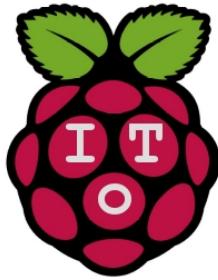


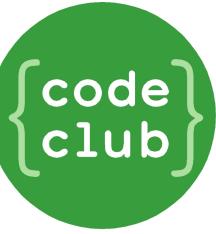
Internet of Things



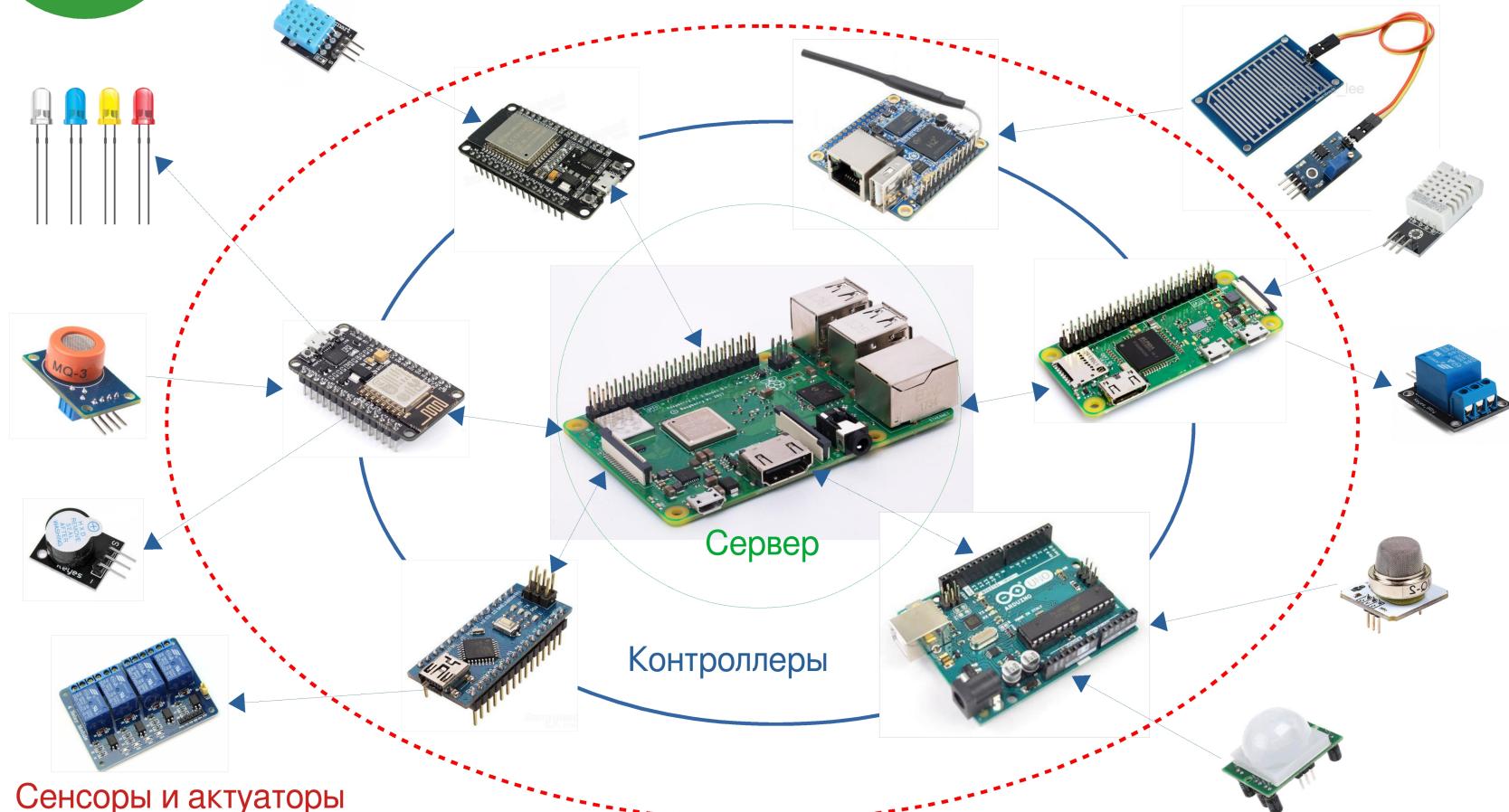
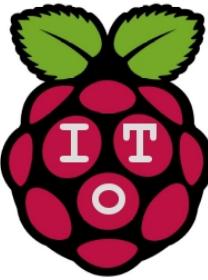
Domoticz — сервер IoT

Шадринск
2018-2019

M. B. Шохирев

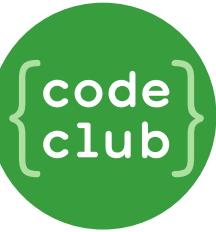


Сервер IoT

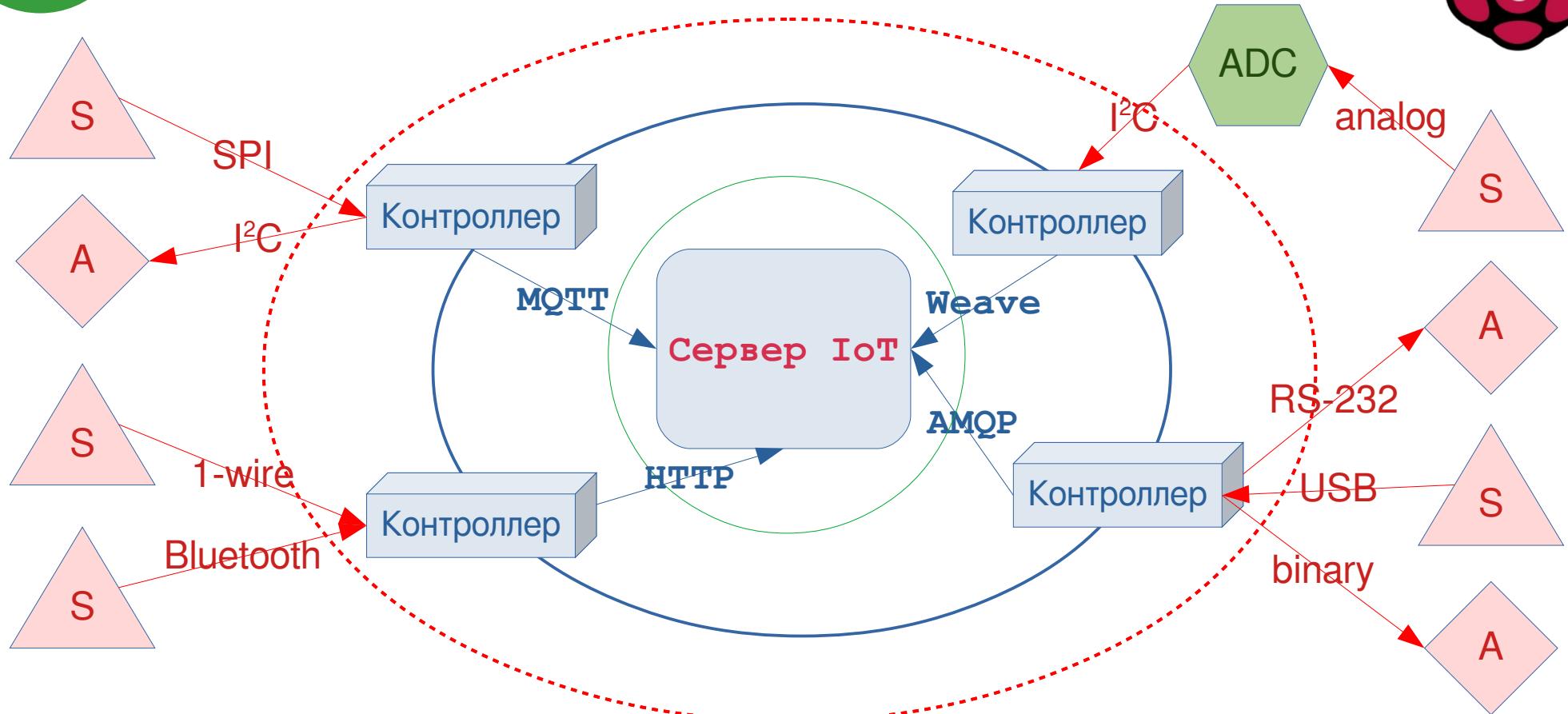
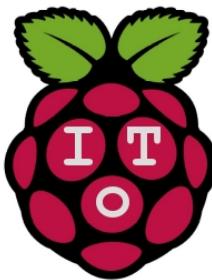


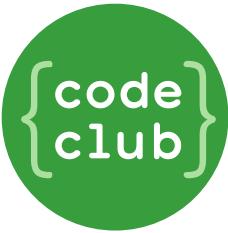
Сенсоры и актуаторы

Сервер IoT собирает сведения от «умных» вещей, накапливает их, анализирует данные, оценивает ситуацию, реагирует на изменения ситуации, выдаёт команды управления устройствами, оповещает или информирует человека.

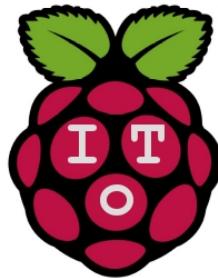


Связь с сервером





Серверы автоматизации



FHEM** — сервер на Perl для автоматизации домашнего хозяйства: включение ламп / жалюзи / обогрева / т. д., а также для регистрации данных об окружающей среде и потреблении энергии.



HomeAssistant.io** — платформа для домашней автоматизации с открытым исходным кодом на Python 3 для отслеживания и управления всеми устройствами дома, с простой и быстрой установкой.



MajorDoMo** — контроллер управления домашней автоматикой, мощный и гибкий инструмент, с помощью которого можно создать систему «умного дома».



Domogik** — бесплатное решение для мониторинга и управления домом с голосовым управлением и автоматизацией на сценариях.



SmartThings* позволяет организовать мониторинг обстановки и управление домом, используя смартфон.



Jeedom* — программное обеспечение для домашней автоматизации, включающее управление безопасностью и использованием энергии, автоматизацию отопления, управление автоматическими устройствами дома (на RPi).



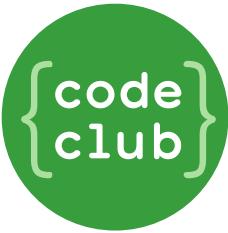
Yeti Smart Home* подключает разные умные домашние устройства, независимо от их бренда и технологии, и позволяет контролировать и автоматизировать их все из одного приложения.



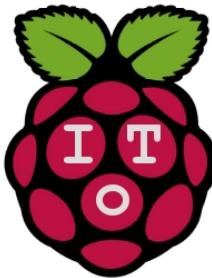
Xiaomi Smart Home — набор умных устройств и единая система управления домашним хозяйством с минимальным участием человека.

* free

** open source



Платформы IoT



openHAB** — платформа для домашней автоматизации с открытым исходным кодом на Java, которая объединяет в одном решении широкий спектр различных интеллектуальных домашних систем и технологий.



ioBroker** — интеграционная платформа для IoT, ориентированная на автоматизацию зданий, интеллектуальный учет, обслуживание на основе окружающей среды, автоматизацию процессов, визуализацию и регистрацию данных.

androidthings

Android Things* (Project Brillo) — платформа IoT от Google для объединения на Android подключаемых устройств для широкого спектра потребительских, торговых и промышленных применений.

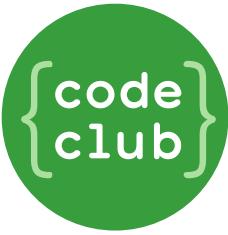


mozilla IoT** — проект, реализующий WoT (Web of Things), который включает программный шлюз **Things Gateway** на **Raspberry Pi**, который представляет собой универсальную прослойку для организации доступа к различным категориям потребительских и IoT-устройств, скрывающую за собой особенности каждой платформы. Представление строится на **Things UI**. Для настройки реакции на события движок обработки правил.

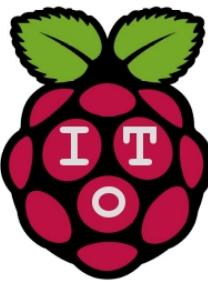
+ Коммерческие
продукты

* free

** open source



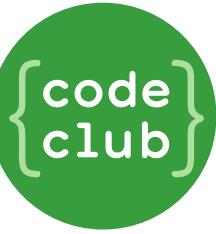
Коммерциализация IoT



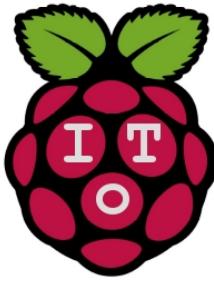
Ведущие компании индустрии ИТ участвуют в разделе растущего рынка IoT, предлагая свои решения.



	Cloud Services	Azure IoT	AWS IoT	iCloud	ARTIK Cloud/SmartThings
Application Protocols	Weave	AMQP	MQTT	HomeKit	MQTT
Network Protocols	WiFi/Thread	WiFi	WiFi	WiFi/BLE	WiFi/ZigBee/BLE/Thread
Operating Systems	Linux/Android Things	Windows IoT	Linux/AWS Greengrass	iOS	Linux/ARTIK

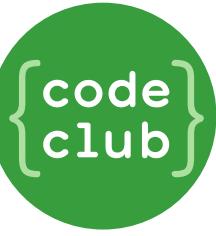


Domoticz

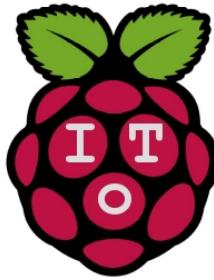


Domoticz — это автономный сервер для автоматизации, который собирает, отображает и накапливает данные о состоянии устройств и окружающих условиях (счётчики, датчики УФ-излучения, давления, влажности, температуры, протечки, открытия окон и дверей и т. д), управляет устройствами (выключатели, реле, регуляторы, светильники, вентиляторы, шторы, обогреватели, моторы и т. п.).





Domoticz

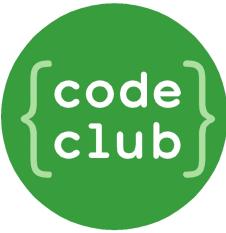


Domoticz — это свободная программа с открытым исходным кодом и API. Для него написано множество подключаемых модулей. Он работает под управлением различных ОС на разных аппаратных платформах, включая Raspberry Pi, имеет масштабируемый веб-интерфейс на HTML5, который автоматически адаптируется для настольных и мобильных устройств.

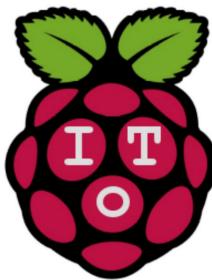
The screenshot shows the Domoticz web interface with a dark blue header bar containing tabs: Панель (Panel), План помещения (Room Plan), Переключатели (Switches), Температура (Temperature), Погода (Weather), Вспомогательные (Auxiliary), and Настройка (Settings). A dropdown menu 'Комната: Все' (Room: All) is open. Below the header is a section titled 'Свет/Выключатели:' (Light/Switches) displaying 18 items:

Название	Статус	Последний контакт
Лента Кухня	Выключен	Last Seen: 2018-11-16 08:03:59
Шлюз	Выключен	Last Seen: 2018-11-16 07:48:10
Гардероб	Выключен	Last Seen: 2018-11-16 08:05:00
Спальня лампочка	Включен	Last Seen: 2018-11-09 08:22:17
Розетка	Включен	Last Seen: 2018-11-16 07:48:11
Розетка USB	Выключен	Last Seen: 2018-11-16 12:52:22
Дверь вход	Закрыто	Last Seen: 2018-11-15 20:56:57
Дверь балкон	Закрыто	Last Seen: 2018-11-15 08:01:03
Дверь ванная	Открыто	Last Seen: 2018-11-16 07:48:11
Вентилятор	Выключен	Last Seen: 2018-10-25 01:17:57
Вытяжка	Включен	Last Seen: 2018-11-16 08:01:26
Ванная	Выключен	Last Seen: 2018-11-15 22:28:09
Зеркало	Включен	Last Seen: 2018-11-16 08:07:07
Вход	Выключен	Last Seen: 2018-11-15 20:57:55
Кухня	Выключен	Last Seen: 2018-10-27 02:38:36
Бар	Выключен	Last Seen: 2018-11-13 18:14:29
Спальня	Выключен	Last Seen: 2018-10-05 20:34:42
Бра	Выключен	Last Seen: 2018-10-25 00:11:44
Ночник	Выключен	Last Seen: 2018-11-15 08:14:31
Торшер	Выключен	Last Seen: 2018-11-15 08:14:31
Зал	Выключен	Last Seen: 2018-10-25 19:48:02
Отключение света, вся квартира	Выключен	Last Seen: 2018-09-23 15:45:43
Балкон выключатель	Выключен	Last Seen: 2018-11-16 07:48:11
Розетка балкон	Включен	Last Seen: 2018-11-16 07:48:11





Domoticz ↔ MQTT

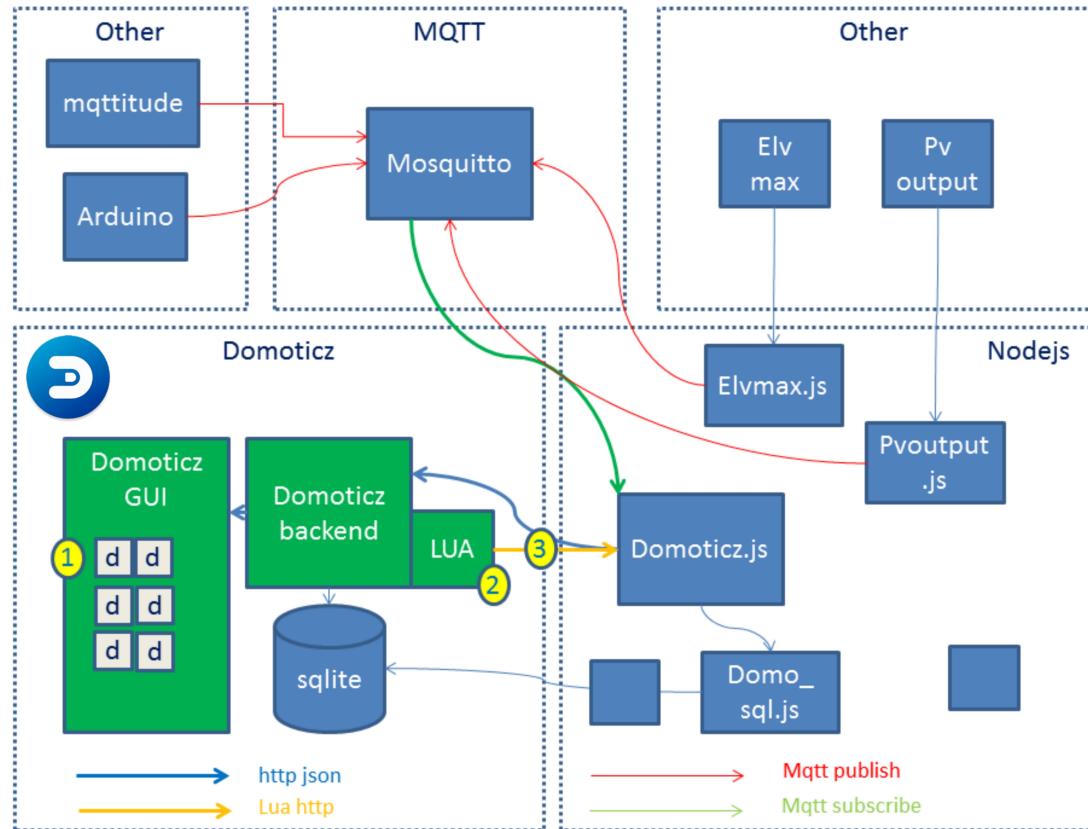


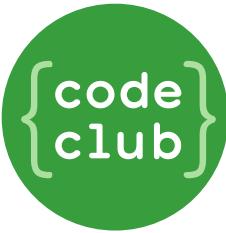
Domoticz может «общаться» со многими «умными» устройствами напрямую по их «родным» протоколам (RFXtrx433, Z-Wave, SmartMeter.io и т. д.).

Но поскольку разнообразных устройств великое множество, и у них самые разные интерфейсы, то для них Domoticz может публиковать события, а также реагировать на действия, запрошенные внешними клиентами, посредством встроенного интерфейса MQTT или через брокер MQTT (например, Mosquitto).

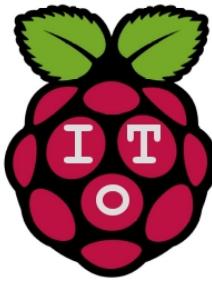
В Domoticz для приёма входящих и отправки исходящих сообщений MQTT определены 2 темы:

domoticz/in
domoticz/out





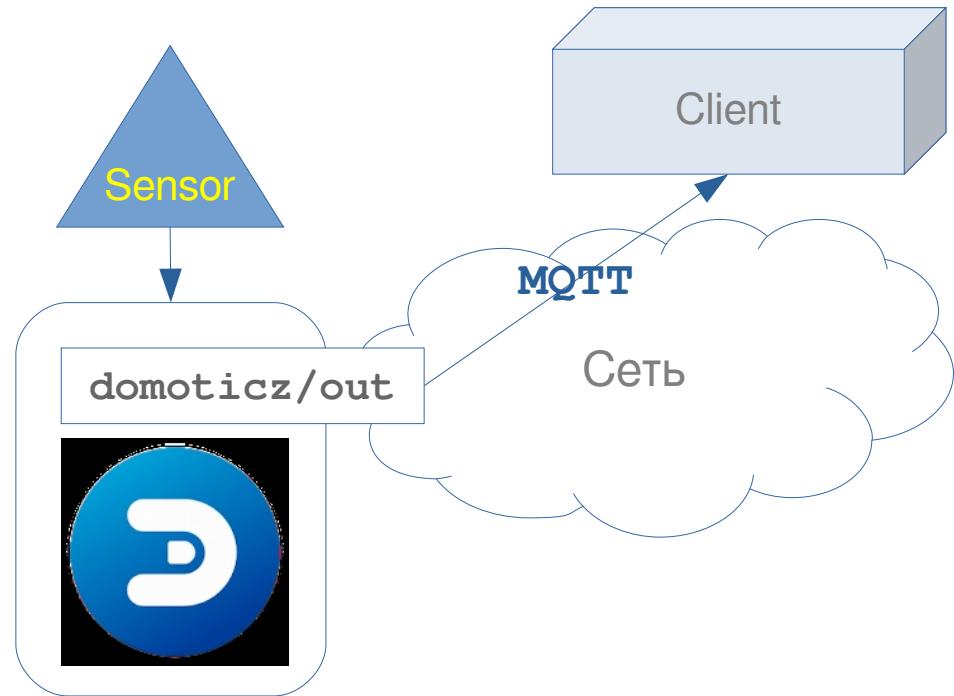
Domoticz → MQTT + JSON

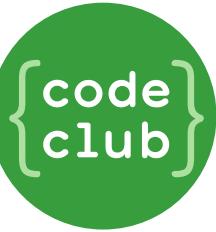


Сообщения для Domoticz оформляются в формате JSON по правилам открытого API. В этих сообщениях "idx" обозначает номер устройства, зарегистрированного на сервере Domoticz. Исходящие сообщения публикуются в теме `domoticz/out`:

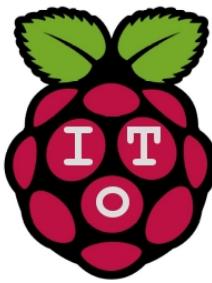
Domoticz → MQTT (domoticz/out)

```
{  
    "Battery" : 255,  
    "RSSI" : 12,  
    "description" : "t° на плате RPi",  
    "dtype" : "Temp",  
    "id" : "1",  
    "idx" : 2,  
    "name" : "RPi Internal Temperature",  
    "nvalue" : 0,  
    "stype" : "LaCrosse TX3",  
    "svalue1" : "58.0",  
    "unit" : 1  
}
```





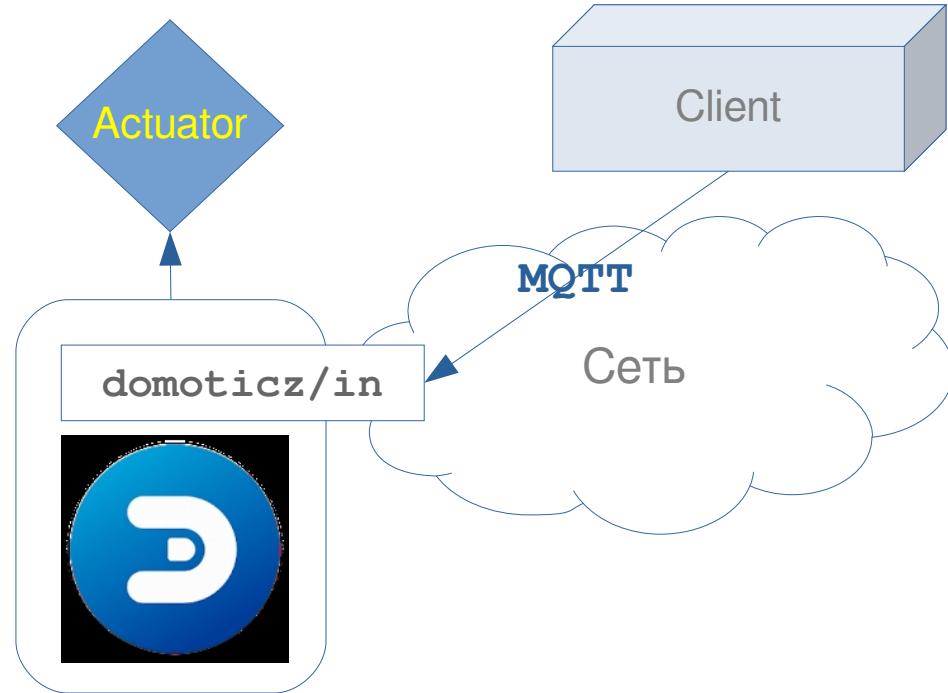
Domoticz ← MQTT + JSON

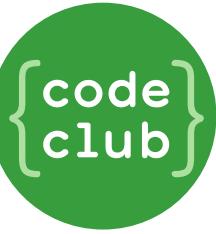


Сервер Domoticz может принимать по MQTT от внешних клиентов данные для учёта и отображения, а также команды для управления подключенными устройствами.
Входящие сообщения для Domoticz на нужно направлять в тему `domoticz/in`:

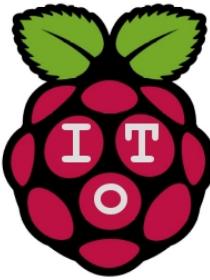
MQTT → Domoticz (domoticz/in)

```
{  
    "idx" : 7, "name" : "TestTemp",  
    "nvalue" : 0, "svalue" : "25.5"  
}  
  
{ "command": "switchlight", "idx": 2406,  
"switchcmd": "On", "description" :  
"Включить светильник"}  
  
{ "command": "switchlight", "idx": 2406,  
"switchcmd": "Set Level", "level": 100,  
"description" : "Установить уровень  
освещения"}
```



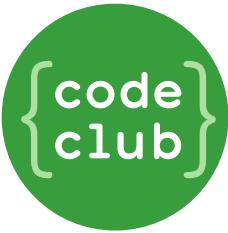


JSON

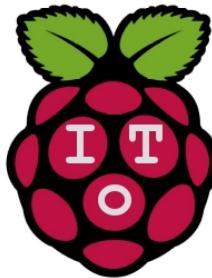


Формат **JSON** (JavaScript Object Notation) предназначен для текстового представления данных при их передаче по сети. Правила записи очень простые:

- **Запись** (хэш) — неупорядоченное множество пар ключ:значение, заключённое в фигурные скобки «{ }». Ключ описывается строкой, между ним и значением стоит символ «:». Пары ключ:значение отделяются друг от друга запятыми, например: { "firstName": "Юрий", "lastName": "Гагарин", birthday: "1934.03.09", age: 34 }
- Массив (список) — упорядоченное множество значений. Массив заключается в квадратные скобки «[]». Индексация значений — с 0. Значения (могут быть разного типа, включая подмассивы) разделяются запятыми, например: [3.1415926535897932384626433832795, 2.7182818284590452353602874713527]
- Числа, например: 123, -456.789
- Литералы: true, false, null
- Стока — упорядоченное множество из нуля или более символов в Unicode, заключённое в двойные кавычки (""). Символы могут быть указаны с использованием escape-последовательностей, начинающихся с обратной косой черты «\» (поддерживаются варианты \', \", \\, \/, \t, \n, \r, \f и \b), или записаны шестнадцатеричным кодом в кодировке Unicode в виде \uFFFF, например: "Язык программирования Ruby\n"



Domoticz ← HTTP ← JSON



По умолчанию сервер Domoticz слушает запросы по HTTP на порту 8080: отображает web-интерфейс и принимает команды управления.

Например, чтобы передать в Domoticz для устройства номер 7 значение температуры 25.5, нужно отправить команду с помощью такого запроса HTTP GET:

```
http://server:8080/json.htm?type=command&param=udevice&idx=7&nvalue=0&svalue=25.5
```

Это соответствует MQTT- сообщению в формате JSON:

```
{ "type": "command", "param": "udevice", "idx": 7, "nvalue": 0, "svalue": 25.5 }
```

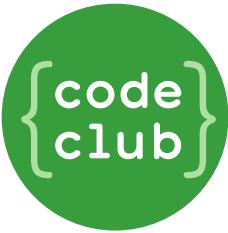
При успешном выполнении команды Domoticz ответит таким сообщением в формате JSON:

```
{ "status" : "OK", "title" : "Update Device" }
```

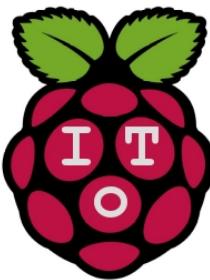
Если команда не выполнена (например, был указан несуществующий номер устройства), Domoticz ответит сообщением об ошибке:

```
{ "status" : "ERR" }
```

Такое «общение» лучше организовать по протоколу HTTPS с регистрацией по имени и паролю.



Domoticz ← HTTP



`http://server:8080/json.htm?type=command&command=switchlight&switchcmd=On&idx=2406`

Отправить устройству № 2406 команду «`switchlight`» с параметром «`On`», которое включит светильник.



On

Передать данные «25.5» от датчика температуры для отображения значения устройства № 7.

25.5 °C

`server:8080/json.htm`



`http://server:8080/json.htm?type=command¶m=udevice&idx=7&nvalue=0&svalue=25.5`

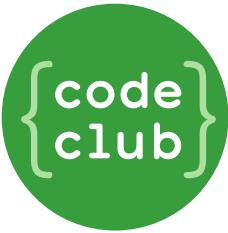
Client 1

HTTP

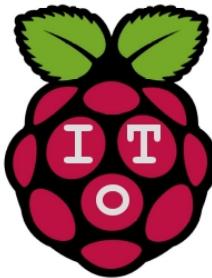
Сеть

HTTP

Client 2



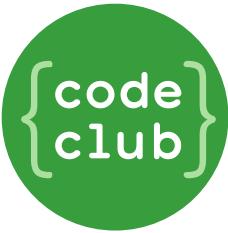
URL-кодирование



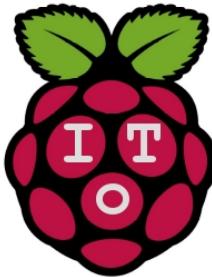
Формат записи данных при их передаче в адресной строке командой GET по протоколу HTTP называется URL-кодированием, которое выполняется по таким правилам:

- URL записывается так: `http://server:порт/ресурс/подресурс?список&параметров`
- При записи URL дозволены только латинские буквы, цифры и некоторые спецсимволы.
- Пробелы в URL не допускаются, если они необходимы — кодируются как `%20`
- Символы национальных алфавитов записываются в виде их шестнадцатеричных кодов, например строка «Пи» должна быть записана как `%D0%9F%D0%B8`
- Список параметров отделяется от ресурса вопросительным знаком «`?`»
- Параметры записываются в виде `имя=значение`
- Пары параметров соединяются символом «`&`»: `name1=value2&name2=value2`

`http://localhost:8080/json.htm?type=command¶m=udevice&idx=7&nvalue=0&svalue=25.5`

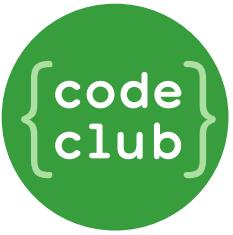


Источники

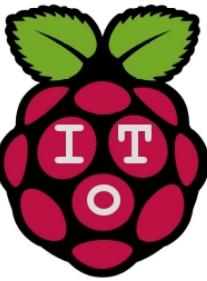


Ссылки на Интернет-ресурсы:

- <https://www.openhab.org/>
- <http://www.ioBroker.net/>
- <https://developer.android.com/things/index.html>
- <https://iot.mozilla.org/>
- <https://openweave.io/>
- <https://www.jeedom.com/site/en/>
- <https://www.smarthings.com/>
- <http://domogik.org/en/>
- <http://fhem.de/fhem.html>
- <https://majordomohome.com/> <https://mjdm.ru/>
- <https://xiaomi-smarthome.ru/>
- Domoticz на Raspberry Pi: установка, настройка, добавление первого датчика
- Добавляем датчик температуры DHT11/DHT22, AM2302 через GPIO в Domoticz
- <https://www.domoticz.com/wiki/MQTT>, https://www.domoticz.com/wiki/Domoticz_API/JSON_URL's
- <https://4pda.ru/forum/index.php?showtopic=927171>



Свой сервер?



В каких случаях может понадобиться
написать свой
сервер управления?