УДК 004.42

<https://doi.org/10.36906/AP-2020/38>

СИСТЕМА СБОРА ДАННЫХ НА БАЗЕ LABVIEW С ИСПОЛЬЗОВАНИЕМ ПРОТОКОЛА MQTT

Григорьев К. А.

*Иркутский национальный исследовательский
технический университет
г. Иркутск, Россия*

Ченский А. Г.

*канд. физ.-мат. наук
Иркутский национальный исследовательский
технический университет
г. Иркутск, Россия*

Аннотация: В статье рассматривается система дистанционного сбора данных с распределенной сети датчиков на основе оборудования компании National Instruments в рамках концепции Промышленного Интернета Вещей. Приведена структура системы и рассмотрены основные принципы взаимодействия ее частей.

Ключевые слова: LabVIEW, myRIO, MQTT, Промышленный Интернет Вещей

Система сбора данных (Data Acquisition System) — это система, осуществляющая сбор информации с датчиков, установленных на объекте исследования, предварительную обработку, накопление и передачу ее в компьютер.

В систему сбора данных также входят и управляющие средства: линии цифрового ввода-вывода, цифро-аналоговые преобразователи. То есть, система сбора данных включает в себя несколько программных и аппаратных уровней. Таким образом, проектирование и отладка современных систем сбора данных требует от разработчиков теоретических знаний и практических навыков в различных областях, таких, как аналоговая и цифровая электроника, низко- и высокоуровневое программирование, разработка пользовательского интерфейса и т.д.

Современная среда графического программирования существенно упрощает разработку и прототипирование систем обработки данных, автоматизации и управления. Одну из лидирующих позиций в этой области занимает платформа LabVIEW компании National Instruments. Благодаря своей гибкости и масштабируемости, LabVIEW может использоваться на всех этапах: от моделирования и разработки прототипов продуктов до широкомасштабных производственных испытаний. Последние версии LabVIEW позволяют создавать распределенные системы измерений [1].

Контроллер NI myRIO

Основной компонент системы сбора данных – контроллер NI myRIO-1900. Ядром myRIO является система на кристалле (SoC) Xilinx Zynq-7010, состоящая из двухъядерного микропроцессора ARM Cortex-A9 с тактовой частотой 667 МГц и программируемой логической интегральной схемы (ПЛИС) Artix-7 с 28000 ячеек (<https://clck.ru/T7vKd>).

Основные характеристики NI MyRIO:

-Xilinx Zynq-7010 с двухъядерным процессором Cortex-A9 и FPGA с 28000 ячеек;

-порт myRIO exPansion(MXP) содержит:

- 4 аналоговых входа; 6 цифровых входа/выхода; 2 аналоговых выхода;
- 3 выхода ШИМ (PWM);
- 1 UART;
- 1 I2C;
- 1 SPI;

-miniSystems Port (MSP) содержит:

питание;

- 2 аналоговых выхода; 4 аналоговых входа; 8 цифровых входа/выхода;

-Сеть – Wi-Fi;

-USB – 1 USB host port, 1 USB порт для устройств и подключения к PC;

-Аудио вход, Audio выход

-Пользовательские кнопки и кнопка сброса, диоды питания, статуса и Wi-Fi, 4 определяемые пользователем светодиода, акселерометр.

-Питание: (6...16)V.

Рис. 1. Структурная схема контроллера myRIO

Описание MQTT. Структура протокола, основные понятия

MQTT (Message Queue Telemetry Transport) — это легкий, компактный и открытый протокол обмена данными созданный для передачи данных на удаленных локациях, где требуется небольшой размер кода и есть ограничения по пропускной способности канала. Вышеперечисленные достоинства позволяют применять его в системах M2M (Машинно-Машинное взаимодействие) и IoT (Промышленный Интернет вещей).

Протокол MQTT работает на прикладном уровне поверх TCP/IP и использует по умолчанию 1883 порт (8883 при подключении через SSL).

Обмен сообщениями осуществляется между клиентом (client), который может быть издателем или подписчиком (publisher/subscriber) сообщений, и брокером (broker) сообщений (рис. 2). Издатель отправляет данные на MQTT брокер, указывая в сообщении определенную тему, топик (topic), где они будут храниться. Подписчик отправляет запрос на подключение к топику и получает данные от одного или множества издателей, в зависимости от подписки. Формат сообщений не ограничен.

Надежность MQTT обеспечивает качество обслуживания (QoS), которое выбирают клиенты:

0 QoS — publisher/subscriber доставит сообщение один раз, без подтверждения.

1 QoS — publisher/subscriber доставит сообщение хотя бы 1 раз с необходимым подтверждением.

2 QoS — publisher/subscriber доставит сообщение один раз с помощью четырехэтапного рукопожатия.

Также надежность обеспечивается функцией Wills. Сообщение клиента, которое необходимо отправить брокеру, при незапланированном отключении клиента, сообщение также будет иметь topic и QoS.

Рис. 2. Взаимодействие между подписчиком, издателем и брокером

В MQTT протоколе для защиты передачи данных существует аутентификация клиентов. При соединении клиента с брокером (подключение к брокеру производится через TLS/SSL) есть возможность установления USERNAME и PASSWORD. В системе также реализован ClientID (<https://clck.ru/T7v6r>). Но, несмотря на это сохраняются такие угрозы как утечка чувствительных данных, взлом конечных устройств, получение управления устройствами IoT и искажение поступающих данных.

Протокол хорошо разделяет и конкретизирует такие операции, как подписка и публикация информации в топики, но формат данных оставляет на усмотрение приложений. Из-за этого разные системы управления используют разные форматы данных и по факту несовместимы между собой.

Клиент MQTT на LabVIEW myRIO

Программа клиента выполняет функции по связи с сервером, опросу топиков, визуализации и сохранению полученных данных. Общая логика работы состоит из трех основных частей: настройка, подключение к серверу, работа с данными (рис. 3).

Пользователь взаимодействует с программой через Web-интерфейс или задает настройки через файл на USB носителе, который подключается к контроллеру myRIO. В настройки входит: IP адрес и порт сервера, параметры подключения к серверу, список топиков, идентификатор клиента.

Рис. 3. Общая структурная схема программы

Программа запускается кнопкой Start или автоматически, при наличии соответствующей настройки. После этого на сервер отправляется запрос на подключение и ожидается его подтверждение. После его получения, на сервер отправляется список топиков, от которых клиент хочет получать данные (рис. 4).

Рис. 4. Блок-диаграмма, подключение к серверу и топикам

Клиент ожидает, пока к нему не поступят новые данные из топиков, после чего отправляет их на сохранение в файл. Структура файла сохранения: имя топика, дата, время, данные (рис. 5).

Параллельно с этим они хранятся в оперативной памяти контроллера и выводятся на график в Web-интерфейсе.

Лицевая панель Web-интерфейса представлена на рисунке 6. В выпадающем меню Topic можно выбрать топик для отображения на графике Waveform Graph. На нем будут выведены все значения за определенный момент времени.

В поле Name File указано общее имя для файлов сохранения данных. При создании нового файла, его имя автоматически дополняется меткой времени от модуля часов реального времени.

Рис. 5. Блок-диаграмма, получение данных из топиков и сохранение в файл

Рис. 6. Лицевая панель Web-интерфейса, вкладки DATA и Setting

Разработанная система сбора данных может быть использована для организации распределенной сети контроля и мониторинга различного назначения. Использование в качестве среды разработки LabVIEW позволяет легко изменять функционал системы и модернизировать интерфейс пользователя.

В дальнейшем планируется применение ПЛИС на базе контроллера NI MyRIO для обеспечения аппаратурной поддержки алгоритмов предварительной обработки данных, повышения производительности систем и качества принятия решений в реальном масштабе времени.

Литература

1. Баран Е. Д. LabVIEW FPGA. Реконфигурируемые измерительные и управляющие системы. М.: ДМК Пресс, 2009. 448 с.

©Григорьев К. А., Ченский А. Г., 2020