МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение

высшего образования

«Новосибирский национальный исследовательский государственный университет»

(Новосибирский государственный университет, НГУ)

Структурное подразделение Новосибирского государственного университета –

Высший колледж информатики Университета (ВКИ НГУ)

КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

1. Направление подготовки: 09.03.01 Информатика и вычислительная техника
2. Образовательная программа: 09.03.01 Информатика и вычислительная техника

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА**

РАЗРАБОТКА АДАПТИВНОЙ СИСТЕМЫ АВТОМАТИЗАЦИИ УСКОРИТЕЛЬНОГО ИСТОЧНИКА ЭПИТЕПЛОВЫХ НЕЙТРОНОВ

утверждена приказом по ВКИ НГУ № 02/3-204 от «27» апреля 2017 г.

|  |  |  |  |
| --- | --- | --- | --- |
| Кошкарева Алексея Михайловича, | группа 14214 | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| (фамилия, имя, отчество студента) |  | (подпись студента) |  |

|  |  |  |
| --- | --- | --- |
| **«К защите допущена»** |  | **Руководитель ВКР** |
| Заведующий кафедрой, к.ф-м.н |  | д.физ.-мат. наук. , в.н.с |
| (ученая степень, звание) |  | (ученая степень, звание) |
| Попов Л.К, /\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | Таскаев С.Ю., /\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| (ФИО) / (подпись) |  | (ФИО) / (подпись) |
| «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_г. |  | «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_г. |

Дата защиты: «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_ 20\_\_г.

Новосибирск

2017

СОДЕРЖАНИЕ

[ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ 5](#_Toc493528327)

[ВВЕДЕНИЕ 6](#_Toc493528328)

[1. ПОСТАНОВКА ЗАДАЧИ 7](#_Toc493528329)

[1.1. Описание предметной области 7](#_Toc493528330)

[1.2. Формулировка задачи 7](#_Toc493528331)

[1.3. Функциональные требования 8](#_Toc493528332)

[1.4. Характеристики выбранных технических средств 8](#_Toc493528333)

[1.4.1. Пирометр Optris 3ML SF 8](#_Toc493528334)

[1.4.2. Потокомер Proteus 04004SN2 TPD 9](#_Toc493528335)

[1.4.3. Блок питания PSU 6-200 9](#_Toc493528336)

[1.4.4. Внешняя периферия ADAM 6015 9](#_Toc493528337)

[1.4.5. Внешняя периферия ADAM 6024 9](#_Toc493528338)

[1.4.6. Внешняя периферия ADAM 6066 9](#_Toc493528339)

[1.5. Описание ускорителя 10](#_Toc493528340)

[2. АЛГОРИТМЫ РЕШЕНИЯ ЗАДАЧИ 12](#_Toc493528341)

[2.1. Начальная конфигурация ускорителя 12](#_Toc493528342)

[2.2. Предлагаемая конфигурация 14](#_Toc493528343)

[2.3. Ведение журнала 15](#_Toc493528344)

[2.3.1. Общий принцип 15](#_Toc493528345)

[2.3.2. Ведение журнала в устаревшей программе 16](#_Toc493528346)

[2.3.3. Перехват ошибок в программе 16](#_Toc493528347)

[2.4. Конфигурация программы 16](#_Toc493528348)

[3. АРХИТЕКТУРА ПРОГРАММЫ УПРАВЛЕНИЯ 18](#_Toc493528349)

[3.1. Устройства 19](#_Toc493528350)

[3.1.1. Охлаждаемые диафрагмы 19](#_Toc493528351)

[3.1.1.1. Принцип работы 19](#_Toc493528352)

[3.1.1.2. Измерение по трем точкам 22](#_Toc493528353)

[3.1.1.3. Звуковое оповещение 23](#_Toc493528354)

[3.1.2. Пирометр 23](#_Toc493528355)

[3.1.3. Шиберы 25](#_Toc493528356)

[3.1.4. Пищалка 300В 26](#_Toc493528357)

[3.1.5. Измеритель мощности на Li мишени 26](#_Toc493528358)

[3.1.6. Ионный источник 28](#_Toc493528359)

[3.1.7. Ребутатор 29](#_Toc493528360)

[3.1.8. Тепловизор Flir T650SC 29](#_Toc493528361)

[3.1.9. Пути развития системы 30](#_Toc493528362)

[3.2. Каналы 30](#_Toc493528363)

[3.2.1. Реальные каналы 33](#_Toc493528364)

[3.2.2. Виртуальные каналы 33](#_Toc493528365)

[3.3. Модули 33](#_Toc493528366)

[3.4. Утилиты 33](#_Toc493528367)

[3.5. Разработанная программа управления 34](#_Toc493528368)

[3.5.1. Блок «Основное» 34](#_Toc493528369)

[3.5.2. Блок «Управление» 35](#_Toc493528370)

[3.5.3. Блок «Флюенс» 36](#_Toc493528371)

[3.5.4. Блок «Ребутатор» 37](#_Toc493528372)

[3.5.5. Блок «Источник H-» 38](#_Toc493528373)

[3.5.6. Блок «Все каналы» 39](#_Toc493528374)

[4. РАЗРАБОТКА БИБЛИОТЕК 41](#_Toc493528375)

[4.1. ModuleWizard 41](#_Toc493528376)

[4.1.1. Общий принцип 41](#_Toc493528377)

[4.1.2. Пример 43](#_Toc493528378)

[4.2. LogWizard 43](#_Toc493528379)

[4.2.1. FormLog 43](#_Toc493528380)

[4.2.2. TimeLog 44](#_Toc493528381)

[4.3. GraphWizard 44](#_Toc493528382)

[4.3.1. Компонент векторный радар 45](#_Toc493528383)

[4.3.2. Компонент колонка с вводом 46](#_Toc493528384)

[4.3.3. Компонент лампочка 47](#_Toc493528385)

[4.3.4. Компонент спидометр 47](#_Toc493528386)

[4.3.5. Компонент график реального времени 48](#_Toc493528387)

[5. ОТЛАДКА И ТЕСТИРОВАНИЕ 50](#_Toc493528388)

[ЗАКЛЮЧЕНИЕ 51](#_Toc493528389)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 52](#_Toc493528390)

[6. ПРИЛОЖЕНИЕ А 53](#_Toc493528391)

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

|  |  |
| --- | --- |
| **БНЗТ** | Бор-нейтронозахватная терапия; |
| **Система** **управления** | Совокупность модулей сбора данных, программного обеспечения и базы данных |
| **Децентрализованная система управления** | Это система управления, у которой модули сбора данных расположены рядом с объектом измерения. При этом их много, и рядом с каждым объектом измерения стоит свой измеритель. |
| **Периферия** | Опрашиваемое или управляемое устройство; |
| **АЦП** | Аналого-цифровой преобразователь; |
| **ЦАП** | Цифро-аналоговый преобразователь; |
| **ЦВх** | Цифровой вход (однобитный АЦП); |
| **ЦВых** | Цифровой выход (однобитный ЦАП). |

ВВЕДЕНИЕ

Целью данной работы является реализация автоматизированной системы управления, контролирующей оборудование во время эксперимента. В ИЯФ СО РАН осуществляется разработка ускорительного источника нейтронов [1, 2], предназначенного для проведения бор-нейтронозахватной терапии [1] (БНЗТ) злокачественных опухолей в условиях онкологической клиники. Данный метод терапии очень эффективен в отношении ряда неизлечимых в настоящее время радиорезистентных опухолей, например, таких, как глиобластома мозга и метастазы меланомы. Все существующие исследования БНЗТ успешно проведены при помощи ядерных реакторов, но из-за их нестабильности требуется создание безопасного ускорителя. Именно такой ускоритель разрабатывается в Институте ядерной физики, и для его дальнейшего развития требуется создание новой системы автоматизации на базе нового оборудования.

Спецификой ускорителя является ее постоянная модернизация и внедрение диагностик разного типа. Для стабильной работы ускорителя необходимо создание гибкой и масштабируемой системы автоматизации, позволяющей управлять подготовкой к работе, осуществлять проведение необходимых экспериментов и выключать ускоритель после работы c минимальным участием оператора.

Отсутствие гибкой, масштабируемой и централизованной системы управления сдерживает внедрение ускорителя в клиническую практику. Решение этой проблемы описывается в данной работе.

1. ПОСТАНОВКА ЗАДАЧИ
   1. Оп**исание предметной области**

Ускоритель состоит из множества устройств, таких как: вакуумные насосы, шиберы, источники питания, измерительное оборудование и прочие элементы. Для получения на ускорителе оптимальных параметров, позволяющих проводить бор-нейтронозахватную терапию, требуется децентрализованная и надежная система управления (далее система), обеспечивающая контроль и сбор данных над всеми узлами ускорителя.

Текущая конфигурация ускорителя не позволяет быстро добавить опрашиваемое / управляемое устройство (далее периферия), что неприемлемо в условиях постоянной модификации ускорителя. Поэтому требуется разработка и внедрение новой системы управления.

* 1. Формулировка задачи

Целью работы являлась реализация системы автоматизированного управления ускорителем БНЗТ, а именно: разработка программы оператора (далее консоль), сопряжение периферии с консолью и отображение данных оператору.

Контроль управляющих узлов источниками питания должен осуществляться при помощи заложенных в них протоколов связи (например, Standard Commands for Programmable Instruments (SCPI)). Диагностика состояния ускорителя осуществляется через измерение токов, напряжений и температур.

В качестве измерителей параметров были выбраны промышленный дистанционные устройства ввода-вывода, управляемые консолью оператора через Ethernet (протокол Modbus).

Так-же требуется автоматическое ведение журнала всех измерений с периодом не менее 1 раза в секунду.

Были поставлены и решены следующие задачи:

* Изучена текущая конфигурация ускорителя;
* Произведен анализ требуемых устройств;
* Изучены принципы работы с периферийными устройствами;
* Разработана консоль оператора;
* Получен опыт отладки многопоточных приложений;
* Налажена связь консоли оператора с устройствами управления и диагностики;
* Разработана база данных для хранения данных об эксперименте.

Программирование осуществлялось на языке «C#» с использованием следующих библиотек:

* графической библиотеки ChartView;
* Modbus;
* SCPI;
* SQL client.

В процессе создания программы управления была использована система контроля версий git, а также был создан личный репозиторий на RaspberryPi.

* 1. Функциональные требования

Система должна соответствовать следующим требованиям:

* Автоматически поддерживать обновления с сервера;
* обладать длительной и стабильной работоспособностью;
* работать в условии постоянных пробоев и электромагнитных наводок;
* система должна быть легко масштабируемой;
* обладать возможностью применения в других задачах;
* в удобной для оператора форме выводить данные на экран;
* система должна состоять из легко заменяемых узлов ввода / вывода;
* система должна записывать журнал всех измерений в базу данных.
  1. Характеристики выбранных технических средств

В процессе разработки системы было принято решение использовать промышленные устройства ввода / вывода и блоки питания, проверенные производителем.

* + 1. Пирометр Optris 3ML SF

Для дистанционного измерения температуры на облучаемой мишени был выбран пирометр с интерфейсом связи ETHERNET Optris 3ML SF с диапазоном температур от 50 до 400 С°. В системе он виден как COM порт, и общение происходит путем приема / передачи управляющих команд в виртуальный COM порт.

* + 1. Потокомер Proteus 04004SN2 TPD

Для измерения потока воды был использован потокомер Proteus 04004SN2 TPD с диапазоном измерения 0.8 – 9.5 литров / мин. Значение потока требуется снимать аналого-цифровым преобразователем (АЦП). В качестве АЦП был выбран ADAM 6024.

* + 1. Блок питания PSU 6-200

Для проведения эксперимента на новом ионном источнике был выбран промышленный блок питания Gwinstek PSU 6-200 6V 200A 1200W с интерфейсом связи ETHERNET и протоколом SCPI.

* + 1. Внешняя периферия ADAM 6015

Для измерения температур были выбраны промышленные модули ADAM 6015 с интерфейсом связи ETHERNET и протоколом Modbus. Они имеют 7 каналов измерения температуры RTD (resistance temperature detector). Так же для сравнения были использованы модули ICPCON PET Pet7215 с точно такими же характеристиками для сравнения производителей по качеству. На ускорителе используются резисторы типа PT100 (ЧЭПТ-1) с трехпроводной схемой подключения (подробнее в п. 3.1.5).

* + 1. Внешняя периферия ADAM 6024

Для диагностики напряжений, токов и логических входов были выбраны промышленные модули ADAM 6024 с интерфейсом связи ETHERNET и протоколом Modbus. Они имеют 6 АЦП, 2 ЦАП, 2 ЦВх,.2 ЦВых.

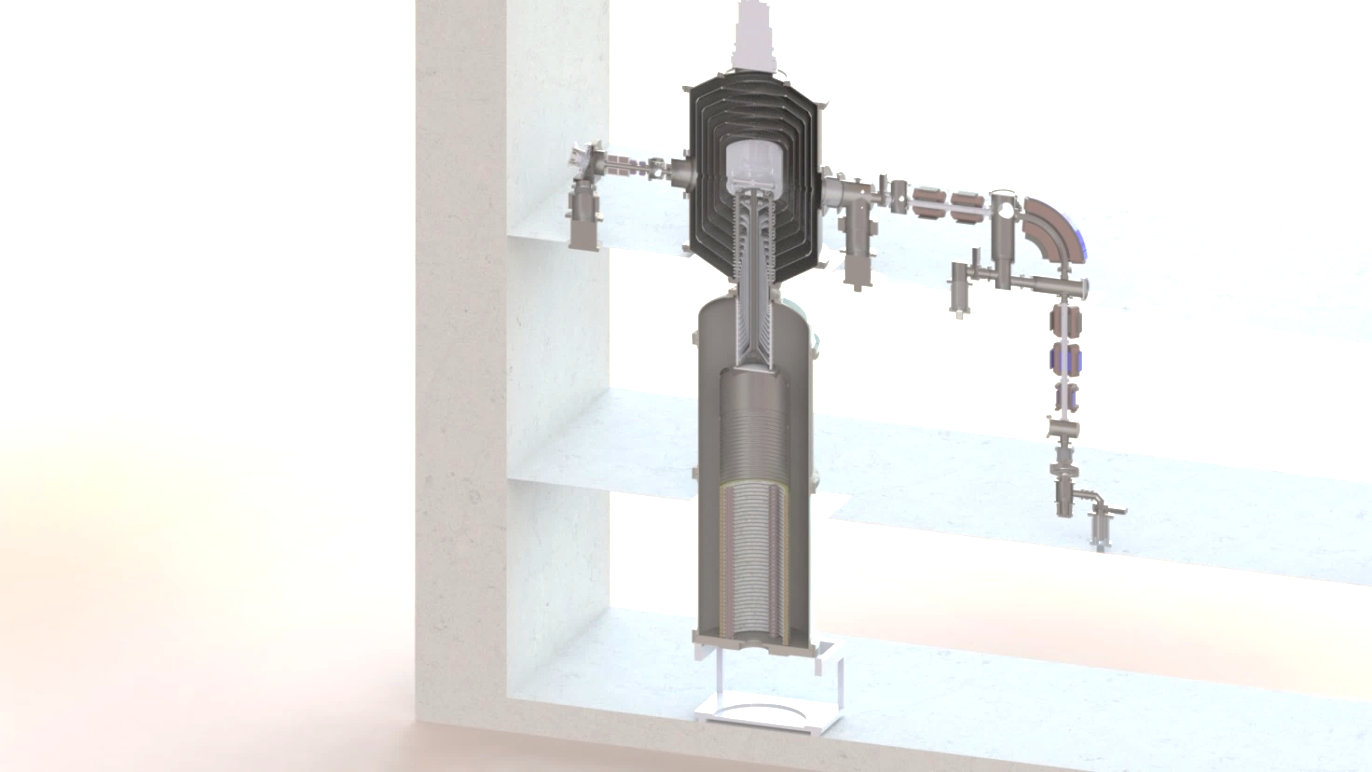
* + 1. Внешняя периферия ADAM 6066

Для управления шиберами, напуском аргона и блоком питания супрессора вторичной эмиссии были выбраны промышленные модули ADAM 6066 с интерфейсом связи ETHERNET и протоколом Modbus. Они имеют 6 каналов реле и 6 каналов цифровых входов (ЦВх).

* 1. Описание ускорителя

Ускоритель расположен на территории ИЯФ СО РАН в специальном радиационно-защищенном бункере. Его схематичное описание, габариты и фотография представлены на рис. 1, рис. 2 и рис. 3.

Обслуживание ускорителем затрудняется тем, что в процессе эксперимента радиационный фон может достигать единицы зиверт. Это означает, что за 10 минут человек получит примерно 200 годовых доз. Кроме того, в процессе работы некоторые узлы находятся под потенциалом 1 000 000 вольт. Поэтому все манипуляции должны происходить удаленно.



Ионный источник

Ускоритель

Поворотный магнит

Фокусирующие линзы

Источник на 1 МВ

Li мишень

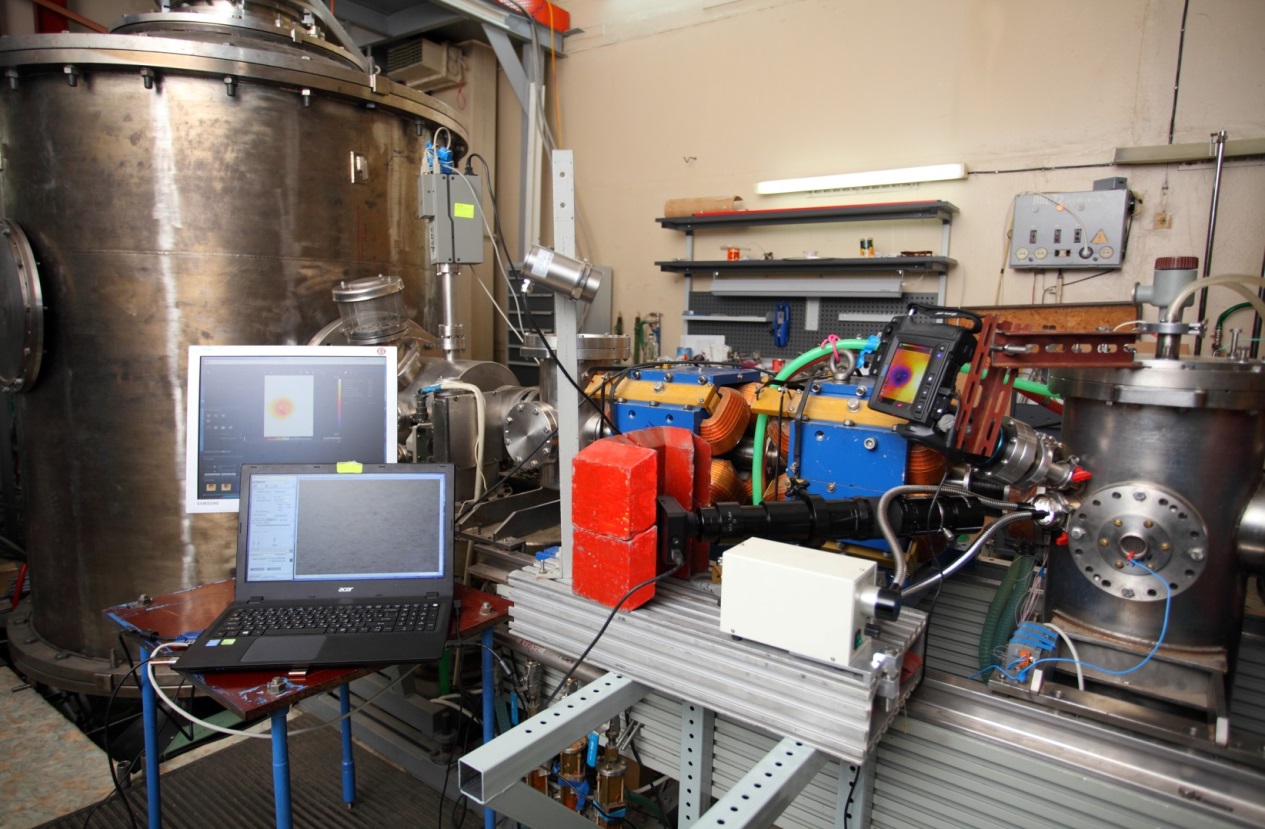
10 метров

6 метров

1. Схема экспериментального ускорителя



1. Фото экспериментального ускорителя



Пирометр

Газовый анализатор

Тепловизор

Компьютер управления

Охлаждаемая диафрагма

Ускоритель

1. Расположение устройств (выскоэнергитичный тракт)
2. АЛГОРИТМЫ РЕШЕНИЯ ЗАДАЧИ
   1. Начальная конфигурация ускорителя

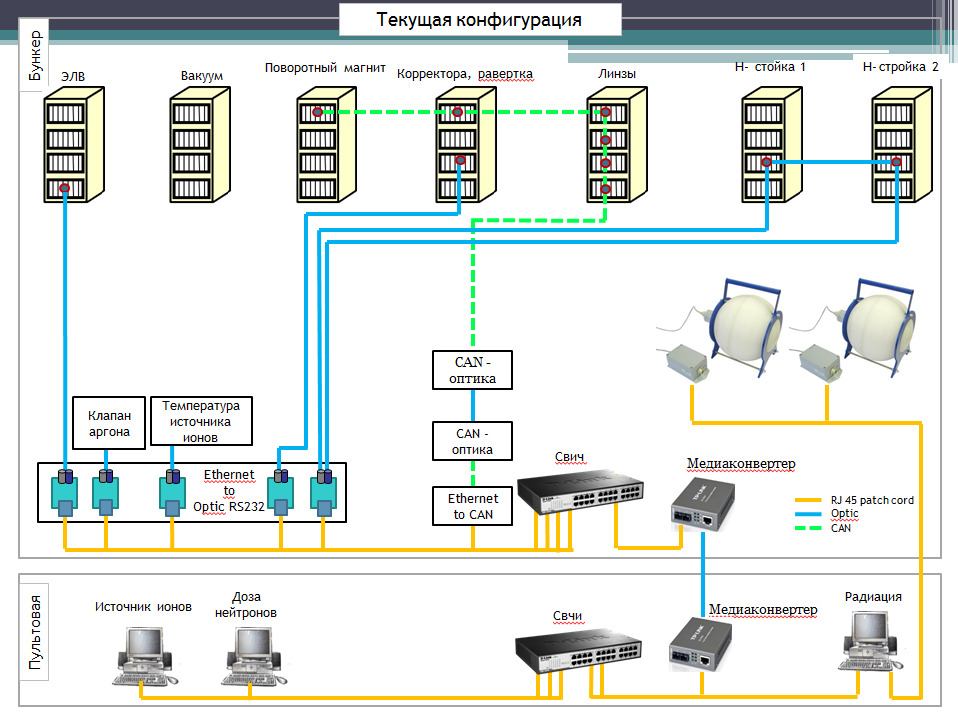
Изначально весь ускоритель был разработан из модулей, произведенных в ИЯФ СО РАН. Система управления проработала десять лет, что показывает ее надежность, но в настоящий момент она устарела. При попытке ее модернизировать или просто починить возникают проблемы и новые неполадки. В разработку этой системы было вовлечено 3 лабораторий и более 15 человек. По истечении 10 лет некоторые сотрудники уже уволились, а другие – уже не помнят, как устроена система. Например, для измерения радиации используется старая система измерения, которая работает на Windows 95 (рис. 4).

1. Программа измерения радиации на Windows 95

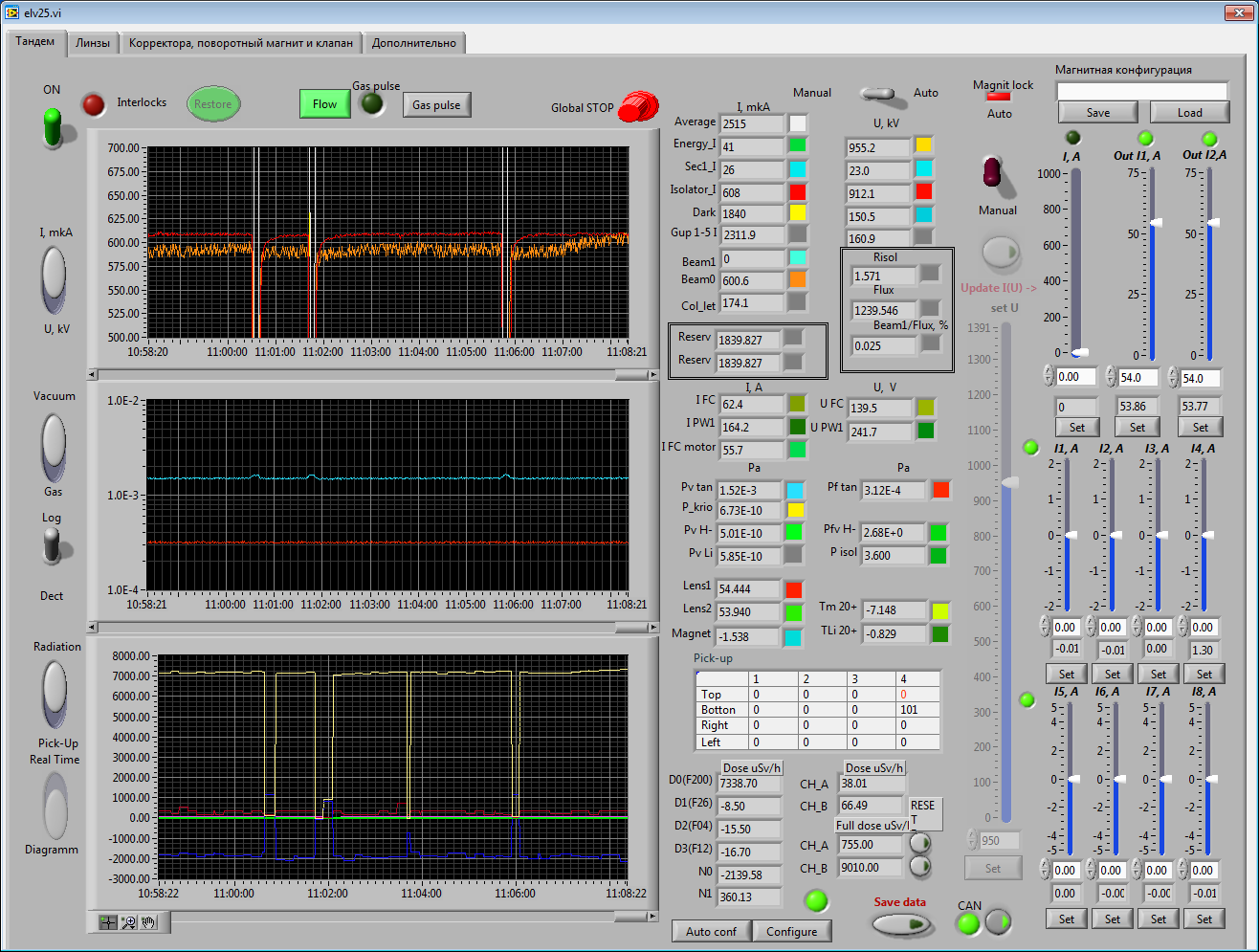
Система разработана таким образом, что часть критичных узлов была не защищена от электромагнитных наводок, и из-за этого зависала.

Устаревшая система имеет централизованный характер. Это означает, что все измерения приходят в стойку измерения в центре зала. При измерении напряжения на другом конце ускорителя при пробоях на длинном проводе образуются электромагнитные наводки, которые значительно уменьшают точность измерения, они могут также вывести АЦП из строя. Упрощенная схема расположения узлов показана на рис. 5.



1. Упрощенная схема расположения узлов устаревшей системы

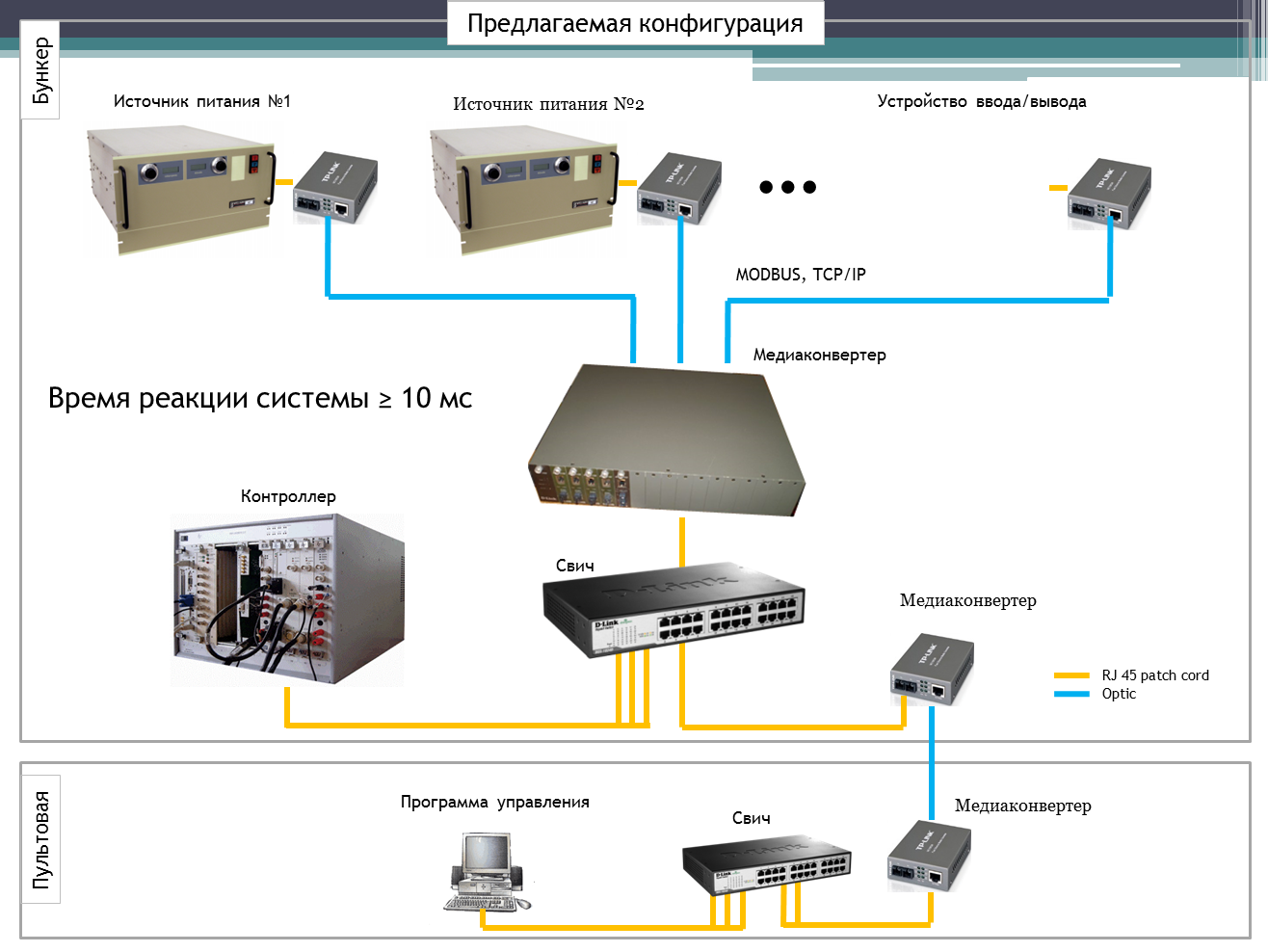
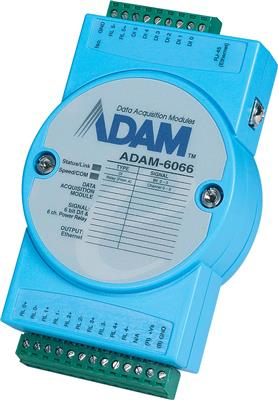
Интерфейс программы управления разработан без интуитивно понятного интерфейса. При этом в программе все еще указываются диагностики, которые были исключены пять лет назад. Человек, который разрабатывал устаревшую программу, уволился, а поддержка устаревшей программы занимает больше времени, чем создание новой. Программа приведена на рис. 6.



1. Устаревшая программа управления
   1. Предлагаемая конфигурация

Предлагаемая система управления имеет децентрализованный характер. Децентрализованная система управления – это система управления, у которой модули сбора данных расположены рядом с объектом измерения. При этом их много, и рядом с каждым объектом измерения стоит свой измеритель.

В процессе разработки системы было принято решение использовать промышленные устройства ввода / вывода и блоки питания, проверенные производителем. Общая структура измерения и контроля ускорителем на базе новой системы управления представлена на рис. 7.



1. Общая структура новой системы управления

В данной работе решается задача создания и обкатки новой системы управления не на всей, а только на части ускорителя, так как в настоящее время проводятся эксперименты, и времени на полную замену нет. Полная замена оборудования будет произведена только после серии тестов из-за соображений безопасности.

* 1. Ведение журнала
     1. Общий принцип

Для ведения журнала измерений была использована разработанная ранее библиотека LogWizard (подробнее в п. 4.2). Запись журнала ведется локально на компьютер в файл Excel и в базу данных, настроенной на мини-компьютере OrangePi. Запись файлов в Excel файл ведется из-за исторических соображений. Иногда физикам проще взять Excel файл и работать с ним, чем делать запрос в SQL.

Поэтому в настоящий момент ведется разработка утилиты выборки данных из базы данных и конвертация их в файл Excel.

Кроме значений системы в журнал регистрируются сообщения о текущем состоянии системы, которое отображается внизу экрана на панели оператора. Пример сообщений представлен на рис. 8.



1. Окно состояния системы управления
   * 1. Ведение журнала в устаревшей программе

Так как на ускорителе все-еще используется старая программа, требуется записывать ее данные в ту же базу данных. Для этого приходится анализировать файл логов, который создает старая программа.

Для решения этой задачи первым делом открывался файл с журналом этой программы. После этого файл анализировался, и отправлялся в базу данных. Однако когда старая программа записывала данные, а программа-анализатор пыталась прочитать – возникал конфликт ресурсов. В этот момент старая программа выводила сообщение об ошибке и зависала, а из-за этого ускоритель оставался без контроля на неопределенный срок.

В ближайшее время будет модифицирован код устаревшей программы так, что она будет сама писать данные в базу данных.

* + 1. Перехват ошибок в программе

Для удобства отладки программы был разработан механизм сохранения лога о возникшей ошибке на компьютер. После сохранения лога происходит отправка письма на электронный адрес разработчика с описанием проблемы. Это позволяет разработчику всегда быть в курсе текущей ситуации с программой и с ускорителем.

* 1. Конфигурация программы

Созданная программа поддерживает возможность считывать конфигурацию из файла. Был выбран конфигурационного файла был выбран XML. Это позволило хранить конфигурацию в иерархической структуре, описать каждый узел с определенным набором опций и предоставить файл в удобном для оператора виде.

В случае выхода какого-нибудь устройства из строя оператору достаточно переподключить измерение на другой модуль в свободный канал, подправить конфигурацию и продолжить работу. Обученному оператору потребуется не более 10 минут на переконфигурирование системы.

1. АРХИТЕКТУРА ПРОГРАММЫ УПРАВЛЕНИЯ

Для масштабируемости системы была разработана архитектура на парадигме объектно-ориентированного программирования (ООП). Основная идея заключается в том, что разделить работу на три уровня:

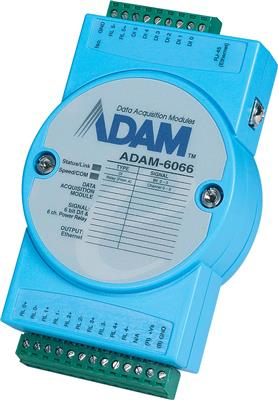
1. Устройства;
2. Каналы;
3. Модули.

Модули – это устройства ввода / вывода.

Устройства – совокупность модулей, объединенных для диагностики и управления определенного узла ускорителя.

А канал – связующая часть между первым и вторым.

В качестве примера рассмотрим измеритель мощности (о нем подробнее в п. 3.1.5). Это устройство, которое измеряет мощность Li мишени по входной температуре, выходной температуре и потоку воды. После измерения, параметры рассчитываются по формуле, и оператору выводится только рассчитанная мощность. Однако при желании всегда можно посмотреть «сырые данные». Логическая схема измерителя мощности представлена на рис. 9.



Измеритель мощности

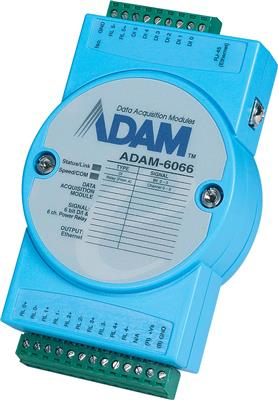
Li

мишень

Вода охлаждения



Измеритель температуры



Измеритель напряжения



**Устройство**

**Каналы**

**Модули**

Входная температура

Выходная температура

Значение потока

Вычисление мощности

1. Логическая схема измерителя мощности

Идея в том, что устройству не важно, откуда и каким образом считываются данные. Можно в любой момент начать считывать значения температур с двух разных модулей, но с точки зрения устройства ничего не поменялось. Пример эквивалентной системы представлен на рис. 10.

Измеритель мощности

Li

мишень

Вода охлаждения



Измерители температуры

Измеритель напряжения



**Устройство**

**Каналы**

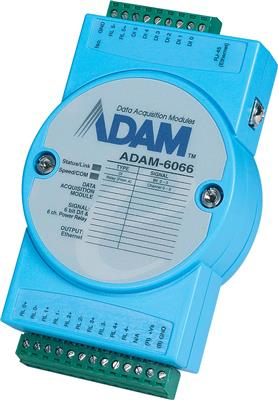
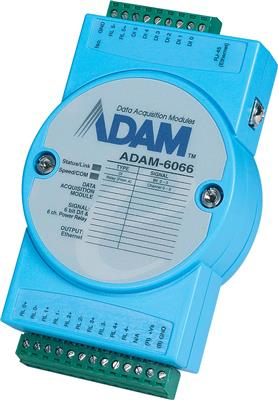
**Модули**

Входная температура

Выходная температура

Значение потока

Вычисление мощности



1. Эквивалентная схема измерителя мощности

На рис. 9 и рис. 10 изображены эквивалентные схемы. Для уровня устройства ничего не изменилось. За сбор данных отвечает канальный уровень. Он опрашивает все модули и передает информацию на уровень устройства.

Каждый уровень более детально описан в следующих пунктах.

* 1. Устройства

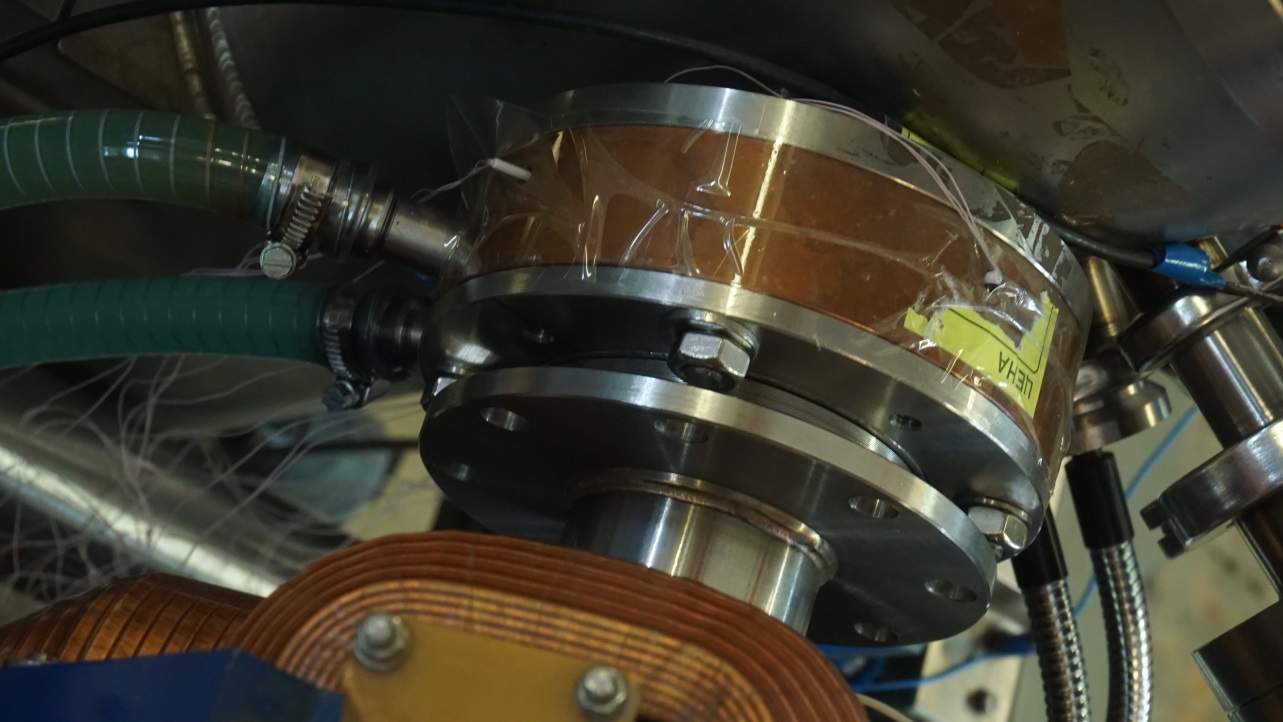
Устройства – совокупность модулей, объединенных для диагностики и управления определенного узла ускорителя. Этот узел может состоять из множества разных компонент, но предназначение у него одно. Иногда устройство может состоять из вложенных устройств. Такая возможность была заложена для объединения схожих узлов в одну категорию. Далее перечислены узлы ускорителя, которые реализованы в программе управления.

* + 1. Охлаждаемые диафрагмы
       1. Принцип работы

Охлаждаемые диафрагмы – это своего рода ограничители, которые отсекают пучок более 30 мм. Пока установлена только одна диафрагма, но в дальнейшем они будут стоять по всему тракту пучка. Фотография диафрагмы приведена на рис. 11 и рис. 12.



1. Фотография охлаждаемой диафрагмы.



1. Фотография установленной охлаждаемой диафрагмы.

В диафрагме установлено 4 термодатчика Pt100 для измерения отклонения пучка. Два датчика расположены на горизонтали, и два по вертикали. Схематическое расположение датчиков показано на рис. 13.

Отклонение пучка определяется по следующим формулам:

Таким образом, зная отклонение в градусах и длину вектора, отклонение можно отобразить оператору, что и было реализовано в данной работе. Панель отклонения пучка представлена на рис. 13.



1. Панель отклонения пучка.
   * + 1. Измерение по трем точкам

В один момент потребовалось диагностировать отклонение пучка после поворотного магнита. Для этого на тракт пучка было установлено 3 термодатчика, так как на момент установки диагностики был только один модуль измерения температуры с 7 каналами. Четыре уже были заняты охлаждаемой диафрагмой, оставалось только три. Схематическое расположение датчиков показано на рис. 13. Формулы расчета отклонения представлены ниже:

* + - 1. Звуковое оповещение

В некоторых ситуация критично, когда пучок отклоняется от оси тракта. Для оповещения оператора был разработан алгоритм, который оповещает оператора об отклонении.

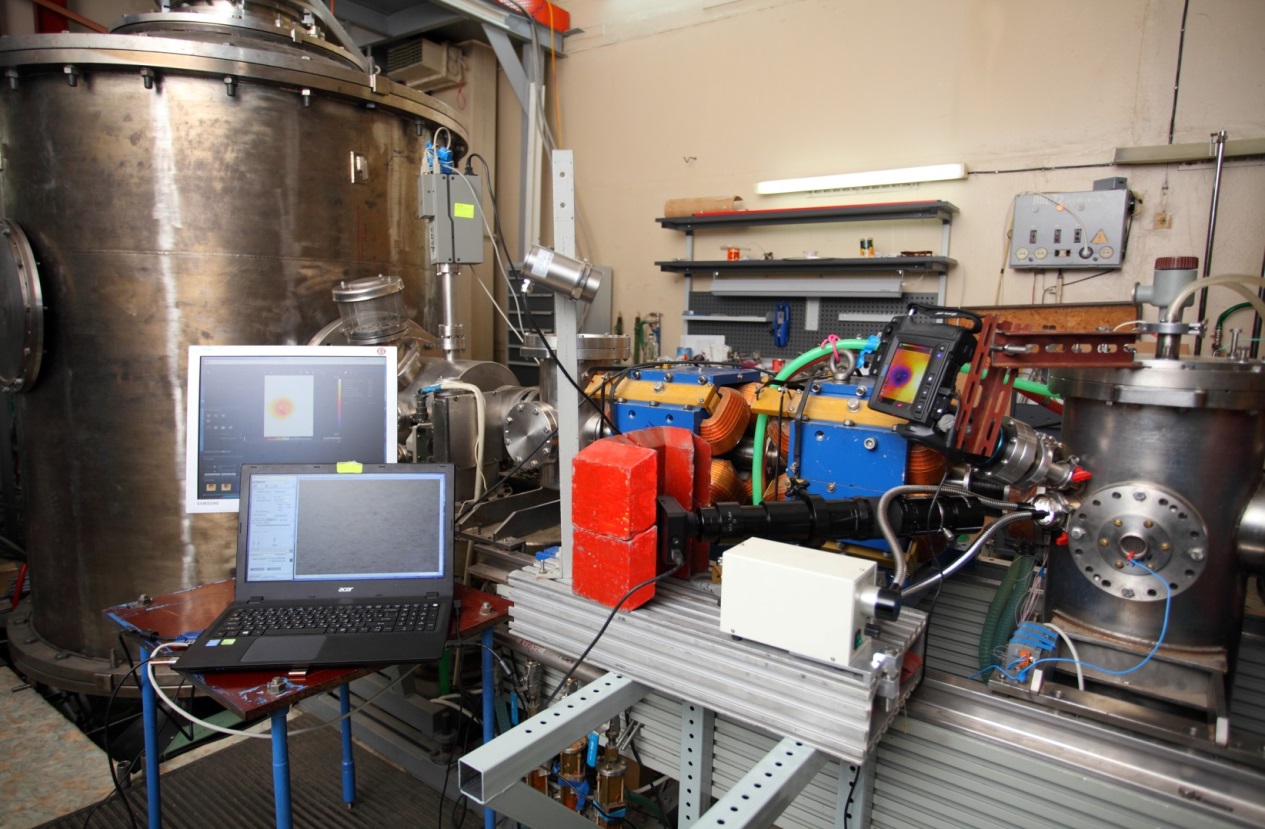
В первых версиях просто загоралась лампочка, потом были взяты голосовые оповещения из google переводчика. В последней версии была использована начитанная озвучка нашей коллеги Лилии (анг. Lilia) из University of science and technology HOUARI BOUMEDIENE, Algeria, которая проходила в нашей лаборатории обучение в аспирантуре. Интересно то, что она говорит по-русски с акцентом. Именно этот фактор должен привлечь внимание оператора.



1. Фотография Лилии
   * 1. Пирометр

Пирометр – устройство дистанционного измерения температуры. Это устройство было использовано для эксперимента, проводимого совместно с Японскими коллегами для изучения эффекта блистеринга в материале при длительном облучении протонным пучком.

При выборе устройства был выбран пирометр 3ML SF CB3 с диапазоном температур от 50 до 400 С° и интерфейсом связи ETHERNET. В системе он виден как COM порт. Общение происходит путем приема / передачи определенных команд в виртуальный COM порт. Фотография устройства представлена на рис. 15.



Пирометр

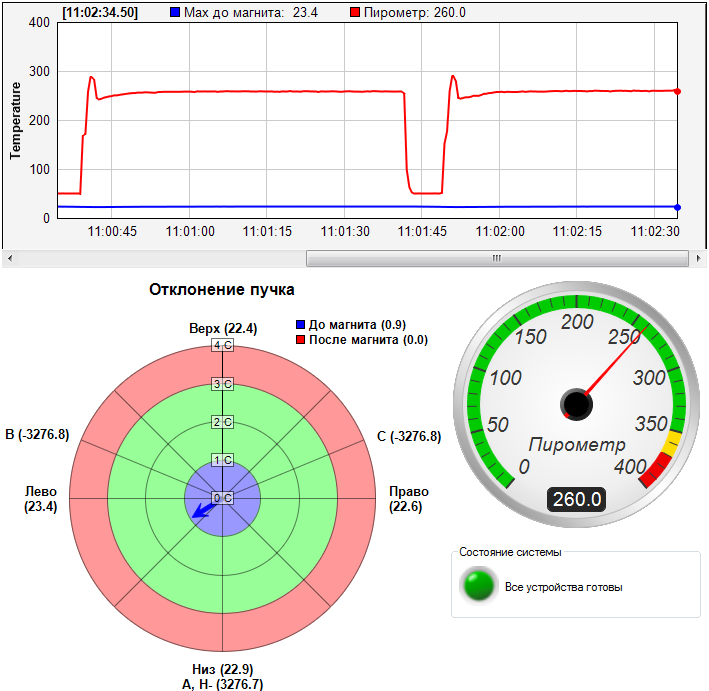
Тепловизор

Охлаждаемая диафрагма

1. Пирометр на ускорителе

В процессе работы с данным устройством возникли трудности снятия реальной температуры из-за изменения коэффициента вторичной эмиссии. В процессе проведения эксперимента менялась отражающая способность материала, и температура, была не такой, как в реальности. Для этого была проведена калибровка по термопаре.

Показания пирометра отображаются на круглом индикаторе и на графике реального времени. Панель пирометра представлена на рис. 16.

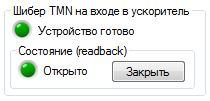
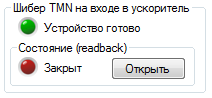


1. Панель пирометра и охлаждаемой диафрагмы
   * 1. Шиберы

Шиберы – своего рода заслонки в вакуумных системах. Они требуются для того, чтобы изолировать вакуумные объемы. Для одного эксперимента потребовалось дистанционно открывать и закрывать шибер из пультовой. Для этого были использованы шиберы с выводами, которые нужно замкнуть для открытия, и разомкнуть для закрытия. В качестве модуля периферии был выбран промышленный модуль ADAM 6066 с интерфейсом связи ETHERNET и протоколом Modbus. Он имеет 6 каналов реле и 6 каналов цифровых входов (ЦВх). Для данного эксперимента использовался один канал реле. Фото шибера представлено на рис. 17. Панель управления шибером изображена на рис. 18.

|  |  |
| --- | --- |
|  |  |

1. Фото шибера

1. Панель управления шибером (два состояния)
   * 1. Пищалка 300В

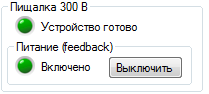
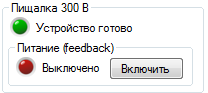
Для лучшего проведения пучка через тракт ускорителя требуется активировать / деактивировать блок питания супрессора вторичной эмиссии. Для решения этой задачи был использован блок питания на 300 вольт. В первое время он стоял на пультовой, и во время эксперимента его включали и выключали. Из-за характерного писка его прозвали пищалкой. Источник питания изображен на рис. 19.

фото

1. Источник питания «Пищалка»

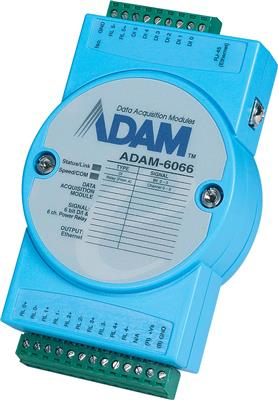
Впоследствии его перенесли в бункер, но возможность включения и отключения должна была сохраниться. Для этого был использован тот же модуль, что и в п. 3.1.3. Питание источника проходит через реле АДАМа. Но для эксперимента требовалась обратная связь по включению. Для этого на выходе источника через делитель был подключен цифровой вход (ЦВх) АДАМа.

Таким образом, оператор получил возможность удаленно контролировать состояние блока питания. Панель управления источником питания изображена на рис. 20.

1. Панель управления источником питания (два состояния)
   * 1. Измеритель мощности на Li мишени

Измеритель мощности на Li мишени – это устройство, которое измеряет мощность Li мишени по входной, выходной температуре воды и потоку воды. После измерения, параметры рассчитываются по формуле, и оператору выводится только рассчитанная мощность. Принципиальная схема измерителя представлена на рис. 21.



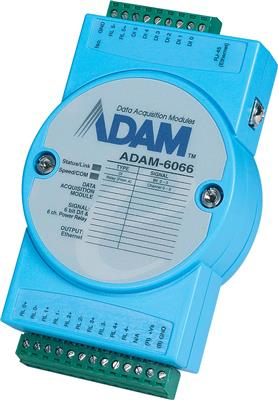
Li

мишень

Вода охлаждения



Измеритель температуры



Измеритель напряжения



**Каналы**

**Модули**

Входная температура

Выходная температура

Значение потока

ADAM 6015

ADAM 6024

Proteus 04004SN2 TPD

1. Принципиальная схема измерителя мощности на Li мишени

Для измерения потока воды был использован потокомер Proteus 04004SN2 TPD с диапазоном измерения 0.8 – 9.5 литров / мин. Значение потока требуется снимать аналого-цифровым преобразователем (АЦП). В качестве АЦП был выбран ADAM 6024.

Для диагностики напряжений, токов и логических входов были выбраны промышленные модули ADAM 6024 с интерфейсом связи ETHERNET и протоколом Modbus. Они имеют 6 АЦП, 2 ЦАП, 2 ЦВх,.2 ЦВых.

Для измерения температур были выбраны промышленные модули ADAM 6015 с интерфейсом связи ETHERNET и протоколом Modbus. Они имеют 7 каналов измерения температуры RTD (resistance temperature detector). Так же для сравнения были использованы модули ICPCON PET Pet7215 с точно такими же характеристиками для сравнения производителей по качеству. На ускорителе используются резисторы типа PT100 (ЧЭПТ-1) с трехпроводной схемой подключения. Схема подключения представлена на рис. 22.

C

B

A

Pt100

Модуль измерения

1. Схема подключения RTD резисторов тремя проводами

Такая схема позволяет устанавливать измеритель и резистор на любое расстояние друг от друга. Устройство измерения производит замер сопротивления по формуле:

Эта формула корректно работает только тогда, когда все три провода одинаковой марки. Такое включение используется для вычитания паразитного сопротивления проводов.

Формула расчета мощности приведена ниже:

Где

– мощность,

– удельная теплоемкость воды,

– плотность воды,

– длина трубы,

– плотность воды,

– значения температуры на входе и выходе,

– скорость потока воды.

* + 1. Ионный источник

В настоящее время проводятся эксперименты с ионным источником на специальном стенде. Для проведения эксперимента требуется запитать поворотный магнит от промышленного источника Gwinstek PSU 6-200. Этот блок общается с консолью через ETHERNET по протоколу SCPI. Требовалось управлять блоком дистанционно и считывать состояние устройства.

* + 1. Ребутатор

Начальная конфигурация системы управления имела централизованную структуру. Это значит, что все сенсоры подключены к одной стойке измерения в центре зала. Из-за этого на длинных проводах образуются большие электромагнитные наводки, которые имеют импульсный характер, а они вызывают сбой и зависание оборудования. Ранее требовалось заходить в зал и перезагружать устройства вручную.

Но в процессе эксперимента радиационный фон может достигать единицы зиверт. Это означает, что за 10 минут человек получит примерно 200 годовых доз. Для примера на рис. 23 изображена толщина двери входа в ускоритель.

фото

1. Толщина двери входа в ускоритель

Для устранения этой проблемы было принято *временное* решение: подключить эти устройства к периферии с реле и дистанционно перезагружать их из пультовой. Проверка включения осуществляется пингованием устройства.

Данное решение очень некрасиво, но оно нужно для сохранения жизни персонала и обеспечения безопасности ускорителя в целом. В скором времени зависающее оборудование будет заменено и проблема будет устранена.

* + 1. Тепловизор Flir T650SC

Для диагностики прохождения пучка через ускоритель используется тепловизор Flir T650SC. Фото тепловизора представлено на Рис. 24. Изображение с тепловизора представлено на Рис. 25.



1. Тепловизор



1. Фото из тепловизора

В настоящий момент интеграция изображения из тепловизора в программу обсуждается.

* + 1. Пути развития системы

В будущем в программу будут добавлены следующие узлы:

* Профилометр пучка на вакуумном вводе движения;
* Безконтактный измеритель тока;
* Вакуумная система;
* Интегрирован двух проволочный профилометр D-Pace;
* Гамма и нейтронный детектор;
* Газовый масс-спектрометр;
* Система управления ионным источником **Ошибка! Источник ссылки не найден.**.
  1. Каналы

Каналы – связующее звено между устройством и модулем. В канале хранится информация об отображаемом имени и имени для логов, а также цвет на графике. Эти данные считываются из конфигурационного файла, пример которого описан в приложении А.

При описании модуля в библиотеке модулей указывается его функционал. Это делается по средствам реализации интерфейсов. В процессе описания модуля указывается лишь требуемый интерфейс.

Интерфейс это фактически регламент взаимодействия. Класс, который реализует интерфейс, обязан реализовывать все его методы. Например, у интерфейса Реле будут объявлены функции чтения, функции записи и массив, в котором хранятся значения Реле.

Таким образом, было создано 6 интерфейсов:

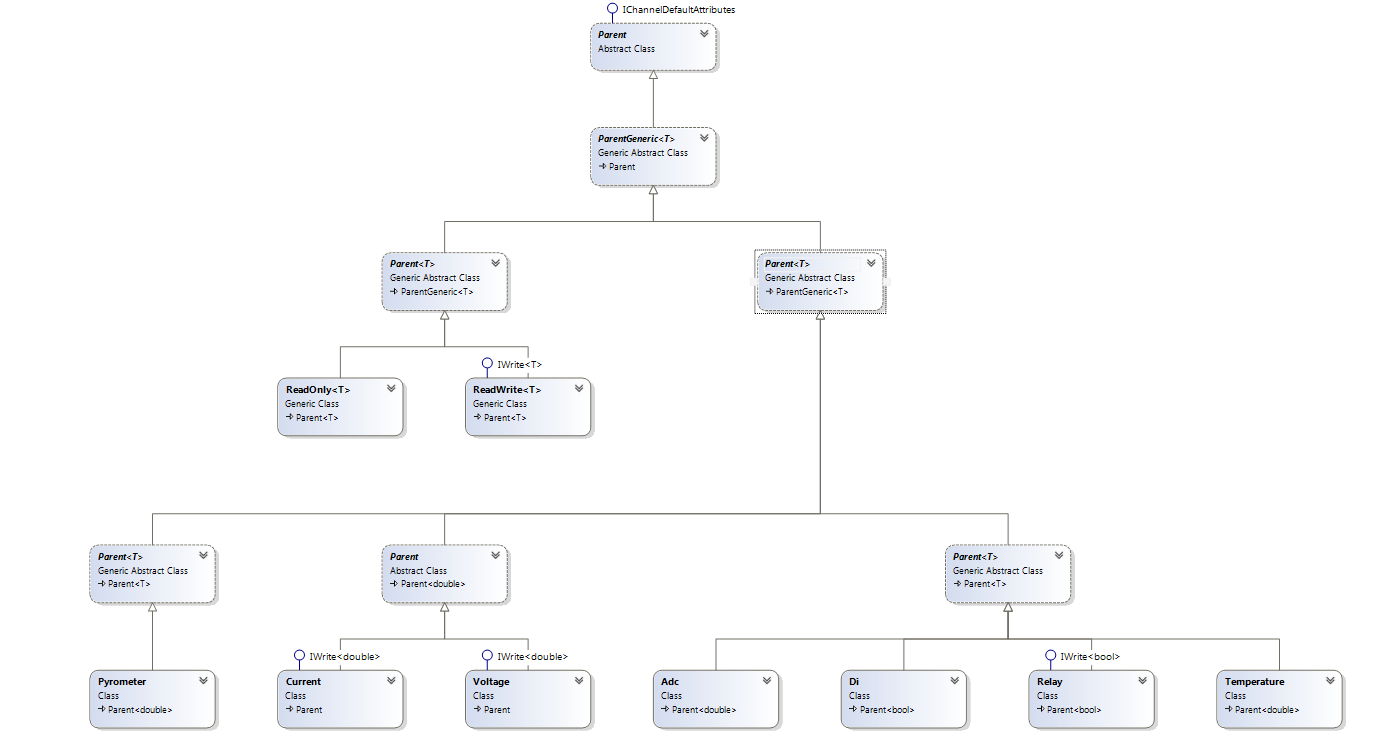
* АЦП;
* ЦАП;
* ЦВх;
* ЦВых;
* Реле;
* Температура.

При этом возможность записи есть только у ЦАП, ЦВых и реле. Можно было бы еще добавить температуру, но на практике установка температуры еще ни разу не была нужна. Возможность считывания данных есть у всех каналов по умолчанию.

Удобство каналов заключается в том, что при описании канала указывается требуемый интерфейс, а при чтении конфигурации происходит проверка модуля на наличие требуемого интерфейса. Это означает, что можно выбирать любой модуль, у которого есть требуемый функционал.

Таким образом, в случае выхода какого-нибудь устройства из строя оператору достаточно переподключить измерение на другой модуль, реализующий нужный интерфейс, подправить конфигурацию и продолжить работу. Обученному оператору потребуется не более 10 минут на переконфигурирование всей системы.

Каналы могут быть двух типов: бинарные и вещественные. Также они логически разделены на реальные и виртуальные. Их различия будут описаны ниже. Блок-схема классов представлена на рис. 26.



Классы реальных устройств

Абстрактные классы

Temperature

Relay

Di

ADC

Modbus

SCPI

COM

Real

Current

Pyrometer

Voltage

ReadWrite

ReadOnly

Virtual

Channel

1. Блок-схема каналов
   * 1. Реальные каналы

Реальные каналы – это каналы, которые связаны с реальным устройством, таким как устройство ввода / вывода или блок питания.

* + 1. Виртуальные каналы

Виртуальные каналы – это каналы, которые не связаны с конкретным модулем. Это канал, значение которого рассчитывается по заданной формуле.

В качестве примера рассмотрим потокометр Proteus 04004SN2 TPD. У него значение потока выводится значением напряжения. 0 вольт соответствует нулевому потоку, 10 вольт – 9,5 л/с. Соответственно значение на канале АЦП будет от 0-10 В, что для оператора может быть не понятно и не нужно. Оператору интересен поток.

Для решения этой задачи можно создать виртуальный канал и указать у него функцию перерасчета из напряжения в поток, и именно ее выводить на экран.

Также виртуальный канал используется в измерителе мощности (п 3.1.5). В нем мощность рассчитывается на основании двух температур и потока воды в системе охлаждения. Сложность функции расчета не ограничена.

* 1. Модули

Модули – устройства ввода / вывода, которые выполняют установку / измерение таких величин, как температура, напряжение, ток. Для модулей была разработана библиотека ModuleWizard (подробнее в п. 4.1). Основная идея заключается в том, что на каждое устройство создается свой поток, в котором опрашивает всю периферию и складывает ее в свой буфер. Далее канальный уровень передает информацию в устройство.

В разработанной библиотеке ModuleWizard были созданы устройства с заданным в них функционалом, посредством интерфейсов (см п. 3.2).

В общем случае, примером модулей могут быть блоки питания или устройства удаленного ввода / вывода.

* 1. Утилиты

Для решения некоторых задач потребовалось создания еще одного уровня. Он не связан с предыдущими, так как не использует модули.

До сих пор единственным применением модулей была задача расчета флюенса. Флюенс – это интеграл тока пучка. Этот параметр очень важен, когда мы проводим совместный эксперимент на тему блистеринга с коллегами из Японии.

Реализация была построена на анализе логов устаревшей программы, которая раз в секунду добавляла значение всех каналов в Excel файл. Для решения этой задачи был написан алгоритм, который по заданной дате искал файлы, кешировал их для быстроты дальнейшей работы, а после считал интеграл по току на мишени.

В первых версиях программы флюенс считался автоматически каждую минуту, но возникла проблема считывания логов с устаревшей программы. Перед изучением файла они сначала копируются во временную папку, и только потом кэшируется и анализируются. Во время копирования файла он блокируется операционной системой, и программа оператора вылетает с ошибкой, что не допустимо.

* 1. Разработанная программа управления

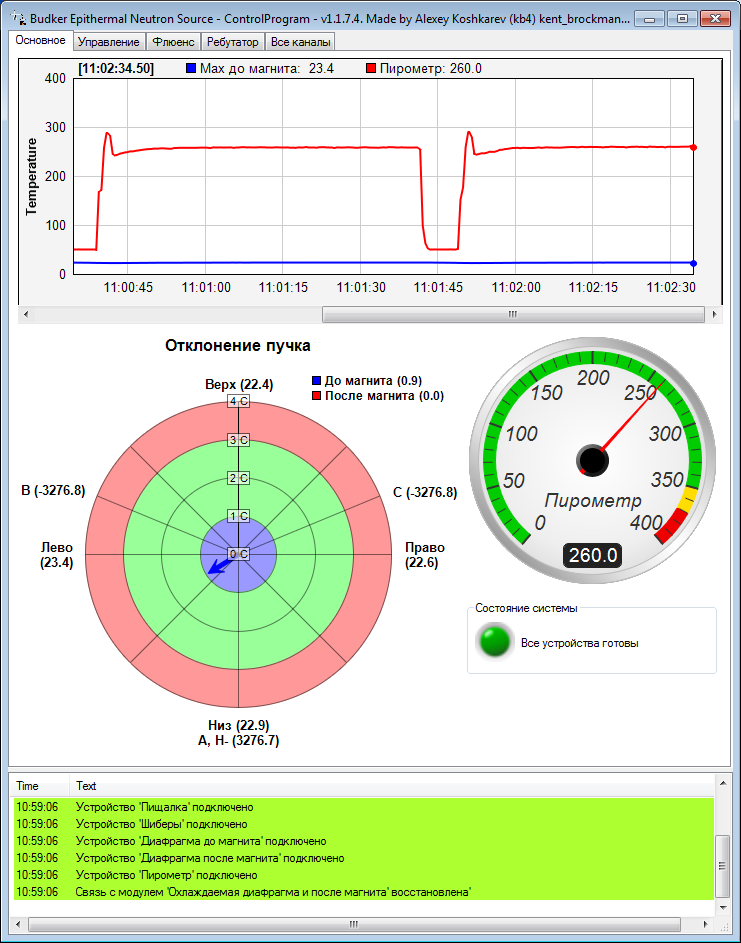
Программа управления выполняет функцию сбора данных с модулей ввода / вывода, управления ими по запросу оператора и сохранение измерений в базу данных. Из-за того, что новая система заменила только часть устаревшей системы, программа оператора имеет ограничения по размерам окна. Поэтому программа была разбита на следующие логические блоки:

* Основное;
* Управление;
* Флюенс;
* Ребутатор;
* Источник H-;
* Все каналы.

Подробнее о каждом разделе будет рассказано в следующих разделах.

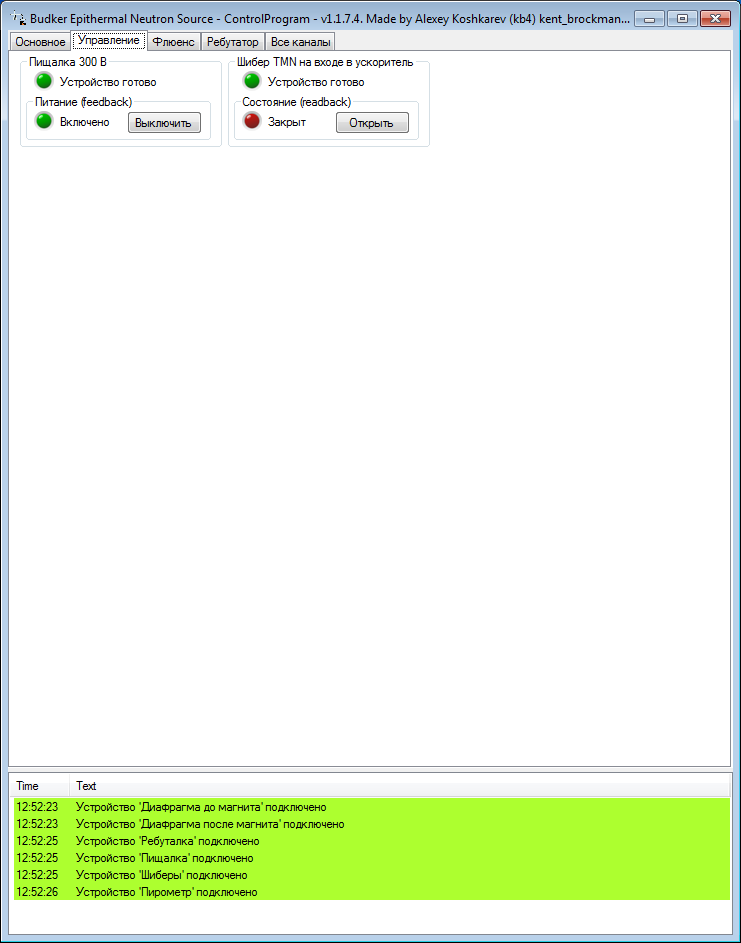
* + 1. Блок «Основное»

В этом разделе показывается общее состояние ускорителя, а именно: состояние подключения всех устройств, температуры на охлаждаемых диафрагмах и пирометра. Этот блок приведен на рис. 27.



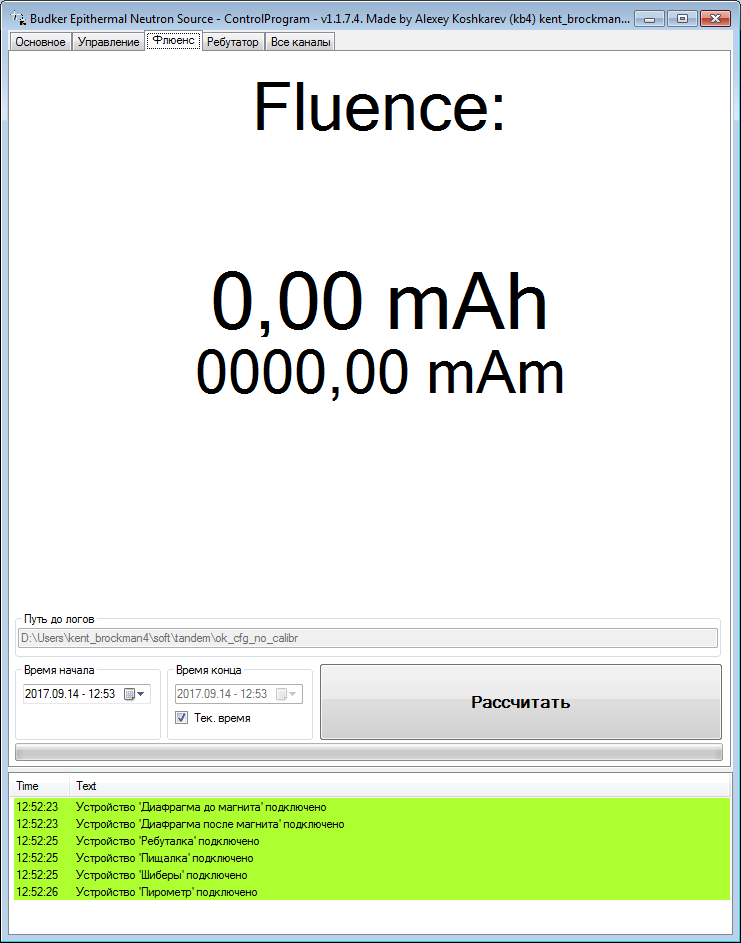
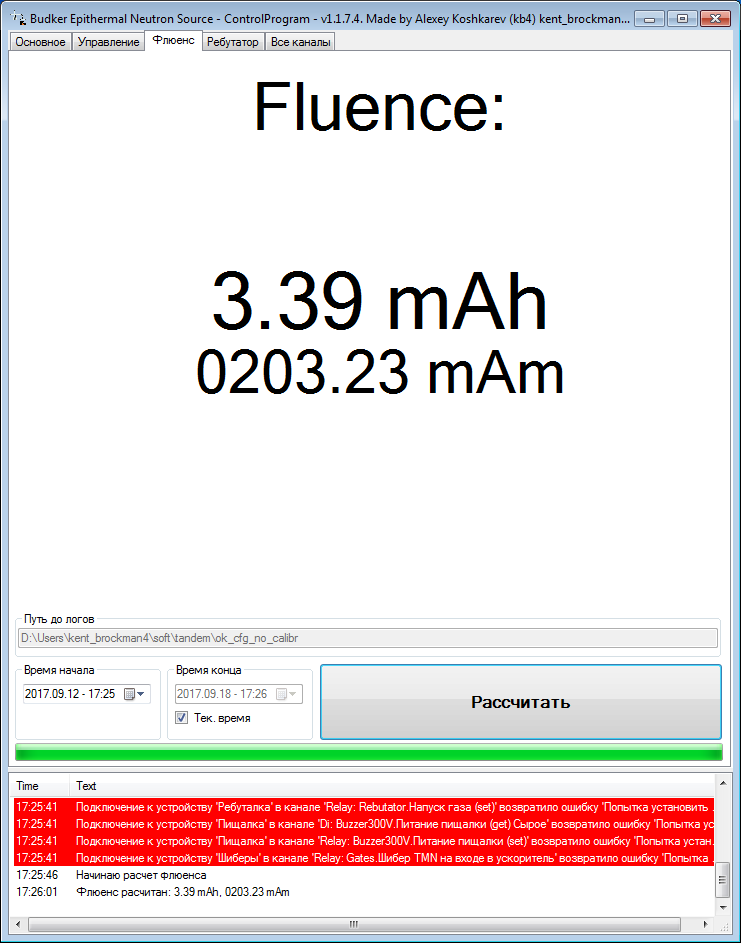
1. Блок «Основное»
   * 1. Блок «Управление»

В этом разделе предоставлен интерфейс управления узлами, такими как: вакуумные шиберы и блок питания супрессора вторичной эмиссии. Подробнее об этих устройствах рассказано в п. 3.1.3 и п. 3.1.4 соответственно. Этот раздел приведен на рис. 28.



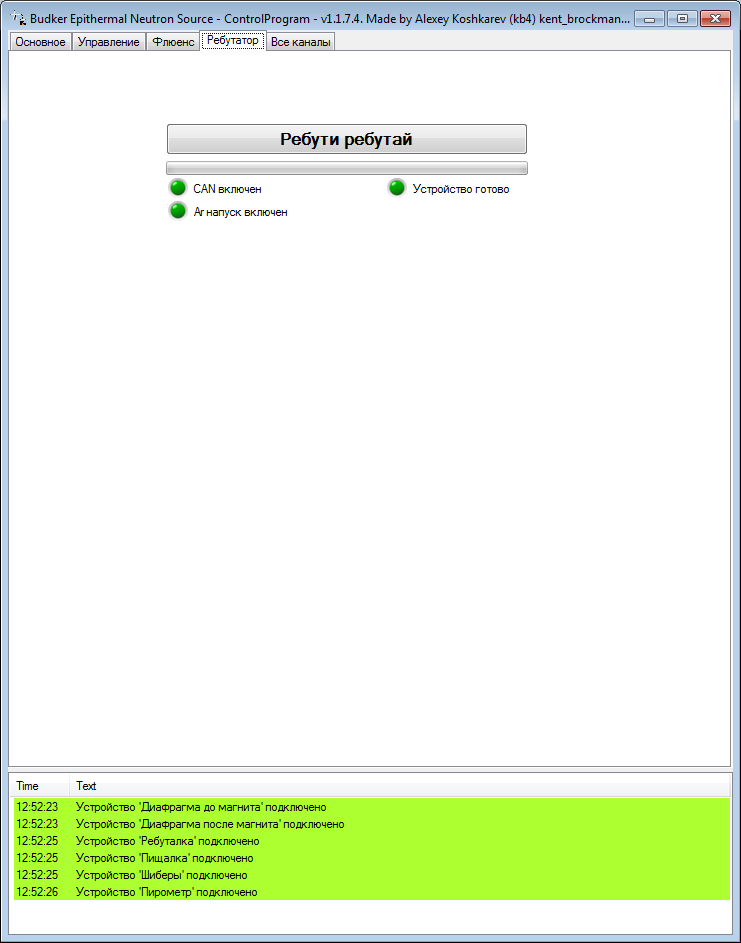
1. Блок «Управление»
   * 1. Блок «Флюенс»

В этом разделе предоставлен интерфейс расчета флюенса за произвольный период. Флюенс – интеграл по времени от плотности потока частиц. Этот параметр очень важен в процессе проведения экспериментов над блистерингом. Подробнее данный функционал описан в п. 3.4. Раздел программы управления приведен на рис. 29.



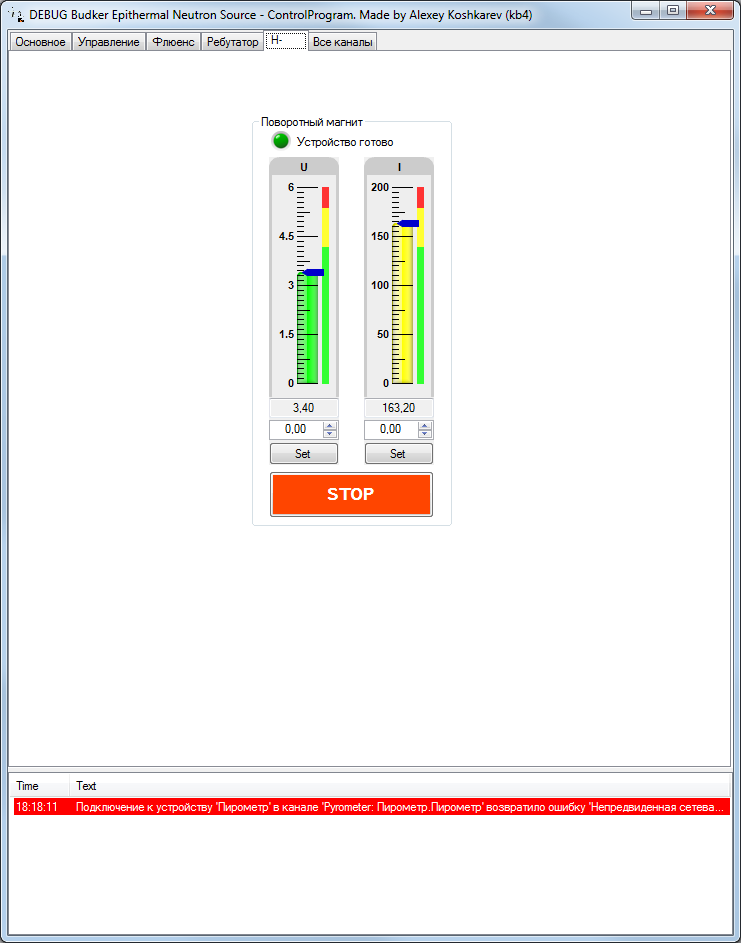
1. Блок «Флюенс»
   * 1. Блок «Ребутатор»

Данный раздел является временной мерой для восстановления работы ускорителя по средством перезагрузки нескольких устройств. Подробнее данный функционал описан в п. 3.1.6. Раздел программы управления приведен на рис. 30.



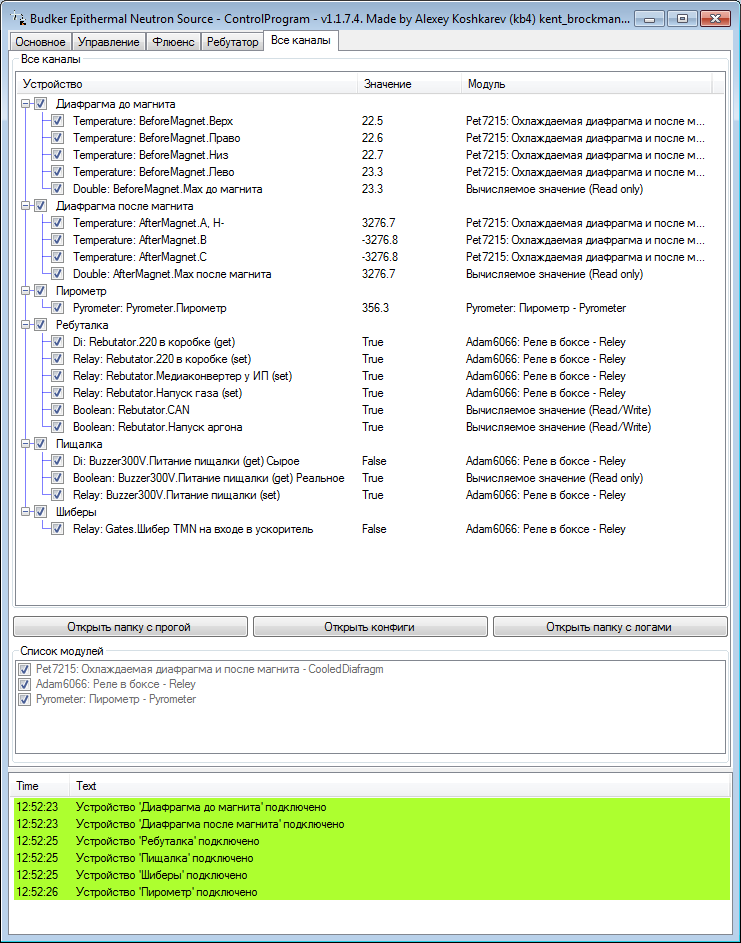
1. Блок «Ребутатор»
   * 1. Блок «Источник H-»

В этом разделе предоставлен интерфейс работы с блоком питания поворотного магнита ионного источника (H-). Изначально планировалось разместить данный функционал в разделе управления, но этот источник временно будет стоять на специальном стенде, на котором весь функционал ускорителя не нужен. Оператор будет управлять только источником питания. Что бы не загромождать интерфейс не нужным в данный момент функционалом, было принято решение разместить интерфейс работы с блоком питания на отдельную вкладку. Подробнее данный функционал описан в п. 3.1.6. Раздел программы управления приведен на рис. 31.

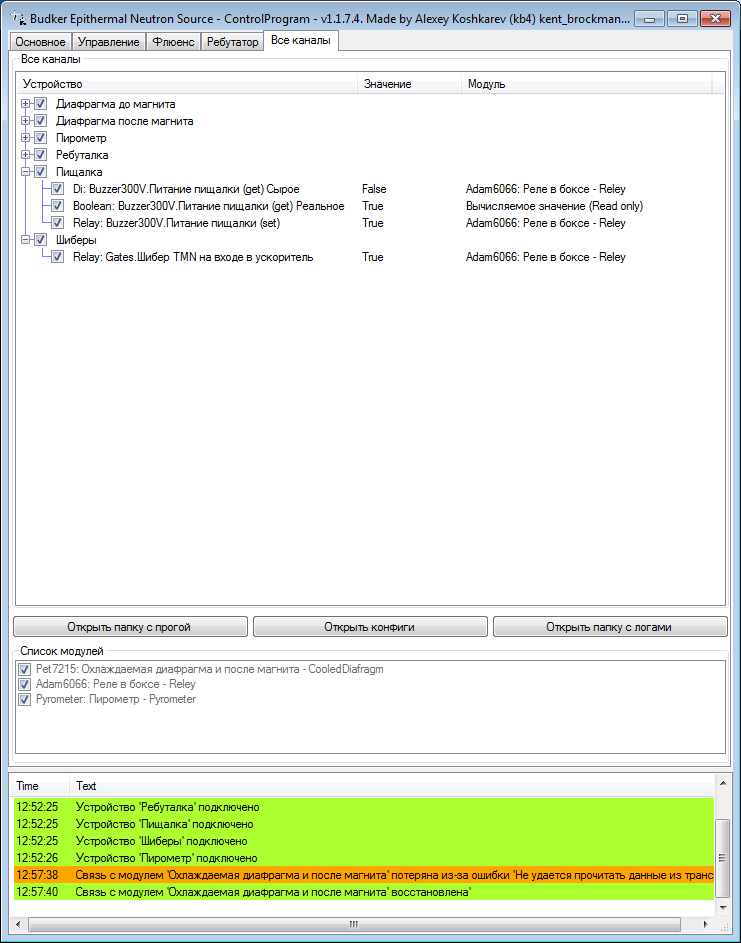
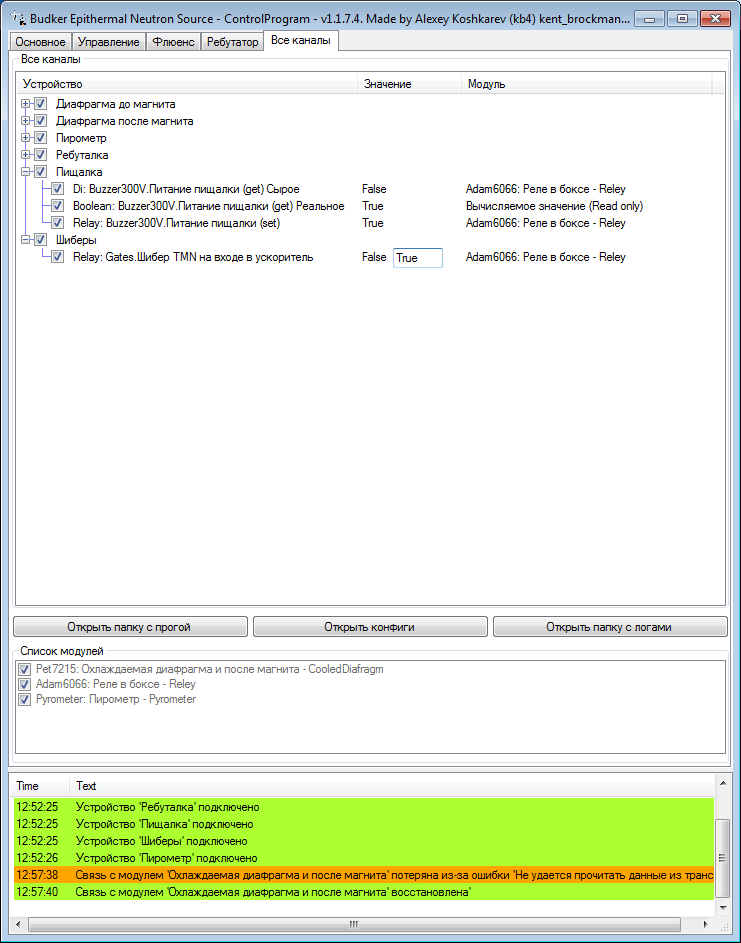


1. Блок источник «H-»
   * 1. Блок «Все каналы»

В данном разделе оператор может изучить «сырые данные» со всех устройств и каналов. Данный функционал очень полезен ученым для самостоятельного оценивания ситуации. Раздел программы управления приведен на рис. 32. Этот интерфейс разработан в том числе для отладки. По двойному клику можно вручную изменить значение канала на устройстве. Пример изменения значения указан на рис. 33. При попытке изменения значения на канале, у которого нет возможности записи, система сообщит об ошибке с описание проблемы.



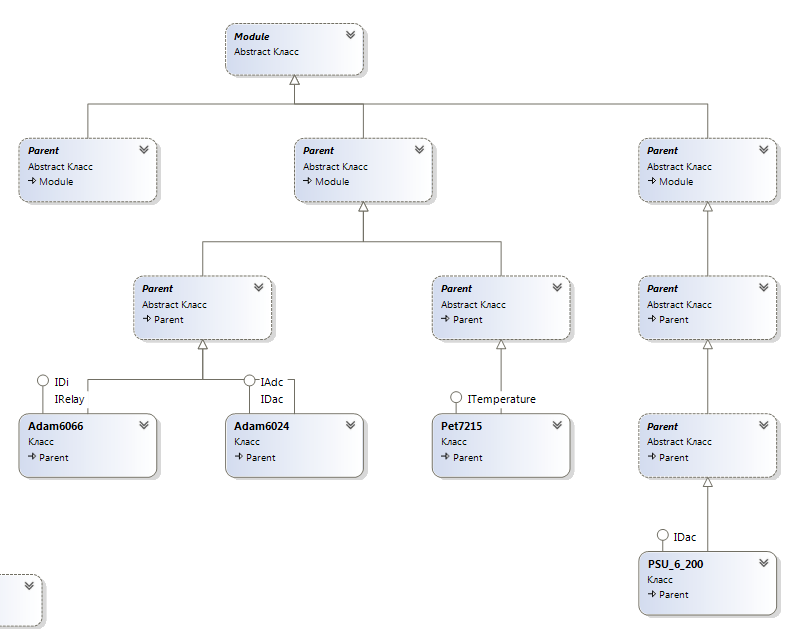
1. Блок «Все каналы»



1. Ручное изменение значение канала на устройстве
2. РАЗРАБОТКА БИБЛИОТЕК
   1. ModuleWizard
      1. Общий принцип

Библиотека ModuleWizard была разработана в процессе создания программы. Задача этой библиотеки – описать все модули управления и предоставить функционал, позволяющий в удобном для программиста виде использовать устройства.

В первую очередь были описаны модули внешней периферии ADAM и ICP CON. После был реализован пирометр, а далее источник питания PSU 6 200. На рис. 34 изображена блок-схема классов и реализации интерфейсов.



ADAM

ICP DAC

COM port

Modbus

SCPI

Power supply

Module

Gwinstek

PSU 6 200

Pet 7215

ADAM 6024

ADAM 6066

Пирометр



Классы реальных устройств

Абстрактные классы

1. Блок схема классов

В классе ‘Module’ описаны основные принципы работы с любыми устройствами.

Сначала происходит подготовка потока: присваивание имени для отладки, создание переменных в потоке устройства. Далее совершаться попытка подключения к модулю. При успешном подключении первым делом производиться считывание выходных регистров устройства, а потом в бесконечном цикле происходит сбор данных и, при необходимости, установка пользовательских значений. Блок схема процесса приведена на рис. 35.



1. Блок схема процесса работы модулей

Удобство библиотеки заключается в том, что методы «Подготовка потока», «Подключение», «Считывание начального состояния модуля» и «Чтение и запись значений на модуль» определяются в дочерних классах, которые могут иметь многоуровневую структуру. На каждом уровне реализуется определенные методы этого уровня. Это сделано для обобщения логики работы с устройствами.

* + 1. Пример

В качестве примера отдельно рассмотрим ветку Modbus (рис. 34). На уровне Module описаны общие принципы работы модулей.

На уровне Modbus описано подключение к устройству по протоколу Modbus и функции проверки статуса подключения (подключен / не подключен). Также на этом уровне описаны функции чтения / записи в устройство.

На уровне Adam / IcpDac описаны функции перерасчета сырых значений АЦП / ЦАП в реальное напряжение и температуру. У каждого производителя функции пересчета разные.

На уровне Adam6066, Adam 6024, IcpDac описан функционал устройства, то есть реализация интерфейсов. Также в каждом устройстве в каждом типе периферии (АЦП, ЦАП, Температура) описываются адреса, по которым хранятся данные о модуле. Например, у Adam6066 данные о реле начинаются с 0x10, а ЦВх с 0x00. Такая же структура у остальных ветвей.

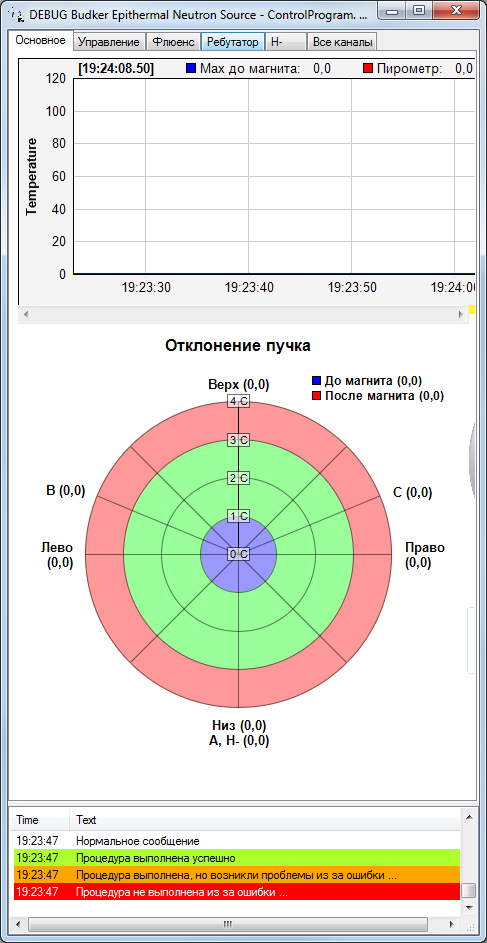
* 1. LogWizard

Библиотека LogWizard была разработана в процессе создания программы управления. В этой библиотеке были собраны воедино механизмы отображения и записи логов в файл и базу данных. Разработанные компоненты описаны ниже.

* + 1. FormLog

Компонент FormLog был разработан для отображения оператору общего состояния системы. Иногда не достаточно обычного индикатора, который может отобразить два или три состояния, а требуется вывести сообщение об ошибке с ее описанием.

Удобство заключается в том, что при выводе сообщения указывается тип сообщения (хорошо, предупреждение, плохо) и время сообщения. Пример использования представлен на рис. 36



1. Пример использования компонента FormLog
   * 1. TimeLog

Компонент TimeLog был разработан для сохранения журнала измерений в Excel файл. В первых версиях библиотеки была использована библиотека для работы с Excel файлами от Microsoft. Но из-за ее неприемлемого исполнения была разработана собственная библиотека с меньшим функционалом, но на порядки производительнее и не требующая частого обращения к жесткому диску.

В программе пользователь лишь вызывает функцию записи в буфер данных. После заполнения буфера пользователь отправляет команду на сохранение строки файла. После многократного повторения этой операции образуется массив строк. Библиотека настроена таким образом, что запись массива строк на диск происходит автоматически раз в минуту. Это позволило значительно разгрузить жесткий диск, и в случае вылета программы сохранятся все данные, кроме последних 60 секунд.

Массив строк пишется в текстовый файл с расширением xls. Формат строки такой, что данные пишутся парами: строковое представление данных, а потом знак табуляции, и так со всеми параметрами. При открытии данных журнала Excel’ем, они успешно открываются и распределяются по колонкам автоматически.

* 1. GraphWizard

Библиотека GraphWizard была разработана в процессе создания программы управления. В этой библиотеке реализовано отображение графической части интерфейса. В данной библиотеке за основу берутся компоненты из графической библиотеки ChartDir.

Проблема использования ChartDir состоит в том, что для реализации компонента требуется написание большого количества кода, а если компонентов несколько, то читаемость программы резко падает, что снижает качество поддержки программного продукта.

В первых версиях библиотеки был описан класс, в конструкторе которого указывалось полотно ChartDir, которое требовалось разместить на форме заранее, а так же параметры отображения. Например, у Led индикатора эти параметры такие: текст при успехе, предупреждении и ошибки. Это неудобно, так как компонент полотна не очевидно связан с сущностью Led.

Для решения этой задачи был разработан компонент WinForms, внутри которого было определено полотно, методы отображения на нем. Например, у компонента BarReadWrite (подробнее в п. 4.3.2) внешними полями были Min, Max, Avarage1,2 и Value. При изменении Value происходила автоматическая отрисовка по заданному параметру. Также в этом компоненте было реализовано событие при нажатии на клавишу Enter. Все компоненты наследуются от Control’а, так что поля x, y, width, height и пр. реализованы автоматически.

В данной библиотеке были разработаны компоненты, которые описаны ниже.

* + 1. Компонент векторный радар

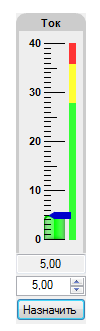
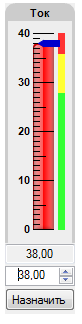
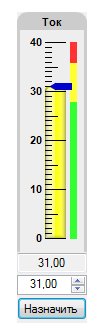
Этот компонент был создан для отображения отклонения пучка. У этого компонента задача отобразить температуру на каждом температурном канале в градусах цельсия и вектор отклонения, который задается в полярной системе координат, то есть углом и длинной. Компонент изображен на рис. 37. Так же на графике изображена легенда и длинна вектора.

В настоящее время измеряется отклонение в двух точках, так что рисуются два вектора.



1. Изображение векторного радара
   * 1. Компонент колонка с вводом

Этот компонент был создан для ввода-вывода данных в пользовательском интерфейсе. Этот компонент изображен на рис. 38.

1. Компонент колонка с вводом в трех состояниях

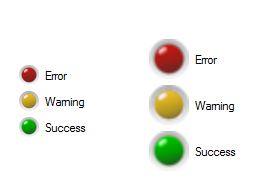
Компонент имеет вид колонки, которая заполняется от минимума до максимума. Он может быть в трех состояниях:

* Зеленый, все хорошо;
* Желтый, предупреждение;
* Красный, опасно.

Этим компонентом управляется и диагностируется источник питания ионного источника.

* + 1. Компонент лампочка

Компонент имеет вид лампочки, что должно быть интуитивно понятно пользователю. Внешний вид индикатора представлен на 4.3.4.



1. Компонент Led- индикатор в трех состояниях

Она может быть в трех состояниях:

* Зеленая, все хорошо;
* Желтая, предупреждение;
* Красная, ошибка.

Для каждого состояния можно настроить отдельный текст. Так же доступно два размера индикатора.

* + 1. Компонент спидометр

Компонент имеет вид стрелочного спидометра. Данный вид отображения позволяет пользователю на интуитивно понятном уровне понимать, является ли значение в пределах нормы. Этот компонент изображен на рис. 40.



1. Компонент спидометр

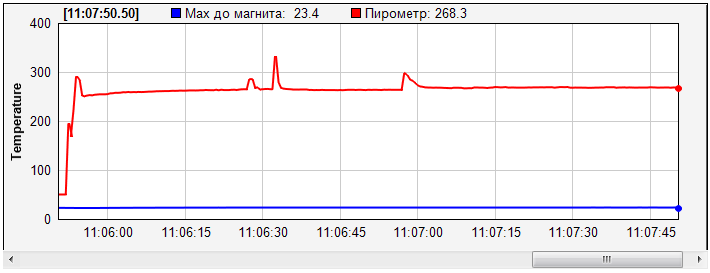
Индикатор имеет три зоны значения:

* Зеленая, все хорошо;
* Желтая, предупреждение;
* Красная, опасно.

Для удобства пользователь на индикаторе изображено название канала и значение в числовом виде.

* + 1. Компонент график реального времени

Данный компонент предназначен для отображения данных в реальном времени. В этом графике отображаются только те параметры, которые интересны пользователю c разрезом по времени. Этот компонент изображен на рис. 41.



1. Компонент график реального времени (масштаб две минуты)

Этот компонент позволяет масштабировать график и менять время начала и конца. Пример тех же данных, только в другом масштабе изображён на рис. 42. Компонент позволяет показать данные на определённый момент времени. При наведении курсора сверху указываются значения.



1. Компонент график реального времени (масштаб час)
2. ОТЛАДКА И ТЕСТИРОВАНИЕ

Для отладки библиотек были разработаны модульные тесты на библиотеки LogWizard и GraphWizard. Для проверки работоспособности программы ее не выключали три дня подряд. Программа работала на компьютере, при этом вся аппаратура была выключена. При включении модулей измерения связь автоматически восстанавливалась.

В старых версиях программы журнал велся в файл Excel в формате xls. Как оказалось, в этом формате было ограничение по количеству строк. Их количество не должно превышать 65535, а при превышении этого числа программа останавливалась с критической библиотекой.

В последних версиях библиотеки ModuleWizard обнаружилось, что одно устройство некорректно считывает данные при переподключении. В настоящее время ведется отладка библиотеки. Данная проблема решается перезагрузкой программы, это значит, что где-то просто не очищается буфер.

ЗАКЛЮЧЕНИЕ

В ходе выполнения дипломной работы была изучена существующая схема автоматизации, написан программный код, обеспечивающий возможность проведения экспериментов на базе нового оборудования. Была завершена разработка системы автоматизации для ускорителя БНЗТ. Продукт был интегрирован в текущую систему автоматизации и начался этап отладки. Проведено тестирование системы в условиях электромагнитных помех. Разработанная система управления может быть без труда использована для решения других задач.

В ходе работы было освоено новое оборудование, изучены протоколы общения устройств, получен опыт работы по изучению сложных комплексов и систем. В дальнейшем планируется расширять список устройств, управляемых новой системой управления, и разработать язык автоматического управления ускорителем, как это было сделано в предыдущей работе [5, 6].

Таким образом, удалось создать систему управления для ускорителя, которая в дальнейшем позволит создавать пучок такой мощности, какой необходим для облучения пациентов, с минимальным участием оператора.

Выражаю искреннюю благодарность моему научному руководителю Таскаеву Сергею Юрьевичу за помощь в подготовке этой работы, Зубареву Петру Васильевичу за помощь в изучении ускорителя и всем сотрудникам лаборатории БНЗТ за поддержку.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. B. Bayanov, V. Belov, E. Bender, M. Bokhovko, G. Dimov, V. Kononov, O. Kononov, N. Kuksanov, V. Palchikov, V. Pivovarov, R. Salimov, G. Silvestrov, A. Skrinsky, S. Taskaev, *Accelerator based neutron source for the neutron-capture and fast neutron therapy at hospital*, 1998 *Nuclear Instr. and Methods in Physics Research* **A 413/2-3** 397-426.
2. S. Taskaev, *Accelerative source of epithermal neutrons*, 2015 *Phys. Element. Particles and Atomic Nucleus* **46** 1770-1830.
3. *Neutron Capture Therapy: Principles and Applications*. Eds.: W. Sauerwein, A. Wittig, R. Moss, Y. Nakagawa. Springer-Verlag, 2012.
4. С.Ю. Таскаев и В.В. Каныгин. *Бор-нейтронозахватная терапия*. Новосибирск: Изд-во СО РАН, 2016 г.
5. Дипломная работа
6. Питер
7. ПРИЛОЖЕНИЕ А

Пример конфигурационного файла

<?xml version="1.0" encoding="utf-8" ?>

<Root>

<Application>

<Base

ShowFullFormLog="true"

ExcelLogFiles="D:\Users\kent\_brockman4\soft\tandem\ok\_cfg\_no\_calibr\Log\BENS Control Program"

/>

</Application>

<Modules>

<Modbus>

<Pet7215>

<CooledDiafragm displayName="Охлаждаемая диафрагма и после магнита" IP="192.168.0.152"/>

</Pet7215>

<Adam6024>

</Adam6024>

<Adam6066>

<Reley displayName="Реле в боксе" IP="192.168.0.103"/>

</Adam6066>

</Modbus>

<COM>

<Pyrometer displayName="Пирометр" port="COM7" baudRate="115200" dataBits="8"/>

</COM>

<SCPI>

<PSU6200>

<HMinusMagnet displayName="H- Питание магнита" IP="192.168.0.15"/>

</PSU6200>

</SCPI>

</Modules>

<Devices>

<ChillingDiaphragms displayName="Охлаждаемые диафрагмы" voiceEnabled="true">

<BeforeMagnet displayName="Диафрагма до магнита" vectorGraphColor="0x0000FF" vectorGraphDisplayName="До магнита" onVectorLengthVeryBigMessageName ="до магнита" soundFileName="cooledDiaphragm01+">

<Up displayName="Верх" module="CooledDiafragm" channel="0"/>

<Right displayName="Право" module="CooledDiafragm" channel="1"/>

<Down displayName="Низ" module="CooledDiafragm" channel="2"/>

<Left displayName="Лево" module="CooledDiafragm" channel="3"/>

<MaxTemperature displayName="Max до магнита" color="0x0000FF"/>

</BeforeMagnet>

<AfterMagnet displayName="Диафрагма после магнита" vectorGraphColor="0x00FF00" vectorGraphDisplayName="После магнита" onVectorLengthVeryBigMessageName ="после магнита" soundFileName="temperatureAfterMagnet">

<A displayName="A, H-" module="CooledDiafragm" channel="4"/>

<B displayName="B" module="CooledDiafragm" channel="5"/>

<C displayName="C" module="CooledDiafragm" channel="6"/>

<MaxTemperature displayName="Max после магнита" color="0x00FF00"/>

</AfterMagnet>

</ChillingDiaphragms>

<StrippingTarget>

<TemperatureIn displayName="Входная температура" module="StrippingTargetTemperature" channel="0"/>

<TemperatureOut displayName="Выходная температура" module="StrippingTargetTemperature" channel="1"/>

<FlowMeter>

<FlowAdc displayName="Поток АЦП" module="StrippingTargetFlowAdc" channel="5"/>

<TemperatureAdc displayName="Темпеатура АЦП" module="StrippingTargetFlowAdc" channel="4"/>

<PressureAdc displayName="Давление АЦП" module="StrippingTargetFlowAdc" channel="3"/>

<FlowVirtual displayName="Поток (LPM) Virtual" color="0x000000"/>

<TemperatureVirtual displayName="Темпеатура Virtual" color="0x000000"/>

<PressureVirtual displayName="Давление Virtual" color="0x000000"/>

</FlowMeter>

<Power displayName="Мощность" color="0x0000FF"/>

</StrippingTarget>

<Rebutator displayName="Ребуталка">

<Power220Get displayName="220 в коробке (get)" module="Reley" channel="0"/>

<Power220Set displayName="220 в коробке (set)" module="Reley" channel="0"/>

<MediaConverterOnRackSet displayName="Медиаконвертер у ИП (set)" module="Reley" channel="1"/>

<ArgonGasControllerSet displayName="Напуск газа (set)" module="Reley" channel="2"/>

<CAN displayName="CAN" IP="192.168.0.250"/>

<ArgonGasController displayName="Напуск аргона" IP="192.168.0.90"/>

</Rebutator>

<Buzzer300V displayName="Пищалка" defaultState="true">

<PowerGet displayName="Питание пищалки (get) Сырое" module="Reley" channel="1"/>

<PowerGetReal displayName="Питание пищалки (get) Реальное"/>

<PowerSet displayName="Питание пищалки (set)" module="Reley" channel="3"/>

</Buzzer300V>

<Gates displayName="Шиберы">

<TMNAcceleratorInSet displayName="Шибер TMN на входе в ускоритель" module="Reley" channel="4"/>

</Gates>

<Pyrometer displayName="Пирометр">

<Pyrometer displayName="Пирометр" color="0xFF0000" module="Pyrometer"/>

</Pyrometer>

<HMinus displayName="H-">

<RotateMagnet displayName="Поворотный магнит">

<Voltage displayName="U" module="HMinusMagnet"/>

<Current displayName="I" module="HMinusMagnet"/>

</RotateMagnet>

</HMinus>

<TestDevice displayName="Тестовое устройство">

<TestValue displayName="Тестовое значение"/>

</TestDevice>

</Devices>

<Utilites>

<FluenceCounter displayName="Расчет флюенса" dataFolder="D:\Users\kent\_brockman4\soft\tandem\ok\_cfg\_no\_calibr"/>

</Utilites>

</Root>