# Ovladač systému REXYGEN pro MQTT (modul MQTTDrv)

Uživatelská příručka

REX Controls s.r.o.

Verze 3.0.3 Plzeň 3.11.2024

# Obsah

1	MQTTDrv modul systému REXYGEN	<b>2</b>
	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
<b>2</b>	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

### 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 přeposílá je dál zařízením, která si o zasílání 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

Cást ovladače pro cílové zařízení, která slouží pro běh MQTTDrvna Windows 10/11 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 popsaných 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

### 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 tvpu MODULE a musí být papejen na výstup Modules bloku EVEC. Blok musí být

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 Studioprogramu. 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)

W r	nqtt_ex	ec.mdl	- REXYGE	N Studio	- [m	qtt_ex	ec]								_		$\times$
File	Edit	View	Project	Target	Tool	s Se	ttings	Window	Help								
	d 🗎		a 🗈	Ē ⊃	C	•	3 4		💧 🖒 🖪		<b>Б</b> Ф	🔳 💘 Y	vii 🕕 🗸				
	matt e	xec 🗌		83		Bloc	k prop:	erties						?	×		3
١٣	1 -					Block											-
						Block n	name:					Block type:					
	Modul	esp				MOTT						exedib\IC	DRV				
	Drive	ers	prev ne	axt	E	- Block t	ype des	scription:									
	Archiv	-	MOTT			REXYO	GEN inp	ut/output dr	iver								
						Open b	block do	cumentation	ı				Togale quick reference				
	QTa	ask >															
	Leve	10	• prev ne	extb		Param	neters	Options	Style								
			mqtt_ta	isk		Scala	ar parar	meters									
	Leve	11>					Par	rameter	Value	2	Т	ype	Description				
	Leve	12>				1	modul	e	MQTTDrv	<b></b>	string		Module name				
						2	classna	ame	MQTTDrv		string		I/O driver class name				
	Leve	13>				3	cfgnan	ne	mqtt_cfg.r	io	string		Configuration file name			2	
	EXEC					4 1	factor		5		long		Execution factor			2	
Ľ						5 :	stack		10240		long		Stack size [bytes]			3	
					н.	6	pri		3		long		Driver thread logical priority				
					н.	7 1	timer		off		bool		Driver is a source of time				
					н.												
					н.												
					н.												
					н.												
					н.												
						module	e: Modu	ile name									
						mouule	c. modu	ac righte									
					1	🎘 Cor	nfigure	2					ОК	Can	cel		
					-	-	-	-	-	-	-	-		-	-		
Proje	t:								Ta	rget: N	ot conne	ected			CAP	NUM S	

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 využívaný.

W MQTT I/O driver config	juration			
	<b>/Q</b> T		ORG	
Connection Will				
Host address:	broker.hivemq.com	n		
Host port:	1883		-	
Client id:	RexygenMQTT			
Username:				
Password:			۲	
Ping period [s]:	5		-	
Reconnection timeout [s]:	5,00		•	
	ОК		Cancel	]

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).

₩ MQTT I/O driver configuration - □ ×					
	Г	.ORG			
Connection Will					
Will enabled:					
Will topic: will_topic					
Will QoS: 0			-		
Will retain:					
Will message:			_		
1 Good bye #dying					
	ОК	Cano	el		

Obrázek 2.3: Konfigurace poslední vůle (MQTT~Will)

# 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,..., Level3 hlavního bloku EXEC.

#### 3.1 Signály přímých vstupů a výstupů ovladače

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

mpt.nov.ml - 707/001 basis				
The Ball Very Propel Suppl	Not beings Window Help			
0000 A 300 P	C 9950 B 555 55	.0 27 2 0.		
Sec	🖉 ngrjask		(and (a) (2)	
	Figher of connection to MOTT broker	677_0-1460.552	•	
	Configuration of Classifi Configuration of Charmanne Configuration of Planearond		templants templantena templantena	
ind) indu Edit				
			a attack	

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 \_ (podtržítko) 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 příkladů 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 \_ (podtržítko) znaky na začátku tagu. Pro více informací o blocích MqttPublish a MqttSubscribe lze nahlédnout do referenční příručky funkčních bloků [3].

# 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*.

### Odstranění problémů

V případě, že diagnostické nástroje systému REXYGEN (např. Watch režim programu REXYGEN Studio) reportují neočekávané či chybové hodnoty vstupů a výstupů, je vhodné vyzkoušet požadovanou funkcionality mimo systéme 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  $\rightarrow$  Licensing...  $\rightarrow$  Export.
- Krátký a výstižný popis problému.
- Konfigurační soubory systému REXYGEN (.mdl and .rio soubory) redukované na svou nejjednodušší podobu, která stále demonstruje problematické chování.

# Literatura

- [2] REX Controls s.r.o.. Začínáme s řídicím systémem REXYGEN, 2020.  $\rightarrow$ .
- [3] REX Controls s.r.o.. Funkční bloky systému REXYGEN Referenční příručka, 2020.
  →.

Referenční číslo dokumentace: 16843