

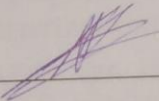
Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ»

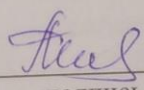
Кафедра «Компьютерные технологии»
Специальность 6М070400 – Вычислительная техника и
программное обеспечение

Допущен к защите:
зав. кафедрой «Компьютерные технологии»
д. ф. –м. н., профессор _____ Куралбаев З.К.
« _____ » _____ 2016 г.

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ пояснительная записка

Разработка интерфейса для динамического моделирования
многоступенчатого газового компрессора

Магистрант _____  _____
подпись (Ф.И.О.)

Руководитель диссертации _____  _____
подпись (Ф.И.О.)

Рецензент _____
подпись (Ф.И.О.)

Алматы, 2016 г.

Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ»

Факультет: «Аэрокосмические и информационные технологии»
Специальность: 6М070400 – Вычислительная техника и программное обеспечение
Кафедра: «Компьютерные технологии»

ЗАДАНИЕ

на выполнение магистерской диссертации

Магистранту Кузуримову Айдару Руслановичу
(фамилия, имя, отчество)

Тема диссертации Разработка интерфейса для динамического моделирования многоступенчатого газового компрессора
утверждена Ученым советом университета № от « »
Срок сдачи законченной диссертации « »

Цель исследования Разработка приложения (модуля) для динамического создания схем с 1,2,3 ступенчатыми компрессорами. Приложение позволит строить модели мало пожиматных в работе схем с 1,2,3 ступенями, а также поможет проводить эксперименты без разрушения настоящих компрессоров

Перечень подлежащих разработке в магистерской диссертации вопросов или краткое содержание магистерской диссертации:

Анализ существующих программ (комплексов для моделирования); Описание задачи для разработки; Описание полученных результатов после разработки приложения; Проведение тестов, сравнение с реальной моделью компрессора а также существующего оборудования; расчет экономических части; итд.

Перечень графического материала (с точным указанием обязательных чертежей)

Презентации (слайдов Power Point).

Рекомендуемая основная литература

- 1) Развитие теории и применение, Ковицкий М.И., Сеткова Е.В. - Новосибирск 2000
- 2) Johan Liedman, Robert Marnon, Dynamic simulation of a centrifugal compressor system
- 3) Теория гидравлических цепей Меренков А.П., Хасиев В.Я. Москва: Наука
- 4) Основы моделирования на VisuaBasic.Net, Никита Кузьмин, БХВ - Петербург 2009

ГРАФИК
подготовки магистерской диссертации

Наименование разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю	Примечание
1. Библиографический обзор по теме диссертации	25.04.15	
2. Анализ существующих методов моделирования и обзор параметров существующих параметров (объектов)	29.06.15	
3. Обзор методов программирования для разработки алгоритмов моделирования	29.06.15	
4. Разработка приложения	7.07.15	
5. Проведение тестов в сравнении с реальными параметрами	10.10.15	
5.1. Динамический запуск	5.11.15	
5.2. Статический анализ	10.11.15	
5.3. Реакция на внешние факторы	15.11.15	
6. Написание МД	1.12.15	
7. Подготовка презентации	5.01.16	
8. Подготовка к защите	15.01.16	

Дата выдачи задания _____

Заведующий

кафедрой _____

(подпись)

(Ф.И.О.)

Руководитель

диссертации _____

(подпись)

(Ф.И.О.)

Задание принял к исполнению

магистрант _____

(подпись)

(Кудряшев А.Р.)

(Ф.И.О.)

АННОТАЦИЯ

Диссертационная работа содержит 82 страниц машинописного текста, 5 таблицу, 60 рисунков, список использованных источников – 29 наименований.

АВТОМАТИЗАЦИЯ, ИЗМЕРЕНИЕ, РЕГУЛИРОВАНИЕ, УПРАВЛЕНИЕ, ТЕМПЕРАТУРА, ГАЗОВЫЙ КОМПРЕССОР, ГАЗО-ЖИДКОСТНЫЙ СЕПАРАТОР, ОХЛАДИТЕЛЬ, ГАЗОВЫЙ КЛАПАН, ГАЗОВАЯ ТРУБА, АНТИПОМПАЖНОЕ РЕГУЛИРОВАНИЕ, РАСХОД ГАЗА.

Диссертационная работа содержит описание приложения для динамического построения 1,2,3 ступенчатых компрессоров разных типов схем, для построения нужной схемы по сжатию газа с соответствующим оборудованием. Моделирование позволяет оценить реальные процессы в таких оборудоваиях как газовый компрессор, газо-жидкостный охладитель, газо-жидкостный сепаратор, газовый клапан, труба. Представленные расчеты основаны на реальных данных и показывают практическую значимость полученных результатов. Каждое из оборудований описывается математической моделью с соответствующими формулами, которые приводят модель к реальному технологическому процессу.

Предлагаемая имитационная модель учитывает факторы, влияющие на процесс реального сжатия газа в газовых компрессорах. Логика алгоритмов моделирования адекватно отражает реальные технологические процессы в представленных оборудованиях.

Цель проекта – разработка динамической системы моделирования для разных количеств ступеней сжатия газа со всем сопутствующим оборудованием и с разными схемами подключения.

Область применения: внедрение проекта возможно в обучающих целях для операторов по системе управления компрессорной станцией, также данная модель позволяет смоделировать работу реального компрессора при загрузке в модель паспортных данных выбранного типа компрессора и сопутствующего оборудования (сепаратор, охладитель, клапан и т.д.).

Диссертациялық

29 атаулары - ТӨЖ Машина басу, 5 үстел, 60 қайраткерлері, пайдаланылған әдебиеттер тізімінен 82 беттен тұрады.

Автоматтандыру, өлшеу, бақылау, пайдалану, температура, газ компрессорлары, газ-сұйық сепараторы, салқын, газ клапандар, газ құбырлары, бақылау асқын, газ ағыны.

Диссертациялық жұмыс қарқынды тиісті жабдықтармен газ қажетті қолмен қысу салу үшін схемаларын әр түрлі 1,2,3 кезеңі компрессорлар, салу қолданбаның сипаттамасын қамтиды. Модельдеу бізге осындай газ сияқты нақты процестер компрессор, газ-сұйық суыту, газ-сұйық сепараторы, газ клапаны, құбыр жабдықталған бағалауға мүмкіндік береді. ұсынылған есептеулер нақты деректерге негізделген және нәтижелерін практикалық маңыздылығын көрсетіп берді. Жабдықтарды Әрбір технологиялық процестің нақты моделін әкелуі тиісті формулалар бар математикалық моделі сипатталған.

Ұсынылып отырған имитациялық моделін факторлар нақты газ сығымдау газ компрессорлар процесіне әсер ескереді. Логика модельдеу алгоритмдері лайықты ұсынылған жабдықтарды нақты процестерді көрсетеді.

Жобаның мақсаты - барлық байланысты жабдықтармен және түрлі байланыс диаграммалары газ сығымдау кезеңдерінде түрлі сомалар үшін динамикалық жүйесі модельдеу дамыту.

Қолданылу саласы: компрессор станциясының басқару жүйесін операторлары үшін жоба ықтимал оқыту жүзеге асыру, бұл модель таңдалған түрі компрессордың және тиісті жабдықтар (бөлгіш, салқын, клапандар және т.б.) моделі төлқұжат деректерге жүктелген кезде нақты компрессор модельдеуге мүмкіндік береді ретінде.

ANNOTATION

Dissertation work contains 82 pages of the typewritten text, 5 table, 60 drawings, the list of the used sources – 29 names.

AUTOMATION, MEASUREMENT, REGULATION, MANAGEMENT, TEMPERATURE, GAS COMPRESSOR, GAZO-ZHIDKOSTNYY SEPARATOR, COOLER, GAS VALVE, GAS PIPE, ANTIPOMPAZHNY REGULATION, GAS CONSUMPTION.

Dissertation work contains the description of the application for dynamic creation of 1,2,3 step compressors of different types of schemes, for creation of the necessary scheme on compression of gas with the corresponding equipment. Modeling allows to estimate real processes in such oborudovaiya as the gas compressor, a gazo-liquid cooler, a gazo-liquid separator, the gas valve, a pipe. The presented calculations are based on real yielded and show the practical importance of the received results. Each of oborudovaniye is described by mathematical model with the corresponding formulas which lead model to real technological process.

The offered imitating model considers the factors influencing process of real compression of gas in gas compressors. The logic of algorithms of modeling adequately reflects real technological processes in presented the oborudovaniyakh. The project purpose – development of dynamic system of modeling for different quantities of steps of compression of gas with all accompanying equipment and with different schemes of connection.

Scope: introduction of the project possibly in the training purposes for operators on a control system of compressor station, also this model allows to simulate operation of the real compressor when loading in model of passport data of the chosen type of the compressor and the accompanying equipment (a separator, a cooler, the valve, etc.).

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	9
Глава 1. Разработка общего алгоритма и описаний к программным и техническим средствам	10
1.1 Условные обозначения	10
1.2 Разработка общего алгоритма решения задачи	10
1.2.1 Разработка математических моделей	
1.2.2 Разработка платформы	12
1.3 Выбор программных средств решения задач	13
Глава 2. Обзор систем моделирования и инженерных расчетов	15
2.1 Обзор системы моделирования Hysys	15
2.2 Анализ работы платформ моделирования	19
Глава 3. Обзор платформы моделирования, принципы работы элементов моделирования и их настройка	21
3.1 Кнопки	21
3.2 Объекты моделирования	22
3.2.2 Компрессора	22
3.2.3 Газо-жидкостный фильтр	24
3.2.4 Теплообменник	25
3.2.5 Клапана	25
3.2.5.1 Обычный клапан	26
3.2.5.2 Антипомпажный клапан	27
3.2.6 Мультиплексор (миксер)	28
3.2.7 Разделитель (узел распределения)	29
3.2.8 Газовые трубы	29
3.2.9 Источник газового потока	30
3.2.10 Двигатель компрессора	31
3.3 Панель меню	32
3.3.1 Меню «Файл»	32
3.3.2 Меню «Редактировать»	33
3.3.3 Меню «Опции»	33
3.4 Окно антипомпажного управления	34
3.4.1 Кнопки управления	35
3.4.2 Дискретные сигналы процесса	36
3.4.3 Аналоговые сигналы процесса	37
3.4.3.1 Аналоговые входные сигналы процесса	37
3.4.3.2 Сигналы аналогового расчета антипомпажного процесса	38
3.4.3.3 Антипомпажный клапан PV/SP	38
3.4.4 Диалоговое окно диаграммы помпажных линий	39
3.4.5 Антипомпажные линии	39
Глава 4. Последовательность запуска приложения и работа с ним	41
4.1 Создание нового проекта. Модель одноступенчатого компрессора	41

4.1.1 Открыть приложение	41
4.1.2 Создание нового проекта	41
4.1.3 Элементы подключения	42
4.1.4 Перейти в режим выполнения	46
4.2 Сохранение и открытие созданной структуры	48
Глава 5. Проверка адекватности ПО реальными данными	49
5.1 Динамический режим: Запуск компрессора	51
5.2 Стационарный режим	53
5.3 Динамический режим: Останов компрессора	54
5.4 Проверка реакции системы на внешние воздействия и изменения параметров системы	56
Глава 6. Экономическая часть	58
6.1 Расчет трудоемкости программного продукта	58
6.2 Расчет численности работников и затрат на основную и дополнительную заработную плату	59
6.3 Расчет эксплуатационных расходов	61
6.4 Составление схемы затрат на разработку программного обеспечения	64
6.5 Расчет отпускной цены	64
6.6 Расчет технико-экономических показателей	65
ЗАКЛЮЧЕНИЕ	68
СПИСОК ЛИТЕРАТУРЫ	69
ПРИЛОЖЕНИЕ А	70

ВВЕДЕНИЕ

Компьютерные комплексы моделирования режимов работы газового компрессора являются общепринятым компонентом диспетчерских систем управления и представляют одну из базовых компонент комплексной автоматизации газотранспортных предприятий. Внедрение информационных технологий и автоматизация диспетчерского управления создает большие возможности для решения таких задач как оптимизационные задачи, задачи анализа ситуаций и расчета показателей функционирования системы управления.

Целью диссертации является создание платформы которая поможет в общих чертах объяснить студенту или ново принятому сотруднику станции как и по каким принципам работает газовый компрессор, и как управляется АПК(анти-помпажный клапан) в реальных условиях.

Задачи по созданию платформы для моделирования процесса:

- Разработать блоки формул и принцип их обработки
- Разработать принцип работы объектов на данной платформе
- Разработать дружелюбный и простой интерфейс, приближенный к реальным SKADA системам.

Цель исследования газовый центробежный компрессор. Принципы его работы и условия, при которых математическая модель будет работать с максимально приближенными значениями к реальной модели. А также будет работать со всеми элементами установленными в схему и соблюдать все законы физики.

Практическая значимость проекта заключается в том, что эта платформа будет использоваться в настоящем университете в Иране. На кафедре автоматизации, для обучения и как стенд по работе с компрессором и SKADA системой.

1 Разработка общего алгоритма и описаний к программным и техническим средствам

1.1 Условные обозначения

G	Массовый расход газа	[кг/час]
MW	Молекулярная масса газа	[г/моль]
P	Давление газа	[бар]
Z	Степень сжатия газа	[-]
T	Температура газа	[°K]
R	Универсальная газовая постоянная	[дж/(моль*К)]
Q	Объемный расход газа	[м ³ /час]
ρ	Плотность газа	[кг/м ³]
n	Скорость вращения компрессора	[об/мин]
n_0	Номинальная скорость вращения компрессора	[об/мин]
ε_0	Приведенная степень сжатия	[-]
α	Показатель политропы	[-]

1.2 Разработка общего алгоритма решения задачи

1.2.1 Разработка математических моделей

Программное обеспечение представляет собой интерфейс, позволяющий пользователю создавать различные технологические схемы с участием таких элементов как газовый компрессор, охладитель, газожидкостный сепаратор, клапан, антипомпажная линия. Пользователь может выбирать данные элементы из библиотеки и создавать требуемые схемы, соединяя элементы между собой в любой последовательности. Также пользователю предлагается библиотека для выбора компонентного состава газа.

Одним из основных этапов создания программного обеспечения для моделирования технологических процессов является создание математической модели. Важной расчётной частью при моделировании режимов работы газового компрессора является задача потокораспределения. Для решения задач потокораспределения в разработанном ПО применялась модель гидравлических цепей, в основу которых положены законы Кирхгофа [2, 3]. Гидравлическая цепь состоит из узлов и связывающих их ветвей. Узлы это объёмы, в которых соблюдаются законы сохранения массы и энергии, выраженные в форме интегральных уравнений баланса расходов и тепловой энергии. Каждая ветвь цепи характеризуется своим интегральным законом, связывающим перепад давления на данной ветви с расходом газа в ней.

Все данные, необходимые для построения статической и динамической моделей системы должны быть взяты с места установки газового

компрессора. Программное обеспечение включает в себя моделирование следующих оборудования:

- Центробежный компрессор
- Охладитель газа
- Газо-жидкостный сепаратор
- Клапан
- Труба, узел разделения/объединения потоков газа

При построении математической модели процесса система моделирования рассматривает все возможные значения параметров процесса, т.к. точность модели является самым критичным фактором для успешного результата моделирования.

Основными расчётными параметрами системы являются давление, температура, плотность, массовый расход, объёмный расход. Плотность газа рассчитывается согласно формуле 1

$$\rho = \frac{P \cdot 100 \cdot MW}{Z \cdot R \cdot T} \quad (1.1)$$

где значения давления, температуры, степени сжатия, молекулярного веса газа берутся на входе либо на выходе элемента в зависимости от того где рассчитывается плотность.

Массовый расход через элементы схемы зависит от величины пропускной способности и перепада на том или ином элементе и сводится к формуле 2

$$G = k \cdot \sqrt{\Delta P} \quad (1.2)$$

где k это размерность элемента (пропускная способность), в случае клапана k дается в паспортных данных, обычно как C_v , выраженное в USGPM (US Gallon Per Minute).

Объёмный расход рассчитывается как отношение массового расхода к плотности (формула 3).

$$Q = \frac{G}{\rho} \quad (1.3)$$

В разработанном программном обеспечении центральное место уделяется модели газового компрессора. Выходные параметры компрессора рассчитываются в зависимости от входных параметров и от кривых компрессора (кривая зависимости степени сжатия от расхода, кривая зависимости эффективности от расхода). Ниже представлены формулы расчёта выходных значений температуры и давления компрессора.

$$T_{\text{выход}} = \left[1 + \left(\frac{n}{n_0} \right)^2 \cdot \frac{(Z \cdot R \cdot T)_0}{(Z \cdot R \cdot T)_{\text{выход}}} \cdot (\epsilon_0^\alpha - 1) \right] \cdot T_{\text{вход}} \quad (1.4)$$

$$P_{\text{выход}} = \left[1 + \left(\frac{n}{n_0} \right)^2 \cdot \frac{(Z \cdot R \cdot T)_0}{(Z \cdot R \cdot T)_{\text{выход}}} \cdot (\epsilon_0^\alpha - 1) \right]^{\frac{1}{\alpha}} \cdot P_{\text{вход}} \quad (1.5)$$

1.2.2 Разработка платформы

Разработка любой платформы для моделирования – это трудоемкий, длительный и творческий процесс, который не может быть до конца формализован ввиду разнообразия возникающих нестандартных задач. Наиболее эффективным способом разработки сложных систем признано модульное программирование – когда систему разрабатывают по частям, которые называются программными модулями.

Программный модуль - это любой фрагмент описания процесса, оформляемый как самостоятельный программный продукт. Каждый программный модуль создается, компилируется и отлаживается отдельно от других модулей программы, и тем самым, физически разделен с другими модулями программы. Более того, каждый разработанный программный модуль может включаться в состав разных программ, если выполнены условия его использования, декларированные в документации по этому модулю. Таким образом, программный модуль может рассматриваться и как средство борьбы со сложностью программ, и как средство борьбы с дублированием в программировании (т.е. как средство накопления и многократного использования программистских знаний).[19]

Для оценки приемлемости использования программного модуля используются следующие основные критерии:

- размер модуля
- прочность модуля
- сцепление с другими модулями
- рутинность модуля

Размер модуля измеряется числом содержащихся в нем операторов или строк. Модуль не должен быть слишком маленьким или слишком большим. Маленькие модули приводят к громоздкой модульной структуре программы и могут не окупать накладных расходов, связанных с их оформлением. Большие модули неудобны для изучения и изменений, они могут существенно увеличить суммарное время повторных трансляций при отладке программы. Обычно рекомендуются программные модули размером от нескольких десятков до нескольких сотен операторов.

Прочность модуля - это мера его внутренних связей. Чем выше прочность модуля, тем больше связей он может спрятать от внешней по отношению к нему части программы и, следовательно, тем больший вклад в упрощение программы он может внести.

Функционально прочный модуль - это модуль, выполняющий одну какую-либо определенную функцию. При реализации этой функции такой модуль может использовать и другие модули. Такой класс программных модулей рекомендуется для использования.

Информационно прочный модуль - это модуль, выполняющий несколько операций над одной и той же структурой данных (информационным объектом), которая считается неизвестной вне этого

модуля. Для каждой из этих операций в таком модуле имеется свой вход со своей формой обращения к нему. Такой класс следует рассматривать как класс программных модулей с высшей степенью прочности.[10]

Сцепление модуля - это мера его зависимости по данным от других модулей.

Рутинность модуля - это его независимость от предыстории обращений к нему. Модуль будем называть рутинным, если результат обращения к нему зависит только от значений его параметров.

Методы — это процедуры и функции, описанные внутри класса и предназначенные для операций над его полями. В состав класса входит указатель на специальную таблицу, где содержится вся информация, нужная для вызова методов. От обычных процедур и функций методы отличаются тем, что им при вызове передается указатель на тот объект, который их вызвал. Поэтому обрабатываться будут поля именно того объекта, который вызвал метод.

Принципы, составляющие суть объектно-ориентированного программирования.

Существует принципы, составляющие суть объектно-ориентированного программирования. Таких принципов три – инкапсуляция, наследование и полиморфизм.[15]

Для обеспечения надежности нежелателен прямой доступ к полям объекта: чтение и обновление их содержимого должно производиться посредством вызова соответствующих методов. Это правило называется инкапсуляцией.

Вторым принципом ООП является наследование. Этот простой принцип означает, что если вы хотите создать новый класс, лишь немного отличающийся от старого, то совершенно нет необходимости в переписывании заново уже существующих полей и методов. Вы объявляете, что новый класс является потомком или дочерним классом старого класса называемого предком или родительским классом, и добавляете к нему новые поля, методы и свойства — иными словами, то, что нужно при переходе от общего к частному.

Последним принципом является полиморфизм. Чтобы смоделировать некоторую совокупность явлений или процессов средствами ООП, нужно выделить их самые общие, типовые черты. Те из них, которые не изменяют своего содержания, должны быть реализованы в виде статических методов. Те же, которые изменяются при переходе от общего к частному, лучше облечь в форму виртуальных методов. [17]

1.3 Выбор программных средств решения задач

В данной магистерской диссертации в качестве платформы для программирования взята .Net Framework V4. Платформа .NET Framework - это компонент Microsoft Windows, доступный для пользователей Windows через множество каналов распространения, в число которых входят веб-узел центра обновления Майкрософт и центр загрузки Microsoft. Кроме того, этот компонент установлен на некоторые новые компьютеры. [11]

Упрощение установки коммерческих приложений: Многие коммерческие приложения Майкрософт и других производителей требуют наличия .NET Framework для поддержки их основных функций. Если платформа .NET Framework установлена, то эти приложения проще установить.

Упрощение обновления: После установки платформы .NET Framework она автоматически обновляется с помощью службы Microsoft Update. Если вы получаете автоматические обновления через панель управления, то все установленные на компьютере версии .NET Framework будут всегда обновляться последними пакетами обновления.[13]

Реализация дипломного проекта проводится в системе программирования VisualStudio 2010. Уже с более ранних версии система VisualStudio снабжена необходимым набором драйверов для доступа к самым известным форматам баз данных, удобными и развитыми средствами для доступа к информации, расположенной как на локальном диске, так и на удаленном сервере. В поставку продукта входит большое количество коллекций визуальных компонент для построения отображаемых на экране окон, что необходимо для создания удобного интерфейса между пользователем и исполняемым кодом.

В смысле проектирования VisualStudio мало чем отличается от проектирования в интерпретирующей среде, однако после выполнения компиляции мы получаем код, который выполняется в 30-50 раз быстрее, чем тоже самое, сделанное при помощи интерпретатора. В VisualStudio компиляция производится непосредственно в родной машинный код, в то время как существуют компиляторы, превращающие программу в так называемый р-код, который затем интерпретируется виртуальной р-машиной. Это не может не сказаться на фактическом быстродействии готового приложения.[27]

2 Обзор систем моделирования и инженерных расчетов

2.1 Обзор системы моделирования Hysys

Продукты канадской компании Hyprotech Ltd. Hysim позволяет выполнять статическое моделирование практически всех основных процессов газопереработки, нефтепереработки и нефтехимии. Особый акцент сделан на работу с уравнением состояния Пенга-Робинсона. Программа имеет расширенный набор модификаций уравнения состояния Пенга-Робинсона, включающих работу с несимметричными коэффициентами бинарного взаимодействия и различными правилами смещения, модификации для работы с водой, гликолями и аминами. Пакет имеет оригинальный, весьма совершенный алгоритм расчета ректификационных колонн, практически не имеет ограничений в отношении набора задаваемых спецификаций и сложности колонны. Программа имеет табличный ввод данных, по которому затем строится изображение схемы в формате AUTOCAD.[5] Дополнительный пакет Hysprop позволяет эффективно обрабатывать экспериментальные данные по свойствам чистых компонентов и затем использовать полученные корреляции в расчетах. В 1996 году фирма представила новую разработку – Hysys, разработанную специально для 32 разрядных платформ PC/Windows (а ныне – и 64 разрядных) и пользователи Hysim начали переход на Hysys (хотя многие и до сих пор работают в Hysim). В настоящее время вышла версия 7.3 программы. Поскольку в 2002 году компания Hyprotech Ltd была приобретена Aspen Technologies, то в данный момент программный продукт называется Aspen Hysys и входит в состав пакета инженерного модуля AspenONE Engineering пакета AspenONE.

AspenONE. Продукты американской компании Aspen Technologies Inc широко известны в России, в том числе среди студентов химико-технологических специальностей и специалистов нефтегазовой отрасли. Aspen HYSYS является, пожалуй, одним из самых известных продуктов компании.[23] Aspen HYSYS представляет собой программный пакет, предназначенный для моделирования в стационарном режиме, проектирования химико-технологических производств, контроля производительности оборудования, оптимизации и бизнес-планирования в области добычи и переработки углеводородов и нефтехимии. Наряду с возможностью статического моделирования технологических схем, позволяет в той же среде производить динамическое моделирование отдельных процессов и всей технологической цепочки, а также разрабатывать и отлаживать схемы регулирования процессов (рисунок 2.1). Имеется возможность выполнять расчеты основных конструктивных характеристик сепарационного оборудования, емкостей, теплообменной аппаратуры, тарельчатых и насадочных ректификационных колонн и оценку стоимости оборудования. Программа имеет развитый графический интерфейс,

поддерживает технологию OLE-2 и хорошо интегрирована с офисными приложениями Microsoft (подробное описание см. далее).

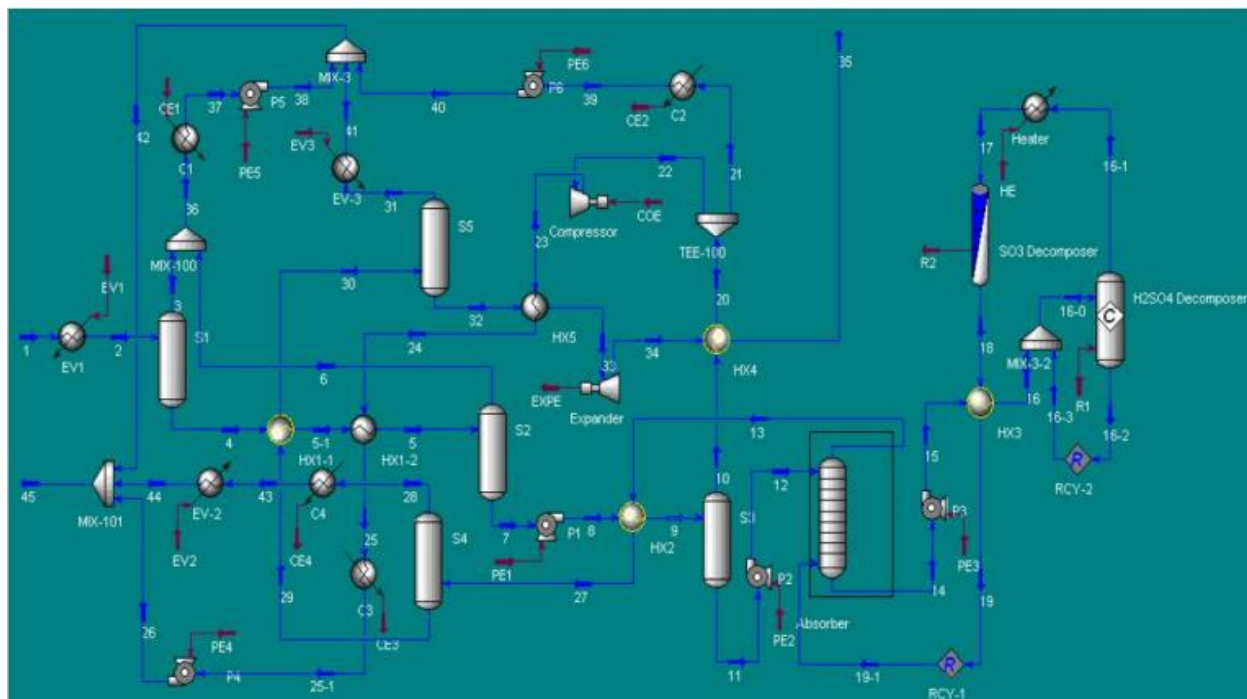


Рисунок 2.1 - Модель в Aspen Hysys

Для моделирования процессов в стационарном режиме помимо HYSYS также служит другой известный продукт компании – Aspen Plus. Этот пакет был разработан для Unix-платформ DEC-alpha и Solaris, являясь, таким образом, высокопроизводительным приложением для рабочих станций. В настоящий момент широко используется на Windows.[6]

Aspen Plus, как уже отмечалось выше, – это система для статического моделирования процессов, основанных на химическом и фазовом превращении. Имеет широкий набор алгоритмов, который постоянно расширяется, благодаря тому, что Aspen Plus является системой с открытыми стандартами. Система имеет развитый графический интерфейс (рисунок 2.2). Имеется возможность выполнять расчеты основных конструктивных характеристик и оценку стоимости оборудования.[26]

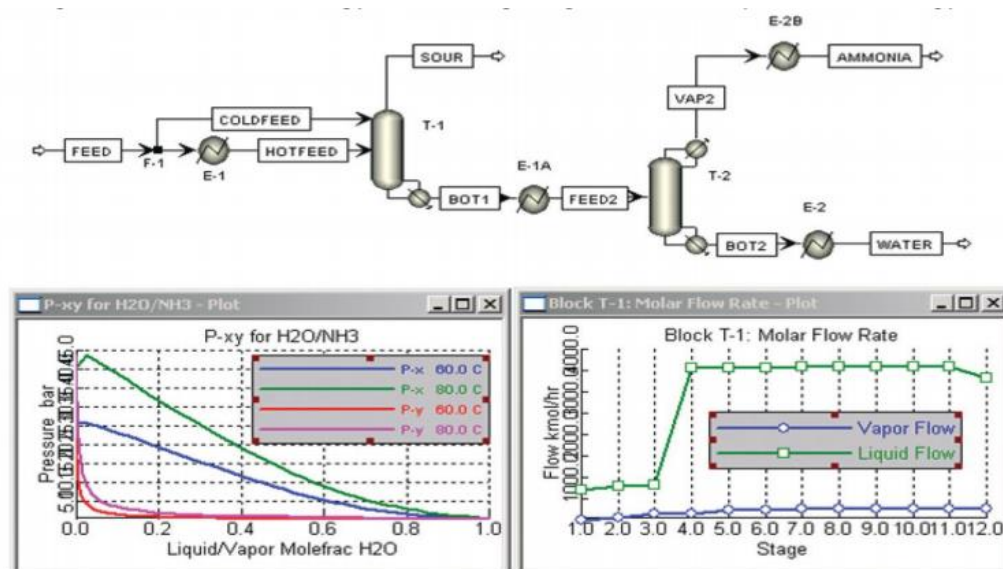


Рисунок 2.2 - Модель процесса в Aspen Plus

В настоящий момент оба этих программных продукта входят в состав пакета AspenONE V7 (текущая версия – 7.3), предназначенного для повышения эффективности производства и управления цепочками поставок, а также проектирования и моделирования технологических процессов. В то же время они продаются и как отдельные ключевые программные продукты компании.[7]

Aspen HYSYS Dynamics Aspen HYSYS Dynamics представляет собой динамическую моделирующую систему для создания нестационарных моделей и моделирования переходных процессов (рисунок 2.3). Это 12 средство для динамических расчетов позволяет также эффективно управлять жизненным циклом технологического процесса. Aspen HYSYS Dynamics позволяет создавать полную модель завода для оценки проекта на предмет рентабельности, управляемости, безопасности, а также для улучшения работы завода в течение всего его жизненного цикла. При использовании динамического моделирования в HYSYS Dynamics достигаются такие цели, как:

- сокращение цикла пуска/наладки и выхода на рабочий режим;
- определение оптимальных стратегий смены сырья;
- настройка регуляторов и определение оптимальных эксплуатационных режимов;
- проверка безопасности работы производства в динамике. Возможно проведение динамического расчета трубопроводов в модуле HYSYS Upstream Dynamics.

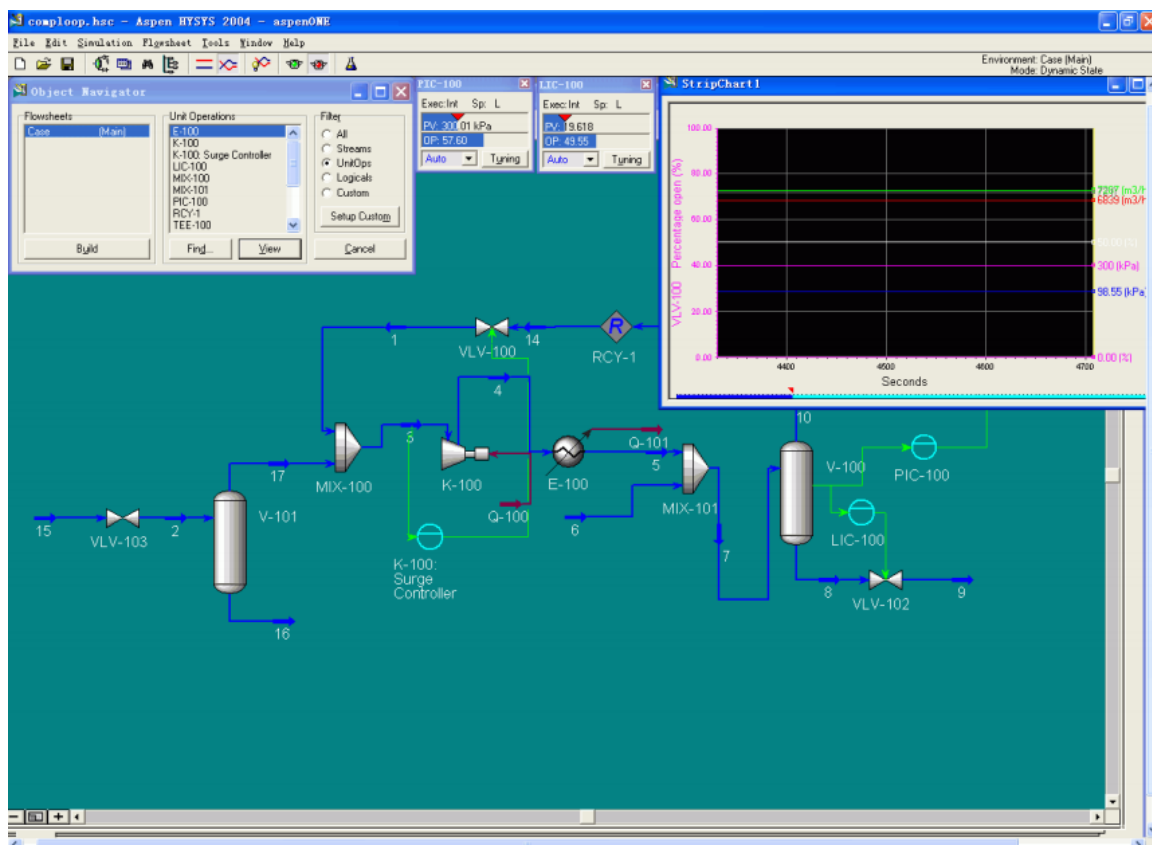


Рисунок 2.3 - Hysys Dynamics

Ключевые особенности HYSYS Dynamics (часть из них пояснены выше при рассмотрении HYSYS)[8]:

- Использование строгих термодинамических моделей.
- Единая среда для моделирования в стационарном и динамическом режимах. При использовании преимуществ интегрированных систем стационарного и динамического моделирования становится достижимым оптимальный баланс между проектными показателями в стационарном режиме и управляемостью установки в динамике.
- Удобный графический интерфейс (PFD – Process Flowsheet Diagram).
- Обширная библиотека модульных операций.
- Расчет свойств потока в динамическом режиме обеспечивает быстрое и точное моделирование поведения системы во времени. С HYSYS Dynamics можно получать результаты в реальном времени для больших моделей. Возможности HYSYS Dynamics значительно расширяются благодаря использованию таких продуктов, как:

- HYSYS Upstream Dynamics: позволяет оценивать поведение трубопроводов с 13 двух и трехфазными потоками на стадии проектирования процесса, когда решения сильно влияют на стоимость проекта. Это может привести к значительной экономии средств и изоляции трубопроводов, увеличить достоверность системного проектирования через уточнение процедур запуска/останова и улучшить безопасность путем моделирования разрыва трубопровода, а также сброса давления.

- HYSYS OLGA Transient: Позволяет использовать программный продукт OLGAS2000 от ScandPower с HYSYS Dynamics, включая динамическое моделирование многофазных потоков нефти, воды и газа в резервуарах и трубопроводах. Это дает возможность инженерам связывать потоки для моделирования сетей трубопроводов, совмещенных с моделированием технологического процесса.
 - Открытая архитектура
 - Расчет пользовательских моделей: для улучшения функциональности модели можно использовать пользовательские модели, созданные с помощью стандартных языков программирования (Microsoft Visual Basic, C++ или FORTRAN). А также использовать программный пакет Aspen Custom Modeler, предназначенный для создания и усовершенствования пользовательских моделей технологических процессов, который можно использовать внутри HYSYS Dynamics.
 - Расчет динамики работы как отдельных аппаратов, так и установки в целом: необходимо для высокопроизводительного функционирования систем обучения операторов, расчета опасных или неприбыльных режимов работы в динамике, осуществления подробных исследований по надежности и эффективности отдельных производственных единиц.
 - Большой набор возможностей задания автоматической системы регулирования: различные регуляторы и логические операции, а также интеграция с системой усовершенствованного регулирования DMCplus.
 - Дополнительные возможности: матрица «причина-следствие» и планировщик событий хорошо подходят для моделирования процедур запуска/останова, а также для автоматизации технологической схемы.
- Основные экономические выгоды использования HYSYS Dynamics:
- Улучшенное проектирование установки: инженеры могут быстро определить наиболее выгодный, надежный и удобный для регулирования проект;
 - Улучшенный контроль установки позволяет исключить проблемы при запуске, возможные отклонения характеристик продукта от заданных спецификаций и незапланированные простои; – повышение рентабельности установки благодаря улучшению эксплуатационных качеств.[9]

2.2 Анализ работы платформ моделирования

Безусловно, у комплекса Hysys множество преимуществ. Но эти преимущества в большинстве случаев используют уже профессионалы или люди хорошо понимающие как этот комплекс работает или работает та или иная система которую они моделируют. Обычным студентам или новичкам на станции будет сложно по ней учиться или понять основные параметры работы компрессора и сопутствующего оборудования.

Для обучения студентов нужно было создать более простую платформу на которой можно было бы создавать схемы с компрессором и следить за изменением параметров и соответственно эти параметры должны были быть максимально приближенные к реальным. После это было решено, что нужно создать платформу с 0 и математическую модель для каждого объекта платформы. Создать интерфейс, и способ обработки каждого объекта установленного в схему и передачу данных между ними. Данные Nysys максимально приближены к реальным данным, а это значит, что проверить работу наших математических моделей можно в нем. И если они сходились, значит все было сделано правильно. Мы нашли тесты проделанные на платформу Nysys и взяли их за основу. Далее в диссертации буду проведены тесты на актуальность нашей платформы и платформы Nysys. Возможно они могут быть не 100% верны так, как в тестах, что уже были проведены не было все параметров данных каждого объекта в схеме и данных, что не было мы заменили на те, что используются на газоперекачивающих станциях.

Что касается анализа платформ для моделирования можно найти много, но они достаточно сложны для обычного студента и стоимость их не оправдывает себя на фоне столь не большой задачи как обучить студентов, настроить понятийный аппарат человека который хочет понять как работает компрессор. Понятно, что посмотреть в реальной жизни студентам на компрессор является немного несбыточной мыслью. Чтобы понять как работает ему надо с чего то начать, и это «чего то» наша платформа. В нашей платформе он сможет понять почему и как работает компрессор и система его управления. Зачем и как работает антипомпажная система и почему без нее компрессор не может работать. Как можно проводить регулирование и управление процессом работы компрессора и его вспомогательного оборудования.

Моделирование систем управления уже развита, сейчас множество компаний ее усовершенствуют и вносят правки для большего КПД в работе управления и физических процессах проходящих на станциях где задействована данная система. В наши интересы не входило использовать дорогие и сложные системы для обучения и настройки понятийного аппарата у студентов или новых сотрудников станций.

Нужность нашей программы для многих университетов должна быть на высоком уровне. Потому что в газонефтяной промышленности всегда нужны хорошие инженеры, которые знают как работает оборудование на станции. И наша главная задача помочь студентам понять как работает система работы газового компрессора. Интерфейс мы решили приблизить максимально к реальным SCADA системам, чтобы при переходе на рабочую станцию оператора людям было уже понятно, что и как работает на новой системе.

3 Обзор платформы для моделирования, принципы работы элементов моделирования и их настройка

Перед нами стояла задача создать продукт который мог просто быть понят студентами а это значило:

- 1) Интерфейс должен быть понятным и максимально приближенным к реальным SCADA системам.
- 2) Большое количество нужных инструментов для детальной схемы газо-перекачки
- 3) Отображение процесса и параметров каждого отдельного объекта схемы.
- 4) Хорошая оптимизация программы для работы на маломощных ПК.
- 5) Удобное и просто управление.

На рисунке 3.1 изображен основной экран программы Compressor Surge loop Simulation. Здесь отображена модель одноступенчатого газового компрессора но можно использовать элементы с панели инструментов для создания любой другой структуры или модели.

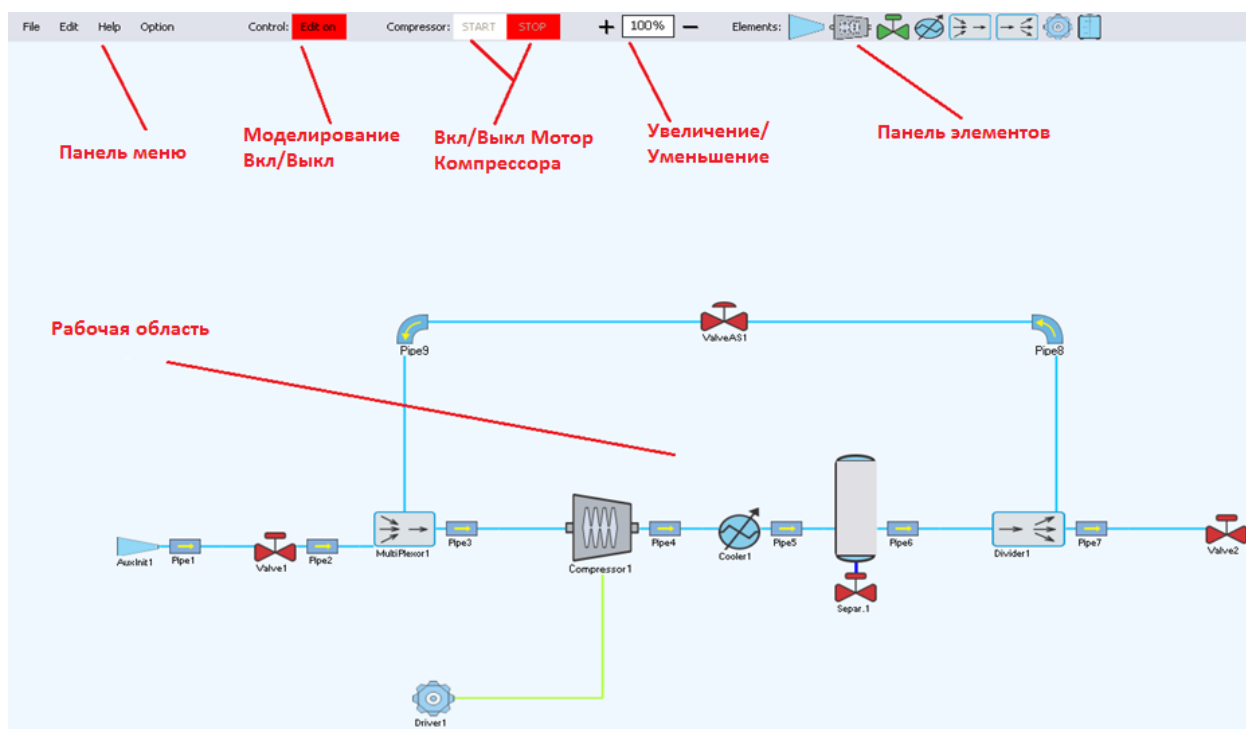


Рисунок 3.1 - Основной экран

3.1 Кнопки

Edit on/off - кнопка используется для запуска и остановки процесса моделирования. Любые изменения в структуре (добавления или удаления объектов) возможно только в состоянии выключенного моделирования.

START

“START” – используется для запуска процесса симуляции. После нажатия на кнопку “START” мотор для компрессора начинает набирать скорость до того момента пока не достигнет уставки скорости.

STOP

“STOP” – используется для остановки процесса симуляции. После как была нажата кнопка “STOP” мотор для компрессора начинает останавливаться до полной остановки.

3.2 Объекты моделирования

Применение приложения состоит из разных объектов которые симулируют реальное оборудование. Основные объекты: *компрессор(compressor)* – это математическая модель реального газового компрессора; *фильтр(separator)* – это математическая модель реального газо-жидкостного(2 ступенчатого) фильтра (конденсатный барабан); *теплообменник (heat exchange)* - это математическая модель реального газового теплообменника (в качестве охлаждающей жидкости использует воду); *Кран(valve)* – это математическая модель реального крана; *двигатель компрессора(compressor driver)*, *Входной поток газа(gas flow input)*, *газовый миксер (gas mixer)* мультиплексор (multiplexor) and *распределитель(tie)(divider)*. На рисунке 3.2 показаны все объекты.

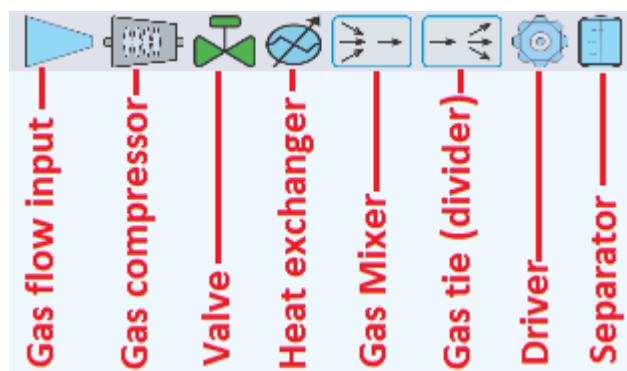
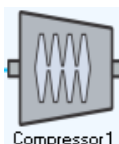


Рисунок 3.2 - Панель элементов

3.2.2 Компрессор



Компрессор, один из основного оборудования в имитируемых модели. Это позволяет построить различные структуры с компрессором: параллельный, последовательный, параллельно последовательный, и т.д. Каждый компрессор имеет собственные параметры, такие как кривой компрессора, номинальная скорость, политропного эффективность,

потенциал. Рисунок 1.3 это диалоговое окно для включения компрессора параметров. Вкладка «Основные параметры» содержит входные и выходные параметры компрессора, вкладка «Газовый состав» показывает текущий состав газа через компрессор. Эти две вкладки являются одинаковыми для всех остальных объектов (же количество и тип параметров).

Для того, чтобы привлечь пользователей кривой компрессора необходимо ввести поток массива и соответствующего Политропный головы; эти параметры будут отличаться для различных типов компрессоров. Эти массивы привлечет красная линия на кривой. Затем специальная функция внутри этого блока будет вычислить полиномиальными коэффициентами (a, b, c, d) функции $dx^3+cx^2+bx^1+a$, которой будет приблизительно красная линия, на кривой светло-голубая линия является функция приближения. На Рисунке 1.3, мы можем видеть, что светло-голубая линия приближается тот красный с высокой точностью. Аппроксимирующей массив коэффициентов [a, b, c, d] отправляет соответствующий компрессор, где с помощью этих коэффициентов вычисляет Политропный пик для текущего объемного потока через компрессор по следующей формуле:

$$Hn=d*(Flow)^3 + c*(Flow)^2 + b*(Flow)^1 + a \quad (3.1)$$

где Hn - Политропный пик, поток – фактический объем потока на входе в компрессор.

Это диалоговое окно также позволяет пользователям сохранить файл («Сохранить как» в меню) массивы кривой компрессора, рекомендуется *.txt формате, и загрузить ранее сохраненные массивы из файла («загрузить» в меню). После « Hn » используется для расчета компрессора в разряд температуры и давления.

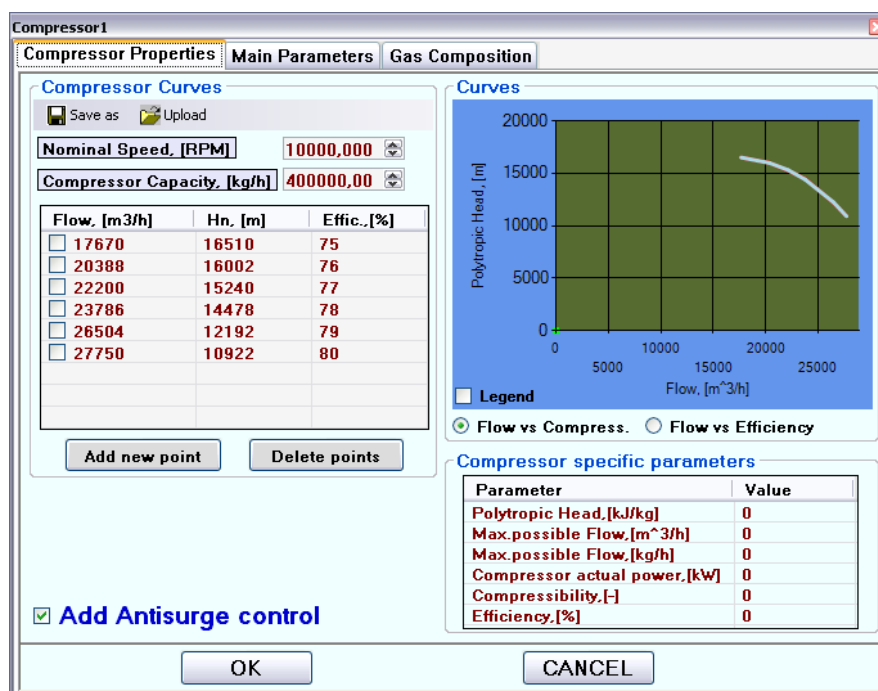


Рисунок 3.3 - Диалоговое окно для компрессора

3.2.3 Газо- жидкостный фильтр



Газ жидкость разделитель используется для разделения газа и водяного пара. В модели, показанной на Рисунке 3.4 используется «разделитель 1» для удаления жидкости при работе компрессора. На Рисунке 3.4 показано диалоговое окно для газо-жидкостного разделителя.

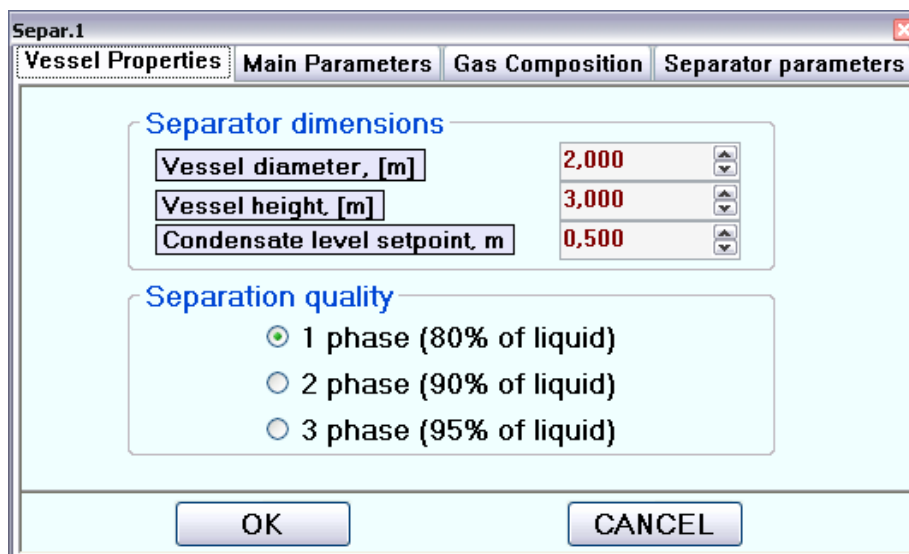


Рисунок 3.4 - Диалоговое окно для фильтра

В диалоговом окне поле пользователь может задать параметры для разделителя: резервуар диаметр, высота резервуара, уставка уровня конденсата, качество очистки. Уровня уставки это сравнивается с текущим уровнем конденсата, если заданное значение больше, чем текущее значение жидкого сливной клапан открывается. Количество разделения фаз является параметром, который имитирует эффективность процесса разделения: «1 этап» – означает, что 80% конденсата будут удалены, «этап 2» - означает, что 90% конденсата будут удалены, «3 этап» - означает, что 95% конденсата

будет удалено. Если состав газа не будет содержать воду, сепаратор конденсата уровень будет неизменным. Вкладка «Параметры разделителя» содержит вычисляемые определенный параметр разделителя.

3.2.4 Теплообменник



Теплообменник используется для охлаждения горячего газа.

В модели, показанной на Рисунке 3.1 используется «кулер 1» для охлаждения газа на части компрессора разряда.

На Рисунке 3.5 показано диалоговое окно для теплообменника.

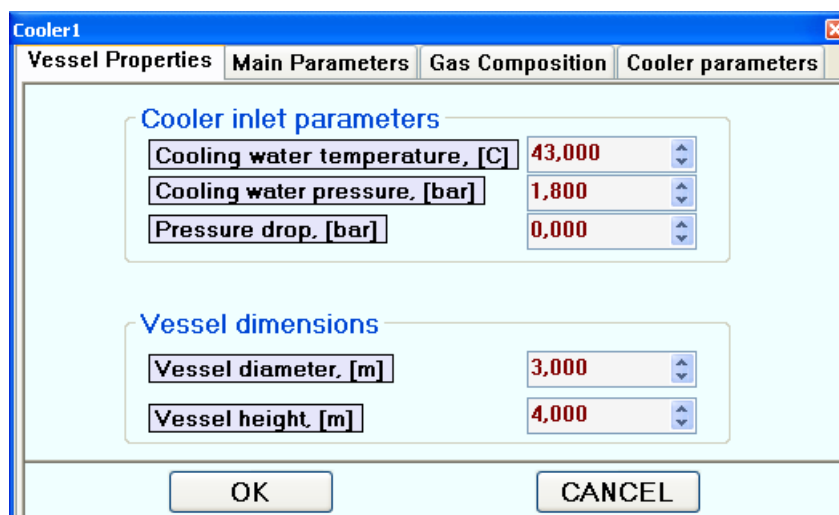
The image shows a software dialog box titled 'Cooler1'. It has four tabs: 'Vessel Properties', 'Main Parameters', 'Gas Composition', and 'Cooler parameters'. The 'Cooler parameters' tab is selected. Inside the dialog, there are two main sections. The first section, 'Cooler inlet parameters', contains three input fields: 'Cooling water temperature, [C]' with a value of 43,000, 'Cooling water pressure, [bar]' with a value of 1,800, and 'Pressure drop, [bar]' with a value of 0,000. The second section, 'Vessel dimensions', contains two input fields: 'Vessel diameter, [m]' with a value of 3,000 and 'Vessel height, [m]' with a value of 4,000. At the bottom of the dialog are 'OK' and 'CANCEL' buttons.

Рисунок 3.5 - Диалоговое окно для теплообменника

В диалоговом окне поле пользователь может установить параметры теплообменника: давление охлаждающей жидкости (bar), температура охлаждающей жидкости (C), падение давления через теплообменник. Для уменьшения температуры газа, существует необходимость увеличить давление воды и уменьшить температуру воды.

3.2.5 Клапана

Приложение содержит два основных типа клапана: обычный клапан и антипомпажный клапан. Общий клапан может быть выбран непосредственно из панели элементов, антипомпажный клапан может быть добавлен из диалогового окна компрессора, позволяя «Добавить контроль антипомпажа». Рисунок 3.6.

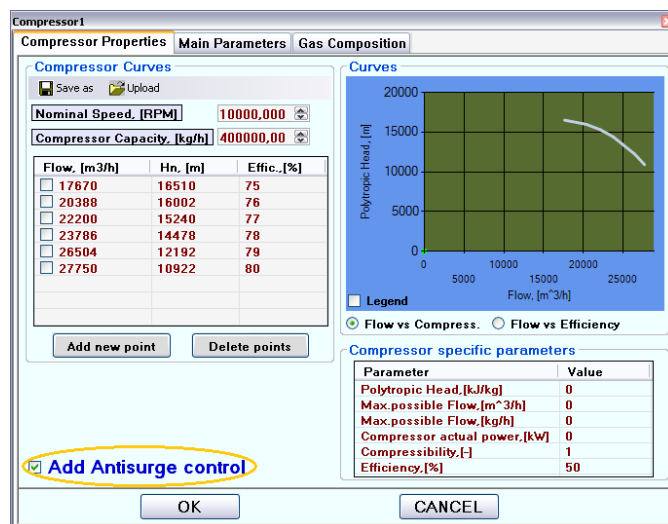


Рисунок 3.6 - Добавление антипомпажного клапана

3.2.5.1 Обычный клапан



Модель для имитации одноступенчатого компрессора (Рисунок 3.1) содержит 2 обычных газовых клапана:

«Valve1» – газовый клапан на входной части компрессора;

«Valve2» – газовый клапан компрессора разгрузочной части;

На Рисунке 3.7 показано диалоговое окно для общих клапанов. В свойствах «клапан» вкладку пользователь может выбрать соответствующие параметры для клапана:

«Значение CV» – значение CV для клапана или мощность клапана в US GPM.

«Значение C1» – C1 значение согласно данным из паспорта клапана.

«Константа падение давления» – падение постоянного давления на клапане [бар].

«Клапан отключения» — если этот флажок установлен, после нажатия кнопки «STOP» вентиль будет идти в полностью закрытую позицию (0%)

«Позиция клапана» – ручной значение для позиции клапана от 0 до 100%

«Время открытия клапана» – время в секундах, необходимого для изменения положения клапана от 0% до 100% или наоборот от 100% до 0%.

«Пропускное давление» – Включение этого флажка позволяет держать разгрузочный клапан давления при постоянном давлении, «Const. press. drop» будет отключена.

«Характеристики потока» – позволяет выбрать характеристики клапана: линейная, быстрое открытие или равнопроцентная (Рисунок 3.8).

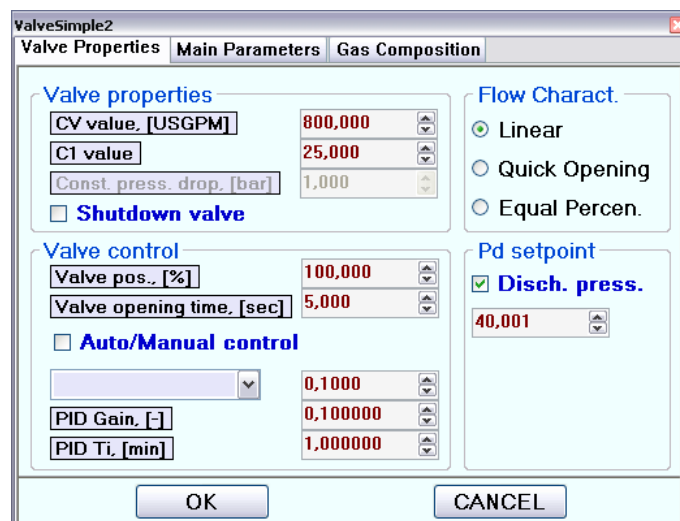


Рисунок 3.7 - Диалоговое окно для клапана

«Авто/ручной контроль» – если этот флажок отключен клапан будет открыт по уставке назначенной в поле «Valve pos.». Если этот флажок включен клапан будет открыт или закрыт для того чтобы держать уставку выходного объемного расхода и температуры, указанной в поле ниже.

«PID Gain» – является пропорциональной увеличением для положения клапана регулятора (применяется, если включен «Авто/ручной контроль»)

«PID Ti» – это коэффициент время интеграции для клапана положение регулятора (применяется, если включен «Авто/ручной контроль»)

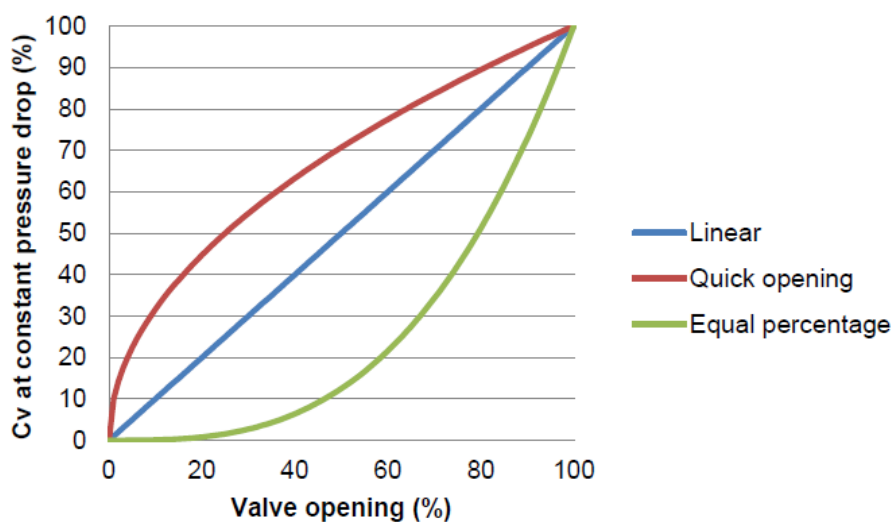


Рисунок 3.8 - Кривые характеристик обычных клапанов

3.2.5.2 Антипомпажный клапан



Приложение для имитации 1 стадии компрессора (Рисунок 3.1) содержит 1 антипомпажный клапан:

«ValveAS1» – антипомпажный клапан используется для предотвращения перенапряжения на «Compressor1»

Диалоговое окно для клапана антипомпажной содержит сведения о составе газа и основных параметров на входе и разгрузки стороны клапана. Значение CV антипомпажного клапана также может быть изменено из диалогового окна. Другие параметры антипомпажного клапана и антипомпажного контроллера будут рассматриваться в главе «Экран антипомпажного управления».

3.2.6 Мультиплексор (миксер)



Мультиплексор имитирует газовый смеситель. Мультиплексор имеет два входа и один выход. Модель для имитации одноступенчатого газового компрессор (Рисунок 3.1) содержит 1 мультиплексор:

«Multiplexer1» – используется для смешивания потоков от входного потока газа и потока газа из газового компрессора антипомпажного клапана. Выходной поток газа идет на вход компрессора.

Давление нагнетания в мультиплексор равны максимальному значению, добавляются объемные потоки, потоки массы добавляются.

На Рисунке 3.9 показано диалоговое окно для мультиплексора. Диалоговое окно содержит только основные параметры и газового состава через входы и выход.

MultiPlexor1			
Main Parameters		Gas Composition	
Parameter name	Input 1	Input 2	Output
Pressure, [bar]	0,65	0,65	0,65
Temperature, [C]	41,7	42,784	42,773
Flow, [m3/h]	0	0,01	0,01
Flow, [kg/h]	0	0,008	0,008
Gas density, [kg/m3]	0,826	0,829	0,829
Average gas molar weight, [kg/kmol]	33,155	33,366	33,364
Critical gas pressure, [bar]	46,836	44,406	44,43
Critical gas temperature, [K]	294,189	289,255	289,304
Compressibility factor, [-]	0,996	0,996	0
Gas constant, [kJ/(kg*K)]	0,251	0,249	0,249
Specific gravity, [-]	1,145	1,152	1,152

Рисунок 3.9 - Диалоговое окно для мультиплексора

3.2.7 Разделитель (узел распределения)



Разделитель имитирует разделение или распределение газа в разные направления. Делитель имеет один вход и 4 выхода. Модель для имитации одноступенчатого газового компрессора (Рисунок 3.1) содержит разделитель:

«Divider1» – делит поток газа на выходе из газового компрессора для двух газовых потоков: газовый поток антипомпажного клапана и поток газа в разгрузочный клапан.

На Рисунке 3.10 показано диалоговое окно для газового разделителя. Диалоговое окно содержит только основные параметры и газового состава через входы и выходы.

Divider1					
Main Parameters Gas Composition					
Parameter name	Input	Out1	Out2	Out3	Out4
Pressure, [bar]	0,6	0,6	0,6	0,6	0,6
Temperature, [C]	42,8	42,8	42,8	42,8	42,8
Flow, [m3/h]	0,6	0	0	0	0
Flow, [kg/h]	0,5	0	0	0	0
Gas density, [kg/m3]	0,8	0,8	0,8	0,8	0,8
Average gas molar weight, [kg/kmol]	33,4	33,4	33,4	33,4	33,4
Critical gas pressure, [bar]	44,4	44,4	44,4	44,4	44,4
Critical gas temperature, [K]	289,3	289,3	289,3	289,3	289,3
Compressibility factor, [-]	1	1	1	1	1
Gas constant, [kJ/(kg*K)]	0,2	0,2	0,2	0,2	0,2
Specific gravity, [-]	1,2	1,2	1,2	1,2	1,2

Рисунок 3.10 - Диалоговое окно для делителя

3.2.8 Газовые трубы



Трубы газовые фактически отсутствует на панели объектов, они добавляется автоматически, когда пользователь подключает элементы друг с другом. Каждая труба имеет диалоговое окно (Рисунок 3.11). В диалоговом окне пользователь может ввести такие параметры, как «Длина трубы», «Диаметр трубы», «Динамическая вязкость». В зависимости от этих параметров выходные данные объемного расхода и давления будут изменяться.

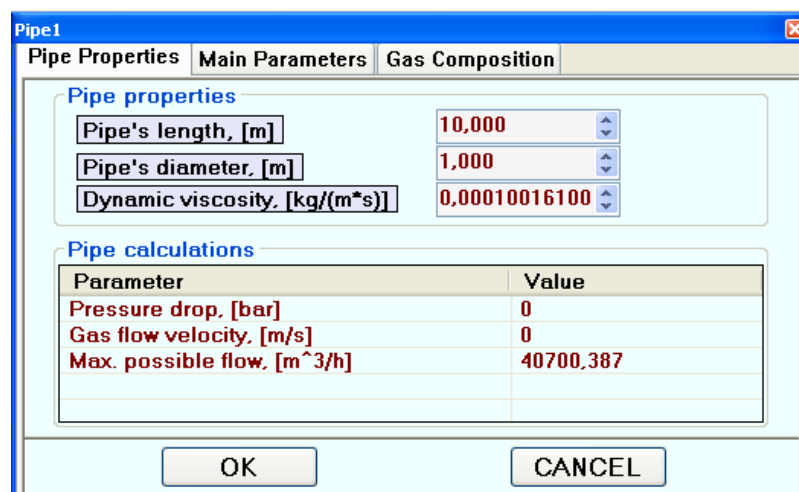


Рисунок 3.11 - Диалоговое окно для газовой трубы

3.2.9 Источник газового потока



«AuxInit» - это элемент, который имитирует источник потока газа с определенными параметрами: давления, температуры, состава. На Рисунке 3.12 показано диалоговое окно для источника потока газа.

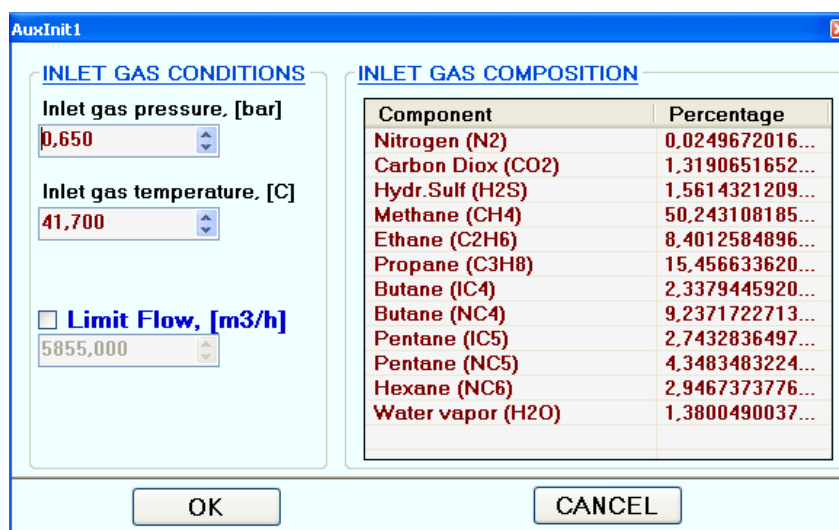


Рисунок 3.12 - Диалоговое окно для элемента источника потока газа

«Давление газа на входе» – давление имитируемого потока газа от источника газа.

«Температура газа на входе» – температура имитируемого потока газа от источника газа.

«Предел потока» – если этот флажок включен, то максимальный объем потока будет ограничиваться полем ниже этого флажка. Если этот флажок отключен, объем потока зависит от минимальной емкости между элементами (газовые трубы, клапаны, компрессоров и т.д.).

«Состав газа в потоке» – список, где пользователь может задать процент каждого газа, дважды щелкнув на столбец процентажа соответствующего газа.

После нажатия на кнопку «ОК» может появиться окно из-за того, что общая сумма компонентов газа не равна 100, диалоговое окно «Нормализации» (Рисунок 3.13), которое после нажатия кнопки «НОРМАЛИЗОВАТЬ» распределяет дополнительную или отсутствующую часть процента состава газа между всех газов согласно текущему проценту газа и выравнивает его на 100%.

Current composition	Normalized composition
2.02496720169535	1.98526196244642
1.31906516529008	1.29320114244125
1.56143212096655	1.53081580486917
50.2431081853629	49.2579492013362
8.40125848965599	8.23652793103528
15.4566336200107	15.1535623725595
2.33794459203025	2.29210254120613
9.23717227137466	9.05605124644574
2.74328364977473	2.68949377428895
4.34834832243074	4.26308659061837
2.94673737768159	2.88895821341332
1.3800490037265	1.35298921933971

TOTAL SUMM 102

RESIDUAL 2.000000000

THE TOTAL SUMM OF INPUT GAS PROPORTIONS IS NOT EQUAL TO 100!

NORMALIZE?

NORMALIZE EXIT

Рисунок 3.13 - Диалоговое окно для нормализации состава газа

3.2.10 Двигатель компрессора



«Двигатель» - элемент используется в качестве мотора для компрессора. В диалоговом окне драйвера можно указать компрессор(компрессора), которые будут использовать выбранный драйвер в качестве мотора для себя. На Рисунке 3.14 показано диалоговое окно драйвера. Элемент «Драйвер» позволяет контролировать параметры как давление, температура, расход на различных элементах (клапаны, охладители, фильтры).

«Выбор элемента» — в этом поле пользователь может выбрать любой элемент выходной параметр (давление, температура, объем потока) согласно которого нужно контролировать скорость.

«Выбранный параметр» – в этом поле пользователь может выбрать контрольный параметр для выбранного элемента.

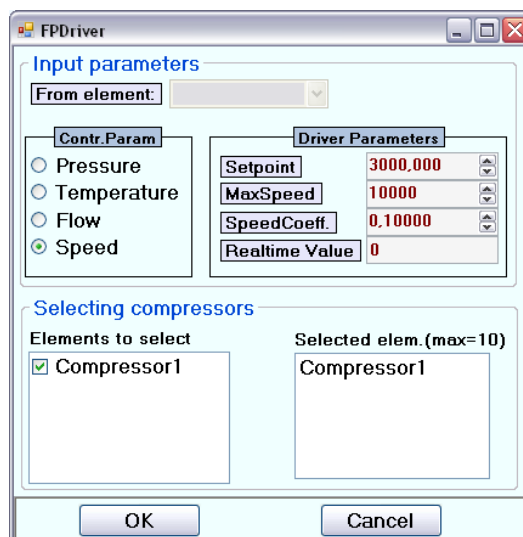


Рисунок 3.14 - Диалоговое окно двигателя

«Уставка» – для параметра управления. Драйвер увеличивает или уменьшает скорость вращения компрессора так чтобы сохранять выбранный параметр максимально близко к уставке.

«Максимальная скорость» – максимально возможная скорость драйвера.

«Коэффициент скорости». Если высокий коэффициент скорости, то драйвер ускоряется быстрее в зависимости от коэффициента.

«Значение в реальном времени» – Текущее значение параметра контроля.

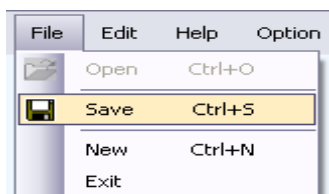
«Элементы для выбора» – в этом списке будут отображены все компрессора, пользователь может подключать драйвер к компрессорам, активировав флажок нужного ему компрессора.

3.3 Панель меню



Меню приложения содержит следующей вкладки: «Файл», «Редактировать», «Справка», «Опции».

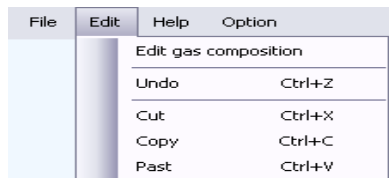
3.3.1 Меню «Файл»



Меню «файл» позволяет создать новый проект кнопкой «New», открыть сохраненный проект с помощью кнопки «Открыть», сохранить текущую версию проекта (текущий набор элементов, значения из всех диалоговых окон и т.д.) с помощью кнопки «Сохранить», выход из приложения кнопкой «Выход».

Приложение позволяет сохранять и открывать проекты в формате «xml», однако у файлов сохраненных нашим приложением своя структура и файлы не соответствующие этой структуре не будут открыты.

3.3.2 Меню «Редактировать»



На вкладка «Изменить композицию газа» позволяет изменять компоненты газа в любое время. После выбора вкладки «Редактировать состав газа» появится диалоговое окно для исправления компонентов газа (Рисунок 3.15). Вернуть на шаг назад, вырезать, копировать, вставить также доступны в этом меню.

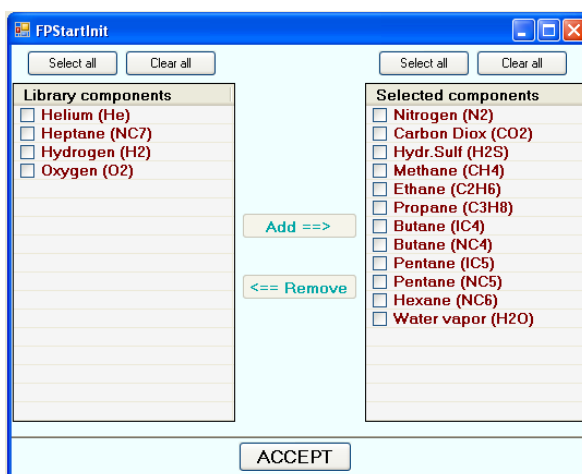
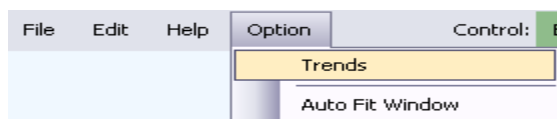


Рисунок 3.15 - Диалоговое окно для выбора компонентов газа

3.3.3 Меню «Опции»



Вкладка «Автоматически выравнивать окно» позволяет равномерно распределить объекты на рабочей области.

Меню «Опции» позволяет открывать тренды вкладкой «Тренды» (Рисунок 3.16). Чтобы добавить параметры для мониторинга тренда выполните следующие шаги:

- 1- Выберите объект, параметры которого будут отслеживаться
- 2- выберите входные, выходные параметры
- 3- Выберите параметры. «_In.*» – означает входные данные, «_Out.*» – означает выходные данные.

- 4- Нажмите кнопку «Добавить». Выбранные параметры будут добавлены в окно выбранные теги (справа). Можно удалить параметры из мониторинга трендов, нажав кнопку «Удалить»
- 5- нажмите кнопку «Согласен». Появится окно трендов (Рисунок 3.17).

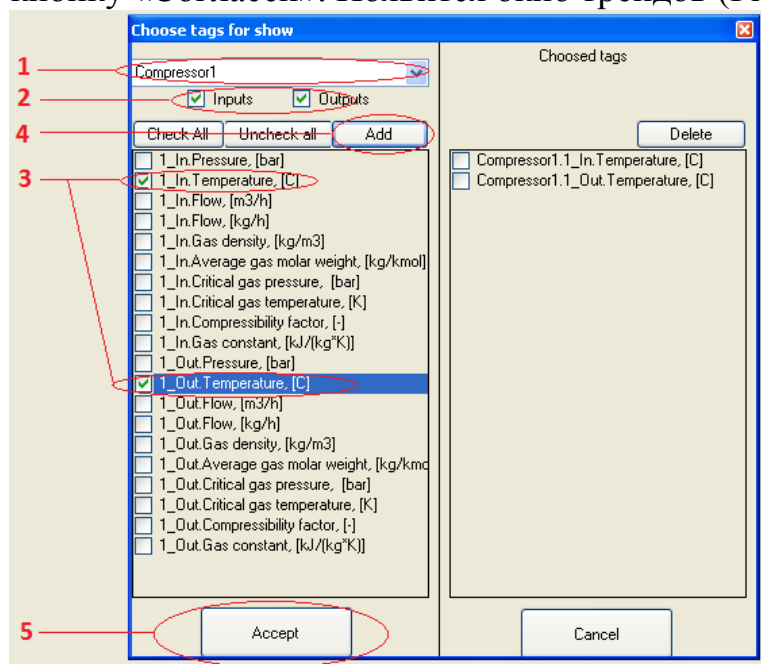


Рисунок 3.16 - Диалоговое окно для выбора параметров для отображения

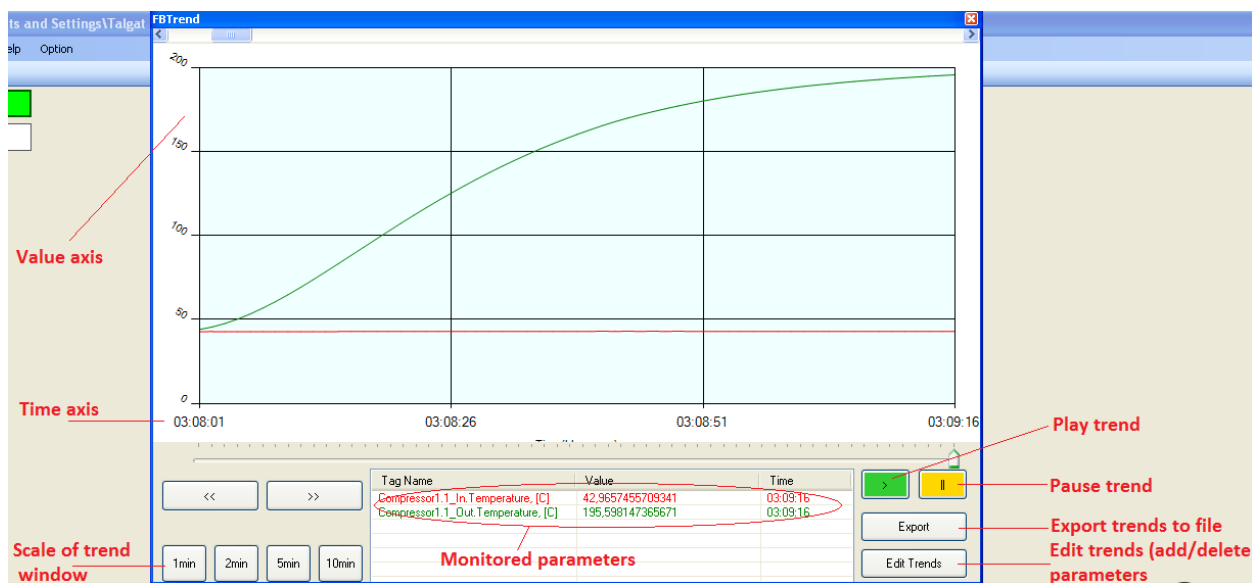


Рисунок 3.17 - Окно трендов

3.4 Окно антипомпажного управления

Каждый компрессор имеет собственный экран управления антипомпажом если включена опция антипомпажного управления (Рисунок 3.18). Антипомпажной управления экран состоит из *кнопок управления, дискретных сигналов процесса, аналоговых сигналов процесса, диалоговые окна, антипомпажные линии*. Пользователь может открыть антипомпажное

управление, правой кнопкой мыши на соответствующий компрессор и выбрав «AntiSurge Face Plate» (Рисунок 3.18).

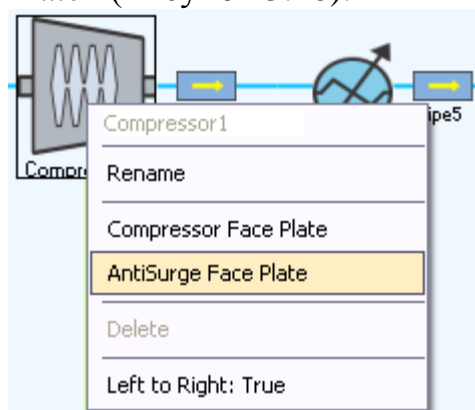


Рисунок 3.18 - Открытие окна антипомпажа

После будет открыто окно антипомпажного управления на выбранном компрессоре. Данное окно отображено на рисунке 3.19.

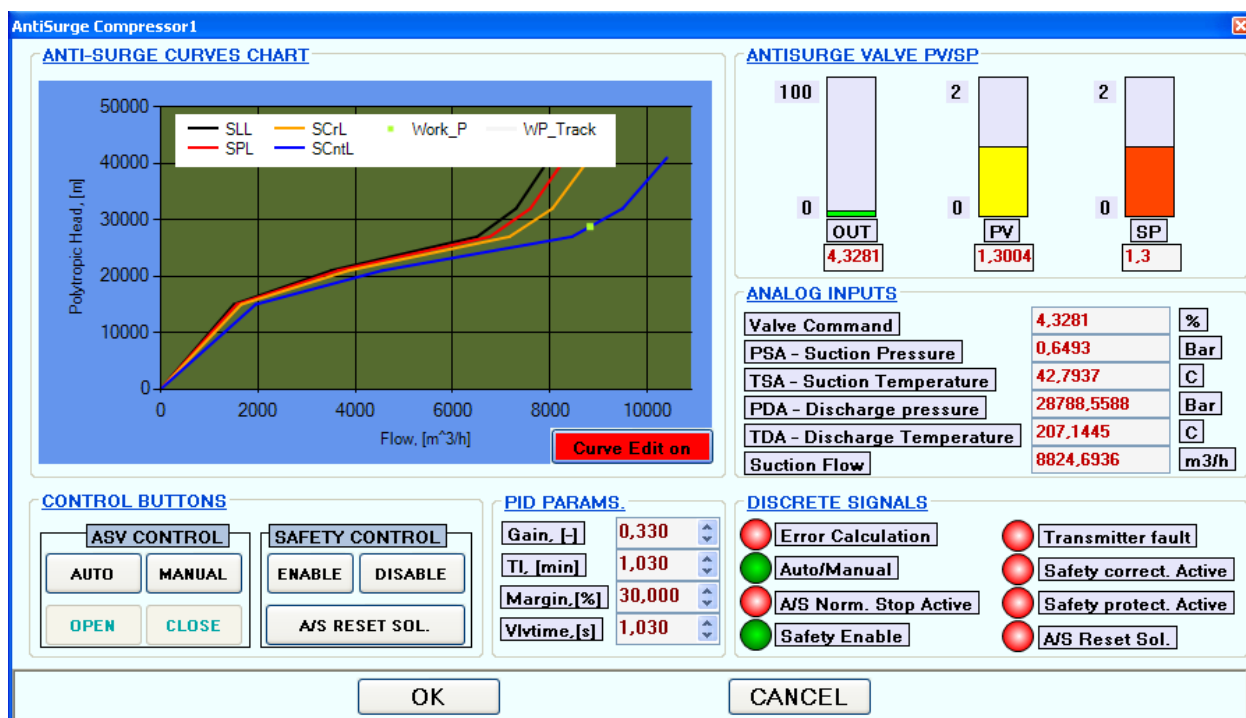


Рисунок 3.19 - Окно управление АПК

3.4.1 Кнопки управления



«Авто», «Ручной» - используются для задания антипомпажного контроллер для автоматического или ручного режима



«Открыть», «Заккрыть» - эти кнопки используются для открытия или закрытия антипомпажного клапана только если антипомпажный контроллер в ручном режиме.



«Включить», «Отключить» - используются для включения или отключения линии защиты от помпажа.



«Сброс антипомпажного соленоида» – эта кнопка используется для сброса электромагнита антипомпажного клапана во время первого запуска приложения или после того, как произошел всплеск помпажа у компрессора.

3.4.2 Дискретные сигналы процесса

Дискретные сигналы процесса приводятся на Рисунке 3.20. Существует 8 сигналов:

1. «Ошибка вычисления» – этот бит равен 1, если существует ошибка расчета системы.
2. «Авто/Ручной» – этот бит равен 1, если выбран режим «AUTO» и равен 0, если выбран режим «Ручной».
3. «АП нормальная остановка активна» – этот бит равен 1, если нажата кнопка «Стоп»
4. «Безопасность включена» – этот бит равен 1, если линия защиты от перенапряжения (SPL) включена. По умолчанию SPL линия включена.
5. «Ошибка передатчика» – если любой датчик изменяет значения слишком быстро, например значение датчика давления уменьшается от 100 баров до 10 бар непосредственно этот бит будет возведен в значение 1.
6. «Активна коррекция безопасности» - это пульс защиты безопасности, этот бит активируется, если «PV» (значение процесса, элемент- столбец желтого цвета) между 1.0404 и 1.1025.
7. «Безопасная защита активна» – это сигнал активируется защитой безопасности, этот бит взводится, если «PV» (значение процесса, элемент- столбец желтого цвета) меньше, чем 1.0404 и значение 0, если «PV» больше или равны, чем допустимое отклонение.
8. «АП перезагрузка соленоида» – этот бит равен 1, если линия защиты от помпажа (SPL) и рабочая точка расположена за линией защиты от помпажа (SPL) более 10 секунд. В результате антипомпажный клапан проходит в полностью открытое положение.

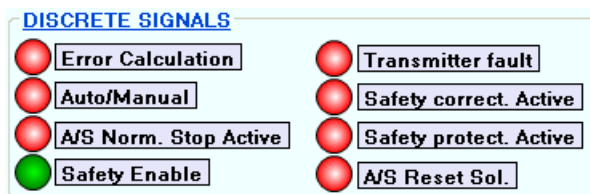


Рисунок 3.20 - Отображение дискретных сигналов

3.4.3 Аналоговые сигналы процесса

На экране антипомпажного управления отображается 3 группы аналоговых сигналов процесса:

1. Группа входных аналоговых сигналов
2. Группа сигналов, связанные с антипомпажным вычислением
3. Группа сигналов антипомпажного клапан PV (значение процесса) и SP (уставка).

3.4.3.1 Аналоговые входные сигналы процесса

Аналоговые входные сигналы процесса представлены на Рисунке 3.21.

- «Команда клапана» – это текущая позиция антипомпажного клапана в % от 0 до 100. 100% - полностью открыт.
- «PSA – давление всасывания» – это давление газа на входе компрессора в барах.
- «TSA – температура всасывания» – это температура газа на входе компрессора в градусах С.
- «PDA – выходное давление» – это давление газа на нагнетательной стороне компрессора в барах.
- «TDA – выходная температура» – это температура газа на нагнетательной стороне компрессора в градусах С.
- «Поток всасывания» – объем расхода газа на входе в компрессор.

ANALOG INPUTS		
Valve Command	4.3281	%
PSA - Suction Pressure	0.6493	Bar
TSA - Suction Temperature	42.7937	C
PDA - Discharge pressure	28788.5588	Bar
TDA - Discharge Temperature	207.1445	C
Suction Flow	8824.6936	m3/h

Рисунок 3.21 - Аналоговые входные сигналы процесса

3.4.3.2 Сигналы аналогового расчета антипомпажного процесса

Процесс сигналы аналогового антипомпажной расчета приведены на Рисунке 3.22.

- «Gain» – это пропорциональное усиление для PID регулятора.
- «Ti» – это время интегрирования для PID регулятора. Пользователь должен ввести соответствующие коэффициенты PID регулятора, эти коэффициенты использует антипомпажный контроллер (ASC) для управления скоростью и эффективностью. ASC увеличение скорости будет быстрее, если увеличит «Gain» и «Ti», но в этом случае существует вероятность возникновения прыжков и колебаний в результате неправильного процесса управления.
- «Margin» — это поле в % от перенапряжения линии управления.
- «Vlvtime» – это время открытия антипомпажного клапан в секундах от полностью закрытого в полностью открытый или из полностью открытого положения в полностью закрытое.

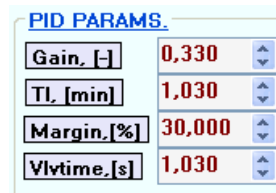


Рисунок 3.22 - Настройка аналоговых сигналов антипомпажного вычисления

3.4.3.3 Антипомпажный клапан PV/SP

На Рисунке 3.23 отображаются параметры позиции антипомпажного клапана, PV (значение процесса), SP (уставка),

- «OUT» – это текущая позиция антипомпажного клапана в % от 0% до 100%. 100% полностью открытый. «OUT» равно «Команде клапана».
- «PV» – переменная антипомпажного контроллера для процесса регулирования.
- «SP» – уставка антипомпажного контроллера. Минимальное значение $(1 + \text{margin}/100)$, максимальное значение $[(1 + \text{margin}/100) \times 2]$. Если «PV» меньше, чем «SP» то «OUT» будет увеличиваться, если «PV» больше, чем «SP» то «OUT» будет уменьшаться.

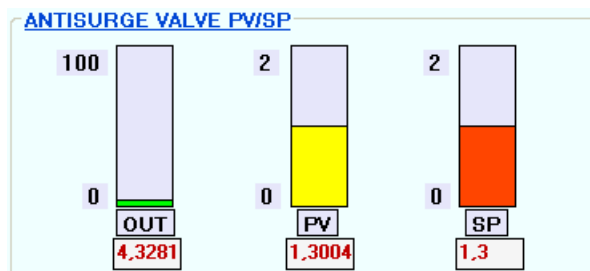


Рисунок 3.23 - Антипомпажной клапан PV/SP

3.4.4 Диалоговое окно диаграммы помпажных линий

Здесь одно диалоговое окно используется для управлением антипомпажом на оперативном экране: диалоговое окно для рисования линии от помпажа.



После нажатия на кнопку «Curve Edit on» появится диалоговое окно редактирования линии помпажа (Рисунок 3.24).

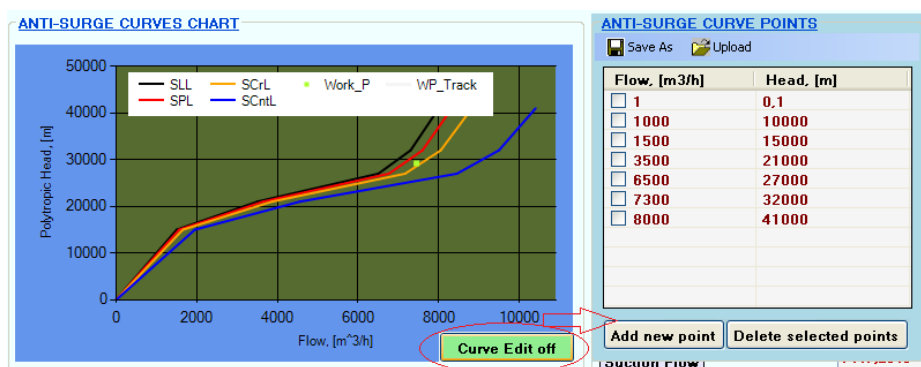


Рисунок 3.24 - Диалоговое окно Потока и Политропных пиков

Пользователь должен ввести объем потока и соответствующие значения Политропных пиков для выбранного типа компрессора. Здесь есть возможность сохранить настройки значений потока и политропных пиков в *.txt файл и загрузить ранее сохраненные значения. Для того, чтобы закрыть диалоговое окно поле, там нужно нажать кнопку «Curve Edit off».

3.4.5 Антипомпажные линии

Экран управления антипомпажом отображает рабочую точку текущего компрессора (зеленая точка), трек рабочей точки (белая линия) и 4 основных антипомпажных линий (Рисунок 3.25).

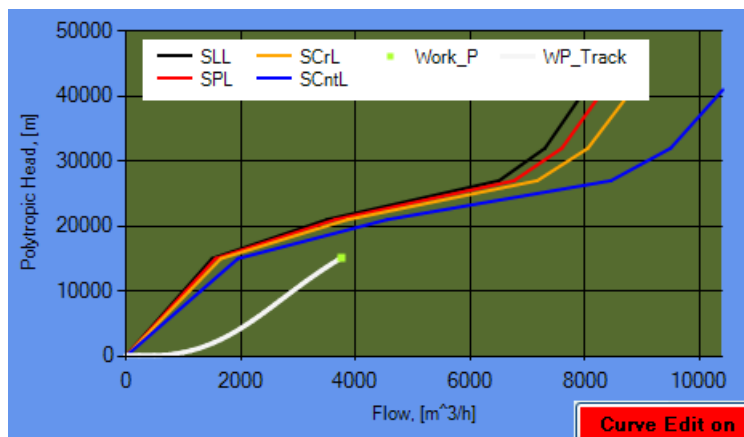


Рисунок 3.25 - Антипомпажные линии и рабочая точка

Координаты для точки работы компрессора определяется как [*«объём потока», «Политропные пики, [м]»*]

Основные линии антипомпажной являются:

- **«SLL»** – это линия ограничения помпажа (черная на Рисунке 3.25). Последняя линия перед тем как компрессор уйдет в помпаж. Эта линия (набор из значений потока для соответствующих Политропных пиков) следует искать в паспортных данных завода-изготовителя для выбранного компрессора. В этом приложении используются следующие значения по умолчанию для «SLL»:

Массив потока [1, 1000, 1500, 3500 6500, 7300, 8000]

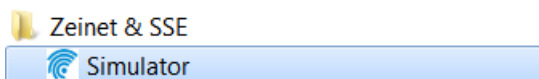
Массив пиков [0.1, 10000, 15000, 21000, 27000, 32000, 41000]

- **«SPL»** – это линия защиты от помпажа (красная на Рисунке 3.25). «SPL» равно «SLL», умноженное на 1.0404: $\text{«SPL»} = 1.0404 * \text{«SLL»}$. Если «SPL» включена (по умолчанию или кнопкой «Включить», описанных выше, «Включить безопасность» бит будет 1) и рабочая точка компрессора достигнет «SPL» защиты компрессора и будет активна на ней или за ней то после 10 секунд-антипомпажный клапан будет открыт до 100%.
- **«SCrL»** – это линия коррекции помпажа. «SCrL» равно «SLL», умноженное на 1.1025 «SCntL» – это линия управления помпажом. Антипомпажный контроллер всегда держит рабочую точку на этой линии, если рабочая точка будет меньше (с левой стороны от «SCntL») то ASC будет открывать клапан, если точка работы будет больше (по правую сторону от «SCntL») ASC закроет клапан. В нормальных условиях «SCntL» равно «SLL» умноженная $(1 + \text{margin}/100)$, но если бит «защита активна», «SCntL» будет постоянно двигать рабочую точку в правую сторону. Максимальное значение равно «SLL», умноженное на $[2 \times (1 + \text{margin}/100)]$.

4 Последовательность запуска приложения и работа с ним

4.1 Создание нового проекта. Модель одноступенчатого компрессора

4.1.1 Открыть приложение

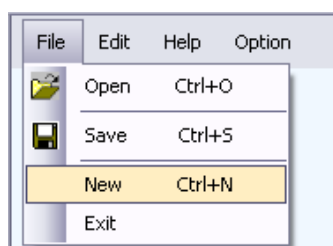


Пуск -> все программы -> Zeinet и SSE -> симулятор -> появиться главное окно приложения (Рисунок 4.1).



Рисунок 4.1 - Главное окно приложения

4.1.2 Создание нового проекта



После выполнения файл -> новый проект, появится окно выбора (Рисунок 4.2). В этом окне выберите соответствующие компоненты газа:

1-Выберите газов из левого списка

2-Нажмите кнопку «Добавить», отдельных газов будет переведен в списке справа. Это возможно для удаления газов из правого списка, выбрав из них и нажав кнопку «Удалить».

3-нажмите кнопку «Согласен». Если в списке справа кнопку «Принять» любой газ не будет отключено.

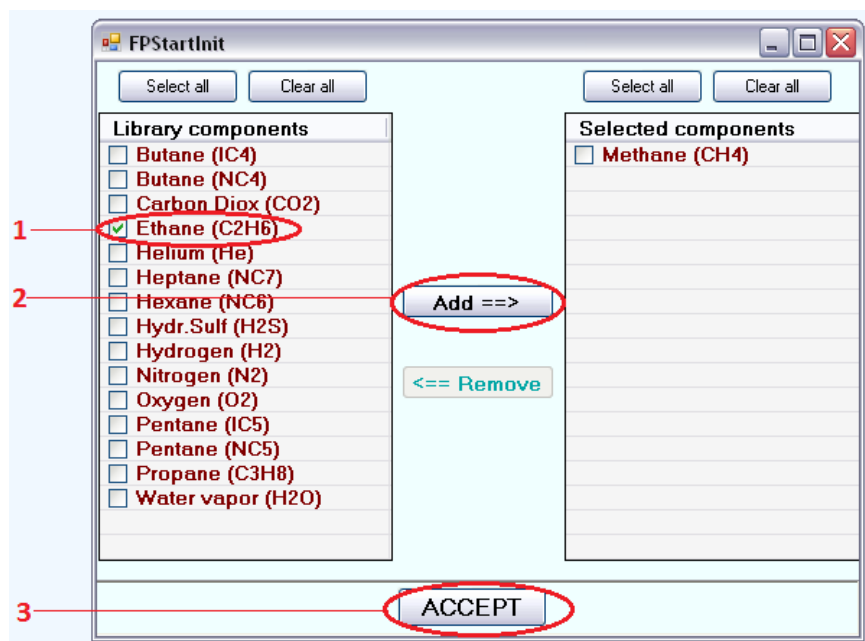


Рисунок 4.2 - Компоненты газа, выбрав окно

После нажатия кнопки «АССЕРТ» элементы появится панель (Рисунок 4.3).



Рисунок 4.3 - Панель элементов

4.1.3 Элементы подключения

Выберите необходимые для моделирования элементы, нажав на элемент на панели. Для создания модели одноступенчатого компрессора нужны следующие элементы:

1) AuxInit, регуляторный клапан, Multiplexer(Mixer), компрессор, охладитель газа, газо-жидкостный сепаратор, разделитель(уровнитель), выходной клапан, антипомпажный клапан, драйвер компрессора. Все эти элементы доступны на панели за исключением антипомпажной клапан, чтобы добавить антипомпажной клапан есть нужно дважды нажать на компрессор и установите флажок «Добавить антипомпажной управления» (Рисунок 3.6).

2) После добавления всех необходимых элементов (Рисунок 4.4) существует необходимость установить связь между элементами.

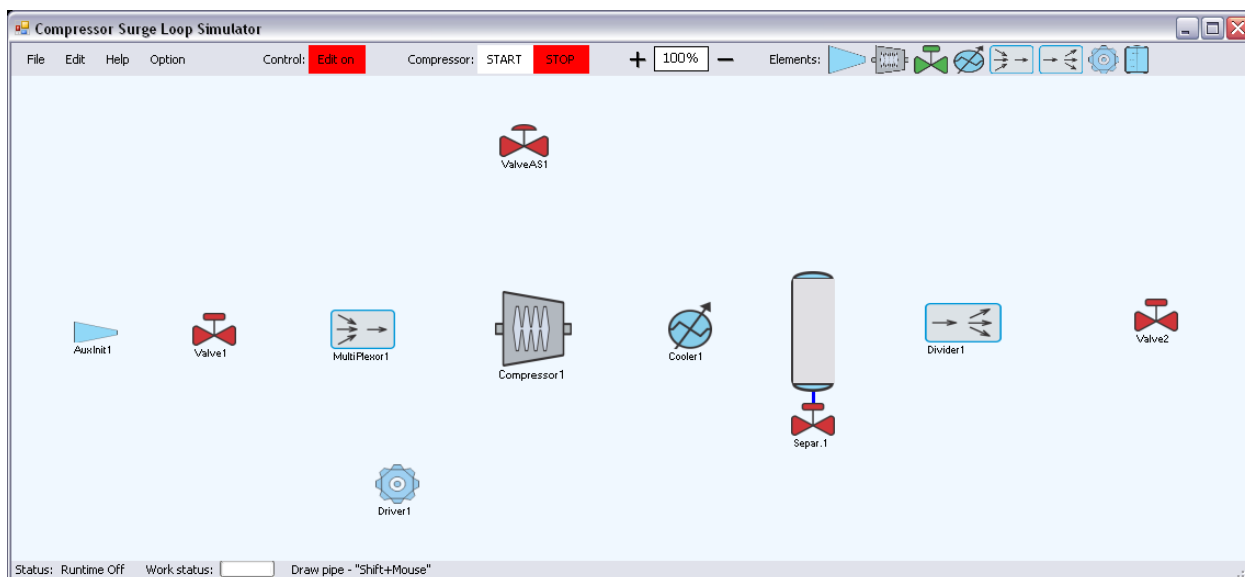


Рисунок 4.4 - Добавление элементов

Для того, чтобы создать связь между элементами, нужно нажать клавишу "Shift" и навести курсор на объект не отпуская клавишу «Shift» => вы увидите доступные входы и выходы будут выделены (желтые квадраты). (Рисунок 4.5)

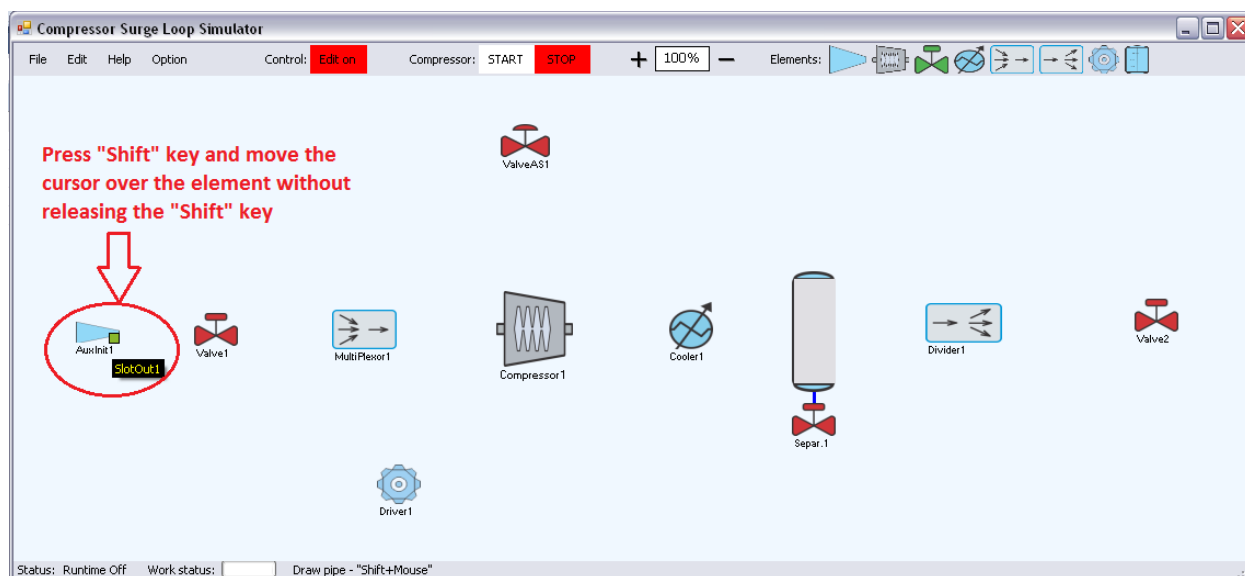


Рисунок 4.5 - Соединение элементов

Следующий шаг это нажать на имеющихся и необходимых входной/выходной (нажмите на желтый квадрат), щелкните не отпуская клавишу «Shift» курсором на следующий элемент, доступные входы и выходы также будут отображаться как желтые квадраты, чтобы необходимый можно было выбрать не обходимые. Если установление соединения между элементами должен появиться элемент «Труба» (Рисунок 4.6).

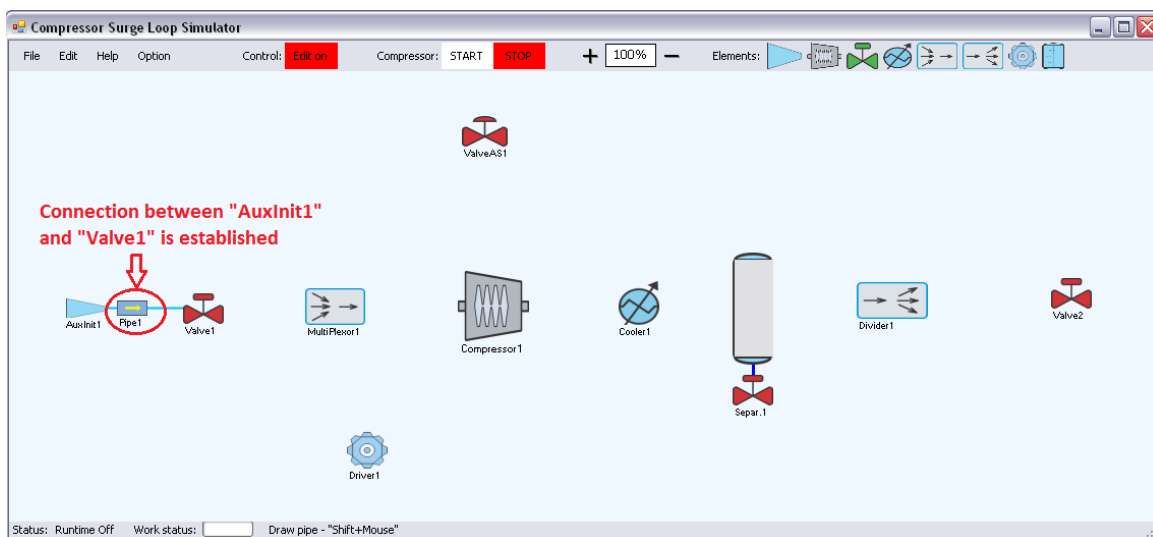


Рисунок 4.6 - Связь между AuxInit1 и Valve1

3) Для нашего примера следующий шаг заключается в создании связи между Valve1 и Multiplexor1. Принцип такой же, как в шаг 2, одно различие, что мультиплексор имеет два входа, пользователь может выбрать любой из этих двух входов. Соединения между Valve1 & Multiplexor1 & Compressor1 & Cooler1 Separator1 & Divider1 указаны на Рисунке 4.7.

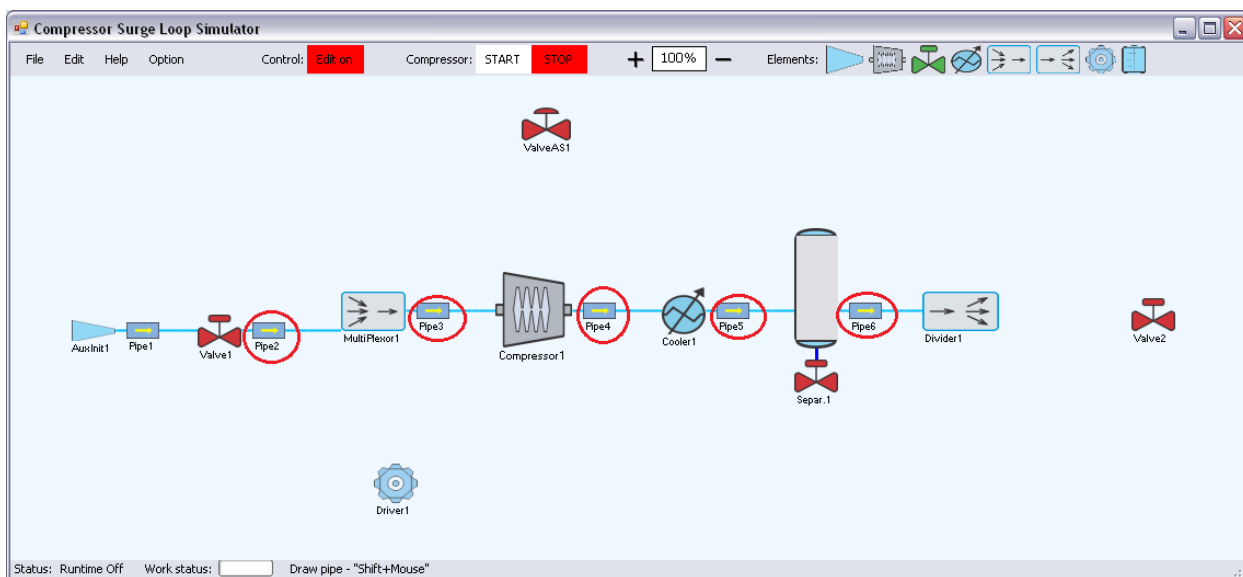


Рисунок 4.7 - Связь между Valve1... Divider1

4) Следующий шаг заключается в подключении элементов делитель (галстук). Делитель имеет одного входа и 4 выхода, пользователь может выбрать любой из этих мероприятий. Для нашего примера мы связаны делителя в 1st выход (верхний желтый квадрат) Valve2 и 2nd выходной клапан антипомпажные входы (ValveAS1), 3-й и 4-й дивайдера выходы не связаны. Затем выход антипомпажного клапана был подключен к мультиплексору в 2nd вход. Структура с разделителем и мультиплексор в соединения показано на Рисунке 4.8.

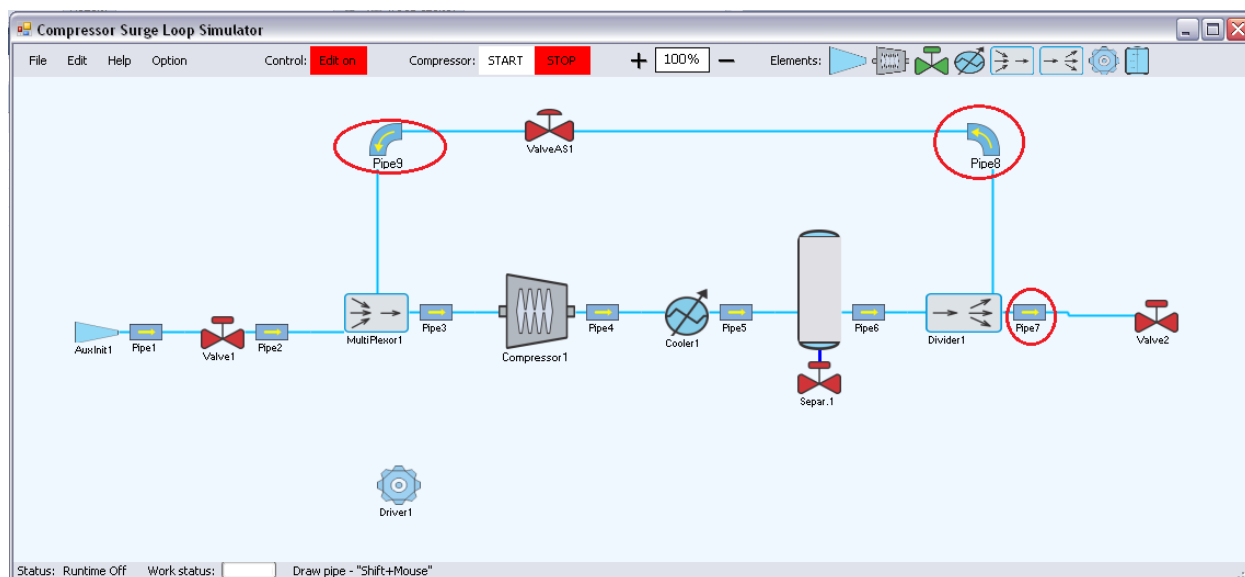
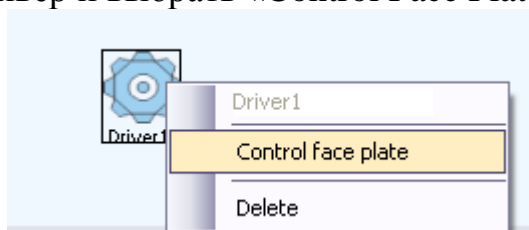


Рисунок 4.8 - Разделитель и мультиплексор соединения

5) Последний шаг заключается в подключении драйвера с компрессором. Сделать это нужно открыть панель драйвера правой кнопкой мыши на элемент драйвер и выбрать «Control Face Plate».



Водитель лицевую панель будет отображаться (Рисунок 4.9). В этом окне нужно будет выбрать:

- 1 – выберите элемент параметр разряд которого будет контролироваться скорость вращения компрессора;
- 2 – Выберите control параметр: давления, температуры и потока. Если «Скорость» выбран «от элемента» combobox будет отключен, потому что скорость контроля применимо только для компрессора;
- 3 – дать setpoint для параметра управления, MaxSpeed водителя, коэффициент скорости для водителя (время, необходимое для достижения заданной скорости);
- 4 – подключение драйвера к необходимым compressor(s) (компрессорам);
- 5 - нажмите кнопку «ОК».

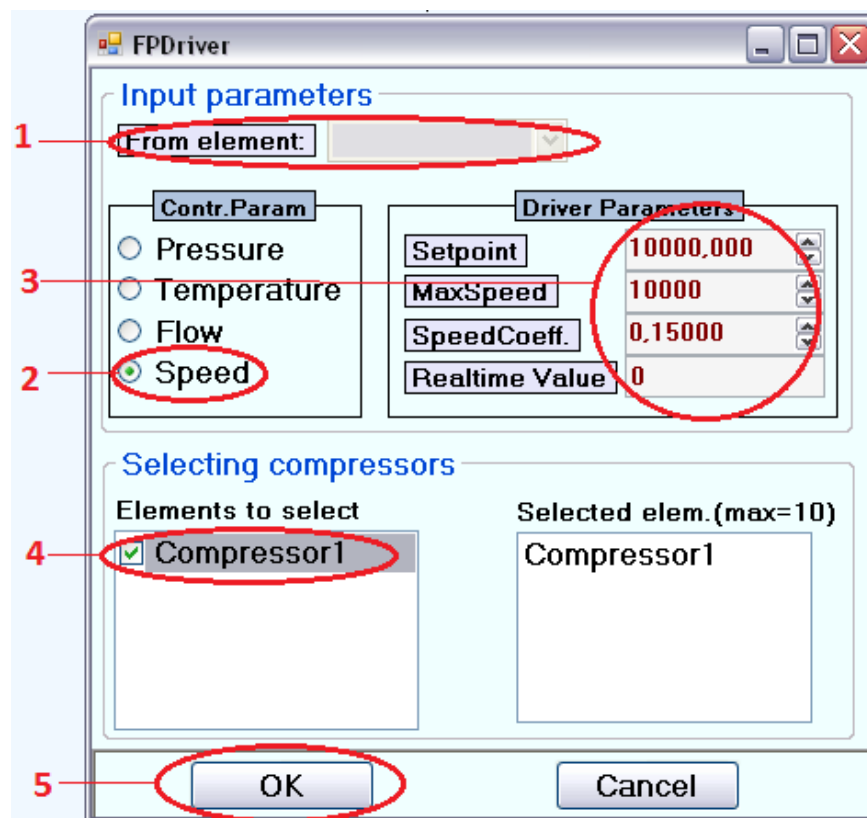


Рисунок 4.9 - Лицевая панель драйвера

В показана структура со всеми необходимыми соединениями для моделирования управления антипомпажной одного цикла (Рисунок 4.10).

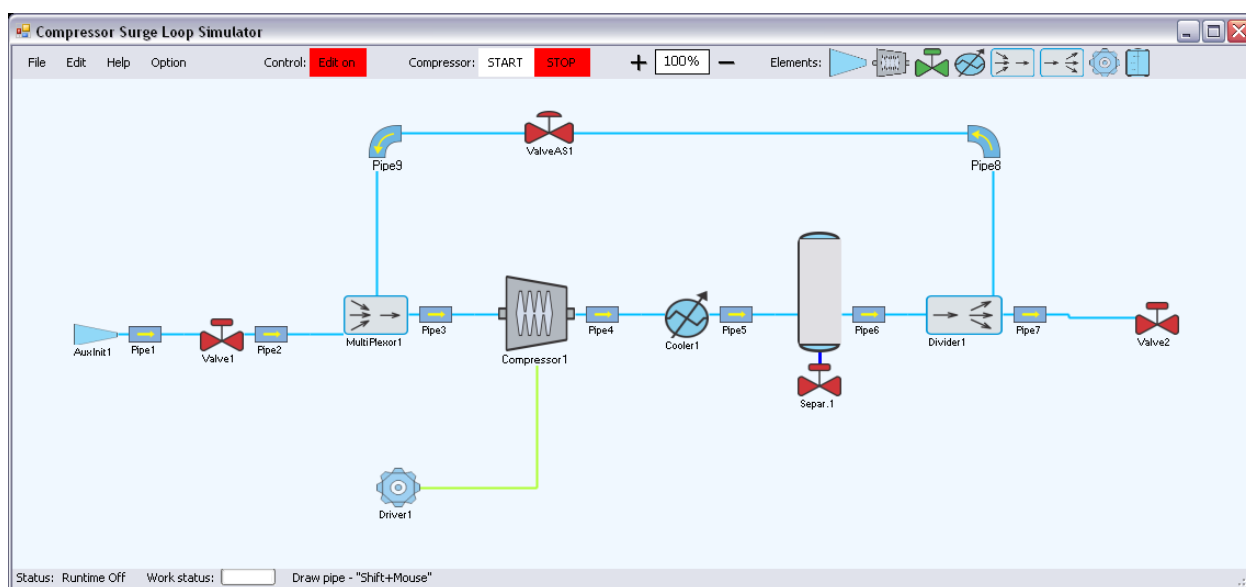


Рисунок 4.10 - Структура со всеми необходимыми соединениями.

4.1.4 Перейти в режим выполнения

Если процедура подключения между элементами делается, это можно перейти в режим выполнения, нажав кнопку «Edit Off» (красная кнопка в верхней части с лево). Приложение в режиме выполнения отображается на

Рисунке 4.11 . В этом режиме пользователь может задать соответствующие параметры для элементов, например, антипомпажной кривой компрессора, CV значение клапана и др. Каждый элемент имеет параметры по умолчанию.

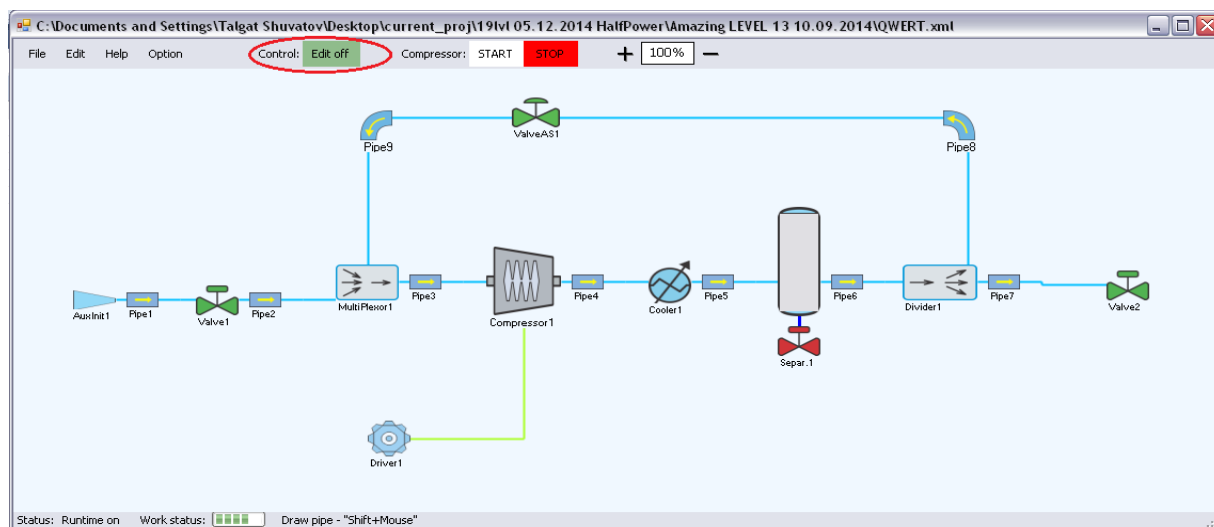


Рисунок 4.11 - Приложения в режиме выполнения

Пользователя может перейти в режиме «старт» компрессора (Рисунок 3.12) и вести наблюдение как параметры меняются, открыв лицевые панели элементов, или по наведению на элементы, где показаны основные параметры, как в/давления, температуры, объемного расхода, массового потока.

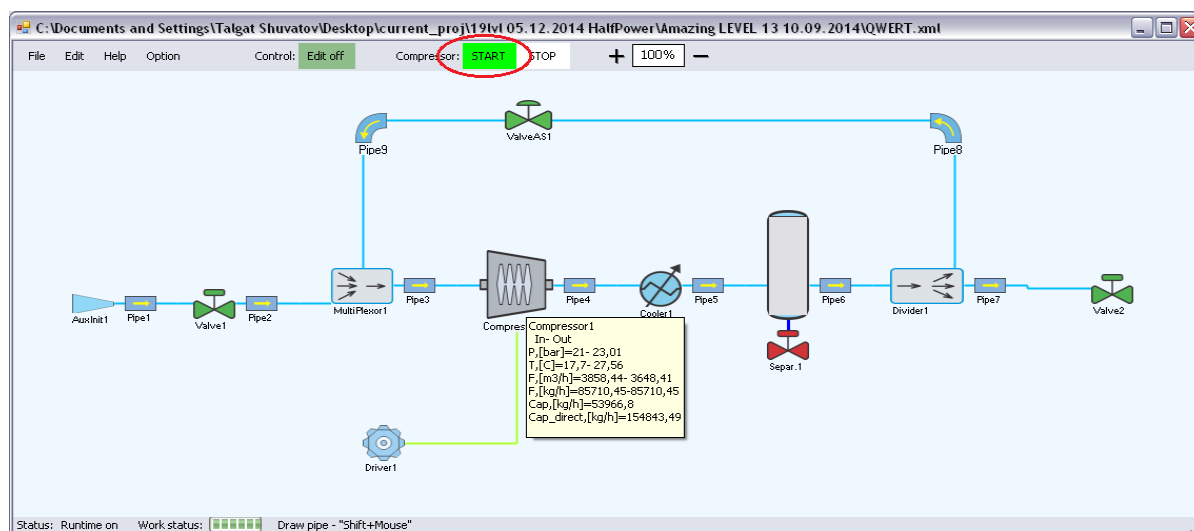


Рисунок 4.12 - Пуск компрессора

4.2 Сохранение и открытие созданной структуры

Текущей структуры могут быть сохранены как файл *.xml файл => Сохранить (Рисунок 4.13).

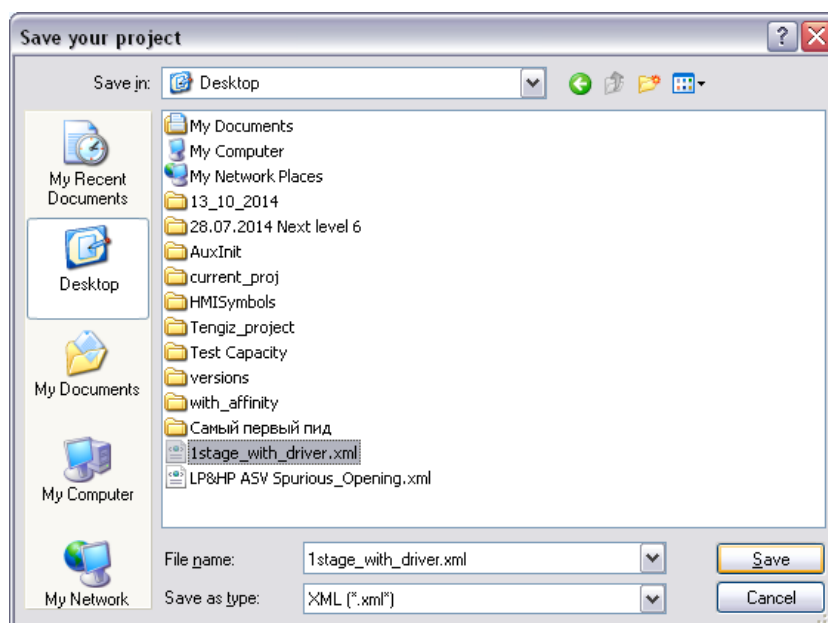


Рисунок 4.13 - Сохранение нынешней структуры

Приложение открывает сохраненную систему по умолчанию, которая сохраняется в реестре, но пользователь может выбрать любой другой сохраненный файл, файл => Открыть (Рисунок 4.14).

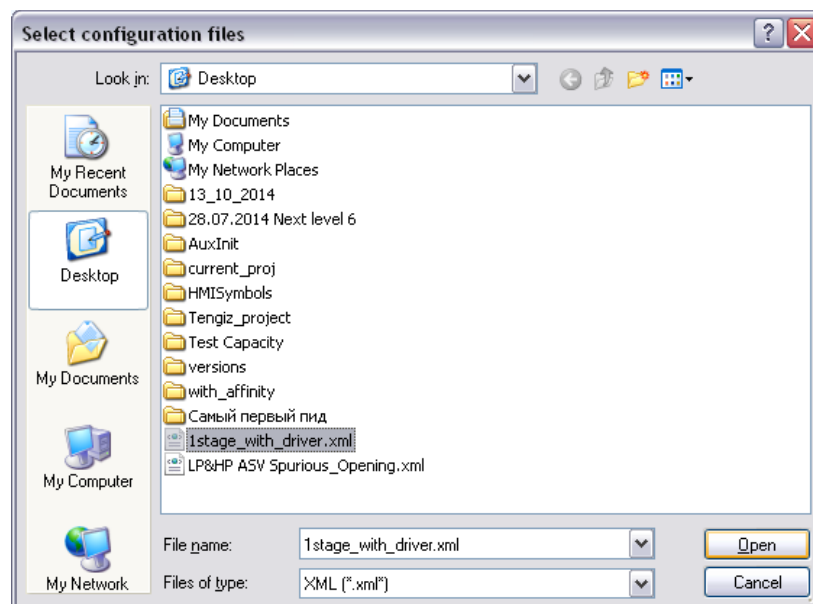


Рисунок 4.14 - Открытие сохраняется структура

5 Проверка адекватности ПО реальным данным

Наиболее важный этап при разработке ПО для моделирования технологических процессов - это проверка адекватности ПО реальным данным, а именно точность расчётов, динамическое поведение моделируемого объекта. Для проверки созданного нами ПО было проведено сопоставление результатов моделирования нашего ПО и данными реального процесса. В качестве реального процесса была выбрана двухступенчатая схема сжатия газа с двумя входами и одним выходом, полное описание которой приводится в [4]. Согласно данной схеме (Рисунок 5.1), в технологическом процессе сжатия газа участвуют 2 компрессора с одним приводом и с одинаковой номинальной скоростью. Также в данную схему включены сопутствующие оборудования, такие как охладитель, газожидкостный сепаратор, клапаны, трубы. Одним из важных элементов схемы является линия рециркуляции с антипомпажным регулированием для каждого из компрессоров.

Было проведено моделирование следующих режимов:

- 1) Динамический режим: Запуск компрессоров
- 2) Стационарный режим
- 3) Динамический режим: Останов компрессоров

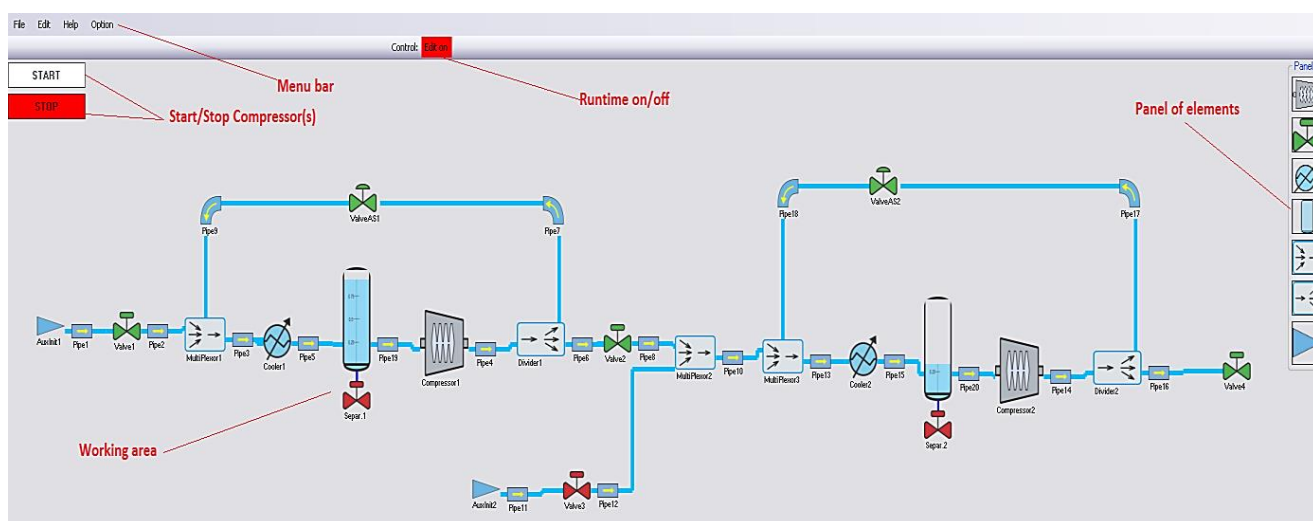


Рисунок 5.1 - Двухступенчатая схема сжатия газа

Важной частью этапа моделирования является инициализация входных параметров системы, таких как композиция газа, давление и температура газа, кривые компрессора (кривая мощности, эффективности, антипомпажная кривая), размер клапанов, диаметр труб и т.д. На Рисунок 5.2 - Рисунок 5.3 представлены диалоговые окна пользователя для ввода параметров компрессора и параметров антипомпажного регулирования.

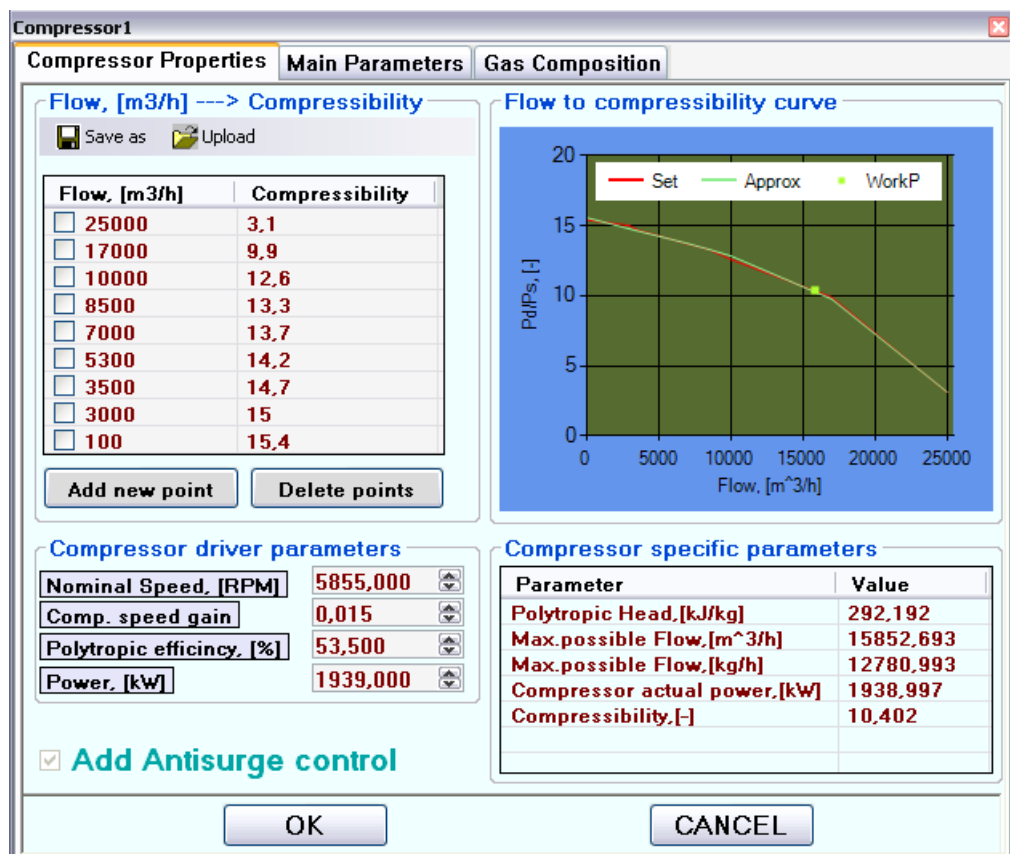


Рисунок 5.2 - Диалоговое окно для ввода параметров компрессора

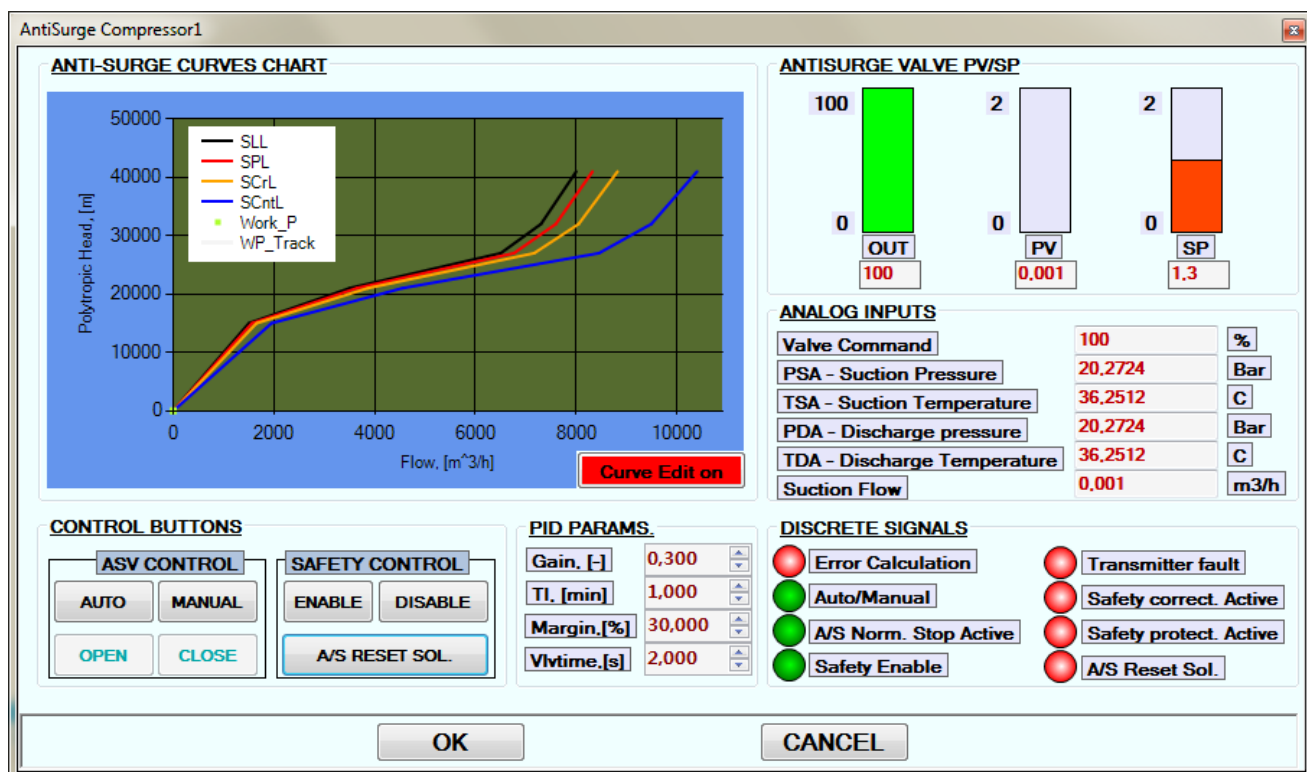


Рисунок 5.3 - Диалоговое окно для ввода параметров антипомпажного регулирования

5.1 Динамический режим: Запуск компрессоров

Данный режим предполагает разгон компрессоров и выход на номинальные обороты при полностью открытых антипомпажных клапанах. Одним из важных параметров для данного режима моделирования является время, требуемое для разгона компрессоров до номинальных оборотов. Для моделируемой и реальной систем это время равно 60-70 секунд. На Рисунках 5.4 - Рисунок 5. показаны графики изменения значений объёмного расхода, давления и температуры при разгоне компрессоров для моделируемой и реальной систем (рисунок слева – результат моделирования, рисунок справа – поведение реальной системы).

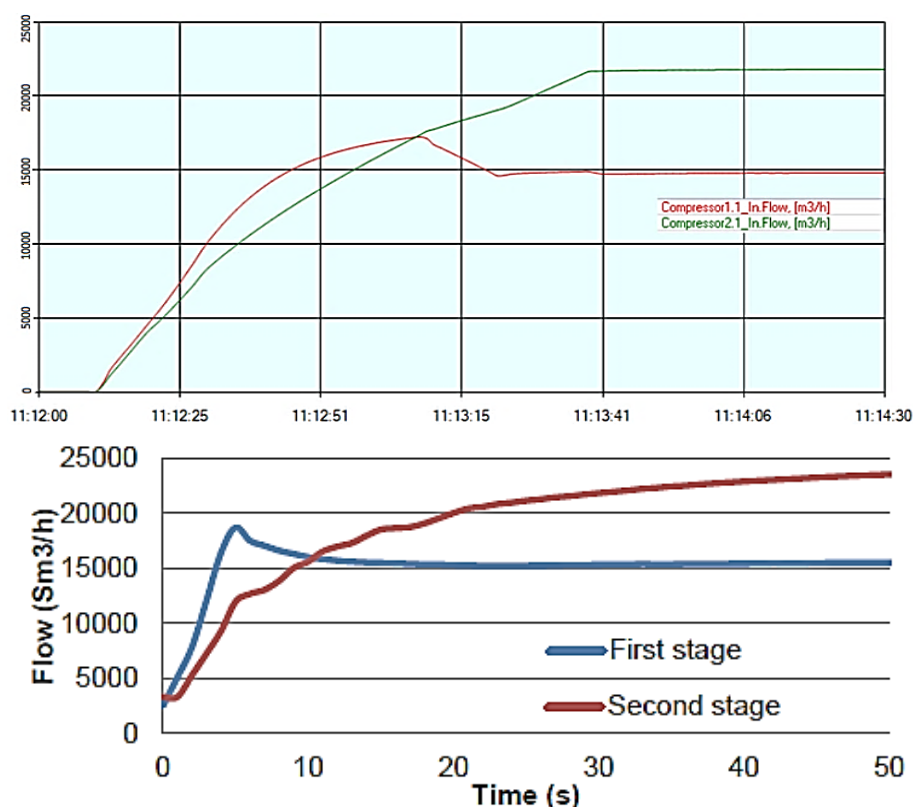


Рисунок 5.4 - Объёмный расход газа при разгоне компрессоров

Как видно по трендам (

Рисунок), характер изменения расхода для моделируемой и реальной системе совпадает.

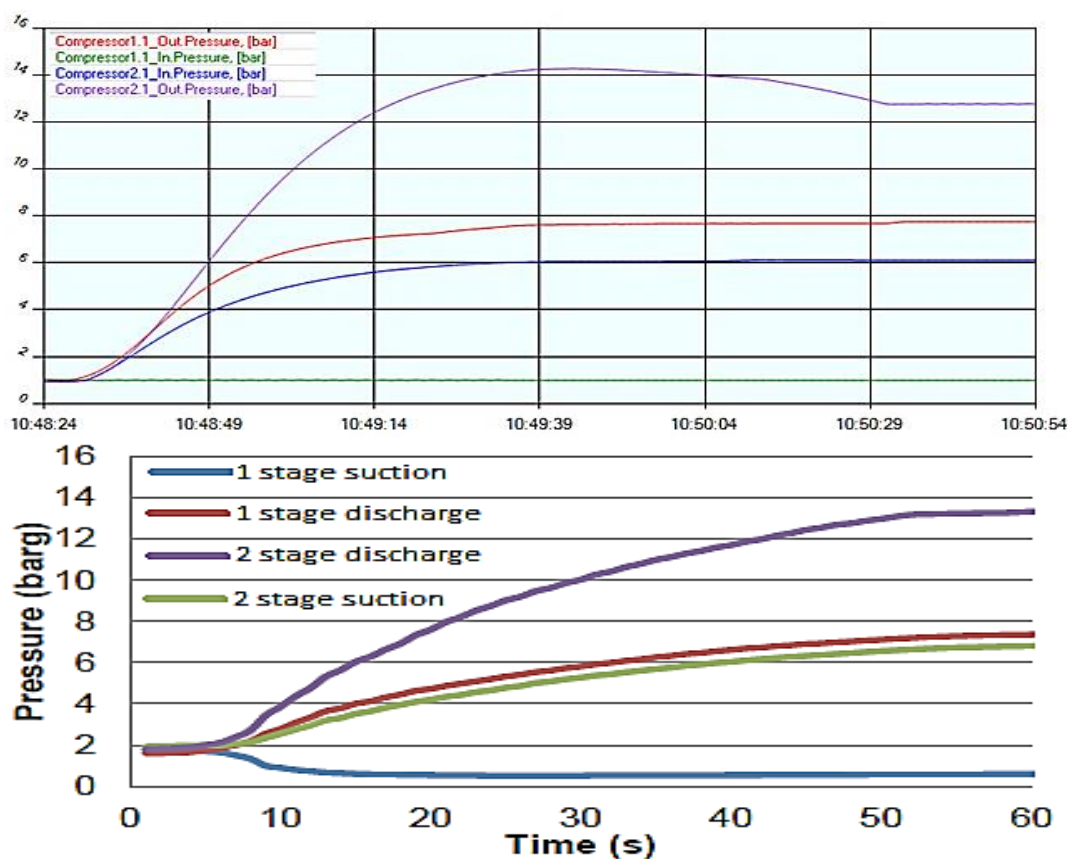


Рисунок 5.5 - Давление газа при разгоне компрессоров

По трендам (Рисунок 5.) видно, что графики давления при разгоне компрессоров для моделируемой и реальной систем схожи. Разница давлений между выходом первого компрессора и входом второго компрессора для моделируемой системы больше чем в реальной системе в виду того, что в моделируемой системе между первой и второй ступенью установлен клапан с постоянным перепадом. Давление на выходе второго компрессора для моделируемой системы сначала возрастает, затем идёт на уменьшение. Это связано с тем, что для моделируемой системы показан 2-х минутный тренд, тогда как для реальной системы показаны первые 60 секунд разгона, что не учитывает дальнейшего изменения давления.

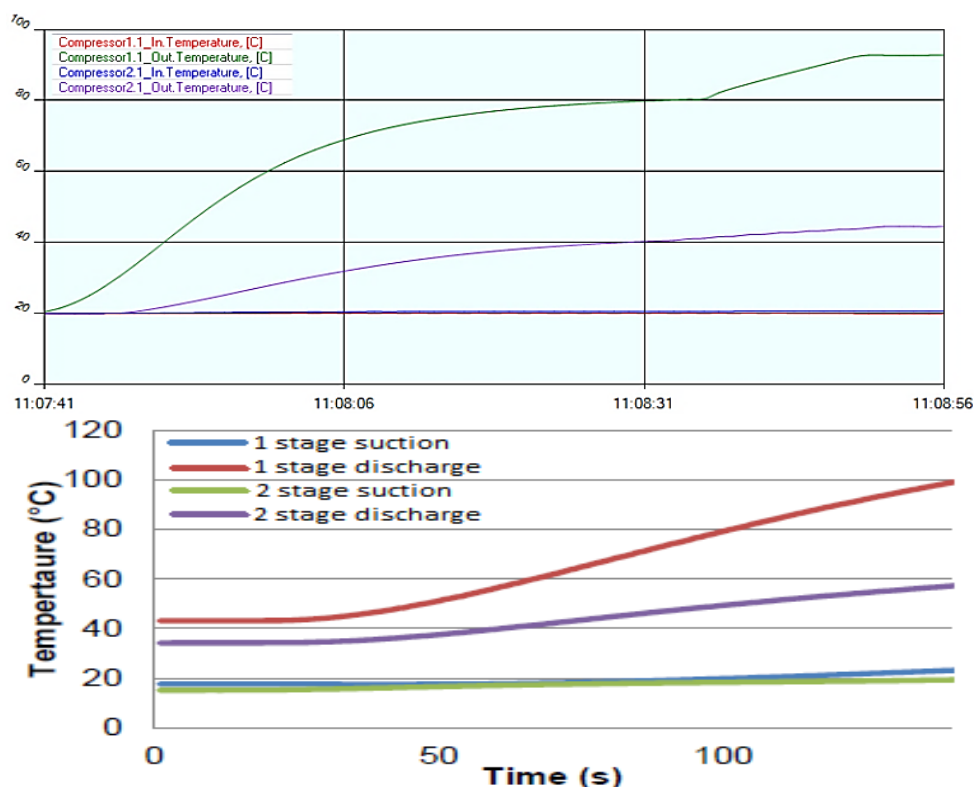


Рисунок 5.6 - Температура газа при запуске компрессоров

Тренды температуры при разгоне компрессоров (Рисунок 5.) показывают небольшие различия между моделируемой и реальной системами. В моделируемой системе начальные значения входных и выходных температур равны 20 °С, для реальной системы начальные значения температур второго компрессора 40 °С. Это связано с тем, что в программном обеспечении не моделируется накопление тепла на стенках компрессора.

5.2 Стационарный режим

Данный режим предполагает работу компрессоров на номинальных оборотах. Результаты представлены в Таблице 1. Данный режим показывает практически одинаковые результаты для моделируемой и реальной систем.

Таблица 5.1 - Результаты моделирования для стационарного режима

Параметр	Компрессор 1			Компрессор 2		
	Данные модели	Реальные данные	Разница	Данные модели	Реальные данные	Разница
MW	27,99	27,92	0,07	27,09	27,02	0,07
Q	15852,66	15853	-0,34	14404,6	14452	-47,4
$P_{вх}$	0,747	0,75	-0,003	5,55	5,58	-0,03
$P_{вых}$	10,701	10,59	0,111	13,19	13,4	-0,21
$T_{вх}$ (°C)	39,8	40	-0,2	50,289	50,4	-0,111
$T_{вых}$ (°C)	165,823	168,1	-2,277	102,632	108,6	-5,968
Мощность (кВт)	1939	2037	-98			

5.3 Динамический режим: Останов компрессоров

Данный режим предполагает останов компрессоров. На Рисунок 5.7 – Рисунок 5. показаны графики изменения значений объёмного расхода, давления и температуры при останове компрессоров (рисунок слева – результат моделирования, рисунок справа – поведение реальной системы).

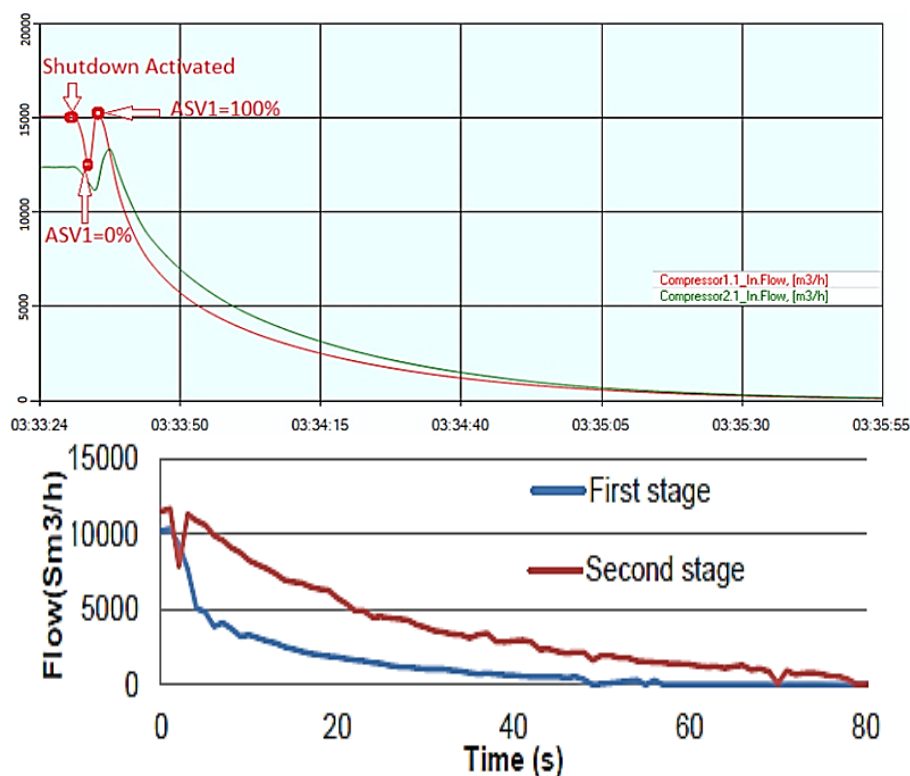


Рисунок 5.7 - Объёмный расход газа при останове компрессоров

По трендам расхода при останове компрессоров (Рисунок 5.7) видно, что для реальной системы есть вероятность помпажа на втором компрессоре, в моделированной системе в первые секунды после останова виден спад расхода для обоих компрессоров, затем его увеличение за счёт полного открытия антипомпажных клапанов.

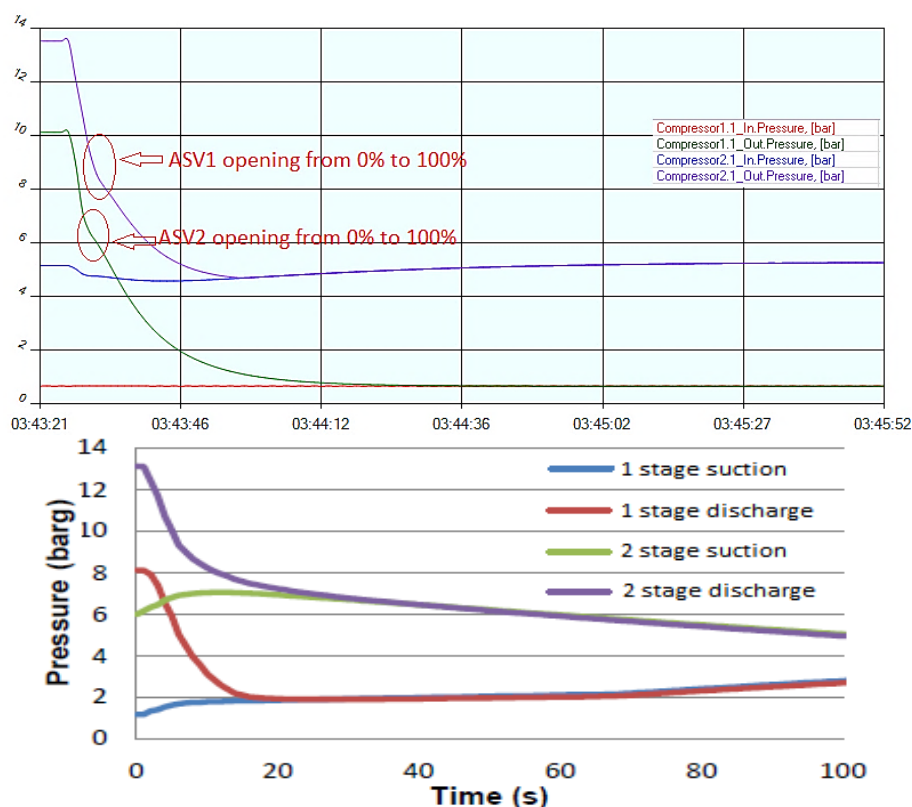


Рисунок 5.8 - Давление газа при останове компрессоров

Тренды по давлению при останове компрессоров (Рисунок 5.) показывают спад выходного давления компрессоров в первые 20 секунд останова. Также можно отметить 2 тенденции для реальной системы: это выравнивание давлений для каждого из компрессоров и дальнейшее выравнивание давления по системе в целом. Для моделированной системы эффекта выравнивания давления по системе нет, в виду того, что первая и вторая ступень инкапсулируются друг от друга клапаном. Как видно из графиков, открытие антипомпажных клапанов мало сказывается на характере изменения давления в момент останова компрессоров.

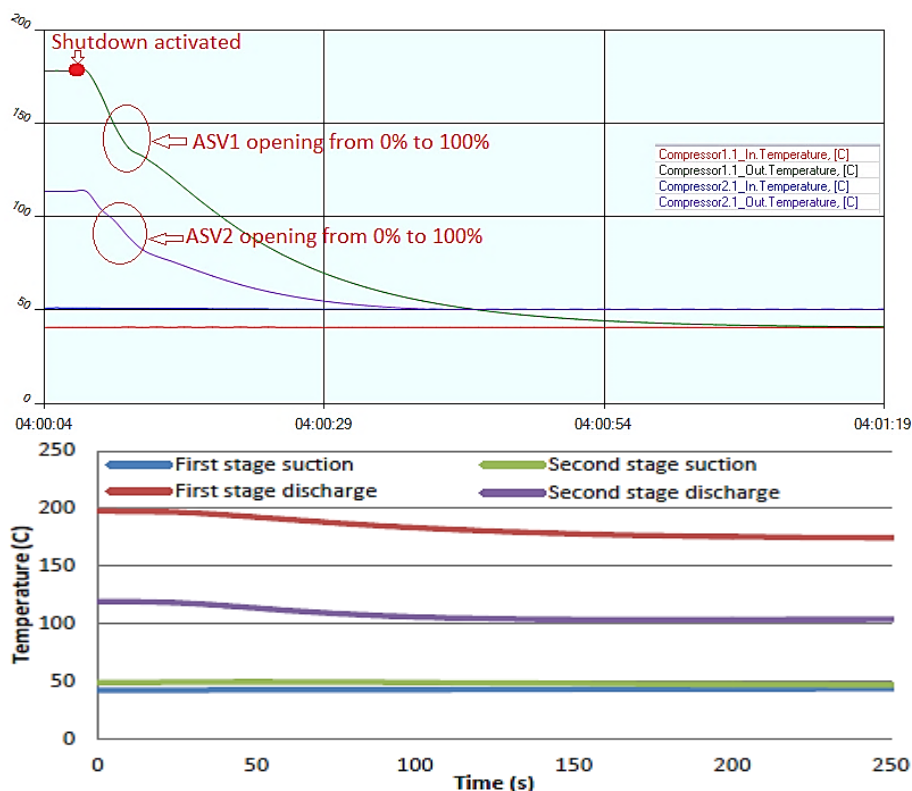


Рисунок 5.9 - Температура газа при останове компрессоров

Как видно из трендов температуры при останове компрессоров (Рисунок 5.), падение температуры для реальной системы занимает больше времени. Как упоминалось ранее, это связано с тем, что в программном обеспечении не моделируется накопление тепла на стенках компрессора.

Останов - один из самых критичных событий для антипомпажной системы регулирования компрессора, т.к. данный режим требует незамедлительной реакции антипомпажной системы регулирования, чтобы не допустить вхождения рабочей точки компрессора в помпажную зону. При тестировании систем антипомпажного регулирования, режим останова используется для подбора антипомпажного клапана (размер клапана, время открытия и т.д.).

5.4 Проверка реакции системы на внешние воздействия и изменения параметров системы

В данном тесте основное внимание уделено антипомпажному регулированию, т.к. при режимах запуска и останова компрессоров антипомпажные клапана находятся в открытом положении, что не показывает реакцию антипомпажного регулятора на внешние воздействия. В качестве воздействий выступают уменьшение расхода на входе в компрессор 1, путём прикрытия клапана 2 (клапан между первым и вторым компрессорами) с 26% до 20% и увеличение расхода, путём открытия клапана 2 с 20% до 26%. При уменьшении расхода, рабочая точка компрессора смещается в помпажную зону, регулятор начинает выдавать команду на открытие антипомпажного клапана, что ведёт к увеличению

расхода за счёт переброса части выходного расхода на вход компрессора (Рисунок 5.). Соответственно при увеличении расхода, регулятор выдаёт команду на закрытие антипомпажного клапана. Как видно из рисунка, антипомпажный регулятор отрабатывает плавно, без колебаний и выравнивает положение рабочей точки на необходимом для безопасной работы компрессора значении расхода.

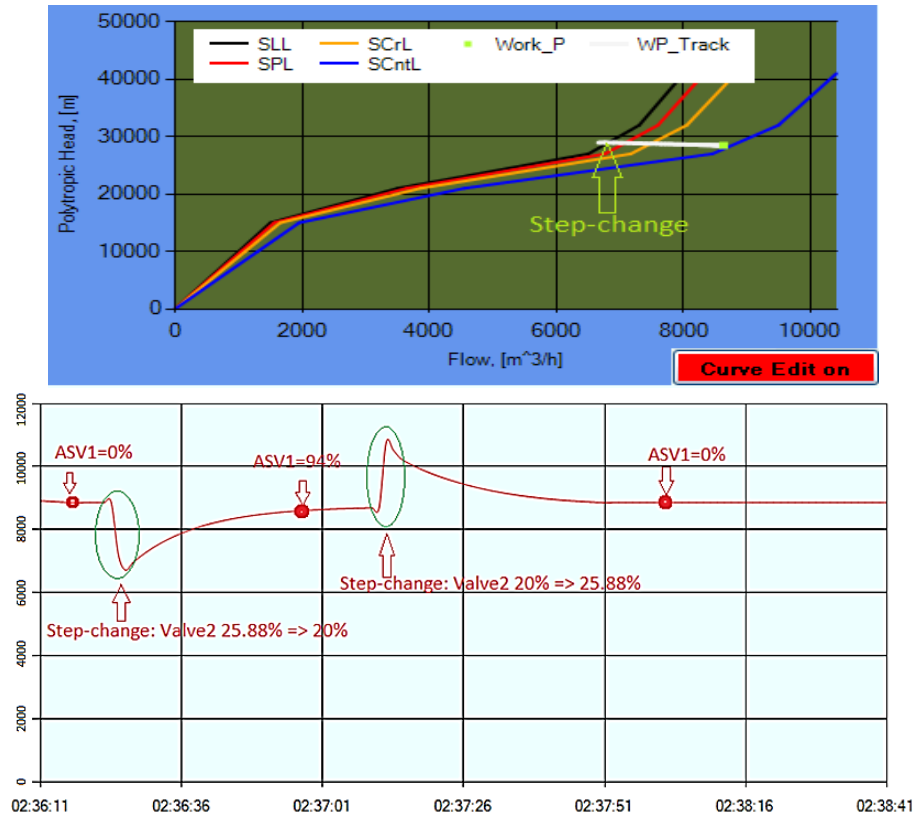


Рисунок 5.10 - Реакция антипомпажного регулятора на падение/увеличение расхода

Из описанных выше тестов можно вынести вердикт в пользу нашей платформы. Данные достаточно приближены к реальным, и управление антипомпажным клапаном хорошо откликается на происходящее в системе. Модели объектов адекватно реагируют на происходящее и корректно обмениваются информацией, что хорошо сказывается на качестве работы каждой из моделей. Пользователь имеет максимальное количество информации для понимания, что происходит в данной схеме. Все это говорит о том, что диссертационный проект удался и выполняет свою задачу на все 100%. И все задачи стоявшие перед платформой она выполняет так же корректно.

6 Экономическая часть

6.1 Расчет трудоемкости программного продукта

Расчет проводится в соответствии с моделью COSOMO II, формулы в которой выведены с использованием статистического подхода – учитывались реальные результаты огромного количества проектов. Автор оригинальной модели – Барри Боэм, ввел ее в 1995 году. Для описания и применения модели требуется информация о размере и сложности программного продукта, цена которого рассчитывается в экономической части дипломного проекта. Размер и основные характеристики программного продукта и являются исходными данными для написания курсовой работы.

На основе определенных объемов и уровней сложности рассчитываются затраты в человеко-месяцах на разработку программного обеспечения по следующей формуле:

Рассчитаем затраты на разработку программы в человеко-месяцах по следующей формуле

$$\text{Затраты} = A * (\text{РАЗМЕР})^B * M_e, \quad (6.1)$$

где A - масштабный коэффициент 2,5

B - показатель, отражающий нелинейную зависимость затрат от размера проекта 1,11

РАЗМЕР – размер программы в тысячах строк, 5 тыс. строк,

M_e – множитель поправки.

Исходя из таблицы 1 вычисляем значение множителя поправки по формуле 2

$$M_e = \prod_{i=1}^7 EM_i, \quad (6.2)$$

где EM_i – числовое значение формирователей затрат.

$M_e = 1 \cdot 0,72 \cdot 1 \cdot 1 \cdot 1 \cdot 1,12 \cdot 1 = 0,80$

Формирование затрат приведено в таблице 12.

Таблица 6.1 - Оценка и числовые значения формирователей затрат

Название фактора затрат	Обозначение	Оценка фактора	Числовое значение
Возможности персонала	ЕМ ₁	Номинальные	1
Надежность и сложность продукта	ЕМ ₂	Низкий	0,72
Требуемое повторное использование	ЕМ ₃	Номинальное, на уровне проекта	1
Трудность платформы	ЕМ ₄	Номинальная	1
Опытность персонала	ЕМ ₅	Номинальная, 1 год	1
Средства поддержки	ЕМ ₆	Низкий	1,12
График	ЕМ ₇	Номинальный, 100%	1

По формуле 1 определяем величину затрат

Затраты=2.5*51.11*0,80=11.93 чел-месяц

Рассчитанные затраты распределяются по этапам в соответствии с приведенным в таблице 6 соотношением.

Таблица 6.2 – Распределение времени по этапам

Этапы разработки программного продукта	Распределение затрат, %	Величина затрат
Анализ и проектирование	40%	4,7
Кодирование, тестирование, отладка	60%	7,15
Итого	100%	11,93

Затраты составляют 11,93 чел-месяц.

6.2 Расчет численности работников и затрат на основную и дополнительную заработную плату

На каждом этапе разработки необходимы работники. На этапе анализа и проектирования - аналитик. Требуемая квалификация: высшее образование, первая или высшая категория. Разряд Единой тарифной сетки, согласно – 14 (тарифный коэффициент 2.55).

На этапе кодирования, тестирования и отладки – программист. Требуемая квалификация: средне специальное образование. Разряд единой тарифной сетки – 9 (тарифный коэффициент 1.78).

Для данного вида работ используется повременная форма оплаты труда. За основу расчета оплаты труда используем Единую тарифную сетку, в которую вписан весь каталог действующих профессий и должностей по разрядам. Отнесение работников к той или иной квалификационно-должностной группе основывается в сетке на сложности их труда.

Рассчитывается размер должностного оклада по формуле

$$\text{Оклад} = \text{МЗП} * K_{\text{тар}} \quad (6.3)$$

где МЗП – минимальная заработная плата (с 01.01.2011 года = 15999 тнг.)

$K_{\text{тар}}$ – тарифный коэффициент, устанавливается в соответствии с ЕТС РК.

$$\text{Оклад}_{\text{аналитика}} = 100000 * 2,55 = 40797,45 \text{ тенге}$$

$$\text{Оклад}_{\text{программиста}} = 100000 * 1,78 = 28478,22 \text{ тенге}$$

Учитывая, что рассчитанный оклад – это стоимость одного человеко-месяца, можно рассчитать затраты на основную заработную плату по формуле 4

$$\text{Основная зарплата} = \text{Затраты (чел. – мес.)} * \text{Оклад (тенге)} \quad (6.4)$$

$$\text{Основная зарплата}_{\text{аналитика}} = 6,68 * 40797,45 = 272526,96 \text{ тенге}$$

$$\text{Основная зарплата}_{\text{программиста}} = 4,45 * 28478,22 = 126728,07 \text{ тенге}$$

Дополнительная заработная плата составляет 10% от основной заработной платы.

$$\text{Дополнительная зарплата}_{\text{аналитика}} = 272526,96 * 10\% = 27252,69 \text{ тенге}$$

$$\text{Дополнительная зарплата}_{\text{программиста}} = 126728,07 * 10\% = 12672,80 \text{ тенге}$$

Таблица 6.3 - Расчет фонда оплаты труда

Наименование должностей	Трудовое количество работ, чел-мес.	Оклад	Количество, чел.	Основная зарплата, тенге	Дополнительная зарплата, тенге	Фонд оплаты труда, тенге
Аналитик	6,68	40797,45	1	272526,96	27252,69	299779,76
Программист	4,45	28478,22	1	126728,07	12672,80	139400,87
ИТОГО:	11,14	69275,67	2	399255,03	39925,49	439180,63

Рассчитаем среднемесячную заработную плату по следующей формуле:

$$ЗП_{CP}^{MEC} = \frac{\Phi OT}{T_{OБЩ}} , \quad (6.5)$$

где ΦOT – общий фонд оплаты труда (таблица 12)

$T_{OБЩ}$ – общая трудоемкость разработки программы в человеко-месяцах (таблица 2)

$$ЗП_{CP}^{MEC} = \frac{439180,63}{11,14} = 39423,75 \text{ тенге}$$

Согласно формуле (5) среднемесячная заработная плата составляет 39423,75 тенге.

6.3 Расчёт эксплуатационных расходов

На основе данных о стоимости оборудования производим расчеты эксплуатационных расходов: амортизационные отчисления, расходы на электроэнергию, прочие расходы на содержание и эксплуатацию оборудования.

Поскольку для разработки ПО используется только компьютер, принимаем величину нормы амортизации $H_a=40\%$ а для программного обеспечения $H_a=15\%$.

Размер годовых амортизационных отчислений рассчитывается по следующей формуле:

$$A_{M}^{год} \equiv \frac{\Phi_{пер} * H_a}{100} \quad (6.6)$$

где $A_{M}^{год}$ – размер амортизационных отчислений по данному виду оборудования в год

$\Phi_{пер}$ – первоначальная стоимость оборудования

H_a – годовая норма амортизации

Согласно формуле 6 получаем размер амортизационных отчислений на системный блок и программное обеспечение

$$\text{Системный блок } A_{M}^{год} \equiv \frac{121367 * 40\%}{100} = 48546,8 \text{ тенге}$$

$$\text{Программное обеспечение } A_{M}^{год} \equiv \frac{11833,5 * 15\%}{100} = 1775,02 \text{ тенге}$$

Определим фонд рабочего времени , за год по следующей формуле 7:

$$\Phi PB_{обор}^{зод} = (\Phi PB_{кал} - ПРАЗД - ВЫХ) * t_{см} * s * (1 - a) \quad (6.7)$$

где $\Phi PB_{кал}$ - календарный фонд времени (366 дней);
 ПРАЗД - количество праздничных дней (11 в год);
 ВЫХ - количество выходных дней (при пятидневной рабочей неделе 104 в год);
 $t_{см}$ - продолжительность рабочего дня (принимается равной 8 часам);
 s - количество смен (принимается односменный график);
 a - коэффициент, учитывающий остановки на ремонт и обслуживание (принимается 0,1).

$$\Phi PB_{обор}^{зод} = (366 - 11 - 104) * 8 * 1 * (1 - 0,1) = 1807,2 \text{ тенге.}$$

Для расчета фонда рабочего времени оборудования, затраченного на создание данного программного продукта, необходимо трудоемкость этапа кодирования, тестирования и отладки, выраженную в человеко-месяцах, перевести в машино-часы. Рассчитывается по формуле

$$\Phi PB_{обор}^{факт} = \frac{\Phi PB_{обор}^{зод} * T_{код}}{12}, \quad (6.8)$$

где $T_{код}$ - трудоемкость этапа кодирования, тестирования и отладки (таблица 2);

12 – количество месяцев в году.

Необходимо отметить, что при таком способе расчета, если для разработки ПО необходимо 2 или более единиц оборудования, $\Phi PB_{обор}^{факт}$ равен общему фонду рабочего времени всех единиц оборудования данного вида. Чтобы узнать фонд времени работы одной единицы оборудования, необходимо разделить $\Phi PB_{обор}^{факт}$ на количество единиц оборудования $n_{об}$.

Исходя из формулы 8

$$\Phi PB_{обор}^{факт} = \frac{1807,2 * 11,14}{12} = 1677,68 \text{ машино-час}$$

Амортизационные отчисления, относимые на себестоимость данного программного продукта, рассчитываются по формуле

$$З_{ам} \equiv \frac{A_m^{зод} * \Phi PB_{об}^{факт}}{\Phi PB_{обор}^{зод} * n_{об}}, \quad (6.9)$$

где $З_{ам}$ - размер амортизационных отчислений, относимых на себестоимость разрабатываемого программного продукта

$n_{об}$ – количество единиц оборудования

Исходя из формулы 9, размер амортизационных отчислений, относимых на себестоимость разрабатываемого ПО:

$$\text{Системный блок } Z_{ам} \equiv \frac{48546,8 * 1677,68}{1807,2 * 1} = 45067,51 \text{ тенге}$$

$$\text{Программное обеспечение } Z_{ам} \equiv \frac{1775,02 * 1677,68}{1807,2 * 1} = 1647,8 \text{ тенге}$$

Рассчитаем расходы на электроэнергию по следующей формуле:

$$Z_{эл} = \text{ФРВ}_{обор}^{факт} * W_{об} * \Pi_{эл}, \quad (6.10)$$

где $W_{об}$ - мощность единицы оборудования;

$\Pi_{эл}$ - цена электроэнергии (принимается равной 8,07 тенге)

Из формулы 10 затраты на электроэнергию для системного блока составят:

$$Z_{эл} = 1677,68 * 0,4 * 8,07 = 5415,55 \text{ тенге}$$

Из формулы 10 затраты на электроэнергию для монитора составят:

$$Z_{эл} = 1677,68 * 0,08 * 8,07 = 1083,11 \text{ тенге}$$

Прочие расходы на содержание и эксплуатацию оборудования включают оплату расходов на ремонт и обслуживание оборудования работниками сторонних организаций. Принимаются в размере 10% от суммы основной и дополнительной заработной платы (фонда оплаты труда).

Результаты расчетов приведены в таблице 9.

Таблица 6.4 - Расчет эксплуатационных расходов

Наименование оборудования	Сумма амортизационных отчислений, тенге	Затраты на электроэнергию, тенге	Прочие расходы на содержание и эксплуатацию, тенге.	Итого эксплуатационных расходов, тенге
Системный блок	45067,51	5415,55		
Монитор		1083,11		
Клавиатура		0		
Мышь		0		
Программное обеспечение	1647,8	0		
ИТОГО:	46714,51	6498,66	43804,58	97017,75

6.4 Составление сметы затрат на разработку программного обеспечения

Смета затрат – общий свод плановых затрат предприятия в денежном выражении на выполнение работ.

Производственная себестоимость – затраты, непосредственно связанные с выполнением данного вида работ. Полная себестоимость – включает производственную себестоимость, административные и управленческие расходы и затраты на реализацию продукции.

Затраты на комплектующие включает приобретение компакт-диска для записи программного обеспечения, что составляет 70 тенге.

Размер отчислений от заработной платы (социальный налог) рассчитывается 11% от ФТО.

Налог аналитика составляет: $H = 194603,83 * 11\% = 21406,42$ тенге

Налог программиста составляет: $H = 203619,27 * 11\% = 22398,11$ тенге

Общая сумма отчислений равна 43804,53

Результаты расчетов статей затрат сведены в таблицу 11.

Таблица 6.5 - Смета затрат на разработку программного продукта

Наименование статей	Сумма, тенге
Материальные затраты: покупные и комплектующие изделия	70,00
Основная заработная плата	398223,1
Дополнительная заработная плата	39822,3
Отчисления от заработной платы специалиста	43804,53
Эксплуатационные расходы	97017,75
Итого полная себестоимость	578937,68

Полная себестоимость составила 578937,68 тенге.

6.5 Расчет отпускной цены

Используя метод ценообразования «издержки + прибыль», на основе рассчитанной себестоимости программного продукта рассчитывается его отпускная цена. Для этого сначала определяется размер прибыли предприятия и расчетная (договорная) цена программного продукта. Расчетная цена предприятия – это доход от реализации программного продукта ($D_{\text{РЕАЛ}}$).

Расчетная цена предприятия рассчитывается по формуле:

$$C_{p.n} = C_{\text{пол}} + P_p \quad (6.11)$$

P_p – прибыль предприятия;

$C_{\text{полн}}$ – полная себестоимость разработки (таблица 7).

Прибыль предприятия рассчитывается по формуле:

$$P_p = \frac{C_{\text{полн}} * R}{100} \quad (6.12)$$

R – рентабельность принимается на среднеотраслевом уровне, равной 20%

$$P_p = \frac{578937,68 * 20\%}{100} = 115787,54 \text{ тенге}$$

$$C_{p.n} = 578937,68 + 115787,54 = 694725,2 \text{ тенге}$$

Отпускная цена предприятия - цена, по которой предприятие, изготовившее продукт, реализует его потребителям. Она отличается от расчетной цены на величину подлежащего к уплате налога на добавленную стоимость.

Отпускная цена предприятия рассчитывается по формуле:

$$C_{\text{отп}} = C_{p.n} + НДС \quad (6.13)$$

НДС – налог на добавленную стоимость, ставка налога равна 12% от суммы добавленной стоимости;

$$НДС = \frac{C_{p.n} * 12\%}{100} \quad (6.14)$$

$$НДС = \frac{694725,2 * 12\%}{100} = 83867,03 \text{ тенге}$$

$$C_{\text{отп}} = 694725,2 + 83867,03 = 778592,23 \text{ тенге}$$

6.6 Расчет технико-экономических показателей

Рентабельность – один из основных стоимостных качественных показателей эффективности производства на предприятии, объединении, отрасли в целом, характеризующий уровень отдачи затрат и степень использования средств в процессе производства и реализации продукции.

Рентабельность продукции – отношение прибыли от реализации продукции к ее полной себестоимости.

Рентабельность продукции рассчитывается по следующим формулам:

$$R_{в.пр} = \frac{ВПр}{C_{полн}} * 100\% , \quad (6.15)$$

$$R_{в.пр} = \frac{115787,54}{694725,2} * 100\% = 16,7\%$$

$C_{полн}$ – полная себестоимость разработки ПО;

$ВПр$ – прибыль от реализации программного продукта.

Валовая прибыль предприятия находится по следующей формуле:

$$ВПр = D_{реал} - C_{полн} \quad (6.16)$$

Где $D_{реал}$ - доход от реализации продукции (расчетная цена предприятия)

$$ВПр = 694725,2 - 578937,68 = 115787,52 \text{ тенге}$$

Производительность труда – продуктивность производственной деятельности людей; измеряется количеством продукции произведенной работником в сфере материального производства за единицу рабочего времени. Рассчитывается по следующей формуле: 17

$$П_t^{нат} = \frac{РАЗМЕР}{T_{общ}} \quad (6.17)$$

РАЗМЕР – размер программного обеспечения, выражается в тысячах LOK (Lines Of Code);

Тобщ– трудоемкость разработки программного продукта, в человеко-месяцах.

$$П_t^{нат} = \frac{РАЗМЕР}{T_{общ}} = \frac{5}{11,93} = 0,41 \text{ чел/месяц}$$

Также рассчитывается производительность труда в денежных измерителях на одного работающего в год:

$$П_t^{ден} = \frac{D_{реал}}{T_{общ}} = \frac{578937,68}{11,93} = 48527,87 \quad (6.18)$$

Все технико-экономические показатели оформлены в виде таблицы 18.

Таблица 6.6 - Техничко-экономические показатели ПО

Наименование показателя	Значение
Рентабельность продукции валовая	16,7%
Валовая прибыль предприятия	115787,52
Производительность труда	0,41
Производительность труда в денежных измерителях	48527,87
Средняя себестоимость	115787,54
Цена реализации	694725,2
Отпускная цена	778592,23

Срок окупаемости капитальных затрат рассчитывается по формуле:

где $\Phi_{пер}$ -расходы на подготовку производства тыс. тенге;

$T_{ок}$ - срок окупаемости, год;

$\Pi_t^{ден}$ - годовой ожидаемый эффект, тыс. тенге.

$$T_{ок} = \frac{C_{полн}}{ВПр} \quad (6.18)$$

$$T_{ок} = \frac{578937,68}{115787,52} = 4,99 \sim 5(месяцев)$$

Таким образом, время окупаемости программного продукта составляет 5 месяцев.

ЗАКЛЮЧЕНИЕ

В магистерской диссертации было проделано следующее:

- 1) Представлено описание программного обеспечения для моделирования режимов работы многоступенчатого газового компрессора;
- 2) Проведён ряд тестов для проверки адекватности программного обеспечения реальным данным;
- 3) Проведены тесты для проверки того, как система, собранная в данном ПО, реагирует на внешние воздействия и изменения основных параметров системы.

По результатам тестирования можно сделать вывод, что моделируемая система для различных режимов работы и параметров даёт результаты близкие к реальным. В свою очередь, это говорит о том, что

- данное программное обеспечение позволяет оценить реальные процессы в таких оборудованиях как газовый компрессор, охладитель газа, газо-жидкостный сепаратор, газовый клапан, труба;
- представленные расчеты основаны на реальных данных и показывают практическую значимость полученных результатов;
- каждое из оборудований описывается математической моделью с соответствующими формулами, которые приводят модель к реальному технологическому процессу;
- предлагаемое программное обеспечение учитывает факторы, влияющие на процесс реального сжатия газа в газовых компрессорах.
- логика алгоритмов моделирования адекватно отражает реальные технологические процессы в представленных оборудованиях.

Практическим применением данного приложения является возможность его внедрения в обучающих целях для исследования процессов, протекающих при различных режимах работы газового компрессора.

СПИСОК ЛИТЕРАТУРЫ

- 1 Герке В.Г., Рубель В.В., Сарданашвили С.А. Проблемы внедрения компьютерных комплексов моделирования, оптимизации и прогнозирования режимов газотранспортных систем.
- 2 Меренков А.П., Хасилев В.Я. Теория гидравлических цепей. – М.: Наука, 1985. – 278 с.
- 3 Новицкий Н.Н., Сеннова Е.В. Гидравлические цепи. Развитие теории и приложения. – Новосибирск: Наука, Сибирская издательская фирма РАН, 2000. – 273 с.
- 4 Johan Liedman, Robert Mansson, Dynamic simulation of a centrifugal compressor system.
(<http://publications.lib.chalmers.se/records/fulltext/179072/179072.pdf>)
- 5 <http://matmetod-popova.narod.ru/theme13.htm>
- 6 <http://mialo.narod.ru/ped/models/etap.htm>
- 7 <HTTP://WWW.HIMI.OGLIB.RU/BGL/167.HTML>
- 8 <http://www.online-electric.ru/articles.php?id=25>
- 9 <http://kbyte.ru/ru/Programming/Sources.aspx?mode=tech&tid=2>
- 10 <http://www.vbnet.ru/samples/>
- 11 <http://vbbook.ru/vb.net/primeru-rabotu-s-access/>
- 12 <http://www.cyberforum.ru/vb-net/thread348151.html>
- 13 <http://www.realcoding.net/articles/glava-11-podderzhka-baz-dannykh-v-vb-net.html>
- 14 <http://gospodaretsva.com/urok-2-graficheskij-interfejs-proekta.html>
- 15 <http://bourabai.kz/einf/vb7/>
- 16 <http://metanit.com/visualbasic/tutorial/3.8.php>
- 17 <http://www.znание-soft.com/site/vbnet/gl13.html>
- 18 <http://compress.ru/article.aspx?id=12038>
- 19 <http://allbest.ru/k-2c0a65625a3ac69a5d43a88521216d37.html>
- 20 <http://www.vbnet.ru/vbguide/showgroup.asp?id=4>
- 21 <http://citforum.edunet.kz/programming/application/rational/>
- 22 <http://famsl.ru/Algorithm.aspx>
- 23 <http://bourabai.kz/einf/vb7/Chapter5.htm>
- 24 <http://compress.ru/article.aspx?id=11974>
- 25 Visual Basic. Освой самостоятельно, Никита Культин, "БХВ-Петербург", 2009
- 26 Visual Basic. Освой самостоятельно. 2-е издание, Никита Культин, "БХВ-Петербург", 2009
- 27 Visual Basic в задачах и примерах, Игорь Сафронов, БХВ-Петербург, 2013
- 28 Visual Basic 2010 на примерах, В. Зиборов, БХВ-Петербург, 2011
- 29 <http://agilesummer.org/programmist/vbmagic/index.htm>


```

End Get
Set(ByVal value As Double())
    ArrInputGasComposition = value
End Set
End Property
Public Property InputGas_Weights As Double()
Get
    Return ArrInputGasWeights
End Get
Set(ByVal value As Double())
    ArrInputGasWeights = value
End Set
End Property

Public Property OutputParameters As Double()
Get
    Return ArrOutputParameters
End Get
Set(ByVal value As Double())
    ArrOutputParameters = value
End Set
End Property
Public Property OutputGas_Composition As Double()
Get
    Return ArrOutputGasComposition
End Get
Set(ByVal value As Double())
    ArrOutputGasComposition = value
End Set
End Property
Public Property OutputGas_Weights As Double()
Get
    Return ArrOutputGasWeights
End Get
Set(ByVal value As Double())
    ArrOutputGasWeights = value
End Set
End Property

Public Property Flow_Head_Polynom As Double()
Get
    Return ArrFlowHeadPolynom
End Get
Set(ByVal value As Double())
    ArrFlowHeadPolynom = value
End Set
End Property
Public Property InPressure_limit As Double
Get
    Return PressureLimit
End Get
Set(ByVal value As Double)
    PressureLimit = value
End Set
End Property
Public Property InCompressor_Speed As Double
Get
    Return n
End Get
Set(ByVal value As Double)
    n = value
End Set
End Property
Public Property OutHS_flow As Double
Get
    Return hs
End Get
Set(ByVal value As Double)
    hs = value
End Set
End Property
Public Property OutHead As Double
Get
    Return Hn
End Get
Set(ByVal value As Double)
    Hn = value
End Set

```

```

End Property
Public Property Flow As Double()
    Get
        Return ArrFlow
    End Get
    Set(ByVal value As Double())
        ArrFlow = value
    End Set
End Property
Public Property PdtoPs As Double()
    Get
        Return ArrPdtoPs
    End Get
    Set(ByVal value As Double())
        ArrPdtoPs = value
    End Set
End Property
Public Property SpecParams As Double()
    Get
        Return ArrSpecParams
    End Get
    Set(ByVal value As Double())
        ArrSpecParams = value
    End Set
End Property
Public Property SpecParamsIn As Double()
    Get
        Return ArrSpecParamsIn
    End Get
    Set(ByVal value As Double())
        ArrSpecParamsIn = value
    End Set
End Property

Dim PipeInName As String = 0
Dim PipeOutName As String = 0
Public Property PipeIn As String
    Get
        Return PipeInName
    End Get
    Set(ByVal value As String)
        PipeInName = value
    End Set
End Property
Public Property PipeOut As String
    Get
        Return PipeOutName
    End Get
    Set(ByVal value As String)
        PipeOutName = value
    End Set
End Property
Public Property RevCapacity As Double()
    Get
        Return Rev_Capacity
    End Get
    Set(ByVal value As Double())
        Rev_Capacity = value
    End Set
End Property
Public Property InputParameters_toFP As Double()
    Get
        Return ArrInputParameters_toFP
    End Get
    Set(ByVal value As Double())
        ArrInputParameters_toFP = value
    End Set
End Property
Public Property OutputParameters_toFP As Double()
    Get
        Return ArrOutputParameters_toFP
    End Get
    Set(ByVal value As Double())
        ArrOutputParameters_toFP = value
    End Set
End Property
Public Property Efficiency As Double()
    Get

```



```

        Return ArrEfficiency
    End Get
    Set(ByVal value As Double())
        ArrEfficiency = value
    End Set
End Property
Public Property Flow_Eff_Polynom As Double()
    Get
        Return ArrFlowEffPolynom
    End Get
    Set(ByVal value As Double())
        ArrFlowEffPolynom = value
    End Set
End Property
#End Region
#End Region
Dim ToolShow As Boolean = False
#Region "INDEXING ELEMENTS"
Dim PiPeSel As Boolean 'для драгэнддропа
Dim MeSel As Boolean
Dim SelectedNumber As Integer
Dim IndexNum As Integer
Dim InputIndexElement_1, InputIndexElement_2, InputIndexElement_3 As Integer
Dim OutputIndexElement_1, OutputIndexElement_2 As Integer
Dim OutputASVlv As String = "0"
Dim OutputASCtrl As String = "0"
Dim Runtime As Boolean
Public Property RuntimeControl As Boolean
    Get
        Return Runtime
    End Get
    Set(ByVal value As Boolean)
        Runtime = value
    End Set
End Property
Public Property PipeControll As Boolean
    Get
        Return PiPeSel
    End Get
    Set(ByVal value As Boolean)
        PiPeSel = value
    End Set
End Property
Public Property ControllSelected As Boolean
    Get
        Return MeSel
    End Get
    Set(ByVal value As Boolean)
        MeSel = value
    End Set
End Property
Public Property IndexInArray As Integer
    Get
        Return SelectedNumber
    End Get
    Set(ByVal value As Integer)
        SelectedNumber = value
    End Set
End Property
Public Property IndexOnForm As Integer
    Get
        Return IndexNum
    End Get
    Set(ByVal value As Integer)
        IndexNum = value
    End Set
End Property
Public Property InputIndex1 As Integer
    Get
        Return InputIndexElement_1
    End Get
    Set(ByVal value As Integer)
        InputIndexElement_1 = value
    End Set
End Property
Public Property InputIndex2 As Integer
    Get
        Return InputIndexElement_2

```

```

End Get
Set(ByVal value As Integer)
    InputIndexElement_2 = value
End Set
End Property
Public Property InputIndex3 As Integer
Get
    Return InputIndexElement_3
End Get
Set(ByVal value As Integer)
    InputIndexElement_3 = value
End Set
End Property
Public Property OutputIndex1 As Integer
Get
    Return OutputIndexElement_1
End Get
Set(ByVal value As Integer)
    OutputIndexElement_1 = value
End Set
End Property
Public Property OutputIndex2 As Integer
Get
    Return OutputIndexElement_2
End Get
Set(ByVal value As Integer)
    OutputIndexElement_2 = value
End Set
End Property
Public Property OutputAntiSurgeControl As String
Get
    Return OutputASCtrl
End Get
Set(ByVal value As String)
    OutputASCtrl = value
End Set
End Property
Public Property OutputAntiSurgeValve As String
Get
    Return OutputASVlv
End Get
Set(ByVal value As String)
    OutputASVlv = value
End Set
End Property
#End Region

#Region "FacePlate and Select"
Dim FacePlateTemp As FPCCompressor
Public FacePlateTempAS As FPCurves
Public ASFacePlate As Boolean = False
Dim QPoint As Point
Dim FacePlateTempTrend As FPTrend
Dim Q2

Public Sub SelectCtrl(ByVal sender As System.Object, ByVal e As System.Windows.Forms.MouseEventArgs) Handles PictureBox1.Click
Try
    ToolTip1.Hide(PictureBox1)
    If e.Button = Windows.Forms.MouseButtons.Right Then
        If Runtime = True Then
            DeleteToolStripMenuItem.Enabled = False
        Else
            DeleteToolStripMenuItem.Enabled = True
        End If
        QPoint = New Point(MyOwner.Location.X + e.Location.X + Me.Location.X + 20, MyOwner.Location.Y + e.Location.Y +
Me.Location.Y + 60)
        If OutputASCtrl <> "0" Then
            ContextMenuStrip1.Items.Item(5).Enabled = True
        Else
            ContextMenuStrip1.Items.Item(5).Enabled = False
        End If
        ContextMenuStrip1.Show(QPoint)

    End If

Catch ex As Exception
    MsgBox("Compressor - Select Control Error")
End Try

```

```

End Sub

Public Sub RefreshFacePlate() 'Handles MyBase.Click
    Try

        If FPopenNow = True Then
            FacePlateTemp.RefreshValue()
        End If
        If FPCurvesopenNOW = True Then
            FacePlateTempAS.RefreshValue()
        End If

        'Catch ex As Exception
        'MyOwner.StopTheTimer()
        'MsgBox("Compressor - Refresh FP Error")
    'End Try
End Sub
#End Region

#Region "Calculated var"
'Optimization Array
Dim Tvihoda, Tvhod, Pvhod, Hn_to_stop, Capac_to_stop, torque, Pvihoda_prev As Double
Dim Start_stop_prev As Boolean
'Constants
Dim n0, R0, T0, e0, a As Double
'Other
Dim pvx, pvix, psr, tvx, tvix, tsr, zvx, zvix, zsr, x1, x2, pvx1, pvix1, Hn, Hn2, Q1, Q2_to_ramp, hs, Ni, Speed_Set, Q2_prev, e0_prev,
CompCap_prev, Speed_prev, PID_FBK, lag As Double
Dim e0_cur, e0_cur2 As Double
'instead of timer to 1 sec
Dim timervar As Integer
Dim curtime As String

#End Region

Private Sub CalculateRange(ByRef Max As Double, ByRef Min As Double, ByRef x As Double, ByRef y As Double)
    If y < Min Then
        x = Min
    ElseIf y > Max Then
        x = Max
    Else
        x = y
    End If
End Sub

Public Sub RED_dim(ByVal Count As Integer)
    ReDim ArrFlow(Count)
    ReDim ArrPdtoPs(Count)
    ReDim ArrEfficiency(Count)
End Sub

Public Sub Revers_Step()
    If PipeOutName <> "0" Then
        MyOwner.RefreshInReversParam(MyOwner.Controls.Item("Panel1").Controls.Item(PipeOutName), Me)
    End If

    "If Rev_Capacity(0) > Compr_Capacity Then
    " Rev_Capacity(0) = Compr_Capacity
    "End If

    Dim Pipe As Pipe
    Pipe = MyOwner.Controls.Item("Panel1").Controls.Item(PipeInName)
    If Rev_Capacity(0) > Pipe.PipeCapacity Then
        Rev_Capacity(0) = Pipe.PipeCapacity
    End If
    Library.CalculateRange(10000000000000000, 1.0E-20, Rev_Capacity(0), Rev_Capacity(0))
End Sub

Public Sub First_Step() 'Расчет перепада давления и приведение остальных параметров массива
    Try
        If PipeInName <> "0" Then
            MyOwner.RefreshInParam(MyOwner.Controls.Item("Panel1").Controls.Item(PipeInName), Me)
        End If

        '*****INITIALIZATION*****
        Dim speed_up, speed_down, acc_time, dec_time As Double
        Call CalculateRange(1000000000, 0.0001, Pvhod, ArrInputParameters(0)) 'PINPUT
        Tvhod = ArrInputParameters(1) + 273.15
        '-----

```

```

Dim SP_speed As Double 'speed setpoint
n0 = ArrSpecParamsIn(2) 'nominal speed

Speed = DriverSpeed
Call Second_Step()
Catch ex As Exception
MyOwner.StopTheTimer()
MsgBox("Compressor - First Step Error")
End Try
End Sub
Public Sub Second_Step() 'CALCULATION OF
Try
Dim Pvihoda, M1, M2, Pkr1, Tkr1, Pkr2, Tkr2, w, H_Hig, Z_ci, m_t, R_1, R_2, d_1, d_2, k, k_div_kml, n_div_nm1, Log_x1, Log_x2,
Max_Flow, Max_Flow_Cur, Max_Head, Min_Flow, Min_Head, Max_Flow_eff, Min_Flow_eff As Double
Dim H_in, Cp_in, k_in, H_out, Cp_out, k_out As Double

Min_Flow = 1000000000000000
Min_Head = 1000000000000000
Min_Flow_eff = 10000000000000000
n = Speed
Dim ArrFlow_cur() As Double = New Double(UBound(ArrFlow)) {}
Dim ArrPdtoPs_cur() As Double = New Double(UBound(ArrPdtoPs)) {}
Dim ArrEfficiency_cur() As Double = New Double(UBound(ArrEfficiency)) {}

'sorting array
Dim i1, tmp_Q, tmp_Hn, tmp_Eff As Long
Dim t As Boolean
t = True
Do While t
t = False
For i1 = 0 To UBound(ArrFlow) - 1
If ArrFlow(i1) > ArrFlow(i1 + 1) Then
tmp_Q = ArrFlow(i1)
tmp_Hn = ArrPdtoPs(i1)
tmp_Eff = ArrEfficiency(i1)

ArrFlow(i1) = ArrFlow(i1 + 1)
ArrPdtoPs(i1) = ArrPdtoPs(i1 + 1)
ArrEfficiency(i1) = ArrEfficiency(i1 + 1)

ArrFlow(i1 + 1) = tmp_Q
ArrPdtoPs(i1 + 1) = tmp_Hn
ArrEfficiency(i1 + 1) = tmp_Eff
t = True
End If
Next i1
Loop

*****CALCULATION OF INLET GAS DENSITY, [kg/m3]*****
Library.CriticalPT(ArrInputGasWeights, ArrInputGasComposition, R0, Pkr1, Tkr1, w, M1)
R_1 = 8.31451 / M1 'Gas constant for natural gas [kJ/(kg*K)]
d_1 = M1 / 28.96 'relative density of the natural gas by air [-]
pvx = Pvhod / Pkr1 'reduced to normal conditions input pressure value [-]
tvx = Tvhod / Tkr1 'reduced to normal conditions input temperature value [-]
Library.Z_Factor_Redl_Kwong(Pvhod, Tvhod, Pkr1, Tkr1, zvx) 'Compressibility factor by the input parameters [-]
Library.Z_Factor_Peng_Robinson(Pvhod, Tvhod, Pkr1, Tkr1, w, H_Hig, zvx) 'Compressibility factor by the input
parameters [-]
Library.CalculateRange(10, 0.2, zvx, zvx)
pvx1 = (100 * Pvhod) / (zvx * Tvhod * R_1) 'Gas density at the compressor inlet [kg/m^3]
Library.CalculateRange(100000, 0.001, pvx1, pvx1)

*****CALCULATION OF e(compressibility), a(efficiency)*****
Dim i_max, i_min As Integer
For i = 0 To UBound(ArrFlow)
ArrFlow_cur(i) = (n / n0) * (ArrFlow(i))
Library.CalculateRange(ArrFlow(i), (1 / 5.5) * (ArrFlow(i)), ArrFlow_cur(i), ArrFlow_cur(i))
ArrPdtoPs_cur(i) = ((n / n0) ^ 2) * (ArrPdtoPs(i))
Library.CalculateRange(ArrPdtoPs(i), (1 / 30.25) * (ArrPdtoPs(i)), ArrPdtoPs_cur(i), ArrPdtoPs_cur(i))
ArrEfficiency_cur(i) = (ArrEfficiency(i))
If ArrFlow_cur(i) > Max_Flow_Cur Then
Max_Flow_Cur = ArrFlow_cur(i)
i_max = i
End If
If ArrFlow(i) > Max_Flow Then
Max_Flow = ArrFlow(i)
End If
End For

```

```

If ArrFlow(i) < Min_Flow_eff Then
    Min_Flow_eff = ArrFlow(i)
End If
If ArrFlow_cur(i) < Min_Flow Then
    Min_Flow = ArrFlow_cur(i)
    i_min = i
End If
If ArrPdtoPs_cur(i) > Max_Head Then
    Max_Head = ArrPdtoPs_cur(i)
End If
If ArrPdtoPs_cur(i) < Min_Head Then
    Min_Head = ArrPdtoPs_cur(i)
End If

Next

Library.Polynoms(ArrFlow_cur, ArrPdtoPs, ArrFlowHeadPolynom)
Library.Polynoms(ArrFlow, ArrEfficiency, ArrFlowEffPolynom)

Q1 = (ArrInputParameters(2)) //Inlet gas flow, [m3/h]
Library.CalculateRange(10000000000000, 0.001, Q1, Q1)

Dim eff As Double
If n > 0 Then
    If Q1 >= Min_Flow And Q1 <= Max_Flow_Cur Then
        If Q1 > 0 And n >= 1 Then
            e0_cur2 = ((n / n0) ^ 2) * (ArrFlowHeadPolynom(3) * ((Q1) ^ 3) + ArrFlowHeadPolynom(2) * ((Q1) ^ 2) +
ArrFlowHeadPolynom(1) * Q1 + ArrFlowHeadPolynom(0)) 'reduced compression ratio
            End If

            ElseIf Q1 < Min_Flow Then
                Library.ETRAPOLATION(ArrFlow_cur(i_min + 1), ArrPdtoPs_cur(i_min + 1), ArrFlow_cur(i_min), ArrPdtoPs_cur(i_min), Q1,
e0_cur2)
            ElseIf Q1 > Max_Flow_Cur Then
                Library.ETRAPOLATION(ArrFlow_cur(i_max - 1), ArrPdtoPs_cur(i_max - 1), ArrFlow_cur(i_max), ArrPdtoPs_cur(i_max), Q1,
e0_cur2)
            End If

            If Q1 <= Max_Flow And Q1 >= Min_Flow_eff Then
                eff = ((ArrFlowEffPolynom(3) * ((Q1) ^ 3) + ArrFlowEffPolynom(2) * ((Q1) ^ 2) + ArrFlowEffPolynom(1) * Q1 +
ArrFlowEffPolynom(0))) 'reduced compression ratio
            ElseIf Q1 > Max_Flow Then
                Library.ETRAPOLATION(ArrFlow(i_max - 1), ArrEfficiency_cur(i_max - 1), ArrFlow(i_max), ArrEfficiency_cur(i_max), Q1,
eff)
            ElseIf Q1 < Min_Flow_eff Then
                Library.ETRAPOLATION(ArrFlow(i_min + 1), ArrEfficiency_cur(i_min + 1), ArrFlow(i_min), ArrEfficiency_cur(i_min), Q1,
eff)
            End If
        Else
            e0_cur2 = 0.01
            eff = 50
        End If
        Library.RAMP(e0_cur2, (e0_cur2 - e0_prev) * 25 / 8, (e0_prev - e0_cur2) * 25 / 8, 0, 0, False, False, e0_cur)
        'e0_prev = e0_cur
        e0_cur = e0_cur2
        a = eff / 100
        Library.CalculateRange(Max_Head, 0.01, e0_cur, e0_cur)
        Library.CalculateRange(1, 0.5, a, a)

        Library.Cp_to_Cv_Enthalpy(ArrInputGasWeights, ArrInputGasComposition, Pvhod, (Tvhod - 273.15), zvx, H_Hig, H_in, Cp_in, k_in) '
isentropic exponent
        k = k_in
        k_div_km1 = k / (k - 1) 'k/(k-1)
        n_div_nm1 = k_div_km1 * a 'n/(n-1), where n - Polytropic exponent

        *****CALCULATION OF Pvihoda(Discharge pressure), Tvihoda(Discharge Temperature)*****
        Dim Pvihoda_cur As Double = 0
        If (n >= 1) And (a > 0) Then
            Pvihoda = Pvhod * (1 + e0_cur / (zvx * R_1 * Tvhod * n_div_nm1 * 100)) ^ n_div_nm1 'Discharge Pressure, bar
            Library.RAMP(Pvihoda_cur, (Pvihoda_prev - Pvihoda_cur) * 25 / 8, (Pvihoda_cur - Pvihoda_prev) * 25 / 8, 0, 0, False, False,
Pvihoda)

            Tvihoda = Tvhod * (1 + (((Pvihoda / Pvhod) ^ ((k - 1) / k)) - 1) / a) 'Discharge temperature, K

```

```

'n-compressor speed (rev/min), n0-compressor rated speed(rev/min), e0-compression ratio, a-polytropic index
Else
    Tvihoda = Tvhod
    Pvihoda = Pvhod
End If
Pvihoda_prev = Pvihoda

*****CALCULATION OF DISCHARGE GAS DENSITY, [kg/m3]*****
Library.CriticalPT(ArrInputGasWeights, ArrInputGasComposition, R0, Pkr2, Tkr2, w, M2)
R_2 = 8.31451 / M2 //Gas constant for natural gas [kJ/(kg*K)]
d_2 = M2 / 28.96 //relative density of the natural gas by air [-]
pvx = Pvihoda / Pkr2 //reduced to normal conditions input pressure value [-]
tvx = Tvihoda / Tkr2 //reduced to normal conditions input temperature value [-]
'Library.Z_Factor_Redl_Kwong(Pvihoda, Tvihoda, Pkr2, Tkr2, zvix) //Compressibility factor by the discharge parameters [-]
Library.Z_Factor_Peng_Robinson(Pvihoda, Tvihoda, Pkr2, Tkr2, w, H_Hig, zvix) //Compressibility factor by the discharge
parameters [-]
Library.CalculateRange(10, 0.2, zvix, zvix)
pvix1 = (100 * Pvihoda) / (zvix * Tvihoda * R_2) //Gas density at the compressor inlet [kg/m^3]
Library.CalculateRange(10000, 0.0001, pvix1, pvix1)

*****CALCULATION OF DISCHARGE VOLUMETRIC FLOW*****
Q2 = Q1 * pvx1 / pvix1 'discharge volumetric flow, m3/h

Hn = e0_cur / 100 //polytropic head -----EDITED

Dim Compr_Cap_cur As Double
Compr_Cap_cur = ArrSpecParamsIn(3) * (n / n0) ^ 0.5
'Compr_Cap_cur = ArrSpecParamsIn(3) * ((n / n0) ^ 2)
Compr_Capacity = Rev_Capacity(0)
CompCap_prev = Compr_Capacity
'Ni = ((Hn / a) * (Q1 / 3600) * pvx1) 'compressor power, kWt
Ni = (Q1 * pvx1 / 3600) * (Hn * 100) * 9.8 / a / 1000 'compressor power, kWt

Call CalculateRange(10000000000, 1, hs, Q1)
If Speed > 0 Then
    torque = 9550 * Ni / Speed 'motor torque
End If

*****CONFIGURATION OF OUTPUT VARIABLE*****
Array.Copy(ArrInputParameters, ArrOutputParameters, 30)
ArrOutputParameters(0) = Pvihoda
Call CalculateRange(1000000000000, 0.0001, ArrOutputParameters(0), ArrOutputParameters(0))
'Call CalculateRange(100000, (Tvhod - 273.15), ArrOutputParameters(1), (Tvihoda - 273.15))
ArrOutputParameters(1) = Tvihoda - 273.15
Call CalculateRange(1000000000000, 0.0001, ArrOutputParameters(2), Q2)
ArrOutputParameters(3) = Pvhod
ArrOutputParameters(4) = Tvhod - 273.15
ArrOutputParameters(5) = n / 1000
ArrOutputParameters(6) = Q2

If ArrInputParameters(22) > Compr_Capacity Then
    ArrOutputParameters(22) = Compr_Capacity 'compressor capacity
End If
'ArrOutputParameters(23) = ArrOutputParameters(23) + (ArrOutputParameters(0) - ArrInputParameters(0))
ArrInputParameters(5) = Speed
ArrInputParameters(6) = Q1
ArrInputParameters(4) = Tvihoda - 273.15
ArrInputParameters(21) = n0
ArrInputParameters(24) = Hn
ArrInputParameters(19) = ArrOutputParameters(0)
Q2_prev = Q2
OutputGas_Weights = ArrInputGasWeights
OutputGas_Composition = ArrInputGasComposition

'PARAMETERS TO FACEPLATE'
Dim UnitCoef(.) As Double = New Double(30, 3) {}
Library.UnitCoeff(UnitCoef)
'Input parameters to FacePlate
ArrInputParameters_toFP(0) = ArrInputParameters(0) * UnitCoef(0, 0)
ArrInputParameters_toFP(1) = ArrInputParameters(1) * UnitCoef(1, 0) + UnitCoef(1, 1)
ArrInputParameters_toFP(2) = Q1 * UnitCoef(2, 0)
ArrInputParameters_toFP(3) = Q1 * pvx1 * UnitCoef(3, 0)
ArrInputParameters_toFP(4) = pvx1 * UnitCoef(4, 0)
ArrInputParameters_toFP(5) = M1 * UnitCoef(5, 0)
ArrInputParameters_toFP(6) = Pkr1 * UnitCoef(6, 0)

```

```

ArrInputParameters_toFP(7) = (Tkr1 - UnitCoef(7, 0)) * UnitCoef(7, 1) + UnitCoef(7, 2)
ArrInputParameters_toFP(8) = zvx * UnitCoef(8, 0)
ArrInputParameters_toFP(9) = R_1 * UnitCoef(9, 0)
ArrInputParameters_toFP(10) = d_1 * UnitCoef(10, 0)

```

'Output parameters to FacePlate

```

ArrOutputParameters_toFP(0) = ArrOutputParameters(0) * UnitCoef(0, 0)
ArrOutputParameters_toFP(1) = ArrOutputParameters(1) * UnitCoef(1, 0) + UnitCoef(1, 1)
ArrOutputParameters_toFP(2) = Q2 * UnitCoef(2, 0)
ArrOutputParameters_toFP(3) = Q2 * pvix1 * UnitCoef(3, 0)
ArrOutputParameters_toFP(4) = pvix1 * UnitCoef(4, 0)
ArrOutputParameters_toFP(5) = M2 * UnitCoef(5, 0)
ArrOutputParameters_toFP(6) = Pkr2 * UnitCoef(6, 0)
ArrOutputParameters_toFP(7) = (Tkr2 - UnitCoef(7, 0)) * UnitCoef(7, 1) + UnitCoef(7, 2)
ArrOutputParameters_toFP(8) = zvix * UnitCoef(8, 0)
ArrOutputParameters_toFP(9) = R_2 * UnitCoef(9, 0)
ArrOutputParameters_toFP(10) = d_2 * UnitCoef(10, 0)

```

```

If MyOwner.Unitttype = 0 Then
    ArrInputParameters_toFP(11) = Ni
    ArrInputParameters_toFP(12) = Hn * 100
    ArrInputParameters_toFP(13) = torque
    ArrInputParameters_toFP(14) = n

```

```

Elseif MyOwner.Unitttype = 1 Then
    ArrInputParameters_toFP(11) = Ni * 1.34102209
    ArrInputParameters_toFP(12) = Hn * 100 * 3.2808399
    ArrInputParameters_toFP(13) = torque
    ArrInputParameters_toFP(14) = n

```

```
End If
```

```

If Double.IsNaN(Ni) = True Then
    Ni = 0.0001
End If
If Double.IsNaN(Hn) Then
    Hn = 0.0001
End If
If Double.IsNaN(torque) Or torque = 0 Then
    torque = 0.0001
End If
If Double.IsNaN(n) Or n = 0 Then
    n = 0.0001
End If
If Double.IsNaN(a) Or a = 0 Then
    a = 0.0001
End If

```

```

If MyOwner.Unitttype = 0 Then
    ArrOutputParameters_toFP(11) = Ni
    ArrOutputParameters_toFP(12) = Hn * 100
    ArrOutputParameters_toFP(13) = torque
    ArrOutputParameters_toFP(14) = n
Elseif MyOwner.Unitttype = 1 Then
    ArrOutputParameters_toFP(11) = Ni * 1.34102209
    ArrOutputParameters_toFP(12) = Hn * 100 * 3.2808399
    ArrOutputParameters_toFP(13) = torque
    ArrOutputParameters_toFP(14) = n
End If

```

'Spec params

```

ArrSpecParams(0) = Hn * 100
ArrSpecParams(1) = Compr_Capacity / pvx1
ArrSpecParams(2) = Compr_Capacity
ArrSpecParams(3) = Ni
If Pvhod > 0 Then
    ArrSpecParams(4) = Pvihoda / Pvhod
End If
ArrSpecParams(5) = a * 100 'efficiency

```

Dim str As String

```

str = str & Me.Name & vbNewLine
str = str & " In- Out" & vbNewLine
str = str & "P,[bar]=" & Math.Round(ArrInputParameters(0), 2) & "-" & Math.Round(ArrOutputParameters(0), 2) & vbNewLine
str = str & "T,[C]=" & Math.Round(ArrInputParameters(1), 2) & "-" & Math.Round(ArrOutputParameters(1), 2) & vbNewLine
str = str & "F,[m3/h]=" & Math.Round(Q1, 2) & "-" & Math.Round(Q2, 2) & vbNewLine
str = str & "F,[kg/h]=" & Math.Round(Q1 * pvx1, 2) & "-" & Math.Round(Q2 * pvix1, 2) & vbNewLine

```

```

str = str & "Cap,[kg/h]=" & Math.Round(Rev_Capacity(0), 2) & vbNewLine
str = str & "Cap_direct,[kg/h]=" & Math.Round(ArrOutputParameters(22), 2) & vbNewLine

If ToolShow = False Then
    ToolTip1.SetToolTip(PictureBox1, str)
Else
    ToolTip1.Show(str, PictureBox1)
End If

M1 = Nothing
M2 = Nothing
Z_ci = Nothing

R0 = Nothing
Tvihoda = Nothing
Tvhd = Nothing
Pvhod = Nothing
Pvihoda = Nothing
Q1 = Nothing

Call RefreshFacePlate()
Catch ex As Exception
    MyOwner.StopTheTimer()
    MsgBox("Compressor - Second Step Error")
End Try
End Sub
Private Sub Compressor_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
    Try
        T0 = 293
        e0 = 1.2
        ToolStripTextBox1.Text = Me.Name
        Dim FacePlate As New FPCompressor
        FacePlate.WhoMeOwned_Controll = Me
        FacePlate.Owner = Me.ParentForm
        FacePlate.Temp = FacePlate
        PictureBox1.Location = New Point(0, 0)
        'DragAndDrop.Percent
        PictureBox1.Size = New Size(MyOwner.Percent * 70 / 100, MyOwner.Percent * 70 / 100)
        PictureBox1.Image = Img.GetThumbnailImage(MyOwner.Percent * 70 / 100, MyOwner.Percent * 70 / 100, Nothing, Nothing)
        Me.Size = New Size(MyOwner.Percent * 70 / 100, MyOwner.Percent * 82 / 100)
        Label1.Location = New Point(0, MyOwner.Percent * 70 / 100)
        Label1.Text = Me.Name
    Catch ex As Exception
        MsgBox("Compressor - Load Error")
    End Try
End Sub
Public FPopenNow As Boolean = False
Private Sub ПервоеМенюToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
    ПервоеМенюToolStripMenuItem.Click
    If FPopenNow = False Then
        Dim FacePlate As New FPCompressor

        If (ArrFlow Is Nothing) Then
            ReDim ArrFlow(5)
            ArrFlow = {17670, 20388, 22200, 23786, 26504, 27750}
        End If
        If (ArrPdtoPs Is Nothing) Then
            ReDim ArrPdtoPs(5)
            ArrPdtoPs = {16510, 16002, 15240, 14478, 12192, 10922}
        End If
        If (ArrEfficiency Is Nothing) Then
            ReDim ArrEfficiency(5)
            ArrEfficiency = {75, 76, 77, 78, 79, 80}
        End If

        For i = 0 To UBound(ArrFlow)
            FacePlate.ListView4.Items.Add(ArrFlow(i))
            FacePlate.ListView4.Items(i).SubItems.Add(ArrPdtoPs(i))
            FacePlate.ListView4.Items(i).SubItems.Add(ArrEfficiency(i))
        Next

        FacePlate.WhoMeOwned_Controll = Me
        FacePlate.Owner = Me.ParentForm

        FacePlate.Show(Me.ParentForm)
        FacePlate.Name = "FP" & Me.Name
        FacePlate.Temp = FacePlate
        FPopenNow = True
    End If
End Sub

```



```

Else
    FacePlateTemp.Focus()
End If
End Sub

Private Sub ВтороеМенюToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
    ВтороеМенюToolStripMenuItem.Click
    If Runtime = True Then
        Dim FacePlate As New FPCurves
        If FPCurvesopenNOW = False Then
            Dim Obj As Antisurge_control
            Obj = MyOwner.Controls.Item("Panel1").Controls.Item(OutputASCtrl)
            If (Obj.Flow_Params Is Nothing) Then
                ReDim FacePlate.Flow_Params(7)
                ReDim FacePlate.Pd_Params(7)
                Obj.Flow_Params = {1, 10000, 15000, 20000, 25000, 27000, 30000} 'FLOW values for AS curve
                Obj.Pd_Params = {0.1, 10000, 15000, 21000, 27000, 32000, 41000} 'Discharge pressure value for AS curve
            End If
            FacePlate.WhoMeOwned_Control = Me.Parent.Controls.Item(OutputASCtrl)
            FacePlate.Owner = Me.ParentForm
            FacePlate.CompressorControl = Me
            FacePlate.Name = "FPAS" & Me.Name
            FacePlate.Show(Me.ParentForm)
            FacePlateTempAS = FacePlate
            FPCurvesopenNOW = True
        Else
            FacePlateTempAS.Focus()
        End If
    Else
        MsgBox("For opening this Antisurge control window go to runtime", MsgBoxStyle.Information, "Open Antisurge control")
    End If
End Sub

Private Sub DeleteToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
    DeleteToolStripMenuItem.Click
    MyOwner.LastAction("Delete")
End Sub

Private Sub ToolStripTextBox1_KeyDown(ByVal sender As System.Object, ByVal e As System.Windows.Forms.KeyEventArgs) Handles
    ToolStripTextBox1.KeyDown
    If e.KeyCode = Keys.Enter Then
        Dim FForm As New DragAndDrop
        FForm = Me.Parent
        FForm.CheckControlName(Me.Name, ToolStripTextBox1.Text, Me)
        Me.Name = ToolStripTextBox1.Text
        ToolStripTextBox1.Enabled = False
    End If
End Sub

Private Sub RenameToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
    RenameToolStripMenuItem.Click
    ToolStripTextBox1.Enabled = True
    ContextMenuStrip1.Show(QPoint)
    ToolStripTextBox1.Focus()
End Sub

Public FPTrendopenNow As Boolean = False

Private Sub PictureBox1_MouseDoubleClick(ByVal sender As Object, ByVal e As System.Windows.Forms.MouseEventArgs) Handles
    PictureBox1.MouseDoubleClick
    If FPopenNow = False Then
        Dim FacePlate As New FPCompressor

        If (ArrFlow Is Nothing) Then
            ReDim ArrFlow(5)
            ArrFlow = {17670, 20388, 22200, 23786, 26504, 27750}
        End If
        If (ArrPdtoPs Is Nothing) Then
            ReDim ArrPdtoPs(5)
            ArrPdtoPs = {16510, 16002, 15240, 14478, 12192, 10922}
        End If
        If (ArrEfficiency Is Nothing) Then
            ReDim ArrEfficiency(5)
            ArrEfficiency = {75, 76, 77, 78, 79, 80}
        End If

        For i = 0 To UBound(ArrFlow)
            FacePlate.ListView4.Items.Add(ArrFlow(i))
        Next
    End If
End Sub

```

```

        FacePlate.ListView4.Items(i).SubItems.Add(ArrPdtoPs(i))
        FacePlate.ListView4.Items(i).SubItems.Add(ArrEfficiency(i))
    Next

    FacePlate.WhoMeOwned_Controll = Me
    FacePlate.Owner = Me.ParentForm

    FacePlate.Show(Me.ParentForm)
    FacePlate.Name = "FP" & Me.Name
    FacePlateTemp = FacePlate
    FOpenNow = True
Else
    FacePlateTemp.Focus()
End If
End Sub
Private Sub PictureBox1_MouseEnter(ByVal sender As Object, ByVal e As System.EventArgs) Handles PictureBox1.MouseEnter
    If ((Control.ModifierKeys And Keys.Shift) = Keys.Shift) And Runtime = False Then
        MyOwner.Connector.DeletSlots()
        MyOwner.RedirectToCr8(Me, IndexNum)
    End If
End Sub
Private Sub Label1_MouseClick(ByVal sender As Object, ByVal e As System.Windows.Forms.MouseEventArgs) Handles
Label1.MouseClick
    If ToolShow = True Then
        ToolShow = False
        ToolTip1.Hide(PictureBox1)
        Label1.ForeColor = Color.Black
    Else
        ToolShow = True
        Label1.ForeColor = Color.Red
    End If
End Sub
Private Sub PictureBox1_MouseLeave(ByVal sender As Object, ByVal e As System.EventArgs) Handles PictureBox1.MouseLeave
    If ToolShow = False Then
        ToolTip1.Hide(PictureBox1)
    End If
End Sub
Private Sub ToolStripMenuItem1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles ToolStripMenuItem1.Click
    Swap()
End Sub
Public Sub Swap()
    Dim BM As New Bitmap(PictureBox1.Image)
    If Rotate = True Then
        ToolStripMenuItem1.Text = "Left to Right: True"
        Rotate = False
        BM.RotateFlip(RotateFlipType.RotateNoneFlipX)
        PictureBox1.Image = BM
        Img = BM
    Else
        ToolStripMenuItem1.Text = "Left to Right: False"
        Rotate = True
        BM.RotateFlip(RotateFlipType.RotateNoneFlipX)
        PictureBox1.Image = BM
        Img = BM
    End If
End Sub

Private Sub Compressor_Resize(ByVal sender As Object, ByVal e As System.EventArgs) Handles Me.Resize
    Try
        PictureBox1.Location = New Point(0, 0)

        PictureBox1.Size = New Size(Me.Width, ((Me.Width + Me.Height) / 2) * 70 / ((ObjHeight + ObjWidth) / 2))
        PictureBox1.Image = Img.GetThumbnailImage(Me.Width, _
            ((Me.Width + Me.Height) / 2) * 70 / ((ObjHeight + ObjWidth) / 2), Nothing, Nothing)
        Label1.Location = New Point(0, _
            ((Me.Width + Me.Height) / 2) * 70 / ((ObjHeight + ObjWidth) / 2))
        Label1.Font = New Font("Microsoft Sans Serif", (((Me.Width + Me.Height) / 2) * 6 / 70))
    Catch ex As Exception
        MsgBox("Can't change size of Compressor")
    End Try
End Sub

Public Hided As Boolean = False
Public Sub HideNam()
    If MyOwner.HidedNemes = True And Hided = False Then
        ObjHeight -= 12
        Hided = True
    End If
End Sub

```