МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ»
Кафедра систем информационной безопасности

«ДОПУЩЕН К ЗАЩИТЕ»
Зав. кафедрой к.п.н., доцент Р. Ш. Бердибаев
_____ «_____» _____2019 г.
(подпись)

ДИПЛОМНЫЙ ПРОЕКТ

На тему: Быстродействующее устройство приведения числа по модулю на трехсумматорном формирователе частичных остатках
Специальность: 5В100200 - «Системы информационной безопасности»
Выполнил Джуманов Алибек Аманкелдиулы Группа СИБ-15-3
Научный руководитель Тынымбаев Сахыбай Тнейбаевич

Консультант:
по экономической части:
к.э.н., профессор Аренбаева М.Г.
(ученая степень, звание, Ф.И.О)
(подпись) «22» мая 2019 г.

по безопасности жизнедеятельности:
к.т.н. ст. преп. Бекбасаров Ш.Ш.
(ученая степень, звание, Ф.И.О)
(подпись) «22» мая 2019 г.

по применению вычислительной техники:
к.т.н. доц. Тынымбаев С.Т.
(ученая степень, звание, Ф.И.О)
(подпись) «27» мая 2019 г.

Нормоконтролер:
Ст. преподаватель м.ф.м. Жакупов Н.Б.
(ученая степень, звание, Ф.И.О)
(подпись) «31» 05 2019 г.

Рецензент:
_____
(ученая степень, звание, Ф.И.О)
_____ «_____» _____2019 г.
(подпись)

Алматы 2019

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН
Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ»

Институт систем управления и информационных технологий

Кафедра систем информационной безопасности

Специальность: 5В100200 - «Системы информационной безопасности»

ЗАДАНИЕ
на выполнение дипломного проекта

Студенту Джуманову Алибеку Аманкелдиулы

Тема проекта: Быстродействующее устройство приведения числа по модулю на трехсумматорном формирователе частичных остатках

Утверждена приказом по университету № 124 от « 26 » 10 2019 г.

Срок сдачи законченного проекта «_____» _____2019 г.

Исходные данные к проекту (требуемые параметры результатов исследования (проектирования) и исходные данные объекта): проект подразумевает разработку быстродействующего устройства приведения чисел по модулю на трехсумматорном формирователе частичных остатков.

Перечень вопросов, подлежащих разработке в дипломном проекте, или краткое содержание дипломного проекта: дипломный проект включает в себя 5 глав, разделенных на подглавы, каждая из которых освещает определенную тематику.

В первой главе дипломного проекта представлена общая теоритическая информация о криптосистемах.

Во второй главе дипломного проекта представлены методы приведения чисел по модулю.

В третей главе подробно описывается разработка схемных решений быстродействующего устройства приведения чисел по модулю.

В четвертой главе приводится технико-экономическое обоснование проекта.

В пятой главе рассматриваются необходимые условия для комфортной разработки программного обеспечения.

Перечень графического материала (с точным указанием обязательных чертежей):

1) схемы криптографических систем защиты;
2) таблицы сравнения оценки алгоритмов;
3) рисунки общих схем;
4) рисунки устройств микросхем.

Основная рекомендуемая литература:

1) Цилькер Б.Я., Орлов С.А. Организация ЭВМ и систем. 2-е изд. -Спб.: Питер, 2011.

2) Е.Ж. Айтхожаева, С.Т. Тынымбаев, Аспекты аппаратного приведения по модулю в асимметричной криптографии: Вестник НАНРК №5, 2014. Алматы. с 88-93

3) Рябко Б.Я, Фионов А.И. Основы современной криптографии для специалистов в информационных технологиях. - М.: Научный мир, 2004.

4) Аманжолова К. Б., Алибаева С. А. Экономика предприятии телекоммуникации: Учебное пособие. - Алматы: АИЭС, 2003

Конструкции по проекту с указанием относящихся к ним разделов проекта

Раздел	Консультант	Сроки	Подпись
Вычислительной комплекс	Тотымбаев С.М.	18.02-27.05.18	
Экономический раздел	Алибаева К.Г.	04.03-22.05.18	
Безопасность жизнедеят-ти	Бейсенбаев Ш.Ш.	14.02-27.05.18	

График
подготовки дипломного проекта

Наименование разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю	Примечание
Теоретические основы криптологии	04.03 - 08.03.2019	
Криптография, как способ решения проблем	11.03 - 15.03.2019	
Анализ криптостойкости открытых ключей по примеру	18.03 - 28.03.2019	
Обзор существующих методов проведения исследования	29.03 - 13.04.2019	
Методы формирования ключей при решении задач	15.04 - 18.04.2019	
Принцип открытого проверения ЭЦП в ассиметр	19.04 - 23.04.2019	
Разработка своих решений блокир. уст.	24.04 - 30.04.2019	
Блокирующие устройства, проверка тока	24.04 - 03.05.2019	
Устройства проверки тока на фильтрацию	03.05 - 13.05.19	
Технико-экономическое обоснование	03.05 - 15.05.2019	
Безопасность жизнедеятельности	04.03 - 22.05.2019	

Дата выдачи задания «___» _____ 2019 г.

Заведующий кафедрой _____ (_____)
(Подпись) (Ф.И.О)

Научный руководитель проекта _____ (Джамбаев С.К.)
(Подпись) (Ф.И.О)

Задание принял к исполнению студент _____ (Джуманов А.А.)
(Подпись) (Ф.И.О)

АННОТАЦИЯ

В дипломном проекте было спроектировано схемные решения быстродействующих устройств приведения чисел по модулю, рассмотрены методы приведения чисел по модулю, аппаратная реализация и аппаратное ускорение выполнения шифрования.

Глава по безопасности жизнедеятельности характеризует благоприятные условия труда. В экономической части были приведены расчеты затрат на создание ПО и прибыль предприятия в случае внедрения предлагаемой модели.

АҢДАТПА

Дипломдық жобада модульдік сандарды келтірудің жоғары жылдамдықты құрылғыларының тізбектік шешімдері әзірленді, модульді күштеу әдістері, аппараттық құралдарды енгізу және шифрлауды аппараттық жеделдетумен орындау қарастырылды.

Өмір сүру қауіпсіздігі тарауында еңбек жағдайлары бойынша қолайлы екені сипатталады. Экономикалық тарауда бағдарламалық құралды құру құны есептелген және кәсіпорынның жобаны іске асырған жағдайдағы пайдасы есептелген.

ANNOTATION

In the diploma project was developed a scheme for solving high-speed coercion modulo and numeric device, considered methods of modification, hardware implementation and hardware acceleration of encryption.

The chapter on life safety is characterized by favorable working conditions. In the economic part, calculations were made of the costs of creating software and the profit of the enterprise in the event of the introduction of the proposed model.

Содержание

Введение

На сегодняшний день одним из больших угроз информации является несанкционированное копирование данных или физическая кража носителя информации. Наиболее эффективным методом борьбы с такими угрозами является хранение и передача особо важных данных в зашифрованном виде. Криптографические методы защиты информации могут быть реализованы программно, аппаратно, программно-аппаратно.

Аппаратное шифрование имеет ряд существенных преимуществ перед программным шифрованием, одним из которых является более высокое быстродействие, чем программная реализация. Аппаратная реализация криптоалгоритма гарантирует его целостность, а шифрование и хранение ключей осуществляется в самой плате шифратора, а не в оперативной памяти компьютера. Это очень важно для обеспечения защищенной реализации самого алгоритма, что также является важным преимуществом аппаратной реализации.

Аппаратная реализация применяется как для симметричных, так и асимметричных криптоалгоритмов. Главным достоинством асимметричных криптосистем с открытым ключом по сравнению с симметричными криптосистемами с секретным ключом является их потенциально высокая безопасность: нет необходимости передавать и убеждаться в подлинности секретных ключей. Главным недостатком криптосистем с открытым ключом является низкое быстродействие, так как в процедурах шифрования и дешифрования используются сложные и громоздкие математические вычисления над очень большими числами.

Для шифрпроцессора асимметричных криптосистем особое значение имеет операция приведения по модулю произведения двух чисел. Именно на основе операции приведения по модулю будет произведена разработка быстродействующего устройства. В данной работе рассматриваются устройства приведения числа по модулю, где для формирования i-го частичного остатка используется только три двоичные сумматоры, что существенно упрощает аппаратную реализацию таких устройств. Целью моей дипломной работы является разработка схемных решений быстродействующего аппаратного устройства приведения чисел по модулю на трехсумматорном ФЧО, что позволит существенно ускорить процесс приведения чисел по модулю, тем самым обеспечив более доступное и практичное использование ассиметричного шифрования.

1. Теоретические основы криптосистем

1.1 Криптографическая защита как способ решения проблемы информационной безопасности

Одним из наиболее надежных способов решения проблемы информационной безопасности является криптографическая защита. На современном этапе развития криптографии особое внимание уделяется аппаратной реализации асимметричных криптоалгоритмов, обладающих высокой потенциальной криптостойкостью. Использование аппаратных средств для выполнения базовых операций асимметричных криптоалгоритмов является одним из путей решения проблемы низкого быстродействия асимметричных криптосистем.

В наше время информация приобрела самостоятельную коммерческую ценность и стала широко распространенным, почти обычным товаром. Ее производят, хранят, транспортируют, продают и покупают, а значит - воруют и подделывают - и, следовательно, ее необходимо защищать. Широкое применение компьютерных технологий и постоянное увеличение объема информационных потоков вызывает постоянный рост интереса к криптографии. В последнее время увеличивается роль программных средств защиты информации, не требующих крупных финансовых затрат в сравнении с аппаратными криптосистемами. Современные методы шифрования гарантируют практически абсолютную защиту данных.

Криптография – наука о защите информации от прочтения ее посторонними. Защита обеспечивается шифрованием, то есть преобразованиями, которые затрудняют выявление защищенного ввода путем ввода без знания специальной информации ключа. Ключ понимается как легко изменяемая часть криптосистемы, которая хранится в секрете и определяет, какое шифрование возможных преобразований выполняется в этом случае. Криптосистема – это семейство обратимых преобразований, выбранных с помощью ключа, которые преобразуют защищенный открытый текст в зашифрованный кадр и обратно.

Криптографическая защита. Криптографическими средствами защиты называются специальные средства и методы преобразования информации, в результате чего его содержание в маске. Основными видами криптографического закрытия являются шифрование и кодирование защищаемых данных. В то же время, шифрования тип закрытия, в которой каждый символ закрытые данные самостоятельно преобразованы; при кодировании защищаемые данные делятся на блоки, имеющие смысловое значение, и каждый такой блок заменяется цифровым, буквенным или комбинированным кодом. Он использует несколько различных систем шифрования: замена, перестановка, гаммирование, аналитическое преобразование зашифрованных данных. Шифры комбинации

широко использованы, когда исходный код последовательно преобразован используя 2 или даже 3 различных шифра.

Криптосистема – это набор аппаратных и программных средств и инструкций, алгоритмов шифрования, которые позволяют зашифровать открытый текст и расшифровать его.

Симметричная криптосистема. Метод кодирования, когда для кодирования и декодирования используется один и тот же ключ и один и тот же алгоритм кодирования, называется симметричным. В симметричном методе кодирования ключ является секретным, закрытым. Вы можете доставить секретный ключ подписчикам:

– физически на электронных носителях информации (дисках, флеш-картах и др.), на пластиковых картах, в виде паролей, о которых администратор сообщает лично;

– канал связи зашифрован, этом случае необходимо, чтобы у абонентов уже были средства для передачи секретной информации.

На практике обычно используется комбинированная модель работы с секретными ключами:

– абонентам физически доставляются долговременные ключи;

– с помощью долговременных ключей шифруются и передаются сеансовые ключи, используемые только в одном сеансе связи;

– на основе сеансовых ключей шифруется секретная информация.

Эта методика использует тот же ключ для шифрования и дешифрования отправителем и получателем, который они договорились использовать до взаимодействия. Если ключ не был скомпрометирован, дешифрование автоматически аутентифицирует отправителя, поскольку только у отправителя есть ключ для шифрования информации, и только у получателя есть ключ для дешифрования информации. Так как отправитель и получатель единственные люди, которые знают этот симметричный ключ, при компрометации ключа будет скомпрометировано только взаимодействие этих двух пользователей. Проблема, которая будет актуальна для других криптосистем, - это безопасное распределение симметричных (секретных) ключей.

Симметричные алгоритмы шифрования используют ключи не очень долго и могут быстро шифровать большие объемы данных.

Как использовать системы с симметричными ключами:

– симметричный секретный ключ надежно создается, распространяется и хранится;

– отправитель создает электронную подпись с помощью расчета хэш-функции для текста и присоединения полученной строки к тексту;

– отправитель использует быстрый симметричный метод шифрования-дешифрования алгоритма вместе с секретным симметричным ключом к полученному пакету (тексту вместе с присоединенной электронной подписью)

для получения зашифрованного текста. это неявная аутентификация, так как только отправитель знает симметричный секретный ключ и может зашифровать этот пакет. только получатель знает симметричный секретный ключ и может расшифровать этот пакет;

– отправитель отправляет зашифрованный текст. симметричный секретный ключ никогда не передается по незащищенным каналам связи;

– получатель использует тот же симметричный алгоритм шифрования-дешифрования вместе с тем же симметричным ключом (который у получателя уже есть) к зашифрованному тексту для восстановления исходного текста и электронной подписи. его успешное восстановление опознает кого-то, кто знает секретный ключ;

– получатель отделяет электронную подпись от текста;

– получатель создает другую электронную подпись с помощью расчета хэш-функции для полученного текста;

Асимметричная криптосистема. Система, в которой для кодирования используйте тот же ключ K_x, а для расшифровки другой ключ, K_2, называется асимметричным. В асимметричном методе кодирования один ключ может быть открытым, второй – закрытым, секретным. С помощью одного ключа общедоступная информация шифруется отправителем, второй ключ используется получателем для расшифровки зашифрованного текста.

Эти системы характеризуются тем, что для шифрования и дешифрования используются разные ключи, связанные между собой некоторой зависимостью. Использование таких шифров стало возможным благодаря К. Шеннону, который предложил построить шифр таким образом, чтобы его раскрытие было эквивалентно решению математической задачи, требующей объема вычислений, превышающих возможности современных компьютеров (например, операции с большими простыми числами и их продуктами). Один из ключей (например, ключ шифрования) может быть сделан публичным, и в этом случае проблема получения общего секретного ключа для связи отпадает. Если сделать ключ дешифрования открытым, можно построить систему проверки подлинности для передаваемых сообщений на основе полученной системы. Поскольку в большинстве случаев один ключ из пары является открытым, такие системы также называются криптосистемами с открытым ключом. Первый ключ не является секретным и может быть опубликован для использования всеми пользователями системы, которые зашифровывают данные. Невозможно расшифровать данные с помощью известного ключа. Для дешифрования данных получатель зашифрованной информации использует второй ключ, который является секретным. Конечно, ключ расшифрования не может быть определен из ключа шифрования.

В асимметричных криптосистемах важно, чтобы сеансовые и асимметричные ключи были сопоставимы с точки зрения уровня безопасности,

который они обеспечивают. При использовании короткого ключа сеанса не имеет значения, насколько велики асимметричные ключи.

Как использовать системы с асимметричными ключами:

– асимметричные открытый и закрытый ключи создаются и распространяются безопасно, а секретный асимметричный ключ передается его владельцу, открытый асимметричный ключ хранится в базе данных X. 500 и управляет центром сертификации. Подразумевается, что пользователи должны верить, что такая система обеспечивает безопасное создание, распространение и администрирование ключей. более того, если создатель ключей и лицо или система, управляющие ими, не совпадают, то конечный пользователь должен верить, что создатель ключей фактически уничтожил их копию;

– создается электронная подпись текста с помощью вычисления его хэш-функции, полученное значение шифруется с помощью асимметричного секретного ключа отправителя, а затем в передаваемый текст добавляется результирующая символьная строка (только отправитель может создать электронную подпись);

– создается секретный симметричный ключ, который будет использоваться для шифрования только этого сообщения или сеанса взаимодействия (сеансовый ключ), затем при помощи симметричного шифрования/дешифрования алгоритма и этого ключа шифруется исходный текст вместе с электронной подписью - производит зашифрованный текст (шифр-текст);

– теперь вам нужно решить проблему с передачей ключа сеанса получателю сообщения;

– отправитель должен иметь открытый ключ асимметричного центра сертификации (ЦС). перехват незашифрованных запросов на этот открытый ключ является распространенной формой атаки. может существовать целая система сертификатов, удостоверяющих подлинность открытого ключа ЦС. стандартный x.509 описывает ряд способов получения пользователями открытых ключей ЦС, но ни один из них не может полностью защитить от подмены открытого ключа ЦС, что наглядно доказывает отсутствие такой системы, в которой можно было бы гарантировать подлинность открытого ключа ЦС;

– отправитель запрашивает асимметричный открытый ключ получателя сообщения от ЦС. этот процесс уязвим для атаки, в которой злоумышленник вмешивается в связь между отправителем и получателем и может изменять трафик, отправляемый между ними. таким образом открытый асимметричный ключ получателя "подписывается" ЦС. это означает, что ЦС использовал свой асимметричный закрытый ключ для шифрования асимметричного отскочившего ключа получателя. только ЦС знает асимметричный закрытый ключ ЦС, поэтому есть гарантии, что открытый асимметричный ключ получателя будет получен из ЦС;

– после получения асимметричный открытый ключ получателя расшифровывается с использованием асимметричного открытого ключа СА и асимметричного алгоритма шифрования/дешифрования. естественно, предполагается, что ЦС не был скомпрометирован. если он окажется скомпрометированным, он отключит всю сеть своих пользователей;

– теперь шифруется сеансовый ключ с использованием асимметричного алгоритма шифрования и расшифровки и асимметричного ключа получателя (полученного от СА и расшифровать);

– зашифрованный сеансовый ключ присоединяется к зашифрованному тексту (который также включает ранее добавлена электронная подпись);

– весь полученный пакет данных (зашифрованный текст, включающий в себя помимо исходного текста его электронную подпись, и зашифрованный сессионный ключ) передается получателю. поскольку зашифрованный ключ сеанса передается по незащищенной сети, он является очевидной целью различных атак;

– получатель выделяет зашифрованный ключ сеанса из полученного пакета;

– теперь получателю необходимо решить проблему с расшифровкой ключа сеанса;

– получатель должен иметь открытый ключ асимметричного центра сертификации (ЦС);

– используя свой секретный асимметричный ключ и тот же асимметричный алгоритм шифрования получатель расшифровывает сеансовый ключ;

– получатель использует тот же симметричный алгоритм шифрования-дешифрования и расшифрованный симметричный (сессионный) ключ к зашифрованному тексту и получает исходный текст вместе с электронной подписью;

– получатель отделяет электронную подпись от исходного текста;

– получатель запрашивает асимметричный открытый ключ отправителя;

– как только этот ключ получен, получатель расшифровывает его с помощью открытого ключа СА и соответствующего асимметричного шифрования-дешифрования алгоритма;

– затем расшифровывает текст хэш-функции с помощью открытого ключа отправителя и асимметричного шифрования-алгоритм расшифровки;

– повторно вычислить хэш-функцию исходного текста;

– эти две хэш-функции сравниваются, чтобы убедиться, что текст не был изменен.

В асимметричной криптосистеме каждый взаимодействующий абонент генерирует пару ключей-открытый, доступный любому пользователю информационной системы, и закрытый, секретный, известный только абоненту.

Рассмотрим взаимодействие двух абонентов – абонента А и абонента Б. абонент в, желающий отправить сообщение абоненту, а, шифрует его открытым ключом абонента л и отправляет по каналу связи. Абонент L, получив это сообщение, расшифровывает его своим секретным ключом.

При использовании алгоритмов открытого ключа существует риск того, что злоумышленник может заменить законный открытый ключ. После подделки открытого ключа злоумышленник может считывать перехваченные зашифрованные сообщения. Защита от подделки сертификация открытого ключа используется для доказательства подлинности открытого ключа.

Недостатки асимметричных шифров являются их низкая производительность и высокие требования к вычислительным ресурсам. Сочетание симметричных и асимметричных методов шифрования повышает эффективность и производительность шифрования: данные шифруются симметричным алгоритмом, а ключи шифрования закрываются асимметричным алгоритмом и передаются партнеру по каналу связи в самом начале сеанса.

Методы хранения информации на современном этапе развития компьютерных технологий динамично улучшается. Это связано с упрощением его хранения в вычислительных системах и несравненно высокой скоростью доступа к нему. В настоящее время информация-это специфический продукт, который можно купить, продать, обменять на что-то другое, и т. д. Информация является стратегическим ресурсом государства. Поэтому защита информации от несанкционированного доступа, кражи, уничтожения и других преступных видов деятельности является актуальной проблемой. Один из наиболее надежных способов обеспечения защиты информации, хранящейся в электронном виде с криптографической защитой. Криптография связана с шифрованием и расшифровкой конфиденциальных данных в каналах связи. Он также используется для исключения возможности искажения информации или подтверждения ее происхождения. Криптографическое преобразование информации обеспечивает ее секретность для лиц, не имеющих ключа, и поддержание необходимой надежности обнаружения несанкционированного искажения. Криптографические средства представляют собой отдельную группу формальных средств защиты, обеспечивающих преобразование открытого текста в зашифрованный путем шифрования исходного кода с использованием криптографических алгоритмов. Они могут быть реализованы в виде программной, аппаратной и программно-аппаратной защиты. Аппаратное шифрование представляет собой специализированное оборудование. Они дороже программных кодеров и сложнее в реализации, но имеют ряд существенных преимуществ перед программным обеспечением: высокую производительность, простоту, безопасность и т.д. Аппаратное шифрование информации имеет три разновидности:

– модули шифрования (они выполняют всю работу с ключами);

– блоки шифрования в каналах связи;

– карты расширения шифрования для установки в персональные компьютеры.

Большинство устройств первого и второго типа являются узкоспециализированными. Карты расширения для персональных компьютеров являются более универсальным аппаратным средством шифрования и обычно могут быть легко настроены для шифрования всей информации, записанной на жесткий диск компьютера, а также всех данных, отправленных на его дискету и последовательные порты. Большинство устройств аппаратного шифрования реализованы в виде карт расширения PCI или устройств типа USB ключ. На современном этапе развития криптографии особое внимание уделяется асимметричным криптографическим алгоритмам. Использование асимметричной пары ключей шифрования (по сравнению с симметричным шифрованием, использующим только один ключ) увеличивает сложность криптоанализа для злоумышленника. Основным недостатком асимметричных криптографических алгоритмов является низкая скорость, так как в процедурах шифрования и дешифрования используются громоздкие арифметические вычисления над числами с высокой битовой шириной (по модулю экспоненты). Поэтому главная проблема асимметричных криптоалгоритмов – это проблема ускорения возведения чисел в степень по модулю. Одним из путей решения этой проблемы является использование аппаратных средств для выполнения базовых операций быстрого возведения чисел в степень по модулю – целочисленного умножения, возведения в квадрат, приведения по модулю.

1.2 Анализ криптосистемы с открытым ключом на примере криптосистемы Диффи-Хелмана

Одна из фундаментальных проблем криптографии – безопасное общение по прослушиваемому каналу. Сообщения должны быть зашифрованы и расшифрованы, но обе стороны должны иметь общий ключ. Если этот ключ передается по тому же каналу, прослушивании тоже получит его, и смысл шифрования исчезнет.

Алгоритм Диффи-Хеллмана позволяет двум сторонам получить общий секретный ключ, используя незащищенный от прослушивания, но защищенный от подмены канал связи. Полученный ключ можно использовать для обмена сообщениями с использованием симметричного шифрования.

Первая публикация данного алгоритма появилась в 70-х годах XX века в статье Диффи и Хеллмана, в которой вводятся основные понятия криптографии с открытым ключом. Алгоритма Диффи-Хеллмана не применяется для шифрования сообщений или формирования электронной подписи. Его основное назначение - распределение ключей. Он позволяет двум или более пользователям обмениваться ключами без посредников, которые затем могут использоваться

для симметричного шифрования. Это была первая криптосистема, позволяющая защищать информацию без использования закрытых ключей, передаваемых по защищенным каналам. Схема распределения открытых ключей, предложенная Диффи и Хеллманом, произвела революцию в мире шифрования, устранив основную проблему классической криптографии – проблему распределения ключей.

Описание алгоритма

Предположим, что оба абонента знают какие-то два числа g и p, которые не являются секретными и также могут быть известны другим заинтересованным лицам. Для того, чтобы создать секретный ключ неизвестен никому, оба абонента генерируют большие случайные числа: первый абонент-число a, второй абонент-число b. Затем первый абонент вычисляет значение $A = g^a \bmod p$ и пересылает его второму, а второй вычисляет параметр $B=g^b \bmod p$ и передает первому. Предполагается, что злоумышленник может получить оба этих значения, но не модифицировать их (то есть, он не имеет возможности вмешаться в процесс передачи). На втором этапе первый абонент на основе результата B вычисляет значение $B^a \bmod p = g^{ab} \bmod p$, а второй абонент на основе b и результата a вычисляет значение $A^b \bmod p = g^{ab} \bmod p$. Как видите, у обоих пользователей получилось одинаковое число: $K = g^{ab} \bmod p$. Они могут использовать его как секретный ключ, потому что здесь злоумышленник столкнется с почти неразрешимой (в разумные сроки) проблемой вычисления $g^{ab} \bmod p$ перехваченным $g^{ab} \bmod p$ и $g^{ab} \bmod p$, если числа p,a,b выбраны достаточно большими.

Когда алгоритм работает, каждая сторона:

– генерирует случайное положительное целое число a-закрытый ключ;

– вместе с удаленной стороной устанавливает открытые параметры p и g (обычно значения p и g генерируются с одной стороны и передаются на другую), где p-случайное простое число, g-первичный корень по модулю p;

– вычисляет открытый ключ A с помощью преобразования по закрытому ключу $a = g^{ab} \bmod p$;

– обмен открытыми ключами с удаленной стороной;

– вычисляет общий секретный ключ K, используя открытый ключ удаленной стороны B и свой закрытый ключ:

$K = B^a \bmod p$

K равна с обеих сторон, потому что

$B^a \bmod p = (g^b \bmod p)^a \bmod p = g^{ab} \bmod p = (g^a \bmod p)^b \bmod p = A^b \bmod p$.

В практических реализациях a и b являются номерами порядка 10^{100}, а p порядка 10^{300}. Число g не должно быть большим и обычно имеет значение в пределах первых десяти.

Криптографическая устойчивость алгоритма Диффи-Хеллмана (то есть сложность вычисления $K=g^{ab} \bmod p$ известными p, g, $A=g^a \bmod p$ и $B=g^b \bmod p$)

основана на предполагаемой сложности задачи дискретного логарифма. Однако, хотя возможность решения задачи дискретного логарифма позволит взломать алгоритм Диффи-Хеллмана, обратное утверждение по-прежнему остается открытым вопросом (другими словами, эквивалентность этих проблем не доказана).

Следует отметить, что алгоритм Диффи-Хеллмана работает только на линиях связи, надежно защищенных от модификации. Если бы она была применима к любым открытым каналам, она давно устранила бы проблему распределения ключей и, возможно, заменила бы всю асимметричную криптографию. Однако в тех случаях, когда в канале возможна модификация данных, существует очевидная возможность вклинивания в процесс генерации ключей "промежуточного злоумышленника" по той же схеме, что и для асимметричной криптографии.

2. Обзор существующих методов приведения чисел по модулю

2.1 Методы формирования остатков при делении на модуль

Существует большое количество разнообразных методов формирования остатков при делении на модуль.

При использовании двоичного (обычного) представления целых положительных чисел можно выделить три способа формирования остатков, но произвольному модулю Р.

Первый способ основан на последовательном формировании остатков (r_i).

Во втором способе для формирования остатка из приводимого числа А вычитывается кратные модули PxI (1=1,3,5... К).

В третьем способе приведения числа по модулю использован принцип. машинного алгоритма двоичного деления чисел А на модуль Р со сдвигом остатков влево.

1) Формирование остатков по модулю Р путем вычисления частичных остатков с дальнейшим суммированием их по модулю

Способ формирования остатка основан на последовательном формировании остатков (r_i) разрядных весов двоичного числа (2^i) от деления по модулю Р с дальнейшим суммированием по модулю Р тех остатков, для которых коэффициенты a_i соответствующих весов равны единице.

Тогда формула для вычисления остатка где от числа А по модулю Р имеет следующий вид:

$$r_A = A \bmod P \left[\sum_{i=0}^{K-1} \left(2^i mod P\right) a_i\right] \bmod P \qquad (2.1)$$

где 2^i- вес i-го разряда числа А (i=0÷К-1), a_i — коэффициент i-го разряда числа А.

Так как для двоичной системы счисления коэффициенты a_i (1 =0...К-1), принимают только два значения (0 или 1), суммируя заранее вычисленные остатки по модулю Р от числа 2^i (1=0,..К-1), для тех ь, для которых коэффициенты а=1, получают остаток по модулю Р от числа А. Частичный остаток от 2. для любого модуля (Р>2) всегда равен единице. Частичный остаток от 2^i в два раза превышает частичный остаток от 2^0 и т.д., т.е. частичный остаток 2^i в два раза превышает частичный остаток от 2^{i-1}. Таким образом, вычисление частичного остатка от 2^i заключается в умножении на два частичного остатка от 2^{i-1} и приведении результата по модулю _А. Операция умножения на два может быть реализована сдвигом всех разрядов умножаемого числа на один разряд влево.

Операция приведения по модулю Р для чисел, не прививающих величину

2P-1, реализуются следующим образом. Если число не превышает величину P, то оно остается без изменения, если же число лежит в интервале от P до 2P-1, то из него вычитается модуль P, а результат является остатком.

На рисунке 2.1 представлена функциональная схема устройства для формирования остатков, на рисунке 2.2 – функциональная схема формирователя частичных остатков, на рисунке 2.3 – функциональная схема сумматора по модулю (СММ).

Устройство формирования остатков по модулю (рисунок 2.1) состоит из n-1 последовательно соединенных ФЧО, схем $И_0 \div И_{k-1}$ и К-1 сумматоров по модулю. Разряды числа А $a_0 \div a_{n-1}$ из регистра А подаются на соответствующие схемы $И_0 \div И_{k-1}$, где логически умножаются со значениями частичного остатка $r_0 \div r_{k-1}$.

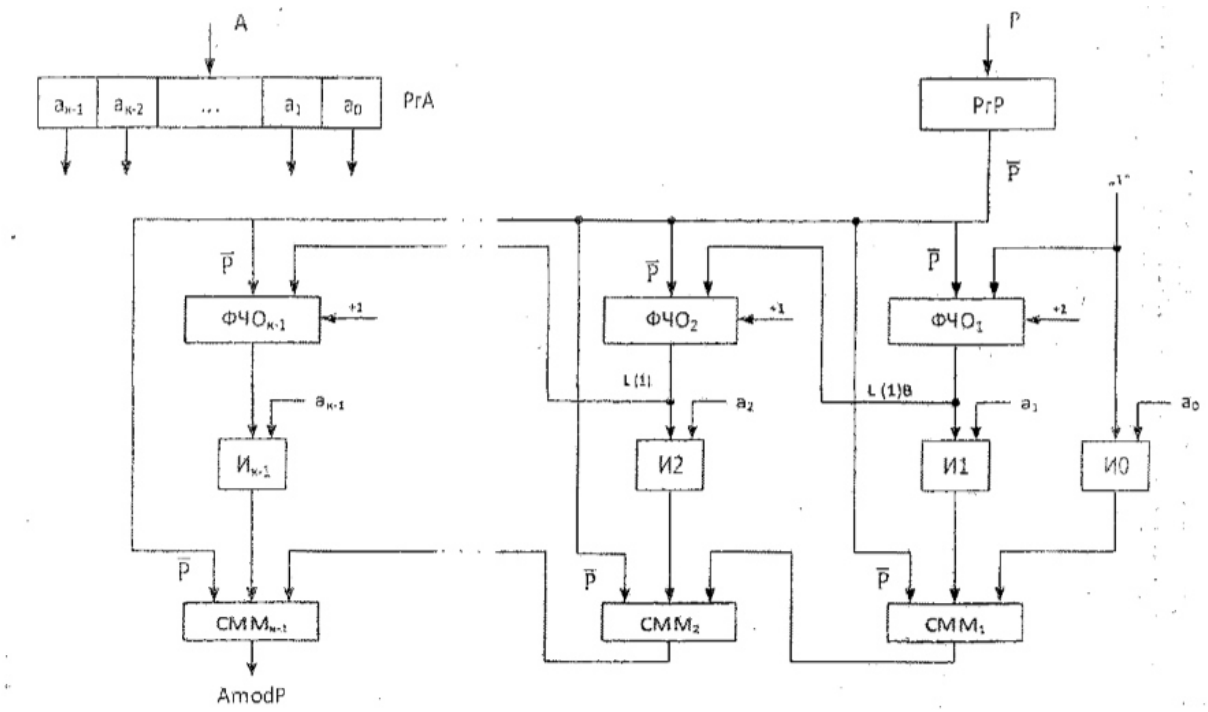

Рисунок 2.1 – Функциональная схема устройства для формирователя остатка по модулю.

Инверсное значение модуля (P) подается на входы $ФЧО_1 \div ФЧО_{k-1}$_ и $СММ_1 = СММ_{k-1}$. На первом шаге вычисления остатка значение $2^0=1$ (частичный остаток r_0) подается на первый вход схемы $И_0$, а на второй вход – a_0. При $a_0=1$ на выходе $И_0$ формируется промежуточный остаток R_0, который на втором шаг вычисления остатка подается на вход $СММ_1$. Одновременно на втором шаге вычисления частичный остаток r_1 с выхода $ФЧО_2$, со сдвигом на один разряд влево подается на вход $ФЧО_1$, также r_1 без сдвига подается на первый вход схемы $И_2$ на второй вход которого подается значение разрядного коэффициента

a_2. При, a_2=1 с выхода $И_2$ r_1 подается на вход сумматора по модулю СММ) и на его выходе формируются промежуточный остаток R_1.

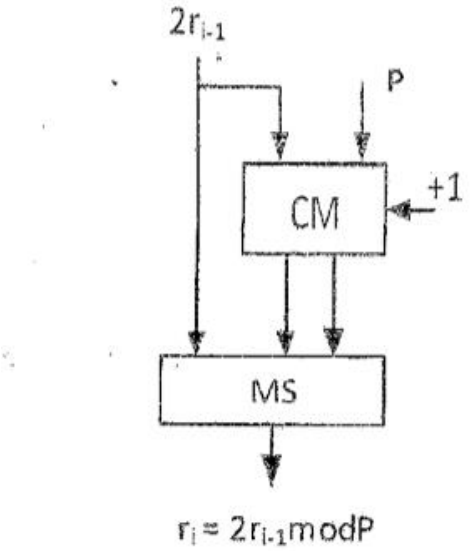

Рисунок 2.2 – Функциональная схема формирователя частичного остатка (ФЧО)

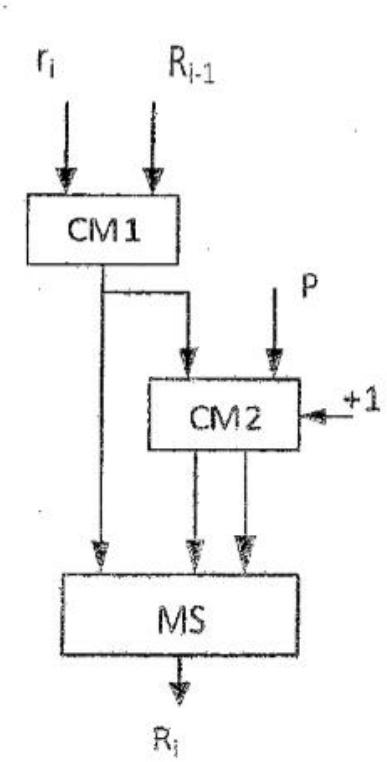

Рисунок 2.3 – Функциональная схема сумматора по модулю (СММ)

После К шагов на выходе $ФЧО_{k-1}$, формируется частичный остаток r_{k-1} при $a_{n-1}=1$ этот остаток поступает на вход $СММ_{k-1}$, формируя на выходе значение R=AmodP.

Время формирования остатка $T_{ф.0i}$ r_i определяется по формуле

$$T_{ф.0i}= 3T_{СМ} \quad (2.2)$$

Количество двоичных сумматоров при этом:

$$Q=3(K-1)N_{СМ} \quad (2.3)$$

где Тсм – время суммирования, К – разряды сумматоров и модуля Р.

Теперь рассмотрим схему, позволяющей ускорить процесс формирования остатка. Для этого формулу (2.1) представим в следующем виде:

$$A=2^{2k}(2a_{k+1}+a_{2k})+ \ldots +2^4(2a_2+a_2)+2^2(2a_1+a_1)+(2a_0+a_0) \quad (2.4)$$

Для вычисления остатка от числа А по модулю Р достаточно в формуле просуммировать частичные остатки по модулю Р от чисел $2^{2i}(2a_{2i} +2_{2i})$.

2.2 Принципы аппаратного приведения по модулю в асимметричной криптографии

По мере развития и усложнения средств, методов и форм автоматизации процессов сбора, хранения и обработки информации повышается ее уязвимость. Защита данных – это совокупность целенаправленных действий и мероприятий по обеспечению безопасности данных. Одним из наиболее надежных способов решения проблемы безопасности данных в компьютерных системах и сетях считается криптографическая защита, обеспечивающая превращение открытого текста в шифртекст путем шифрования исходного текста с помощью криптографических алгоритмов. Шифрование возможно осуществить программно, аппаратно и программно-аппаратно. Аппаратное шифрование имеет ряд существенных преимуществ перед программным шифрованием:

– аппаратные средства шифрования обладают большей скоростью (аппаратная реализация любого алгоритма, в том числе и криптографического, обеспечивает более высокое быстродействие, чем программная реализация);

– аппаратуру шифрования легче физически защитить от проникновения извне, чем программу;

– аппаратуру шифрования проще установить. Поэтому большинство средств криптографической защиты данных реализовано в виде специализированных аппаратных устройств.

Эти устройства встраиваются в линию связи и шифрования информации, передаваемой через нее. Преобладание аппаратного шифрования над

программным шифрованием обусловлено не только вышеуказанными причинами, список преимуществ аппаратных шифраторов значительно шире:

– аппаратная реализация криптоалгоритмов обеспечивает целостность;

– шифрование и хранение ключей осуществляются в самой плате шифратора, а не в памяти компьютера;

– аппаратный датчик случайных чисел создает действительно случайные числа для создания надежных ключей шифрования и электронной цифровой подписи;

– на базе аппаратных шифраторов можно создавать системы защиты информации от несанкционированного доступа и разграничения доступа к компьютеру;

– использование специального шифра процессор для выполнения криптографических преобразований разгружает центральный процессор компьютера; возможна также установка нескольких аппаратных шифраторов на одном компьютере, что еще более повышает скорость обработки информации;

– использование парафазных шин в архитектуре шифр процессор исключает угрозу снятия ключевой информации по колебаниям электромагнитного излучения в цепях "земля - питание" микросхемы, возникающим в ходе криптографических преобразований.

Большинство современных криптосистем используют асимметричное шифрование. Особенностью асимметричных (двухключевых) алгоритмов шифрования является то, что для шифрования и дешифрования информации используются разные ключи. Знание открытого ключа, с помощью которого был зашифрован документ, не позволяет расшифровать этот документ, а знание закрытого (секретного) ключа, позволяющего расшифровать сообщение, не позволяет его зашифровать. Широко известен двойной-основных алгоритмов, таких как RSA, Эль-Гамаля, Диффи-Хеллмана, Фиата-Шамира, Рабина, Окамото-Саранси, Мацумото-Имаси, Шнорра. Основным преимуществом криптосистем с открытым ключом перед симметричными (однократными) криптосистемами с закрытым ключом является их потенциально высокая безопасность: нет необходимости передавать и проверять подлинность закрытых ключей. Основным недостатком криптосистем с открытым ключом является низкая скорость, так как процедуры шифрования и дешифрования используют гораздо более сложные и громоздкие математические вычисления на очень больших числах (например, RSA, Эль-Гамаль и Рабин используют числа с порядками 10309). Поэтому криптосистемы с открытым ключом часто используются для шифрования, передачи и расшифровки только секретного ключа симметричной криптосистемы. Симметричная криптосистема применяется для шифрования и передачи сообщений. Это так называемая схема электронного цифрового конверта. Широкое использование двухключевых средств защиты также связано с электронной цифровой подписью, которая является необходимым условием

электронного документа, предназначенного для защиты этого электронного документа от подделки. В 1997 году был разработан стандарт ANSI X9.30, поддерживая стандарт цифровой подписи (стандарт цифровой подписи), и год более поздно ANSI X9 было введено.31, в котором основное внимание уделяется цифровым подписям RSA, что соответствует фактической ситуации, в частности, для финансовых учреждений.

Разработанные на сегодняшний день криптосистемы открытого ключа основаны на одном из следующих видов необратимых (и комплексных) преобразований: разложение больших чисел на простые множители, вычисление логарифмов в конечном поле, вычисление корней алгебраических уравнений.

На практике наиболее широко распространен алгоритм асимметричного шифрования RSA (Ривеста, Шамира и Адлемана, 1978), который основан на необратимом преобразовании, разложении больших чисел на простые множители. Крипто алгоритм характеризуется хорошей криптографической устойчивостью, которая основана на сложности факторизации больших целых чисел. Алгоритм RSA стал первым полноценным алгоритмом открытого ключа, который может работать как в режиме шифрования данных, так и в режиме электронной цифровой подписи. Он стал де-факто мировым стандартом для открытых систем и рекомендован МКТТ. В настоящее время алгоритм RSA используется во многих стандартах. ISO 9796 описывает RSA как совместимый криптографический алгоритм, который соответствует стандарту безопасности ITU-T X. 509. Кроме того, криптосистема RSA является частью стандартов SWIFT, ANSI X9.31 rDSA и проект стандарта X9.44 для американских банков. Австралийский стандарт управления ключами AS2805.6.5.3 также включает систему RSA. Алгоритм RSA активно реализуется как в виде самостоятельных криптографических продуктов, так и в качестве встроенных средств в приложениях. Например, для защиты баз данных серверы используют встроенные механизмы шифрования, требующие использования RSA .

Алгоритм RSA используется в Интернете, в частности он включен в такие протоколы, как SSL, SHHTP, S-MIME, S/WAN, STT, PCT, IPSEC (Internet Protocol Security) и TLS (которые должны заменить SSL), а также в стандарт PKCS, используемый в важных приложениях. Для разработчиков приложений, использующих PKCS, OSI Implementers' Workshop (OIW) выпустила соглашение, которое конкретно касается алгоритма RSA.

Многие другие стандарты, разрабатываемые в настоящее время, включают либо сам алгоритм RSA, либо его поддержку, либо рекомендуют криптосистему RSA для конфиденциальности и/или аутентификации. Например, включите рекомендации Системы RSA IEEE P1363 и WAP WTLS.

Разработаны специальные процессоры для аппаратной реализации операций шифрования и дешифрования RSA. Эти процессоры, реализованные на сверхбольших интегральных схемах (СБИС), позволяют выполнять операции

RSA, связанные с поднятием больших чисел до очень большой степени по модулю P, за относительно короткое время. Одна из самых быстрых аппаратных реализаций RSA с 512-битным модулем на сверхбольшой интегральной схеме имеет скорость 64 Кбит/с. лучшими из коммерчески доступных VLSI являются процессоры компании CYLINK, выполняющие 1024-битное шифрование RSA. Для сравнения, криптографический программный пакет BSAFE 3.0, реализующий RSA на компьютере Pentium-90, выполняет шифрование с частотой 21,6 Кбит / с для 512-битного ключа и 7,4 Кбит / с для 1024-битного ключа.

Однако аппаратная реализация RSA выполняет операции шифрования и дешифрования примерно в 1000 раз медленнее, чем аппаратная реализация DES – симметричной криптографии. Такой значительный разрыв в производительности возникает из-за того, что RSA использует построение очень больших (многозначных) чисел в очень большой степени по модулю P. лаборатория RSA рекомендует 1024-битные ключи для обычных задач, а для особо важных задач – 2048 бит и более. Например, для обеспечения безопасности в стандарте Республики Казахстан СТ РК 1073-2007 для достижения 3 – го уровня безопасности предусмотрено использование ключа длиной 4000 бит, а для достижения 4-го уровня безопасности-8000 бит. Поэтому все большее внимание теоретиков и практиков криптографии уделяется проблеме ускорения возведения чисел в степень по модулю P.

Определены основные операции над числами, используемые в асимметричных криптографических алгоритмах. Возведение в степень чисел по модулю P (ax mod p) осуществляется с помощью таких операций, как умножение, возведение в квадрат и по модулю. И одним из подходов к повышению производительности криптосистем с открытым ключом является ускорение этих операций.

Наиболее сложной из них является операция приведения по модулю, так как она является остаточной от деления числа на модуль P, а операция деления – наиболее сложная из арифметических операций. И эта операция повторяется много раз, потому что вместо умножения, а затем деления очень большого числа (ax) на модуль, для ускорения возведения в степень по модулю, используется многоступенчатое последовательное умножение с уменьшением по модулю на каждом шаге каждый раз новое произведение. Это также уменьшает разрядность умножаемых чисел и, соответственно, разрядность умножаемого произведения.

Например, если нужно вычислить a16 mod p, то вместо выполнения пятнадцати перемножений и одного приведения по модулю очень большого числа (a*a*a*a*a*a*a*a*a*a*a*a*a*a*a*a) выполняют четыре возведения в квадрат, используя после каждого возведения в квадрат приведение по модулю: a16 mod p = (((a2)2)2)2 mod p = (((a2 mod p)2 mod p)2 mod p)2 mod p. Это

позволяет уменьшить разрядность операндов и ускорить возведение чисел в степень по модулю P. И чем длиннее число, тем заметнее ускорение.

Вычисление ax mod p, где x не является степенью 2, ненамного сложнее. Например, необходимо вычислить a17 mod p. Двоичная запись степени (x) числа позволяет представить x как сумму степеней 2: x = 17(10) = 1 0 0 0 1(2), поэтому 17 = 24+ 20. Тогда

a17 mod p = (a*a16) mod p = (a*(((a2)2)2)2)mod p = (((((a2 mod p)2 mod p)2mod p)2 mod p)*a) mod p.

Такой подход уменьшает трудоемкость вычислений до 1,5xk операций в среднем, где x – степень числа, k – длина числа в битах.

Из этих примеров видно, что используется умножение на a (*a) и возведение a в квадрат (a2), приведение полученных произведений (в том числе a2) по модулю.

На сегодняшний день накоплен большой опыт разработки быстродействующих целочисленных умножителей и квадраторов для различных классов вычислительных систем. Для ускорения основных операций умножения и возведения в квадрат можно использовать массивы двоичных сумматоров, дерево Уоллеса, счетчики Дадда, умножители Харрисона, систолические умножители, ведические умножители, быстрые двоично-десятичные сумматоры (при использовании двоично-десятичной нотации) и др.

Что касается ускорения основной работы литого по модулю, то такой задачи в традиционных вычислительных системах не было. Поэтому высокоскоростное аппаратное решение операции по модулю является ключевой проблемой аппаратной реализации криптографических алгоритмов, использующих возведение в степень чисел по модулю P, включая RSA.

В аппаратной реализации литого модуля могут быть использованы различные подходы, которые приводят к широкому разнообразию структур устройств для получения остатка деления на модули. Эти структуры представлены в различных публикациях, но систематизация и анализ их отсутствует.

Анализ структур и принципов функционирования различных устройств редукции на модуле позволил выявить их характерные признаки:

– последовательное или параллельное выполнение операций возведения в квадрат и получение остатков от деления по модулю;

– одношаговый или многошаговый работы устройства;

– наличие или отсутствие схемы управления (управляющей машины) работой литого модуля;

– использование определенной системы счисления.

С учетом этих характеристик все литейные устройства можно разделить на классы по модулю.

Ниже приводится классификация литейных устройств по модулю на основе вышеуказанных критериев.

1) Классификация по степени параллельности умножения и уменьшения произведения по модулю:

а) параллельно-уменьшение модуля осуществляется в процессе умножения, параллельно. После получения каждого частичного произведения, каждый раз его уменьшают по модулю и далее для продолжения умножения используют не частичное произведение, а его остаток;

б) последовательное-уменьшение модуля осуществляется после получения продукта, последовательно. Выполняет умножение или возведение в квадрат, затем найти остаток от деления по модулю.

2) Классификация по количеству тактов, необходимых для получения баланса в устройстве литье модуло:

а) Многоточные устройства, в которых баланс определяется путем многократного вычитания из исходного числа заданного, а затем полученного модуля положительного остатка, который используется для литья. И здесь возможны два варианта:

– все вычитания осуществляются на одних и тех же узлах, которые многократно циклически участвуют в процессе получения каждого остатка (циклическая организация);

– вычитания реализуются на аппаратном конвейере (конвейерной организации), каждая схема которого используется только один раз. Каждый остаток образуется на своем уровне конвейера, количество которого определяется максимальным количеством положительных остатков;

б) однотактное устройство, выполняющее параллельное вычитание заданных чисел по модулю P и кратное модулю (2P, 3P, 4P,....). Кратные модуля предварительно формируются на дополнительных узлах устройства. В этом случае получается набор остатков, в результате получается наименьший положительный остаток.

3) классификация по наличию устройства автомата управления (UA) по модулю уменьшения:

а) комплексное устройство-представлено в виде набора рабочих и управляющих машин (ОА и УА). UA генерирует управляющие сигналы и управляет процессом приведения модуля, а все операции выполняются в ОА. Работая машина, в свою очередь, посылает информационные сигналы к машине управления, которая служит как направляющий выступ для машины управления в развитии сигнала управления. Это типичный случай классического операционного устройства (ОУ), в синтезе которого применяются известные методы синтеза цифровых машин, в том числе микропрограммных машин (МПА). Здесь доступны следующие параметры:

– машина управления может быть построена в виде схемы-UA с жесткой логикой;

– машина управления может быть построена на основе принципа программного управления-UA с программируемой логикой;

б) автономное устройство - управляющая часть не выделена, все реализовано в виде единой схемы, управляющие сигналы формируются в результате операций.

4) классификация в соответствии с системой счисления, используемой в устройстве литья по модулю:

а) двоичная система счисления;

б) двоично-десятичная система счисления;

в) вспомогательная система счисления с базой 2h, где h-целое число и h>=2. Переход к вспомогательной системе счисления осуществляется условно из двоичной системы счисления путем деления двоичного числа на диады (h=2, 2h =4), на триады (h=3, 2h =8), на тетрады (h=4, 2h =16) и др.

Предложенные критерии и классификации позволяют проводить сравнительную оценку любой аппаратной реализации литого модуля на уровне структуры, так как каждый класс во всех четырех классификациях имеет свои преимущества и недостатки.

Что касается классификации управления схемой, то из описаний и схем устройства патента не ясно, какой метод управления операцией подразумевали авторы. Поэтому устройство может быть реализовано либо как единая схема, либо должно быть дополнено схемой управления. И это будут два разных устройства.

Другое устройство по модулю, которое имеет регистры, сумматор или схемы, Схема сравнения относится к классам 1В, 2А, 3А, 4А, так как оно является последовательным, многоходовым, сложным, использует двоичную систему счисления. Соответственно, он имеет преимущества и недостатки, присущие этим классам.

Предлагаемая классификация устройств приведения по модулю позволяет систематизировать известные структуры устройств и использовать системный подход при их проектировании и анализе.

3. Разработка схемных решений быстродействующего устройства приведения чисел по модулю на трехсумматорном ФЧО

3.1 Быстродействующие устройства приведения числа по модулю

При проектировании аппаратных и программно-аппаратных криптосистем с открытым ключом актуальным становится задача разработки эффективных схем для реализации одной из базовых операций – приведения числа по модулю.

В работе было рассмотрено устройство приведения 2n разрядного числа по n-разрядному модулю за n/2 шагов. В нем для формирования очередного частичного остатка r_i потребовались восемь двоичных сумматоров, что приводит к повышению сложности устройства. В данной работе рассматриваются устройства приведения числа по модулю(УПЧМ), где для формирования i-го частичного остатка используется только три двоичные сумматоры, что существенно упрощает аппаратную реализацию таких устройств (рисунок 3.1).

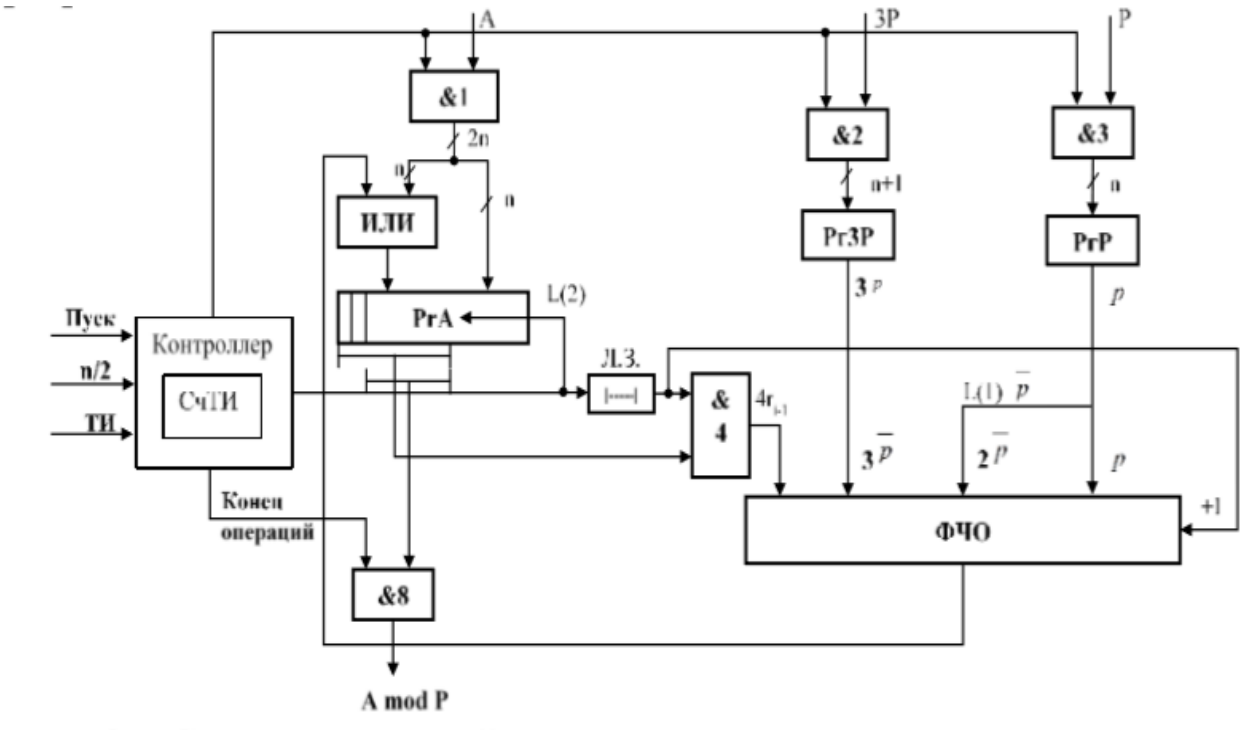

Рисунок 3.1 – Структура быстродействующего устройства приведения числа по модулю последовательного действия

Устройство состоит из регистров РгА, РгР и Рг3Р, формирователя частичного остатка ФЧО, блока схем $\&_1 \div \&_3$, $\&_8$ и ИЛИ, а также контроллера, содержащего счетчик тактовых сигналов СчТИ.

Регистр РгА, состоящий из 2n+2 разряда, предназначен для сдвига на два разряда влево и хранения числа, проводимого А, а n-разрядный РгР и (n+1) - разрядный Рг3Р – для хранения модуля Р и утроенного модуля 3Р, соответственно. Частичный остаток r_i от предыдущего остатка $r_{i-1,}$ сдвинутый на два разряда влево формируется посредством ФЧО.

На вход контроллера подаются сигнал «ПУСК», тактовые сигналы ТИ и двоичный код числа сдвигов n/2.

Основным узлом устройства является ФЧО, который состоит из сумматоров СМ1, СМ2, СМ3, блоков схем $\&_5$, $\&_6$, $\&_7$, а также логического элемента ИЛИ (рисунок 3.2)

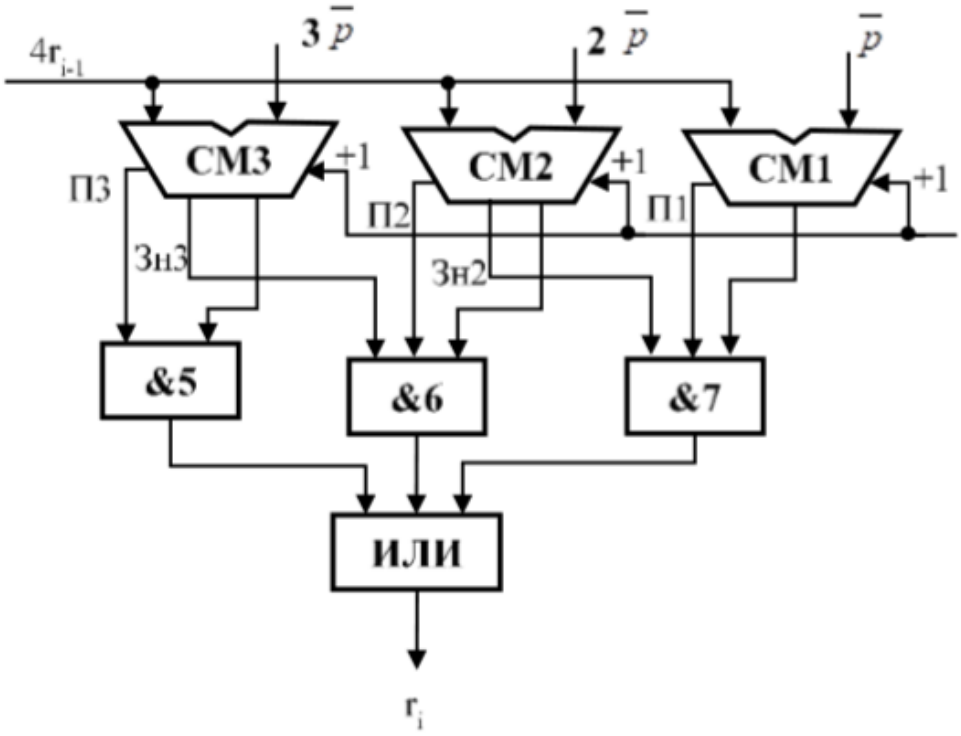

Рисунок 3.2 - Функциональная схема формирователя частичных остатков

На входы сумматоров СМ3 и СМ2 подаются значения $3\overline{P}$ и $2\overline{P}$ из инверсных выходов регистров Рг3Р и РгР со сдвигом влево L (1) $\overline{P}$ на один разряд, а на сумматор СМ1 – значение $\overline{P}$ из регистра РгР без сдвига. Кроме того, в момент выполнения вышеуказанных операций задержанный на ЛЗ тактовые импульсы ТИ также подается в младшие разряды сумматоров, что переводит обратные коды $3\overline{P}$, $2\overline{P}$ и $\overline{P}$ в дополнительные.

На выходе ФЧО значение r_i определяется путем анализа соотношений значений предыдущего остатка r_{i-1}, умноженного на четыре, т.е. $4r_{i-1}$, со значениями кодов модуля – Р, 2Р и 3Р.

В таблице 3.1 приведены виды выполнимых операций, необходимых для вычисления частичного остатка r_i по результатам сравнения $4r_{i-1}$ со значениями модулей.

Таблица 3.1 – Операции для вычисления r_i по условиям сравнения $4r_{i-1}$ с модулями

Условия сравнения	Операции
$4r_{i-1}<P$	$4r_{i-1}$
$P\leq 4r_{i-1}<P$	$4r_{i-1}-P$
$2P\leq 4r_{i-1}<3P$	$4r_{i-1}-2P$
$3P\leq 4r_{i-1}$	$4r_{i-1}-3P$

Для вычисления значения r_i в сумматоре СМ3 выполняется операция $4r_{i-1}-3P$, что соответствует операции сложения в дополнительном коде $4r_{i-1}+3\overline{P}+1$, в сумматоре СМ2 выполняется операция $4r_{i-1}-2P$, которая соответствует операции сложения в дополнительном коде: $4r_{i-1}+\overline{P}+1$. Во всех сумматорах операции выполняются параллельно, поэтому переносы из знаковых разрядов П3, П2, П1 и знаки Зн3, Зн2 и на соответствующих выходах сумматоров вырабатываются одновременно.

В процессе вычислений r_i на выходах сумматоров могут формироваться остатки с положительными и отрицательными знаками. Из них необходимо выбрать наименьший положительный остаток, блокируя другие положительные и все отрицательные остатки.

Условия формирования наименьшего положительного остатка r_i с блокировкой других остатков в зависимости от значения переносов П3, П2, П1 и знаков Зн3, Зн2, Зн1 приведены в таблице 3.2.

Таблица 3.2 – Условия формирования наименьшего положительного остатка r_i

	Переносы			Знаки		Выходы сумматоров			$4r_{i-1}$
	П3	П2	П1	Зн3	Зн2	СМ3	СМ2	СМ1	
1	1	1	1	0	0	r_i	-	-	-
2	0	1	1	1	0	-	r_i	-	-
3	0	0	1	1	1	-	-	r_i	-
4	0	0	0	1	1	-	-	-	$r_i=4r_{i-1}$

Из таблицы 3.2, а также схемы на рисунке 3.2, видно, что при П3=П2=П1 и Зн3=Зн2=0 наименьший положительный остаток r_i, передаваемый на выходы

блока схем $\&_5$ сигналом П3=1, определяется как разница $4r_{i-1}$-3Р с выходов сумматора СМ3. При этом сигналами Зн3=Зн2=0 блокируются передачи кодов с выходов СМ2 и СМ1 блоками схем $\&_6$ и $\&_7$.

При значениях П3=0 и П2=П1=1, Зн3=1 и Зн2=0 положительные разницы формируются на выходах сумматоров СМ2 и СМ1. При этом выходы сумматора СМ3 блокируется сигналом П3=0, а сигналам Зн2=0 блокируется выход сумматора СМ1. Сигналом игналом Зн3=1 с выходов сумматора СМ2 остаток r_i и передается на выход ФЧО. При П3=П2=0 и П1=1 результат r_i формируется и передается с выходов сумматора СМ1. При этом сигналами П3=П2=0 блокируются выходы СМ3 и СМ2.

При П3=П2=П1=0 этими сигналами блокируется выходы СМ3, СМ2 и СМ1 и значение $4r_{i-1}$ останется в старших разрядах регистра РгА.

В целом устройство работает следующим образом. Разрешающий сигнал ПУСК на входах блоков конъюнкторов $\&_1 \div \&_3$ позволяет прием регистром РгА приводимого числа А, регистром Рг3Р – утроенного значения модуля 3Р и регистром РгР – одинарное значение модуля Р. Кроме этого, сигналом «ПУСК» осуществляется передача код числа сдвигов, необходимых для формирования результата, в СчТИ.

После окончания приема операндов контроллером первый тактовый импульс ТИ1 сдвигает содержимое РгА на два разряда влево. При этом формируемое в старших разрядах РгА значение $4r_0$ передается на входы блока логических схем $\&_4$, а на его управляющие входы поступает ТИ1, задержанный на элементе задержки Л.З.

На входы сумматоров ФЧО поступает значение $4R_0$ со схемы $\&_4$, а также значения $3\overline{P}$, $2\overline{P}$, $\overline{P}$ и в зависимости от значений переносов из знаковых разрядов П3, П2, П1 и знаков разницы Зн3, Зн2 параллельно выполняются операций $4R_0+3\overline{P}+1$, $4R_0+2\overline{P}+1$, $4R_0+\overline{P}+1$.

Наименьший положительный остаток из выходов одного из сумматоров посредством схем $\&_5$, $\&_6$ и $\&_7$ и схем блока ИЛИ передается в старшие разряды регистры РгА, который является частичным остатком r_1. Одновременно с формирователем частичного остатка r_1, тактовый импульс ТИ1 уменьшает на единицу содержимое СчТИ.

После приема частичного остатка r_1 на старшие разряды регистра РгА из контроллера в схему поступает тактовый сигнал ТИ2, осуществляющий сдвиг содержимого РгА на два разряда влево. При этом в старших разрядах РгА формируется $4r_i$, который посредством схемы $\&_4$ подается в первые входы сумматоров ФЧО и образуется частичный остаток r_2. Кроме того сигналом ТИ2 уменьшается показание вычитающего счетчика на единицу.

После поступления n/2-го такта ТИ в старших разрядах РгА формируется результат. А в счетчике СчТИ установится 0, и выработает сигнал «Конец

операций», который осуществляет выдачу результата посредством блока схем $\&_8$.

Рассмотрим пример ускоренного приведения числа А по модулю Р. Предположим, что $A=894_{10}=001101111110_2$ и $P=35_{10}=100011_2$. Следовательно, $2P=70_{10}=01000110_2$, $3P=105_{10}=01101001_2$, а значения $r_0=001101_2=13_{10}$, и $4r_0=00110111_2=13*4+3=55_{10}$ получим из записи $A=001101111110_2$.

Для удобства все вычисления выполняются в десятичной системе счисления. Процедура определения искомого остатка R=894mod 35 приведена в талице 3.3.

Таблица 3.3 – Порядок вычсисления A mod P

Тактовые импульсы	Сумматоры	Вывод
ТИ1	$4r_0=55_{10}$ СМ3 СМ2 СМ1 _ 55 _ 55 _ 55 105 70 35 -50 -85 +20 $r_1=20_{10}$	П3 = П2 = 0 П1 = 1
ТИ2	$4r_1=20*4+3=83_{10}$ СМ3 СМ2 СМ1 _ 83 _ 83 _ 83 105 70 35 -22 +13 +48 $r_2=13_{10}$	П3 = 0 П1 = П2 = 1 Зн3=1, Зн2=0
ТИ3	$4r_2=13*4+2=54_{10}$ СМ3 СМ2 СМ1 _ 54 _ 54 _ 54 105 70 35 -51 -16 +19 $R=r_1=19_{10}$	П3 = П2 = 0 П1 = 1

Проверка: R= 894-]$\frac{894}{35}$[35=19_{10}

На рисунке 3.3 приведена матричная схема приведения числа по модулю на основе выше рассмотренного примера для чисел A=$a_{11}a_{10}$ ... a_1a_0; P= p_7p_6 ... p_1p_0.

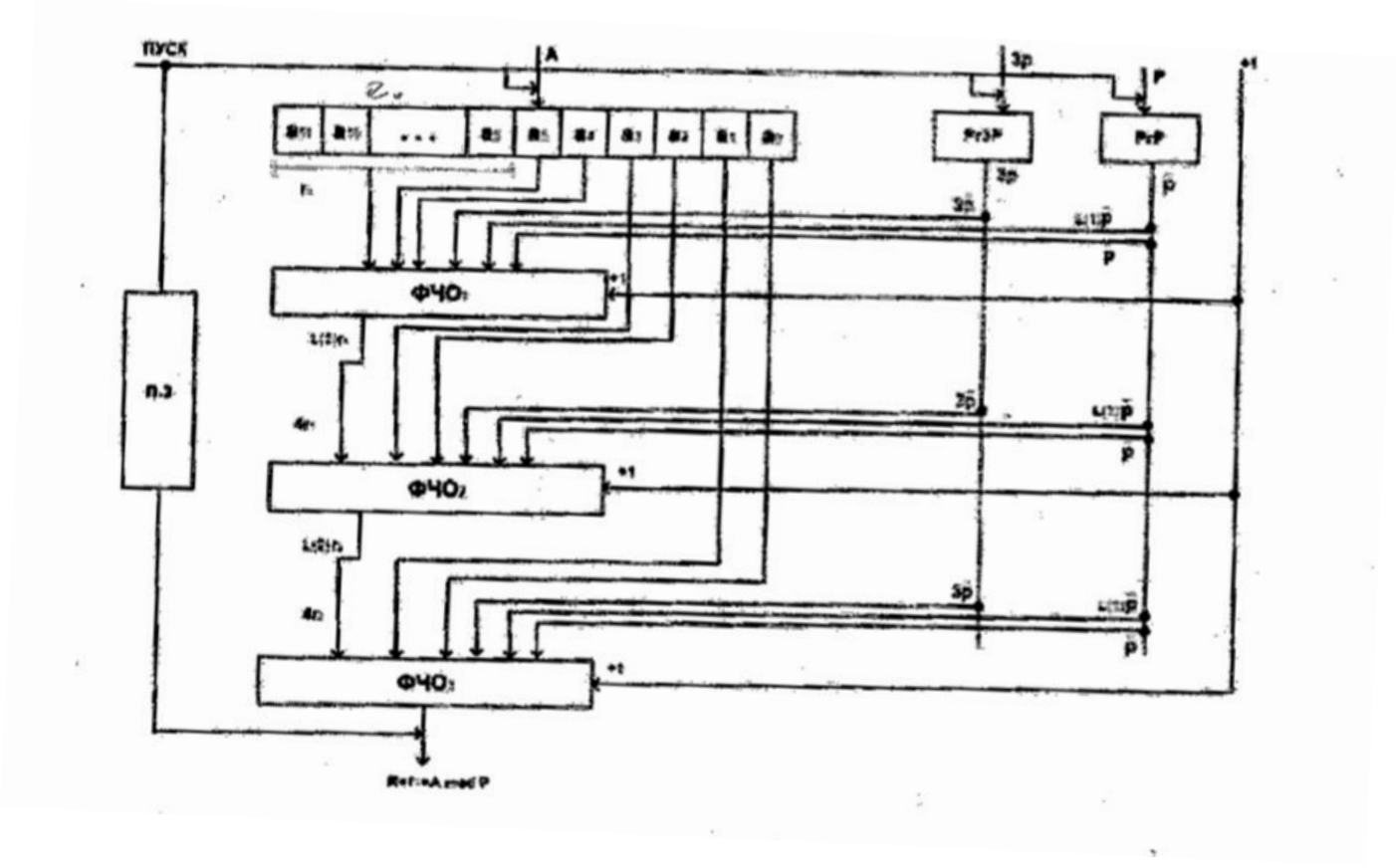

Рисунок 3.3 – Матричная схема приведения числа по модулю

В данном случае частичные остатки r_i на $ФЧО_i$ формируются путем последовательной передачи r_i на вход $ФЧО_{i+1}$ со сдвигом на два разряда влево с присоединением следующих двух разрядов приводимого числа A. Структура $ФЧО_i$ приведена на рисунке 3.4.

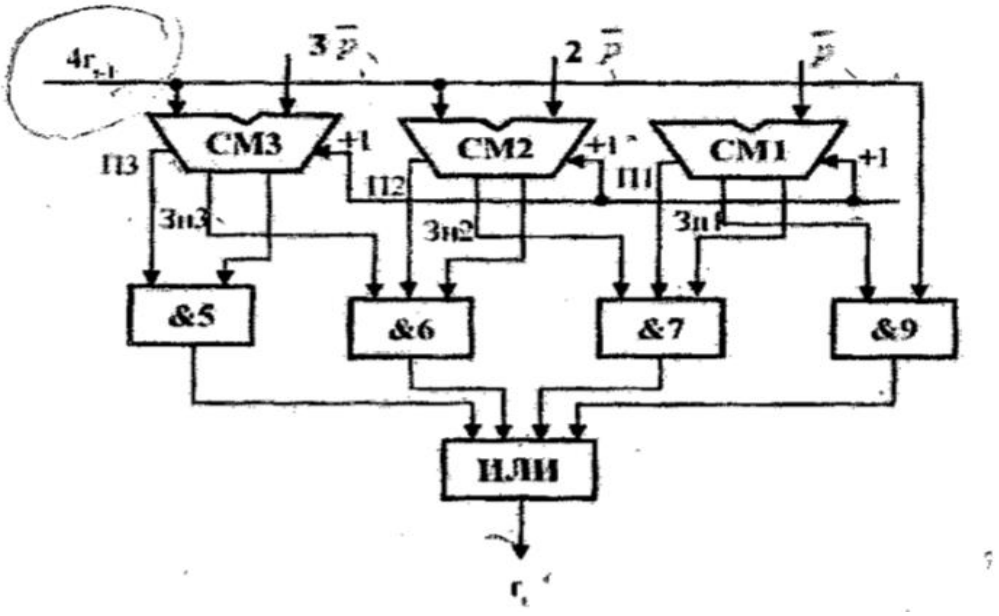

Рисунок 3.4 – Формирователь частичных остатков для матричных схем приведения по модулю

Как видно в состав ФЧО добавлен блок логических &9, формирующий значение $4r_{i-1}$ при Зн3=Зн2=Зн1=1.

После приема операндов в регистры РгА, Рг3Р и РгР по сигналу «ПУСК» разряды $a_{11} \div a_4$ регистра РгА ФЧО подаются на входы $ФЧО_1$. На входы $ФЧО_1$ других ФЧО подаются значенмя $3\overline{P}$, $2\overline{P}$ и $\overline{P}$ из регистров Рг3Р и РгР.

Первый частичный остаток r_i, формируемый $ФЧО_1$, сдвигается на два разряда влево и подается на входы $ФЧО_2$. При этом к коду $4r_1$ присоединяются разряды a_3 и a_2 числа А. Затем формируется частичный остаток r_2 на $ФЧО_2$, который также сдвигается на два разряда влево и с присоединением разрядов a_1 и a_0 подается на входы $ФЧО_3$. Наконец, задержанным сигналом «Пуск» на линии задержки Л.З. выходе $ФЧО_3$ формируется окончательный результат $R=r_3$.

В матричной схеме приведения числа по модулю заложен очень важный потенциал повышения производительности – возможность конвейеризации. На рис. 5 приведена конвейеризованная матричная схема приведения числа по модулю для выше рассмотренного примера, состоящая из К=3 ступеней. Каждая ступень конвейера состоит из ФЧО, буферных регистров $Ргr_i$ и буферных регистров для запоминания тех битов числа А, которые будут участвовать при вычислении следующих частичных остатков. Ускорение вычислений S данной схемы определяется по формуле:

$$S=\frac{NK}{K+(N-1)}; \qquad (3.1)$$

где, N - число входных потоков приводимых чисел и модулей, а К – число ступеней конвейера. При N->∞, ускорение стремится к величине К.

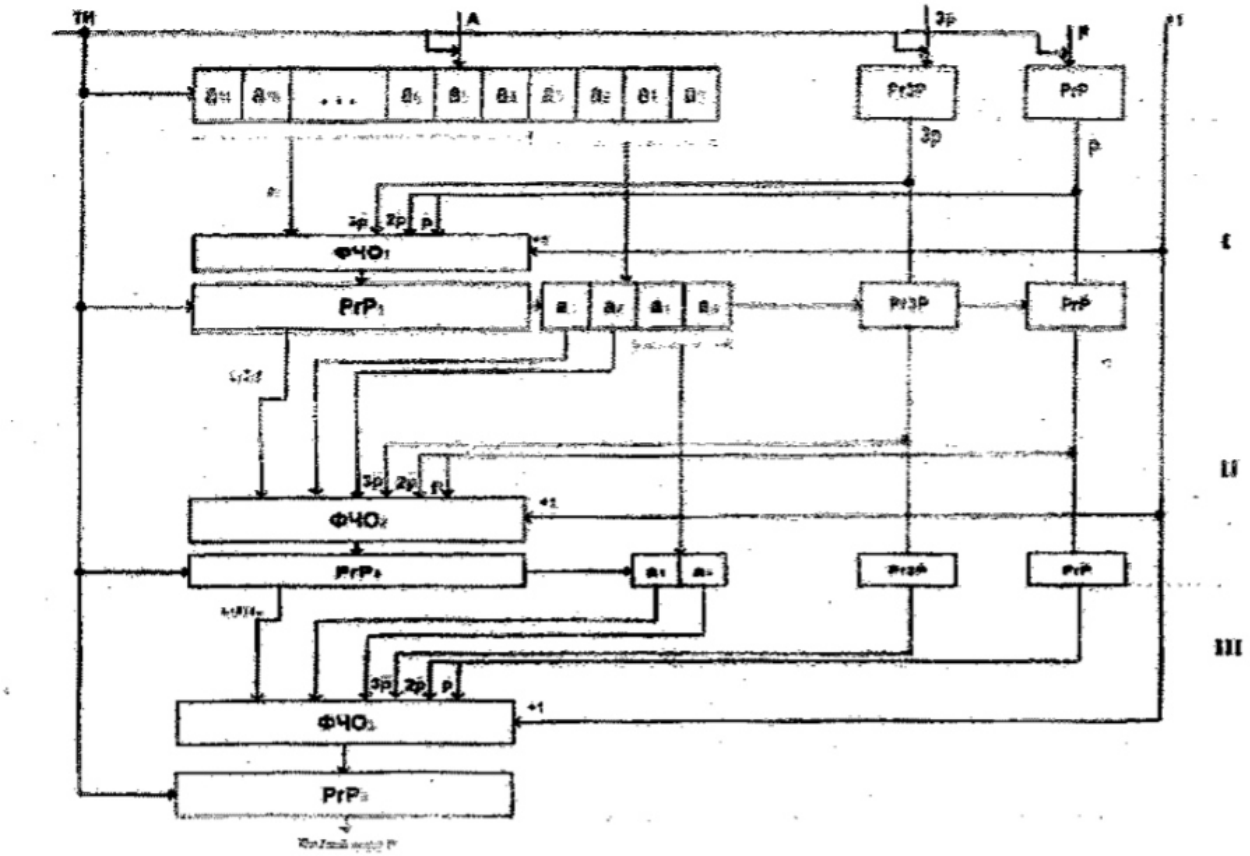

Рисунок 3.5 – Конвейеризованная матричная схема приведения числа по модулю

В заключении следует отметить, что разработанная схема формирователя частичных остатков позволяет построить быстродействующее устройство приведения чисел по модулю.

3.2 Устройство приведения чисел на трех сумматорах и деление

На рисунке 3.6 приведена временная диаграмма работы устройства приведения числа $A_{a7 \div a0} = 187_{10} = 10111011_2$ с разрядностью 8 со значениям модулей $P = 14_{10} = 1110_2$.

Как видно из рисунка, на диаграмме указаны значения трех модулей со значениями $P = 14_{10} = 1110_2$ $P2 = 28_{10} = 11100_2$ и $P3 = 42_{10} = 101010_2$. Значение P является основным модулем, а P2, P3 соответственно P умноженный на целые числа 2 и 3. Вследствий этого количество сумматоров увеличелось на три, но время работы устройства приведения числа по модулю на делителях с блокировкой отрицательных остатков уменьшилось в два раза. Теперь частичный остаток будет вычеслятся по условию сравнения $4r_{i-1}$ с модулями.

Таблица 3.4 – Уловия сравнения и операции

Условия сравнения	Операции
$4r_{i-1} < P$	$4r_{i-1}$
$P \leq 4r_{i-1} < P2$	$4r_{i-1} - P$
$P2 \leq 4r_{i-1} < P3$	$4r_{i-1} - P2$
$P3 \leq 4r_{i-1}$	$4r_{i-1} - P3$

Как видно из таблицы, условия сравнения будут решать какой из сумматоров будет выполнять операцию сложения частичного остатка и модуля. Первое условие гласит о том, что если модуль окажется меньше частичного остатка, то отрицательные остатки блокируется и предыдущий остаток останется неизменным. А алгоритм поиска частного операции деления будет напрямую связано с сумматорм, который будет задействован в формировании частичного остатка. Так как, при получении частичного остатка, только значение одного из сумматоров будет выведено на выход. Но если все три значение суматоров окажется отрицательным, то в этот момент выходные значение сумматоров обнуляются и в общей схеме остается предыдущее значение. В следующей таблице показано, что должно формировать в регистре Divide для получение значения частного.

Таблица 3.5 – Формирующие значения для сумматоров

Операции	Сумматор	Формирующее значение
$4r_{i-1}$	-	00_2
$4r_{i-1} - P$	СМ1	01_2
$4r_{i-1} - P2$	СМ2	10_2
$4r_{i-1} - P3$	СМ3	11_2

На каждом такте в регистре Divide сперва формируется значение по таблице, а после оно конкатенируется с предыдущим значением этого же регистра и на конечном такте работы получается искомое.

Теперь, если перейдти к временной диаграмме на рисунке 3.6, то мы видим что для числа А $r_0 = 11_{10} = 1011_2$. На этой диаграмме после первого тактового импульса ТИ1 содержимое регистра А сдвигается влево на два разряда и в регистре формируется $(4r_0 + a_3a_2) = 46$. Полученное значение удовлетворяет условию $P3 \leq 4r_{i-1}$ и тогда формируется частичный остаток $r_1 = 46 - P3 = 4$, что соответсвует работе сумматору СМ3 в схеме. Так как, рабочее значение получилось с сумматора СМ3, то в регистре Divide будет сформировано значение 11_2. При следующем такте ТИ2 содержимое регистра А сдвигается влево на два разряда и в регистре формируется $(4r_1 + a_1a_0) = 19$. Как видно из полученного значения, оно соответствует условию сравнения $P \leq 4r_{i-1} < P2$, и

тогда формируется частичный остаток $r_2 = 19 - P = 5$. Полученный частичный остаток будет нашим искомым остатком $R = 5$, что можно легко проверить следующей операцией R = 187 mod 14 = 5. Второй и последний такт показывает, что значение сумматора СМ1 является рассчитанным частичным остатком. В этом случае, по таблице в регистре Divide получится значение 01_2, а результатом конкатенации с предыдущим значение будем имет вид 1101_2. А это значение и есть частное операции 187 / 14 = 13

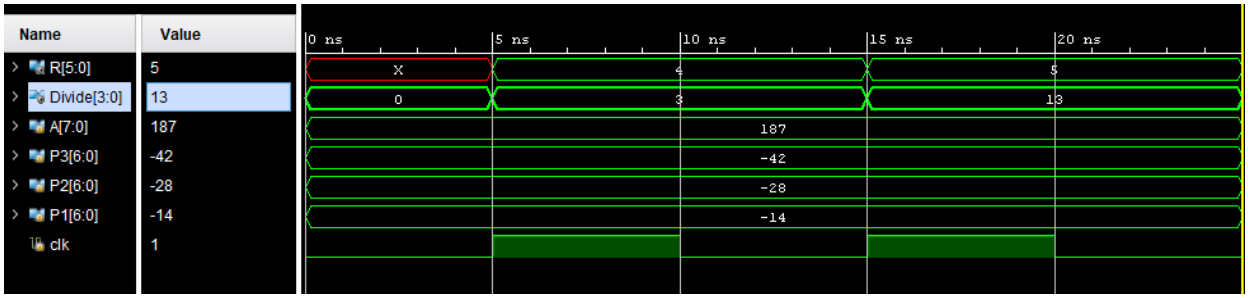

Рисунок 3.6 – Диаграмма работы алгоритма для 8 разрядного числа

Аналогично на рисунке 3.7 приведена диаграмма работы приведения числа по модулю при значениях $A = 27317_{10}$ с разрядностью 16 при $P = 209_{10}$ с разрядностью 8. Как видно из диаграммы модулям $P2$ и $P3$ соответсвуют значения 418 и 627, а значение $r_0 = 106_{10} = 1101010_2$. Так как, значения регистра *A* на каждом такте будет сдвигатся на два, то для решения операции R = 27317 mod 209 достаточно всего четыре такта. Первым тактом ТИ1 будет сформировано $(4r_0 + a_7a_6) = 426$. По таблице условии полученное значение лежит между P2 и P3 и соответственно сформируется частичный остаток $r_1 = 426 - P2 = 8$. В этом такте регистр Divide получить значение 10_2, потому что будет задействован сумматор СМ2. На рисунке видно,что первый такт сформировал значение $10_2 = 2_{10}$. Следующий такт ТИ2 сдвинет влево значения регистра *A* на два разряда и тогда сформируется новое значение $(4r_1 + a_5a_4) = 35$. Полученное значение оказалось меньше всех модулей, а это значит блокировку отрицательного остатка и получение старого значения частичного остатка $r_2 = 35$. На диаграмме это место указано нулем для того, чтобы показать что в этом такте блокировалось отрицательный остаток. Второй такт показывает, что данные ни одного сумматора не будет задействовано в формировании значения регистра Divide. А это значит что по таблице в этот регистр будет записано два логических нуля и результатом конкатенации окажется 1000_2, что соответствует десятичной цифре 8. В этом мы можем удостоверится посмотрев на рисунок 3.7, где показана временная диаграмма. Такт ТИ3 сдвинет влево значения регистра *A* на два разряда и сформирует значение $(4r_2 + a_3a_2) = 141$. Получившое значение соответствует первому условию, а это значит частичный

остаток $r_3 = 141$. И в этом же такте происходит предыдущее действие, так как значение всех сумматоров оказались отрицательными. Значит регистр Divide сперва получить значение 00_2 и результатом конкатенации будет $100000_2 = 32_{10}$. На последнем такте ТИ4, при сдвиге регистра A влева на два разряда получится $(4r_3 + a_1a_0) = 565$. По условию, значение частичного остатка $r_4 = 565 - P2 = 147$. Полученный частичный остаток будет нашим искомым остатком $R = 147$, что соответсвует операцией R = 27317 mod 209 = 147. Последний такт задействует значение сумматора СМ2, а это значит что регистр Divide получит значение 10_2 по таблице и конечным результатом конкатенации будет $10000010_2 = 130_{10}$. Полученное значение будет являтся частным операции деления 27317 / 209 = 130.

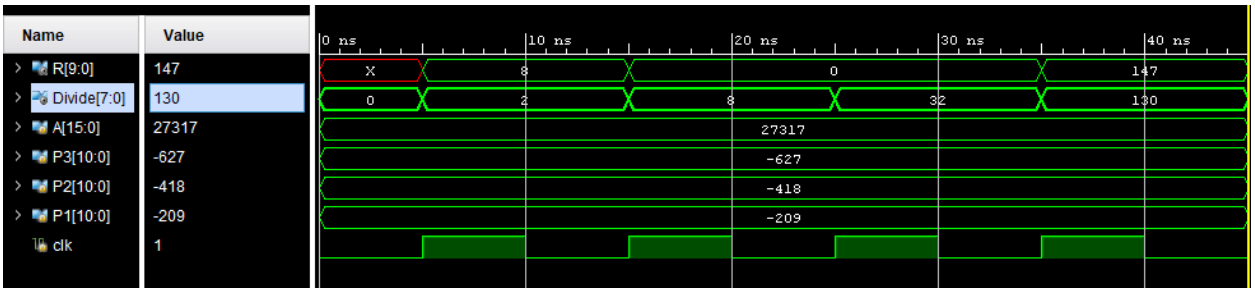

Рисунок 3.7 – Диаграмма работы алгоритма для 16 разрядного числа

Также были вычислены значения остатков для числа A разрядностью 32 и 64. На рисунке 3.8 приведены зависимость затраченных ресурс ПЛИС Artix-7 от разрядности приводимого числа A. На этом рисунке LUT (Look-Up Table) – таблица преобразования), FF (Flip-Flop) – тригеры, BUFG – независимый от архитектуры глобальный буфер., IO(Input/Output) – входы/выходы.

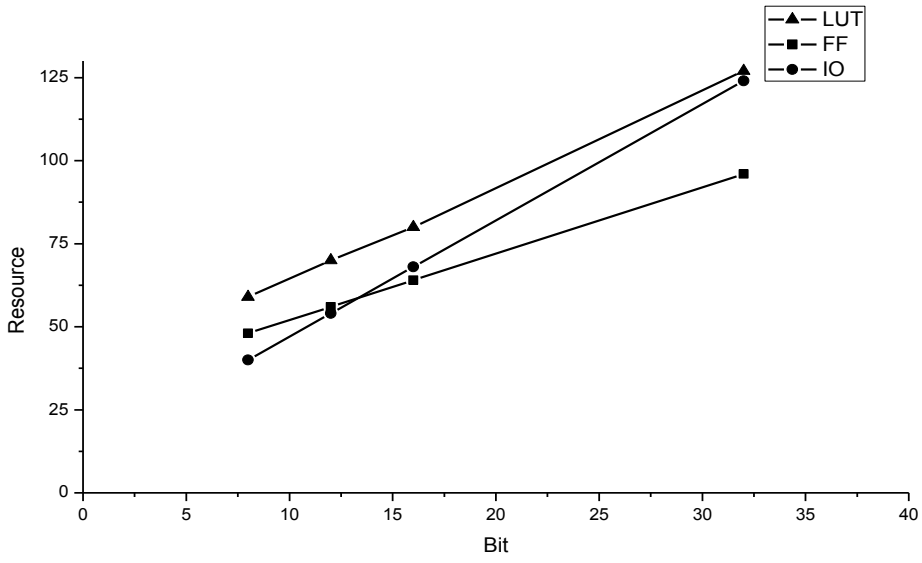

Рисунок 3.8 – Количество затраченных ресурсов

Количество использованных ресурсов LUT и FF не превышает даже 1% от ресурсов Artix-7. Это позволяет использовать этот ПЛИС и для более большого количество разрядных чисел, чем 64. На рисунке 3.9 показана плата Nexys ПЛИС Artix-7.

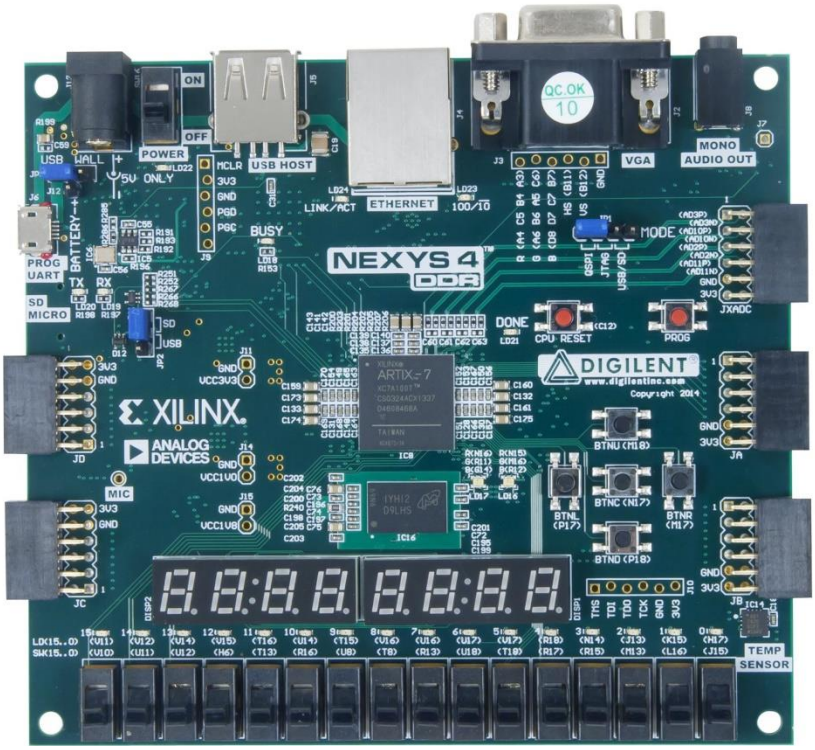

Рисунок 3.9 – Плата Nexys ПЛИС Artix-7

Для определения быстродействия процесса приведения числа по модулю был использован внутренний генератор ПЛИС с частотой 100 МГц. Известно, что время работы алгоритма прямо пропорционально половине длины входной цифры. Зная эти данные можно рассчитать затраченное время на приведения числа к модулю:

$$t = (n/4)/f, \qquad (3.2)$$

здесь n – длина разряда входного числа, f – частота работы ПЛИС. Например, можно определить быстродействие, для 16 разрядного числа, которое будет равно 40 нс.

Программа настройки ПЛИС была написана на языке “Verilog”. Листинг программы представлен в Приложении А.

Вывод

В данной главе дипломного проекта была поставлена задача разработать быстродействующее устройство приведения чисел по модулю. По проведенным

анализам существующих методов приведения чисел было разработано схемные решения устройства приведения чисел по модулю со сдвигом на два разряда частичного остатка, что позволяет ускорить операцию в два раза. При этом был разработан формирователь частичных остатков (ФЧО) на базе трех сумматоров. Для проверки алгоритма приведения чисел по модулю было осуществлено путем реализации устройств на ПЛИС Artix-7. Программа настройки ПЛИС была написана на языке “Verilog”.

4 Технико-экономическое обоснование

Данный дипломный проект подразумевает проектирование быстродействующее устройство приведения числа по модулю на трехсумматорном ФЧО (формирователь частичных остатков).

В экономической части дипломной работы будут рассчитаны все расходы на разработку, требуемых материалов и устройств, затраты на программные обеспечения, затраты на электроэнергию и оплату труда, а также амортизацию основного оборудования.

4.1 Расчет трудоемкости разработки программного продукта

Приведен перечень основных этапов и работ, которые нужно выполнить для определения трудоемкости разработки программного обеспечения. Трудоемкость работы определялась согласно нормам времени на проведение расчетов, анализа и исследований. Форма разделения работ по этапам с указанием трудоемкости их выполнения приведена в таблице 4.1.

Таблица 4.1 – Распределение работ по этапам и оценка их трудоемкости

Этапы разработки ПО	Вид работы	Трудоемкость, чел. час.
Этап 1	Постановка задач	9
Этап 2	Разработка и утверждение технического задания на разработку ПП	12
Этап 3	Поиск и изучение подобных программ и устройств	14
Этап 4	Поиск и изучение сопутствующей литературы	19
Этап 5	Составление аналитических графиков ПО	14
Этап 6	Оформление теоретической части темы дипломного проекта	22
Этап 7	Разработка практической части дипломного проекта	24
Этап 8	Выбор среды разработки программного обеспечения	4
Этап 9	Реализация проекта	32
Этап 10	Отладка программного обеспечения	12
Этап 11	Оформление отчета и выводов	13

Продолжение таблицы 4.1

Этапы разработки ПО	Вид работы	Трудоемкость, чел. час.
Этап 12	Тестирование проекта	12
Этап 13	Итог разработки программного продукта	16
Итого: трудоемкость выполнения дипломного проекта		216

Продолжительность рабочего дня равна 8 часам. (216: 8 = 27)

4.2 Расчет затрат на разработку программного продукта

Определение затрат на разработку программного продукта производится на основе существующей сметы, которая включает следующие статьи:

– материальные затраты;
– затраты на оплату труда;
– социальный налог;
– амортизация основных фондов;

Статья «Материальные затраты» состоит из основных и вспомогательных материалов, энергии, которые необходимы для разработки программного продукта. Расчет затрат на материальные ресурсы производится по форме, приведенной в таблице 4.2.

Таблица 4.2 – Затраты на материальные ресурсы

Наименование материала	Марка	Ед. измерения	Количество	Цена за ед. в тенге	Сумма в тенге
Офисная бумага, А4	Белоснежка	Пачка	2	1500	3000
Наименование материала	Марка	Ед. измерения	Количество	Цена за ед. в тенге	Сумма в тенге
Тетрадь А4 (96 листов)	Magister	Штук	1	250	250
Блокнот (96 листов)	OFFICE	Штук	1	650	650
Ручка	Rotomac	Штук	5	120	600
Карандаш	Hatber	Штук	5	65	325

Продолжение таблицы 4.2

Наименование материала	Марка	Ед. измерения	Количество	Цена за ед. в тенге	Сумма в тенге
Компьютерная мышь (беспроводная)	HP Classic mouse	Штук	1	7990	7990
Итого					12815

При покупке нового ноутбука Acer A71 в нем предусмотрены встроенная операционная система и дополнительное программное обеспечение, поэтому затраты на покупку новой операционной системы Windows 8.1 и лицензионную MS Office производиться не будут.

Таблица 4.3 – Затраты на основное оборудование, необходимые для проекта

Наименование материала	Марка	Ед. измерения	Количество	Цена за ед. в тенге	Сумма в тенге
Ноутбук	Acer A71	Шт.	1	190000	190000
Принтер	Samsung M2070	Шт.	1	48500	48500
Модем	TP-Link TL-WR740N	Шт.	1	8900	8900
Итого					247400

Общая сумма затрат на материальные ресурсы ($З_м$) определяется по формуле:

$$З_м = \sum P_i \times Ц_i, \quad (4.1)$$

где P_i– расход i-го вида материального ресурса, натуральные единицы;
$Ц_i$– цена за единицу i-го вида материального ресурса, тг;
i – вид материального ресурса;
n – количество видов материальных ресурсов.

$З_м = 12815 + 247400 = 260\ 215$ (тг)

Материальные затраты на дипломный проект составят 366 284 тенге. Все материальные затраты лягут на основные средства

4.3 Расчет затрат на электроэнергию

Важно рассчитать затраты на электроэнергию, потому что в процессе работы используется электрооборудование. Время работы оборудования для разработки программного продукта берется равным 224 часов для ноутбуков и модема, данное количество часов было рассчитано в таблице 4.1. Для принтера время работы для разработки программного продукта берется равным 12 часов, так нет необходимости постоянного его использования.

$$Э = З_{эл.эн.обор} + З_{доп.нуж}, \quad (4.2)$$

где $З_{эл.эн.обор}$ – затраты на электроэнергию оборудования;
$З_{доп.нуж.}$ – затраты электроэнергии на дополнительные нужды.

Расходы электроэнергии на оборудование рассчитывается по формуле:

$$З_{эл.эн.обор} = \sum W \times К_{исп} \times S \times T, \quad (4.3)$$

где W – потребляемая мощность, Вт;
$К_{исп}$ – коэффициент использования ($К_{исп}$ = 0,7..0,9);
T – время работы;
S – тариф (1кВт/ч = 18,32 тг).

Сводные результаты расчета затрат на электроэнергию представлены в таблице 4.4.

Таблица 4.4 – Затраты на электроэнергию

Наименование приборов	Паспортная мощность, кВт	Коэффициент мощности	Время работы оборудования, ч	Цена ЭЭ тг/ кВтч	Сумма, тг
Ноутбук	0,6	0,7	216	18,32	1661,9
Модем	0,08	0,9	100	18,32	131,9
Принтер	0,5	0,9	12	18,32	98,9
Кондиционер	0,8	0,9	200	18,32	2638,08
Освещение	0,3	0,7	216	18,32	830,9
Итого					5361,7

$З_{эл.эн.обор}$ = 1661,9+131,9+98,9+2638,08+830,9=5361,7 (тенге)

Затраты на дополнительные потребности берутся по укрупненному показателю в размере 5% от затрат на оборудование:

$$З_{доп.нуж} = 5\% \times З_{эл.эн.обор} \quad (4.4)$$

Затраты на дополнительные потребности рассчитаны по формуле (4.4):

$З_{доп.нуж} = 0,05 \times 5361,7 = 268,08$ (тенге)

Таким образом суммарные затраты на электроэнергию составляют:

Э = 5361,7 + 268,08= 5629,78(тенге)

4.4 Расчет затрат на оплату труда

Над разработкой проекта работают два сотрудника:

– руководитель проекта – он изучает предметную область, проводит анализ требований к системе, занимается внедрением и поддержкой;

– разработчик – создание и реализует модель, занимается тестировкой и отладкой продукта;

Общая сумма затрат на оплату труда ($З_{тр}$) определяется по формуле:

$$З_{тр}=\sum ЧС_i \times Т_i \quad (4.5)$$

где $ЧС_i$– часовая ставка i-го работника, тг;

Ti – трудоемкость разработки модели, чел.×ч;

i – категория работника;

n – количество работников, занятых разработкой ПП.

На этапах разработки, участники разработки задействованы неравноценно, для этого необходимо рассчитать часовую ставку работника, а затем общий размер заработной платы.

Часовая ставка работника может быть рассчитана по формуле:

$$ЧС_i=\frac{ЗП_i}{ФРВ_i} \quad (4.6)$$

где Зпi – месячная заработная плата i-го работника, тг;

ФРВi- месячный фонд рабочего времени i-го работника, час

Месячная заработная плата сотрудников:

Руководитель проекта – 180 000 тг;

Разработчик – 120 000 тг.

$ЧС_i$= 180 000 / 22 × 8 = 1 022,72 тг/ч

$ЧС_i$= 120 000 / 22 × 8 = 681,81 тг/ч

Часовая ставка научного руководителя составляет 1 022,72 (тг/ч), трудоемкость разработки – 90 ч. Часовая ставка разработчика составляет 681,81 (тг/ч), трудоемкость разработки – 216 ч.

Рассчитаем общую сумму затрат на оплату труда по формуле (4.5):

$З_{тр}$ = 1 022,72 × 90 + 681,81 × 216 = 239 315,76 (тенге)

Сводные результаты расчета затрат на оплату труда показаны в таблице 4.5.

Таблица 6.5 – Расчёт основной заработной платы разработчиков.

Категория работника	Квалификация	Трудоемкость разработки ПП, час.	Часовая ставка, тг/ч	Сумма, тг.
Руководитель проекта	Профессор	90	1 022,72	92 044,8
Разработчик	Студент	216	681,81	147270,96
Итого				239 315,76

4.5 Расчет затрат по социальному налогу

Социальный налог – согласно Налоговому кодексу Республики Казахстан составляет 9,5 % от ФОТ (фонда оплаты труда). Следует отметить, что пенсионные отчисления не облагаются социальным налогом.

$$С_н = (ФОТ- ПО)\times 0,095 \quad (4.7)$$

где ПО - отчисления в пенсионный фонд, 10% от ФОТ.

Социальный налог рассчитываем по формуле (4.7):

ПО = 239 315,76× 0,1 = 23 931,576 тенге;
$С_н$ = (239 315,76 – 23 931,576) × 0,095 = 20 461,49 тенге

Сводные результаты расчета затрат представлены в таблице 4.7.

Таблица 4.7 - Начисление социального налога

Категория работника	Количество человек	Заработная плата, тг	Пенсионные отчисления, тг	Социальный налог, тг
Руководитель проекта	1	92 044,8	9 204,48	7 869,75
Разработчик	1	147270,96	14 727,09	12591,66
Итого				20 461,49

4.6 Амортизация основных фондов и прочие затраты

Годовые нормы амортизации ОФ принимаются по налоговому кодексу РК или определяются, исходя из возможного срока полезного использования ОФ. Амортизация основных фондов определяется:

$$А_г = \frac{С_{об} \times Н_а}{100} \quad (4.8)$$

где, $С_{об}$ – стоимость оборудования;

$Н_а$ – норма амортизации (норма амортизация = 20);

По формуле 4.8 рассчитаем сумму амортизационных отчислений за год для ноутбука:

$$А_г = \frac{190000 \times 20}{100} = 38\,000 \text{ тг}$$

Рассчитаем сумму амортизации за время разработки:

$$А_р = \frac{38\,000 \times 27}{365} = 2\,811 \text{ тг}$$

Аналогичным способом рассчитаем сумму амортизации для остального оборудования.

Результаты расчетов приведены в таблице 4.6

Таблица 4.8 - Амортизация основных фондов

Наименование оборудования и ПО	Стоимость оборудования и ПО, тг	Годовая норма амортизации, %	Сумма амортизации за год, тг	Сумма амортизации за время разработки, тг
Ноутбук	190 000	20	38 000	2 811
Принтер	48500	20	9700	744,1
Модем	8900	15	1355	102,4
RAD Studio XE8	0	10	0	0
ИТОГО амортизация основных средств			49055	3657,5

Смета затрат на разработку программного продукта

На основании полученных данных по отдельным статьям составляется смета затрат на разработку программного продукта по форме, приведенной в таблице. Для разработки ПП использовался Интернет, сумма оплаты 22 404,00 тг.

Таблица 4.9 – Смета затрат на разработку программного продукта

Статьи затрат	Сумма, тг
Затраты на оборудование и материальные расходы	260215
Затраты на программное обеспечение	0
Затраты на оплату труда	239 315,76
Социальные налоги	20 461,49

Затраты на электроэнергию	5629,7
Амортизация основных фондов	3657,5

Продолжение таблицы 4.9

Статьи затрат	Сумма, тг
Прочие расходы (интернет)	22 404
Итого по смете	551 683,54

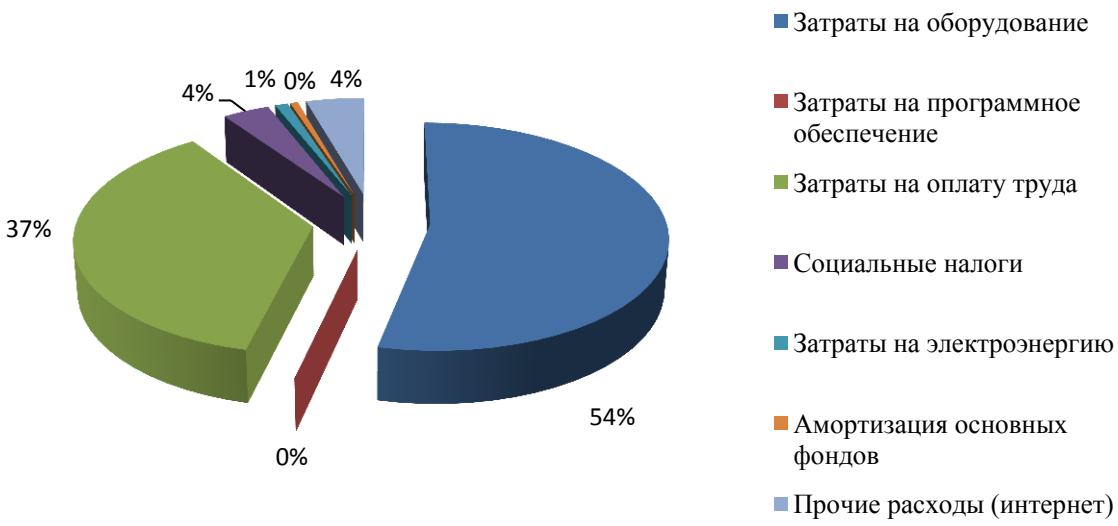

Рисунок 4.1 – Диаграмма структуры затрат

4.7 Определение возможной (договорной) цены программного продукта

Величина возможной (договорной) цены программного продукта устанавливается на основе эффективности, качества и сроков её выполнения на уровне, отвечающем экономическим интересам заказчика (потребителя) и исполнителя.

Договорная цена ЦД для прикладных программных продуктов рассчитывается по формуле:

$$Ц_Д = З_{НИР}(1 + \frac{Р}{100}) \quad (4.9)$$

где $З_{НИР}$- затраты на разработку ПП, тг;
Р – средний уровень рентабельности ПП. % (принимается в размере 20%).

$Ц_Д$ = 551 683,54× (1 + 20/100) = 551 683,54+110 336,7 = 662 020,24 тенге

Далее определяется цена реализации с учетом налога на добавленную стоимость (НДС), ставка (НДС) устанавливается законодательно. Налоговым Кодексом РК. На 2017 год ставка НДС установлена в размере 12%.

Цена реализации с учетом НДС рассчитывается по формуле:

$$Ц_р = Ц_д + Ц_д \times НДС \quad (4.10)$$

662 020,24 + 662 020,24 × 0,12 = 662 020,24 + 79 442,4=
= 741 462,67 тенге

Рассчитанную возможную цену ПП можно округлить до 742 000,00 тенге.

Вывод

Данная глава дипломного проекта содержит экономические расчеты, которые позволяют определить затраты необходимые для разработки программного продукта. Расчеты включают в себя:

– расчет трудоемкости разработки программного продукта;
– расчет затрат на разработку программного продукта;
– расчет затрат на электроэнергию;
– расчет затрат на оплату труда;
– расчет затрат по социальному налогу;
– амортизация основных фондов и прочие затраты.

Договорная цена программного продукта будет равна 742 000,00 тенге. Смета затрат на разработку программного продукта будет равна 551 683,54тенге. Прибыль (рентабельность) будет равна 110 336,7 тенге.

5. Безопасность жизнедеятельности

5.1 Анализ условий труда

В данной дипломной работе я разрабатываю аппаратный метод шифрования. Для её разработки требуется небольшое закрытое помещение, для 5 человек, сотрудник службы безопасности и несколько разработчиков имеющие навыки в области шифрования, криптографии и знания ЭВМ. В помещении есть 5 рабочих мест со стационарными компьютерами модели Asus ROG которые работают достаточно тихо и не вызывают шума, и так же аппаратными устройствами шифрования. Помещение площадью 280 м2 длиной 10 метров шириной 7 метров и высотой 4 метра очень хорошо освещаемо благодаря источнику освещения люминесцентных ламп, мощностью 50 Вт/м2 и остеклению площадью 24м2. Но в помещении полностью отсутствует система кондиционирования для благоприятной работы сотрудников, в следствии в разделе БЖД задаюсь целью рассчитать системы кондиционирования и подобрать соответствующую модель по основным характеристикам.

Ход работы:

1) Определить внутренние и наружные нагрузки и произвести расчеты тепловых нагрузок.
2) Произвести расчеты количества воздуха для подачи в помещение.
3) По произведенным расчетам и значениям нужно подобрать модель кондиционера.
4) Привести характеристики выбранного кондиционера
5) Показать расположение кондиционера, рабочие места, и место подачи воздуха.

Таблица 5.1 – Исходные данные

Наименование	Данные
Город	Алматы;
Параметры помещения (Д х Ш х В), м	10 х 7 х 4;
Данные по оборудованию	кол-во 5 шт.;
Мощность Роб, кВт/ч	0,5;
КПД η	0,75;
Данные по ист. света	мощ. N ос. уст., Вт/$м^2$ = 50;
Вид ист. св.	люминиц. лампы;
Число сотрудников, из них	мужчины = 4, женщины = 1;
Окна	кол-во 4;
Площадь 1 окна, $м^2$	6;
Общая площадь остекления, $м^2$	24$м^2$;

Продолжение таблицы 5.1

Вид	остекление в один-х метал. переплет, загрязнение умеренное;
Расчетное время суток, ч.	10-11;
Температура в помещении, ^{0}C	летом 23, зимой 21;
Вид положения работы	умеренная работа

5.2 Расчёт тепловых нагрузок в помещении: внутренние и наружные

В помещениях различного назначения в основном бывают тепловые нагрузки, возникающие снаружи помещения (наружные); а также тепловые нагрузки, возникающие внутри помещения (внутренние).

Наружные тепловые нагрузки.

Данные нагрузки представлены следующими составляющими:

– тепловые поступления или тепловые потери в результате разности температур снаружи и внутри здания через стены, потолки, полы, окна и двери.

– летом температура снаружи объекта является выше чем внутри, поэтому происходит приток тепла снаружи в внутрь помещения; а зимой наоборот температура снаружи меньше;

– тепловые поступления от солнечного излучения через окна; эта нагрузка проявляется в форме ощущаемого тепла;

– тепловые поступления от инфильтрации.

Тепловые нагрузки бывают положительными в случаях когда она зависит от времени года и времени суток. Теплопоступления и теплопотери в результате разности температур определяются по формуле 5.1:

$$Q_{огр}=V_{пом}*X_0*(t_{Нрасч}-t_{Врасч}),\ Вт \qquad (5.1)$$

где $V_{пом.}$ – объем помещения, м^3 : $V_{пом}= 10\cdot17\cdot4=680$ м^3; $X_о$ – удельная тепловая характеристика, Вт/м^3 ^{0}C: $X_0=0.42$ Вт/м^3 ^{0}C;

$t_{Нрасч}$ – наружная температура (параметр А). Для холодного периода – средняя температура самого холодного месяца в 13 часов, для теплого периода – средней температуре самого жаркого месяца в 13 часов.

$t_{Врасч}$ – внутренняя температура, выбирается с учетом комфортных условий или технологических требований, предъявляемых к производственным процессам.

Для летнего времени года:

$$t_{Нрасч} = 31\ ^0C$$
$$t_{Врасч} = 24\ ^0C$$
$$Q_{огр.}= 680\cdot0,42\cdot7=1999,2\ Вт$$

Для зимнего времени года

$$t_{Нрасч} = -14\ ^0C$$

$$t_{Врасч} = 21\ ^0C$$

$$Q_{огр.} = 680 \cdot 0{,}42 \cdot 35 = 9996\ Вт$$

Избыточная теплота солнечного излучения в зависимости от типа стекла почти до 90% поглощается средой помещения, остальная часть отражается. Максимальная тепловая нагрузка достигается при максимальном уровне излучения, которое имеет прямую и рассеянную составляющие. Интенсивность излучения зависит от ширины местности, времени года и времени суток.

Теплопоступление от солнечного излучения через остекление определяется по формуле 5.2:

$$Q_р = (q^I F_0^{II} + q^{II} F_0^{II}) * \beta_{с.з.} \qquad (5.2)$$

где q^I, q^{II} – тепловые потоки от прямой и рассеянной солнечной радиации, Вт/м^2 ;

F^I_o, F^{II}_o – площади светового проема, облучаемые и необлучаемые прямой солнечной радиацией, м^3;

βс. з. – коэффициент теплопропускания. βс. з. = 0.15

При отсутствии наружных затеняющих козырьков, ребер и т. д. для периода облучения остекления солнцем, когда его лучи проникают через окно в помещение $F^I_o = F_o$; $F^{II}_o = 0$, (1.3):

$$Q_р = q^I F_0 * \beta_{с.з.} = (q_{вп} + q_{вр}) * K_1^c * K_2 * \beta_{с.з.} * n * S_0, \text{ Вт} \qquad (5.3)$$

где $Q_{вп}$; $q_{вр}$ – тепловые потоки от прямой рассеянной радиации, Вт/м^2. По таблице 5 [1] для широты в 440 СШ до полудня в 11-12 ч. при расположении З:

$$Q_{вп} = 69\ Вт/м^2\ ;\ q_{вр} = 74\ Вт/м^2\ ;$$

$F_o = nS_o = 4 \cdot 6 = 24$ м^2 – площадь светового проема (n – число окон; S_o – площадь 1 окна);

K_1 – коэффициент затемнения остекления переплетами (K^C_1 – для облученных проемов). По таблице 6 [1]:

$K^C_1 = 0.72$;

K_2 – коэффициент загрязнения остекления. По таблице 7 [1]:

$K_2 = 0.9$.

Тогда:

$Q_р = (69+74)*0{,}72*0{,}9*0{,}15*4{,}5 = 62{,}54$ Вт.

По таблице 5 [1] для широты в 440 СШ до полудня в 11-12 ч. при расположении В:

$Q_{вп} = 211$ Вт/м^2; $q_{вр} = 89$ Вт/м^2 ;

$F_o = nS_o = 4 \cdot 72 = 24$ м^2 – площадь светового проема (n – число окон; S_o – площадь 1 окна);

Тогда:

$Q_р = (211+89)*0{,}72*0{,}9*0{,}15*4{,}5 = 131{,}22$ Вт.

Тогда общее теплопоступление солнечного излучения с обеих окон равно:

$Q_р$=62,54+131,22=193,76 Вт.

Внутренние тепловые нагрузки.

Внутренние нагрузки в жилых, офисных или относящихся к сфере обслуживания помещениях слагаются в основном из тепла:

– выделяемые людьми;

– выделяемые лампами и осветительными, электробытовыми приборами;

– выделяемые компьютерами, печатающими устройствами фотокопировальными машинами пр.

Другим источником тепла в помещениях различного назначения могут быть: отапливаемое производственное оборудование, горячие материалы, в том числе жидкости и различные полуфабрикаты, продукты сгорания и химических реакций.

Поступление тепла от людей зависит от интенсивности выполняемой работы и параметров окружающего воздуха. Тепло, выделяемое человеком, состоит из осязаемого, то есть передаваемого в воздух конвекцией и излучением, и скрытого тепла, расходуемого на испарение влаги с поверхности кожи и из легких.

Летом в 24 ^{0}С один человек выделяет тепло 61 Вт-102, Вт. Женщина выделяет 85% от нормы тепловыделений взрослого мужчины. Тогда выделение видимого тепла в помещении будет:

$Q_л^я$ = 61*4+61*1*0,85=295,85 Вт.

А выделение общего тепла:

$Q_л^о$ = 102*4+102*1*0,85=494,7 Вт.

По таблице 8 зимой при 20 ^{0} С один мужчина выделяет явного тепла 82 Вт, а общего – 103 Вт. Женщина выделяет 85% от нормы тепловыделений взрослого мужчины. Тогда выделение явного тепла в помещении составит:

$Q_л^я$ = 82*4+82*1*0,85=397,7 Вт.

А выделение общего тепла:

$Q_л^о$=103*4+103*1*0,85=499,55 Вт.

Теплопоступление от осветительных приборов, оргтехники и оборудования рассчитывается следующим образом. Теплопоступление от ламп определяется по формуле (5.4):

$$Q_{осв} = \eta * N_{осв} * F_{пол} \quad (5.4)$$

где η – коэффициент перехода электрической энергии в тепловую (для люминесцентных ламп η=0.5-0.6);

$N_{осв}$ – установленная мощность ламп (N=50 Вт/м^2);

$F_{пол}$ – площадь пола:

$F_{пол}$=17·11=70

Тогда:

$Q_{осв}$=0,5·50·70= 1750 Вт

Тепло, выделяемое производственным оборудованием, определяется по формуле (5.5):

$$Q_{об} = N_{уст} * K \quad (5.5)$$

$Q_{об}$=1,8*10^3*5*0,95=8550 Вт.

Теплопритоки, возникающие за счет находящейся оргтехники, – это 30% мощности оборудования:

$Q_{орг}$ = 1,8*10^3*5*0,3=2700 Вт.

5.3 Расчёт количества воздуха, необходимое для подачи в помещение

На основании выполненных расчетов составим баланс теплопоступлений в помещении:

Лето: $Q_{изб}$=295,85+686,25+1750+8550+2700+1999,2 = 15981, Вт

Зима: $Q_{изб}$=295,85+686,25+1750+8550+2700 +9986 =23968, Вт

Так как тепловой баланс для лета больше зимнего теплового баланса, то рассчитаем тепло напряжённость воздуха по формуле:

$$Q_н = \frac{Q_{изб.лето} \times 860}{V_{пом}}$$

$Q_н$=$\frac{23968 \cdot 860}{680} = 30312{,}4$ ккал/м3

При $Q_н$>20ккал/м3 , Δt= 8 °С.

Определение количества воздуха, необходимое для поступления в помещение:

$$L = \frac{Q_{изб} \times 860}{C \times \Delta t \times \gamma}$$

L= $\frac{23968 \cdot 860}{0{,}24 \cdot 8 \cdot 1{,}206 \cdot 10^4} = 890{,}8$ м3/час, где

С=0,24 ккал/(кг°С) – теплоемкость воздуха,

γ=1,206 кг/$м^3$ – удельная масса приточного воздуха.

Определение кратности воздухообмена:

N=$\frac{593{,}5}{680} = 1.31$ $час^{-1}$

5.4 По найденному значению количества воздуха подбираем соответствующую модель кондиционера

На основании полученных данных выберите тип стены сплит-системы кондиционирования. Основные технические характеристики кондиционера приведены в таблице 5.2 ниже.

Таблица 5.2 – Основные технические характеристики настенного кондиционера серии TOSHIBA RAV-SM1102CT-E

Эл. питание В/Гн	Произв. по холоду, кВт	Потр. эл мощн, кВт	Потреблток, А	Произв. по теплу, кВт	Размер (внешн. блок) мм	Расход воздуха, м3 /ч	Размер (внутр. блок) мм
220/50/1	14,07	7300	10,2	7330	L 845 H 700 B 320	8000	L 735 H 620 B 310

5.6 Приводим схему расположения кондиционера в помещении и схему подачи воздуха

Внешний блок представляет собой устройство, состоящее из компрессора, вентилятор и конденсатор. Он установлен таким образом, что высокая температура в конденсаторе может продуваться холодным воздухом. Обычно его можно установить на стене здания, на крышах или чердаках.

Внутренний блок обеспечивает охлаждение или нагрев воздуха в помещении, тем самым обеспечивая необходимый комфорт в рабочей среде. Во внутренних блоках обычно задают заданную температуру, которая распределяется равномерно и работает практически бесшумно (уровень шума 37-41дБ)

Кондиционер управляется специальным пультом дистанционного управления, который имеет такие функции, как: отопление, охлаждение, сушка, вентиляция, ночной режим; установка точной температуры, которая будет поддерживаться в будущем; несколько типов режима кондиционера: установите таймер, который включит или выключит кондиционер в данный момент времени; автоматически отрегулируйте положение направляющих жалюзи и тем самым измените направление воздушного потока.

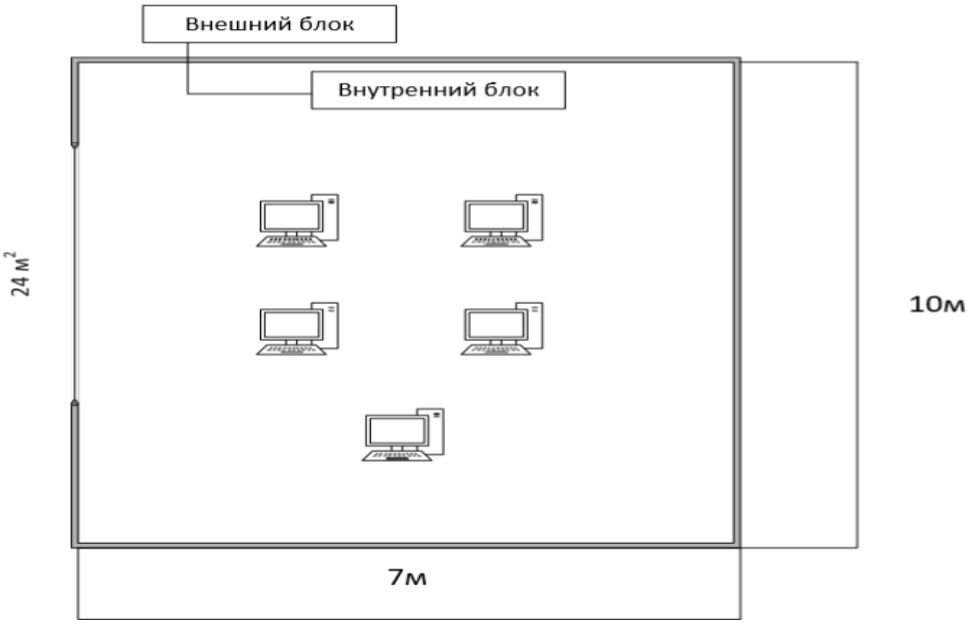

Рисунок 5.1 – Схема расположения кондиционера в производственном помещении

Вывод

В данной главе были рассчитаны тепловые нагрузки в помещении. По данным из расчетов были так же рассчитаны наружные и внутренние нагрузки. Основываясь на эти расчеты был выбран кондиционер, который будет обеспечивать комфортные условия труда, приведены его характеристики. Была выбрана одна модель кондиционера, учитывая нагрузки и площадь помещения.

Обеспечение воздушного комфорта в жилых и производственных помещениях зависит от систем аспирации, вентиляции, отопления и кондиционирования воздуха.

Задача кондиционирования воздуха состоит в выполнении вентиляции и отопления, а также в поддержании таких параметров воздушной среды, при которых каждый человек благодаря своей индивидуальной системе автоматической терморегуляции организма чувствовал бы себя комфортно, не замечая влияния этой среды. В итоге был произведен расчет кондиционирования помещения и выбора типа и вида кондиционера для данного помещения.

В данной главе были рассчитаны тепловые нагрузки в помещении. По данным из расчетов были так же рассчитаны наружные и внутренние нагрузки. Основываясь на эти расчеты был выбран кондиционер, который будет обеспечивать комфортные условия труда, приведены его характеристики. Была выбрана одна модель кондиционера, учитывая нагрузки и площадь помещения.

Список литературы

1 Цилькер Б.Я., Орлов С.А. Организация ЭВМ и систем. 2-е изд. -Спб.: Питер, 2011.

2 Е.Ж. Айтхожаева, С.Т. Тынымбаев, Аспекты аппаратного приведения по модулю в асимметричной криптографии: Вестник НАНРК №5, 2014. Алматы. с 88-93

3 Рябко Б.Я, Фионов А.И. Основы современной криптографии для специалистов в информационных технологиях. - М.: Научный мир, 2014.

4 Аманжолова К. Б., Алибаева С. А. Экономика предприятии телекоммуникации: Учебное пособие. - Алматы: АИЭС, 2003

5 Ростовцев А.Г., Маховенко Е.Б. Теоретическая криптография. – СПб.: Профессионал, 2015.

6 Протокол Диффи-Хеллмана, URL: https://www.securitylab.ru/analytics/ 478912.php (дата обращения 15.03.2019г.)

7 Введение в криптографию, URL:http://citforum.ru/ security/cryptography/ crypto1.shtml (дата обращения 28.03.2019г.)

8 Белов С.В. Безопасность жизнедеятельности. -М.: Высшая школа 2014.

9 Абдимуратов Ж.С., Мананбаева С.Е. Безопасность жизнедеятельности. Методические указания к выполнению раздела «Расчет производственного освещения» в выпускных работах для всех специальностей. -Алматы: АИЭС, 2009

10 Дюсебаев М.К. Безопасность жизнедеятельности: методические указания к выполнению раздела дипломных проектов. -Алматы: АИЭС, 2003

11 Комплексная оценка эффективности мероприятий, направленных на ускорение научно-технического прогресса. Методические рекомендации и комментарии по их применению. -М.: Издательство Москва, 2003

12 Шепеленко Г.И. Экономика, организация и планирование производства на предприятии. Учебное пособие. -Ростов-на-дону: «МАРТ», 2004.

Приложение А

```
module mod2(
output [N/2+1:0] R,
input [N-1:0] A,
input signed [N/2+2:0] P3,
input signed [N/2+2:0] P2,
input signed [N/2+2:0] P1,
input clk
);
Parameter N=8;
Integer i=0;
Reg [N+1:0] temp;
reg signed [N/2+2:0] addr1;
reg signed [N/2+2:0] addr2;
reg signed [N/2+2:0] addr3;
reg [N/2+1:0]ri;
reg [N+1:0] memory [0:1];
always@(posedge clk)
begin
if(i<N/4)
begin
memory[i] = A;
temp = memory[0];
temp = temp <<2;
addr3 = temp[N+1:N/2] + P3;
addr2 = temp[N+1:N/2] + P2;
addr1 = temp[N+1:N/2] + P1;
outAddr3 = addr3[N/2+1:0] & {(N/2+2) { (addr3[N/2+2])}};
outAddr2 = (addr2[N/2+1:0] & {(N/2+2) { (addr3[N/2+2])}} & {(N/2+2){
(addr2[N/2+2])}};
outAddr1 = (addr1[N/2+1:0] & {(N/2+2) { (addr2[N/2+2])}} & {(N/2+2){
(addr1[N/2+2])}};
ri = outAddr3 | outAddr2 | outAddr1;
temp [N+1:N/2] = (ri == 10'b0) ? temp[N+1:N/2]: ri;
memory[0] = temp;
i=i+1;
end
end
assign R=ri;
endmodule
```