

УДК 621.31 (073)

DOI: 10.21209/2227-9245-2019-25-10-31-39

**ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ СИСТЕМЫ SCADA ДЛЯ УПРАВЛЕНИЯ  
ЭЛЕКТРОПРИВОДОМ УСТАНОВКИ ШАХТНОГО ПОДЪЕМА****THE SOFTWARE OF THE SCADA SYSTEM FOR CONTROL OF THE ELECTRIC DRIVE  
OF MINE RISE INSTALLATION**

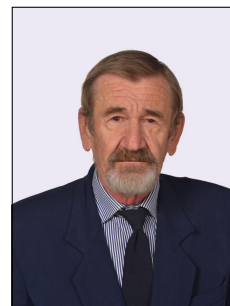
**Е. Д. Тельманова,**  
Уральский государственный  
горный университет,  
г. Екатеринбург  
telmanova\_rsvpu@mail.ru

**E. Telmanova,**  
Ural State Mining University,  
Yekaterinburg, Russia



**В. Т. Трапезников,**  
Уральский государственный  
горный университет,  
г. Екатеринбург  
trapeznikov.v@mail.ru

**V. Trapeznikov,**  
Ural State Mining University,  
Yekaterinburg, Russia



**Р**ассматривается вопрос проектирования и безопасной эксплуатации инженерных сооружений в производственной среде шахты. Внедрение системы SCADA обеспечит высокий уровень безопасности при перемещении людей посредством мониторинга, визуализации, централизованного контроля и управления процессом передвижения установки шахтного подъема. Представлено обоснование способа реализации SCADA-системы для управления электроприводом шахтного подъемника. Система SCADA иерархически выстроена по четырем уровням: уровень устройств; уровень транспортов; уровень транспортных протоколов; уровень контроллеров логического уровня КЛУ.

Приведены решения с использованием наиболее стабильной и безопасной технологии. Показаны основные подходы к работе с программным обеспечением (ПО) контроллеров и системы OpenSCADA. Для управления электроприводом установки шахтного подъема посредством технологии SCADA осуществлен выбор технических средств, таких как одноплатный компьютер Raspberry Pi, преобразователь частоты SINAMICS V20, контроллер Arduino. При выборе SCADA-системы выполнен анализ существующих систем по принципам их взаимодействия с внешними устройствами. В качестве системного ПО определены: операционная система персонального компьютера GNU/Linux; дистрибутив Raspbian; встроенное ПО преобразователя частоты; встроенное ПО контроллера. Для разработки прошивок и их загрузки в память микроконтроллера Arduino IDE выбрана требуемая среда. В качестве прикладного ПО рассмотрена OpenSCADA и все программы, написанные в ней, а также встраиваемая система управления базами данных (СУБД) – SQLite, которую SCADA использует для хранения данных.

Описаны процесс настройки выходных транспортов и особенности обмена данными в системе OpenSCADA. Представлена нестандартная графическая оболочка SCADA-системы, которая обеспечивает мониторинг процесса перемещения установки шахтного подъема, а также позволяет работать оператору в режиме прямого управления электроприводом при возникновении нештатных ситуаций

**Ключевые слова:** установка шахтного подъема; управление; OpenSCADA; программное обеспечение; электропривод; преобразователь частоты; мониторинг; безопасность; горные машины; внешние устройства

**I**n the article the question of design and safe operation of engineering constructions in the production environment of the mine is considered. System implementation of SCADA will provide the high level of safety when moving people by means of monitoring, visualization, the centralized control and process control of movement of mine rise installation. Justification of a way of SCADA system implementation for control of the electric drive of the mine elevator is presented. The SCADA systems about four levels are built hierarchically: level of devices; level of transports; level of transport protocols; level of controllers of the logic level of KLU.

Solutions, with use of the stablest and safe technology are given. Basic approaches to work with the software of controllers and the OpenSCADA system are shown. For control of the electric drive of mine rise installation by means of SCADA technology the choice of technical means, such as single board computer Raspberry Pi, frequency converter SINAMICS V20, Arduino controller is carried out. When choosing SCADA system the analysis of the existing systems by the principles of their interaction with external devices was made. As the system software were defined: operating system of the GNU/Linux personal computer; Raspbian distribution kit; built in by frequency converter software; built in by controller software. For development of firmwares and their memory load of the Arduino IDE microcontroller the required environment is selected. As the application software OpenSCADA and all programs written in it and also, the built-in DBMS – to SQLite which SCADA uses for data storage is considered.

In the article process of setup of output transports and feature of data exchange in the OpenSCADA system are described. The non-standard graphic cover of SCADA system which provides monitoring of process of movement of installation of mine rise is presented and also allows to work to the operator in the mode of direct control of the electric drive at emergence of emergency situations

**Key words:** *installation of mine rise; management; OpenSCADA; software; electric drive; frequency converter; monitoring; security; mining cars; external devices*

**В**ведение. Внедрение в производственную среду шахт системы SCADA позволит не только добиться роста производительности, но и обеспечит необходимый уровень безопасности при производстве горных работ. Система SCADA способна обеспечить визуализацию, мониторинг, контроль и управление оборудованием для очистки угля, конвейеров, шахтных подъемников. Позволяет вести мониторинг всех аспектов безопасности и инфраструктуры (электроснабжение, шахтные воды, атмосферное давление, сжатый воздух, газовыделение), а также отслеживать движение людей и транспортных средств. Любое контролируемое событие отображается с помощью видеосигналов для сотрудников диспетчерского пункта.

В настоящее время внедрение программного обеспечения третьего уровня SCADA для централизованного контроля и управления автоматизированным электроприводом горных машин становится все более актуальным. Объясняется это общей тенденцией к увеличению глубин шахт и, как следствие, максимальных скоростей установок шахтного подъема [1]. Это требует обеспечения устойчивой информации о процессе перемещения механизма с изображением мнемосхем или анимации. Необходима повышенная защита от несанкционированного доступа к системам управления, а также система предупреждения о возникновении событий, приводящих к аварийным ситуациям. Все эти функции с помощью оператора может осуществлять SCADA-система.

**Объект исследования.** Система SCADA для управления оборудованием в производственной среде шахт.

**Предмет исследования.** Процесс проектирования системы SCADA для управления электроприводом установки шахтного подъема.

**Цель исследования.** Создание системы управления автоматизированным электроприводом подъемной установки для безопасного перемещения людей в среде шахт.

**Методы и методология исследования.** При разработке программного обеспечения системы SCADA на уровне логических контроллеров применен граф конечного автомата лифта. Для обеспечения взаимодействия OpenSCADA с силовым преобразователем используются числа с плавающей запятой в формате IEEE 754, что расширяет диапазон данных с плавающей запятой и является более компактным с точки зрения хранения.

Для разработки программного обеспечения применена объектно-ориентированная методология Real [5].

**Результаты исследования и их обсуждение.** Одним из этапов построения SCADA-системы вспомогательной установки шахтного подъема является выбор ПО и технических средств для системы мониторинга, контроля и управления электрооборудованием установки.

В нашем случае для приводного двигателя установки шахтного подъема (трехфазный асинхронный двигатель переменного тока с короткозамкнутым ротором) и для таких технических средств, как однофазный ком-

пьютер Raspberry Pi, преобразователь частоты SINAMICS V20, контроллер Arduino, в качестве системного ПО определено следующее:

- операционная система персонального компьютера GNU/Linux;
- дистрибутив Raspbian, превращающий Raspberry Pi в маршрутизатор;
- встроенное ПО преобразователя частоты (установлено на заводе);
- встроенное ПО контроллера (разработано специально для данной установки).

Для разработки прошивок и загрузки их в память микроконтроллера выбрана среда – Arduino IDE.

К прикладному ПО также относится SCADA-система, все программы, написанные в ней, а также компактная встраиваемая СУБД – SQLite, которую SCADA использует для хранения данных.

При выборе SCADA-системы выполнен анализ существующих систем по принципам их взаимодействия с внешними устройствами (датчиками, счетчиками, контроллерами и т. д.):

- системы, имеющие в своем составе большое количество драйверов, обеспечивающих возможность взаимодействия со всеми поддерживаемыми устройствами (“Trace mode”);
- системы, не имеющие средств прямого обмена информацией, но поддерживающие подключение OPC-серверов, обеспечивающих обмен данными (“MasterSCADA”) [3].

Также учитывался преобразователь частоты SINAMICS V20, управление которым осуществляется с использованием протокола USS, который узкоспециализирован и не поддерживается ни “Trace mode”, ни “MasterSCADA”.

Одной из SCADA-систем, поддерживающих USS, является OpenSCADA [2]. Эта система, обеспечивая поддержку всех типовых функций современных SCADA-систем, позволяет реализовать пользовательские протоколы, содержит библиотеки для разработки, интерфейсные приложения, конфигурационные утилиты, внутренние и внешние интерфейсы.

Взаимодействие различных частей разрабатываемой системы SCADA представим в виде иерархически выстроенной системы, состоящей из четырех уровней:

- 1) уровень устройств;
- 2) уровень транспортов;

3) уровень транспортных протоколов;

4) уровень контроллеров логического уровня КЛУ.

На 1-м уровне расположен преобразователь частоты с подключенным двигателем, сконфигурированный для дистанционного управления с помощью конвертера RS-485/USB (подключен к ПК). Операционная система ОС определяет его как устройство с последовательным интерфейсом. Аналогичным образом контроллер, на дискретные входы которого поступает информация о состоянии кнопок поста приказов и постов вызова, состояний кнопок «Стоп» и данных с энкодера, подключается к ПК. Устройство также является устройством с последовательным интерфейсом.

В SCADA-системе на 2-м уровне каждому из подключенных устройств, или устройств, подключение которых ожидается, ставятся в соответствие наименования, чтобы получить возможность впоследствии их различать, а также задаются параметры передачи, чтобы обмен данными реализовался.

На 3-м уровне происходит составление пакета в соответствии с правилами, предписываемыми тем или иным протоколом обмена данными, а так же разбор ответа. Данные, полученные от различных источников, интерпретируются по-разному (формируются посылки, различным образом). Поэтому, несмотря на схожие интерфейсы передачи, протоколы обмена данными различаются.

На 4-м уровне данные, получаемые от внешних устройств, должны обновлять (актуализировать) состояние системы. На этом уровне находится контроллер логического уровня “SIMATIC S7”. Этот контроллер, выполняя свою программу, должен посылать преобразователю частоты сообщения о пуске, остановке, торможении, сбросе ошибки, запрашивать текущую частоту напряжения на выходе инвертора [9]. Контроллер не является самостоятельным и не принимает сам решений о том, какое действие сейчас будет выполнять. Вторым контроллером на 4-м уровне является “Arduino”. Его программа также не является управляющей, лишь запрашивая у физического контроллера актуальные данные о состоянии кнопок и пр. [6]. Далее предоставляет их для использования в общий доступ внутри SCADA-системы.

Третьим контроллером является КЛУ «Лифт», так как управляющая программа

вспомогательной установки шахтного подъема реализуется на основе конечного автомата лифтовой установки. Эта программа, получая данные от других контроллеров логического уровня и оператора SCADA-системы, принимает решение о текущих действиях.

Для успешного управления преобразователем частоты по протоколу USS необходимо придерживаться ряда правил [10]. Структура кадра на канальном уровне протокола USS имеет следующий вид: «STX | LGE | ADR | DATA | BCC» (символ «|» применен для разграничения элементов кадра). Здесь содержатся следующие элементы кадра:

– STX – элемент размером 1 байт, являющийся стартовым байтом посылки; в соответствии с протоколом USS он должен всегда иметь значение 0x02 (здесь и везде далее: числа, запись которых начинается с символов «0x», записаны в шестнадцатеричной системе счисления);

– LGE – элемент размером 1 байт, указывающий на длину передаваемого сообщения, максимальное значение которого 256 (значение вычисляется как сумма длин полей ADR, DATA и BCC);

– ADR – элемент размером 1 байт, содержащий адрес устройства, которому предназначается сообщение (лежит в пределах 0...31);

– DATA – элемент кадра переменной длины, являющийся самим сообщением (максимальная длина, как можно заключить из описания поля LGE – 254 байта);

– BCC – элемент размером 1 байт, являющийся контрольной суммой (вычисляется последовательным применением ко всем предыдущим байтам кадра операции или с начальным значением 0).

Элемент DATA состоит из двух полей: PKW и PZD. Поле PKW состоит из двух пар двухбайтовых слов, первое из которых – идентификатор параметра, а второе – его значение.

Настройка выходных транспортов в OpenSCADA позволяет определить те информационные каналы, которые будут использованы для обмена данными. Для этого в созданном проекте OpenSCADA необходимо перейти в раздел «Транспорты», а затем – в «Последовательные интерфейсы». Так как главным устройством является персональный компьютер, а силовой преобразователь является подчиненным, инициировать обмен

может только компьютер. Поэтому достаточно добавить «Выходной транспорт», назвав его “SS7”. В окне настройки транспорта в строке «Адрес» указываем идентификатор последовательного интерфейса, скорость обмена и параметры передачи (количество бит в посылке, контроль четности и количество стоп-бит). При этом идентификатор интерфейса будет иметь вид “/dev/ttyUSBx”, где x – порядковый номер в системе, скорость обмена и параметры передачи берутся из справочника [10]. Окончательный вид строки – “/dev/ttyUSB0:9600:8E1”. Обмен данными с контроллером происходит также по последовательному интерфейсу, и только ПК может являться инициатором обмена. Поэтому создается еще один «Выходной транспорт», с названием “plc”. Идентификатор интерфейса будет иметь иной вид, так как в контроллере применена FTDI микросхема USB-UART. Поэтому в поле «Адрес» записываем “/dev/ttyACM0:57600:8E1”.

Для реализации протокола USS необходимо перейти в раздел «Транспортные протоколы» и создать «Пользовательский протокол», назвав его “SS7\_USS”. После открытия окна настройки созданного протокола необходимо перейти во вкладку «Выход» и выбрать Java-подобный язык выходной программы “JavaLikeCalc.JavaScript”. В текстовое поле «Выходная программа» необходимо поместить программу, которая будет вызываться каждый раз, когда с более высокого уровня иерархии будет приходиться запрос на обмен данными. Отметим, что запрос всегда должен сопровождаться информацией об адресе, к которому он направляется (ADDR), а также содержимым полей PKW, PZD и SPD. Все указанные данные при вызове выходной программы находятся в объекте “io”, доступ к полям которого осуществляется методом “attr”.

Требуемый алгоритм функционирования следующий. Запрос для уровня транспортов начинает формироваться со значения 0x020E (0x0E соответствует длине пакета 14 байт). Можно сделать так, чтобы длина пакета вычислялась каждый раз при выполнении выходной программы, но в рамках данной разработки нет необходимости работать с пакетами переменной длины (пакетами, с несколькими полями PKW), поэтому для упрощения длина пакета задается константой. Затем запрос дополняется параметрами

ADDR, PKW, PZD, SPD. При этом значение параметра SPD является вещественным числом в пределах 0,0...50,0, в то время как преобразователь частоты ожидает значение 0x0000...0x4000. Поэтому SPD предварительно нормализуется. При этом обмен данными на транспортном уровне производится в двоичном виде, а запрос до сих пор формировался как символьная строка с шестнадцатеричными числами. Для приведения формируемого запроса к необходимому виду используется функция "strEnc2Bin" из модуля со специальными функциями Special.FLibSys, который подключается в начале. В финале вычисляется контрольная сумма полученного запроса и добавляется в его конец. С помощью метода "messIO" объекта транспорта "tr" запрос отправляется, а ответ записывается в переменную resp. Полученный ответ помещается в узел обмена данными с верхними уровнями "io" с помощью метода "setAttr".

Для формирования программы, обеспечивающей взаимодействие с силовым преобразователем, в разделе «Сбор данных» OpenSCADA в «Библиотеке устройств» необходимо создать «шаблон» программы с названием "SS7". Желаемый алгоритм функционирования следующий. При первом запуске программы (проверяется с помощью флага f\_start) должен быть создан узел обмена данными с указанием используемого протокола (USS), а так как состояние привода в момент запуска программы неизвестно, для исключения аварийных ситуаций на вход остановки движения должен быть искусственно подан сигнал «Стоп».

Так как программа имеет четыре входа ("start", "stop", "fStop", "resetError"), то при поступлении на них сигналов программа должна отправить преобразователю сообщение о необходимости начала движения либо об остановке на выбеге, либо о торможении, либо о сбросе состояния ошибки соответственно. При отсутствии какого-либо сигнала программа отправляет запрос на получение текущей частоты напряжения на выходе инвертора, помещая результат в переменную invFreq. Также присутствует входная переменная, куда поступают: адрес преобразователя, к которому осуществляется обращение, переменная, отражающая направление вращения, и переменная, где находится задание по скорости.

Поле «Атрибут» переменных start, stop, fStop, resetError, address, speed и dir опреде-

ляется как «Полный доступ», так как названные переменные должны устанавливаться с более высоких уровней иерархии (из КЛУ «Лифт» или из графической оболочки). «Атрибут» переменной invFreq определяется как «Только чтение», так как изменение его значения недопустимо. Вид сообщений, отправляемых на преобразователь частоты, определяется содержанием полей PKW и PZD. Ответом на запрос частоты напряжения на выходе инвертора со стороны преобразователя является стандартная посылка, в теле которой находится число с плавающей запятой (десятичная дробь) с одинарной точностью (4 байта на число), которое соответствует стандарту IEEE 754 [4].

Необходимо отметить, что обмен данными в SCADA-системе происходит в виде строк, а число с плавающей запятой – двоичные данные. В OpenSCADA не существует функции перевода части строки в число с плавающей запятой (слишком частный случай), но существует функция объединения двух целочисленных двухбайтовых значений в одно число с плавающей запятой одинарной точности "floatMergeWord". При этом предполагается, что данные в целочисленных значениях – это знак мантиссы, мантисса и показатель экспоненты. То есть, OpenSCADA также использует числа с плавающей запятой в формате IEEE 754, что и делает возможным дальнейшую работу. Таким образом, алгоритм для получения скорости следующий: выделить из строки ответа блок символов длиной четыре байта; сформировать два числа (знак мантиссы, мантисса, первая часть показателя экспоненты и вторая часть показателя экспоненты), учитывая, что порядок байт – «сначала старший», использовать системную функцию и получить число с плавающей запятой.

Программа лифта – последняя ступень в иерархической модели на уровне логических контроллеров SCADA. Реализовать программу лифта (установки шахтного подъема) можно в виде конечного автомата [7]. Состояниями конечного автомата будут: ожидание команды, движение и ошибка. Граф конечного автомата лифта представлен на рис. 1.

Прикладная программа взаимодействия с контроллером Arduino должна действовать, согласно следующему алгоритму: отправлять запрос на уровень транспортных протоколов для получения актуальных данных от кон-

троллера; извлекать из узла обмена данные о наличии ошибок передачи и ответ контроллера; в случае отсутствия ошибок извлекать из ответа контроллера данные о состоянии кнопок поста приказов, постов вызова, кно-

пок «Стоп»; в случае необходимости генерировать управляющие воздействия для КЛУ «Лифт». Входные и выходные переменные приведены в таблице.

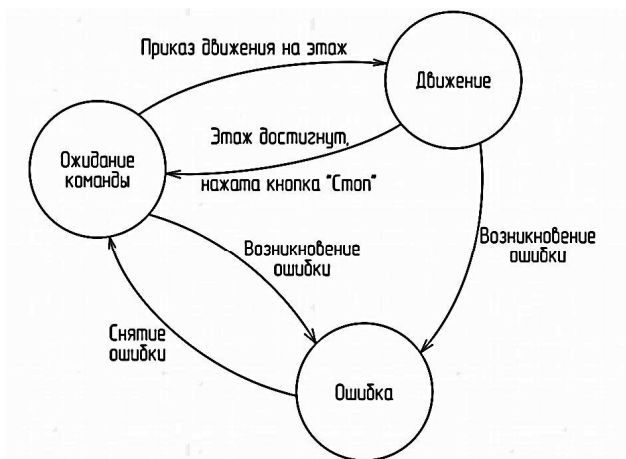


Рис. 1. Граф конечного автомата лифта (установка шахтного подъема) / Fig. 1. Elevator state machine graph (installation of mine rise)

Входы и выходы программы КЛУ «Контроллер-Arduino» / Inputs and outputs of "Controller-Arduino" CLU program

Идентификатор / Identifier	Имя / Name	Тип / Type	Режим / Mode	Атрибут / Attribute	Конфигурация / Configuration
DeltaEndImp	Разница в количестве импульсов энкодера / Difference in number of impulses of an encoder	Целый / Whole	Выход / Exit	Полный доступ / Full access	Связь / Communication
SelectedFloor	Выбранный уровень / The selected level	Целый / Whole	Выход / Exit	Не атрибут / Not attribute	Связь / Communication
CurrentFloor	Текущий уровень / Current level	Целый / Whole	Вход / input	Не атрибут / Not attribute	Связь / Communication
StartMove	Команда к началу движения / Command to a start of movement	Логический / Logical	Выход / Exit	Не атрибут / Not attribute	Связь / Communication
StopPressed	Нажата кнопка «Стоп» / Depressed key «Feet»	Логический / Logical	Выход / Exit	Не атрибут / Not attribute	Связь / Communication

Обнаружение факта нажатия на какую-либо кнопку осуществляется с помощью операции «побитового И» из байта, отражающего состояние порта, далее выделяется необходимый бит, а затем он инвертируется, так как подтягивающие резисторы подтягивают входы к логической единице, и нажатие кнопки приводит к соединению соответствующего входа с нулем.

Интерфейс обмена данными с контроллером определяется техническими возможностями контроллера и ПК, а также из соображений унификации с уже существующим интерфейсом связи между компьютером и силовым преобразователем и является последовательным интерфейсом (UART). При этом никаких ограничений к протоколу не существует. Более того – и со стороны

SCADA-системы, и со стороны контроллера реализация выбранного протокола может быть написана самостоятельно. Поэтому для простейшего протокола, исходя из целей максимальной простоты, минимальной длины посылки и хорошей сохранности данных, параметры UART-интерфейса могут быть следующими: скорость обмена – 57 600 бод, количество бит данных – 8, контроль четности – «контроль на четность», количество стоп-бит – 1. Обмен данными инициирует ПК, отправляя посылку со стартовым битом «?» (0x3F), номером текущего уровня (1 байт) и контрольной суммой (1 байт), вычисляемой по уже известному алгоритму, изначально взятому из протокола USS.

Графическая оболочка SCADA-системы выполнена с учетом, что она должна предоставлять пользователю возможность работать в двух режимах: в режиме прямого управления приводом, при котором оператор может задавать скорость, направление

вращения, запускать, останавливать или тормозить электропривод шахтного подъемника и в автоматическом (в соответствии с программой). Действия в ручном режиме необходимы для отладочных целей. В режиме управления «Подъемник» оператору должна предоставляться вся информация, касающаяся программы контроллеров логического уровня «Подъемник». А именно, текущее состояние конечного автомата, текущий этаж, данные с энкодера, наличие или отсутствие блокировки дверей кабины, наличие ошибки. Оператор должен иметь возможность сбросить состояние ошибки и управлять подъемником, отправляя приказы движения на один из этажей. Вид графической оболочки представлен на рис. 2.

На графическом поле разработанной оболочки верхнего уровня размещены различные элементы кнопок, текстовых полей, полей для ввода чисел, выпадающих списков, элементов-индикаторов.

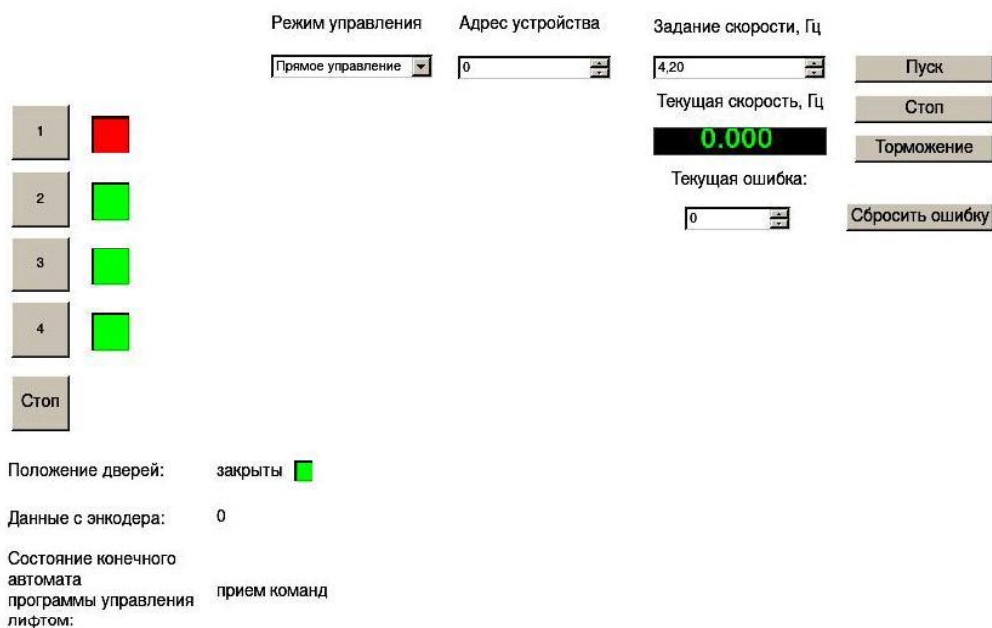


Рис. 2. Графическая оболочка SCADA-системы / Fig. 2. SCADA Graphics Shell

**Заключение.** Система SCADA состоит из множества различных технологий для разных целей: от регистрации и сохранения данных до протоколов связи и стандартов доступа к данным. Поэтому при выборе способа реализации системы важно знать об этих технологиях. SCADA часто представляет собой

обширную систему со многими компонентами, которые должны работать вместе. В то же время система должна быть безопасной. По названным причинам не всегда рационально, лучше использовать новейшие технологии, а реализовать наиболее стабильные и безопасные. Особенно при проведении горных

работ. Не все компоненты, в частности на нижнем уровне (ПЛК и RTU), способны использовать новейшие технологии для связи и т. д.

При выборе программного обеспечения SCADA необходимо учитывать множество факторов, как в отношении экономической целесообразности, так и в отношении технических аспектов [8]. Кроме того, необходимо учитывать контролируемые переменные. С экономической точки зрения важен срок службы программного обеспечения SCADA,

так как система может потребовать значительных инвестиций, а срок ее службы будет составлять 5...15 лет. В течение этого периода необходима уверенность в возможности получения техподдержки, обновления, возможности расширения и т.д.

Исходя из приведенных представлений, выполнена разработка SCADA-системы для управления электроприводом вспомогательной установки шахтного подъема.

## Список литературы

1. Дацковский Л. Х., Роговой В. И., Кузнецов И. С., Жидков А. А., Воликов А. А. Электропривод современных шахтных подъемных машин // Известия Тульского государственного университета. Серия «Технические науки». 2010. № 3. С. 157–165.
2. Дом – OpenSCADAWiki. URL: <http://www.oscada.org/wiki/Home/ru> (дата обращения: 08.07.2019). Текст: электронный.
3. Ицкович Э. Л. Современные SCADA-программы разных производителей: их свойства и отличия, важные для потенциальных заказчиков // Автоматизация в промышленности. 2007. № 4. С. 25–30.
4. Шундеев А. С. Введение в стандарт IEEE 754 // Программная инженерия. 2013. № 3. С. 44–47.
5. Alvarez J. M., Diaz M., Llopis L., Pimentel E., Troya J. M. Object-oriented methodology for embedded real-time systems // The computer journal. 2003. Vol. 46, No. 2. P. 123–145.
6. Arduino Uno. URL: <https://www.farnell.com/datasheets/1682209.pdf> (дата обращения: 08.07.2019). Текст: электронный.
7. Bang-Jensen J., Gutin G. Z. Digraphs: theory, algorithms and applications. London: Springer-Verlag, 2009. 754 p.
8. Reynders D., Mackay S., Wright E. Practical industrial data communications: best practice techniques. Oxford: Butterworth-Heinemann, 2004. 432 p.
9. SINAMICS V20: советы по началу работы: компактное руководство по эксплуатации. URL: <https://www.prosoft.ru/cms/f/465022/Советы+по+началу+работы+SINAMICS+V20.pdf> (дата обращения: 08.07.2019). Текст: электронный.
10. Universal serial interface protocol, USS protocol specification. URL: [https://www.cache.industry.siemens.com/dl/files/253/24178253/att\\_101238/v1/uss\\_24178253\\_spec\\_76.pdf](https://www.cache.industry.siemens.com/dl/files/253/24178253/att_101238/v1/uss_24178253_spec_76.pdf) (дата обращения: 08.07.2019). Текст: электронный.

## References

1. Datskovsky L. Kh., Rogovoi V. I., Kuznetsov I. S., Zhidkov A. A., Volikov A. A. *Izvestiya Tulsogo gosudarstvennogo universiteta. Seriya "Tehnicheskie nauki"* (Bulletin of the Tula State University. Series "Engineering"), 2010, no. 3, pp. 157–165.
2. *Dom – OpenSCADAWiki* (Home – OpenSCADAWiki). URL: <http://www.oscada.org/wiki/Home/ru> (Date of access: 08.07.2019). Text: electronic.
3. Itskovich E. L. *Avtomatizatsiya v promyshlennosti* (Automation in industry), 2007, no. 4, pp. 25–30.
4. Shundeev A. S. *Programmnyaya inzheneriya* (Software Engineering), 2013, no. 3, pp. 44–47.
5. Alvarez J. M., Diaz M., Llopis L., Pimentel E., Troya J. M. *The computer journal* (The computer journal), 2003, vol. 46, no. 2, pp. 123–145.
6. *Arduino Uno* (Arduino Uno). URL: <https://www.farnell.com/datasheets/1682209.pdf> (Date of access: 08.07.2019). Text: electronic.
7. Bang-Jensen J., Gutin G. Z. *Digraphs: theory, algorithms and applications* (Digraphs: theory, algorithms and applications). London: Springer-Verlag, 2009. 754 p.
8. Reynders D., Mackay S., Wright E. *Practical industrial data communications: best practice techniques* (Practical industrial data communications: best practice techniques). Oxford: Butterworth-Heinemann, 2004. 432 p.
9. *SINAMICS V20: sovety po nachalu raboty: kompaktnoe rukovodstvo po ekspluatatsii* (SINAMICS V20: getting started tips: a compact instruction manual). URL: <https://www.prosoft.ru/cms/f/465022/Tips+++++SINAMICS+V20.pdf> (Date of access: 08.07.2019). Text: electronic.



10. *Universal serial interface protocol, USS protocol specification* (Universal serial interface protocol, USS protocol specification). URL: [https://www.cache.industry.siemens.com/dl/files/253/24178253/att\\_101238/v1/uss\\_24178253\\_сpec\\_76.pdf](https://www.cache.industry.siemens.com/dl/files/253/24178253/att_101238/v1/uss_24178253_сpec_76.pdf) (Date of access: 08.07.2019). Text: electronic.

#### Коротко об авторах

---

*Тельманова Елена Дмитриевна*, канд. пед. наук, доцент кафедры электрификации горных предприятий, Уральский государственный горный университет, г. Екатеринбург, Россия. Область научных интересов: проектирование, эксплуатация инженерных сооружений в шахтах  
telmanova\_rsvpu@mail.ru

*Трапезников Владислав Тимофеевич*, доцент кафедры электрификации горных предприятий, Уральский государственный горный университет, г. Екатеринбург, Россия. Область научных интересов: проектирование, эксплуатация инженерных сооружений в шахтах  
trapeznikov.v@mail.ru

#### Briefly about the authors

---

*Elena Telmanova*, candidate of pedagogical sciences, associate professor, Electrification of Mining Enterprises department, Ural State Mining University, Yekaterinburg, Russia. Sphere of scientific interests: design, operation of engineering structures in mines

*Vladislav Trapeznikov*, associate professor, Electrification of Mining Enterprises department, Ural State Mining University, Yekaterinburg, Russia. Sphere of scientific interests: design, operation of engineering structures in mines

#### Образец цитирования

---

*Тельманова Е. Д., Трапезников В. Т. Программное обеспечение системы scada для управления электроприводом установки шахтного подъема // Вестник Забайкальского государственного университета. 2019. Т. 25, № 10. С. 31–39. DOI: 10.21209/2227-9245-2019-25-10-31-39.*

*Telmanova E., Trapeznikov V. The software of the scada system for control of the electric drive of installation of mine rise // Transbaikal State University Journal, 2019, vol. 25, no. 10, pp. 31–39. DOI: 10.21209/2227-9245-2019-25-10-31-39.*

Статья поступила в редакцию: 12.09.2019 г.  
Статья принята к публикации: 04.12.2019 г.