

УДК 004.738.5

ИНТЕРНЕТ ВЕЩЕЙ: ОРГАНИЗАЦИЯ АВТОМАТИЗИРОВАННОГО ПОЛИВА

Правосудов А.Р., Ямашкин С.А.

ФГБОУ ВО «Национальный исследовательский Мордовский государственный университет им. Н.П. Огарева», Саранск, e-mail: pravosudov_2015@mail.ru, yamashkinsa@mail.ru

В статье рассматриваются аспекты разработки и внедрения технологии интернета вещей для решения задач управления техническими и организационными системами, в частности, в области растениеводства. Предлагается технологическое решение реализации системы автоматического полива растений. Поэтапно рассмотрен процесс создания устройства интернета вещей для полива растений и веб-интерфейса для управления этим устройством. Исследованы средства взаимодействия устройств интернета вещей и принципы их работы. Подобраны необходимые аппаратные комплектующие для создания устройства. Спроектировано и изготовлено устройство, позволяющее осуществлять полив как в ручном, так и в полностью автоматическом режиме на основании данных о влажности почвы. Разработан веб-интерфейс в соответствии с современными стандартами веб-разработки, через который пользователь может удаленно управлять устройством. Для обеспечения возможности управления устройством из любой точки земного шара данный интерфейс был загружен на хостинг. В результате работы получен аппаратно-программный комплекс, реализующий возможность удаленного управления автоматизированным поливом растений. Полученный программно-аппаратный комплекс фактически является прототипом промышленного образца для реализации на открытом рынке устройств интернета вещей. Целевой аудиторией данного проекта являются люди, желающие по тем или иным причинам добиться автономного полива растений.

Ключевые слова: интернет вещей, автоматический полив, аппаратно-программный комплекс, растениеводство, удаленное управление

INTERNET OF THINGS: ORGANIZATION OF AUTOMATED IRRIGATION

Pravosudov A.R., Yamashkin S.A.

National Research Mordovia State University named after N.P. Ogarev, Saransk, e-mail: pravosudov_2015@mail.ru, yamashkinsa@mail.ru

The article discusses aspects of the development and implementation of the Internet of Things technology for solving problems of managing technical and organizational systems, in particular, in the field of crop production. A technological solution for the implementation of an automatic plant watering system is proposed. The process of creating an Internet of Things device for watering plants and a web interface for controlling this device is considered in stages. The means of interaction of Internet of Things devices and the principles of their operation are investigated. The necessary hardware components have been selected to create the device. A device has been designed and manufactured that allows watering, both manually and in fully automatic mode, based on data on soil moisture. A web interface has been developed, in accordance with modern web development standards, through which the user can remotely control the device. In order to be able to control the device from anywhere in the world, this interface has been uploaded to the hosting. As a result of the work, a hardware and software complex was obtained that implements the possibility of remote control of automated watering of plants. The resulting hardware and software complex is actually a prototype of an industrial design for the implementation of Internet of Things devices on the open market. The target audience of this project are people who want to achieve autonomous watering of plants due to various reasons.

Keywords: internet of things, automatic irrigation, hardware and software system, crop production, remote control

На сегодняшний день развитие информационного общества достигло той точки, когда количество устройств, подключенных к интернету, в несколько раз превышает общее число людей на планете. Возникла новая концепция развития технологий сети Интернет – интернет вещей. Интернет вещей (Internet of Things, IoT) – это множество физических объектов, подключенных к интернету и обменивающихся данными [1]. Концепция интернета вещей имеет огромное множество потенциальных областей применения в области управления техническими и организационными системами: «умные» бытовые приборы, идентификация и отслеживание товаров, «умные» здания, охрана здоровья, мониторинг окружающей сре-

ды, сбор информации и управление потреблением энергии и многое-многое другое. В определенном смысле особняком стоит промышленное применение, обозначаемое термином «Промышленный интернет вещей» (Industrial Internet of Things – IIoT). Это понятие близко связано с концепцией Индустрии 4.0, являясь ее составной частью [2].

Интернет вещей может способствовать мощному прорыву в разрезе сельского хозяйства, как в целом, так и, в частности, для растениеводства. Вот несколько примеров применения концепции интернета вещей для растениеводства:

– создание закрытых садовых систем с обеспечением необходимых условий для развития растений;

– съемка земель с помощью беспилотных летательных аппаратов и анализ полученных данных;

– умные системы полива и удобрения для повышения урожайности, организации эффективного водопотребления и умного освещения в теплицах.

Благодаря использованию умных систем можно добиться повышения общей урожайности и эффективности при выращивании растений.

Помимо промышленных в настоящее время в Европе набирают популярность домашние садовые системы и гроубоксы (от англ. growbox – ящик для выращивания). В данных системах создаются благоприятные условия для выращивания растений, и люди могут получать свежие овощи и фрукты у себя дома круглый год. Как правило, данные комплексы включают некий набор датчиков и актуаторов, обеспечивающих своевременный полив и освещение растений [3].

Цель исследования – разработка прототипа аппаратно-программного комплекса для автоматического полива растений на базе технологий интернета вещей как примера решения задач управления техническими и организационными системами.

Для достижения поставленной цели были последовательно решены следующие задачи:

1. Проектирование и создание устройства, способного осуществлять удаленный сбор сведений о влажности почвы и полив растений.

2. Проектирование и реализация веб-интерфейса для удаленного доступа и управления функциями.

Материалы и методы исследования

На этапе проектирования в качестве средства взаимодействия устройства и веб-

сайта было принято решение использовать протокол передачи данных MQTT. MQTT – это стандартный OASIS протокол обмена сообщениями для интернета вещей. Он разработан как крайне легковесный способ обмена сообщениями для публикации/подписки, который идеально подходит для подключения удаленных устройств с небольшим объемом кода и минимальной пропускной способностью сети. MQTT сегодня используется в самых разных отраслях промышленности, таких как автомобилестроение, производство, телекоммуникации, нефть и газ и т.д. [4].

Все устройства, использующие протокол MQTT, называются mqtt-клиентами и работают по принципу издатель/подписчик. Устройства могут одновременно являться как издателями, которые публикуют сообщения брокеру, так и подписчиками, которые принимают сообщения от брокера. Брокер – это программа на сервере или сам сервер, который перенаправляет полученные сообщения от издателей к подписчикам [5]. В разрабатываемой системе актором (mqtt-клиентом) может являться как человек, управляющий устройством через веб-интерфейс, так и само устройство для полива.

Общая схема взаимодействия всех компонентов разрабатываемой системы представлена на рис. 1.

Для реализации проекта были использованы следующие технологии: в качестве языка программирования для аппаратной платформы был выбран язык C++ с фреймворком Wiring (Arduino Wiring). В основе веб-интерфейса лежат стандартные технологии: HTML, CSS и JavaScript. Общение веб-интерфейса с базой данных на сервере осуществляется с помощью языка PHP.

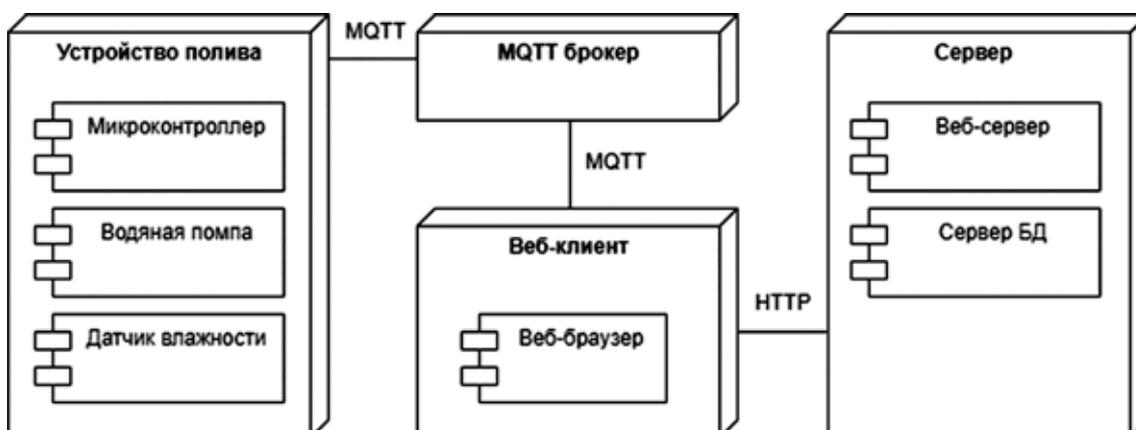


Рис. 1. Диаграмма развертывания разрабатываемого аппаратно-программного комплекса

Проектирование и создание прототипа устройства для полива

Согласно поставленным задачам разрабатываемое устройство должно обеспечивать следующие функции:

- включение/выключение подачи воды по команде;
- постоянная фиксация текущих значений влажности почвы;
- возможность включения автоматического «умного» полива на основании данных влажности.

На основании выделенных функций, для идентификации сущностей, составим топики (текстовые идентификаторы), на которые по протоколу MQTT будут поступать значения. Для распознавания конкретного устройства будем использовать его mac-адрес. Далее представлены топики:

- mac/pump – для управления насосом;
- mac/soilHum – для получения значения влажности почвы;
- mac/soilHum/threshold – для изменения порога влажности;
- mac/autoMode – для управления автоматическим режимом.

В самом устройстве на месте «mac» стоит его реальный mac-адрес, который определяется программно.

Так как на многих современных предприятиях и в подавляющем большинстве домов и офисов действуют сети Wi-Fi, для выхода в интернет был выбран этот способ. Поэтому в качестве мозга устройства был выбран микроконтроллер ESP8266, благодаря его встроенной функции поддержки Wi-Fi [6].

Для упрощения задачи изготовления прототипа была использована отладочная плата NodeMCU с чипом ESP8266. Помимо платы с контроллером необходимы следующие компоненты:

- водяная помпа (в нашем случае шестеренчатая помпа на базе мотора rs360sh);
- датчик влажности почвы FC-28 (YL-38);
- понижающий преобразователь напряжения на базе LM2596 0-30V;
- 12-вольтовый источник питания;
- транзистор 2N2222 (NPN);
- металлопленочный резистор на 5,1 кОм;
- одноканальный переключающий релейный модуль с управлением 5 вольт (например GSMIN HW-482 для среды Arduino).

В дополнение к перечисленному для сборки всей схемы понадобилась одна макетная плата MB102 с модулем питания и проводами. Схематическое изображение готового прототипа представлено на рис. 2.

Так как наш модуль реле управляется с помощью 5В, мы можем оба вывода реле подключить к шине, на которую модуль питания подает +5В. А транзистор при открытии будет подводить землю к минусовому контакту реле, тем самым открывая его.

Проектирование и реализация веб-интерфейса

Для ускорения процесса верстки и определения конечного результата был необходим макет реализуемого веб-интерфейса с желаемым UX/UI дизайном. Данный макет создан в графическом редакторе Figma. Затем в соответствии с макетом производилась верстка сайта.

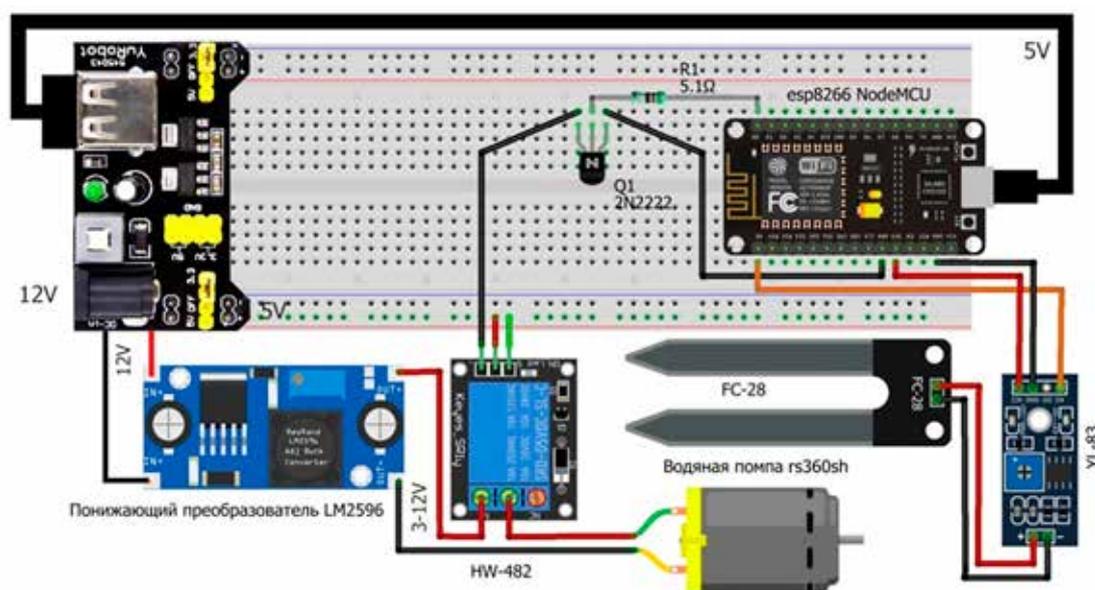


Рис. 2. Схематическое изображение прототипа

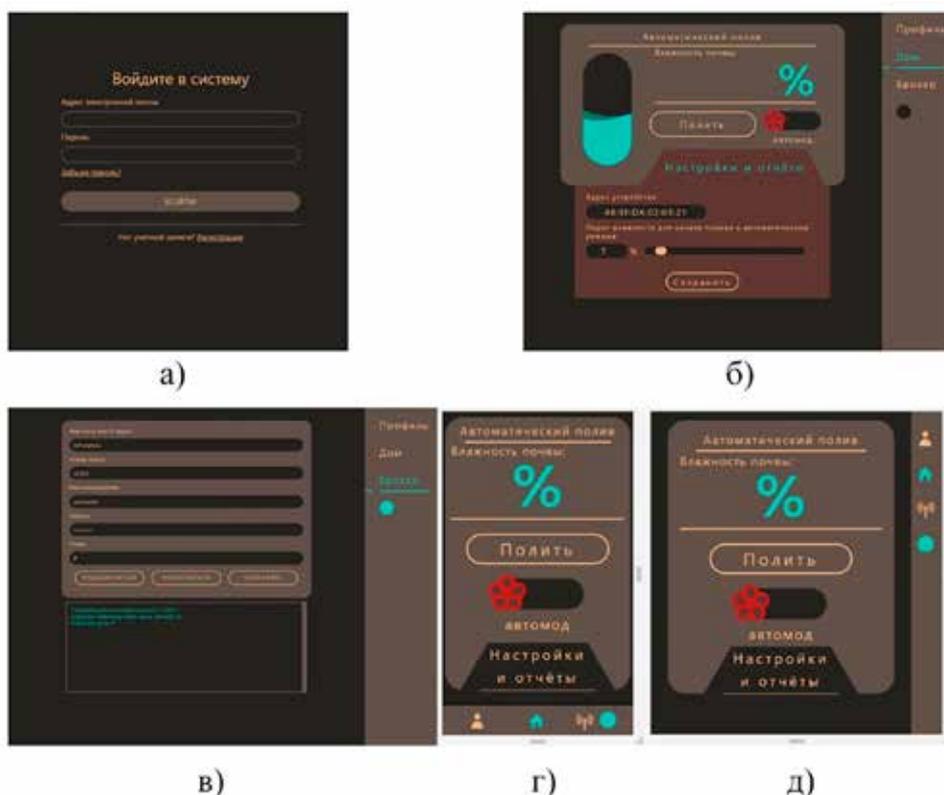


Рис. 3. Сверстанный веб-интерфейс: а) форма авторизации; б) вкладка с виджетом при ширине экрана более 1200px; в) вкладка с брокером при ширине экрана более 1200px; г) вкладка с виджетом при ширине экрана 320px; д) вкладка с виджетом при ширине экрана 490px

Чтобы убрать различия в базовых стилях и сделать отображение страницы во всех браузерах одинаковым, использован файл `Normalize.css`.

Сегодня больше половины всего интернет-трафика относится к сеансам со смартфонов. Доступ к смартфону есть практически в любой ситуации, в отличие от стационарного компьютера или ноутбука. Доступность смартфона довольно удобно вписывается в концепцию удаленного управления устройствами. Поэтому было принято решение сделать веб-интерфейс адаптивным под различные устройства.

Чтобы сделать интерфейс *адаптивным*, были использованы медиазапросы. Медиазапросы были описаны под стандартные размеры ширины экрана (1200px, 1024px, 768px, 480px). Помимо этого в процессе верстки был сделан упор на использование относительных единиц измерения.

В процессе разметки были использованы семантические теги в соответствии со стандартом HTML5, стилизация выполнена на чистом CSS. Основные страницы полностью сверстанного веб-интерфейса представлены на рис. 3.

При ширине экрана меньше 768px было принято решение текстовое описание вкладок меню заменить иконками (рис. 1, г, д). Для этих целей с помощью специальных онлайн-сервисов был создан собственный иконочный шрифт, содержащий необходимый набор иконок. Взаимодействие веб-интерфейса с MQTT брокером реализовано на языке JavaScript, с использованием библиотеки `Paho MQTT` для JS. Проект `Paho` был создан для обеспечения надежных реализаций открытых и стандартных протоколов обмена сообщениями с открытым исходным кодом, предназначенных для новых, существующих и появляющихся приложений для межмашинных коммуникаций (M2M) и интернета вещей [7]. Первоначально библиотека была создана Эндрю Бэнксом из IBM и передана в дар Eclipse компанией IBM в 2013 г. Библиотека считается очень стабильной и используется во многих веб-приложениях на базе MQTT [8].

Для защиты от несогласованного управления устройством посторонними лицами и возможности использования нескольких устройств было необходимо создать базу данных пользователей. Так как на большин-

стве хостингов используются базы данных MySQL, в качестве базы данных была использована именно она. Взаимодействие сайта с базой данных осуществляется с помощью языка программирования PHP. Этот язык был выбран благодаря своей широкой распространенности в данной сфере и обилию примеров.

На PHP были созданы обработчики форм авторизации и регистрации. При успешной авторизации идентификатор пользователя сохраняется в файлах cookie. Помимо этого реализован скрипт выхода из системы, скрипт получения всех данных пользователя по его идентификатору из таблицы баз данных. Реализовано сохранение данных пользователя и настроек виджета в базу данных по нажатию кнопки. А также сохранение положения переключателя автоматического режима, т.е. статуса автоматического режима, при изменении статуса с помощью ajax запросов.

Для управления нашим устройством из любой точки мира, необходимо разместить сайт в глобальной сети Интернет. В качестве хостинг-провайдера был выбран beget. Далее приобретен домен, а также к домену был подключен бесплатный SSL сертификат Let's Encrypt. Затем домен был прикручен к встроенному тестовому сайту, на который впоследствии были загружены все файлы и данные разработанного веб-интерфейса.

Стоит заметить, при загрузке на хостинг в php файлах, где используется `include_once`, необходимо прописывать путь с использованием глобального массива `$SERVER`. Например: `include_once $SERVER['DOCUMENT_ROOT'] . '/template/auth_template.php'`. После проделанных манипуляций веб-интерфейс стал доступен по приобретенному домену (на момент написания работы, веб-интерфейс доступен по домену `aka-poseidon.ru`).

Заключение

В процессе реализации проекта был исследован ряд областей и аспектов разработки устройств и веб-сайтов. Проведен анализ сферы интернета вещей, в процессе которого было получено подтверждение актуальности выбранного направления разработки. Исследованы возможные перспективы применения таких технологий в домашнем и промышленном растениеводстве. Представленное решение ориентировано на задачи управления техническими и организационными системами на основе технологий интернета вещей.

Итогом проекта является спроектированное с выявлением необходимых аппаратных

модулей и изготовленное устройство, способное осуществлять удаленный сбор сведений о влажности почвы и полив. Согласно выявленным требованиям к устройству, в среде разработки Arduino IDE был написан необходимый для работы программный код, дающий возможность взаимодействовать устройству с брокером. Предусмотрена настройка параметров порога влажности для регулирования автоматического полива.

Создан веб-интерфейс отвечающий всем современным стандартам веб-разработки. Выполнено написание скриптов, обеспечивающих корректное функционирование сайта, и скриптов, обеспечивающих межмашинную связь. Веб-интерфейс был размещен на публичном хостинге и стал доступен из глобальной сети Интернет.

Результатом работы является разработанная система, представляющая собой программно-аппаратный комплекс – готовое устройство и веб-интерфейс. Полученный программно-аппаратный комплекс фактически является прототипом промышленного образца для реализации на открытом рынке устройств интернета вещей. При том, что система разрабатывалась для решения конкретной проектно-ориентированной задачи автоматизированного полива растений, в ней реализованы качественные требования расширяемости и повторного использования, что делает возможным использование разработки в последующих проектах.

Список литературы

1. Что такое IoT и что о нем следует знать // Хабр. [Электронный ресурс]. URL: <https://habr.com/ru/company/otus/blog/549550/> (дата обращения: 23.11.2022).
2. Ковалевский Ю. Интернет вещей: от концепции к реальности // Электроника: НТБ. 2017. № 1. С. 135–138.
3. Горностаев М.Н., Лунина А.Е., Тукмачев Д.В., Чукай А.А., Мороз А.А. История появления грубобоксов и возможности их использования в домашних условиях // Научная сессия ТУСУР–2020: материалы Международной научно-технической конференции студентов, аспирантов и молодых ученых (Томск, 13–30 мая 2020 г.). В 3 ч. Томск: В-Спектр, 2020. Ч. 3. С. 202–204.
4. MQTT – The Standard for IoT Messaging. [Электронный ресурс]. URL: <https://mqtt.org/> (дата обращения: 01.12.2022).
5. Салкин Д.А., Душутин С.С. Устройство дистанционного контроля и управления для систем домашней автоматизации // Современные наукоемкие технологии. 2018. № 10. С. 113–119.
6. kolban-ESP32. [Электронный ресурс]. URL: <https://www.robotlinkmarket.com/Data/EditorFiles/datasheet/kolban-ESP32.pdf> (дата обращения: 01.12.2022).
7. Eclipse Paho | The Eclipse Foundation. [Электронный ресурс]. URL: <https://www.eclipse.org/paho/> (дата обращения: 01.12.2022).
8. Sutton J. Paho JavaScript – MQTT Client Library Encyclopedia. [Электронный ресурс]. URL: <https://www.hivemq.com/blog/mqtt-client-library-encyclopedia-paho-js/> (дата обращения: 18.12.2022).