Выпускная квалификационная работа

Прототип приложения для настройки и конфигурации устройств дуговой защиты

по программе профессиональной переподготовки:

«Разработчик прикладного программного обеспечения (Языки C и C++)»

Выполнила: Леонова Татьяна Александровна

Руководитель: Старший преподаватель ВИШ ИДО
Брык Иван Юрьевич

2022

Цель работы:

Разработать прототип приложения для настройки и конфигурации устройств дуговой защиты

Актуальность:

Данное приложение призвано для увеличения скорости настройки и конфигурации устройства на производстве одного из предприятий, занимающихся дуговой защитой.

Предметная область

Дуговая защита – вид быстродействующей защиты от КЗ, основанный на регистрации спектра света открытой электрической дуги. При написании работы использовалось устройство на основе ЗДЗ волоконно-оптического типа. Данный тип ЗДЗ реагирует на световую вспышку от электрической дуги. В качестве датчика, реагирующего на световую вспышку от электрической дуги, используется волоконно-оптический датчик (ВОД).

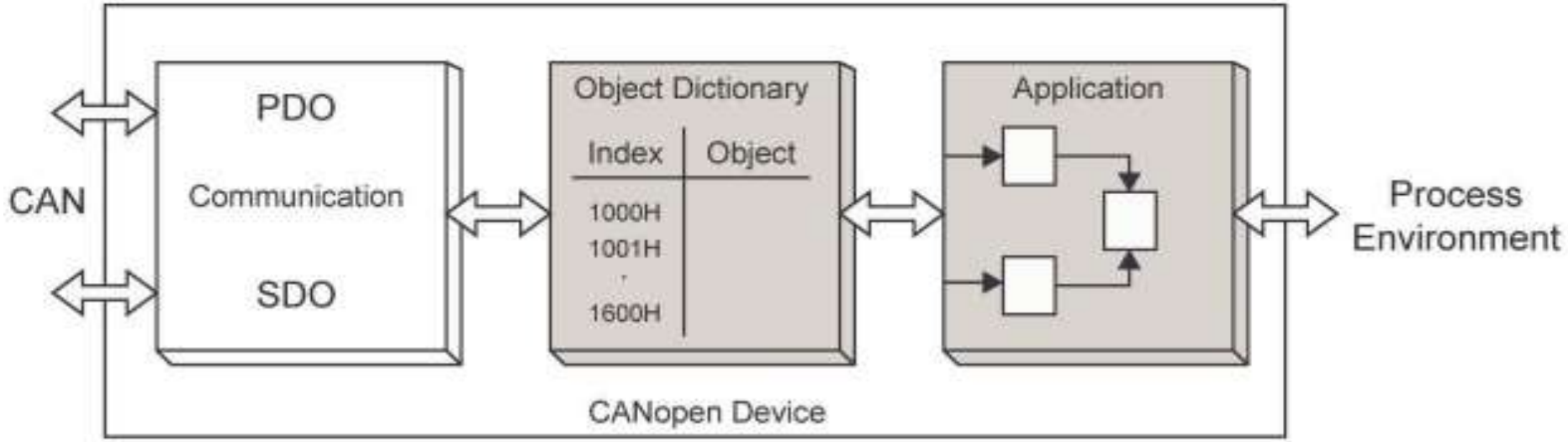
Структура устройства

УДЗ ловит световое воздействие от дуги и воздействует на один из многих аварийных выключателей на подстанции.

ПО позволяет конфигурировать (задавать настройки из Словаря объектов) и настраивать (выбирать какие датчики включены или выключены в выбранном блоке) блоки отдельно от УДЗ и в его составе.

Описание протокола CANopen

CAN-шина – это магистральная шина для высоконадёжной передачи данных по последовательному каналу в широковещательном режиме. CAN использует короткие сообщения, максимум 94 бита.



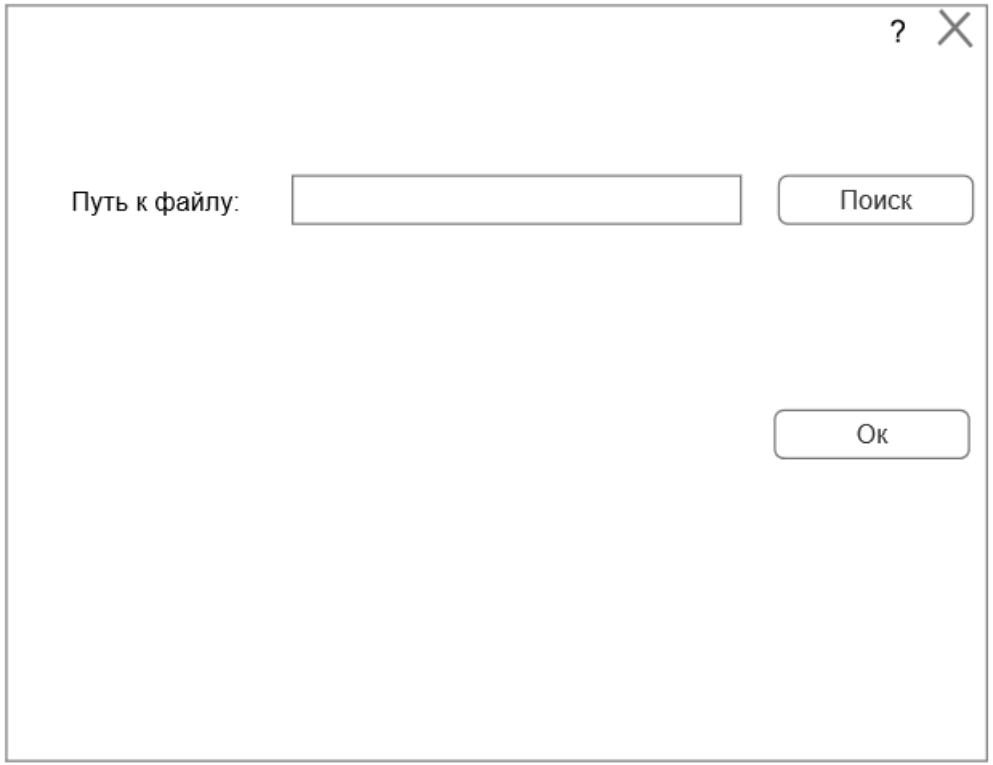
Задачи:

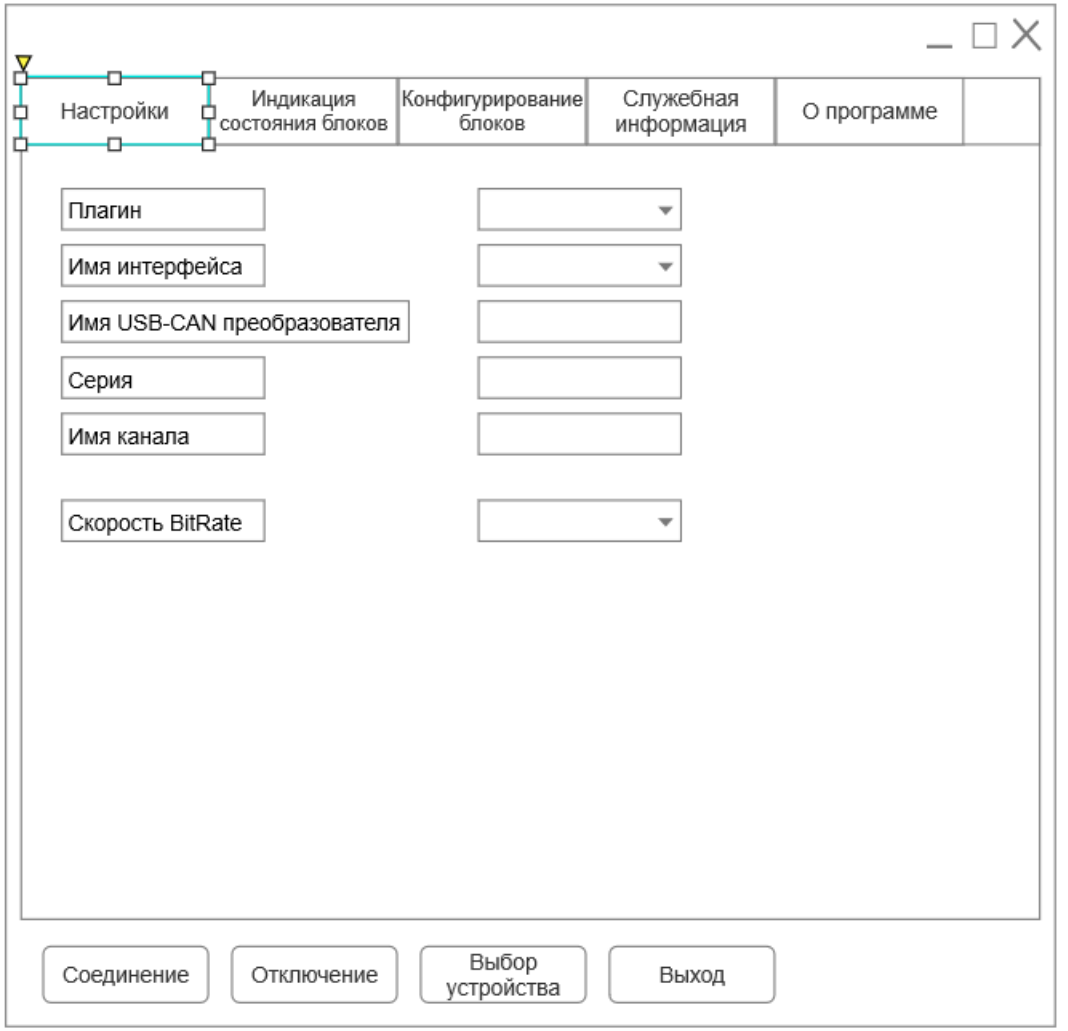
- Разработка функциональных требований к ПО;
- Проработка документации на протокол обмена данными CANopen;
- Применение готовых библиотек Qt для работы с протоколом CAN;
- Разработка классов для работы с пользовательским интерфейсом;
- Разработка классов для работы с протоколом CANopen;
- Применение готовых библиотек для работы с XML файлами;
- Разработка классы для работы с датчиками;
- Проверка работоспособности приложения с помощью ручного тестирования;
- Анализ полученных результатов;

Функциональные требования:

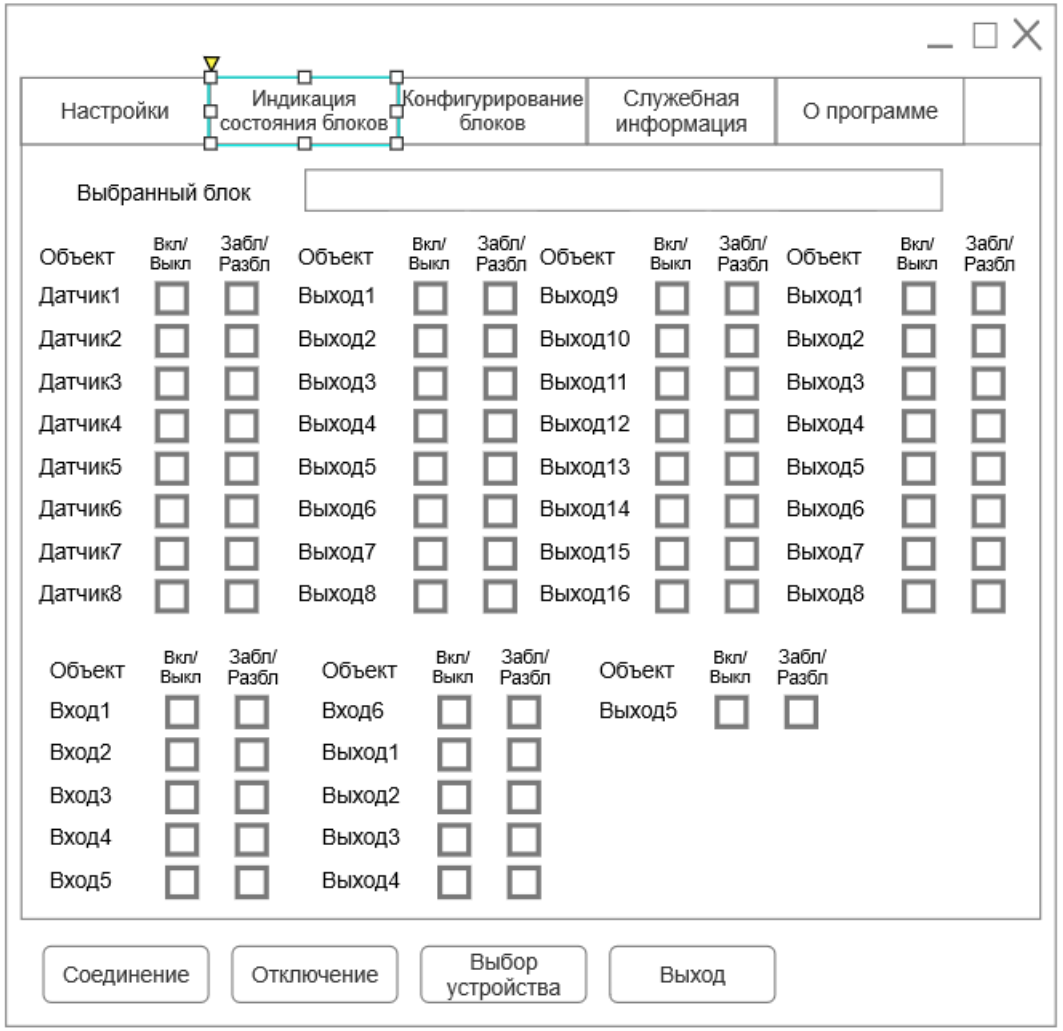
- Программа должна иметь пользовательский интерфейс
  - Окно с запросом пути до файла конфигурации (*.ini);
  - Окно выбора типа блока;
- Программа должна предоставлять:
  -Подключение устройств к ПО по CANopen протоколу;
  -Обмен данными между ПО и устройствами по протоколу CANopen;
  -Индикацию датчиков;
  -Парсинг XML файлов: чтение данных из файла настроек;

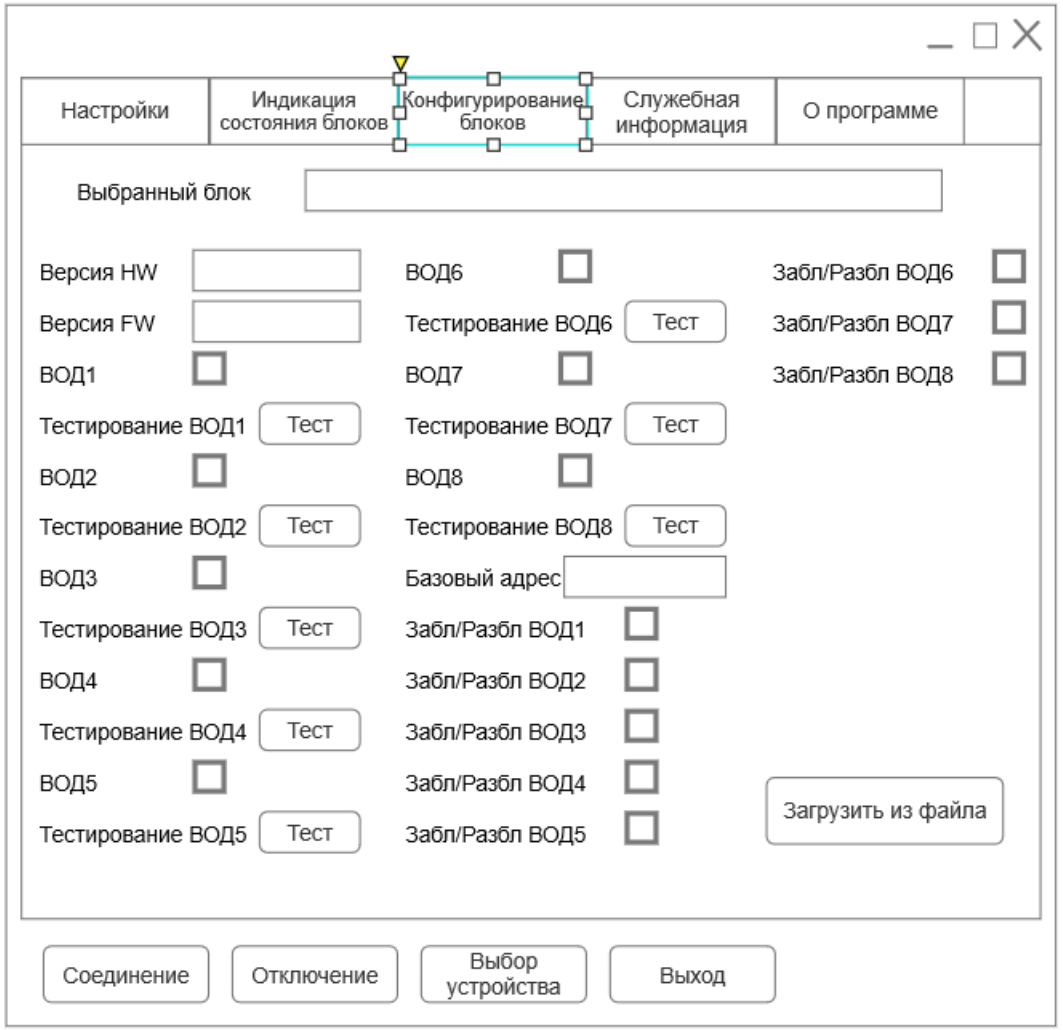
Требования графическому интерфейсу:



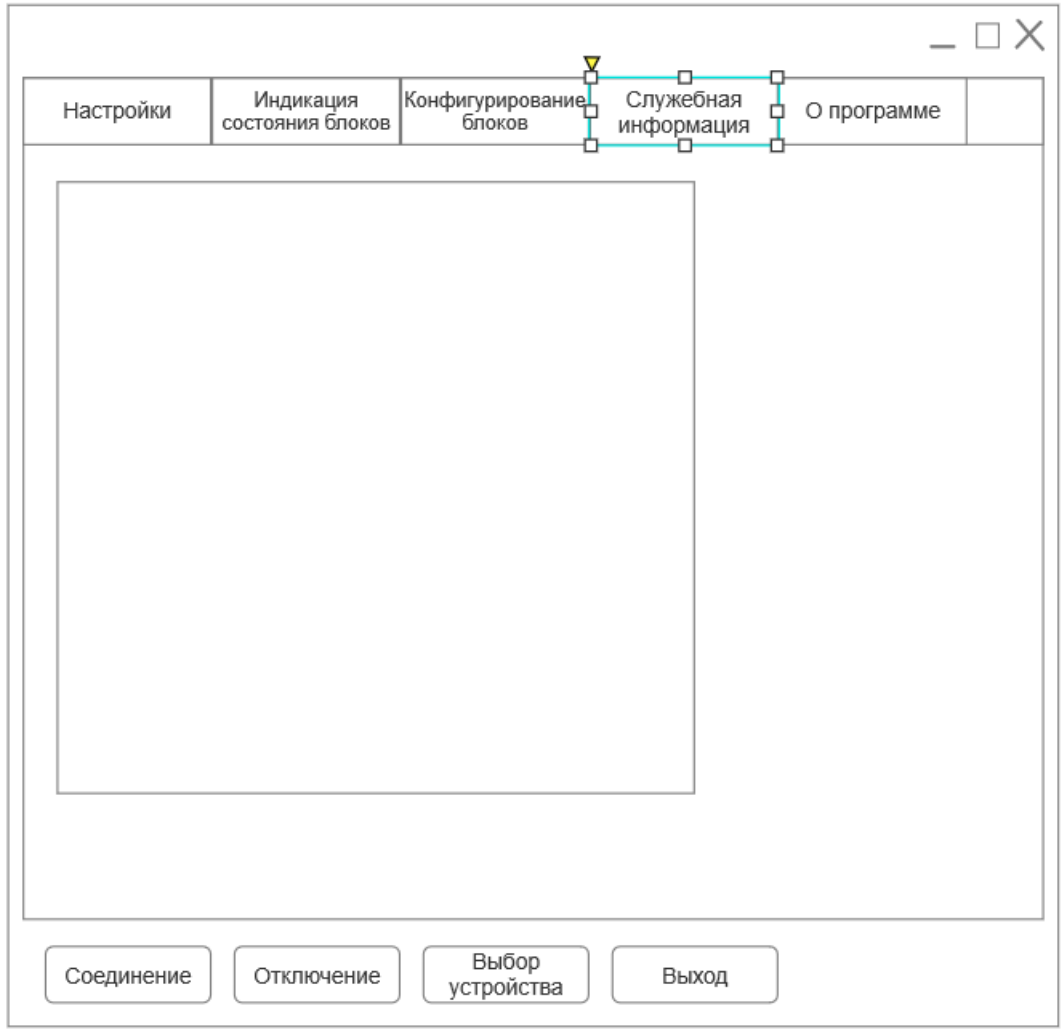


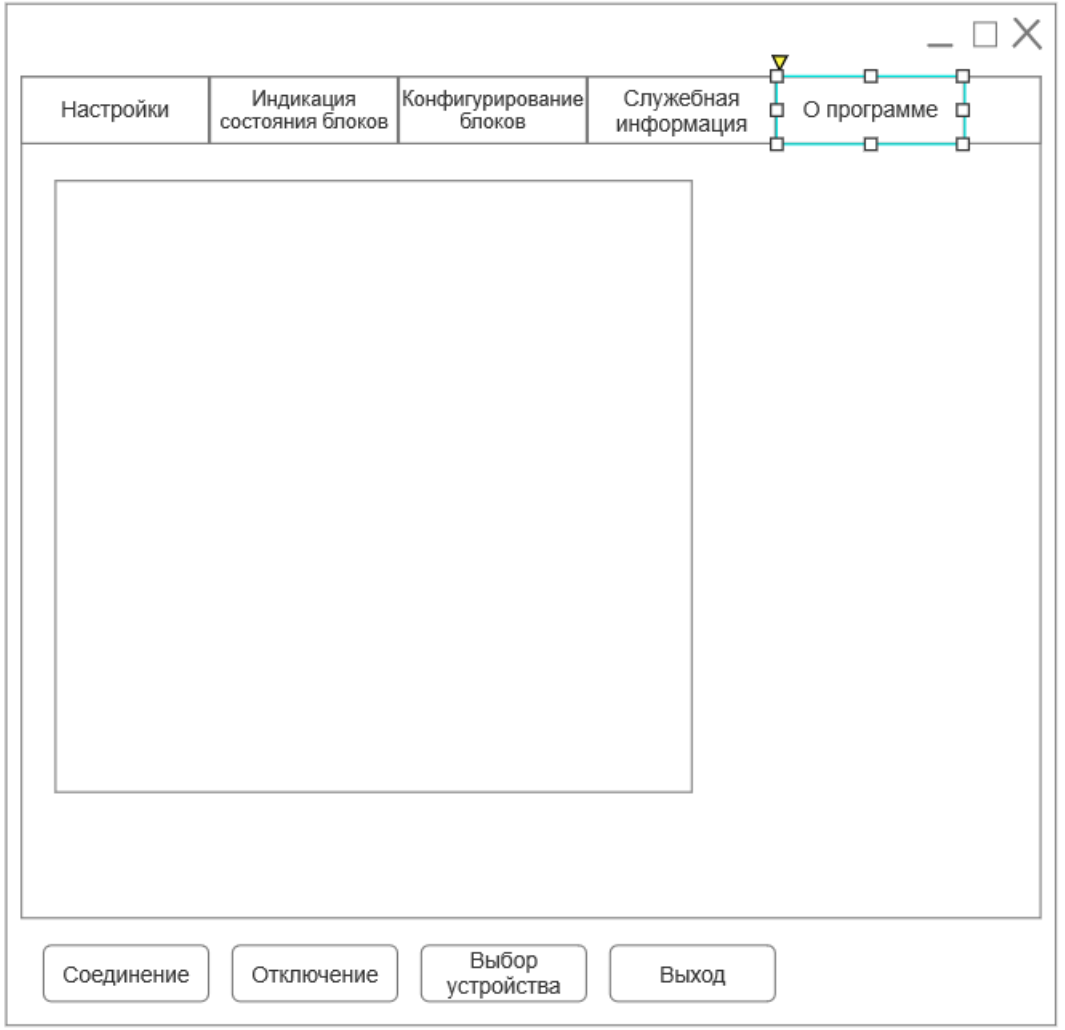
Требования графическому интерфейсу:



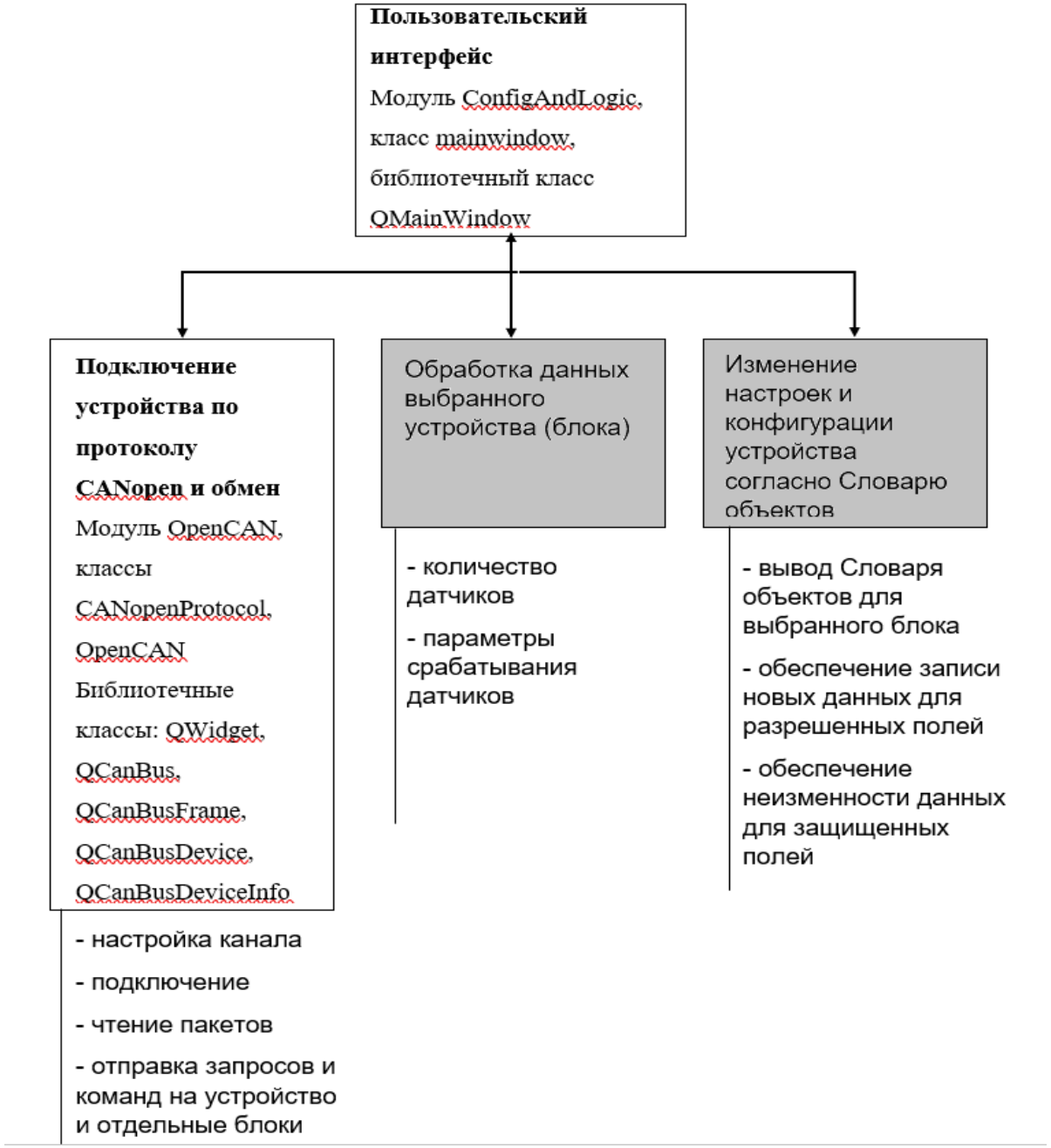


Требования графическому интерфейсу:

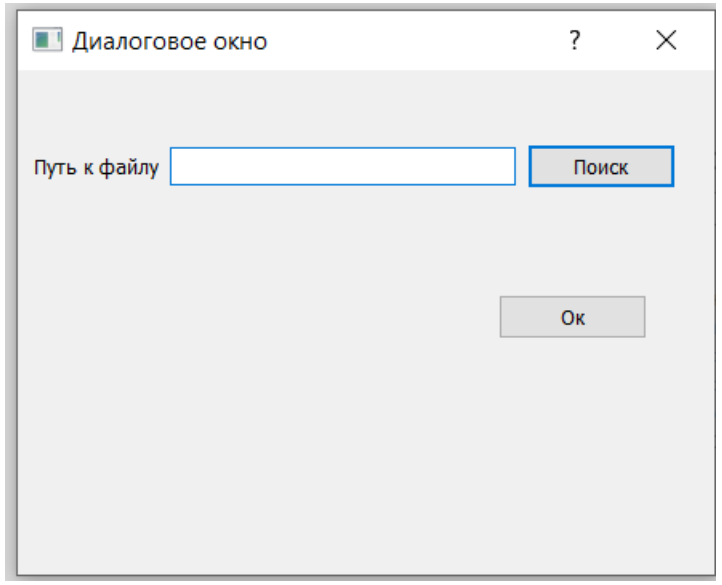




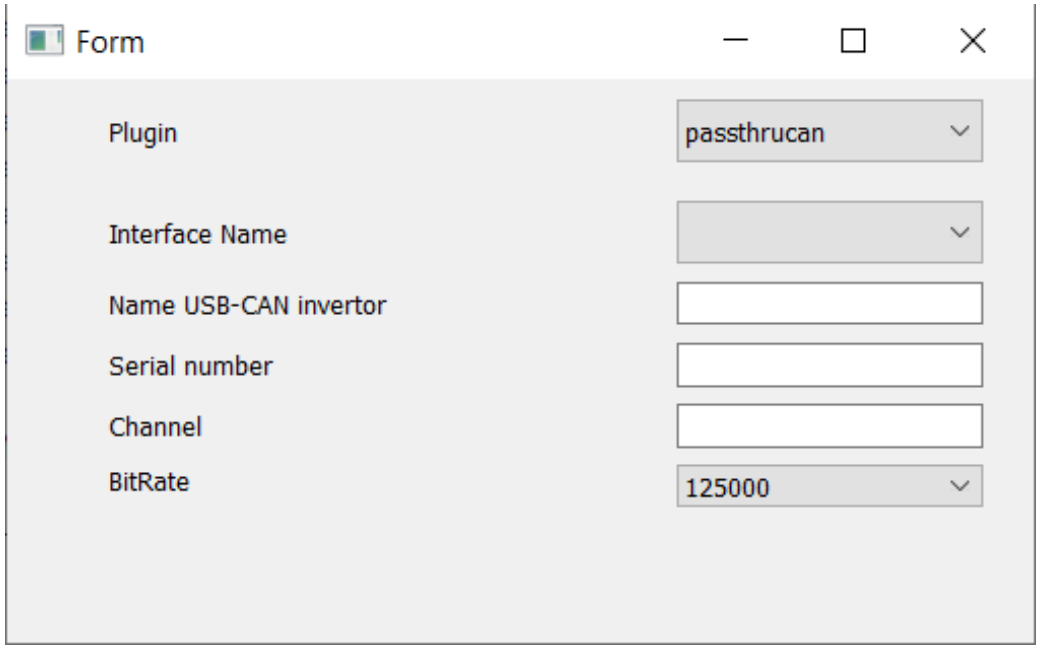
Архитектура программы

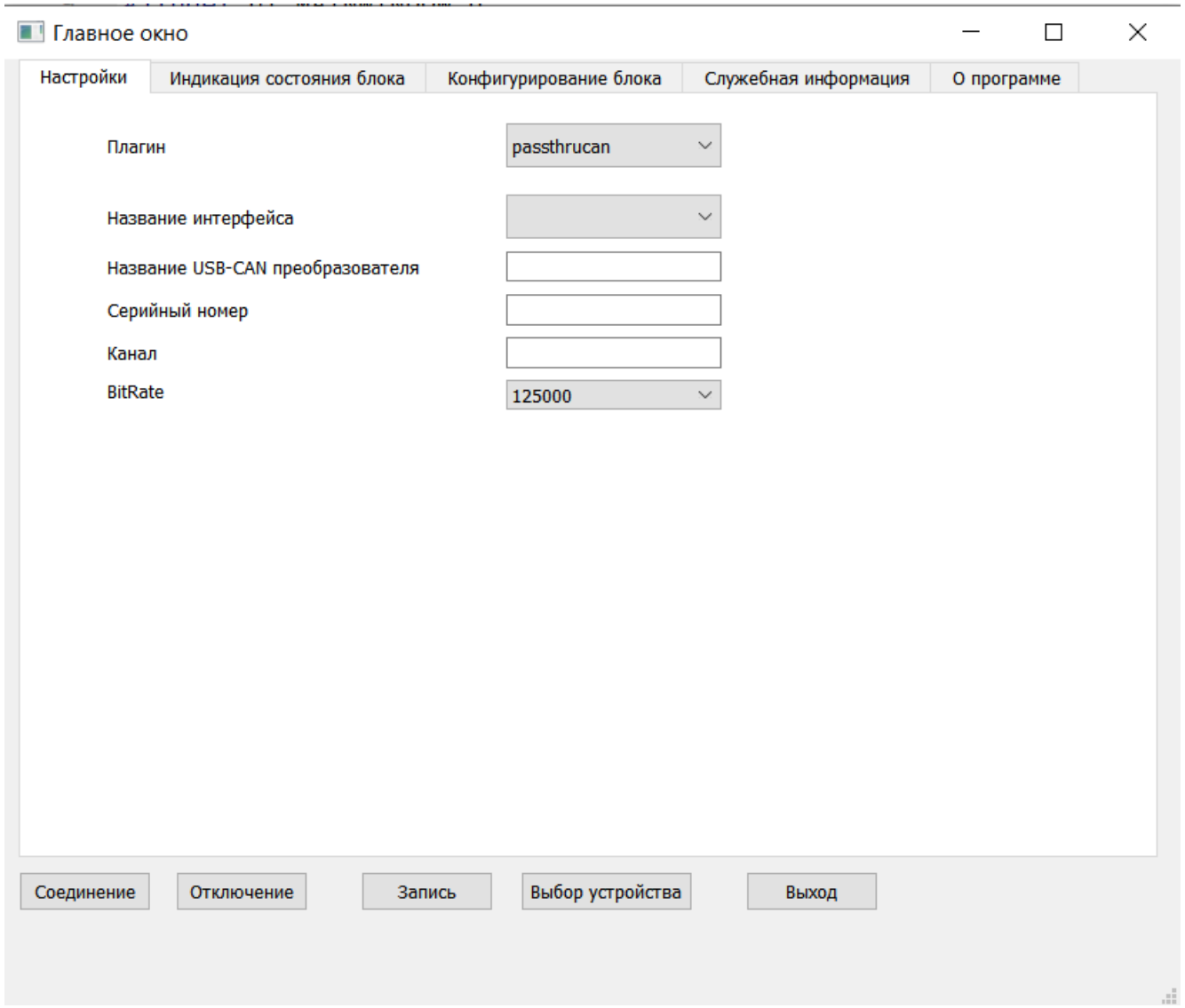


Реализация графического интерфейса



Реализация графического интерфейса





Классы, необходимые для работы с протоколом CANopen

Имя класса	Описание класса	Методы класса
class CANopenProtocol	Обеспечение чтения и записи данных по протоколу CANopen	CANopenProtocol(); ~CANopenProtocol(); QCanBusFrame writeFrame(uint& id, QString& data);

class OpenCAN	Обеспечение чтения и записи данных по протоколу CANopen	explicit OpenCAN(QWidget *parent = nullptr); ~OpenCAN();
		void ChangeSettings();
		Set settings();
		void Connect();
		void Disconnect();
		void writeMessage();
		bool getFlag(); QList<QString> text(); QList<QString> getText();
		QList<QString> getList();
		private slots:
		void pluginChanged(QString name);
		void interfaceChanged(QString name);
		void on_Stop_clicked();
		public slots:
		void read();
		void RequestProductCode();
		void createListS();
		void getResponse();

Проверка работоспособности программы

№	Сценарий проверки	Результат
1	Запуск программы	Проведено 50 проверок, количество ошибок 0
2	Поиск *.ini файла на компьютере	Проведено проверок 50, ошибок 0, файл запускается из любой директории
3	Выбор плагина	Проведено проверок 50, если выбрать неверный плагин, программа завершается. При выборе верного плагина программа работает корректно
4	Проверка соединения с устройством	Проведено проверок 50, устройство подключается корректно, полученные пакеты выводятся во вкладке «Служебная информация»
5	Проверка отключения устройства	Проведено проверок 50, устройство отключается корректно без ошибок
6	Проверка отправки запроса на сброс устройства	Проведено проверок 50, ошибок 0, после отправки запроса, устройство посылает новые пакеты данных с другими Id, что можно проследить во вкладке «Служебная информация»
7	Проверка выхода из приложения	Проведено проверок 50, программа закрывается без ошибок, при этом устройство отключается корректно, если оно не было отключено ранее, кнопкой «Отключение»

Заключение

- Разработаны функциональные требования
- Проработана документация для работы с протоколом CANopen
- Проработаны библиотечные классы для работы с протоколом CAN
- Разработан графический интерфейс
- Разработаны классы для подключения устройства и обмена данными.
- Проведено ручное тестирование во время отладки для подтверждения корректности работы программы.