

Восточно-Казахстанский технический университет
им. Д. Серикбаева

УДК 681.5:004.272.22

На правах рукописи

САГЫНГАНОВА ИНДИРА КЕНЕСОВНА

**Автоматизированная система управления теплопунктами
с применением конвейерной обработки информации**

6D070200 – Автоматизация и управление

Диссертация на соискание степени
доктора философии (PhD)

Научные консультанты:
Квасов А.И.,
доктор технических наук, профессор,
академик КазНАЕН
Маркин В.Б.,
доктор технических наук, профессор
Алтайского государственного
технического университета
им. Ползунова,
Российская Федерация

Республика Казахстан
Усть-Каменогорск, 2020

СОДЕРЖАНИЕ

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ	4
ВВЕДЕНИЕ	5
1 СОВРЕМЕННОЕ СОСТОЯНИЕ ОРГАНИЗАЦИИ РАБОТЫ ТЕПЛОВЫХ ПУНКТОВ И ВОЗМОЖНОСТИ КОНТРОЛЯ И РЕГУЛИРОВАНИЯ ИХ РАБОТЫ	9
1.1 Основные подходы к проектированию монтажу и обслуживанию тепловых пунктов	9
1.2 Существующие системы автоматизации работы тепловых пунктов.....	11
1.3 Организация обработки информации при автоматизации теплового пункта.....	16
1.4 Факторы, влияющие на внедрение систем автоматизации тепловых пунктов.....	21
1.5 Формулировка цели и постановка задач исследования.....	22
Выводы по разделу.....	23
2 КОНВЕЙЕРНАЯ ОБРАБОТКА ДАННЫХ	25
2.1 Комбинированные методы получения и обработки информационных потоков в централизованных системах теплоснабжения.....	25
2.2 Организация конвейерного выполнения задач в распределенных АСУ тепловым пунктом.....	29
2.3 Использование систолических массивов для создания систем автоматизации с высокой скоростью передачи данных от тепловых пунктов.....	32
2.4 Методика обработки данных тепловых пунктов на основе технологии работы центрального процессора с процессами и потоками.....	43
Выводы по разделу.....	47
3 КОНВЕЙЕРНАЯ СИСТЕМА РЕГУЛИРОВАНИЯ РАБОТЫ ТЕПЛОПУНКТА	48
3.1 Автоматизация и модернизация тепловых пунктов.....	48
3.2 Автоматизация системы конвейерного управления тепловыми пунктами.....	50
Выводы по разделу.....	56
4 ИССЛЕДОВАНИЕ КОНВЕЙЕРНОГО РЕЖИМА РАБОТЫ ТЕПЛОПУНКТОВ	57
4.1 Разработка структуры программно-технического комплекса и используемое программное обеспечение в системах теплоснабжения.....	57
4.2 Программа для создания и редактирования схем теплоснабжения тепловых пунктов с возможностью считывания	

реальных данных.....	62
4.3 Экспериментальные исследования эффективности применения автоматизации тепловых пунктов с использованием конвейерной обработки информации.....	70
Выводы по разделу.....	79
ЗАКЛЮЧЕНИЕ	80
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	81
ПРИЛОЖЕНИЕ А – Свидетельство об авторском праве. Код программы «Конструктор».....	88
ПРИЛОЖЕНИЕ Б – Акт внедрения.....	100
ПРИЛОЖЕНИЕ В – Диплом конкурса «Жас Ғалым - 2019».....	101

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

В настоящей диссертации использованы следующие обозначения и сокращения:

RTC – часы реального времени в тепловычислителе

SCADA – Supervisory Control And Data Acquisition (диспетчерское управление и сбор данных)

SCADA-система – программное обеспечение для сбора информации в реальном времени с удаленных точек (объектов) для обработки, анализа и возможного управления удаленными объектами

АРМ – автоматизированное рабочее место

АТП – автоматизированный тепловой пункт

БТП – блочный (или модульный) тепловой пункт

ГВС – система горячего водоснабжения

И – символьный индикатор тепловычислителя

ИП – измерительный преобразователь

ИТП – индивидуальный тепловой пункт

МП – микросхема генератора сигнала сброса и компаратора пропадания питания

ОЗУ – оперативное запоминающее устройство

ОМК – основной микроконтроллер тепловычислителя с встроенной памятью программ

ПК – персональный компьютер

ПЛМ – программируемая логическая матрица

РР – регулятором расхода

СДУ – система диспетчерского управления и сбора данных

СМК – микроконтроллер связи и шифрования данных тепловычислителя

СОД – система обработки данных

ПТК – программно-технический комплекс

ТП – тепловой пункт

ТЭЦ – разновидность тепловой электростанции, которая не только производит электроэнергию, но и является источником тепловой энергии в централизованных системах теплоснабжения

ЦТП – центральный тепловой пункт

В работе использованы традиционные для электротехнической литературы единицы измерения: А, В, Ом, Гн, Вт, кВА, кВт, Гц, м, мм, мм², об/мин, м/с, кПа, МПа, и т.д.

ВВЕДЕНИЕ

Актуальность темы диссертационного исследования. Низкая средняя температура воздуха и резко-континентальный холодный климат, длительная продолжительность отопительного сезона по сравнению со странами Европы, а также централизованное теплоснабжение, характеризующееся протяженными тепловыми сетями, в сочетании с низкой энергоэффективностью производства и передачей тепловой энергии являются характерной особенностью теплоэнергетических систем Казахстана.

Потребление тепловой энергии по данным Агентства РК по статистике и планам Правительства с 2010 года по 2020 год вырастет на 15 тыс.Гкал и достигнет 110 тыс.Гкал [1].

Несмотря на наличие объективных причин высокой энергоемкости ВВП Республики Казахстан, существует значительный потенциал повышения энергоэффективности и энергосбережения, так как энергоемкость ВВП Республики Казахстан в два раза превышает аналогичный показатель большинства развитых стран [2].

Низкой энергоэффективностью характеризуются производство и передача тепловой энергии. По экспертным данным перерасход топлива на производство электроэнергии составляет 10-15%, на теплоснабжение – 15-20%. Затраты на внедрение энергосбережения примерно в 5 раз ниже, чем на создание новых источников энергии [1]. Сложной задачей является определение тепловых потерь, персонифицировать которые из-за разветвленности тепловых распределительных сетей, количества котельных и насосно-перекачивающих и дроссельных станций, нерациональной конфигурации внутридомовых сетей теплоснабжения непросто.

В связи с этим, необходимость создания и внедрения современных информационно-управляющих систем для теплоснабжающих комплексов городов Казахстана не вызывает сомнения.

В последние годы все больше используются средства для автоматического регулирования систем отопления и горячего водоснабжения (ГВС) как в центральных (ЦТП), так и в индивидуальных (ИТП) тепловых пунктах зданий и сооружений – это, например, комплексы автоматических регуляторов в котельных и тепловых пунктах зданий. Экономия энергоресурсов в таких системах достигает до 10% годового их расхода.

Основная идея работы – разработка и внедрение микроконтроллерной SMART-системы для реализации рациональных режимов теплообеспечения, использующих в процессе получения и обработки информации о параметрах теплопотребления и принятия управляющих решений переменные циклы опроса параметров теплоснабжения на тепловых пунктах, с учетом их динамических характеристик и с последующей конвейерной обработкой информации на верхнем уровне автоматизированных систем централизованного теплоснабжения.

Цель диссертационной работы. Автоматизация тепловых пунктов на основе современной микроконтроллерной системы, использующей в процессе получения информации о параметрах теплопотребления и принятия управляющих решений способы конвейерной обработки данных, и обеспечивающая повышение эффективности теплообеспечения.

Объект исследования. Теплоэнергетика, тепловые сети, тепловые пункты.

Предмет исследования. Автоматизированная система управления тепловыми пунктами на основе конвейерной обработки данных.

Методы исследования. Для решения поставленных задач применялись основные положения и методы теории автоматического управления, математического моделирования.

Задачи исследований:

1) на основе анализа современного состояния систем автоматизации и методов обработки информации параметров теплоснабжения тепловых пунктов разработать комбинированные методы получения и конвейерной обработки информационных потоков в централизованных системах теплоснабжения в распределенных АСУ тепловыми пунктами;

2) разработать структуру программно-технического комплекса с анализом используемого программного обеспечения в системах теплоснабжения;

3) разработать программы для создания и редактирования схем теплоснабжения тепловых пунктов с возможностью считывания реальных данных;

4) провести экспериментальные исследования эффективности применения автоматизации тепловых пунктов с использованием конвейерной обработки данных.

Научная новизна заключается в следующем:

1) предложены принципы и методы измерения и обработки информации в централизованных системах теплоснабжения, отличающиеся переменными циклами опроса параметров процессов на тепловых пунктах и учитывающие их динамические характеристики, с последующей конвейерной обработкой информации на верхнем уровне автоматизированных систем централизованного теплообеспечения;

2) предложены принципы разработки распределенной системы, сочетающей в себе на разных иерархических уровнях программно-аппаратные средства, обрабатывающие и минимизирующие потоки информации в системах централизованного теплоснабжения.

Практическая ценность работы заключается:

1) В разработке новой, внесенной в государственный реестр прав на объекты, охраняемые авторским правом № 4095 от «18» июня 2019 года, программы для ЭВМ «Конструктор», предназначенной для создания и редактирования схем теплоснабжения тепловых пунктов с возможностью считывания реальных параметров тепловых процессов и адаптированной к разработанному и внедренному в Казахстане программному обеспечению для

систем теплообеспечения «ТГИД».

2) Во внедрении в эксплуатацию в жилых и производственных объектах жилищно-коммунального хозяйства АТП (автоматизированных тепловых пунктов).

Основные научные положения и результаты исследований, выносимые на защиту. Принципы и методы измерения и обработки информации в централизованных системах теплоснабжения, сочетающие переменные циклы опроса параметров теплоснабжения на тепловых пунктах с учетом их динамических характеристик, и конвейерную обработку информации на верхнем уровне автоматизированных систем централизованного теплоснабжения.

Обоснованность и достоверность научных положений, выводов и рекомендаций основывается на корректном использовании теории оптимального управления, теории автоматического управления, методов математического моделирования, а также на внедрении разработанных рекомендаций.

Апробация работы. Основные материалы и результаты диссертационной работы докладывали и обсуждались: на научных семинарах кафедры «Приборостроение и автоматизация технологических процессов» ВКГТУ им. Д.Серикбаева, на Международной научно-технической конференции студентов, магистрантов и молодых ученых «Творчество молодых инновационному развитию Казахстана», на XVIII Международной конференции молодых специалистов по микро и нанотехнологиям и электронным приборам (EDM 2017) (Республика Алтай, Россия), на Международной научно-практической конференции «Вычислительные и информационные технологии в науке, технике и образовании» (CITech–2018), на III International Scientific-Practical Conference (Seoul, Korea, March 28-30, 2018), на Республиканском конкурсе инновационных проектов молодых ученых «Жас Ғалым - 2019», где представленный проект по теме диссертационных исследований занял первое место, на научных семинарах в Институте информационных и вычислительных технологий МОН РК и кафедры Искусственный интеллект и Big data факультета Информационных технологий КазНУ им. Аль-Фараби.

Личный вклад автора. Постановка проблемы, формулирование всех рассмотренных задач, поиск путей их решения, а также приведенные в диссертации научные и практические результаты, их анализ, формирование итоговых выводов и внедрение осуществлены лично автором диссертации.

Публикации. Основные научные результаты докторской диссертации опубликованы в 19 научных трудах, в том числе 4 публикации в изданиях, рекомендованных Комитетом по контролю в сфере образования и науки МОН РК, 3 публикации, входящих в информационную базу компаний Scopus и Web of Science, из них 2 статьи в периодических изданиях с процентилем соответственно 75 и 26%, 10 публикаций в международных научно-практических конференциях, в том числе 3 в зарубежье. Получено 1

свидетельство об интеллектуальной собственности (СИС) РК.

Структура и объем работы. Диссертация состоит из содержания, списка сокращений, введения, основной части из четырех глав, заключения. Объем диссертации составляет 101 страниц машинописного текста, содержит 47 рисунков, 8 формул и 7 таблиц, списка использованных источников, включающий 90 наименований, 3 приложения.

1 СОВРЕМЕННОЕ СОСТОЯНИЕ ОРГАНИЗАЦИИ РАБОТЫ ТЕПЛОВЫХ ПУНКТОВ И ВОЗМОЖНОСТИ КОНТРОЛЯ И РЕГУЛИРОВАНИЯ ИХ РАБОТЫ

1.1 Основные подходы к проектированию монтажу и обслуживанию тепловых пунктов

Эксплуатационное регулирование системы отопления проводят с целью обеспечения теплоподачи в отапливаемые помещения соответствующей текущей теплопотребности. В зависимости от места проведения регулирования в системе теплоснабжения различают центральное, групповое, местное и индивидуальное регулирование [3-4].

Центральное регулирование на тепловой станции при теплоснабжении различных по назначению зданий (жилые, общественные, производственные и др.) и режиму теплопотребления их инженерных систем (отопление, горячее водоснабжение, вентиляция и др.) не может обеспечить устойчивой работы систем отопления.

Устойчивость работы повышается при приближении места проведения регулирования к теплопотребителю за счет более полного учета различных факторов, определяющих теплопотребность помещений отапливаемых зданий.

Тепловой пункт (ТП) – совокупность оборудования, находящаяся как в отдельном, так и в одном помещении с потребителями, включающая элементы тепловых энергоустановок, делающая возможным присоединение этих установок к теплосети, работоспособность установок, контроль за режимами потребления тепла, преобразование, управление параметрами теплоносителя и распределение теплоносителя по типам потребления.

Проектирование ИТП (ЦТП) сопровождается дальнейшим согласованием объектов в органах государственного надзора и сдачей теплоснабжающим организациям. Проектирование тепловых пунктов (ИТП, ЦТП, БТП) производится компанией на основании установленных нормативно-правовых документов паспортов систем теплового снабжения объектов и на основе анализа возможности внедрения в данном случае того или иного проектного решения, учитывая специфические особенности объекта (объектов) потребления.

Через тепловой пункт регулируется работоспособность теплового оборудования, контролируется режим потребления тепла, распределяется теплоноситель по типам потребления (системы отопления, горячего водоснабжения, вентиляции и кондиционирования), настраиваются и изменяются параметры теплоносителя [5].

Тепловые пункты обязательно имеются как в жилых, так и в офисных, производственных и складских помещениях. Обслуживание тепловых пунктов зависит от их типов [5].

Центральный тепловой пункт (ЦТП) – это такой тепловой пункт, который предназначен для обслуживания нескольких зданий. Это могут быть как

промышленные объекты, так и непроизводственные комплексы. Индивидуальный тепловой пункт (ИТП) предназначен для обслуживания одного здания или его части. С помощью тепловых сетей тепловые пункты соединяются с потребителем тепла и теплогенерирующим предприятием: теплоэлектроцентралью или котельной. Теплосеть, соединяющая ЦТП или ИТП с источником тепла называется первичной, а та, что соединяет тепловой пункт с потребителем – вторичной. Обслуживание вторичных тепловых сетей – неотъемлемая часть эксплуатации теплового пункта здания или комплекса сооружений. Индивидуальные тепловые пункты зачастую не имеют вторичной тепловой сети, в отличие от ЦТП [5].

Качественное и своевременное обслуживание ИТП и ЦТП обеспечивает бесперебойное выполнение тепловым пунктом своих функций. Функции теплового пункта [5]:

- контроль и оптимизация параметров теплоносителя;
- преобразование вида теплоносителя;
- защита систем теплоснабжения, снижение риска создания аварийной ситуации;
- распределение теплоносителя по системам отопления, водоснабжения и вентиляции;
- включение и выключение систем теплоснабжения;
- контроль над расходом теплоносителя и тепла, а также обеспечение необходимого расхода теплоносителя. на эту характеристику влияют тепловые потери и особенности объекта в сочетании с заданными параметрами;
- сокращение тепловых потерь.

Исправное функционирование ИТП или ЦТП, обусловленное грамотной эксплуатацией и регулярным обслуживанием, ведет к повышению уровня комфорта в помещениях. А это, в свою очередь, является залогом уютной атмосферы в жилых помещениях, создания необходимых условий труда, роста производительности. Правильное обслуживание тепловых пунктов снижает риск создания аварийной ситуации. Следствием этого является снижение расходов на эксплуатацию ИТП или ЦТП, отсутствие необходимости в срочном и дорогостоящем ремонте оборудования или его преждевременной замене, а также бесперебойная работа организации, занимающей объект недвижимости. Более того, благодаря регулярному обслуживанию теплового пункта, за год расходы электроэнергии сокращаются на 30-35%. Наконец, систематическое обслуживание тепловых пунктов позволяет продлить срок службы оборудования ИТП или ЦТП и увеличить межремонтный период [6].

ЦТП могут располагаться как в отдельном специальном небольшом сооружении, так и в подвальном помещении одного из объектов. Но, в любом случае, для бесперебойной службы ЦТП необходима правильная эксплуатация и качественное обслуживание [5].

Начинается оно, конечно же, с проектировки и установки оборудования. Рекомендуется для монтажа и обслуживания пользоваться услугами одной

эксплуатирующей компании: гораздо удобнее, когда устанавливают оборудование, а затем обслуживают его одни и те же специалисты. За услугами по обслуживанию тепловых пунктов могут обращаться как юридические, так и физические лица [5].

Эффективное обслуживание ЦТП подразумевает, прежде всего, регулярность. Одним из главных мероприятий, связанных с обслуживанием центральных тепловых пунктов, является подготовка ЦТП к отопительному сезону. Подготовка теплового пункта к зимнему периоду – это большой комплекс работ [7].

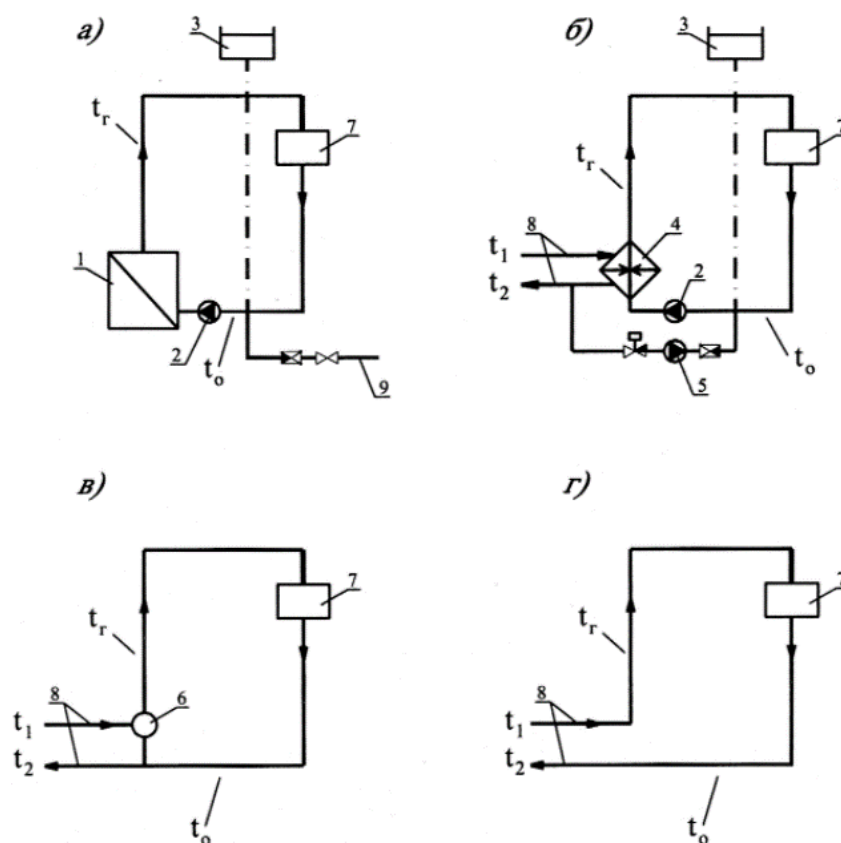
1.2 Существующие системы автоматизации работы тепловых пунктов

При централизованном теплоснабжении ТП может быть как местным - индивидуальным для теплопотребляющих систем конкретного здания, так и групповым - центральным для систем группы зданий. ИТП размещается в специальном помещении здания, ЦТП чаще всего представляет собой отдельностоящее одноэтажное строение [8]. Проектирование тепловых пунктов ведётся в соответствии с нормативными правилами [4, с. 4].

Роль теплогенератора при независимой схеме присоединения теплопотребляющих систем к наружной тепловой сети (рисунок 1.1б) выполняет водоводяной теплообменник [8, с. 7].

Для нагрева воды в системах горячего водоснабжения, особенно в индивидуальном жилом доме, больше подходит не скоростной, а емкостной водонагреватель (рисунок 1.1в). Его объём определяется исходя из расчётного количества одновременно работающих точек водоразбора и предполагаемых индивидуальных особенностей водопотребления в доме [8, с. 7].

Общим для всех схем, изображенных на рисунке 1.1, является применение насоса для искусственного побуждения движения воды в теплопотребляющих системах. В первых двух схемах (рисунок 1.1а и 1.1б) насос включают непосредственно в системы здания. В зависимых схемах (рисунки 1.1в и 1.1г) насос помещают на тепловой станции, и он создаёт давление, необходимое для циркуляции воды, как в наружных теплопроводах, так и в местных теплопотребляющих системах [8, с. 7]. Насос, действующий в замкнутых кольцах систем, заполненных водой, не поднимает, а только перемещает воду, создавая циркуляцию, и поэтому называется циркуляционным. В отличие от циркуляционного насоса насос в системе водоснабжения перемещает воду, поднимая её к точкам разбора. При таком использовании насос называют повысительным [8, с. 7]. В процессах заполнения и возмещения потери (утечки) воды в системе отопления циркуляционный насос не участвует. Заполнение происходит под воздействием давления в наружных теплопроводах, в водопроводе или, если этого давления недостаточно, с помощью специального подпиточного насоса [8, с. 7].



а – местное теплоснабжение от собственной водогрейной котельной; б – независимая схема присоединения при централизованном теплоснабжении; в – зависимая схема со смешением воды при централизованном теплоснабжении; г – зависимая прямоточная схема при централизованном теплоснабжении; 1 – водогрейный котёл (теплогенератор); 2 – циркуляционный насос; 3 – расширительный бак; 4 – теплообменник; 5 – подпиточный насос; 6 – смесительная установка; 7 – теплопотребляющие инженерные системы здания; 8 – наружная тепловая сеть; 9 – наружный водопровод

Рисунок 1.1 – Принципиальные схемы водяного теплоснабжения зданий

Существующие схемы автоматизации центральных (ЦТП) и индивидуальных (ИТП) тепловых пунктов.

В соответствии с правилами по проектированию тепловых пунктов автоматизация центральных и индивидуальных тепловых пунктов должна обеспечивать:

- регулирование подачи теплоты (теплового потока) в системы отопления в зависимости от изменения параметров наружного воздуха;
- ограничение максимального расхода воды из тепловой сети на ТП путем

перекрытия клапана регулятора теплоты на отопление;

- поддержание требуемого перепада давлений в подающем и обратном трубопроводах тепловых сетей на вводе в ИТП;

- поддержание заданной температуры воды, поступающей в систему горячего водоснабжения здания.

МЗТА разработал и с 1998 г. выпускает устройства управляющие РУНТ, предназначенные для автоматизации ЦТП, ИТП и других объектов теплоснабжения. РУНТ осуществляют регулирование температур в системах отопления и горячего водоснабжения, управление основным и резервным насосами отопления, горячего водоснабжения, подпитки [9].

Функции устройства РУНТ определяются установленными в них исполнениями регуляторов и наличием или отсутствием встроенного модуля для управления насосами подпитки, а именно:

- регулирование температуры прямой воды на отопление в зависимости от температуры наружного воздуха по отопительному графику;

- ограничение температуры обратной воды (на превышение) по заданному графику в зависимости от температуры наружного воздуха;

- регулирование температуры в системе горячего водоснабжения;

- автоматическое включение насосов в системах отопления и ГВС, переключение на резервный насос в случае выхода из строя основного с необходимыми задержками и блокировками;

- ручное управление каждым насосом в отдельности;

- возможность автоматического выключения отопления в зависимости от температуры наружного воздуха;

- сигнализация о нормальной работе системы (датчики, насосы и т.д.);

- диагностика отказов устройства;

- автоматическое включение насоса подпитки (РУНТ 312);

- переключение на резервный насос подпитки в случае выхода из строя основного насоса (РУНТ 312);

- экономичный режим регулирования ночью и в выходные дни (РУНТ 411);

- автоматическое включение насоса подпитки (РУНТ 411, 412);

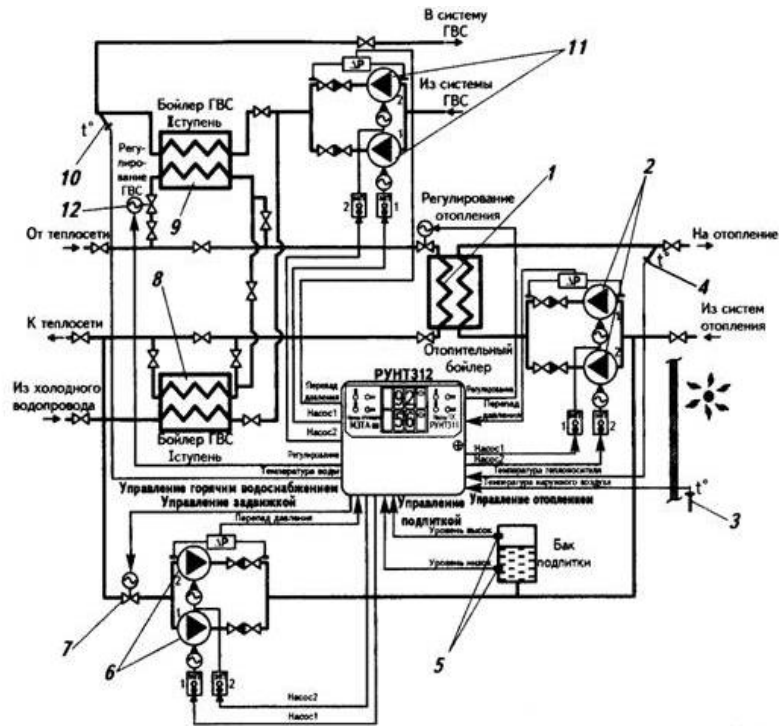
- переключение на резервный насос подпитки в случае выхода из строя основного насоса (РУНТ 411, 412).

На рисунке 1.2 представлена принципиальная схема применения устройства РУНТ-312. Устройство служит для автоматического управления теплоснабжением при независимой схеме присоединения системы отопления и двухступенчатой схеме горячего водоснабжения.

Управление расходом теплоты системой отопления здания производится по сигналу датчика 3 температуры наружного воздуха t_n и датчика 4 контроля начальной температуры воды t_{wT1} поступающей в систему отопления.

В ЦТП здания устанавливаются два водо-водяных теплообменника 8 и 9

для двухступенчатого нагрева водопроводной воды на нужды горячего водоснабжения здания. В первой ступени 8 водопроводная вода нагревается теплотой обратной воды сети теплоснабжения. Если датчик 10 фиксирует, что на горячее водоснабжение поступает вода с температурой ниже $60\text{ }^{\circ}\text{C}$, то последует через регулятор РУНТ 312 команда на открытие автоматического вентиля 12 для поступления горячей воды из подающего трубопровода сети теплоснабжения в теплообменник 9 - второй ступени нагрева воды горячего водоснабжения до $t_{\text{wтв}} = 60\text{ }^{\circ}\text{C}$.



1 - водо-водяной теплообменник для нагрева циркуляционной воды в системе отопления здания; 2 - два циркуляционных насоса (один запасной) с электронным управлением частоты вращения для системы отопления; 3 - датчик контроля температуры наружного воздуха; 4 - датчик контроля начальной температуры воды в системе отопления; 5 - датчик контроля уровня воды в системе отопления; 6 - два подпиточных насоса (один резервный) подачи воды из системы теплоснабжения для подпитки контура циркуляции системы отопления; 7 - автоматический запорный клапан на трубопроводе забора подпиточной воды из обратного трубопровода системы централизованного теплоснабжения; 8 - водо-водяной теплообменник первой ступени нагрева водопроводной воды на горячее водоснабжение здания; 9 - водо-водяной теплообменник второй ступени нагрева водопроводной воды на горячее водоснабжение здания; 10 - датчик контроля температуры воды горячего водоснабжения на уровне $60\text{ }^{\circ}\text{C}$; 11 - насосы (один резервный) циркуляции воды в системе горячего водоснабжения здания; 12 - автоматический вентиль на трубопроводе подачи горячей воды из подающего трубопровода сети теплоснабжения

Рисунок 1.2 – Принципиальная схема применения автоматического регулятора РУНТ 312 в ЦТП для теплоснабжения систем отопления и горячего водоснабжения зданий

Современные системы отопления зданий должны работать эффективно, то есть не только обеспечивать комфортный температурный режим во всех помещениях здания, но и рационально распределять тепловую энергию внутри здания с минимальными потерями. Эти требования могут обеспечить только мероприятия по автоматизации системы отопления и тепловых пунктов зданий, включая установку узла учета тепловой энергии. Комплексная автоматизация системы отопления предполагает регулирование параметров теплоносителя в тепловом пункте, индивидуальное управление подачей теплоты от отопительных приборов (радиаторные терморегуляторы - термостаты), автоматическое поддержание гидравлических режимов в трубопроводной сети (стояки, магистрали) системы отопления, экономичный расход тепловой энергии. Применение данных методов регулирования определено СП РК «Отопление, вентиляция и кондиционирование» [4, с. 68] и обязательно при проектировании систем отопления и тепловых пунктов новых зданий.

Все задачи регулирования решаются за счет средств автоматизации теплового пункта, в том числе с помощью оборудования фирмы «Саутер», что дает не только значительную экономию потребления тепловой энергии, особенно в переходный период (весна, осень), но и обеспечивает устойчивую и экономичную работу системы отопления в отопительный период [10].

Как правило, системы отопления существующих зданий присоединены к тепловым сетям по зависимой схеме, реже по независимой, через теплообменники. Для многих существующих жилых и административных зданий, как правило, актуальны две проблемы: величина оплаты тепловой энергии и распределение тепловой энергии внутри здания, которые могут быть решены различными способами, различающимися размером капитальных затрат и достигаемой при этом величины экономии тепловой энергии. Первым и самым мало затратным способом достижения экономии тепловой энергии и относительной стабилизации распределения тепла по зданию является наладка существующих тепловых пунктов здания, которая сводится к анализу фактических параметров теплоносителя в системе отопления, на основании которого выполняются проверочные расчеты сопла элеваторов. Данный способ актуален как при «недотопах» (плохом распределении тепла внутри здания), так и при «перетопах» (избыточном теплоснабжении) и необходим в случае установки узла учета тепловой энергии для обеспечения заданной величины теплоснабжения (на основании тепловой нагрузки здания). Эффект от данного мероприятия при определенных условиях (наладка не проводилась долгое время, хорошие параметры теплоносителя в тепловых сетях) может достигать 100% и более. Наладку существующих тепловых пунктов (элеваторов) рекомендуется выполнять при изменении параметров теплоносителя на вводе в здание (изменения в режимной карте).

Вторым способом, обеспечивающим устойчивое и экономичное теплоснабжение здания, является автоматизация теплового пункта, которую

рекомендуется выполнять одновременно или после установки узла учета тепловой энергии.

Автоматизированный узел управления, состоящий из насоса и комплекта автоматики (регулятора расхода, датчиков, контроллера), поддерживает температуру теплоносителя в системе отопления здания в соответствие с температурным графиком по температуре наружного воздуха, обеспечивает стабильный гидравлический режим в системе и не чувствителен к перепадам давления в тепловых сетях на вводе в здание. Совместно с применением балансировочных вентилей или регуляторов перепада давления на ветках и стояках, автоматизированный узел управления позволяет более рационально распределить тепло внутри здания. Кроме того, установка автоматизированного узла управления позволяет получить значительную экономию тепловой энергии в периоды оттепелей и переходный период (осень, весна). Замена элеватора на автоматизированный узел управления не влечет за собой дополнительных работ по переделке системы отопления, в случае же объединения нескольких тепловых пунктов здания в один узел управления необходима, как правило, перекладка части трубопроводов (магистралей) по подвалу [11].

1.3 Организация обработки информации при автоматизации теплового пункта

Информационные технологии (ИТ) обработки данных предназначены для решения хорошо структурированных задач, по которым имеются необходимые входные данные, известны алгоритмы и другие стандартные процедуры их обработки. Технология обеспечивает выполнение основного объема работ в автоматическом режиме с минимальным участием человека [12].

Технологический процесс обработки данных включает [12, с. 83]:

– подготовительный этап, на котором осуществляется подготовка к решению задачи (создание справочников, введение в память компьютера необходимых постоянных данных и др.);

– начальный этап, связанный с операциями по сбору, регистрации и размещению документов в базовые массивы (возможна обработка документов, заполненных вручную, однако более эффективным является электронное документирование);

– основной, завершающий этап работы, обеспечивающий получение необходимых отчетных форм, когда из компьютерной базы данных извлекаются рабочие массивы, подлежащие группировке по соответствующим ключевым признакам, подсчету по ним итоговых данных с распечаткой в дальнейшем полученных отчетных документов.

Выбор того или иного варианта технологии определяется прежде всего как объемно-временными особенностями решаемых задач, периодичностью, срочностью, требованиями к скорости связи пользователя с ЭВМ, так и режимных возможностей технических средств [12, с. 85].

Конвейерная обработка улучшает использование аппаратных ресурсов для заданного набора процессов, каждый из которых применяет эти ресурсы заранее предусмотренным способом. Хорошим примером конвейерной организации является сборочный транспортер на производстве, на котором изделие последовательно проходит все стадии вплоть до готового продукта.

Преимущество этого способа состоит в том, что каждое изделие на своем пути использует одни и те же ресурсы, и как только некоторый ресурс освобождается данным изделием, он сразу же может быть использован следующим изделием, не ожидая, пока предыдущее изделие достигнет конца сборочной линии. Если транспортер несет аналогичные, но не тождественные изделия, то это последовательный конвейер; если же все изделия одинаковы, то это векторный конвейер.

Последовательные конвейеры. На рисунке 1.3а представлена схема устройства обработки команд, в котором имеются четыре ступени: выборка команды из памяти, декодирование, выборка операнда, исполнение.



а – ступени конвейера; б – временная диаграмма работы

Рисунок 1.3 – Схема четырехступенного устройства обработки команд

Ускорение обработки в данном устройстве измеряется отношением времени T_s , необходимого для последовательного выполнения L заданий (т.е. выполнения L циклов на одной обрабатывающей ступени), ко времени T_r выполнения той же обработки на конвейере. Обозначим через t_i время обработки на i -й ступени, а через t_j – соответствующее время для самой

медленной ступени (рисунок 1.3б). Тогда, если L заданий (команд) проходят через конвейер с n ступенями, эффективность конвейера определяется выражением:

$$\frac{T_s}{T_p} = \frac{L \sum_{i=1}^n t_i}{\sum_{i=1}^n t_i + (L-1)t_j}; \quad (1.1)$$

для

$$t_i = t_j \rightarrow \frac{nL}{n+L-1} \quad (1.2)$$

Конвейерный подход в обработке информации широко применяется во всех современных быстродействующих процессорах, таких как Pentium Intel Core i7, AMD A10. Но длина такого конвейера может составлять 20-30 ступеней, в зависимости от типа выполняемой программы и тактовой частоты процессора [14].

Рассмотрим еще один пример конвейерной обработки данных. Дополним наш конвейер 5 ступеню и введем буквенные обозначения:

H1 – считывание команды в процессор (выборка команды);

H2 – декодирование команды (определение КОП и типа операндов);

H3 – считывание операндов (определение местонахождения операндов и вызов их из регистров);

H4 – выполнение команды;

H5 – запись результата.

Выполнение команд в таком конвейере представлено в таблице 1.1 [15].

Таблица 1.1 – Порядок выполнения команд в 5-ступенчатом конвейере

Номер команды	Такт								
	1	2	3	4	5	6	7	8	9
i	H1	H2	H3	H4	H5				
i+1		H1	H2	H3	H4	H5			
i+2			H1	H2	H3	H4	H5		
i+3				H1	H2	H3	H4	H5	
i+4					H1	H2	H3	H4	H5

Из таблицы видно, что при увеличении команды на n-единиц, ступени смещаются ровно на столько же тактов.

Далее рассмотрим производительность идеального конвейера [16, 17].

Очевидно, что в каждом такте выполняются разные этапы обработки команд, соответственно время такта будет вычисляться путем максимального выполнения всех этапов. К тому же, нужно учесть, что для перехода команды

от одного этапа обработки к другому так же требуется время, связанное с хранением промежуточных результатов обработки в регистры буфера.

Допустим, для выполнения отдельных этапов обработки потребуются следующие затраты времени (в некоторых условных единицах) [15, с. 60]:

$$T_{H1} = 20; T_{H2} = 15; T_{H3} = 20; T_{H4} = 25; T_{H5} = 20$$

Тогда, предполагая, что дополнительное время составляет $t = 5$ единиц, вычислим время такта [15, с. 60]:

$$T = \max\{T_{H1}=20, T_{H2}=15, T_{H3}=20, T_{H4}=25, T_{H5}=20\} + \Delta t = 30 \quad (1.3)$$

Теперь проанализируем время выполнения одной команды и некоторой группы команд при последовательной и конвейерной обработке данных [15, с. 60].

При последовательной обработке время выполнения N команд будет следующим [15, с. 60]:

$$T_{\text{посл}} = N \times (T_{H1} + T_{H2} + T_{H3} + T_{H4} + T_{H5}) = 100N \quad (1.4)$$

Из анализа таблицы 1.1 следует, что при конвейерной обработке после того, как получен результат выполнения первой команды, то результат очередной команды появляется в следующем такте работы процессора. Таким образом, время при конвейерной обработке N команд будет вычисляться следующей формулой [15, с. 61]:

$$T_{\text{конв}} = 5T + (N-1) \times T \quad (1.5)$$

Бесспорно, что при определено длительной работе, быстродействие конвейера будет значительно превышать быстродействие, которое достигается при последовательной обработке команд. Это увеличение будет тем больше, чем меньше длительность такта конвейера и чем больше количество выполненных за исследуемый период команд. Уменьшение длительности такта может достигаться разделением выполнения команды на большое число этапов, каждый из которых включает в себя сравнительно простые операции и поэтому будет выполняться за более короткий промежуток времени [15, с. 61].

Длительность обработки некоторого количества команд (1, 2, 5, 10, 100, 1000) при последовательном и конвейерном выполнении представлена в таблице 1.2 [15, с. 61].

Таблица 1.2 – Оценка эффективности конвейерной обработки

Количество команд	Время	
	Последовательное выполнение	Конвейерное выполнение
1	100	150
2	200	180
5	500	270
10	1000	420
100	10000	3120
1000	100000	30120

Несмотря на достоинства конвейерной обработки команд, в ней часто возникают ошибки (конфликты), которые замедляют и блокируют выполнение очередной команды в предназначенном для нее такте. Такие конфликты могут быть вызваны по разным причинам, но в целом их можно отнести к структурным [18, 19].

В результате проведенного анализа принципа конвейерной обработки данных, была получена следующая концептуальная модель, представленная на рисунке 1.4 [15, с. 62].

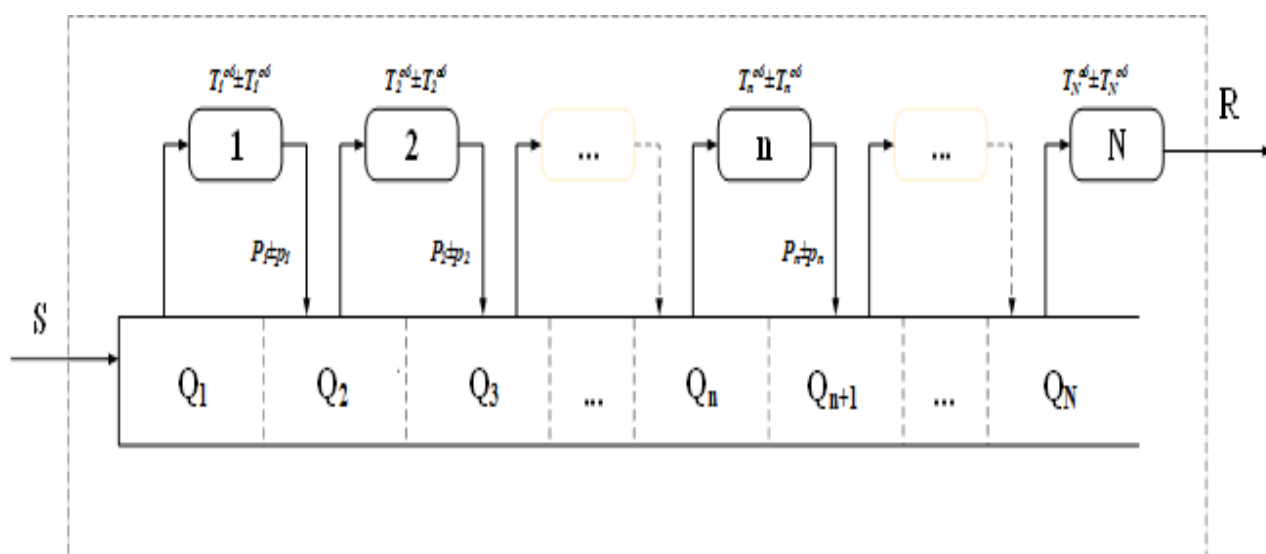


Рисунок 1.4 – Концептуальная схема системы с конвейерной обработкой данных

На рисунке 1.4 показано [15, с. 61]:

S – задания, поступающие на обработку;

$Q_1, Q_2, Q_3, \dots, Q_n, Q_{n+1}, \dots, Q_N$ – очереди заданий, поступающих на обработку;

$T_1^{об} \pm T_1^{об}, T_2^{об} \pm T_2^{об}, T_n^{об} \pm T_n^{об}, \dots, T_N^{об} \pm T_N^{об}$ – единица условного времени, затраченная на обработку 1, 2, ..., N-го задания.

$P1 \pm p1, P2 \pm p2, Pn \pm pn, \dots$ – страницы, на которые увеличивается объем задания на выходе 1, 2, ..., N-го процессоров;

R – задания, которые обработались.

Таким образом, конвейерная обработка способствует улучшению в использовании аппаратных ресурсов для определенного набора процессов, каждый из которых применяет данные ресурсы заранее предусмотренным методом.

1.4 Факторы, влияющие на внедрение систем автоматизации тепловых пунктов

Рассмотрим основные факторы, показывающие необходимость автоматического регулирования расхода тепла на отопление в тепловых пунктах. Они могут быть классифицированы в соответствии с таблицей 1.3.

Таблица 1.3 – Основные факторы, вызывающие необходимость местного регулирования

Наименование фактора	Характеристика
Несоответствие статических характеристик системы отопления режиму центрального регулирования расхода тепла	Значения расчетных температур воздуха в зданиях (жилые дома, школы, детские сады, больницы и пр.). Значения расчетных температур наружного воздуха. Несоответствие поверхности нагрева приборов, установленных в здании, температурному режиму тепловой сети. Неодинаковое охлаждение воды в трубах при транспорте теплоносителя до различных зданий
Неодинаковые динамические характеристики отапливаемых зданий, абонентских систем отопления и участков тепловой сети от генератора тепла до данного здания	Разная тепловая устойчивость зданий. Неодинаковые динамические характеристики систем отопления (радиаторных, панельных, лучистого отопления). Разные величины транспортного запаздывания в тепловой сети (до данного здания)

Наименование фактора	Характеристика
Влияние на регулируемую величину возмущений (температура и расход теплоносителя), действующих между источником тепла и вводом в здание	Работа установок горячего водоснабжения. Включение установок приточной вентиляции. Переключения в тепловой сети
Неоднородный характер теплопотребления	Невозможность осуществления центрального регулирования по отопительному графику на всем диапазоне тепловых нагрузок

Анализируя указанные факторы (таблица 1.3) становится ясно, что при отсутствии местного автоматического регулирования в ряде случаев могут иметь место серьезные нарушения теплового режима зданий [20].

1.5 Формулировка цели и постановка задач исследования

На основании проведенного аналитического обзора исследования разработок и применения систем автоматизации для различных объектов теплоэнергетики можно сделать следующие выводы:

- для повышения качества и экономичности теплоснабжения зданий и сооружений необходимо повсеместное применение систем автоматизации;
- местное регулирование производится в абонентских вводах и тепловых пунктах и преследует цель скорректировать режим центрального регулирования расхода тепла;
- автоматическая стабилизация теплогидравлического режима означает защиту системы отопления от влияний разного рода возмущающих воздействий тепловой сети (работа установок горячего водоснабжения с параллельно включенными подогревателями, переключения в сети и пр.);
- система автоматического управления комплексно обеспечивает решение задач перспективного и текущего планирования учета и статистики параметров теплопотребления.

Таким образом, опираясь на вышеизложенное, сформулирована цель диссертационного исследования: повышение эффективности теплообеспечения за счет автоматизации тепловых пунктов, использующих в процессе получения информации о параметрах теплопотребления и принятия управляющих решений способы конвейерной обработки данных.

Для достижения поставленной цели необходимо решение следующих научно-технических задач:

- 1) анализ современного состояния систем автоматизации и методов обработки информации параметров теплоснабжения тепловых пунктов;
- 2) разработка комбинированных методов получения и конвейерной

обработки информационных потоков в централизованных системах теплоснабжения в распределенных АСУ тепловыми пунктами;

3) разработка системы регулирования работы теплового пункта на основе конвейерной обработки данных;

4) разработка структуры программно-технического комплекса и проведение анализа используемого программного обеспечения в системах теплоснабжения;

5) разработка программы для создания и редактирования схем теплоснабжения тепловых пунктов с возможностью считывания реальных данных;

6) экспериментальные исследования эффективности применения автоматизации тепловых пунктов с использованием конвейерной обработки данных.

Выводы по разделу

1 Проведен аналитический обзор применения тепловых пунктов и их оборудования, рассмотрены параметры работы теплового пункта. Показана актуальность применения систем автоматизации при работе с тепловыми пунктами.

2 Рассмотрены основные существующие режимы обработки информации. Выбрана наиболее актуальная схема обработки данных – способ конвейерной обработки.

3 Рассмотренные варианты регулирования обосновывают эффективное использование конвейерной обработки данных при управлении тепловыми пунктами в городских условиях.

4 Осуществлено обоснование, постановка вопросов, сформулированы цели и задачи диссертационного исследования.

2 КОНВЕЙЕРНАЯ ОБРАБОТКА ДАННЫХ

2.1 Комбинированные методы получения и обработки информационных потоков в централизованных системах теплоснабжения

Известно, что внедрение и промышленная эксплуатация информационно-управляющих систем для теплоснабжения ориентировочно обеспечит [21]:

- экономию мощностей по выработке тепла и перекачиванию теплоносителя;
- снижение аварийности в теплоснабжающих системах мегаполисов;
- снижение эксплуатационных затрат на электроэнергию.

Рассмотрим основные направления исследований и технические решения по сбору, передаче и обработке информации на базовом объекте систем теплоснабжения – тепловом пункте.

Эксплуатация, наладка и развитие теплоснабжающих комплексов потребовало системного подхода к исследованиям энергетических проблем, а также к решению задач, наладки и эксплуатации систем энергетики. Важное влияние на развитие этого направления оказали исследования и разработки В.К. Аверьянова и С.И. Быкова [22, 23], Б.Н. Громова [24], Г.В. Монахова [25, 26], Н.И. Карасева [27, 28] и многих других. В этих работах решались задачи информатизации теплоснабжающих систем мегаполисов, которые рассматривались как объекты контроля. Так работы [29, 30] направлены на системную интеграцию информационных и математических моделей, образующих базы данных и знаний для системы поддержки принятия решений по наладочным мероприятиям теплоснабжающих систем мегаполисов. При расчетах их тепловых режимов были решены задачи анализа установившихся гидравлических режимов теплоснабжающей системы, разработано математическое обеспечение системы поддержки принятия решений, содержащее модели движения теплоносителя в технологических элементах теплоснабжающей системы; модели установившихся гидравлических режимов теплоснабжающих систем в различных режимных ситуациях и т.д.

На рисунке 2.1 представлена обобщенная схема АСУТП системы централизованного теплоснабжения города [31].

Многоуровневая распределенная система управления объектами системы централизованного теплоснабжения строится по объектному принципу.

Объект информатизации представлен следующими агрегированными технологическими установками: ИТ - источник тепла, представленный технологической структурой ТЭЦ или РК (районной котельной); ЦТП - центральный тепловой пункт (используется для обслуживания группы потребителей (зданий, промышленных объектов); ИТП - индивидуальный тепловой пункт потребителя (используется для обслуживания одного потребителя (здания или его части); НС - сетевая насосная станция на подающем или обратном трубопроводах магистральной сети; КП - контролируемый пункт для концентрации измерительной и управляющей

информации на трубопроводной сети [31, с.35].

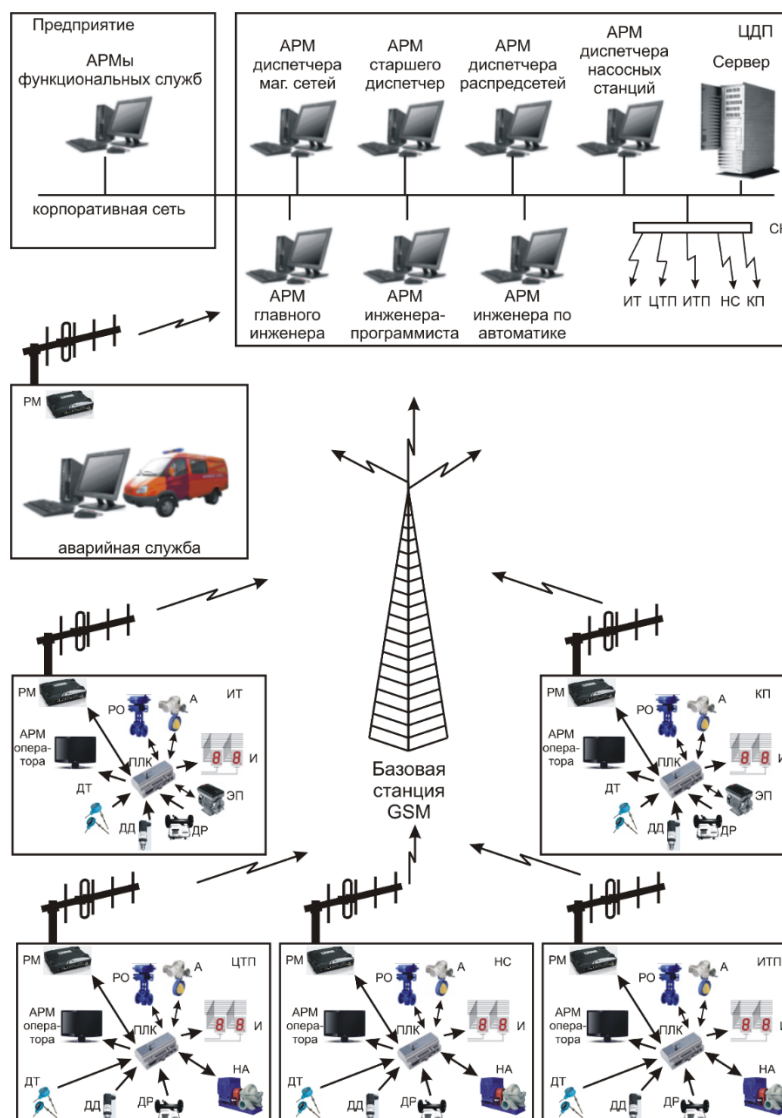


Рисунок 2.1 – Обобщенная схема АСУТП системы централизованного теплоснабжения города

Идеология таких систем была ориентирована на недостаточные возможности систем тепловой автоматики, а также на режимные расчеты систем с учетом различных эксплуатационных и сезонных факторов. Это в большей степени не управляющие, а информационные системы.

Для обработки значительного объема информации эффективны методы конвейерной обработки данных [32-35]. Применительно к системам централизованного теплоснабжения города эти методы описаны в [36].

Однако, централизованный подход, при котором объектом контроля и управления для информационно - управляющей системы является теплоснабжающий комплекс крупного города, требует организации двухсторонних потоков функционально избыточной информации от тепловых

пунктов к серверам и от серверов к тепловым пунктам. Циклический опрос информации по объектам контроля каждого теплового пункта занимает большие интервалы времени, что значительно увеличивает время реакции на события и снижает эффективность управления объектами тепловых пунктов. При любом отказе на верхнем уровне вся система становится неработоспособной.

Применение стандартных технологий циклического опроса всех локальных объектов, входящих в централизованную систему теплоснабжения, без учета динамических характеристик измеряемых параметров, в числе и при конвейерной обработке этой информации, приводит к значительному увеличению времени сбора, обработки, передачи информации и принятия решений в централизованной системе теплоснабжения.

В последнее время в тепловой автоматике на базе программно-аппаратных средств, включая современные интеллектуальные датчики и промышленные контроллеры, начали интенсивно внедряться системы, ориентированные на решение локальных задач автоматизации на уровне тепловых пунктов [37]. Тепловой пункт (ТП) - один из главных элементов системы централизованного теплоснабжения зданий, выполняющий функции приема теплоносителя, преобразования (при необходимости) его параметров, распределения между потребителями тепловой энергии и учета ее расходования. В зависимости от предназначения, условий присоединения потребителей к тепловой сети, требований заказчика и других ТП может состоять из ряда отдельных функциональных узлов.

Для упрощения процесса проектирования, комплектации и монтажа ТП могут изготавливаться в заводских условиях и поставляться на объект строительства в виде готовых блоков — блочный тепловой пункт (БТП). БТП представляет собой собранные на раме в общую конструкцию отдельные функциональные узлы, как правило, в комплекте с приборами и устройствами контроля, автоматического регулирования и управления. Фирма Danfoss предлагает для применения в Казахстане стандартные автоматизированные блочные тепловые пункты полной заводской готовности, предназначенные для присоединения к тепловой сети различных систем теплоснабжения и выполненные по типовым технологическим схемам с применением водоподогревателей на базе паяных или разборных пластинчатых теплообменников собственного производства [38].

В таких децентрализованных системах при их большом количестве, а в городах имеется значительное количество тепловых пунктов, в процессе работы возникают конфликтные режимы, когда локальные системы не справляются с управлением режимами, и необходимо централизованное перераспределение энергетических потоков.

Но увеличение количества задач, решаемых локальными системами автоматизации на тепловых пунктах, приводит к многократному увеличению измеряемой и обрабатываемой информации. При нескольких отводящих

магистральных на каждом тепловом пункте измерению подлежат десятки параметров. Полный цикл опроса всех тепловых пунктов города может составлять несколько часов.

В то же время такие параметры, как температура изменяются медленно, а расходы и давление имеют более высокое быстродействие. Срабатывание защит, выявление нештатных ситуаций требуют еще более высокого быстродействия.

Предлагается концепция измерения и обработки информации в централизованных системах теплоснабжения, базирующаяся на переменных циклах опроса параметров теплоснабжающих систем на тепловых пунктах с учетом их динамических характеристик с последующей конвейерной обработкой информации на верхнем уровне автоматизированных систем централизованного теплоснабжения.

Выделяется три группы параметров, подлежащих контролю на тепловом пункте:

- 1) контроль срабатывания защит;
- 2) контроль давления, расхода и электропотребления;
- 3) контроль температур.

Опрос параметров теплового пункта может быть осуществлен с разными периодами. В тепловом пункте выделяются группы параметров с различающимися динамическими характеристиками.

На тепловой пункт поступают задающие воздействия для приводов насосных станций.

На центральные диспетчерские пункты периодически передается информация о температуре теплоносителя, усредненные данные о давлении и расходе на тепломагистрали и электропотреблении. Оперативно передается информация о срабатывании защит.

Обработка информации на центральных диспетчерских пунктах осуществляется конвейерными методами.

Решение таких задач может обеспечить иерархическая структура, в которой на верхний уровень передается информация, необходимая для контроля и управления всей теплоснабжающей системой, а задачи управления и контроля каждого объекта теплового пункта передаются локальным подсистемам, которые задают переменные циклы опроса в зависимости от динамических характеристик технологических параметров (температура теплоносителей, давление, расход и т.п.) [31, с.37].

Предложенная концепция, сочетающая переменные циклы опроса на каждом уровне системы теплоснабжения с конвейерной обработкой информации на центральных диспетчерских пунктах, снизит нагрузки на каналы передачи информации, существенно уменьшит трафик при передаче информации от локальных тепловых пунктов на центральные диспетчерские пункты по сотовой связи, увеличит быстродействие и сократит время принятия решений в централизованных системах электроснабжения.

2.2 Организация конвейерного выполнения задач в распределенных АСУ теплопунктом

В 2008 году ГУ «Управление энергетики и жилищно-коммунального хозяйства ВКО» РК совместно с акиматами городов и районов проведена инвентаризация теплоисточников и объектов инженерной инфраструктуры. Проведенный в Восточно-казахстанской области анализ показал, что средний процент износа котельного оборудования составляет - 70%, тепловых сетей - 65%, водопроводных и канализационных - 85%, очистных сооружений и КНС - 75% и 70% соответственно [39]. Разработан проект программы модернизации объектов инженерной инфраструктуры Восточно-Казахстанской области на 2009-2013 годы и направлен в Канцелярию Премьер-министра Республики Казахстан [40].

Теоретические и методологические аспекты типизации задач в модульных системах обработки данных (СОД) отражены в работах отечественных и зарубежных авторов, таких как А.Г. Мамиконов [41], В.В. Кульба, С.А. Косяченко [42, 43], А.С. Миронов, Е.Н. Сидоров, А.А. Ашимов [44], Ю.Ю. Кесс, В.М. Ревако [45], А.В. Товмасын, Б. Дюран, П. Оделл [46], Х. Берж, О. Оре [47]. Модели и методы обработки информации в технических и организационно – технологических системах рассмотрены в работах В.В. Воеводина [48-50], Б.А. Головкина, К.Г. Самофалова [51], Г.М. Луцкого, А.Б. Барских, Е.Л. Шлимовича [52], А.П. Шабанова, Д. Филлипса, А. Гарсиа-Диаса [53], Р.В. Конвея, В.Л. Максвелла, Л.В. Миллера [54]. Конвейерное выполнение типовых задач в распределенных АСУ определяет новый этап в проектировании АСУ, требует модельно алгоритмического обеспечения выполнения задач и адаптации методики типизации задач для СОД конвейерного типа [55].

Предлагаемая технология обработки данных предоставляет возможность формирования показателей эффективности работы отдельных процессоров, реализующих конвейерные планы, СОД и АСУ теплопунктом в целом. Реализация этой технологии даст существенный эффект на практике, так как совершенствуется внутренняя технология работы АСУ. Помимо этого описанный подход дает следующие преимущества: увеличение пропускной способности конвейерной системы обработки данных; обеспечение однородности функций конвейерной системы обработки данных, что позволяет снизить требования к АСУ теплопункта; уменьшение времени и улучшение качества коммуникаций в системе, связывающие и координирующие работу нескольких теплопунктов [36, с. 103; 56].

Рассмотрим модельно-алгоритмические процедуры, позволяющие проследить влияние процесса накопления информации в АСУ на эффективность планирования задач при их конвейерном выполнении. Для класса задач в распределенных АСУ, проблема минимизации ресурсов решается в общем виде с помощью метода оценки достаточной мощности системы обработки данных (СОД) [36, с. 103].

В основе метода лежит математический аппарат теории массового обслуживания с ожиданием (с очередью). Задача разработки модели, позволяющей получить оценку мощности СОД с накоплением информации, является в настоящее время актуальной. Данная задача относится к проблеме минимизации ресурсов АСУ тепловых пунктов. С использованием в качестве инструмента известного математического аппарата для оценки мощности СОД будем использовать модель оценки влияния процесса накопления информации на эффективность конвейерной обработки данных в АСУ [36, с. 104; 56, с. 160]:

$$P(\leq T_{\text{зад}}) \geq \frac{1}{Q} \left(P_{\text{const}}^{k=1}(0) + P_{\text{const}}^{k=2}(2) + P_{\varphi\{\tau[V(t)]\}}^{k \geq 3}(j \leq J) \right), \quad (2.1)$$

где $T_{\text{зад}}$ – заданное максимально-допустимое значение времени ожидания требованием обслуживания;

$P(\leq T_{\text{зад}})$ – заданное минимально-допустимое значение вероятности превышения $T_{\text{зад}}$;

Q – максимальное количество требований, обслуживаемых в непрерывном интервале времени (интервале занятости);

$V(t)$ – объём записанной информации; значение $V(t)$ изменяется со временем;

$\tau[V(t)]$ – длительность одного интервала обслуживания. Является зависимой от объёма $V(t)$ информации величиной и определяется как [36, с. 104]:

$$\tau[V(t)] = \frac{\tau_{\text{const}} + \tau_{\text{var}}[V(t)]}{M}, \quad (2.2)$$

где M – количество процессоров в СОД;

τ_{const} – постоянная составляющая интервала обслуживания, определяемая в основном временем, затрачиваемым непосредственно на выполнение работ по обслуживанию требования;

$\tau_{\text{var}}[V(t)]$ – составляющая интервала обслуживания, которая зависит от объёма информации. Параметр $\tau_{\text{var}}[V(t)]$ определяется временем, затрачиваемым на выполнение действий с информацией и принятие решений;

k – порядковое место требования в интервале занятости;

j – время ожидания для k -го требования, выраженное в количестве интервалов обслуживания ($j=1,2,\dots,J$);

$P_{\text{const}}^{k=1}(0)$ – вероятность того, что требование, поступившее в СОД, не ожидает обслуживания. В одном интервале занятости таким требованием может быть только первое требование ($k=1$), при этом $P_{\text{const}}^{k=1}(0)$ для любого значения Q является постоянной, равной 1 [36, с. 104];

$P_{\text{const}}^{k=2}(0)$ – вероятность того, что длительность ожидания обслуживания второго, по порядку обслуживания в системе, требования ($k=2$), равна одному интервалу обслуживания. В одном интервале занятости $P_{\text{const}}^{k=2}(0)$ для любого

значения Q является постоянной, равной 1 [36, с. 104];

$P_{\varphi\{\tau[V(t)]\}}^{k \geq 3} (j \leq J)$ – вероятность того, что любое требование, начиная с третьего, по порядку обслуживания в интервале занятости, ожидало не более j интервалов обслуживания;

J – максимально-допустимое время ожидания, выраженное в количестве интервалов обслуживания. Параметр J соответствует параметру $T_{\text{зад}}$ и является зависимым от параметра $\tau [V(t)]$ [36, с. 104].

Модель позволяет получить зависимость между максимальным количеством Q требований, обслуживаемых в СОД в непрерывном интервале времени (интервале занятости), и объёмом $V(t)$ записанной информации при условии соблюдения заданных значений времени T ожидания требованиями обслуживания и вероятности $P(\leq T_{\text{зад}})$ его неперевышения. Рассмотренная модель позволяет определить зависимость между мощностью СОД и объёмом информации, постепенно накапливаемой и осваиваемой процессорами конвейерного плана обработки данных. Важным применением модели является выбор стратегии для организации процесса обслуживания требований с возможностью количественной оценки различных вариантов при распределении основных задач между процессорами [36, с. 105; 56].

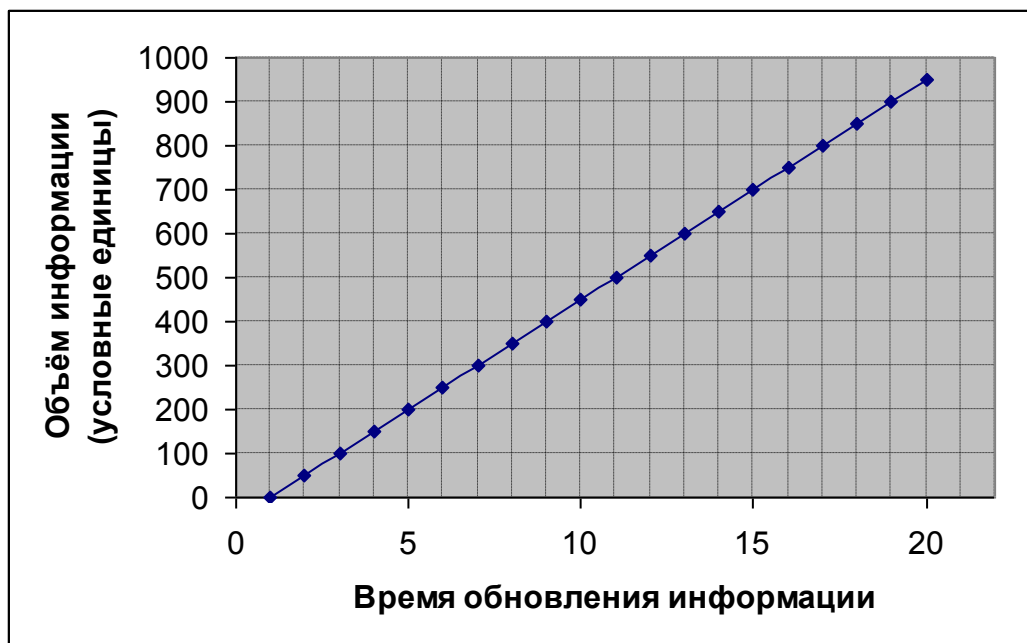


Рисунок 2.2 – Зависимость объёма записанной информации от времени её обновления

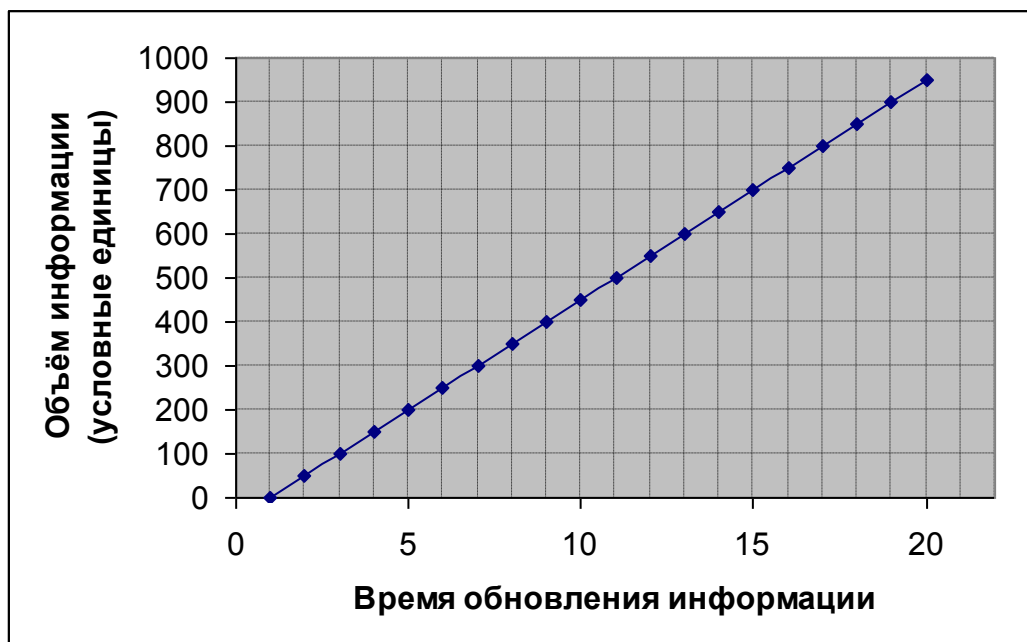


Рисунок 2.3 – $\tau_{\text{var}}[V(t)]$ в зависимости от времени обновления информации

Это позволит создать программный комплекс для управления тепловыми станциями с конвейерной обработкой данных.

2.3 Использование систолических массивов для создания систем автоматизации с высокой скоростью передачи данных от теплопунктов

Будем считать, что при построении систолических систем используются только плоские многоугольники. Они имеют одинаковые конфигурации и размеры, пристыковываются друг к другу без зазоров и все вместе находятся в одной плоскости, образуя некоторый плоский массив. Подобные систолические системы и называются систолическими массивами.

Прежде чем обсуждать аспекты конструирования и функционирования систолических массивов, рассмотрим простой пример. Это один из первых систолических массивов, построенных с момента возникновения самой идеи создания систем такого типа. Предположим, что необходимо построить специализированную вычислительную систему, на которой достаточно быстро реализуется операция вычисления матрицы $D = C + AB$, где [50, с. 484]

$$A = \begin{bmatrix} a_{11} & a_{12} & & 0 \\ a_{21} & a_{22} & a_{23} & \\ a_{31} & a_{32} & \dots & \dots \\ 0 & a_{41} & \dots & \dots \\ \dots & & & \dots \end{bmatrix}, B = \begin{bmatrix} b_{11} & b_{12} & b_{13} & 0 \\ b_{21} & b_{22} & \dots & \dots \\ \dots & b_{32} & \dots & \dots \\ 0 & \dots & & \dots \end{bmatrix}, C = \begin{bmatrix} c_{11} & c_{12} & c_{13} & c_{14} & 0 \\ c_{21} & c_{22} & c_{23} & \dots & \dots \\ c_{31} & c_{32} & \dots & \dots & \\ c_{41} & \dots & \dots & & \\ 0 & \dots & & & \end{bmatrix} \quad (2.1)$$

Здесь все матрицы – ленточные порядка n . Матрица A имеет одну диагональ выше и две диагонали ниже главной. Матрица B имеет одну

диагональ ниже и две диагонали выше главной. Матрица C имеет по три диагонали выше и ниже главной [50, с. 484].

Пусть в нашем распоряжении имеется достаточно большое число функциональных устройств, выполняющих скалярную операцию $C + DE$ и осуществляющих одновременно передачу данных. Такие операции реализуются либо на четырех-, либо на шестиугольных систолических ячейках (рисунок 2.5) [50, с. 484].

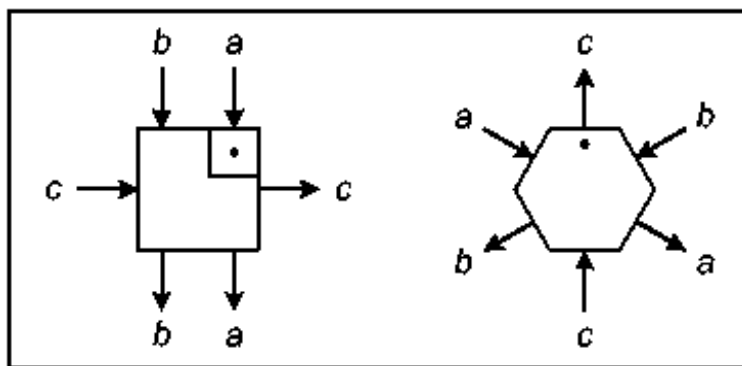


Рисунок 2.5 – Систолические ячейки с операцией $F + DE$

Здесь каждое функциональное устройство имеет три входа a, b, c и три выхода a, b, c . Входные (in) и выходные (out) данные в общем случае связаны соотношениями [50, с. 484]:

$$c_{out} = c_{in} + a_{in}b_{in}, \quad b_{out} = b_{in}, \quad a_{out} = a_{in}. \quad (2.2)$$

Если в момент выполнения операции какие-то данные не поступают, то они заменяются нулями. Точка указывает ориентацию систолической ячейки на плоскости [50, с. 485].

На рисунке 2.6 представлен систолический массив, реализующий матричную операцию $C + AB$ с указанием расположения и схемы подачи входных данных. При построении систолического массива использованы шестиугольные систолические ячейки. Вся плоскость покрыта регулярной косоугольной решеткой, образованной двумя семействами равноотстоящих параллельных прямых, пересекающихся под углами соответственно 60° и 120° . В узлах решетки размещены функциональные устройства и входные данные. Система работает по тактам. За каждый такт все данные перемещаются в соседние узлы по направлениям, указанным стрелками [50, с. 485].

На рисунке 2.6 показано состояние в некоторый момент времени. В следующий такт все данные переместятся на один узел и элементы a_{11}, b_{11}, c_{11} окажутся в одном функциональном устройстве, находящемся на пересечении штриховых линий. Следовательно, будет вычислено выражение $c_{11} + a_{11}b_{11}$. В этот же такт данные a_{12} и b_{21} приблизятся вплотную к функциональному

устройству, находящемуся в вершине систолического массива. Далее, в последующем такте все данные снова переместятся на один узел в направлении стрелок и в верхнем функциональном устройстве окажутся элементы a_{12} , b_{21} и результат срабатывания функционального устройства, находящегося снизу, т. е. $c_{11} + a_{11}b_{11}$. Следовательно, будет вычислено выражение $c_{11} + a_{11}b_{11} + a_{12}b_{21}$. Это есть элемент d_{11} матрицы D [50, с. 486].

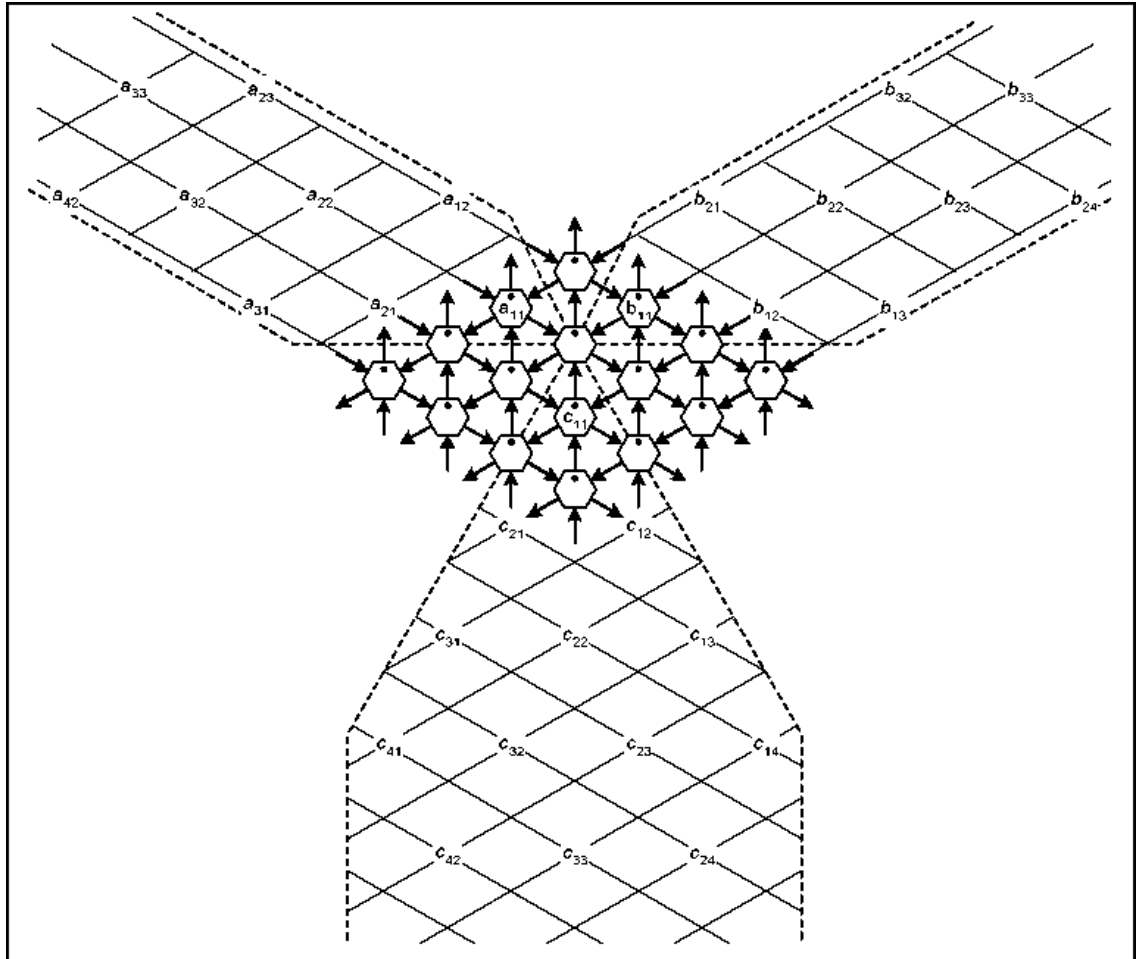


Рисунок 2.6 – Систолический массив для перемножения ленточных матриц

Продолжая потактовое рассмотрение процесса, можно убедиться, что на выходах функциональных устройств, соответствующих верхней границе систолического массива, периодически через три такта будут выдаваться элементы матрицы D . На каждом выходе будут появляться элементы одной и той же диагонали. Примерно через $3n$ тактов будет закончено вычисление всей матрицы D . Загруженность каждой систолической ячейки асимптотически равна $1/3$ [50, с. 486].

Для разработки методологии построения систолических массивов исследуем сначала вид графов алгоритмов, реализуемых подобными вычислительными системами. Не будем принимать во внимание, что выходные данные функциональных устройств могут совпадать с какими-то входными

данными, и что каждое устройство имеет более одного выхода, а не один, как чаще всего предполагалось до сих пор. На данном этапе важнее не то, какие операции выполняют функциональные устройства, а то, как они будут управляться и в каком временном режиме смогут работать. Разрабатывая методологию построения систолических массивов, необходимо учесть, во-первых, возможность локального управления работой системы и, во-вторых, возможность реализации всего спектра временных режимов, включая наискорейшие. Первую особенность мы сохраним безусловно и будем строить лишь такие систолические массивы, в которых систолические ячейки сами управляют своей работой и моментами срабатывания. Принимая во внимание вторую особенность, мы будем строить систолические массивы, реализующие алгоритм по возможности максимально быстро. Кроме этого, мы сохраним еще одну особенность, подмеченную в реально созданных систолических массивах. Именно, любая систолическая ячейка, начав работать в какой-то момент времени, продолжает срабатывать каждый такт или с постоянным периодом, пока не закончится весь процесс ее функционирования. Другими словами, в работе систолических ячеек не бывает как больших, так и неконтролируемых перерывов [50, с. 487].

Очевидно, что плоскость можно покрыть без зазоров как четырех-, так и шестиугольниками одного размера. Если задан какой-то лежащий в плоскости систолический массив, то он вырезает из покрытия плоскости соответствующую ему односвязную фигуру. Выберем в плоскости массива два семейства параллельных равноотстоящих прямых линий таким образом, чтобы центр любой систолической ячейки находился в точке пересечения двух прямых. Эти семейства выбираются однозначно в случае четырехугольников и неоднозначно в случае шестиугольников. На основе выбранных семейств построим аффинную систему координат. Пусть она такова, что центры всех систолических ячеек описываются двумерными векторами с целочисленными координатами. Соседние ячейки всегда имеют равными одну из координат и различающуюся на 1 другую координату. Обозначим оси координат x , y . Теперь возьмем любую ось t , не лежащую в плоскости x , y , и установим на ней направление. При $t=0, 1, 2, \dots$ проведем плоскости, параллельные плоскости x , y . Предположим, что $t = 0$ соответствует плоскости систолического массива [50, с. 487].

Будем считать t осью времени. Пусть систолический массив начинает работать при $t=0$ и продолжает функционировать с тактом, равным 1. Построим ориентированный граф. Вершинами являются точки трехмерного пространства, в которых координаты x , y соответствуют систолической ячейке, срабатывающей в момент t . Дуги символизируют передачу информации от ячейки, которая произвела ее в момент t , к ячейке, которая получит ее в момент $t + 1$. Очевидно, что число различных дуг, рассматриваемых как векторы в трехмерном пространстве, не превосходит числа различных по направлению передач информации от одной систолической ячейки к другой соседней в

самом систолическом массиве. При сделанных предположениях координаты векторов-дуг по осям x , y равны 0, 1 или -1 , а по оси t координата всегда равна 1. Таким образом, если ячейки систолического массива работают в синхронном режиме, то систолический массив реализует алгоритм, граф которого изоморфен подграфу бесконечного регулярного графа [50, с. 488].

Таким образом, систолические массивы как специализированные вычислительные системы могут реализовывать лишь вполне определенный класс алгоритмов, которые относятся к регулярным алгоритмам. Реально используемые нами алгоритмы являются регулярными изначально. Вспомогательные алгоритмы легко сводятся к регулярным [50, с. 488].

Самой характерной чертой систолических массивов является отсутствие каких-либо дополнительных линий связи при соединении входов и выходов систолических ячеек. Соединение отдельных ячеек между собой лишь в местах соприкосновения приводит к тому, что в систолических массивах все связи имеют минимально возможные длины и, следовательно, оказываются минимальными временные затраты на передачу информации от одних функциональных устройств к другим. Подразумевая именно эту особенность, говорят, что в систолических массивах реализуется принцип близкодействия. Это, в свою очередь, приводит к тому, что в графах связей систолических массивов дуги никогда не пересекаются. Точнее, дуги могут иметь общими только концевые вершины. Такие графы называются плоскими [50, с. 488].

Как видно из рисунка 2.5 и формулы (2.2), систолические ячейки могут выполнять функции переносчика информации. Это очень важная их черта. Она помогает в вычислительных системах организовывать транспортные связи. Если некоторое данное должно использоваться во многих устройствах, то необходимо применение специальных способов его рассылки. Один из способов реализует шинную связь. Рассылаемое данное поступает на шину и одновременно принимается несколькими устройствами. Для систолических массивов такой способ неприемлем. В подобных вычислительных системах транспортные связи оказываются более подходящими. В этом случае данное поступает в устройство, используется в нем как операнд и, не изменяясь, пересылается по транспортной магистрали в соседнее устройство, где используется, пересылается дальше и т. д. [50, с. 488].

При преобразовании графов алгоритмов многократная передача информации через систолические ячейки означает замену "длинных" дуг графа путем, составленным из "коротких" дуг. Данный путь не обязан быть прямолинейным. Обратим особое внимание на то, что такая замена может увеличить длину критического пути графа алгоритма и, следовательно, увеличить время наискорейшей реализации. Вводя транспортные связи, об этом следует помнить [50, с. 489].

Теперь ключевыми моментами методологии создания систолических массивов становятся умение строить и исследовать графы алгоритмов. Для регулярных графов операция гомоморфной свертки может быть выполнена как

операция однократного или даже многократного ортогонального проектирования. Выбор направления проектирования ограничен лишь требованием, чтобы последняя проекция представляла собой плоский граф. В целом при увеличении числа вершин, лежащих на прямых с выбранным направлением, повышается загруженность систолических ячеек. Если на каждой такой прямой все вершины связаны одним путем, то сохраняется весь спектр временных реализаций алгоритма. Во всех остальных случаях возможность наискорейших реализаций не гарантируется [50, с. 489].

Рассмотрим механизм использования систолических массивов. Предположим, что граф алгоритма имеет вид, показанный на рисунке 2.7. Подобные графы свойственны используемым при вычислении расчётных параметров вычислительным методам линейной алгебры и математической физики. Этот граф регулярный. На горизонтальных или вертикальных прямых, а также на прямых, параллельных прямой, проходящей через вершины с кружками, либо нет ни одной вершины, либо все вершины связаны одним путем. Проектируя граф вдоль таких прямых на перпендикулярные им прямые линии, получаем различные схемы одномерных систолических массивов (рисунок 2.7). Все систолические массивы имеют возможность реализовать весь спектр временных режимов, в том числе, наискорейшие. Тем не менее, они имеют разное число устройств и, следовательно, разную их загруженность. Схемы передач информации также различные. В двух из них осуществляется односторонняя передача, в одной – двухсторонняя. Наиболее простой представляется схема систолического массива, расположенная на рисунке 2.7 справа. Однако учет дополнительных факторов, таких как подача к систолическим ячейкам входных данных алгоритма, может в некоторых случаях сделать более предпочтительными другие схемы [50, с. 490].

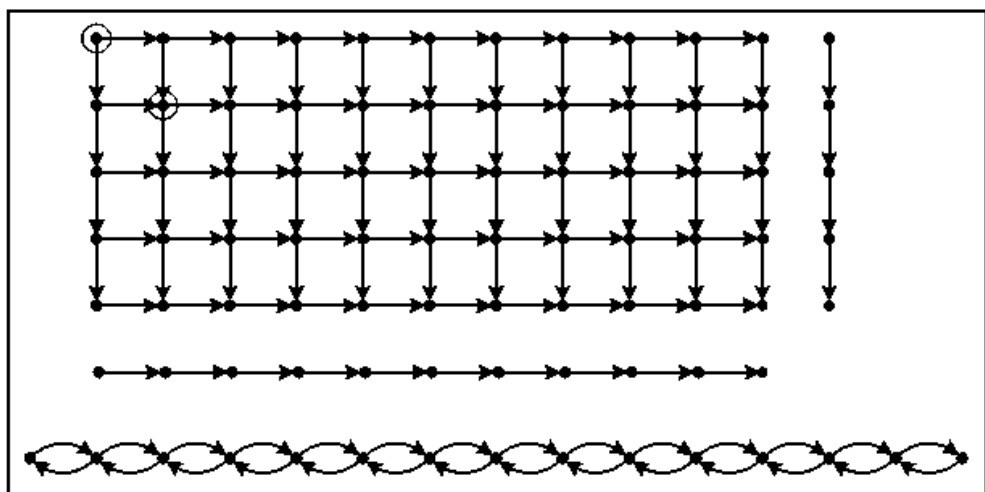


Рисунок 2.7 – Одномерные систолические массивы

В каждый такт работы систолического массива срабатывают какие-то его

ячейки. Исходя из принципов построения подобных вычислительных систем, можно утверждать, что соответствующие этим ячейкам операции не могут быть связаны между собой информационно. Следовательно, процесс функционирования систолического массива в действительности реализует некоторую строгую параллельную форму графа алгоритма. Ярусы параллельной формы описываются работающими одновременно систолическими ячейками, номера ярусов – временами срабатывания ячеек [50, с. 490].

Граф систолического массива мы получаем путем ортогонального проектирования графа алгоритма вдоль прямой линии. Обозначим ее направляющий вектор через p . Если граф алгоритма регулярный, то, существует вектор q , который образует острые углы со всеми базовыми векторами. Поэтому на любой гиперплоскости с направляющим вектором q не могут находиться вершины, связанные дугами. Другими словами, совокупность гиперплоскостей, проходящих через вершины графа алгоритма и имеющих в качестве направляющего вектор q , разбивает все вершины на ярусы строгой параллельной формы. Если существует хотя бы одна дуга, параллельная вектору p , то векторы p и q не могут быть ортогональными [50, с. 490].

Выберем подходящий вектор q и будем реализовывать на систолическом массиве соответствующую параллельную форму. Проведем через вершины графа алгоритма гиперплоскости с направляющим вектором q . Перенумеруем их подряд в направлении этого вектора. Пусть наименьший номер гиперплоскости, содержащий хотя бы одну вершину, равен 1. Эти и только эти вершины определяют систолические ячейки, которые будут срабатывать в 1-й такт. Гиперплоскость с номером 2 определяет те и только те ячейки систолического массива, которые будут срабатывать во 2-й такт, и т. д. Тем самым фиксируется программа работы каждой систолической ячейки. В процессе работы систолического массива в его плоскости в соответствии с номерами гиперплоскостей распространяется фронт вычислений, показывая, как систолические ячейки вступают в процесс функционирования и как они его заканчивают. Каждая систолическая ячейка будет срабатывать не обязательно каждый такт, но обязательно периодически. Период для всех ячеек один и тот же. Он на 1 больше числа гиперплоскостей, помещающихся между соседними точками на прямой с направляющим вектором p [50, с. 491].

Построим модель систолического массива для необходимой нам при расчёте коэффициентов задачи вычисления матрицы $A + BC$, где A , B , C – заданные матрицы порядка n . Если элементы матрицы $A + BC$ находятся по обычным формулам, то для $n = 3$ граф алгоритма будет таким, как показано более жирными стрелками на рисунке 2.8. Легко проверить, что один и тот же элемент матрицы B или C требуется для выполнения операций, соответствующих вершинам, лежащим на прямой, параллельной оси j или i . Элементы матрицы A требуются для выполнения лишь операций, соответствующих начальным вершинам путей графа алгоритма. На рисунке 2.8

линиями без стрелок обозначены будущие транспортные связи. Направление передачи информации определяется выбором порядка выполнения операций [50, с. 491].

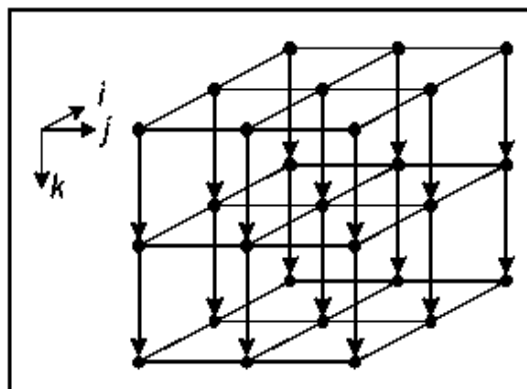


Рисунок 2.8 – Граф алгоритма и транспортные связи

Если спроектировать граф алгоритма на плоскость, перпендикулярную оси k , то проекция будет состоять только из изолированных вершин. В этом случае реализовать алгоритм за минимально возможное время можно лишь при рассылке начальных данных B, C с помощью шинных связей, параллельных осям j, i . Если для реализации операции $A + BC$ мы хотим построить систолический массив, то должны фигуру на рисунке 2.8 превратить в регулярный граф или, другими словами, установить направления передач информации. Пусть эти направления совпадают с направлениями осей i, j . Тем самым фигура на рисунке 2.8 становится регулярным координатным графом. Заметим, что при этом длина критического пути графа сразу увеличивается в три раза [50, с. 492].

Спроектируем полученный граф на плоскости, перпендикулярные оси i , вектору с координатами $(1, 1, 1)$ и оси j . Соответствующие схемы систолических массивов представлены на рисунке 2.9. Сразу видны различия множеств систолических ячеек и конфигураций связи. Кроме этого, нетрудно заметить общность схем на рисунках 2.9б и 2.8. Будем считать, что реализуется параллельная форма, определяемая гиперплоскостями с направляющим вектором $q = (1, 1, 1)$. В систолических массивах на рисунке 2.9а и 2.9в любая систолическая ячейка, начиная с некоторого момента, срабатывает каждый такт, на рисунке 2.9б – один раз в три такта. Штриховыми линиями отмечено распространение фронта вычислений. На рисунке 2.9а и 2.9в фронт распространяется по диагонали квадрата от угла, ближайшего к началу координат, к наиболее удаленному углу, на рисунке 2.9б от центра к границе.

Разные проекции приводят не только к различиям в числе вершин и конфигураций связей, но и к различиям в организации процессов. Так, в проекции на рисунке 2.9б имеется достаточное число связей для выполнения операций без использования какой-либо памяти. В проекциях на рисунке 2.9а и

2.9в связей меньше. Поэтому здесь каждая систолическая ячейка должна иметь по одной ячейке памяти для хранения соответственно элементов c_{kj} , b_{ik} матриц C , B . Это приводит, в свою очередь, к различиям в процессах загрузки и разгрузки систолических массивов.

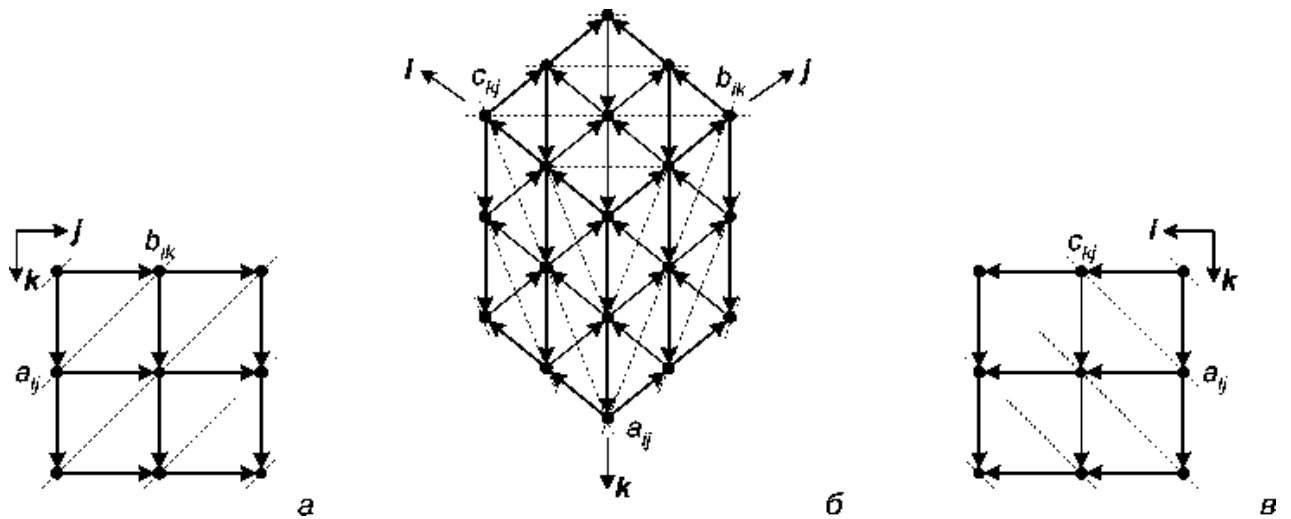


Рисунок 2.9 – Систолические массивы для операции $A + BC$

Заметим, что ни один из систолических массивов не позволяет реализовать операцию $A + BC$ за минимально возможное для нее время. Данный факт является ограничением за использование транспортных связей и он не противоречит предположению, что любое функциональное устройство, осуществляющее все необходимые операции после слияния вершин, может работать в тех же условиях передачи информации, что и в граф-машине. Но введение транспортных связей как раз и нарушает данные условия, что диктуется, в конечном счете, потребностью организации множественных рассылок.

Схема систолического массива и порядок подачи данных и выдачи результатов для $n = 4$ приведены на рисунке 2.10. Общее время решения системы порядка n приблизительно равно $5n$. Каждая ячейка срабатывает один раз в три такта. При этом средняя загруженность ячеек систолического массива будет около 0,1 [50, с. 496].

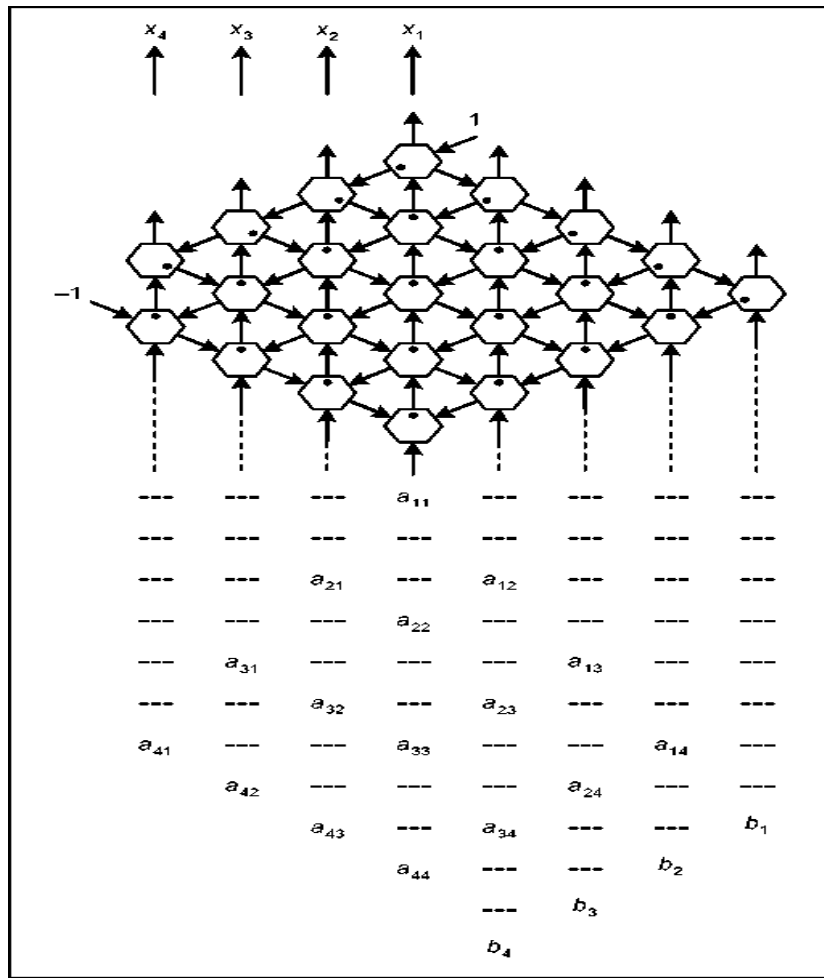


Рисунок 2.10 – Ввод данных и получение результатов

Рассмотрим теперь ортогональную проекцию координатного графа алгоритма для $n = 4$ по направлению $(0, 0, 1)$. Она представлена на рисунке 2.11. В новых координатах $\zeta=j, \eta = k$ ребро 1 и грань 2 перешли в линию $(1, 1) - (n, n)$, грани 3, 5 и внутренность пирамиды перешли на всю область, за исключением этой линии. Каждая ячейка систолического массива теперь будет срабатывать каждый такт после начала функционирования [50, с. 497].

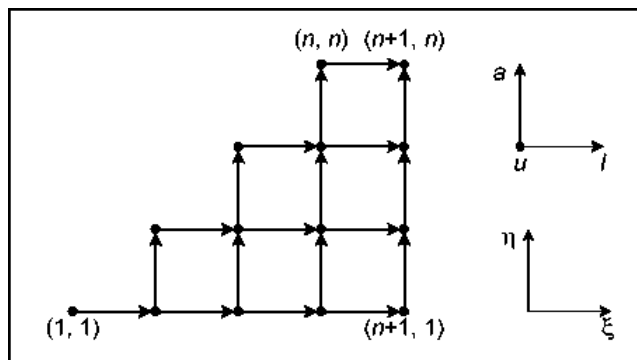


Рисунок 2.11 – Ортогональная проекция систолического массива

Схема систолического массива, порядок подачи данных и выдачи результатов приведены на рисунке 2.12. Результаты вычислений будут выдаваться из ячейки $(n+1, n)$ в последовательные такты, начиная с $(3n + 2)$ -го. Все ячейки массива меняют выполняемые ими функции во время счета, хотя в целом они делают примерно то же, что и в случае первой проекции. Загруженность ячеек систолического массива приблизительно равна 0,25.

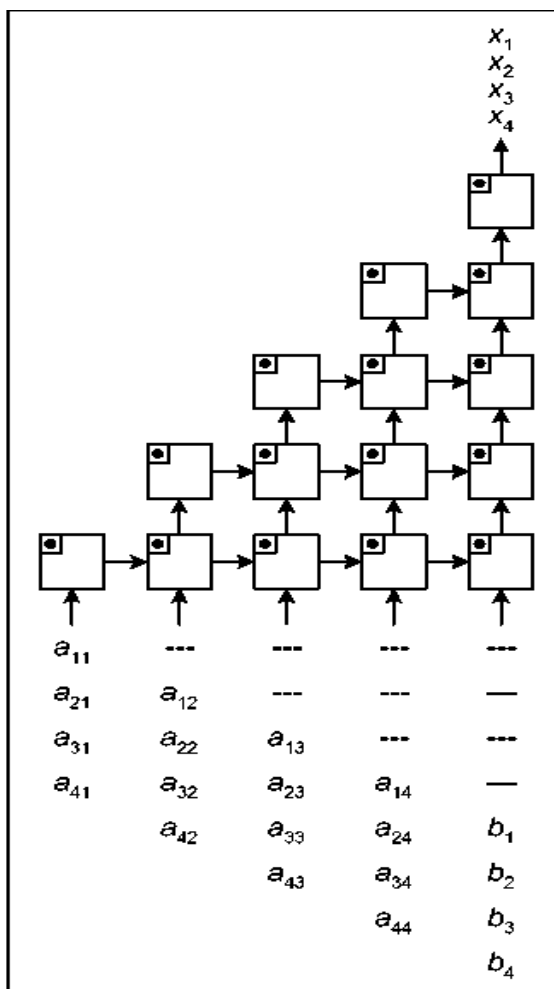


Рисунок 2.12 – Перемещение информации

Эти примеры показывают, что на одиночной задаче загруженность систолических массивов может оказаться не очень высокой. Особенно заметно она уменьшается из-за достаточно длинных этапов загрузки и разгрузки массивов. На загруженность влияет также выбор прямой, вдоль которой осуществляется проектирование графа алгоритма. Чем больше длина пути, связывающего соседние сливаемые вершины, тем реже срабатывает соответствующая систолическая ячейка и, следовательно, тем ниже ее загруженность [50, с. 499].

Для использования систолических массивов была разработана технология,

позволяющая осуществлять контроль и управление данными, поступающими от большого количества источников. Данная технология основана на параллельно-конвейерной обработке данных, которая реализована в виде стандартных функций в современных языках программирования и позволяют распараллеливать работу как с данными хранящимися внутри компьютера, так и обрабатывать данные множества компьютеров, объединенных в единую сеть. Однако в реальных условиях мы имеем ситуацию, когда сервер данных достаточно удалён от периферийного оборудования и компьютеров пользователя. Это приводит к тому, что с увеличением объёма поступающих данных время обработки увеличивается, тем самым может быть отложена обработка оперативной информации, имеющую важную составляющую, влияющую на безопасность или режимы работы периферийных устройств [64].

2.4 Методика обработки данных тепловых точек на основе технологии работы центрального процессора с процессами и потоками

Для повышения эффективности работы с большим потоком данных и организации приоритетов работы с данными предлагается перенести технологию работы центрального процессора с потоками на работу вычислительной машины с данными, поступающими из сети.

Удачным в персональном компьютере является организация работы центрального процессора с потоками. Данную организацию выгодно перенести с работой с реальными данными.

Для понимания механизма взаимодействия процессора с потоками необходимо дать некоторые определения и информацию, связанную с потоками и процессами [65, 66].

В нашем случае для возможности переноса будут рассматриваться пользовательские потоки.

"Одновременное" (или "конвейерно-параллельное") выполнение потоков подразумевает, что в том случае, если имеется всего один процессор с одним ядром, в один момент времени может выполняться только один поток. Однако операционная система может быстро переключать процессор с выполнения одного потока на другой и вследствие высокой частоты процессоров (обычно операционная система выделяет на обработку потока 0,6 нс) у пользователя возникает иллюзия одновременной работы нескольких программ. Такая ситуация называется псевдопараллельное или "конвейерно-параллельное" выполнение потоков.

Наглядно, можно показать конвейерно-параллельное выполнение потоков на рисунке 2.13 [58, с. 128].

Важным при работе процессора с потоками (процессами) является и возможность определения приоритетов потокам обрабатываемым процессором. Чем выше приоритет, тем больше времени выделяется на обработку данного потока.

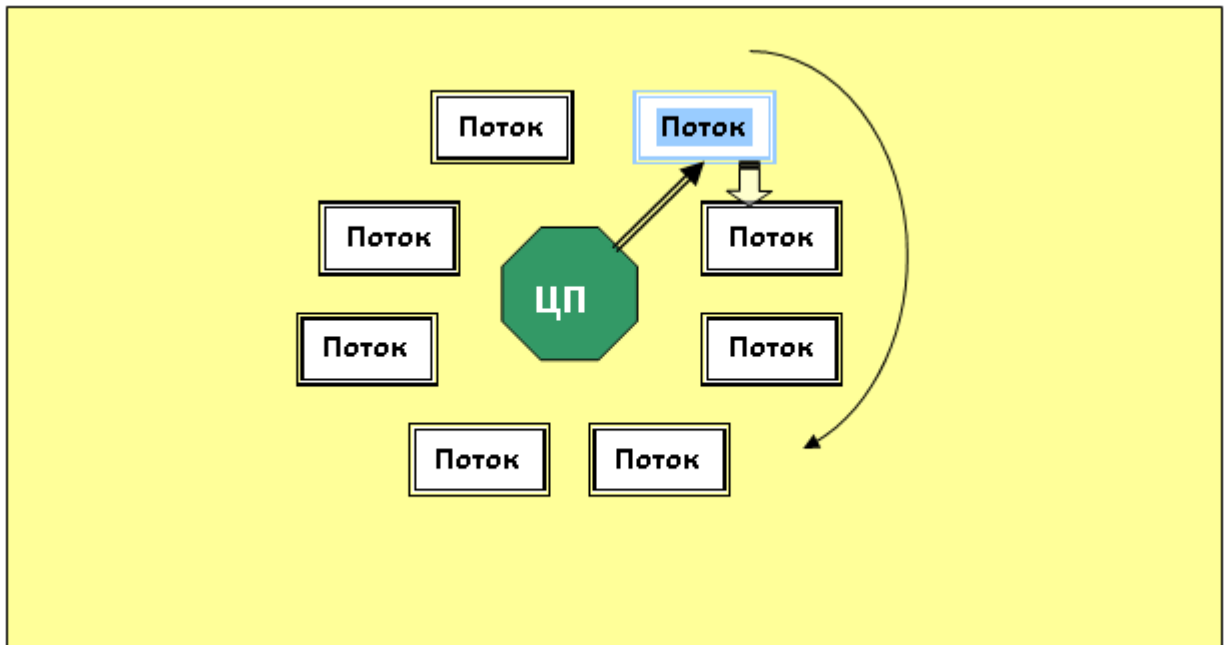


Рисунок 2.13 – Выделение операционной системой потокам кванты времени

Рассмотрим возможность применения данной технологии для обработки данных тепловыпуска.

Основными задачами тепловых пунктов являются [69]:

- преобразование вида теплоносителя;
- контроль и регулирование параметров теплоносителя;
- распределение теплоносителя по системам теплоснабжения;
- отключение систем теплоснабжения;
- защита систем теплоснабжения от аварийного повышения параметров теплоносителя;
- учет расходов теплоносителя и тепла.

Очевидно, что при выполнении данных задач необходимо учитывать приоритет задачи и работать не с одним тепловым пунктом, а с тем количеством тепловых пунктов, которые обслуживаются центральным тепловым пунктом.

Для этого стоит использовать стандартные функции центрального процессора по работе с процессами (рисунок 2.14). Для того чтобы из обычного любого приложения создать процесс нам необходима функция `CreateProcess`. Новый процесс запускает заданный исполняемый файл в контексте безопасности вызывающего процесса [70].

```

BOOL CreateProcess(LPCTSTR lpApplicationName, // указатель на имя исполняемого
файла
LPTSTR lpCommandLine, // указатель на командную строку
LPSECURITY_ATTRIBUTES lpProcessAttributes, // указатель на атрибуты безопасности
процесса
LPSECURITY_ATTRIBUTES lpThreadAttributes, // указатель на атрибуты безопасности
потока
BOOL bInheritHandles, // указатель на флаг наследования
DWORD dwCreationFlags, // флаги создания
LPCVOID lpEnvironment, // указатель на новый блок среды
LPCTSTR lpCurrentDirectory, // указатель на имя текущего каталога
LPSTARTUPINFO, // указатель на структуру
STARTUPINFO LPPROCESS_INFORMATION lpProcessInformation // указатель на
структуру PROCESS_INFORMATION );

```

Рисунок 2.14 – Функции центрального процессора по работе с процессами

В нашем случае исполняемый модуль LPCTSTR lpApplicationName выполняет обслуживание теплопункта, т.е. можно запустить необходимое количество процессов, которые будут контролировать работу всех теплопунктов входящих в обслуживание центрального теплопункта. Очень важно, что можно обеспечить безопасность работы теплопункта поставив нужный дескриптор безопасности как в процессе для соответствующего теплопункта, так и в потоке, отвечающему за определённую задачу.

Рассмотрим более детально работу предлагаемой методики обработки данных поступающих с теплопунктов. Схема работы такой системы показана на рисунке 2.15

Суть состоит в том, что для обслуживания каждого теплопункта создаётся процесс, который состоит из 6 потоков. Каждый поток соответствует задаче, выполняемой в теплопункте. Так потоку №1 соответствует преобразование вида теплоносителя, потоку № 2 соответствует контроль и регулирование параметров теплоносителя, потоку № 3 - распределение теплоносителя по системам теплоснабжения, потоку №4 - отключение систем теплоснабжения, потоку № 5 - защита систем теплоснабжения от аварийного повышения параметров теплоносителя, потоку №6 - учет расходов теплоносителя и тепла [68, с. 188; 71].

Для понимания работы персонального компьютера с теплопунктами необходимо рассмотрим функцию создания потока в операционной системе Windows.

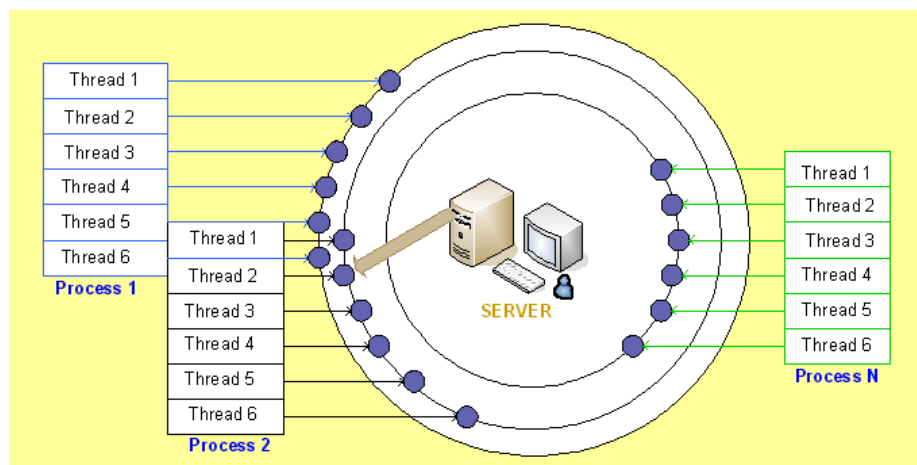


Рисунок 2.15 – Схема обслуживания персональным компьютером процессов

Для понимания работы персонального компьютера с теплопунктами необходимо рассмотрим функцию создания потока в операционной системе Windows.

При запуске нового процесса с помощью функции `CreateProcess` идентификатор потока хранится в члене `dwThreadId` структуры `PROCESS_INFORMATION`, которая заполняется после возврата вызова функции. Ссылка потока находится в члене `hThread` этой же структуры.

Поток создается в результате вызова функции `CreateThread` (рисунок 2.16).

В предлагаемой технологии использования процессов и потоков для обработки данных теплопунктов параметр `ThreadFunc` отражает определённую выполняемую теплопунктом задачу.

```

CreateThread(
    NULL, // default security attributes
    0, // use default stack size
    ThreadFunc, // thread function
    &dwThrdParam, // argument to thread function
    0, // use default creation flags
    &dwThreadId); // returns the thread identifier

```

Рисунок 2.16 – Функция `CreateThread`

Необходимо отметить о возможности введения приоритетов при обработке данных поступающих с теплопунктов. Это относится к созданию приоритета для определённого теплопункта, так и приоритета задачи при обработке данных поступающих с теплопункта. Это можно сделать по аналогии организации приоритетов при работе с процессами и потоками. Так класс приоритета процесса изменяем вызовом функции `SetPriorityClass`:

BOOL SetPriorityClass(HANDLE hProcess, DWORD fdwPriority)

Эта функция меняет класс приоритета процесса, определяемого описателем hProcess, в соответствии со значением параметра fdwPriority. Последний должен содержать одно из значений, в соответствии с приоритетом. Поскольку SetPriorityClass принимает описатель процесса, можно изменить приоритет любого процесса, выполняемого в системе, — если его описатель известен и есть соответствующие права доступа.

Для изменения приоритета первичного потока используем функцию:

BOOL SetThreadPriority(HANDLE hThread, int nPriority),

здесь параметр hThread указывает на поток, чей приоритет изменяется.

Предложенная методика позволит обеспечить высокое быстродействие обработки данных теплопунктов, а также увеличить надёжность и безопасность за счёт введения приоритета выполнения потока.

Выводы по разделу

1 Предложена концепция, сочетающая переменные циклы опроса на каждом уровне системы теплоснабжения с конвейерной обработкой информации на центральных диспетчерских пунктах. Это снизит нагрузки на каналы передачи информации, существенно уменьшит трафик при передаче информации от локальных тепловых пунктов на центральные диспетчерские пункты по сотовой связи, увеличит быстродействие и сократит время принятия решений в централизованных системах электроснабжения.

2 При включении однотипных последовательностей задач в конвейерный план для распределенных АСУ увеличивается эффективность работы, что применимо к задачам автоматизации тепловых пунктов.

3 Приведены основные принципы конвейерного способа обработки данных и показано, что наличие механизмов конвейерного выполнения задач в АСУ приводит к увеличению эффективности работы АСУ.

4 Рассмотрена технология использования систолических массивов, позволяющая осуществлять контроль и управление данными, поступающими от большого количества источников. Данная технология основана на параллельно-конвейерной обработке данных, которая реализована в виде стандартных функций в современных языках программирования и позволяют распараллеливать работу как с данными хранящимися внутри компьютера, так и обрабатывать данные множества компьютеров, объединенных в единую сеть.

5 Предложена методика обработки данных теплопунктов на основе технологии работы центрального процессора с процессами и потоками, которая позволит обеспечить высокое быстродействие обработки данных теплопунктов, а также увеличить надёжность и безопасность за счёт введения приоритета выполнения потока.

3 КОНВЕЙЕРНАЯ СИСТЕМА РЕГУЛИРОВАНИЯ РАБОТЫ ТЕПЛОПУНКТА

3.1 Автоматизация и модернизация тепловых пунктов

Для установки приборов, контролирующих работу теплопункта предварительно осуществляется модернизация работы теплопунктов [39, с. 324] (рисунок 3.1).



а



б

а) ТП до модернизации; б) ТП после модернизации

Рисунок 3.1 – Фотографии теплопункта

Основным элементом модернизированного теплопункта является тепловычислитель [72].

Принцип действия тепловычислителя основан на получении каналу внутренней связи от измерительных преобразователей (ИП) значений исходных физических величин: давлений (P), температур (T_1 , T_2), объемных расходов теплоносителя (V) и дальнейшем определении по заданному алгоритму количества потребленной тепловой энергии. При этом в тепловычислителе производится индикация и хранение в заданной форме накопленной информации. В случае необходимости накопленная информация может

передаваться по каналам внешней связи (RS-232C либо телефонный модем) потребителю [72, с. 72].

Структурная схема подключения измерительных преобразователей к тепловычислителю представлена на рисунке 3.2 [72, с. 72].

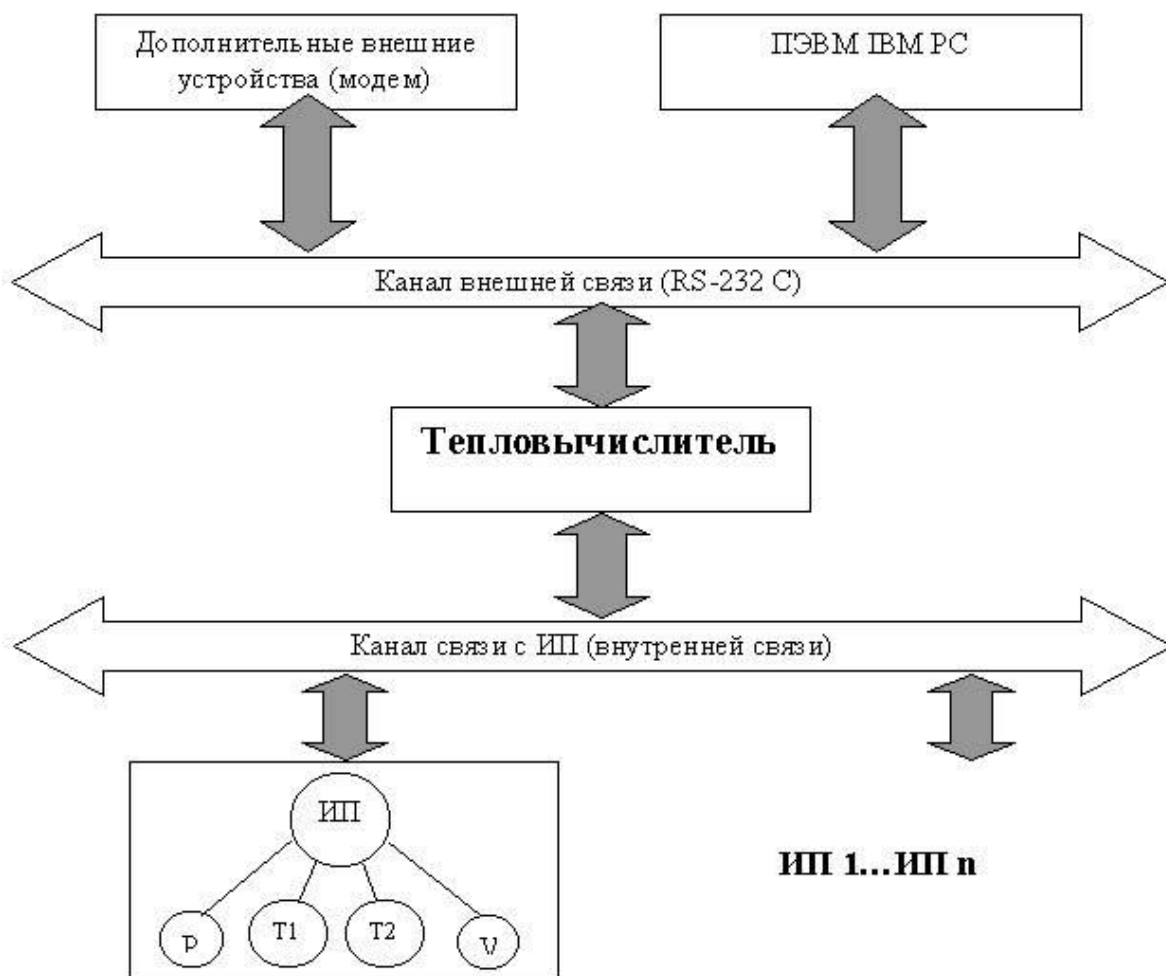


Рисунок 3.2 – Структурная схема подключения измерительных преобразователей (ИП) к тепловычислителю

Одним из основных направлений решения задачи повышения качества и экономичности теплоснабжения является автоматизация регулирования расхода тепла при работе тепловых пунктов.

Через тепловой пункт регулируется работоспособность теплового оборудования, контролируется режим потребления тепла, распределяется теплоноситель по типам потребления (системы отопления, горячего водоснабжения, вентиляции и кондиционирования), настраиваются и изменяются параметры теплоносителя.

Основными функциями теплового пункта являются [73]:

- контроль и оптимизация параметров теплоносителя;

- преобразование вида теплоносителя;
- защита систем теплоснабжения, снижение риска создания аварийной ситуации;
- распределение теплоносителя по системам отопления, водоснабжения и вентиляции;
- включение и выключение систем теплоснабжения;
- контроль над расходом теплоносителя и тепла, а также обеспечение необходимого расхода теплоносителя (на эту характеристику влияют теплопотери и особенности объекта в сочетании с заданными параметрами);
- сокращение теплопотерь.

В связи с развитием жилищно-коммунального хозяйства возникают предпосылки для перехода на оплату каждым владельцем или съемщиком жилья фактически потребленной им тепловой энергии. Цены на тепло, предоставляемое предприятиями теплосетей, являются ориентировочными по максимально возможному потреблению. Для решения этой проблемы нужны объективные методы контроля количества потребленного тепла. Таким образом, для экономии энергетических ресурсов и, в то же время, обеспечения комфортных условий в помещениях необходима система, совмещающая в себе функции измерения фактически потребленной тепловой энергии и регулирования процесса использования тепла [36, с. 104]. На основании вышесказанного можно сделать вывод, что необходима интеграция данных, которая включает объединение данных, находящихся в различных источниках, и предоставление данных пользователям в унифицированном виде. Этот процесс становится существенным как в коммерческих задачах (когда двум похожим компаниям необходимо объединить их базы данных), так и в научных (комбинирование результатов исследования из различных биоинформационных репозиториях, для примера). Роль интеграции данных возрастает, когда увеличивается объём и необходимость совместного использования данных.

3.2 Автоматизация системы конвейерного управления тепловыми пунктами

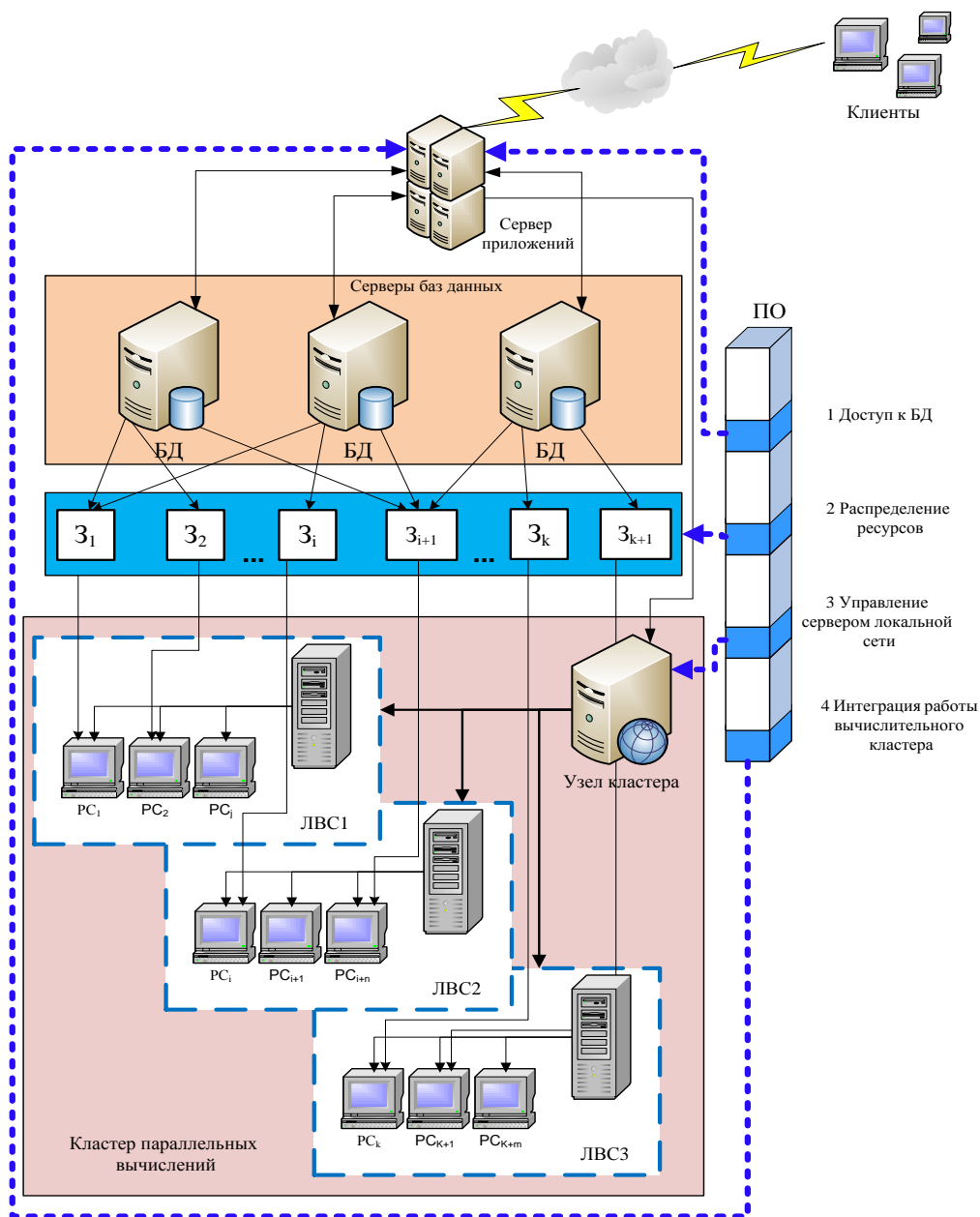
Для решения задач интеграции информационных ресурсов тепловых сетей, в отличие от известных подходов [74], где данные из различных баз данных конвертируются в одну базу данных, предлагается параллельная работа с различными базами данных.

Здесь архитектура интегрированной информационной системы представляет собой модель, которая объединяет элементы «клиент-серверной» архитектуры и использование вычислительного кластера с параллельной обработкой распределенной информации (рисунок 3.4) [75-78].

Разработанная архитектура информационной системы имеет 4 уровня.

Первый уровень – «клиент-серверная» основа информационной системы, осуществляющая управление параллельной работой кластера вычислительных машин и взаимодействие с клиентскими (пользовательскими) станциями [77, с. 154].

Второй уровень – это уровень работы с данными. Ключевыми элементами данного уровня являются серверы баз данных, осуществляющие взаимодействие с базами данных, имеющими различные форматы данных или различные платформы, основанные на различных механизмах хранения данных, таких как реляционные базы данных и объектно-ориентированные базы данных [77, с. 154].



Архитектура ИС с параллельной обработкой данных на базе вычислительного кластера

БД – база данных;
 Z_i – i -тая задача;
 PC – рабочая станция;
 ЛВС – локальная вычислительная сеть;
 ПО – программное обеспечение

Рисунок 3.4 – Архитектура интегрированной ИС

Третий уровень – это уровень решаемых системой информационных задач, который устанавливает алгоритм решения и выделяет используемые в информационной системе записи из имеющихся на втором уровне БД [77, с. 154].

Четвертый уровень отвечает за реализацию параллельной обработки данных. Он основан на компьютерном кластере, позволяющем осуществлять параллельные вычисления с координацией сервером узла кластера. Сервер узла кластера выполняет диспетчерские функции по распределению задач между рабочими станциями кластера в соответствии с указаниями сервера-приложений ИС [77, с. 154].

Последние годы во всем мире происходит бурное внедрение вычислительных кластеров – локальных сетей, с узлами из рабочих станций или персональных компьютеров специально собранных для использования в качестве многопроцессорной вычислительной системы (суперкомпьютера) [77, с. 154].

Мировой опыт создания вычислительных кластеров представлен немалым количеством примеров от скромных 20-30 узловых кластеров в учебных или научных лабораториях университетов, до гигантских вычислительных систем состоящих из 1000 – 2000 рабочих станций, созданных в рамках специальных проектов [77, с. 154].

Для построения вычислительных кластеров обычно используются общедоступные компьютеры на базе процессоров Intel или AMD, стандартные сетевые технологии Ethernet или Fast Ethernet, распространенные операционные системы Windows и коммуникационная библиотека MPI [79], реализующая связь между ветвями параллельного вычислительного процесса. Таким образом, сегодня вычислительные кластеры стали общедоступной и сравнительно дешевой альтернативой традиционных суперкомпьютерам. На многих классах задач и при достаточно большом числе узлов кластерные системы достигают производительности сравнимой с суперкомпьютерной [50, с. 40]. Схематично взаимодействие программной реализации информационной системы с аппаратной составляющей показана также на рисунке 3.4. Модуль информационной системы для взаимодействия с базами данных представлен на рисунке 3.5.

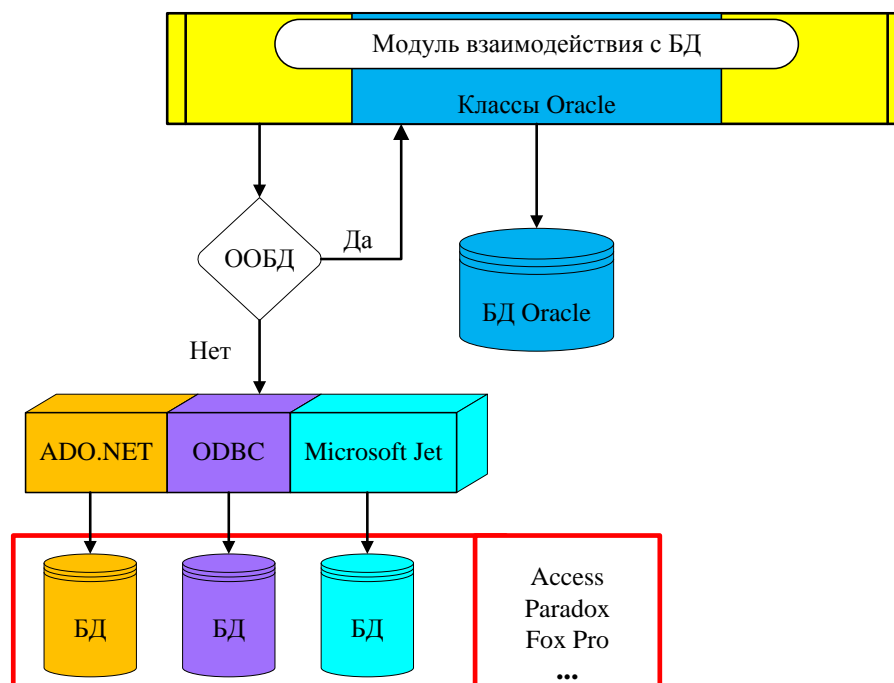


Рисунок 3.5 – Схема модуля взаимодействия с базами данных

Модуль связывается с программной платформой NET Framework. NET Framework дает нам множество способов для работы с базами данных. Платформа .NET Framework имеет собственную технологию доступа к данным - ADO.NET (ActiveX Data Object для .NET). ADO.NET включает в себя управляемые классы, позволяющие приложениям .NET подключаться к базам данных, а также выполнять работу с данными и управлять автономными данными. ADO.NET технология позволяет работать с данными в Microsoft SQL Server, Microsoft Access, Microsoft Excel, Microsoft Outlook, Microsoft Exchange, Oracle, OLE DB, ODBC, XML в автономном режиме с помощью объектов DataSet [80]. Объекты DataSet позволяют нам извлекать копии связанных между собой локальных таблиц данных из MS Access. После этого модуль работает с содержимым DataSet, при этом не требуется подключение к источнику данных, а также позволяет отправлять обратно измененные данные для обработки с помощью соответствующего адаптера данных. Схема взаимодействия с базой данных через ADO.NET представлена на рисунке 3.6.

Для работы с базами данных в том числе с Paradox в модуле реализован открытый интерфейс доступа к базам данных ODBC (Open DataBase Connectivity). ODBC позволяет нашему модулю взаимодействовать с различными базами данных и при этом нет необходимости беспокоиться о тонкостях взаимодействия с несколькими источниками.

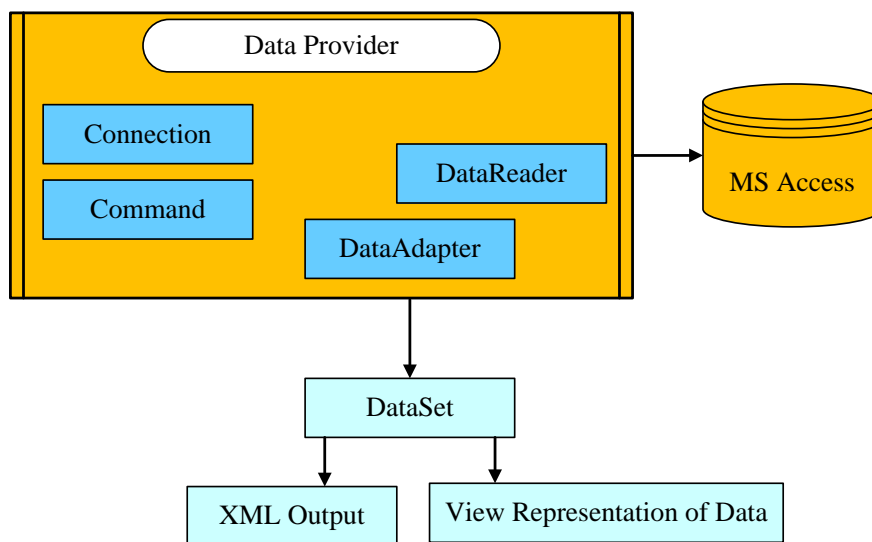


Рисунок 3.6 – Схема взаимодействия с базой данных через ADO.NET

Работа с базой данных через ODBC API осуществляется через установку соединения с базой данных. Схема доступа модуля к базе данных Paradox через интерфейс ODBC приведена на рисунке 3.7. В системе Microsoft Driver Manager (odbc32.dll) взаимодействует с odbcint.ini и odbc.ini. Для работы Microsoft Driver Manager, который позволяет подключать драйверы, используется ODBC Administrator (odbcad32.exe, odbccp32.dll и odbccp32.cpl). Запуская ODBC Administrator мы можно устанавливать имя базы, подключать драйверы, редактировать данные и т.д. Для работы с базой данных в системе есть соответствующий драйвер. Затем выполняется запрос и, после получения данных, соединение закрывается.

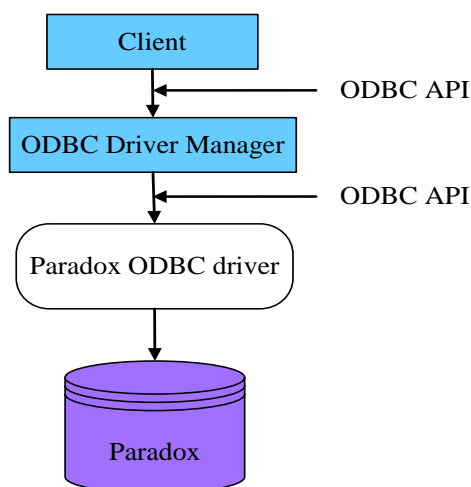


Рисунок 3.7 – Схема доступа к базе данных Visual FoxPro через интерфейс ODBC

Для работы с базами данных Visual FoxPro в модуле используется технология Microsoft JET Engine. Один из трех модулей Microsoft Jet Database Engine содержит драйверы ISAM, библиотеки DLL, которые дают возможность доступа к базам данных ISAM [81], среди которых в том числе и Visual FoxPro. Еще один из модулей DAO предоставляет API. API дает возможность доступа к базам данных JET с использованием произвольного языка программирования, что и используется нами в модуле информационной системы для взаимодействия с базами данных Visual FoxPro.

Часть модуля информационной системы для взаимодействия с базой данных Oracle разработана в интегрированной среде разработки приложений баз данных PowerBuilder. PowerBuilder был выбран, так как он использует "родные" интерфейсы для подключения к Oracle и патентованную технологию работы с данными - DataWindow. Технология PBNI в PowerBuilder устраняет недостаток значительного времени компиляции для анализа.

На заключительном этапе работы программа, используя коммуникационную библиотеку MPI, передает управление в соответствии с распределенными задачами рабочим станциям кластера. Рабочие станции самостоятельно осуществляют обработку данных из соответствующего хранилища и производят вычисления, необходимые для решения соответствующей данной рабочей станции задачи (рисунок 3.9). При этом основное приложение осуществляет с заданной периодичностью опрос всех рабочих тепловых пунктов, участвующих в решении поставленной задачи. Опрос производится циклически путем перебора всех рабочих тепловых пунктов, и в случае решения всех задач происходит запуск последнего модуля, интегрирующего полученные результаты и обеспечивающего документацию по проектам строительства, ремонта или реконструкции тепловых сетей [77, с. 155].

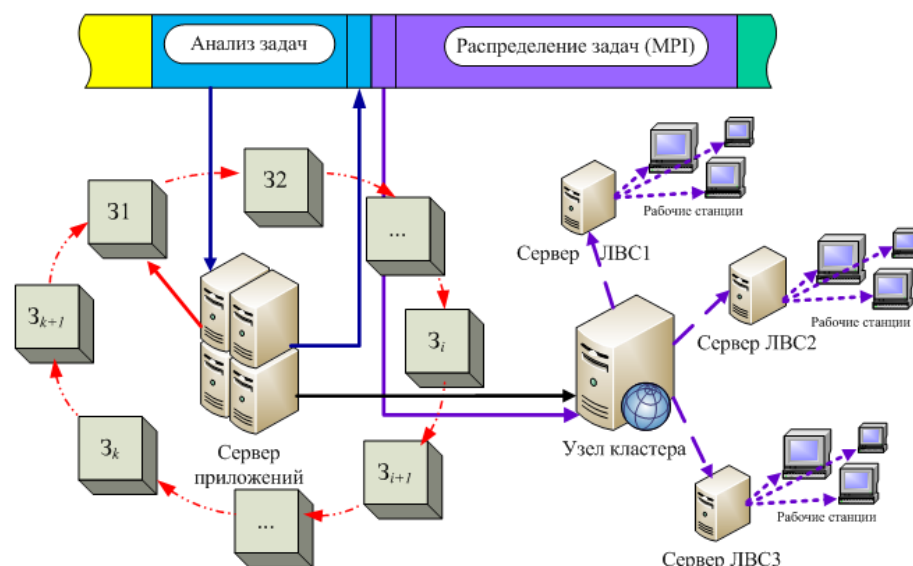


Рисунок 3.8 – Схема распределения ресурсов информационной системы

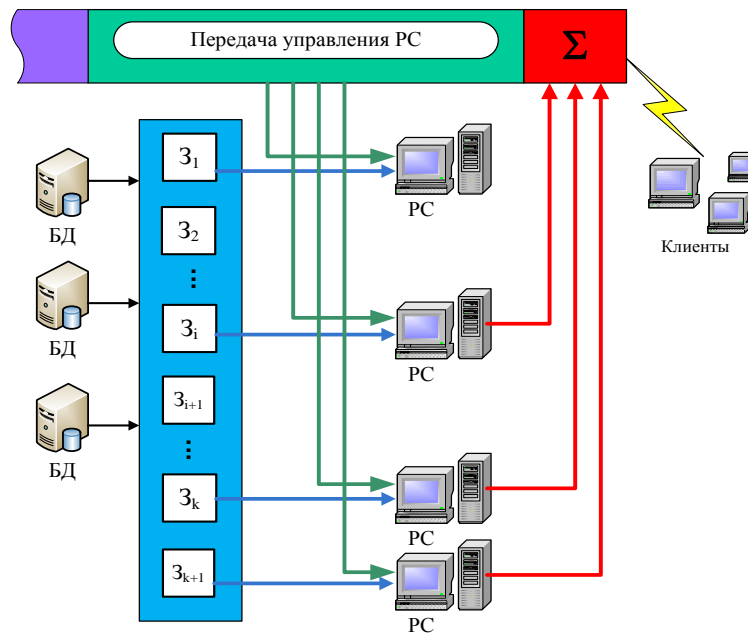


Рисунок 3.9 – Схема управления ИС рабочими станциями вычислительного кластера

Клиентские приложения, выполняющие функции автоматизации работы подразделений органа управления тепловыми сетями используют полученные на сервере приложения готовые результаты для формирования документации по работе тепловых сетей. В нашем случае, это позволит оперативно обрабатывать данные по тепловым пунктам тепловых сетей, хранящиеся в разных базах данных, и при этом управлять объектами с удаленным доступом [77, с. 155].

На основе вышеизложенного подхода в диссертационной работе была разработана информационная модель обслуживания тепловых пунктов, которая была реализована в созданном программном обеспечении.

Выводы по разделу

1 Представлены технические решения по модернизации работы тепловых пунктов на основе современных систем автоматизации

2 Разработана и описана архитектура, представляющая собой гибридную модель, интегрированной информационной системы. Она объединяет элементы «клиент-серверной» архитектуры и использование вычислительного кластера с параллельной обработкой распределенной информации.

3 Предложена гибридная ИС, которая позволяет оперативно решать задачи управления ТП с использованием кластера компьютеров, созданного на базе имеющегося технического обеспечения теплогенерирующих и теплообеспечивающих компаний.

4 ИССЛЕДОВАНИЕ КОНВЕЙЕРНОГО РЕЖИМА РАБОТЫ ТЕПЛОПУНКТОВ

4.1 Разработка структуры программно-технического комплекса и используемое программное обеспечение в системах теплоснабжения

Применение программно-технического комплекса (ПТК) служит для реализации функций автоматизированного контроля и управления технологическими объектами и сооружениями [82].

ПТК включает средства получения информации о значениях технологических параметров (датчики), аппаратуру сбора и передачи информации (устройства телемеханики), средства вычислительной техники (ЭВМ и дополнительные устройства к ним) аппаратуру представления информации оперативному персоналу (видеотерминалы, печатающие устройства и др.), средства связи, исполнительные механизмы [82].

Структура ПТК в первую очередь определяется средствами и характеристиками взаимосвязи отдельных компонентов комплекса (контроллеров, пультов оператора, удаленных блоков ввода-вывода), т.е. сетевыми возможностями. Гибкость и разнообразие структур ПТК зависят от [84]:

- числа имеющихся сетевых уровней;
- возможных типов связи (топологий) на каждом уровне сети: шина, звезда, кольцо;
- параметров сети каждого уровня: типов кабеля, допустимых расстояний, максимального количества узлов (компонентов комплекса), подключаемых к каждой сети, скорости передачи информации, методе доступа компонентов к сети (случайный по времени доставки сообщений или гарантирующий время их доставки).

Сбор данных и управление оборудованием может осуществляться по выделенным телефонным линиям, GSM и радиоканалам, Ethernet, GPRS, посредством модемного соединения. При выборе канала связи предпочтение отдается GSM-технологии по следующим причинам:

- независимость от рельефа местности (возможность связи с объектами, не находящимися в зоне прямой видимости);
- доступность оборудования по цене и простоте установки;
- возможность подключения к автоматизированной системе оперативного диспетчерского управления (АСОДУ) большого числа абонентов;
- отсутствие внешней антенны и ретранслятора, компактность.

Обмен данными по GSM-сети может быть реализован с помощью прямого соединения или по GPRS каналу.

Вариант структурной схемы ПТК по учету и контролю тепловой энергии будет иметь вид, представленный на рисунке 4.1.

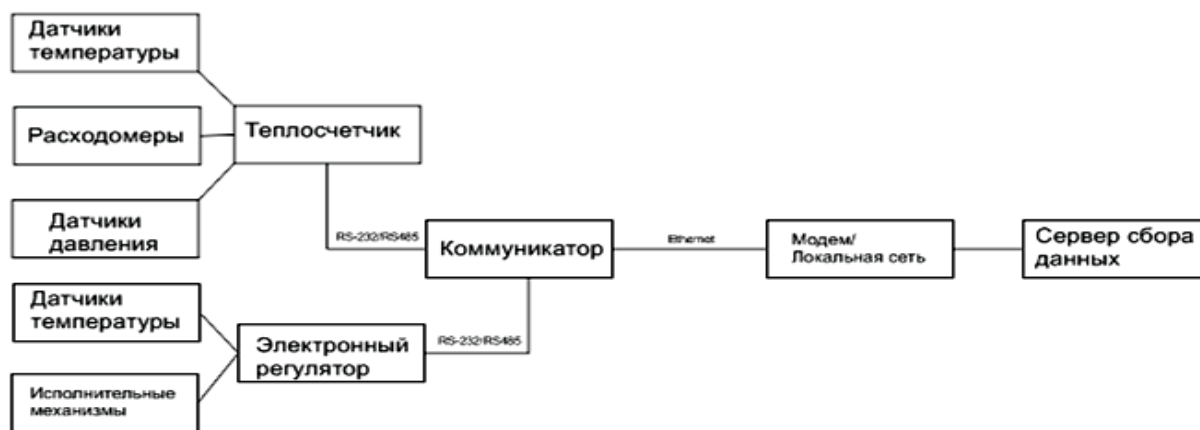


Рисунок 4.1 – Структурная схема ПТК по учету и контролю тепловой энергии

Данная структура полностью соответствует типовой структуре получения-обработки информации в ПТК (рисунок 4.2), которую в ходе диссертационной работы усовершенствуем с использованием принципов конвейерной обработки данных.

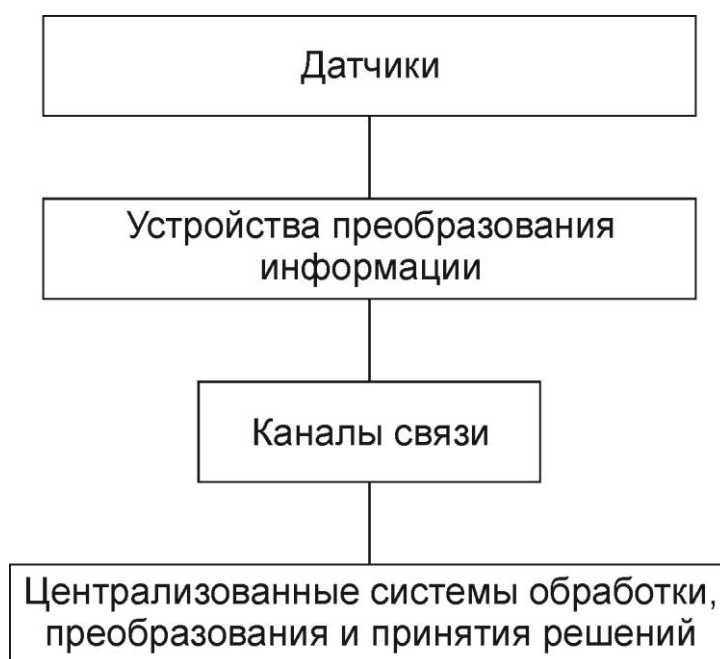


Рисунок 4.2 – Типовая структура получения-обработки информации в ПТК

Указанные выше свойства ПТК характеризуют возможность распределения аппаратуры в производственных цехах; объем производства, который может быть охвачен системой автоматизации, реализованного на данном ПТК; возможность переноса блоков ввода-вывода непосредственно к датчикам и исполнительным механизмам.

Рассмотрим программное обеспечение, созданное коллективом отечественных разработчиков ПКФ «Сириус» (г. Караганда, РК) -

Информационно-графический программный комплекс (ИГС) ТГИД-07 (рисунок 4.3).

Назначение ТГИД-07 - моделирование и пространственно-технологический анализ установившихся теплогидравлических режимов централизованных систем теплоснабжения, с целью управления режимами эксплуатации и комплексного развития сложных теплоснабжающих систем, с возможностью контролинга рассчитанных и измеренных параметров работы системы.

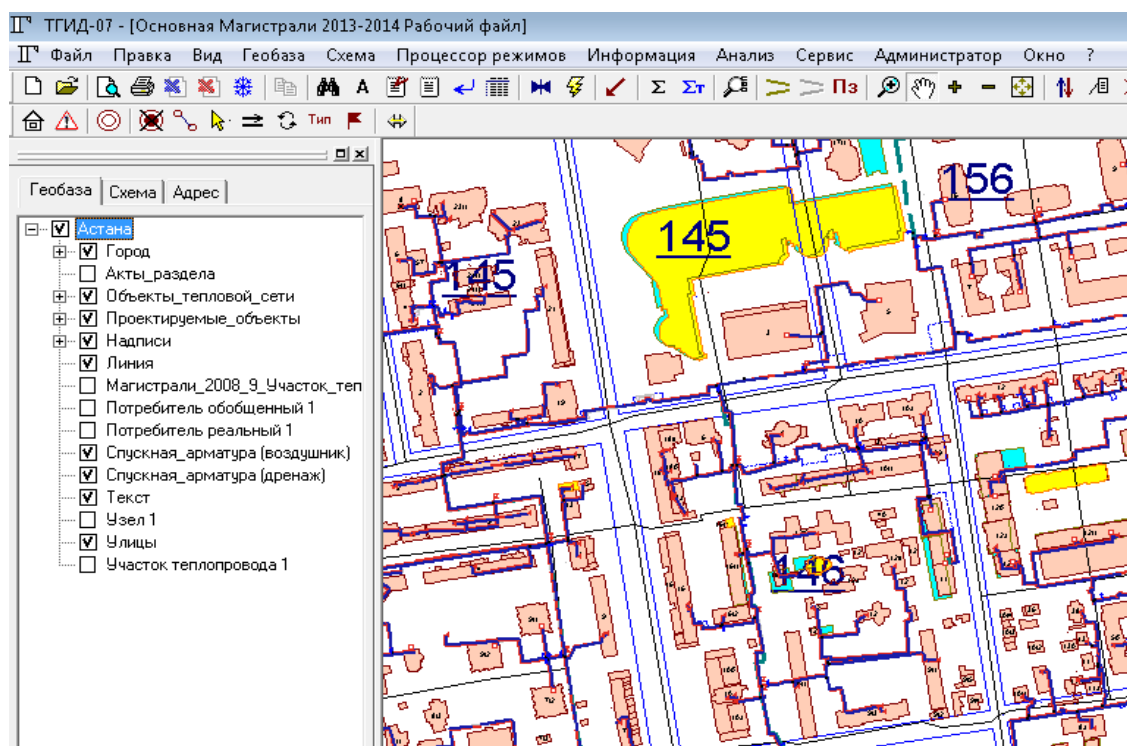


Рисунок 4.3 – Главное окно ТГИД-07

Область применения - создание автоматизированных рабочих мест в центральных и районных диспетчерских службах, режимных службах, производственно-технических отделах энергетических компаний с совместной выработкой электрической и тепловой энергии, с возможностью интеграции с ГИС, со SCADA-системами, с автоматизированными системами учета энергоресурсов, с автоматизированными системами управления производством.

Функциональные возможности ТГИД-07 обеспечивают:

- организацию бесперебойного теплоснабжения потребителей города;
- снижение числа аварийных ситуаций и инцидентов в работе систем теплообеспечения города;
- оперативное управление системами теплоснабжения и энергообеспечения объектов городской инфраструктуры;
- оптимизацию теплопотребления промышленными и жилыми объектами города;

- возможность планирования обслуживания и ремонтов оборудования по его фактическому техническому состоянию;

- регулирование режимов работы и противоаварийной защиты оборудования всех объектов теплообеспечения на базе использования современных средств автоматизации и распределенных управляющих программно-технических комплексов с высокой эксплуатационной надежностью.

Минимальная программно-аппаратная платформа для работы ТГИД-07 включает в себя:

- пульта операторов для отдельных автоматизированных рабочих мест технологических служб на базе типовых ПК офисного исполнения (операционная система Windows) с установленной сетевой версией ИГС ТГИД-07;

- сетевую аппаратуру (сетевой коммутатор, шлюз или мост для связи с корпоративной сетью энергетического комплекса);

- центральный концентратор измеренной информации – сервер БД на базе Windows Server 2008 с установленным MS SQL поддерживающий функции диспетчерского управления. Основное программное обеспечение сервера БД может быть представлено SCADA-программой, реализующей основные функции визуализации измеряемой и контролируемой информации, передачи данных и команд системе контроля и управления.

При внедрении ИГС ТГИД-07 на теплоснабжающие и теплообеспечивающие предприятия и компании города или района происходит инсталляция автоматизированных рабочих мест (АРМ) в центральных и районных диспетчерских службах, в режимных службах и в производственно-технических отделах (рисунок 4.4).

В состав ТГИД-07 включены следующие АРМ:

- АРМ службы режимов. Назначение – синтез и наладка сезонных и перспективных теплогидравлических режимов теплоснабжения в условиях нормальной эксплуатации и аварийных ситуаций;

- АРМ оперативно-диспетчерской службы. Назначение – мониторинг и оперативное управление режимом эксплуатации теплоснабжения, синтез гидравлических режимов теплоснабжения в условиях аварийных ситуаций;

- АРМ службы наладки и учета потребителей. Назначение – учет и подготовка абонентов к отопительному сезону;

- АРМ производственного отдела. Назначение – анализ состояния и планирование ремонта оборудования;

- АРМ технического отдела. Назначение – оформление и ведение документации на подключение новых потребителей.

На рисунке 4.5 приведено детальное разложение работы АРМ службы режимов теплоснабжающего и/или теплообеспечивающего предприятия или компании города. Одним из основных компонентов для расчета теплогидравлического режима функционирования системы теплоснабжения

города является информация с ТП, а также с тепловых пунктов, оснащенных авторегуляторами отопления и ГВС.

Исходя из вышесказанного, была создана оригинальная программа для ЭВМ, которая позволяет производить управление системой объектов тепловых сетей, проводить вычисления, расчет и выполнять другие действия (Приложение А).

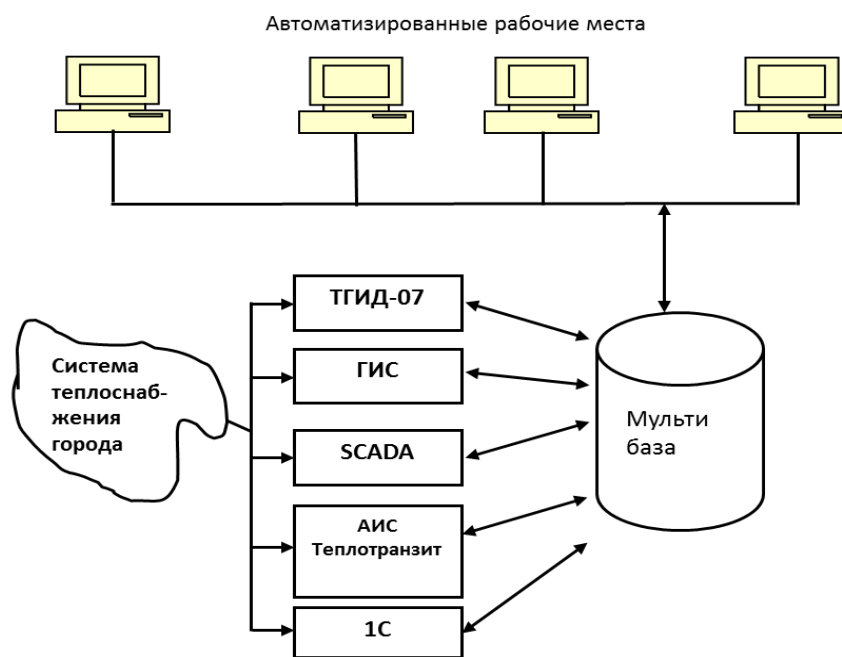


Рисунок 4.4 – Концептуальная архитектура ТГИД-07

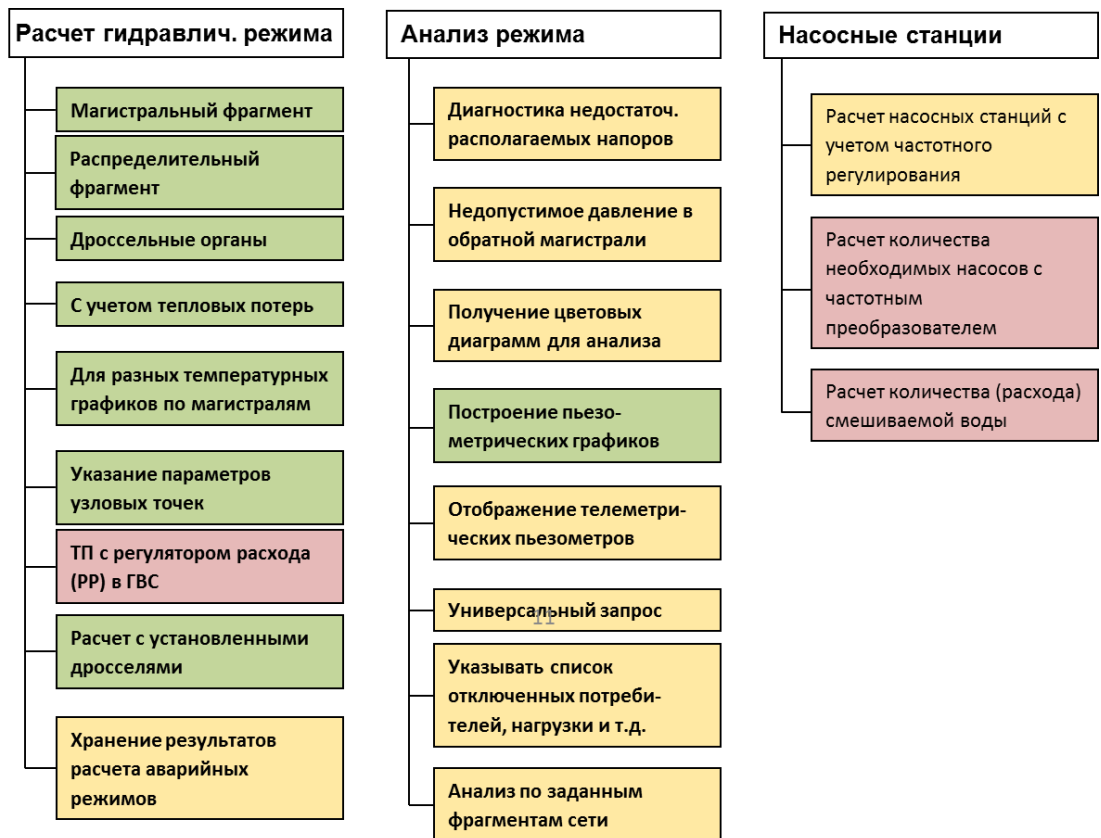


Рисунок 4.5 – Детальное разложение работы АРМ службы режимов

При этом была использована методика, основанная на информационной модели системы управления тепловыми пунктами с использованием конвейерной обработки данных.

4.2 Программа для создания и редактирования схем теплоснабжения тепловых пунктов с возможностью считывания реальных данных

Программа предназначена для создания и редактирования схем тепловых пунктов теплоснабжения. Содержит ряд как стационарных элементов (участок трубопровода, батарея, тройник, угол и т.п.), так и ряд динамических (узел смешения, насос, кран, вентиль и т.п.). Позволяет включать в схему размещения оборудования теплового пункта отображающие приборы учета параметров теплоснабжения (манометр, термометр, расходомер, теплосчетчик), значения которых программно можно связать с архивной базой измеренных значений соответствующих приборов, а также получать текущие значения в реальном времени. Содержит библиотеку различных схем тепловых пунктов теплоснабжения (зависимое, независимое, зависимое элеваторное, с ГВС, с рециркуляцией и т.д.). Может функционировать как независимое приложение. Также существует возможность включение программы в качестве модуля в составе другой программы-оболочки, например, в составе диспетчерской геоинформационной программы, считывающей показания с приборов учета теплоснабжения на карте города [86].

Функциональные возможности:

- создание и редактирование схемы теплового пункта теплоснабжения;
- сохранение схемы теплового пункта теплоснабжения в библиотеке схем;
- отображение на схеме значений показаний приборов учета теплоснабжения;
- кодирование приборов учета теплоснабжения на сохраненных схемах с созданием базы приборов для интеграции со сторонней программной оболочкой.
- работа в качестве независимого приложения для изучения вариантов схем теплоснабжения тепловых пунктов.

Основные технические характеристики: количество файлов для работы программы – один. Объем программы для ЭВМ (компилированного кода программы (выполняемого .exe файла)) 424 кБайт. Может быть дополнено библиотекой разработанных схем: бинарными файлами с расширением CON. Количество и размер файлов зависит от полноты создания и наполнения библиотеки.

Язык программирования программы: Microsoft Visual C++ 6.0.

Тип реализующей ЭВМ: персональный компьютер с архитектурой x86, x64. Данная программа для ЭВМ предназначена для работы только в семействе операционных систем Microsoft Windows 32/64 версии не ниже Windows XP Service Pack 3.

Минимальные аппаратные средства компьютера для работы данной программы для ЭВМ:

- процессор не ниже Intel Pentium I 100 МГц;
- объем оперативной памяти не менее 256 Мб;
- объем свободного пространства жесткого диска не менее 1 Мб;
- манипулятор типа мышь, клавиатура.
- графический монитор с рабочей областью экрана не менее 800x600 точек и цветовая палитрой High Color не менее 16 бит.

Программа может распространяться копированием компилированного кода программы (выполняемого .exe файла «Constructor.exe») и имеющейся библиотекой бинарных файлов с расширением CON.

Для запуска программы необходимо запустить файл «Constructor.exe» (находится в папке «Конструктор\Release»), после чего на экране появляется окно программы с пустой рабочей областью для создания схемы (рисунок 4.6).

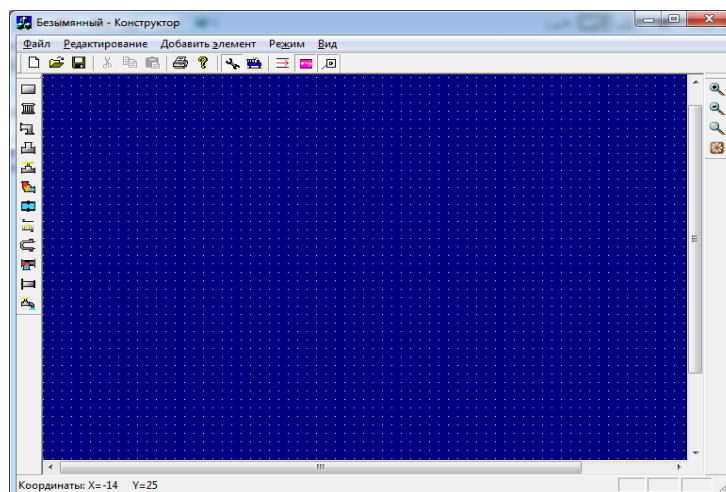


Рисунок 4.6 – Основное окно программы «Конструктор»

Справа расположены кнопки с примитивами элементов для создания схемы. Кнопки продублированы в пункте меню «Добавить элемент» (рисунок 4.7).

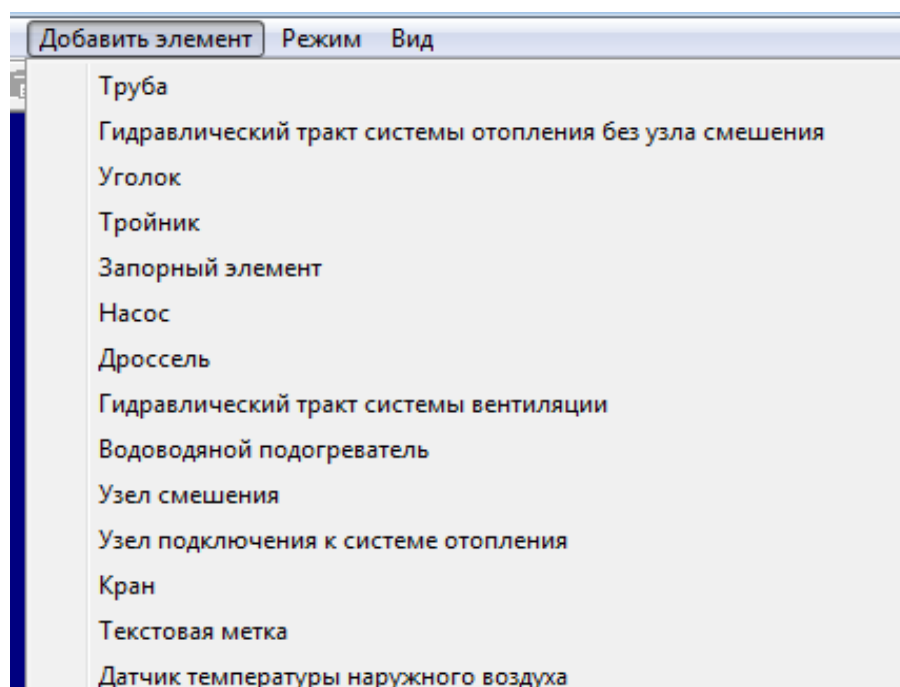


Рисунок 4.7 – Команды меню «Добавить элемент» программы «Конструктор»

Меню «Файл» содержит типовой набор команд (Открыть схему/Сохранить схему/Сохранить как... и т.п.), а также список последних открытых схем (рисунок 4.8).

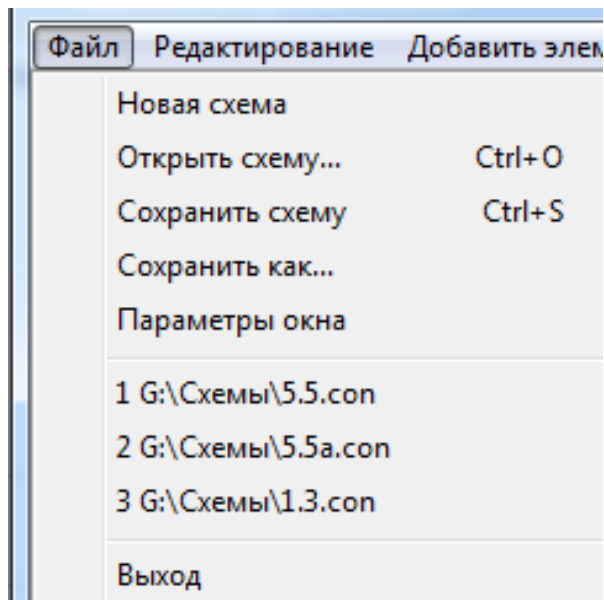


Рисунок 4.8 – Команды меню «Файл» программы «Конструктор»

Пример отрисованной схемы теплового пункта по зависимой схеме отопления с безэлеваторным типом подключения с вентиляцией (рисунок 4.9).

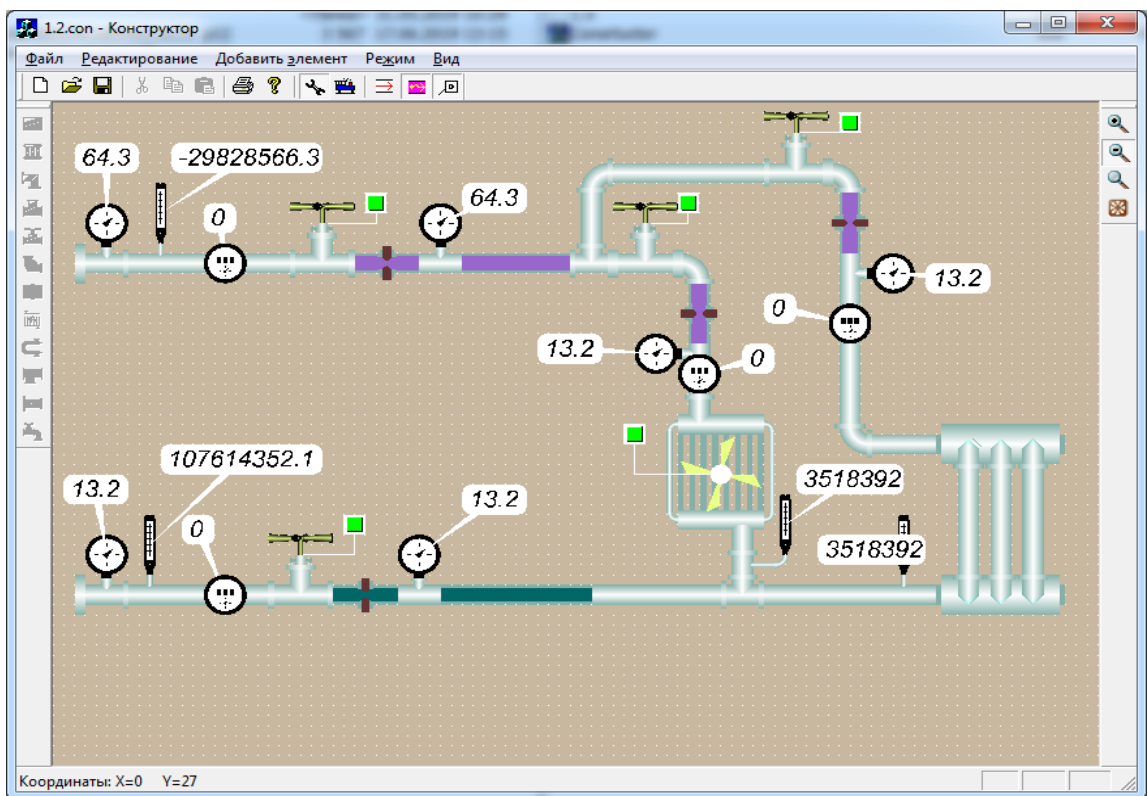


Рисунок 4.9 – Пример отрисованной схемы теплового пункта в программе «Конструктор»

Существует возможность включить режим отображения показаний значений приборов учета тепловой энергии (меню «Режим» пункт «Схема\Показывать названия приборов») (рисунок 4.10).

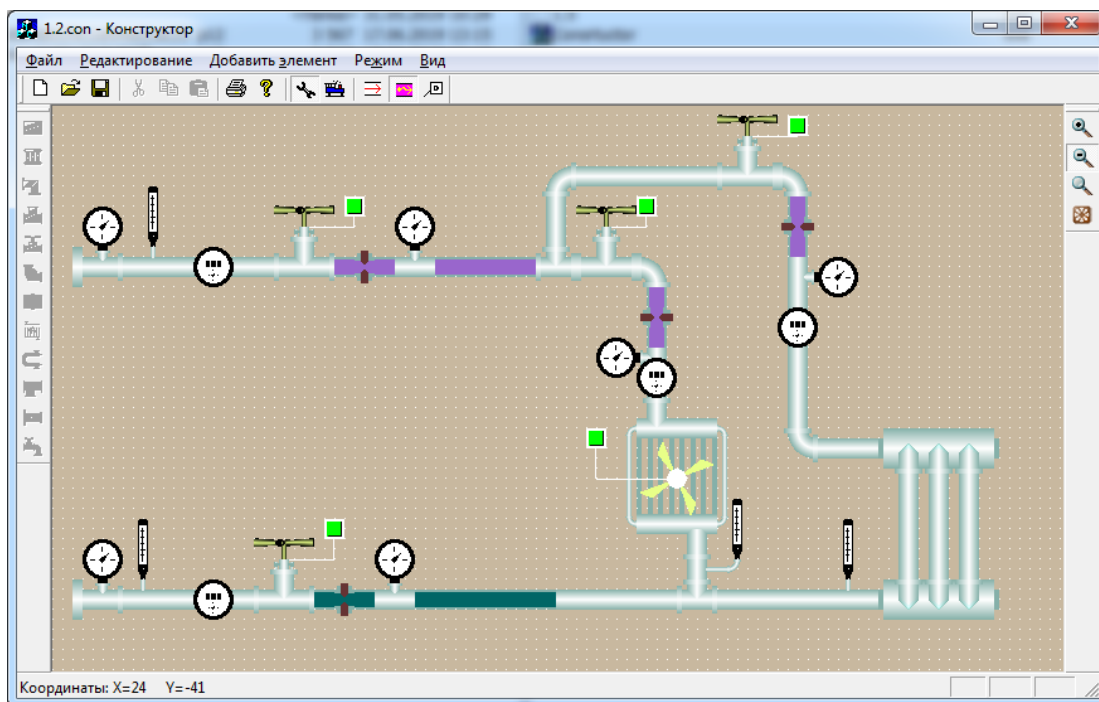


Рисунок 4.10 – Режим отображения показаний значений приборов учета

Каждый элемент схемы, включая приборы учета, имеет свои свойства (параметры построения). Окно свойств вызывается выбором строки «Свойства...» из контекстного меню элемента (рисунок 4.11).

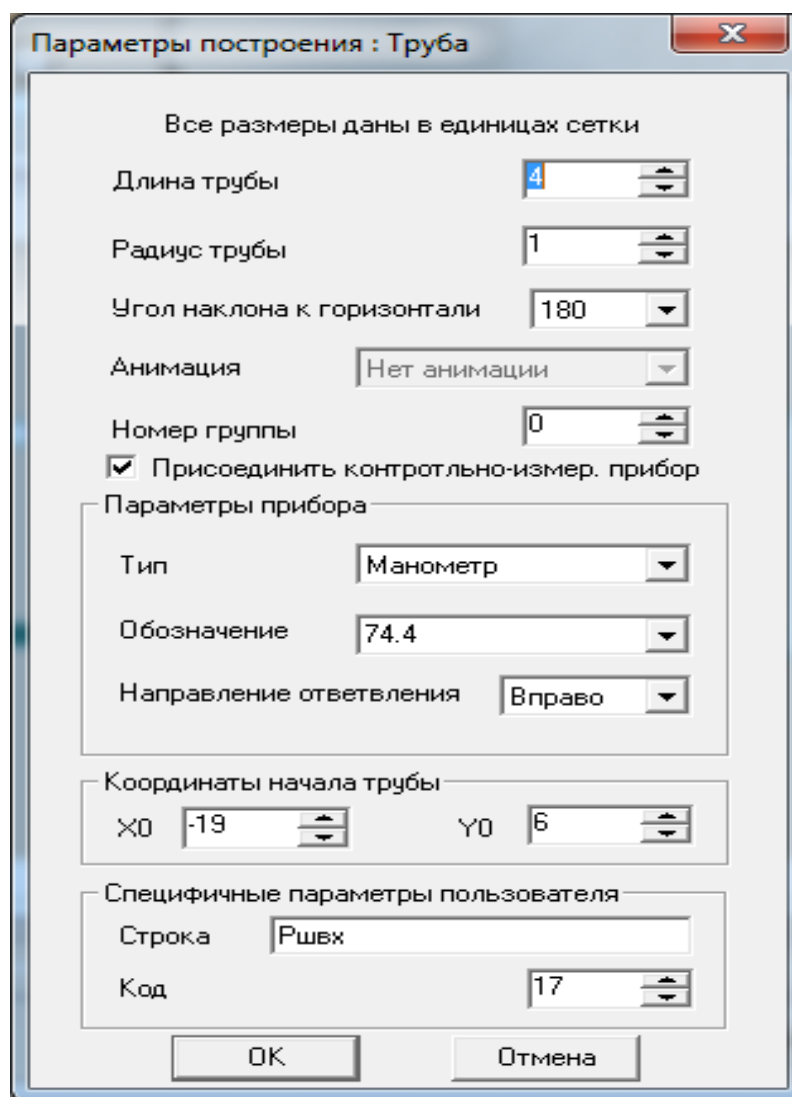


Рисунок 4.11 – Свойства объекта

В разделе «Параметры прибора» (см. рисунок 4.11) имеется возможность выбора типа прибора учета и варианта отображения показаний прибора.

Здесь же в свойствах объектов (параметры построения) имеется возможность указать специфические параметры пользователя для интеграции схемы теплового пункта со сторонней программной-оболочкой.

Результатами работы программы являются:

- созданная схема теплового пункта теплоснабжения;
- набор бинарных файлов (библиотека схем тепловых пунктов);
- отображение на схеме значений показаний приборов учета теплоснабжения;
- кодирование приборов учета теплоснабжения на сохраненных схемах с созданием базы приборов для интеграции со сторонней программной-оболочкой;
- работа в качестве независимого приложения для изучений вариантов схем теплоснабжения тепловых пунктов.

В программе для ЭВМ «Конструктор» реализована идея конвейерной обработки данных. Создаются два потока – первый связан с процессом скачивания данных с прибора учета и перебирает в цикле записанные тепловые узлы (ТУ) и скачивает значения с их в БД, ранее сгенерированную. В результате, имитируется процесс скачивания данных с приборов – сервер поочередно опрашивает каждый узел и считывает данные. После опроса каждого узла происходит задержка по времени, необходимая для корректной записи и обработки данных. Время задержки настраивается в меню «Параметры соединения - Настройки». Там же можно настроить, сколько максимальной число записей в таблице хранения данных. После достижения максимума запись будет производиться в «стек». Запуск и остановка потока происходит при активизации с панели меню. Чем больше узлов, по которым происходит считывание данных, тем медленнее идет процесс обновления для каждого узла.

Второй поток отвечает за обновление параметров выбранного на карте узла. Он запускается автоматически при запуске программы. При выделении на карте ТУ данный поток в непрерывном режиме обращается к БД и скачивает последние записи по данному ТУ и обновляет в соответствующем поле.

Созданная программа для ЭВМ «Конструктор» была интегрирована в ИГС ТГИД-07, а именно поставляет данные в сложные объекты БД «Расчетная схема»: ненагруженный узел (камера, павильон и т.п.), КРТП (контрольно-распределительный тепловой пункт), насосная станция, источник тепла, потребитель реальный.

Входная форма «Реальный потребитель» несет в себе следующие данные и функциональность (рисунок 4.12). Анимационная технологическая схема теплового пункта потребителя, в определенные точки которой можно выводить как результаты измерений, так и расчета параметров режима. Информационная схема объекта представлена в девяти вкладках (Основные, Нагрузки, Дроссели, Оборудование, Здание, Архив, Расчет оборудования, Результат расчета), в каждой из которых можно изменять значение любого из свойств, на основе встроенных списков, функций или из любых источников легитимных данных. Во вкладке «Результат расчета» можно обозревать рассчитанные параметры фактического теплогидравлического режима данного потребителя.

В результате интеграции программы для ЭВМ «Конструктор» в ИГС ТГИД-07 типовую структуру получения-обработки информации в ПТК (рисунок 4.2) можно трансформировать в следующую (рисунок 4.13). Здесь программа для ЭВМ «Конструктор», реализующая принципы конвейерной обработки, вошла в состав верхнего уровня управления. Программный код «Конструктор» приведен в приложении А.

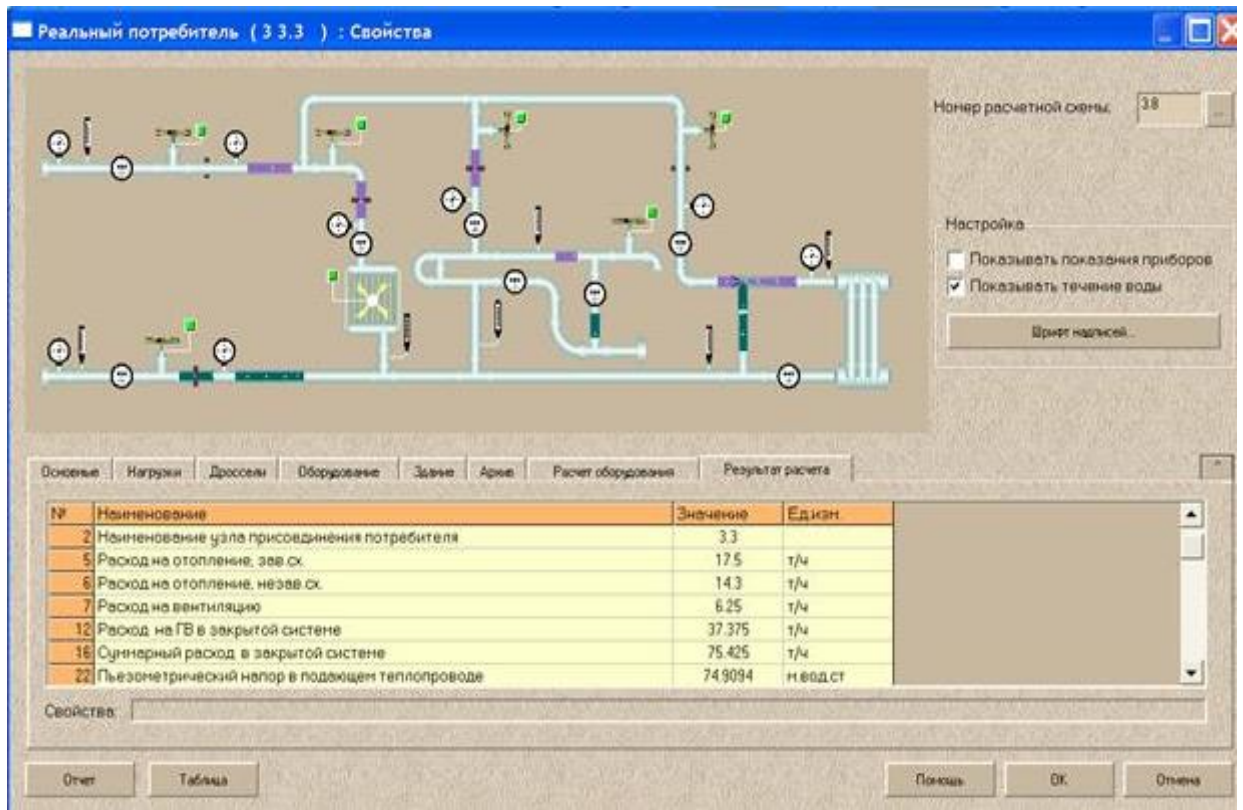


Рисунок 4.12 – Пример отображения данных результатов расчета теплогидравлического режима для объекта расчетной схемы тепловой сети "Реальный потребитель"



Рисунок 4.13 – Структурная схема получения-обработки информации в ПТК с конвейерной обработкой данных на верхнем уровне

4.3 Экспериментальные исследования эффективности применения автоматизации тепловых пунктов с использованием конвейерной обработки информации

Для практической апробации расчетов в среде ТГИД-07 с установленным программным модулем «Конструктор» были выбраны 2 тепловые пункта в г. Караганда. Это тепловые пункты в жилом доме по адресу г. Караганда, мкр. "Аэропорт городской", уч.79 и в Драматическом театре им. Станиславского по адресу г. Караганда, пр. Нурсултана Назарбаева, 19/1. Внешний вид автоматизированного ТП в жилом доме представлен на рисунке 4.14, а его технологическая схема с табличными данными о тепловых нагрузках и спецификации оборудования приведены на рисунке 4.15 и таблицах 4.1 и 4.2.



Рисунок 4.14 – Внешний вид автоматизированного ТП по адресу г. Караганда, мкр. "Аэропорт городской", уч.79

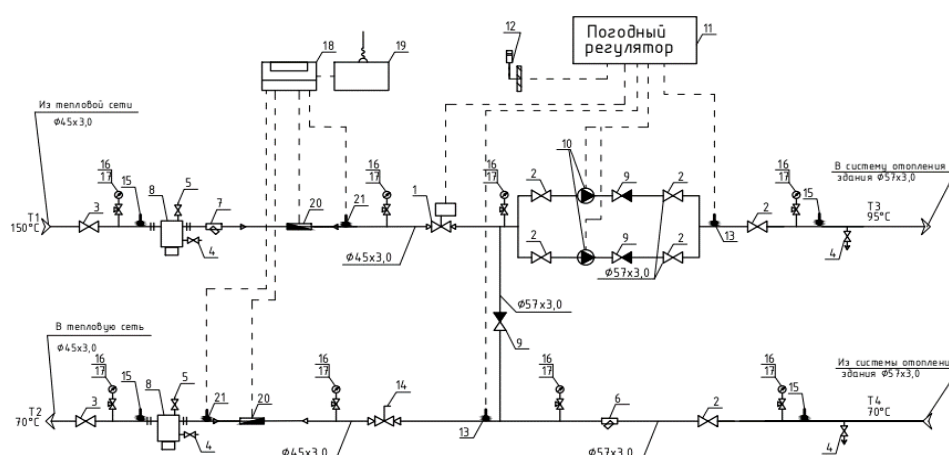


Рисунок 4.15 – Технологическая схема автоматизированного ТП по адресу г. Караганда, мкр. "Аэропорт городской", уч.79

Таблица 4.1 – Таблица тепловых нагрузок на ТП по адресу г. Караганда, мкр. "Аэропорт городской", уч.79

Наименование здания (сооружения, помещения)	Объем м ³	Периоды года при тн, °С	Расход тепла, ккал/час				
			на отопление	на вентиляцию	на горячее водоснабжение		общий
					макс.	средний	
Жилой дом		-32	46700	-	-	-	46700

Таблица 4.2 – Спецификация оборудования ТП по адресу г. Караганда, мкр. "Аэропорт городской", уч.79

Поз.	Обозначение	Наименование	Кол.	Примечание
1	"Honeywell"	Регулирующий клапан V5832B2091 Ду25, Kvs 10 с электроприводом M6410L2031	1	
2	"Also"	Кран шаровой Ду50 под приварку	6	
3	"Also"	Кран шаровой Ду40 под приварку	2	
4	"Genebre"	Кран шаровой Ду25 муфтовый	4	
5	"Genebre"	Кран шаровой Ду 5 муфтовый	2	
6	"Genebre"	Фильтр муфтовый Ду50	1	
7	"Genebre"	Фильтр муфтовый Ду40	1	
8		Грязевик Ду40	2	
9		Клапан обратный Ду50	3	
10	"IMP"	Насос циркуляционный системы отопления GHN 32/80-180 (раб. т. G=2,3 м ³ /ч, H=5м)	2	1 резерв
11	"ТРИТОН"	Погодный регулятор Тритон 001.3	1	
12	"ТРИТОН"	Датчик температуры наружного воздуха RT100	1	
13	"ТРИТОН"	Датчик температуры погружной RT100	2	
14		Клапан балансировочный Ду32	1	
15		Термометр биметаллический ТБ-63-50 (0-160С)	4	
16		Манометр (0-16 кгс/см ²)	8	
17	116186к	Кран трехходовой под манометр	8	
18	"Термотроник"	Тепловычислитель ТВ7-01	1	
19	"Teleofis"	GPRS-модем	1	
20	"Kamstrup"	Расходомер ультразвуковой Ultraflow54 Ду20, Q1,5 м ³ /ч	1	

21	"Термико"	Комплект термопреобразователей сопротивления КТПТР-03	1 ком.
----	-----------	---	-----------

Технологическая схема автоматизированного ТП в Драматическом театре им. Станиславского по адресу г. Караганда, пр. Нурсултана Назарбаева, 19/1 представлена на рисунке 4.16, схема автоматизации – на рисунке 4.17, а его табличные данные о тепловых нагрузках и спецификации оборудования приведены в таблицах 4.3 и 4.4.

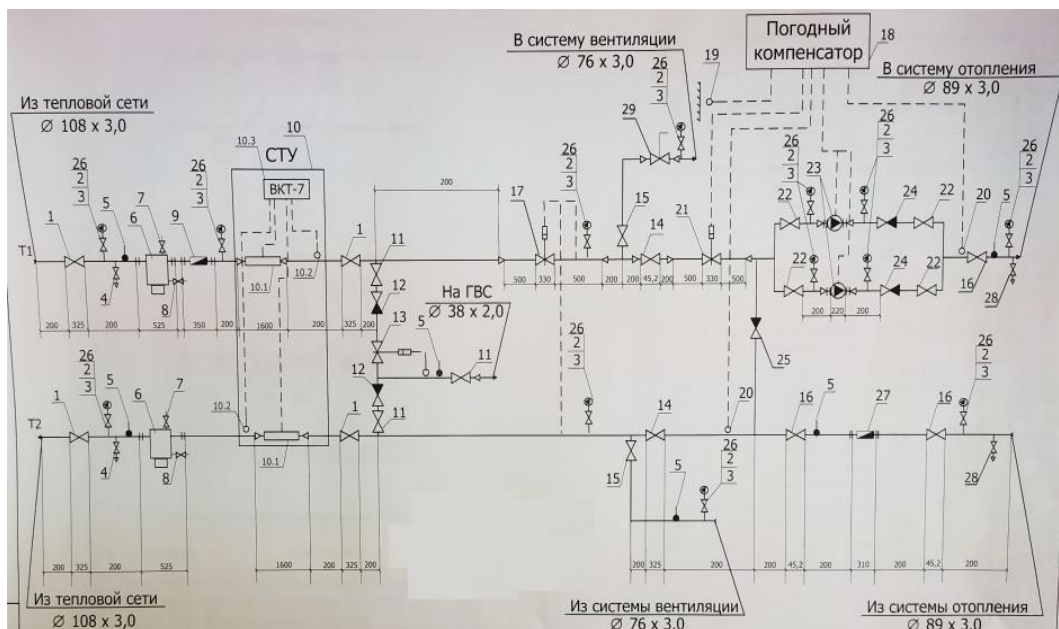


Рисунок 4.16 – Технологическая схема автоматизированного ТП по адресу г. Караганда, пр. Нурсултана Назарбаева, 19/1

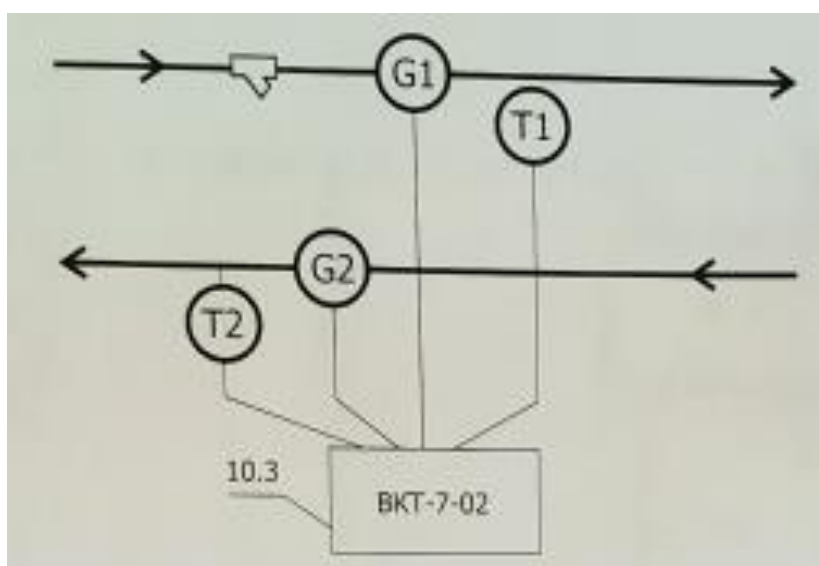


Рисунок 4.17 – Схема автоматизации ТП по адресу г. Караганда, пр. Нурсултана Назарбаева, 19/1

Таблица 4.3 – Таблица тепловых нагрузок на ТП по адресу г. Караганда, пр. Нурсултана Назарбаева, 19/1

Наименование здания (сооружения, помещения)	Объем м ³	Периоды года при тн, °С	Расход тепла, Гкал/час				общий
			на отопление	на вентиляцию	на горячее водоснабжение		
					макс.	средний	
Драматический театр им. Станиславского		-32	0,895	0,588	0,021	-	1,004

Таблица 4.4 – Спецификация оборудования ТП по адресу г. Караганда, пр. Нурсултана Назарбаева, 19/1

Поз.	Обозначение	Наименование	Кол.
1	2	3	4
1	«Vexve» (Финляндия)	Кран шаровый под приварку Тип 100, Ду100	4
2	ГОСТ 2405-88	Манометр МПЗуф	13
3	11б18бк	Кран трехходовойц под манометр	13
4	«Ferro» (Польша)	Кран шаровый, муфтовый, Ду25	2
5	«Wlka» (Германия)	Термометр биметаллический	6
6		Грязевик, Ду100	2
7	«Ferro» (Польша)	Кран шаровый, муфтовый, Ду15	2
8	«Ferro» (Польша)	Крам шаровый, муфтовый, Ду50	2
9	«TiS» (Италия)	Фильтр фланцевый Ду100	1
10		Система учета тепла:	
10.1	«Kamstrup» (Дания)	Расходомер «Ultraflow», Ду65	2
10.2	«Kamstrup» (Дания)	Термопреобразователь Pt-500 (Комплект)	1
10.3	«Теплоком» (Россия)	Тепловычислитель ВКТ-7-2	1
10.4	«Kamstrup» (Дания)	Гильза под термопреобразователь Pt-500	3
11	«Vexve» (Финляндия)	Кран шаровый под приварку, Ду15	3
12	«Ferro» (Польша)	Клапан обратный, муфтовый, Ду15	2
13	«Samson» (Германия)	Регулятор температуры проходной, Тип43-1, Ду15	1
14	«Vexve» (Финляндия)	Кран шаровый под приварку Тип 100, Ду65	1
15	«Vexve» (Финляндия)	Кран шаровый над приварку Тип 100, Ду80	2
16	«TiS» (Италия)	Дисковый поворотный затвор, Ду80	3
17	«IMI» Великобритания	Регулятор перепада давления, DA616, Ду50	1
19	«Samson» (Германия)	Погодный компенсатор TROVIS 5432	1
19	«Samson» (Германия)	Датчик температуры наружного воздух 5227-2	1
20	«Samson» (Германия)	Датчик температуры накладной 5267-02	2

Продолжение таблицы 4.4

1	2	3	4
21	«IMI» Великобритания	Клапан CV 216 RGA, Ду50, с электроприводом MC 55	1
22	«Vexve» (Финляндия)	Край шаровый под приварку Тип 100, Ду40	4
23	«Grundfos» (Германия)	Насос циркуляционный UPS 40-120F H=55м, G=11.5 м ³ /ч, N=0.300kW	2
24	«Ferro» (Польша)	Капан обратный, муфтовый, Ду40	2
25	«TiS» (Италия)	Клапан обратный, межфланцевый Ду65	1
26	с. 5.903-13, в3	Труба кольцевая T193.00.001	13
27	«TiS» (Италия)	Фильтр фланцевый Ду80	1
28	«Ferro» (Польша)	Кран шаровый, муфтовый, Ду25	2
29	«IMI» Великобритания	Клапан балансировочный, STAD, Ду65, уставка 4.1	1

В каждом тепловом пункте обрабатываются 9 параметров, а также их максимальное и минимальное значения и границы интервалов.

Это параметры:

- значение температуры сетевой воды в подающем трубопроводе, °С;
- значение температуры сетевой воды в обратном трубопроводе, °С;
- значение температуры циркуляции, °С;
- расход сетевой воды в подающем трубопроводе, тонн/час;
- расход в обратном трубопроводе, тонн/час;
- значение давления сетевой воды в подающем трубопроводе, кгс/см;
- значение давления сетевой воды в обратном трубопроводе, кгс/см
- значение температуры внутри помещения, °С;
- количество отпущенного тепла, Гкал.

Также в режиме реального времени фиксируются:

- текущие дата и время соединения;
- параметры связи;
- время отклика при соединении – для отладочного режима.

При практической апробации (Приложение Б) считывания актуальных данных о теплоносителе в ТП возникла ситуация потери накопленных значений показаний температура сетевой воды в подающем трубопроводе. В результате на верхний уровень управления уходила неверная информация. Было предложено, проверено и отлажено следующее решение. На уровне промышленного контроллера, осуществляющего сбор данных о теплоносителе, дополнить имеющейся программный код, отвечающий за считывание параметров теплопотребления и передачу пакета данных на верхний уровень управления, «Блоком защиты от стирания данных». Данный блок представлен на рисунке 4.18. На рисунке 4 «Блок MemoryA1» реализует функцию

сохранения информации для счетчика расхода сетевой воды, установленного в подающем трубопроводе, а «Блок MemoryR1» – для счетчика расхода сетевой воды, установленного в обратном трубопроводе.

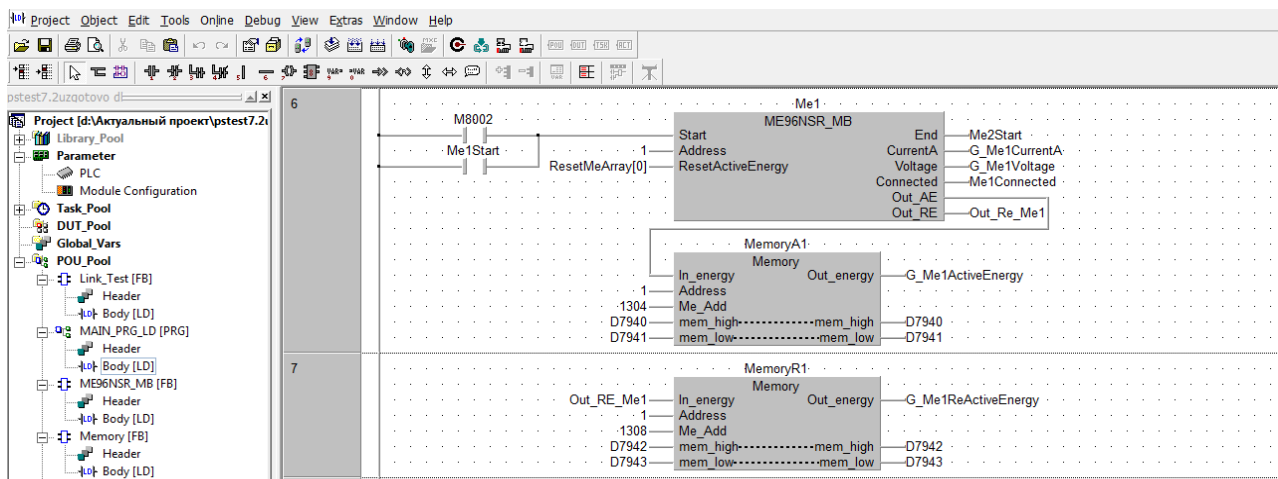


Рисунок 4.18 – Блок защиты от стирания данных, реализованный на промышленном контроллере Mitsubishi FX3U

На рисунке 4.19 представлено внутренний код «Блока защиты от стирания данных», реализованный на промышленных языках FBD и LD.

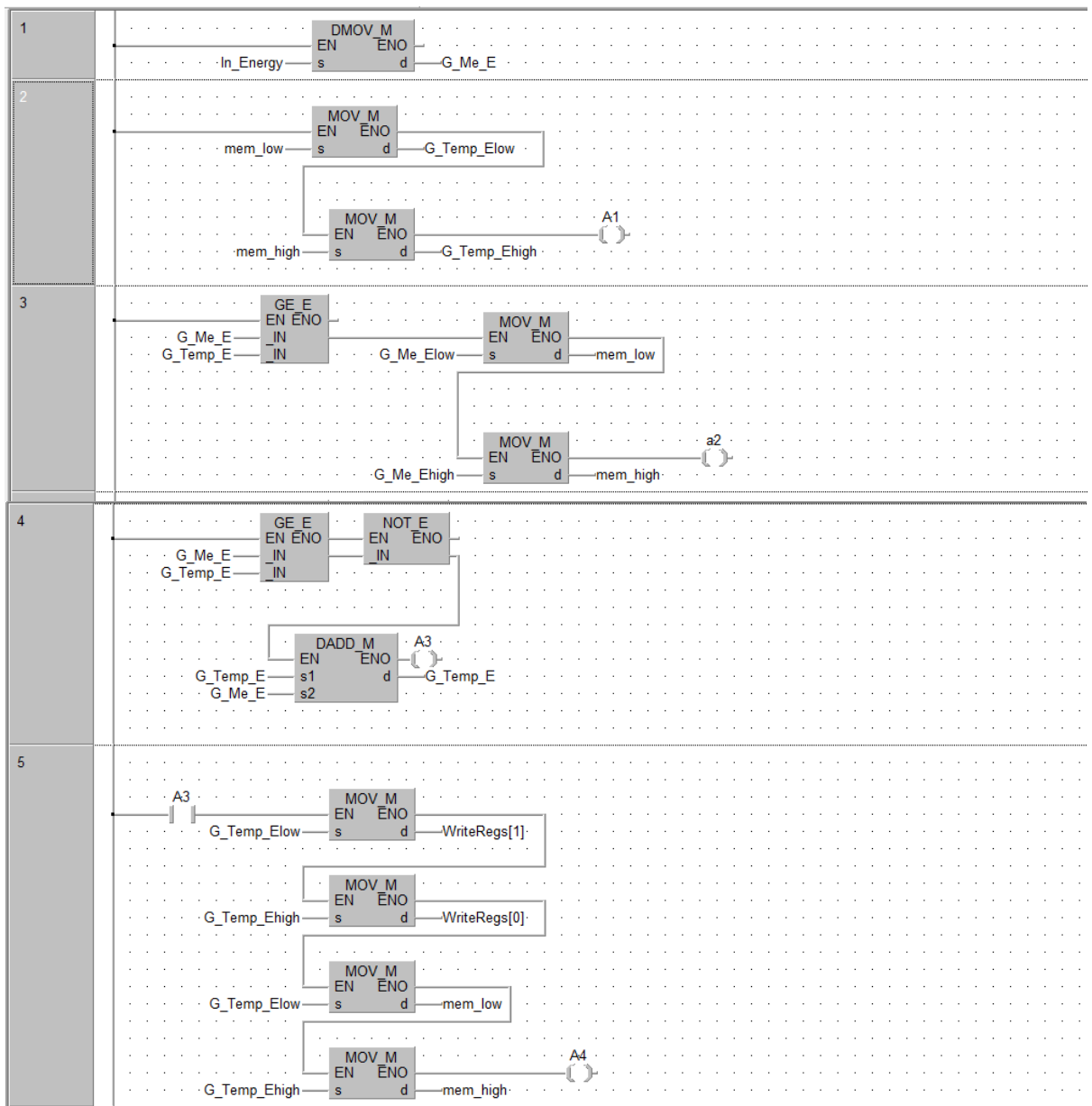
Также руководствуясь принципами конвейерной обработки данных некоторые комбинационные логические элементы, такие как сумматоры или умножители, быть ускорены путем увеличения количества логических элементов в коде программы [87]. Это привело к уменьшению времени цикла процессора и, таким образом, увеличилась скорость обработки инструкций.

В результате модификации программного кода контроллера, в том числе за счет инструкций, основанных на принципе конвейерной обработки данных структурную схему получения-обработки информации в ПТК (рисунки 4.2 и 4.13) можно представить в следующем виде (рисунок 4.20). Анализируя схему на рисунке 4.20 можно говорить о двухуровневой обработки данных. При апробации данного способа в расчет длительности одного интервала обслуживания $\tau[V(t)]$ (формула 2.2) за счет появления переменных по длительности циклах считывания-опроса параметров теплоснабжения был введен поправочный коэффициент q , определенный экспериментальным путем и принимающий свои значения в диапазоне от 1.05÷1.25. Формула 2.2 принимает вид:

$$\tau[V(t)] = \frac{\tau_{const} + \tau_{var}[V(t)]}{M * q} \quad (4.1)$$

За счет разделения обработки информации на уровни (уровень контроллера в ТП и верхний смарт-уровень), а также в использовании ПО

«Конструктор», в которой реализована идея конвейерной обработки данных, скорость получения-обработки информации как на ТП, так и в ЦТП, возросла на 5÷15%.



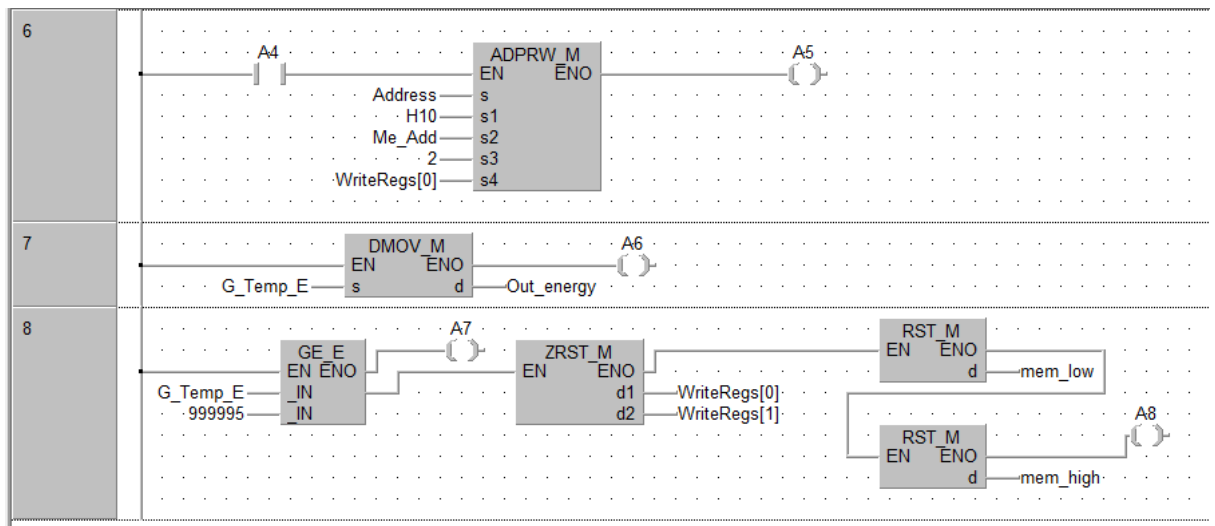


Рисунок 4.19 – Внутренний код Блока защиты от стирания данных Memory

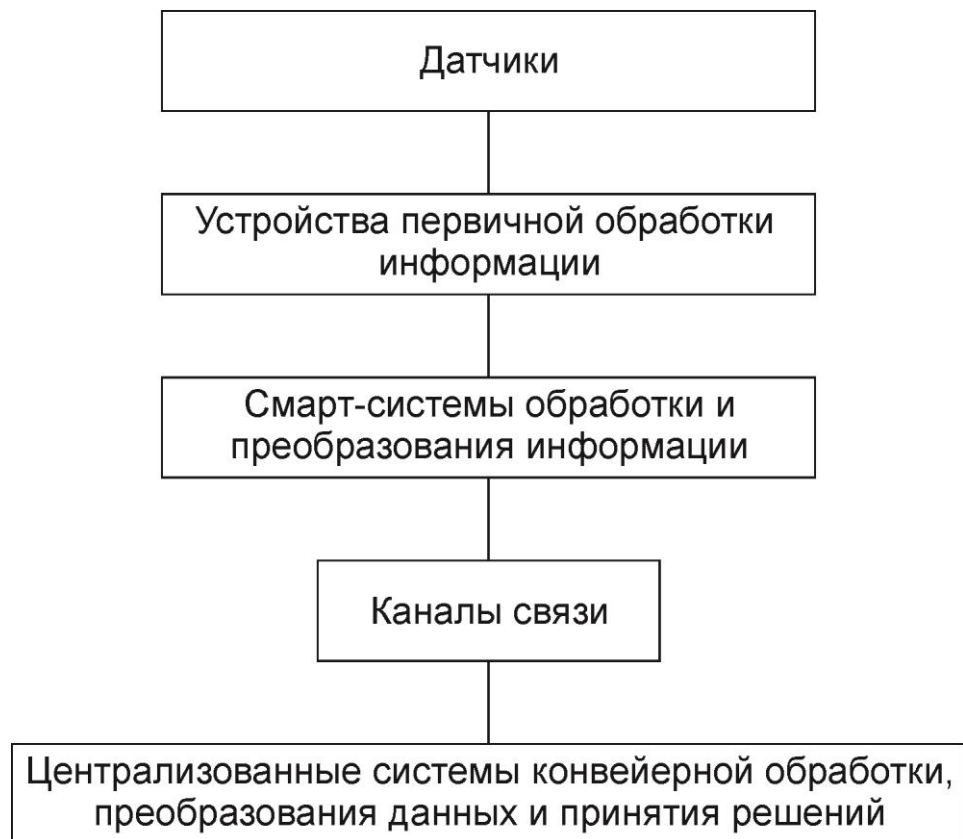


Рисунок 4.20 – Структурная схема получения-обработки информации в ПТК с конвейерной обработкой данных на верхнем уровне и включением конвейерной обработки данных на уровне контроллера

Применение разработанных программ и принципов позволило повысить надежность, экономичность, как тепловых пунктов, так и системы отопления в

целом. Благодаря проводимому мониторингу в реальном времени с использованием конвейерного способа обработки данных при работе тепловых пунктов подтверждена важность наличия в АСУ способов и механизмов конвейерного выполнения задач.

Этот вывод следует из теории развития производства и преобразования информации: все однотипные последовательности задач продуктивнее организовать в конвейерный план исполнения; наличие механизмов конвейерного выполнения в АСУ задач приводит к увеличению эффективности работы АСУ.

На Республиканском конкурсе «Жас Галым - 2019», организованным во исполнение Послания Президента Республики Казахстан, определившего 2019 год «Годом молодежи», проект «Автоматизация процессов управления параметрами системы теплоснабжения тепловых пунктов», основанный на материалах диссертационного исследования Сагынгановой И.К. старшего преподавателя ВКГТУ им. Д.Серикбаева занял первое место и был отмечен дипломом (Приложение В). Республиканский конкурс инновационных проектов «Жас Галым - 2019», организованный и проведенный Казахстанской национальной академией естественных наук, ставил своей целью стимулировать научную деятельность начинающих молодых ученых и поддержать их в реализации перспективных научно-исследовательских проектов, выполненных в ходе осуществления ими научно-исследовательской деятельности.

В настоящее время распределение и регулирование тепловой энергии как внутри, так и снаружи зданий в соответствии с потребностью являются одними из основополагающих подходов энергосбережения как в Казахстане, так и во всех развитых странах [88]. Несмотря на развитие и совершенствование возобновляемых источников энергии в Казахстане [89], в декабре 2017 года председатель Казахстанской электроэнергетической ассоциации Шаймерден Уразалинов, говоря о проблемах в секторе теплоснабжения отметил следующее [90]:

– имеется нерешенность организационно-финансовых вопросов для проведения реконструкции и модернизации отдельных составляющих систем теплоснабжения;

– действующие системы теплоснабжения в зданиях, присоединенные к централизованной системе теплоснабжения городов, как правило, не имеют автоматики и в незначительном количестве обеспечены приборами учета тепловой энергии и теплоносителя на тепловых пунктах.

При этом использование современных технологий управления тепловыми пунктами, объединенными в единую сеть, позволит значительно сэкономить электрическую энергию и более правильно осуществить распределение тепла в жилых и промышленных помещениях

Выводы по разделу

В настоящей главе получены следующие результаты:

- 1) предложена модифицированная структура программно-технических комплексов для теплоснабжающих систем, с выделением функциональных подсистем и использованием принципов конвейерной обработки информации;
- 2) создана и описана новая, внесенная в государственный реестр прав на объекты, охраняемые авторским правом № 4095 от «18» июня 2019 года, программа для ЭВМ «Конструктор», предназначенная для создания и редактирования схем теплоснабжения тепловых пунктов с возможностью считывания реальных данных параметров теплоснабжения (Приложение А);
- 3) проведены экспериментальные исследования эффективности применения автоматизации тепловых пунктов с использованием конвейерной обработки информации на реальных объектах (Приложение Б);
- 4) предложена модификация структурной схемы получения-обработки информации в ТП за счет использования конвейерной обработкой данных на верхнем уровне и включением конвейерной обработки данных на уровне контроллера;
- 5) за счет разделения обработки информации на уровни (уровень контроллера в ТП и верхний смарт-уровень), а также за счет использования созданного ПО «Конструктор», в котором реализована идея конвейерной обработки данных, скорость получения-обработки информации как на ТП, так и в ЦТП, возросла на 5÷15%.

ЗАКЛЮЧЕНИЕ

В диссертационной работе проведены теоретические и практические исследования и решена актуальная научная проблема повышения эффективности теплообеспечения за счет автоматизации тепловых пунктов, использующих в процессе в процессе получения информации о параметрах теплопотребления и принятия управляющих решений конвейерный способ обработки данных

В процессе диссертационных исследований получены следующие результаты:

1) выполнен аналитический обзор применения ТП и их оборудования, рассмотрены параметры работы теплового пункта и предложена концепция, использующая переменные циклы опроса на уровне тепловых пунктов системы теплоснабжения;

2) предложена методика обработки данных на основе технологии работы центрального процессора с процессами и потоками, которая позволит обеспечить высокое быстродействие обработки данных с тепловых пунктов и диспетчерских центров, а также увеличить надёжность и безопасность за счёт введения приоритета выполнения потока;

3) разработана гибридная информационная система, представляющую собой модель, объединяющую компоненты «клиент-серверной» архитектуры и использование вычислительного кластера с параллельной обработкой информации;

4) создана программа для ЭВМ «Конструктор», предназначенная для создания и редактирования схем теплоснабжения тепловых пунктов с возможностью считывания реальных данных параметров теплоснабжения. С ее использованием проведены экспериментальные исследования эффективности применения автоматизации тепловых пунктов с апробацией конвейерной обработки информации на реальных объектах;

5) за счет разделения обработки информации на уровни (уровень контроллера в ТП и верхний смарт-уровень), а также за счет использования созданного ПО «Конструктор», в котором реализована идея конвейерной обработки данных, скорость получения-обработки информации как на ТП, так и в ЦТП, возросла на 5÷15%.

Результаты диссертационной работы прошли успешную апробацию на объектах, оснащенных АТП (автоматизированными тепловыми пунктами), реализованные компанией МВ «Инжиниринг» (г.Караганда), занимающейся комплексными инженерными решениями для систем тепло- и водоснабжения.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Антонов О. Зеленая энергетика Казахстана в 21 веке: мифы, реальность и перспективы. www.polpred.com. 15.02.2019.
- 2 «Концепция развития топливно-энергетического комплекса Республики Казахстан до 2030 года»: утв. Постановлением Правительства Республики Казахстан от 28 июня 2014 года, № 724.
- 3 Богослоский В.Н., Сканава А.Н. Отопление: учебник для вузов. – М.: Стройиздат, 1991. – 735 с.
- 4 Свод правил Республики Казахстан: СП РК 4.02-101-2012. Отопление, вентиляция и кондиционирование воздуха. – Алматы: АО «КазНИИСА», 2015. – 89 с.
- 5 Котельные и тепловые пункты. www.ottim.ru. 10.05.2019.
- 6 Мухин О.А. Автоматизация систем теплогасоснабжения и вентиляции: учебное пособие для вузов. – Минск: Высш.шк., 1986. – 304 с.
- 7 Пырков В.В. Современные тепловые пункты. Автоматика и регулирование. – Киев: И ДП «Такі справи», 2007. – 252 с.
- 8 Певной П.А. Современное здание. Инженерные системы. – М., 2006. – 82 с.
- 9 Устройства управляющие РУНТ. www.mzta.ru. 16.05.2019.
- 10 Калмаков А.А., Кувшинов Ю.Я., Романова С.С., Щелкунов С.А. Автоматика и автоматизация систем теплогасоснабжения и вентиляции: учебник для вузов / под ред. В.Н. Богословского. – М.: Стройиздат, 1986. – 479 с.
- 11 Пырков В.В. Гидравлическое регулирование систем отопления и охлаждения. Теория и практика. - Киев: И ДП «Такі справи», 2005. – 304 с.
- 12 Рагулин П.Г. Информационные технологии: электрон. учебник. – Владивосток: ТИДОТ Дальневост. ун-та, 2004. – 208 с.
- 13 Информационные технологии управления: учебное пособие для вузов / под ред. проф. Г.А. Титоренко. – Изд 2-е., доп. – М.: ЮНИТИ-ДАНА, 2003. – 439 с.
- 14 Основы организации вычислительных систем. Конвейерная организация. www.lcard.ru. 10.07.2016.
- 15 Лагарникова А.В. Принцип организации обработки данных на примере пятиступенчатого конвейера // Научное сообщество студентов. Междисциплинарные исследования: электрон. сборн. ст. по матер. IV студ. междунар. науч.-практ. конф. – Новосибирск: Изд. АНС «СибАК», 2016. – С. 59-63.
- 16 Конвейерная обработка данных. Суперскаляризация. www.sites.google.com. 12.07.2016.
- 17 СИЗИТ. Оценка производительности идеального. www.xsieit.ru. 10.07.2016.
- 18 Архитектура микропроцессоров. Конвейерная организация работы

микропроцессора. www.intuit.ru. 10.07.2016.

19 Архитектура ЭВМ. www.rushkolnik.ru. 13.07.2016.

20 Уваров А.В. Автоматизация инженерных систем современных зданий и комплексов // Промышленные АСУ и контроллеры. – 2005. – № 9. – С.15-19.

21 Томилова Н.И. Разработка информационной системы поддержки принятия решений по проведению наладочных и перспективных мероприятий в теплоснабжающих системах мегаполисов: дис. ... канд. техн. наук. – Караганда, 2010. – 147 с.

22 Аверьянов В.К., Быков С.И. Вероятностно-статистическое описание режима работы системы теплоснабжения // Изв. Вузов. Серия Энергетика. – М., 1979. – № 11. – С. 55-60.

23 Аверьянов В.К., Быков С.И. Теплообмен в помещениях при программном отпуске тепла // ИФЖ. – 1982. – Т. 43, № 3. – С. 406-412.

24 Громов Н.К. О схемах автоматизации абонентских установок крупных городских систем централизованного теплоснабжения // Водоснабжение и санитарная техника. – М., 1981. – № 12. – С. 8-9.

25 Монахов Г.В., Сибирякова Л.П., Карасёв Н.И., Фольгарт В.И. Программная реализация математических моделей переменных тепловых режимов городских тепловых сетей // Автоматизированные системы научных исследований коллективного пользования: сб. науч. тр. науч.-производ. объединения по автоматизации управления городским хозяйством и вычислительными работами. – М., 1985. – С. 81-88.

26 Монахов Г.В., Войтинская Ю.А. Моделирование управления режимами тепловых сетей. – М.: Энергоатомиздат, 1995. – 221 с.

27 Карасёв Н.И., Фольгарт В.И., Дирксен В.Л. и др. Об одной модели установившегося потокораспределения в больших трубопроводных системах // Энергетика: изв. высш. учеб. заведений. – М., 1989. – № 1. – С. 113-117.

28 Карасёв Н.И., Фольгарт В.И., Дирксен В.Л. и др. Моделирование установившихся гидравлических режимов больших теплоснабжающих систем с дроссельным управлением // Электрон. моделирование. – 1990. – № 1. – С. 72-76.

29 Карасев Н.И., Томилова Н.И. Синтез допустимых гидравлических режимов распределительных фрагментов теплоснабжающих систем // Труды университета. – Караганда, 2008. – Вып. 3. – С. 75-77.

30 Карасев Н.И., Томилова Н.И. Методы декомпозиции гидравлических цепей теплоснабжающих систем // Труды университета. – Караганда: КарГТУ, 2008. – Вып. 2. – С. 72-75.

31 Сагынганова И.К., Квасов А.И., Калинин А.А. Комбинированные методы получения и обработки информационных потоков в централизованных системах теплоснабжения // Вестник Казахской национальной академии естественных наук №4/2019, 2019. С. 34-37.

32 Основы автоматизации насосных станции. www.nasosinfo.ru. 15.02.2019.

- 33 Кульба В.В., Косяченко С.А., Лебедев В.Н. Автоматизированные информационно-управляющие системы социально-экономических и организационных структур // Проблемы управления. – 2009. – Вып. 3. – № 1. – С. 73-86.
- 34 Самофалов К.Г. Основы теории многоуровневых конвейерных вычислительных систем / под. ред. Г.М. Луцкий. – М.: Радио и связь, 1989. – 271 с.
- 35 Конвей Р.В. Теория расписаний / под. ред. В.Л. Максвелл, Л.В. Миллер. – М.: Наука, 1975. – 360 с.
- 36 Рыбакова Д.А., Бакланов А.Е., Квасов А.И. Использование конвейерной системы обработки данных для регулирования работы тепловых пунктов // Вестник ВКГТУ. – 2014. – № 3. – С. 102-106.
- 37 Автоматизированная насосная станция. www.ngpedia.ru. 15.02.2019.
- 38 Стандартные автоматизированные блочные тепловые пункты фирмы Danfoss. Пособие. – М.: ЗАО «Данфосс», 2006. – 47 с.
- 39 Rybakova D. Application of conveyor data processing of work of thermal points for possibility of decrease in losses of thermal energy // Зеленая экономика – будущее человечества: матер. междунар. науч.-практ. конф. – Усть-Каменогорск: ВКГТУ, 2014. – С. 323-332.
- 40 Стратегический план государственного учреждения «Управление энергетики и коммунального хозяйства Восточно-Казахстанской области» на 2010 – 2014 годы. www.pandia.ru. 15.02.2019.
- 41 Novikov DA, Ashimov A.A., Sultanov B.T., Adilov Zh.M., Borovsky Yu.V., Alshanov R.A. Macroeconomic Analysis and Parametric Control of a National Economy. – М.: Springer, 2013. – 256 p.
- 42 Мамиконов А.Г., Кульба В.В. и др. Проектирование подсистем и веньев АСУ. – М.: Высш. шкл., 1975. – 248 с.
- 43 Микрин Е.А., Кульба В.В. и др. Информационное обеспечение систем организационного управления (теоретические основы): в 3 ч. – М.: Изд-кий дом физ. и мат. лит-ры, 2012. – 528 с.
- 44 Novikov D.A., Ashimov A.A., Sultanov B.T., Adilov Zh.M., Borovsky Yu.V., Alshanov R.A. Macroeconomic Analysis and Parametric Control of a National Economy. – М.: Springer, 2013. – 256 p.
- 45 Revako V.M. Methods and models in the tasks of managing production. – М.: V. and., Tallinn, 1979. – 171 p.
- 46 Benjamin S. Duran, Patrick L. Odell. Cluster analysis. – Springer-Verlag Berlin-Heidelberg-New York, 1974. – 129 p.
- 47 Оре О. Теория графов. – М.: СССР, 2008. – 352 с.
- 48 Воеводин В.В. Математические основы параллельных вычислений. – Изд. МГУ, 1991. – 345 с.
- 49 Воеводин В.В. Информационная структура алгоритмов и программ. – Изд. МГУ, 1997. – 139 с.
- 50 Воеводин В.В., Воеводин, В.В. Параллельные вычисления. – СПб.:

БХВ-Петербург, 2002. – 608 с.

51 Samofalov K.G., Rimankevich A.M., Valuisky V.N., Kanevsky Yu. S., Pinevich M.M. Applied theory of digital automata. – Head and Publishing House, 1987. – 375 p.

52 Petrov Yu.A., Shlimovich E.L., Iriupin Yu.V. Integrated automation of enterprise management – M.: Finance and Statistics, 2001. – 160 p.

53 Phillips D., Garcia-Días A. Methods of network analysis / trans. with english. – M.: Mir, 1984. – 496 p.

54 Конвей Р.В., Максвелл В.Л., Миллер Л.В. Теория расписаний. – М: Наука, 1975. – 360 с.

55 Богданова О.В. Модельно-алгоритмическое обеспечение конвейерного выполнения задач в распределенных АСУ: дис. ... канд. тех. наук. – Красноярск, 2008. – 202 с.

56 Квасов А.И., Бакланов А.Е., Сагынганова И.К., Анапьянова С.Б. Система конвейерной обработки информации в современных тепловых сетях // Сб. науч. тр. «Актуальные научные исследования в современном мире. – 2018. – Вып. 2 (34), Ч. 6. – С. 158-162.

57 Sagynganova K.I., Markin V.B. The organizations of the tasks implementation in the distributed automatic control systems of heat supply stations // Известия Национальной академии наук Республики Казахстан. Серия геологии и технических наук. – 2019. – № 1 (433). – С. 63-67.

58 Rybakova D.A., Sagynganova K.I., Kumargazhanova S.K., Baklanov A.E., Shvets O.Y., Application of a CPU streaming technology to work of the computer with data coming from the network on the example of a heating station // Матер. XVIII междунар. конф. молодых специалистов по микро/нанотехнологиям и электронным приборам. – Республика Алтай, 2017. – С. 128-131.

59 Шлимович Е.Л. Концепция управленческого учета, основанная на методах «конвейерного» управления. www.avacco.ru. 25.04.2019.

60 Богданов Е.В. Использование методов конвейерного управления в реинжиниринге бизнес-процессов: дис. ... магистр. – Красноярск, 2017. – 85 с.

61 Заручевская Г.В. Применение мелкозернистого локально-параллельного программирования при решении задач математической физики методом сеток: дис. ... канд. техн. наук. – Петрозаводск, 2008. – 198 с.

62 Kung H.T. Why systolic architecture? // Computer. – 1982. – Vol. 15, № 1. – P. 37-46.

63 Молдован Д.И. О разработке алгоритмов для систолических матриц – СБИС. ТИИЗР, 1983. – Т. 71, № I. – С. 140-149.

64 Сагынганова И.К., Квасов А.И., Бакланов А.Е. Использование систолических массивов для создания информационных сетей с высокой скоростью передачи данных от тепловых пунктов // Вестник Казахской национальной академии естественных наук. – 2019. – № 2. – С. 27-34.

65 Побегайло А.П. Системное программирование в Windows. – СПб., 2006. – 1056 с.

66 Тапенамбум Э. Современные операционные системы. – Изд. 2-е. – СПб., 2007. – 1038 с.

67 Бакланова О.Е., Бакланов А.Е., Сагынганова И.К. Mathematical modeling of data processing pipeline technology in heat-networks // Матер. междунар. науч.-практ. конф. «Вычислительные и информационные технологии в науке, технике и образовании». – Усть-Каменогорск, ВКГТУ им. Д. Серикбаева, 2018. – С. 200-204.

68 Квасов А.Е., Бакланов И.К., Сагынганова И.К., Мамырбекова Г.К. Использование методики обработки потоков для мониторинга работы теплосетями // Prospects for the Development of Modern Science: materials of the iii international scientific-practical conference. – Seoul, 2018. – С. 184-189.

69 Пырклов В.В. Современные тепловые пункты. Автоматика и регулирование. – Киев: И ДП «Такісправи», 2007. – 252 с.

70 Работа с процессами. Функции CreateProcess и TerminateProcess www.makak.ru. 25.04.2019.

71 Сағынғанова И.К. Жылумен қамту жүйесі параметрлерін басқару үдерістерін автоматтандыру // Вестник Казахской национальной академии естетствених наук. – 2019. – № 1. – С. 36-39.

72 Györök G., Sagynganova I., Rybakova D. Methodological aspects of doctoral research in terms of conveyor technology // Матер. междунар. конф. «Современные тенденции подготовки технических кадров и преподавания английского языка в условиях индустриально-инновационного развития Республики Казахстан» и XIII нац. конф. KazTEA "25-летие независимого Казахстана: к новым свершениям вместе с английским языком" – Усть-Каменорск: ВКГТУ, 2016. – С. 71-749.

73 Сагынганова И.К. Поточковая обработка данных тепловых пунктов в реальном времени // Труды университета. – Караганда: КарГТУ, 2018. – Вып. 4. – С. 159-162.

74 Информационный портал. parallel.ru. 25.04.2019.

75 Сагынганова И.К., Бакланов А.Е. Методика конвейерной обработки данных на примере городских тепловых сетей // Матер. III междунар. науч.-техн. конф. студентов, магистрантов и молодых ученых «Творчество молодых инновационному развитию Казахстана». – Усть-Каменогорск: ВКГТУ им. Д. Серикбаева, 2017. – С. 293-298.

76 Györök G., Baklanov A., Baklanova O., Grigoryeva S., Kumargazhanova S., Sagynganova I., Vais Y. The Development of Hybrid IP Architecture for Solving the Problems of Heating Networks (using pipeline-parallel data processing technology). // Acta Polytechnica Hungarica, Hungry, 2020. – № 17 (1). – P. 123-140.

77 Сагынганова И.К. Разработка гибридной архитектуры информационных систем для решения задач тепловых сетей // Вестник Национальной инженерной академии РК. – 2018. – № 4 (70). – С. 153-155.

78 Сагынганова И.К. О вопросах внедрения цифровизации и информационных технологий на примере систем теплоснабжения предприятий

агропромышленного комплекса // Издәністер, нә тижелер. – Казахский национальный аграрный университет, 2018. – № 4. – С. 148-153.

79 Шпаковский Г.И., Серикова Н.В. Программирование для многопроцессорных систем в стандарте MPI. – Минск: БГУ, 2002. – 323 с.

80 Informational portal. en.wikipedia.org. 25.04.2019.

81 Informational portal Bauman National Library. www.ru.bmstu.wiki. 25.04.2019.

82 Приказ Министерства жилищно-коммунального хозяйства Республики Беларусь. О Правилах технической эксплуатации систем водоснабжения и водоотведения населенных мест: утв. от 06.04.1994 № 23 www.pravo.levonevsky.org. 18.10.2016.

83 Эгильский И.С. Автоматизированные системы управления технологическими процессами подачи и распределения воды. – Л.: Стройиздат, Ленингр. отд-ние, 1988. – 216 с.

84 Частотное регулирование электроприводов. www.e-audit.ru. 20.10.2016.

85 Карасёв Н.И., Крицкий А.В., Томилова Н.И., Цок Г.Н. Информатизация систем централизованного теплоснабжения в инфраструктуре жизнеобеспечения мегаполисов Казахстана // Тр. междунар. науч.-практ. конф. «Состояние, проблемы и задачи информатизации в Казахстане», посв. 70-летию КазНТУ им К.И. Сатпаева и 10-летию Международной Академии Информатизации (МАИН): – Алматы: Изд-во РИО, 2004. – 708 с.

86 Сагынганова И.К., Калинин А.А. Программа «Конструктор» для создания и редактирования схем теплоснабжения тепловых пунктов с возможностью считывания реальных данных (программа для ЭВМ). Свидетельство о внесении сведений в государственный реестр прав на объекты, охраняемые авторским правом № 4095 от «18» июня 2019 года. – РГП «Национальный институт интеллектуальной собственности» МЮ РК.

87 Шаих Б.Н., Сагынганова И.К., Квасов А.И. Конвейерная система регулирования теплового режима зданий // Матер. V междунар. науч.-техн. конф. студ., магистр. и мол. ученых «Творчество молодых инновационному развитию Казахстана». – Усть-Каменогорск: ВКГТУ им. Д. Серикбаева, 2019. – С. 272-274.

88 Galkin S.V., Sagynganova I.K. Rational principles water supply of cooling systems of thermal power plants in east Kazakhstan // Матер. IX междунар. конф. «Эффективное использование ресурсов и охрана окружающей среды – ключевые вопросы развития горно-металлургического комплекса» и XII междунар. Казахстанско-Российской науч. конф. «Перспективные технологии, оборудование и аналитические системы для материаловедения и наноматериалов» – Оскемен: ВКГТУ им. Д. Серикбаева, 2015 г. – С. 347-349.

89 Вилемзон И.С., Сагынганова И.К. Возможности возобновляемых источников энергии в Казахстане // Матер. междунар. науч.-техн. конф. студентов, магистрантов и молодых ученых «Творчество молодых инновационному развитию Казахстана», Часть I. – Оскемен: ВКГТУ им. Д.

Серикбаева, 2015 г. – С. 74-75.

90 Бакланов А.Е., Сагынганова И.К. Разработка новых алгоритмов контроля и управления в современных тепловых пунктах // Матер. междунар. науч.-практич. конф. «Вычислительные и информационные технологии в науке, технике и образовании» (СITech-2018). Том 1, Часть I. – Оскемен: ВКГТУ им. Д. Серикбаева, 2018 г. – С. 72-78.

ПРИЛОЖЕНИЕ А



Рисунок А.1 – Авторское свидетельство

Код программы «Конструктор»

```
#include "stdafx.h"
#include "Consrtuctor.h"
#include "ConsrtuctorDoc.h"
#include "ConsrtuctorView.h"
#include "About\TubeAbout.h"
#include "About\BatteryAbout.h"
#include "About\TurnAbout.h"
#include "About\TrinityAbout.h"
#include "About\VentilAbout.h"
#include "About\NasosAbout.h"
#include "About\DrosselAbout.h"
#include "About\VentilatorAbout.h"
#include "About\VodovodoAbout.h"
#include "About\SoploAbout.h"
#include "About\ConnectorAbout.h"
#include "About\LabelAbout.h"
#include "About\ExtttermoAbout.h"
#include "About\WinParam.h"
#include "MainFrm.h"
CConsrtuctorView::CConsrtuctorView()
{listik=500;
AboutDialog=NULL;
element=0;
Dragging=false;
Selecting=false;
SelectPrepared=false;
worksp.DrawVect=0;
r.left=r.top=r.bottom=r.right=0;
for (int i=0;i<14;i++)
{objects[i].grdat.group=0;
  objects[i].grdat.groupOn=true;}
//Объект Труба : заполнение первоначальных параметров
objects[1].X0=0;
objects[1].Y0=0;
objects[1].Radius=1;
objects[1].Angle=0;
objects[1].Height=10;
objects[1].animate=0;
objects[1].Bounds.left=objects[1].X0;
objects[1].Bounds.top=objects[1].Y0;
objects[1].Bounds.height=2;
objects[1].Bounds.width=10;
objects[1].Pribor=1;
objects[1].Right=1;
strcpy(objects[1].Text,"p1");
////////////////////////////////////
////
// CConsrtuctorView message handlers
void CConsrtuctorView::OnSize(UINT nType, int cx, int cy)
{CCustomView::OnSize(nType, cx, cy);
  RubberRect.right=RubberRect.left;
  RubberRect.bottom=RubberRect.top;}
//-----
```



```

-----
void CConsrtuctorView::OnDraw(CDC* pDC)
{wglMakeCurrent(pdc->m_hDC, hGLRC);
  if (element!=-1)display();}
//-----
-----
void CConsrtuctorView::OnMouseMove(UINT nFlags, CPoint point)
{VECTORXY xy;
  xy.X=getGLX(point.x);
  xy.Y=getGLY(point.y);
  MOUSEXY Mxy;
  if ( xy.X>((long)xy.X+0.5) )Mxy.X=((long)xy.X+1);
  else
  {if ( xy.X<((long)xy.X-0.5) )Mxy.X=((long)xy.X-1);
   else Mxy.X=(long)xy.X;}
  if ( xy.Y>((long)xy.Y+0.5) )Mxy.Y=((long)xy.Y+1);
  else
  {if ( xy.Y<((long)xy.Y-0.5) )Mxy.Y=((long)xy.Y-1);
   else Mxy.Y=(long)xy.Y;   }
  CString str="Координаты: X=";
  char buffer[100];
  str+=itoa(Mxy.X,buffer,10);
  str+="      Y=";
  str+=itoa(Mxy.Y,buffer,10);
  ((CMainFrame*)(AfxGetApp()->m_pMainWnd))->SetStatusText(str);

  // TODO: Add your message handler code here and/or call default
  switch (DebugMode)
  {case 0:
    { m.X=getGLX(point.x);
      m.Y=getGLY(point.y);
      MoveDrag(point);
      Move(m);
      break;}
    case 1:
      {OnZoomInMouseMove(point);break;}
    case 3:
      {OnSelectMouseMove(point);break;}
    }
    CCustomView::OnMouseMove(nFlags, point);}
//-----
-----
void CConsrtuctorView::GetLeftTop(RECTAN *rect)
{if (!Dragging)
  {if(element!=0)
    {rect->left=int((m.X-rect->width/2.0)/grid)*grid;
     rect->top=int((m.Y+rect->height/2.0)/grid)*grid;}
   else
   {rect->left=int((m.X-Shift[0])/grid)*grid;
    rect->top=int((m.Y+Shift[1])/grid)*grid;}
  }
//-----
-----
void CConsrtuctorView::Move(const MOUSEXY m)
{CRgn Reg;
  //GetClientRect(&rectClient);

```

```

#define zazor 10
CRect Client;
Client.left=rectClient.left+zazor;
Client.top=rectClient.top+zazor;
Client.bottom=rectClient.bottom-zazor;
Client.right=rectClient.right-zazor;
Reg.CreateRectRgnIndirect (&Client);
CPoint point;
point.x=getX(m.X);
point.y=getY(m.Y);
if(!Reg.PtInRegion(point))
{DrawRect(r);
  r.bottom=r.top;
  r.right=r.left;
  DrawRect(r);
  return;}
if (Dragging)
{DrawElem(&objects[duo_selected.place.Type]);}
else
{switch (element)
  {case 0 :
{
      DrawRect(r);
      r.bottom=r.top;
      r.right=r.left;
      DrawRect(r);
      break; }
    default : {if (element>0)DrawElem(&objects[element]);break;}
  }
void CConsrtuctorView::DrawElem(ELEMENT_DATA * elem)
{
  GetLeftTop(&elem->Bounds);
  DrawRect(r);
  r=getBRect (&elem->Bounds);
  DrawRect(r);}
//-----
void CConsrtuctorView::OnKeyDown(UINT nChar, UINT nRepCnt, UINT nFlags)
{
  // TODO: Add your message handler code here and/or call default
  if (nChar==VK_ESCAPE)
  {
    element=0;
    r.bottom=r.top;
    r.right=r.left;
    DebugMode=0;
    Zooming=0;
    ZoomPrepared=0;
    if (AfxGetApp()->m_pMainWnd!=NULL)
    {
      if (DebugMode==0)for (int i=0;i<ELEM_KOL;i++)
      ((CMainFrame*)(AfxGetApp()->m_pMainWnd))->DrawElem[i]=0;
    }
    //ЧТОБЫ МОЖНО БЫЛО СКРОЛЛИТЬ ReleaseCapture();
    EndSelMode();
    ::SetClassLong(this->m_hWnd,GCL_HCURSOR,(long)old);
  }
  if (nChar==VK_DELETE) DeleteObject();
  if (nChar==VK_UP)
  {
    m.Y+=grid;
    Move(m);}
  if (nChar==VK_DOWN)
  {m.Y-=grid;

```

```

        Move(m); }
    if (nChar==VK_LEFT)
    {
        m.X-=grid;
        Move(m); }
    if (nChar==VK_RIGHT)
    {
        m.X+=grid;
        Move(m); }
    if (nChar==VK_RETURN)
    {
        int el=element;
        AddElement();
        display();
        element=el;}
    CCustomView::OnKeyDown(nChar, nRepCnt, nFlags);}
//-----
void CConsrtuctorView::OnLButtonDown(UINT nFlags, CPoint point)
{
    CConsrtuctorDoc * m_pDoc=GetDocument();
    SetCapture();
    m.X=getGLX(point.x);
    m.Y=getGLY(point.y);
    switch (DebugMode)
    {
        case 0://простой режим(расстановка объектов)
        {
            if (!MouseOnSub(true))
                {StartSelectMode(point); }
            break; }
        case 1:
            {ZoomIn(point);break;}//режим Масштабирования ZOOM_IN
        case 3:
            {OnSelectMouseDown(nFlags,point);break;}//режим группового
перетаскивания }
        CCustomView::OnLButtonDown(nFlags, point);
    }
//-----
// Режим выделения и перетаскивания компонентов
void CConsrtuctorView::StartSelectMode(CPoint point)
{
    if (element!=0) return;
    if (MouseOnObject())
    {
        if(type_selected==NONE) {EndSelMode();return;}
        SelStack.clear();
        SelStack.push_back(duo_selected);
        Selecting=false;
        SelectPrepared=true;
        StartDragElem(point);
        DebugMode=3; }
    else
    {
        SetCapture();
        SelStack.clear();
        DrawRubberRect(RubberRect);
        RubberRect.left=point.x;
        RubberRect.top=point.y;
        RubberRect.right=RubberRect.left;
        RubberRect.bottom=RubberRect.top;
        Selecting=1;
        element=-1;
    }
}

```

```

        DebugMode=3; }
    }
//-----
-----
void CConsrtuctorView::OnSelectMouseMove(CPoint point)
{
    m.X=getGLX(point.x);
    m.Y=getGLY(point.y);
    if (Selecting)
    {
        DrawRubberRect (RubberRect);
        RubberRect.right=point.x;
        RubberRect.bottom=point.y;
        DrawRubberRect (RubberRect);    }
    if (SelectPrepared&&(Dragging))
    {
        Move(m); }
}
//-----
-----
void CConsrtuctorView::OnSelectMouseUP()
{
    ReleaseCapture();
    if (Selecting)
    {
        if (!SelectPrepared)
        {
            type_selected=NONE;
            SelectPrepared=true;
            Selecting=false;
            DrawRubberRect (RubberRect);
            FillSelStack();
            element=0;
            RubberRect.right=RubberRect.left;
            RubberRect.bottom=RubberRect.top;}

        if (SelectPrepared&&(Dragging))
        {
            MoveSel();}
    }
}
//-----
-----
void CConsrtuctorView::MoveSel()
{
    MOUSEXY delta;
    CConsrtuctorDoc * m_pDoc=GetDocument();
    if (!SelectPrepared) return;
    DrawRect(r);
    ELEMENT_DATA ell;
    CTempl * obj;
    obj=m_pDoc->GetTempl(duo_selected);
    obj->GetData(&ell);
    delta.X=round_to_dig(getGLX(r.left))-ell.Bounds.left;
    delta.Y=round_to_dig(getGLY(r.top))-ell.Bounds.top;
    PAIR::iterator itr;
    for(itr=SelStack.begin();itr!=SelStack.end();itr++)
    {
        obj=m_pDoc->GetTempl(*itr);
        obj->GetData(&ell);
        ell.Bounds.left+=delta.X;
        ell.Bounds.top+=delta.Y;
        obj->UpdateFromBounds(&ell);
        m_pDoc->Analyse(obj); }
    Dragging=false;
    r.right=r.left;
}

```

```

        r.bottom=r.top;
        CompileList();
        element=0;    }
//-----
-----
void CConstrtuctorView::OnSelectMouseDown(UINT nFlags,CPoint point)
{
    CConstrtuctorDoc * m_pDoc=GetDocument();
    if (!SelectPrepared) return;
    m.X=getGLX(point.x);
    m.Y=getGLY(point.y);
    bool result=false;
    PAIR::iterator itr;
    CTempl * obj;
    for(itr=SelStack.begin();itr!=SelStack.end();itr++)
    {
        obj=m_pDoc->GetTempl(*itr);
        if (obj->PointInRgn(&m))
        {
            if ((nFlags&MK_SHIFT)==MK_SHIFT)
            {
                SelStack.erase(itr);
                obj->SetFocus(0);
                if (SelStack.empty())
                {
                    EndSelMode();
                    OnLButtonDown(nFlags,point);
                    return; }
            else
            {
                type_selected=itr->place.Type;
                duo_selected=(*itr);
                StartDragElem(point); }
            result=true;
            break;        }
        if (!result)
        {
            if ((nFlags&MK_SHIFT)==MK_SHIFT)
            {
                if (MouseOnObject(false))
                {
                    if (type_selected==NONE) {EndSelMode();return;}
                    SelStack.push_back(duo_selected);
                    StartDragElem(point); }
                else
                {
                    EndSelMode();
                    OnLButtonDown(nFlags,point);
                    return; }
            else
            {
                EndSelMode();
                OnLButtonDown(nFlags,point);
            }
        }
    }
//-----
-----
void CConstrtuctorView::EndSelMode()
{CConstrtuctorDoc * m_pDoc=GetDocument();
    PAIR::iterator itr;
    CTempl * obj;
    for(itr=SelStack.begin();itr!=SelStack.end();itr++)
    {
        obj=m_pDoc->GetTempl(*itr);
        if (obj!=NULL) obj->SetFocus(FALSE);
    }//m_pDoc->SetZeroFocusAll();
    SelStack.clear();
    Selecting=0;
    SelectPrepared=0;
}

```

```

        RubberRect.right=RubberRect.left;
        RubberRect.bottom=RubberRect.top;
        type_selected=NONE;
        Dragging=false;
        DebugMode=0; }
//-----
-----
void CConstrtuctorView::FillSelStack()
{
    CConstrtuctorDoc * m_pDoc=GetDocument();
    FillSelStackFrom(&m_pDoc->tubeV);
    FillSelStackFrom(&m_pDoc->batV);
    FillSelStackFrom(&m_pDoc->turnV);
    FillSelStackFrom(&m_pDoc->triV);
    FillSelStackFrom(&m_pDoc->ventV);
    FillSelStackFrom(&m_pDoc->nasV);
    FillSelStackFrom(&m_pDoc->drosV);
    FillSelStackFrom(&m_pDoc->airbatV);
    FillSelStackFrom(&m_pDoc->vodovodoV);
    FillSelStackFrom(&m_pDoc->soploV);
    FillSelStackFrom(&m_pDoc->connV);
    FillSelStackFrom(&m_pDoc->kranV);
    FillSelStackFrom(&m_pDoc->labV);
    FillSelStackFrom(&m_pDoc->extermV);
    if (SelStack.empty()) EndSelMode();}
//-----
-----
template <class V>
void CConstrtuctorView::FillSelStackFrom(V * vect)
{CConstrtuctorDoc * m_pDoc=GetDocument();
    ELEMENT_DATA ell;
    V::iterator itr;
    CDuo dd;
    for(itr=vect->begin();itr!=vect->end();itr++)
    {
        itr->GetData(&ell);
        CRgn Reg;
        Reg.CreateRectRgnIndirect (&RubberRect);
        if(Reg.PtInRegion(getX(ell.Bounds.left),getY(ell.Bounds.top)))
        {
            if(Reg.PtInRegion(getX(ell.Bounds.left+ell.Bounds.width),getY(ell.Bounds.top-ell.Bounds.height)))
            {
                dd=m_pDoc->GetDuo(itr);
                SelStack.push_back(dd);
                itr->SetFocus(1);}
//-----
-----
void CConstrtuctorView::StartDragElem(CPoint point)
{
    CConstrtuctorDoc * m_pDoc=GetDocument();
    if (m_pDoc==NULL) return;
    ELEMENT_DATA * str=&objects[duo_selected.place.Type];
    CTempl * vect=m_pDoc->GetTempl(duo_selected);
    vect->GetData(str);
    m.X=getGLX(point.x);
    m.Y=getGLY(point.y);
    Shift[0] = m.X-str->Bounds.left;
    Shift[1] = str->Bounds.top-m.Y;
    Dragging=true;

```

```

        element=-1;
        Move(m);}
//-----
-----
// Режим выделения и перетаскивания компонентов
////////////////////////////////////
////
CDuo CConstrtuctorView::AddElement()
{
    CDuo duo;
    duo.place.Type=-1;
    CConstrtuctorDoc * m_pDoc=GetDocument();
    if (m_pDoc==NULL) return duo;
    duo.place.Type=element;
    if (element<1) return duo;
    duo=m_pDoc->AddElement(element,&objects[element]);
    CompileList();
    element=0;
    for (int i=0;i<ELEM_KOL;i++) ((CMainFrame*)(AfxGetApp()-
>m_pMainWnd)->DrawElem[i]=0;
    return duo;}
//-----
-----
CRect CConstrtuctorView::getBRect(const RECTAN *t)
{
    CRect r;
    r.left=getX(t->left);
    r.top=getY(t->top);
    r.bottom=r.top-getY(t->height)+getY(0);
    r.right=r.left+getX(t->width)-getX(0);
    return r;}
//-----
-----
template <class V>
void CConstrtuctorView::DrawWhenCompileElem(V * vect)
{
    V::iterator itr;
    for(itr=vect->begin();itr!=vect->end();itr++)itr->Draw(true);
    if (worksp.DrawVect)DrawVectors();}
//-----
-----
void CConstrtuctorView::OnLButtonDblClk(UINT nFlags, CPoint point)
{
    //if (element!=0) return;
    if (IsDebugModeEquivTo0())OnLeftDblClick();
    CCustomView::OnLButtonDblClk(nFlags, point);}
//-----
-----
bool CConstrtuctorView::IsDebugModeEquivTo0()
{
    bool res=false;
    if ((DebugMode==3)&&(SelStack.size()==1) )res=true;
    if (DebugMode==0)res=true;
    return res;}
//-----
-----
void CConstrtuctorView::OnLeftDblClick()
{
    if (MouseOnObject())
    {
        CConstrtuctorDoc * m_pDoc=GetDocument();
        switch (type_selected)
        {

```

```

        case TUBE: {AboutElem((CTubeAbout*)NULL);break;}
        case BATTERY: {AboutElem((CBatteryAbout*)NULL);break;}
        case TURN: {AboutElem((CTurnAbout*)NULL);break;}
        case TRINITY: {AboutElem((CTrinityAbout*)NULL);break;}
        case VENTIL: {AboutElem((CVentilAbout *)NULL);break;}
        case NASOS: {AboutElem((CNasosAbout *)NULL);break;}
        case DROSSEL: {AboutElem((CDrosselAbout *)NULL);break;}
        case AIRBATTERY: {AboutElem((CVentilatorAbout *)NULL);break;}
        case VODOBATTERY: {AboutElem((CVodovodoAbout *)NULL);break;}
        case SOPLO: {AboutElem((CSoploAbout *)NULL);break;}
        case CONNECTOR: {AboutElem((CConnectorAbout *)NULL);break;}
        case KRAN: {AboutElem((CVentilAbout *)NULL);break;}
        case LABEL: {AboutElem((CLabelAbout *)NULL);break;}
        case EXTERM: {AboutElem((CExtTermoAbout *)NULL);break;}
        case SOPLO_OBJECT: {AboutElem((CSoploObjAbout *)NULL);break;}
    //-----
    -----
template <class A>
void CConstrtuctorView::AboutElem(A *ab)
{
    CConstrtuctorDoc * m_pDoc=GetDocument();
    if (m_pDoc==NULL) return;
    ELEMENT_DATA * str=&objects[duo_selected.place.Type];
    CTempl * ActiveItr=m_pDoc->GetTempl(duo_selected);
    ActiveItr->GetData(str);
    A about(str);
    if (about.DoModal()==IDOK)
    {
        element=-1;
        ActiveItr->Update(str);
        m_pDoc->Analyse(ActiveItr);
        CompileList();
        element=0;    }
    //-----
    -----
void CConstrtuctorView::OnLButtonUp(UINT nFlags, CPoint point)
{
    CConstrtuctorDoc * m_pDoc=GetDocument();
    if (m_pDoc==NULL) return;
    ReleaseCapture();
    switch (DebugMode)
    {
        case 0://режим Обычный (расстановка компонентов)
            {
                EndDragText();
                AddElement();
                break;    }
        case 1://режим Масштабированя ZOOM_IN
            {OnZoomInMouseUP(); break;}
        case 2://режим Масштабированя ZOOM_OUT
            {ZoomOut(point);break;}
        case 3://режим выделения и перетаскивания
            {OnSelectMouseUP(); break;}

        CCustomView::OnLButtonUp(nFlags, point);}
    //-----
    -----
void CConstrtuctorView::DeleteObject()
{
    CConstrtuctorDoc * m_pDoc=GetDocument();
    if (m_pDoc==NULL) return;
    if (DebugMode!=3) return;

```



```

CString Message="Вы точно хотите удалить выделенные объекты?";
if (AfxMessageBox(Message,MB_YESNO|MB_DEFBUTTON2)==IDYES)
{
    StopMovie();
    PAIR::iterator itr;
    PAIR::iterator follower_itr;
    for(itr=SelStack.begin();itr!=SelStack.end();itr++)
    {m_pDoc->DeleteElement(*itr,true);

    for(follower_itr=itr;follower_itr!=SelStack.end();follower_itr++)
        {if (follower_itr->place.Type==itr->place.Type)
            {
                if (follower_itr->place.Number>itr-
>place.Number) follower_itr->place.Number--;}

        SelStack.clear();
        type_selected=NONE;
        CompileList();
        OnKeyDown(VK_ESCAPE,0,0);
        StartMovie();}
//-----
}

bool CConsrtuctorView::CanAcsessObject()
{
    return((type_selected!=NONE)&&(element==0));
}//-----

void CConsrtuctorView::PasteObject()
{
    CConsrtuctorDoc * m_pDoc=GetDocument();
    if (m_pDoc==NULL) return;
    if (DebugMode!=3) return;
    StopMovie();
    PAIR::iterator itr;
    PAIR followet_SelStack;
    followet_SelStack.clear();
    for(itr=SelStack.begin();itr!=SelStack.end();itr++)
    {
        CDuo dd=*itr;
        m_pDoc->PasteElement(dd,true);
        followet_SelStack.push_back(dd);}
    SelStack.clear();
    SelStack.insert(SelStack.begin(),followet_SelStack.begin(),followet_
SelStack.end());
    followet_SelStack.clear();
    CompileList();
    StartMovie();}
//-----

void CConsrtuctorView::ChangeWinParam()
{
    CWinParam winP(&sc,this);
    if (winP.DoModal()==IDOK)
    {
        sc=winP.GetScreenStruct();
        UpdateScrollSizes();
        GetClientRect(&rectClient);
        sc.Wrect.height=(rectClient.bottom)*2*sc.YL2/(float)sc.CY;
        sc.Wrect.width=(rectClient.right)*2*sc.XL2/(float)sc.CX;
        sc.Wrect.top=sc.Wrect.left=0;
        origin=0;
}

```

```

        InitDisplay();    }
//-----
-----
#define zedd 2
void CConstrtuctorView::DrawVectors()
{
    CConstrtuctorDoc * m_pDoc=GetDocument();
    if (m_pDoc==NULL) return;
    UINT ext,inter;
    ext=0;
    inter=0;
    CDuo duoDraw;
    CTempl * eleDraw;
    VECTORXY first;
    VECTORXY last;
    VECTORXY lenth;
    lenth.X=2;
    lenth.Y=0.5;
    GLdouble params[4];
    glGetDoublev(GL_CURRENT_COLOR,params);
    VECTOR_OF_PAIR::const_iterator PairsV_itr;
    for (PairsV_itr=m_pDoc->PairsV.begin();PairsV_itr!=m_pDoc-
>PairsV.end();++PairsV_itr)
    {
        double R,G,B;
        double col;
        if (ext==0)col=0;
        else col=ext/( (m_pDoc->PairsV.size()-1)*1.0 );
        if (col<0.5) R=1-2*col;
        else R=0;
        if (col<0.5) G=2*col;
        else G=2*(1-col);
        if (col<0.5) B=0;
        else B=2*(col-0.5);
        glColor3d(R,G,B);
        PAIR::const_iterator p_itr;
        for (p_itr=PairsV_itr->begin();p_itr!=PairsV_itr->end();++p_itr)
        {
            duoDraw=(*p_itr);
            eleDraw=m_pDoc->GetTempl(duoDraw);
            if (eleDraw==NULL)continue; //return;
            CNode nodeFirst,nodeSecond;
            bool Found=false;
            UINT gl_drawmode=GL_TRIANGLES;
            if ((duoDraw.first>=0)&&(duoDraw.second>=0))
            {
                nodeFirst=eleDraw->GetNode(duoDraw.first);
                nodeSecond=eleDraw->GetNode(duoDraw.second);
                Found=true;}
            else
            {if ( (duoDraw.first==-2)&&(duoDraw.second>=0) )
                {if (m_pDoc-
>GetSecondPin(eleDraw,duoDraw.second)<0) Found=false;
                else
                {nodeFirst=eleDraw->GetNode(m_pDoc-
>GetSecondPin(eleDraw,duoDraw.second) );
                    nodeSecond=eleDraw->GetNode(duoDraw.second);
                    gl_drawmode=GL_LINE_STRIP;
                    Found=true;}}

```

```

        else
        {if ( (duoDraw.second===-2)&&(duoDraw.first>=0) )
            {if ( m_pDoc-
>GetSecondPin(eleDraw,duoDraw.first)<0 ) Found=false;
                else
                {
                    nodeFirst=eleDraw->GetNode(duoDraw.first);
                    nodeSecond=eleDraw->GetNode(m_pDoc-
>GetSecondPin(eleDraw,duoDraw.first));
                    gl_drawmode=GL_LINE_STRIP;
                    Found=true; }
                }
            if (Found)
            {
                first.X=nodeFirst.X;
                first.Y=nodeFirst.Y;
                last.X=nodeSecond.X;
                last.Y=nodeSecond.Y;
                Get2Points(first,last,lenth);
                glBegin(GL_LINES);
                glVertex3d(nodeFirst.X,nodeFirst.Y,zedd);
                glVertex3d(nodeSecond.X,nodeSecond.Y,zedd);
                glEnd();
                glBegin(gl_drawmode);
                glVertex3d(first.X,first.Y,zedd);
                glVertex3d(nodeSecond.X,nodeSecond.Y,zedd);
                glVertex3d(last.X,last.Y,zedd);
                glEnd(); }
            inter++; }
        ext++;}
        glColor3dv(params); }
//-----
-----
void CConsrtuctorView::OnProperties()
{ OnLeftDb1Click(); }
//-----
-----
LRESULT CConsrtuctorView::OnAction(WPARAM wParam, LPARAM lParam)
{ CConsrtuctorDoc * m_pDoc=GetDocument();
    if (m_pDoc==NULL) return FALSE;
    return m_pDoc->OnAction(wParam,lParam); }

```

ПРИЛОЖЕНИЕ Б

ТОО «МВ Инжиниринг»,
г.Караганда,
ул.Лободы 46/2, оф.13

АКТ

О практическом использовании результатов диссертационной работы САГЫНГАНОВОЙ И.К. на тему «Автоматизация теплового пункта с использованием конвейерной обработки данных»

Разработанные в диссертационной работе Сагынгановой И.К. информационные модели, основанные на автоматизации системы управления тепловыми пунктами с использованием конвейерной обработки данных, нашли свое отражение в созданном Сагынгановой И.К. программном обеспечении – программе для ЭВМ «Конструктор» для создания и редактирования схем теплоснабжения тепловых пунктов с возможностью считывания реальных данных, и имеющем следующие функциональные возможности:

- создание и редактирование схемы теплового пункта теплоснабжения;
- отображение на схеме реальных измеренных значений показаний приборов учета теплоснабжения;
- кодирование приборов учета теплоснабжения на сохраненных схемах с созданием базы приборов для интеграции со сторонней программной оболочкой;
- работа в качестве независимого приложения для изучений вариантов схем теплоснабжения тепловых пунктов;

прошли успешную апробацию на объектах, оснащенных АТП (автоматизированными тепловыми пунктами), реализованные компанией МВ «Инжиниринг», занимающейся комплексными инженерными решениями для систем тепло- и водоснабжения. При этом обеспечивалось снижение временных затрат на разработку технической документации при реализации проектов АТП, повысилась эффективность контроля и учета тепловой энергии на теплообеспечивающих предприятиях города Караганды.

Директор
ТОО «МВ Инжиниринг»



Михалченко А.В.

Рисунок Б.1- Акт внедрения

ПРИЛОЖЕНИЕ В



Рисунок В.1 - Диплом