

Knihovna KnxLib

TXV 003 80.01
první vydání
prosinec 2012
změny vyhrazeny

Historie změn

Datum	Vydání	Popis změn
Prosinec 2012	1	První vydání, popis odpovídá KnxLib_v11

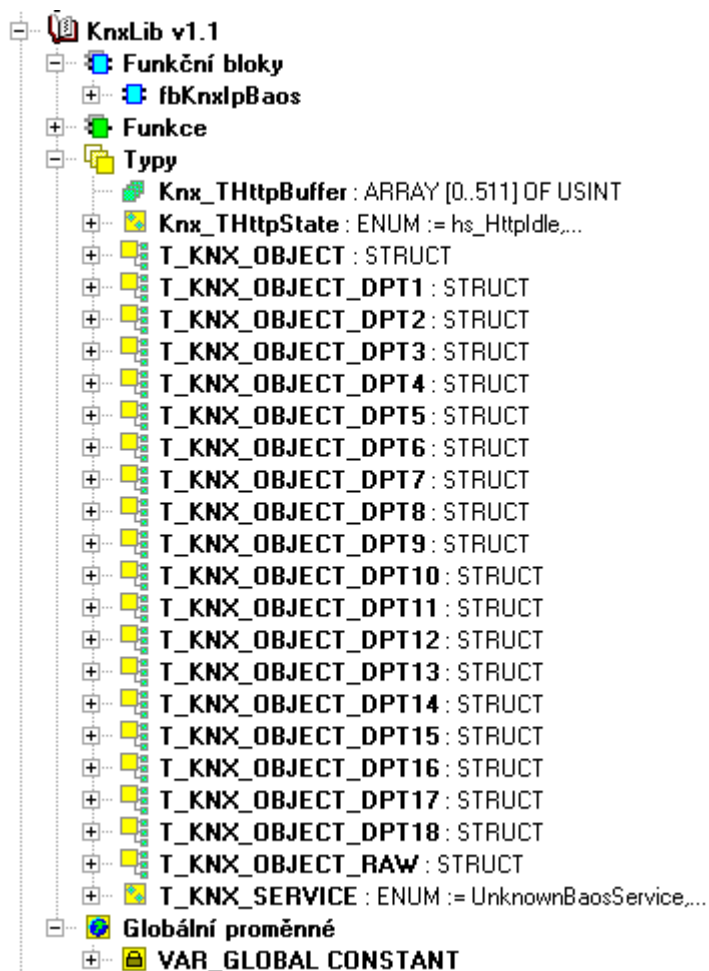
Obsah

1 Úvod	3
1.1 Propojení systémů.....	4
1.2 Princip činnosti.....	5
2 Konstanty	6
3 Datové typy	7
3.1 Typ T_KNX_OBJECT_DPT1.....	8
3.2 Typ T_KNX_OBJECT_DPT2.....	8
3.3 Typ T_KNX_OBJECT_DPT3.....	9
3.4 Typ T_KNX_OBJECT_DPT4.....	9
3.5 Typ T_KNX_OBJECT_DPT5.....	10
3.6 Typ T_KNX_OBJECT_DPT6.....	10
3.7 Typ T_KNX_OBJECT_DPT7.....	11
3.8 Typ T_KNX_OBJECT_DPT8.....	11
3.9 Typ T_KNX_OBJECT_DPT9.....	12
3.10 Typ T_KNX_OBJECT_DPT10.....	12
3.11 Typ T_KNX_OBJECT_DPT11.....	13
3.12 Typ T_KNX_OBJECT_DPT12.....	13
3.13 Typ T_KNX_OBJECT_DPT13.....	14
3.14 Typ T_KNX_OBJECT_DPT14.....	14
3.15 Typ T_KNX_OBJECT_DPT15.....	15
3.16 Typ T_KNX_OBJECT_DPT16.....	15
3.17 Typ T_KNX_OBJECT_DPT17.....	16
3.18 Typ T_KNX_OBJECT_DPT18.....	16
3.19 Typ T_KNX_OBJECT_RAW.....	17
4 Globální proměnné	17
5 Funkce	17
6 Funkční bloky	17
6.1 Funkční blok fbKnxlpBaos.....	18
7 Nastavení parametrů komunikace v PLC	22
8 Příklad použití	26

1 ÚVOD

Knihovna KnxLib je určena pro komunikaci PLC systémů Tecomat s inteligentní elektroinstalací pro řízení budov systému KNX. KNX instalace jsou k PLC připojeny prostřednictvím modulu KNX IP BAOS 772 přes rozhraní Ethernet protokolem TCP/IP.

Následující obrázek ukazuje strukturu knihovny KnxLib v prostředí Mosaic



Pokud chceme funkce z knihovny KnxLib použít v aplikačním programu PLC, je třeba nejprve přidat tuto knihovnu do projektu. Současně s knihovnou KnxLib se do projektu automaticky přidají knihovny ComLib, ToStringLib, TimeLib a JsonLibEx protože knihovna KnxLib využívá některé funkce z těchto knihoven. Knihovna KnxLib je dodávána jako součást instalace prostředí Mosaic od verze v2012.4.

POZOR !!!

Knihovna KnxLib vyžaduje, aby byl v PLC naprogramovaný aplikační profil TXF 689 09 AP KNX BAOS LICENCE. Tento profil není součástí standardního PLC a je třeba jej objednat zvlášť ke každému PLC. Bez tohoto profilu bude funkční blok pro komunikaci s ústřednou fungovat pouze omezenou dobu (4 hodiny pro testovací a ladicí účely). Poté blok přestane komunikovat a vyhlásí chybu aplikačního profilu. K obnovení funkce dojde po vypnutí a zapnutí napájení PLC (opět na omezenou dobu) nebo po naprogramování aplikačního profilu TXF 689 09 (bez časového

omezení). Podrobnosti o práci s aplikačními profily viz dokumentace TXV 003 39.01 Aplikační profily pro PLC Tecomat.

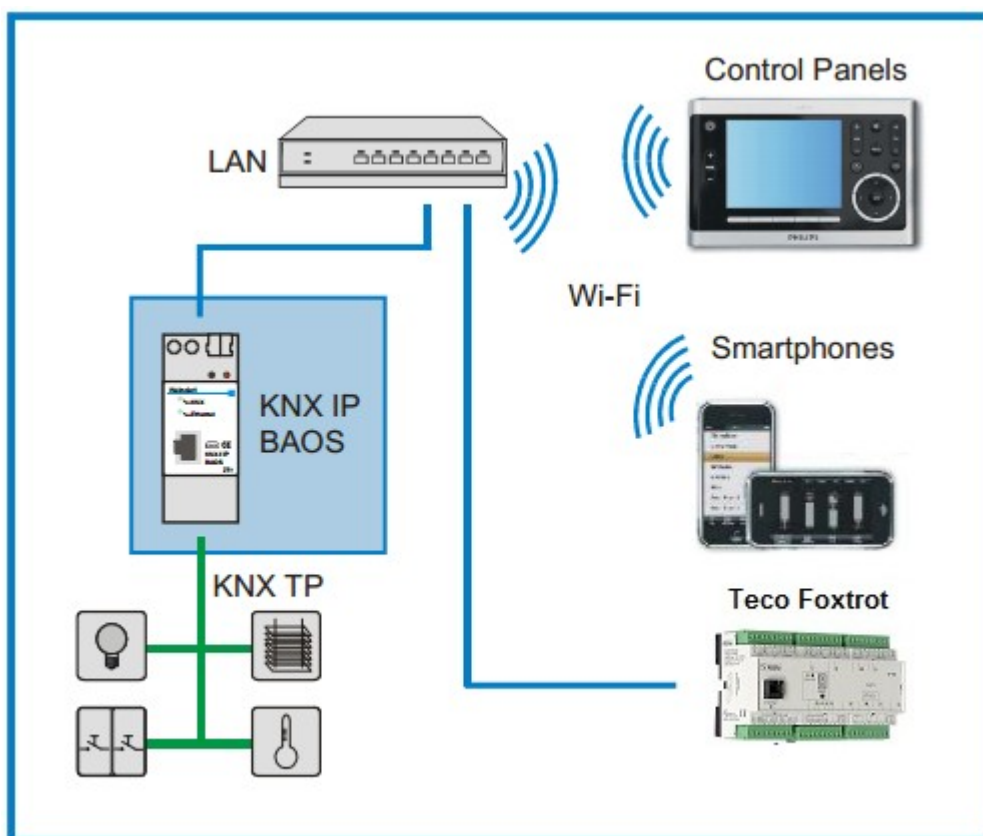
Knihovna KnxLib není podporovaná na systémech TC-650, u systému TC700 nelze knihovnu použít s procesorovými moduly CP-7002, CP-7003 a CP-7005.

Funkce z knihovny KnxLib jsou podporovány v centrálních jednotkách řady K a L (TC700 CP-7000, CP-7004 a CP-7007, všechny varianty systému Foxtrot) od verze v7.5. Doprogramování aplikačního profilu do PLC uživatelem je podporováno firmwarem centrální jednotky od verze v7.3.

Objednací číslo dokumentace ke knihovně KnxLib je TXV 003 80.01.

1.1 Propojení systémů

Systémy Tecomat jsou s instalací KNX propojeny prostřednictvím modulu KNX IP BAOS 772. Tento modul umožňuje jak výměnu dat mezi PLC Tecomat a sítí prvlů KNX, tak konfiguraci KNX sítě softwarem ETS4. Modul KNX IP BAOS 772 je připojen na sběrnici KNX a z pohledu KNX instalace představuje plnohodnotné KNX zařízení. S osobním počítačem (s programem ETS4) a s PLC je modul propojen přes síť LAN.



1.2 Princip činnosti

Modul KNX IP BAOS 772 je v síti KNX plně konfigurovatelný softwarem ETS4, kde je možno založit až 1000 objektů (Group Objects) a navázat je na skupinové adresy v síti KNX. Informace o těchto objektech poskytuje vestavěný server ve formátu JSON (JavaScript Object Notation). PLC tyto informace zpracuje a získané hodnoty uloží do proměnných v paměti PLC.

Každý objekt definovaný v modulu KNX IP BAOS 772 má definovaný datový typ. Ten určuje, jaké hodnoty budou objektem přenášeny. K dispozici jsou následující možnosti:

<i>Typ KNX objektu</i>	<i>Datový typ v PLC</i>	<i>Význam</i>
DPT 01	T_KNX_OBJECT_DPT1	Binary - 1 Bit
DPT 02	T_KNX_OBJECT_DPT2	Binary controlled - 2 Bits
DPT 03	T_KNX_OBJECT_DPT3	Dim up/down - 4 Bit
DPT 04	T_KNX_OBJECT_DPT4	Character -1 Byte
DPT 05	T_KNX_OBJECT_DPT5	Scaling – 1 Byte (8 Bit Unsigned Value)
DPT 06	T_KNX_OBJECT_DPT6	Signed Value – 1 Byte
DPT 07	T_KNX_OBJECT_DPT7	Unsigned Value - 2 Bytes
DPT 08	T_KNX_OBJECT_DPT8	Signed Value - 2 Byte s
DPT 09	T_KNX_OBJECT_DPT9	Float Value - 2 Bytes
DPT 10	T_KNX_OBJECT_DPT10	Time - 3 Bytes
DPT 11	T_KNX_OBJECT_DPT11	Date - 3 Bytes
DPT 12	T_KNX_OBJECT_DPT12	Unsigned Value - 4 Bytes
DPT 13	T_KNX_OBJECT_DPT13	Signed Value - 4 Bytes
DPT 14	T_KNX_OBJECT_DPT14	Float Value - 4 Bytes
DPT 15	T_KNX_OBJECT_DPT15	Access data - 4 Bytes
DPT 16	T_KNX_OBJECT_DPT16	Character string - 14 Bytes
DPT 17	T_KNX_OBJECT_DPT17	Scene - 1 Byte
DPT 18	T_KNX_OBJECT_DPT18	Scene Controlled - 1 Byte
DPT unknown	T_KNX_OBJECT_RAW	Max. 14 Byte, Data as array, byte-by-byte

V programu PLC pak musí být pro každý objekt založena proměnná odpovídajícího datového typu (viz popis typů T_KNX_OBJECT_DPT1, ..., T_KNX_OBJECT_DPT18 dále). Hodnoty těchto proměnných v PLC jsou komunikací udržovány v souladu se stavem objektů KNX. Pokud program PLC změní hodnotu některé proměnné, komunikace automaticky zapíše novou hodnotu do KNX objektu v modulu KNX IP BAOS 772 a modul zajistí odeslání této hodnoty do sítě KNX. PLC má tedy možnost jak číst tak zapisovat hodnoty z KNX instalace.

Výměna dat probíhá po síti LAN, přičemž jak PLC tak modul KNX IP BAOS 772 zveřejňují nové hodnoty objektů pouze při změně. To znamená, že komunikace není zahlcena cyklickým čtením resp. zápisem, naopak pokud se „nic neděje“ systémy pouze udržují spojení a přenosový kanál je volný a připravený přenést potřebné informace ve chvíli, kdy v PLC či v KNX síti dojde k nějakým změnám sledovaných objektů.

2 KONSTANTY

Knihovna KnxLib obsahuje následující konstanty:

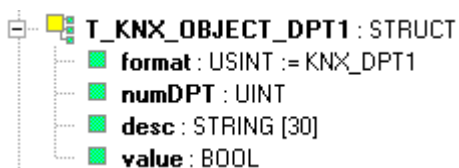
<i>Jméno</i>	<i>Typ</i>	<i>Hodnota</i>	<i>Význam</i>
Kód datového typu			
KNX_UNKNOWN	USINT	0	Neznámý datový typ
KNX_DPT1	USINT	1	Kód pro typ T_KNX_OBJECT_DPT1
KNX_DPT2	USINT	2	Kód pro typ T_KNX_OBJECT_DPT2
KNX_DPT3	USINT	3	Kód pro typ T_KNX_OBJECT_DPT3
KNX_DPT4	USINT	4	Kód pro typ T_KNX_OBJECT_DPT4
KNX_DPT5	USINT	5	Kód pro typ T_KNX_OBJECT_DPT5
KNX_DPT6	USINT	6	Kód pro typ T_KNX_OBJECT_DPT6
KNX_DPT7	USINT	7	Kód pro typ T_KNX_OBJECT_DPT7
KNX_DPT8	USINT	8	Kód pro typ T_KNX_OBJECT_DPT8
KNX_DPT9	USINT	9	Kód pro typ T_KNX_OBJECT_DPT9
KNX_DPT10	USINT	10	Kód pro typ T_KNX_OBJECT_DPT10
KNX_DPT11	USINT	11	Kód pro typ T_KNX_OBJECT_DPT11
KNX_DPT12	USINT	12	Kód pro typ T_KNX_OBJECT_DPT12
KNX_DPT13	USINT	13	Kód pro typ T_KNX_OBJECT_DPT13
KNX_DPT14	USINT	14	Kód pro typ T_KNX_OBJECT_DPT14
KNX_DPT15	USINT	15	Kód pro typ T_KNX_OBJECT_DPT15
KNX_DPT16	USINT	16	Kód pro typ T_KNX_OBJECT_DPT16
KNX_DPT17	USINT	17	Kód pro typ T_KNX_OBJECT_DPT17
KNX_DPT18	USINT	18	Kód pro typ T_KNX_OBJECT_DPT18
KNX_RAW	USINT	19	Kód pro typ T_KNX_OBJECT_RAW

3 DATOVÉ TYPY

V knihovně KnxLib jsou definovány následující datové typy:

Jméno	Typ	Význam
<i>T_KNX_OBJECT_DPT1</i>	STRUCT	Stav objektu DPT1
<i>T_KNX_OBJECT_DPT2</i>	STRUCT	Stav objektu DPT2
<i>T_KNX_OBJECT_DPT3</i>	STRUCT	Stav objektu DPT3
<i>T_KNX_OBJECT_DPT4</i>	STRUCT	Stav objektu DPT4
<i>T_KNX_OBJECT_DPT5</i>	STRUCT	Stav objektu DPT5
<i>T_KNX_OBJECT_DPT6</i>	STRUCT	Stav objektu DPT6
<i>T_KNX_OBJECT_DPT7</i>	STRUCT	Stav objektu DPT7
<i>T_KNX_OBJECT_DPT8</i>	STRUCT	Stav objektu DPT8
<i>T_KNX_OBJECT_DPT9</i>	STRUCT	Stav objektu DPT9
<i>T_KNX_OBJECT_DPT10</i>	STRUCT	Stav objektu DPT10
<i>T_KNX_OBJECT_DPT11</i>	STRUCT	Stav objektu DPT11
<i>T_KNX_OBJECT_DPT12</i>	STRUCT	Stav objektu DPT12
<i>T_KNX_OBJECT_DPT13</i>	STRUCT	Stav objektu DPT13
<i>T_KNX_OBJECT_DPT14</i>	STRUCT	Stav objektu DPT14
<i>T_KNX_OBJECT_DPT15</i>	STRUCT	Stav objektu DPT15
<i>T_KNX_OBJECT_DPT16</i>	STRUCT	Stav objektu DPT16
<i>T_KNX_OBJECT_DPT17</i>	STRUCT	Stav objektu DPT17
<i>T_KNX_OBJECT_DPT18</i>	STRUCT	Stav objektu DPT18
<i>T_KNX_OBJECT_RAW</i>	STRUCT	Stav objektu DPT RAW

3.1 Typ T_KNX_OBJECT_DPT1

Knihovna : *KnxLib*

Datový typ *T_KNX_OBJECT_DPT1* je struktura, které obsahuje informace o stavu KNX objektu typu DPT1 (1 Bit Boolean). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých poček struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT1)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *value* aktuální stav datapointu (BOOL hodnota)

3.2 Typ T_KNX_OBJECT_DPT2

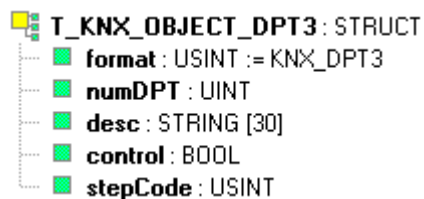
Knihovna : *KnxLib*

Datový typ *T_KNX_OBJECT_DPT2* je struktura, které obsahuje informace o stavu KNX objektu typu DPT2 (2 Bit Control). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých poček struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT2)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *control* aktuální stav datapointu (BOOL hodnota)
- *code* aktuální stav datapointu (BOOL hodnota)

3.3 Type T_KNX_OBJECT_DPT3

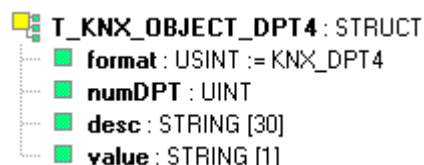
Knihovna : *KnxLib*

Datový typ *T_KNX_OBJECT_DPT3* je struktura, které obsahuje informace o stavu KNX objektu typu DPT3 (4 Bit, Control Dimming, Control Blinds). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých položek struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT3)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *control* aktuální stav datapointu (BOOL hodnota)
- *stepCode* aktuální stav datapointu (USINT hodnota 0,...,3)

3.4 Typ T_KNX_OBJECT_DPT4

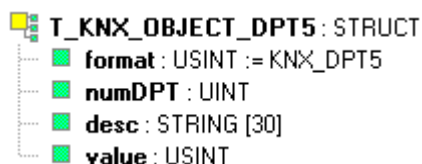
Knihovna : *KnxLib*

Datový typ *T_KNX_OBJECT_DPT4* je struktura, které obsahuje informace o stavu KNX objektu typu DPT4 (Character Set, 1 Byte). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých položek struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT4)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *value* aktuální stav datapointu (jeden ASCII znak)

3.5 Typ T_KNX_OBJECT_DPT5

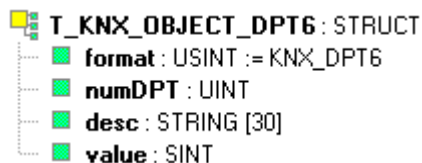
Knihovna : *KnxLib*

Datový typ *T_KNX_OBJECT_DPT5* je struktura, které obsahuje informace o stavu KNX objektu typu DPT5 (Unsigned Value, 1 Byte). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých položek struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT5)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *value* aktuální stav datapointu (USINT hodnota 0,...,255)

3.6 Typ T_KNX_OBJECT_DPT6

Knihovna : *KnxLib*

Datový typ *T_KNX_OBJECT_DPT6* je struktura, které obsahuje informace o stavu KNX objektu typu DPT6 (Signed Value, 1 Byte). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých položek struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT6)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *value* aktuální stav datapointu (SINT hodnota -128,...,127)

3.7 Typ T_KNX_OBJECT_DPT7

Knihovna : *KnxLib*

```

T_KNX_OBJECT_DPT7 : STRUCT
├── format : USINT := KNX_DPT7
├── numDPT : UINT
├── desc : STRING [30]
└── value : UINT

```

Datový typ *T_KNX_OBJECT_DPT7* je struktura, které obsahuje informace o stavu KNX objektu typu DPT7 (Unsigned Value, 2 Byte). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých poček struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT7)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *value* aktuální stav datapointu (UINT hodnota 0,...,65535)

3.8 Typ T_KNX_OBJECT_DPT8

Knihovna : *KnxLib*

```

T_KNX_OBJECT_DPT8 : STRUCT
├── format : USINT := KNX_DPT8
├── numDPT : UINT
├── desc : STRING [30]
└── value : INT

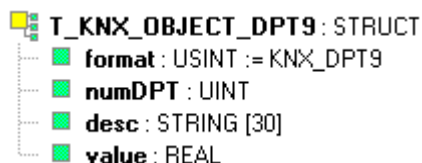
```

Datový typ *T_KNX_OBJECT_DPT8* je struktura, které obsahuje informace o stavu KNX objektu typu DPT8 (Signed Value, 2 Byte). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých poček struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT8)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *value* aktuální stav datapointu (INT hodnota -32768,...,32767)

3.9 Typ T_KNX_OBJECT_DPT9

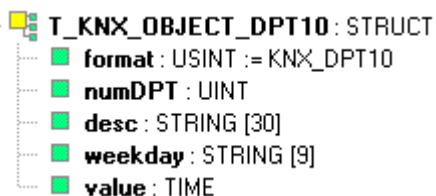
Knihovna : *KnxLib*

Datový typ *T_KNX_OBJECT_DPT9* je struktura, které obsahuje informace o stavu KNX objektu typu DPT9 (Float, 2 Byte). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých poček struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT9)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *value* aktuální stav datapointu (REAL hodnota)

3.10 Typ T_KNX_OBJECT_DPT10

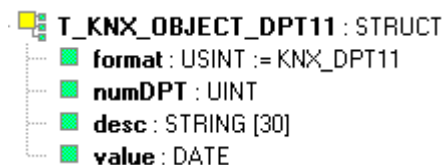
Knihovna : *KnxLib*

Datový typ *T_KNX_OBJECT_DPT10* je struktura, které obsahuje informace o stavu KNX objektu typu DPT10 (TIME, 3 Byte). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých poček struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT10)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *weekday* aktuální stav datapointu - den v týdnu (STRING)
(*'Monday'*, *'Tuesday'*, *'Wednesday'*, *'Thursday'*, *'Friday'*,
'Saturday', *'Sunday'*)
- *value* aktuální stav datapointu - čas (TIME hodnota)

3.11 Typ T_KNX_OBJECT_DPT11

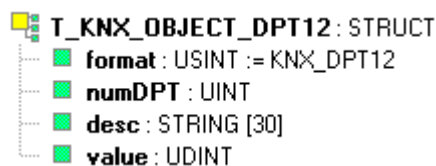
Knihovna : *KnxLib*

Datový typ *T_KNX_OBJECT_DPT11* je struktura, které obsahuje informace o stavu KNX objektu typu DPT11 (Date, 3 Byte). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých poček struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT11)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *value* aktuální stav datapointu (DATE hodnota)

3.12 Typ T_KNX_OBJECT_DPT12

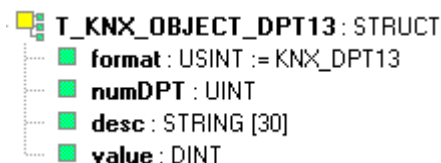
Knihovna : *KnxLib*

Datový typ *T_KNX_OBJECT_DPT12* je struktura, které obsahuje informace o stavu KNX objektu typu DPT12 (Unsigned Value, 4 Byte). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých poček struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT12)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *value* aktuální stav datapointu (UDINT hodnota)

3.13 Typ T_KNX_OBJECT_DPT13

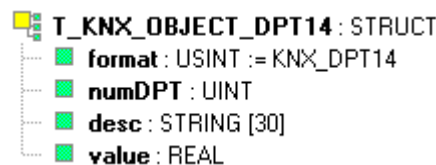
Knihovna : *KnxLib*

Datový typ *T_KNX_OBJECT_DPT13* je struktura, které obsahuje informace o stavu KNX objektu typu DPT13 (Signed Value, 4 Byte). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých pořek struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT13)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *value* aktuální stav datapointu (DINT hodnota)

3.14 Typ T_KNX_OBJECT_DPT14

Knihovna : *KnxLib*

Datový typ *T_KNX_OBJECT_DPT14* je struktura, které obsahuje informace o stavu KNX objektu typu DPT14 (Float, 4 Byte). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých pořek struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT14)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *value* aktuální stav datapointu (REAL hodnota)

3.15 Typ T_KNX_OBJECT_DPT15

Knihovna : *KnxLib*

```

T_KNX_OBJECT_DPT15 : STRUCT
├── format : USINT := KNX_DPT15
├── numDPT : UINT
├── desc : STRING [30]
├── code : UDINT
├── index : UDINT
├── flagError : BOOL
├── flagPermission : BOOL
├── flagReadDirection : BOOL
└── flagEncrypted : BOOL

```

Datový typ *T_KNX_OBJECT_DPT15* je struktura, které obsahuje informace o stavu KNX objektu typu DPT15 (Access, 14 Byte). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých poček struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT6)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *code* aktuální stav datapointu (UDINT hodnota)
- *index* aktuální stav datapointu (UDINT hodnota)
- *flagError* aktuální stav datapointu (BOOL hodnota)
- *flagPermission* aktuální stav datapointu (BOOL hodnota)
- *flagReadDirection* aktuální stav datapointu (BOOL hodnota)
- *flagEncrypted* aktuální stav datapointu (BOOL hodnota)

3.16 Typ T_KNX_OBJECT_DPT16

Knihovna : *KnxLib*

```

T_KNX_OBJECT_DPT16 : STRUCT
├── format : USINT := KNX_DPT16
├── numDPT : UINT
├── desc : STRING [30]
└── value : STRING [14]

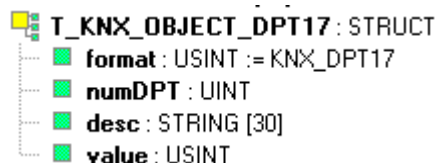
```

Datový typ *T_KNX_OBJECT_DPT16* je struktura, které obsahuje informace o stavu KNX objektu typu DPT16 (String, 14 Byte). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých poček struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT16)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *value* aktuální stav datapointu (STRING, 14 znaků)

3.17 Typ T_KNX_OBJECT_DPT17

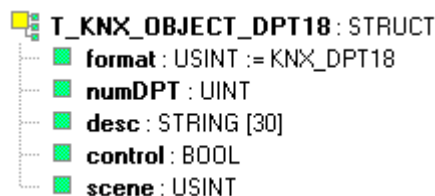
Knihovna : *KnxLib*

Datový typ *T_KNX_OBJECT_DPT17* je struktura, které obsahuje informace o stavu KNX objektu typu DPT17 (Scene Number, 1 Byte). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých poček struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT17)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *value* aktuální stav datapointu (USINT hodnota)

3.18 Typ T_KNX_OBJECT_DPT18

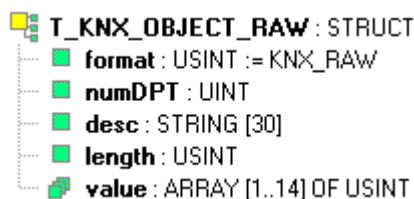
Knihovna : *KnxLib*

Datový typ *T_KNX_OBJECT_DPT18* je struktura, které obsahuje informace o stavu KNX objektu typu DPT18 (Scene Control, 1 Byte). KNX objekty se v modulu KNX IP BAOS 772 označují jako **datapoint**.

Význam jednotlivých poček struktury je následující:

- *format* typ datapointu (viz konstanta KNX_DPT18)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *control* aktuální stav datapointu (BOOL hodnota)
- *scene* aktuální stav datapointu (USINT hodnota)

3.19 Typ T_KNX_OBJECT_RAW

Knihovna : *KnxLib*

Datový typ *T_KNX_OBJECT_RAW* je struktura, které obsahuje informace o stavu KNX objektu typu RAW (Data as array, byte-by-byte).

Význam jednotlivých položek struktury je následující:

- *format* typ datapointu (viz konstanta KNX_RAW)
- *numDPT* číslo datapointu
- *desc* popis datapointu načtený z BAOS serveru
- *length* počet platných položek v poli *value[]*
- *value* aktuální stav datapointu (ARRAY OF USINT)

4 GLOBÁLNÍ PROMĚNNÉ

Knihovna KnxLib neobsahuje žádné globální proměnné.

5 FUNKCE

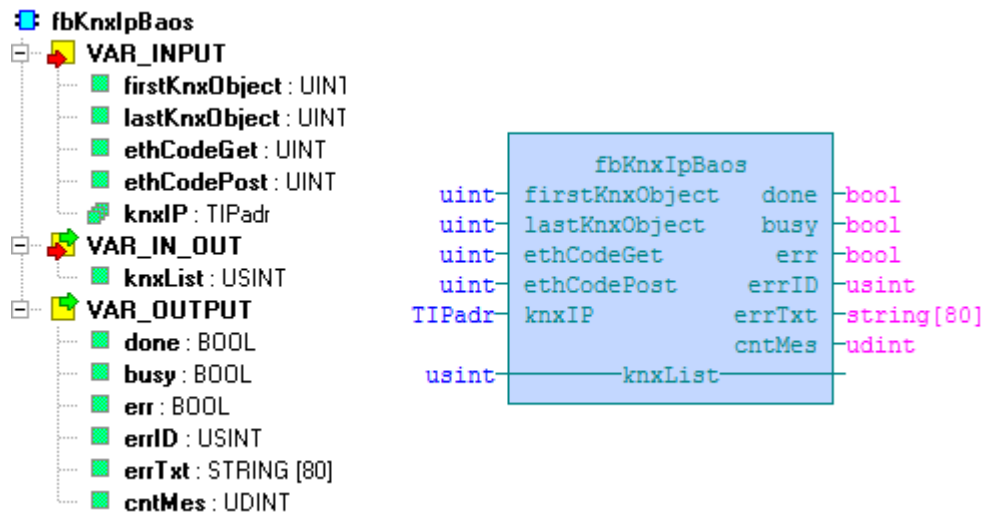
Knihovna KnxLib neobsahuje žádné funkce.

6 FUNKČNÍ BLOKY

V knihovně KnxLib jsou definovány následující funkční bloky:

<i>Funkční blok</i>	<i>Popis</i>
<i>fbKnxIpBaos</i>	Komunikace s instalací KNX

6.1 Funkční blok fbKnxIpBaos

Knihovna : *KnxLib*

Funkční blok *fbKnxIpBaos* komunikuje s instalací KNX přes integrační modul KNX IP BAOS 772. PLC je s modulem propojen přes síť LAN, komunikace probíhá HTTP protokolem na portu 80. Jedna instance bloku *fbKnxIpBaos* navazuje s modulem KNX IP BAOS 772 dvě spojení. První se používá pro načítání aktuálního stavu KNX objektů, druhé je využíváno pro změnu stavu KNX objektů (tj. pro zápis nových hodnot do KNX sítě). V HW konfiguraci PLC musí být tedy nastaveny dvě spojení ETH1_uni s následujícími parametry: délka přijímací zóny 512 bytů, délka vysílací zóny 512 bytů, typ protokolu TCP master, vzdálená IP adresa 0.0.0.0, vzdálený port 80, místní port 0. Do vstupní proměnné *ethCodeGet* je nutné nastavit kód kanálu, přes který budou načítána data z KNX sítě (např. *ETH1_uni0*). Do proměnné *ethCodePost* se pak nastaví kód kanálu, přes který budou data zapisována do KNX sítě (např. *ETH1_uni1*). IP adresa modulu KNX IP BAOS 772 se zadává v proměnné *knxIP*.

Vstupní proměnná *firstKnxObject* udává číslo prvního datapointu, na který se bude funkční blok *fbKnxIpBaos* dotazovat. Proměnná *lastKnxObject* udává číslo posledního komunikovaného datapointu.













Po startu programu se nejprve naváže spojení s modulem KNX IP BAOS 772 a načtou se popisy všech datapointů (KNX objektů) podle vstupních proměnných *firstKnxObject* a *lastKnxObject*. Například pokud je *firstKnxObject* =1 a *lastKnxObject* =20 tak budou načteny informace o objektech *datapoint1* až *datapoint20* včetně. Pro každý datapoint musí být v programu PLC založena proměnná, jejíž struktura odpovídá typu datapointu podle konfigurace v programu ETS4. Například pokud je *datapoint1* v modulu KNX IP BAOS 772 konfigurován programem ETS4 jako typ DPT4, pak musí být v programu PLC definovaná proměnná typu *T_KNX_OBJECT_DPT4*. Adresu této proměnné je pak třeba zapsat do seznamu KNX objektů. Odkaz na seznam KNX objektů se předává při volání instance funkčního bloku *fbKnxIpBaos* v proměnné *knxList*.

Výstupní proměnná *done* je nastavena na dobu jednoho cyklu na hodnotu TRUE pokaždé, když funkční blok *fbKnxIpBaos* dokončí zpracování informací získaných od modulu KNX IP BAOS 772. Proměnná *busy* signalizuje, že probíhá komunikace s modulem KNX IP BAOS 772 resp. zpracování dat získaných komunikací.

Pokud má výstupní proměnná *err* hodnotu TRUE, znamená to, že v činnosti bloku *fbKnxIpBaos* došlo k chybě. Pak proměnná *errID* obsahuje kódové číslo chyby a proměnná *errTxt* obsahuje textový popis chyby.

Výstupní proměnná *cntMes* udává počet úspěšně přenesených a zpracovaných zpráv.

Popis proměnných:

	Proměnná	Typ	Význam
VAR_INPUT			
	<i>firstKnxObject</i>	UINT	Číslo prvního načítaného KNX objektu (datapointu)
	<i>lastKnxObject</i>	UINT	Číslo posledního načítaného KNX objektu (datapointu)
	<i>ethCodeGet</i>	UINT	Kód komunikačního kanálu použitého pro čtení (viz ComLib) <i>ETH1_uni0</i> ethernet ETH1, spojení uni0 <i>ETH1_uni7</i> ethernet ETH1, spojení uni7
	<i>ethCodePost</i>	UINT	Kód komunikačního kanálu použitého pro zápis (viz ComLib) <i>ETH1_uni0</i> ethernet ETH1, spojení uni0 <i>ETH1_uni7</i> ethernet ETH1, spojení uni7
	<i>knxIP</i>	TIPadr	IP adresa modulu KNX IP BAOS 772
VAR_IN_OUT			
	<i>knxList</i>	ARRAY	Pole adres proměnných, do kterých se ukládají hodnoty KNX objektů (datapointů)
VAR_OUTPUT			
	<i>done</i>	BOOL	TRUE na dobu 1 cyklu při dokončení zpracování nových informací od modulu KNX IP BAOS 772
	<i>busy</i>	BOOL	TRUE pokud probíhá komunikace nebo zpracování dat
	<i>err</i>	BOOL	TRUE znamená, že při komunikaci s ústřednou došlo k chybě
	<i>err_ID</i>	USINT	Číslo chyby 0 ... bez chyb 200 ... chybí aplikační profil TXF 68905 249 ... chybné parametry <i>firstKnxObject</i> resp. <i>lastKnxObject</i> 250 ... jiná odpověď BAOS serveru než 200 OK 251 ... vypršel timeout komunikace 252 ... nepodařilo se získat všechna data od BAOS serveru 253 ... velikost http hlavičky překročila 512 bytů 254 ... chybně zadaná IP adresa modulu KNX IP BAOS 772 255 ...chybně nastavené parametry kanálu ETH1
	<i>err_txt</i>	STRING	Textový popis chyby
	<i>cntMes</i>	UDINT	Celkový počet přijatých zpráv od modulu KNX IP BAOS 772

Jednoduché použití funkčního bloku *fbKnxIpBaos* pro načítání stavu a ovládání jednoho KNX objektu (*datapoint1*) ukazuje následující příklad. Předpokládejme, že v parametrech modulu KNX IP BAOS 772 je programem ETS4 nastaven objekt datapoint 1 jako typ „DPT 01 – Binary – 1 Bit“ a v komentáři u tohoto datapointu je zadáno např. „my first datapoint“.

V PLC programu musíme nejprve založit proměnnou, jejíž datový typ odpovídá typu DPT 01. V ukázkovém programu je to globální proměnná *datapoint1* : *T_KNX_OBJECT_DPT1*. Název proměnné je možno volit libovolně, avšak pro udržení pořádku je dobré nazvat proměnnou stejně jako v programu ETS4. Tam se první KNX objekt v pořadí jmenuje *datapoint1*, takže proměnnou v PLC programu nazveme stejně. Datový typ proměnné musí být *T_KNX_OBJECT_DPT1*, protože typ proměnné v PLC musí odpovídat typu, který je nastaven v programu ETS4.

V dalším kroku je třeba založit seznam adres všech proměnných, které budeme mezi KNX a PLC synchronizovat. Tento seznam budeme udržovat v proměnné *knxObjectList* : *ARRAY[1..1] OF UDINT*. Pro tento příklad stačí pole obsahující pouze jeden prvek, typicky pak bude rozměr pole dán počtem přenášených KNX objektů. Indexy pole *knxObjectList* je nutné volit tak, aby odpovídaly číslům přenášených datapointů. Pokud prvním přenášeným objektem bude *datapoint1*, tak první index pole *knxObjectList* bude 1. Vzhledem k tomu, že bude přenášen pouze jeden objekt, tak i poslední index pole bude 1.

Po startu PLC programu se do seznamu *knxObjectList* musí uložit adresy proměnných, které odpovídají KNX objektům nastaveným v modulu KNX IP BAOS 772. Pro objekt *datapoint1* uložíme adresu PLC proměnné *datapoint1* do položky *knxObjectList[1]*. To zajistí následující příkaz: *knxObjectList[1] := PTR_TO_UDINT(ADR(datapoint1))*.

Pak už je možné volat instanci bloku *KnxIpBaos()*, která zajistí komunikaci s modulem KNX IP BAOS 772.

```

VAR_GLOBAL
  datapoint1      : T_KNX_OBJECT_DPT1;
  knxObjectList  : ARRAY[1..1] OF UDINT; // list of datapoints
END_VAR

PROGRAM prgMain
  VAR
    init          : BOOL;
    KnxIpBaos    : fbKnxIpBaos;
  END_VAR

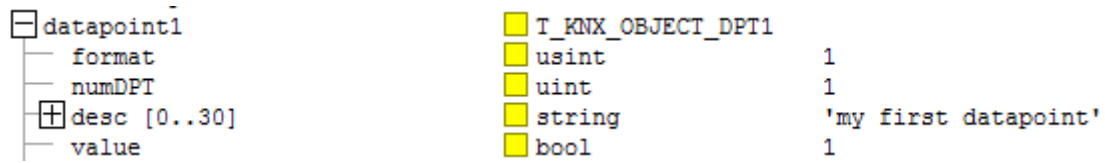
  IF NOT init THEN
    // add datapoint1 to list of KNX objects
    knxObjectList[1] := PTR_TO_UDINT(ADR(datapoint1));
    init := TRUE;
  END_IF;

  KnxIpBaos( firstKnxObject := 1, lastKnxObject := 1,
             ethCodeGet := ETH1_uni0, ethCodePost := ETH1_uni1,
             knxIP := STRING_TO_IPADR('192.168.134.180'),
             knxList := void(knxObjectList));
END_PROGRAM

```


Po zahájení komunikace uloží blok *fbKnxIpBaos* do proměnné *datapoint1.numDPT* číslo datapointu (pro kontrolu) a do proměnné *datapoint1.desc* popis datapointu načtený z modulu KNX IP BAOS 772. Poté se nastaví hodnota datapointu do proměnné *datapoint1.value*. Když se hodnota objektu *datapoint1* v KNX síti změní, promítne se tato změna automaticky do proměnné *datapoint1.value*. Naopak pokud z PLC programu změníme hodnotu proměnné *datapoint1.value*, pak bude nová hodnota zapsaná automaticky do objektu *datapoint1* v modulu KNX IP BAOS 772. Odtud se nová hodnota rozešle všem prvkům sítě KNX, které sdílí stejnou skupinovou adresu, jako objekt *datapoint1*.

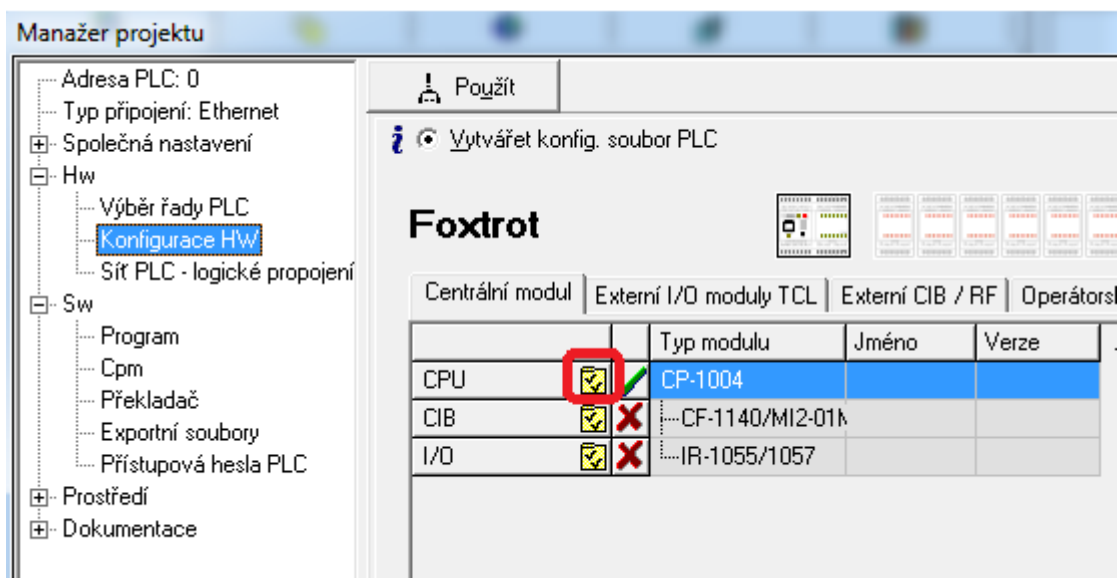
Stav proměnné *datapoint1* v PLC ukazuje následující obrázek.



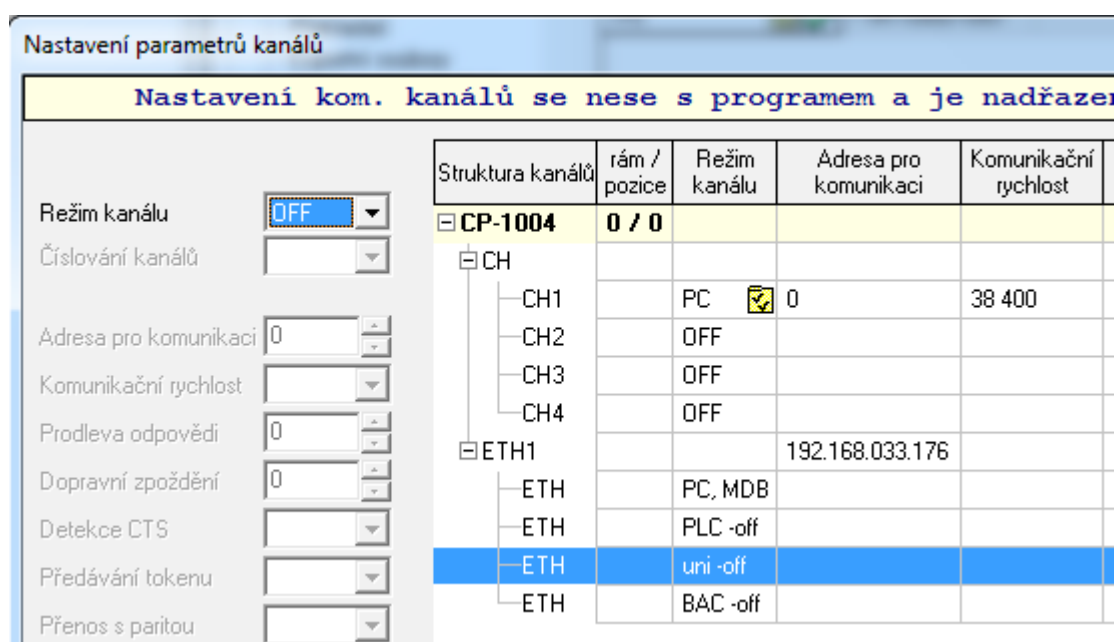
Položky *format* a *numDPT* využívá při komunikaci blok *fbKnxIpBaos*. Položka *desc* obsahuje popis datapointu zadaný v programu ETS4. Do těchto položek se nesmí programem PLC zapisovat. Položka *value* obsahuje hodnotu KNX objektu. Tuto hodnotu lze programem PLC číst i zapisovat.

7 NASTAVENÍ PARAMETRŮ KOMUNIKACE V PLC

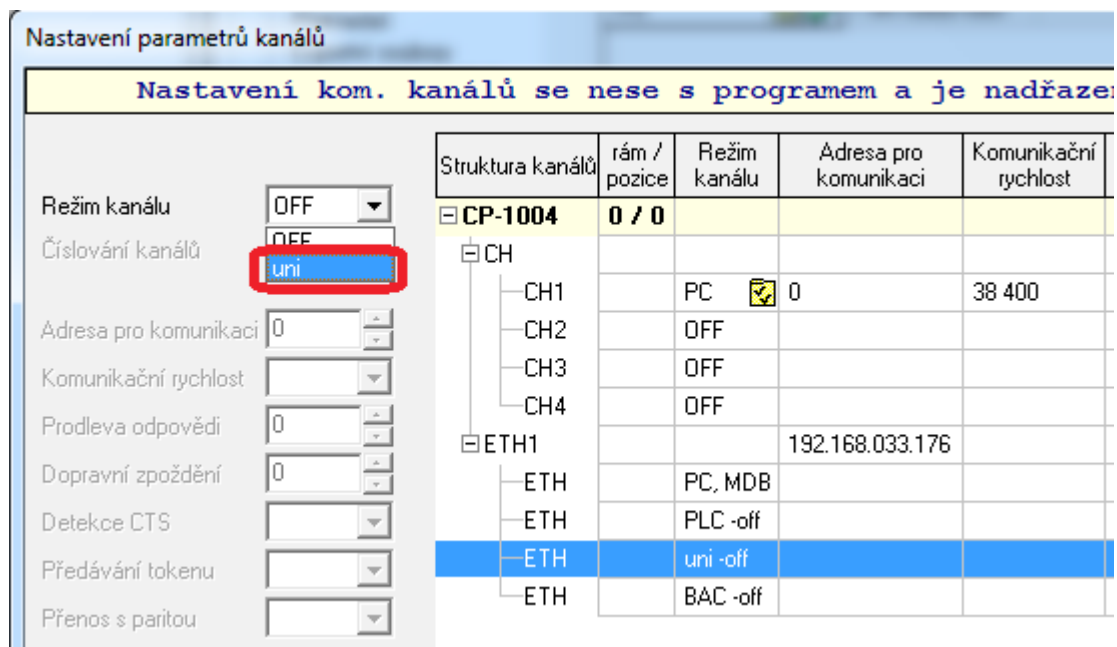
Komunikace PLC s modulem KNX IP BAOS 772 probíhá HTTP protokolem. Jedna instance bloku *fbKnxIpBaos* navazuje s modulem KNX IP BAOS 772 dvě spojení. Pro tato spojení je třeba nejprve zapnout podporu režimu UNI na rozhraní ethernet. Toto se v prostředí Mosaic provede pomocí Manažeru projektu. Po spuštění Manažera projektu (např. CTRL+ALT+F11) vybereme myší uzel HW konfigurace. Dále je třeba vyvolat dialog pro nastavení komunikačních kanálů centrální jednotky PLC, což se provede kliknutím na ikonu  v řádku CPU.



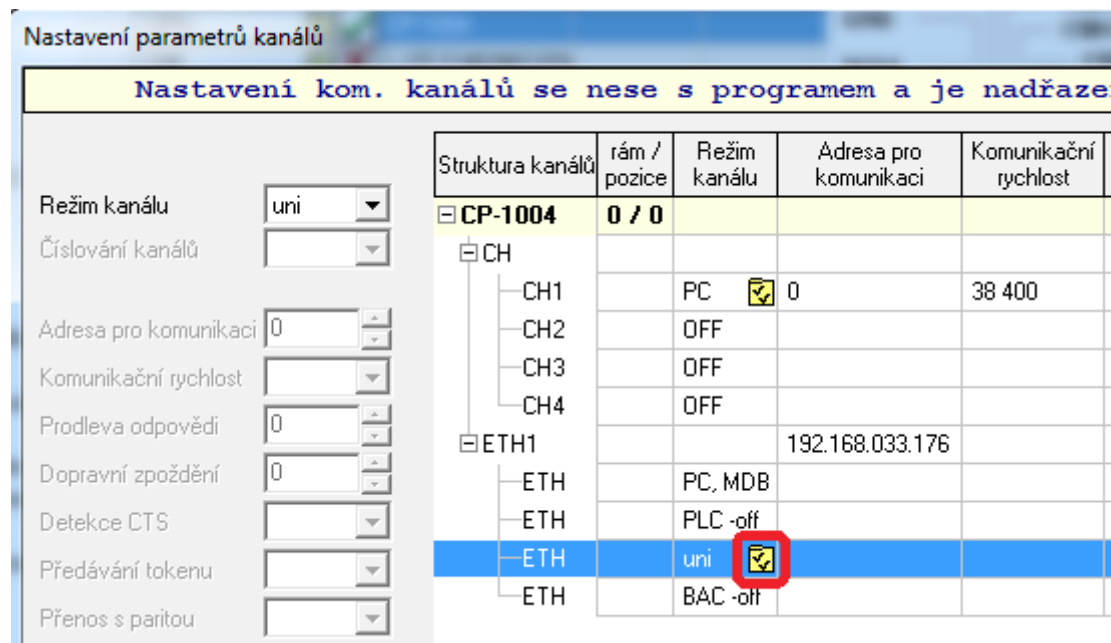
Poté klikneme na řádek s nastavením režimu uni pro rozhraní Ethernet (viz řádek ETH – uni-off) a ten se zbarví modře. V novém projektu je uni režim pro rozhraní ethernet vypnutý (viz pole Režim kanálu = OFF).



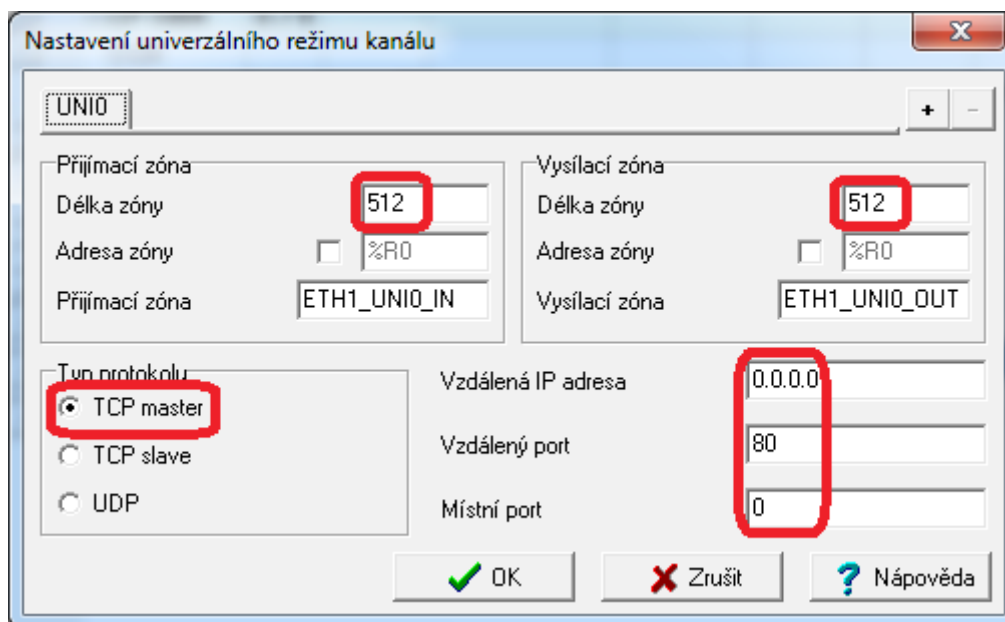
Poté je třeba zvolit režim kanálu UNI, což se provede pomocí rozbalovacího menu jak ukazuje následující obrázek.



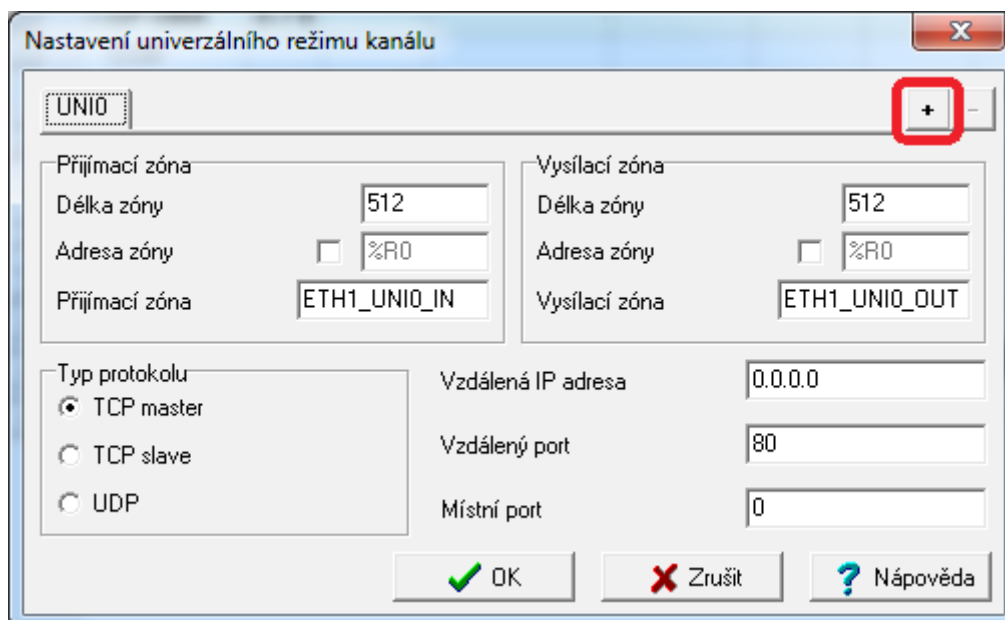
Následující obrázek ukazuje jak bude vypadat dialog po nastavení režimu Uni pro kanál ethernet. Kliknutím na ikonu v řádku ETH-uni a vyvoláme dialog pro nastavení parametrů komunikace v režimu UNI.



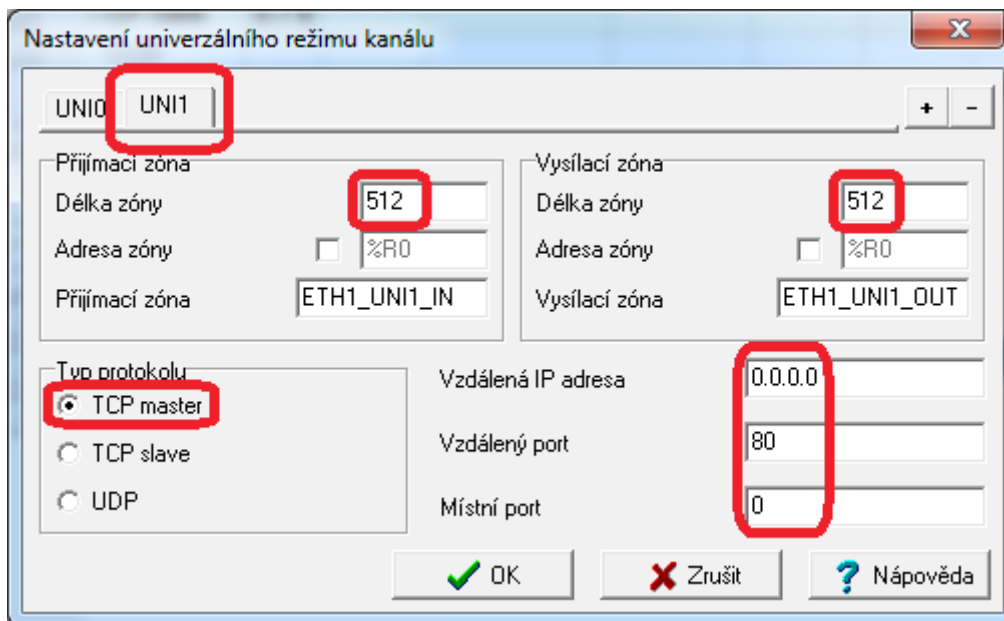
Objeví se dialog s názvem „Nastavení univerzálního režimu kanálu“. V něm nastavíme následující parametry pro první ethernet spojení (*ETH1_uni0*): zvolíme délku přijímací zóny 512 bytů, délku vysílací zóny 512 bytů, typ protokolu TCP master, vzdálená IP adresa 0.0.0.0, vzdálený port 80, místní port 0.



Dále je třeba přidat další spojení, což se provede kliknutím na klávesu +.



V okně nastavení univerzálního režimu kanálu přibude záložka pro další spojení (*ETH1_uni1*). Toto spojení nastavíme stejně jako v prvním případě: zvolíme délku přijímací zóny 512 bytů, délku vysílací zóny 512 bytů, typ protokolu TCP master, vzdálená IP adresa 0.0.0.0, vzdálený port 80, místní port 0.



Po stisku tlačítka OK je ethernet rozhraní PLC nastaveno pro komunikaci modulem KNX IP BAOS 772.

Poznámka.

IP adresa modulu KNX IP BAOS 772 se zadává jako parametr při volání instance bloku *fbKnxIpBaos*. Na závěr je třeba přeložit program a nahrát ho do PLC.

8 PŘÍKLAD POUŽITÍ

