

Ovladač systému REXYGEN pro MQTT
(modul MQTDrv)

Uživatelská příručka

REX Controls s.r.o.

Verze 2.50.10
Plzeň
3.9.2020

Obsah

1	MQTTDrv modul systému REXYGEN	2
1.1	Úvod	2
1.2	Instalace ovladače na cílovém zařízení	2
1.2.1	Zařízení s operačním systémem Windows	2
1.2.2	Zařízení s operačním systémem Linux	2
2	Zavedení ovladače do projektu	4
2.1	Přidání MQTTDrv ovladače	4
2.2	Konfigurační dialog MQTTDrv ovladače	4
3	Zapojení vstupů a výstupů a použití funkčních bloků ovladače v řídicím algoritmu	8
3.1	Signály přímých vstupů a výstupů ovladače	8
3.2	Funkční bloky ovladače	9
4	Příklady	10
5	Odstranění problémů	11
	Literatura	12

Kapitola 1

MQTTDrv modul systému REXYGEN

1.1 Úvod

V tomto manuálu je popsán MQTTDrv ovladač pro komunikaci přes protokol MQTT v rámci systému REXYGEN. Ovladač byl vyvinut společností REX Controls.

MQTT je jednoduchý protokol pro výměnu zpráv určený pro malé senzory a mobilní zařízení s důrazem kladeným na optimalizaci pro komunikaci po nespolehlivých sítích a sítích s vysokou latencí. Typická architektura sítě založené na MQTT komunikaci se skládá z jediné stanice typu *Broker*, jednoho či více zařízení, která produkují zprávy a tedy zastávají úlohu zařízení typu *Publisher*, a jednoho či více zařízení, která zprávy konzumují a tím pádem fungují jako zařízení typu *Subscriber*. Viz obr. 1.1.

Zprávy komunikované přes MQTT protokol jsou organizované do tématických celků – *topics*. Každá zpráva má *topic* definovaný při odeslání. *Broker* sbírá všechny zprávy ze sítě a preposílá je dál zařízením, která si o zaslání zpráv z daného *topic* zažádala. Více informací naleznete ve specifikaci MQTT protokolu [1] (pouze anglicky).

Z pohledu MQTT protokolu může REXYGEN vystupovat v roli *Publisher* a/nebo v roli *Subscriber*. REXYGEN nelze použít jako *Broker*, ale měl by být kompatibilní s jakýmkoliv zařízením typu *Broker* s podporou MQTT ve verzi 3.1.1.

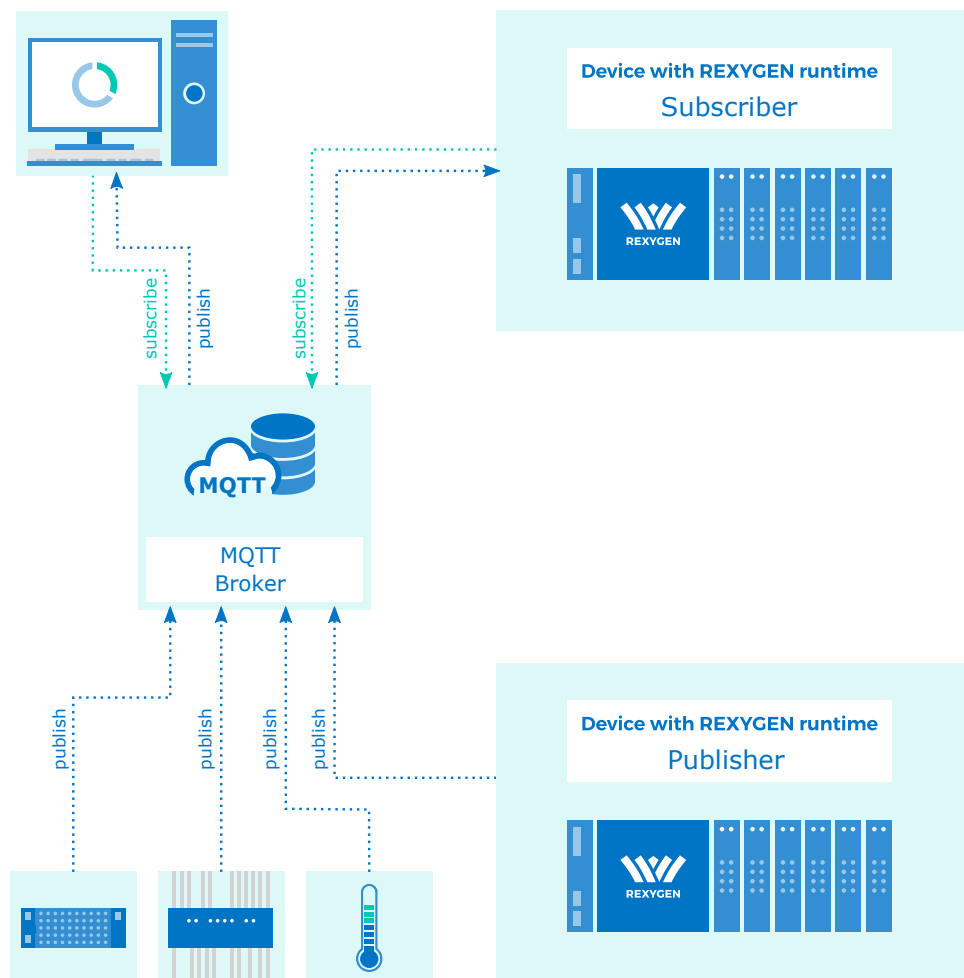
1.2 Instalace ovladače na cílovém zařízení

1.2.1 Zařízení s operačním systémem Windows

Část ovladače pro cílové zařízení, která slouží pro běh MQTTDrvna Windows 7/8/10 je součástí instalace vývojových nástrojů systému REXYGEN.

1.2.2 Zařízení s operačním systémem Linux

Pokud na cílovém zařízení nainstalovaný RexCore runtime modul, pak je nutné ho nainstalovat podle kroků popsanych v návodu *Začínáme s řídicím systémem REXYGEN* [2] Instalace obsahuje všechny dostupné ovladače včetně MQTTDrv.



Obrázek 1.1: Příklad architektury založené na MQTT protokolu

Pro instalaci ovladače odděleně může být z terminálu použit příkaz
`sudo apt-get install rex-mqttdrvt`

Kapitola 2

Zavedení ovladače do projektu

Ovladač je zaveden do projektu v momentě, kdy je ovladač přidán do hlavního souboru a vstupy a výstupy jsou zapojeny do řídicího algoritmu.

2.1 Přidání MQTTDrv ovladače

Hlavní soubor projektu se zavedeným ovladačem je zobrazen na obrázku 2.1.

Pro zavedení ovladače musí být do hlavního souboru bloku přidány 2 bloky. První z bloků je typu `MODULE` a musí být napojen na výstup `Modules` bloku `EXEC`. Blok musí být přejmenován na `MQTTDrv`.

Druhým z bloků je typu `IODRV` a musí být napojen na výstup `Drivers` bloku `EXEC` v hlavním souboru projektu. Jméno tohoto bloku (`MQTT`, viz obr. 2.1) musí být zároveň prefixem všech vstupních a výstupních signálů, které ovladač poskytuje.

Mezi nejvýznamnější parametry bloku `IODRV` patří:

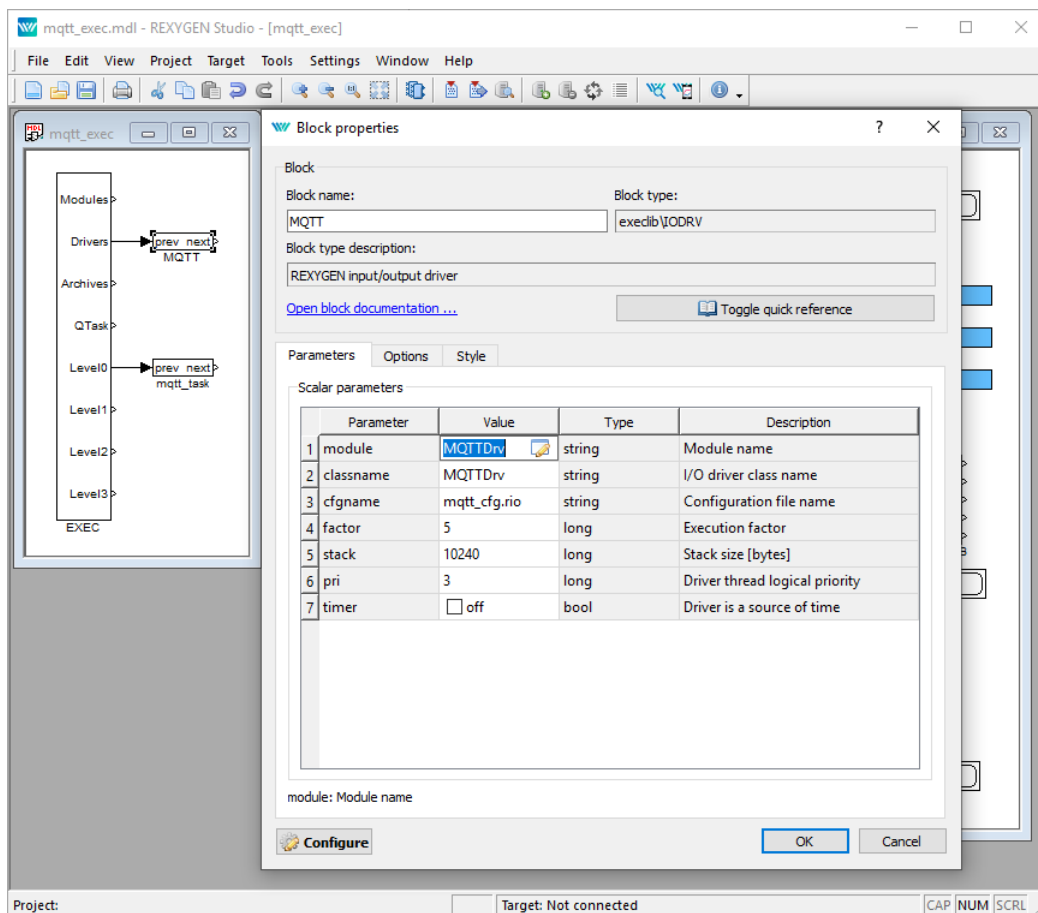
- `module` – jméno modulu, na který je ovladač napojen, v tomto případě `MQTTDrv` – je nutné dodržovat velikosti písmen!
- `classname` – třída ovladače, v tomto případě `MQTTDrv`
- `cfgname` – jméno konfiguračního souboru, např. `mqtt_cfg.rio`
- `factor` – násobek parametru `tick` bloku `EXEC` definující periodu spouštění ovladače

Výše zmíněné parametry bloku `IODRV` jsou konfigurovány v `REXYGEN Studio` programu. Konfigurační dialog je také zobrazen na obrázku 2.1.

Tlačítko `Configure` otevírá konfigurační dialog `MQTTDrv` ovladače, který je popsán v kapitole 2.2.

2.2 Konfigurační dialog MQTTDrv ovladače

Konfigurační dialog se zobrazí z `REXYGEN Studio` kliknutím na tlačítko `Configure` v parametrickém dialogu bloku `IODRV` (přejmenovaného na `MQTT`, viz kapitola 2.1)

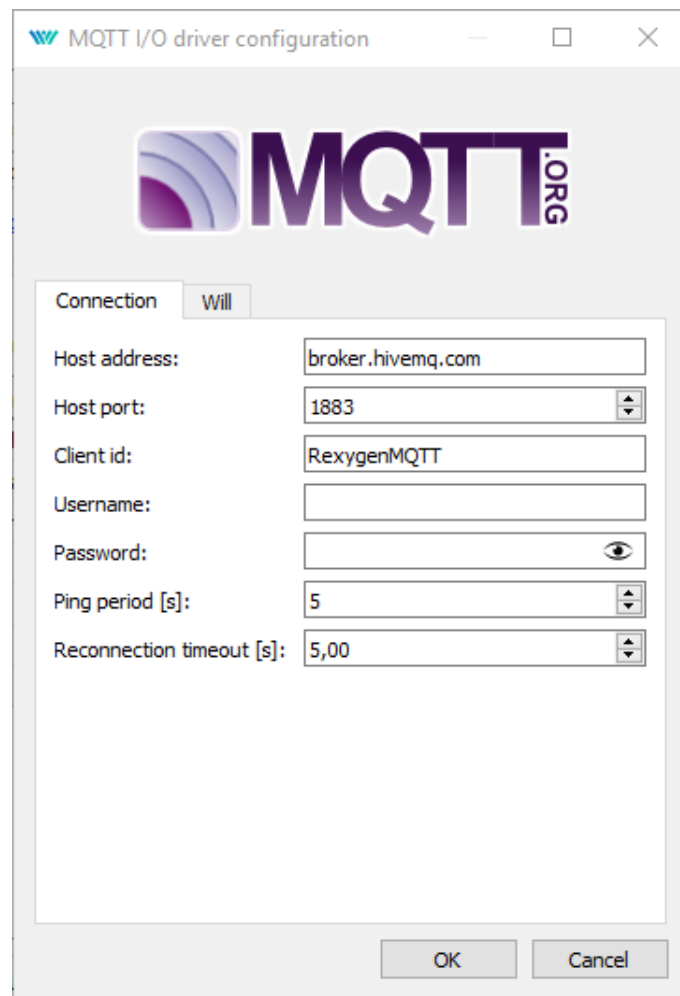


Obrázek 2.1: Příklad hlavního souboru projektu se zavedeným MQTTDrv ovladačem

Záložka *Connection* zobrazená na obr. 2.2 obsahuje konfiguraci připojení k externímu MQTT Broker zařízení. Příklady ze skupiny MQTT dodávané s instalací používají veřejně dostupná zařízení typu *Broker*. Parametr *Client id* by měl být unikátní pro každé zařízení připojené k danému zařízení typu *Broker*.

Parametry *Username* a *Password* nejsou povinné a měli by být použité pouze v případě, že zařízení typu *Broker* vyžaduje autentizaci. Další možnost, jak nastavit parametry *Username* a *Password*, je použít patřičné vstupy ovladače (viz kapitola 3).

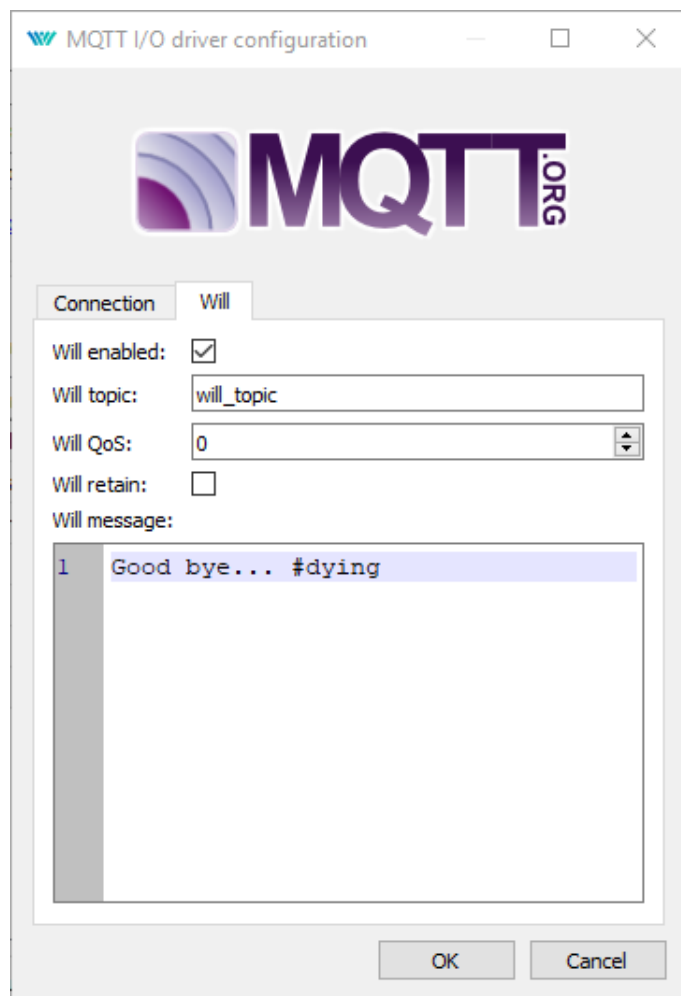
Parametr *Ping period* určuje, jak často má ovladač odesílat zprávu typu *ping* pro udržení stálé komunikace. V průběhu procesu připojování musí klient zařízení typu *Broker* určit parametr *keep-alive*. Pokud zařízení typu *Broker* po čas daný tímto parametrem nepřijme od klienta žádnou zprávu, pak by *Broker* měl končit spojení a považovat klienta za odpojeného ze sítě. Ve výchozím nastavení je tato hodnota určena jako trojnásobek hodnoty parametru *Ping period*. Pokud je hodnota parametru nastavena na 0, pak nejsou posílány žádné zprávy typu *ping* a *keep-alive* mechanismus není využíván.



Obrázek 2.2: Konfigurace MQTT připojení

Poslední parametr *Reconnection timeout* nastavuje periodu, za kterou se ovladač pokusí obnovit připojení po ztrátě spojení se zařízením typu *Broker*. Nastavení hodnoty na 0 znamená, že ovladač se pokusí navázat spojení, jakmile je to možné.

Sekce *Will* zobrazená na obr. 2.3 obsahuje nastavení odeslání zprávy poslední vůle v případě ztráty spojení s klientem. Více informací naleznete ve specifikaci MQTT protokolu [1] (pouze anglicky).



Obrázek 2.3: Konfigurace poslední vůle (*MQTT Will*)

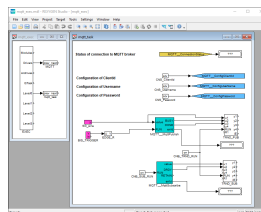
Kapitola 3

Zapojení vstupů a výstupů a použití funkčních bloků ovladače v řídicím algoritmu

Vstupy a výstupy ovladače musí být zapojeny uvnitř schémat konkrétních úloh konfigurace (v `.mdl` souborech). Samostatné úlohy (bloky `QTASK` or `TASK`) jsou napojeny na výstupy `QTask`, `Level0`, `...`, `Level13` hlavního bloku `EXEC`.

3.1 Signály přímých vstupů a výstupů ovladače

Použití vstupů a výstupů ovladače `MQTTDrv` je znázorněno na obr. 3.1.



Obrázek 3.1: Ukázka použití vstupních a výstupních vlajek `MQTTDrv` ovladače

První blok typu `From` umožňující uživateli číst informaci o stavu spojení má `Goto` tag nastavený na `MQTT__ConnectionStatus`. Bloky typu `Goto` umožňující uživateli nastavit `ClientId`, `Username` a `Password` mají `Goto` tag nastavený na `MQTT__ConfigClientId`, `MQTT__ConfigUserName` and `MQTT__ConfigPassword`. Bloky vždy musí mít jméno ovladače (doporučená hodnota je `MQTT`) jako prefix následovaný dvěma `_` (podtržítka) znaky na začátku tagu.

Všechny vstupní a výstupní vlajky `MQTTDrv` ovladače jsou dostupné v příkladu 0407-00, který je součástí instalace vývojových nástrojů `REXYGEN`. Nejaktuálnější verze pří-

kladů je dostupná na adrese <https://github.com/rexcontrols/REXexamples/archive/v2.50.zip>.

3.2 Funkční bloky ovladače

Ovladač sám o sobě se stará o udržování spojení se zařízením typu *Broker* a o komunikaci přes socket. Pro odeslání zpráv přes MQTT protokol je nutné použít funkční blok `MqttPublish`. Pro přijímání zpráv slouží blok `MqttSubscribe`. Použití bloků `MQTTDrv` ovladače demonstruje obr. 3.1. Bloky vždy musí mít jméno ovladače (doporučená hodnota je `MQTT`) jako prefix následovaný dvěma `_` (podtržítka) znaky na začátku tagu. Pro více informací o blocích `MqttPublish` a `MqttSubscribe` lze nahlédnout do referenční příručky funkčních bloků [3].

Kapitola 4

Příklady

Pro snadný start s MQTT protokolem lze využít následující příklady jako referenci a jejich modifikováním lze dosáhnout cíleného chování dané aplikace.

- 0407-01 MQTT/MQTT Data Exchange – Příklad demonstruje komunikaci mezi zařízeními typu *Publisher* a *Subscriber* kdy oba jsou implementovány v systému REXYGEN.
- 0302-09 IoT Integrations/ThingSpeak MQTT API – Příklad demonstruje komunikaci mezi systémem REXYGEN a platformou *ThingSpeak IoT*. REXYGEN může zastávat roli zařízení typu *Publisher* a/nebo *Subscriber*.

Kapitola 5

Odstranění problémů

V případě, že diagnostické nástroje systému REXYGEN (např. REXYGEN Diagnostics) reportují neočekávané či chybové hodnoty vstupů a výstupů, je vhodné vyzkoušet požadovanou funkcionality mimo systému REXYGEN. Existuje velké množství volně dostupných softwarových nástrojů, které mohou být použity pro monitorování a simulaci MQTT komunikace, jako například [mqtt-spy](#). Je také vhodné zkontrolovat konfiguraci ovladače – mezi nejčastější problémy patří:

- Chybné nastavení připojení k zařízení typu *Broker*
- Chybné nastavení parametrů *topic*

V případě, že dané vstupy a výstupy fungují v kombinaci s jinými softwarovými prostředky a nefungují v kombinaci se systémem REXYGEN, prosím, reportujte nám tento problém. Preferovaný způsob je zasláním e-mailu na adresu support@rexygen.com. Připojte prosím tyto informace v popisu problému, abychom byli schopni problém vyřešit co nejdříve:

- Identifikace systému REXYGEN, který je používán. Stačí jednoduše provést export z programu REXYGEN Studio volbou v menu Target → Licensing... → Export.
- Krátký a výstižný popis problému.
- Konfigurační soubory systému REXYGEN (.mdl and .rio soubory) redukované na svou nejjednodušší podobu, která stále demonstuje problematické chování.

Literatura

- [1] OASIS. MQTT Version 3.1.1, 2014.
- [2] REX Controls s.r.o.. *Začínáme s řídicím systémem REXYGEN*, 2020. [→](#).
- [3] REX Controls s.r.o.. *Funkční bloky systému REXYGEN – Referenční příručka*, 2020. [→](#).