

Как создать приложение промышленного Интернета вещей с помощью ThingsPro и протокола MQTT

Техническая поддержка
support@moxa.ru

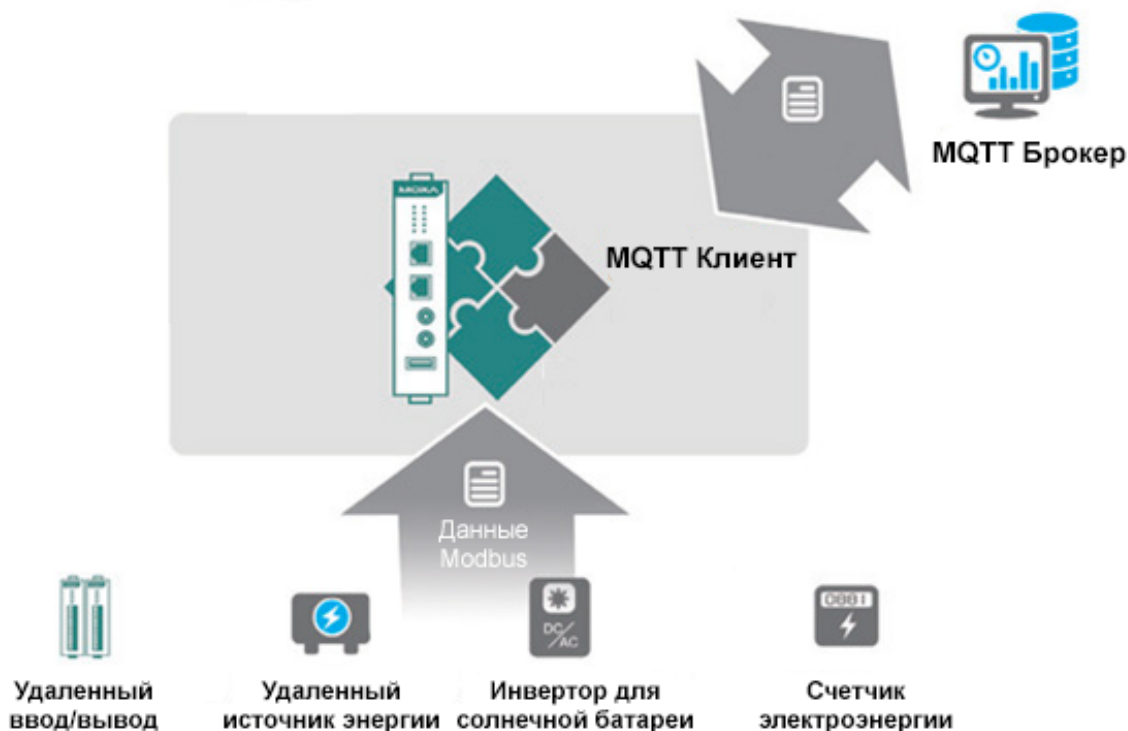
ОГЛАВЛЕНИЕ

| | |
|---------------------------------------|---|
| 1 ВВЕДЕНИЕ | 1 |
| 2 ВАРИАНТ ПРИМЕНЕНИЯ | 2 |
| 3 ИСПОЛЬЗУЕМОЕ В ПРИМЕРЕ ОБОРУДОВАНИЕ | 2 |
| 4 РЕШЕНИЯ | 2 |

Версия от 31 марта 2017 года

1 Введение

Программный пакет компании Мохэ ThingsPro™ 2.0 упрощает разработку приложений IoT и облегчает сбор данных и управление удаленными устройствами. ThingsPro обеспечивает сбор данных с конечных устройств по Modbus-протоколам, передачу информации по кабельным и беспроводным сетям связи, даёт возможность централизованного управления устройствами. Всё это позволяет пользователям сосредоточиться на разработке приложения, а не на интеграции устройств в систему.



Для пользователей, самостоятельно разрабатывающих программы для сбора данных и управления устройствами, ThingsPro предоставляет возможность передавать данные с полевых устройств Modbus на MQTT брокер без самостоятельного программирования преобразователя. В пакете ThingsPro 2.0 реализован протокол Modbus и поддерживается механизм MQTT клиент, поэтому его конфигурирование заключается лишь в настройке таблиц опроса данных с устройств и настройке подключения к устройству MQTT брокер. Затем пользователь может загружать собранные данные в MQTT брокер в фоновом режиме. В данном документе мы описываем, как настроить шаблон опроса MODBUS-устройства с помощью ThingsPro, извлечь оттуда данные и загрузить их на удаленный HTTP-сервер.

2 Вариант применения

С помощью ThingsPro 2.0 возможна отправка MQTT брокеру изменяющихся данных в режиме реального времени. В данном примере ThingsPro используется в качестве регистратора Modbus-данных, а интерфейс MQTT-клиента используется для создания соединения с MQTT брокером.

3 Используемое в примере оборудование

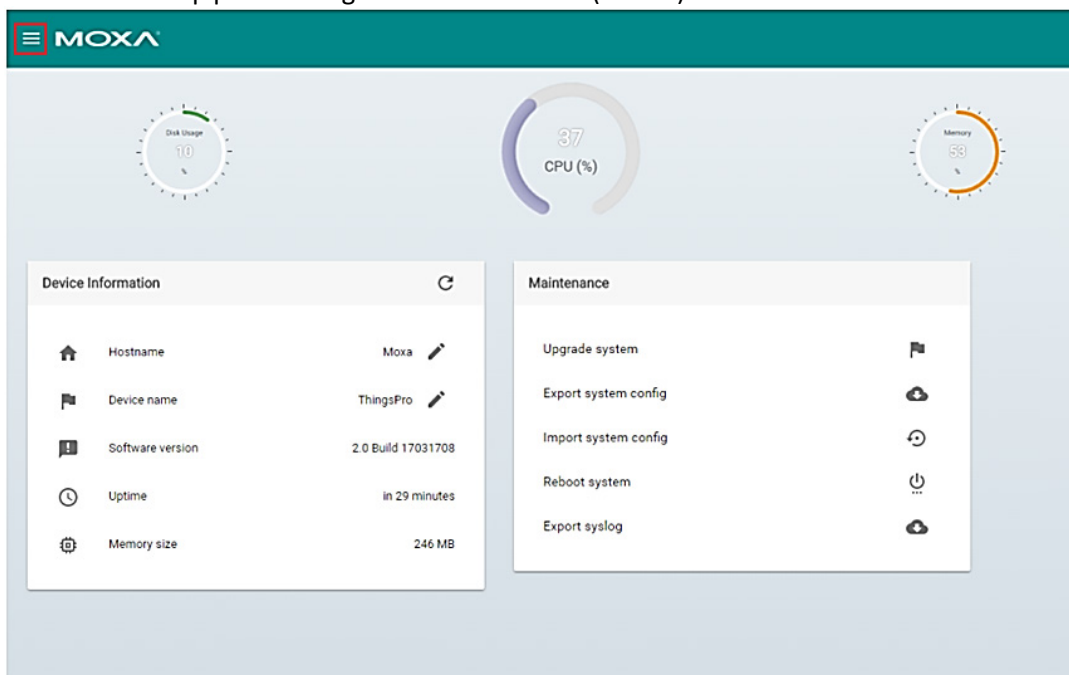
1. Компьютер UC-8112 с доступом в интернет, программный пакет ThingsPro 2.0 Gateway и SD-карта
Подробную информацию об установке компьютера UC-8112 вы можете найти в руководстве *UC-8112-ME-T Series Hardware User's Manual*.
2. ПК/ноутбук с установленными браузером Chrome и [Microsoft Device Explorer](#).
3. Modbus-устройство (в данном примере используется ioLogik E2242).

4 Решения

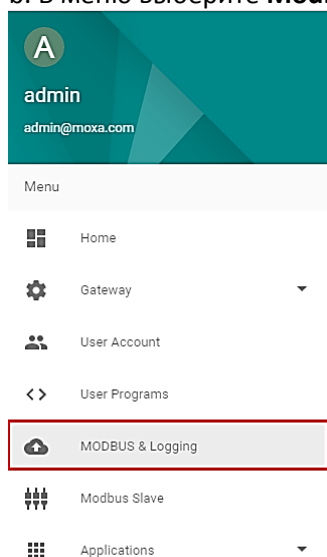
Следуйте следующим шагам для создания собственного приложения IoT с помощью ThingsPro и MQTT брокера:

1. Выбор шаблона устройства

- a. В web-интерфейсе ThingsPro нажмите  (Меню)



b. В меню выберите **Modbus & Logging**



c. Откройте вкладку **EQUIPMENT TEMPLATE**

d. Просмотрите список шаблонов (**Template List**), выберите тот, который соответствует вашему Modbus-устройству, или создайте новый.

Подробнее о создании нового шаблона – в руководстве *ThingsPro User's Manual*.

The screenshot shows the 'Template Management' interface with three tabs: EQUIPMENT TEMPLATE, MODBUS DEVICE, and LOG UPLOAD. The 'EQUIPMENT TEMPLATE' tab is active, displaying a 'Template List' table. The table has columns for Name, Tag Count, Template Action, and Tag Action. The list contains 10 entries for ioLogik-E1210 through ioLogik-E1260.

| <input type="checkbox"/> | Name ↑ | Tag Count | Template Action | Tag Action |
|--------------------------|---------------|-----------|-----------------|------------|
| <input type="checkbox"/> | ioLogik-E1210 | 16 | | |
| <input type="checkbox"/> | ioLogik-E1211 | 31 | | |
| <input type="checkbox"/> | ioLogik-E1212 | 32 | | |
| <input type="checkbox"/> | ioLogik-E1213 | 28 | | |
| <input type="checkbox"/> | ioLogik-E1214 | 18 | | |
| <input type="checkbox"/> | ioLogik-E1240 | 8 | | |
| <input type="checkbox"/> | ioLogik-E1241 | 8 | | |
| <input type="checkbox"/> | ioLogik-E1242 | 20 | | |
| <input type="checkbox"/> | ioLogik-E1260 | 6 | | |

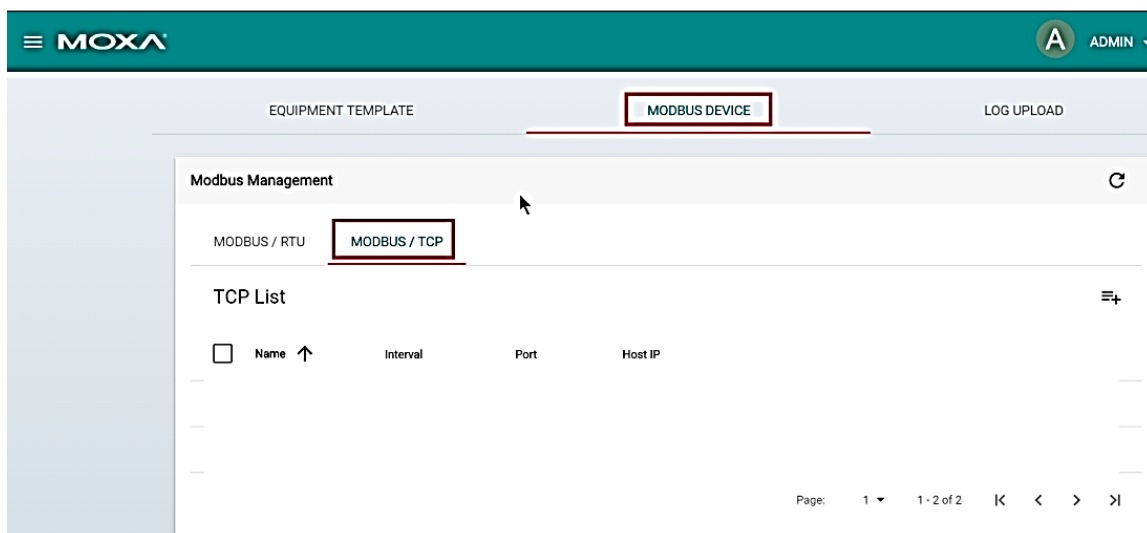
В данном документе мы используем для примера Modbus-устройство **ioLogik E2242**, которое уже указано в списке шаблонов.


2. Создание образа Modbus-устройства

a. Откройте вкладку **MODBUS DEVICE**.

b. Настройте Modbus-устройство с помощью последовательного или Ethernet-интерфейса.

Поскольку подключенное устройство (E2242) использует протокол Modbus TCP, перейдите на вкладку **MODBUS/TCP tab**.



- i. Нажмите  (Add TCP interface)
- ii. Заполните настройки TCP-интерфейса и нажмите **SAVE**.

Edit TCP Interface Settings ✕

Interface Name *
E2242

Host IP *
192.168.4.123


Port *
502

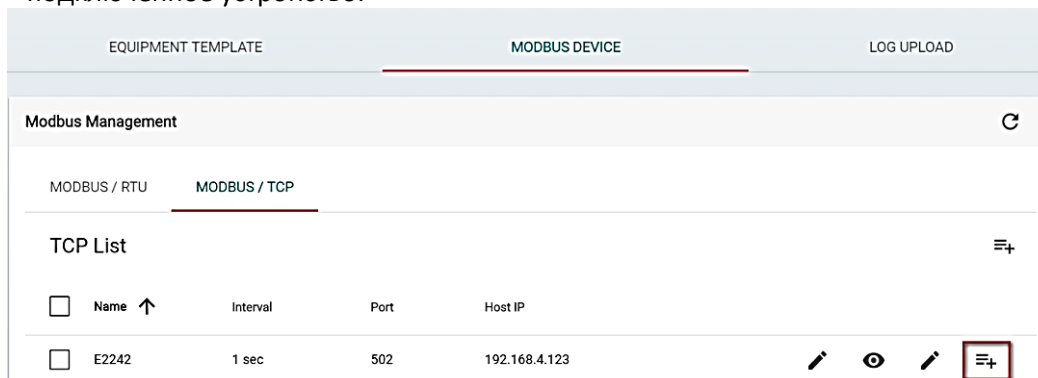
Interval *
1000

Response Timeout *
5000

SAVE

Интерфейс Modbus/TCP для E2242 будет добавлен в раздел **TCP List**.

- iii. Нажмите  (Add connected device), соответствующую E2242, чтобы добавить подключенное устройство.



- iv. Заполните имя устройства (Device Name).
- v. Выберите шаблон (Template) для совмещения интерфейса MODBUS/TCP и Modbus-устройства.
- vi. Укажите Unit ID для устройства и нажмите **SAVE**.

Device Name *
E2242

Template *
ioLogik-E2242

Unit ID *
1

TEST SAVE

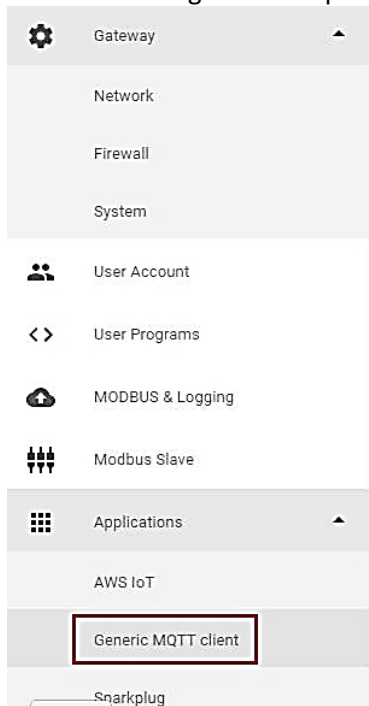
Теперь ThingsPro будет непрерывно опрашивать данное устройство.

Для проверки статуса устройств нажмите  (Show connected devices).

Если **Request Status** – зеленый, ThingsPro готов отправлять запросы на устройство Modbus TCP и получать данные с него.

3. Используйте MQTT брокер для обновления данных в режиме реального времени.

а. В меню ThingsPro выберите в разделе **Applications** подраздел **Generic MQTT client**.



б. Заполните реквизиты MQTT брокера.



Enable

Update on change

Target Host *

Port *
1883

Keepalive (sec.)
60

User Name

Password (include no., alphabet and >5 characters)

Topic *
00:90:e8:57:c0:fd

QoS
0

Client ID
00:90:e8:57:c0:fd

Retain

Clean Session

More Options

Enable TLS

SAVE

- c. Выберите параметр **Enable** (Включить).
d. Нажмите **SAVE**.

Интерфейс MQTT клиента будет использовать данную информацию для создания соединения с MQTT брокером. После установления соединения данные будут загружаться в MQTT брокер в режиме реального времени.