

**Документация, содержащая описание функциональных характеристик программного обеспечения и информацию, необходимую для установки и эксплуатации программного обеспечения**

**Подсистема автоматизированного контроля «Конструктор состояний контроллера управления оборудованием»**

## Оглавление

1. Введение .....	3
2. Назначение системы .....	3
3. Уровень подготовки пользователей Комплекса .....	4
4. Краткое описание возможностей .....	4
4.1. Серверная часть .....	4
4.2. Интерфейс пользователя .....	4
4.2.1. Интерфейс пользователя предоставляет возможности: .....	4
4.2.2. Компоненты интерфейса пользователя.....	4
5. Виды деятельности и условия применения.....	4
5.1. Виды деятельности и функции. ....	4
5.2. Программные и аппаратные требования к системе .....	5
5.3. Поддерживаемые операционные системы:.....	5
5.4. Рекомендованные настройки безопасности веб-браузеров: .....	5
5.5. Требования к одноплатному компьютеру для установки программного комплекса:.....	5
5.6. Установка программного комплекса.....	6
6. Работа пользователя в панели управления .....	6
6.1. Вход в систему. Главная страница. ....	6
6.2. Вкладка [Порты]. Описание физического порта.....	7
6.3. Вкладка Переменные. Создание переменных. ....	8
6.4. Вкладка Команды. Создание команд. ....	8
6.5. Вкладка Фразы. Создание фраз. ....	9
6.6. Вкладка Состояния. Создание описания. ....	9
6.7. Вкладка Монитор .....	10

## 1. Введение

Подсистема автоматизированного контроля «Конструктор состояний контроллера управления оборудованием» (далее Комплекс) - программный комплекс, предназначенный для визуального конструирования схемы состояний; входных условий перехода между состояниями; задания способов реакций состояний; а также обработки состояний портов ввода, управления состояниями портов вывода в соответствии со схемой перехода; запуска внешних команд и сохранения схемы состояний.

Комплекс можно использовать для прикладных задач автоматизации и управления процессами в реальной сфере экономики, а также как тренажер для обучения робототехнике и схемотехнике.

Комплекс представляет собой конструктор, который настраивается под различные практические задачи и потребности и может управлять состояниями аппаратного обеспечения (приводы, двигатели, воротные группы, технологии умного дом и т.д.), взаимодействующего с пользователем с помощью датчиков и исполнительных устройств.

Комплекс может использоваться как в практическом управлении данным оборудованием, так и в его тестировании и настройке. В Комплексе реализована возможность задавать и конструировать условия работы оборудования в различных состояниях соответствии с потребностями пользователя без подключения к действующему объекту.

Комплекс может интегрироваться с дополнительным ПО и внешними системами посредством интерфейса Комплекса, без использования сложных интеграционных процессов.

Комплекс состоит из следующих элементов:

- Серверная часть, предоставляет собой интерфейс взаимодействия между клиентом и модулем управления портами на базе http протокола;
- Интерфейс пользователя, представляет собой html страницу с меню и активной зоной текущего компонента.

В разработке Комплекса использовались

Технологии и средства разработки:

- IntelliJ Idea (в редакции «Community Edition») - интегрированная среда разработки программного обеспечения для языков программирования, в частности, JavaScript, Python. Имеет разрешительную лицензию на свободное программное обеспечение Apache Software Foundation, которая даёт пользователю право использовать программное обеспечение для любых целей, свободно изменять и распространять изменённые копии, использовать лицензируемый код в собственном закрытом программном обеспечении.
- IntelliJ PhpStorm – интеллектуальный редактор для PHP, HTML, JavaScript, интегрированная среда разработки для PHP. Лицензия на проприетарное ПО (международная компания JetBrains).

Языки программирования:

Серверная часть: python 3, bash.

Интерфейс пользователя: JavaScript, html.

Операционная система: серверная часть linux arm 32/64 либо linux x86-64

Веб-сервер: встроена в серверную часть библиотека flask.

Правообладателем подсистемы автоматизированного контроля «Конструктор состояний контроллера управления оборудованием» является Акционерное общество «Благонадежный Поставщик», что подтверждается Свидетельством о государственной регистрации программы для ЭВМ № 2022680571 от 02.11.2022 г.

## 2. Назначение системы

Программное обеспечение представляет собой комплекс, состоящий из серверной части, запускаемой на одноплатном компьютере, и интерфейса пользователя, работающего в браузере.

**Назначение Комплекса:**

визуальное конструирование схемы состояний, входных условий перехода между состояниями,

задание способов реакций состояний;

обработка состояний портов ввода, управление состояниями портов вывода в соответствии со схемой перехода, запуск внешних команд;  
сохранение схемы состояний.

### **3. Уровень подготовки пользователей Комплекса**

Пользователь – администратор, имеющий доступ к панели управления Комплексом для создания, моделирования и управления состояниями требуемого оборудования.

Администратору необходимо знать соответствующую предметную область (оборудование, для управления которым используется Комплекс), иметь навыки и опыт работы с браузерами, операционными системами, иметь представление о программировании, устройстве одноплатного компьютера, назначении портов ввода/вывода, иметь начальные знания схемотехники и электроники.

### **4. Краткое описание возможностей**

Подсистема автоматизированного контроля «Конструктор состояний контроллера управления оборудованием» (далее Комплекс) решает спектр задач по конструированию, обработке, управлению и сохранению схемы состояний оборудования.

В Комплексе реализованы следующие возможности:

- задавать и конструировать условия работы действующего оборудования в различных состояниях, в соответствии с потребностями пользователя;
- наблюдать за процессом переключения состояний, состоянием входных и выходных портов действующего объекта;
- проводить тестирование, моделирование, настройку и отладку состояний и условий работы оборудования, без подключения к действующему объекту.

Необходимые исходные условия применения Конструктора для пользователя:

- описан процесс, который требуется автоматизировать;
- наличие одноплатного компьютера;
- наличие датчиков (входные устройства) и исполнительных устройств (выходные устройства: двигатели, другие контроллеры, лампы сигнализации и т.д.).

Далее можно приступить к конструированию и настройке состояний на Комплексе.

#### **4.1. Серверная часть**

##### **4.1.1. Серверная часть выполняет следующие функции:**

- работает с портами ввода/вывода микрокомпьютера и обрабатывает переключение состояний;
- реагирует на изменение состояния в соответствии с заданными условиями в интерфейсе пользователя, в зависимости от изменения состояний портов ввода;
- хранит настройки схемы состояний.

##### **4.1.2. Компоненты серверной части:**

- Автомат обработки состояний: программный модуль, считывающий уровни сигнала портов ввода, хранящий значения переменных, параметры текущего состояния, отслеживающий изменения и переключение текущего состояния, запуск внешних команд.
- Внешние команды выполнения: список внешних команд и программ операционной системы с параметрами для запуска.
- Эмулятор состояний: режим работы без физического управления портами ввода/вывода.

#### **4.2. Интерфейс пользователя**

##### **4.2.1. Интерфейс пользователя предоставляет возможности:**

- визуальная настройка входных условий и переходов между состояниями, в зависимости от условий (уровня сигнала портов ввода, значения переменных);
- настройка выходных параметров (уровни сигнала портов вывода, значения переменных, запуск внешних процессов, обработка таймаута);

- мониторинг состояний – работа с эмулятором состояний, без физического управления портами ввода/вывода.

#### **4.2.2. Компоненты интерфейса пользователя**

Интерфейс пользователя представляет собой конструктор состояний для переключения текущего компонента и включает в себя:

- Порты;
- Переменные;
- Команды;
- Фразы;
- Состояния;
- Мониторинг состояний.

### **5. Виды деятельности и условия применения**

#### **5.1. Виды деятельности и функции.**

Функционал Комплекса для администратора включает в себя:

- создание и редактирование конфигурации системы;
- создание описания для физических портов;
- создание и описание списка переменных;
- создание и описание списка команд;
- создание и описание списка фраз;
- создание и редактирование списка состояний;
- просмотр состояния аппаратного обеспечения действующего объекта, возможность наблюдать за процессом переключения состояний, состоянием входных и выходных портов;
- мониторинг, тестирование и настройка состояний и условий работы оборудования без подключения к действующему объекту.

#### **5.2. Программные и аппаратные требования к системе**

Для корректной работы с комплексом необходима следующая конфигурация автоматизированного рабочего места администратора. Минимальные требования к системе:

- 1 ядро;
- 1 Гб доступной памяти на 1 ядро системы.
- Процессор:
  - архитектура: x86\_64/32, arm64/32;
  - количество процессоров: 1;
  - количество ядер: от 1;
  - тактовая частота: от 1.0 ГГц.;
- 512 мб свободной оперативной памяти;
- 1 Гб свободного дискового пространства;
- сетевая карта ethernet - 10 мбит/с.;
- ОС Microsoft Windows XP и выше (Vista, 7, 8/8.1, X) либо другая поддерживающая запуск браузера.

#### **5.3. Поддерживаемые операционные системы:**

- Red Hat Enterprise Linux (RHEL) (32-bit or 64-bit);
- Novel SUSE Linux Enterprise Server (SLES) (32-bit or 64-bit);
- Microsoft Windows (32-bit or 64-bit);
- Apple Mac OS;
- Apple iOS;
- Solaris SPARC;
- Fedora;
- Debian Linux;
- HP-UX;
- FreeBSD;
- CentOS;

– Ubuntu.

Поддерживаемые веб-браузеры:

Google Chrome 9+, Firefox 4+, Opera 15+, Safari 5.1+, Internet Explorer 11, Microsoft Edge.

Персональный компьютер.

#### 5.4. Рекомендованные настройки безопасности веб-браузеров:

- Cookies;
- Javascript;
- AJAX;
- DHTML.

#### 5.5. Требования к одноплатному компьютеру для установки комплекса:

- Совместимость с Raspberry-PI 3+ или выше;
- Операционная система Raspbian Linux 32-bit или 64-bit или совместимая;
- Установлены пакеты python 3, pip3, flask-python3, Rpi.GPIO-python3;
- Свободное дисковое пространство 20Мбайт.

#### 5.6. Установка комплекса

**1 вариант.** Пользователю передается архив файлов в формате zip для развертывания в системе одноплатного компьютера.

Действия по установке:

- Развернуть архив в каталоге /home/pi/gpsrv;
- Файл doc/gpsrv.service скопировать в /etc/systemd/system;
- Выполнить команды `sudo systemctl enable gpsrv.service; sudo systemctl start gpsrv.service`
- В браузере на рабочем месте администратора открыть url: <http://<IP>:8084/gp/static>, где IP — это IP адрес либо имя DNS одноплатного компьютера.

**2 вариант.** Установка с помощью бинарного образа.

Пользователю передается ссылка на бинарный образ. Дальнейшие действия пользователя:

- пользователь записывает образ на электронный носитель micro-sd card;
- устанавливает micro-sd в одноплатный компьютер raspberry pi 3;
- подключает одноплатный компьютер raspberry pi 3 к локальной сети.

В случае подключения по ethernet настройки производятся автоматически. Для настроек по wifi требуется дополнительная конфигурация устройства, логин и пароль для входа на устройство pi/raspberry.

Конфигурация wifi в ОС linux debian производится стандартным способом (например <https://wiki.debian.org/ru/WiFi/HowToUse>).

В браузере на рабочем месте администратора открыть url: <http://<IP>:8084/gp/static>, где IP — это IP адрес либо имя DNS одноплатного компьютера.

## 2. Работа пользователя в панели управления

(на примере аппаратного обеспечения «Воротная группа» <https://100car.rutp.ru/gp/>)

### 1. Вход в систему. Главная страница.

Каждый Комплекс имеет одного пользователя, который управляет Комплексом, подключаясь к своему микрокомпьютеру.

У микрокомпьютера имеется собственный ip-адрес, который с рабочего места администратора через локальную сеть или Интернет открывается в браузере.

На главной странице панели управления размещены:

Слева - блок управления, включает в себя вкладки:

- [Порты];
- [Переменные];
- [Команды];
- [Фразы];

- [Состояния];
- [Монитор].

Справа: конфигурация системы: визуальная конфигурация портов на микрокомпьютере и общие настройки параметров.

Для создания конфигурации необходимо заполнить поля:

- [Название] – название конфигурации;
- [Частота] – частота опроса портов в секунду;
- [CRM id] (пример для интеграции в CRM-системой в создании команды).

Дополнительные функции:

- [Скачать] файл конфигурации;
- [Загрузить] файл конфигурации;
- [Скачать бэкап].
- [Загрузить]/[Удалить] файл (например, файлы mp3 для использования в некоторых командах, которые проигрывают файлы).

После внесения/редактирования данных в полях нажимаем [Применить].

## 2. Вкладка [Порты]. Описание физического порта.

Данная вкладка позволяет создать перечень портов, которые будут участвовать в работе, контроле и мониторинге оборудования, в соответствии с техническими и бизнес-процессами работы данного оборудования. используя все физические порты (ввода и вывода) микрокомпьютера, подключенного к аппаратному обеспечению.

Порты ввода (входные порты, IN) - электронные устройства, на которые извне поступают какие-либо входные сигналы от датчиков, кнопок или других систем, предназначенных для управления оборудованием (в нашем примере: сигналы, возникающие при срабатывании датчиков звука, температуры и т.д.: подъехала/не подъехала машина, машина в боксе / нет, открыты/закрыты ворота).

Через порты вывода (выходные порты, OUT) осуществляется управление аппаратным оборудованием, то есть передаются выходные сигналы: людям, исполнительным устройствам или в другие системы (в нашем примере: сигнал на закрытие/открытие ворот, сигнал на переключение света светофора, включение/выключение света в боксе).

Описание порта:

Порт на устройстве:

Тип порта:  IN  OUT

Тип подтяжки:  PULLUP  PULLDOWN  нет

Начальный:  LOW  HIGH

GPIO	Тип	Подтяжка	Состояние	Описание	Удалить
GPIO 1	IN	HIGH	LOW	Светофор Красный	<input type="button" value="Удалить"/>
GPIO 2	IN	нет	нет	Линия мойки	<input type="button" value="Удалить"/>
GPIO 3	IN	нет	нет	Линия ворот	<input type="button" value="Удалить"/>
GPIO 4	IN	нет	нет	Линия перед воротами	<input type="button" value="Удалить"/>
GPIO 9	IN	HIGH	LOW	Включить освещение!	<input type="button" value="Удалить"/>

В верхней части вкладки размещено изображение с названиями портов одноплатного компьютера (из документации Raspberry).

Далее блок для добавления порта и его описания:

**[Описание порта]** – текстовое поле, описать коротко, но максимально понятно, какое действие происходит;

**[Порт на устройстве]** – выбрать из списка свободных портов. Занятые порты в списке отсутствуют.

**[Тип порта]:**

- **IN (вход)** - входной порт, с которого считываются и анализируются данные.
- **OUT (выход)** - выходной порт, на котором устанавливается уровень сигнала (1 – включился, 0 – выключился).

**[Тип подтяжки]** - поле активно только для входных (IN) портов. Использование данного параметра зависит от задач в прикладном ПО.

- **PULLUP** - порт изначально включен, резистор с повышением напряжения;
- **PULLDOWN** - порт изначально выключен, резистор с понижением напряжения;

**[Начальное состояние]** - можно задавать только для выходных (OUT) портов.

- **LOW** – низкое состояние, выключен;
- **HIGH** - высокое состояние, включен.

Заполнив все поля, нажимаем **[Добавить]**. Порт добавляется в список. У каждого порта есть опция **[Удалить]**.

### 3. Вкладка **[Переменные]**. Создание переменных.

Переменные используются для выстраивания сложной логики работы системы, так не всегда хватает одного датчика или исполнительного устройства для описания состояния.

Переменная может принимать только два значения — «истина» и «ложь», Переменные могут менять значение в зависимости от конкретного порта и реализации состояния.

Интерфейс пользователя для описания списка переменных:

**[Название переменной]** - текстовое поле, латинскими буквами, без пробелов;

**[Описание]** - текстовое поле, описать коротко, но максимально понятно, какое действие происходит;

**[Начальное значение]** - указать начальное значение переменной, которое будет установлено при запуске Комплекса. Любое текстовое значение, в зависимости от предметной области, в которой применяется данный Комплекс.

Заполнив все поля, нажимаем **[Добавить]**. Переменная добавляется в список. У каждой переменной есть опция **[Удалить]**.



Название переменной	Описание переменной	Начальное значение	
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="button" value="Добавить"/>
Укажите название переменной      Укажите краткое описание переменной      Укажите начальное значение переменной			
_instatetime = 0 (Системная) Время в состоянии			
avtobox = 0 Авто в боксе?			<input type="button" value="Удалить"/>
closebox = 0 Ворота закрыты?			<input type="button" value="Удалить"/>
emergency = 0 безопасность			<input type="button" value="Удалить"/>
garantV = 0 Гарантированный выезд			<input type="button" value="Удалить"/>
garantZ = 0 Гарантированный въезд			<input type="button" value="Удалить"/>

#### 4. Вкладка [Команды]. Создание команд.

Команда	Описание команды	
<input type="text"/>	<input type="text"/>	<input type="button" value="Добавить"/>
Задайте команду из каталога gp/bin      Укажите краткое описание		
1 playmp3 проигрывание аудио файла .mp3		<input type="button" value="Удалить"/>
5 test.sh тест		<input type="button" value="Удалить"/>
6 tg Отправка сообщение в Telegram		<input type="button" value="Удалить"/>

Команды – это список указаний в операционной системе, которые могут быть выполнены при переключении состояний. Команды – это скрипты, которые к Комплексу не относятся, но могут храниться в каталоге и исполняться. Команды создаются индивидуально для конкретной предметной области. Через Команды возможна интеграция Комплекса с внешними системами и дополнительным ПО.

Команды (скрипты) должны содержаться в каталоге gp/bin (каталог, куда установлена серверная часть Комплекса). Если при описании какого-либо состояния потребуется команда (например, включить голосовое сообщение, направить уведомление в ctm-систему), ее необходимо заранее сохранить в каталоге.

Интерфейс пользователя для описания списка команд:

- **[Название команды]** – текстовое поле, ввести название команды, как в каталоге;
- **[Описание команды]** – текстовое поле, коротко, но максимально понятно описать команду.

#### 5. Вкладка [Фразы]. Создание фраз.

Вкладка представляет собой справочник фраз, имеющих короткий ключ. Фразы могут сопровождать различные состояния оборудования и действия пользователя в прикладном ПО.

Фразы используются как параметры для Команды при создании состояний.

При создании состояний используется короткий ключ фразы, пользователь увидит полную фразу.

Интерфейс пользователя для описания списка фраз:

**[Ключ]** - текстовое поле, короткое название латинскими буквами.

**[Фраза]** – текстовое поле, фраза, которая подставляется вместо ключа при выполнении команды или состояния с использованием фразы.

Ключ	Фраза	
<input type="text"/>	<input type="text"/>	<input type="button" value="Добавить"/>
Краткий ключ для фразы	Полная фраза	
<input type="checkbox"/> alert	Внимание!	<input type="button" value="Удалить"/>
<input type="checkbox"/> hello	Привет!	<input type="button" value="Удалить"/>

## 6. Вкладка [Состояния]. Создание Состояния.

Состояние ✕

Укажите краткое описание состояния

**Входные условия**

Порт **Переменная**

HIGH (1)  
 LOW (0)

Порты

Переменные

**Выходные параметры**

Порт **Переменная** **Команда** **Таймаут**

HIGH (1)  
 LOW (0)

Задержка, с  Длительность, с

0 без задержки 0 без ограничения

Установка немедленно !

Порты

Переменные

Команды

Вкладка [Состояния] представляет собой:

- 1) конструктор, с помощью которого, используя заранее созданные и описанные Порты, Переменные, Фразы, Команды, можно конструировать различные состояния, описывающие алгоритмы действий и процессы аппаратного обеспечения.
- 2) список созданных состояний.

Алгоритм создание нового состояния:

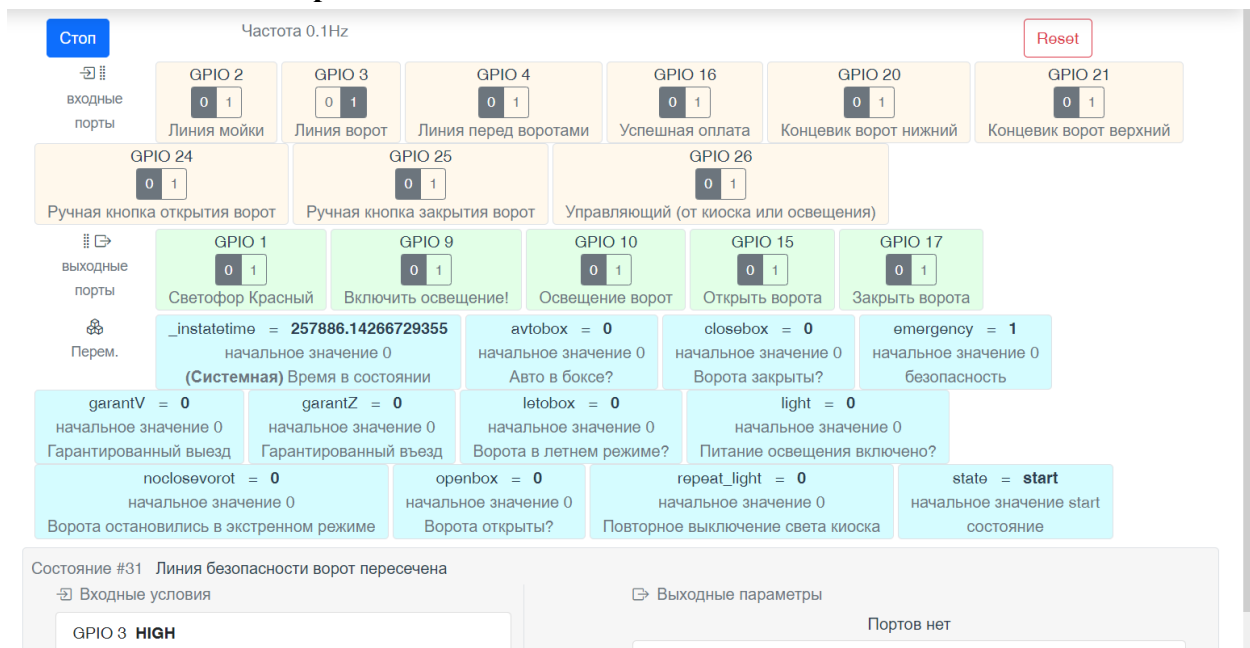
1. Нажимаем на [Новое состояние].
2. Открывается диалоговое окно, в поле [Краткое описание] —вносим описание для отображения в списке состояний.
3. В блоке [Входные условия] устанавливаем условия уровня порта ввода и условия переменных:
  - Во вкладке [Порт]: выбираем порт из списка созданных входных портов (с типом IN). Задаем для порта условие: устанавливаем чекбокс [HIGH] или [LOW]. Нажимаем [Добавить]. Входной порт появился в списке внизу. Добавляем при необходимости другие порты.
  - Во вкладке [Переменная] выбираем, при необходимости, переменную из списка, проверяем условие. Нажимаем [Добавить]. Входная переменная появилась в списке внизу. Добавляем при необходимости другие переменные.
  - Входные условия готовы.
4. В блоке [Выходные параметры] выставляем уровни порта вывода, устанавливаем условия переменной, запуск внешней команды и действие по таймауту.

- Во вкладке [Порт] из списка выходных портов (тип OUT) выбираем нужный. Устанавливаем для выбранного порта условие: четбокс [HIGH] или [LOW]. При необходимости устанавливаем временные параметры действия (в сек.) в полях [Задержка], [Длительность], четбокс [Установка немедленно]. Нажимаем [Добавить].
- Во вкладке [Переменная] добавляем при необходимости переменную, устанавливаем ее значение, добавляем переменную.
- Во вкладке [Команды] выбираем команду из списка и указываем, при необходимости, ее [Параметры]. В [Параметрах] указывается дополнительная информация (добавить [Фразу], название файла, место хранения файла и т.д.) Параметры задаются через пробел.
- Во вкладке [Таймаут] можно задать [Команды] и [Переменные], а также временные значения для них: в ситуации, если система находится продолжительное время в текущем состоянии, можно выставить условия, при которых сработает таймаут (выставится [Переменная] или сработает [Команда]).
- Выходные параметры готовы. Созданное состояние добавлено в список состояний.

5. Во вкладке [Состояния] есть возможность:

- [Править] (редактировать) состояние;
- [Удалить] состояние;
- Задать активность состоянию (с помощью переключателя)

## 7. Вкладка [Монитор].



Во вкладке [Монитор] пользователь имеет возможность:

- 1) наблюдать за процессом переключения состояний, состоянием входных и выходных портов действующего объекта.

В этом случае администратор наблюдает текущее состояние системы, в зависимости от текущего состояния портов в режиме реального времени.

- 2) проводить тестирование, настройку и отладку состояний и условий работы оборудования без подключения к действующему объекту.

Если система не подключена к действующему устройству, то во вкладке [Монитор] можно имитировать состояния входящих портов для отладки и тестирования работы оборудования, а именно:

- включить микрокомпьютер, во вкладке [Монитор] выбрать входящий порт (можно одновременно выбрать несколько входных портов, в зависимости от созданных состояний);
- поменять состояние входящего порта на противоположное (высокое-низкое, 0-1);
- далее система проверяет и выставляет то состояние, в котором используется данный порт(ы).

Администратор анализирует состояние: правильно ли меняется состояние согласно описанию процессов, проверяет на наличие ошибок, достаточности и достоверности данных, правильности выставления переменные и т.д.

Для завершения сеанса работы в панели управления необходимо выключить микрокомпьютер.