

Control Web — Ovladač pro komunikaci s přístroji na sběrnici M-BUS

Ovladač MBUS je určen pro spojení systému **Control Web** s přístroji pro měření různých fyzikálních veličin připojenými na sběrnici M-BUS. Komunikuje přes standardní sériové rozhraní počítače RS-232 (RS-485, RS422), USB nebo prostřednictvím sítě ETHERNET.

Vlastnosti ovladače

- Čtení hodnot z měřičů fyzikálních veličin do systému **Control Web**.
- Komunikace s přístroji na sběrnici M-BUS přes speciální převodník na standardní sériové rozhraní (COM), USB nebo ETHERNET.
- Jeden ovladač komunikuje přes jedno komunikační rozhraní (COM port, USB, ETHERNET) přes převodník na sběrnici M-BUS, na které může být připojeno více zařízení. V případě, že je potřeba připojit různá zařízení na více sběrnicích M-BUS, je možno v aplikaci definovat více ovladačů.
- Načtení všech hodnot, které poskytuje připojený přístroj a možnost jejich uložení do souboru CSV pro další zpracování.

Seznam sekcí:

[Vlastnosti ovladače](#)
[Činnost ovladače](#)
[Parametry ovladače](#)
[Procedury ovladače](#)
[Kanály, mapovací soubor ovladače a datové typy](#)
[Zpracování a stavy výjimek ovladače](#)
[Chybové kódy](#)
[Příklad použití ovladače](#)

Činnost ovladače

Ovladač čte data z přístrojů na sběrnici M-BUS a předává je do aplikace systému **Control Web**. Sběrnice M-BUS je specifická, proto ovladač ke své činnosti vyžaduje hardwarový komunikační převodník pro tuto sběrnici. Existuje celá řada převodníků, které převádějí komunikaci ze standardních komunikačních rozhraní počítače na sběrnici M-BUS. Typické a ovladačem podporované převodníky jsou:

- RS-232 na M-BUS
- USB (virtuální sériový port) na M-BUS
- ETHERNET (TCP/IP) na M-BUS

Hodnoty, které získává z měřičů, mohou být v různých tvarech, podle veličiny, kterou reprezentují. Zpravidla se jedná o číselné údaje, časové údaje nebo různé textové informace. Dvoustavové (logické) veličiny se obvykle přenášejí jako čísla s hodnotou 0 nebo 1. Číselné typy mohou mít v zařízení různé formáty a rozsahy a ovladač je automaticky konvertuje do typu požadovaného uživatelem resp. tvůrcem aplikace. Jednotlivé hodnoty se do aplikace předávají prostřednictvím kanálů ovladače. Číslování kanálů je zcela volné a záleží na tvůrci aplikace jaká čísla si zvolí. Platí pravidlo, že všechny hodnoty, které jedno zařízení poskytuje, tvoří souvislou řadu (interval) kanálů. Tento interval se určí prvním a posledním členem. Ovladač umožňuje komunikaci s více zařízeními na sběrnici a pro každé zařízení je třeba definovat kanály tak, aby se jednotlivé intervaly vzájemně nepřekrývaly.

Kromě kanálů s jednotlivými hodnotami veličin v přístroji jsou ještě pro každé zařízení k dispozici speciální kanály s označením resp. identifikátorem zařízení a dále dva souhrnné kanály, které obsahují všechny veličiny ze zařízení ve formě dvou tabulek. Tabulky jsou předávány v typu **string**, který je členěn po řádcích (oddělené znaky **CR** a **LF**). Jeden řádek tvoří položku obsahující sledovanou veličinu a její popis. První tabulka je v "čitelné" podobě, kde jsou jednotlivé položky odděleny mezerami a zarovnány do sloupců pro snazší přehlednost. Druhá tabulka je určena pro programové zpracování a je ve formátu CSV, kde jsou jednotlivé položky odděleny znakem středník.

Kromě kanálů, prostřednictvím kterých lze číst data z přístrojů, má ovladač také procedury, pomocí kterých je možné z přístrojů získat data. Tyto procedury vrací identifikátor a adresu zařízení, obsah všech veličin v zařízení, mohou měnit nastavení komunikace atd.

V instalaci ovladače se nachází jednoduchá aplikace 'Config.cw' pro **Control Web**, která slouží jako konfigurator. Pro první připojení zařízení je velice výhodné tuto aplikaci použít. Tato aplikace je vytvořena bez použití kanálů a využívá pouze procedury ovladače. Funguje tak, že vyčte všechny hodnoty ze zařízení a zobrazí je v tabulce, ve které se dá jednoduše zvolit, které veličiny chceme sledovat. Zároveň se dají vytvořit konfigurační soubory PAR a DMF a dokonce celá jednoduchá aplikace, která tyto hodnoty čte a zobrazuje.

Jak postupovat

Připojit na sběrnici M-BUS pouze zařízení, se kterým chceme komunikovat nebo ostatní zařízení vypnout.

Načíst do systému **Control Web** aplikaci Config.cw.

Upravit parametry ComDriver v souboru Config.par.

Upravit parametry komunikace (rychlost, parita, ..) v souboru Config.par.

Spustit aplikaci Config.cw.

Vybrat komunikační rozhraní (COM Port / IP). Jestliže se používá převodník ETHERNET/M-BUS, je potřeba znát jeho IP adresu a IP port, na kterém komunikuje a ten zadat.

Spustit komunikaci s přístrojem (Read device).

Jestliže zařízení odpovídá, naplní se tabulka jednotlivými veličinami.

Pak je možno označit veličiny, které chceme v cílové aplikaci zpracovávat.

Zadat jméno cílové aplikace a vygenerovat soubory CW, PAR a DMF (Save files).

Načíst a spustit automaticky vygenerovanou aplikaci (Open folder).

Parametry ovladače

Pomocí souboru parametrů je možno nastavit parametry komunikace a definovat kanály ovladače, popřípadě nastavit další doplňující parametry. Celý soubor parametrů je rozdělen do několika sekcí. Název sekce je uveden v hranatých závorkách. V sekci jsou uvedeny za jménem a znakem = jednotlivé parametry. Na jednom řádku může být definován jeden parametr.

Upozornění:

U názvů sekcí a parametrů je třeba dodržet správný zápis velkých a malých písmen.

Parametry pro základní nastavení ovladače — sekce [Settings]

Tato sekce obsahuje parametry, které slouží ke konfiguraci ovladače z hlediska komunikace. Způsob zápisu parametrů a jejich možné hodnoty:

```
[Settings]
ComDriver = <LinkDll> <ComId>
NumRepeat = <N>
Timeout = <N>
InputBufferSize = <N>
OutputBufferSize = <N>
MaxExceptions = <N>
Trace = none | log | xtrace | windows
InterMessageDelay = <N>
Readback = true | false
```

Základní syntaktické kategorie použité v konfiguraci ovladače:

Objekt	Popis
<i>LinkDll</i>	řetězec obsahující jméno DLL knihovny linkového ovladače.
<i>ComId</i>	řetězec obsahující jméno komunikačního portu.
<i>N</i>	číselná hodnota. Může být zapsána dekadicky nebo hexadecimálně (ukončena znakem H).

Následuje výčet a popis jednotlivých parametrů:

ComDriver	jméno knihovny ovladače sériového rozhraní (linkové vrstvy) a jméno komunikačního portu (<i>COMx</i>), který je nainstalován v prostředí Windows. Jméno standardního ovladače je 'CWCOMM7.DLL'. Lze použít také ovladač 'CWCOMM7x.DLL'. Pro komunikaci přes ETHERNET se jako jméno <i>LinkDLL</i> uvede klíčové slovo tcpip a jako <i>ComId</i> se zadá IP adresa a IP port např. 192.168.19.160:10001 .
NumRepeat	počet opakování požadavků na komunikaci po výskytu chyby. Implicitní hodnota je 1.
Timeout	čas čekání na odezvu v milisekundách. Nedojde-li do této doby odpověď od zařízení, bude vrácena do aplikace chyba komunikace. Implicitní hodnota je 3000 ms.
InputBufferSize	velikost přijímacího komunikačního bufferu. Ovladač si zpravidla tuto hodnotu nastavuje sám. Implicitní hodnota je 1024 byte.
OutputBufferSize	velikost vysílacího komunikačního bufferu. Ovladač si zpravidla tuto hodnotu nastavuje sám. Implicitní hodnota je 1024 byte.
MaxExceptions	velikost fronty pro nezpracované výjimky od ovladače. Implicitní nastavení je 4 096.
Trace	zapnutí trasovacích výpisů komunikace. Je-li nutno sledovat komunikaci ovladače se zařízením, je možno zvolit tímto parametrem cíl trasovacích výpisů. Možné hodnoty parametru jsou:

Hodnota	Význam
none	žádné trasovací výpisy
log	výpisy jsou nasměrovány do <i>Okna zpráv</i> záložky <i>Ladicí výpisy</i> systému Control Web
xtrace	výpisy jsou nasměrovány do souboru 'XTRACE.TXT'
windows	výpisy jsou nasměrovány do ladicího okna systému Windows

Upozornění:

Trasovací výpisy zpomalují činnost aplikace, popřípadě mohou vést ke zmenšování prostoru v paměti nebo na disku. Proto se **nedoporučuje** používat trasovací výpisy ve finální aplikaci.

- InterMessageDelay** minimální časová prodleva před následujícím požadavkem na komunikaci v milisekundách. Ovladač nezačíná další komunikaci dříve, než uplyne nastavený čas od naposledy přijaté odpovědi zařízení. Nastavuje se automaticky podle komunikační rychlosti. U komunikace přes ETHETNET se neuplatňuje.
- IgnoreComOnStartup** je-li tento parametr nastaven na **true**, bude se při startu aplikace ignorovat chyba při pokusu otevřít sériový port. To je vhodné zejména, pokud je port nedostupný nebo když v okamžiku startu aplikace neexistuje (týká se hlavně virtuálních sériových portů - převodníky USB/RS-232 a podobně).

Příklad parametrů v sekci **[Settings]**:

```
[Settings]
ComDriver = CWCOMM7.DLL COM1
NumRepeat = 0
Timeout = 5000
InterMessageDelay = 250
IgnoreComOnStartup = true
Trace = none
```

Konfigurace kanálů ovladače — sekce [Channels]

Tato sekce definuje jednotlivé kanály nebo skupiny kanálů ovladače a jejich reprezentaci v prostoru veličin zařízení.

Následuje způsob zápisu parametrů a jejich možné hodnoty. Každé skupině (bloku) kanálů odpovídá jeden řádek textu:

```
[Channels]
Device = <Address>, <ChInfo>, <ChInfoCSV>, <ChId>, <ChFirstValue>, <ChLastValue>
Device = ...
...
```

Address adresa zařízení. Hodnota může být 0 až 255. Jejich význam je následující:

Hodnota	Význam
0	je zpravidla nastavena v zařízení z výroby a je nutné ji změnit, pokud jich má být na sběrnici více.
1 až 250	rozsah adres pro typické použití.
251, 252	tyto adresy se nepoužívají.
253	tato adresa určuje, že se k zařízení přistupuje pomocí tzv. síťové adresy. Ovladač tento typ zpráv nepodporuje.
254	slouží jako univerzální adresa, na kterou odpoví každé zařízení. Tuto adresu má tedy význam použít v případě, že je na sběrnici připojeno pouze jedno zařízení. Můžeme tak komunikovat se zařízením s neznámou adresou.
255	slouží jako univerzální adresa pro vyslání zprávy do všech zařízení. Na tuto zprávu žádné zařízení neodpovídá. Ovladač tento typ zpráv nepodporuje.

ChInfo

číslo kanálu, který vrátí seznam (tabulku) všech hodnot, které zařízení poskytuje. Kanál musí být typu **string**. Je formátován jako text, kde na každém řádku je jedna veličina. Jednotlivé řádky odděleny znaky CR, LF.
Řádek s veličinou obsahuje několik údajů, které blíže specifikují veličinu:

Hodnota	Význam
RECORD	pořadové číslo veličiny
TYPE	symbol S nebo N, který znamená typ veličiny, kde S = text (string), N = číslo (number)
VALUE	hodnota veličiny
UNIT	jednotka veličiny
MEDIUM	typ hodnoty, např. okamžitá (Instantaneous value), maximální (Maximum value), chybová (Value during error state), atd.
QUANTITY	popis veličiny, např. výkon (Power), objem (Volume), energie (Energy), atd.

Jednotlivé položky jsou odděleny skupinou mezer a jsou zarovnány tak aby byly přehledně zobrazeny pomocí neproporciálního fontu (např. Courier).
Příklad obsahu kanálu:

```
VARIABLE RECORDS
```

1: N 4.000000	s	Instantaneous value	Averaging Duration
2: N 4.000000	s	Instantaneous value	Averaging Duration
3: N 2252200000000.000000	J	Instantaneous value	Energy
4: N 59201.700000	m ³	Instantaneous value	Volume
5: N 0.000000	W	Instantaneous value	Power
6: N 0.000000	m ³ /h	Instantaneous value	Volume flow

ChInfoCSV číslo kanálu, který vrátí podobně jako kanál **ChInfo** seznam (tabulku) všech hodnot, které zařízení poskytuje. Kanál musí být typu **string**. Obsah řetězce má formát CSV. Položky jsou odděleny znakem středník a jednotlivé řádky jsou odděleny znaky CR, LF.
Příklad obsahu kanálu:

```
VARIABLE RECORDS
RECORD;TYPE;VALUE;UNIT;MEDIUM;QUANTITY
1;N;4.000000;s;Instantaneous value;Averaging Duration
2;N;4.000000;s;Instantaneous value;Averaging Duration
3;N;2252200000000.000000;J;Instantaneous value;Energy
4;N;59201.700000;m^3;Instantaneous value;Volume
5;N;0.000000;W;Instantaneous value;Power
6;N;0.000000;m^3/h;Instantaneous value;Volume flow
```

ChId číslo kanálu, který vrátí jméno zařízení. Kanál musí být typu **string**.

ChFirstValue číslo prvního kanálu bloku, který obsahuje všechny načtené veličiny ze zařízení. V tomto kanále bude k dispozici hodnota první veličiny ze zařízení. V následujícím kanále bude následující veličina atd. Typ kanálu se definuje v souboru **DMF** a měl by odpovídat typu veličiny. Pokud typy neodpovídají, ovladač se pokusí hodnotu převést do požadovaného typu, který je v souboru **DMF**.

ChLastValue číslo posledního kanálu bloku veličin.

Příklad parametrů v sekci **[channels]** pro dvě zařízení na sběrnici M-BUS:

```
[Channels]
Device = 1, 100, 101, 102, 1001, 1012
Device = 2, 200, 201, 202, 2001, 2076
```

Pro snazší definici souborů **PAR** a **DMF** je k dispozici konfigurátor, který se spojí se zařízením, načte jeho data a umožní výběr veličin, které chceme zpracovávat. Výsledkem jsou oba soubory a kromě toho také jednoduchá aplikace pro ověření funkčnosti ovladače. Konfigurátor je vytvořen jako aplikace v prostředí **Control Web**, která využívá tento ovladač. Zdrojový text aplikace tedy může posloužit také jako ukázka použití ovladače. Najdete je v ukázkových aplikacích pro ovladač a vede na něj také odkaz z menu Start systému Windows.

Parametry linkové komunikační vrstvy – sekce **[comm]**

Linková komunikační vrstva představuje prostředek pro přístup na standardní sériové rozhraní počítače. Tato vrstva umožňuje mimo jiné sdílet jedno sériové rozhraní více ovladači. V systému **Control Web** je linková komunikační vrstva reprezentována dvěma DLL knihovnami ('CWXLINK.DLL' a 'CWCOMM.DLL'). Pro její konfiguraci slouží v souborech parametrů ovladače sekce **[comm]**. V této sekci je možno buď přímo definovat parametry sériové komunikace nebo uvést odkaz (přesměrování) na samostatný konfigurační soubor s parametry sériové komunikace. V případě přesměrování obsahuje sekce **[comm]** pouze jediný parametr – **file**:

```
[comm]
file=c:\cw\par\comm.par
```

Struktura parametrů linkové vrstvy:

```
[comm]
file = <file>
device = <comdevice>
```

Pokud je definován parametr **device** a existuje sekce jména **comdevice**, budou konfigurační parametry přednostně získány z této sekce, jinak budou použity parametry přímo ze sekce **[comm]**.

Povinné parametry:

```
[comm]
rx_frame_buffer = <N>
tx_frame_buffer = <N>
baudrate = <N>
databits = <N>
stopbits = one | 1 | oneandhalf | 1.5 | two | 2
parity = none | no | even | mark | odd | space
cts_flow = true | false
dsr_flow = true | false
dtr_control = disable | low | enable | high | handshake | toggle | toggle_neg
rts_control = disable | low | enable | high | handshake | toggle | toggle_neg
dsr_sense = low | high
rx_interchar_timeout = <N>
```

```

rx_char_timeout = <N> | 0
rx_timeout = <N> | 0
tx_char_timeout = <N> | 0
tx_timeout = <N> | 0

```

Nepovinné parametry:

```

priority = idle | low | below_normal | normal | above_normal | high | realtime
mode = full duplex | half duplex
pre_key = <N>
hold_key = <N>
rx_buffer = <N>
tx_buffer = <N>
tx_continue_xon_xoff = true | false
tx_xon_xoff = true | false
rx_xon_xoff = true | false
xon_tresh = <N>
xoff_tresh = <N>
error_xlat = true | false
discard_null = true | false
xon_char = <N>
xoff_char = <N>
err_char = <N>
eof_char = <N>
evt_char = <N>

```

Syntaktické kategorie:

Objekt	Popis
<i>file</i>	jméno souboru u parametru file
<i>comdevice</i>	název sekce u parametru device
<i>N</i>	číselná hodnota

Popis jednotlivých parametrů:

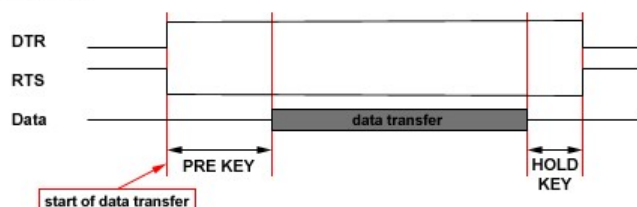
Parametr	Význam
file	soubor parametrů. Tento parametr je výhodné použít v případech, kdy více ovladačů používá jedinou linkovou vrstvu. Parametry lze potom udržovat v jediném souboru.
device	jméno zařízení, například <i>COM1</i> . Pokud je tento parametr uveden, bude použito zařízení tohoto jména. Pokud zároveň existuje sekce stejného jména, budou následující parametry získány z této sekce. Pokud sekce tohoto jména neexistuje, parametry jsou načteny z aktuální sekce [comm] .
priority	priorita komunikačního threadu. Implicitní hodnota je normal .
mode	režim činnosti (obousměrná/jednosměrná komunikace). Implicitní hodnota je full duplex .
pre_key	určuje prodlevu před zahájením přenosu. Stav signálů DTR/RTS, pokud pracují v toggle módu, je po dobu prodlevy již změněn na aktivní.
hold_key	určuje prodlevu po ukončení přenosu. Signály DTR/RTS, pokud pracují v toggle módu, zůstávají po tuto dobu prodlevy ještě aktivní.
rx_frame_buffer	velikost sekundárního bufferu pro vstup. Přípustný rozsah <96;65535>, parametr nemá implicitní hodnotu.
tx_frame_buffer	velikost sekundárního bufferu pro výstup. Přípustný rozsah <96;65535>, parametr nemá implicitní hodnotu.
rx_buffer	velikost bufferu pro vstup; rozsah <96;65535>, implicitně 4096.
tx_buffer	velikost bufferu pro výstup; rozsah <96;65535>, implicitně 4096.
baudrate	komunikační rychlost.
databits	počet data bitů.
stopbits	počet stop bitů.
parity	parita přenosu.
cts_flow	povoluje CTS řízení (handshake).
dsr_flow	povoluje DSR řízení (handshake).
dtr_control rts_control	chování DTS a RTS, význam hodnot viz dále.
dsr_sense	určuje, je-li DSR aktivní v úrovni high nebo low (high = pozitivní handshake, low = negativní handshake).
tx_continue_xon_xoff	povoluje zastavení vysílání pokud je vstupní buffer plný a byl vyslán znak XOFF (xoff_char). Pokud je nastaveno true , vysílání pokračuje od okamžiku, kdy je ve vstupním bufferu nejméně

	xoff_tresh znaků volných a ovladač vyslal XOFF znak (xoff_char) pro přibrzdění příjmu. Pokud je nastaveno false , vysílání nepokračuje do okamžiku než je ve vstupním bufferu alespoň xon_tresh znaků volných a ovladač vyslal XON znak (xon_char) pro obnovení příjmu. Implicitní hodnota je false .
tx_xon_xoff	povoluje XON/XOFF (software) řízení pro výstup. Implicitní hodnota je false .
rx_xon_xoff	povoluje XON/XOFF (software) řízení pro vstup. Implicitní hodnota je false .
xon_tresh	minimální volná kapacita vstupního bufferu pro vyslání XON. Implicitní hodnota je 50 procent.
xoff_tresh	ovlivňuje maximální počet znaků ve vstupním bufferu pro vyslání XOFF. Maximální počet přípustných znaků lze spočítat odečtením uvedené hodnoty od velikosti vstupního bufferu (rx_buffer). Implicitní hodnota je 80 procent.
error_xlat	povoluje nahrazení znaků přijatých s chybou parity znakem err_char . Pokud je nastaveno true a kontrola parity povolena, dochází k nahrazení. Implicitní hodnota je false .
discard_null	povoluje odstranění prázdných (NULL) znaků. Pokud je nastaveno true , každý NULL znak je ihned po přijetí odstraněn ze vstupního bufferu. Implicitní hodnota je false .
xon_char	dekadický kód znaku XON.
xoff_char	dekadický kód znaku XOFF.
err_char	dekadický kód znaku, který je určen pro nahrazení znaků přijatých s chybou parity.
eof_char	dekadický kód znaku EOF.
evt_char	dekadický kód znaku EVT.
rx_interchar_timeout	meziznaková prodleva pro příjem v milisekundách.
rx_char_timeout	prodleva na jeden přijatý znak v milisekundách.
rx_timeout	konstantní prodleva pro příjem v milisekundách.
tx_char_timeout	prodleva na vyslání jednoho znaku v milisekundách.
tx_timeout	konstantní prodleva pro vysílání v milisekundách.

Hodnoty pro definici chování DTS a RTS (parametry **rts_control** a **dtr_control**):

Hodnota	Význam
disable	řízení vypnuto, trvale v úrovni low
low	stejně jako disable
enable	řízení vypnuto, trvale v úrovni high
high	stejně jako enable
handshake	pozitivní handshake
toggle	signál mění stav během přenosu dat na high (pozitivní klíč), jinak trvale v low
toggle_neg	signál mění stav během přenosu dat na low (negativní klíč), jinak trvale v high

```
dtr_control = toggle
rts_control = toggle_neg
pre_key = ...
hold_key = ...
```



Chování DTR/RTS v módech toggle/toggle_neg

Upozornění:

Parametry **rx_interchar_timeout**, **rx_char_timeout** a **rx_timeout** ovlivňují propustnost na vstupu. Zvýšením hodnoty **rx_timeout** lze dosáhnout zvýšení efektivity pro rozsáhlé přenosy dat (pozor na kapacitu vstupního bufferu!), ale dojde také k prodloužení prodlev při příjmu separátních bloků dat. Nulové hodnoty parametrů způsobí nastavení optimálních hodnot podle parametrů komunikace.

Parametry **tx_char_timeout** a **tx_timeout** ovlivňují propustnost na výstupu. Zvýšením hodnot lze docílit vyšší efektivity pro rozsáhlé přenosy dat, zvláště při povoleném handshake. Příliš malé hodnoty mohou způsobit problémy při vysílání, včetně potenciálně velkého počtu neúspěšných pokusů o vyslání. Nulové hodnoty parametrů způsobí nastavení optimálních hodnot podle parametrů komunikace.

Příklad parametrů v sekci **[comm]**:

```
[comm]
baudrate = 2400
parity = even
databits = 8
stopbits = 1
rx_buffer = 1024
tx_buffer = 1024
rx_frame_buffer = 1024
tx_frame_buffer = 1024
cts_flow = disable
dsr_flow = disable
dtr_control = disable
rts_control = disable
dsr_sense = low
rx_interchar_timeout = 0
rx_char_timeout = 0
rx_timeout = 0
tx_char_timeout = 0
tx_timeout = 0
```

Procedury ovladače

Aplikace může volat ovladači určité procedury, pomocí nichž je možno řídit chování ovladače nebo získat důležité informace. To se děje prostřednictvím systémové procedury:

```
core.DriverQueryProc( DriverName : string; Param1 : any; &Param2 : any )
```

Její parametry mají následující význam:

Parametr	Význam
DriverName	symbolické jméno ovladače, definované v aplikaci,
Param1	slouží k zadání povelu (jména procedury ovladače),
Param2	je použit jako parametr procedury nebo návratová hodnota procedury.

DriverName a **Param1** se zadávají jako textové řetězce. **Param2** je různého typu podle významu procedury. Některé procedury, sloužící k nastavení parametrů, mohou skončit chybou. Výsledek naposled vykonané procedury je možno získat voláním procedury **GetResult**.

Přehled a význam jednotlivých procedur ovladače (parametr Param1)

GetVersion

Vrátí v Param2 řetězec popisující jméno a verzi ovladače.

EnableException

Voláním této procedury se povolí další výjimka od ovladače.

GetErrorCode

Vrátí číslo chyby právě zpracovávané výjimky ovladače.

GetErrorString

Vrátí popis chyby právě zpracovávané výjimky ovladače.

GetExcStatus

Vrátí kód právě zpracovávané výjimky ovladače.

SendRequest

Odešle do zařízení s adresou zadanou v parametru Param2 požadavek na přečtení všech hodnot. Jakmile zařízení odpoví, ovladač vygeneruje výjimku a pomocí procedury **ReadSummary** nebo **ReadSummaryCSV** lze vyčíst tabulku všech hodnot.

GetDeviceAddress

V parametru Param2 vrátí adresu zařízení, které odpovědělo na volání procedury **SendRequest**.

ReadId

V parametru Param2 typu **string** vrátí jméno (identifikátor) zařízení, které odpovědělo na volání procedury **SendRequest**.

ReadSummary

V parametru Param2 typu **string** vrátí tabulku všech hodnot ze zařízení, které odpovědělo na volání procedury **SendRequest**. Tabulka je formátovaná pomocí mezer.

ReadSummaryCSV

V parametru Param2 typu **string** vrátí tabulku všech hodnot ze zařízení, které odpovědělo na volání procedury

SendRequest. Tabulka je ve formátu CSV a jako oddělovač je použit znak středník.

SetTraceOutput

Nastaví cíl trasovacích výpisů komunikace. Parametr udává cíl výpisů:

Hodnota	Význam
0	žádné trasovací výpisy
1	výpisy jsou nasměrovány do ladicího okna systému Windows
2	výpisy jsou nasměrovány do <i>Okna zpráv</i> záložky <i>Ladicí výpisy</i> systému Control Web
3	výpisy jsou nasměrovány do souboru 'XTRACE.TXT'

SetLink

Nastaví jméno knihovny linkového ovladače pro otevření komunikačního rozhraní pomocí procedury **OpenCom**. Pro sériový port se použije jméno 'CVCMM7.DLL' nebo 'CVCMM7x.DLL'. Pro komunikaci přes ETHERNET se použije klíčové slovo **tcpip**.

SetDevice

Nastaví jméno COM portu pro otevření pomocí procedury **OpenCom**.

GetDevice

Vrátí aktuální jméno COM portu (nemusí se shodovat se jménem nastaveným pomocí procedury **SetDevice**).

OpenCom

Otevře COM port definovaný procedurou **SetDevice** a vrátí popis chyby.

CloseCom

Zavře aktuální COM port.

IsOpen

Vrátí aktuální stav COM portu — otevřen/zavřen.

SetComParams

Nastaví parametry COM portu. Parametr je ve tvaru: '**baud=n, data=n, parity=n, stop=n**'.

GetComParams

Vrátí aktuální parametry COM portu.

SetBaudRate

Nastaví komunikační rychlost, např. 19200.

GetBaudRate

Vrátí aktuální komunikační rychlost.

SetDataBits

Nastaví počet datových bitů komunikace. Možné hodnoty jsou 5, 6, 7, 8.

GetDataBits

Vrátí počet datových bitů.

SetParity

Nastaví paritu přenosu:

Hodnota	Význam
0 (none)	bez parity
1 (odd)	lichá
2 (even)	sudá
3 (mark)	značka
4 (space)	mezera

GetParity

Vrátí aktuální paritu přenosu. Hodnoty jsou stejné jako u procedury **SetParity**.

SetStopBits

Nastaví počet stop bitů komunikace:

Hodnota	Význam
0	1 stop bit
1	1.5 stop bitu
2	2 stop bity

GetStopBits

Vrátí aktuální počet stop bitů komunikace.

SetTimeout

Nastaví timeout komunikace v milisekundách. Je to doba, kterou ovladač čeká po odeslání požadavku na odpověď od zařízení. Pokud odpověď do této doby nepřijde, vznikne chyba timeout.

GetTimeout

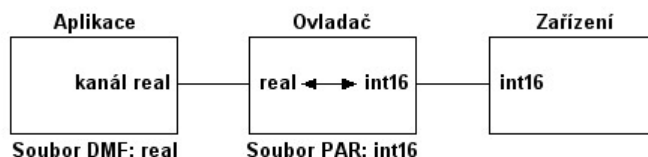
Vrátí aktuální hodnotu timeoutu komunikace.

Kanály, mapovací soubor ovladače a datové typy

Při definování kanálů ovladače je nutno kromě zápisu bloků v souboru parametrů vytvořit také mapovací soubor ('DMF') pro překlad aplikace. Tento soubor obsahuje seznam čísel kanálů ovladače a jejich typy. Ovladač používá pouze číselné typy a řetězce, protože hodnoty získané ze zařízení jsou buď čísla nebo text. Pokud je u číselných typů známý rozsah (velikost typu), je možno definovat typ kanálu podle typu veličiny v zařízení. Jestliže si nejsme jisti, je možno zadat typ **real**. Dokonce je možno i pro číselné hodnoty zadat u kanálů typ **string** a ovladač tyto hodnoty automaticky převede.

Automatická konverze typů

Ovladač automaticky konvertuje typy u veličin do typů, které jsou definovány pro odpovídající kanály v souboru 'DMF'. Např. v souboru 'DMF' mohou být všechny číselné kanály typu **real**, přestože v zařízení jsou hodnoty prezentovány třeba jako **int16**.



Příklad automatické konverze typu int16 na real.

Zpracování a stavy výjimek ovladače

Při výskytu chyby komunikace vznikne událost, kterou ovladač může přenést jako výjimku do aplikace. V aplikaci musí být definován objekt (virtuální přístroj), který má definován parametr **driver_exception** se symbolickým jménem tohoto ovladače. V tomto případě je takový přístroj výjimkou aktivován. V rámci jeho činnosti je třeba zapsat na kanál č.2 hodnotu **true** nebo zavolat proceduru ovladače **EnableException**. Tím se povolí vyvolání další výjimky, jakmile nastane nebo pokud není fronta výjimek prázdná. Přečtením kanálu č.1 je možno zjistit stav výjimky. Při použití kanálu č.2 je třeba pamatovat na to, aby měl virtuální přístroj nastaven parametr **send_same_data**. Pokud aplikace výjimky nezpracovává, přidávají se do fronty, jejíž maximální velikost je určena parametrem **MaxExceptions** v sekci **[Settings]**.

Kódy pro stavy výjimek ovladače

Hodnota	Význam
0	žádná výjimka od ovladače
1	výjimka signalizuje chybu při čtení dat. Její kód a popis je možno získat voláním procedury ovladače GetErrorCode nebo čtením kanálu č.6.
2	výjimka signalizuje chybu při zápisu dat. Její kód a popis je možno získat voláním procedury ovladače GetErrorCode nebo čtením kanálu č.6.
3	změna dat v zařízení. U asynchronních komunikací nebo v případě, že ovladač vystupuje jako podřízená stanice (slave), se takto signalizuje změna dat v zařízení.
4	chyba procedury ovladače. Vykonání některých procedur ovladače může zabrat poměrně dlouhou dobu. Proto běží zpravidla ve zvláštním threadu, aby nezdržovaly běh aplikace. Pokud procedura skončí chybou, způsobí generování výjimky s tímto kódem.

Chybové kódy

Ovladač generuje následující chybové kódy. Tyto kódy jsou k dispozici na kanále č.6. Kód a popis chyby je také možno získat voláním procedur ovladače **GetErrorCode** a **GetErrorString**. Při vzniku chyby se současně vypíše hlášení do *Okna zpráv* systému **Control Web**.

Kódy definované systémem **Control Web**:

Hodnota	Význam
0	úspěšná operace
1	chyba timeoutu

2	neplatná hodnota
3	neexistující kanál

Obecné chybové kódy ovladače:

Hodnota	Význam
1000	vnitřní chyba
1001	chyba kontrolního součtu
1002	vstupní komunikační buffer je plný
1003	nesprávná adresa stanice
1004	nesprávný kód povelu (funkce)
1005	přijatá data mají nesprávnou délku
1006	chyba potvrzení
1007	nesprávná odpověď
1008	chyba otevření COM portu
1009	chyba zavření COM portu
1010	chyba nastavení zařízení
1011	chyba nastavení COM parametrů
1012	chyba nastavení rychlosti
1013	chyba nastavení délky slova
1014	chyba nastavení parity
1015	chyba nastavení stop bitů
1016	chyba nastavení timeoutu
1017	chyba zápisu bloku
1018	výstupní fronta je plná
1019	vstupní fronta je plná
1020	nesprávný parametr procedury
1021	data nejsou k dispozici
1022	neznámá procedura ovladače
1023	nesprávný identifikátor bloku
1024	neplatný offset bloku
1025	zařízení není připojeno
1026	datový element je určen pouze pro čtení
1027	neplatný číselný rozsah datového elementu nebo nesprávný formát dat v textovém tvaru
1028	neplatné záhlaví rámce

Příklad použití ovladače

Součástí instalace je několik příkladů použití ovladače. Tyto příklady naleznete v ukázkových aplikacích ovladače, na které vede odkaz z menu Start systému Windows. Součástí příkladů jsou také soubory '*.DMF' a '*.PAR'. Před spuštěním příkladů si nejprve tyto soubory upravte podle aktuální konfigurace.