



CODESYS V3.5

Библиотека OwenYandexHome



Руководство пользователя

11.02.2022

версия 2.0

Оглавление

1	Цель документа.....	4
2	Описание библиотеки OwenYandexHome.....	5
2.1	Установка библиотеки	5
2.2	Добавление библиотеки в проект CODESYS.....	6
2.3	Общие типы данных и POU	7
2.3.1	Перечисление ERROR.....	7
2.3.2	Перечисление DEVICE_TYPE_REQUEST	8
2.3.3	Перечисление DEVICE_TYPE	9
2.3.4	ФБ SmartDevice	10
2.4	Умения и датчики. Структура типов данных библиотеки	12
2.5	Типы данных для умений	14
2.5.1	Структура CAPABILITY_OBJECT.....	14
2.5.2	Перечисление CAPABILITY_TYPE	14
2.5.3	Перечисление CAPABILITY_FUNCTION	15
2.5.4	Структура CAPABILITY_PARAMETERS.....	16
2.5.5	Структура CAPABILITY_STATE	16
2.5.6	Структура COLOR_SETTING_PARAMETERS	17
2.5.7	Перечисление COLOR_SETTING_MODEL_TYPE.....	17
2.5.8	Структура COLOR_SETTING_TEMPERATURE_K_TYPE.....	17
2.5.9	Перечисление COLOR_SETTING_SCENE_TYPE	18
2.5.10	Структура RANGE_PARAMETERS	18
2.5.11	Перечисление RANGE_UNIT_TYPE	19
2.5.12	Структура RANGE_TYPE	19
2.5.13	Структура MODE_PARAMETERS	19
2.5.14	Перечисление MODE_TYPE	20
2.5.15	Структура ON_OFF_STATE	21
2.5.16	Структура TOGGLE_STATE	22
2.5.17	Структура MODE_STATE	22
2.5.18	Структура RANGE_STATE	22
2.5.19	Структура COLOR_SETTING_STATE	22
2.5.20	Структура COLOR_SETTING_VALUE_HSV_TYPE.....	23
2.6	Типы данных для встроенных датчиков.....	24
2.6.1	Структура PROPERTY_OBJECT.....	24
2.6.2	Перечисление PROPERTY_TYPE	24
2.6.3	Перечисление PROPERTY_FUNCTION	24
2.6.4	Структура PROPERTY_PARAMETERS	25
2.6.5	Структура PROPERTY_STATE	25

1 Цель документа

2.6.6	Структура FLOAT_PARAMETERS.....	26
2.6.7	Перечисление FLOAT_UNIT_TYPE	26
2.6.8	Структура EVENT_PARAMETERS	26
2.6.9	Перечисление EVENT_TYPE	26
2.6.10	Структура FLOAT_STATE	27
2.6.11	Структура EVENT_STATE.....	27
3	Определение идентификаторов (ID) устройств	28
3.1	Добавление устройств Яндекс в приложение Яндекс	28
3.2	Добавление устройств других производителей в приложение Яндекс.....	34
4	Получение токена	37
5	Примеры.....	39

1 Цель документа

Настоящее руководство описывает настройку обмена с устройствами [платформы умного дома Яндекс](#) для контроллеров ОБЕН, программируемых в среде **CODESYS V3.5**, с использованием библиотеки **OwenYandexHome**. Руководство предназначено для пользователей, которые обладают базовыми навыками работы с **CODESYS** и ПЛК, поэтому общие вопросы (например, создание и загрузка проектов) в данном документе не рассматриваются. Основная информация приведена в документах **CODESYS V3.5. Первый старт** и **CODESYS V3.5. FAQ**, которые доступны на сайте [ОБЕН](#) в разделе **CODESYS V3/Документация**.

Библиотека использует [REST API платформы умного дома](#) для отправки запроса устройствам. Поддерживаются как [устройства Яндекса](#), так и устройства сторонних производителей, поддерживающих интеграцию с платформой умного дома Яндекс и полностью соответствующих спецификации протокола (версия API – **v1.0**). Устройства, которые входят в платформу умного дома Яндекс, имеют отметку «работает с Алисой». Для возможности работы с устройствами требуется предварительно [добавить их в приложение Яндекс](#).

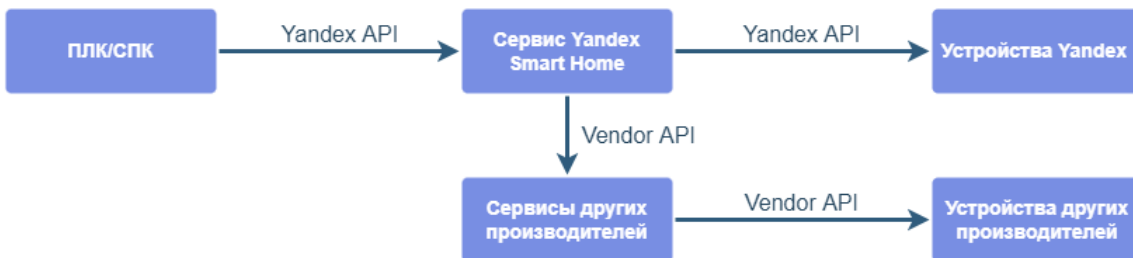


Рисунок 1.1 – Структурная схема взаимодействия устройств платформы умного дома

Список устройств, с которыми тестировалась работа библиотеки:

- Умная лампочка Яндекса;
- Умная розетка Sibling Powerswitch-Lux;
- Умная Wi-Fi RGB лампочка Sibling;
- Умный беспроводной выключатель Sibling Powerlight-M1;
- Умная розетка Xiaomi Smart Plug (WIFI) ZNCZ05CM Mi;
- Умная лампа Aqara LED Light Bulb;
- Умная розетка Aqara Smart Plug;
- Выключатель 2x1-полюсный Aqara QBKG03LM;
- Умные жалюзи SmartBlinds ver.3;
- Умная лампа Tuya;
- Умная Wi-Fi розетка Tuya Smart Power Plug WR01;
- Датчик качества воздуха Aqara TVOC Air quality monitor;
- Чайник REDMOND SkyKettle RK-G240S Bluetooth v4.0.



ПРИМЕЧАНИЕ

Для использования библиотеки требуется версия **CODESYS V3.5 SP14 Patch 3** или выше.



ПРИМЕЧАНИЕ

Работа библиотеки поддерживается только на контроллерах ОБЕН.

2 Описание библиотеки OwenYandexHome

2.1 Установка библиотеки

Библиотека **OwenYandexHome** доступна для загрузки на сайте компании [OVEN](#) в разделе **CODESYS V3/Библиотеки**. Для каждой версии CODESYS доступна своя версия библиотеки.

Для установки библиотеки в **CODESYS** в меню **Инструменты** следует выбрать пункт **Репозиторий библиотек**, нажать кнопку **Установить** и указать путь к файлу библиотеки:

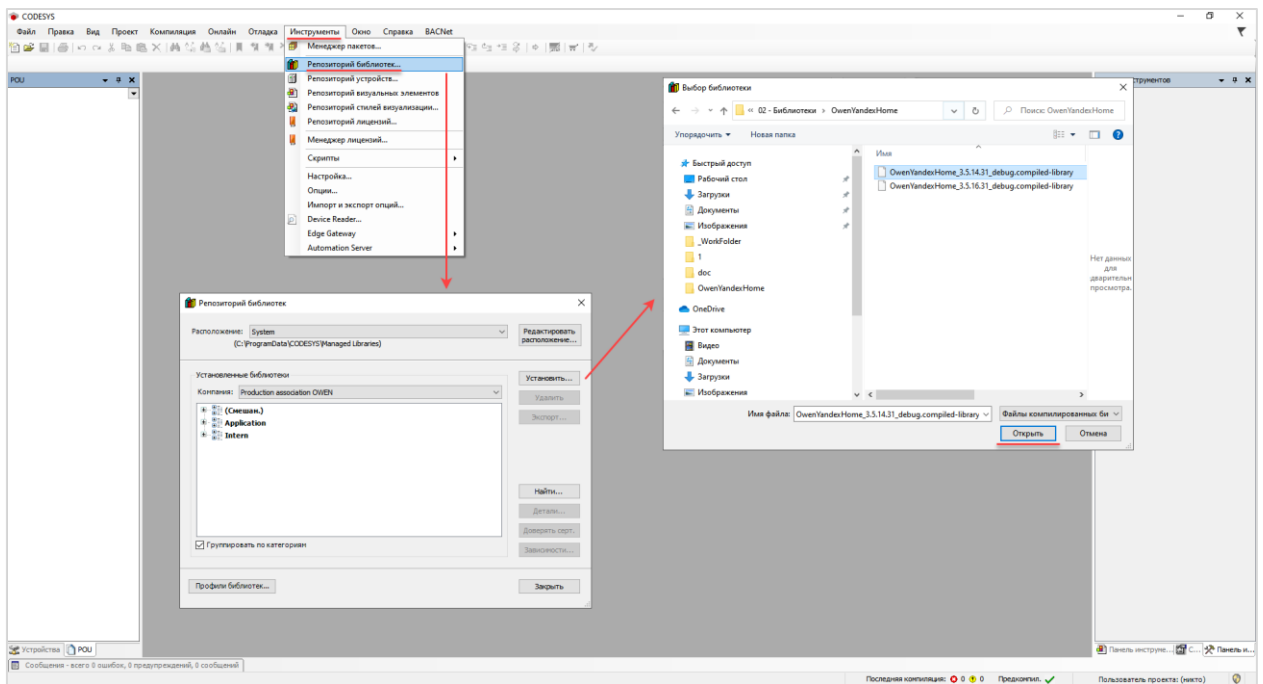


Рисунок 2.1 – Установка библиотеки OwenYandexHome в среду CODESYS

2.2 Добавление библиотеки в проект CODESYS

Для добавления библиотеки **OwenYandexHome** в проект **CODESYS** в **Менеджере библиотек** следует нажать кнопку **Добавить библиотеку**, в появившемся списке выбрать библиотеку **OwenYandexHome** и нажать **ОК**.

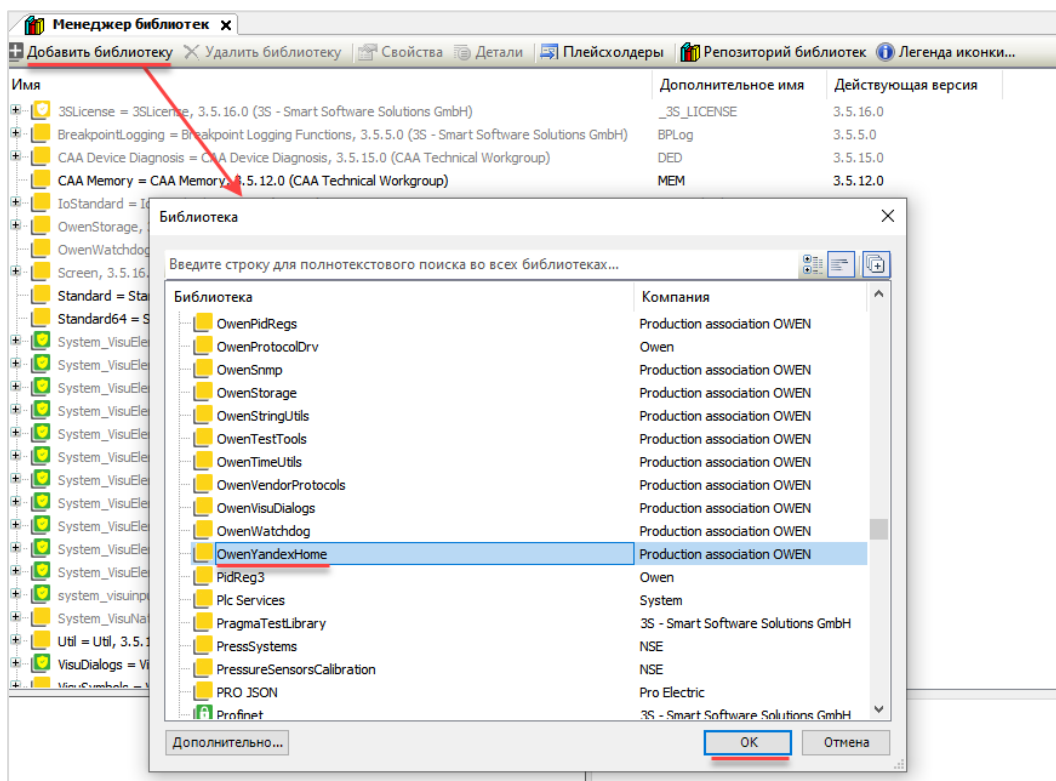


Рисунок 2.2 – Добавление библиотеки OwenYandexHome

После добавления библиотека появится в списке **Менеджера библиотек**:

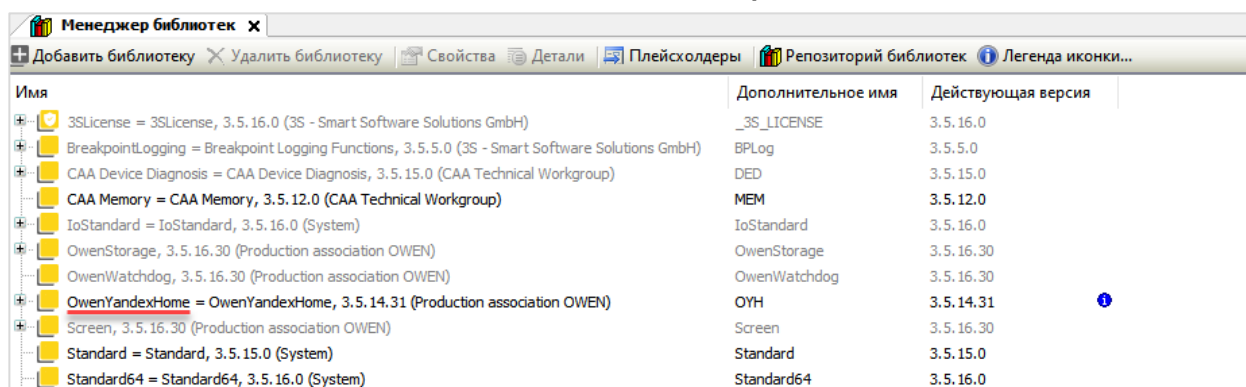


Рисунок 2.3 – Список библиотек проекта



ПРИМЕЧАНИЕ

При обращении к типам данных и POU библиотеки следует перед их названием указывать префикс **OYH** (пример: **OYH.ERROR**).

2.3 Общие типы данных и POU

2.3.1 Перечисление ERROR

Перечисление **ERROR** описывает ошибки и сообщения, которые могут возникнуть во время вызова ФБ [SmartDevice](#).

Таблица 2.3.1 – Описание элементов перечисления ERROR

Название	Значение	Описание	Пример
NO_ERROR	0	Нет ошибок	
DOOR_OPEN	1	Открыта дверца	Запуск стирки при неплотно закрытой дверце стиральной машины
LID_OPEN	2	Открыта крышка	Запуск программы приготовления на мультиварке с открытой крышкой
REMOTE_CONTROL_DISABLED	3	Удаленное управление устройством отключено	Попытка управления устройством, у которого отключена функция удаленного управления
NOT_ENOUGH_WATER	4	Недостаточно воды	Запуск чайника или увлажнителя без воды
LOW_CHARGE_LEVEL	5	Низкий уровень заряда	Запуск робота-пылесоса с разряженной батареей
CONTAINER_FULL	6	Контейнер полон	Запуск робота-пылесоса с переполненным контейнером для пыли
CONTAINER_EMPTY	7	Контейнер пуст	Запуск кофеварки с пустым контейнером для зерен
DRIP_TRAY_FULL	8	Сливной поддон полон	Запуск кофеварки с переполненным сливным поддоном
DEVICE_STUCK	9	Устройство застряло	Команда застрявшему роботу-пылесосу
DEVICE_OFF	10	Устройство выключено	Команда выключенному устройству
FIRMWARE_OUT_OF_DATE	11	Прошивка устарела	Команда устройству с устаревшей версией прошивки, которая не позволяет выполнить указанную команду
NOT_ENOUGH_DETERGENT	12	Недостаточно моющего средства	Запуск посудомоечной машины без моющего средства
HUMAN_INVOLVEMENT_NEEDED	13	Требуется вмешательство человека	Запуск стиральной машины с перегруженным барабаном
DEVICE_UNREACHABLE	14	Устройство недоступно	Запуск устройства, отключенного от электричества или от сети интернет
DEVICE_BUSY	15	Устройство занято	Команда стиральной машине, которая уже выполняет программу стирки
INTERNAL_ERROR	16	Неизвестная внутренняя ошибка	Любая неклассифицируемая ошибка на стороне платформы партнера
INVALID_ACTION	17	Недопустимое действие	Команда изменения яркости света на стиральной машине

Название	Значение	Описание	Пример
			без соответствующего умения
INVALID_VALUE	18	Недопустимое значение	Команда на установку значения вне допустимого диапазона – например, включение 150-го канала на телевизоре при доступных 100
NOT_SUPPORTED_IN_CURRENT_MODE	19	Не поддерживается в текущем режиме работы устройства	Команда на изменение температуры кондиционера, находящегося в режиме вентиляции (в данном режиме охлаждение или нагрев воздуха не происходит)
ACCOUNT_LINKING_ERROR	20	Ошибка в OAuth2 токене пользователя	Попытка управлять устройством с просроченным или инвалидированным OAuth2 токеном
NOT_OWEN_DEVICE	51	Данное устройство (ПЛК) не является контроллером ОВЕН	
INVALID_REQUEST_TYPE	52	Указан неподдерживаемый тип запроса (см. вход eTypeRequest)	
INVALID_CAPABILITY_POINTER	53	Передан некорректный указатель на массив умений устройства (см. вход pArrayCapabilityObject)	
INVALID_CAPABILITY_CNT	54	Указанное количество умений устройства не входит в допустимые ограничения	
INVALID_PROPERTY_POINTER	55	Передан некорректный указатель на массив датчиков устройства (см. вход pArrayPropertyObject)	
INVALID_PROPERTY_CNT	56	Указанное количество датчиков устройства не входит в допустимые ограничения	
TOKEN_EMPTY	57	Строка sToken пустая	
SID_EMPTY	58	Строка sid пустая	
RESPONSE_FAILURE	59	Получен некорректный ответ на запрос	

2.3.2 Перечисление DEVICE_TYPE_REQUEST

Перечисление **DEVICE_TYPE_REQUEST** описывает типы запросов, которые поддерживает ФБ [SmartDevice](#).

Таблица 2.3.2 – Описание элементов перечисления **DEVICE_TYPE_REQUEST**

Название	Значение	Описание
GetState	1	Запрос об актуальном состоянии устройства
SetActions	2	Запрос на выполнение массива умений устройства

2.3.3 Перечисление DEVICE_TYPE

Перечисление **DEVICE_TYPE** описывает типы устройств платформы умного дома Яндекс.

Таблица 2.3.3 – Описание элементов перечисления DEVICE_TYPE

Название	Значение	Описание
light	1	Лампочка, светильник, ночник, люстра
socket	2	Умная розетка
switch	3	Настенный выключатель света, тумблер, автомат в электрическом щитке, умное реле, умная кнопка
thermostat	4	Водонагреватель, теплый пол, обогреватель, электровентилятор
thermostat_ac	5	Кондиционер
media_device	6	DVD-плеер и другие медиаустройства, которые умеют воспроизводить звук и видео
media_device_tv	7	Холодильник, духовой шкаф, кофеварка, мультиварка
media_device_tv_box	8	ИК-пульт от ТВ-приставки, умная ТВ-приставка
media_device_receiver	9	ИК-пульт от ресивера, AV-ресивер, спутниковый ресивер
cooking	10	Холодильник, духовой шкаф, кофеварка, мультиварка
cooking_coffee_maker	11	Кофеварка, кофемашина
cooking_kettle	12	Умный чайник, термопот
cooking_multicooker	13	Мультиварка
openable	14	Дверь, ворота, окно, ставни
openable_curtain	15	Шторы, жалюзи
humidifier	16	Увлажнитель воздуха
purifier	17	Очиститель воздуха, мойка воздуха
vacuum_cleaner	18	Робот-пылесос
washing_machine	19	Стиральная машина
dishwasher	20	Посудомоечная машина
iron	21	Утюг, парогенератор
sensor	22	Датчик температуры, датчик влажности, датчик открытия двери, датчик движения (см. PROPERTY_FUNCTION)
other	23	Остальные устройства, не подходящие под существующие типы

2.3.4 ФБ SmartDevice

Функциональный блок **SmartDevice** используется для отправки запроса устройству платформы умного дома.



ВНИМАНИЕ

Не допускается использование в проекте более одного экземпляра блока. Для опроса нескольких устройств следует последовательно вызывать один и тот же экземпляр блока с разными аргументами. Новый запрос должен отправляться только после получения ответа на текущий (после появления **TRUE** на выходе **xDone**).

По переднему фронту на входе **xExecute** происходит однократная отправка запроса устройству с идентификатором **sid** (процедура определения идентификатора описана в [п. 3](#)). На входе **sToken** указывается токен приложения (процесс получения токена описан в [п. 4](#)). Вход **eTypeRequest**, представляющий собой перечисление типа [DEVICE_TYPE_REQUEST](#), определяет тип выполняемого запроса – **GetRequest** (чтение) или **SetRequest** (запись).

При выполнении запроса **GetRequest** полученная от устройства информация об умениях размещается под указателем **pArrayCapabilityObject**, а информация о датчиках – под указателем **pArrayPropertyObject**. Пользователь должен передать на вход **pArrayCapabilityObject** массив типа **ARRAY [1..x] OF CAPABILITY OBJECT**, а на вход **pArrayPropertyObject** – **ARRAY [1..x] OF PROPERTY OBJECT**. На входах **usiCntCapabilityObject** и **usiCntPropertyObject** следует указать размерность этих массивов (не более **26** для **usiCntCapabilityObject** и не более **23** для **usiCntPropertyObject**). Если число умений/датчиков устройства заранее неизвестно, то рекомендуется при первом вызове блока использовать максимально возможную размерность массивов, а далее ориентироваться на значения выходов **usiCntCapability** и **usiCntProperty** (см. ниже).

При выполнении запроса **SetRequest** из массива, размещенного под указателем **pArrayCapabilityObject**, в устройство будут отправлены значения полей структуры [CAPABILITY OBJECT](#) **eFunction** и **stState**. Массив под указателем **pArrayPropertyObject** при отправке этого запроса не используется.

В случае получения корректного ответа выход **xDone** принимает значение **TRUE** на один цикл ПЛК. При выполнении запроса **GetRequest** после этого появляются значения на выходах:

- **eType** (перечисление типа [DEVICE_TYPE](#)) – тип устройства;
- **xState** – состояние устройства (**TRUE** – устройство доступно, **FALSE** – нет связи с устройством; может не поддерживаться устройством);
- **usiCntCapability** – число прочитанных умений устройства;
- **usiCntProperty** – число прочитанных датчиков устройства.

В случае ошибки выход **xError** принимает значение **TRUE**, а на выходе **eError** отображается код ошибки из перечисления [ERROR](#). На выходе **sMessage** отображается текстовое описание ошибки (может не поддерживаться устройством).

Для отправки нового запроса следует создать передний фронт на входе **xExecute**.

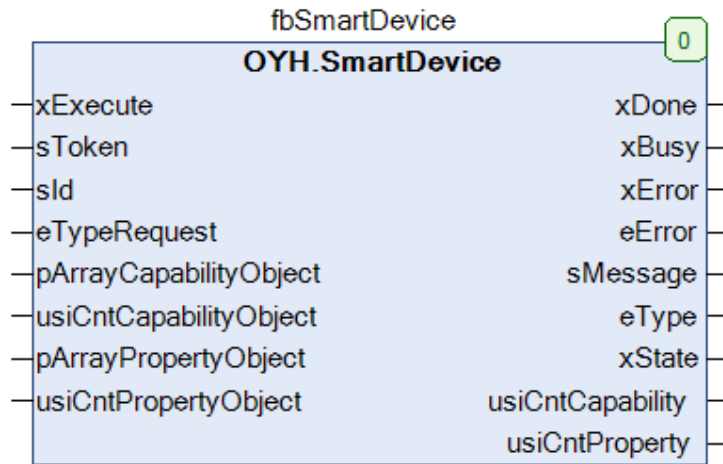


Рисунок 2.3.1 – Внешний вид ФБ SmartDevice на языке CFC

Таблица 2.3.4 – Описание входов и выходов ФБ SmartDevice

Название	Тип	Описание
Входы		
xExecute	BOOL	По переднему фронту происходит однократная отправка запроса
sToken	STRING	Токен приложения (процесс получения токена описан в п. 4)
sId	STRING	Идентификатор устройства (см. п. 3)
eTypeRequest	DEVICE_REQUEST_TYPE	Тип запроса (GetRequest или SetRequest)
pArrayCapabilityObject	POINTER TO CAPABILITY_OBJECT	Указатель на массив умений устройства (ARRAY [1..usiCntCapabilityObject] OF CAPABILITY_OBJECT)
usiCapabilityObject	USINT(0..26)	Число элементов массива под указателем pArrayCapabilityObject
pArrayPropertyObject	POINTER TO PROPERTY_OBJECT	Указатель на массив датчиков устройства (ARRAY [1..usiCntPropertyObject] OF PROPERTY_OBJECT)
usiPropertyObject	USINT(0..23)	Число элементов массива под указателем pArrayPropertyObject
Выходы		
xDone	BOOL	Принимает TRUE на один цикл ПЛК при получении ответа
xBusy		TRUE – блок находится в работе
xError	BOOL	Принимает значение TRUE в случае возникновения ошибки
eError	ERROR	Статус работы ФБ (или код ошибки)
sMessage	STRING(255)	Сообщение об ошибке от сервера Yandex (может не поддерживаться устройством)
eType	DEVICE_TYPE	Тип устройства
xState	BOOL	TRUE – устройство доступно, FALSE – нет связи с устройством; может не поддерживаться устройством
usiCntCapability	USINT	Количество прочитанных умений устройства
usiCntProperty	USINT	Количество прочитанных датчиков устройства

2.4 Умения и датчики. Структура типов данных библиотеки

[Умение \(capability\)](#) – это свойство устройства, которое описывает его возможности. Платформа умного дома Яндекс поддерживает следующие типы умений:

- **on_off** – удаленное включение и выключение устройства (аналог нажатия кнопки питания на устройстве или его пульте управления);
- **color_setting** – Управление цветом для светящихся элементов в устройстве;
- **mode** – переключение режимов работы устройства (например, переключение между температурными режимами работы кондиционера: «Охлаждение», «Нагрев» или «Авто»);
- **range** – управление параметрами устройства, которые имеют диапазон (например, яркость лампы, громкость звука, температура обогревателя);
- **toggle** – управление параметрами устройства, которые могут находиться только в одном из двух состояний (например, кнопки, тумблеры и подобные им элементы управления, которые включают или выключают какую-либо дополнительную функцию устройства).

[Встроенные датчики \(properties\)](#) — свойства устройства, которые описывают его состояние. Датчики обычно используют для контроля за окружающей обстановкой. У устройства могут отсутствовать встроенные датчики. Платформа умного дома Яндекс поддерживает следующие типы встроенных датчиков:

- **float** – отображение показаний датчиков, встроенных в устройство, в числовом формате;
- **event** – отображение показаний датчиков событий, встроенных в устройство (датчики могут иметь одно из заданных состояний).

В библиотеке **OwenYandexHome** каждое умение устройства представлено структурой [CAPABILITY_OBJECT](#), а каждый встроенный датчик – структурой [PROPERTY_OBJECT](#). Эти структуры включают в себя вложенные структуры и перечисления, используемые для описания параметров всех поддерживаемых типов умений и датчиков. На рис. 2.4.1 и рис. 2.4.2 представлены схемы связей между типами данных библиотеки. Описание этих типов данных приведено в [п. 2.5](#) (умения) и [2.6](#) (встроенные датчики).

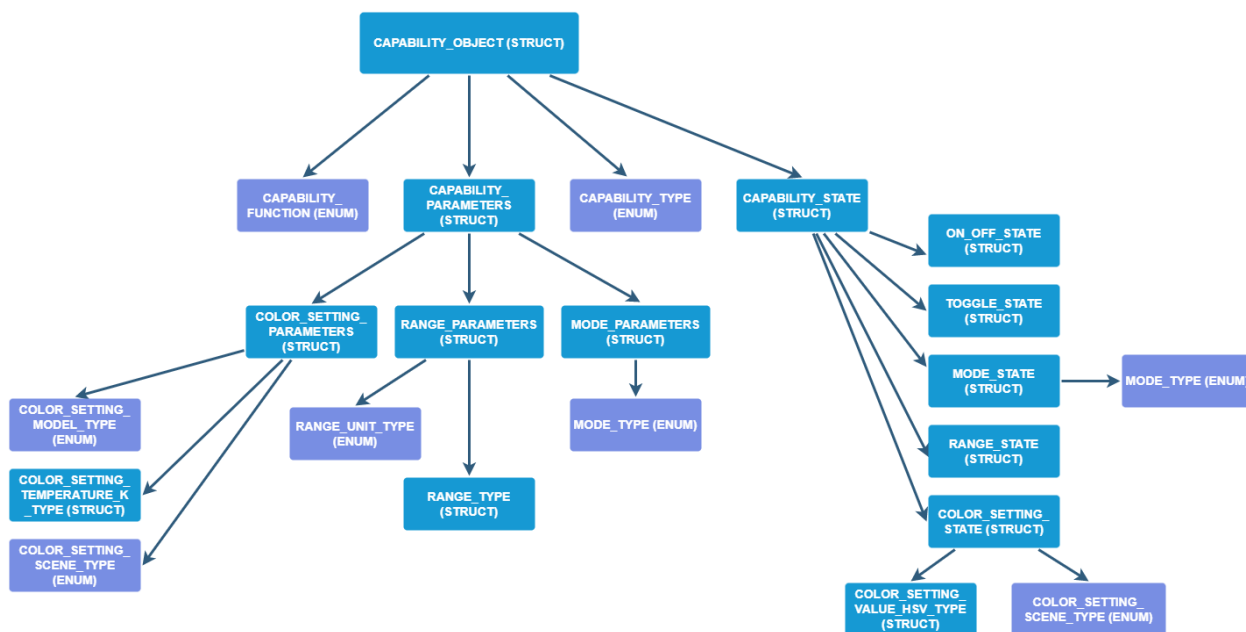


Рисунок 2.4.1 – Схема связей между типами данных, описывающими умения

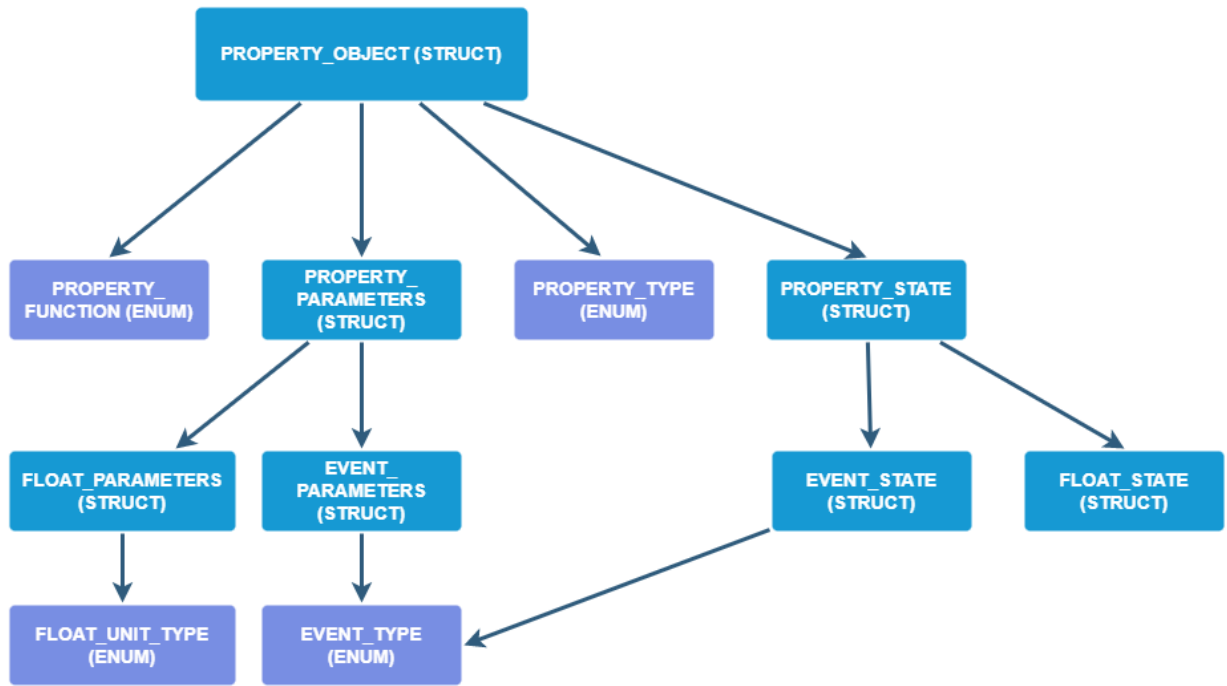


Рисунок 2.4.2 – Схема связей между типами данных, описывающими встроенные датчики

2.5 Типы данных для умений

2.5.1 Структура CAPABILITY_OBJECT

Структура **CAPABILITY_OBJECT** описывает умение устройства. Массив таких структур размещается под указателем **pArrayCapabilityObject** блока [SmartDevice](#).

Таблица 2.5.1 – Описание элементов структуры CAPABILITY_OBJECT

Название	Тип	Описание
eType	CAPABILITY_TYPE	Тип умения
eFunction	CAPABILITY_FUNCTION	Функция в пределах типа умения
xRetrievable	BOOL	Признак доступности для данного умения запроса состояния (GetState)
stParams	CAPABILITY_PARAMETERS	Параметры умения
stState	CAPABILITY_STATE	Текущее состояние умения
dtLastUpdated	DT	Время последнего обновления состояния умения
eError	ERROR	Код ошибки применения умения
sErrorMessage	STRING(255)	Сообщение об ошибке применения умения (может отсутствовать)

2.5.2 Перечисление CAPABILITY_TYPE

Перечисление **CAPABILITY_TYPE** описывает типы умений.

Таблица 2.5.2 – Описание элементов перечисления CAPABILITY_TYPE

Название	Значение	Описание
on_off	1	Удаленное включение и выключение устройства (аналог нажатия кнопки питания на устройстве или его пульте управления). Является базовым умением для большинства устройств
color_setting	2	Управление цветом для светящихся элементов в устройстве
mode	3	Переключение режимов работы устройства (например, переключение между температурными режимами работы кондиционера: «Охлаждение», «Нагрев» или «Авто»)
range	4	Управление параметрами устройства, которые имеют диапазон (например, яркость лампы, громкость звука, температура обогревателя)
toggle	5	Управление параметрами устройства, которые могут находиться только в одном из двух состояний (например, кнопки, тумблеры и подобные им элементы управления, которые включают или выключают какую-либо дополнительную функцию устройства)

2.5.3 Перечисление CAPABILITY_FUNCTION

Перечисление **CAPABILITY_FUNCTION** описывает конкретную функцию в пределах типа умения.

Таблица 2.5.2 – Описание элементов перечисления **CAPABILITY_FUNCTION**

Название	Значение	Описание
Функции умения on_off		
on_off_on	1	Включение и выключение устройства
Функции умения color_setting		
color_setting_hsv	2	Изменение цвета в соответствии с моделью HSV
color_setting_rgb	3	Изменение цвета в соответствии с моделью RGB
color_setting_temperature_k	4	Изменение температуры цвета в кельвинах
color_setting_scene	5	Изменение тем и сценариев освещения
Функции умения mode		
mode_cleanup_mode	6	Установка режима уборки
mode_coffee_mode	7	Установка режима работы кофеварки
mode_dishwashing	8	Установка режима мытья посуды
mode_fan_speed	9	Установка режима работы скорости вентиляции (например, в кондиционере, вентиляторе или обогревателе)
mode_heat	10	Установка режима нагрева
mode_input_source	11	Установка источника сигнала
mode_program	12	Установка какой-либо программы работы
mode_swing	13	Установка направления воздуха в климатической технике
mode_tea_mode	14	Установка режима приготовления чая
mode_thermostat	15	Установка температурного режима работы климатической техники (например, в кондиционере)
mode_work_speed	16	Установка скорости работы
Функции умения range		
range_brightness	17	Изменение яркости световых элементов
range_channel	18	Изменение канала (например, телевизионного)
range_humidity	19	Изменение влажности
range_open	20	Открывание чего-либо (открывание штор, окна)
range_temperature	21	Изменение температуры. Может, например, обозначать температуру нагрева чайника, обогревателя или температуру кондиционера в каком-либо его режиме
range_volume	22	Изменение громкости устройства
Функции умения toggle		
toggle_backlight	23	Включение подсветки
toggle_controls_locked	24	Блокировка управления и включение детского режима
toggle_ionization	25	Включение ионизации

Название	Значение	Описание
toggle_keep_warm	26	Включение поддержания тепла
toggle_mute	27	Выключение звука на устройстве
toggle_oscillation	28	Включение вращения (например, включение вращения вентилятора)
toggle_pause	29	Временная остановка текущей деятельности устройства (например, временная остановка работающего робота-пылесоса)

2.5.4 Структура CAPABILITY_PARAMETERS

Структура **CAPABILITY_PARAMETERS** описывает параметры умения устройства. Параметры доступны только для чтения; они описывают характеристики устройства – например, поддерживаемую цветовую модель, диапазон значений функции и т.д. Так как структура [CAPABILITY_OBJECT](#) является универсальной, то в данной структуре пользователь должен использовать только одну из вложенных структур (в зависимости от значения поля **CAPABILITY_OBJECT.eType**).

Таблица 2.5.4 – Описание элементов структуры CAPABILITY_PARAMETERS

Название	Тип	Описание
stColor_setting	COLOR_SETTING_PARAMETERS	Параметры умения color_setting
stMode	MODE_PARAMETERS	Параметры умения mode
stRange	RANGE_PARAMETERS	Параметры умения range

2.5.5 Структура CAPABILITY_STATE

Структура **CAPABILITY_STATE** описывает состояние умения устройства и может использоваться для его изменения. Так как структура [CAPABILITY_OBJECT](#) является универсальной, то в данной структуре пользователь должен использовать только одну из вложенных структур (в зависимости от значения поля **CAPABILITY_OBJECT.eType**).

Таблица 2.5.5 – Описание элементов структуры CAPABILITY_STATE

Название	Тип	Описание
xNull	BOOL	TRUE – текущее состояние умения еще не передавалось в экосистему умного дома Яндекс
stOn_off	ON_OFF_STATE	Текущее состояние умения on_off
stColor_setting	COLOR_SETTING_STATE	Текущее состояние умения color_setting
stMode	MODE_STATE	Текущее состояние умения mode
stRange	RANGE_STATE	Текущее состояние умения range
stToggle	TOGGLE_STATE	Текущее состояние умения toggle

2.5.6 Структура COLOR_SETTING_PARAMETERS

Структура **COLOR_SETTING_PARAMETERS** описывает параметры умения типа **color_setting**.

Таблица 2.5.6 – Описание элементов структуры **COLOR_SETTING_PARAMETERS**

Название	Тип	Описание
xSupportColorModel	BOOL	Признак поддержки схемы изменения цвета (eColorModel)
xSupportTemperatureK	BOOL	Признак поддержки изменения температуры цвета в кельвинах (stTemperature_k)
usiCntScene	USINT(0..18)	Количество поддерживаемых объектов scene в массиве aeScenes (0 – нет ни одного поддерживаемого сценария освещения)
eColorModel	COLOR_SETTING_MODEL_TYPE	Тип схемы изменения цвета
stTemperature_k	COLOR_SETTING_TEMPERATURE_K_TYPE	Параметры температуры цвета в кельвинах (например, огненный белый – 1500, белый – 4500, холодный белый – 6500)
aeScenes	ARRAY [1..18] OF COLOR_SETTING_SCENE_TYPE	Массив объектов scene , которые описывают темы и сценарии освещения

2.5.7 Перечисление COLOR_SETTING_MODEL_TYPE

Перечисление **COLOR_SETTING_MODEL_TYPE** описывает схемы изменения цвета для умения типа **color_setting**.

Таблица 2.5.7 – Описание элементов перечисления **COLOR_SETTING_MODEL_TYPE**

Название	Значение	Описание
color_model_hsv	1	Изменение цвета в соответствии со схемой HSV
color_model_rgb	2	Изменение цвета в соответствии со схемой RGB

2.5.8 Структура COLOR_SETTING_TEMPERATURE_K_TYPE

Структура **COLOR_SETTING_TEMPERATURE_K_TYPE** описывает параметры температуры цвета в кельвинах для умения типа **color_setting**.

Таблица 2.5.8 – Описание элементов структуры **COLOR_SETTING_TEMPERATURE_K_TYPE**

Название	Тип	Описание
udiMin	UDINT	Минимальная допустимая температура цвета в кельвинах
udiMax	UDINT	Максимальная температура цвета в кельвинах

2.5.9 Перечисление COLOR_SETTING_SCENE_TYPE

Перечисление **COLOR_SETTING_SCENE_TYPE** описывает сценарии освещения для умения типа **color_setting**.

Таблица 2.5.9 – Описание элементов перечисления COLOR_SETTING_SCENE_TYPE

Название	Значение	Описание
scene_alarm	1	Тревога
scene_alice	2	Алиса
scene_candle	3	Свеча
scene_dinner	4	Обед
scene_fantasy	5	Фантазия
scene_garland	6	Гирлянда
scene_jungle	7	Джунгли
scene_movie	8	Кино
scene_neon	9	Неон
scene_night	10	Ночь
scene_ocean	11	Океан
scene_party	12	Вечеринка
scene_reading	13	Чтение
scene_rest	14	Отдых
scene_romance	15	Романтика
scene_siren	16	Сирена
scene_sunrise	17	Рассвет
scene_sunset	18	Закат

2.5.10 Структура RANGE_PARAMETERS

Структура **RANGE_PARAMETERS** описывает параметры умения типа **range**.

Таблица 2.5.10 – Описание элементов структуры RANGE_PARAMETERS

Название	Тип	Описание
xRandom_access	BOOL	Возможность устанавливать произвольные значения функции. Если эта возможность выключена, пользователю будет доступно только последовательное изменение значений — в большую или меньшую сторону (например, изменение громкости телевизора при работе через ИК пульт)
xRangeNone	BOOL	TRUE – отсутствует поле range (объект с описанием диапазона значений функции)
eUnit	RANGE_UNIT_TYPE	Единицы измерения значений функции
stRange	RANGE_TYPE	Объект с описанием диапазона значений функции

2.5.11 Перечисление RANGE_UNIT_TYPE

Перечисление **RANGE_UNIT_TYPE** описывает единицы измерения для умения типа **range**.

Таблица 2.5.11 – Описание элементов перечисления RANGE_UNIT_TYPE

Название	Значение	Описание
range_unit_none	1	Отсутствует единица измерения. Отсутствует для функции: CAPABILITY_FUNCTION.range_channel и CAPABILITY_FUNCTION.range_volume
range_unit_percent	2	Проценты
range_unit_celsius	3	Градусы Цельсия
range_unit_kelvin	4	Кельвины

2.5.12 Структура RANGE_TYPE

Структура **RANGE_TYPE** описывает параметры диапазона значений функции умения типа **range**.

Таблица 2.5.12 – Описание элементов структуры RANGE_TYPE

Название	Тип	Описание
rMin	REAL	Минимальное допустимое значение (зависит от функции умения)
rMax	REAL	Максимальное допустимое значение (зависит от функции умения)
rPrecision	REAL	Минимальный шаг изменения значений внутри диапазона. Значение по умолчанию: 1.0

2.5.13 Структура MODE_PARAMETERS

Структура **MODE_PARAMETERS** описывает параметры умения типа **mode**.

Таблица 2.5.13 – Описание элементов структуры MODE_PARAMETERS

Название	Тип	Описание
usiCntMode	USINT(1..71)	Количество поддерживаемых объектов в массиве modes (количество поддерживаемых режимов работы функции умения)
aeModes	ARRAY [1..71] OF MODE_TYPE	Массив объектов, которые описывают режимы работы функции

2.5.14 Перечисление MODE_TYPE

Перечисление **MODE_TYPE** описывает режимы работы для умения типа **mode**.

Таблица 2.5.14 – Описание элементов перечисления MODE_TYPE

Название	Значение	Описание
mode_auto	1	Автоматический режим
mode_eco	2	Экономичный режим
mode_turbo	3	Турбо
mode_cool	4	Охлаждение
mode_dry	5	Режим осушения
mode_fan_only	6	Вентиляция
mode_heat	7	Обогрев
mode_preheat	8	Подогрев (разогрев, предварительный нагрев, предварительный разогрев)
mode_high	9	Высокая скорость
mode_low	10	Низкая скорость
mode_medium	11	Средняя скорость
mode_max	12	Максимальный
mode_min	13	Минимальный
mode_fast	14	Быстрый
mode_slow	15	Медленный
mode_express	16	Экспресс
mode_normal	17	Нормальный (обычный)
mode_quiet	18	Тихий (ночной)
mode_horizontal	19	Горизонтальный
mode_stationary	20	Неподвижный (статичный, фиксированный)
mode_vertical	21	Вертикальный
mode_one	22	Первый
mode_two	23	Второй
mode_three	24	Третий
mode_four	25	Четвертый
mode_five	26	Пятый
mode_six	27	Шестой
mode_seven	28	Седьмой
mode_eight	29	Восьмой
mode_nine	30	Девятый
mode_ten	31	Десятый
mode_americano	32	Американо
mode_cappuccino	33	Капучино
mode_double_espresso	34	Двойной эспрессо
mode_espresso	35	Эспрессо
mode_latte	36	Латте
mode_black_tea	37	Черный чай
mode_flower_tea	38	Цветочный чай
mode_green_tea	39	Зеленый чай
mode_herbal_tea	40	Травяной чай

Название	Значение	Описание
mode_oolong_tea	41	Чай улун
mode_puerh_tea	42	Чай пуэр
mode_red_tea	43	Красный чай
mode_white_tea	44	Белый чай
mode_glass	45	Мойка стекла
mode_intensive	46	Интенсивный
mode_pre_rinse	47	Ополаскивание
mode_aspic	48	Холодец
mode_baby_food	49	Детское питание
mode_baking	50	Выпечка
mode_bread	51	Хлеб
mode_boiling	52	Варка
mode_cereals	53	Крупы
mode_cheesecake	54	Чизкейк
mode_deep_fryer	55	Фритюр
mode_dessert	56	Десерты
mode_fowl	57	Дичь
mode_frying	58	Жарка
mode_macaroni	59	Макароны
mode_milk_porridge	60	Молочная каша
mode_multicooker	61	Мультиповар
mode_pasta	62	Паста
mode_pilaf	63	Плов
mode_pizza	64	Пицца
mode_sauce	65	Соус
mode_slow_cook	66	Томление
mode_soup	67	Суп
mode_steam	68	Пар
mode_stewing	69	Тушение
mode_vacuum	70	Вакуум
mode_yogurt	71	Йогурт

2.5.15 Структура ON_OFF_STATE

Структура **ON_OFF_STATE** описывает состояние умения типа **on_off**.

Таблица 2.5.15 – Описание элементов структуры **ON_OFF_STATE**

Название	Тип	Описание
xValue	BOOL	TRUE — устройство включено; FALSE — устройство выключено

2.5.16 Структура TOGGLE_STATE

Структура **TOGGLE_STATE** описывает состояние умения типа **toggle**.

Таблица 2.5.16 – Описание элементов структуры **TOGGLE_STATE**

Название	Тип	Описание
xValue	BOOL	TRUE — функция включена; FALSE — функция выключена

2.5.17 Структура MODE_STATE

Структура **MODE_STATE** описывает состояние умения типа **mode**.

Таблица 2.5.17 – Описание элементов структуры **MODE_STATE**

Название	Тип	Описание
eValue	MODE_TYPE	Режим работы функции

2.5.18 Структура RANGE_STATE

Структура **RANGE_STATE** описывает состояние умения типа **mode**.

Таблица 2.5.18 – Описание элементов структуры **RANGE_STATE**

Название	Тип	Описание
rValue	REAL	Значение функции для данного умения
xRelative	BOOL	TRUE — новое значение функции будет рассчитываться относительно текущего путем изменения этого значения на заданное rValue ; FALSE — для функции умения будет выставлено значение rValue

2.5.19 Структура COLOR_SETTING_STATE

Структура **COLOR_SETTING_STATE** описывает состояние умения типа **color_setting**.

Таблица 2.5.19 – Описание элементов структуры **COLOR_SETTING_STATE**

Название	Тип	Описание
udiValue_rgb	UDINT(0..16777215)	Значение цвета в соответствии с моделью RGB (в диапазоне 0...16#FFFFFF)
stValue_hsv	COLOR_SETTING_VALUE_HSV_TYPE	Значение цвета в соответствии с моделью HSV
udiValue_temperature_k	UDINT	Значение температуры цвета в кельвинах
eValue_scene	COLOR_SETTING_SCENE_TYPE	Текущий сценарий освещения

2.5.20 Структура COLOR_SETTING_VALUE_HSV_TYPE

Структура **COLOR_SETTING_VALUE_HSV_TYPE** описывает значение цвета в соответствии с моделью [HSV](#) для умения типа **color_setting**.

Таблица 2.5.20 – Описание элементов структуры COLOR_SETTING_VALUE_HSV_TYPE

Название	Тип	Описание
uiH	UINT(0..360)	Цветовой тон (hue), изменяется в диапазоне 0...360 градусов
usiS	USINT(0..100)	Насыщенность (saturation), изменяется в диапазоне 0...100
usiV	USINT(0..100)	Значение цвета или яркость (value), изменяется в диапазоне 0...100

2.6 Типы данных для встроенных датчиков

2.6.1 Структура PROPERTY_OBJECT

Структура **PROPERTY_OBJECT** описывает встроенные датчики устройства. Массив таких структур размещается под указателем **pArrayPropertyObject** блока [SmartDevice](#).

Таблица 2.6.1 – Описание элементов структуры PROPERTY_OBJECT

Название	Тип	Описание
eType	PROPERTY_TYPE	Тип датчика
eFunction	PROPERTY_FUNCTION	Функция в пределах типа датчика
xRetrievable	BOOL	Признак доступности для данного устройства запроса состояния (GetState)
stParams	PROPERTY_PARAMETERS	Параметры датчика
stState	PROPERTY_STATE	Текущее состояние датчика
dtLastUpdated	DT	Время последнего обновления состояния датчика

2.6.2 Перечисление PROPERTY_TYPE

Перечисление **PROPERTY_TYPE** описывает типы встроенных датчиков.

Таблица 2.6.2 – Описание элементов перечисления PROPERTY_TYPE

Название	Значение	Описание
float	1	Отображение показаний датчиков, встроенных в устройство, в числовом формате
event	2	Отображение показаний датчиков событий, встроенных в устройство (датчики могут иметь одно из заданных состояний)

2.6.3 Перечисление PROPERTY_FUNCTION

Перечисление **PROPERTY_FUNCTION** описывает конкретную функцию в пределах типа датчика.

Таблица 2.6.3 – Описание элементов перечисления PROPERTY_FUNCTION

Название	Значение	Описание
Функции датчиков типа float		
float_amperage	1	Отображение текущего потребления тока
float_battery_level	2	Отображение уровня заряда аккумулятора
float_co2_level	3	Отображение показаний уровня углекислого газа
float_humidity	4	Отображение показаний влажности
float_illumination	5	Отображение уровня освещенности
float_pm1_density	6	Отображение уровня загрязнения воздуха частицами PM1
float_pm2_5_density	7	Отображение уровня загрязнения воздуха частицами PM2.5
float_pm10_density	8	Отображение уровня загрязнения воздуха частицами PM10

Название	Значение	Описание
float_power	9	Отображение текущей потребляемой мощности
float_pressure	10	Отображение давления
float_temperature	11	Отображение показаний температуры
float_tvoc	12	Отображение уровня загрязнения воздуха органическими веществами
float_voltage	13	Отображение текущего напряжения
float_water_level	14	Отображение показаний уровня воды
Функции датчиков типа event		
event_vibration	15	Отображение событий физического воздействия: вибрация, падение, переворачивание
event_open	16	Отображение событий открытия/закрытия дверей, окон и т. п.
event_button	17	Отображение событий нажатия кнопки
event_motion	18	Отображение событий, связанных с наличием движения в области действия датчика
event_smoke	19	Отображение событий наличия дыма в помещении
event_gas	20	Отображение событий наличия газа в помещении
event_battery_level	21	Отображение событий заряда батареи
event_water_level	22	Отображение событий, связанных с уровнем воды
event_water_leak	23	Отображение событий протечки воды

2.6.4 Структура PROPERTY_PARAMETERS

Структура **PROPERTY_PARAMETERS** описывает параметры встроенного датчика устройства. Так как структура [PROPERTY_OBJECT](#) является универсальной, то в данной структуре пользователь должен использовать только одну из вложенных структур (в зависимости от значения поля **PROPERTY_OBJECT.eType**).

Таблица 2.6.4 – Описание элементов структуры PROPERTY_PARAMETERS

Название	Тип	Описание
stFloat	FLOAT_PARAMETERS	Параметры датчика типа float
stEvent	EVENT_PARAMETERS	Параметры датчика типа event

2.6.5 Структура PROPERTY_STATE

Структура **PROPERTY_STATE** описывает текущее состояние датчика устройства. Так как структура [PROPERTY_OBJECT](#) является универсальной, то в данной структуре пользователь должен использовать только одну из вложенных структур (в зависимости от значения поля **PROPERTY_OBJECT.eType**).

Таблица 2.6.5 – Описание элементов структуры PROPERTY_STATE

Название	Тип	Описание
xNull	BOOL	TRUE – текущее состояние датчика еще не передавалось в экосистему умного дома Яндекс
stFloat	FLOAT_STATE	Текущее состояние датчика типа float
stEvent	EVENT_STATE	Текущее состояние датчика типа event

2.6.6 Структура FLOAT_PARAMETERS

Структура **FLOAT_PARAMETERS** описывает параметры датчика типа **float**.

Таблица 2.6.6 – Описание элементов структуры **FLOAT_PARAMETERS**

Название	Тип	Описание
eUnit	FLOAT_UNIT_TYPE	Единицы измерения датчика

2.6.7 Перечисление FLOAT_UNIT_TYPE

Перечисление **FLOAT_UNIT_TYPE** описывает единицы измерения для датчика типа **float**.

Таблица 2.6.7 – Описание элементов перечисления **FLOAT_UNIT_TYPE**

Название	Значение	Описание
float_unit_ampere	1	Амперы
float_unit_percent	2	Проценты
float_unit_ppm	3	Миллионные доли
float_unit_lux	4	Люксы
float_unit_density_mcg_m3	5	мкг/м3
float_unit_watt	6	Ватты
float_unit_pressure_atm	7	Атмосферы
float_unit_pressure_pascal	8	Паскали
float_unit_pressure_bar	9	Бары
float_unit_pressure_mmhg	10	Миллиметры ртутного столба
float_unit_celsius	11	Градусы Цельсия
float_unit_kelvin	12	Кельвины
float_unit_volt	13	Вольты

2.6.8 Структура EVENT_PARAMETERS

Структура **EVENT_PARAMETERS** описывает параметры датчика типа **event**.

Таблица 2.6.8 – Описание элементов структуры **EVENT_PARAMETERS**

Название	Тип	Описание
usiCntEvent	USINT(1..15)	Количество поддерживаемых объектов в массиве aeEvents (количество поддерживаемых событий функцией датчика)
aeEvents	ARRAY [1..15] OF EVENT_TYPE	Массив объектов, которые описывают события, поддерживаемые датчиком

2.6.9 Перечисление EVENT_TYPE

Перечисление **EVENT_TYPE** описывает возможные события для датчика типа **event**.

Таблица 2.6.9 – Описание элементов перечисления **EVENT_TYPE**

Название	Значение	Описание
event_tilt	1	Перевоорачивание
event_fall	2	Падение

Название	Значение	Описание
event_vibration	3	Вибрация
event_opened	4	Открыто
event_closed	5	Закрыто
event_click	6	Одиночное нажатие
event_double_click	7	Двойное нажатие
event_press	8	Долгое нажатие
event_detected	9	Обнаружено
event_not_detected	10	Не обнаружено
event_high	11	Высокий уровень
event_low	12	Низкий уровень
event_normal	13	Нормальный уровень
event_dry	14	Нет протечки
event_leak	15	Протечки

2.6.10 Структура FLOAT_STATE

Структура **FLOAT_STATE** описывает состояние датчика типа **float**.

Таблица 2.6.10 – Описание элементов структуры **FLOAT_STATE**

Название	Тип	Описание
rValue	REAL	Значение встроенного датчика типа float

2.6.11 Структура EVENT_STATE

Структура **EVENT_STATE** описывает состояние датчика типа **event**.

Таблица 2.6.11 – Описание элементов структуры **EVENT_STATE**

Название	Тип	Описание
eValue	EVENT_TYPE	Значение встроенного датчика типа event

3 Определение идентификаторов (ID) устройств

3.1 Добавление устройств Яндекс в приложение Яндекс

Перейдите в приложение Яндекс и выберите вкладку **Устройства**:

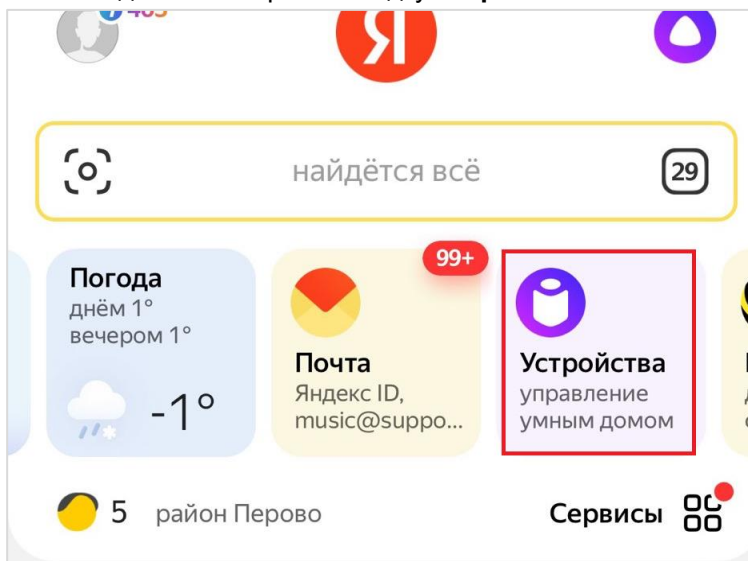


Рисунок 3.1.1 – Внешний вид приложения Яндекс

Нажмите кнопку добавления нового устройства:

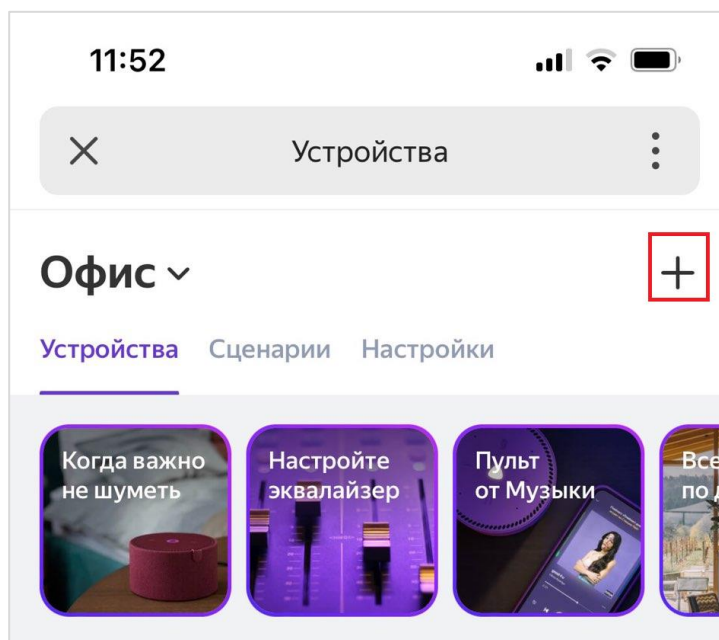


Рисунок 3.1.2 – Добавление нового устройства

Выберите нужный тип устройства:

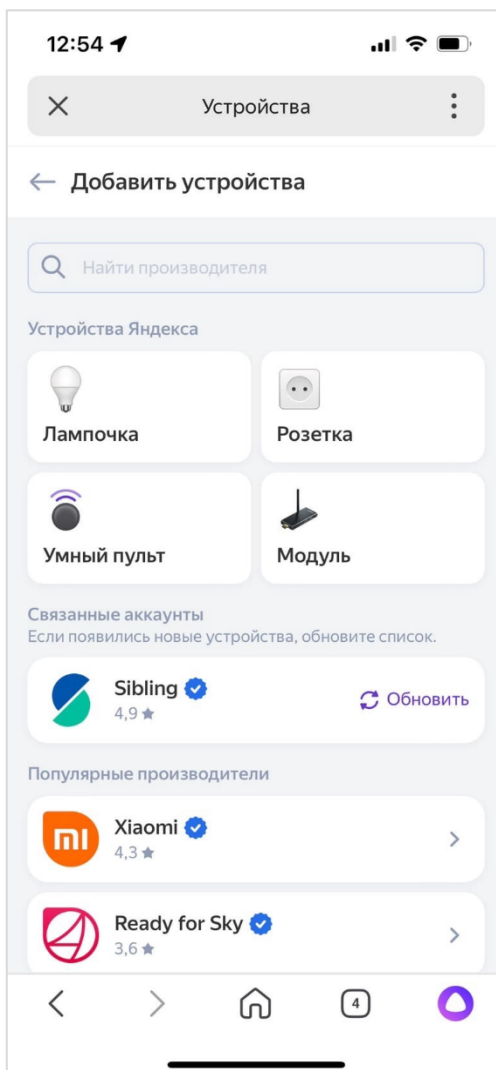


Рисунок 3.1.3 – Выбор типа устройства

Выберите дом, в который добавляется устройство:

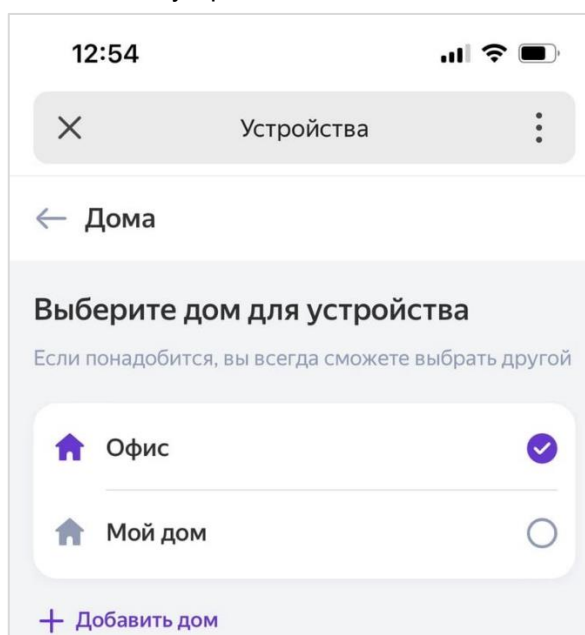


Рисунок 3.1.4 – Выбор дома

Следуя инструкциям в приложении, подключите устройство к Wi-Fi:



Рисунок 3.1.5 – Подключение устройства (1)

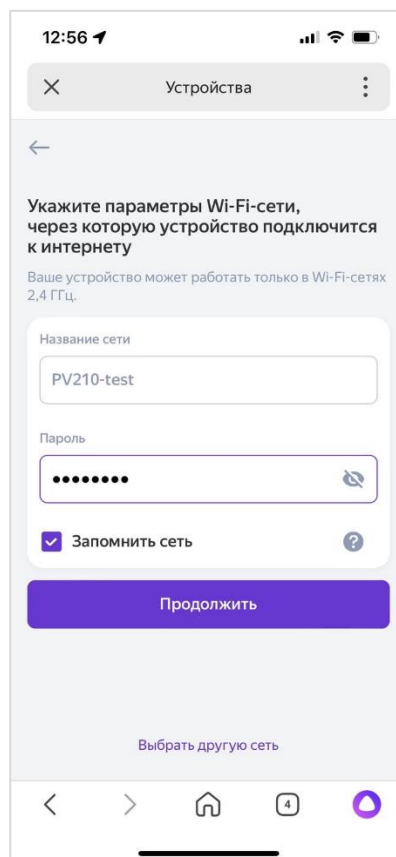


Рисунок 3.1.6 – Подключение устройства (2)

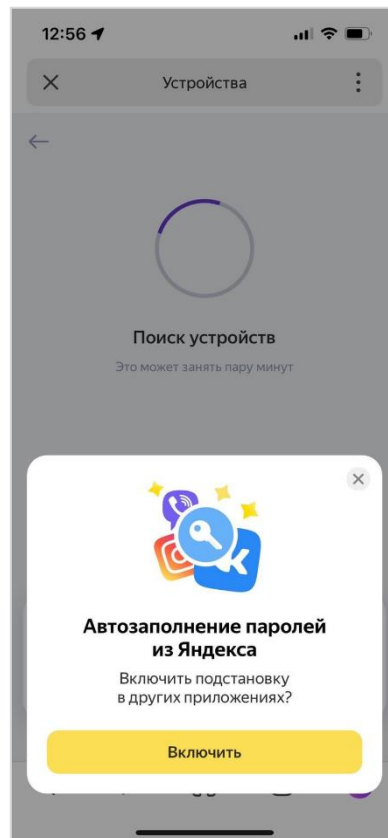


Рисунок 3.1.7 – Подключение устройства (3)

Нажмите на добавленное устройство:

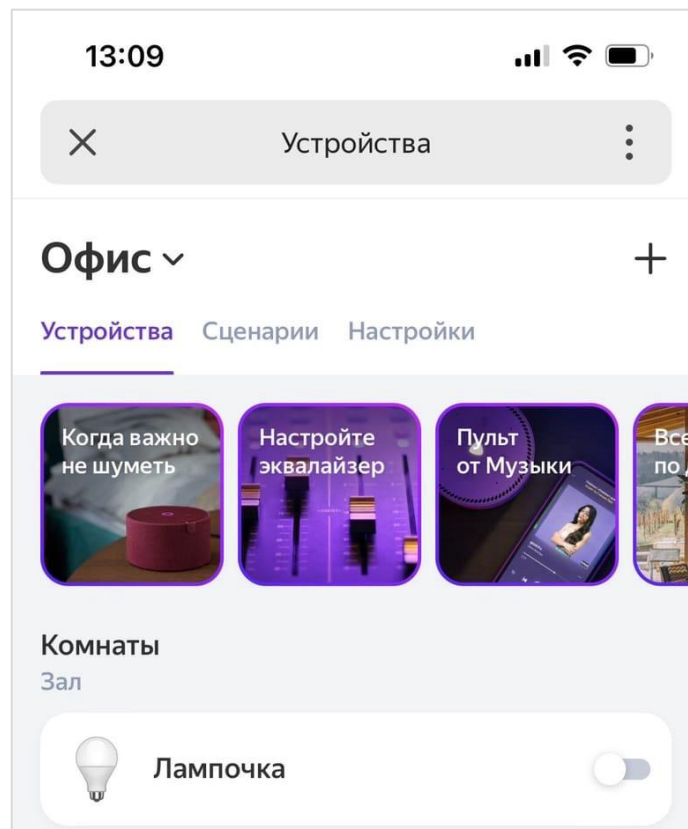


Рисунок 3.1.8 – Переход к добавленному устройству

Перейдите в его настройки:

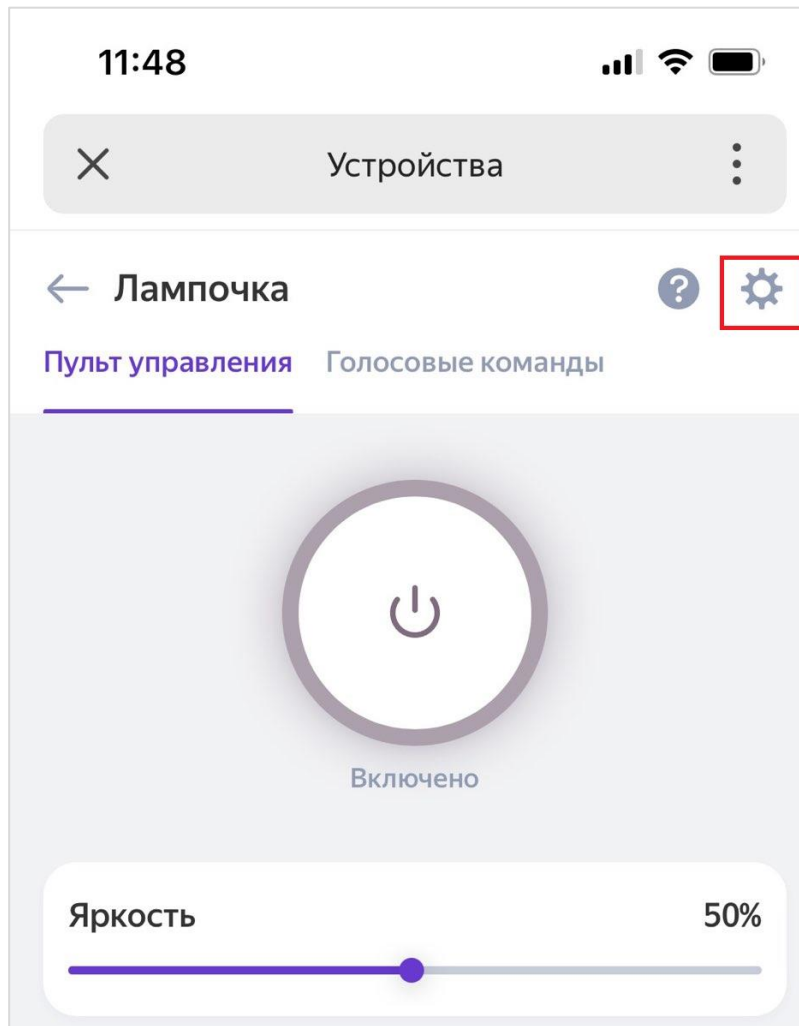


Рисунок 3.1.9 – Переход к настройкам устройства

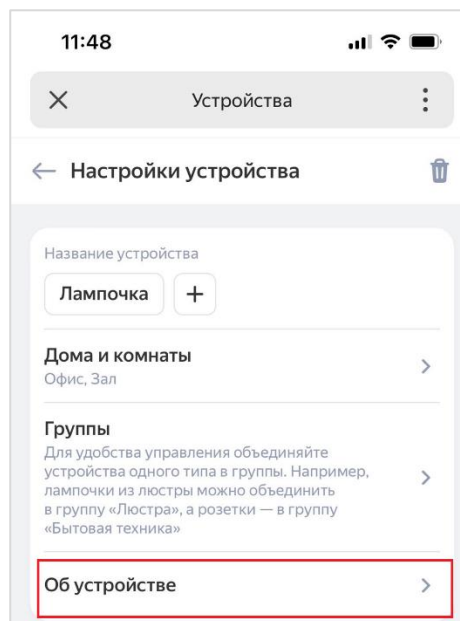


Рисунок 3.1.10 – Настройки устройства

На этой вкладке отображается идентификатор, который необходимо передать на вход **sld** ФБ [SmartDevice](#):

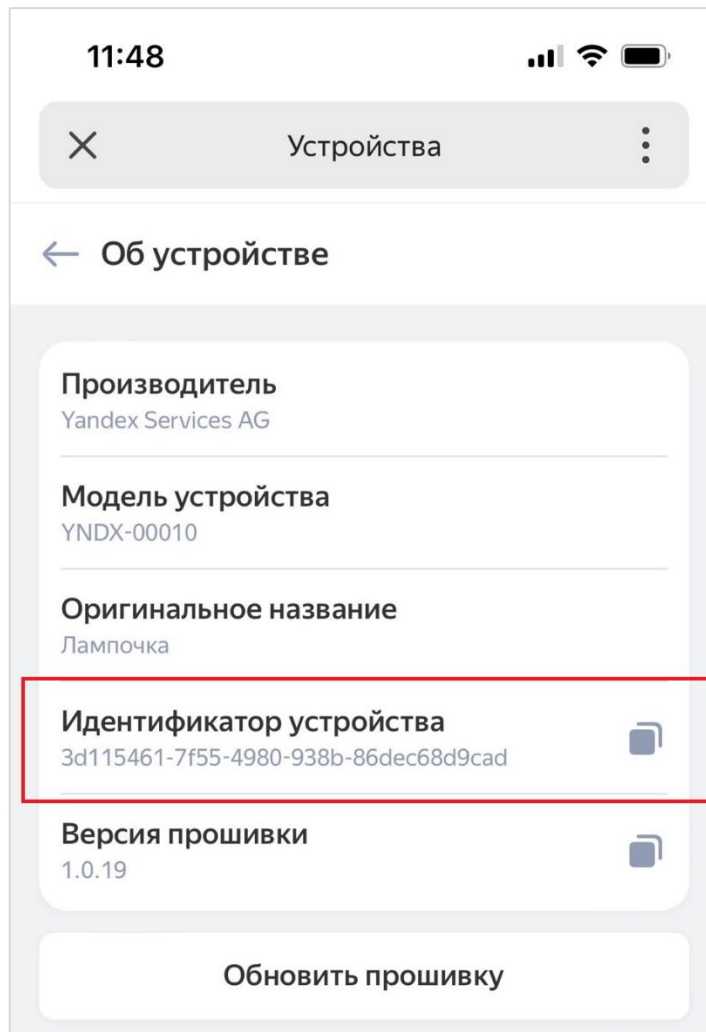


Рисунок 3.1.11 – Отображение идентификатора (ID) устройства

3.2 Добавление устройств других производителей в приложение Яндекс

Установите приложение для вашего устройства и перейдите в него.

Нажмите кнопку добавления нового устройства:

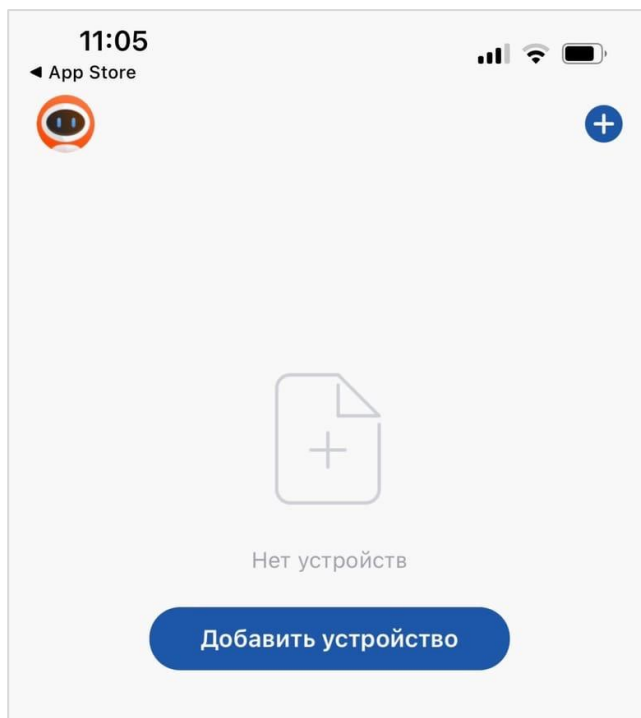


Рисунок 3.2.1 – Добавление устройства

Выберите нужный тип устройства:

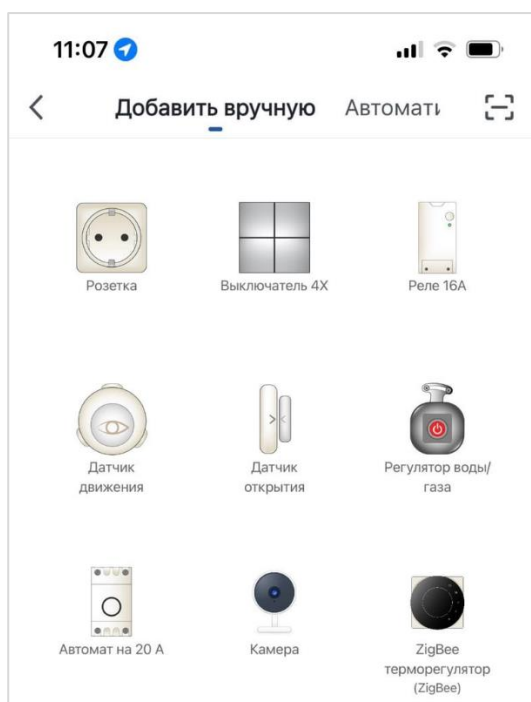


Рисунок 3.2.2 – Выбор типа устройства

Следуя инструкциям в приложении, подключите устройство к Wi-Fi:



Рисунок 3.2.3 – Подключение устройства (1)

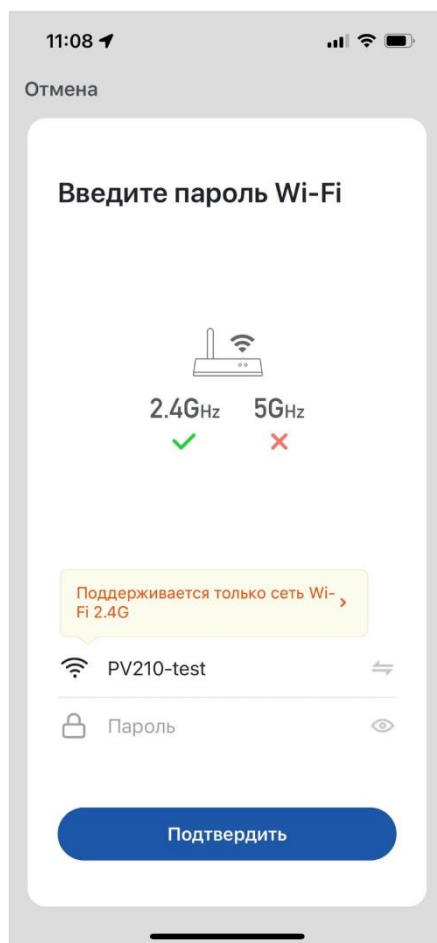


Рисунок 3.2.4 – Подключение устройства (2)

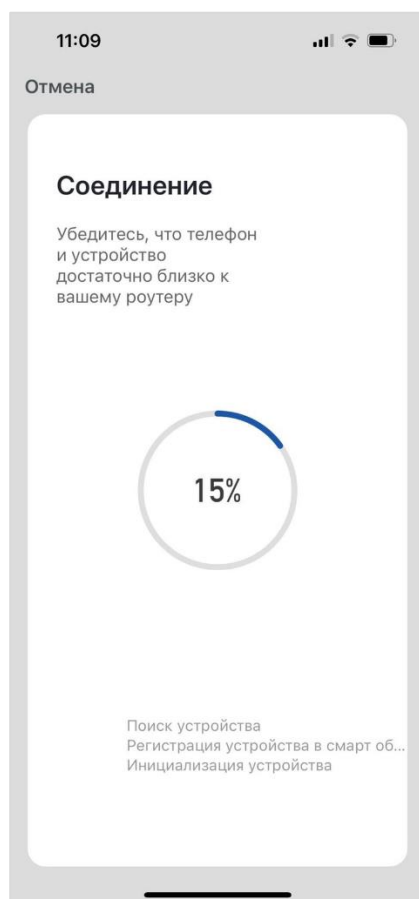


Рисунок 3.2.5 – Подключение устройства (3)

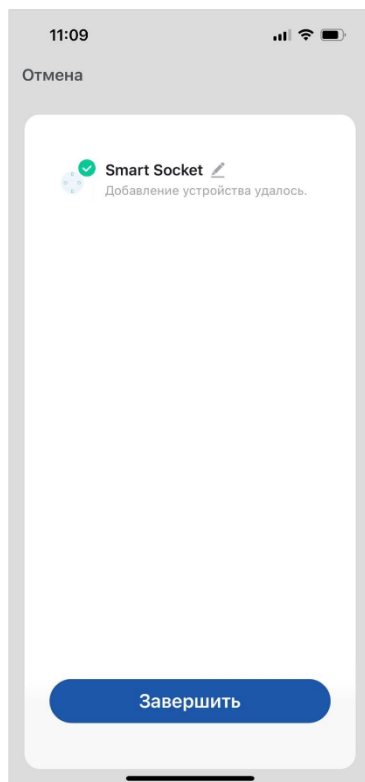


Рисунок 3.2.6 – Подключение устройства (4)

После этого перейдите в приложения Яндекс и добавьте устройства там, следуя инструкции из [п. 3.1](#) (начиная с рис. 3.1.3). Посмотрите его идентификатор (см. рис. 3.1.11) – его необходимо будет передать на вход **sld** ФБ [SmartDevice](#).

4 Получение токена

Для доступа к устройствам требуется токен.

Для его получения следует войти в аккаунт Яндекс и перейти по ссылке: <https://oauth.yandex.ru/>

Нажмите кнопку **Зарегистрировать новое приложение**:

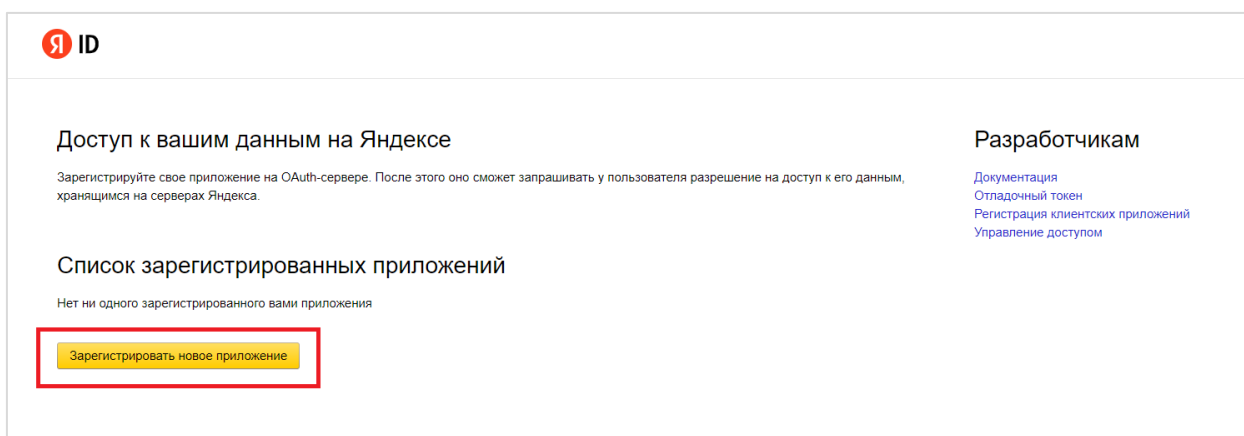


Рисунок 4.1 – Регистрация нового приложения

Введите название приложения, выберите платформу **Веб-сервисы** и нажмите кнопку **Подставить URL для разработки** (см. рис. 4.3):

The screenshot shows the 'Redaction of application' (Редактирование приложения) page. At the top, there are two tabs: 'Зарегистрированные приложения' (Registered applications) and 'Редактирование приложения' (Application editing), with the latter being active. The main heading is 'Редактирование приложения' (Application editing). Below it, a note says 'Информация о возможностях сервиса находится на сайте документации' (Information about service capabilities is on the documentation site). The form contains several fields: 'Название приложения*' (Application name*) with the value 'owen-yandex' and a note that it appears on the permissions page and in the registered applications list; 'Описание приложения:' (Application description:) with an empty text area and a note that it appears in the application list; 'Иконка приложения:' (Application icon:) with the text 'Иконка не выбрана' (Icon not selected) and a 'Загрузить' (Upload) button, and a note that it appears in the application list and on the permissions page with a 1MB limit; and 'Ссылка на сайт приложения' (Application website link) with an empty text field and a note that it appears in the application list.

Рисунок 4.2 – Настройки приложения

Платформы

Отметьте галочкой платформы, на которых будет доступно ваше приложение

iOS приложение

Android приложение

Веб-сервисы

Callback URI #1:

Адрес, на который пользователь возвращается после того, как он разрешил или отказал приложению в доступе (соответствует redirect_uri протокола OAuth)

[Подставить URL для разработки](#)

API авторизации в Яндекс.Браузере и Приложении Яндекс

Рисунок 4.3 – Выбор платформы и Callback URI (нажмите кнопку Подставить URL для разработки)

Разрешите доступ к устройствам умного дома:

Маркет для производителей

Партнёрский интерфейс Яндекса

Транспорт уведомлений push.yandex.ru

Трекер

Умный дом Яндекса

Просмотр списка устройств умного дома

Управление устройствами умного дома

Рисунок 4.4 – Разрешение доступа к устройствам умного дома

Нажмите кнопку **Сохранить изменения**:

Параметры приложения

Модерация приложения: Не требуется

Время жизни токена: Не менее, чем 1 год

После сохранения настроек существующие токены продолжат работу

Рисунок 4.5 – Сохранение изменений и получение токена

После этого на экране появится токен, который необходимо записать на вход **sToken** ФБ [SmartDevice](#).

5 Примеры

Пример управления умной лампочкой Яндекс: [Example_OwenYandexLamp_3514v1.projectarchive](https://www.projectarchive.com/Example_OwenYandexLamp_3514v1)

Пример создан в среде **CODESYS V3.5 SP14 Patch 3** и подразумевает запуск на контроллере **СПК1xx [M01]** с таргет-файлом **3.5.14.34**. В случае необходимости запуска проекта на другом устройстве следует изменить таргет-файл в проекте (**ПКМ** на узел **Device** – **Обновить устройство**).

При переносе проекта в другие версии CODESYS потребуется обновить библиотеку **OwenYandexHome** до соответствующей версии (например, для **CODESYS V3.5 SP16 Patch 3** потребуется обновить библиотеку до версии **3.5.16.x**).