

Knihovna DataBoxLib

TXV 003 56.01
první vydání
prosinec 2010
změny vyhrazeny

Historie změn

Datum	Vydání	Popis změn
Prosinec 2010	1	První vydání, popis odpovídá DataBoxLib_v14

OBSAH

<i>1 Úvod</i>	3
<i>2 Datové typy</i>	5
<i>3 Konstanty</i>	5
<i>4 Globální proměnné</i>	6
<i>5 Funkce</i>	6
5.1 Funkce ReadBlockFromDBx.....	7
5.2 Funkce ReadFromDataBox.....	8
5.3 Funkce SizeOfDataBox.....	10
5.4 Funkce WriteBlockToDBx.....	11
5.5 Funkce WriteToDataBox.....	12
<i>6 Funkční bloky</i>	14
<i>7 Příklad použití</i>	14

1 ÚVOD

Knihovna DataBoxLib je standardně dodávaná jako součást programovacího prostředí Mosaic. Knihovna obsahuje funkce umožňující práci s pamětí databox, která je standardní součástí procesorových jednotek řady C, G a K.

Paměť DataBox je typu CMOS RAM a její obsah je zálohovan baterií. Aplikační program PLC může tuto paměť číst nebo do ní zapisovat prostřednictvím funkcí knihovny DataBoxLib. Procesor PLC systému tuto paměť nijak standardně neošetřuje, pouze zajišťuje zálohování obsahu této paměti během výpadku napájení. Inicializace DataBoxu (např. nulování) je věcí aplikačního programu PLC stejně jako případná kontrola integrity dat. Při zápisu nového programu do PLC se obsah této paměti nemění.

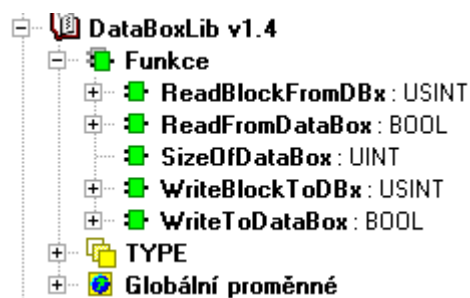
Paměť DataBox je také možno číst nebo přepisovat pomocí služeb READN a WRITEN protokolu EPSNET (komunikační kanál PLC pracující v režimu PC). To znamená, že data v této paměti jsou dostupná z komunikace (sériová linka, USB, Ethernet). V prostředí Mosaic je paměť DataBox přístupná stejným způsobem jako zápisníková paměť, tedy lze ji i ručně editovat

Výhodou této paměti je rychlost přístupu k uloženým datům, neomezený počet zápisů do paměti a její velikost. Paměť DataBox integrální součástí procesorové jednotky, což znamená, že jí nelze ze systému jednoduše odebrat (na rozdíl např. od SD/MMC karty).

Při ukládání dat do této paměti je třeba brát v potaz výdrž zálohovací baterie, která určuje dobu, po kterou může být PLC systém vypnutý bez toho, že by došlo ke ztrátě dat uložených v DataBoxu.

Paměť DataBox je tedy vhodná pro uložení větších objemů dat. To mohou být např. záznamy o průběhu řízení (logy, databáze) nebo naopak receptury, podle kterých se modifikuje řízení technologie. Paměť DataBox může být také využita jako vyrovnávací buffer při ukládání větších objemů dat na SD/MMC kartu, čímž se sníží počet zápisů na kartu.

Následující obrázek ukazuje strukturu knihovny DataBoxLib v prostředí Mosaic



Pokud chceme funkce z knihovny DataBoxLib použít v aplikačním programu PLC, je třeba nejprve přidat tuto knihovnu do projektu. Knihovna je dodávaná jako součást instalace prostředí Mosaic od verze v2.0.0. Simulátor PLC v prostředí Mosaic podporuje simulaci DataBox paměti od verze v2.2.0.0.

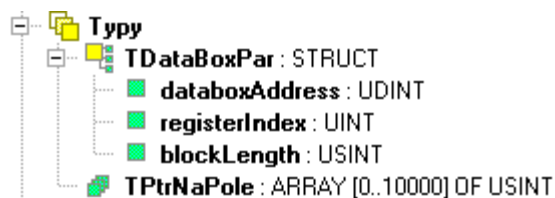
Funkce z knihovny DataBoxLib jsou podporovány v centrálních jednotkách řady K (TC700 CP-7000 a CP-7004, všechny varianty systému Foxtrot) ve všech verzích, dále v jednotkách řady C (TC700 CP-7001, CP-7002 a TC650 CP-6053) a řady G (TC700 CP-7003).

Velikosti paměti DataBox v centrálních jednotkách Tecomat udává následující tabulka.

Velikost paměti DataBox	Typ PLC systému	Typ centrální jednotky	Dostupný adresový prostor
512 KB	Foxtrot	CP-10xx	0 - 16#07FFFF
512 KB	TC700	CP-7000	0 - 16#07FFFF
128 KB	TC700	CP-7001	0 - 16#01FFFF
128 KB nebo volitelně 3 MB	TC700	CP-7002	0 - 16#01FFFF 0 - 16#2FFFFFFF
volitelně 3 MB	TC700	CP-7003	0 - 16#2FFFFFFF
512 KB nebo volitelně 3.5 MB	TC700	CP-7004	0 - 16#07FFFF 0 - 16#37FFFF
2.5 MB	TC700	CP-7005	0 - 16#27FFFF
512 KB nebo volitelně 3.5 MB	TC700	CP-7007	0 - 16#07FFFF 0 - 16#37FFFF

2 DATOVÉ TYPY

V knihovně DataBoxLib jsou definovány následující datové typy:



Datový typ *TDataBoxPar* je struktura využívaná vnitřně v knihovně DataBoxLib. Význam jednotlivých položek je následující:

<i>Identifikátor</i>	<i>Typ</i>	<i>Význam</i>
<i>TDataBoxPar</i>	STRUCT	Parametry operace s DataBoxem
<i>.databoxAddress</i>	UDINT	Adresa paměti DataBox
<i>.registerIndex</i>	UINT	Index registru R (proměnná PLC)
<i>.blockLength</i>	USINT	Velikost přenášeného bloku

Datový typ *TPtrNaPole* je využíván pro určení proměnné v zápisníkové paměti PLC.

<i>Identifikátor</i>	<i>Typ</i>	<i>Význam</i>
<i>TPtrNaPole</i>	ARRAY OF USINT	Zástupný typ pro určení zdroje / cíle v paměti PLC

3 KONSTANTY

V knihovně DataBoxLib nejsou definovány žádné konstanty.

4 GLOBÁLNÍ PROMĚNNÉ

V knihovně DataBoxLib jsou definovány následující globální proměnné.



Konstanta `MAX_LENGTH_DATABOX_BLOCK` udává maximální možnou velikost bloku, který je možná přečíst nebo zapsat jedním zavoláním funkce `ReadFromDataBox()` resp. `WriteToDataBox()`.

5 FUNKCE

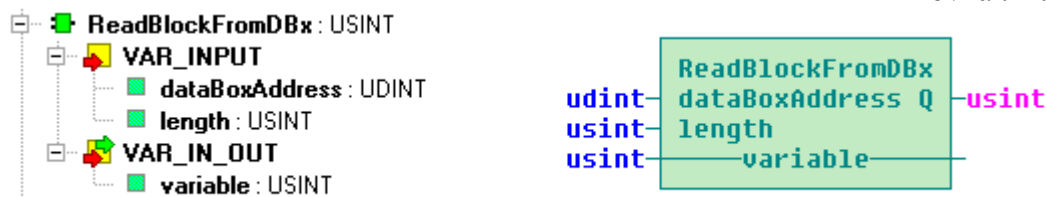
Knihovna DataBoxLib obsahuje následující funkce:



<i>Funkce</i>	<i>Popis</i>
<i>ReadBlockFromDBx</i>	Přenos bloku paměti z DataBoxu do proměnné (max 255 bytů)
<i>ReadFromDataBox</i>	Přenos bloku paměti z DataBoxu do proměnné
<i>SizeOfDataBox</i>	Test aktuální velikost paměti v DataBoxu
<i>WriteBlockToDBx</i>	Přenos bloku paměti z proměnné do DataBoxu (max 255 bytů)
<i>WriteToDataBox</i>	Přenos bloku paměti z proměnné do DataBoxu

5.1 Funkce ReadBlockFromDBx

Knihovna : *DataBoxLib*



Funkce *ReadBlockFromDBx* přenesení blok paměti z DataBoxu do proměnné. Maximální velikost bloku je 255 bytů. Funkce vrací počet přenesených bytů.

Tato funkce je podporovaná na všech centrálních jednotkách uvedených v kap.1 ve všech verzích firmware. Do knihovny DataBoxLib je funkce zařazena od verze DataBoxLib_v10.

Popis proměnných :

	<i>Proměnná</i>	<i>Typ</i>	<i>Význam</i>
VAR_INPUT			
	<i>dataBoxAddress</i>	UDINT	Počáteční adresa v paměti DataBox, ze které se bude číst
	<i>length</i>	USINT	Velikost přenášeného bloku (počet bytů, max 255)
VAR_IN_OUT			
	<i>variable</i>	USINT	Proměnná, do které budou uložena data přečtená z DataBoxu
ReadBlockFromDBx			
	<i>Návratová hodnota</i>	USINT	Počet skutečně přenesených bytů

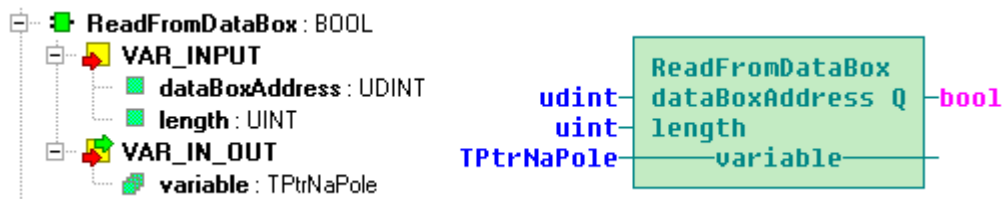
Příklad programu s voláním funkce *ReadBlockFromDBx* :

```
PROGRAM exampleReadFromDBx
VAR
  data : ARRAY[0..99] OF USINT;
  size : USINT;
  result : BOOL;
END_VAR

size := ReadBlockFromDBx( dataBoxAddress := 16#100,
                          length := sizeof(data),
                          variable := void(data));
result := size = sizeof(data);
END_PROGRAM
```

Viz také Funkce WriteBlockToDBx

5.2 Funkce ReadFromDataBox

Knihovna : *DataBoxLib*

Funkce *ReadFromDataBox* načte data z paměti DataBox do proměnné, jejíž jméno je uvedeno v parametru *variable*. Velikost načítaných dat musí odpovídat velikosti proměnné. Maximální velikost načítaného bloku je 10000 bytů. Funkce vrací TRUE, pokud byla všechna požadovaná data přenesena.

Upozornění !!!

- proměnná musí být umístěna v paměťovém prostoru %R resp. %M
- proměnná nesmí být typu BOOL nebo ARRAY[] OF BOOL

Tato funkce je podporovaná na všech centrálních jednotkách uvedených v kap.1 ve všech verzích firmware. Do knihovny DataBoxLib je funkce zařazena od verze DataBoxLib_v10.

Popis proměnných :

	<i>Proměnná</i>	<i>Typ</i>	<i>Význam</i>
VAR_INPUT			
	<i>dataBoxAddress</i>	UDINT	Počáteční adresa v paměti DataBox, ze které se bude číst
	<i>length</i>	UINT	Počet načítaných bytů
VAR_IN_OUT			
	<i>variable</i>	USINT	Proměnná, do které budou uložena data přečtená z DataBoxu
ReadFromDataBox			
	<i>Návratová hodnota</i>	BOOL	TRUE pokud se podaří načíst požadovaná data, jinak FALSE

Příklad programu s voláním funkce *ReadFromDataBox* :

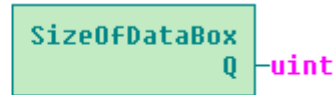
```
PROGRAM exampleReadFromDataBox
VAR
  record : STRING;
  result : BOOL;
END_VAR

IF SizeOfDataBox() <> 0 THEN
  result := ReadFromDataBox ( dataBoxAddress := 16#100,
                             length := sizeof(record),
                             variable := void(record));
ELSE
  result := FALSE;
END_IF;
END_PROGRAM
```

Viz také Funkce WriteToDataBox, Funkce SizeOfDataBox

5.3 Funkce SizeOfDataBox


Knihovna : *DataBoxLib*
 **SizeOfDataBox** : UINT



Funkce *SizeOfDataBox* vrací velikost DataBoxu v kilobytech.

Tato funkce je podporovaná na všech centrálních jednotkách uvedených v kap.1 ve všech verzích firmware. Do knihovny DataBoxLib je funkce zařazena od verze DataBoxLib_v10.

Popis proměnných :

	<i>Proměnná</i>	<i>Typ</i>	<i>Význam</i>
<i>SizeOfDataBox</i>			
	<i>Návratová hodnota</i>	UINT	Velikost paměti DataBox v kilobytech

Příklad programu s voláním funkce *SizeOfDataBox* :

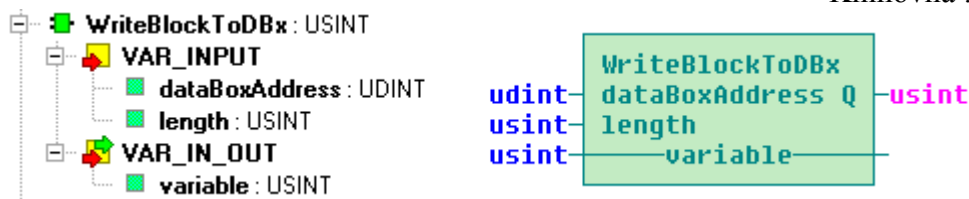
```
PROGRAM exampleReadFromDataBox
VAR
  record : STRING;
  result : BOOL;
END_VAR

IF SizeOfDataBox() <> 0 THEN
  result := ReadFromDataBox ( dataBoxAddress := 16#100,
                             length := sizeof(record),
                             variable := void(record));
ELSE
  result := FALSE;
END_IF;
END_PROGRAM
```

Viz také Funkce ReadFromDataBox, Funkce WriteToDataBox

5.4 Funkce WriteBlockToDBx

Knihovna : *DataBoxLib*



Funkce *WriteBlockToDBx* přenesení blok paměti z proměnné do DataBoxu. Maximální velikost bloku je 255 bytů. Funkce vrací počet přenesených bytů.

Tato funkce je podporovaná na všech centrálních jednotkách uvedených v kap.1 ve všech verzích firmware. Do knihovny DataBoxLib je funkce zařazena od verze DataBoxLib_v10.

Popis proměnných :

	Proměnná	Typ	Význam
VAR_INPUT			
	<i>dataBoxAddress</i>	UDINT	Počáteční adresa v paměti DataBox, do které se bude zapisovat
	<i>length</i>	USINT	Počet zapisovaných bytů (max 255 bytů)
VAR_IN_OUT			
	<i>variable</i>	USINT	Proměnná, která obsahuje data, jež budou zapsána do DataBoxu
WriteBlockToDBx			
	<i>Návratová hodnota</i>	USINT	Počet skutečně zapsaných bytů

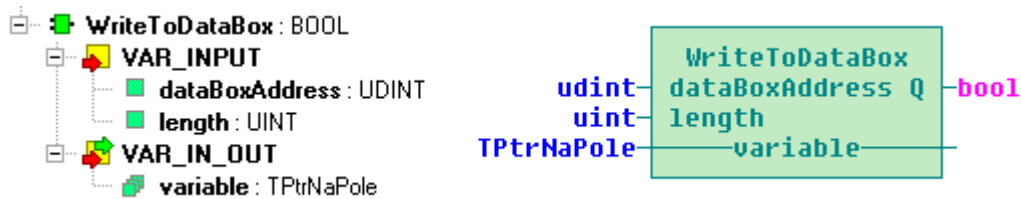
Příklad programu s voláním funkce *WriteBlockToDBx* :

```
PROGRAM exampleWriteToDBx
VAR
  data   : ARRAY[0..99] OF USINT;
  i      : USINT;
  size   : USINT;
  result : BOOL;
END_VAR

FOR i := 0 TO 99 DO data[i] := i; END_FOR;
size := WriteBlockToDBx( dataBoxAddress := 16#100,
                        length := sizeof(data),
                        variable := void(data));
result := size = sizeof(data);
END_PROGRAM
```

Viz také Funkce ReadBlockFromDBx

5.5 Funkce WriteToDataBox

Knihovna : *DataBoxLib*

Funkce *WriteToDataBox* přenesení blok paměti z proměnné do DataBoxu. Maximální velikost bloku je 10000 bytů. Funkce vrací TRUE, pokud byla všechna požadovaná data přenesena.

Upozornění !!!

- proměnná musí být umístěna v paměťovém prostoru %R resp. %M
- proměnná nesmí být typu BOOL nebo ARRAY[] OF BOOL

Tato funkce je podporovaná na všech centrálních jednotkách uvedených v kap.1 ve všech verzích firmware. Do knihovny DataBoxLib je funkce zařazena od verze DataBoxLib_v10.

Popis proměnných :

	<i>Proměnná</i>	<i>Typ</i>	<i>Význam</i>
VAR_INPUT			
	<i>dataBoxAddress</i>	UDINT	Počáteční adresa v paměti DataBox, do které se bude zapisovat
	<i>length</i>	USINT	Počet zapisovaných bytů
VAR_IN_OUT			
	<i>variable</i>	USINT	Proměnná, která obsahuje data, jež budou zapsána do DataBoxu
WriteToDataBox			
	<i>Návratová hodnota</i>	BOOL	TRUE, pokud byla všechna požadovaná data přenesena

Příklad programu s voláním funkce *WriteToDataBox* :

```
PROGRAM exampleWriteToDataBox
VAR
  record : STRING := 'This record will be saved to DataBox';
  result : BOOL;
END_VAR

IF SizeOfDataBox() <> 0 THEN
  result := WriteToDataBox( dataBoxAddress := 16#100,
                           length := sizeof(record),
                           variable := void(record));
ELSE
  result := FALSE;
END_IF;
END_PROGRAM
```

Viz také Funkce ReadFromDataBox, Funkce SizeOfDataBox

6 *FUNKČNÍ BLOKY*

V knihovně DataBoxLib nejsou definovány žádné funkční bloky.

7 *PŘÍKLAD POUŽITÍ*