

Данный проект демонстрирует пример использования ПЛК NLScon-RSB в качестве MQTT-клиента для обмена данными по сети Ethernet.

Для разработки проекта использованы среда Codesys 3.5.16 patch 4 и установочные пакеты компонентов для модулей RealLab! CoDeSys Linux package.

Для работы проекта понадобятся:

- библиотека MQTT Client Library (Stefan Roßmann Engineering Solutions).

Ссылка на библиотеку*: [GitHub - rossmann-engineering/CoDeSys-MQTT-library](https://github.com/rossmann-engineering/CoDeSys-MQTT-library)

- MQTT-брокер HiveMQ Public Broker

Ссылка: [HiveMQ Dashboard \(mqtt-dashboard.com\)](https://dashboard.hivemq.com/)

- приложение для MQTT-клиента, например, MQTTX

Ссылка для скачивания: [MQTTX Download](https://mqttx.io/)

*В архиве MQTT_client.zip кроме архива проекта находится версия библиотеки MQTT Client Library с изменениями, внесенными с целью исправить ошибку при приеме сообщений (см. [Subscribe message missing first character · Issue #3 · rossmann-engineering/CoDeSys-MQTT-library · GitHub](https://github.com/rossmann-engineering/CoDeSys-MQTT-library/issues/3)).

Для использования ПЛК в качестве MQTT-клиента понадобится функциональный блок FB_MQTTClient библиотеки MQTT Client Library. Экземпляр данного ФБ в CFC-диаграммах Codesys выглядит следующим образом:



Рис. 1. Экземпляр FB_MQTTClient

Для обмена данными понадобится настроить несколько входов этого ФБ:

Вход FB_MQTTClient	Тип переменной	Пояснение	Присвоенное значение
i_xEnable	BOOL	По нарастающему фронту происходит автоматическое подключение к MQTT-брокеру	TRUE
i_sBrokerAddress	STRING	IP или URL MQTT-брокера	broker.hivemq.com
i_uiPort	UINT	Порт MQTT-брокера	1883
i_xAutoReconnect	BOOL	TRUE – автоматическое переподключение после	TRUE

		возникновения исключения	
i_sPayload	STRING	Сообщение для публикации	hello
i_sTopicPublish	STRING	Топик для публикации сообщений	mqttTest_pub
i_sTopicSubscribe	STRING	Топик для подписки	mqttTest_sub
i_xPublish	BOOL	Нарастающий фронт – публикация сообщения из i_sPayload в топик i_sTopicPublish.	
i_xSubscribe	BOOL	Нарастающий фронт – подписка на топик топик i_sTopicSubscribe.	
q_sLastReceivedMessage	STRING	Последнее сообщение, полученное от топика, на который подписался клиент	
q_sLastReceivedMessageTopic	STRING	Топик от которого пришло последнее полученное сообщение	

Вызов экземпляра FB_MQTTClient на ST выглядит следующим образом:

```
fbMqttClient(i_xEnable      := xEnable,
             i_sBrokerAddress := sBrokerAddress,
             i_uiPort        := uiPort,
             i_xAutoReconnect := TRUE,
             i_sPayload      := sPayload,
             i_sTopicPublish := sTopicPublish,
             i_sTopicSubscribe := sTopicSubscribe,
             i_xPublish      := xPublish,
             i_xSubscribe    := xSubscribe,
             q_sLastReceivedMessage => sLastReceivedMessage,
             q_sLastReceivedMessageTopic => sLastReceivedTopic);
```

Рис.2. Вызов экземпляра FB_MQTTClient

Чтобы подключиться к MQTT-брокеру, нужно знать его IP (или URL) и номер порта. В данном проекте в качестве MQTT-брокера использован [HiveMQ Dashboard \(mqtt-dashboard.com\)](https://hivemq.com/mqtt-dashboard.com/) (см. рис. 3).

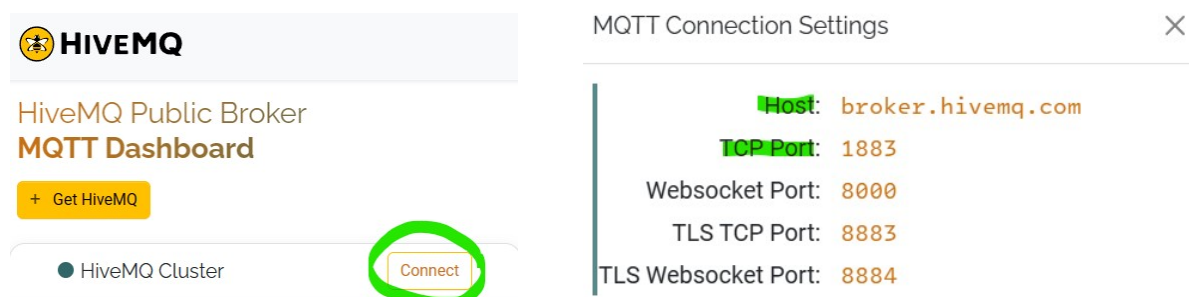


Рис. 3. Адрес и порт MQTT-брокера

После загрузки и запуска проекта на ПЛК с экрана визуализации можно будет отправлять и принимать сообщения (см. рис. 4). Для проверки работы можно выбрать один и тот же топик для подписки и для публикации. Тогда сообщение, опубликованное на вкладке

Publish (нажать на кнопку Опубликовать), должно появиться в поле на вкладке Subscribe (Нажать на кнопку Подписаться).

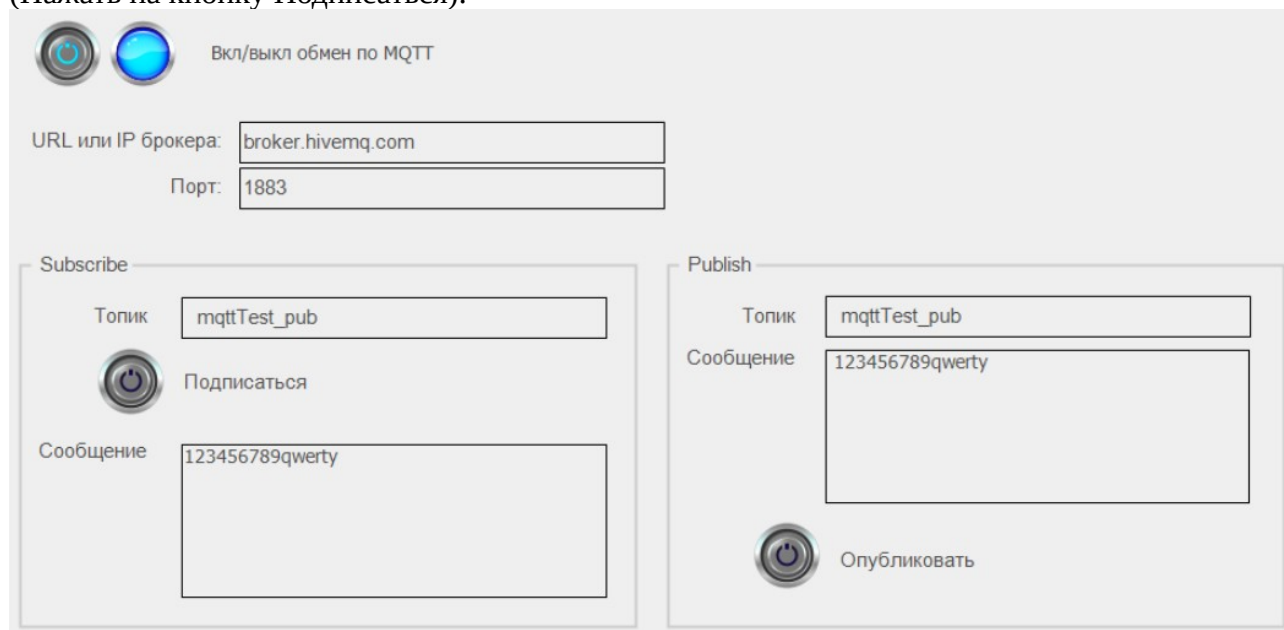


Рис. 4. Управление подпиской и публикацией с экрана визуализации на ПЛК

Для проверки работы с разными топами подписки и публикации используется еще один MQTT-клиент - приложение MQTTX на ПК. Для подключения MQTT-клиента в приложении MQTTX требуется следующее:

1. Создать новое подключение к брокеру

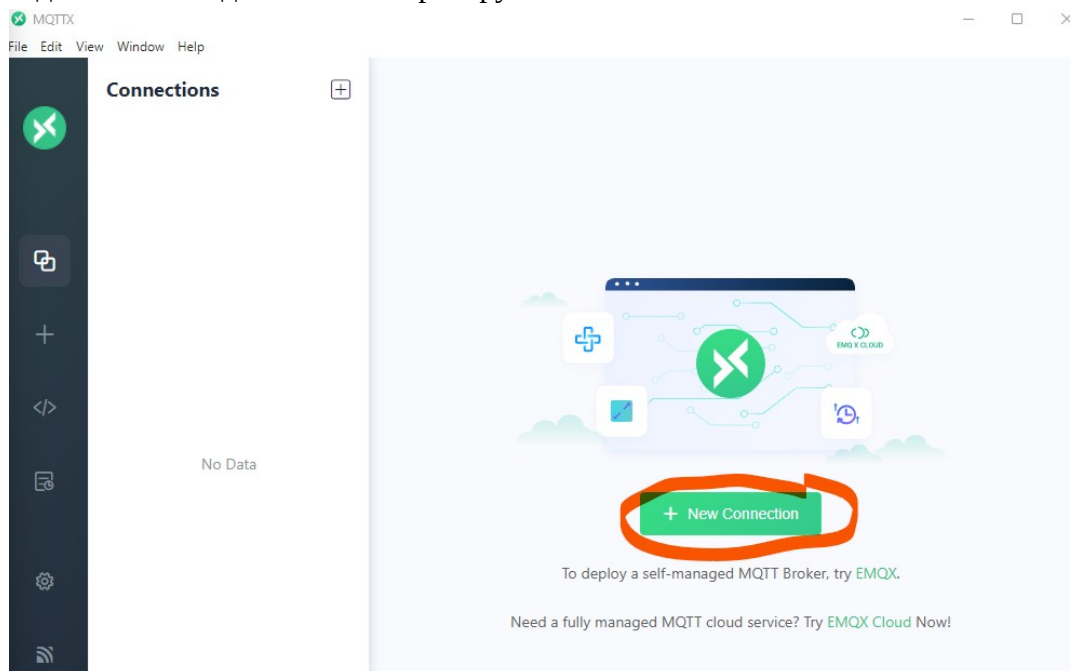


Рис. 5. Создание нового подключения в MQTTX

2. Указать имя подключения (Name), адрес брокера (Host), номер порта (Port) и версию протокола, соответствующую используемой библиотеке (MQTT Version = 3.1.1). Затем в правом верхнем углу — подключение (Connect).

Connections + < Back **New** Connect

General

* Name ⓘ

* Client ID ⓘ ⓘ

* Host

* Port ^ v

Username

Password

SSL/TLS ☐

No Data

Advanced ▲

MQTT Version v

Connect Timeout ^ v (s)

Keep Alive ^ v (s)

Auto Reconnect ☒

Reconnect Period ^ v (ms)

Clean Session ☒

Рис. 6. Настройка подключения к брокеру broker.hivemq.com

При успешном подключении появится уведомление и окно MQTTX примет вид рис. 7

Connections + **broker.hivemq.com** ✓ 0 ⏻ ✎ ...

● broker.hivemq.com@...

+ New Subscription

Plaintext v All Received Published

Рис. 7. Вид окна MQTTX при успешном подключении к брокеру

3. Для подписки на топик нужно выбрать New Subscription и указать название топика, например, mqttTest_pub. Далее — подтвердить подписку (Confirm).

* Topic

mqttTest_pub

* QoS

0 At most once

Color

#D7F6C3

Alias

Cancel Confirm

Рис. 8. Выбор топика для подписки

4. На экране визуализации во вкладке Publish сформировать сообщение для публикации и нажать кнопку Опубликовать. Сообщение должно появиться в MQTTX в области с очередью сообщений:

Connections + broker.hivemq.com 31

broker.hivemq.com@...

+ New Subscription

mqttTest_pub QoS 0

Plaintext

All Received Published

123456

2023-09-14 09:50:15:620

Topic: mqttTest_pub QoS: 0

123456789qwerty

2023-09-14 09:50:25:143

Рис. 9. Прием сообщения в MQTTX

5. Для отправки сообщения нужно использовать нижнюю часть окна MQTTX (см. рис. 10). Потребуется указать имя топика для публикации (mqttTest_sub), само сообщение и нажать на кнопку отправки. Отправленное сообщение должно появиться на экране визуализации ПЛК во вкладке Subscribe в поле Сообщение.

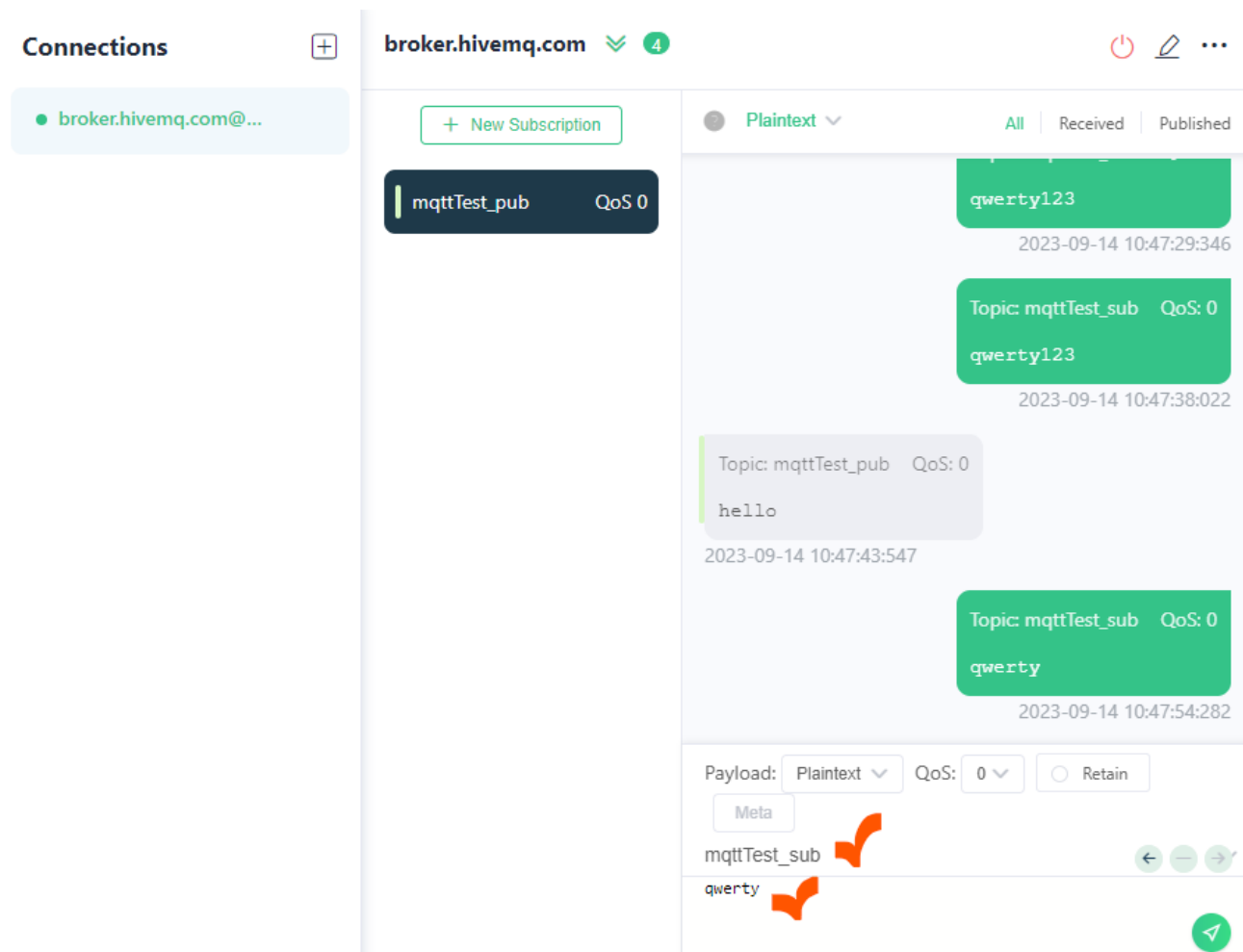


Рис. 10. Отправка сообщения из MQTTX

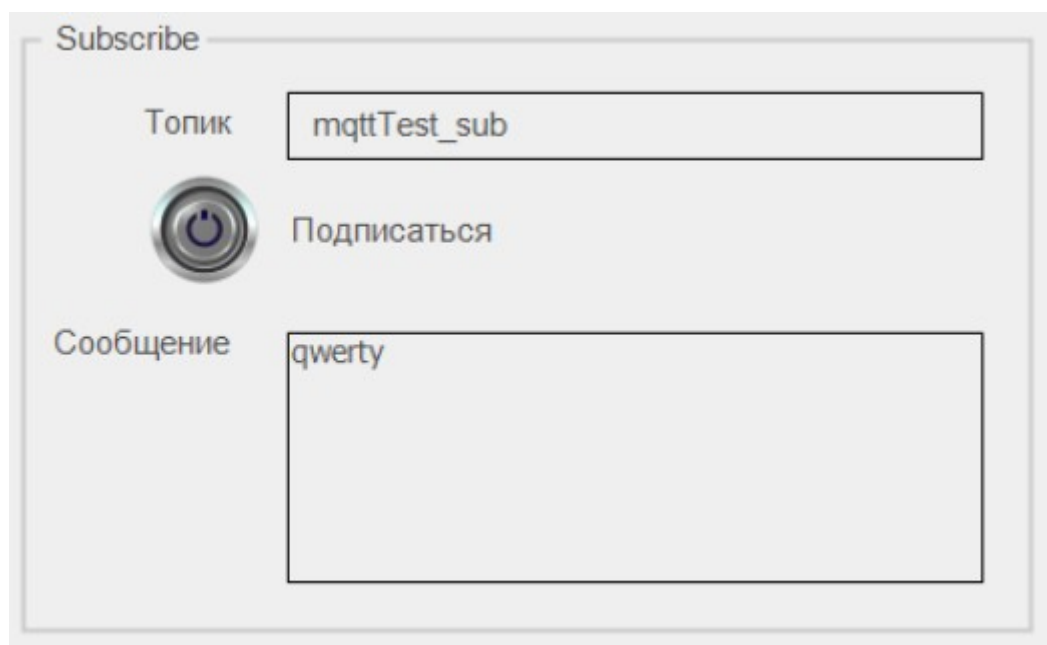


Рис. 11. Полученное сообщение с топином mqttTest_sub