

Knihovna EpsnetLib

TXV 003 73.01
třetí vydání
červenec 2021
změny vyhrazeny

Historie změn

Datum	Vydání	Popis změn
Září 2012	1	První vydání, popis odpovídá EpsnetLib_v11
Prosinec 2018	2	Popis odpovídá EpsnetLib_v14
Červenec 2021	3	Popis odpovídá EpsnetLib_v17

OBSAH

1 Úvod	3
2 Datové typy	4
3 Konstanty	5
4 Globální proměnné	5
5 Funkce	5
5.1 Funkce EpsnetCheckFrame	6
5.2 Funkce EpsnetCompleteFrame	7
5.3 Funkce EpsnetEthCmd	8
5.4 Funkce EpsnetEthCmdRW	11
5.5 Funkce EpsnetEthCmdWriteBool	14
5.6 Funkce EpsnetEthLongCmd	17
5.7 Funkce EpsnetEthLongCmdRW	19
5.8 Funkce EpsnetMasCmd	21
5.9 Funkce EpsnetMasCmdRW	23
5.10 Funkce EpsnetMasCmdWriteBool	24
6 Funkční bloky	26
6.1 Funkční blok fbEpsnetEth	27
6.2 Funkční blok fbEpsnetMultiEth	31
6.3 Funkční blok fbEpsnetLongEth	36
6.4 Funkční blok fbEpsnetMas	40
6.5 Funkční blok fbEpsnetSlave	43

1 ÚVOD

Knihovna EpsnetLib je standardně dodávána jako součást programovacího prostředí Mosaic. Knihovna obsahuje funkce a funkční bloky umožňující komunikaci protokolem Epsnet v roli master na rozhraní Ethernet.

Pokud chceme funkce z knihovny EpsnetLib použít v aplikačním programu PLC, je třeba nejprve přidat tuto knihovnu do projektu. Knihovna je dodávána jako součást instalace prostředí Mosaic od verze 2012.3.

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

2 DATOVÉ TYPY

V knihovně EpsnetLib jsou definovány následující datové typy:

Identifikátor	Typ	Význam
<i>TEpsnetCommand</i>	STRUCT	Příkaz pro funkční blok fbEpsnetEth
<i>TEpsnetCommandMas</i>	STRUCT	Příkaz pro funkční blok fbEpsnetMas
<i>TEpsnetCommands</i>	ENUM	Komunikační služby
<i>TEpsnetEthCmdNos</i>	ARRAY	Čísla příkazů ke spuštění fbEpsnetMultiEth
<i>TEpsnetLongCommand</i>	STRUCT	Příkaz pro funkční blok fbEpsnetLongEth
<i>TEpsnetZone</i>	ENUM	Oblasti zápisníku
<i>TEpsnetLongFrame</i>	STRUCT	Zpráva s datovým polem
<i>TEpsnetShortFrame</i>	STRUCT	Zpráva bez datového pole
<i>TEthEpsnetLongFrame</i>	STRUCT	Zpráva s datovým polem (Ethernet)
<i>TEthEpsnetShortFrame</i>	STRUCT	Zpráva bez datového pole (Ethernet)
<i>TEthEpsnetMaxFrame</i>	STRUCT	Datové pole maximální velikosti Epsnet UDP

3 KONSTANTY

V knihovně EpsnetLib nejsou definovány žádné konstanty.

4 GLOBÁLNÍ PROMĚNNÉ

V knihovně EpsnetLib nejsou definovány žádné globální proměnné.

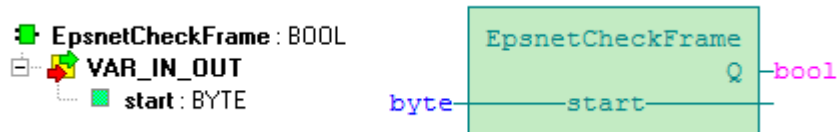
5 FUNKCE

Knihovna EpsnetLib obsahuje následující funkce:

Funkce	Popis
<i>EpsnetCheckFrame</i>	Kontrola kontrolního součtu, včetně kontroly rámce
<i>EpsnetCompleteFrame</i>	Výpočet kontrolního součtu – zapíše do rámce včetně koncového znaku
<i>Příkazy pro fbEpsnetEth a fbEpsnetMultiEth:</i>	
<i>EpsnetEthCmd</i>	Nastavení příkazu pro čtení nebo zápis bloku bytů
<i>EpsnetEthCmdRW</i>	Nastavení příkazu pro čtení a zápis bloku bytů
<i>EpsnetEthCmdWriteBool</i>	Nastavení příkazu pro zápis jednoho bitu
<i>Příkazy pro fbEpsnetLongEth:</i>	
<i>EpsnetEthLongCmd</i>	Nastavení příkazu pro čtení nebo zápis dlouhých bloků bytů
<i>EpsnetEthLongCmdRW</i>	Nastavení příkazu pro čtení a zápis dlouhých bloků bytů
<i>Příkazy pro fbEpsnetMas:</i>	
<i>EpsnetMasCmd</i>	Nastavení příkazu pro čtení nebo zápis bloku bytů
<i>EpsnetMasCmdRW</i>	Nastavení příkazu pro čtení a zápis bloku bytů
<i>EpsnetMasCmdWriteBool</i>	Nastavení příkazu pro zápis jednoho bitu

5.1 Funkce EpsnetCheckFrame

Knihovna : *EpsnetLib*

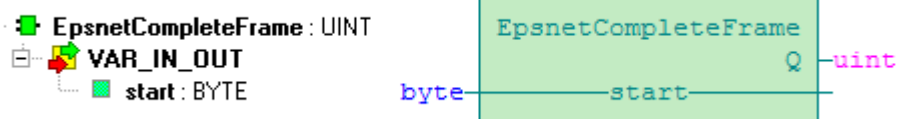


Funkce *EpsnetCheckFrame* provádí kontrolu kontrolního součtu, včetně kontroly rámce zprávy protokolu EPSNET.

Popis proměnných :

	Proměnná	Typ	Význam
VAR_IN_OUT			
	<i>start</i>	BYTE	První byte rámce (úvodní znak)
EpsnetCheckFrame			
	<i>Návratová hodnota</i>	BOOL	TRUE pokud je rámec platný

5.2 Funkce *EpsnetCompleteFrame*

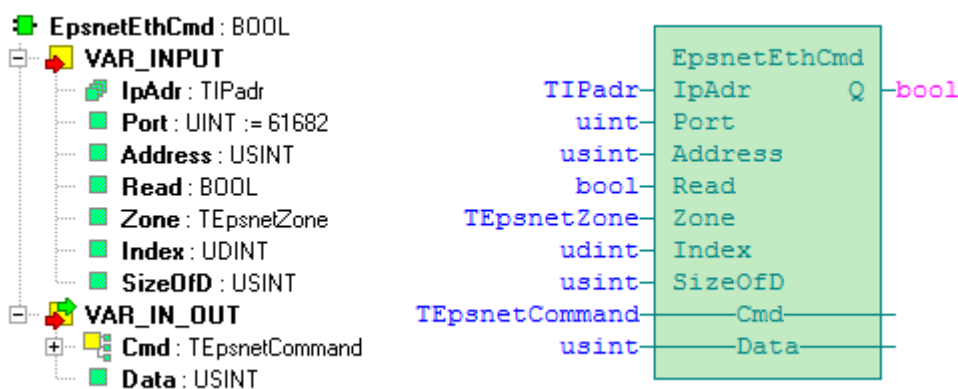
Knihovna : *EpsnetLib*

Funkce *EpsnetCompleteFrame* vypočte kontrolní součet rámce a zapíše jej na příslušné místo. První byte rámce se předává na vstupu *start*. Uživatel musí zajistit, aby za prvním byte bylo deklarováno dostatek prostoru pro zápis kontrolního součtu a koncového znaku, což lze zajistit použitím proměnné typu *TEpsnetLongFrame* nebo pole o velikosti 255 bytů.

Popis proměnných :

	Proměnná	Typ	Význam
VAR_IN_OUT			
	<i>start</i>	BYTE	První byte rámce (úvodní znak 1 nebo 2)
EpsnetCompleteFrame			
	<i>Návratová hodnota</i>	UINT	Délka zprávy včetně kontrolního součtu a koncového znaku. Nula pokud se nenajde správný úvodní znak nebo špatná délka

5.3 Funkce *EpsnetEthCmd*












Knihovna : *EpsnetLib*

Funkce *EpsnetEthCmd* slouží k nastavení jedné položky v řídicím poli příkazů pro blok *fbEpsnetEth* nebo *fbEpsnetMultiEth*. Funkce provede nastavení příkazu pro čtení nebo zápis bloku bytů. IP adresa cílového PLC se předává na vstupu *IpAdr*. Pokud je PLC za NAT je možné nastavit jiné než výchozí číslo UDP portu na vstupu *Port*. Vstup *Address* lze použít pokud má PLC systém podřízené stanice na sériových linkách v MPC režimu s povoleným tunelováním. V ostatních případech lze nechat výchozí hodnotu 0. Bit *Read* určuje zda budou data z PLC čtená (hodnota TRUE) nebo zapisována (hodnota FALSE).

Adresa bloku dat v paměti dotazovaného PLC je určena dvojicí vstupů *Zone* a *Index*. Vstup *SizeOfD* určuje délku komunikovaného bloku dat. Uživatel musí zajistit, aby proměnná, která slouží pro zápis či čtení dat a jejíž první byte se předává na vstupu *Data*, měla velikost minimálně rovnou hodnotě vstupu *SizeOfD*.

Odpovídající položka v poli příkazů se předává na vstupu *Cmd*.

Popis proměnných :

	Proměnná	Typ	Význam
VAR_INPUT			
	<i>IpAdr</i>	TIPadr	IP adresa
	<i>Port</i>	UINT	Číslo portu
	<i>LocalPort</i>	UINT	Číslo lokálního portu
	<i>Address</i>	USINT	Adresa podřízené stanice
	<i>Read</i>	BOOL	FALSE = zápis, TRUE = čtení
	<i>Zone</i>	TEpsnetZone	Oblast zápisníku
	<i>Index</i>	UDINT	Index prvního čteného byte v zápisníku
	<i>SizeOfD</i>	USINT	Velikost proměnné pro čtení (max. 246) nebo zápis (max. 241) v bytech
VAR_IN_OUT			
	<i>Cmd</i>	TEpsnet-Command	Výsledný příkaz v poli
	<i>Data</i>	USINT	První byte cílové/zdrojové proměnné
EpsnetEthCmd			
	<i>Návratová hodnota</i>	BOOL	TRUE pokud jsou vstupní parametry platné

Příklad programu volání funkce *EpsnetEthCmd*. Příklad realizuje čtení a zápis proměnné typu *DATE_AND_TIME* s pevnou pauzou mezi komunikacemi (odměřováno časovačem *CommTimer*):

```
VAR_GLOBAL CONSTANT
  MAX_CMD1 : UINT := 2;
END_VAR

VAR_GLOBAL
  Commands1 : ARRAY [0..MAX_CMD1-1] OF TEpsnetCommand;
  CommandWrite AT Commands1[0];
  CommandRead AT Commands1[1];
END_VAR

PROGRAM prgExampleEpsnetEthCmd
  VAR
    EpsnetEth : fbEpsnetEth;
    CommTimer : TON;
    IPAddr : TIPAdr := [192,168,33,175];
    Data : ARRAY [0..1] OF DATE_AND_TIME;
    CmdNo : UINT;
  END_VAR

  //write blok
  EpsnetEthCmd(IpAdr := IPAddr, Read := false,
    Zone := epszR, Index := 60000,
    SizeOfD := sizeof(Data[0]), Cmd := CommandWrite,
    Data := void(Data[0])) ;

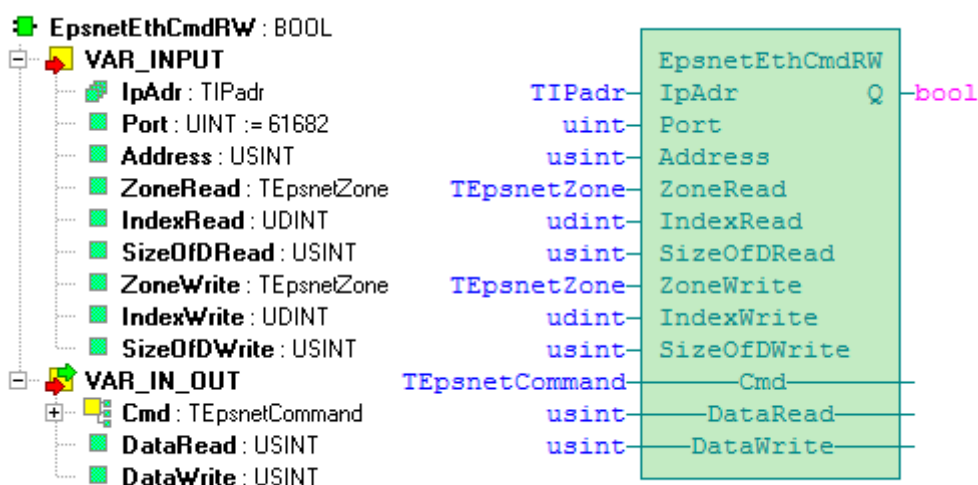
  //read blok
  EpsnetEthCmd(IPAddr, 61682, 17744, 0, true, epszR, 60100,
    sizeof(Data[1]), CommandRead, void(Data[1]));

  Data[0] := GetDateTime();

  //time between messages
  CommTimer(IN := NOT EpsnetEth.Busy, PT := T#1s);

  //communication
  EpsnetEth(Req := CommTimer.Q, MyAddress := 0,
    chanCode := ETH1_uni0,
    CmdNo := CmdNo, MaxCmd := MAX_CMD1,
    Timeout := T#1s,
    Commands := Commands1[0]);

  //next command
  IF EpsnetEth.Done OR EpsnetEth.Error THEN
    CmdNo := CmdNo + 1;
    IF CmdNo > 1 THEN
      CmdNo := 0;
    END_IF;
  END_IF;
END_PROGRAM
```















5.4 Funkce *EpsnetEthCmdRW*Knihovna : *EpsnetLib*

Funkce *EpsnetEthCmdRW* slouží k nastavení jedné položky v řídicím poli příkazů pro blok *fbEpsnetEth* nebo *fbEpsnetMultiEth*. Funkce provede nastavení příkazu pro současné čtení a zápis bloku bytů. IP adresa cílového PLC se předává na vstupu *IpAdr*. Pokud je PLC za NAT je možné nastavit jiné než výchozí číslo UDP portu na vstupu *Port*. Vstup *Address* lze použít pokud má PLC systém podřízené stanice na sériových linkách v MPC režimu s povoleným tunelováním. V ostatních případech lze nechat výchozí hodnotu nula.

Adresy bloků dat v paměti dotazovaného PLC jsou určena dvojicí vstupů *ZoneRead* a *IndexRead* pro čtený blok a *ZoneWrite* a *IndexWrite*. Vstupy *SizeOfDRead* a *SizeOfDWrite* určují délku komunikovaných bloků dat. Uživatel musí zajistit, aby proměnná, která slouží pro zápis dat a jejíž první byte se předává na vstupu *DataWrite*, měla velikost minimálně rovnou hodnotě vstupu *SizeOfDWrite*. Stejně tak platí, že proměnná, která slouží pro čtení dat a jejíž první byte se předává na vstupu *DataRead* musí mít velikost minimálně rovnou hodnotě vstupu *SizeOfDRead*.

Odpovídající položka v poli příkazů se předává na vstupu *Cmd*.

Popis proměnných :

	Proměnná	Typ	Význam
VAR_INPUT			
	<i>IpAdr</i>	TIPadr	IP adresa
	<i>Port</i>	UINT	Číslo portu
	<i>LocalPort</i>	UINT	Číslo lokálního portu
	<i>Address</i>	USINT	Adresa podřízené stanice
	<i>ZoneRead</i>	TEpsnetZone	Oblast zápisníku
	<i>IndexRead</i>	UDINT	Index prvního čteného byte v zápisníku
	<i>SizeOfDRead</i>	USINT	Velikost proměnné pro čtení (max. 246)
	<i>ZoneWrite</i>	TEpsnetZone	Oblast zápisníku
	<i>IndexWrite</i>	UDINT	Index prvního čteného byte v zápisníku
	<i>SizeOfDWrite</i>	USINT	Velikost proměnné pro zápis (max. 237) v bytech
VAR_IN_OUT			
	<i>Cmd</i>	TEpsnet-Command	Výsledný příkaz v poli
	<i>DataRead</i>	USINT	První byte cílové proměnné
	<i>DataWrite</i>	USINT	První byte zdrojové proměnné
EpsnetEthCmdRW			
	<i>Návratová hodnota</i>	BOOL	TRUE pokud jsou vstupní parametry platné

Příklad programu volání funkce *EpsnetEthCmdRW*. Program stejně jako příklad pro *EpsnetEthCmd* realizuje čtení a zápis proměnné typu `DATE_AND_TIME` s pevnou pauzou mezi komunikacemi (odměřováno časovačem *CommTimer*). Na rozdíl od předchozímu příkladu v tomto případě proběhne čtení i zápis najednou:

```
VAR_GLOBAL
  CommandRW : TEpsnetCommand;
END_VAR

PROGRAM prgExampleEpsnetEthCmdRW
  VAR
    EpsnetEth : fbEpsnetEth;
    CommTimer : TON;
    IPAddr : TIPAdr := [192,168,33,175];
    Data : ARRAY [0..1] OF DATE_AND_TIME;
  END_VAR

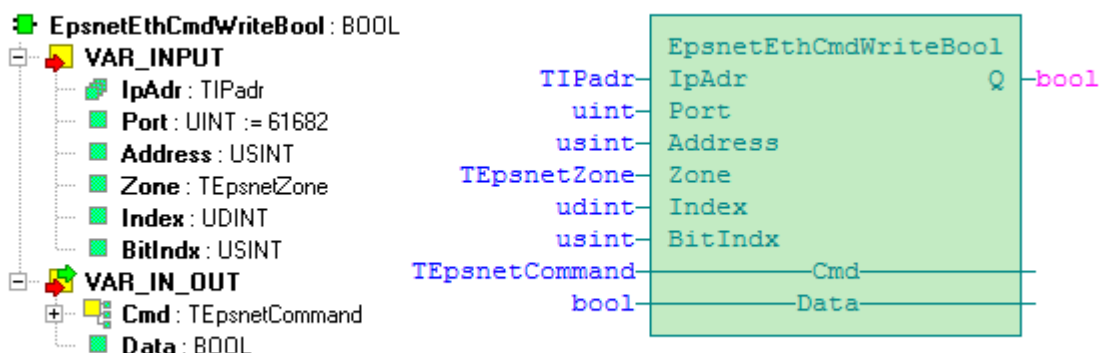
  EpsnetEthCmdRW(IpAdr := IPAddr,
    ZoneRead := epszR,
    IndexRead := 50100,
    SizeOfDRead := sizeof(Data[1]),
    ZoneWrite := epszR,
    IndexWrite := 50000,
    SizeOfDWrite := sizeof(Data[0]),
    Cmd := CommandRW,
    DataRead := void(Data[1]),
    DataWrite := void(Data[0]));

  Data[0] := GetDateTime();

  //time between messages
  CommTimer(IN := NOT EpsnetEth.Busy, PT := T#1s);

  //communication
  EpsnetEth(Req := CommTimer.Q, MyAddress := 0,
    chanCode := ETH1_uni1,
    CmdNo := 0, MaxCmd := 1,
    Timeout := T#1s,
    Commands := CommandRW);
END_PROGRAM
```

5.5 Funkce *EpsnetEthCmdWriteBool*

Knihovna : *EpsnetLib*











Funkce *EpsnetEthCmdWriteBool* slouží k nastavení jedné položky v řídicím poli příkazů pro blok *fbEpsnetEth* nebo *fbEpsnetMultiEth*. Funkce provede nastavení příkazu pro zápis jednoho bitu. Tento zápis neovlivní hodnoty ostatních bitů v bytu, do kterého patří.

IP adresa cílového PLC se předává na vstupu *IpAdr*. Pokud je PLC za NAT je možné nastavit jiné než výchozí číslo UDP portu na vstupu *Port*. Vstup *Address* lze použít pokud má PLC systém podřízené stanice na sériových linkách v MPC režimu s povoleným tunelováním. V ostatních případech lze nechat výchozí hodnotu 0.

Adresa zapisovaného bitu v paměti dotazovaného PLC je určena trojicí vstupů *Zone*, *Index* a *BitIndx*. Vstup *BitIndx* je indexem bitu v rámci byte a může nabývat hodnot 0 až 7. Uživatel musí zajistit, aby proměnná, která se předává na vstupu *Data*, byla typu `BOOL`.

Odpovídající položka v poli příkazů se předává na vstupu *Cmd*.

Popis proměnných :

	Proměnná	Typ	Význam
VAR_INPUT			
	<i>IpAdr</i>	TIPadr	IP adresa
	<i>Port</i>	UINT	Číslo portu
	<i>LocalPort</i>	UINT	Číslo lokálního portu
	<i>Address</i>	USINT	Adresa podřízené stanice
	<i>Zone</i>	TEpsnetZone	Oblast zápisníku
	<i>Index</i>	UDINT	Index prvního čteného byte v zápisníku
	<i>BitIndx</i>	USINT	Index bitu v byte (0 až 7)
VAR_IN_OUT			
	<i>Cmd</i>	TEpsnet-Command	Výsledný příkaz v poli
	<i>Data</i>	USINT	Zdrojová proměnná (musí být typu BOOL!)
EpsnetEthCmd			
	<i>Návratová hodnota</i>	BOOL	TRUE pokud jsou vstupní parametry platné

Příklad programu s voláním funkce *EpsnetEthCmdWriteBool*. V příkladu se zapisují jednotlivé bity a zpětně se čte hodnota bytu, do kterého zápis probíhá. Zápis je řízen trojicí proměnných *SetBit* – odstartuje zápis a čtení, *BitIndx* – určuje index bitu, který je modifikován, *BitValue* – určuje jaká hodnota do bitu bude zapsána:

```

VAR_GLOBAL CONSTANT
  MAX_CMD2 : UINT := 2;
END_VAR

VAR_GLOBAL
  Commands2 : ARRAY [0..MAX_CMD2-1] OF TEpsnetCommand;
  CommandReadByte AT Commands2[0];
  CommandWriteBit AT Commands2[1];
END_VAR

PROGRAM prgExampleEpsnetEthCmdWriteBool
  VAR
    EpsnetEth : fbEpsnetEth;
    IPAddr    : TIPadr := [192,168,33,175];
    Data      : BYTE;
    SetBit    : BOOL;
    BitIndx   : USINT;
    BitValue  : BOOL;
    CmdNo     : UINT := 1;
    ErrCnt    : UDINT;
  END_VAR

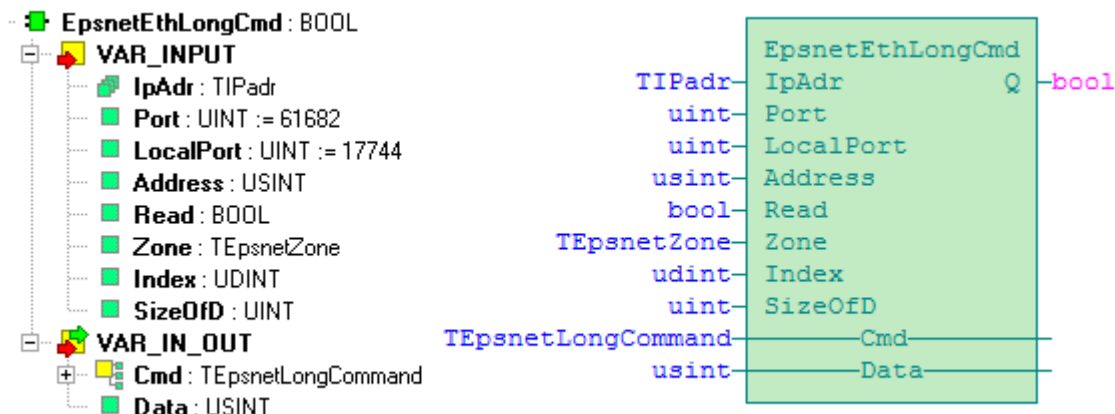
  //read byte
  EpsnetEthCmd(IPAddr, 61682, 17744, 0, true, epszR, 40000,
               sizeof(Data), CommandReadByte, void(Data));

  //write bit
  EpsnetEthCmdWriteBool(IpAdr := IPAddr,
                        Zone := epszR, Index := 40000,
                        BitIndx := BitIndx,
                        Cmd := CommandWriteBit,
                        Data := BitValue);

  //communication
  EpsnetEth(Req := SetBit AND NOT EpsnetEth.Busy,
            MyAddress := 0,
            chanCode := ETH1_uni2,
            CmdNo := CmdNo, MaxCmd := MAX_CMD2,
            Timeout := T#1s,
            Commands := Commands2[0]);

  //next command
  IF EpsnetEth.Done OR EpsnetEth.Error THEN
    IF CmdNo = 1 THEN
      CmdNo := 0;
    ELSE
      CmdNo := 1;
      SetBit := false;
    END_IF;
    IF EpsnetEth.Error THEN
      ErrCnt := ErrCnt + 1;
    END_IF;
  END_IF;
END_PROGRAM

```


5.6 Funkce *EpsnetEthLongCmd*Knihovna : *EpsnetLib*












Funkce *EpsnetEthLongCmd* slouží k nastavení jedné položky v řídicím poli příkazů pro blok *fbEpsnetLongEth*. Funkce provede nastavení příkazu pro čtení nebo zápis bloku bytů. IP adresa cílového PLC se předává na vstupu *IpAdr*. Pokud je PLC za NAT je možné nastavit jiné než výchozí číslo UDP portu na vstupu *Port*. Vstup *Address* lze použít pokud má PLC systém podřízené stanice na sériových linkách v MPC režimu s povoleným tunelováním. V ostatních případech lze nechat výchozí hodnotu 0. Bit *Read* určuje zda budou data z PLC čtená (hodnota TRUE) nebo zapisována (hodnota FALSE).

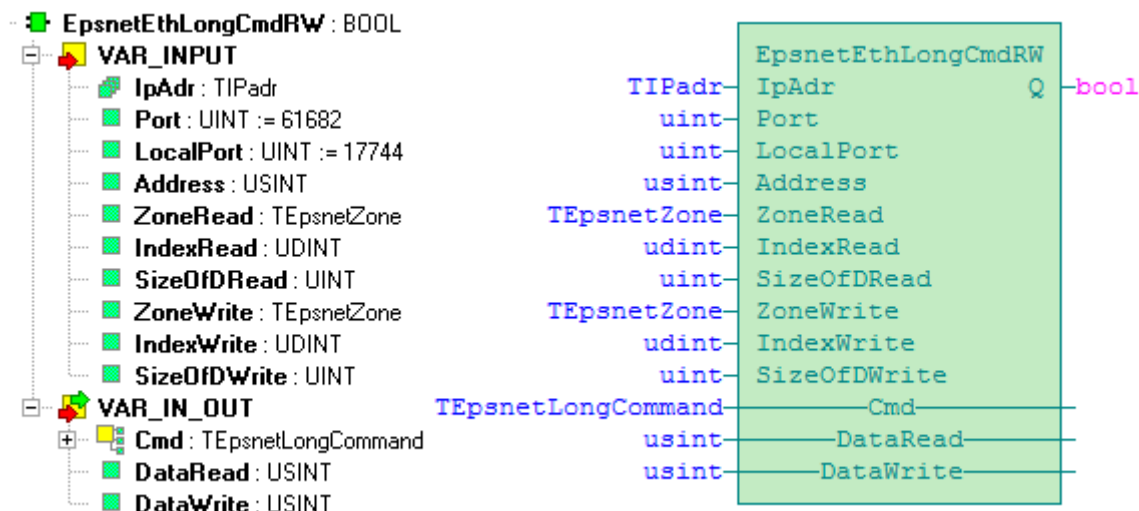
Adresa bloku dat v paměti dotazovaného PLC je určena dvojicí vstupů *Zone* a *Index*. Vstup *SizeOfD* určuje délku komunikovaného bloku dat. Uživatel musí zajistit, aby proměnná, která slouží pro zápis či čtení dat a jejíž první byte se předává na vstupu *Data*, měla velikost minimálně rovnou hodnotě vstupu *SizeOfD*.

Odpovídající položka v poli příkazů se předává na vstupu *Cmd*.

Příklad viz *fbEpsnetLongEth*

Popis proměnných :

	Proměnná	Typ	Význam
VAR_INPUT			
	<i>IpAdr</i>	TIPadr	IP adresa
	<i>Port</i>	UINT	číslo portu
	<i>LocalPort</i>	UINT	číslo lokálního portu
	<i>Address</i>	USINT	adresa podřízené stanice
	<i>Read</i>	BOOL	FALSE = zápis, TRUE = čtení
	<i>Zone</i>	TEpsnetZone	oblast zápisníku
	<i>Index</i>	UDINT	index prvního čteného byte v zápisníku
	<i>SizeOfD</i>	UINT	velikost proměnné pro čtení (max. 1230) nebo zápis (max. 1205)
VAR_IN_OUT			
	<i>Cmd</i>	TEpsnetLong-Command	výsledný příkaz v poli
	<i>Data</i>	USINT	první byte cílové/zdrojové proměnné
EpsnetEthLongCmd			
	<i>Návratová hodnota</i>	BOOL	nastavení příkazu pro čtení nebo zápis bloku bytů

5.7 Funkce *EpsnetEthLongCmdRW*Knihovna : *EpsnetLib*















Funkce *EpsnetEthLongCmdRW* slouží k nastavení jedné položky v řídicím poli příkazů pro blok *fbEpsnetLongEth*. Funkce provede nastavení příkazu pro současné čtení a zápis bloku bytů. IP adresa cílového PLC se předává na vstupu *IpAdr*. Pokud je PLC za NAT je možné nastavit jiné než výchozí číslo UDP portu na vstupu *Port*. Vstup *Address* lze použít pokud má PLC systém podřízené stanice na sériových linkách v MPC režimu s povoleným tunelováním. V ostatních případech lze nechat výchozí hodnotu 0.

Adresy bloků dat v paměti dotazovaného PLC jsou určena dvojicí vstupů *ZoneRead* a *IndexRead* pro čtený blok a *ZoneWrite* a *IndexWrite*. Vstupy *SizeOfDRead* a *SizeOfDWrite* určují délku komunikovaných bloků dat. Uživatel musí zajistit, aby proměnná, která slouží pro zápis dat a jejíž první byte se předává na vstupu *DataWrite*, měla velikost minimálně rovnou hodnotě vstupu *SizeOfDWrite*. Stejně tak platí, že proměnná, která slouží pro čtení dat a jejíž první byte se předává na vstupu *DataRead* musí mít velikost minimálně rovnou hodnotě vstupu *SizeOfDRead*.

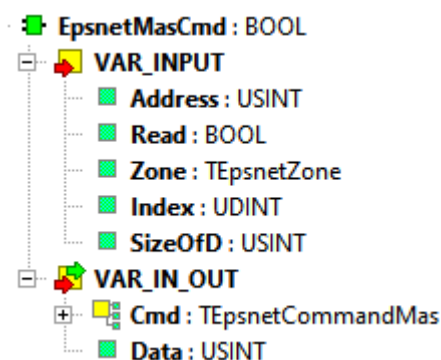
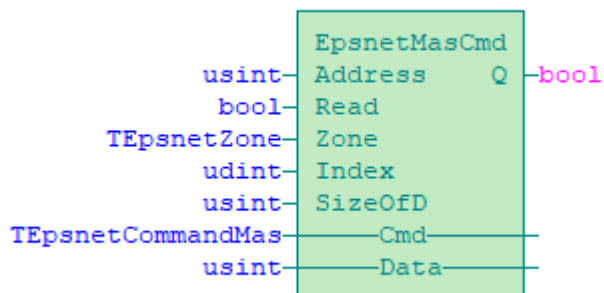
Odpovídající položka v poli příkazů se předává na vstupu *Cmd*.

Příklad viz *fbEpsnetLongEth*

Popis proměnných :

	Proměnná	Typ	Význam
VAR_INPUT			
	<i>IpAdr</i>	TIPadr	IP adresa
	<i>Port</i>	UINT	číslo portu
	<i>LocalPort</i>	UINT	číslo lokálního portu
	<i>Address</i>	USINT	adresa podřízené stanice
	<i>ZoneRead</i>	TEpsnetZone	oblast zápisníku
	<i>IndexRead</i>	UDINT	index prvního čteného byte v zápisníku
	<i>SizeOfDRead</i>	UINT	velikost proměnné pro čtení (max. 1230)
	<i>ZoneWrite</i>	TEpsnetZone	oblast zápisníku
	<i>IndexWrite</i>	UDINT	index prvního čteného byte v zápisníku
	<i>SizeOfDWrite</i>	UINT	velikost proměnné pro zápis (max. 1185)
VAR_IN_OUT			
	<i>Cmd</i>	TEpsnetLong-Command	výsledný příkaz v poli
	<i>DataRead</i>	USINT	cílová/zdrojová proměnná
	<i>DataWrite</i>	USINT	cílová/zdrojová proměnná
EpsnetEthLongCmdRW			
	<i>Návratová hodnota</i>	BOOL	nastavení příkazu pro čtení a zápis bloku bytů

5.8 Funkce *EpsnetMasCmd*









Knihovna : *EpsnetLib*

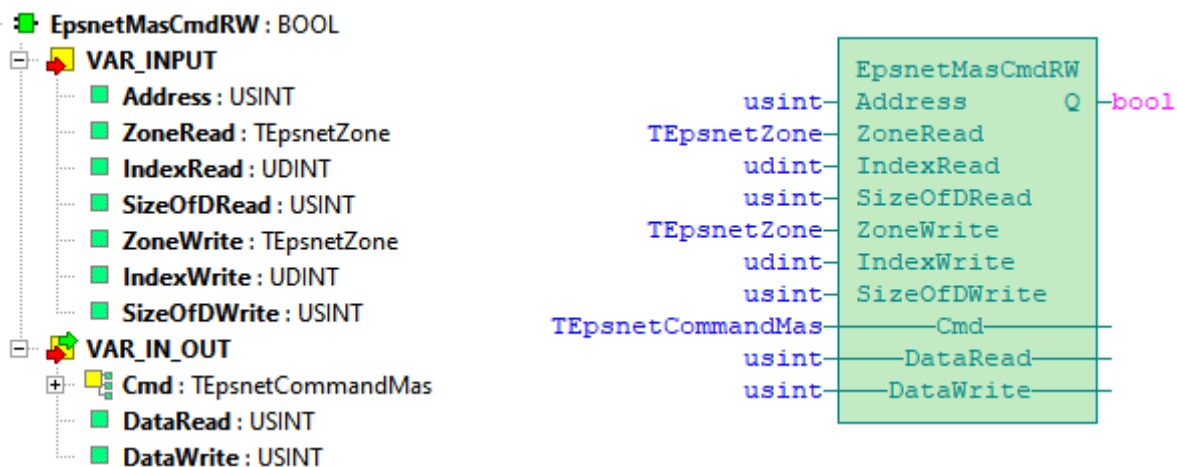
Funkce *EpsnetMasCmd* slouží k nastavení jedné položky v řídicím poli příkazů pro blok *fbEpsnetMas*. Funkce provede nastavení příkazu pro čtení nebo zápis bloku bytů. Vstup *Address* určuje adresu cílového PLC na seriové lince. Bit *Read* určuje zda budou data z PLC čtená (hodnota TRUE) nebo zapisována (hodnota FALSE).

Adresa bloku dat v paměti dotazovaného PLC je určena dvojicí vstupů *Zone* a *Index*. Vstup *SizeOfD* určuje délku komunikovaného bloku dat. Uživatel musí zajistit, aby proměnná, která slouží pro zápis či čtení dat a jejíž první byte se předává na vstupu *Data*, měla velikost minimálně rovnou hodnotě vstupu *SizeOfD*.

Odpovídající položka v poli příkazů se předává na vstupu *Cmd*.

Popis proměnných :

	Proměnná	Typ	Význam
VAR_INPUT			
	<i>Address</i>	USINT	adresa podřizené stanice
	<i>Read</i>	BOOL	FALSE = zápis, TRUE = čtení
	<i>Zone</i>	TEpsnetZone	oblast zápisníku
	<i>Index</i>	UDINT	index prvního čteného byte v zápisníku
	<i>SizeOfD</i>	USINT	velikost proměnné pro čtení (max. 246) nebo zápis (max. 241)
VAR_IN_OUT			
	<i>Cmd</i>	TEpsnetCommandMas	výsledný příkaz v poli
	<i>Data</i>	USINT	první byte cílové/zdrojové proměnné
EpsnetMasCmd			
	<i>Návratová hodnota</i>	BOOL	nastavení příkazu pro čtení nebo zápis bloku bytů pro blok fbEpsnetMas

5.9 Funkce *EpsnetMasCmdRW*Knihovna : *EpsnetLib*

Funkce *EpsnetMasCmdRW* slouží k nastavení jedné položky v řídicím poli příkazů pro blok *fbEpsnetMas*. Funkce provede nastavení příkazu pro současné čtení a zápis bloku bytů. Vstup *Address* určuje adresu cílového PLC na seriové lince.

Adresy bloků dat v paměti dotazovaného PLC jsou určena dvojicí vstupů *ZoneRead* a *IndexRead* pro čtený blok a *ZoneWrite* a *IndexWrite*. Vstupy *SizeOfDRead* a *SizeOfDWrite* určují délku komunikovaných bloků dat. Uživatel musí zajistit, aby proměnná, která slouží pro zápis dat a jejíž první byte se předává na vstupu *DataWrite*, měla velikost minimálně rovnou hodnotě vstupu *SizeOfDWrite*. Stejně tak platí, že proměnná, která slouží pro čtení dat a jejíž první byte se předává na vstupu *DataRead* musí mít velikost minimálně rovnou hodnotě vstupu *SizeOfDRead*.

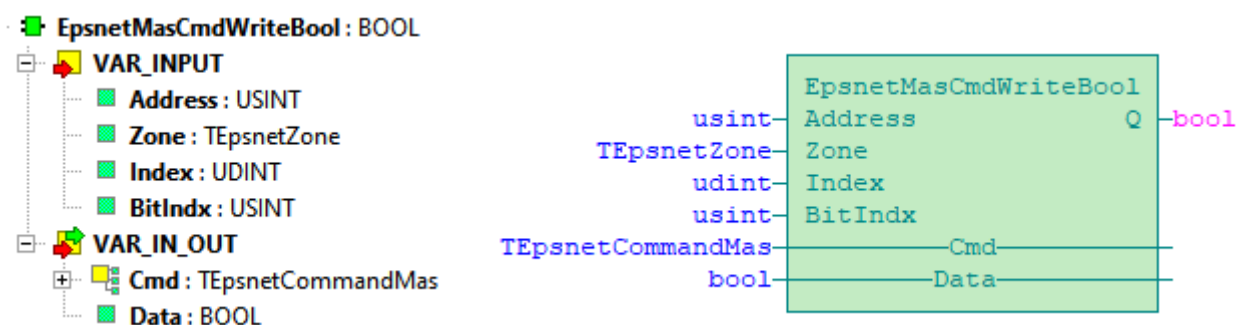
Odpovídající položka v poli příkazů se předává na vstupu *Cmd*.

Popis proměnných :

	Proměnná	Typ	Význam
VAR_INPUT			
📌	Address	USINT	adresa podřízené stanice
📌	ZoneRead	TEpsnetZone	oblast zápisníku
📌	IndexRead	UDINT	index prvního čteného byte v zápisníku
📌	SizeOfDRead	USINT	velikost proměnné pro čtení (max. 246)
📌	ZoneWrite	TEpsnetZone	oblast zápisníku
📌	IndexWrite	UDINT	index prvního čteného byte v zápisníku
📌	SizeOfDWrite	USINT	velikost proměnné pro zápis (max. 237)
VAR_IN_OUT			
📌	Cmd	TEpsnetCommand- Mas	výsledný příkaz v poli
📌	DataRead	USINT	cílová/zdrojová proměnná
📌	DataWrite	USINT	cílová/zdrojová proměnná
EpsnetMasCmdRW			
📌	Návratová hodnota	BOOL	nastavení příkazu pro čtení a zápis bloku bytů pro blok fbEpsnetMas

5.10 Funkce EpsnetMasCmdWriteBool

Knihovna : EpsnetLib










Funkce *EpsnetMasCmdWriteBool* slouží k nastavení jedné položky v řídicím poli příkazů pro blok *fbEpsnetMas*. Funkce provede nastavení příkazu pro zápis jednoho bitu. Tento zápis neovlivní hodnoty ostatních bitů v bytu, do kterého patří.

Vstup *Address* určuje adresu cílového PLC na seriové lince.

Adresa zapisovaného bitu v paměti dotazovaného PLC je určena trojicí vstupů *Zone*, *Index* a *BitIdx*. Vstup *BitIdx* je indexem bitu v rámci byte a může nabývat hodnot 0 až 7. Uživatel musí zajistit, aby proměnná, která se předává na vstupu *Data*, byla typu *BOOL*.

Odpovídající položka v poli příkazů se předává na vstupu *Cmd*.

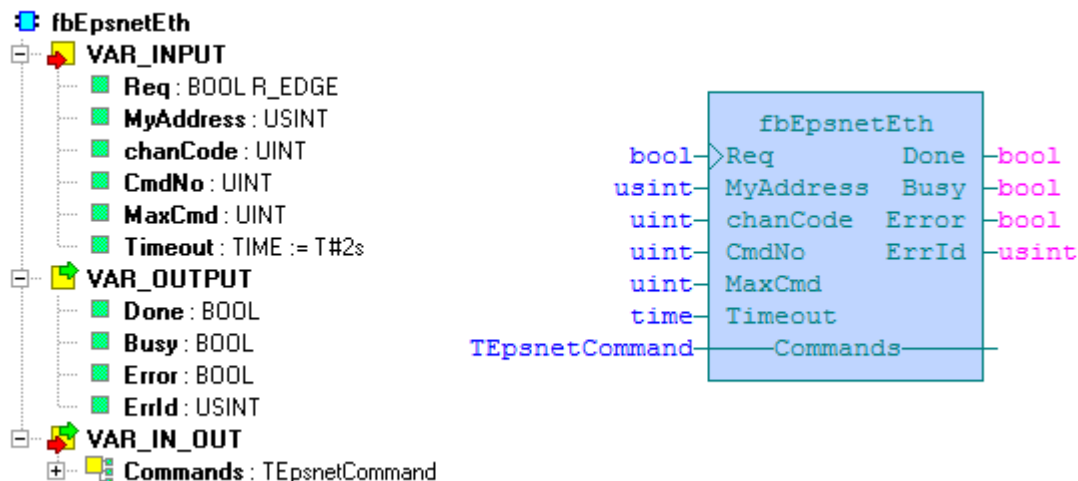
Popis proměnných :

	Proměnná	Typ	Význam
VAR_INPUT			
	<i>Address</i>	USINT	adresa podřízené stanice
	<i>Zone</i>	TEpsnetZone	oblast zápisníku
	<i>Index</i>	UDINT	index prvního čteného byte v zápisníku
	<i>BitIndx</i>	USINT	index bitu v byte (0 až 7)
VAR_IN_OUT			
	<i>Cmd</i>	TEpsnetCommandMas	výsledný příkaz v poli
	<i>Data</i>	BOOL	zdrojová proměnná (musí být typu BOOL!)
EpsnetMasCmdWriteBool			
	<i>Návratová hodnota</i>	BOOL	nastavení příkazu pro zápis jednoho bitu pro blok fbEpsnetMas

6 FUNKČNÍ BLOKY

V knihovně EpsnetLib je definován následující funkční blok:

Funkční blok	Popis
<i>fbEpsnetEth</i>	Zakládní blok pro komunikaci protokolem EPSNET UDP v roli nadřazené stanice, využívá jednu zprávu na UDP paket
<i>fbEpsnetMultiEth</i>	Rozšířený blok <i>fbEpsnetEth</i> o možnost odeslat až 5 EPSNET zpráv jedním UDP paketem
<i>fbEpsnetLongEth</i>	Blok pro čtení a zápis delších souvislých zón. Automaticky skládá příkazy čtení a zápisu, aby plně využil možnosti vložit až 5 EPSNET zpráv do jednoho UDP paketu
<i>fbEpsnetMas</i>	Blok pro komunikaci protokolem EPSNET po sériové lince
<i>fbEpsnetSlave</i>	Základní implementace podřazené stanice EPSNET (PC režim) na UNI kanálu.

6.1 Funkční blok *fbEpsnetEth*Knihovna : *EpsnetLib*

Funkční blok *fbEpsnetEth* slouží ke pro komunikaci protokolem EPSNET UDP v pozici nadřazené stanice. Blok implementuje ze souboru služeb protokolu EPSNET následující služby: READN – čtení z datové paměti, WRITEN – zápis do datové paměti, WANDRN – zápis do a čtení z datové paměti, WRITEB – zápis do bitů datové paměti. Funkční blok si volí automaticky odpovídající službu podle požadovaného příkazu. Příkazy jsou definovány v poli, jehož první položka se předává na vstupu *Cmd*. Uživatel musí zajistit, aby rozměr pole příkazů byl větší nebo roven hodnotě vstupu *MaxCmd*.












Jednotlivé příkazy se definují pomocí funkcí *EpsnetEthCmd*, *EpsnetEthCmdRW*, *EpsnetEthCmdWriteBool*.

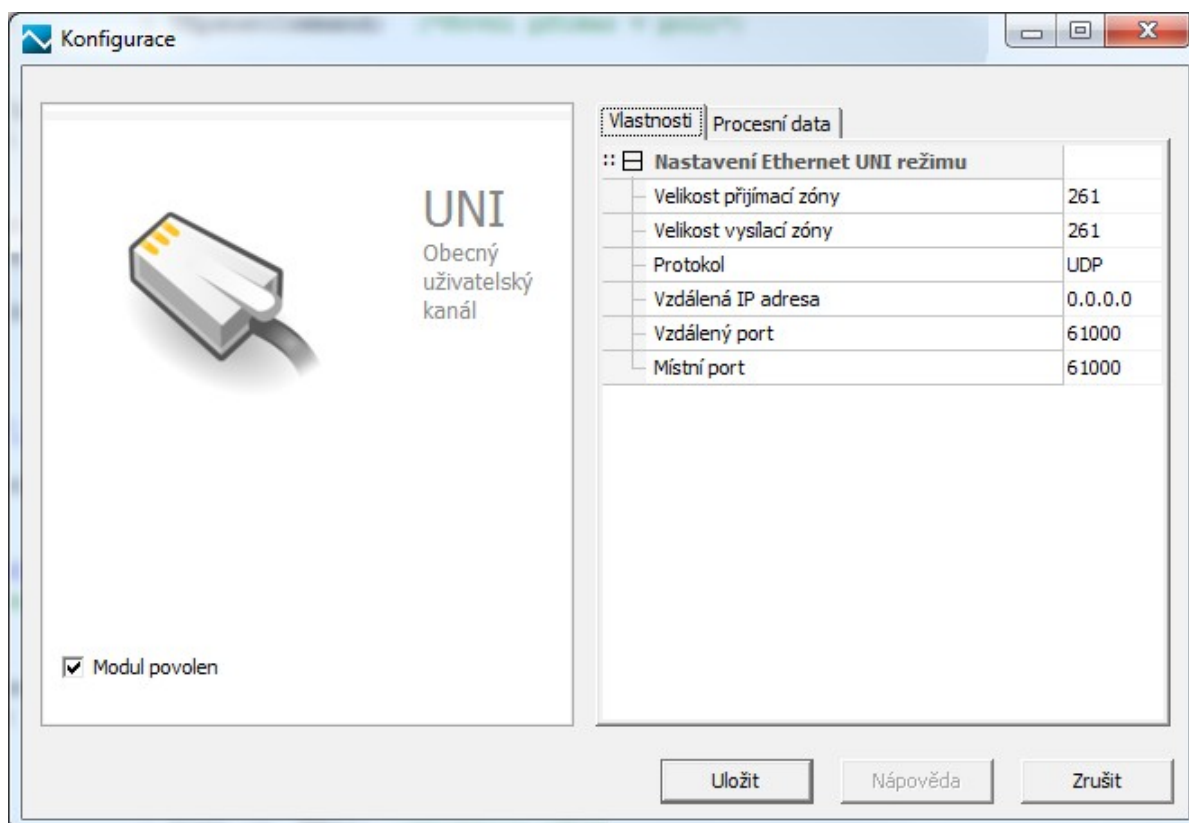
Komunikace probíhá přes spojení na ethernet kanálu v režimu UNI podle konstanty na vstupu *chanCode*. Spojení musí mít následujícími parametry: režim UDP, délka přijímací a vysílací zóny 261 bytů. Pokud spojení není aktivní nebo nemá správné délky zón, blok indikuje chybu na výstupech *Err* hodnotou TRUE a *ErrId* hodnotou 255. Adresu a porty nastavuje blok dle parametrů jednotlivých příkazů.

Vstup *MyAddress* udává zdrojovou adresu v protokolu EPSNET. Tato adresa, v případě použití tunelování na sériové rozhraní podřazených stanic přes režim MPC, nesmí být stejná jako adresa podřazených stanic. V případě běžné komunikace může být ponechána výchozí hodnota – nula.

Vykonání příkazu se spustí náběžnou hranou na vstupu *Req*. Číslo příkazu, který se má vykonat, je dáno vstupem *CmdNo* (prvnímu příkazu odpovídá číslo 0). Po dobu komunikace je nastaven výstup *Busy*. Po ukončení komunikace je nastaven na dobu jednoho cyklu výstup *Done* v případě úspěšné komunikace nebo výstup *Error* v případě chybné odpovědi nebo pokud odpověď nedorazí v čase daném vstupem *Timeout*. Chybu komunikace blíže specifikuje výstup *ErrId*. Významy jednotlivých hodnot výstup *ErrId* jsou uvedeny v tabulce popisu proměnných.

Popis proměnných :

	Proměnná	Typ	Význam
VAR_INPUT			
	Req	BOOL R_EDGE	Spuštění příkazu
	MyAddress	USINT	Vlastní adresa
	chanCode	UINT	Kód kanálu <i>ETH1_uni0, ..., ETH1_uni7, ETH2_uni0, ..., ETH2_uni7</i>
	CmdNo	UINT	Index příkazu v poli (0 až <i>MaxCmd-1</i>)
	MaxCmd	UINT	Počet příkazů v poli
	Timeout	TIME	Timeout příkazu
VAR_IN_OUT			
	Cmd	TEpsnetCommand	První příkaz v poli
VAR_OUTPUT			
	Done	BOOL	Nastaví se na TRUE po dobu jednoho cyklu pokud komunikace proběhla úspěšně
	Busy	BOOL	TRUE znamená, že probíhá komunikace
	Error	BOOL	Nastaví se na TRUE po dobu jednoho cyklu pokud nastala chyba
	ErrId	USINT	Chybový kód: <i>errID = 0</i> bez chyby <i>errID = 1</i> odpověď nepřišla v čase <i>Timeout</i> <i>errID = 2</i> přijatá zpráva má chybný kontrolní součet <i>errID = 3</i> <i>CmdNo</i> \geq <i>MaxCmd</i> <i>errID = 4</i> neodpovídající číslo odpovědi <i>errID = 5</i> neočekávaná odpověď <i>errID = 6</i> přijatá zpráva má neplatnou délku <i>errID = 12..33</i> chybové odpovědi z protokolu EPSNET <i>errID = 252</i> neplatný index bitu <i>errID = 253</i> překročena maximální délka zprávy <i>errID = 254</i> neznámá služba <i>errID = 255</i> chybné nastavení spojení na ethernet kanálu



Nastavení spojení na ethernetovém kanálu v režimu UNI pro funkční blok fbEpsnetEth

Příklad programu s funkčním blokem *fbEpsnetEth*. Program realizuje distribuci dat do tří podřízených systémů. Zpětně je vyčítán čítač sekund ze systémových registrů. V příkladu je použita proměnná z knihovny *SysLib*:

```

VAR_GLOBAL CONSTANT
  MAX_CMD3 : UINT := 3;
END_VAR

VAR_GLOBAL
  Commands3 : ARRAY [0..MAX_CMD3-1] OF TEpsnetCommand;
  CommandDataExchange1 AT Commands3[0];
  CommandDataExchange2 AT Commands3[1];
  CommandDataExchange3 AT Commands3[2];
END_VAR

PROGRAM prgExampleEpsnetEth
  VAR
    EpsnetEth : fbEpsnetEth;
    IPAdrs : ARRAY [0..2] OF TIPadr :=
      [[192,168,33,154],[192,168,33,175],[192,168,33,177]];
    Ticks : ARRAY [0..2] OF USINT;
    Data : ARRAY [0..99] OF USINT;
    ErrCnt : USINT;
    CmdNo : UINT;
  END_VAR
  VAR_TEMP
  END_VAR

  EpsnetEthCmdRW (IpAdr := IPAdrs[0],

```

```
ZoneRead := epszS,
IndexRead := 6,
SizeOfDRead := 1,
ZoneWrite := epszR,
IndexWrite := 30000,
SizeOfDWrite := SIZEOF(Data),
Cmd := CommandDataExchange1,
DataRead := Ticks[0],
DataWrite := void(Data));

EpsnetEthCmdRW(IpAdr := IPAdrs[1],
ZoneRead := epszS,
IndexRead := 6,
SizeOfDRead := 1,
ZoneWrite := epszR,
IndexWrite := 30000,
SizeOfDWrite := SIZEOF(Data),
Cmd := CommandDataExchange2,
DataRead := Ticks[1],
DataWrite := void(Data));

EpsnetEthCmdRW(IpAdr := IPAdrs[2],
ZoneRead := epszS,
IndexRead := 6,
SizeOfDRead := 1,
ZoneWrite := epszR,
IndexWrite := 30000,
SizeOfDWrite := SIZEOF(Data),
Cmd := CommandDataExchange3,
DataRead := Ticks[2],
DataWrite := void(Data));

EpsnetEth(Req := System_S.R_EDGE_1SEC,
chanCode := ETH1_uni3, CmdNo := CmdNo,
MaxCmd := MAX_CMD3, Timeout := T#700ms,
Commands := Commands3[0]);

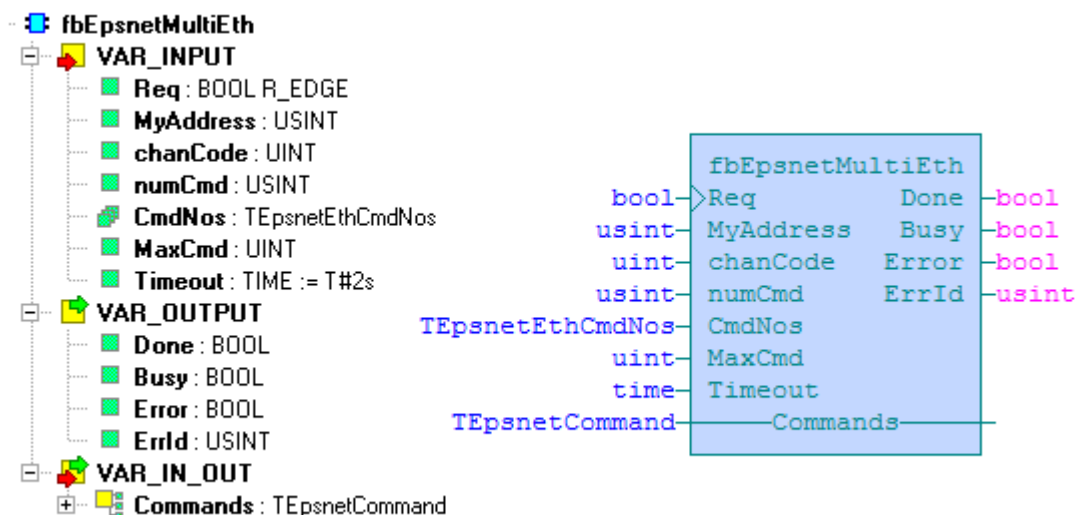
Data[0] := Ticks[0];
Data[1] := Ticks[1];
Data[2] := Ticks[2];

IF EpsnetEth.Error THEN
  ErrCnt := ErrCnt + 1;
END_IF;

IF EpsnetEth.Done OR ErrCnt > 1 THEN
  CmdNo := CmdNo + 1;
  IF CmdNo = MAX_CMD3 THEN
    CmdNo := 0;
  END_IF;
  ErrCnt := 0;
END_IF;

END_PROGRAM
```

6.2 Funkční blok *fbEpsnetMultiEth*

Knihovna : *EpsnetLib*

Funkční blok *fbEpsnetMultiEth* slouží ke pro komunikaci protokolem EPSNET UDP v pozici nadřazené stanice. Blok implementuje stejné služby jako *fbEpsnetEth*, ale umožňuje odeslat až pět příkazů najednou. K tomu využívá skládání více EPSNET zpráv do jednoho UDP paketu. Podmínkou tedy je, aby všechny příkazy byly pro jednu IP adresu.













Stejně jako pro *fbEpsnetEth* jsou příkazy definovány v poli, jehož první položka se předává na vstupu *Cmd*. Uživatel musí zajistit, aby rozměr pole příkazů byl větší nebo roven hodnotě vstupu *MaxCmd*.

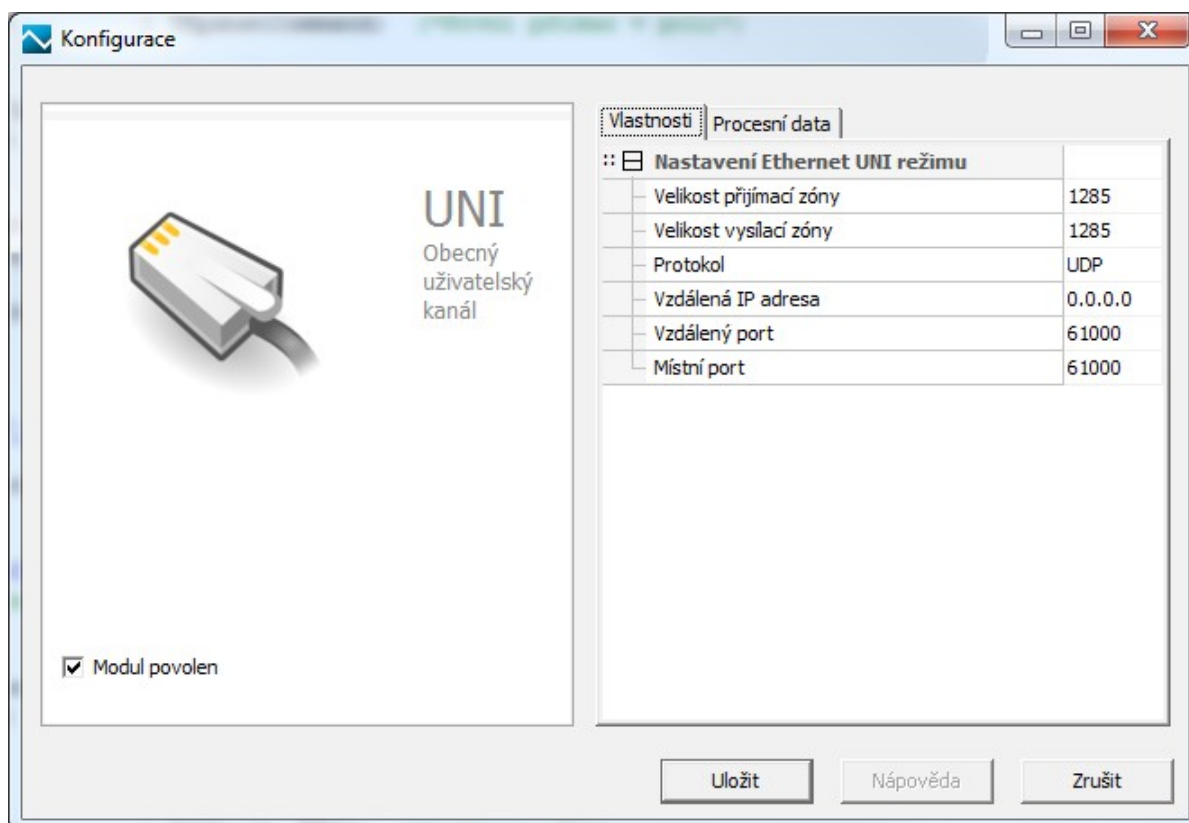
Pro definici příkazů se používají stejné funkce *EpsnetEthCmd*, *EpsnetEthCmdRW*, *EpsnetEthCmdWriteBool*.

Komunikace probíhá přes spojení na ethernet kanálu v režimu UNI podle konstanty na vstupu *chanCode*. Spojení musí mít následujícími parametry: režim UDP, délka přijímací a vysílací zóny 1285 bytů. Pokud spojení není aktivní nebo nemá správné délky zón, blok indikuje chybu na výstupech *Err* hodnotou TRUE a *ErrId* hodnotou 255. Adresu a porty nastavuje blok dle parametrů jednotlivých příkazů.

Vykonání příkazů se spustí náběžnou hranou na vstupu *Req*. Číslo příkazu, které se má vykonat, jsou dána vstupem *CmdNos* (prvnímu příkazu odpovídá číslo 0). Na vstupu *CmdNos* se očekává vždy pole pěti čísel. Počet zároveň vykonaných příkazů může být nižší a je dán vstupem *numCmd*. Po dobu komunikace je nastaven výstup *Busy*. Po ukončení komunikace je nastaven na dobu jednoho cyklu výstup *Done* v případě úspěšné komunikace nebo výstup *Error* v případě chybné odpovědi nebo pokud odpověď nedorazí v čase daném vstupem *Timeout*. Chybu komunikace blíže specifikuje výstup *ErrId*. Významy jednotlivých hodnot výstup *ErrId* jsou uvedeny v tabulce popisu proměnných.

Popis proměnných :

	Proměnná	Typ	Význam
VAR_INPUT			
	<i>Req</i>	BOOL R_EDGE	Spuštění příkazů
	<i>MyAddress</i>	USINT	Vlastní adresa
	<i>chanCode</i>	UINT	Kód kanálu ETH1_uni0, ..., ETH1_uni7, ETH2_uni0, ..., ETH2_uni7
	<i>numCmd</i>	USINT	Počet příkazů k vykonání z pole <i>CmdNos</i>
	<i>CmdNos</i>	TEpsnetEthCmdNos	Indexy příkazů v poli (0 až <i>MaxCmd</i> -1)
	<i>MaxCmd</i>	UINT	Počet příkazů v poli
	<i>Timeout</i>	TIME	Timeout příkazu
VAR_OUTPUT			
	<i>Done</i>	BOOL	Nastaví se na TRUE po dobu jednoho cyklu pokud komunikace proběhla úspěšně
	<i>Busy</i>	BOOL	TRUE znamená, že probíhá komunikace
	<i>Error</i>	BOOL	Nastaví se na TRUE po dobu jednoho cyklu pokud nastala chyba
	<i>ErrId</i>	USINT	Chybový kód: <i>errID</i> = 0 bez chyby <i>errID</i> = 1 odpověď nepřišla v čase <i>Timeout</i> <i>errID</i> = 2 přijatá zpráva má chybný kontrolní součet <i>errID</i> = 3 <i>CmdNo</i> >= <i>MaxCmd</i> <i>errID</i> = 4 neodpovídající číslo odpovědi <i>errID</i> = 5 neočekávaná odpověď <i>errID</i> = 6 přijatá zpráva má neplatnou délku <i>errID</i> = 12..33 chybové odpovědi z protokolu EPSNET <i>errID</i> = 252 neplatný index bitu <i>errID</i> = 253 překročena maximální délka zprávy <i>errID</i> = 254 neznámá služba <i>errID</i> = 255 chybné nastavení spojení na ethernet kanálu
VAR_IN_OUT			
	<i>Commands</i>	TEpsnet-Command	První příkaz v poli



*Nastavení spojení na ethernetovém kanálu v režimu UNI
pro funkční blok fbEpsnetMultiEth*

Příklad programu s funkčním blokem *fbEpsnetMultiEth*. Program realizuje distribuci dat do dvou podřízených systémů. Ze systému s IP adresou 192.168.134.193 je jednou zprávou vyčten obraz vstupů ze zóny registrů X a zapsán obraz výstupů do zóny registrů Y společně se zápisem a čtením zóny registrů R. V systému s adresou 192.168.134.194 jsou čteny a zapisovány registry v zóně R a zároveň je ve čtené zóně aleternován bit. Díky tomu, že čtení a zápis je odesláno v jednom paketu, jsou operace synchronní a alternaci bitu lze v podřízeném systému využít pro detekci nového obsahu v zapisovaných registrech.

```
PROGRAM prgTestEpsnetMulti
VAR
  EpsnetMultiEth : fbEpsnetMultiEth;

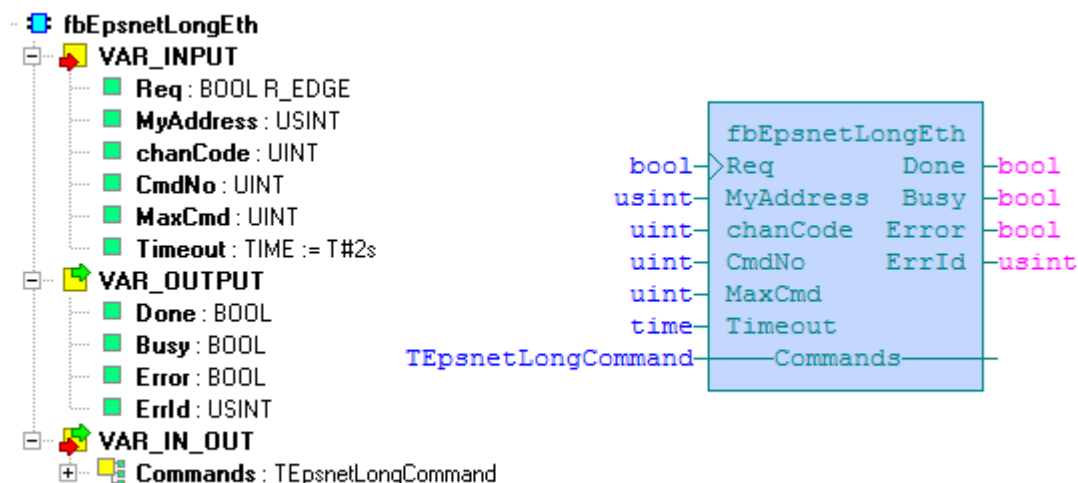
  Commands : ARRAY [0..7] OF TEpsnetCommand;

  ipAdr : ARRAY [0..1] OF TIPAdr :=
    [[192,168,134,193],[192,168,134,194]];
  cmdNos : ARRAY [0..1] OF TEpsnetEthCmdNos :=
    [[0,1,2,3,4],[5,6,7,0,0]];
  cmdNums : ARRAY [0..1] OF USINT :=
    [5,3];
  idx : USINT;
```

```
imageOfX : ARRAY [0..479] OF USINT;  
imageOfY : ARRAY [0..479] OF USINT;  
dataToRead1 : ARRAY [0..119] OF USINT;  
dataToRead2 : ARRAY [0..119] OF USINT;  
dataToWrite : ARRAY [0..59] OF USINT;  
start      : BOOL := 1;  
alt        : BOOL;  
  
trueMes   : UDINT;  
falseMes  : UDINT;  
END_VAR  
  
EpsnetEthCmd(IpAdr := ipAdr[0],  
  Read := true, Zone := TEpsnetZone#epszX,  
  Index := 0, SizeOfD := 240,  
  Cmd := Commands[0], Data := void(imageOfX[0]));  
  
EpsnetEthCmd(IpAdr := ipAdr[0],  
  Read := true, Zone := TEpsnetZone#epszX,  
  Index := 240, SizeOfD := 240,  
  Cmd := Commands[1], Data := void(imageOfX[240]));  
  
EpsnetEthCmd(IpAdr := ipAdr[0],  
  Read := false, Zone := TEpsnetZone#epszY,  
  Index := 0, SizeOfD := 240,  
  Cmd := Commands[2], Data := void(imageOfY[0]));  
  
EpsnetEthCmd(IpAdr := ipAdr[0],  
  Read := false, Zone := TEpsnetZone#epszY,  
  Index := 240, SizeOfD := 240,  
  Cmd := Commands[3], Data := void(imageOfY[240]));  
  
EpsnetEthCmdRW(IpAdr := ipAdr[0],  
  ZoneRead := TEpsnetZone#epszR,  
  IndexRead := 60000,  
  SizeOfDRead := SIZEOF(dataToRead1),  
  ZoneWrite := TEpsnetZone#epszR,  
  IndexWrite := 61000,  
  SizeOfDWrite := SIZEOF(dataToWrite),  
  Cmd := Commands[4],  
  DataRead := void(dataToRead1),  
  DataWrite := void(dataToWrite));  
  
EpsnetEthCmd(IpAdr := ipAdr[1],  
  Read := false, Zone := TEpsnetZone#epszR,  
  Index := 61000, SizeOfD := SIZEOF(dataToWrite),  
  Cmd := Commands[5], Data := void(dataToWrite));  
  
EpsnetEthCmd(IpAdr := ipAdr[1],  
  Read := true, Zone := TEpsnetZone#epszR,  
  Index := 60000, SizeOfD := SIZEOF(dataToRead2),  
  Cmd := Commands[6], Data := void(dataToRead2));  
  
EpsnetEthCmdWriteBool(IpAdr := ipAdr[1],  
  Zone := TEpsnetZone#epszR,  
  Index := 60000, BitIndx := 0,  
  Cmd := Commands[7], Data := alt);
```

```
EpsnetMultiEth(  
  chanCode := ETH1_uni1,  
  numCmd := cmdNums[idx],  
  CmdNos := cmdNos[idx],  
  MaxCmd := 8,  
  Commands := Commands[0]);  
  
EpsnetMultiEth.Req := false;  
  
IF start OR EpsnetMultiEth.Done OR EpsnetMultiEth.Error THEN  
  start := false;  
  idx := idx + 1;  
  IF EpsnetMultiEth.Error THEN  
    falseMes := falseMes + 1;  
  ELSIF EpsnetMultiEth.Done THEN  
    trueMes := trueMes + 1;  
    IF idx = 2 THEN  
      alt := NOT dataToRead2[0].0;  
      idx := 0;  
    END_IF;  
  END_IF;  
  EpsnetMultiEth.Req := true;  
END_IF;  
  
END_PROGRAM
```

6.3 Funkční blok *fbEpsnetLongEth*

Knihovna : *EpsnetLib*

Funkční blok *fbEpsnetLongEth* slouží ke pro komunikaci protokolem EPSNET UDP v pozici nadřazené stanice. Blok je určen pro přenos větších datových zón a implementuje služby pro čtení a zápis registrů, jmenovitě READN – čtení z datové paměti, WRITEN – zápis do datové paměti a WANDRN – zápis do a čtení z datové paměti. Narozdíl od bloků *fbEpsnetEth* a *fbEpsnetMultiEth* položky v poli příkazů nemusí odpovídat jedné zprávě EPSNET, ale popisují přenos jedním UDP paketem, který může obsahovat až 5 EPSNET zpráv. Výsledkem je možnost jednou komunikací číst zóny velikosti až 1230 bytů a zapisovat zóny velikosti 1205 bytů (resp. 1185 pokud je zápis kombinován se čtením).












Příkazy jsou definovány v poli, jehož první položka se předává na vstupu *Cmd*. Uživatel musí zajistit, aby rozměr pole příkazů byl větší nebo roven hodnotě vstupu *MaxCmd*.

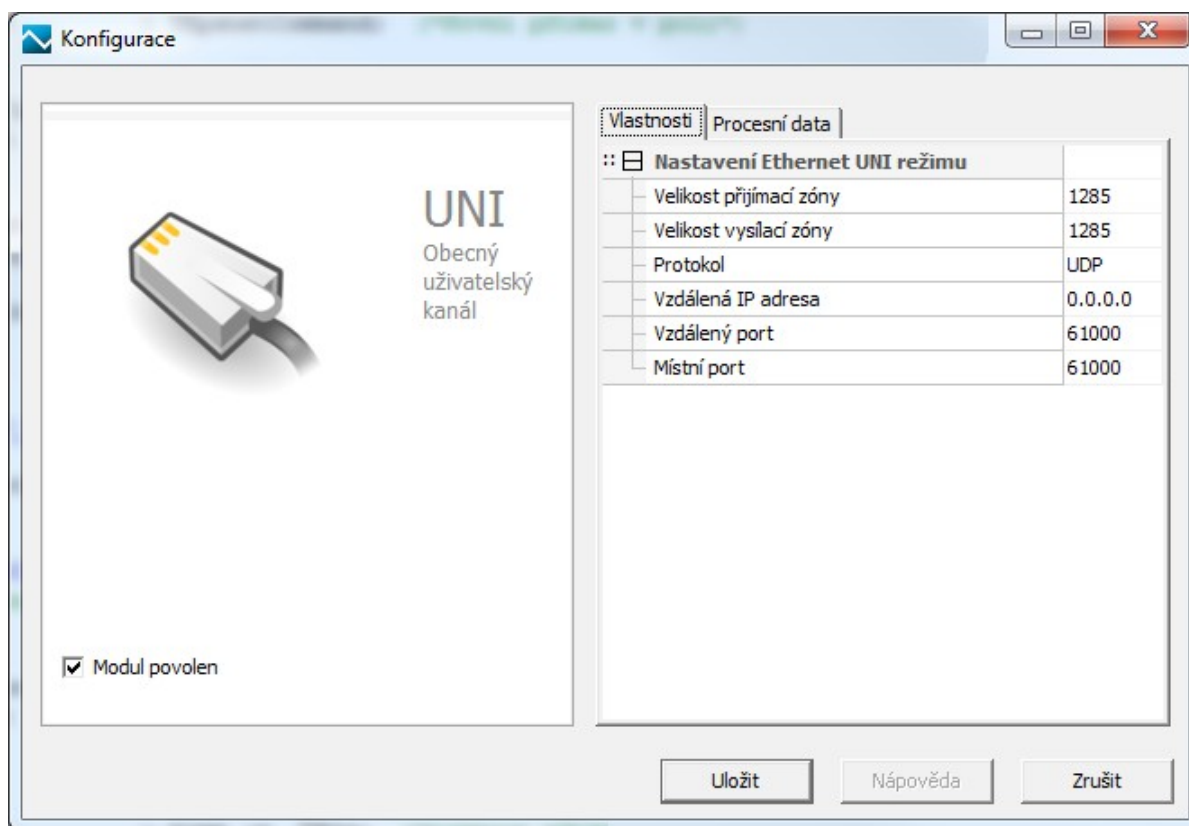
Jednotlivé příkazy se definují pomocí funkcí *EpsnetEthLongCmd*, *EpsnetEthCmdLongRW*.

Komunikace probíhá přes spojení na ethernet kanálu v režimu UNI podle konstanty na vstupu *chanCode*. Spojení musí mít následujícími parametry: režim UDP, délka přijímací a vysílací zóny 1285 bytů. Pokud spojení není aktivní nebo nemá správné délky zón, blok indikuje chybu na výstupech *Err* hodnotou TRUE a *ErrId* hodnotou 255. Adresu a porty nastavuje blok dle parametrů jednotlivých příkazů.

Vykonání příkazu se spustí náběžnou hranou na vstupu *Req*. Číslo příkazu, který se má vykonat, je dáno vstupem *CmdNo* (prvnímu příkazu odpovídá číslo 0). Po dobu komunikace je nastaven výstup *Busy*. Po ukončení komunikace je nastaven na dobu jednoho cyklu výstup *Done* v případě úspěšné komunikace nebo výstup *Error* v případě chybné odpovědi nebo pokud odpověď nedorazí v čase daném vstupem *Timeout*. Chybu komunikace blíže specifikuje výstup *ErrId*. Významy jednotlivých hodnot výstup *ErrId* jsou uvedeny v tabulce popisu proměnných.

Popis proměnných :

	Proměnná	Typ	Význam
VAR_INPUT			
	<i>Req</i>	BOOL R_EDGE	Spuštění příkazu
	<i>MyAddress</i>	USINT	Vlastní adresa
	<i>chanCode</i>	UINT	Kód kanálu ETH1_uni0, ..., ETH1_uni7, ETH2_uni0, ..., ETH2_uni7
	<i>CmdNo</i>	UINT	Index příkazu v poli (0 až <i>MaxCmd</i> -1)
	<i>MaxCmd</i>	UINT	Počet příkazů v poli
	<i>Timeout</i>	TIME	Timeout příkazu
VAR_OUTPUT			
	<i>Done</i>	BOOL	Nastaví se na TRUE po dobu jednoho cyklu pokud komunikace proběhla úspěšně
	<i>Busy</i>	BOOL	TRUE znamená, že probíhá komunikace
	<i>Error</i>	BOOL	Nastaví se na TRUE po dobu jednoho cyklu pokud nastala chyba
	<i>ErrId</i>	USINT	Chybový kód: <i>errID</i> = 0 bez chyby <i>errID</i> = 1 odpověď nepřišla v čase <i>Timeout</i> <i>errID</i> = 2 přijatá zpráva má chybný kontrolní součet <i>errID</i> = 3 <i>CmdNo</i> >= <i>MaxCmd</i> <i>errID</i> = 4 neodpovídající číslo odpovědi <i>errID</i> = 5 neočekávaná odpověď <i>errID</i> = 6 přijatá zpráva má neplatnou délku <i>errID</i> = 12..33 chybové odpovědi z protokolu EPSNET <i>errID</i> = 252 neplatný index bitu <i>errID</i> = 253 překročena maximální délka zprávy <i>errID</i> = 254 neznámá služba <i>errID</i> = 255 chybné nastavení spojení na ethernet kanálu
VAR_IN_OUT			
	<i>Commands</i>	TEpsnet-Long-Command	První příkaz v poli



*Nastavení spojení na ethernetovém kanálu v režimu UNI
pro funkční blok fbEpsnetLongEth*

Příklad programu s funkčním blokem *fbEpsnetLongEth*. Program realizuje distribuci dat do dvou podřízených systémů. Ze systému s IP adresou 192.168.134.193 je jednou zprávou vyčten obraz vstupů ze zóny registrů X a zapsán obraz výstupů do zóny registrů Y. V systému s adresou 192.168.134.194 jsou čteny registry dvěma zprávami (velikost přesahuje maximální délku přenositelnou jedním paketem). Společně s vyčtením druhé části je zpětně zapisován registr s číslem inkrementovaným při každém přenosu, umožňující kontrolu komunikace na podřízené straně.

```
PROGRAM prgTestEpsnetLong
VAR
  EpsnetLongEth : fbEpsnetLongEth;

  Commands : ARRAY [0..2] OF TEpsnetLongCommand;

  ipAdr : ARRAY [0..1] OF TIPAdr :=
    [[192,168,134,193],[192,168,134,194]];
  imageOfX : ARRAY [0..1199] OF USINT;
  imageOfY : ARRAY [0..479] OF USINT;
  dataToRead : ARRAY [0..1999] OF USINT;
  dataToWrite : USINT;
  start : BOOL := 1;

  trueMes : UDINT;
  falseMes : UDINT;
END_VAR

EpsnetEthLongCmdRW(IpAdr := ipAdr[0],
```

```
ZoneRead := TEpsnetZone#epszX,
IndexRead := 0,
SizeOfDRead := SIZEOF(imageOfX),
ZoneWrite := TEpsnetZone#epszY,
IndexWrite := 0,
SizeOfDWrite := SIZEOF(imageOfY),
Cmd := Commands[0],
DataRead := void(imageOfX[0]),
DataWrite := void(imageOfY[0]);

EpsnetEthLongCmd(IpAdr := ipAdr[1],
Read := true, Zone := TEpsnetZone#epszR,
Index := 60000, SizeOfD := 1230,
Cmd := Commands[1], Data := void(dataToRead[0]));

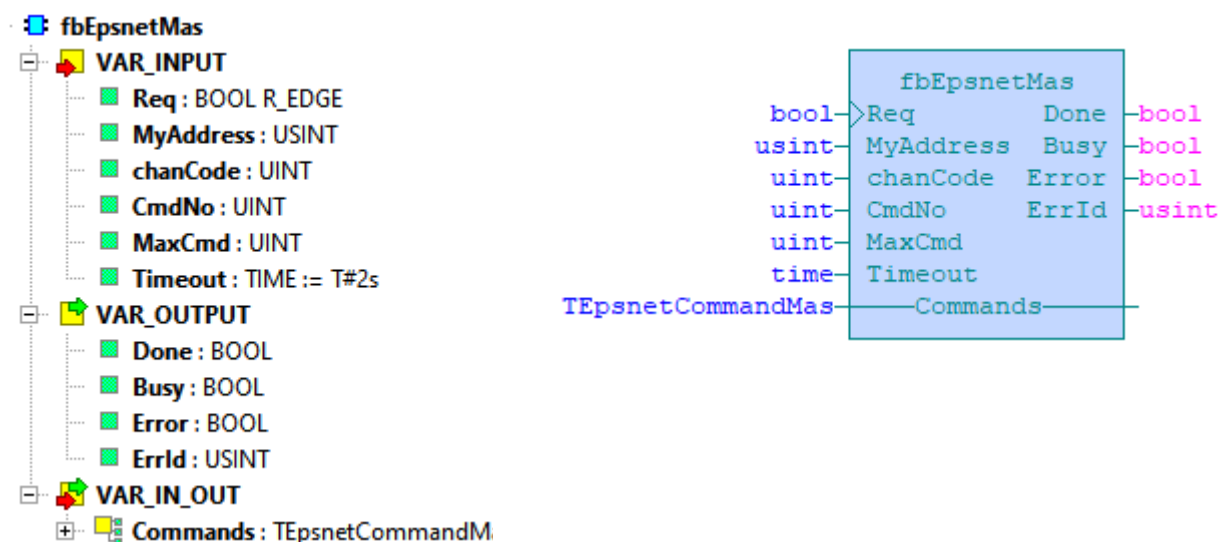
EpsnetEthLongCmdRW(IpAdr := ipAdr[1],
ZoneRead := TEpsnetZone#epszR,
IndexRead := 61230,
SizeOfDRead := SIZEOF(dataToWrite),
ZoneWrite := TEpsnetZone#epszR,
IndexWrite := 60000,
SizeOfDWrite := 770,
Cmd := Commands[2],
DataRead := void(dataToRead[1230]),
DataWrite := void(dataToWrite));

EpsnetLongEth(
chanCode := ETH1_uni1,
MaxCmd := 3,
Commands := Commands[0]);

EpsnetLongEth.Req := false;

IF start OR EpsnetLongEth.Done OR EpsnetLongEth.Error THEN
start := false;
IF EpsnetLongEth.Error THEN
falseMes := falseMes + 1;
ELSIF EpsnetLongEth.Done THEN
trueMes := trueMes + 1;
IF EpsnetLongEth.CmdNo = 2 THEN
dataToWrite := dataToWrite + 1;
END_IF;
dataToWrite := dataToWrite + 1;
END_IF;
EpsnetLongEth.CmdNo := EpsnetLongEth.CmdNo + 1;
IF EpsnetLongEth.CmdNo = 3 THEN
EpsnetLongEth.CmdNo := 0;
END_IF;
EpsnetLongEth.Req := true;
END_IF;

END_PROGRAM
```

6.4 Funkční blok *fbEpsnetMas*Knihovna : *EpsnetLib*

Funkční blok *fbEpsnetMas* slouží ke pro komunikaci protokolem EPSNET v pozici nadřazené stanice. Blok implementuje ze souboru služeb protokolu EPSNET následující služby: READN – čtení z datové paměti, WRITEN – zápis do datové paměti, WANDRN – zápis do a čtení z datové paměti, WRITEB – zápis do bitů datové paměti. Funkční blok si volí automaticky odpovídající službu podle požadovaného příkazu. Příkazy jsou definovány v poli, jehož první položka se předává na vstupu *Cmd*. Uživatel musí zajistit, aby rozměr pole příkazů byl větší nebo roven hodnotě vstupu *MaxCmd*.












Jednotlivé příkazy se definují pomocí funkcí *EpsnetMasCmd*, *EpsnetMasCmdRW*, *EpsnetMasCmdWriteBool*.

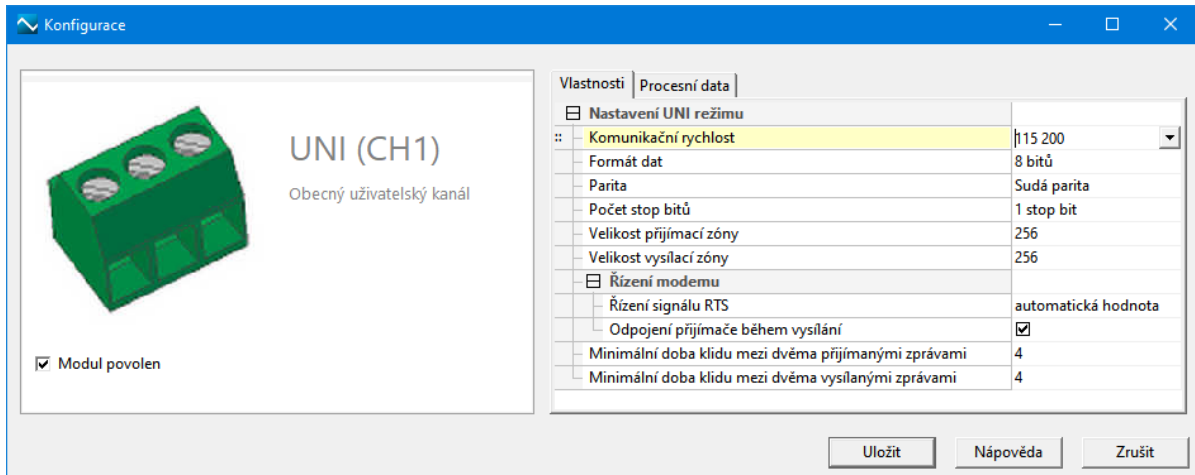
Komunikace probíhá přes spojení na seriovém kanálu v režimu UNI podle konstanty na vstupu *chanCode*. Spojení musí mít následujícími parametry: 8 bitů, sudá parita, 1 stop bit, délka přijímací a vysílací zóny 256 bytů. Pokud spojení není aktivní nebo nemá správné délky zón, blok indikuje chybu na výstupech *Err* hodnotou TRUE a *ErrId* hodnotou 255.

Vstup *MyAddress* udává zdrojovou adresu v protokolu EPSNET. Tato adresa, nesmí být stejná jako adresa podřazených stanic.

Vykonání příkazu se spustí náběžnou hranou na vstupu *Req*. Číslo příkazu, který se má vykonat, je dáno vstupem *CmdNo* (prvnímu příkazu odpovídá číslo 0). Po dobu komunikace je nastaven výstup *Busy*. Po ukončení komunikace je nastaven na dobu jednoho cyklu výstup *Done* v případě úspěšné komunikace nebo výstup *Error* v případě chybné odpovědi nebo pokud odpověď nedorazí v čase daném vstupem *Timeout*. Chybu komunikace blíže specifikuje výstup *ErrId*. Významy jednotlivých hodnot výstup *ErrId* jsou uvedeny v tabulce popisu proměnných.

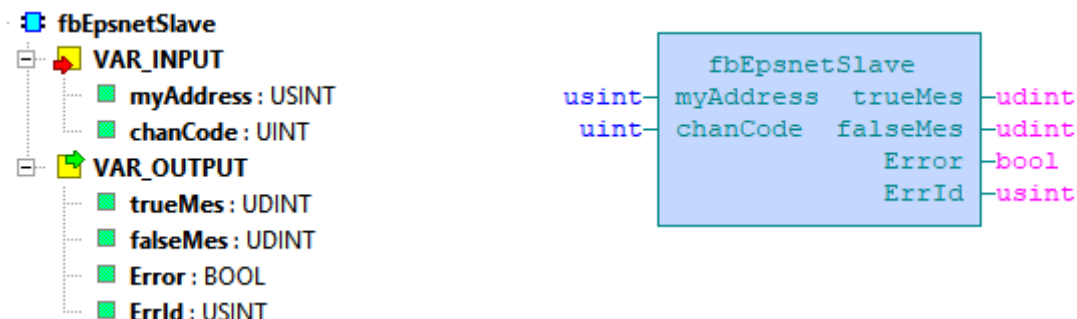
Popis proměnných :

	Proměnná	Typ	Význam
VAR_INPUT			
	<i>Req</i>	BOOL R_EDGE	spuštění příkazu
	<i>MyAddress</i>	USINT	vlastní adresa
	<i>chanCode</i>	UINT	kód kanálu CH1_uni, CH2_uni,...
	<i>CmdNo</i>	UINT	index příkazu v poli (0 až MaxCmd-1)
	<i>MaxCmd</i>	UINT	počet příkazů v poli
	<i>Timeout</i>	TIME	timeout příkazu
VAR_OUTPUT			
	<i>Done</i>	BOOL	komunikace proběhla úspěšně
	<i>Busy</i>	BOOL	probíhá komunikace
	<i>Error</i>	BOOL	nastala chyba
	<i>ErrId</i>	USINT	chybový kód: <i>errID</i> = 0 bez chyby <i>errID</i> = 1 odpověď nepřišla v čase <i>Timeout</i> <i>errID</i> = 2 přijatá zpráva má chybný kontrolní součet <i>errID</i> = 3 <i>CmdNo</i> >= <i>MaxCmd</i> <i>errID</i> = 4 neodpovídající číslo odpovědi <i>errID</i> = 5 neočekávaná odpověď <i>errID</i> = 6 přijatá zpráva má neplatnou délku <i>errID</i> = 12..33 chybové odpovědi z protokolu EPSNET <i>errID</i> = 252 neplatný index bitu <i>errID</i> = 253 překročena maximální délka zprávy <i>errID</i> = 254 neznámá služba <i>errID</i> = 255 chybné nastavení spojení na sériovém kanálu
VAR_IN_OUT			
	<i>Commands</i>	TEpsnet- Command- Mas	První příkaz v poli



Nastavení spojení na sériovém kanálu v režimu UNI pro funkční blok fbEpsnetMas

6.5 Funkční blok *fbEpsnetSlave*

Knihovna : *EpsnetLib*

Funkční blok *fbEpsnetSlave* slouží ke pro komunikaci protokolem EPSNET v pozici podřízené stanice.

Blok implementuje ze souboru služeb protokolu EPSNET následující služby: *CONNECT*, *IDENT*, *SETTID*, *READN*, *WRITEN*, *WANDRN*, *READB*, *WRITEB*.

Komunikace probíhá přes spojení na sériovém kanálu v režimu UNI podle konstanty na vstupu *chanCode*. Spojení musí mít následujícími parametry: 8 bitů, sudá parita, 1 stop bit, délka přijímací a vysílací zóny 256 bytů. Pokud spojení není aktivní nebo nemá správné délky zón, blok indikuje chybu na výstupech *Err* hodnotou TRUE a *ErrId* hodnotou 255.







Vstup *MyAddress* udává adresu v protokolu EPSNET. Tato adresa musí být v síti EPSNET unikátní.

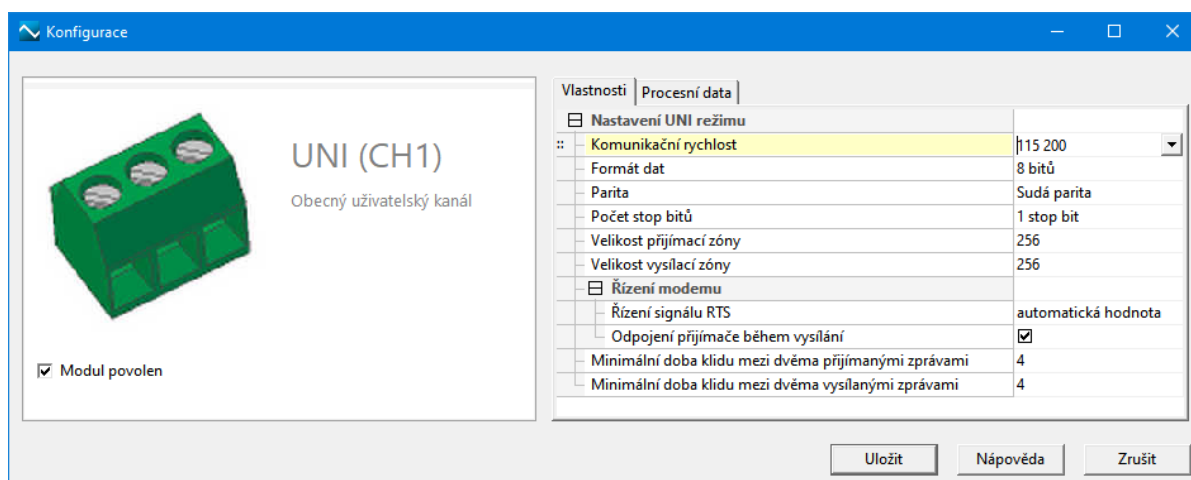
Výstupy *trueMes* a *falseMes* udávají počet dobrých a špatných přijatých zpráv.

Výstup *Error* indikuje, že došlo k chybě.

Významy jednotlivých hodnot výstup *ErrId* jsou uvedeny v tabulce popisu proměnných.

Popis proměnných :

Proměnná	Typ	Význam
VAR_INPUT		
 myAddress	USINT	vlastní adresa
 chanCode	UINT	kód kanálu CH1_uni, CH2_uni,...
VAR_OUTPUT		
 trueMes	UDINT	počet dobrých zpráv
 falseMes	UDINT	počet špatných zpráv
 Error	BOOL	nastala chyba 13 – bloky čtené v komunikační službě READN / READND se nevejdou do zprávy 16 – nulová délka bloku v komunikační službě WRITEN 17 – chyba parametru WRITEB 19 – chybné parametry komunikační služby WANDRN / WANDRND v části zápisu 20 – chybné parametry komunikační služby WANDRN / WANDRND v části čtení 23 – nepodporovaná služba 144 – chybný počáteční znak 152 – chyba kontrolního součtu
 ErrId	USINT	číslo chyby



Nastavení spojení na sériového kanálu v režimu UNI pro funkční blok fbEpsnetSlave

Příklad volání funkčního bloku realizujícího podřízenou stanici s adresou 1 na sériovém kanálu CH1:

```
PROGRAM prgExampleEpsnetSlave
  VAR
    EpsnetSlave : fbEpsnetSlave;
  END_VAR

  EpsnetSlave(myAddress := 1, chanCode := CH1_uni);

END_PROGRAM
```


TXV 003 73.01

Výrobce si vyhrazuje právo na změny dokumentace. Poslední aktuální vydání je k dispozici na internetu www.tecomat.cz