

5. Chih-Ming H., Feng-Li L., Jun-An T., Jun-An L., Bo-Chiuan C. Road Detection Based on Bread-first Search in Urban Traffic Scenes // 8th Asian Control Conf. (ASCC). –2011. – С. 1393–1397.
6. Hougardy S. The Floyd-Warshall algorithm on graphs with negative cycles // Inf. Process. Lett. –2018. –Т. 110. – № 8–9. – С. 279–281.
7. Cui S.-G., Wang H., Yang L. A simulation study of A-star algorithm for robot path planning // 16th International Conference on Mechatronics Technology. –2012. – С. 506–509.
8. Djojo M. A., Karyono K. Computational load analysis of Dijkstra, A*, and Floyd-Warshall algorithms in mesh network // In Proceedings of 2013 International Conference on Robotics, Biomimetics, Intelligent Computational Systems, ROBIONETICS 2013. –2013. –С. 104–108.
9. Bolodurina, I., Parfenov D. The optimization of traffic management for cloud application and services in the virtual data center // Parallel Computing Technologies (PaCT) : proceedings of 14th International Conference, 4-8 sept. 2017, Nizhny Novgorod, Russia. –2017. – С. 418–426.

СИСТЕМА ОПЕРАТИВНОЙ ИНДИКАЦИИ ПАРАМЕТРОВ ТЕХНОЛОГИЧЕСКОГО ПРОЦЕССА

Д.А. Журман, А.С. Фадеев
(г. Томск, Томский политехнический университет)
e-mail: daz18@tpu.ru

SYSTEM OF INDICATION OF PARAMETERS OF TECHNOLOGICAL PROCESS

D.A. Zhurman, A.S. Fadeev
(Tomsk, Tomsk Polytechnic University)

Abstract. The aim of the article is to create a system for indicating the operational values of the technological process parameters on mobile devices using the Raspberry Pi microcomputer.

The text gives valuable information about industrial protocols such as Modbus and Ethernet. It gives a detailed analysis of Python Modbus libraries. The article provide review several ready-made solutions for creating a system for the operational display of process parameters on mobile devices, namely: Master-SCADA 4D of the INSAT company, SCADA SimpLight by Simp Lite and development of Siberia Electrical Systems.

As a result, the following was developed: a server for transferring process parameters, a mobile client, a parameter caching function for the server, and a request structure.

Keywords: Raspberry Pi, Siemens S7-200, Modbus, TCP, Kivy, PLC, server, client, request, parameter.

Введение. Система оперативной индикации параметров технологического процесса является неотъемлемой составляющей любого производства, так как благодаря данной системе можно выявить отклонение процесса от заданных пределов, отказы технологического оборудования, и получить информацию о технологическом процессе, также произвести сигнализацию об аварийной ситуации.

На многих производствах даже незначительная задержка в выявлении отказа технологического оборудования может привести к значительным экономическим потерям, либо к чрезвычайной ситуации. Для того, чтобы оперативно предупредить рабочий персонал об отклонении процесса от заданных пределов и аварийной ситуации, индикацию параметров можно осуществлять на смартфоне или смарт часах.

К сожалению, готовые решения до сих пор стоят достаточно дорого. Также существующие системы привязаны к конкретному оборудованию, а иногда входят в состав SCADA, поэтому внедрение их на предприятие требует полного переоборудования производства или переустановки SCADA.

Целью данной работы является создание системы индикации оперативных значений параметров технологического процесса на мобильных устройствах при помощи микрокомпьютера Raspberry Pi. Индикация системы должна осуществляться на дисплее, подключенном к микрокомпьютеру, и на смартфоне или смарт часах под управлением операционной системы Android. Подключение к промышленному контроллеру может производиться с помощью основных промышленных протоколов таких как Modbus или Ethernet. Данная система не должна быть привязана к конкретному оборудованию, должна иметь возможность работать параллельно с уже установленной SCADA на предприятии, а также обладать низкой стоимостью. Предприятие, которое планирует использовать разрабатываемую систему, не должно проводить никакого переоборудования для ее установки.

Для реализации цели работы были поставлены следующие задачи:

1. Настройка точки доступа Wi-Fi на микрокомпьютере Raspberry Pi.
2. Подключение к промышленному логическому контроллеру.
3. Создание сервера, который обеспечивает передачу параметров технологического процесса.
4. Создание клиентского приложения.

Обзор решений, представленных на рынке. На данный момент уже существует несколько готовых решений по созданию системы оперативной индикации параметров технологического процесса на мобильные устройства, а именно: MasterSCADA 4D компании «ИнСАТ», SCADA SimpLight фирмы ООО «Симп Лайт» и разработка фирмы ООО "Электротехнические системы Сибирь". Однако каждое решение обладает своими недостатками.

MasterSCADA 4D – это российская вертикально-интегрированная SCADA-система с многоуровневой клиент-серверной архитектурой [1].

К недостаткам данного решения для индикации параметров технологического процесса можно отнести:

1. Высокая стоимость. В зависимости от используемого сервера MasterSCADA 4D стоимость клиентского приложения может составлять от 11900 рублей до 10752000 рублей [1].
2. В случае использования облачного сервера для индикации параметров необходимо обеспечивать дополнительные средства кибербезопасности для того, чтобы обеспечить сохранность данных.

SCADA SimpLight представляет собой программное обеспечение для построения систем управления и диспетчеризации различных автоматизируемых объектов [2].

К недостаткам данного решения для индикации параметров технологического процесса можно отнести:

1. Высокая стоимость. В зависимости от количества тегов, количества мобильных клиентов, драйверов и дополнительных опций стоимость SCADA SimpLight может составлять от 10000 рублей до 600000 рублей [2].
2. Ограниченность поддерживаемого оборудования. Данная SCADA поддерживает оборудование только определенных производителей, указанных в документации.
3. Связь с контроллером осуществляется только через MODBUS или OPC. В системе отсутствует поддержка таких распространенных протоколов как Ethernet или Profibus.

Разработка фирмы ООО "Электротехнические системы Сибирь" представляет собой техническое решение по мониторингу и управлению системами АСУ ТП, выполненными на базе оборудования MITSUBISHI ELECTRIC [3].

К недостаткам данного решения для индикации параметров технологического процесса можно отнести:

1. Связь с контроллером только через протокол Ethernet. В системе отсутствует поддержка таких распространенных протоколов как MODBUS или Profibus.

2. Ограниченность поддерживаемого оборудования. Данная SCADA поддерживает оборудование только определенных производителей, а именно MITSUBISHI ELECTRIC.

Из вышеприведенного обзора существующих технологий и решений следует, что основными недостатками являются:

1. Высокая стоимость.
2. Ограниченность поддерживаемого оборудования.
3. В случае, если на предприятии уже установлены контроллеры других производителей то, для использования данного решения необходимо переоборудование предприятия, что влечет за собой остановку процесса и финансовые и материальные потери, связанные с этим.

Проектирование системы оперативной индикации параметров технологического процесса. На рисунке 1 представлена архитектура разрабатываемой системы, которая позволяет устранить недостатки существующих систем.



Рисунок 1 – Архитектура разрабатываемой системы

В работе для проверки работоспособности системы оперативной индикации параметров технологических процессов используется ПЛК Siemens Simatic S7-200.

Для того, чтобы передавать параметры технологического процесса необходимо разработать сервер, который будет собирать параметры с контроллера и передавать на мобильные устройства. В качестве устройства, на котором будет работать сервер, был выбран микрокомпьютер Raspberry Pi под управлением операционной системы Raspbian. В сравнении с конкурентами компьютерами Raspberry Pi имеет неплохую производительность и низкую стоимость, а также очень примечательна его стабильная работа и надежность. Кроме того, он имеет очень низкую энергопотребляемость, поэтому данный микрокомпьютер отлично подходит для передачи технологических параметров [4].

Для передачи данных технологических параметров мобильным устройствам использовался протокол TCP (Transmission Control Protocol), а для создания точки доступа Wi-Fi применялся протокол DHCP (протокол динамической настройки узла).

Raspberry Pi имеет возможность подключения к контроллерам по протоколам Modbus, благодаря открытым библиотекам.

Подключение к ПЛК по протоколу Modbus может осуществляться через один из четырех входов USB или интерфейсы GPIO посредством специальных адаптеров и переходников.

Несмотря на то, что Raspberry Pi поддерживает и другие языки программирования, управление портами GPIO, через которые может осуществляться подключение к ПЛК, осуществляется только с помощью языка Python, который использовался в данной работе.

Так как язык Python не содержит встроенных средств для создания приложений для операционной системы Android, для создания интерфейса программы отображения парамет-

ров технологического процесса и создания приложения для операционной системы Android используется библиотека Kivy.

Несмотря на все свои достоинства у Python отсутствует возможность создания исполняемых файлов и пакетов установки написанных программ. Для решения данной проблемы в работе используется специальная утилита – Buildozer.

Благодаря тому, что в разрабатываемой системе в качестве сервера передачи данных используется одноплатный микрокомпьютер Raspberry Pi обеспечивается низкая энергопотребляемость. Так как операционная система микрокомпьютера основана на Debian, она является более безопасной по отношению к Windows из-за меньшего количества вредоносных программ ввиду меньшей популярности. Также данный микрокомпьютер обеспечивает низкую стоимость относительно специальных серверных компьютеров.

Использование промышленных протоколов Modbus и Ethernet обеспечивает совместимость с различными типами промышленного оборудования.

Так как для разработки приложения для оперативной индикации параметров технологического процесса используется библиотека Kivy, обеспечивается совместимость с основными операционными системами, что позволяет запускать приложение как на мобильных устройствах, так и на микрокомпьютере.

Разработка системы оперативной индикации параметров технологического процесса. Raspberry Pi можно использовать в качестве беспроводной точки доступа с автономной сетью. Raspberry Pi 3 имеет встроенный модуль Wi-Fi, поэтому отсутствует необходимость в подключении дополнительных устройств.

Чтобы создать точку доступа на Raspberry Pi, было установлено программное обеспечение точки доступа. Так как автономная сеть настраивается в качестве сервера, Raspberry Pi был назначен статический IP-адрес и интерфейс беспроводного порта. Чтобы настроить статический IP-адрес, был отредактирован файл конфигурации dhcpd.conf и установлены следующие параметры:

1. interface = wlan0;
2. static ip_address=192.168.4.1.

Затем было установлено программное обеспечение DHCP-сервера для назначения подключаемым устройствам сетевого адреса [5]. После этого был отредактирован файл конфигурации dnsmasq.conf и установлены следующие параметры:

1. interface = wlan0;
2. dhcp-range=192.168.4.2,192.168.4.50;

Затем был отредактирован файл hostapd.conf и установлены следующие параметры: используемый интерфейс, драйвер сети, название сети и пароль.

После запуска утилит hostapd и dnsmasq настроенная Wi-Fi сеть стала доступна для подключения при введении указанного пароля.

Используемый в данной работе контроллере Siemens S7-200 оснащен двумя портами RS485, однако только порт 0 может осуществлять передачу данных по протоколу Modbus.

Для подключения к ПЛК был использован адаптер USB-RS485 на основе микросхемы MAX485, которому для работы достаточно 5 В. Адаптер имеет 2 контакта для передачи данных (А и В), которые необходимо было подключить к контактам порта 0. Питание адаптера осуществляется от USB порта микрокомпьютера Raspberry Pi [6].

В разрабатываемой системе Siemens S7-200 выступает в качестве ведомого устройства (Slave), а микрокомпьютер Raspberry Pi в качестве ведущего (Master). Такой выбор режимов работы устройств обусловлен тем, что Modbus Master способен получать данные одновременно с нескольких контроллеров, работающих в режиме Modbus Slave. Это позволяет подключать одновременно несколько ПЛК к микрокомпьютеру и считывать с них параметры [6].

Для активации протокола Modbus и разрешения считывания параметров в контроллер была загружена программа, реализованная на языке LD, в которой были установлены следующие параметры соединения:

1. Скорость передачи 9600 Бод.
2. Установлена проверка на четность.
3. Задержка отсутствует.
4. Разрешено обращение ко всем цифровым входам и выходам S7-200.
5. Разрешено обращение ко всем аналоговым входам S7-200.
6. Разрешен доступ к 1000 регистрам временного хранения информации.

Ключевым параметром является проверка на четность, поскольку при ее отключении или включении проверки на нечетность, прочитать параметры с ПЛК не удалось. Остальные параметры могут принимать другие значения в зависимости от поставленной задачи.

Считывать значения параметров с контроллера Siemens S7-200 можно через специальное программное обеспечение STEP 7-Micro/WIN, однако данное ПО разработано только для операционных систем Windows. Поэтому после того, как программа была загружена в ПЛК, было необходимо разработать собственную программу для работы по протоколу Modbus для Raspberry Pi.

Основными библиотеками Python для использования протокола Modbus являются PyModbus, Modbus-TK, MinimalModbus.

После успешного подключения к контроллеру было произведено сравнение библиотек PyModbus, Modbus-TK, MinimalModbus.

Библиотека PyModbus не обеспечивала стабильного подключения к контроллеру, так как некоторые запросы, посылаемые данной библиотекой, не возвращали никакого результата.

Библиотека MinimalModbus смогла обеспечить стабильное подключение к контроллеру, но позволяет считывать не более 40 параметров в секунду.

Библиотека Modbus-TK смогла обеспечить стабильное подключение к контроллеру, и позволила считывать не более 480 параметров в секунду. Однако контроллер Siemens S7-200 способен обработать не более 75 параметров в 1 запросе на чтение.

Таким образом, Modbus-TK способна считывать наибольшее количество параметров за 1 запрос и обеспечивает стабильное подключение к ПЛК, поэтому в дальнейшей работе было принято решение для создания сервера использовать данную библиотеку.

Создание сервера передачи параметров технологического процесса. Одной из главных частей системы оперативной индикации параметров технологического процесса является сервер, основной функцией которого является передача параметров технологического процесса, полученных с ПЛК, мобильному устройству. Данный сервер был создан на языке программирования Python с использованием открытой библиотеки socketserver, которая позволяет осуществлять передачу данных по протоколу TCP.

При одновременном подключении нескольких клиентов, формировалось несколько одновременных запросов на чтение параметров с контроллера, что приводило к ошибке, так как контроллер не может обрабатывать несколько запросов одновременно. Для решения проблемы были использованы потоки, работающие в асинхронном режиме.

Поток – это виртуальный процессор, имеющий свой собственный набор регистров и стек, аналогичных регистрам настоящего центрального процессора, при этом другие потоки также могут их использовать [7].

При подключении нового клиента создается новый поток, который обрабатывает запрос на чтение параметров только этого клиента независимо от остальных потоков. При отключении клиента поток, отвечающий за обработку его запросов, уничтожается. Благодаря тому, что потоки работают в асинхронном режиме, запросы на чтение формируются в разное время, что позволяет избежать ошибок, связанных с одновременной обработкой нескольких запросов контроллером.

В случае, если количество параметров больше 75, то для их получения, сервер формирует несколько запросов к контроллеру. Затем полученные с контроллера параметры, принадлежащие одному запросу клиента, объединяется и отправляются на мобильное устройство.

В ходе выполнения работы было обнаружено, что, если нескольким клиентам требуется получить одни и те же параметры технологического процесса, сервер и контроллер формировали и обрабатывали несколько одинаковых запросов. С учетом того, что контроллер не может обрабатывать более 4 запросов в секунду, это приводило к существенному снижению скорости обработки. Для решения данной проблемы в сервере реализована возможность кэширования параметров технологического процесса.

Функция кэширования работает следующим образом: при получении сервером запроса от клиента, сервер проверяет наличие данного запроса в списке всех запросов, полученных сервером за последнюю секунду. Если запрос не обнаружен, то сервер формирует новый запрос на чтение параметров с контроллера и сохраняет запрос клиента и параметры, удовлетворяющие данному запросу, в списки. В случае если запрос найден в списке всех запросов, полученных сервером за последнюю секунду, то клиенту отправляются параметры, сохраненные при обработке этого запроса. Сохраненные запросы и параметры удаляются по истечении 1 секунды. Применение возможности кэширования позволило увеличить скорость работы системы в 4 раза.

Создание клиентского приложения для операционной системы Android. После того, как был написан программный код сервера, необходимо было написать программу, которая бы получала и считывала полученные значения, отправленные с сервера.

В мобильном приложении были реализованы следующие функции:

1. Окно подключения, где пользователю необходимо указать IP адрес и порт сервера.
2. Окно индикации параметров технологического процесса, где отображаются параметры выбранной группы. Значения параметров обновляются автоматически.
3. Нажатие на значение параметра вызывает окно настройки сигнализации. В данном окне пользователь может задать верхнюю и нижнюю границу параметра. Если значение параметра выйдет за указанные пределы, то возникнет окно сигнализации, а после закрытия окна, напротив значения будет отображаться стрелка, указывающая на превышение или понижения значения.
4. Окно настройки, где пользователь может задать количество параметров каждого типа и начальный адрес.

После написания программного кода необходимо было создать пакет приложения для операционной системы Android. Для этого была использована специальная утилита Buildozer. После запуска программы был сформирован файл конфигурации, где необходимо было указать название программы, режим работы, режим отображения и требования к устройству.

После компиляции приложения был получен установочный файл арк, который был загружен и установлен на смартфон. После установки программы на устройство под управлением системы Android и ее запуска было осуществлено подключение к серверу и успешно получены параметры технологического процесса.

Вывод. Таким образом, разработанная система была успешно протестирована на контроллере Siemens S7-200 с передачей параметров на смартфон, в ее работе никаких сбоев и ошибок не наблюдалось.

Благодаря тому, что в данной системе используется протокол Modbus, ее применение не ограничивается контроллером Siemens S7-200. Она может быть применена для чтения параметров с любых контроллеров, которые поддерживают протокол Modbus.

Разработанная система может быть применена на различных предприятиях, где используется АСУ ТП, для того, чтобы оперативно оповещать рабочий персонал об отклонении процесса от заданных пределов и аварийной ситуации.

ЛИТЕРАТУРА

1. Официальный сайт ИнСАТ [Электронный ресурс] / MasterSCADA 4D // URL: <https://insat.ru/products/?category=1536>. – дата обращения 01.05.2019.
2. Официальный сайт SimpLight [Электронный ресурс] / SIMP HMI // URL: <https://simplight.ru/mobile/#more>. – дата обращения 01.05.2019.
3. Электротехнические системы Сибирь [Электронный ресурс] / Мониторинг и управление системами АСУ ТП на мобильных платформах // URL: http://es-electro.ru/novosti/monitoring_i_upravlenie__sistemami_asu_tp_na_mobilnyh_platformah_s_os_android/. – дата обращения 01.05.2019.
4. Портал о микроконтроллере Raspberry Pi [Электронный ресурс] / Аналоги одно-платного мини-ПК Raspberry Pi // URL: <https://myraspberrypi.ru/analogi-raspberrypi.html>. – дата обращения 01.05.2019.
5. Официальная документация к Raspberry Pi [Электронный ресурс] / Access point // URL: <https://www.raspberrypi.org/documentation/configuration/wireless/access-point.md>. – дата обращения 01.05.2019.
6. Официальный сайт Siemens [Электронный ресурс] / документация Modbus // URL: https://www.siemens-ru.com/doc/12_Modbus_r.pdf. – дата обращения 01.05.2019.
7. Сайт Microsoft [Электронный ресурс] / Поток и работа с ними // URL: <https://docs.microsoft.com/ru-ru/dotnet/standard/threading/threads-and-threading>. – дата обращения 01.05.2019.

АНАЛИЗ ГОРОДСКОЙ ТРАНСПОРТНОЙ СЕТИ НА ОСНОВЕ ГЕОИНФОРМАЦИОННЫХ СИСТЕМ В ТРАНСПОРТНЫХ ЗОНАХ ПРОМЫШЛЕННОГО ГОРОДА

О.А. Лебедева

*(г. Ангарск, ФГБОУ ВО «Ангарский государственный технический университет»)
kravhome@mail.ru*

ANALYSIS OF THE CITY TRANSPORT NETWORK BASED ON GEOINFORMATION SYSTEMS IN THE TRANSPORT ZONES INDUSTRIAL CITY

O.A. Lebedeva

(Angarsk, Angarsk State Technical University)

Abstract. Analysis of transport accessibility is a standard task requiring the search for the optimal solution, the solution of which is necessary for the harmonious development of the urban environment in general, and the effective organization of the transportation process in particular. The article discusses the indicators necessary for determining transport accessibility based on an analysis of the urban transport system and calculation methods using geographical information systems. The main parameter is time-based accessibility from the central part of the city to other transport zones. To determine the rank of the transport zone, the following statistical characteristics are required: the density of the street network in urban areas; number of jobs, number of equipped parking spaces; the number of objects of gravity in transport zones.

A decision support system based on geographic information systems can be carried out by two methods: the choice of a relatively ideal solution and simple additive weighing. Analysis of transport zones can improve the quality of statistical data on urban transport environment. The article presents the models, the use of which is necessary for the efficient operation of urban transport, as well as indicators necessary for the stable functioning of the infrastructure. It can be integrated into planning systems and sustainable development of urban transport and planning.

Keywords: geoinformation is, city transport, transport zoning, transport planning.

Оценка эффективности работы транспортной системы одна из важнейших задач транспортного моделирования. Для решения этой задачи необходимо выбрать пару пунктов