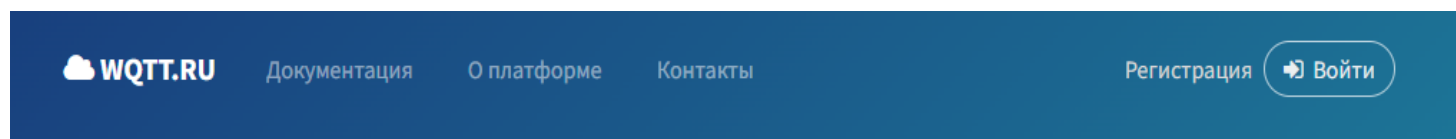


Голосовое управление лампой

“Fiery Led Lamp” через Яндекс Алису

Сперва нужно создать подключение к брокеру MQTT. Я воспользуюсь бесплатным брокером в интернете – CloudMQTT (<https://www.wqtt.ru/>). Далее я буду рассматривать настройку через него.

Регистрируемся на сайте CloudMQTT.



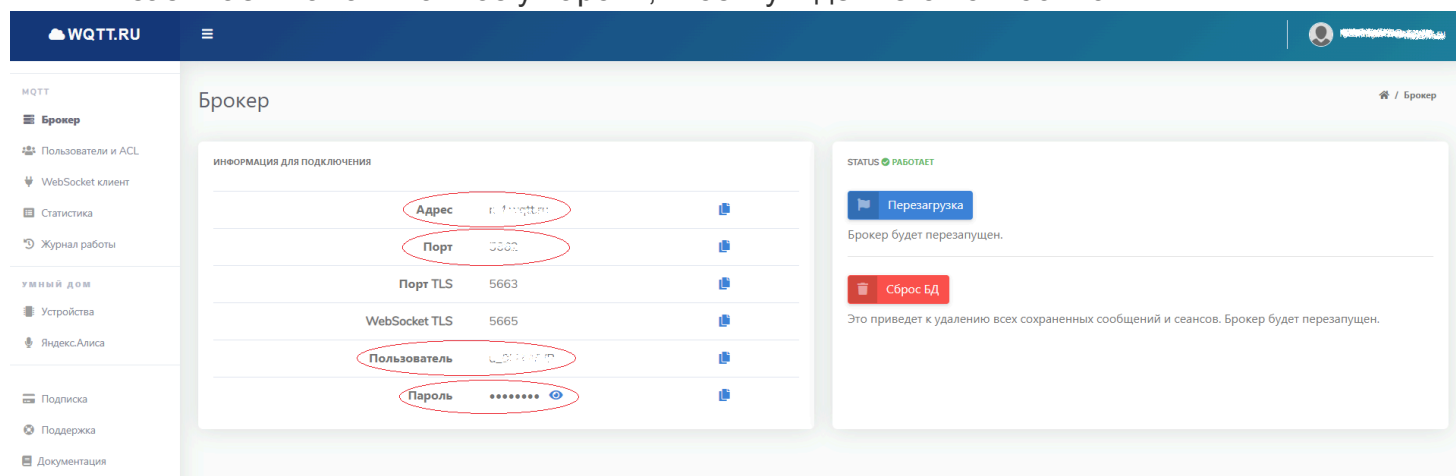
В прошивке лампы на закладке «Constants.h» разрешаем управление через MQTT

```
556 // --- ВНЕШНЕЕ УПРАВЛЕНИЕ -----
557 #define USE_MQTT                (true)                // true - используется mqtt клиент, false - нет
558 #if USE_MQTT
```

и на закладке «MqttManager.h» указываем настройки MQTT сервера: сервер, логин, пароль, порт. Остальные настройки не трогаем.

```
static const char MqttServer[]      PROGMEM = "w1.mqtt.ru";    // строка с IP адресом MQTT брокера
static const uint16_t MqttPort      = 5665;                  // порт MQTT брокера
static const char MqttUser[]        PROGMEM = "user";          // пользователь MQTT брокера
static const char MqttPassword[]    PROGMEM = "password";      // пароль пользователя MQTT брокера
static const char MqttClientIdPrefix[] PROGMEM = "LedLamp_";    // id клиента MQTT брокера (к нему будет добавлен ESP.getChipId)
```

Все настройки берем из личного кабинета CloudMQTT в разделе «Брокер». Не забываем нажать на глаз у пароля, чтобы увидеть его полностью.



После перепрошивки лампы с новыми параметрами сервера, мы должны увидеть в принятых сообщениях (раздел «WebSocket клиент»), что лампа подключена. Там же можно посмотреть ее номер ID_ESP – он нам потом понадобится. Если не видим в логе лампу – проверяем настройки сервера в прошивке, особенно номер порта.

1 Правило MQTT Включи лампу

Активационная фраза: Часть текста которая содержится в сообщении (строгое совпадение) Ответ Кузи: Ответ голосового навыка в ответе можно использовать {1}- найденный параметр из запроса

MQTT-брокер:  

 [Пример настройки cloudmqtt.com](#) 

Topic: QoS: RETAIN:

Payload:

Можно использовать метки заменители: {1} в это место будет подставлено значение найденное во фразе, {in} - исходный запрос Алисы

Client ID: Если оставить пустое поле, то Client ID будет сгенерен автоматически

Поиск значений в фразе:

Пример для "Все выше перечисленно"
Фраза: Включить громкость на 100 процентов и басы на 10.5
 [Пример поиска значений](#) 

Метки заменители для этого текста будут такие:
{on1} - 1
{ion1} - 0
{d1} - 100
{f1} - 10.5
{in} - исходный текст
{in_command} - чуть преобразованный текст
Если в тексте есть повторы цифр или вкл/выкл то они будут найдены аналогично {d2} {f2} {on2}

И заполняем поля

В качестве Topic для всех правил указываем «LedLamp/LedLamp_**ID_ESP**/cmdnd», где ID_ESP – номер лампы, просмотренный в сообщениях MQTT сервера.

В качестве «Client ID» можно указать что угодно – у меня «CID-Kuzya», что бы было понятно, откуда подключение, ведь практически так же можно настроить мобильное приложение для управления устройствами через MQTT.

В «Payload» указываем команду MQTT. Список команд можно посмотреть в прошивке на закладке «MqttManager.h», так же продублирую их в приложении. Для передачи команды от «Алисы» к «Домовенку Кузе» нужно в команде поставить параметр {value}.

«Поиск значений в фразе» ставим «Не искать»

Правило MQTT Включи эффект номер

Активационная фраза: Ответ Кузи

Часть текста которая содержится в сообщении (строгое совпадение) Ответ голосового навыка в ответе можно использовать (1)- найденный параметр из запроса

MQTT-брокер:

[Пример настройки cloudmqtt.com](#)

Topic: QoS: RETAIN:

Payload:

Можно использовать метки заменители: {1} в это место будет подставлено значение найденное во фразе, {in} - исходный запрос Алисы

Client ID:

Если оставить пустое поле, то Client ID будет сгенерен автоматически

Поиск значений в фразе:

Пример для "Все выше перечисленно"
Фраза: Включить громкость на 100 процентов и басы на 10.5

[Пример поиска значений](#)

Метки заменители для этого текста будут такие:

У себя я настроил следующие команды:

- Включи лампу
- Выключи лампу
- Включи эффект номер
- Яркость эффекта
- Громкость

Создаем виртуальные устройства: «Лампа» для управления яркостью и «Телевизор» для управления каналами (эффектами).

[Управление умным домом](#)

Управление умным домом

Перед началом работы посмотрите [видео инструкцию](#)

[Описание сервиса](#)

Код для объединения устройств с этими правилами:

[Виртуальные устройства умного дома](#) [Объединенные устройства](#)

[Добавить правило HTTP \(GET\)](#) [Добавить правило MQTT](#) [Добавить правило IFTTT](#) [Выход](#)

Указываем ранее сформированные правила для Вкл/Выкл, Яркость

🏠 / Управление умным домом / Виртуальные устройства

Виртуальные устройства домовенка Кузи

В этом разделе вы можете создать виртуальные устройства умного дома для приложения Яндекс

Работает с Алисой

Создать новое устройство

💡 Лампа

🔌 Розетка

🔌 Выключатель

🌡️ Термостат

🔒 Замок

🪟 Шторы

☕ Кофемашина

🍵 Чайник

🤖 Робот-пылесос

💧 Увлажнитель

🌬️ Очиститель воздуха

❄️ Кондиционер

📺 Телевизор

💡 RGB лампа

Устройства

Лампа Комната ▾

Название устройства

Лампа

Место расположения

Комната

Название устройства, будет участвовать в активационной фразе

Устройства группируются по комнатам

Включение выключение

Правило на включение

Включи лампу MQTT ▾

В правило будут переданы метки заменители {value} - 1 или 0 при выключении, если не используется правило выключения

Правило на выключение

Выключи лампу MQTT ▾

Если параметр не указан то будет использовано правило Включения, В правило будут переданы метки заменители {value} - 0

Запрос состояния лампы

Нет ▾

Запрос состояния должен вернуть JSON с параметром value=(true|false)

Управления яркостью

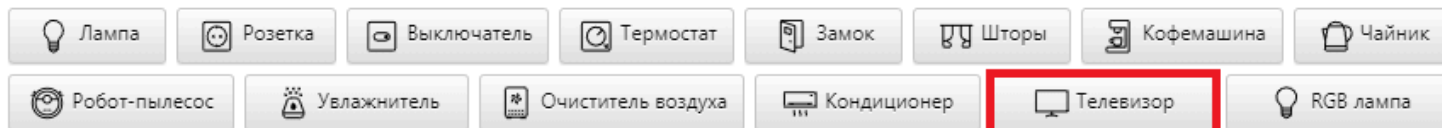
Правило для управления яркостью

Установи яркость MQTT ▾

В правило будут переданы метки заменители {value} - яркость

Создать новое устройство

работает
с Алисой



Устройства

Лампа Комната ▾



Название устройства

Лампа

Место расположения

Комната

Название устройства, будет участвовать в активационной фразе

Устройства группируются по комнатам

Включение выключение

Правило на включение

Нет

В правило будут переданы метки заменители
{value} - 1 или при выключении 0

Правило на выключение

Нет

Если параметр не указан то будет использовано правило Включения, В правило будут переданы метки заменители
{value} - 0

Запрос состояния вкл выкл

Нет

Запрос состояния, должно вернуть json с параметром value=(true/false)

Управление громкостью

Запрос громкости

Нет

Запрос громкости, должно вернуть json с параметром value=0, где 0 текущая громкость

Правило для управления громкостью

Сделай громкость MQTT

В правило будут переданы метки заменители
{value} - громкость, должно вернуть json с параметром value=0, где 0 текущая громкость

Управление каналами

Запрос канала

Нет

Запрос канала, должно вернуть json с параметром value=0, где 0 текущий канал

Правило для управления каналом

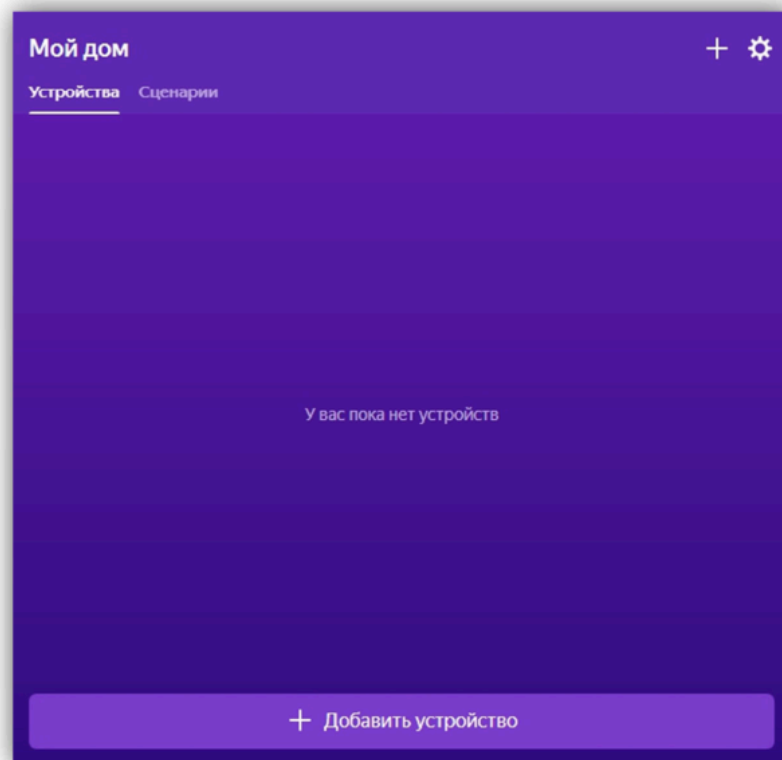
Включи эффект MQTT

В правило будут переданы метки заменители
{value} - канал, должно вернуть json с параметром value=0, где 0 текущий канал

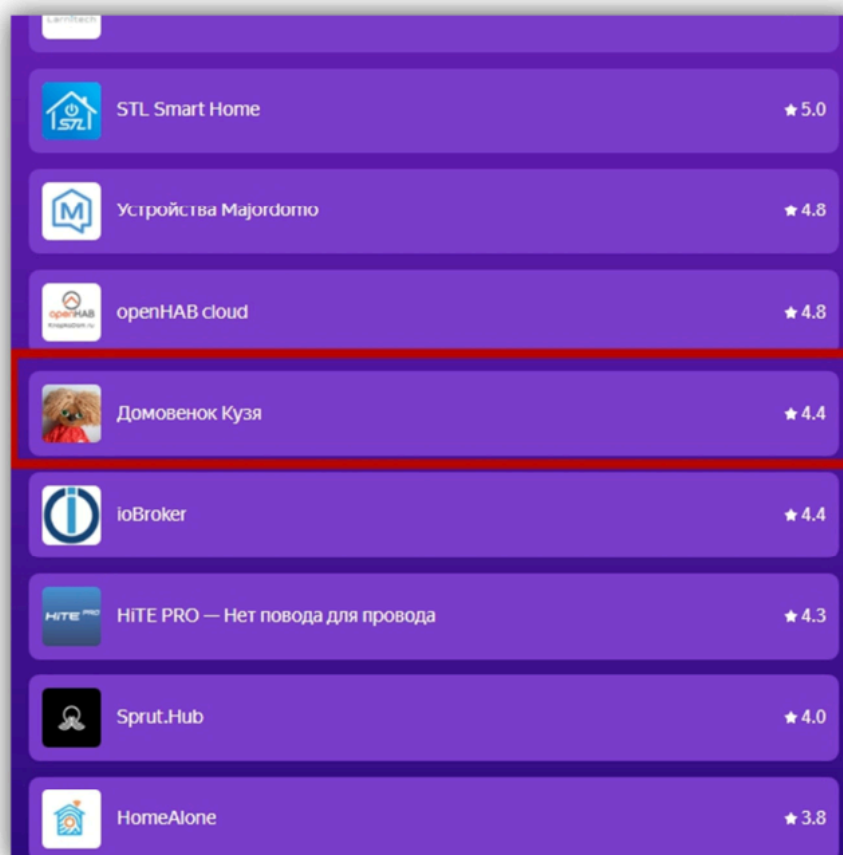
Указываем правила для Громкости и переключения эффекта

Название устройства будет известно Алисе, так же, как и расположение

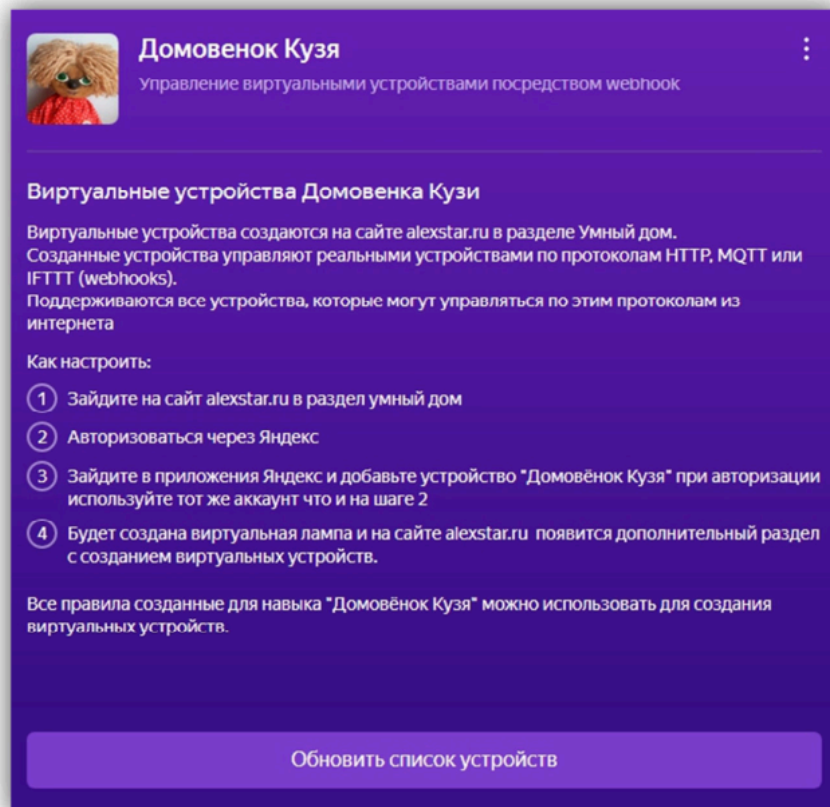
Далее, нужно добавить этот виртуальный телевизор в Алису. Для этого переходим в настройки умного дома от Яндекс по адресу <https://quasar.yandex.ru/skills/iot>



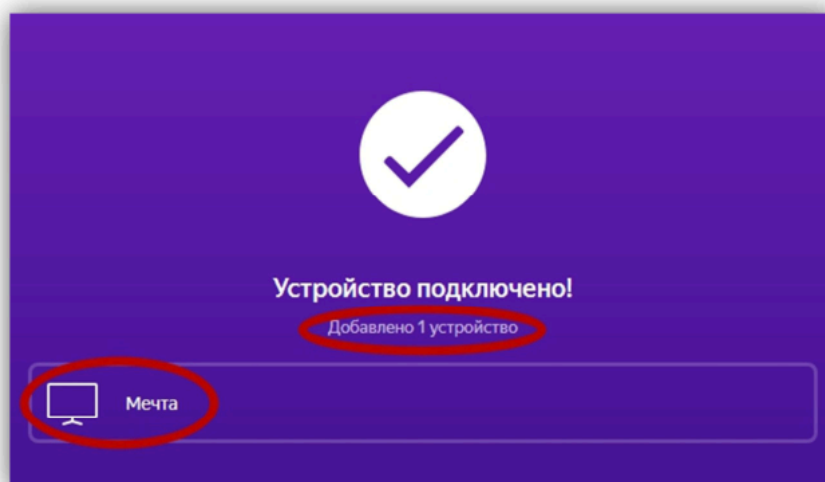
Жмем «+ Добавить устройство» и выбираем «Домовенок Кузя»



Далее «Обновить список устройств»

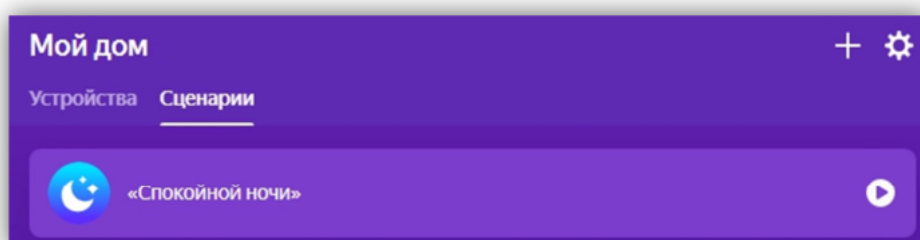


И все, теперь Алиса знает нашу лампу и может ей управлять.



Теперь можно управлять устройством напрямую через Алису «Алиса, включи Лампу», «Алиса, выключи все в комнате», «Алиса включи на лампе двенадцатый канал», «Алиса включи на лампе следующий канал».

Настало время для создания сценариев



Приложение. Список команд MQTT для навыка «Домовенок Кузя»

- P_ON - включить Лампу
- P_OFF – выключить Лампу
- SO_ON - включить звук
- SO_OFF - выключить звук
- EFF# – сделать активным эффект #, нумерация с нуля до 103
для управления через «Алису» указываем EFF{value}
- BRI# – установить яркость, диапазон от 1 до 255
для управления через «Алису» указываем BRI{value}
- SPD# – установить скорость, диапазон от 1 до 255
для управления через «Алису» указываем SPD{value}
- SCA# – установить масштаб, диапазон от 1 до 100
для управления через «Алису» указываем SCA{value}
- VOL# – установить громкость, диапазон от 1 до 30
для управления через «Алису» указываем VOL{value}
- BTN ON – разблокировать кнопку на лампе
- BTN OFF – заблокировать кнопку на лампе