Әл-Фараби атындағы Қазақ ұлттық университеті

ӘОЖ 004.4’233 Қолжазба құқығында

ТЕМИРБЕКОВА ЖАНЕРКЕ ЕРЛАНОВНА

"Есептеу кластерін және жүйенің қауіпсіздігін қамтамасыз ету үшін AtmelAVR микроконтроллерін қолдану"

6D070400- Есептеу техникасы және бағдарламалық қамтамасыз ету

Философия докторы (PhD)
дәрежесін алу үшін дайындалған диссертация

Отандық ғылыми жетекші
Физика математика ғылымдарының кандидаты, профессор
Пыркова Анна Юрьевна
Әл-Фараби атындағы Қазақ ұлттық университеті,
Алматы, Қазақстан

Шетелдік ғылыми жетекші:
PhD, профессор, Manuel Martins Barata
ISEL техникалық университеті, Лиссабон, Португалия

Қазақстан Республикасы
Алматы, 2023

МАЗМҰНЫ

НОРМАТИВТІК СІЛТЕМЕЛЕР 4

БЕЛГІЛЕУЛЕР МЕН ҚЫСҚАРТУЛАР 5

КІРІСПЕ 6

1 ІОТ ҚҰРЫЛҒЫЛАР КЛАСТЕРІНДЕГІ ДЕРЕКТЕРДІ ҚОРҒАУҒА АРНАЛҒАН ӘДІСТЕР МЕН ҚҰРЫЛҒЫЛАРҒА ТАЛДАУ ЖАСАУ 6

1.1 AtmelAVR тобындағы микроконтроллерді таңдау және негіздеу 13

1.2 Әртүрлі AtmelAVR микроконтроллердің сенімділігін есептеу 28

1.3 IoT құрылғылар кластеріндегі деректерді қорғауға арналған әдістер 30

1.3.1 Гомоморфты шифрлау криптожүйесі 30

1.3.2 Гомоморфты шифрлау криптожүйелерін AtmelAVR микроконтроллерінде зерттеу 33

2 ATMELAVR МИКРОКОНТРОЛЛЕРІНДЕ ҚОЛДАНЫЛАТЫН ГОМОМОРФТЫ ШИФРЛАУ АЛГОРИТМІН ЖЕТІЛДІРУ 36

2.1 Айнымалысы бар көпмүшелер сақинасындағы гомоморфты шифрлау 36

2.2 Рационал коэффициенттері бар көпмүшелер өрісіндегі гомоморфты шифрлау 45

2.3 Пэйе криптожүйесі 49

2.4 Толық гомоморфты шифрлау принципі 51

3 IoT ҚҰРЫЛҒЫЛАР КЛАСТЕРІНІҢ ҚАУІПСІЗДІГІН ҚАМТАМАСЫЗ ЕТУ ҮШІН ATMELAVR МИКРОКОНТРОЛЛЕРІНДЕ КІТАПХАНА АРХИТЕКТУРАСЫН ҚҰРУ 53

3.1 Жетілдірілген айнымалысы бар көпмүшелер сақинасындағы толық гомоморфты шифрлауға кітапхана құру 53

3.2 Жетілдірілген рационал коэффициенттері бар көпмүшелер өрісіндегі гомоморфизмге кітапхана құру 58

3.3 Пэйе гомоморфты шифрлау кітапханасы 60

3.4 Микроконтроллер мен кітапхана байланысы 63

3.5 Жетілдірілген гомоморфты шифрлау алгоритмдеріне жасақтама модуль құру 67

4. ATMELAVR МИКРОКОНТРОЛЛЕРІНДЕГІ КІТАПХАНА ЖҰМЫСЫНЫҢ САПАСЫН БАҒАЛАУ ЖӘНЕ СОНДАЙ-АҚ БАСҚА КІТАПХАНА ЖҰМЫСЫМЕН САЛЫСТЫРУ 74

4.1 Құрылған кітапханалардың өнімділігін AtmelAVR (Atmega 328) микроконтроллерде тестілеу 74

4.2 Құрылған кітапханаларды басқа авторлардың жұмыстарымен салыстырулар жүргізу 80

ҚОРЫТЫНДЫ 84

ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ 85

ҚОСЫМША А – Жарияланымдар тізімі 92

ҚОСЫМША Ә – Авторлық куәлік 94

НОРМАТИВТІК СІЛТЕМЕЛЕР

Берілген диссертацияда келесі стандарттарға сәйкес сілтемелер қолданылды:

1. «Диссертацияны безендіру нұсқаулығы», Қазақстан Республикасы БҒМ ЖАК 28 қыркүйек, 2004 жыл. №377-3 ж.

2. ГОСТ 7.32-2001. Ғылыми зерттеу жұмысының есебі. Безендіру ережелері мен құрылымы.

3. ГОСТ 7.1-2003. Библиографиялық таспа. Библиографиялық сипаттау.

ҚЫСҚАРТУЛАР МЕН БЕЛГІЛЕУЛЕР

ARM – Advanced RISC Machine- жетілдірілген RISC машинасы
BGV – Zvika Brakerski, Craig Gentry and Vinod Vaikuntanathan фамилиялардың қысқартуы
BLE – Bluetooth Low Energy –төмен энергия тұтыну блютусы
CISC – Complex instruction set computer- күрделі нұсқаулар жиынтығы бар компьютер
CPU – Central processing unit-орталық процессор
CS – Chip Select- чипті таңдау
EEPROM – Electrically Erasable Programmable Read-Only Memory- электрлік өшірілетін қайта бағдарламаланатын тұрақты сақтау құрылғысы
FHE – Fully homomorphic encryption-толық гомоморфты шифрлау
GHS – Global Harmonized System – ғаламдық үйлестірілген жүйе
I²C – Inter-Integrated Circuit- интерваларалық сызба
IDE – Integrated Development Environment-интеграцияланған даму ортасы
IoT – Internet of things – интернет заттары
ISP - In-System Programming – ішкі сұлбалық бағдарламалау
MISO – Master In Slave Out
MOSI —Master Out Slave In
PIC – peripheral interface controller - перифериялық интерфейс контроллері
RAM – Random Access Memory-жедел жады
RISC – Reduced instruction set computer- қысқартылған командалар жиынтығы бар компьютер
RSA – Rivest, Shamir, Adleman фамилиялардың қысқартуы, ашық кілтті криптографиялық алгоритм
SCK – Serial Clock- сериялық сағат
SPI – Serial Peripheral Interface – тізбектелген перифериялы интерфейс
TFHE – Totall fast fully homomorphic encryption – толық жәлдам гомоморфты шифрлау
VCC – Voltage Collector to Collector-коллекторға коллектор қысымы
АСТ – аналогты-сандық түрлендіргіш
ТГШ – толық гомоморфты шифрлау

КІРІСПЕ

Зерттеу тақырыбының өзектілігі IoT (Internet of Things, Интернет заттары) құрылғылары денсаулық сақтау саласынан бастап, өндіріске дейін барлық секторларда кең қолданысқа енген. IoT технологиялардың негізгі мақсаты – интернетке қосылған құрылғылардың бір-бірімен байланысуына, деректер алмасуына, деректерді сақтауына және пайдаланушының талаптарына сәйкес есептеулер жүргізуіне мүмкіндік беру.

IoT нұсқасының бірі медициналық заттардың интернеті (IoMT) денсаулық сақтау саласының өсіп келе жатқан саласы және қашықтан бақылау, деректерді жинау және талдау үшін қосылған медициналық құрылғыларды пайдалануды қамтитын IoT мамандандырылған қолдануының бірі болып табылады. Atmel AVR микроконтроллерлері IoMT құрылғыларында энергия қуатын аз тұтынуына, жоғары өнімділікке және сенімділікке байланысты кеңінен қолданылады. Atmel AVR микроконтроллерлерін пайдаланатын IoMT құрылғыларының мысалдары:

1. Тағатын денсаулық мониторлары;
2. Ақылды инсулин қаламдары;
3. Мөлшерленген доза ингаляторы;
4. Науқастарды қашықтықтан бақылау жүйелері;
5. Таблеткалары бар ақылды бөтелкелер;
6. Телемедицина құрылғылары.

Қорыта келе, Atmel AVR микроконтроллері төмен қуат тұтынуына, шағын өлшемдеріне және жоғары есептеу мүмкіндігіне байланысты IoMT пайдалану үшін жан-жақты және сенімді таңдау болып табылады.

IoT құрылғылары 2024 жылға қарай 83 миллиардқа жетеді деп болжанғанымен, бұл құрылғылардың қауіпсіздігі басты алаңдаушылықты тудырады, тиісті қауіпсіздік шараларын қолданбаған жағдайда, кез келген IoT қосылған құрылғының жұмыс жасау мүмкіндігін, сонымен қатар пайдаланушы деректерін ұрлауға қауіпі бар. Palo Alto Networks 2022 жылғы есептемесі бойынша, IoT құрылғысының барлық трафигінің 98%-ы шифрланбаған, бұл желідегі жеке және құпия деректерді құпия түрде сақталмағанын көрсетеді және шабуылдаушыларға шифрланбаған желілік трафикті тыңдауға, жеке немесе құпия ақпаратты жинауға, содан кейін бұл деректерді шабуылшы өзінің жеке мақсаттарына пайдалануына мүмкіндік береді. SAM Seamless Network мәліметтері бойынша, 2021 жылы 1,5 миллиардтан астам IoT құрылғыларына шабуыл жасалынды, соның ішінде 900 миллионға жуық фишингтік шабуылдар.

IoT құрылғылары өздігінен компьютерлердің аналогы емес, олар басынан аяғына дейін ресурстарды қажет ететін кез келген тапсырманы орындай алмайды, олар оның кейбір бөлігін ғана орындайды, ал қалған бөліктерін басқа IoT құрылғылары аяқтайды. IoT белгілі бір топта немесе кластерде жұмыс істейді, олар қандай да бір мәселені бірлесіп шешеді. IoT құрылғылар кластерінің арасында жіберілетін ақпаратты қорғау үшін, олар шифрланған болуы керек және біз жұмыстың жалпы нәтижесін бұзбауымыз үшін

шифрланған деректермен шифрланбаған күйде сияқты операцияларды орындай алуымыз керек. Бұл мүмкіндікті біз медицинадағы, тұрмыстық электроникадағы және өндірістегі IoT құрылғыларын басқаратын AtmelAVR микроконтроллерлерінде (DFRobot Beetle BLUE, Atmega 328, Atmega 32u4, Atmega 2560) іске асырылуы мүмкін гомоморфты шифрлау арқылы жасай аламыз.

Соңғы жылдары әлемде IoT құрылғыларына арналған ТГШ (толық гомоморфты шифрлау) бойынша көптеген жұмыстар пайда болды. Суджой С.Р., Гоюри П., Дипика Н. (2021) жұмыстарында толық гомоморфты шифрлау алгоритмдерін IoT қосымшалары мен құрылғыларына қолдануға болатынын көрсетеді, сонымен қатар деректердің құпиялылығын сақтай отырып, есептеу жылдамдығын жоғарлатуды қамтамасыз етуге бағытталған.

Горан Д., Милан М., Павле В. (2021) «IoT құрылғысында гомоморфты шифрлауды енгізуді бағалау» жұмыстарында BFV және BGV гомоморфты шифрлау механизмдерінің ерекшеліктерін бағалап, есептеу өнімділігін өлшеді. Raspberry Pi 4 B моделіне негізделген IoT платформасындағы шифрлау схемаларын бағалап, гомоморфты шифрлау операцияларын ендірілген құрылғыларда қолдануға болатындығын көрсетеді және ең алдымен құпиялылықты жақсартуға және қосымшаларды жылдамдату үшін жоғары өткізу қабілеті мен төмен кідіріспен қамтамасыз етуге бағытталған.

Ресейлік ғылыми қоғамдастық өкілдерінің ішінде келесі ғалымдардың еңбектерін ерекше атап өтуге болады: И.Б.Саенко, В.А. Десницкий (Москва қ.), И.В.Котенко (Сверлоск қ.), П.Д. Зегжда (Санкт-Петербург).

ҚР БҒМ ҒК Ақпараттық және есептеуіш технологиялар институтының ғалымдары: Бияшев Р.Г., Нысанбаева С.Е., Капалова Н.А., Күнболат А. ғалымдардың еңбектерінде деректерді криптографиялық қорғау құралдарын зерттеген.

Жасалған талдауларды ескере отырып, IoT қосымшалары мен құрылғыларының қауіпсіздігін тиімді қамтамасыз ететін әдістер, алгоритмдер қажеттілігі туындайды және қарастырылып отырған тақырыптың **өзекті мәселе** екендігін анықтайды.

Диссертациялық жұмыстың мақсаты. IoT құрылғалар арасында деректердің қауіпсіз сақталуын және алмасуын қамтамасыз ету үшін әртүрлі AtmelAVR микроконтроллерде шифрланған деректерге барлық арифметикалық операцияларды орындауға мүмкіндік беретін толық гомоморфты шифрлау кітапханаларының архитектурасын құру және іске асыру.

Диссертациялық жұмыстың мақсатын жүзеге асыратын **зерттеу міндеттері:**

1. IoT құрылғылар кластеріндегі деректерді қорғауға арналған әдістер мен құрылғыларға талдау жасау;

2. AtmelAVR микроконтроллерінде қолданылатын гомоморфты шифрлау алгоритмін жетілдіру;

3. IoT құрылғылар кластерінің қауіпсіздігін қамтамасыз ету үшін AtmelAVR (DFRobot Beetle BLUE, Atmega 328, Atmega 32u4, Atmega 2560, ESP

32) микроконтроллерінде кітапхана архитектурасын құру, бағдарламалық жасақтама құру;

4. AtmelAVR микроконтроллеріндегі кітапхана жұмысының сапасын бағалау және қолданыстағы белгілі кітапхана жұмысымен салыстыру мақсатында есептеу эксперименттерін жүргізу.

Зерттелу нысаны. IoT құрылғылар арасында деректерді қауіпсіз жіберу.

Зерттеу пәні. Микроконтроллердің көмегімен деректерді қорғау жолдары.

Зерттеу әдістері. Микроконтроллердегі ақпараттарды өңдеу әдісі, IoT кластерлерін қорғау үшін микроконтроллерлерді пайдалану тиімділігін талдау және бағалау әдістері, гомоморфты шифрлау әдісі.

Зерттеу жұмысының жаңалығы. IoMT құрылғылар тобында деректердің қауіпсіздігін қамтамасыз ету үшін және жіберілетін ақпараттық құпиялылығын бұзбай, осы деректерді өңдеу мақсатында IoMT құрылғылардың бірлесіп жұмыс жасауын қамтамасыз ету үшін алғаш рет AtmelAVR (DFRobot Beetle BLUE, Atmega 328, Atmega 32u4, Atmega 2560, ESP 32) микроконтроллерінде гомоморфты шифрлау алгоритмдердің кітапхана архитектурасы құрылды және іске асырылды. Құрылған кітапхана мен белгілі Кренделев С.Ф. және Абрамов А. кітапханаларының өнімділігін бағалау бойынша эксперименттер барысында диссертациялық жұмыста ұсынылған кітапхана қосылу мен пайдаланудың қарапайымдылығын көрсетті, сонымен қатар деректерді есептеу жылдамдығы шамамен 1,52 есе жоғары болды.

Жұмыстың теориялық маңыздылығы.

Бүтін сандармен жұмыс істеуге және оларға барлық арифметикалық амалдарды орындауға мүмкіндік беретін толық гомоморфты шифрлау алгоритмдерін микроконтроллерлер мен IoMT құрылғыларында деректерді өңдеу процестерін жетілдіру және бейімдеу.

Жұмыстың практикалық маңыздылығы.

Микроконтроллерге арналған кітапхананың архитектурасын әзірлеу және оның жұмысын оңтайландыру мақсатында кітапхананың модульдері мен әдістері арасында мәліметтер алмасу схемасын, әдістемесін және тәртібін белгілеу.

Қорғауға шығарылатын негізгі тұжырым.

Зерттеу барысында әртүрлі деректер құрылымдарымен жұмыс істеу үшін SD картамен, SD модулмен және бағдарламашпен толықтырылған AtmelAVR микроконтроллерлер тобында әзірленген, IoMT құрылғылар жүйесінде деректерді қауіпсіз жіберу үшін гомоморфты шифрлау алгоритмдерінің архитектурасы қорғауға ұсынылады.

Сенімділік дәрежесі мен апробациялау нәтижелері. Жұмыстың ғылыми нәтижелері төмендегі халықаралық ғылыми-әдістемелік конференциялар мен ғылыми семинарларда баяндалып, талқыланды:

1) «Көліктегі инновациялық технологиялар: білім, ғылым, тәжірибе» атты XLI, XLII Халықаралық ғылыми-практикалық конференциясы;

2) Международная научно-практическая конференция «Актуальные и перспективные направления развития научно-технологического прогресса»;

4) International Research Conference on Technology, Science, Engineering and Economy held Seattle, USA;

5) «Физика – математика ғылымдарының қазіргі білім беру кеңістігіндегі рөлі» VI халықаралық ғылыми-практикалық конференциясы;

6) The 5th International Conference on Energy, Environmental and Information System (ICENIS 2020), Semarang, Indonesia //E3S Web of Conferences.

Сондай-ақ бұл тақырып бірнеше рет әл-Фараби атындағы Қазақ ұлттық университетінің Ақпараттық технологиялар факультетінің «Компьютерлік ғылымдар» кафедрасында, ҚР БҒМ ҒК Ақпараттық және есептеуіш технологиялар институтының семинарларында талқыланды.

Әрбір басылымды дайындаудағы докторанттың үлесі. Жарияланған мақалалар мен ғылыми еңбектер диссертация тақырыбы бойынша зерттеу нәтижелерін сипаттайды. Ғылыми зерттеу жұмыстарын орындау барысында 12 мақала жарияланды және 1 авторлық куәлік алынды, оның ішінде, Scopus деректер базасында индекстелетін журналдарда 2 ғылыми мақала жарыққа шықты:

1. Pyrkova A.Yu., Temirbekova Zh.E. “Compare encryption performance across devices to ensure the security of the IoT”, Indonesian Journal of Electrical Engineering and Computer Science, -2020. -Vol. 20. -No. 2. – P. 894-902. (Scopus базасы бойынша процентилі - 45). Q3

2. Temirbekova Zh.E., Pyrkova A.Yu. “Improving teachers’ skills to integrate the microcontroller technology in computer engineering education”, Education and information technology, -2022 doi: 10.1007/s10639-021-10875-8 (Scopus базасы бойынша процентилі - 95). Q1

Қазақстан Республикасы Ғылым және жоғары білім министрлігінің Ғылым және жоғары білім саласындағы сапаны қамтамасыз ету комитеті мен ұсынылған басылымда жарияланған мақалалар (3):

1. Temirbekova Zh.E., B.K. Alymbayeva. “Using Atmel AVR microcontrollers for safety-performance computing” // Вестник КазНИТУ, -2017. №2, – С. 192 - 195

2. Pyrkova A.Yu., Temirbekova Zh.E. “Possibilities of using a BLE Nano Kit microcontroller to develop cryptographic libraries” // Вестник КазНИТУ:, - 2018. №2, – С. 477 - 481

3. Pyrkova A.Yu., Temirbekova Zh.E. “Performing symmetric encryption mbed platform” // Вестник КазНИТУ, - 2018. №2, – С. 473 – 476

Scopus деректер базасында индекстелген халықаралық ғылыми-тәжірибелік конференциялар жинақтарында 2 ғылыми мақала жарияланған:

1. Temirbekova Zh.E., Pyrkova A.Yu. “Using FHE in a binary ring Encryption and Decryption with BLE Nano kit microcontroller” //E3S Web of Conferences 202 (ICENIS 2020), -2020. 15002

2. Temirbekova Zh.E., Pyrkova A.Yu., Abdiakhmetova Zh. “Library of fully homomorphic encryption on a microcontroller” //2022 International Conference on Smart Information Systems and Technologies 28-30 April, 2022, Nur-Sultan, doi:10.1109/SIST54437.2022.9945722.

Халықаралық ғылыми конференциялар жинақтарында 5 ғылыми мақала жарияланды:

1. Temirbekova Zh.E “Programming microcontroller AVR Atmega8” // «Көліктегі инновациялық технологиялар: білім, ғылым, тәжірибе» атты XLI Халықаралық ғылыми-практикалық конференцияның материалдары, 3-4 сәуір 2017 ж., Алматы, Казахстан, (I том), 102-104 б.

2. Pyrkova A.Yu, Temirbekova Zh.E “Use homomorphic encryption for data security” // «Көліктегі инновациялық технологиялар: білі, ғылым, тәжірибе» атты XLII Халықаралық ғылыми-практикалық конференцияның материалдары, 2018 ж., Алматы, Казахстан, (I том), 83 -85 б.

3. Temirbekova Zh.E “For data security symmetric encryption algorithm” // Международная научно-практическая конференция «Актуальные и перспективные направления развития научно-технологического прогресса», 30 января, 2020 года, Россия, г. Кемерово, С. 26-30

4. Pyrkova A.Yu, Temirbekova Zh.E. “Using microcontrollers to ensure data security”, International Research Conference on Technology, Science, Engineering and Economy held Seattle, USA, February 28th, 2020, P. 52-60

5. Темирбекова Ж.Е. “Толық гомоморфты шифрлеу алгоритмінің кітапханасы” «Физика – математика ғылымдарының қазіргі білім беру кеңістігіндегі рөлі» VI халықаралық ғылыми-практикалық конференция материалдар жинағы, 7 желтоқсан 2021 ж., Атырау, Казахстан, 326-331 б.

Жұмыс көлемі мен құрылымы. Диссертациялық жұмыс кіріспеден, төрт бөлімнен, қорытындыдан, 104 пайдалынылған әдебиеттер тізімінен және екі қосымшадан тұрады. Зерттеу жұмысының жалпы көлемі – 94 бет оның ішіне 46 сурет, 17 кесте кіреді.

Кіріспеде диссертациялық жұмыстың өзектілігін негіздейді. Жұмыстың мақсаты, зерттеу жұмысының объектісі мен пәні тұжырымдалды. Ғылыми жаңалығы мен практикалық маңыздылығы анықталды. Жүргізілген зерттеу нәтижелері сипатталған. Зерттеу және жариялау нәтижелерін апробациялау туралы ақпарат берілген.

Бірінші бөлімде әртүрлі AtmelAVR микроконтроллердің архитектурасы қарастырылып, шолу берілген. Диссертациялық жұмысқа қатысты қолданылған терминдер мен ұғымдар берілген. Әртүрлі AtmelAVR микроконтроллердің сенімділігі есептеліп, соның негізінде диссертациялық жұмыста қолданылатын микроконтроллер анықталды. AtmelAVR микроконтроллерінде әртүрлі гомоморфты шифрлау криптожүйелеріне эксперименттік есептеулер жүргізілді. Микроконтроллерде эксперименттік есептеулер негізінде тиімді гомоморфты шифрлау алгоритмдері көрсетілген. Осы тақырып бойынша ғылыми еңбектерге сілтеме жасалып, шолу жүргізілді.

Екінші бөлімде гомоморфты шифрлаудың жетілдірілген әдістері: С.Ф. Кренделев алгоритміне азайту және бөлу операциялары қосылды, А.Абрамовтың алгоритміне – азайту операциясы қосылды. Жетілдірілген толық гомоморфты шифрлау іс-әрекет диаграмма түрінде көрсетілді.

Үшінші бөлімде AtmelAVR микроконтроллері үшін жетілдірілген толық гомоморфты шифрлау кітапхана архитектурасы жүзеге асырылды. Құрылған кітапхана архитектурасы микроконтроллермен байланысу схемасы зерттелді. Atmega 328 микроконтроллерінде орнатылған кітапхана архитектурасы көрсетілді. Жетілдірілген шифрлау алгоритмі үшін құрылған бағдарламалық жасақтама модулі туралы ақпарат берілген. Жетілдірілген шифрлау алгоритмі блок-схемалар негізінде түсіндірілген (кілтті генерациялау, шифрлау, гомоморфтылық, кері шифрлау). Бағдарламалық жасақтаманың негізгі жүйелік талаптары, жұмыс істеу түсіндірмелері көрсетілді.

Төртінші бөлімде ұсынылған архитектура негізінде құрылған кітапхана AtmelAVR микроконтроллерінде есептеу жылдамдықтарына эксперименттік зерттеу жұмыстары жүргізілді. Нәтижелері диаграмма түрінде көрсетілді. Құрылған кітапханаларды басқа белгілі авторлардың жұмыстарымен салыстырулар жүргізіліп, диссертациялық жұмыста ұсынылған, яғни жетілдірілген толық гомоморфты шифрлау салыстырмалы түрде есептеу жылдамдығы шамамен 1,52 есе жылдам жұмыс жасайтыны көрсетілді.

Қорытындыда диссертацияда алынған негізгі нәтижелер тұжырымдалды.

Автор ғылыми кеңесшілерге: физика математика ғылымдарының кандидаты, профессор А.Ю. Пырковаға және шетелдік жетекші М.М. Баратаға тапсырма қойылымын анықтағаны және жұмысты жазудағы қолдауы үшін ризашылығын білдіреді.

1 ІОТ ҚҰРЫЛҒЫЛАР КЛАСТЕРІНДЕГІ ДЕРЕКТЕРДІ ҚОРҒАУҒА АРНАЛҒАН ӘДІСТЕР МЕН ҚҰРЫЛҒЫЛАРҒА ТАЛДАУ ЖАСАУ

Соңғы жылдары ақпараттық және коммуникациялық технологиялардың дамуы өндірістік процесстердің жұмысын айтарлықтай жақсартты. Осы саладағы заманауи тенденциялардың бірі коммуникациялық байланыс жүйелеріне, бұлтты есептеу технологияларына, деректерді өңдеу орталықтарына негізделген IoT жүйелері. IoT көптеген процесстерді автоматтандыруға мүмкіндік береді және жасанды интеллект алгоритмдерін, әртүрлі микроконтроллерлерді үйлестіре отырып өз бетінше шешім қабылдай алады.

Қазіргі таңдағы IoT қосымшалары мен құрылғылары денсаулық сақтау саласынан бастап, өндіріске дейін барлық секторларда кең қолданысқа енген. Juniper Research [1] зерттеуі бойынша IoT құрылғыларының жалпы саны: 2016 ж. – 23,2 млрд., 2018 ж. – 35,3 млрд., 2020 жылы 50,1 миллиардт, ал 2024 жылға қарай 83 миллиардқа жететінін көрсетті. Құрылғылардың жалпы саны 2024 жылға қарай 83 миллиардқа жетеді деп болжанғанымен, бұл құрылғылардың қауіпсіздігі басты алаңдаушылықты тудырады. Тиісті қауіпсіздік шараларын қолданбаған жағдайда, кез келген IoT қосылған құрылғының жұмыс жасау мүмкіндігін, сонымен қатар пайдаланушы деректерін урлауға қауіпі бар.

Микроконтроллерлер енгізілген жүйелерде және IoT құрылғыларында деректерді өңдеу, байланыс және басқару сияқты әртүрлі тапсырмалар үшін кеңінен қолданылады. Микроконтроллер көмегімен деректерді қорғау рұқсатсыз кіруді болдырмау және деректердің құпиялылығын, тұтастығын және қолжетімділігін қамтамасыз ету үшін бірге жұмыс істейтін аппараттық және бағдарламалық қамтамасыз ету қауіпсіздік шараларының комбинациясын жүзеге асыруды қамтиды. Микроконтроллер көмегімен деректерді қорғау үшін шифрлауды қолдануға болады. Деректерді қорғаудың сенімді әдістерінің бірі – гомоморфты шифрлау. Микроконтроллерді деректерді жадта сақтау немесе желі арқылы жіберу алдында шифрлау үшін бағдарламалауға болады. Шифрланған деректерді микроконтроллерде қауіпсіз сақтауға болатын арнайы кілт арқылы ғана шешуге болады.

Гомоморфты шифрлау - алдын ала шифрланған деректерді кері шифрламай, шифрланған түрде деректерді өңдеуге және талдауға мүмкіндік беретін жаңа криптографиялық әдіс. Бұл криптожүйе әсіресе деректердің құпиялылығы мен қауіпсіздігі маңызды болып табылатын IoT қолданбалары үшін пайдалы. Кәзіргі таңда, IoT құрылғылары мен косымшалары үшін гомоморфты шифрлау саласындағы ғалымдардың ғылыми еңбектерін атап айтсақ :

Звика Бракерски және т.б. [2] 2012 жылы толықтай гомоморфты шифрлау схемаларының есептеу күрделілігін айтарлықтай төмендететін жаңа толық гомоморфты шифрлау схемасын ұсынды. Бұл есептеу ресурстары шектеулі IoT қолданбаларында тәжірибелік тұрғыда қолдануға мүмкіндік берді.

Илиар Чилотти және т.б. [3] 2016 жылы мүмкіндігі шектеулі ресурстары бар құрылғыларда іске асырылуы мүмкін IoT құрылғылары үшін гомоморфты шифрлау схемасын ұсынды. Схема шамамен есептеулерге негізделген, бұл гомоморфты шифрлау есептеу уақытын төмендетті.

Хуэй Ван және т.б. [4] 2017 жылы IoT деректерінің құпиялылығы үшін әртүрлі пішімдері мен өлшемдері бар бірнеше деректер көздерін өңдей алатын гомоморфты шифрлау схемасын ұсынды. Схема шифрланған деректермен икемді және тиімді есептеуді қамтамасыз ететін шифрлау кілтін басқару тәсілін пайдаланады.

Xiaolei Dong және т.б. [5] 2020 жылы IoT құрылғылары үшін гомоморфты шифрлауға негізделген деректерді сақтау және іздеу схемасын ұсынды. Бұл схема деректердің құпиялылығы мен қауіпсіздігін сақтай отырып, шифрланған деректерді тиімді сақтауға және алуға мүмкіндік береді.

Бабенко Л.К., Толомоненка Е.А. [6] жұмыстарында IoT құрылғылар арасында деректерді қауіпсіз жіберу және олармен жұмыс жасау үшін, яғни жеңіл криптография шифрланған деректерді өңдеуге мүмкіндік бермегендіктен, гомоморфты шифрлау алгоритмін қолдану арқылы мәселені шешуге болатынын көрсетеді.

Тұтастай алғанда, авторлар IoT қосымшаларын қорғау және деректердің құпиялығын сақтау үшін гомоморфты шифрлау қолдану тиімді деген тұжырымға келді.

1.1 AtmelAVR тобындағы микроконтроллерді таңдау және негіздеу

Микроконтроллерлер – бұл әртүрлі электрондық құралдарды басқаруға арналған микросхема [7]. Классикалық дербес компьютерлердің белгілі микропроцессорларда жад пен перифериялық құрылғылар жоқ, осыған байланысты оларға белгілі бір шина жүйесі арқылы қосылған сыртқы құрылғылар қажет. Микроконтроллерлерде компьютерді қалыптастыру үшін қажет сызбаның бәрі бар болғандықтан, көп жағдайда оларды жұмыс істеуге мәжбүр етудің жалғыз қажеттілігі - бағдарламалық қамтамасыздандыру ету. Микроконтроллер нақты тапсырмаларды орындауға арналған, енгізу-шығару байланысы анықталған нақты қолдану құралдары. Кіріске байланысты кейбір өңдеуді орындау керек және шығыс жеткізіледі. Микроконтроллерлердің қосымшалары ерекше екенін ескере отырып, оларды оңтайландыруға болады, осынған байланысты олардың мөлшері мен шығындарын едәуір азайтылады.

Микроконтроллерге дербес компьютерлерден табуға болатын жылдам процессор қажет емес, сонымен қатар жадының көлемі әлдеқайда аз. Микроконтроллерлер көптеген ресурстарды қажет етпейтіндіктен, олардың қуат көзін тұтынуы өте аз болып келеді. Бұл микроконтроллерлерді батареяларды қолдану арқылы іске қосуға мүмкіндік береді, ал кейбір жағдайларда олар батареясыз бірнеше жыл бойы үздіксіз жұмыс істей алады.

Перифериялық құрылғылар микроконтроллерлердің маңызды компоненттері болып табылады. Перифериялық құрылғылар микроконтроллерге қоршаған ортамен өзара әрекеттесуге мүмкіндік береді

және олар процессорға кейбір функциялармен көмектесе алады. Қазіргі заманғы микроконтроллерлерде белгілі бір қосымшаның қазіргі қажеттіліктеріне байланысты қолдануға болатын көптеген сыртқы құрылғылар бар. Ең көп таралған перифериялық құрылғылардың мысалдары төменде келтірілген [8-15]:

–GPIO-жалпы мақсаттағы енгізу/шығару интерфейсі;

–АСТ-аналогты-сандық түрлендіргіш;

–Тізбектелген протокол контроллері-SPI(Serial Peripheral Interface), UART;

–Таймерлер;

–Жад-RAM, Flash, EEPROM.

Соңғы жылдары микроконтроллерлердің саны өсіп келеді, олардың құрамына микросхемадағы аналогты-цифрлық түрлендіру қондырғылары кіреді [16-20]. Қазіргі уақытта шығарылатын микроконтроллерлердің ішінде ATMEL фирмасының AVR отбасының микроконтроллерлері ерекшеленеді. Бұл микроконтроллерлердің тұтыну деңгейі төмен, функционалдығы жоғары бола отырып, құны төмен, жылдамдығы жоғары және бағдарламаларды бірнеше рет жазу мүмкіндігі бар.

AVR тобының микроконтроллерлері қазіргі уақытта өнеркәсіптік, коммерциялық және тұрмыстық мақсатта қолданылатын электронды жабдықтарды, әртүрлі құрылғылар мен механизмдерді басқаруды автоматтандыру үшін компьютерлерде кеңінен қолданылады. Осы сериядағы микроконтроллерлердің құны төмен, ауқымы кең және көптеген мүмкіндіктері олардың үлкен танымал болуына ықпал етті [21-24].

Сондай-ақ, көптеген басқа да перифериялық құрылғылар қол жетімді, бірақ олардың микроконтроллерлерде болуы нақты модельдерді таңдауға, сондай-ақ өндірушіге байланысты болып келеді. Перифериялық құрылғылардың көптеген түрлерінің қол жетімділігі, қуатты аз тұтыну, шағын өлшем және бағдарламалық жасақтаманы оңтайландырудың жоғары мүмкіндігі IoT қосымшаларында микроконтроллерлер қолданылатын негізгі факторлар болып табылады [25].

Қазіргі уақытта көптеген танымал компаниялар микроконтроллерді шығарумен айналысады. Сонымен қатар, операциялық күшейткіштер немесе кернеу реттегіштері сияқты стандартты өнімдерден айырмашылығы, әртүрлі компаниялардың микроконтроллері бір-бірінен ерекшеленеді. Олардың көпшілігінде әртүрлі командалық жүйелер бар. Бастапқы жағдайда деректердің қауіпсіздігін қамтамасыз ету үшін AVR [17], 8051 [18], PIC [19], ARM [20] микроконтроллері қарастырылды. Әртүрлі микроконтроллердің разрядтылығы, энергияны тұтыну, кірістірілген жадтың көлемі мен типтері, тактілік жиілігі, перифериялық құрылғылардың құрамы мен бағасы бойынша салыстырулар жүргізілді.

1.1 – кесте. AVR, 8051, PIC, ARM микроконтроллерін салыстыру

	AVR	**8051**	**PIC**	**ARM**

Разрядтылығы	8/32 бит	8 бит	8/16/32 бит	32 бит
Энергияны тұтыну	2,0 бастап 6,4 мА дейін	17 бастап 27 мА дейін	18 бастап 32 мА дейін	6,0 бастап 17,4 мА дейін
Кірістірілген жадтың көлемі мен типтері	256 кбайт дейін; SRAM 512 Б-қа дейін; EEPROM 512 Б дейін.	256 байт 4 кбайт	SRAM: 128 кБ-қа дейін, flash-жады: 512 кбайт	flash-жады: 64 кбайт; SRAM: 20 кбайт
Тактілік жиілігі	520 МГц	60 МГц	200 МГц	1,0 ГГц
Перифериялық құрылғылардың құрамы	3-тен 48-ге дейін кіріс және шығыс сызықтары	40-қа дейін кіріс және шығыс сызықтары	8-ден 16-ға дейін кіріс және шығыс сызықтары	31 кіріс және шығыс сызықтары
$\boldsymbol{P(t) = e^{-\lambda t}}$ $\boldsymbol{t = 2000}$сағ	0,983	0,892	0,917	0,932
$\boldsymbol{a = \lambda_i t}$	0,021	0,095	0,089	0,067
Бағасы	2000-7000 тг	5000-37600	6120-45000	6150-42500

1.1 кесте бойынша, микроконтроллерлерді салыстыру кезінде маңызды параметрлердің бірі тактілік жиілік болып табылады, ол микроконтроллердің жұмыс істеу жылдамдығын анықтайды. Тактілік жиілігі неғұрлым жоғары болса, микроконтроллер соғұрлым жылдам жұмыс істейді, оның сыртқы оқиғаларға жауап беруі соғұрлым дәл болады. Осы параметр негізінде екі микроконтроллердің жоғары тактілік жиіліктері бар: AVR және ARM. Аталған микроконтроллерді салыстыру барысында AVR микроконтроллері Arduino платформасының және бағдарламалау материалының болуының артықшылығына ие. Кірістірілген жадтың көлемі мен типтерін, сонымен қатар бағасын салыстыру кезінде AVR микроконтроллері жақсы көрсеткіштер көрсетті.

Талдау барысында жоғары қозғалтқышты басқару таймерлері және әртүрлі интерфейстерді қоса алғанда, барлық қажетті перифериялық құрылғыларды қамтитын AVR микроконтроллері таңдалды. Бұл біріктірілген мүмкіндіктер ұсынылған архитектуралардағы аз перифериялық бай микроконтроллерлерге қарағанда үлкен артықшылық береді.

Микроконтроллердің негізгі ерекшелігі оны бағдарламалауға болады. Микроконтроллерді кішкентай өлшемі мен қарапайым сәулетінің арқасында әртүрлі перифериялық құрылғылары бар әртүрлі қосымшаларда оңай қолдануға болады. Микроконтроллердің дербес компьютерлерден айырмашылығы, микроконтроллер процессор мен перифериялық құрылғылардың функцияларын бір кристалда біріктіреді, жедел жады және код пен деректерді сақтауға

арналған тұрақты сақтау құрылғысы бар, бірақ есептеу ресурстары салыстырмалы түрде аз болып келеді [21-25].

Кез келген микроконтроллерді жалпы үш құрамдас бөлікке бөлуге болады:

1. Арифметикалық логикалық құрылғы немесе процессор деп аталатын есептеу блогы. CPU сандармен әртүрлі операцияларды орындауға мүмкіндік береді, осыған байланысты жүйенің маңызды бөлігі болып табылады. Осы орындалатын операциялардың реттілігі бағдарлама деп аталады [26]. Әр операция сан түрінде кодталады және микроконтроллердің жадына жазылады.

2. Микроконтроллердің екінші негізгі бөлігі жад модулі болып табылады. Бұл мамандандырылған электрондық құрылғы, бұл әрқайсысында бір санды сақтауға болатын ұяшықтар жиынтығы. Дәл осы жерде жазылған бағдарлама және басқа микроконтроллер командалары сақталады.

3. Микроконтроллердің үшінші құрамдас бөлігі енгізу/шығару порттары деп аталады. Егер процессор мен жады микроконтроллер корпусының ішкі бөлігінде болса және біз оларды көрмейтін болсақ, онда енгізу-шығару порттары әрдайым сыртқы бөлігінде орналасады және олар кішкентай металл аяқтар түрінде болады.

ATmega328 бұл Atmel picoPower AVR RISC негізіндегі 8 биттік микроконтроллер. AVR микроконтроллерлер тобына кіретін Atmega328 бүгінде әртүрлі деңгейдегі электрониканы құруда кеңінен қолданылады. Микроконтроллер сонымен қатар әуесқой бағдарламашылардың ортасында өте кең таралған шағын тақтасы бар Arduino Uno-ның негізі болып табылады.

Әдетте, аталған құрылғыларды бағдарламалау SPI-сериялық перифериялық интерфейсті қолдану арқылы жүзеге асырылады. Деректер шинасы арқылы бірқатар сигналдар өтеді [27]:

- MOSI-деректерді енгізу;
- MISO-ның таралуының шығуы;
- SCK-синхрондау кірісі.

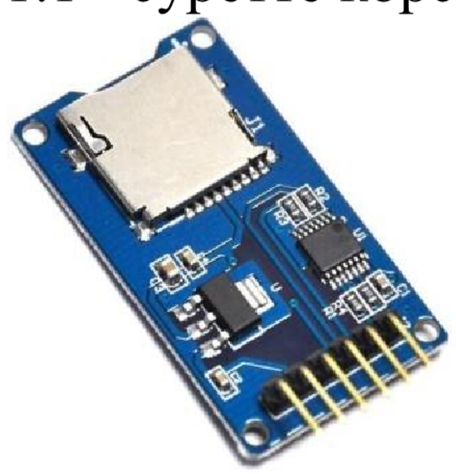
SCK-синхрондау кірісін бағдарламашы жасайды және екі байланысты құрылғы арасында синхронды алуды қамтамасыз етеді. SPI интерфейсі синхронды байланыс шинасы болып табылады.

Микроконтроллердегі мәтіндік деректерді шифрлау үшін микроконтроллерлер мен microSD картасы қолданылады. Келесі қадамда microSD картасынан шифрланған файлдар оқылады және жазылады. MicroSD картасын Arduino тақтасымен интерфейстеу үшін microSD модулін (SPI байланыс протоколы) қолданады. Микроконтроллерлері бар microSD картасын пайдалану, әсіресе деректерді тіркеу немесе файлдық жүйеге (SPIFFS) сәйкес келмейтін файлдарды сақтау үшін пайдалы [28]. Микроконтроллерлер Arduino IDE көмегімен бағдарламаланады.

Жұмыста әртүрлі файлдармен жұмыс істеу үшін SD және microSD карталары пайдаланылды. Олар датчиктерден және түрлендіргіштерден оқылған ақпаратты сақтай алады. Arduino тақталарындағы жадтың жалпы көлемі 4 Кбайтқа дейін. SD карталарын пайдалану арқылы әртүрлі көлемдегі ақпаратты сақтауға мүмкіндік береді. SD картасын Arduino тақтасына қосу

үшін жад модул картасы пайдаланылды. Оның резисторы, кернеу реттегіші және карта ұясы бар. Модульді пайдалану жад картасындағы деректерді оқуды, жазуды және сақтауды қолдайды.

MicroSD картасына арналған модульді дайындау. Деректерді microSD картасына сақтау үшін төмендегі 1.1 - суретте көрсетілген модуль қажет.



Сурет 1.1 - microSD карта модулі

AtmelAVR тобындағы микроконтроллер арқылы microSD карта модулін пайдаланбас бұрын оны пішімдеу керек. Ол үшін келесі әрекеттер орындалады:

- компьютерге microSD картасы салынады. «Осы компьютерге» өтіп, microSD картасы белгішесін тінтуірдің оң жақ түймешігімен басылады. Пайда болған мәзірде «Формат» түймесін басылады. Жаңа терезе пайда болады. «FAT32» таңдап, «Бастау» түймесін басып, нұсқаулар орындалады. MicroSD картасына арналған модуль қосылады. MicroSD картасын пішімдегеннен кейін оны модульге қосу қажет.

Микроконтроллерде мәтіндік деректерді шифрлау үшін Arduino тақтасымен microSD картасы қолданылады. Келесі қадамда microSD картасына шифрланған файлдар оқытылады және жазылады. MicroSD картасын Arduino тақтасымен интерфейстеу үшін microSD карта модулі (SPI байланыс протоколы) қолданылады.

Микроконтроллерлер арқылы SD картасына деректерді тіркеудің принципиалдық схемасы Fritzing бағдарламалық ортасында сызылған. ATmega328 микроконтроллерінің принципиалдық схемасы 1.2-суретте көрсетілген.

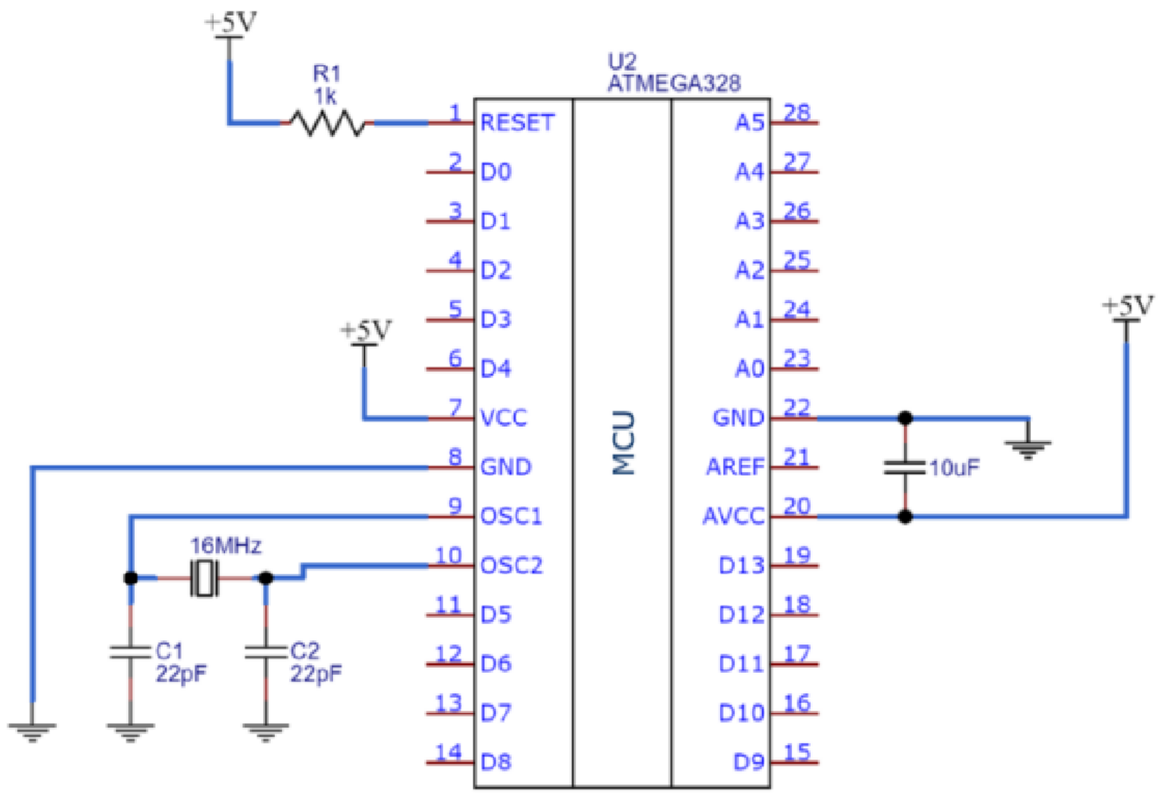
Кесте 1.2 – MicroSD карта модуліне Arduino Uno деректерді тіркеу

micro-SD картасы	Vcc	GND	CS	SCK		MISO	MOSI
Arduino Uno	+5V	GND	4	13	12	11	

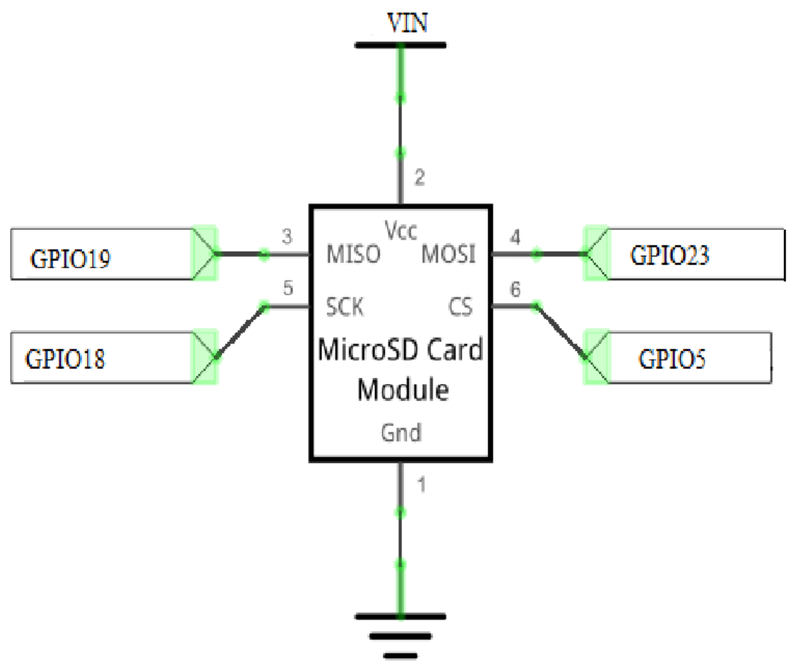
MicroSD картаны қосу үшін 6 контакт пайдаланылады және өзара әрекеттесу SPI интерфейсі арқылы жүзеге асырылады. Картаны қосу үшін ATmega328 микроконтроллері, карта модулі және 6 сым қажет. MicroSD картасына Arduino Uno деректерді тіркеу 1.2-кестеде көрсетілген.

Сандық енгізу-шығару келесідей қосылған: Arduino Uno тақтасындағы MOSI мен MISO контактілері D11, SCK - D13-ке, CS – 4-ке, VCC - +5 V-қа, GND – GND-қа. Тақтада 3,3 және 5 вольтке қосуға арналған қосқыштар бар.

MicroSD картаның қуат көзі 3,3 вольтты құрайды, сондықтан бірдей қуат көзі бар микроконтроллерді пайдалану қажет, әйтпесе кернеу деңгейінің түрлендіргіштері қажет болады. ATmega328 микроконтроллеріне MicroSD карта модулімен байланысын және ATmega328 микроконтроллерінің принципиалдық схемасын 1.2 – суреттен көруге болады.



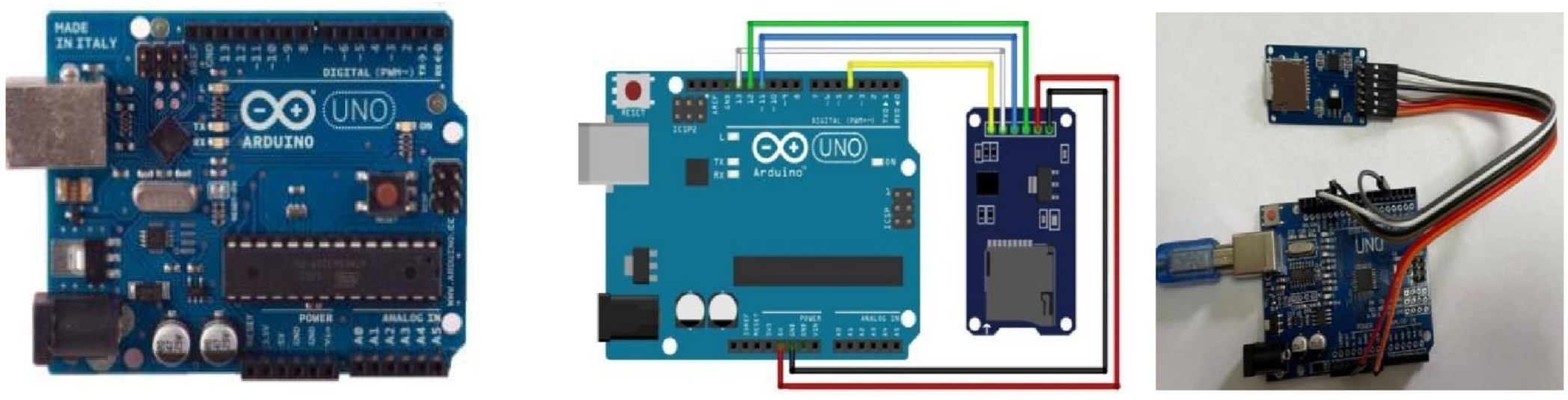
а)



ә)

Сурет 1.2 – а) ATmega328 микроконтроллерінің принципиалдық схемасы ә) MicroSD карта модулі мен ATmega328 микроконтроллердің байланысу принципиалдық схемасы

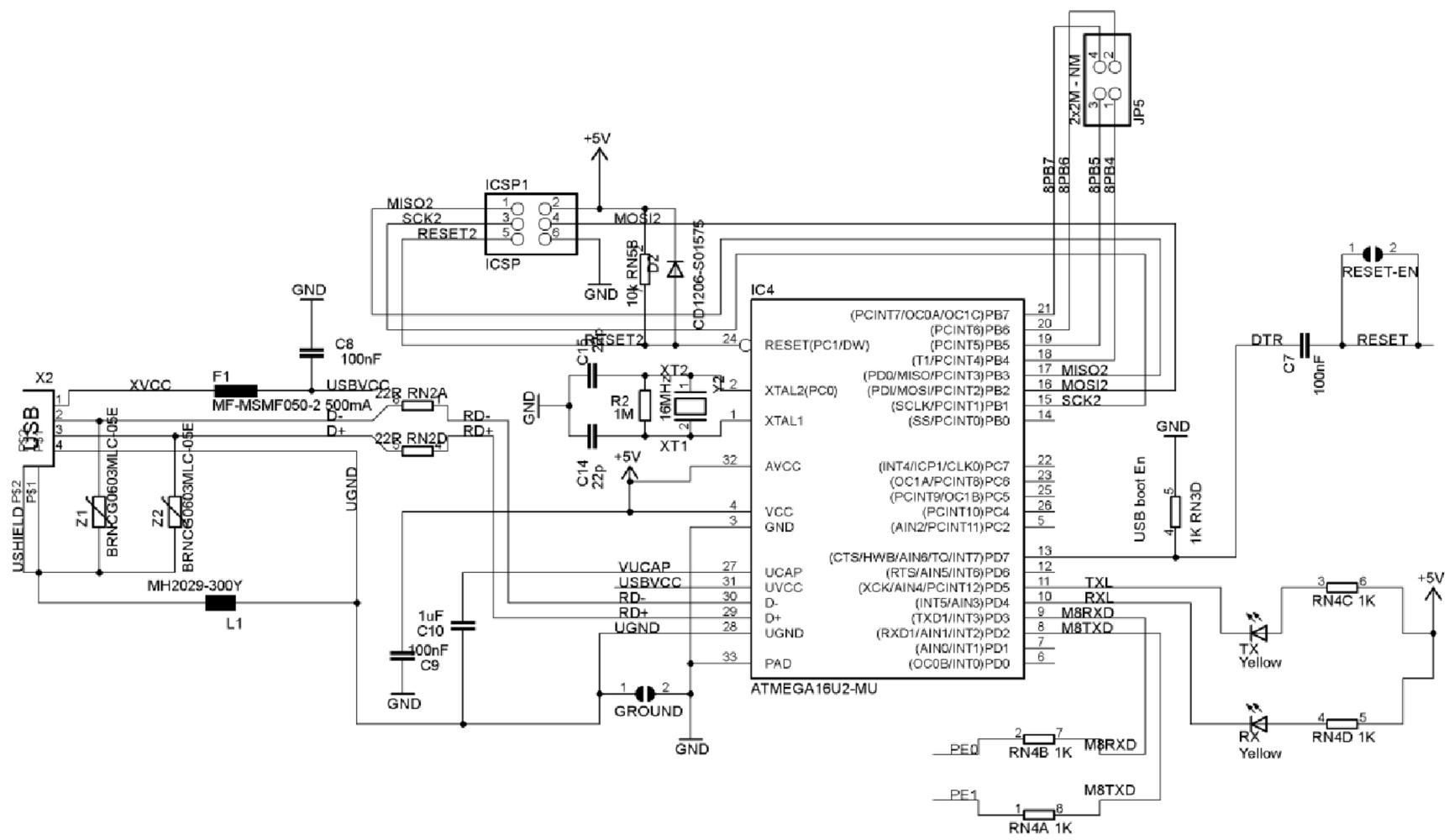
Екі микроконтроллерге арналған Arduino UNO тақтасында сәйкесінше 16 МГц жиіліктегі екі кварц резонаторы орнатылған, бұл жоғары жылдамдық яғни, 16 миллион секундына импульстарды береді [29]. Аталған микроконтроллер дәл осы импульстарды қарастырады, содан кейін олардың саны бойынша кез келген процедураның басталуынан қанша уақыт өткендігі туралы қорытынды жасайды. ATmega328 микроконтроллеріне microSD карта модулімен байланысын 1.3 – суреттен көруге болады.



Сурет 1.3 –ATmega328 микроконтроллеріне microSD карта модулімен байланысту

AtmelAVR тобындағы келесі қарастырылатын микроконтроллер ATmega 2560**.** ATmega 2560 микроконтроллерлері классикалық AVR тобынан интерфейс функцияларының кеңейтілген жиынтығымен, қосымша командаларды қолдауымен, жад сыйымдылығын арттырумен, кірістірілген аналогті сандық түрлендіргішімен, архитектуралық мүмкіндіктерді және бағдарламалық қамтамасыз ету үйлесімділігін сақтай отырып ерекшеленеді.

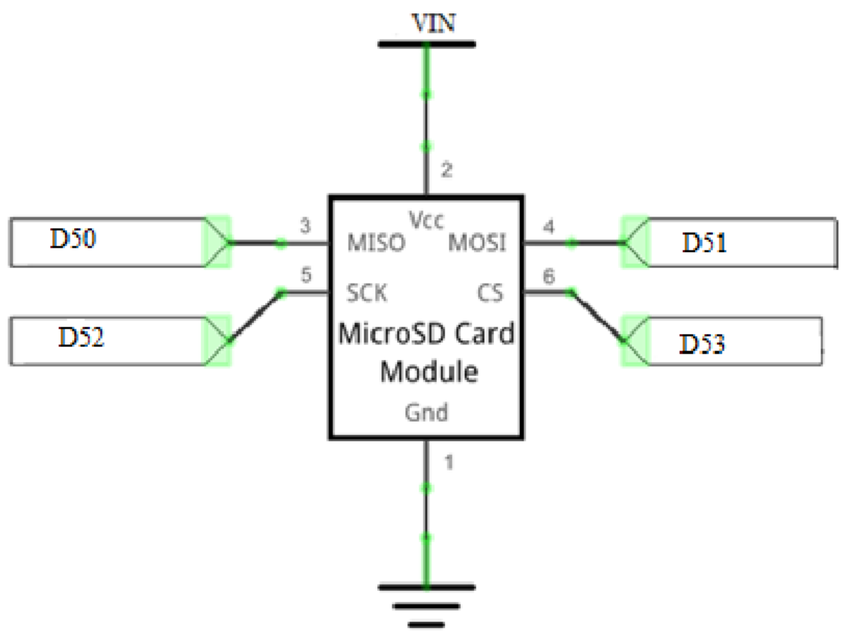
ATmega 2560 микроконтроллері RISC процессорымен ұсынылған және 16 МГц жиілікте жұмыс істейді [30]. Оның чипінің кристалында компьютерлік жүйенің жалпы тұжырымдамасына қатысты барлық құрылғылар орналасқан, атап айтқанда: жедел және қайта бағдарламаланатын, сонымен қатар flash жады, интерфейс көпірлері, мультипликатор. 1.4 – суретте Atmega 2560 микроконтроллерінің принципиалдық схемасы көрсетілген.



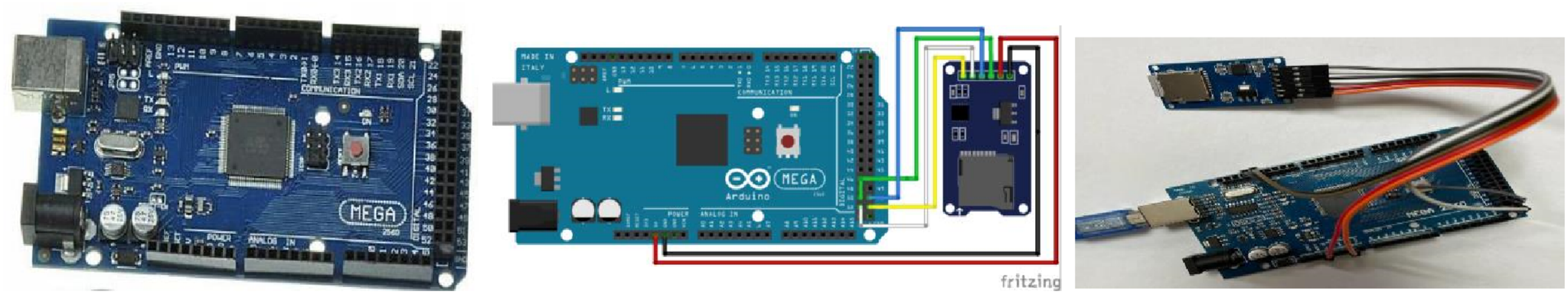
Сурет 1.4 – Atmega 2560 микроконтроллерінің принципиалдық схемасы[30]

ATMega2560 – тың басқа өндірушілердің контроллер модельдерімен салыстырғанда ең үлкен артықшылығы – бұл әмбебаптылық, Arduino IDE процессорының кодын әзірлеудің дамыған жүйесі, мүмкіндіктердің құжатталуы және көптеген кеңейту модульдерінің болуы.

Сондай – ақ, құрылғының өзі командаларды орындау уақытының маңызы зор, олардың күрделілігіне қарамастан барлық нұсқаулар үшін бірдей. 1.5 – суретте ATmega 2560 микроконтроллеріне MicroSD картасын байланысту қарастырылған. 1.6 – суретте ATmega 2560 микроконтроллеріне microSD карта модулімен байланысу принципиалдық схемасы көрсетілген.



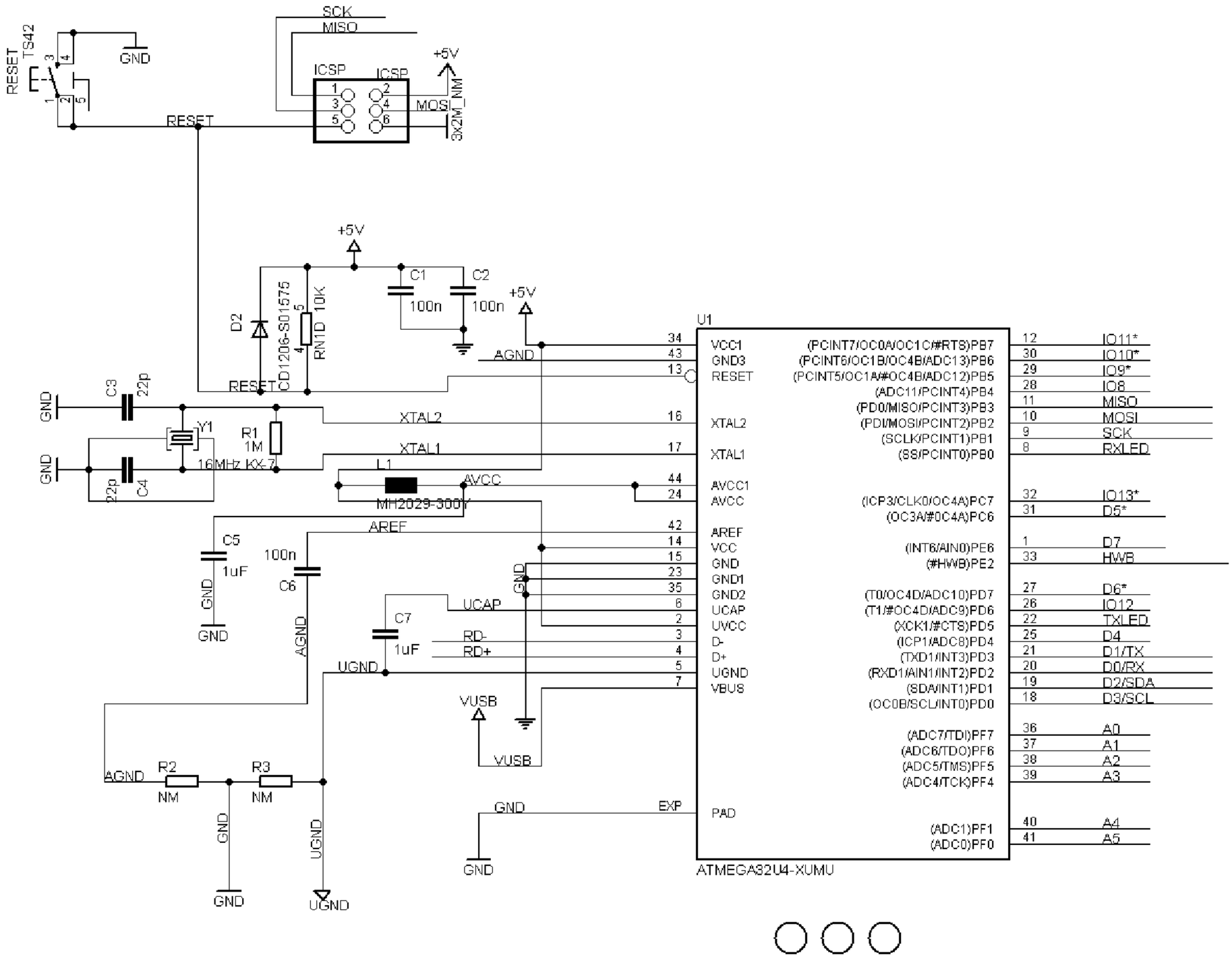
Сурет 1.5 – MicroSD модулі мен Atmega 2560 микроконтроллердің байланысу принципиалдық схемасы



Сурет 1.6 – ATmega 2560 микроконтроллеріне microSD карта модулімен байланысу

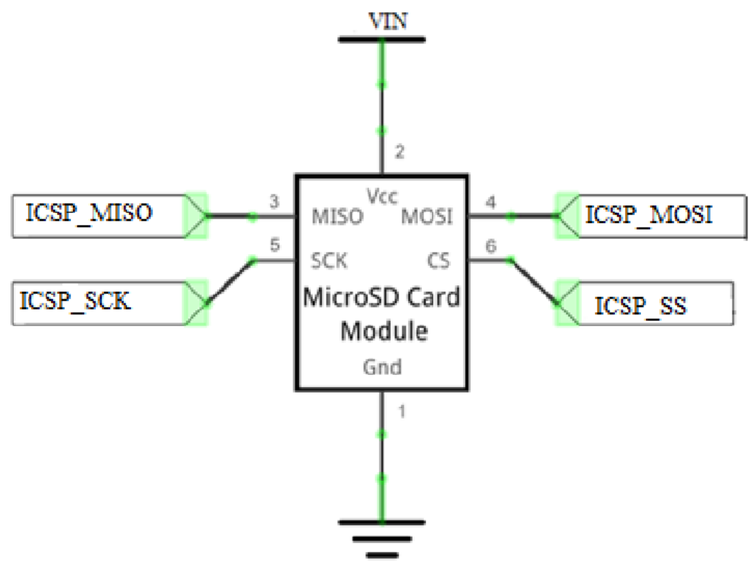
ATmega32U4 - RISC AVR негізінде жасалған үнемді 8 биттік микроконтроллер. Бір синхрондау циклі үшін көптеген нұсқауларды орындау арқылы ATmega32U4 1 MIPS/МГц өнімділігіне жетеді, бұл әзірлеушіге қуат тұтыну мен өнімділік қатынасын оңтайландыруға мүмкіндік береді [31]. 1.7 – суретте ATmega32U4 микроконтроллерінің принципиалдық схемасы бейнелеген.

AVR ядросы әртүрлі нұсқаулық жиынтығын қолдайды және жалпы мақсаттағы 32 жұмыс регистрін қамтиды. Барлық 32 регистрлер арифметикалық логикалық құрылғыға тікелей қосылған, бұл бір нұсқаулықта екі регистрді көрсетуге және оны бір синхрондау циклінде орындауға мүмкіндік береді [32]. Нәтижесінде, мұндай архитектура бағдарлама кодын орындаудың жақсы тиімділігіне ие және дәстүрлі CISC микроконтроллерлерінің мүмкіндіктерінен асып түсетін дәрежеге жетеді.

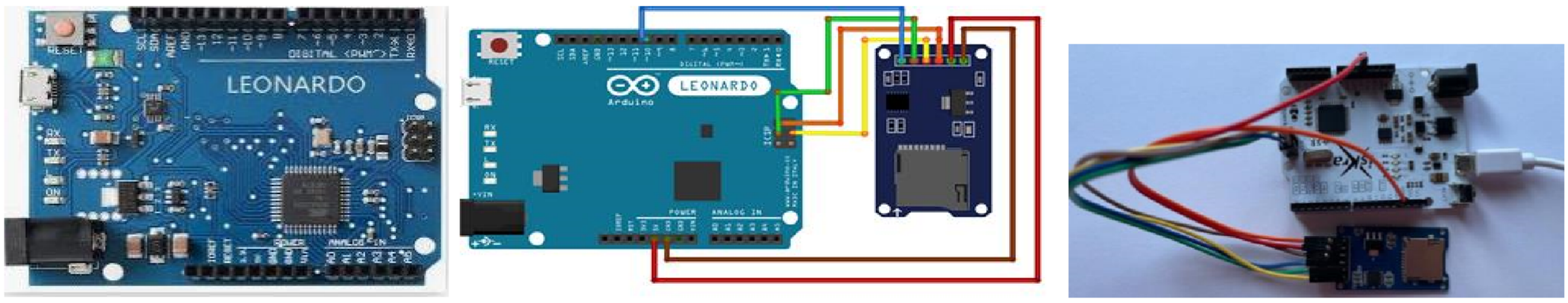


Сурет 1.7 – ATmega32U4 микроконтроллерінің принципиалдық схемасы

8 биттік RISC бір кристалындағы орталық процессордың ішкі жүйелік өздігінен бағдарламаланатын флэш-жадымен үйлесуі ATmega32U4 микроконтроллерін кірістірілген басқару мәселелерін шешудің тиімді және арзан құралына айналдырады [33]. 1.8 және 1.9 - суреттерде ATmega32U4 микроконтроллеріне microSD карта модулімен байланыстыру схемалары көрсетілген.



Сурет 1.8 – ATmega32U4 микроконтроллері мен SD картаның байланысу принципиалдық схемасы



Сурет 1.9 – ATmega32U4 микроконтроллеріне microSD карта модулімен байланысту

Atmega32u4 микроконтроллерін жобалауға арналған аппараттық және бағдарламалық құралдардың толық жиынтығын қолдайды, соның ішінде C-компиляторлар, макроассемблерлер, бағдарламалық реттегіш/ симуляторлар, схема ішіндегі эмуляторлар және бағалау жиынтықтары.

DFRobot Beetle BLUE – бұл Dfrobot өндірушісінің кішкентай микроконтроллері, ол ATmega328-де кіріктірілген Bluetooth 4.0-мен жұмыс істейді. Шын мәнінде, бұл атақты Bluno (UNO + Bluetooth) шағын нұсқасы және онымен толық үйлесімді микроконтроллер [34].

Beetle Blue (бұрынғы атауы Blue Beetle Nano kit) - Bluetooth 4.0 (BLE) бар Arduino Uno негізіндегі тақта. Бұл нарықтағы ең кішкентай BLE тақтасы. Ол қосымша кітапхана мен драйверлерсіз кодтарды жүктеу үшін стандартты IDE Arduino ортасын қолданады. 1.10 және 1.11 - суреттерде Bluno Beetle V1.0 микроконтроллерінің microSD картаға жалғану принципиалдық схемасы және байланыстыру сызбасы бейнеленген.

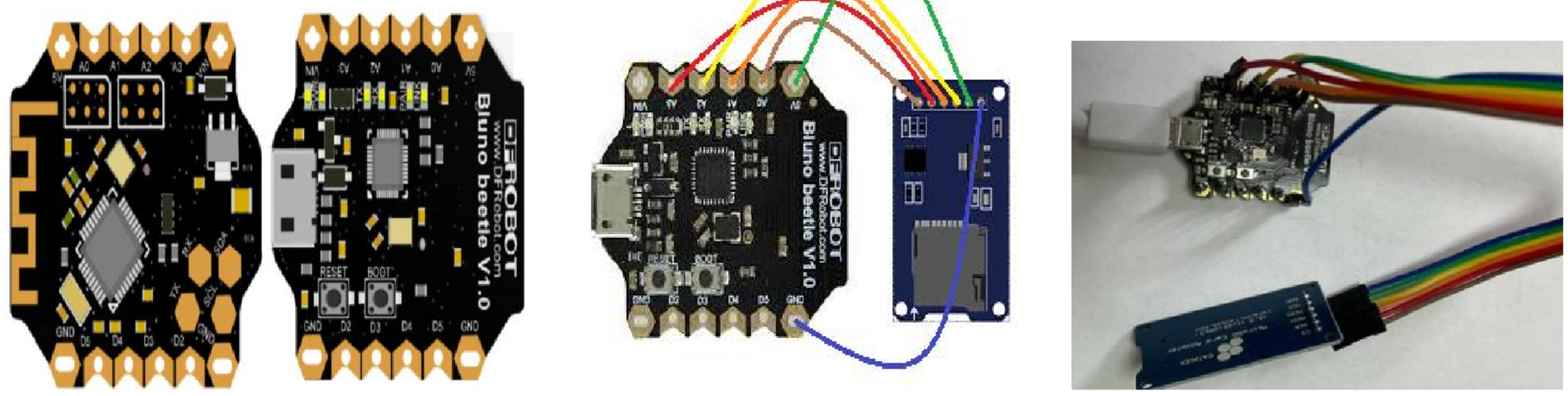
Micro-SD картасында деректер сақталынады. Сақталған деректерді micro-SD картасын қолдану арқылы оқытылады және жазылады. Bluno Beetle V1.0 жүйесіндегі қосылу мүмкіндігі бойынша micro-SD модулі қарапайым SD карталарына арналған модульден еш айырмашылығы жоқ [35]. Модуль Bluno Beetle V1.0 жүйесіне SPI шинасы арқылы қосылады. 1.3 – кестеде MicroSD картасына Arduino Uno деректерді тіркеу келтірілген.

Кесте 1.3 – MicroSD картасына Arduino Uno SPI порттарын тіркеу

micro-SD карта	Vcc	GND	CS	SCK	MISO	MOSI
Bluno Beetle	+5V	GND	A0	A3	A2	A1



Сурет 1.10 – Bluno Beetle V1.0 микроконтроллерінің microSD картаға жалғану принципиалдық схемасы



Сурет 1.11 – Bluno Beetle V1.0 микроконтроллеріне microSD карта модулімен байланыстыру

Beetle BLE-бұл өте кішкентай және арзан тақташа, Arduino үйлесімді, Bluetooth 4.0. Beetle BLUE-ді DIY жобалары, мастер-кластар, сыйлық жобалары, электронды тоқыма және білім беру сияқты бір реттік жобалар үшін пайдалануға болады [36]. Жабдықты сатып алуға тым көп ақша таба алмайтын студенттер мен өндірушілер үшін Beetle олар үшін керемет шешім бола алады.

«Құрылғы» ыңғайлы бағдарламалау және күйін келтіру үшін microUSB портымен жабдықталған, сымдарды бекітуге арналған V-жиектері бар, Bluetooth HID және IBeacon, сондай-ақ сымсыз бағдарламалауды қолдайды.

ESP32 микроконтроллерін Espressif Systems компаниясы әзірлеген. ESP 32-интеграцияланған Wi-Fi және Bluetooth контроллері бар бір кристаллдағы жүйе. ESP 32 микроконтроллерінде Tensilica Xtensa LX6 ядросы қолданылады. ESP 32 бар тақталар өңдеу қабілеті жоғары, перифериясы дамыған сонымен бірге бағасы басқа микроконтроллерімен салыстырғанда төмен болғандықтан, өте танымал.

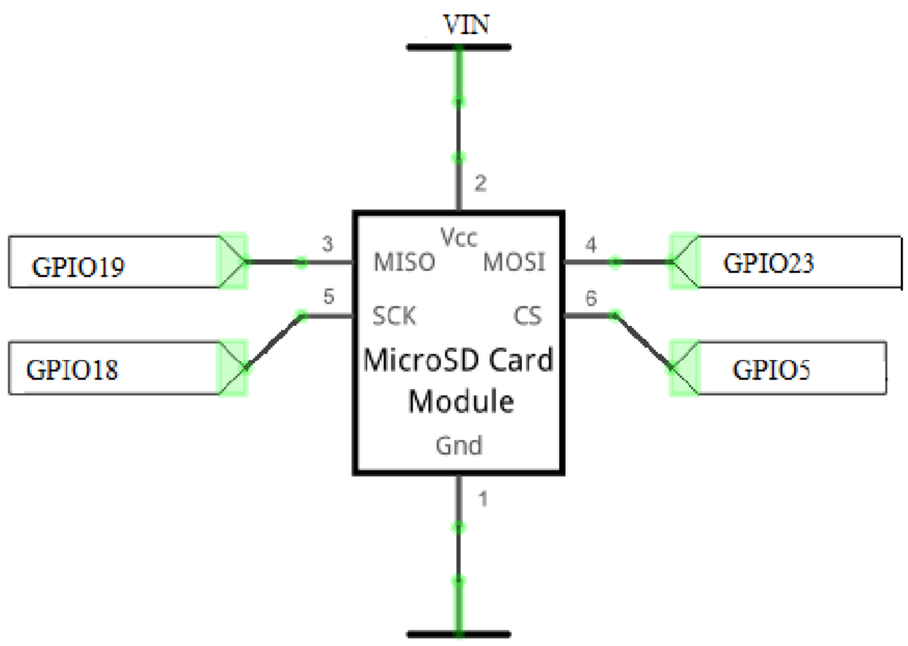
SD картасының модулі ESP32-ге SPI интерфейсі арқылы қосылған. ESP32 микроконтроллеріндегі SPI байланыстары 5, 18, 19 және 23 байланыстар болып табылады [37]. Байланыстар төмендегі 1.4 - кестеде жіктеледі.

Кесте 1.4 - microSD карта модулі мен ESP32 микроконтроллерінің байланысы

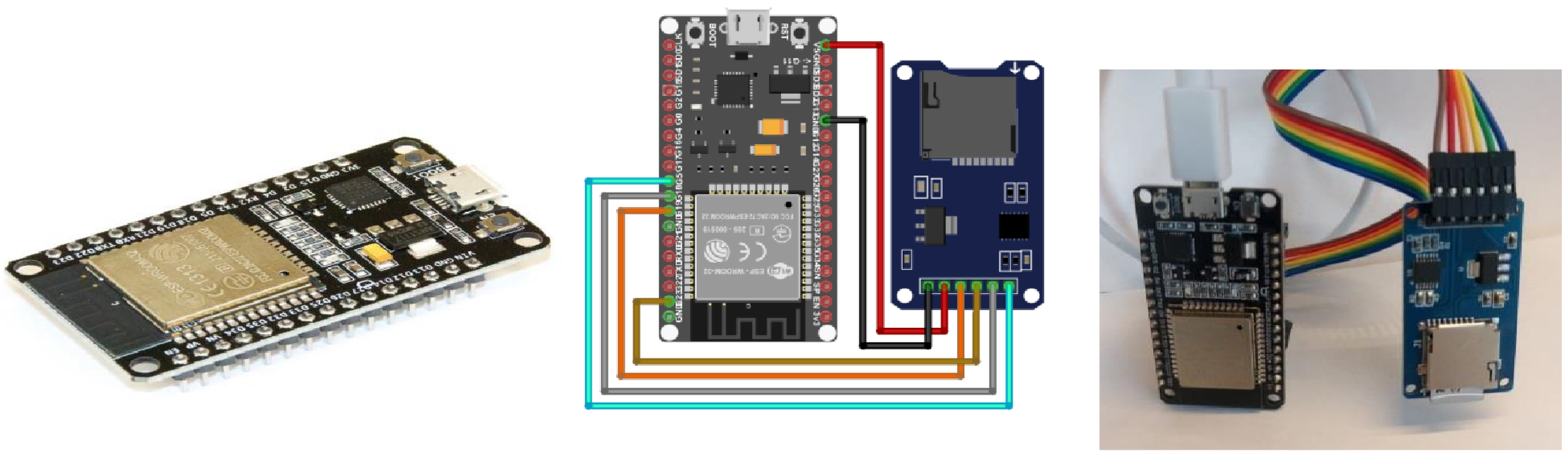
microSD карта модулі	ESP32 микроконтроллері
Vcc	VIN
GND	GND
CS	GPIO5
SCK	GPIO18
MISO	GPIO19
MOSI	GPIO23

ESP 32 тақтасы соңғы жылдарда пайда болғанына қарамастан, ол мобильді қосымшаларда, электроникада, ең бастысы IoT қосымшалары мен құрылғыларды, коммерциялық жобаларда кеңінен қолданылады [39]. 1.12 және

1.13 суреттерде ESP32 микроконтроллеріне microSD карта модулімен байланыстыру принципиалдық схемасы көрсетілген.

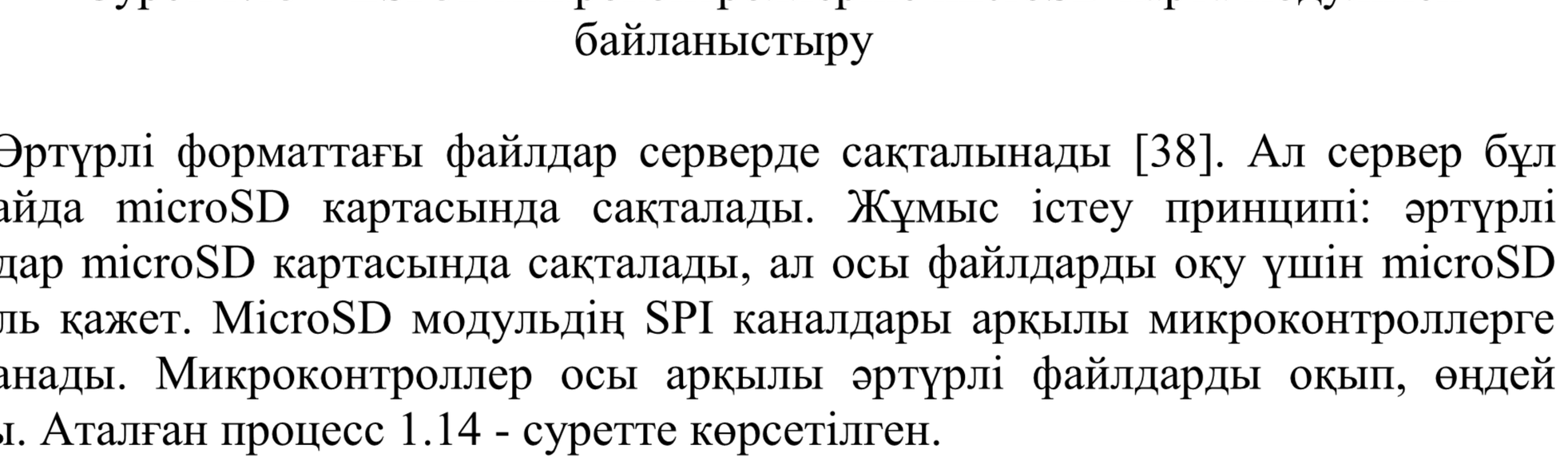


Сурет 1.12 – ESP32 микроконтроллерінің microSD карта модулімен байланыстыру принципиалдық схемасы



Сурет 1.13 – ESP32 микроконтроллеріне microSD карта модулімен байланыстыру

Әртүрлі форматтағы файлдар серверде сақталынады [38]. Ал сервер бұл жағдайда microSD картасында сақталады. Жұмыс істеу принципі: әртүрлі файлдар microSD картасында сақталады, ал осы файлдарды оқу үшін microSD модуль қажет. MicroSD модульдің SPI каналдары арқылы микроконтроллерге жалғанады. Микроконтроллер осы арқылы әртүрлі файлдарды оқып, өңдей алады. Аталған процесс 1.14 - суретте көрсетілген.



Сурет 1.14 – MicroSD карта модуль арқылы ESP32 микроконтроллерімен файлды өңдеу.

ESP 32 тақтасында Alibaba тобының IoT жарық диодты білезігі салынды. Бұл сымсыз экран, онда әрбір білезік пиксель сияқты жұмыс істейді. Сонымен қатар, микроконтроллер негізінде қатысуды бақылаудың биометриялық жүйесі және климаттық жағдайларды талдауға байланысты жобалар жүзеге асырылды. ESP 32 тақтасы негізінде температура, ылғалдылық және қысым сенсорлары бар метеостанцияны кез-келген адам жинап, өңдей алады. ESP 32 музыкалық ойнатқыштарда, дауыстық басқару көмекшісі, аудио гарнитурада кең ауқымда қолданылады.

Алғашқы микроконтроллерлердің архитектурасы қазіргіге қарағанда әлдеқайда қарапайым болған, сондықтан олар ассемблер тілінде бағдарламалар жазылған [40]. Микроконтроллерлердің күрделілігінің артуымен бағдарламалық жасақтаманы тек ассемблер тілінде жазу мүмкін болмады. Қазіргі уақытта өте жақсы C/C ++ компиляторлары бар, олар бағдарламалық жасақтаманың барлық дизайнын жеңілдетеді және микроконтроллерлерге операциялық жүйелерді енгізуге мүмкіндік береді. Қазіргі таңда, ассемблер тілі әлі де қолданылады, бірақ тек C/C ++тілінде жазуға болмайтын аппараттық процедураларда ғана қолданылады. 1.5 кестеде AtmelAVR микроконтроллер тобындағы әртүрлі микроконтроллерлер жіктеліп сипатталған.

Кесте 1.5 – AtmelAVR микроконтроллердің сипаттамалары

Микроконтроллер	ATmega328	ATmega 2560	ATmega 32U4	DFRobot Beetle BLE	ESP32
Қолданыстық кернеуі	5 В	5 В	2.7-5.5 В	5 В	2.2-3.6 В
Ұсынылған кіріс кернеуі	7-12 В	5- 7В	7-12 В	7-12 В	3,7-20 В
Шекті кіріс кернеуі	6-20 В	7-12 В	6-20 В	6-20 В	7-22 В
Сандық Кіріс/Шығыс	14	54	20	4	21
Аналогтық кірістер	6	16	12	4	15
Кіріс/шығысқа арналған тұрақты ток	40 мА	150 мА	40 мА	40 мА	30 мА
Шығару үшін Тұрақты ток	50 мА	800 мА	50 мА	50 мА	40 мА
Флеш-жад	32 Кб	256 Кб	32 Кб	32 Кб	520 Кб
Жедел жады	2 Кб	8 Кб	2.5 Кб	2 Кб	448 Кб

EEPROM	1 Кб	4 Кб	2 Кб	1 Кб	512 б
Тактілік жиілік	16 МГц	16 МГц	16 МГц	16 МГц	240 МГц

Микроконтроллер бағдарламалық жасақтамасын жасаудың бірнеше стратегиясы бар. Бағдарламалық стратегияны таңдау микроконтроллерді нақты қолданудың күрделілігіне байланысты. Бұл өте қарапайым жүйелер болуы мүмкін, олардың міндеті батырманы басқан кезде жауап беру болып табылады. Екінші жағынан, көптеген параметрлерді үнемі қадағалап отыратын және нақты уақыт режимінде өз нәтижелерін орнатуды қажет ететін жүйелер сияқты күрделі жүйелер бар. Аталған жүйелер әртүрлі, сондықтан олардың бағдарламалық жасақтамасы әртүрлі стратегияларға сәйкес жасалған.

Arduino-ның танымал болуына байланысты тақта өндірушілерінің саны жүздегенге жете бастады. Seeeduino Mega тақтасы, Arduino Mega тақтасымен толық үйлесімді. Arduino Duemilanove немесе Diecimila үшін жасалған көптеген (shield) кеңейту модульдерімен үйлесімді. Ол көптеген енгізу/шығару түйреуіштері, 70-ке дейін цифрлық енгізу/шығару, 16 аналогтық кіріс, 14 PWM (импульстік ен модуляциясы) және 4 аппараттық сериялық порттары бар ATmega2560 процессорымен жабдықталған. Seeed Shield өнімдерімен толық үйлесімді, Эбен Аптон басқаратын Raspberry Pi Foundation ағылшын компаниясымен әзірленген. Raspberry Pi – бұл банк картасының өлшеміндей болатын бір тақталы компьютер, бастапқыда ақпараттық технолониялар оқытуға арналған бюджеттік жүйе ретінде жасалған, бірақ кейінірек кеңірек қолданыс пен атаққа ие болды. Raspberry Pi бір тақталы компьютер үлгілерінің көпшілігі банк картасының өлшеміндей төрт қабатты баспа тақтасында толығымен жинақталған түрде таратылады. Raspberry Pi-нің ең қызықты мүмкіндіктерінің бірі - GPIO (жалпы мақсаттағы енгізу/шығару) порттарының болуы. Осыған байланысты оны әртүрлі құрылғыларды басқару үшін пайдалануға болады [41].

Arduino IDE ортасы кәсіби деңгейі орташа пайдаланушыларға бағытталған. Аталған орта қарапайым автоматтандыру және робототехника жүйелерін құруға арналған аппараттық және бағдарламалық қамтамасыз етеді. Arduino тақтасы - жадысы бар және көптеген стандартты кірістер мен шығыстармен жабдықталған дербес процессор. Оған әртүрлі құрылғылар мен механизмдер, сенсорлар, қозғалтқыштар және т.б. оңай қосылады. Arduino-ның негізгі сериялары: Uno, Nano, Lilypad, Mega, Leonardo, Micro, Due. Ең көп таралғаны Uno сериясы болып келеді. Олардың барлығы өлшемі бойынша ғана емес, сонымен қатар пішіні, сипаттамалары және сәйкесінше бағасы бойынша ерекшеленеді. Бұл тақталардың әрқайсысы бір-бірінен ерекше және әртүрлі жағдайлар үшін артықшылықтары бар [42]. Осыдан шығатын қорытынды: нақты жобалар немесе қосымшалар үшін Arduino тақтасы қолайлы.

Тақталардың үлкен таңдауы мен олардың модификацияларының ішінен орнатудың қарапайымдылығы, құны, ауқымдылығы, сондай-ақ құрамдас бөліктер мен сенсорлардың құны сияқты берілген критерийлерге неғұрлым

сәйкес келетінін таңдау керек. Тақталардың негізгі сипаттамаларын салыстырсақ ең көп таралған тақта ретінде Arduino Uno болып табылады.

Arduino Mega әзірленген жүйе үшін таңдалды, өйткені ол аналогтық және цифрлық кірістердің ең көп санына ие. Сондықтан, қажет болса, сенсорлар мен кеңейту карталарын қосу арқылы жүйені масштабтауға болады. Бұл нақты қажеттіліктерге арналған жүйені әзірлеуде көбірек икемділікке мүмкіндік береді.

Компьютерге USB кабелі немесе айнымалы тұрақты ток адаптері, сонымен қатар қайта зарядталатын батарея арқылы қуат беру арқылы Arduino Mega тақташасымен жұмыс істеуге болады. Arduino Mega тақташасы Arduino Uno платформаларына арналған барлық кеңейту тақталарымен үйлесімді болып келеді. Arduino тақташасы бұл стандартты жеке компьютерлерге қарағанда қоршаған физикалық ортамен тығыз әрекеттесетін электронды құрылғыларды (электронды конструктор) жобалау құралы. Бұл «физикалық компьютерлерге» арналған ашық бағдарламалық жасақтама платформасы, бағдарламалық жасақтаманы жазу үшін қазіргі заманғы ортасы бар қарапайым баспа тақтасына салынған. Arduino-ға қосылуға болатын әртүрлі сандық және аналогтық датчиктерден сигналдарды қабылдау және әртүрлі атқарушы құрылғыларды басқару мүмкіндігі бар электрондық құрылғыларды құру үшін қолданылады. Arduino негізіндегі құрылғы жобалары дербес жұмыс істеу мүмкіндігіне ие, сонымен қатар компьютердегі бағдарламалық жасақтамамен өзара әрекеттесе алады. Бастапқы ашық деректерді бағдарламаларды әзірлеу ортасы тегін жүктеу үшін қол жетімді [43-44]. Processing мультимедиялық бағдарламалау ортасына негізделген «физикалық есептеу» үшін ұқсас Wiring платформасы Arduino бағдарламалау тілін іске асырған.

IDE – интеграцияланған даму ортасы болып табылады, аталған ортаны қолдана отырып, кәсіби немесе қарапайым бағдарламашылар бағдарламаларды жазады және қарапайым мәтіндік редакторларды қолданғанға қарағанда тезірек және ыңғайлы болып келеді.

Arduino IDE ыңғайлы мәтіндік редакторға бағдарламалар жасауға, оларды машина кодына құрастыруға және Arduino тақтасының барлық нұсқаларына жүктеуге мүмкіндік береді. Arduino қауымдастығының ресми сайтынан қосымшаны толығымен тегін жүктеуге болады [45].

Arduino IDE-ң төмендегідей артықшылықтарын атап көрсетсе болады [46]:

- қол жетімділігі жоғары;
- қолдануға және түсінуге тиімді интерфейс;
- Windows операциялық жүйелерінің барлық нұсқаларымен бағдарлама үйлесімді;
- бағдарламалауға қажетті құралдардың болуы;
- бағдарламалау тілдерінің бірнеше нұсқалары бар;
- сақтау, экспорттау, тексеру, іздеу, орын ауыстыру функциялары бар.

Осындай артықшылықтарға негізделе отырып, микроконтроллерде кітапханаларын іске асыру мақсатында Arduino интеграцияланған даму ортасы таңдалынды. microSD картасы мен файлдарды микроконтроллермен

байланыстыру үшін, алдымен келесі кітапханаларды қосу керек: файлдарды өңдеу үшін FS.h, microSD картасымен әрекеттесу үшін SD.h және SPI байланыс протоколын пайдалану үшін SPI.h қолданылды.

String, String.h, math.h кітапханалары математикалық операциялармен, сонымен қатар нөлмен аяқталатын жолдармен және әртүрлі жады функцияларымен жұмыс істеуге мүмкіндік береді [47]. ReadFile() функциясы файлдың мазмұнын оқиды және оны сериялық мониторға басып шығарады. Алдыңғы функциялар сияқты, файлдық жүйені және файл жолын SD аргументі ретінде жібереді.

Мазмұнды файлға жазу үшін writeFile() функциясын пайдалануға болады. Аргумент ретінде SD файлдық жүйе, файл жолы және хабарлама алынады.

1.2 Әртүрлі AtmelAVR микроконтроллердің сенімділігін есептеу

Қазіргі таңда заманауи микроконтроллерлер әзірлеушіге өз ойларын жүзеге асыруға кең мүмкіндіктер ашылды. Сенімділік – объектінің берілген режимдерде, пайдалану, техникалық қызмет көрсету, жөндеу, сақтау және тасымалдау шарттарында барлық функцияларды орындау мүмкіндігін сипаттайтын барлық параметрлер мәндерін белгіленген шектерде уақытында сақтау қасиеті [48-50].

Сенімділік көрсеткіші – объектінің белгілі бір уақыт (жұмыс уақыты) аралығында үздіксіз жұмыс қабілеттілігін сақтау қасиеті. Әртүрлі AtmelAVR микроконтроллердің 2000 сағат үздіксіз жұмыс істеу қабілеттілігі есептелінді. Алдымен ESP32 микроконтроллердің ақаусыз жұмыс істеу ықтималдығы есептелінді. ESP32 микроконтроллердің негізгі көрсеткіштері 1.6 – кестеде көрсетілген.

Кесте 1.6 - ESP32 микроконтроллердің негізгі көрсеткіштері

қоректену кернеуі, В	3,6
тұтыну тоғы, А	20
қоршаған ортаның температурасы, °С	40 °С –дан + 85 °С дейін
бұзылуға дейінгі уақыты, сағ	2000

t уақыт аралығындағы *n* кездейсоқ оқиғалардың (істен шығулардың) болу ықтималдығы 1.1 формуламен анықталады:

$$P_n(t) = \frac{a^n}{n!} e^{-a}, \quad (1.1)$$

мұндағы, $a = \lambda t, t$ уақыт аралығындағы істен шығудың орташа саны.

Сенімділік көрсеткіштерін есептеу үшін жүйенің құрылымдық компоненттері істен шыққанға дейінгі орташа жұмыс уақыты қабылданады: ESP32 микроконтроллері = 17 200 сағ.

ESP32 Микроконтроллердің істен шығу қарқындылығы:

$\lambda = \frac{1}{17200} = 0{,}58 * 10^{-5}\,\frac{1}{\text{сағ}}$ $\lambda_i = \frac{1}{T_i}$.

Тоқтаусыз жұмыс істеу ықтималдығы: $P(t) = e^{-\lambda t}$.

$P(t) = 2{,}718^{(-0{,}58*10^5)*2000} = 0{,}98$,

$P(t) = 0{,}98 * 100\% = 98\%$.

Егер t = 2000 сағ. уақыт ішінде 0-2 істен шығу ықтималдығын анықтасақ,

$\lambda = 0{,}58 * 10^{-5}\,\frac{1}{\text{сағ}}$.

1. t уақытындағы істен шығудың орташа саны : $a = \lambda t = 0{,}01163$;

2. Істен шығудың болмау ықтималдығы $P_0(2000) = 2{,}718^{-0{,}01163} = 0{,}988$;

3. Бір істен шығу ықтималдығы: $P_1(2000) = \frac{0{,}01163^1}{1} 2{,}718^{-0{,}01163} = 0{,}011$;

4. Екі істен шығу ықтималдығы : $P_2(2000) = \frac{0{,}01163^2}{2} 2{,}718^{-0{,}01163} = 0{,}00007$.

ESP32 микроконтроллерінің сенімділігін бағалау нәтижесінде 2000 сағат ішінде ақаусыз жұмыс істеу ықтималдығы 98% құрады, бұл дұрыс пайдаланған кезде аталған құрылғының жоғары сенімділігін көрсетеді.

Әртүрлі AtmelAVR микроконтроллердің (ATmega2560, ATmega32u4, Atmega328, DFRobot Beetle BLE, ESP32, ATtiny44) 2000 сағат аралығында істен шығу қарқындылығы, тоқтаусыз жұмыс істеу ықтималдығы, t уақытындағы істен шығудың орташа саны, бір және екі істен шығу ықтималдығы есептелініп, алынған нәтиже 1.7 – кестеде көрсетілді.

Кесте 1.7 - Әртүрлі AtmelAVR микроконтроллердің сенімділігі

	ATmega2560	ATmega32u4	Atmega328	DFRobot Beetle BLE	ESP32	ATtiny44
T_0	90000 сағ	80000 сағ	75000 сағ	60000 сағ	95000 сағ	30000 сағ
МК-дің істен шығу қарқындылығы $\lambda_i = \frac{1}{T_i}$	1,1 $* 10^{-5}\,\frac{1}{\text{сағ}}$	12,5 $* 10^{-6}\,\frac{1}{\text{сағ}}$	13,3 $* 10^{-5}\,\frac{1}{\text{сағ}}$	16,7 $* 10^{-6}\,\frac{1}{\text{сағ}}$	1,5 $* 10^{-5}\,\frac{1}{\text{сағ}}$	3,3 $* 10^{-5}\,\frac{1}{\text{сағ}}$
Тоқтаусыз жұмыс істеу ықтималдығы $P(t) = e^{-\lambda t}$ $t = 2000\text{сағ}$	0,978	0,975	0,969	0,962	0,983	0,932
t уақытын-	0,02222	0,025	0,0267	0,033	0,021	0,067

дағы істен шығудың орташа саны $a = \lambda_i t$						
Бір істен шығу ықтималдығы: $P_1(2000)$	$\frac{0{,}02222^{1}}{1} e^{-0}$ $= 0{,}217$	$\frac{0{,}025^{1}}{1} e^{-0{,}0}$ $= 0{,}024$	$\frac{0{,}026^{1}}{1} e^{-}$ $= 0{,}025$	$\frac{0.033^{1}}{1} e^{-0.03}$ $= 0{,}031$	$\frac{0{,}021^{1}}{1} e^{-0.021}$ $= 0{,}2$	$\frac{0{,}067^{1}}{1} e^{-0.06}$ $= 0{,}062$
Екі істен шығу ықтималдығы: $P_2(2000)$	$\frac{0{,}02222^{2}}{2} e^{-0}$ $= 0{,}0002$	$\frac{0{,}025^{2}}{2} e^{-0{,}0}$ $= 0{,}00003$	$\frac{0{,}026^{2}}{2} e^{-0{,}0}$ $= 0{,}000032$	$\frac{0.033^{2}}{2} e^{-0.03}$ $= 0{,}0005$	$\frac{0{,}021^{2}}{2} e^{-0.021}$ $= 0{,}00021$	$\frac{0{,}067^{2}}{2} e^{-0.06}$ $= 0{,}002$

AtmelAVR микроконтроллерінің сенімділігін бағалау нәтижесінде 2000 сағат ішінде ақаусыз жұмыс істеу ықтималдығы ATtiny44 қарағанда ATmega2560, ATmega32u4, Atmega328, DFRobot Beetle BLE, ESP32 дұрыс пайдаланған кезде бұл құрылғылардың жоғары сенімділігін көрсетті, осыған байланысты аталған микроконтроллер таңдалынып алынды.

1.3 IoT құрылғылар кластеріндегі деректерді қорғауға арналған әдістер

1.3.1 Гомоморфты шифрлау криптожүйесі

Кәзіргі таңдағы криптожүйенің жаңа бағыттарының бірі - гомоморфты криптография. Оның айрықша ерекшелігі – криптографияның бұл түрі шифрланған деректерді алдын ала құпия кілтсіз өңдеуге мүмкіндік береді, осылайша шифрланған деректермен жасалатын операциялардың нәтижесі шифрды шешкеннен кейін ашық деректердегі операциялардың нәтижесіне тең болады [51]. Бұл криптография мәселелерінің бірі – жалпы сеанс кілттерін генерациялау, сақтау және таратуды шешеді. Бұл деректер қауіпсіздігінің деңгейін арттырады – сервер шифрланған деректерді алады, оларды өңдейді және шифрланған нәтижені қайтарады, ал ашық деректер мен шифрлау кілттері желілік өзара әрекеттесу кезінде қауіпсіз сегменттен шықпайды.

Толық гомоморфты шифрлау идеясын алғаш рет 1978 жылы криптографиялық RSA ашық кілт алгоритмін Рональд Ривест, Ади Шамир мен Майкл Дертузоспен бірге ұсынған [52]. Гомоморфты шифрлау-шифрланған деректерді ерікті есептеуді жүзеге асыратын шифрлау түрі. Алайда, бастапқы кезеңдерде мұндай шифрлаумен криптожүйені құру әрекеттері сәтсіз болды. Көптеген жылдар бойы толық гомоморфты шифрлау мүмкін екендігі түсініксіз болды, дегенмен мұндай жүйені құру әрекеттері бірнеше рет жасалды. Мысалы,

1982 жылы Шафи Голдвассер мен Сильвио Микали [53] ұсынған криптографиялық төзімділіктің жоғары деңгейіне ие болды, бірақ жартылай гомоморфты (гомоморфты тек қосу арқылы) және тек бір битті шифрлай алды.

Ең тиімді және толық гомоморфты криптографиялық жүйені құру жабық есептеулердің аутсорсингінде практикалық іске асыруды қамтамасыз ете алады, мысалы бұлтты есептеулер болуы мүмкін. Гомоморфты шифрлауды қолдану әр нақты қызметке мәліметтер бермей, әртүрлі қызметтерді біріктіре алады.

Гомоморфтылық қасиет көптеген криптографиялық жүйелерде қолданылады, атап айтқанда қауіпсіз дауыс беру жүйелерінде, соқтығысуларға төзімді хэш-функциялардың барлық түрлерінде, іздеу жүйелерінің жабық ақпаратын құруға және бұлтты есептеулерде. Гомоморфтылық қасиет өңделген деректердің құпиялылығына кепілдік береді.

Қазіргі уақытта гомоморфты шифрлау саласында белсенді зерттеулер жүргізілуде. Оның дамуының негізгі бағыттары ретінде келесілерді атап айтуға болады:

Біріншіден, өзгермейтін матрицалық көпмүшелерді қолдана отырып, симметриялы толық гомоморфты шифрлауды жасау. Ресей криптограф ғалымы Ф.Б. Буртыка осы салада жұмыс істейді, ол шифрлауды екі раундта жүргізуді ұсынды: бірінші қадамда сақинасының элементтері болып табылатын ашық мәтіндер алынады, екінші қадамда олар құпия векторды қолдана отырып матрицаларға кодталады, үшінші қадамда бұл матрицалар құпия өзгермейтін матрицалық көпмүшені қолдана отырып, матрицалық көпмүшелерде көрсетіледі. Ал, кері шифрлау екі раундта да іске асырылады [54-55].

Екіншіден, сандарды факторизациялау есебіне негізделген симметриялы толық гомоморфты сызықтық криптожүйелерді дамыту. Осы бағытта жұмыс істейтін ресейлік ғалымдардың ішінде А.В. Трепачева [56], П. К. Бабенканы [57] атауға болады. Бұл жүйелердің криптографиялық тұрақтылығы үлкен сандарды факторизациялау мәселесін шешудің күрделілігін қолданумен негізделген.

Үшіншіден, гомоморфты шифрлаудың шекті жүйелерін әзірлеу және бұлтты есептеулерде ақпаратты қорғау. Бұл бағыттағы зерттеулерді Н.П. Варновский, С.А.Мартишин, М.В. Храпченко, А.В. Шакуров айналысады. Нәтижесінде қосымша ашық кілтті қажет етпейтін және криптографиялық арнайы серверлерде орындайтын аса тиімді емес және проблемалық қайта шифрлау процедурасын (bootstrapping) алмастыратын жүйе алынды [58].

Ресей Федерациясында гомоморфты шифрлауды дамытудың төртінші бағыты-қытайлық қалдық теоремасы негізінде бұлтты ортада есептеу құпиялылығын қамтамасыз ету әдістерін зерттеу. Бұл бағытта Н.И. Червяков, Н.Н. Кучеров жұмыс істейді, олар бұлтты есептеулердің қауіпсіздігін қамтамасыз ету үшін құпияны бөлудің шекті схемаларының қолданылуын, сондай-ақ Шамир құпиясын бөлу схемасының қолданылмауын зерттейді [59].

Осылайша, қазіргі уақытта Ресейде гомоморфты шифрлауды дамытудың келесі төрт бағытын бөліп көрсетуге болады: өзгермейтін матрицалық көпмүшелерді қолдана отырып, симметриялы толық гомоморфты шифрлауды

дамыту, сандарды факторизациялау есебіне негізделген симметриялы толық гомоморфты сызықтық криптожүйелерді дамыту, бұлтты есептеулерде гомоморфты шифрлаудың шекті жүйелерін және ақпаратты қорғауды дамыту, сонымен қатар есептеу құпиялылығын қамтамасыз ету әдістерін зерттеу.

XX ғасырдың соңында жартылай гомоморфты криптографиялық жүйелердің көптеген нұсқалары жарық көрді; олардың ішіндегі ең танымалы-көбейту операциясы бойынша гомоморфты RSA жүйесі.

Алғашқы толық гомоморфты криптографиялық жүйелер XXI ғасырдың басында пайда бола бастады. Крейг Джентри өзінің диссертациялық жұмысында [60] бүтін тор теориясына негізделген жүйені енгізуді ұсынды. Жүйенің негізінде деректерге сандық шу қосу идеясы бойынша іске асырылады, шифрланған торларды құпия кілтті білместен қалпына келтіруге мүмкіндік бермейді. Джентри жүйесі деректерді шифрлау, шифрланған деректерге математикалық операциялар жүргізу, кері шифрлаудан тұрады. Есептеулерді орындау кезінде деректердегі сандық шу жоғарылайды, нәтижесінде белгілі бір операцияларды орындағаннан кейін нәтижені дұрыс қалпына келтіру мүмкін болмайды. Джентридің басты жетістігі-оның жұмысы деректерді толық ашпай шуды азайту механизмін сипаттайды, оны жүктеу деп атайды. Осылайша, осындай түрлендірулерді қолдана отырып, шексіз көптеген қосу және көбейту амалдарын орындауға болады.

Мұнда гомоморфты шифрлау әдісі қолданылады, өйткені ол деректерді оқу кезінде алынған математикалық есептеулерді орындай алады. Бұлтты есептеулерде құпия деректерді шифрланған форматта сақтай отырып, бірақ егер шифрланған деректер үшін қандай-да бір есептеулер жүргізу керек болса, кілт бұлтты қызмет жеткізушілеріне берілуі керек, бұл деректерді пайдалануға әкелуі мүмкін. Бұлтты қызмет жеткізушілерімен кілтті бөліспеу үшін гомоморфты шифрлау әдісі қолданылады. Есептеулер іздеу, сұрыптау, қосу, көбейту, шифрланған мәтін үшін орындалады [61].

Көптеген криптографиялардың ішінде гомоморфты шифрлау ерекше өнімділігімен ғалымдардың назарын аударды [62]. Кәдімгі криптография шифрланған деректерде есептеулерді тікелей орындай алмайды, бірақ гомоморфты шифрлау бұл уақытта гомоморфты шифрлау жұмысының нәтижелері автоматты түрде шифрлануы мүмкін. Гомоморфты шифрлауды қолдану перспективасы қауіпсіз көп партиялы есептеу, электронды дауыс беру, шифрланған мәтінді іздеу, шифрланған поштаны сүзу, мобильді шифр сияқты салаларда кең қолданылады. Қорытындылай келе, қауіпсіздікті талдау одан әрі зерттеу жолын тексереді және көрсетеді.

Осы саладағы зерттеулердің басталуы 1978 жылы Рональд Линн Ривест, Леонард Адлеман және Михаил Дертузос «Құпиялылықты сақтайтын деректер банктері мен гомоморфизмдер туралы» мақала жариялаған кезде басталды [63]. Банктегі пайдаланушы деректерінің құпиялылығын сақтау мәселесін қарастыра отырып, ғалымдар шифрланған мәліметтермен аралық шифрсыз жұмыс істеу үшін арнайы «Құпиялылық гомоморфизмін» қолдануды және осылайша жабдықты үнемдеуді және деректерді сақтау үшін басқа фирмалардың

ресурстарын пайдалануды жалғастыруды ұсынды. Гомоморфты шифрлау тарихындағы осы мақаланың маңызды нәтижесі іс жүзінде тиімді жүзеге асырылатын толық гомоморфты жүйені құрудың ашық математикалық міндеті болды.

S алгебрасының тасымалдаушы жиынтығынан, $\{f_1, f_2 ...\}$ алгебралық амалдар, $\{p_1, p_2, ...\}$ предикаттар және $\{s_1, s_2, ...\}$ константалар жиынтығынан құрылған алгебралық жүйе болсын. Болашақты осындай жүйені $A =< S; f_1, f_2, ...; p_1, p_2, ...; s_1, s_2, ... >$ деп белгіленеді.

Мысалы, осы жазбада бүтін сандар үшін дәстүрлі алгебралық жүйе $<Z, +, -, *, \div, \leq,, 0, 1>$ осы сияқты көрсетіледі. U пайдаланушысының алгебралық жүйесі және C қашықтағы жүйенің алгебралық жүйесі келесідей болсын:

$U =< S; f_1, f_2, ..., f_k; p_1, p_2, ..., p_l; s_1, s_2, ..., s_m >$,

$C = < S'; f_1', f_2', ..., f_k'; p_1', p_2', ..., p_l'; s_1', s_2', ..., s_m' >$.

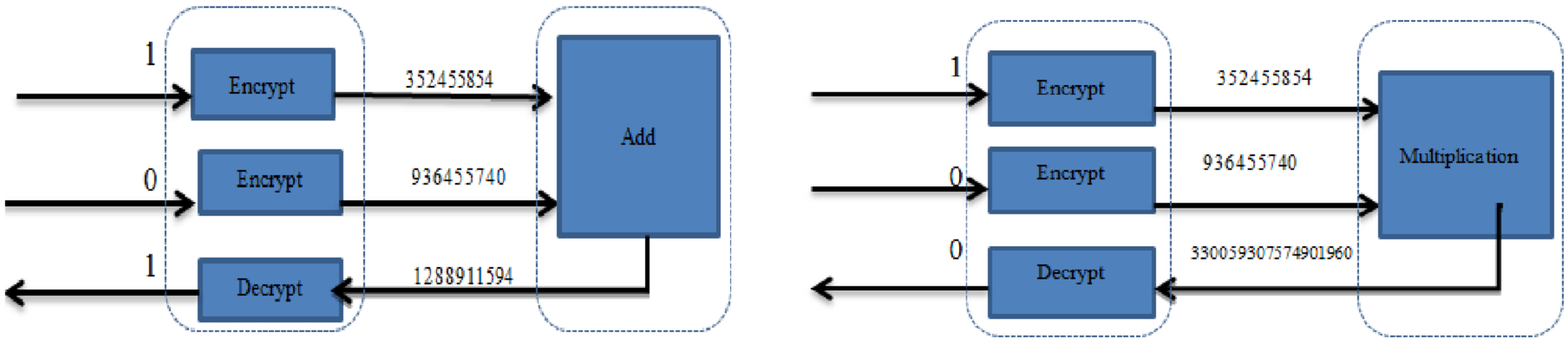
Бұл жағдайда кері шифрлау келесідей түрде болады:

$\varphi: S' \rightarrow S$ шифрлау $\varphi^{-1}: S \rightarrow S'$ түрде көрсетіледі. Пайдаланушыда $\{d_1, d_2, ...\}$ ақпарат жиынтығы бар, мұндағы $d_i \in S \; \forall_i$, ол C жүйесінен қандай да бір операциялар жасағысы келеді. Алайда жіберілетін ақпаратты алдын ала φ^{-1} шифрлау білімін қолдана отырып, $\{\varphi^{-1}(d_1), \varphi^{-1}(d_2), ...\}$ жиынтығын алу арқылы қорғау керек.

Бұл кезеңде шифрлау гомоморфтығын анықтау қажет. Гомоморфты шифрлау ашық мәтіндерге қатысты кез-келген алгебралық операцияларға қатысты гомоморфизмнің қосымша талаптарын қанағаттандыратын шифрлау функциясы болып табылатын қарапайым криптография деп аталады [64]. Гомоморфты криптожүйенің негізі екі түрі бар:

1. Жартылай гомоморфты криптожүйесі – мұнде көбейту немесе азайту екі арифметикалық амалдың біреуіне ған қатысты гомоморфты;

2. Толығымен гомоморфты – қосу мен көбейту арифметикалық операцияларына қатысты гомоморфты. Толық гомоморфты шифрлау арифметикалық операциялары 1.14 - суретте көрсетілген.



а) ә)

Сурет 1.14 – а) Шифрланған деректерді гомоморфты қосу ә) Шифрланған деректерді гомоморфты көбейту

1.3.2 Гомоморфты шифрлау криптожүйелерін AtmelAVR микроконтроллерінде зерттеу

ТГШ негізінде деректерді қорғау – шифрланған деректерді алдымен кері шифрлаймай-ақ есептеуге мүмкіндік беретін қауіпсіздіктің жаңа түрі. Дегенмен, практикалық ТГШ шешімі бүгінгі күні іске асыру үшін қол жетімді емес. Тиімді және қауіпсіздік компонентін ескере отырып, процесс уақытын талдау үшін ең танымал ассиметриялық және симметриялық гомоморфты шифрлау алгоритмдері: Бенало [65], Эль-Гамаль [66], RSA [67], Пэйе [68], Джентридің биттік гомоморфты шифрлау [69], А.Абрамовтың рационал коэффициенттері бар көпмүшелер өрісіндегі гомоморфты шифрлану [70] және С. Ф. Кренделевтің айнымалысы бар көпмүшелер өрісіндегі гомоморфты шифрлау [71] зерттелді. Әрбір алгоритмдердің гомоморфтық белгілері мен кемшіліктері мен ұқсастықтары зерттелді. Сонымен қатар, әрбір алгоритмдер үшін шифрлау, кері шифрлау уақыттары есептеліп салыстырулар жүргізілді. Шифрлау және кері шифрлау операциясының өнімділігін салыстыруға арналған тестілеу 8 итерация үшін орташа жұмыс уақыты есептелінді. Эксперименталды нәтижесі 1.8-кестеде көрсетілген.

Кесте 1.8 – 40 мб өлшемді файл бойынша салыстыру

Криптожүйе	Бенало	Эль-Гамаль	RSA	Пэйе	Джентри	Абрамов	Кренделев
Кілт генерациясы (мс)	23,67	10,808	12,25	23,31	8,51	15,03	20,12
Шифрлеу уақыты (мс)	6772,98	4983,26	2133,65	1672,12	13108,17	1048,69	751,84
Кері шифрлау уақыты (мс)	2736,36	3412,96	1391,59	1336,23	4098,52	530,52	403,96
Деректер жадын пайдалану (байт)	204	202	206	201	216	178	175
Бағдарлама жадын пайдалану (байт)	1689	1382	1802	1567	2976	992	867
Ток күші (Вт)	1,15	1,03	1,21	1,12	2,24	0,63	0,57
Кернеу (Вольт)	2,584	2,508	2,607	2,520	2,897	2,361	2,336
Қуат (Вт)	1,15	1,03	1,21	1,12	2,24	0,63	0,57
ГШ Категориялары	ЖГШ	ЖГШ	ЖГШ	ЖГШ	ТГШ	ТГШ	ТГШ
Гомоморфтық белгілер	Қосу	Көбейту	Көбейту	Қосу	Қосу және көбейту	Қосу және көбейту	Қосу және көбейту

Ұқсастықтары	Гомоморфтылығы

Салыстырулар негізінде А.Абрамов ұсынған рационал коэффициенттері бар көпмүшелер өрісіндегі гомоморфты шифрлау, С. Ф. Кренделев ұсынған айнымалысы бар көпмүшелер өрісіндегі гомоморфты шифрлау және Пәйе алгоритмдері AtmelAVR микроконтроллерінде уақыт бойынша тиімді екені анықталды.

.

2 ATMELAVR МИКРОКОНТРОЛЛЕРІНДЕ ҚОЛДАНЫЛАТЫН ГОМОМОРФТЫ ШИФРЛАУ АЛГОРИТМІН ЖЕТІЛДІРУ

2.1 Айнымалысы бар көпмүшелер сақинасындағы гомоморфты шифрлау

Криптография ежелден бері жіберілген деректерді құпия сақтай отырып, қауіпті ортада ақпараттың қауіпсіз берілуін қамтамасыз етеді. Бұл ғылым үнемі дамып отырады.

Гомоморфты шифрлау, оның барлық артықшылықтарына қарамастан, болашақта шешуді қажет ететін бірқатар мәселелерді тудырады:

- берілетін деректердің тұтастығын қамтамасыз ету қажеттілігі;
- шифрланған деректермен операциялардың жоғары күрделілігі;
- деректермен операцияларды орындағаннан кейін коэффициенттердің жылдам өсуі.

Қазіргі таңда гомоморфты шифрлаудың дамуы мен жетілдірілуі өзекті мәселе болып табылады [72]. Осы саладағы алгоритмдер мен схемаларды жақсарту оны қазіргі уақытта симметриялық шифрлау қолданылатын көптеген салаларда қолданылады:

- бұлтты есептеу;
- фотосуреттерді бұлттық өңдеу;
- электрондық дауыс беру (сайлау);
- ақпаратты қауіпсіз іздеу.

2011 жылы С.Ф Кренделев. өзінің толық гомоморфты криптожүйесін ұсынды, оның жұмысы айнымалысы бар көпмүшелер өрісіндегі гомоморфизмге негізделген. Әрбір $a \in Z_n$ саны $a(x) = a_0 + a_1x + ... + a_kx^k$ көпмүшесімен байланысты, мұнда k және a_i кездейсоқ таңдалады. Құрылымы бойынша a_0 және b_0 сандарының екі көпмүшелік кескіні үшін олардың $a(x) + b(x)$ қосындысының және $a(x)*b(x)$ көбейтіндісінің бос мүшесі $a(0) + b(0)$ және a(0)*b(0).

Сонда $\varphi : Z_n[x] \rightarrow Z_n[y]$, $x = c_0 + c_1y + ... + c_ty^t = \varphi(y)$ - қосу мен көбейтуді сақтайтын гомоморфизм болып табылады. С.Ф. Кренделевтің іске асырылуы Джентриге қарағанда тиімдірек. Сонымен қатар, оның бірқатар кемшіліктері бар:

1. Көпмүшелердің дәрежелерінің шексіз өсуі есептеудің тиімсіздігіне әкелуі мүмкін;

2. Барлық амалдар шын мәнінде бос мүшелерде орындалғанымен, жадта сақтау және үлкен дәрежелі көпмүшеліктер бойынша есептеулер жүргізу керек;

3. С.Ф. Кренделев жүйесінде шифрланған деректерге бөлу және азайту операциялары орындалмаған.

Диссертациялық жұмыста С.Ф. Кренделев ұсынған жүйеге шифрланған деректерге бөлу және азайту операциялары қосылды.

Есептің математикалық тұжырымы:

«Гомоморфты шифрлау» ұғымы енгізілген «privacy homomorphism» [73] тұжырымдамасына негізделген:

Анықтама 1 (Құпиялылық гомоморфизмі). Екі алгебралық жүйе берілсін

$$U =< S; f_1..., f_k; p_1,..., p_l; s_1,.., s_m >,$$

$$U =< S'; f_1'..., f_k'; p_1',..., p_l'; s_1',.., s_m' >,$$

мұндағы, S және S'-жиындар; $f_1,...,f_k$, $f_1',...,f_k'$-функциялар; $p_1,...,p_l$,, $p_1',...,p_l'$-предикаттар, тиісінше S және S' жиындарында анықталған, а $\{s_1,...,s_m\} \subset S$, $\{s_1',...,s_m'\} \subset S'$- белгілі тұрақтылар.

$\varphi : S \to S'$ кері функция бар делік, онда φ privacy homomorphism деп аталады, егер келесі шарттар орындалса:

$$\forall i \in \{1,...,k\}: f_i(a_1,...,a_n) = \varphi^{-1}\left(f_i'(\varphi(a_1),...,\varphi(a_n),)\right)$$

$$\forall i \in \{1,...,l\}: p_i(a_1,...,a_n) = p_i'(\varphi(a_1),...,\varphi(a_n),)$$

$$\forall i \in \{1,...,m\}: \varphi(s_i) = s_i'$$

Көпмүшелерге операциялар.

Толық гомоморфты шифрлауды құру үшін қажетті бірінші ішесепті қарастырайық – бүтін сандар мен көпмүшелердің сақиналары арасындағы сәйкестікті орнату, сондай-ақ олардағы қосу және көбейту амалдары.

Айталық екі көпмүше берілсін $a(x), b(x) \in Z[x]$:

$$a(x) = a_0 + a_1 x + ... + a_n x^n,$$

$$b(x) = b_0 + b_1 x + ... + b_n x^n,$$

мұндағы, $n > 0$. Жалпылықты жоғалтпай, бұл көпмүшелердің бірдей дәрежелері бар деп болжауға болады, әйтпесе, төменгі дәрежелі көпмүшені нөлдік коэффициенттері бар жоғары мүшелердің қажетті санымен толықтыруға болады.

Көпмүшелердің сақинасындағы қосу және көбейту амалдары келесідей анықталады[74]:

$$a(x) + b(x) = a_0 + b_0 + (a_1 + b_1)x + ... + (a_n + b_n)x^n,$$

$$a(x) * b(x) = a_0 b_0 + (a_0 b_1 + b_0 a_1)x + ... + a_n b_n x^{2n}.$$

Осы формулалардан көрініп тұрғандай, алынған көпмүшенің бос мүшесі сәйкесінше бастапқы көпмүшелердің бос мүшелерінің қосындысы мен көбейтіндісіне тең болады.

Z сақинасынан $Z[x]$ көпмүшеліктерінің сақинасына дейінгі кескінді құру мәселесінің шешімі ретінде келесі алгоритм қолданылды:

Алгоритм 2.1 Көпмүшені бүтін санға сәйкестендіру.

1) Айталық $z \in Z$ — көпмүшеге сәйкес келетін қандай да бір бүтін сан. Айталық сан берілсін бұл $n > 0$ —көпмүшенің дәрежесі, $x_0 = \frac{p}{q}, x_0 \in Q$

2) $a_0, a_1, ...a_n \in Z$ коэффициенттері кездейсоқ таңдалынатын $f(x) = a_0 + a_1 x + ... + a_n x^n$ көпмүше құрылады.

3) $f(x_0) = f\left(\frac{p}{q}\right) = a_0 + a_1\left(\frac{p}{q}\right) + ... + a_n\left(\frac{p}{q}\right)^n$ есептеледі, осыдан $q^n f\left(\frac{p}{q}\right) = q^n a_0 + q^{n-1} p a_1 + ... + p^n a_n$, $q^n f\left(\frac{p}{q}\right) \in Z$ алынады.

4) z санға сәйкес келетін $g_z(x)=q^n f(x)-q^n f\left(\frac{p}{q}\right)+z$ көпмүше есептелінеді.

Алгоритм 2.2 Бүтін санды көпмүшеге сәйкестендіру (2.1-ге кері алгоритм).

1) 1-алгоритмнің нәтижесінде алынған $g_z(x)$ көпмүше берілсін.

2) Содан кейін $g_z(x_0)=q^n f(x_0)-q^n f\left(\frac{p}{q}\right)+z=z$ бастапқы сан.

Тұжырым 2.1 Алгоритм 2.1 бойынша құрастырылған $h:Z\to Z[x]$ — бейнелеу. Онда берілген бейнелеу инъективті.

Дәлелдеу коэффициенттерді кездейсоқ таңдаудан айқын түрде келеді. Z сақинасынан көпмүшеліктер сақинасына дейін гомоморфизмді құру.

Енді $\boldsymbol{Z}$ сақинасынан $\boldsymbol{Z[x]}$ сақинасына гомоморфизм құру мәселесін қарастырайық.

Анықтама 2.1 (сақина гомоморфизмі). *f: A-> B* болсын, мұндағы *A* және *B* қосу, көбейту, нөл және бір болатын сақиналар.

Теорема 2.1. 3.1-ші алгоритм бойынша құрылған Z бүтін сандар сақинасы, $Z[x]$ көпмүшелер сақинасы, *P(x)*-бейнелеу берілсін. Сонда *P(x)* бұл $P:Z\to Z[x]$ гомоморфизмі.

Дәлелдеу: $z_1,z_2\in Z$. (1) қасиет орындалатынын көрсетейік, яғни

$P(z_1+_z z_2)=P(z_1)+_{z[x]}P(z_2)$.

$$P(z_1)+_{z[x]}P(z_2)=q^n f_1(x)-q^n f_1\left(\frac{p}{q}\right)+z_1+q^n f_2(x)-q^n f_2\left(\frac{p}{q}\right)+z_2=$$

$$=q^n a_0+q^n a_1 x+\ldots+q^n a_n x^n-\left(q^n a_0+q^{n-1}pa_1+\ldots+p^n a_n\right)+z_1+$$

$$+q^n b_0+q^n b_1 x+\ldots+q^n b_n x^n-\left(q^n b_0+q^{n-1}pb_1+\ldots+p^n b_n\right)+z_2=$$

$$=\left(a_1+b_1\right)\left(q^n x-q^{n-1}p\right)+\ldots+\left(a_n+b_n\right)\left(q^n x-p^n\right)+z_1+z_2=$$

$$=q^n c_0+q^n c_1 x+\ldots+q^n c_n x^n-\left(q^n c_0+q^{n-1}pc_1+\ldots+p^n c_n\right)+z_1+z_2$$

$$=P(z_1+_z z_2)$$

мұндағы, $c_i=a_i+b_i, i\in\{0\ldots n\}$

Жоғарыда келтірілген есептеулер негізінде шифрлау және кері шифрлау алгоритмін құруға болады.

С.Ф. Кренделев жүйесіне шифрланған деректерге бөлу операциясын енгізу: Анықтама 2.1 Айталық Z – қандай да бір өріс. Z өрісіндегі бір айнымалыдағы көпмүше пішіннің формальды қосындысы болып табылады:

$$f(x)=a_n x^n+a_{n-1}x^{n-1}+a_1 x+a_0, \qquad 2.1$$

мұндағы, $a_i\in Z, i\in\{0,1,\ldots,n\}\, n\in N$.

Ескерету. Z өрісінің кез келген элементі нөлдік дәрежелі көпмүше, нөлдік коэффициенттері бар ерікті дәрежелі көпмүше – нөлдік көпмүше, Z өрісінің

бірлігі – бірлік көпмүше болып саналады және оларды сәйкесінше $\vartheta(x)$ және $E(x)$ арқылы белгіленеді.

Z өрісіндегі бір айнымалыдағы барлық көпмүшелердің жиынында келесі ережелер бойынша көпмүшелерді қосу және көбейту операцияларын анықтауға болады. Айталық $f(x)$ – 3.1 түріндегі көпмүше және $g(x) = b_n x^n + b_{n-1} x^{n-1} + b_1 x + b_0$, мұндағы $b_i \in Z,\ j \in \{0,1,\dots,m\}\, m \in N$ – 2.2 түріндегі көпмүше [75-77].

Анықтама 2.2 [78] $f(x)$ және $g(x)$ көпмүшелердің қосындысын, $h(x) = c_k x^k + c_{k-1} x^{k-1} + c_1 x + c_0$, түріндегі $h(x)$ көпмүшесін айтамыз, мұндағы $c_i = a_i + b_i,\ j \in \{0,1,\dots,k\}, k = max\{n,m\}$
(2.3)

Z өрісіндегі $Z[x]$ көпмүше сақинасында, бүтін сандар сақинасында орындалғандай, көпмүшеліктерге бөлу операциясын анықтауға болады.

Анықтама 2.3 [79] (3.1) түріндегі $f(x)$ көпмүшесі, Z өрісіндегі $f(x) = g(x) * h(x)$ болатындай $h(x)$ көпмүше бар болса, онда $g(x)$ көпмүшесіне бөлінеді.

Есептеуге ыңғайлы болу үшін көпмүшелерді белгілеуде нөлдік коэффициенттер пайда болды.

Көпмүшелердің бөлінгіштігінің кейбір қарапайым қасиеттерін атап өтсек, олар бүтін сандардың бөлінгіштік қасиеттеріне ұқсас.

Қасиет 2.1 Егер бір уақытта $f(x)$ көпмүшесі $g(x)$ – қа бөлінсе және керісінше бөлінетін болса, онда көпмүшелер тек нөлдік дәреже коэффициентімен немесе басқаша айтқанда, жай ғана $c: f(x) = cg(x)$ сандық көбейткішпен ерекшеленеді.

Бұл жағдайда $f(x)$ және $g(x)$ көпмүшелері ассоциативті деп аталады.

Қасиет 2.2 Егер $f(x)$ көпмүшесі $g(x)$ және $h(x)$ екі көпмүшелерінің әрқайсысына бөлінетін болса, онда ол олардың көбейтіндісіне де бөлінеді.

Қасиет 2.3 $Z[x]$ сақинасындағы көпмүшелердің бөлінгіштік қатынасы рефлексивті және транзитивті.

Қасиет 2.4 Егер $f(x)$ көпмүшесі нөлге тең емес $g(x)$ көпмүшесіне бөлінсе, онда кез келген ассациативті $f(x)$ көпмүшесі $g(x)$-қа бөлінетін болады.

Қасиет 2.5. Егер $f(x)$ және $g(x)$ екі көпмүшелерінің әрқайсысы нөлдік емес $h(x)$ көпмүшелігіне бөлінетін болса, онда олардың кез келген сызықтық комбинациясы да $h(x)$- ке бөлінеді [80-81].

Анықтама 2.4 Егер теңдік орындалатын $q(x)$ және $r(x)$ көпмүшеліктерінің жұбы болса, $f(x)$ көпмүшесі, $g(x)$ көпмүшесінің нөлдік емес қалдығына бөлінеді деп аталады:

$$f(x) = g(x) * q(x) + r(x), \tag{2.4}$$

мұндағы, $r(x) < g(x)$ дәрежесі немесе $r(x) = \vartheta(x)$.

Көпмүшелер сақинасында, сондай-ақ бүтін сандар сақинасында қалдықпен бөлу теоремасын дәлелдеуге болады.

Теорема 2.3 $f(x)$ және $g(x)$, көпмүшелерінің кез келген жұбы үшін, мұндағы $g(x)$ - нөлдік емес көпмүше болса, (2.1) шарттарын қанағаттандыратын $q(x)$ және $r(x)$, көпмүшелерінің жалғыз жұбы болады.

Теорема 2.4 $x = c$: $f(x) = (x - c) * q(x) + f(c)$ жағдайда, $f(x)$ көпмүшені $x - c$ екі мүшеге бөлудің қалдығы $f(x)$ көпмүшенің мәніне тең [81].

$f(x)$ көпмүшесі келесі түрде жазылсын::

$$f(x) = a_0x^n + a_1x^{n-1} + a_{n-1} + a_n \quad (a_0 \neq 0) \tag{2.5}$$

Оны келесі түрде көрсетейік:

$$f(x) = (x - c) * (b_0x^{n-1} + b_1x^{n-2} + b_{n-2}x + b_{n-1}) + r \tag{2.6}$$

Теңдіктің оң жағындағы жақшаларды ойша ашып, бірдей дәрежедегі коэффициенттерді (2.5) және (2.6) теңдіктерінде теңестірсек, келселер алынады:

$$\begin{gathered} b_0 = a_0, \\ b_1 = a_1 + cb_0, \\ b_2 = a_2 + cb_1, \\ \ldots\ldots\ldots\ldots\ldots \\ b_k = a_k + cb_{k-1}, \\ \ldots\ldots\ldots\ldots\ldots \\ b_{n-1} = a_{n-1} + cb_{n-2}, \\ \ldots\ldots\ldots\ldots\ldots \\ r = a_n + cb_{n-1}. \end{gathered} \tag{2.7}$$

(2.7) әдетте кесте түрінде жазылады және Горнер схемасы деп аталады [82]. Горнер схемасы 2.1-кесте түрінде көрсетілген.

Кесте 2.1 Горнер схемасы

	a_0	a_1	...	a_k	...	a_n
$x = c$	b_0	$b_1 = a_1 + cb_0$	...	$b_k = a_k + cb_{k-1}$	...	$r = a_n + cb_{n-1}$

Әлбетте, $r = f(c)$, b_i коэффициенттері $f(x)$ –ті $x - c$-ға бөлу кезіндегі $q(x)$ бөліндісінің коэффициенттері болып табылады.

Теорема 2.5 Z өрісіндегі кез келген с саны үшін n дәрежелі (2.5) түріндегі $f(x)$ көпмүшесі әрқашан және оның үстіне бірегей түрде келесі түрде ұсынылуы мүмкін:

$$f(x) = b_0 + b_1(x - c) + b_2(x - c)^2 + \cdots + b_n(x - c)^n, b_n \neq 0, \tag{2.8}$$

(2.8) түріндегі бейнелеу $f(x)$ мәнін $x - c$ айырмасының дәрежелерінде жіктеу деп аталады.

Енді Z сақинасынан $Z[x]$ сақинасында гомоморфизм құру мәселесін қарастырайық.

Анықтама 2.1 (сақина гомоморфизмі). *f: A-> B* болсын, мұндағы *A* және *B* қосу, көбейту, нөл және бір болатын сақиналар. Егер келесі шарттар орындалса, *f* - сақиналы гомоморфизм [83-86]:

$$f(a +_A b) = f(a) +_B f(b)\ [12] \quad (2.9)$$

$$f(a -_A b) = f(a) -_B f(b) \quad (2.10)$$

$$f(a *_A b) = f(a) *_B f(b)\ [12] \quad (2.11)$$

$$f(a /_A b) = f(a) /_B f(b) \quad (2.12)$$

$$f(0_A) = 0_B\ [12] \quad (2.13)$$

$$f(1_A) = 1_B\ [12] \quad (2.14)$$

Z бүтін сандар сақинасы, *Z[x]* көпмүше сақинасы және 2.1 – алгоритм бойынша құрастырылған $P(x)$ берілсін. Сонда $P(x)$, $P: Z \to Z[x]$ гомоморфизмі.

Шифрлау және кері шифрлау алгоритмі

Кілтті генерациялау. x_0 берілген шифрдың құпия кілті, айталық z_1, $z_2 \in Z$, $z_1 > z_2$ болсын, (3.12) қасиетінің орындалатынын көрсетейік, яғни:

$$P(z_1 /_z z_2) = P(z_1) /_{z[x]} P(z_2)$$

$$P(z_1) = q^n f_1(x) - q^n f_1\left(\frac{p}{q}\right) + z_1 = q^n a_0 + q^n a_1 x + q^n a_2 x^2 + \cdots - q^n a_0 -$$

$$-q^{n-1} a_1 p - q^{n-2} a_2 p^2 - \cdots + z_1 = a_1 q^{n-1}(qx - p) + a_2 q^{n-2}(q^2 x^2 - p^2) + \cdots z_1$$

$$\frac{P(z_1)}{P(z_1)} = \frac{a_1 q^{n-1}(qx - p) + a_2 q^{n-2}(q^2 x^2 - p^2) + \cdots + z_1}{b_1 q^{n-1}(qx - p) + b_2 q^{n-2}(q^2 x^2 - p^2) + \cdots + z_2}$$

$$P\left(\frac{z_1}{z_2}\right) = c_1 q^{n-1}(qx - p) + c_2 q^{n-2}(q^2 x^2 - p^2) + \cdots + \frac{z_1}{z_2}$$

$\frac{P(z_1)}{P(z_1)} = P\left(\frac{z_1}{z_2}\right)$ үшін c_i – ді $c_0 = \forall$ деп таңдауға болады.

$c_i = \frac{a_i z_2 - b_i z_1}{z_2 * P_{[x]}(z_2)}$, $i = \overline{1, n}$ $\qquad \frac{a_i z_2 - b_i z_1}{z_2 * P_{[x=p/q]} z_2}$

$$\frac{a_1 q^{n-1}(qx - p) + a_2 q^{n-2}(q^2 x^2 - p^2) + \cdots + z_1}{b_1 q^{n-1}(qx - p) + b_2 q^{n-2}(q^2 x^2 - p^2) + \cdots + z_2} =$$

$$\frac{a_1 z_2 - b_1 z_1}{z_2 * b_1 q^{n-1}(qx - p) + b_2 q^{n-2}(q^2 x^2 - p^2) + \cdots + z_2} * q^{n-1}(qx - p)$$

$$+ \frac{a_2 z_2 - b_2 z_1}{z_2 * b_1 q^{n-1}(qx - p) + b_2 q^{n-2}(q^2 x^2 - p^2) + \cdots + z_2} * q^{n-2}(q^2 x^2 - p^2) + \cdots + \frac{z_1}{z_2}$$

$$a_1q^{n-1}(qx-p)+a_2q^{n-2}(q^2x^2-p^2)+\cdots+z_1=\frac{a_1z_2-b_1z_1}{z_2}*q^{n-1}(qx-p)+$$

$$+\frac{a_1z_2-b_1z_1}{z_2}*q^{n-2}(q^2x^2-p^2)+\cdots+\frac{z_1}{z_2}*b_2q^{n-2}(q^2x^2-p^2)+\cdots+z_2$$

$$a_1q^{n-1}(qx-p)+a_2q^{n-2}(q^2x^2-p^2)+\cdots+z_1$$

$$=q^{n-1}(qx-p)\left[\frac{a_1z_2-b_1z_1}{z_2}+\frac{z_1}{z_2}*b_1\right]$$

$$+q^{n-2}(q^2x^2-p^2)\left[\frac{a_1z_2-b_1z_1}{z_2}+\frac{z_1}{z_2}*b_2\right]+\cdots+z_1$$

$$a_1q^{n-1}(qx-p)+a_2q^{n-2}(q^2x^2-p^2)+\cdots+z_1$$

$$=a_1q^{n-1}(qx-p)+a_2q^{n-2}(q^2x^2-p^2)+\cdots+z_1$$

$\frac{P(z_1)}{P(z_1)}=P\left(\frac{z_1}{z_2}\right)$ болғандықтан P бейнесі гомоморфизм болып табылады.

Дәлелдеу. $z_1, z_2 \in Z$ болсын, (2.10) қасиетінің орындалатынын көрсетейік, яғни:

$$P(z_1-_z z_2)=P(z_1)-_{z[x]}P(z_2),$$

$$P(z_1)-_{z[x]}P(z_2)=(q^nf_1(x)-q^nf_1\left(\frac{p}{q}\right)+z_1)-(q^nf_2(x)-q^nf_2\left(\frac{p}{q}\right)+z_2=$$

$$=q^na_0+q^na_1x+\cdots+q^na_nx^n-q^na_0-q^{n-1}pa_1-\cdots-p^na_n+z_1-q^nb_0-$$

$$q^nb_1x-\cdots-q^nb_nx^n+q^nb_0+q^{n-1}pb_1+\cdots+p^nb_n-z_2=(a_1-b_1)(q^nx-$$

$$q^{n-1}p)+(a_2-b_2)(q^nx^2-q^{n-2}p^2)+\cdots+(a_n-b_n)(q^nx^n-p^n)+z_1-z_2.$$

Бірініші бөлігіне екінші бөлігі тең екенін көрсетейік:

$$P(z_1-_z z_2)=(q^nf_1(x)-q^nf_1\left(\frac{p}{q}\right)+z_1)-(q^nf_2(x)-q^nf_2\left(\frac{p}{q}\right)+z_2=$$

$$q^n(a_0-b_0)+q^n(a_1-b_1)x+q^n(a_2-b_2)x^2-q^n(a_0-b_0)-q^{n-1}p(a_1-b_1)-$$

$$q^{n-2}p^2(a_2-b_2)+z_1-z_2=(a_1-b_1)(q^nx-q^{n-1}p)+(a_2-b_2)(q^nx^2-$$

$$q^{n-2}p^2)+\cdots+(a_n-b_n)(q^nx^n-p^n)+z_1-z_2.$$

Сондай-ақ, P бейнелеу (3) және (4) қасиеттерге сәйкес келетін 1-ден 1-ге және 0-ден 0-ге дейін салыстыратыны анық. Сондықтан P бейнелеу гомоморфизм болып табылады.

Кілттерді генерациялау. $x_0=\frac{p}{q}$ мұндағы $p\in Z, q\in N$ - кез-келген сандар.

Алгоритм 2.3 (шифрлау). Шифрлау алгоритмі $Enc: Z\to Z[x]$ келесідей көрсетіледі.

1) $z\in Z$ — шифрланатын сан болсын, $z_1 > z_2$, $z_2 \neq 0$;

2) Коэффициенттер $a_0, a_1, \ldots a_n \in Z$ кездейсоқ таңдалатындай көпмүшені тұрғызамыз $f(x)=a_0+a_1x+\ldots+a_nx^n$;

3) $f(x_0)=f\left(\frac{p}{q}\right)=a_0+a_1(\frac{q}{p})+\ldots+a_n(\frac{q}{p})^n$ есептелінеді. Осыдан $q^nf\left(\frac{p}{q}\right)=q^na_0+q^{n-1}pa_1+\ldots+p^na_n$ алынады, мұндағы $q^nf\left(\frac{p}{q}\right)\in Z$;

4) Сонда z үшін шифрлау көпмүшесі келесідей болады:

$$g_z(x)=q^nf(x)-q^nf(x)+z\in Z[x]$$

Алгоритм 2.4 (кері шифрлау). Кері шифрлау алгоритмін қарастырайық $Dec: Z[x] \to Z$ бұл 2.2 алгоритмге кері.

1) Көпмүше $g_{z'}'' \in Z[x]$- шифрланған деректер $x_0 \in Q$ - құпия кілт.

2) Кері шифрлау үшін x_0 нүктесіндегі $g_{z'}''$ есептелінеді.

3) Сонда $z' = g_{z'}''(x_0) \in Z$ -кері шифрланған деректер.

2.1 – теоремаға сәйкес, $z_1, z_2 \in Z$ үшін яғни келесі қасиеттер орындалса, шифрлау гомоморфты:

$$z_1 + z_2 = Dec(Enc(z_1) + Enc(z_2)),$$
$$z_1 * z_2 = Dec(Enc(z_1) * Enc(z_2)).$$

С.Ф. Кренделевтің еңбегінде [72] қосу және көбейту операциясы үшін гомоморфты шифрлау қарастырылды, бұл диссертациялық жұмыста азайту және бөлу операцияларын орындайтын толық гомоморфты шифрлау әдісі ұсынылған.

Жүйе жұмысының мысалы:

Key: $x_0 = \frac{11}{7}$.

$Enc: Z \to Z[x]$.

1) $z_1 = 7, z_2 = 5$;

2) $f_1(x) = 2 + 4x - 6x^2 + 3x^3$, $f_2(x) = 5 - 8x + 6x^2 + 9x^3$;

3) $q^n f_1\left(\frac{p}{q}\right) = 7^3 * 2 + 7^2 * 11 * 4 - 6 * 7 * 11^2 + 3 * 11^3 = 175$;

$q^n f_2\left(\frac{p}{q}\right) = 7^3 * 5 - 8 * 7^2 * 11 + 6 * 7 * 11^2 + 11^3 * 9 = 14464$;

4) z_1 және z_2 үшін шифрланатын көпмүше:

$g_{z_1} = 1029x^3 - 2058x^2 + 1372x - 1060$, $g_{z_2} = 3087x^3 + 2058x^2 - 2744x - 12744$;

Жүйенің азайту және бөлу гомоморфизмі:

$g_{z_1} - g_{z_2} = -2058x^3 - 4116x^2 + 4116x + 11684$;

$\frac{g_{z_1}}{g_{z_2}} = \frac{1}{3} + \frac{-2744x^2 + 2286\frac{2}{3}x + 3188}{3087x^3 + 2058x^2 - 2744x - 12744}$;

$Dec: Z[x] \to Z$.

$g_{z_1}x_0 - g_{z_2}x_0 = -2058\left(\frac{11}{7}\right)^3 - 4116\left(\frac{11}{7}\right)^2 + 4116 * \frac{11}{7} + 11684 = 2$;

$$\frac{g_{z_1}x_0}{g_{z_2}x_0} = \frac{1}{3} + \frac{-2744\left(\frac{11}{7}\right)^2 + 2286\frac{2}{3} * \frac{11}{7} + 3188}{3087\left(\frac{11}{7}\right)^3 + 2058\left(\frac{11}{7}\right)^2 - 2744 * \frac{11}{7} - 12744} = 1{,}4.$$

Жүйенің мәтіндермен жұмыс жасау мысалы:

Шифрлау алгоритмі $Enc: Z \to Z[x]$ келесі түрде көрсетіледі:

1) Шифрланатын сөз «Informatics» болсын

2) Коэффициенттері a_0=8, a_1=-2, a_2=1-> $8-2x+x^2$ көпмүше құрылады

3) Символдарды ASCII кестесіне ауыстырылады

4) Сонда шифрланатын көпмүше z_I. z_n, z_f, z_o, z_r, z_m, z_a, z_t, z_i, z_c, z_s келесі түрде болады:

$$g_{z_I}(x) = 7^2\left(8-2x+x^2\right)-\left(7^2*8+7^{2-1}*5*(-2)+7^0*5^2*(-1)\right)+73 = 49x^2-98x+118;$$

$$g_{z_n}(x) = 7^2\left(8-2x+x^2\right)-\left(7^2*8+7^{2-1}*5*(-2)+7^0*5^2*(-1)\right)+110 = 49x^2-98x+155;$$

$$g_{z_f}(x) = 7^2\left(8-2x+x^2\right)-\left(7^2*8+7^{2-1}*5*(-2)+7^0*5^2*(-1)\right)+102 = 49x^2-98x+147;$$

$$g_{z_o}(x) = 7^2\left(8-2x+x^2\right)-\left(7^2*8+7^{2-1}*5*(-2)+7^0*5^2*(-1)\right)+111 = 49x^2-98x+156;$$

$$g_{z_r}(x) = 7^2\left(8-2x+x^2\right)-\left(7^2*8+7^{2-1}*5*(-2)+7^0*5^2*(-1)\right)+114 = 49x^2-98x+159;$$

$$g_{z_m}(x) = 7^2\left(8-2x+x^2\right)-\left(7^2*8+7^{2-1}*5*(-2)+7^0*5^2*(-1)\right)+109 = 49x^2-98x+154;$$

$$g_{z_a}(x) = 7^2\left(8-2x+x^2\right)-\left(7^2*8+7^{2-1}*5*(-2)+7^0*5^2*(-1)\right)+97 = 49x^2-98x+142;$$

$$g_{z_t}(x) = 7^2\left(8-2x+x^2\right)-\left(7^2*8+7^{2-1}*5*(-2)+7^0*5^2*(-1)\right)+116 = 49x^2-98x+161;$$

$$g_{z_i}(x) = 7^2\left(8-2x+x^2\right)-\left(7^2*8+7^{2-1}*5*(-2)+7^0*5^2*(-1)\right)+105 = 49x^2-98x+150;$$

$$g_{z_c}(x) = 7^2\left(8-2x+x^2\right)-\left(7^2*8+7^{2-1}*5*(-2)+7^0*5^2*(-1)\right)+99 = 49x^2-98x+144;$$

$$g_{z_s}(x) = 7^2\left(8-2x+x^2\right)-\left(7^2*8+7^{2-1}*5*(-2)+7^0*5^2*(-1)\right)+115 = 49x^2-98x+160.$$

Кері шифрлау $Dec: Z[x] \to Z$ келесі түрде көрсетіледі:

$$g''_{z_I} = 49*\left(\frac{25}{49}\right)-98*\left(\frac{5}{7}\right)+118 = 73;$$

$$g''_{z_n} = 49*\left(\frac{25}{49}\right)-98*\frac{5}{7}+155 = 110;$$

$$g''_{z_f} = 49*\left(\frac{25}{49}\right)-98*\frac{5}{7}+147 = 102;$$

$$g''_{z_o} = 49*\left(\frac{25}{49}\right)-98*\frac{5}{7}+156 = 111;$$

$$g''_{z_r} = 49*\left(\frac{25}{49}\right)-98*\frac{5}{7}+159 = 114;$$

$$g''_{z_m} = 49*\left(\frac{25}{49}\right)-98*\frac{5}{7}+154 = 109;$$

$$g''_{z_a} = 49*\left(\frac{25}{49}\right)-98*\frac{5}{7}+142 = 97;$$

$$g''_{z_t} = 49*\left(\frac{25}{49}\right)-98*\frac{5}{7}+161 = 116;$$

$$g''_{z_i} = 49*\left(\frac{25}{49}\right)-98*\frac{5}{7}+150 = 105;$$

$$g''_{z_c} = 49*\left(\frac{25}{49}\right)-98*\frac{5}{7}+144 = 99;$$

$$g''_{z_s} = 49*\left(\frac{25}{49}\right)-98*\frac{5}{7}+160 = 115.$$

Кері шифрланған сандар (73, 110, 102, 111, 114, 109, 97, 116, 105, 99, 115) ASCII кестесіне мәні бойынша аударылады. Бастапқы «Informatics» сөзі алынады.

Айнымалысы бар көпмүшелер сақинасындағы толық гомоморфты шифрлауға қосылған азайту және бөлу операциялардың псевдо коды 2.1 – суретте көрсетілген.

Algorithm 1

$size \leftarrow RANDOM$
$n \leftarrow RANDOM$
$numbers1[size], numbers2[size]$
$a[size][n+1]$
$kopmushe1[size][n+1], kopmushe2[size][n+1]$
$p, q \leftarrow RANDOM$
$x0 \leftarrow p \div q$
$ENCRYPTION(numbers1,numbers2,size,n,a,p,q)$
$GOMOMORPHISM$
-
for $i \leftarrow 0$ to $size$ **do**
    for $j \leftarrow 0$ to $n+1$ **do**
        $gomAzaitu[i][j] \leftarrow (kopmushe1[i][j] - kopmushe2[i][j])$
    end for
end for
/
for $i \leftarrow 0$ to $size$ **do**
    $resKopmushe \leftarrow kopmushe1[i]/kopmushe2[i]$
    $qaldyq \leftarrow kopmushe1[i]\ mod\ kopmushe2[i]$
    $bolu[i] \leftarrow resKopmushe + qaldyq \div kopmushe2[i]$
end for
$DECRYPTION(gomAzaitu,bolu,x0)$

Сурет 2.1 – Ұсынылған азайту және бөлу операцияларының псевдо коды

2.2 Рационал коэффициенттері бар көпмүшелер өрісіндегі гомоморфты шифрлау

2015 жылы А. Абрамов рационал коэффициенттері бар көпмүшеліктер өрісіндегі гомоморфизмге негізделген жүйені[70] ұсынды.

Сонымен, гомоморфтық жүйенің анықтамасы бойынша φ түрлендіру C-дан U-ға гомоморфизм болу қажет.

Пайдаланушы $f_1(d_1, d_2)$ мәнін есептегісі келеді. Содан кейін ол жүйеден $f_1'(\varphi^{-1}(d_1), \varphi^{-1}(d_2))$ мәнін есептеуді сұрайды. φ функциясы гомоморфизм болғандықтан, ф $\varphi(f_1'(\varphi^{-1}(d_1), \varphi^{-1}(d_2))) = f_1(d_1, d_2)$ ақиқат болғанына байланысты, жүйенің гомоморфты мәні орындалады.

Сондай-ақ, C жүйелері мен φ функциясына төмендегідей шектеулер қойылады:

1. φ және φ^{-1} функциялары қарапайым және жылдам есептелетін болуы қажет;

2. C -дан f_i' мен p_i' рационалды түрде есептелуі керек;

3. d_i үшін ұсынылған жадыдан $\varphi^{-1}(d_i)$, шифрланған мәнді сақтауға қажетті жады аспауы керек;

4. $\varphi^{-1}(d_i)$ мәндерінің үлкен жиынын ескере отырып, φ функциясын дәл болжау мүмкін емес;

5. d_i мәндерінің жиынын және $\varphi^{-1}(d_i)$ кейбір мәндерін ескере отырып, φ функциясын дәл болжау мүмкін емес.

Іске асырылған жүйенің сипаттамасы

Жүйенің құпия кілтін таңдау:

Бірінші қадамда $r < \max\{a_1, a_2, ..., a_n\}$ бүтін саны таңдалады. Әрбір a_i –дың мәні r негізі бойынша санау жүйесіне түрлендіріліп, $a_i(x) = a_{i0} + a_{i1}x + ... + a_{im}x^m$,

$a_i(r) = a_i$ түрінде көпмүше алынады. Алынған көпмүшелер рационал коэффициенттері бар өрістің элементтері болып табылады. Құпия кілт ретінде:

1. $s(x) = r$ теңдеуінің дәл бір бүтін шешімі болатындай бүтін коэффициенттері $s(x)$ болатын көпмүше. Күрделі шешімдердің саны ерікті түрде таңдалады және жүйенің қауіпсіздік дәрежесі ретінде қызмет етеді;

2. $s(x_r) = r$; болатындай x_r бүтін саны;

3. Жоғарыда сипатталған r бүтін саны.

Шифрлау:

a_i бүтін санының $a_i(x)$ көпмүшелік көрінісі $c_i(x) = a_i(x) \circ s(x)$ шифрланған мәтін көпмүшесімен байланысты. Енді бастапқы функцияның есебі $f((c_1(x), c_2(x), ..., c_n(x))$ есептеуіне келтірілді.

Кері шифрлау:

$R(x) = f\ (a_1(x), ..., a_k(x))$ көпмүшесінің мәнін есептеу талап етілсін. $s(x)$ шифрлау түрлендіруі қолданылғандықтан, қашықтағы машинадан алынған нәтиже $R(s(x))$ түрінде ұсынуға болады. Демек, қажетті мәнді есептеу үшін x_r-ді қою қажет, өйткені көрсеткіші бойынша $s(x_r) = r$, ал $R(r)$ - қажетті мән, өйткені шифрлау гомоморфты және алгоритмнің бірінші қадамның қасиеттері бойынша гомоморфты болды. (санды көпмүше ретінде көрсету).

А. Абрамов [70] еңбегінде қосу, көбейту, алу арифметикалық амалдары үшін гомоморфты шифрлау қарастырылды, бұл диссертациялық жұмыста азайту операциясының гомоморфты шифрлау әдісі ұсынылған [88].

Шифрлау арқылы Z_n бүтін сандар сақинасынан формальды $Z_n[x]$ айнымалысына қатысты көпмүшелер сақинасына Hom бейнелеуін құрайық:

$Hom: Z_n \to Z_n[x]$.

Теорема 2.6 Hom бейнелеуі Z_n және $Z_n[x]$ сақиналарының гомоморфизмі болып табылады.

Дәлелдеу: Сақина гомоморфизм анықтамасы бойынша, келесі шарттар орындалатынын көрсету қажет:

$$Hom(a -_z b) = Hom(a) -_{z[x]} Hom(b), \qquad (2.15)$$

Кез келген $a, b \in Z$ және $a > b$ үшін.

$r: r < \{a, b\}$ санау жүйесі негізінде таңдалынады, онда a және b бейнелеуі $Z[x]$ –ке сәйкесінше $a(x) = a_m x^m + a_{m-1} x^{m-1} + \cdots + a_x + a_0$ және $b(x) = b_k x^k + b_{k-1} x^{k-1} + \cdots + b_k + b_0$ болады, мұндағы $a_{10} = (a_m, a_{m-1}, \dots, a_0)_r$ және $b_{10} = (b_k, b_{k-1}, \dots, b_0)_r$ болады.

$Hom(a) -_{Z[x]} Hom(b) = a(x) - b(x) = a_m x^m + \cdots + a_{k+1} x^{k+1} + (a_k - b_k) x^k + \dots + (a_0 - b_0) = Hom(a -_z b)$.

Жүйе жұмысына мысал:

a және b сандарының қосындысын, көбейтіндісін және бөлімін есептеу талап етілсін, мұндағы $a = 195$, $b = 23$. $r < \max\{195, 23\} = 195$ сандары таңдалынды. Алгоритмге келесі шарт қосылды: $a > b$

$r = 8$ болсын, сонда:

$a_r(x) = 3x^2 - 12x + 15$ $\quad$ $b_r(x) = 2x + 7$

Құрылуы бойынша, $a_r(r) = a,\ b_r(r) = b$

$s(x)$ шифрлау көпмүшелік түрлендіруін енгіземіз. Бұл келесіге тең болсын $s(x) = x - 2$, сонда:

$x_r = 10$, өйткені x_r – бұл $s(x_r) = r$ теңдеуінің бүтін түбірі.

Қортындылай келе, жүйенің құпия кілті $(s(x),\ x_r,\ r) = (x - 2,\ 10,\ 8)$ түрде болады. Осылайша a және b үшін шифрланған түрлендіру алынады:

$c_a(x) = a(x) \circ s(x) = 3(x-2)^2 - 12(x-2) + 15 = 3x^2 - 12x + 15,$

$c_b(x) = b(x) \circ s(x) = 2(x - 2) + 7 = 2x + 3.$

Енді келесі мәндерді есептеу қажет:

1. $S(x) = c_a(x) + c_b(x) = 3x^2 - 10x + 18;$

$S(x_r) = S(10) = 218 = a + b.$

2. $M(X) = c_a(x) * c_b(x) = 6x^3 - 15x^2 - x + 45;$

$M(x_r) = M(10) = 4485 = ab.$

3. $D(x) = c_a(x)/c_b(x) = (3x^2 - 12x + 15)/(2x + 3);$

$D(x_r) = D(10) = 195/25.$

Ұсынылған арифметикалық азайту амалы келесі түрде орындалады:

4. $A(x) = c_a(x) - c_b(x) = 3x^2 - 14x + 12;$

$A(x_r) = A(10) = 172 = a - b.$

Мәтінді шифрлау мысалы:

Мысал ретінде «Informatics» сөзі алынды.

Шифрлау:

1. Бастапқыда әрбір әріп ASCII кестесіне ауыстырылады. Сонда I – 73, n – 110, f – 102, o – 111, r – 114, m – 109, a – 97, t – 116, i – 105, c – 99, s – 115;

2. $r=7$ болсын, әрбір әріп жетілік санау жүйесіне ауыстырылып, алынған мән көпмүше түрінде жазылады [70]:

$a_I(x) = 133_7 = x^2 + 3x + 3$ $\quad$ $a_a(x) = 166_7 = x^2 + 6x + 6$

$a_n(x) = 215_7 = 2x^2 + x + 5$ $\quad$ $a_t(x) = 224_7 = 2x^2 + 2x + 4$

$a_f(x) = 204_7 = 2x^2 + 4$ $\quad$ $a_i(x) = 210_7 = 2x^2 + x$

$a_o(x) = 216_7 = 2x^2 + x + 6$ $\quad$ $a_c(x) = 201_7 = 2x^2 + 1$

$a_r(x) = 222_7 = 2x^2 + 2x + 2$ $\quad$ $a_s(x) = 223_7 = 2x^2 + 2x + 3$

$a_m(x) = 214_7 = 2x^2 + x + 4$

3. $s(x)$ шифрлау көпмүшелік түрлендіруін енгізіледі. Ол $s(x)=x-3$-ке тең болсын, x_r бұл $s(x_r) = r$ теңдеуінің бүтін түбірі болғандықтан $x_r = 10$.

Қорыта келе, жүйенің құпия кілті $(s(x),\ x_r,\ r) = (x - 3,\ 10,\ 7)$ түрде болады, осылайша шифрланған көпмүше алынады:

$c_I(x) = I(x) \circ S(x) = (x-3)^2 + 3(x-3) + 3 = x^2 - 3x + 3;$

$c_n(x) = n(x) \circ S(x) = 2(x-3)^2 + x - 3 + 5 = 2x^2 - 11x + 20;$

$c_f(x) = f(x) \circ S(x) = 2(x-3)^2 + 4 = 2x^2 - 12x + 22;$

$c_o(x) = o(x) \circ S(x) = 2(x-3)^2 + x - 3 + 6 = 2x^2 - 11x + 21;$

$c_r(x) = r(x) \circ S(x) = 2(x-3)^2 + 2(x-3) + 2 = 2x^2 - 10x + 14;$

$c_m(x) = m(x) \circ S(x) = 2(x-3)^2 + x - 3 + 4 = 2x^2 - 11x + 19;$

$c_a(x) = a(x) \circ S(x) = (x-3)^2 + 6(x-3) + 6 = x^2 - 3;$

$$c_t(x) = t(x)°S(x) = 2(x-3)^2 + 2(x-3) + 4 = 2x^2 - 10x + 16;$$

$c_i(x) = i(x) \circ S(x) = 2(x-3)^2 + x - 3 = 2x^2 - 11x + 15$;

$c_c(x) = c(x) \circ S(x) = 2(x-3)^2 + 1 = 2x^2 - 12x + 19$;

$c_s(x) = s(x) \circ S(x) = 2(x-3)^2 + 2(x-3) + 3 = x^2 - 10x + 15$.

Деректерді кері шифрлау 2.2-кестеде көрсетілген.

Кесте 2.2 – Рационал коэффициенттері бар көпмүшелер өрісіндегі мәндерді кері шифрлау

Шифрланған көпмүше	Құрия кілтті қою арқылы кері шифрлау
$c_I(x) = x^2 - 3x + 3$	$S(x_r) = S(10) = 73 \rightarrow I$
$c_n(x) = 2x^2 - 11x + 20$	$S(x_r) = S(10) = 110 \rightarrow n$
$c_f(x) = 2x^2 - 12x + 22$	$S(x_r) = S(10) = 102 \rightarrow f$
$c_o(x) = 2x^2 - 11x + 21$	$S(x_r) = S(10) = 111 \rightarrow o$
$c_r(x) = 2x^2 - 10x + 14$	$S(x_r) = S(10) = 114 \rightarrow r$
$c_m(x) = 2x^2 - 11x + 19$	$S(x_r) = S(10) = 109 \rightarrow m$
$c_a(x) = x^2 - 3$	$S(x_r) = S(10) = 97 \rightarrow a$
$c_t(x) = 2x^2 - 10x + 16$	$S(x_r) = S(10) = 116 \rightarrow t$
$c_i(x) = 2x^2 - 11x + 15$	$S(x_r) = S(10) = 105 \rightarrow i$
$c_c(x) = 2x^2 - 12x + 19$	$S(x_r) = S(10) = 99 \rightarrow c$
$c_s(x) = x^2 - 10x + 15$	$S(x_r) = S(10) = 115 \rightarrow s$

Шифрланған көпмүшеге құпия кілтті қою арқылы бастапқы «Informatics» сөзі алынды.

Жүйенің гомоморфтылығын көрсету:

Ұсынылған арифметикалық азайту амалы келесі түрде орындалады:

$c_I(x) - c_n(x) = -x^2 + 8x - 17 \rightarrow A(x_r) = A(10) = -37 = I - n$;

$c_n(x) - c_f(x) = x - 2 \rightarrow A(x_r) = A(10) = -8 = n - f$;

$c_f(x) - c_o(x) = -x + 1 \rightarrow A(x_r) = A(10) = -39 = f - 0$;

$c_o(x) - c_r(x) = -x + 7 \rightarrow A(x_r) = A(10) = -3 = o - r$;

$c_r(x) - c_m(x) = x - 5 \rightarrow A(x_r) = A(10) = 5 = r - m$;

$c_m(x) - c_a(x) = x^2 - 11x + 22 \rightarrow A(x_r) = A(10) = 12 = m - a$;

$c_a(x) - c_t(x) = -x^2 + 10x - 19 \rightarrow A(x_r) = A(10) = -19 = a - t$;

$c_t(x) - c_i(x) = x - 1 \rightarrow A(x_r) = A(10) = 9 = t - i$;

$c_i(x) - c_c(x) = x - 4 \rightarrow A(x_r) = A(10) = 6 = i - c$;

$c_c(x) - c_s(x) = -2x + 4 \rightarrow A(x_r) = A(10) = -16 = c - s$.

Рационал коэффициенттері бар көпмүшелер өрісіндегі шифрланған деректерге азайту операциясы ұсынылды. Ұсынылған азайту опрециясының псевдо кодын 2.2 – суреттен көруге болады.

Algorithm 1

$size \leftarrow RANDOM$
$numbers1[size], numbers2[size]$
$r \leftarrow RANDOM$
$S(x) \leftarrow RANDOM$
$numbase1[size][], numbase2[size][]$
xr
ENCRYPTION(numbers1,numbers2,r,S(x))
GOMOMORPHISM
-
for $i \leftarrow 0$ to $size$ **do**
    $numbase1[i][] \leftarrow numbers1[i]$ *r sanau juiesinde*
    $numbase2[i][] \leftarrow numbers2[i]$ *r sanau juiesinde*
    for $j \leftarrow 0$ to $numbase1[i].Length$ **do**
        $numbaseRes[i][j] \leftarrow numbase1[i][j] - numbase2[i][j]$
    end for
end for
DECRYPTION(numbaseRes,S(x),xr)

Сурет 2.2 - Ұсынылған азайту операцияның псевдо коды

2.3 Пэйе криптожүйесі

Пэйе криптожүйесі [89] – ашық кілтті криптографияға негізделген асимметриялық алгоритм. Ол қиын деп есептелетін қалдықтардың *n*-ші класын есептеу есебіне негізделген. Пэйе аддитивті гомоморфты криптожүйе болып табылады. Бұл криптожүйе электронды кассалық операциялар, қауіпсіз электронды дауыс беруді жүзеге асыру үшін кеңінен қолданылады. Пэйе гомоморфты шифрлау және кері шифрлау процесін өңдеу ресурстары шектеулі кез келген құрылғы үшін есептеу тұрғысынан қымбат [90]. Сондықтан өңдеу ресурстары шектеулі құрылғыларын қолдайтын жақсартылған және жылдам шифрлау және кері шифрлау нұсқасы қажет. Пэйе гомоморфты шифрлау үшін жақсартылған және жылдам шифрды шешу нұсқасы қолданылып, сол бойынша әр түрлі AtmelAVR микроконтроллерде кітапхана құрылды. Бұл криптожүйені сипаттайтын негізгі түрлендірулер төмендегідей:

Кілтті генерациялау:

$$p, q. \quad \mathrm{Gcd}\left(pq,\ (p-1)(q-1)\right) = 1,$$

$$n = p * q \text{ және } \lambda = lcm(p-1)(q-1),$$

$$\mu = \left(L\left(g^{\lambda} mod n^2\right)\right)^{-1} mod n,$$

мұндағы, $L(u) = div\left(\frac{u-1}{n}\right)$

$public\ key\ (n)$;

$private\ key\ (p, q, n)$.

Шифрлау:

$$r \epsilon Z_n ,\ \ \gcd(r, n) = 1,$$

$$\mathrm{c} = g^m * r^n mod. \qquad (2.16)$$

Кері шифрлау:

$$h = \left[L\left(g^{\varphi/n} mod n^2\right)\right]^{-1} mod n;$$

$$h_p = \left(\frac{g^{p-1} mod p^2 - 1}{p}\right)^{-1} mod p\ ,\ h_q = \left(\frac{g^{q-1} mod q^2 - 1}{q}\right)^{-1} mod q;$$

$$m = L\left(c^{\varphi/n} mod n^2\right) * h mod n;$$

$$m_p = \left(\frac{c^{p-1} mod p^2 - 1}{p}\right) * h_p mod p\ , \quad m_q = \left(\frac{c^{q-1} mod q^2 - 1}{q}\right) * h_q mod q. \quad (2.17)$$

Гомоморфтылық қасиеті:

1. Ашық мәтіндердің гомоморфты қосылуы:

$$D(E(m_1) * E(m_2) mod n^2) = m_1 + m_2 mod n. \quad (2.18)$$

2. Ашық мәтіндердің гомоморфты көбейтілуі:

$$D(E(m_1, r_1)^{m_2} mod n^2) = m_1 * m_2 mod n, \quad (2.19)$$

$$D(E(m_2, r_2)^{m_1} mod n^2) = m_1 * m_2 mod n.$$

3. Ашық мәтіндердің гомоморфты азайтылуы:

$$D(E(m_1) * E(m_2)^{-1} mod n^2) = (m_1 - m_2) mod n. \quad (2.20)$$

Жүйенің жұмыс жасау мысалы:

Кілтті генерациялау:

1. $p = 41, q = 31.$ $\gcd\left(pq,\ (p-1)(q-1)\right) = 1$ $m_1 = 97,\ m_2 = 98,$

2. $n = 1271$ λ=120,

3. $\mu = 519,\ g = 1272,\ r = 683.$

Шифрлау:

$c_1 = 1559022, c_2 = 684812.$ (2.16)

Кері шифрлау:

$m_1 = 97,\ m_2 = 98.$ (2.17)

1. Ашық мәтіндердің гомоморфты қосылуы:

$1559022 * 684812 mod 811801 = 1615441$ $D = \frac{94169^{120} mod 1615441 - 1}{1271} *$

$519 mod 1271 = 522 * 519 mod 1271 = 195$ (2.18)

2. Ашық мәтіндердің гомоморфты көбейтілуі:

$1559022^{98} mod 1615441 = 1054652$ $D = \frac{1054652^{120} mod 1615441 - 1}{1271} *$

$519 mod 1271 = 633 * 519 mod 1271 = 609$ (2.19)

3. Ашық мәтіндердің гомоморфты азайтылуы:

$1559022 * 684812^{-1} mod 1615441 = 399180$

$D = \frac{399180^{120} mod 1615441 - 1}{1271} * 519 mod 1271 = 1151 * 519 mod 1271 = 1270$

(2.20)

Гомоморфты Пэйе криптожүйесінде шифрланған деректерге қосу, азайту, көбейту операциялары орындалды. Орындалған операциялардың пседо кодын 2.3 – суреттен көруге болады.

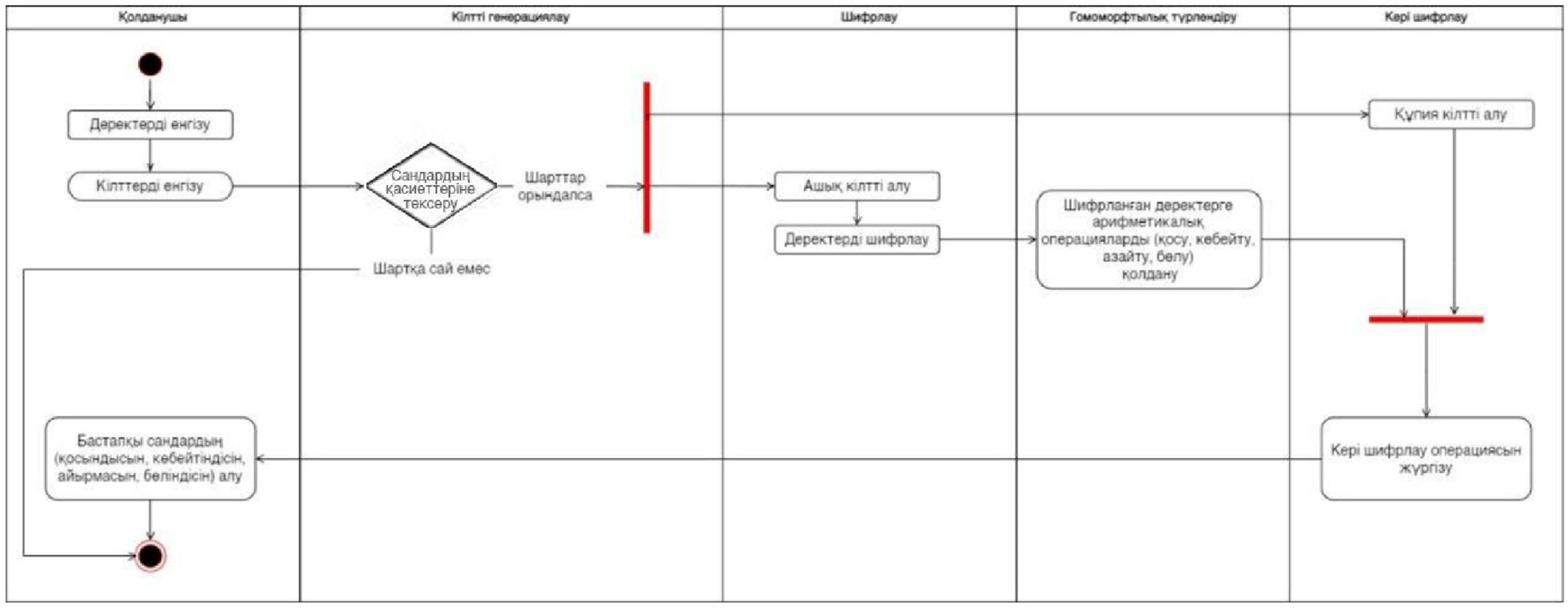
Algorithm 1

$size \leftarrow RANDOM$
$numbers1[size], numbers2[size]$
$c1[size], c2[size]$
$p, q \leftarrow PrimeNumber$
if $GCD(pq, (p-1)(q-1)) \neq 1$ **then**
  $Incorrect. GCD(pq, (p-1)(q-1))$ *must be* 1 *and p and q must be prime*
$numbers$
else
  $n \leftarrow p \times q$
  $\lambda \leftarrow LCM(p-1)(q-1)$
  $g \leftarrow n+1$
  $\mu \leftarrow ((g^{\lambda} mod n^2 - 1)/n)^{-1} \ mod \ n$
  $ENCRYPTION$
  for $i \leftarrow 0$ *to* $size$ **do**
    $r \leftarrow RANDOM$
    $c1[i] \leftarrow g^{numbers1[i]} \times r^n \ mod \ n^2$
    $c2[i] \leftarrow g^{numbers2[i]} \times r^n \ mod \ n^2$
  end for
  $GOMOMORPHISM$
  $+$
  for $i \leftarrow 0$ *to* $size$ **do**
    $buf \leftarrow c1[i] \times c2[i] \ mod \ n^2$
    $numebers1[i] + numbers2[i] modn \leftarrow ((buf^{\lambda} mod n^2 - 1)/n) \times \mu \ mod \ n$
  end for
  $*$
  for $i \leftarrow 0$ *to* $size$ **do**
    $buf \leftarrow c1[i]^{numbers2[i]} \ mod \ n^2$
    $numebers1[i] \times numbers2[i] modn \leftarrow ((buf^{\lambda} mod n^2 - 1)/n) \times \mu \ mod \ n$
  end for
  $-$
  for $i \leftarrow 0$ *to* $size$ **do**
    $buf \leftarrow c1[i] \times c2[i]^{-1} \ mod \ n^2$
    $numebers1[i] - numbers2[i] modn \leftarrow ((buf^{\lambda} mod n^2 - 1)/n) \times \mu \ mod \ n$
  end for
end if

Сурет 2.3 - Гомоморфты Пэйе криптожүйесінде орындалған операциялар

2.4 Толық гомоморфты шифрлау принципі

Гомоморфты шифрлаудың модификацияланған әдістері: айнымалысы бар көпмүшелер өрісінде гомоморфты шифрланған мәліметтерге азайту және бөлу операциялары, рационал коэффициенттері бар көпмүшелер өрісіндегі гомоморфты шифрланған мәліметтерге – азайту операциялары қосылды. 2.4 – суретте модификацияланған толық гомоморфты шифрлау іс-әрекет диаграмма түрінде көрсетілген.



Сурет 2.4 – Толық гомоморфты шифрлау принципі

3 IoT ҚҰРЫЛҒЫЛАР КЛАСТЕРІНІҢ ҚАУІПСІЗДІГІН ҚАМТАМАСЫЗ ЕТУ ҮШІН ATMELAVR МИКРОКОНТРОЛЛЕРІНДЕ КІТАПХАНА АРХИТЕКТУРАСЫН ҚҰРУ

3.1 Жетілдірілген айнымалысы бар көпмүшелер сақинасындағы толық гомоморфты шифрлауға кітапхана құру

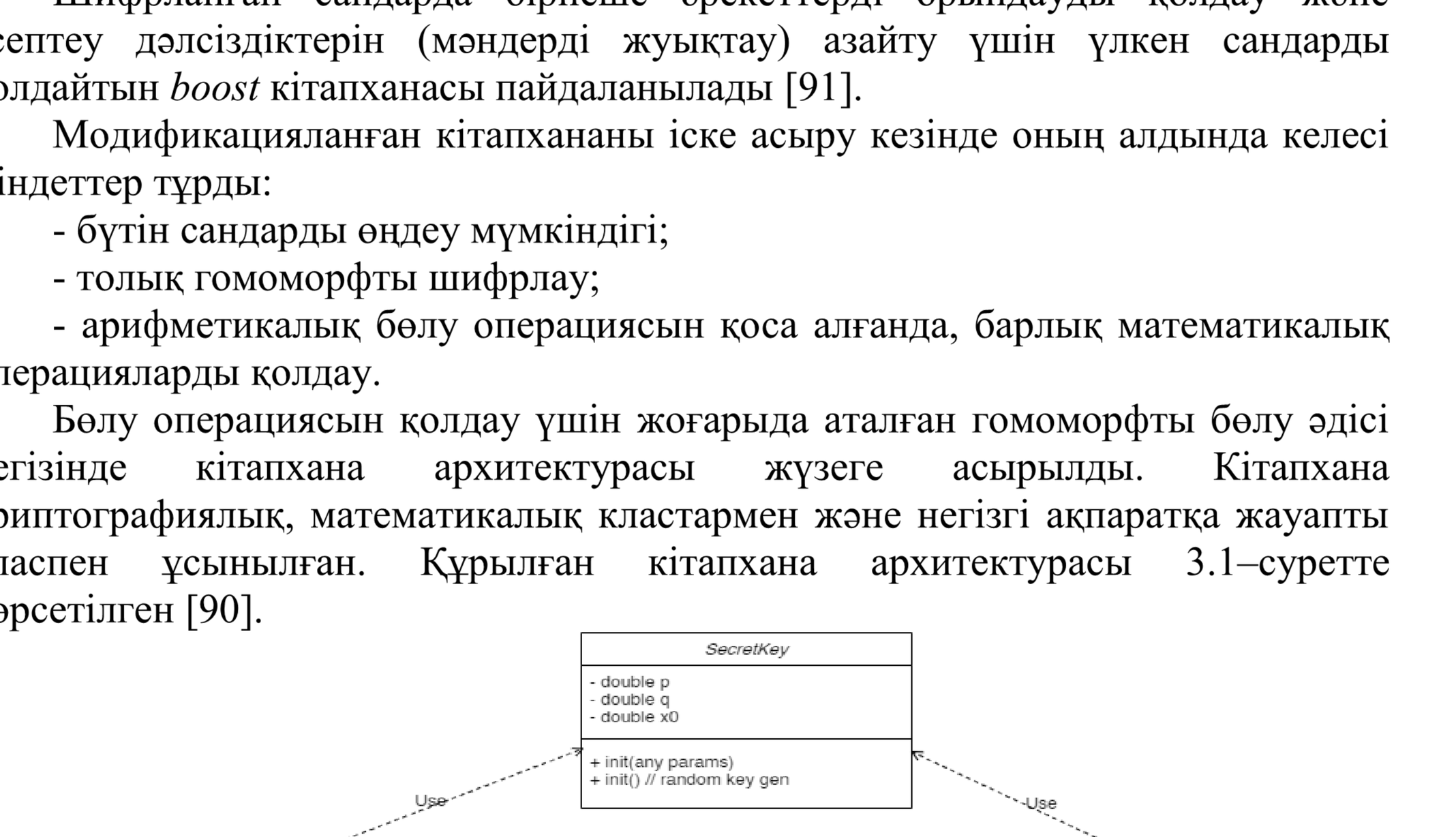
Arduino IDE біріктірілген даму ортасында, C++ тілінде әртүрлі микроконтроллер үшін айнымалысы бар көпмүшелер сақинасындағы гомоморфты модификацияланған айнымалысы бар көпмүшелер сақинасындағы толық гомоморфты шифрлауға статикалық кітапхана құрылды.

Шифрланған сандарда бірнеше әрекеттерді орындауды қолдау және есептеу дәлсіздіктерін (мәндерді жуықтау) азайту үшін үлкен сандарды қолдайтын *boost* кітапханасы пайдаланылады [91].

Модификацияланған кітапхананы іске асыру кезінде оның алдында келесі міндеттер тұрды:

- бүтін сандарды өңдеу мүмкіндігі;
- толық гомоморфты шифрлау;
- арифметикалық бөлу операциясын қоса алғанда, барлық математикалық операцияларды қолдау.

Бөлу операциясын қолдау үшін жоғарыда аталған гомоморфты бөлу әдісі негізінде кітапхана архитектурасы жүзеге асырылды. Кітапхана криптографиялық, математикалық кластармен және негізгі ақпаратқа жауапты класпен ұсынылған. Құрылған кітапхана архитектурасы 3.1–суретте көрсетілген [90].



Сурет 3.1 - Жетілдірілген айнымалысы бар көпмүшелер сақинасындағы толық гомоморфты шифрлаудың кітапхана архитектурасы

Secret Key класы криптографиялық алгоритмде қолданылатын құпия кілт туралы деректермен жұмыс істейді. Бұл жаңа кілтті құру, кездейсоқ генерациялау және оны пайдалану мүмкіндігін қамтамасыз етеді. Ағымдағы кітапхананың іске асырылуы кілттер мен көпмүше коэффициенттерін генерациялау үшін ағымдағы уақытқа қатысты автоматты рандомизациясы бар стандартты кітапханадан кездейсоқ сандар генераторын пайдаланады. Стандартты кітапханадан псевдокездейсоқ сандар генераторы криптографиялық тұрғыдан сенімсіз деп танылады, өйткені ол сызықты конгруентті әдісті пайдаланады. Осыған байланысты кездейсоқ генераторлық енгізулерді пайдалануға мүмкіндік беретін интерфейс енгізілді. Келісім бойынша, Boost кітапханасынан кездейсоқ сандар генераторын пайдалану ұсынылады. Ол детерминирленген емес кездейсоқ сандар генерациясын пайдаланады және криптографиялық тұрғыдан қауіпсіз.

Encrypted Data – негізгі деректер түрі гомоморфты шифрланған сан болатын модуль болып табылады. Шифрлау класында ашық мәтіндер мен құпия кілттерін (деректерді шифрлау операциясы) қолдану арқылы жаңа шифр мәтіндерін құру, шифрлау кілті негізінде шифрланған деректерден ашық деректерді алу (мәліметтерді шифрлау операциясы) мүмкіндіктері іске асырылады.

Homomorphism – шифрланған деректерге негізгі азайту, қосу, бөлу, көбейту математикалық операцияларын орындайды. Шифрлау мен кері шифрлау алдын ала генерацияланған кілттерді қолдану арқылы немесе құпия параметрлерді беру көмегімен іске асырылады. Сонымен қатар класс көпмүшелерге қажетті барлық математикалық операцияларды атап айтсақ бөлу, көбейту, азайту және қосуды іске асырады.

Decrypted Data құпия кілттің көмегімен шифрланған деректерді және гомоморфтылықты кері шифрлайды.

Гомоморфизмге тексеру. Қосу операциясы бойынша гомоморфтылықты тексеру үшін resBuf деген массив ашылады. Оның өлшемі қай көпмүшенің ұзындығы үлкен, соған сәйкес алынады. Әрі қарай resBuf-тың барлық элементтерін 0 деп алынады. Кейін цикл бойынша әр индекске сәйкес келетін индекс көпмүше мәндері қосылады. Қосынды мәнін есептеу үшін resu=0 деген айнымалыны ашып, resu+=resbuff[0]*x^0. Дәл осылай көпмүшенің қалған бөліктеріне жасалынады.

Азайту операциясы бойынша гомоморфтылықты тексеру Алгоритмді қосуға тексерген сияқты resBuf массивіне бірінші көпмүшенің сандарын алып, одан екінші көпмүшенің сандары азайтылады.

Көбейту операциясы бойынша гомоморфтылықты тексеру. Kob деген массив ашылады. Оның өлшемі [(бірінші көпмүше өлшемі)*(екінші көпмүше өлшемі)]. Екі бағанға бөлінеді, бірінші бағанда x-тың алдында тұрған коэффициенттер, ал екінші бағанда сол x-тың дәрежелері. Resu массиві ашылып, оған бірдей дәрежелер қосылады.

Бөлу операциясы бойынша гомоморфтылықты тексеру. Шифрланған деректерге бөлу операциясын орындау үшін Горнер схемасы қолданылды.

Көпмүшені көпмүшеге бөлгенде, бөлінді және қалдық алынды. Кітапхананың жалпы іске асырылуы 3.1-кестеде көрсетілген.

Кесте 3.1 AtmelAVR микроконтроллерде кітапхананы іске асыру функциялары [90]

Екі сан бойынша гомоморфтылықты көрсету		
Функцияның атауы	Функцияның сигнатурасы	Сипаттамасы
EncrZ1	EncrZ1(double q, double p, double n, double z1, double a[]);	Гомоморфты *Z1* үшін шифрлау, *q*-натурал, *p*-бүтін, *z1,a[ ]* - кездейсоқ сандар.
EncrZ2	EncrZ2(double q, double p, double n, double z2, double b[]);	Гомоморфты *Z2* үшін шифрлау, *q*-натурал, *p* - бүтін, *z2, b[ ]* - кездейсоқ сандар.
DecrZ1	DecrZ1(double q, double p, double n, double z1, double a[]);	Гомоморфты *Z1* үшін кері шифрлау, *q* - натурал, *p*-бүтін, *z1, a[ ]* - кездейсоқ сандар.
DecrZ2	DecrZ2(double q, double p, double n, double z2, double b[]);	Гомоморфты *Z2* үшін кері шифрлау. *q* - натурал, *p*-бүтін, *z2, b[ ]* - кездейсоқ сандар.
DecrZ1kosuZ2	DecrZ1kosuZ2(double q, double p, double n, double z1, double z2, double a[], double b[]);	Шифрлаудың гомоморфтылығына дәлел ретінде *Z1, Z2* – көпмүшелерге қосу амалын жүзеге асырады.
DecrZ1kobituZ2	DecrZ1kobituZ2(double q, double p, double n, double z1, double z2, double a[], double b[]);	Шифрлаудың гомоморфтылығына дәлел ретінде *Z1* мен *Z2* көпмүшелерге көбейту амалын жүзеге асырады.
DecrZ1azaituZ2	DecrZ1azaituZ2(double q, double p, double n, double z1, double z2, double a[], double b[]);	Шифрлаудың гомоморфтылығына дәлел ретінде *Z1* мен *Z2* көпмүшелерге азайту амалын жүзеге асырады.
DecrZ1boluZ2	DecrZ1boluZ2(double q, double p, double n, double z1, double z2, double a[], double b[]);	Шифрлаудың гомоморфтылығына дәлел ретінде *Z1* мен *Z2* көпмүшелерге бөлу

		амалын жүзеге асырады.
Koef	Koef(double n);	*n > 0* және *a[ ]* коэффициенттері кездейсоқ тәртіпте берілген көпмүше коэффициетін алады.
print_array	print_array(double a[], double n);	Толық гомоморфты шифрлау үшін көпмүше қажет. Функция көпмүшені экранға шығарады.
Random	Random(int n)	Еркін кездейсоқ сан генетор функциясы.
P	double P()	*P* үшін бүтін кездейсоқ сан қабылдайды.
Q	double Q()	*Q* үшін натурал кездейсоқ сан қабылдайды.
X	double X()	*X=P/Q* осы шифрдың құпия кілтін анықтайды.
Кез-келген ақпарат бойынша гомоморфтылық		
Encr	Encr(double q, double p, double n, double z1, double a[]);	ASCII кестесінен мәтіндегі сөздердің әріптерін іздеп, алынған әріпке сәйкес әріпке ASCII кодын табады. Алынған әрбір кодты көпмүшеге айналдыру арқылы шифрлайды, мұндағы *q*-натурал, *p*-бүтін, *z1, a[ ]* - кездейсоқ сандар.
Decr	Decr(double q, double p, double n, double z1, double a[]);	Алынған полиномды кері шифрлайды. Кері шифрланған сандарды ASCII кестесінен іздейді. Алынған мән бойынша әріптерді жазады, мұндағы . *q* - натурал, *p* - бүтін, *z1, a[ ]* - кездейсоқ сандар.
SetSecretKey	SetSecretKey(int x)	Құпия кілттің мәнін өз қалауымыз бойынша бірден меншіктейтін

		функция
incrdecr	int incrdecr(float& stime, float& dtime);	Шифрлау және кері шифрлау уақытын анықтайтын функция.

Жүйенің барлық талаптарын қанағаттандыру үшін *EncrZ1(double q, double p, double n, double z1, double a[]), EncrZ2(double q, double p, double n, double z2, double a[])* келесі әрекеттерді орындауы қажет:

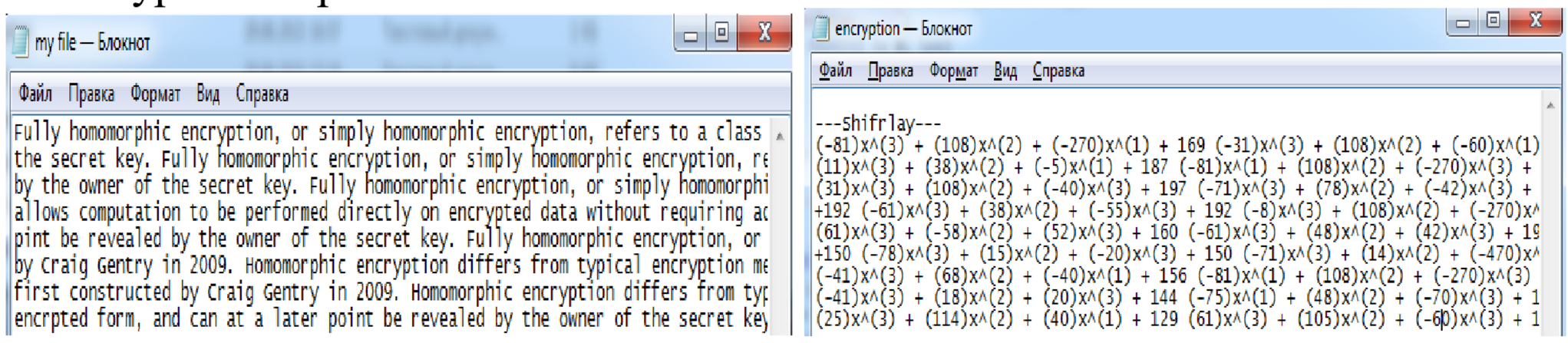
- алдын ала берілген коэффициенттері бар көпмүшелерді құру;
- көпмүшелерге қосу, алу, көбейту, бөлу, сонымен қатар тұрақты шамаға көбейту сияқты қарапайым математикалық амалдарды орындау;
- нөлді қоса алғанда, айнымалының кез келген берілген дәрежесінде коэффициент мәнін көрсету;
- кез келген дәрежедегі коэффициент мәнін кез келген алдын ала анықталған мәнге өзгерту;
- берілген нүктедегі көпмүшенің мәнін есептеу.

Қосу және көбейту амалдарын жүзеге асыру үшін сәйкесінше $O(n)$ және $O(n^2)$ уақытында жұмыс істейтін классикалық алгоритмдер қолданылды. Тұрақты шамаға көбейту операциясы да $O(n)$ бойынша орындалады. Көпмүшенің нүктедегі мәнін есептеу үшін көбейту амалдарының санын азайтуға және қажетті мәнді $O(n)$ есептеуге мүмкіндік беретін Горнер схемасы қолданылады [76].

Шифрланған көпмүшелерді алғаннан кейін келесі функциялар *DecrZ1(double q, double p, double n, double z1, double a[]), DecrZ2(double q, double p, double n, double z2, double a[])* құпия кілттің көмегімен деректерді кері шифрлайды.

Берілген шифрлаудың гомоморфты екенін дәлелдеу үшін кітапханада *void DecrZ1kosuZ2(double q, double p, double n, double z1, double z2, double a[], double b[])DecrZ1kobituZ2(double q, double p, double n, double z1, double z2, double a[], double b[]);DecrZ1azaituZ2(double q, double p, double n, double z1, double z2, double a[], double b[]);DecrZ1boluZ2(double q, double p, double n, double z1, double z2, double a[], double b[])* функциялар қолданылады.

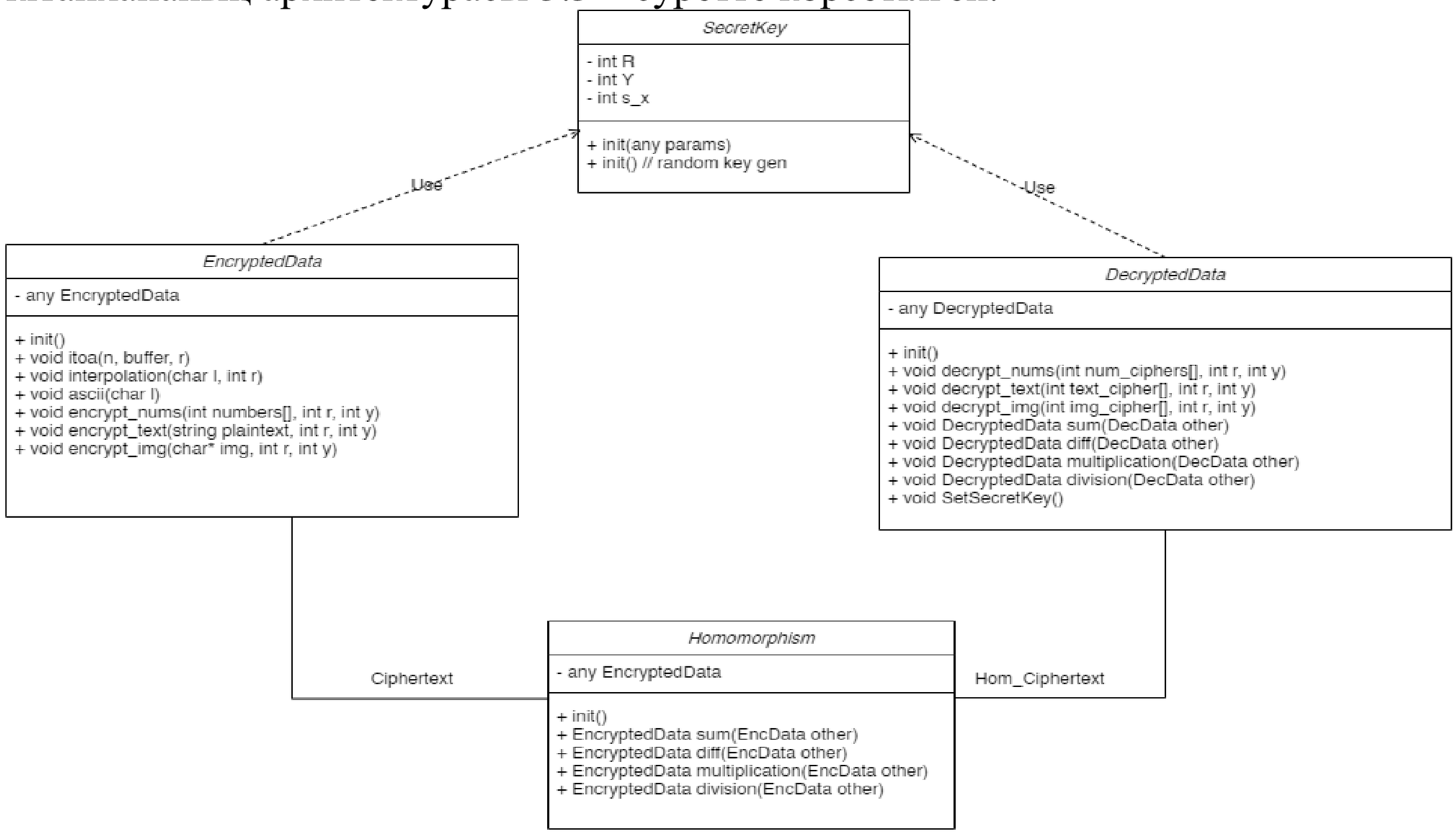
Көпмүшелерді құру үшін Koef(double n) функциясы қолданылады. Мұндағы коэффициенттер кездейсоқ таңдалынады. Айнымалысы бар көпмүшелер сақинасындағы толық гомоморфты шифрлау және кері шифрлау 3.2 – суретте көрсетілген.



Сурет 3.2 - Айнымалысы бар көпмүшелер сақинасындағы толық гомоморфты шифрлау және кері шифрлау

3.2 Жетілдірілген рационал коэффициенттері бар көпмүшелер өрісіндегі гомоморфизмге кітапхана құру

Рационал коэффициенттері бар көпмүшелер өрісіндегі гомоморфизмнің кітапханасы әртүрлі AtmelAVR микроконтроллерінде құрылды. Құрылған кітапхананың архитектурасы 3.3 – суретте көрсетілген.



Сурет 3.3 – Рационал коэффициенттері бар көпмүшелер өрісіндегі гомоморфизмнің кітапхана архитектурасы

Secret key класында құпия кілтті генерациялау функцияларын жүзеге асырады: $key\ ((int\ R, int\ Y, float\ \&\ ut)$ функциясына берілген n бүтін санды ескере отырып, $x + i * y, x - i * y$ комплекс сандар жұбын қайтарады, мұндағы $|x| < n$ және $0 < y < n$, мұндағы x және y бүтін. Осылайша құрылған жұптар кейіннен кілттік көпмүшенің түбірлеріне айналады; Содан кейін ол кіріс ретінде жүйенің қауіпсіздік параметрі және деректерді шифрлау үшін деректер болып табылатын бүтін сек мәнін қабылдайды. Функция дәйекті түрде күрделі түбірлер жұбын жасайды, содан кейін бір бүтін түбірді алады.Ол шығысында функция жүйенің жабық кілтін береді: шифрлаушы көпмүшенің коэффициенттерінің бүтін массиві, теңдеудің бүтін түбірі, және шифрланған сандарды аударған санау жүйесінің негізі.

Encryption класы бастапқы деректерді шифрлауға арналған функцияларды жүзеге асырады: функцияның интерполяциясы $(char\ l, int\ r)$.

Функция кіріс ретінде шифрлау үшін бүтін санын және алдын ала жасалған y жүйесінің жабық кілтін қабылдайды. Жұмыс барысында жоғарыда сипатталғандай негізі r болатын санау жүйесіне қатысты бүтін санды көпмүшеге айналдырады; нәтижесінде коэффициенттердің бүтін массиві қайтарылады. Шифрлау функциясы $(char\ l, int\ r, int\ y)$

Функцияның жұмысы кезінде сан алдымен құпия кілт негізіндегі көпмүше ретінде көрсетіледі; онда композиция функциясы құпия кілтті көпмүше және нәтижелі көпмүше үшін орындалады. Шығару санның бүтін коэффициенттері бар көпмүше ретіндегі шифрланған көрінісі болып табылады.

Кері шифрлау модулі қабылданған деректердің шифрын ашу функциясын жүзеге асырады:

Кері шифрлау функциясы $(char\ l, int\ r, int\ y)$. Функция кіріс ретінде бүтін коэффициенттері res және бұрын жасалған жеке кілт кілті бар көпмүшені қабылдайды. Құпия кілттен түбірді ауыстырғаннан кейін алынған көпмүшенің шифрланған мәні қайтарылады.

Функция кіріс ретінде бүтін коэффициенттері бар көпмүшені және бұрын жасалған y жеке кілтін қабылдайды. Бұрын сипатталған кілттен түбірді ауыстырғаннан кейін алынған көпмүшенің шифрланған мәні қайтарылады.

Бастапқыда деректерді шифрлау үшін құпия кілті жасалады. Құпия кілтті берілген параметрлер негізінде жасауға немесе кездейсоқ түрде жасауға болады. Құпия кілтті жасағаннан кейін шифрлық мәтін объектілерін құру және олардың бастапқы инициализациясын орындау қажет. Бастапқыда жасалған нысандарда деректерді шифрлауды орындауға болады. Ол үшін сәйкес әдісті шақырып, оған қарапайым мәтінді және құпия шифрлау кілтін параметр ретінде беру керек. Шифрланған деректерге әртүрлі математикалық операцияларды орындауға болады. Деректермен операцияларды орындағаннан кейін құпия кілттің көмегімен олардың шифрын ашуға және деректермен орындалатын операциялардың нәтижесін алуға болады [91-95]. Барлық операциялар ретімен орындалады. Кітапханада қолданылатынфункциялар 3.2 – кестеде көрсетілген.

Кесте 3.2 – Кітапханада қолданылатын функциялар [74]

Функцияның атауы	Функцияның сигнатурасы (сигнатура функции)	Сипаттамасы
$itoa$	$itoa\ (n, buffer, r);$	n-дің әрқайсысын r негізімен санау жүйесіне аударып, көпмүше түрінде алады.
$km(unsigned\ n. int\ r)$	$string\ km(unsigned\ n. int\ r)$	r негізі бойынша санау жүйесіне ауыстырылған сандарды көпмүше түрінде жазылады.
enc	$int\ enc\ (int\ a, int\ d)$	Массивтен алынған

		сандарды көпмүшеге аударады.
$enct\ (int\ a, int\ ad, int\ j)$	$int\ enct\ (int\ a, int\ ad, int\ j)$	Көпмүше түрінде берілген мәндерге құпия кілтті қою арқылы шифрлеу процесі жүзеге асады.
$key\ (int\ R, int\ Y, float\&\ ut)$	$int\ key\ (int\ R, int\ Y, float\&\ ut);$	Құпия кілтті қайтаратын функция
$ascii(char\ l);$	$void\ ascii(char\ l);$	Шифрлауға берілген мәндерді ASCII кодына ауыстыратын функция
$decrypt(char\ l, int\ r, int\ y)$	$void\ decrypt(char\ l, int\ r, int\ y)$	Шифрланған мәндерді құпия кілттің көмегімен кері шифрлау процесі іске асады.

Гомоморфты шифрлау схемасын жүзеге асыру

Қарастырылып отырған схеманың негізгі кезеңдері кілттерді генерациялау, шифрлау, гомомоморфизм және кері шифрлау болып табылады. Тиісінше, осы схемамен жұмыс істеуге арналған клиент коды 4 негізгі класқа бөлінеді.

1. SecretKey класы. Бұл класс құпия кілтпен жұмыс істеу логикасын инкапсуляциялайды; жалпы интерфейсте осы класс данасын жасаудың статикалық әдісі орналастырылған. Енгізу параметрлері генерацияланған көпмүшенің дәрежесі және генерацияда қолданылатын санау жүйесінің негізі болып табылады. Шығару кезінде көпмүше мен оның түбірінен тұратын құпия кілт алынады.

2. Encryption класы. Сандардың кіріс тізбегін шифрлау әдістерінің жиынтығын қамтиды. Негізгі әдіс BigInteger пішіміндегі санды енгізу ретінде қабылдайды. Әдістің өзінде санау жүйесінің кездейсоқ негізі құрылады және кездейсоқ көпмүше пайда болады. Әдіс шифрлау нәтижесі болып табылатын Polynom класының данасын қайтарады.

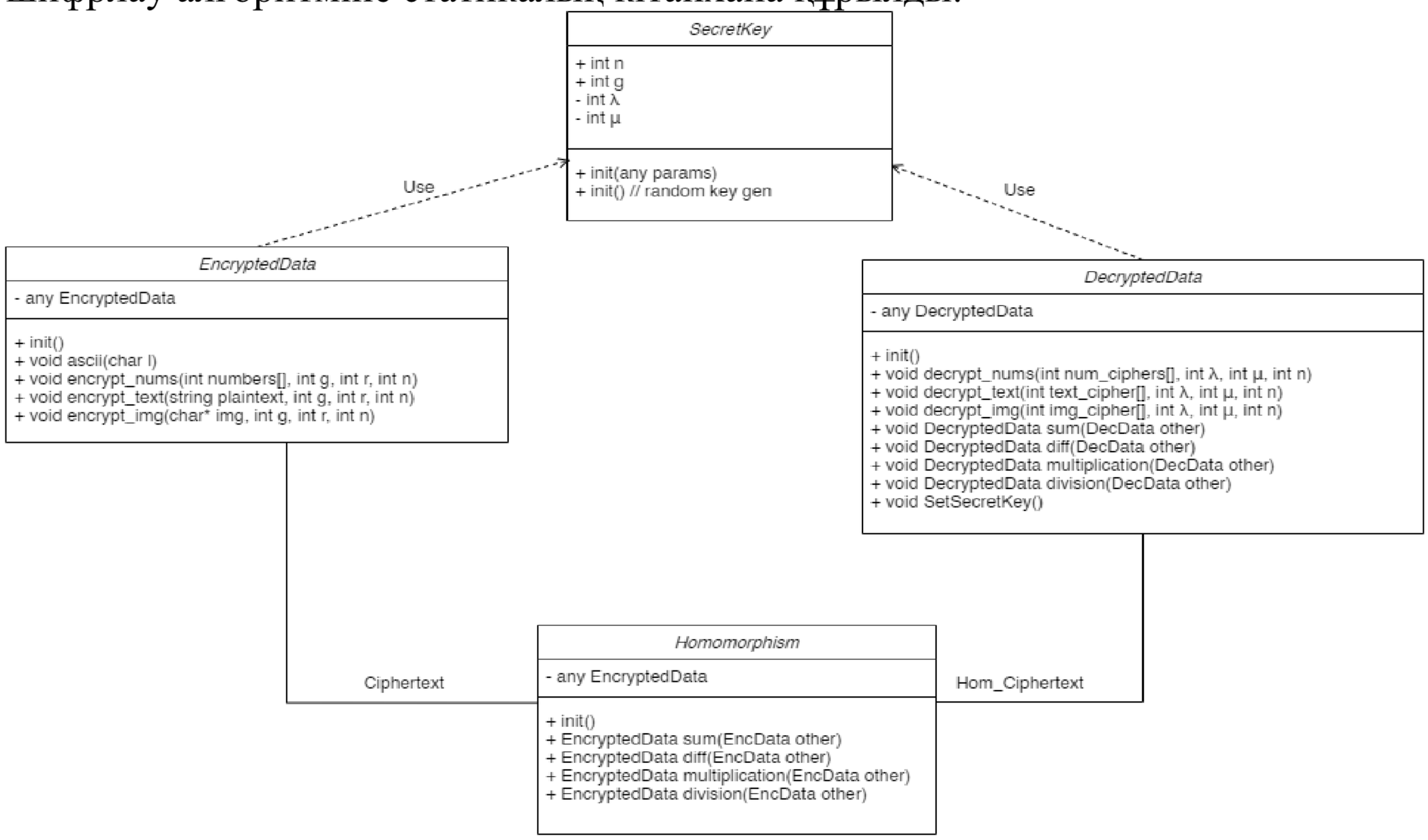
3. Homomorphism класы. Шифрланған деректерге барлық арифметикалық операцияларды орындайды.

3. Decryption класы. Шифрланған деректерді кері шифрлау операциясы үшін SecretKey класының данасы өрісі пайдаланылады - құпия кілтті генерациялау сатысында есептелген көпмүшенің түбірі. Осылайша, бұл класта SecretKey және Polynom нысандарының мәндерінен бастапқы шифрланған нөмірді есептейтін әдіс болып табылады.

3.3 Пэйе гомоморфты шифрлау кітапханасы

Пэйе гомоморфты шифрлаудың аритектурасы құрылды. Құрылған архитектура 3.4 – суретте көрсетілген. Құрылған архитектураға сәйкес Arduino

IDE біріктірілген даму ортасында әр-түрлі микроконтроллер үшін Пэйе шифрлау алгоритміне статикалық кітапхана құрылды.



Сурет 3.4 –Пэйе гомоморфты шифрлау архитектурасы

Secret Key. Аталған класс криптографиялық алгоритмде қолданылатын құпия кілт туралы деректермен жұмыс істейді. Жаңа кілтті құру, кілтті кездейсоқ генерациялау, ашық кілт, құпия кілт және оны пайдалану мүмкіндігін қамтамасыз етеді.

Encryption. Ашық деректердің мәндерін ASCI кестесінен алып, ашық кілттің көмегімен деректерді шифрлайды.

Homomorphism. Шифрланған деректермен әртүрлі математикалық операцияларды орындайды.

DecryptedData. Шифрланған деректермен операцияларды орындағаннан кейін құпия кілттің көмегімен оларды кері шифрлауға және деректермен орындалатын операциялардың нәтижесін алуға болады.

Кесте 3.3 Пэйе кітапхананы іске асыру функциялары [90]

Екі сан бойынша гомоморфтылық		
Функцияның атауы	Функцияның сигнатурасы	Сипаттамасы
add	add(long long a, long long b);	Екі үлкен бүтін санның қосындысын есептеу. a және b – бүтін сандар.
Modpow	modpow(long long a, long long b, long long c);	Санның белгілі бір дәрежесінің келесі санға

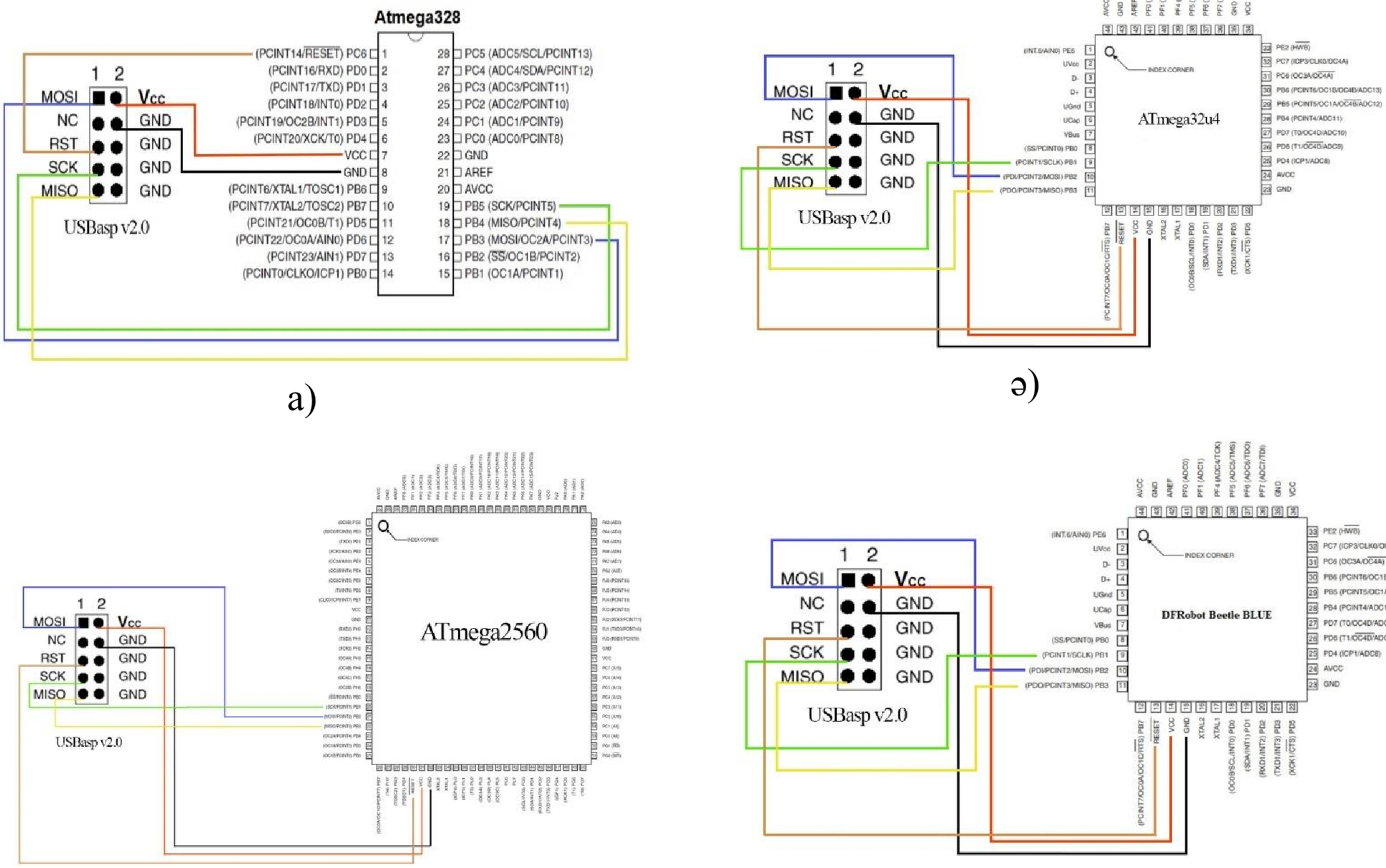
		бөлгендегі қалдығын алу. a, b, c – бүтін сандар $(a^b \% c)$.
reverseMod	reverseMod(long long a, long long b)	Санның -1 дәрежесінің келесі санға бөлгендегі қалдығын алу. a және b – бүтін сандар $(a^{-1} \% b)$.
printSome	printSome(long long A[], long long n);	A[] мәліметтер жиынтығын экранға шығарады (n - жиын ұзындығы).
isPrime	isPrime(int n);	Берілген сан жай сан екенін немесе жай сан еместігін анықтайды n – бүтін сан.
NOD	NOD(long long int a, long long int b);	Екі санның ең үлкен ортақ бөлгішін (ЕҮОБ) табу үшін қолданылады a және b – бүтін сандар.
Nok	Nok(int a, int b);	Екі санның ең кіші ортақ еселігін (ЕКОЕ) табу үшін қолданылады a және b – бүтін сандар.
findG	findG(long long g, long long ly, long long n2, long long n, int size);	
Gomomorph	Gomomorph(long long x, long long ly, long long n, long long n2);	Шифрланған мәліметтерге математикалық операциялар жүргізу арқылы гомоморфты қасиеті дәлелденеді. x – операциялар нәтижесінде алынған мән, ly – құпия кілт, n – бүтін сан.
encryptionPeye	EncryptionPeye(long long int C[], long long int M[], long long int r[], long long g, long long n, long long n2, long long size);	Мәтіндегі әрбір символдарды ASCII кестесінен іздейді. Кестеден алынған мәндер шифрланады. C[] қайтарылатын шифрлар жиынтығы, M[] енгізілген мәліметтері

		жиынтығы, r[] әр шифрлау процесіне қолданылатын бүтін сандар, g,n – ашық кілттер, n2 = n^2
decryptionPeye	decryptionPeye(long long int C[], long long int M2[], long long ly, long long mu, long long n, long long n2, long long size);	Шифрланған мәндерді кері шифрлайды. Кері шифрлау нәтижесінде алынған сандарды ASCII кестеден іздеп, сәйкес символдарды жазады. C[] кері шифрлау жүргізілетін шифрлар жиынтығы, M2[] кері шифрлау нәтижесінде алынатын сандар, ly,mu – құпия кілттер, n – бүтін сан, size – мәліметтер жиынтығының ұзындығы

3.4 Микроконтроллер мен кітапхана байланысы

Әртүрлі микроконтроллердегі модификацияланған С. Ф. Кренделевтің айнымалысы бар көпмүшелер сақинасындағы гомоморфты шифрлау, модификацияланған А. Абрамовтың рационал коэффициенттері бар көпмүшелер өрісіндегі гомоморфты шифрлау, Пэйе криптожүйесінің статикалық кітапханалары құрылды.

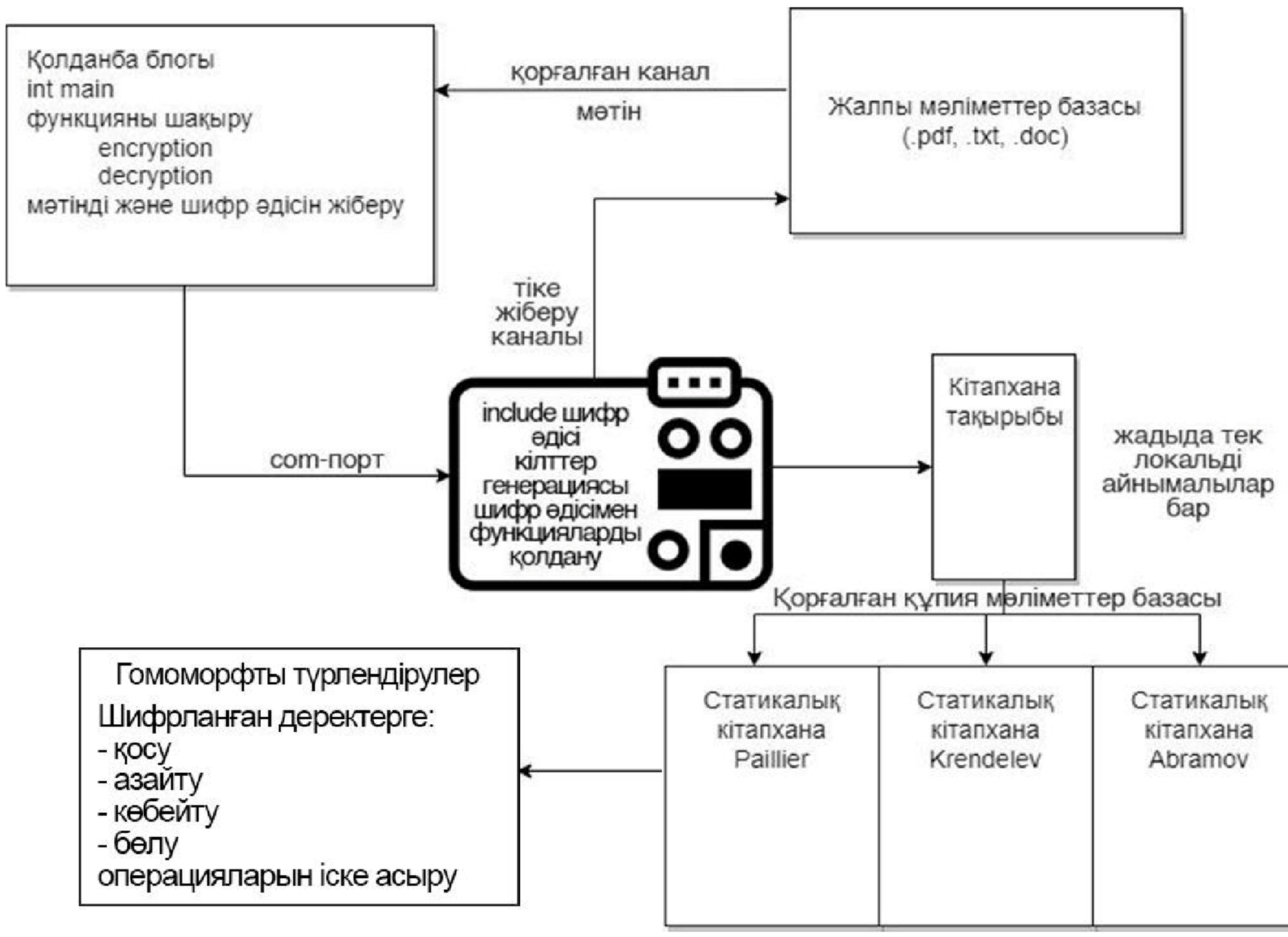
Модификацияланған гомоморфтық алгоритмдерге құрылған кітапханаларды микроконтроллердің бағдарламалық жадысына жазу үшін USBasp V 2.0 бағдарламалағышы қолданылды. AVR USBasp V 2.0 бағдарламалағышы тізбектелген бағдарламалау режимі бар барлық AVR микроконтроллерлеріне арналған қарапайым, ықшам және сенімді бағдарламалау құралы болып табылады. Ішкі сұлбалық бағдарламалау (ISP) функциясының арқасында мұндай бағдарламашыларды пайдалану микропроцессорлық құрылғыны оның қуатын өшірмей, жинақталған түрде жылдам және қайта бағдарламалауға мүмкіндік береді. Эмулятор көгілдір баспа платасына салынған, сол жақта компьютерге қосылу үшін қажет USB порты бар. ATmega8A микроконтроллері сұлбаның ортасында орналасқан, микроконтроллер жанында 12 мегагерц кварц резонаторы орнатылған. Оң жақта бағдарламаланатын контроллер мен эмулятор арасында деректер алмасуды қамтамасыз ететін 10 істікшелі (пин) порт бар. AtmelAVR микроконтроллер мен USBasp V2.0 бағдарламалағыштың қағидалық сұлбаларын 3.5 суреттен көруге болады.



Сурет 3.5 – а) Atmega328 микроконтроллер мен USBasp V2.0, ә) Atmega 32u4 микроконтроллер мен USBAsp V2.0, б) Atmega 2560 микроконтроллер мен USBasp V2.0, в)

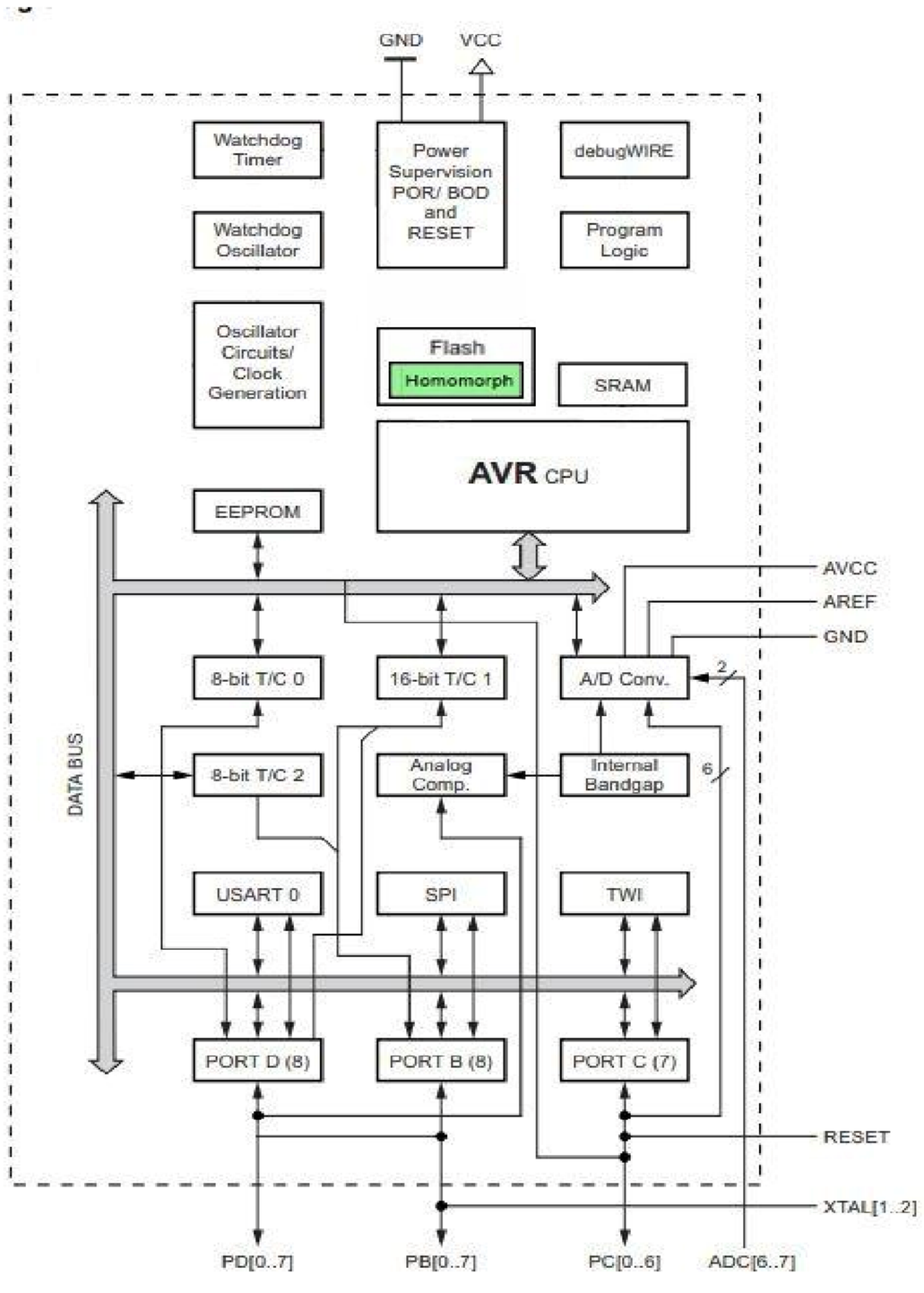
AtmelAVR тобындағы микроконтроллер мен USBasp V2.0 бағдарламалығышты SPI каналы арқылы байланысады. Байланысу келесі түрде болады: VCC мен VCC, RST мен PCINT14/Reset, SCK мен SCK/PCINT5, MISO мен MISO/PCINT5, GND пен GND, MOSI мен MOSI/OC2A жалғанады.

3.6 суреттте, қолданба блогы, мұнда негізгі функциялар шақырылады, яғни кілттерді генерациялау, шифрлау, кері шифрлау, мәтін мен шифр әдісін жіберу. Микроконтроллер мен қолданба блогы Com-порт арқылы байланысады. Негізгі операциялар микроконтроллерде орындалады. Микроконтроллер ашық және құпия кілттерді генерациялау, деректерді шифрлау, кері шифрлау, шифрланған деректерге гомомортылық (қосу, азайту, бөлу, көбейту) орындау, шифрланып қосылған, азайтылған, көбейтілген және бөлінген деректерді кері шифрлау функцияларын орындайды. Ал барлық функциялар кітапханада сақталған. Микроконтроллер кітапхананы шақыру арқылы функциялар орындалады. Жалпы мәліметтер (.pdf, .txt, doc, .png) SD модульде сақталады.

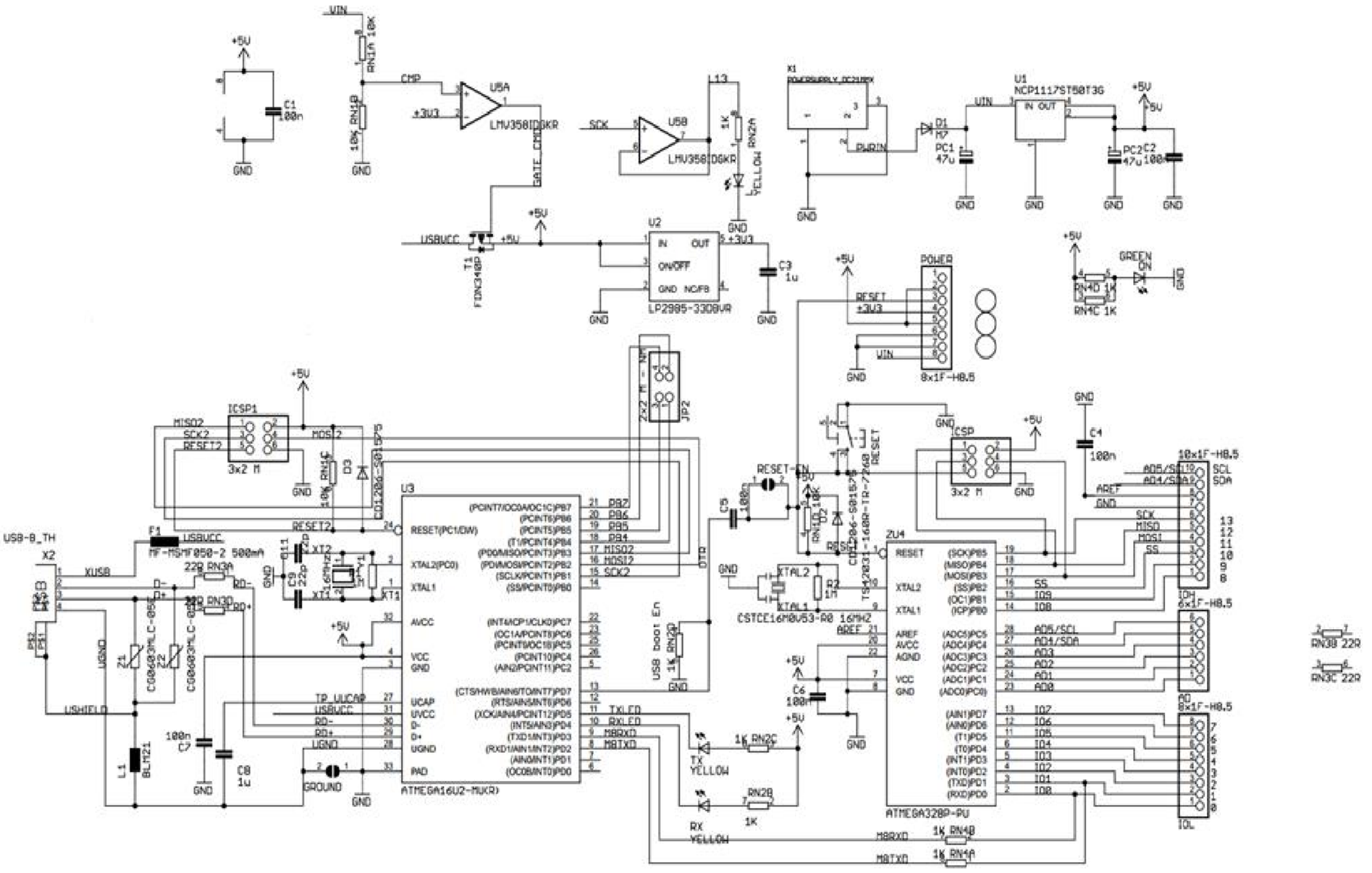


Сурет 3.6 – Микроконтроллер мен кітапхана байланысы

Құрылған кітапхананы клиенттік бағдарламада 3 түрде пайдалануға болады: 1. HomomorphicControllerVersion_01.C, яғни кітапхана компьютерде файл түрінде сақталған, микроконтроллерді компьютерге Com-порт арқылы байланыстырып, керекті функцияларды шақырады. 2. Құрылған кітапхананы GitHub-тан zip арихив түрінде жүктеп, колданушы оны драйвер түрінде орнатып қолданады. 3. Құрылған кітапхана микроконтроллер құрылығысында жазылған архитектурасы 3.7 және 3.8 суреттерден көруге болады.



Сурет 3.7 – Atmega 328 микроконтроллерінде орнатылған кітапхана архитектурасы



Сурет 3.8 – ArduinoUNO тақтасындағы Atmega328 микроконтроллерінің архитектурасы [30]

3.5 Жетілдірілген гомоморфты шифрлау алгоритмдеріне жасақтама модуль құру

Дайындалған бағдарламалық жабдықтамалар C# бағдарламалық тілінде жүзеге асырылды. Жүйелік талаптар: операциялық жүйе – Windows 7/8/10.

«HomomorphicController v1.0» бағдарламалық жабдықтамасының негізгі терезесі 3.9-суретте келтірілген.



Сурет 3.9 - HomomorphicController v1.0 алгоритм бағдарламасының негізгі терезесі

Бағдарламаны іске қосу «HomomorphicController.exe» арқылы жүзеге асырылады. Мәзір терезесі үш бөлімнен тұрады: «Файл», «Параметрлер» және «Шифрлау».

«Файл» батырмасы – бұл деректерді шифрлау, гомоморфтылық операцияларды орындау немесе кері шифрлау үшін файлдарды жүктеуге арналған диалогтық терезе.

«Параметрлер» батырмасы – кілттерді жүктеуге арналған диалогтық терезе. Бұл терезе бізге кілттерді «кездейсоқ» немесе «қолмен енгізу» әдістеріне мүмкіндік береді.

«Шифрлау» батырмасы – шифрлауды және оны іске қосуды таңдауға арналған терезе. Файлдан хабарламаны оқу, шифрлау процесі және нәтижесінде алынған шифрланған деректерді файлға сақтау процестерді жүзеге асырады.

Жетілдірілген айнымалысы бар көпмүшелер сақинасындағы ТГШ алгоритмінің шифрлау процесі:

1. Бастапқы және тұрақты деректерді енгізу. Бағдарламалық жасақтаманың енгізу мәліметтері келесілер болып табылады:

- шифрлауға арналған деректерді енгізу;
- кері шифрлауға арналған деректерді енгізу.

2. «Параметрлер» терезесінен кілт генерациясы іске қосылады. p мен q шарттарға сай болатындай таңдалып алынады. x_0 құпия кілті есептеледі ($x_0 = \frac{p}{q}$). Блок-схемасы төмендегі суретте көрсетілген (сурет 3.10 а).

3. Шифрлау блогы. Деректерді шифрлау блок схемасы сурет 3.10 ә)-де көрсетілген. Шифрлау бөлігінде хабарламаны айнымалысы бар көпмүшеге айналдыру жүзеге асырылады.

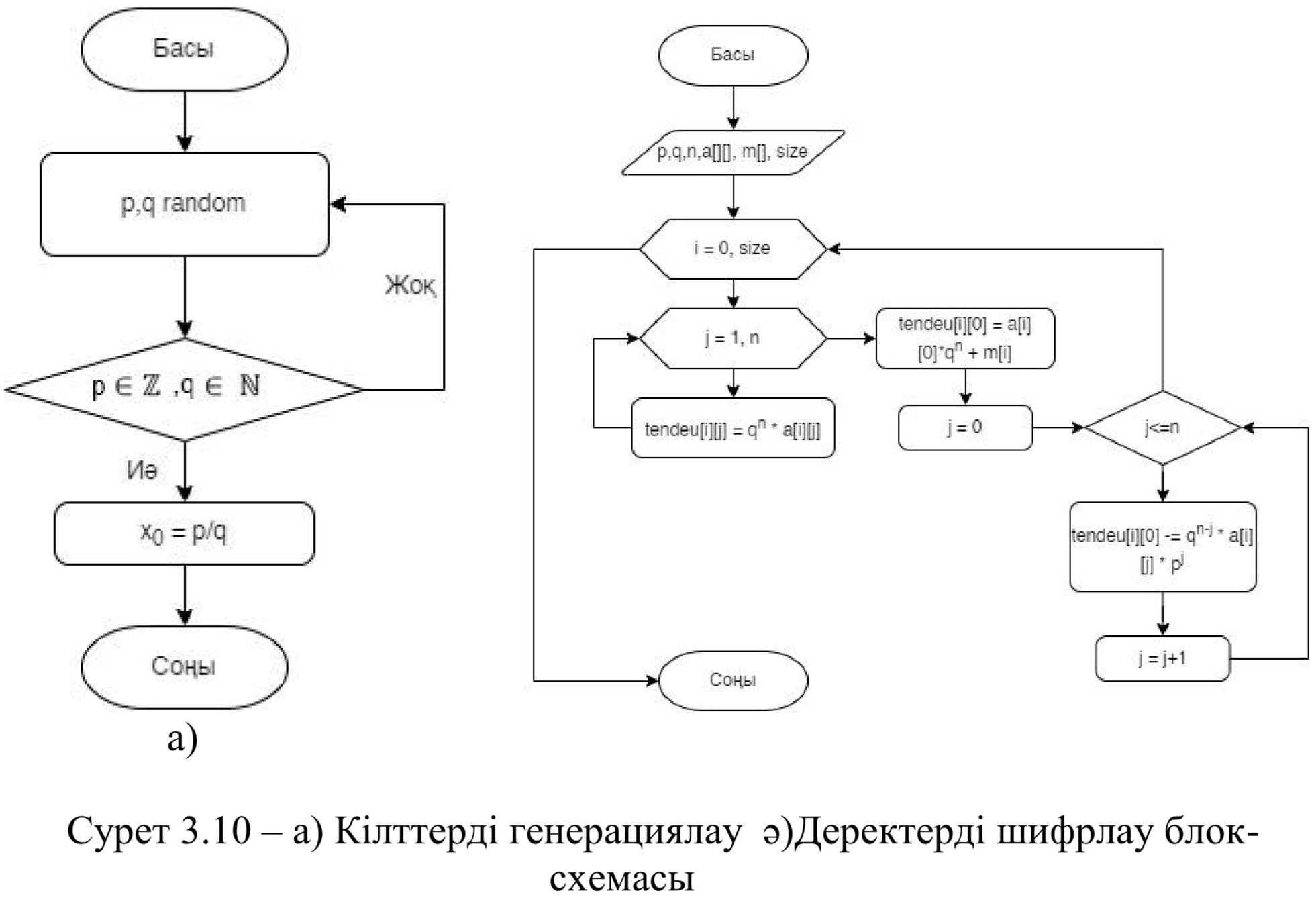
3.1 Қажетті параметрлер ретінде n және коэффициенттер $(a_0, a_1, \dots, a_n)$ енгізіледі.

3.2 Параметрлерді пайдалана отырып $f(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$ көпмүше құрылады.

3.3 p мен q арқылы $q^n f\left(\frac{p}{q}\right) = q^n a_0 + q^{n-1} p a_1 + \dots + p^n a_n$ есептеледі.

3.4 Нәтижесінде $g(x) = q^n f(x) - q^n f\left(\frac{p}{q}\right) + z$ шифрын аламыз.

3.5 Бұл қадамда 3.1.-3.4 пункттері файлдың соңғы блогы шифрланғанға дейін қайталанады.



а)

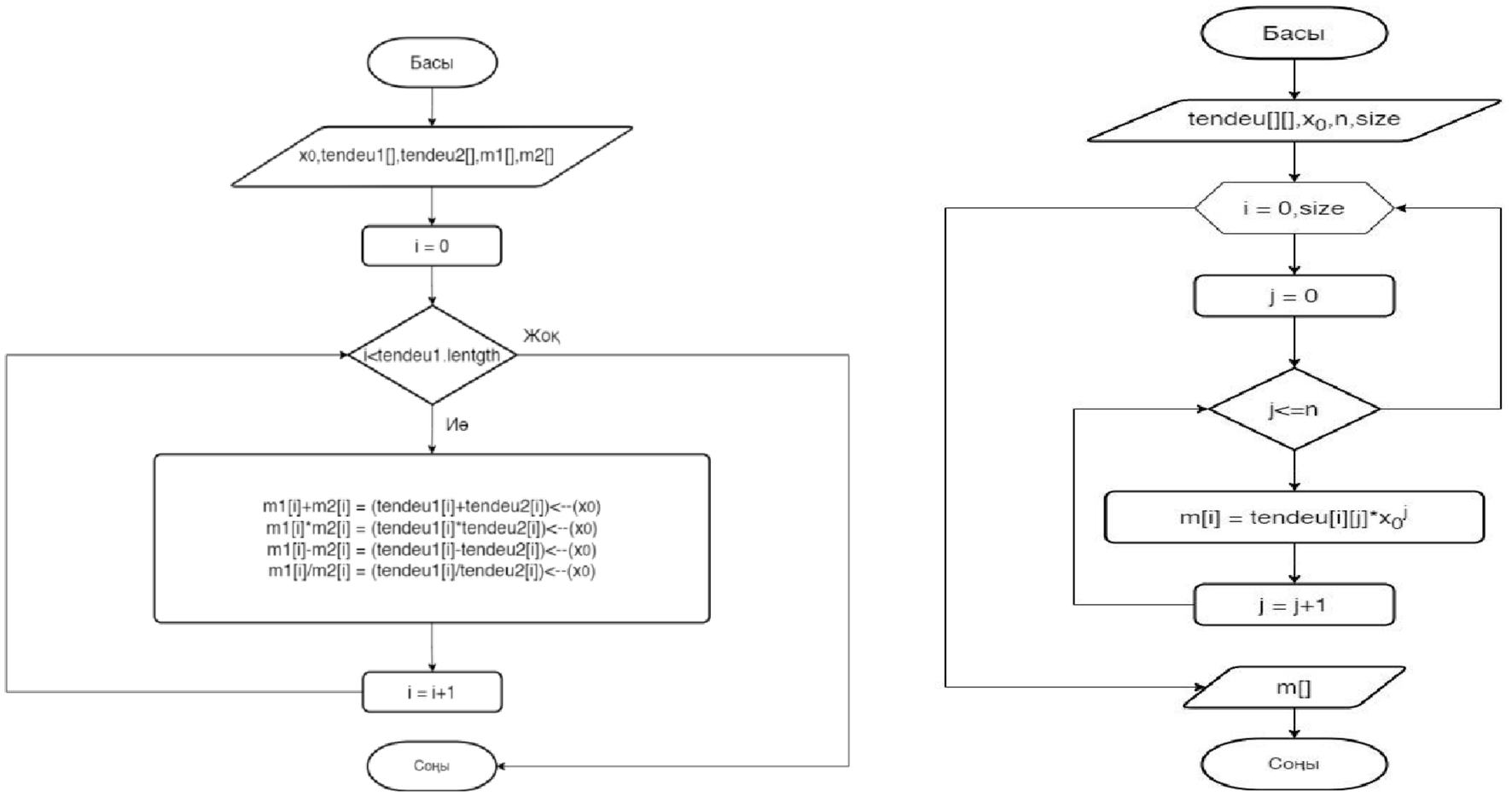
Сурет 3.10 – а) Кілттерді генерациялау ә)Деректерді шифрлау блок-схемасы

4. Гомоморфтылық қасиеті. Алгоритмнің толық гомоморфты шифрлау бөлігі бағдарламалық жабдықтаманың оң жағында орналасқан. Гомоморфты шифрлау шифрланған көпмүшелерге математикалық операциялар қолдану арқылы жүзеге асады (қосу, азайту, көбейту, бөлу). Келесі қасиеттері орындалады: $\boldsymbol{P(z_1 -_z z_2) = P(z_1) -_{z[x]} P(z_2)}$. Блок-схемасы 3.11 а) – суретте көрсетілген.

5. Кері шифрлау блогы. Құпия кілтті қолдану арқылы шифрланған деректер кері шифрланады.

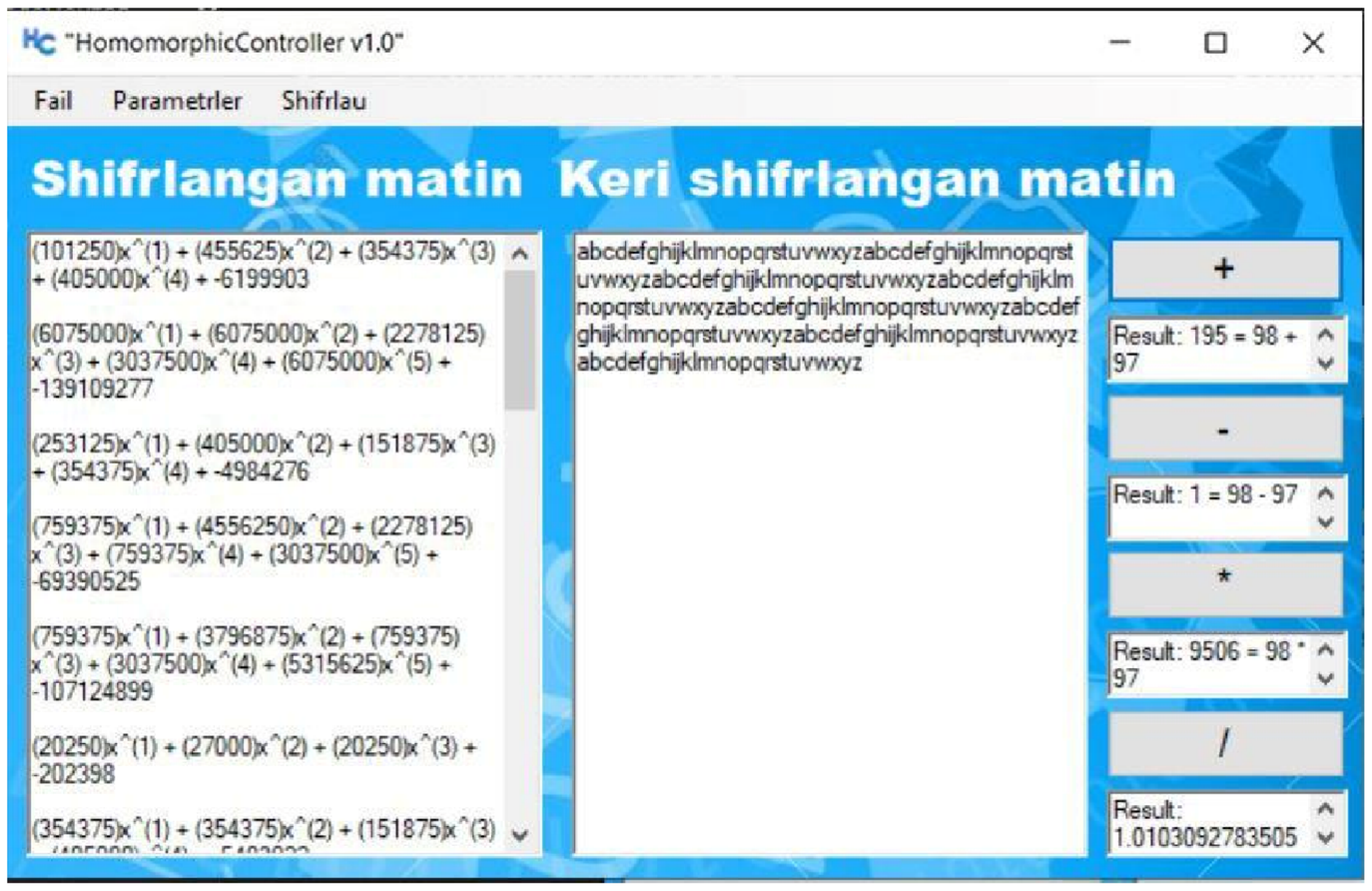
5.1 x_0 құпия кілті $g(x)$ шифрланған көпмүшелерге қолданылады.

5.2 Кері шифрлау шифрланған деректердің соңғы блогы кері шифрланып болғанға дейін орындалады. Нәтижесінде бастапқы ашық деректерге орындалған (қосындысы, айырмасы, көбейтіндісі, бөліндісі) алынады. Блок-схемасы 3.11 ә) – суретте көрсетілген.



Сурет 3.11 – а) Гомоморфтылық орындалу ә) Кері шифрлау блок схемасы

Модификацияланған айнымалысы бар көпмүшелер сақинасындағы ТГШ алгоритміне құрылған бағдарламалық жасақтаманың нәтижесі 3.12 – суретте көрсетілген.



Сурет 3.12 – Модификацияланған айнымалысы бар көпмүшелер сақинасындағы ТГШ алгоритмінің нәтижесі

Жетілдірілген рациональ коэффициенттері бар көпмүшелер өрісіндегі ТГШ алгоритміне бағдарламалық жасақтама модулін құру. Құрылған жасақтама модулі 3.15 – суретте көрсетілген.

1. Қажетті файл ашылады («Шифрлауға арналған файлды ашу» немесе «Кері шифрлауға арналған файлды ашу»).

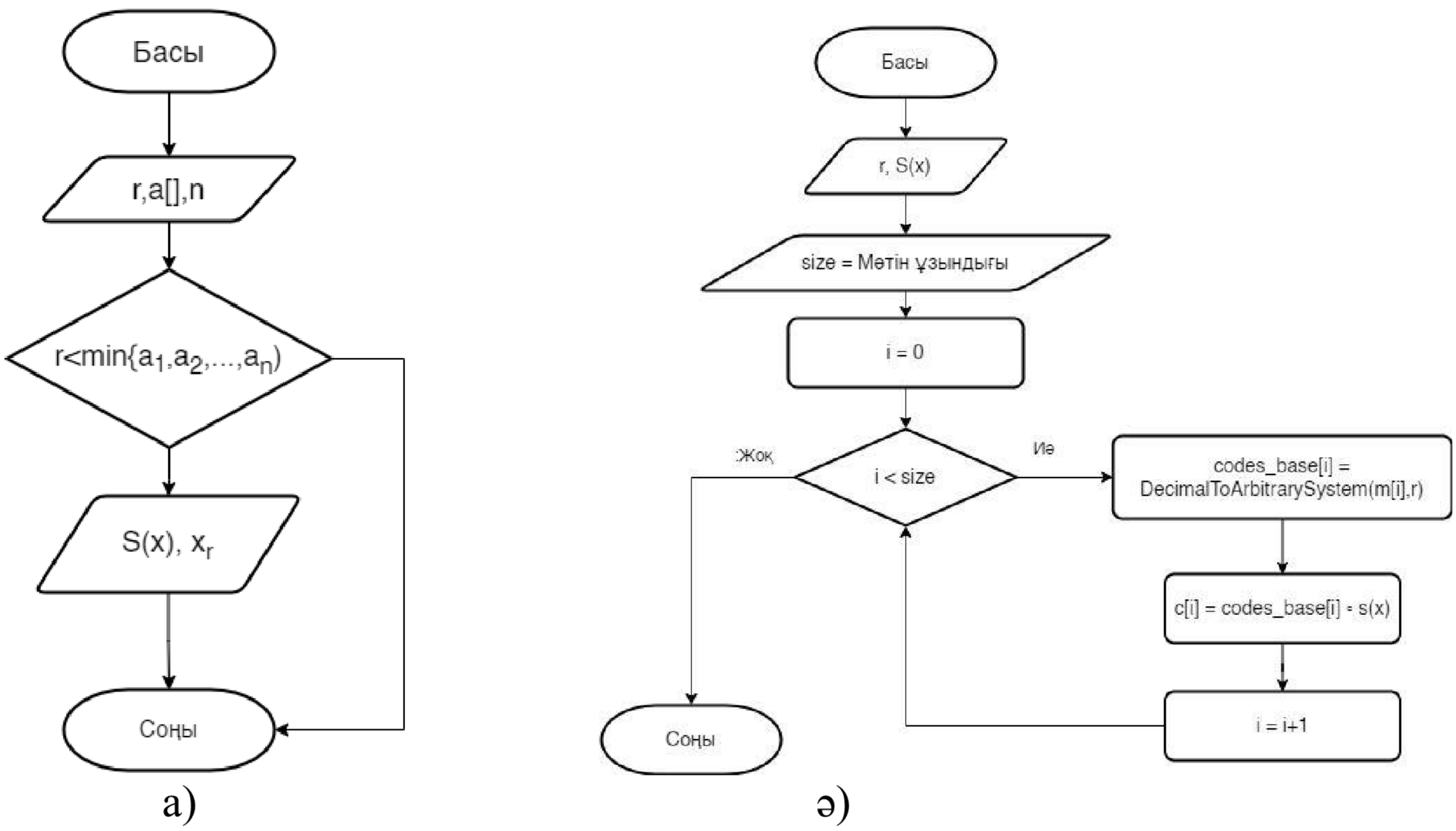
2. «Параметрлер» бөлімінен r және $S(x), x_r$ мәндері алынады, мұндағы, r және $S(x)$ ашық кілт, ал x_r құпия кілт. Шарт бойынша $r < \min\{a_1, a_2, \dots, a_n\}$. Кілттерді генерациялау блок-схемасы 3.13 а) суретте көрсетілген.

3. Шифрлау блогы. Деректерді шифрлау блок-схемасы 3.13 ә) суретте көрсетілген. Мәліметтерді шифрлау келесі қадамдардан тұрады:

3.1 Енгізілген деректерді шифрлау үшін санау жүйесі қолданылып, көпмүше құрылады. Ашық деректер r санау жүйесіне айналдырылады.

3.2 r бойынша санау жүйесінен алынған деректер, көпмүше түрінде жазылады.

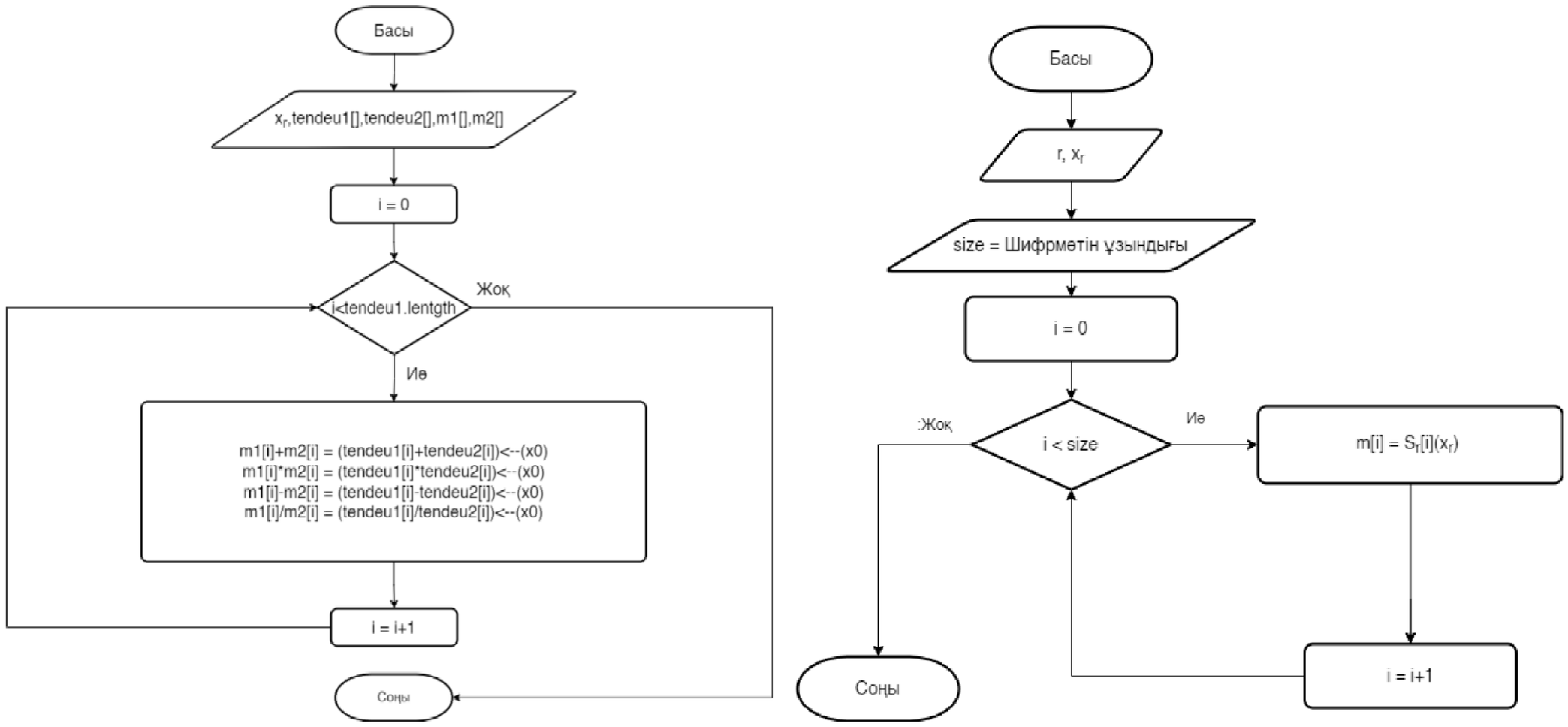
3.3 Кездейсоқ алынған $S(x)$ көпмүшесі арқылы r бойынша санау жүйесі бойынша құрылған көпмүшеге түрлендіру жасалынады.



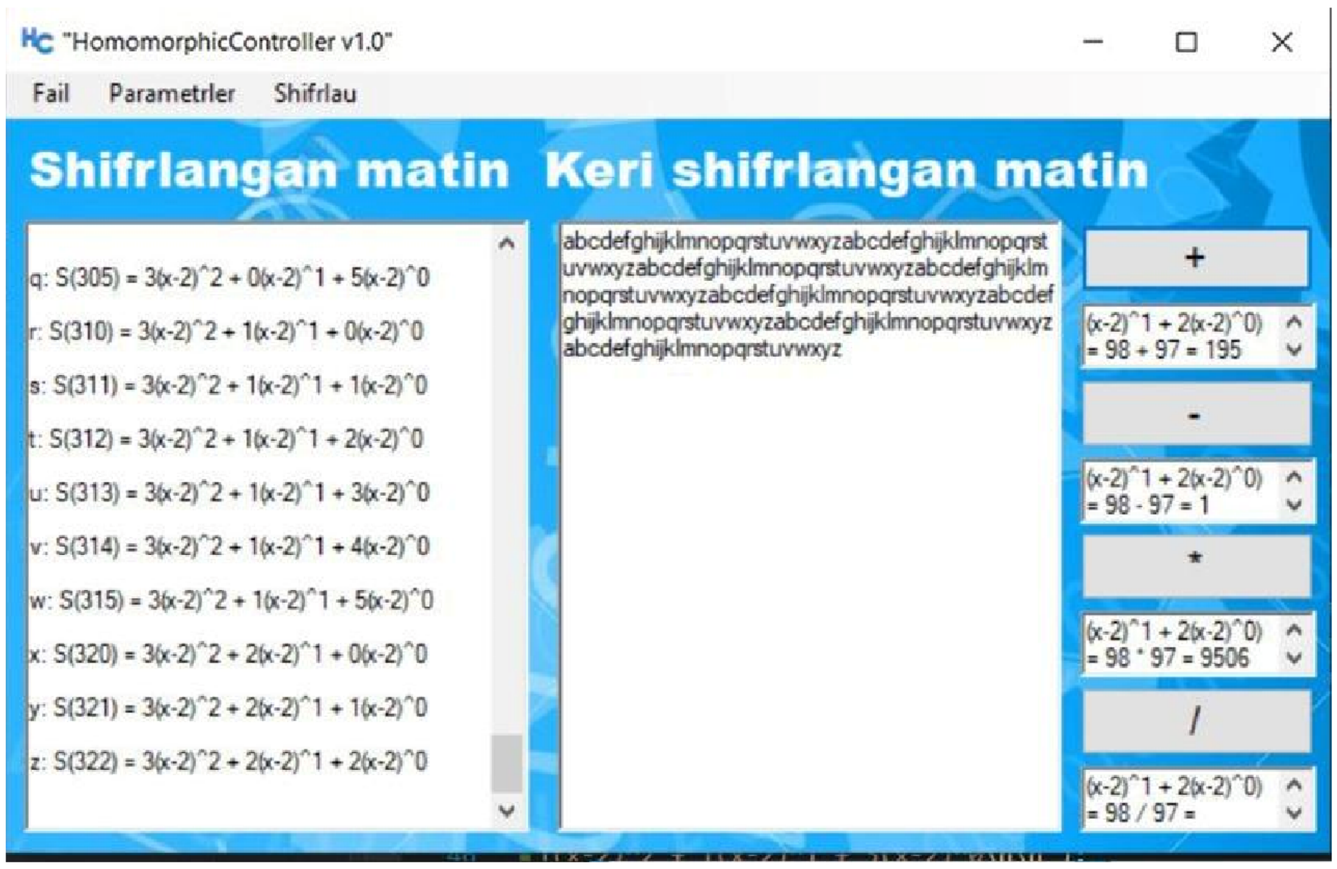
Сурет 3.13 – а)Кілттерді генерациялау ә)Деректерді шифрлау блок схемасы

4. Гомоморфтылық қасиеті. Алгоритмнің толық гомоморфты шифрлау бөлігі бағдарламалық жабдықтаманың оң жағында орналасқан. Гомоморфты шифрлау шифрланған көпмүшелерге математикалық операциялар қолдану арқылы жүзеге асады (қосу, азайту, көбейту, бөлу). Сақина гомоморфизмі бойынша келесі теңестірулер орындалады: $Hom(a -_z b) = Hom(a) -_{z[x]} Hom(b)$. Блок-схемасы 3.14 а) – суретте көрсетілген.

4. Шифрланған деректерді және шифрланған деректерге гомоморфты операцияларды кері шифрлау үшін, құпия x_r кілтін шифрланған көпмүшедегі x-тің орнына қойылады. Деректерді кері шифрлау блок-схемасы 3.14 ә) – суретте көрсетілген.



Сурет 3.14 – а) Гомоморфтылық орындалу ә) Кері шифрлау блок схемасы



Сурет 3.15 – Модификацияланған рационал коэффициенттері бар көпмүшелер өрісіндегі ТГШ алгоритмінің нәтижесі

Пэйе криптожүйесінің ТГШ алгоритміне бағдармалық жасақтама модулін құру. Құрылған бағдарламалық жасақтама 3.18 – суретте бейнеленген.

Шифрлау үдерісі:

1. Бастапқы ашық деректерді енгізу. Шифрлауға арналған файлдар енгізіледі. Файлдардан деректер оқылады.

2. «Параметрлер» бөлімінде p және q $\gcd(pq, (p-1)(q-1))$ шартына сай болғанша кездейсоқ түрде алынады.

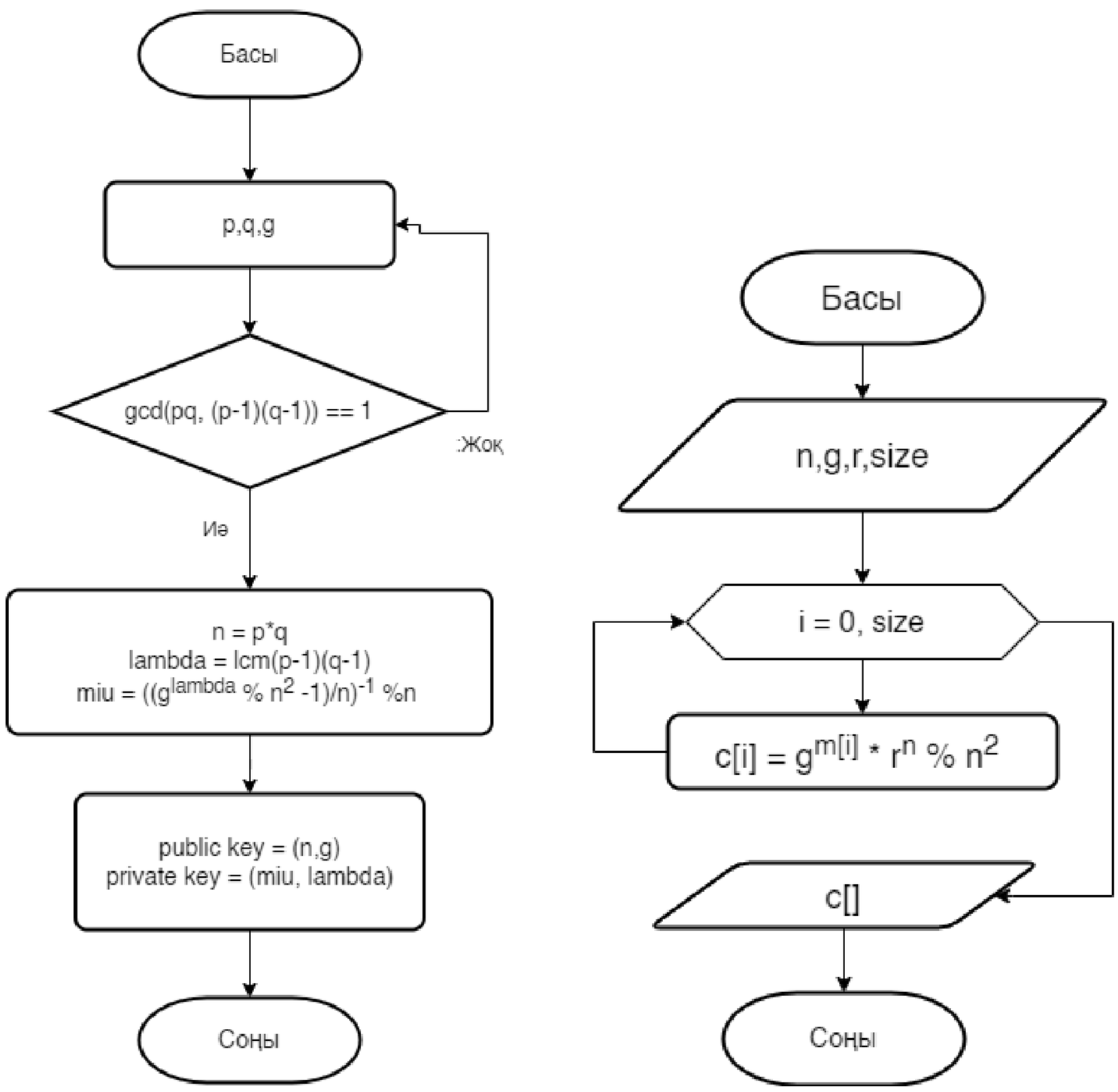
$$n = p * q, \lambda = \text{ЕКОЕ}(p-1, q-1), g = n+1, \mu = \left(\frac{g^{\lambda} mod\, n^2 - 1}{n}\right)^{-1} mod\, n$$

кілттері есептеледі, мұндағы n, g – ашық, μ, λ – құпия кілттер. Ашық және құпия кілттерді генерациялау 3.16 – суретте көрсетілген.

3. Шифрлау блогы математикалық операциялардың көмегімен жүзеге асырылады.

3.1 Кездейсоқ r таңдалады (r бүтін сан)

3.2 $c = g^m * r^n \; mod \; n^2$негізінде ашық деректер шифрланады.



Сурет 3.16 – а) Ашық және құпия кілттерді құру ә) Деректерді кері шифрлау

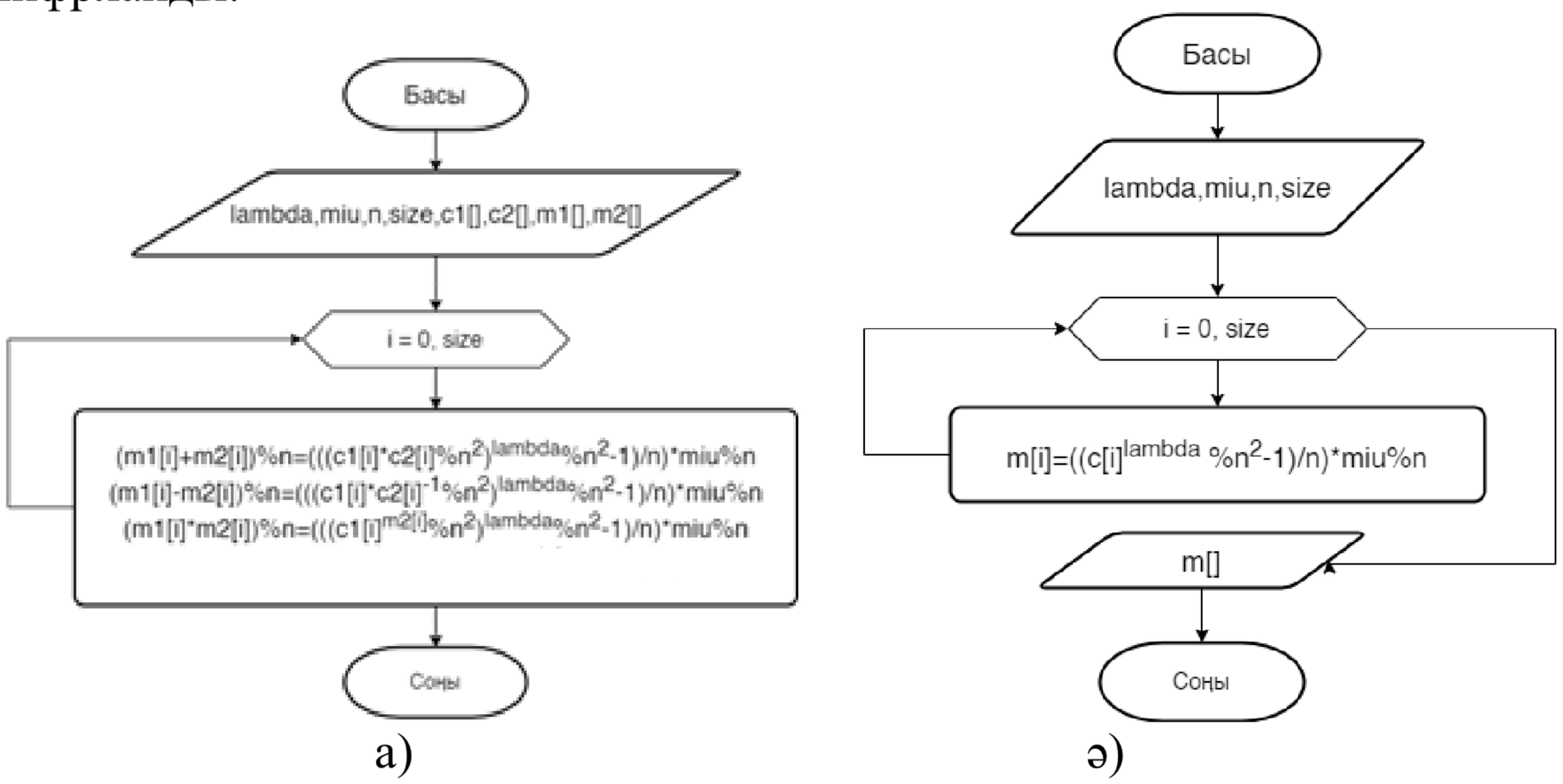
4. Шифрланған деректерге математикалық операциялар төмендегідей іске асырылады: Блок-схемасы 3.17 суретте кескінделген.

4.1 Қосу операциясы: $\frac{(c_1 * c_2 \,\%\, n^2)^{\lambda} \,\%\, n^2 - 1}{n} * \mu \,\%\, n$

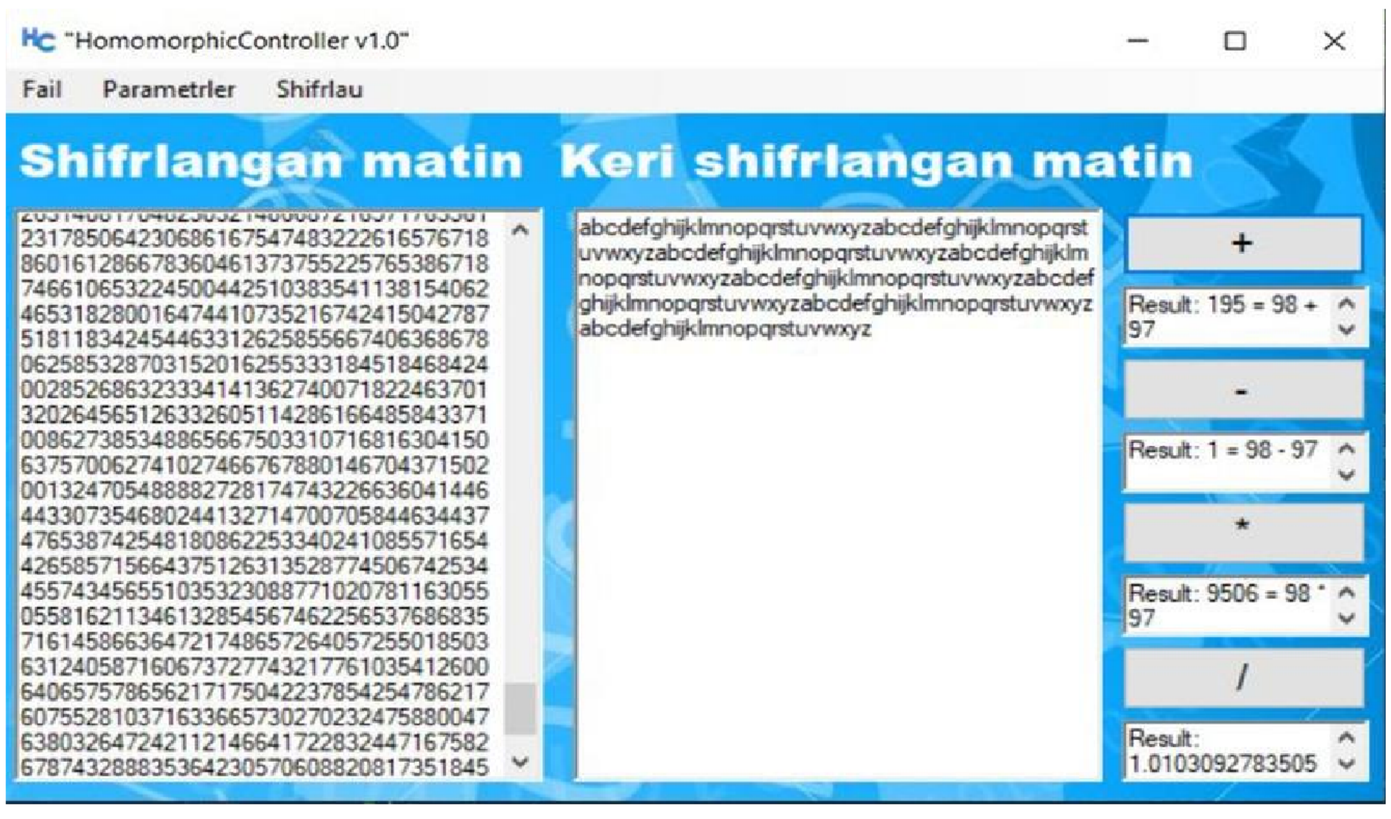
4.2 Азайту операциясы: $\frac{(c_1 * c_2^{-1} \,\%\, n^2)^{\lambda} \,\%\, n^2 - 1}{n} * \mu \,\%\, n$

4.3 Көбейту операциясы: $\frac{(c_1^{m_2} \,\%\, n^2)^{\lambda} \,\%\, n^2 - 1}{n} * \mu \,\%\, n$

4. $m = \left(\frac{c^{\lambda} mod\ n^2 - 1}{n}\right) * \mu\ mod\ n$ формуласы арқылы шифрланған деректерді кері шифрлайды.



Сурет 3.17 – а) Жүйенің гомоморфтылығы ә) Кері шифрлау блок схемасы



Сурет 3.18 –Пэйе криптожүйесінің ТГШ алгоритмінің нәтижесі

Құрылған бағдарламалық жасақтамаға «Homomorphic Controller_Version01» 2021 жылы 21 сәуірде №16878 авторлық куәлік алынды.

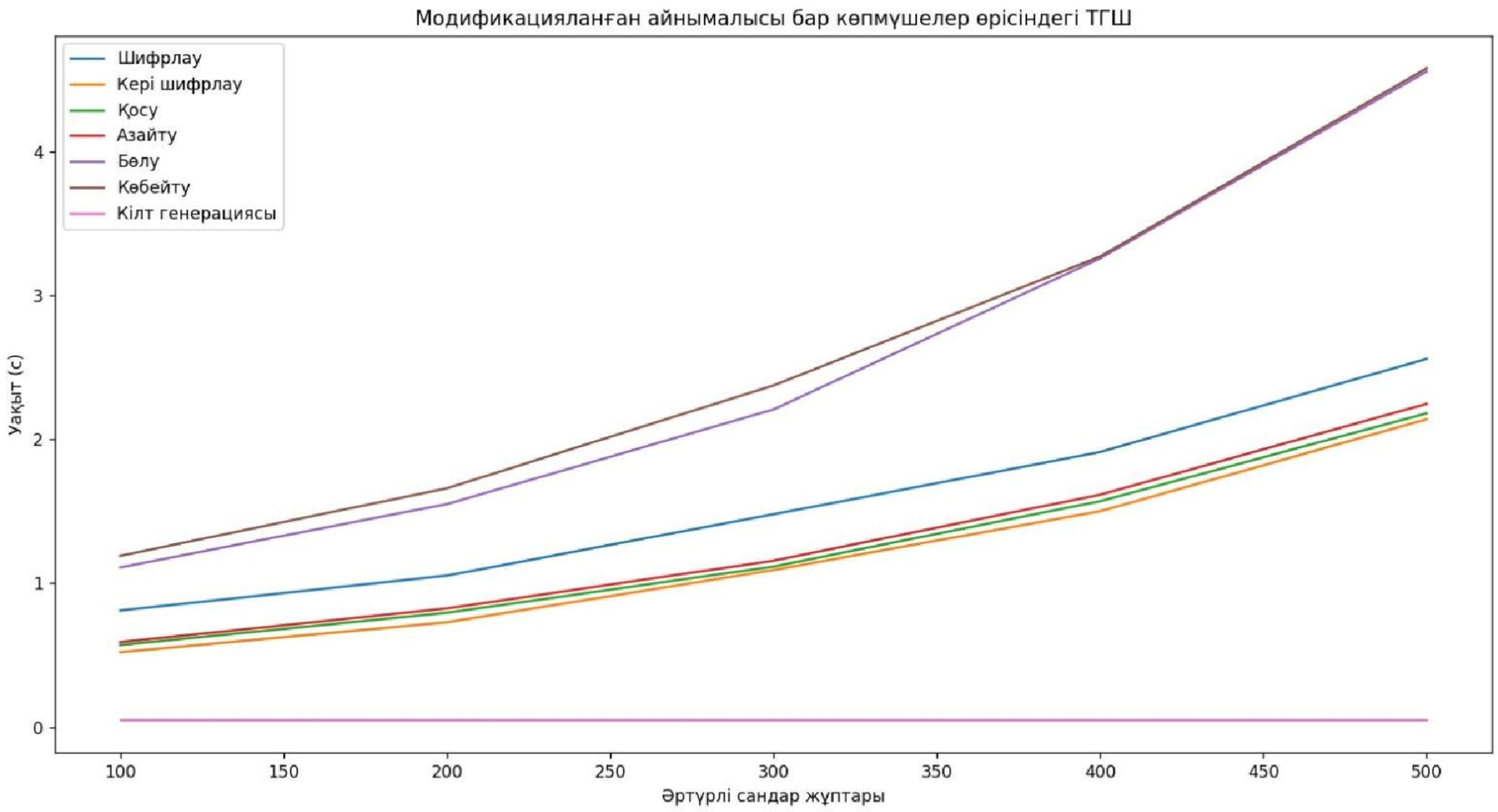
4. ATMELAVR МИКРОКОНТРОЛЛЕРІНДЕГІ КІТАПХАНА ЖҰМЫСЫНЫҢ САПАСЫН БАҒАЛАУ ЖӘНЕ СОНДАЙ-АҚ БАСҚА КІТАПХАНА ЖҰМЫСЫМЕН САЛЫСТЫРУ

4.1 Құрылған кітапханалардың өнімділігін AtmelAVR (Atmega 328) микроконтроллерде тестілеу

Зерттеу жұмысының маңызды міндетінің бірі құрылған кітапханалардың өнімділігін әртүрлі AtmelAVR микроконтроллерде бағалау болып табылады. Бастапқыда модификацияланған айнымалысы бар көпмүшелер сақинасындағы ТГШ Atmega 328 микроконтроллерде екі таңбалы әртүрлі жұп сандары бойынша: кілтті генерациялау, шифрлау, кері шифрлау, (қосу, азайту, көбейту, бөлу) уақыттарына салыстырулар жүргізіліп, алгоритм күрделігі қарастырылды. Нәтижесі 4.1 – кестеден және алгоритмнің орындалу уақытының шифрлау параметрлеріне тәуелділігін 4.1 – суреттен көруге болады.

Кесте 4.1 – Құрылған кітапхананың Atmega 328 микроконтроллерде орындалу уақыты

2 таңбалы, әртүрлі жұп сандары	Кілт генерациялау, шифрлау, қосу, кері шифрлау, с	Кілт генерациялау, шифрлау, азайту, кері шифрлау, с	Кілт генерациялау, шифрлау, көбейту, кері шифрлау, с	Кілт генерациялау, шифрлау, бөлу, кері шифрлау, с
100	1,95	1,92	2,52	2,44
200	2,625	2,605	3,44	3,33
300	3,736	3,727	4,94	4,77
400	5,032	5,027	6,68	6,67
500	6,931	6,947	9,28	9,2



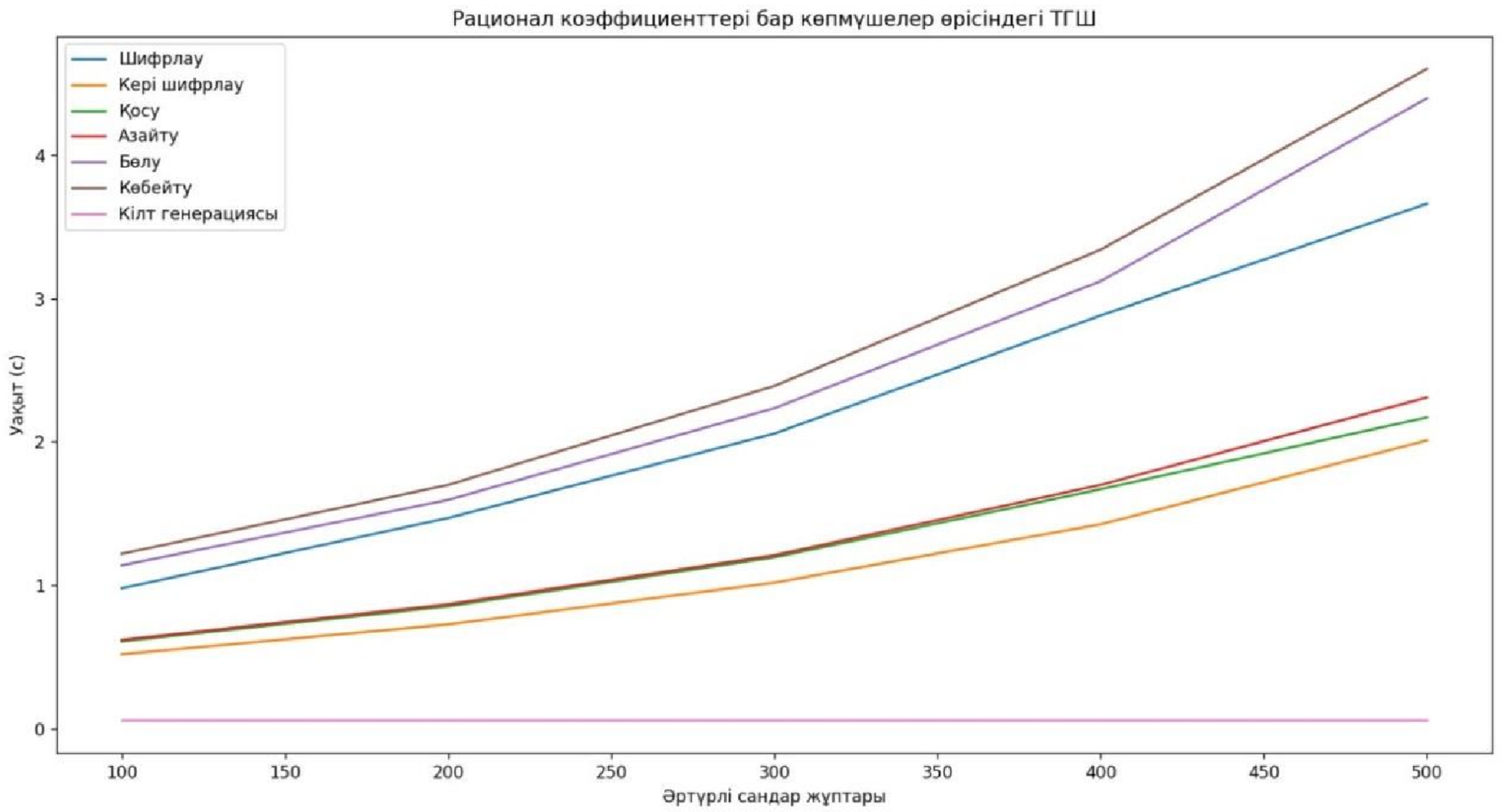
Сурет 4.1 – Алгоритмнің орындалу уақытының шифрлау параметрлеріне тәуелділігі

Қосу операциясымен көбейту операциясын салыстырғанда, көбейту операциясының қосу операциясына қарағанда нашар орындалуы алгоритмнің күрделілігі жоғары екенін көрсетеді. Сондай-ақ жады операциялары процессор уақытының көп бөлігін алатыны анықталды [95-96]. Бұл схемалар нақты бағдарламалық өнімдерде қолдануға жарамды болуы мүмкін деген қорытынды жасауға болады. График сызықтық өсуді көрсетеді, бұл қосу нәтижесінде көпмүше коэффициенттеріне бөлінген жады көлемінің ұлғаюының салдары, сондай-ақ үлкенірек ұзындықтағы деректер бойынша қосу операциясының күрделенуі. Алайда, көпмүшелерді қосу операциясы $O(n)$ алгоритмдік күрделілікке ие болғандықтан, бұл өсуді елеусіз деп санауға болады. Көбейту кезінде квадраттық өсу байқалады және қосудан айырмашылығы, жылдамдықтың айтарлықтай төмендеуі байқалады. Себебі, көпмүшелерді көбейту кезінде олардың коэффициенттері көбейтіледі, бұл коэффициенттерді қосуға қарағанда көбірек жады бөлуді қажет етеді, ал көпмүшелерді көбейту операциясы жоғары алгоритмдік күрделілікке ие болады - $O(n^2)$.

Келесі қадамда, рационал коэффициенттері бар көпмүшелер өрісіндегі ТГШ Atmega 328 микроконтроллерде екі таңбалы әртүрлі жұп сандары бойынша: кілтті генерациялау, шифрлау, кері шифрлау, (қосу, азайту, көбейту, бөлу) уақыттарына салыстырулар жүргізіліп, алгоритм күрделігі қарастырылды. Нәтижесі 4.2 – кестеден және алгоритмнің орындалу уақытының шифрлау параметрлеріне тәуелділігін 4.2 – суреттен көруге болады.

Кесте 4.2 – Құрылған кітапхананың Atmega 328 микроконтроллерде орындалу уақыты

2 таңбалы, әртүрлі жұп сандары	Кілт генерациялау, шифрлау, кері шифрлау, қосу, с	Кілт генерациялау, шифрлау, кері шифрлау, азайту, с	Кілт генерациялау, шифрлау, кері шифрлау, көбейту, с	Кілт генерациялау, шифрлау, кері шифрлау, бөлу, с
100	2,17	2,18	2,78	2,7
200	3,11	3,12	3,95	3,85
300	4,33	4,34	5,52	5,37
400	6,03	6,06	7,7	7,48
500	7,9	8,04	10,33	10,12



Сурет 4.2 – Алгоритмнің орындалу уақытының шифрлау параметрлеріне тәуелділігі

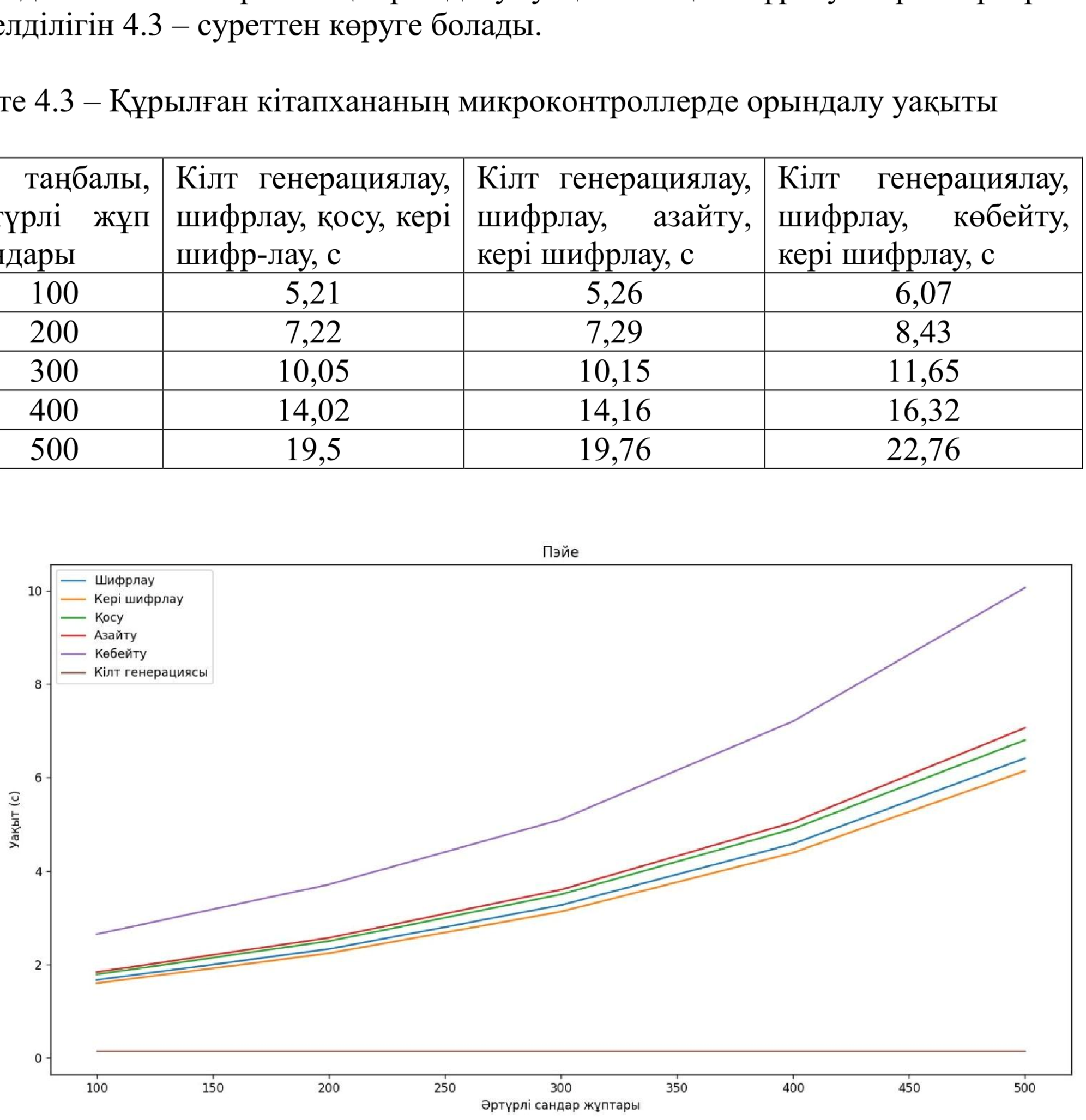
Шифрлау операциясының орындалу уақытының ашық деректердің көлеміне тәуелділігі анықталды. Санау жүйесінің мәндері және шифрлау кезіндегі көпмүшелік дәрежесі кездейсоқ болды. 4.2-суретте график нәтижені көрсетілді. График бойынша, мәндер сызықты өсуде, яғни нақты жүйелерде қолдануға қолайлы екенін көрсетеді. Шифрланған деректердегі қосу және көбейту амалдарын салыстыру қарастырылды. Тестілеуді жеңілдету үшін деректер бірдей, алдын ала анықталған құпия кілтпен шифрланған. Классикалық алгоритмдер *O(n)*-де көпмүшелерді қосуды орындайды, бұл алынған нәтижелермен көрсетілді. Көпмүше ретінде ұсынылған шифрланған деректермен операциялар санының ұлғаюымен уақыттың сызықтық өсуі байқалады. Бұл көпмүшелердің коэффициенттері үшін бөлінген жады

көлемінің ұлғаюына және үлкен өлшемді сандарға қосу операциясының күрделілігінің артуына байланысты. Шифрланған деректерге көбейту операциясы бойынша, қосу операциясынан айырмашылығы, көпмүшелерді көбейтудің квадраттық орындалу уақыты $O(n^2)$ – амалдар санының ұлғаюымен орындалу уақытының жоғары өсуін байқауға болады, бұл қарастырылып отырған жүйенің негізгі кемшіліктерінің бірі болып табылады.

Келесі қадамда, модификацияланған Пэйе криптожүйесі ТГШ Atmega 328 микроконтроллерде екі таңбалы әртүрлі жұп сандары бойынша: кілтті генерациялау, шифрлау, кері шифрлау, (қосу, азайту, көбейту, бөлу) уақыттарына салыстырулар жүргізіліп, алгоритм күрделігі қарастырылды. Нәтижесі 4.3 – кестеден және алгоритмнің орындалу уақытының шифрлау параметрлеріне тәуелділігін 4.3 – суреттен көруге болады.

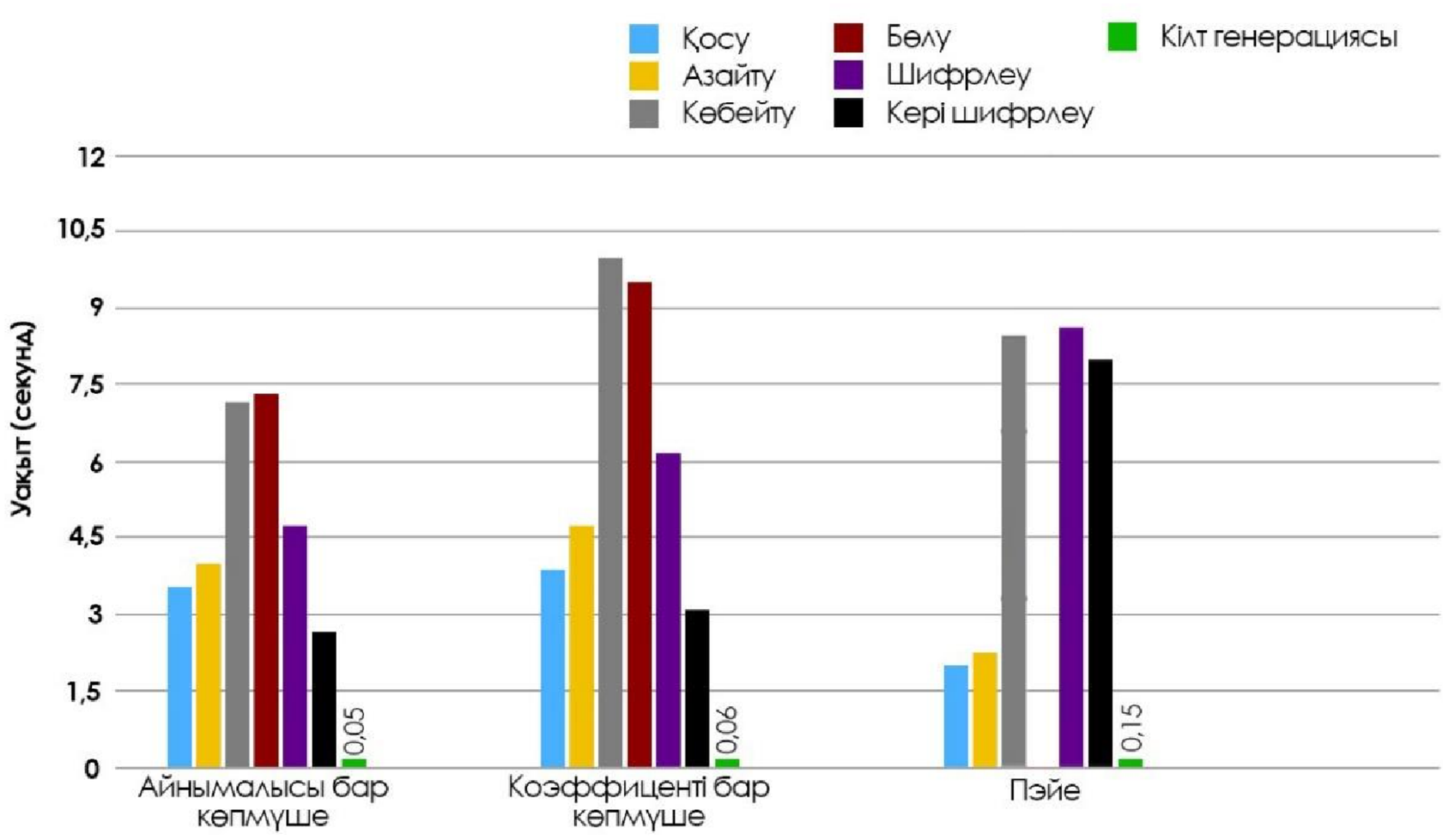
Кесте 4.3 – Құрылған кітапхананың микроконтроллерде орындалу уақыты

2 таңбалы, әртүрлі жұп сандары	Кілт генерациялау, шифрлау, қосу, кері шифр-лау, с	Кілт генерациялау, шифрлау, азайту, кері шифрлау, с	Кілт генерациялау, шифрлау, көбейту, кері шифрлау, с
100	5,21	5,26	6,07
200	7,22	7,29	8,43
300	10,05	10,15	11,65
400	14,02	14,16	16,32
500	19,5	19,76	22,76



Сурет 4.3 – Алгоритмнің орындалу уақытының шифрлау параметрлеріне тәуелділігі

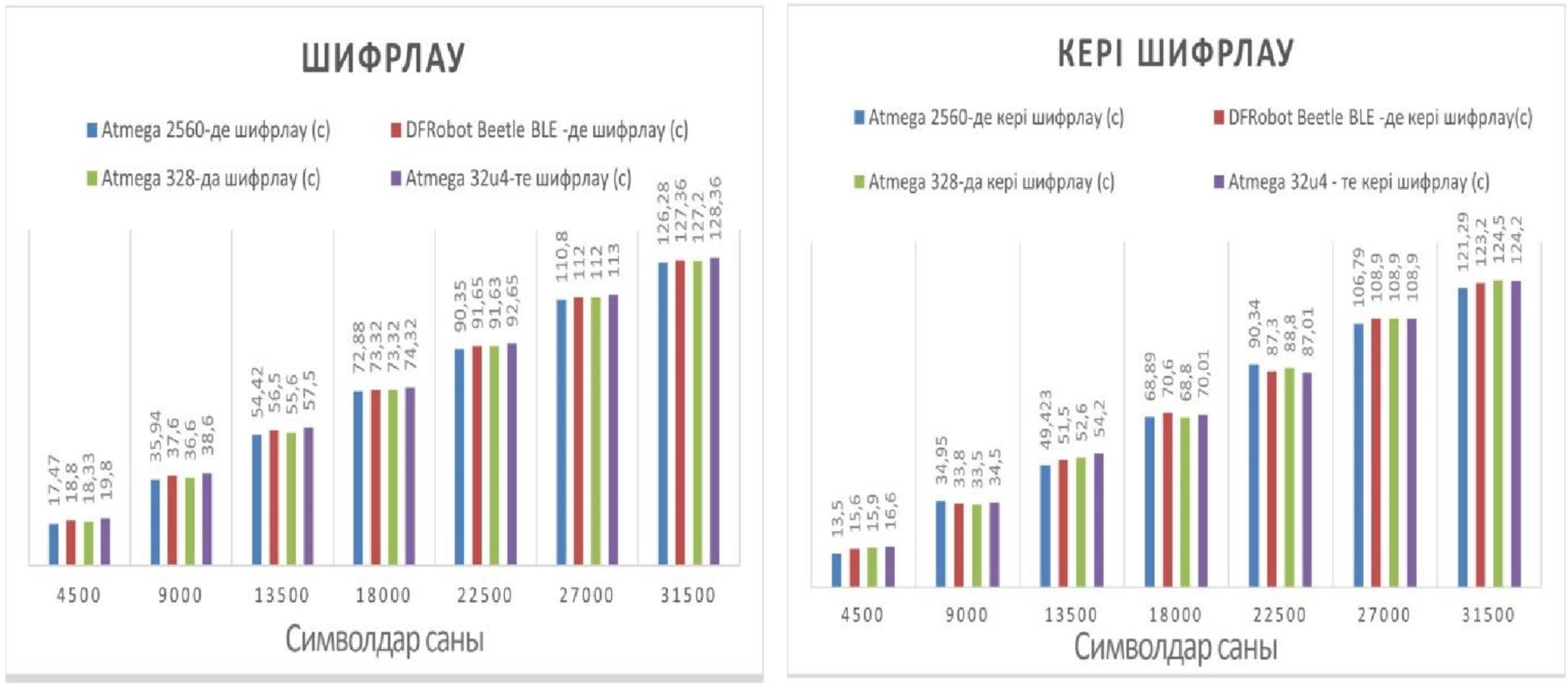
Пэйе алгоритмінде қосу және азайту операцияларының алгоритм күрделілігі $O(1)$ болады. Себебі, қосу операциясында екі шифрды көбейтіп, n^2 бөлгендегі қалдық есептелінеді. Азайту операциясында бір шифрды екіншісіне бөлініп, қалдық есептелінеді. Көбейту операциясында дәреже қолданғандықтан, санның n дәрежесін табу алгоритмінің күрделілігі $O(n)$ болып табылады. Демек, көбейту операциясында хабарламаны n ретінде алынса, алгоритм күрделілігі $O(n)$ болады. Бөлу операциясы көбейтумен сәйкес, тек бұл жерде бір операцияға артық орындалады, себебі шифрдың $-n$ дәрежесі есептелінеді. Демек, алдымен шифрдың n дәрежесін есептеп алып, соңында 1-ді алынған нәтижеге бөлінеді. Осыған байланысты, бөлу операциясының күрделілігі $O(n+1)$.



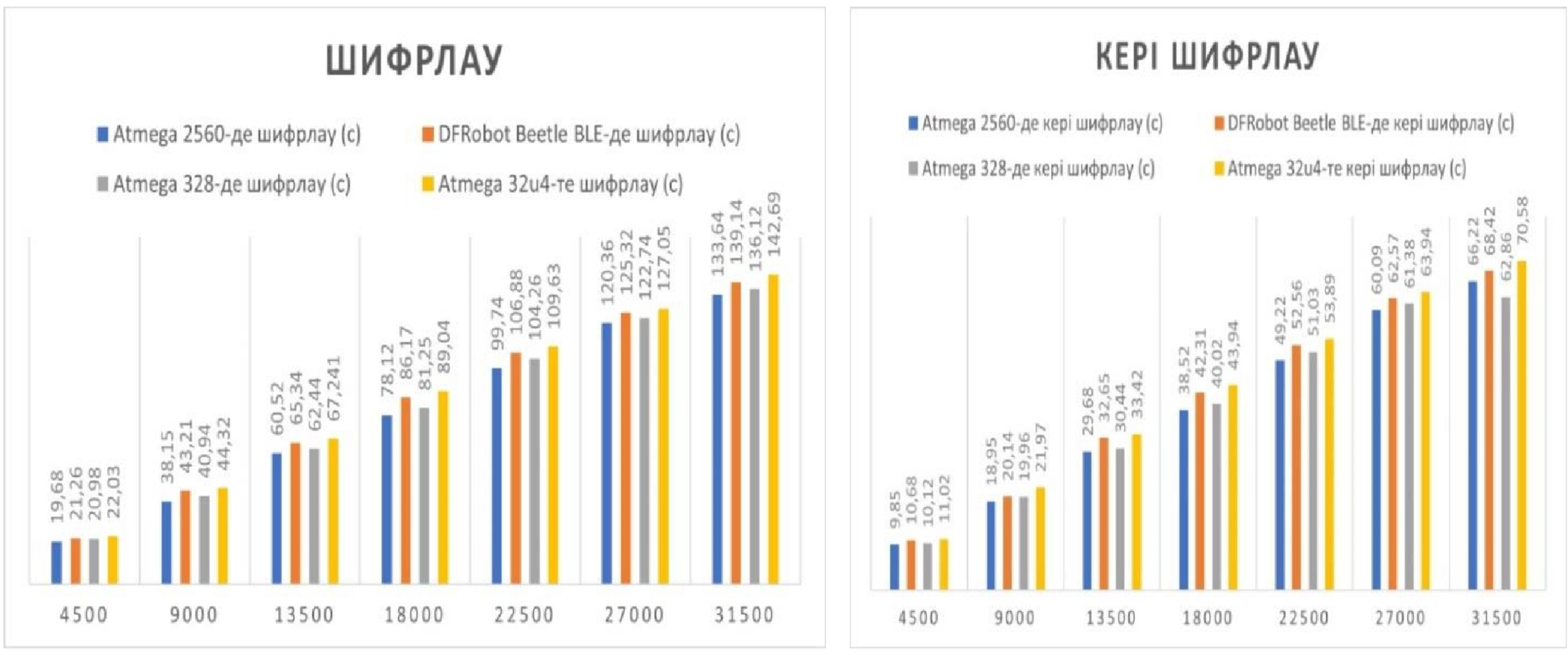
Сурет 4.4 – 500 жұп бойынша Atmega 328 микроконтроллерде үш ТГШ алгоритмдердің жылдамдығын салыстыру

Барлық өткізілген сынақтар ашық деректердің 500 жұп бойынша орындалды. Модификацияланған Гомоморфты шифрлау алгоритмдердің өнімділігін салыстыру барысында ең жылдам қосу, азайту операцияларына ие - Пэйе. Ең жылдам шифрлау, кері шифрлау, көбейту, бөлу және кілт генерациялау операциясы модификацияланған айнымалысы бар көпмүше. Пэйе криптожүйесі асимметриялық болғандықтан кілттерді генерациялау үшін салыстырмалы түрде ұзақтау уақыт алатыны көрінді.

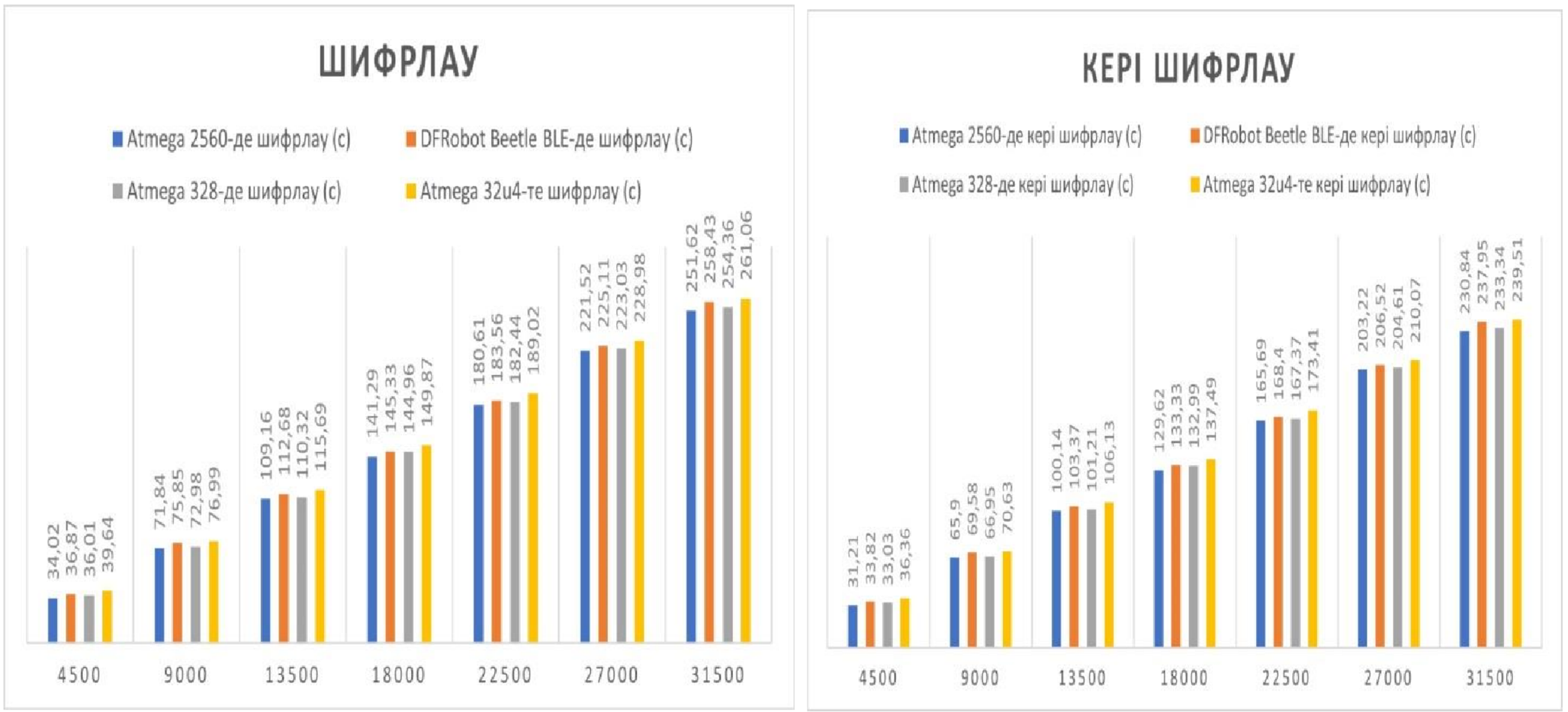
Модификацияланған ТГШ алгоритмдерін әртүрлі AtmelAVR микроконтроллерінде салыстырулар жүргізілді. Салыстыру нәтижелері 4.5, 4.6, 4.7 – суреттерде көрсетілді. Нәтиже бойынша шифрлау және кері шифрлау уақыттары сызықты өседі.



Сурет 4.5 – Айнымалысы бар көпмүшелер сақинасындағы ТГ шифрлау және кері шифрлау жылдамдығын әртүлі микроконтроллерде бағалау



Сурет 4.6 – Рационал коэффициенттері бар көпмүшелер өрісіндегі ТГ шифрлау және кері шифрлау жылдамдығын микроконтроллерде бағалау



Сурет 4.7 – Пэйе шифрлау және кері шифрлау жылдамдығын микроконтроллерде бағалау

4.2 Құрылған кітапханаларды басқа авторлардың жұмыстарымен салыстырулар жүргізу

Толық гомоморфты шифрлау кітапханасын жасау, тек теориялық криптографиядағы ғылыми жаңалық емес, сонымен қатар құпия деректерді практикалық қорғау және өңдеу саласындағы перспективті даму болып табылады. Гомоморфты шифрлау арқылы құрастырылған технологиялар өңделген жеке деректерге қауіп төндіретін тәуекелдерді азайтуға мүмкіндік береді және адам қызметінің көптеген салаларында қолданылуы мүмкін.

Кесте 4.4 – Негізгі функциялары бойынша гомоморфты шифрлау кітапханаларын салыстыру

Негізгі ерекшеліктері	SEAL, Microsoft Research, 2020[79]	Helib, Shai Halevi, et al., 2020 [80]	TFHE, Ilaria Chillotti, et al., 2021 [81]	Fan-Vercauteren	Пэйе [82]	Айны-малысы бар көп-мүшелер [82]	Рацио-нал коэф. бар көп-мүшелер [83]
Ассиметриялық	Иә	Иә	Иә		Иә	Жоқ	Иә
Шифрланған мәтін өлшемі (1 енгізу үшін 1МБ кем емес)	Жоқ	Жоқ	Иә		Иә	Иә	Иә

2 ГБ-тан аз жедел жадта жұмыс істей алады.	Жоқ	Иә	Иә		Иә	Иә	Иә
Қосу, азайту	Иә	Иә	Иә		Иә	Иә	Иә
Көбейту	Иә	Иә	Иә		Иә	Иә	Иә
Бөлу	Жоқ	Жоқ	Жоқ		Жоқ	Иә	Иә
Биттік операциялар	Иә	Иә	Иә		Жоқ	Жоқ	Жоқ
Бүтін, нақты сандармен жұмыс	Жоқ	Жоқ	Жоқ		Иә	Иә	Иә
Микроконтроллер	Жоқ	Жоқ	Жоқ		Иә	Иә	Иә
Бағдарламалау тілі	C++, Python	C++, Python	C		C++	C++	C++

Гомоморфты операциялар әдетте екі шифрланған мәтін арасында орындалады. Егер операндтардың бірі ашық мәтін болуы мүмкін болса, ол өнімділікті айтарлықтай жақсартуы мүмкін. Нәтижедегі шифрлық мәтіннің өлшемі енгізілген шифр мәтінімен бірдей болып қалады және қайта сызықтандыру қадамын өткізіп жіберуге болады. SEAL, Helib, TFHE ашық мәтінмен шифрланған мәтінді қосу, алу және көбейту функцияларын қамтамасыз етеді [97-100]. Диссертациялық жұмыста ұсынылған модификацияланған айнымалысы бар көпмүшелер, модификацияланған рационал коэффициенттері бар көпмүшелер кітапханалары шифрланған мәтінді қосу, азайту және көбейту мен қоса бөлу арифметикалық амалдарын қолдайды. Қазіргі таңдағы гомоморфты шифрлау кітапханалар микроконтроллер үшін жасалмаған. Ұсынылған кітапхана микроконтроллер мен компьютер үшін қолдануға болады.

Екі көпмүшені қосқанда олардың нәтижесі қосылған мүшелердің дәрежесінен аспайтынын, ал көбейтілгенде көбейткіштердің дәрежелерінің қосындысына тең болады. Осыдан келесідей қорытынды жасауға болады: орындалатын қосындылардың саны есептеу өнімділігіне кері әсер етпейді, ал көбейту есептеу өнімділігін баяулауына әкеледі. Дәрежелердің өсу қарқынын азайту үшін бастапқы шифрлау көпмүшелері ретінде минималды дәрежелі көпмүшелерді құру ұсынылады.

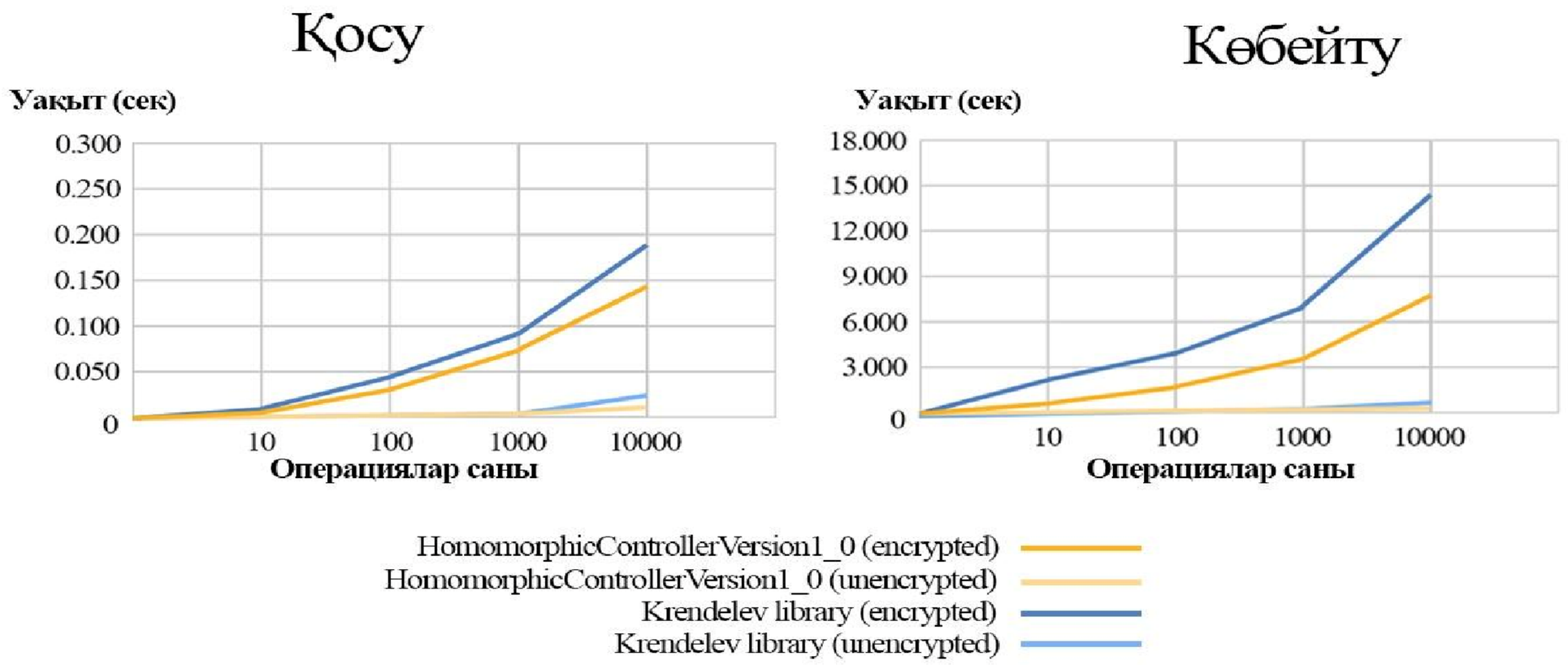
Кітапхананың өнімділігін тестілеу үшін C++ тілінде GMP ұзын арифметикалық кітапхана қолданылды [100-104]. GMP кітапханасы бүтін таңбалы, рационал және нақты сандармен жұмыс жасайды, ерікті дәлдіктегі арифметикаға арналған тегін кітапхана болып табылады. GMP сонымен қатар әртүрлі математикалық функциялардың жиынтығына ие.

GMP кез келген өлшемдегі сандар операциялары үшін жоғары өнімділікті қамтамасыз етеді. Операциялардың жоғары жылдамдығына негізгі арифметикалық тип ретінде толық машина сөзін қолдану арқылы қол жеткізіледі, әртүрлі процессорлар архитектурасы үшін жүзеге асырылатын алгоритмдердің ассемблер кірістірулерін және есептеулер жылдамдығын арттыруға бағытталған басқа да оңтайландыруларды қолдану арқылы жеделдетіледі. Жұмыста Microsoft Visual C++ компиляторы үшін әзірленген GMP кітапханасының MPIR1 нұсқасы пайдаланылды.

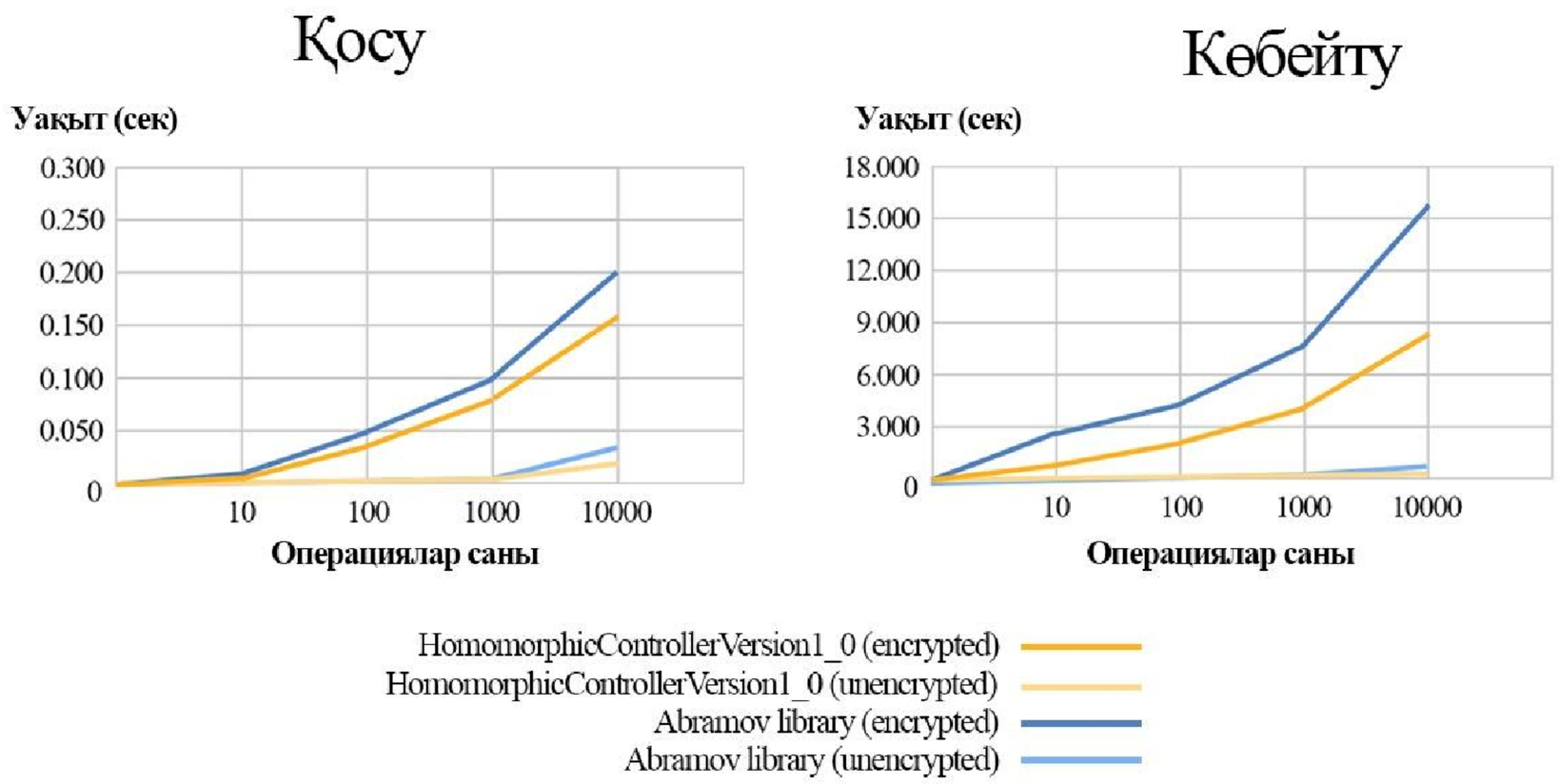
Көпмүшелер коэффициенттерін сақтау үшін C++ тіліндегі стандартты кітапханасы std::vector класымен басқарылатын динамикалық бөлінетін жад пайдаланылды.

Шифрланған деректер бойынша көпмүшелерді қосу, көбейту, азайту және бөлу операцияларына есептеулер орындалды. Барлық арифметикалық операторлар екі нұсқада жүзеге асырылды: нәтижені жаңа объектке орналастырумен (+, -, *, /) және сол операндқа нәтижені тағайындаумен (+=, -=, *=, / =). Көпмүшелердің әр мүшесін есептеу кезінде операторлардың соңғы түрін пайдалану бөлінген жад көлемін азайтуға және сол арқылы есептеулердің өнімділігін арттыруға мүмкіндік береді. Функциядағы жергілікті айнымалылар саны аз болса, компилятор оларды регистрлерге енгізе алады. Параметрлер операторларға сілтеме арқылы беріледі. Барлық жергілікті айнымалылар регистрлерде болады, сондықтан оларға жадтан қол жеткізуге қарағанда өнімділігі жоғары болады.

Шифрланған және шифрланбаған деректер үшін бірдей операцияның орындалу уақыттарына тестілеу жүргізілді. Құрылған кітапхананың өнімділігін тестілеу Intel(R) Core(TM) i3-3220CPU 3,30 ГГц процессоры және 4 ГБ жедел жады бар Windows 10 компьютерінде жасалынды.



Сурет 4.8 – Кітапхананың өнімділігін салыстыру (Айнымалысы бар көпмүшелер сақинасындағы ТГШ)



Сурет 4.9 – Кітапхананың өнімділігін салыстыру (Рационал коэффициенттері бар көпмүшелер өрісіндегі ТГШ)

Шифрланған деректер мен шифрланбаған деректерге қосу және көбейту операцияларының өнімділіктері салыстырылды; 10^8 итерацияның орташа жұмыс уақыты есептелінді. Салыстыру нәтижесінде С.Ф. Кренделев пен А.Абрамов алгоритмдерге қарағанда құрылған кітапхананың өнімділігі 1.5 есе жылдам екенін сурет 4.8 және сурет 4.9-дан көруге болады.

ҚОРЫТЫНДЫ

Диссертацияда AtmelAVR микроконтроллерінде модификацияланған толық гомоморфты шифрлау мәселелері зерттеліп, келесідей нәтижелерге қол жеткізілді:

1. IoT құрылғылар кластеріндегі деректерді қорғауға арналған әдістер мен құрылғыларға талдаулар жасалынды;

2. AtmelAVR микроконтроллерінде қолданылатын гомоморфты шифрлау кітапханасы жетілдірілді;

3. IoT құрылғылар кластерінің қауіпсіздігін қамтамасыз ету үшін AtmelAVR микроконтроллерінде кітапхана архитектурасы құрылды;

4. AtmelAVR микроконтроллеріндегі кітапхана жұмысының сапасын бағалау және сондай-ақ басқа кітапхана жұмысымен салыстырылды.

Қазіргі уақытта ақпарат алмасу кезінде деректерді қорғау дәстүрлі желілер үшін ғана емес, сонымен қатар заттар интернетінің қарқынды дамып келе жатқан сегменті үшін де маңызды міндеттердің бірі болып табылады. Диссертациялық жұмыста AtmelAVR микроконтроллерлерінің пайдаланушыларына деректердің қауіпсіздігін қамтамасыз ету мақсатында HomomorphicController_Version01 кітапханасы іске асырылды. Ол әртүрлі AtmelAVR микроконтроллерлер отбасылары үшін құрастырылған шифрлау кітапханасының файлдарын қамтиды.

Бұл жұмыстың аясында бүтін сандарды гомоморфты бөлу мәселесі қарастырылып, гомоморфты бөлуді жүзеге асыру әдісі ұсынылып, сипатталған және жоғарыда аталған әдісті қолданудың практикалық мысалдары келтірілген. Жұмыстың практикалық құндылығы гомоморфты шифрлау мәселелерінің бірін шешуде жатыр - гомоморфты бөлуді жүзеге асыру гомоморфты шифрлауды бұлттық есептеулер, ақпаратты қорғау мәселелерін шешу және машиналық оқыту сияқты салаларда практикалық қолдану аясын кеңейтуге мүмкіндік береді.

ПАЙДАЛАНЫЛҒАН ӘДЕБИЕТТЕР ТІЗІМІ

1. Smith Sam. IoT Connections to reach 83 billion by 2024 driven by maturing industrial use cases [Internet], https://www.juniperresearch.com/press/iot-connections-to-reach-83-bn-by-2024

2. Brakerski, Z.; Gentry, C.; Vaikuntanathan, V. (Leveled) fully homomorphic encryption without bootstrapping. In Proceedings of the Conference on Innovations in Theoretical Computer Science (ITCS), Cambridge, MA, USA, 8–10 January 2012; ACM: New York, NY, USA, 2012; pp. 309–325.

3. Iliar Cilotti, Shea Sharon. IoT security (internet of things security). [Internet], https://www.techtarget.com/iotagenda/definition/IoT-security-Internet-of-Things-security

4. Hu Wang, Yan Yao, Xiaojun Tong, Qinghua Luo, Xiangyu Chen. “Dynamically Reconfigurable Encryption and Decryption System Design for the Internet of Things Information Security,” Sensors (Basel), vol. 1, p.143, January 2019; doi: 10.3390/s19010143

5. Zhang D.G., Dong D.C., Peng H.T., Research on development of embedded uninterruptable power supply system for IOT-based mobile service, “Computers and Electrical Engineering”, Vol. 38, No. 6, 2020, 1377–1387. DOI: 10.1016/j.compeleceng.2012.04.001.

6. Бабенко П. К. Пакетное симметричное полностью гомоморфное шифрование на основе матричных полиномов // Труды Института системного программирования РАН. 2014. Т. 26. № -5. С. 99–116.

7. Kirichek R., Koucheryavy A. Internet of Things Laboratory Test Bed // Lecture notes in Electrical Engineering. 2020. Vol. 348. pp. 485–494.

8. Кучерявый А. Е. Обзор возможных видов и методов тестирования Интернет Вещей // Информационные технологии и телекоммуникации. 2020. Том 4. № 2. С. 1–11.

9. M. Abomhara and G. M. Koien, "Security and privacy in the Internet of Things: Current status and open issues", *Int'l Conference on Privacy and Security in Mobile Systems (PRISMS)*, pp. 1-8, 2021

10. P. N. Mahalle, B. Anggorojati, N. R. Prasad and R. Prasad, "Identity authentication and capability based access control (iacac) for the internet of things", *J. of Cyber Security and Mobility*, vol. 1, pp. 309-348, 2021.

11. Brakerski, Z.; Vaikuntanathan, V. Fully homomorphic encryption from ring-LWE and security for key dependent messages. In Proceedings of the International Cryptology Conference (CRYPTO) 2011, Santa Barbara, CA, USA, 14–18 August 2011; pp. 505–524.

12. Brakerski, Z.; Vaikuntanathan, V. Efficient fully homomorphic encryption from (standard) LWE. In Proceedings of the IEEE Symposium on Foundations of Computer Science (FOCS) 2011, Palm Springs, CA, USA, 22–25 October 2011; pp. 97–106

13. Renesas. How to solve the 6 top security challenges of embedded IoT design. [Internet], https://www.renesas.com/us/en/document/whp/how-solve-6-top-security-challenges

14. Gentry, C.; Halevi, S.; Smart, N.P. Fully homomorphic encryption with polylog overhead. In Proceedings of the International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT) 2012, LNCS 7237, Cambridge, UK, 15–19 April 2012; pp. 465–482.

15. Halevi, S.; Shoup, V. Algorithms in HElib. In Proceedings of the International Cryptology Conference (CRYPTO) 2014, Santa Barbara, CA, USA, 17–21 August 2014; pp. 554–571.

16. Васильев А. Е., Микроконтроллеры: разработка встраиваемых приложений, изд. «БХВ-Петербург» 2018

17. Микушин А. Занимательно о микроконтроллерах. — М.: БХВ-Петербург, 2016. — ISBN 5-94157-571-8.

18. Жан М. Рабаи, Ананта Чандракасан, Боривож Николич. Цифровые интегральные схемы. Методология проектирования = Digital Integrated Circuits. — 2-е изд. — М.: Вильямс, 2007. — ISBN 0-13-090996-3.

19. Amgad Muneer , Suliman Mohamed Fati , Saddam Fuddah. “Smart health monitoring system using IoT based smart fitness mirror,” TELKOMNIKA Telecommunication, Computing, Electronics and Control, vol. 18, pp. 317-331, Feb 2020.

20. Королев Н. Atmel: 32-разрядные Flash-микроконтроллеры на ядре AVR32 // Компоненты и технологии. 2018. № 11

21. Королев Н., Шабынин А. Архитектура AVR: развитие вширь и вглубь. Часть 1 // Компоненты и технологии. 2017. № 2.

22. Матвеенко, И.П. Методика изучения микроконтроллеров AVR. / И.П. Матвеенко – «Информатизация образования», № 2. – Р 2016.– С.86-95.

23. С.Сокол «Миниатюрные USB-программаторы для микроконтроллеров AVR» –Радио, 2012, №2, с.27-30

24. Евстифеев А.В. Микроконтроллеры AVR семейства Mega. Руководство пользователя. — М.: Издательский дом «Додэка-XXI», 2017. — 592 с: ил.

25. Ревич Ю.В. Практическое программирование микроконтроллеров Atmel AVR на языке ассемблера.- Спб.: БХВ-Петербург,2008. 384 с

26. Хартов В.Я. Микроконтроллеры AVR. Практикум для начинающих. - М.: Изд-во МГТУ им. Н.Э. Баумана, 2007. - 240 с: ил.

27. Atmel 8-bit AVR Microcontroller with 16/32/64K Bytes In-System Programmable Flash, AT-mega164P/V, ATmega324P/V, 644P/V data sheet: 8011I-AVR-05/08, Atmel Corporation, 2325 Orchard Parkway, San Jose, CA 95131.

28. S. Barrett and D. Pack, Microcontroller Fundamentals for Engineers and Scientists, Morgan & Claypool Publishers, 2006.

29. Barrett S, PackD (2006) Microcontrollers Fundamentals for Engineers and Scientists. Morgan and Claypool Publishers. DOI: 10.2200/S00025ED1V01Y200605DCS001

30. Steven F. Barrett. (2013) Arduino Microcontroller Processing for Everyone! Third Edition. Synthesis Lectures on Digital Circuits and Systems 8:4, 1-513.

31. Dung Dang, Daniel J. Pack, Steven F. Barrett. (2016) Embedded Systems Design with the Texas Instruments MSP432 32-bit Processor. Synthesis Lectures on Digital Circuits and Systems 11:3, 1-574.

32. Ganry, Nicolas , Tinlot, Eric Atmel's Rad Tolerant, MCU, ATmegaS128 AVR Available and Perspectives

DASIA 2016, Proceedings of the Conference held 10-12 May, 2016 at Tallinn in Estonia. Edited by L. Ouwehand. ISBN: 978-92-9221-301-5. ESA-SP Vol. 736, 2016, id.33

33. Kim, K.; Choi, S.; Kwon, H.; Liu, Z.; Seo, H. FACE–LIGHT: Fast AES–CTR Mode Encryption for Low-End Microcontrollers. In Proceedings of the International Conference on Information Security and Cryptology, Seoul, Korea, 4–6 December 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 102–114

34. Kshetri N. The evolution of the internet of things industry and market in China: An interplay of institutions demands and supply //Telecommunications Policy. – 2017. – Т. 41. – №. 1. – С. 49-67.

35. Lu Y., Liu Y., Li G., Song G., Liu M., Liu W., Design and application of an automatic packaging machine controller based on ATmega 128, [in:] The 2nd International Conference on Computer and Automation Engineering (ICCAE), 2010, Singapore.

36. Atmel Corporation, July 2010. 8-bit AVR Instruction Set. URL: http://www.atmel.com/dyn/resources/prod_documents/doc0856.pdf.

37. Temirbekova Zh.E “Programming microcontroller AVR Atmega8” // «Көліктегі инновациялық технологиялар: білім, ғылым, тәжірибе» атты XLI Халықаралық ғылыми-практикалық конференцияның материалдары, 3-4 сәуір 2017 ж., Алматы, Казахстан, (I том), 102-104 б.

38. Espressif Systems. ESP32 Series Datasheet. Espressif Systems 53. — «Deleted content about temperature sensor;».

39. Espressif Systems. ESP32-PICO-D4 Datasheet (21 августа 2017).

40. Jim Lindblom. Enginursday: First Impressions of the ESP32. Sparkfun Electronics (21 января 2020).

41. Digi International Inc. (2015). digi.com [Online]. Available: https:// www.digi.com/pdf/ds_xbeewifis6b.pdf

42. Espressif Systems. (2017, April 11) espressif.com [Online]. Available: https://espressif.com/sites/default/files/documentation/ esp32_datasheet_en.pdf

43. N. Kolban, Kolban’s Book on ESP32, USA: Leanpub, 2017

44. Jack J Purdum. Beginning C for Arduino: learn C programming for the Arduino. — 2015.

45. Jorge R Castro. Building a home security system with Arduino: design, build, and maintain a home security system with Arduino Uno. — 2015.

46. Massimo Banzi, Michael Shiloh. Getting started with Arduino. — 2014. — ISBN 978-1-4493-6331-4, 978-1-4493-6329-1.

47. Pyrkova A.Yu, Temirbekova Zh.E. “Using microcontrollers to ensure data security”, International Research Conference on Technology, Science, Engineering and Economy held Seattle, USA, February 28th, 2020, P. 52-60

48. Федухин А.В., Пасько В.П., Муха Ар.А. К вопросу моделирования надежности восстанавливаемой квазимостиковой структуры с учетом тренда параметров надежности составных частей. Математичні машини і системи. 2016. № 1. С. 158–167.

49. Pyrkova A.Yu., Temirbekova Zh.E. “Possibilities of using a BLE Nano Kit microcontroller to develop cryptographic libraries” // Вестник КазНИТУ:, - 2018. №2, – С. 477 - 481

50. Temirbekova Zh.E “Programming microcontroller AVR Atmega8” // «Көліктегі инновациялық технологиялар: білім, ғылым, тәжірибе» атты XLI Халықаралық ғылыми-практикалық конференцияның материалдары, 3-4 сәуір 2017 ж., Алматы, Казахстан, (I том), 102-104 б.

51. Pyrkova A.Yu, Temirbekova Zh.E “Use homomorphic encryption for data security” // «Көліктегі инновациялық технологиялар: білі, ғылым, тәжірибе» атты XLII Халықаралық ғылыми-практикалық конференцияның материалдары, 2018 ж., Алматы, Казахстан, (I том), 83 -85 б.

52. Rivest, R. L. On data banks and privacy homomorphism / R. L. Rivest, L. Adleman, M. L. Dertouzos // Foundations of secure computation. -1978. –Vol. 32, no. 4. –Pp. 169–178.

53. Мао В. Современная криптография: Теория и практика / пер. Д. А. Клюшина — М.: Вильямс, 2005. — 768 с. — ISBN 978-5-8459-0847-6

54. Буртыка Ф. Б. Симметричное полностью гомоморфное шифрование с использованием неприводимых матричных полиномов // Известия ЮФУ. Технические науки. 2014. № 8. С. 107–122.

55. Буртыка Ф. Б. Пакетное симметричное полностью гомоморфное шифрование на основе матричных полиномов // Труды Института системного программирования РАН. 2014. Т. 26. № -5. С. 99–116.

56. Буртыка Ф. Б. Симметричное полностью гомоморфное шифрование с использованием неприводимых матричных полиномов // Известия ЮФУ. Технические науки. 2014. № 8. С. 107–122.

57. Salim, Mikail Mohammed, Inyeung Kim, Umarov Doniyor, Changhoon Lee, and Jong Hyuk Park. 2021. "Homomorphic Encryption Based Privacy-Preservation for IoMT" *Applied Sciences* 11, no. 18: 8757. https://doi.org/10.3390/app11188757

58. Трепачева А. В. Криптоанализ симметричных полностью гомоморфных линейных криптосистем на основе задачи факторизации чисел // Известия ЮФУ. Технические науки. 2015. № 5 (166). С. 89–102.

59. Трепачева А. В., Бабенко Л. К. Формальный криптоанализ полностью гомоморфных систем, использующих задачу факторизации чисел // Информационное противодействие угрозам терроризма. 2015. № 24. С. 283–286.

60. Gentry, Craig. "Fully homomorphic encryption using ideal lattices," In Proceedings of the forty-first annual ACM symposium on Theory of computing, pp. 169-178. 2009.

61. Y. NarasimhaRao, P. Surya Chandra, V. Revathi, N. Suresh Kumar. “Providing enhanced security in IoT based smart weather system,” Indonesian

Journal of Electrical Engineering and Computer Science, vol. 18, No. 1, April 2020, pp. 9~15

62. Варновский Н. П., Шокуров А. В. Гомоморфное шифрование // Труды Института системного программирования РАН. 2007. № 12. С. 27–36.

63. Gentry, Craig, and Dan Boneh. “A fully homomorphic encryption scheme,” PhD thesis, Stanford: Stanford university, 2009.

64. Варновский Н. П., Мартишин С. А., Храпченко М. В., Шокуров А. В. Пороговые системы гомоморфного шифрования и защита информации в облачных вычислениях // Программирование. 2015. № 4. С. 47–51.

65. Van Dijk, Marten, Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. "Fully homomorphic encryption over the integers." In Annual International Conference on the Theory and Applications of Cryptographic Techniques, vol. 12, pp. 24-43. 2010.

66. Оценка стойкости криптосистемы Эль-Гамаля // Технические науки в России и за рубежом: материалы IV междунар науч. конф. (г. Москва, январь 2015 г.). М.: БукиВеди, 2015. С. 14-16.

67. Naehrig, Michael, Kristin Lauter, and Vinod Vaikuntanathan. "Can homomorphic encryption be practical?," In Proceedings of the 3rd ACM workshop on Cloud computing security workshop, pp. 113-124. 2011.

68. Paillier P. Public-Key Cryptosystems Based on Composite Degree Residuosity Classes (англ.) // Advances in cryptology, EUROCRYPT'99. — 1999. — P. 223-238. — ISBN 978-3-540-48910-8. — doi:10.1007/3-540-48910-X_16.

69. Rivest, R.L. On data banks and privacy homomorphism / R.L. Rivest, L. Adleman, M.L. Dertouzos // Foundations of secure computation. -1978. –Vol. 32, no. 4. –Pp. 169-178. http://people.csail.mit.edu/rivest/pubs/RAD78.pdf.

70. Абрамов Андрей. Гомоморфное шифрование. — 2015.

71. А.О. Жиров, О.В. Жирова, С.Ф. Кренделев "Безопасные облачные вычисления с помощью гомоморфной криптографии"

72. Кренделев С.Ф. Гомоморфное шифрование (защищенные облачные вычисления) // РусКрипто’2011. — 2011.

73. Temirbekova Zh.E. “Possibility using homomorphic encryption for data security” //International Scientific Conference of Students and Young Scientists «Farabi alemi» Almaty, Kazakhstan, April 10-12, - 2018, -P. 193-194

74. Alex Biryukov and Dmitry Khovratovich, “Related-key Cryptanalysis of the Full AES-192 and AES-256” "Archived copy". Archived from the original on 2009-09-28. Retrieved 2020-02-16.

75. M. Farooq, M. Waseem, A. Khairi and S. Mazhar, "A Critical Analysis on the Security Concerns of Internet of Things (loT)", *Perception*, vol. 111, 2020.

76. N.P. Smart and F. Vercauteren, “Fully homomorphic encryption with relatively small key and ciphertext sizes, public Key Cryptography,” PKC Springer Berlin Heidelberg, vol. 6056, pp. 420-443, 2010

77. Y. NarasimhaRao, P. Surya Chandra, V. Revathi, N. Suresh Kumar. “Providing enhanced security in IoT based smart weather system,” Indonesian

Journal of Electrical Engineering and Computer Science, vol. 18, No. 1, April 2020, pp. 9~15

78. Gentry, Craig. "Fully homomorphic encryption using ideal lattices," In Proceedings of the forty-first annual ACM symposium on Theory of computing, pp. 169-178. 2009.

79. Gentry, Craig, and Dan Boneh. “A fully homomorphic encryption scheme,” PhD thesis, Stanford: Stanford university, 2009.

80. Van Dijk, Marten, Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. "Fully homomorphic encryption over the integers." In Annual International Conference on the Theory and Applications of Cryptographic Techniques, vol. 12, pp. 24-43. 2010.

81. Naehrig, Michael, Kristin Lauter, and Vinod Vaikuntanathan. "Can homomorphic encryption be practical?," In Proceedings of the 3rd ACM workshop on Cloud computing security workshop, pp. 113-124. 2011.

82. Левитин А. В. Глава 6. Метод преобразования: Схема Горнера и возведение в степень // Алгоритмы. Введение в разработку и анализ — М.: Вильямс, 2006. — С. 284—291. — 576 с. — ISBN 978-5-8459-0987-9

83. Pandey J.G., Goel T., Karmakar A. “A High-Performance and Area-Efficient VLSI Architecture for the PRESENT Lightweight Cipher,” Proceedings of the 17th International Conference on Embedded Systems; Maharashtra, India. pp. 392–397, 6–10 January 2018.

84. А. О. Жиров, О. В. Жирова, С. Ф. Кренделев. Безопасные облачные вычисления с помощью гомоморфной криптографии. Журнал БИТ (безопасность информационных технологий), том 1, 2013. стр. 6–12.

85. A. Zhirov, O. Zhirova, S. F. Krendelev. Practical fully homomorphic encryption over polynomial quotient rings. Internet Security (WorldCIS), 2013 World Congress on. – IEEE, 2013. pp. 70-75. doi: 10.1109/WorldCIS.2013.6751020

86. Pyrkova A.Yu., Temirbekova Zh.E. “Performing symmetric encryption mbed platform” // Вестник КазНИТУ, - 2018. №2, – С. 473 – 476

87. Katz J., Lindell Y. Introduction to Modern Cryptography: Principles and Protocols. — Chapman & Hall/CRC, 2007. — С. 385-395. — ISBN 978-1584885511.

88. Temirbekova Zh.E “For data security symmetric encryption algorithm” // Международная научно-практическая конференция «Актуальные и перспективные направления развития научно-технологического прогресса», 30 января, 2020 года, Россия, г. Кемерово, С. 26-30

89. Temirbekova Zh.E., Pyrkova A.Yu. “Improving teachers’ skills to integrate the microcontroller technology in computer engineering education”, Education and information technology, -2022 doi:10.1007/s10639-021-10875-8 (Scopus базасы бойынша процентилі - 95). Q1

90. Темирбекова Ж.Е., Пыркова А.Ю. Авторлық куәлік, объектінің атауы: HomomorphicController version 1.0, №16878, 2021

91. Temirbekova Zh.E., Pyrkova A.Yu. “Using FHE in a binary ring Encryption and Decryption with BLE Nano kit microcontroller” //E3S Web of Conferences 202 (ICENIS 2020), -2020. 15002 (Scopus).

92. Temirbekova Zh.E., B.K. Alymbayeva. “Using Atmel AVR microcontrollers for safety-performance computing” // Вестник КазНИТУ, -2017. №2, – С. 192 – 195

93. Pyrkova A.Yu., Temirbekova Zh.E. “Compare encryption performance across devices to ensure the security of the IoT”, Indonesian Journal of Electrical Engineering and Computer Science, -2020. -Vol. 20. -No. 2. – P. 894-902. (Scopus базасы бойынша процентилі - 45). Q3

94. Темирбекова Ж.Е. “Толық гомоморфты шифрлеу алгоритмінің кітапханасы”«Физика – математика ғылымдарының қазіргі білім беру кеңістігіндегі рөлі» VI халықаралық ғылыми-практикалық конференция материалдар жинағы, 7 желтоқсан 2021 ж., Атырау, Казахстан, 326-331 б.

95. Gentry Craig, A fully homomorphic encryption scheme // A dissertation submitted to the department of computer science and the committee on graduate students of Standford University. – 2009.

96. Regev O. New lattice-based cryptographic constructions // J. ACM. – 2004. – Vol. 51, No. 6. – P. 899-942.

97. Regev O. On lattices, learning with errors, random linear codes, and cryptography // STOC. – 2005. – P. 84-93.

98. Temirbekova Zh.E., Pyrkova A.Yu., Abdiakhmetova Zh. “Library of fully homomorphic encryption on a microcontroller” //2022 International Conference on Smart Information Systems and Technologies 28-30 April, 2022, Nur-Sultan, doi:10.1109/SIST54437.2022.9945722 (Scopus).

99. Pyrkova A.Yu, Temirbekova Zh.E. “Using microcontrollers to ensure data security”, International Research Conference on Technology, Science, Engineering and Economy held Seattle, USA, February 28th, 2020, P. 52-60

100. Pyrkova A.Yu., Temirbekova Zh.E. “Possibilities of using a BLE Nano Kit microcontroller to develop cryptographic libraries” // Вестник КазНИТУ:, - 2018. №2, – С. 477 – 481

101. Варновский Н.П., Шокуров А.В. Гомоморфное шифрование // Тр. Института системного программирования РАН. – 2007. – № 12. – С. 27-36.

102. Варновский Н.П., Мартишин С.А., Храпченко М.В., Шокуров А.В. Пороговые системы гомоморфного шифрования и защита информации в облачных вычислениях // Програм-мирование. – 2015. – № 4. – С. 47-51.

103. Pyrkova A.Yu, Temirbekova Zh.E “Use homomorphic encryption for data security” // «Көліктегі инновациялық технологиялар: білі, ғылым, тәжірибе» атты XLII Халықаралық ғылыми-практикалық конференцияның материалдары, 2018 ж., Алматы, Казахстан, (I том), 83 -85 б.

104. Temirbekova Zh.E “For data security symmetric encryption algorithm” // Международная научно-практическая конференция «Актуальные и перспективные направления развития научно-технологического прогресса», 30 января, 2020 года, Россия, г. Кемерово, С. 26-30

ҚОСЫМША А – Жарияланымдар тізімі

1. Temirbekova Zh.E., Pyrkova A.Yu. “Improving teachers’ skills to integrate the microcontroller technology in computer engineering education”, Education and information technology, -2022 doi:10.1007/s10639-021-10875-8 **(Scopus базасы бойынша процентилі - 95). Q1**

2. Pyrkova A.Yu., Temirbekova Zh.E. “Compare encryption performance across devices to ensure the security of the IOT”, Indonesian Journal of Electrical Engineering and Computer Science, -2020. -Vol. 20. -No. 2. – P. 894-902. **(Scopus базасы бойынша процентилі - 45). Q3**

3. Pyrkova A.Yu., Temirbekova Zh.E. “Using FHE in a binary ring Encryption and Decryption with BLE Nano kit microcontroller” //E3S Web of Conferences **202 (ICENIS 2020)**, -2020. 15002 (**Scopus**).

4. Temirbekova Zh.E., Pyrkova A.Yu., Abdiakhmetova Zh., Berdaly A. “Library of fully homomorphic encryption on a microcontroller” //2022 International Conference on Smart Information Systems and Technologies 28-30 April, 2022, Nur-Sultan (**Scopus**).

5. Temirbekova Zh.E., B.K. Alymbayeva. “Using Atmel AVR microcontrollers for safety-performance computing” // Вестник КазНИТУ, -2017. №2, – С. 192 - 195

6. Pyrkova A.Yu., Temirbekova Zh.E. “Possibilities of using a BLE Nano Kit microcontroller to develop cryptographic libraries” // Вестник КазНИТУ:, - 2018. №2, – С. 477 - 481

7. Pyrkova A.Yu., Temirbekova Zh.E. “Performing symmetric encryption mbed platform” // Вестник КазНИТУ, - 2018. №2, – С. 473 – 476

8. Temirbekova Zh.E “Programming microcontroller AVR Atmega8” // «Көліктегі инновациялық технологиялар: білім, ғылым, тәжірибе» атты XLI Халықаралық ғылыми-практикалық конференцияның материалдары, 3-4 сәуір 2017 ж., Алматы, Казахстан, (I том), 102-104 б.

9. Pyrkova A.Yu, Temirbekova Zh.E “Use homomorphic encryption for data security” // «Көліктегі инновациялық технологиялар: білі, ғылым, тәжірибе» атты XLII Халықаралық ғылыми-практикалық конференцияның материалдары, 2018 ж., Алматы, Казахстан, (I том), 83 -85 б.

10. Temirbekova Zh.E “For data security symmetric encryption algorithm” // Международная научно-практическая конференция «Актуальные и перспективные направления развития научно-технологического прогресса», 30 января, 2020 года, Россия, г. Кемерово, С. 26-30

11. Pyrkova A.Yu, Temirbekova Zh.E., “Using microcontrollers to ensure data security”, International Research Conference on Technology, Science, Engineering and Economy held Seattle, USA, February 28th, 2020, P. 52-60

12. Темирбекова Ж.Е. “Толық гомоморфты шифрлеу алгоритмінің кітапханасы”«Физика – математика ғылымдарының қазіргі білім беру кеңістігіндегі рөлі» VI халықаралық ғылыми-практикалық конференция материалдар жинағы, 7 желтоқсан 2021 ж., Атырау, Казахстан, 326-331 б.

13. Temirbekova Zh.E. “Possibility using homomorphic encryption for data security” //International Scientific Conference of Students and Young Scientists «Farabi alemi» Almaty, Kazakhstan, April 10-12, - 2018, -P. 193-194

ҚОСЫМША Ә – Авторлық куәлік

ҚР авторлық құқықпен қорғалатын объектілерге құқықтардың мемлекеттік тізіліміне мәліметтер мен олардың өзгерістерін енгізу, 2021 жылы 21 сәуірде №16878, «HomomorphicController_Version01».

ҚАЗАҚСТАН РЕСПУБЛИКАСЫ

РЕСПУБЛИКА КАЗАХСТАН

АВТОРЛЫҚ ҚҰҚЫҚПЕН ҚОРҒАЛАТЫН ОБЪЕКТІЛЕРГЕ ҚҰҚЫҚТАРДЫҢ МЕМЛЕКЕТТІК ТІЗІЛІМГЕ МӘЛІМЕТТЕРДІ ЕНГІЗУ ТУРАЛЫ

КУӘЛІК

2021 жылғы «22» сәуір № 16878

Автордың (лардың) жөні, аты, әкесінің аты (егер ол жеке басын куәландыратын құжатта көрсетілсе):
ТЕМИРБЕКОВА ЖАНЕРКЕ ЕРЛАНОВНА, Пыркова Анна Юрьевна

Авторлық құқық объектісі: **ЭЕМ-ге арналған бағдарлама**

Объектінің атауы: **HomomorphicController version 1.0**

Объектіні жасаған күні: **21.04.2021**

ҚР ӘМ «ҰЛТТЫҚ ЗИЯТКЕРЛІК МЕНШІК ИНСТИТУТЫ» РМК
РГП «НАЦИОНАЛЬНЫЙ ИНСТИТУТ ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ» МЮ РК

Құжат түпнұсқалығын http://www.kazpatent.kz/ru сайтының "Авторлық құқық" бөлімінде тексеруге болады https://copyright.kazpatent.kz

Подлинность документа возможно проверить на сайте kazpatent.kz в разделе «Авторское право» https://copyright.kazpatent.kz

ЭЦҚ қол қойылды

Оспанов Е.К.

Сурет Ә.1 – Авторлық куәлік