

**ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE**

Fakulta elektrotechnická

Katedra řídicí techniky



Bakalářská práce

**Monitorování provozu mostového jeřábu**

Praha, 2010

*Autor:* David Šteffl

*Vedoucí:* Ing. Petr Havel, Ph.D.



České vysoké učení technické v Praze  
Fakulta elektrotechnická

Katedra řídicí techniky

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **David Štefl**

Studijní program: Elektrotechnika a informatika (bakalářský), strukturovaný  
Obor: Kybernetika a měření

Název tématu: **Monitorování provozu mostového jeřábu**

Pokyny pro vypracování:

1. Seznamte se s programovatelnými automaty řady Siemens Simatic a programovacím prostředím STEP 7
2. Navrhněte a implementujte aplikaci, která monitoruje, zpracovává a ukládá potřebná data o provozu mostového jeřábu
3. Aplikace bude umožňovat dávkový způsob přenosu dat do PC a jejich vizualizaci v prostředí MS Excel

Předpokládá se spolupráce s firmou SIMO-CZ s.r.o. a případné nasazení výsledků práce na skutečných provozech instalovaných mostových jeřábů.

Seznam odborné literatury:

Dodá vedoucí práce

Vedoucí: Ing. Petr Havel, Ph.D.

Platnost zadání: do konce zimního semestru 2010/2011

prof. Ing. Michael Šebek, DrSc.  
vedoucí katedry



doc. Ing. Boris Šimák, CSc.  
děkan

V Praze dne 26. 11. 2009



## **Prohlášení**

Prohlašuji, že jsem svou bakalářskou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v přiloženém seznamu.

V Praze dne .....  
.....  
podpis

## **Poděkování**

Rád bych poděkoval svému vedoucímu Ing. Petrovi Havlovi, Ph.D. a také Ing. Martinovi Samkovi za jejich čas a cenné rady při zpracování bakalářské práce.

## **Anotace**

Bakalářská práce se zaměřuje na monitorování provozu mostového jeřábu. Monitorování zajišťuje funkční blok vyvinutý v programátorském prostředí Step 7 od firmy Siemens. Cílem práce je nalezení vhodného způsobu přenosu provozních dat uložených v PLC Simatic S7 do počítače a implementace tohoto způsobu v rámci uživatelské aplikace vytvořené v MS Excel. Zvolený způsob musí umožnit obousměrný přenos dat mezi programovatelnou řídicí jednotkou PLC a počítačem. Aplikace slouží k vizualizaci a archivaci dat z PLC. Součástí práce je přehled a popis různých možných způsobů přenosu dat. Práce vychází z požadavků firmy SIMO-CZ s.r.o.

## **Annotation**

This Bachelor thesis is focused on bridge crane operation monitoring. The monitoring is maintained by a function block designed in the programming environment Step 7 from the Siemens company. The goal of this thesis is to find a suitable transfer method of data stored in PLC Simatic S7 to computer and to implement an user application in MS Excel by using this method. The transfer method has to enable bidirectional data transmission between the programmable logical controller and the computer. The application is designed to visualize and archive the data from PLC. Another part of this thesis is an overview and description of possible transfer methods. The thesis is based on the requierements of the company SIMO-CZ s.r.o.

# **Obsah**

<b>1</b>	<b>Úvod.....</b>	10
1.1	Motivace.....	10
1.2	Konfigurace PLC.....	10
1.3	Hlavní cíle .....	10
1.4	Postup řešení.....	11
<b>2</b>	<b>Možnosti přenosu dat.....</b>	12
2.1	Bezplatné možnosti .....	12
2.1.1	Generovaný soubor AWL .....	12
2.1.2	Funkční bloky AG_SEND a AG_RECV .....	13
2.1.3	Knihovna libnodave.dll.....	14
2.1.4	Open Communication Wizard (OC Wizard) .....	15
2.1.5	S7Backup .....	16
2.2	Placené možnosti .....	17
2.2.1	INDI.AN PLC COM.....	17
2.2.2	ACCON S7 EasyLog .....	18
2.2.3	AGLink S7 komunikační .DLL .....	20
2.2.4	ACCON PG S7 Power Tools.....	21
2.2.5	I/O server eWON.....	22
2.2.6	S7 DB TAGS.....	23
2.2.7	PC-MPI-S7-LINK .....	24
2.2.8	IP-S7-LINK.....	26
2.3	Výsledné doporučení.....	27
2.4	Otestování doporučených možností přenosu dat.....	28
2.4.1	Knihovna libnodave.dll.....	28
2.4.2	Generovaný soubor AWL .....	28
2.4.3	IP-S7-LINK.....	31
2.4.4	AGLink S7 komunikační .DLL .....	31
<b>3</b>	<b>Knihovna libnodave.dll detailně.....</b>	32
3.1	Úvod a licenční politika .....	32
3.2	Podporované programovací jazyky a způsoby připojení .....	33
3.3	Práce s knihovnou .....	33
3.3.1	Navázání spojení s PLC .....	33
3.3.2	Zrušení spojení s PLC.....	36
3.3.3	Čtení dat z PLC.....	37
3.3.4	Zápis dat do PLC .....	39
3.3.5	Další možnosti .....	39

<b>4 Funkční a datový blok.....</b>	<b>41</b>
4.1    Funkční blok .....	41
4.1.1    Obecný popis .....	41
4.1.2    Monitorovací funkční blok.....	43
4.2    Datový blok .....	45
4.2.1    Obecný popis .....	45
4.2.2    Monitorovací datový blok.....	46
<b>5 Vlastní aplikace v MS Excel .....</b>	<b>47</b>
5.1    Pracovní list Nastavení.....	48
5.2    Pracovní list Přenos.....	49
5.2.1    Práce s PLC.....	49
5.2.2    Práce s daty v Excelu.....	50
<b>6 Závěr.....</b>	<b>52</b>
<b>7 Literatura .....</b>	<b>53</b>

# 1 Úvod

Cílem této bakalářské práce je implementovat v PLC samostatný funkční blok naprogramovaný v prostředí Step 7 od firmy Siemens AG, který ukládá důležitá provozní data o provozu mostového jeřábu do vlastního datového bloku. Dále je cílem vytvořit uživatelsky přívětivou a na ovládání nenáročnou aplikaci v prostředí Microsoft Excel, která umožňuje obousměrný dávkový přenos dat z řídicí programovatelné jednotky PLC do osobního počítače či notebooku a jejich následnou vizualizaci. Práce vznikla ve spolupráci s firmou SIMO-CZ s.r.o. a vychází z jejího požadavku na monitorování provozu mostového jeřábu a následný přenos a archivaci dat v prostředí Microsoft Excel.

## 1.1 Motivace

Požadavek na monitorování a zálohování dat spojených s provozem mostového jeřábu vychází z českých technických norem pro sledování stavu jeřábů ČSN ISO 12482-1 z roku 1997 [1]. Hlavním cílem normy je dodržení podmínek bezpečnosti provozování jeřábů a definovaní postupů vedoucích k zajištění jejich dodržení i v dlouhodobém časovém horizontu používání jeřábů. Součástí této normy jsou povinnosti uživatele/vlastníka vést záznamy o používání, inspekčích a údržbě jeřábů stejně tak jako včasné provedení zvláštního posouzení, které musí obsahovat normou definovaná kritéria.

## 1.2 Konfigurace PLC

Monitorování provozu mostového jeřábu obsluhuje řídicí systém SIMATIC S7-300 z kategorie standardních CPU firmy Siemens AG. Všechny jednotky této kategorie disponují programovacím a komunikačním rozhraním MPI. V našem případě se jedná konkrétně o typ řídicího systému 315-2DP se zabudovaným rozhraním PROFIBUS. Na základě požadavků zadávající firmy SIMO-CZ s.r.o. dále vycházíme z možnosti nasazení komunikačního procesoru CP 343-1 rozšiřujícího možnosti komunikace s řídicím systém o průmyslový ethernet.

## 1.3 Hlavní cíle

Jedním z hlavních cílů této práce je vytvořit funkční blok FB ve Step 7, který bude zpracovávat data ze senzorů měřících provozní veličiny mostového jeřábu. Data budou zpracovávána s ohledem na následnou archivaci v rámci požadavků zmíněné normy

ČSN ISO 12482-1. Datové úložiště představuje datový blok DB obsahující všechna vstupní a statická data.

Další hlavní cíl představuje způsob komunikace mezi PLC a PC. Je potřeba nalézt a zvolit vhodnou metodu umožňující obousměrně přenášet data obsažená v datovém bloku. Od této metody se očekává bezproblémové navázání spojení PC s PLC, možnost přistoupit k datům DB a ty uložit v rámci aplikace MS Excel. Aplikace by měla uživateli umožnit uložená data modifikovat a zpětně je nahrát do PLC. Uživatel by měl mít možnost zobrazení grafické závislosti dat a dále by měla aplikace umožnit archivaci výsledného zpracování dat.

## 1.4 Postup řešení

První krok k úspěšnému vyřešení požadavků firmy SIMO-CZ s.r.o. je důkladné zmapování všech dostupných možností přenosu dat mezi PLC a PC. Všechny nalezené možnosti musí vyhovovat požadavkům na obousměrný přenos a možnost archivace dat. Popisem všech takovýchto možností se zabývá následující kapitola 2 *Možnosti přenosu dat*.

Kapitola 3 *Knihovna libnodave.dll detailně* navazuje na výsledky mapování všech dostupných možností přenosu dat. Zde detailně popisují jednu vybranou možnost a způsob jejího uplatnění ve vyvíjené aplikaci.

Realizací monitorovacího funkčního bloku v prostředí Step 7 se zabývá kapitola 4 *Funkční a datový blok*. V této kapitole popisuji zpracování vstupních dat ze senzorů a jejich úpravu do výsledné podoby dle požadavků firmy SIMO-CZ s.r.o.

Podrobný popis implementované aplikace v MS Excel se nachází v kapitole 5 *Vlastní Aplikace v MS Excel*. V této kapitole prezentuji krok za krokem úkony, které jsou potřeba provést před samotnou realizací spojení PC s PLC. Dále zde uvádím veškeré funkce poskytované koncovému uživateli.

## 2 Možnosti přenosu dat

V této kapitole se nachází přehled a popis způsobů přenosu dat mezi PLC a PC. Přehled je výsledkem mapování všech možných způsobů, které by mohli vyhovovat požadavkům firmy SIMO-CZ s.r.o. na obousměrný přenos dat. První část tvoří bezplatné možnosti. Hlavní výhodou je jejich nezávislé použití na licenční politice. Nevýhodou je nutná vlastní implementace aplikace řešící konkrétní problém. Druhou část tvoří způsoby nabízené firmami zabývajícími se touto problematikou. U každé možnosti uvádím způsob připojení a její základní nástin postupu realizace přenosu. V závěru kapitoly uvádím výsledné doporučení a výsledky testování doporučených způsobů s ohledem na požadavky firmy SIMO-CZ s.r.o. Hlavní zohledněné kritérium je možnost realizace spojení mezi PC a PLC přes sběrnici MPI a průmyslový Ethernet a dále možný vývoj aplikace za použití prostředků Microsoft Excel.

### 2.1 Bezplatné možnosti

#### 2.1.1 Generovaný soubor AWL

**Způsob připojení:**

MPI, Ethernet

**Popis:**

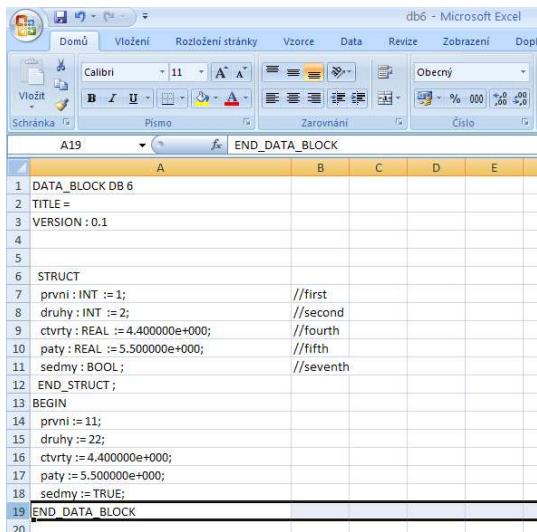
V Simatic Manageru přistoupíme přímo k požadovanému datovému bloku. Vybraný datový blok uložíme a vygenerujeme jeho zdrojový kód přes „File – Generate Source..“. Vygenerovaný zdrojový kód datového bloku exportujeme na libovolné úložiště jako soubor typu „AWL“ [2]. Tento typ souboru je podporován Excelem. Po otevření vidíme strukturu jako je uvedená na Obr. 1. Pomocí VBA (Visual Basic for Application), který je součástí Excelu, upravíme data získaná exportem do požadované podoby pro vizualizaci.

Import je opět záležitostí pár kliknutí. Pro úspěšný import musíme dodržet formátování, které získáme exportem datového bloku. Toho lze docílit pomocí VBA. Hodnoty lze libovolně nastavovat. V Simatic Manageru vybereme možnost importování zdrojového kódu s příponou AWL, ten pak zkompilujeme a tím se obnoví námi vybraný datový blok.

**Nástin provedení:**

- vygenerujeme zdrojový kód vybraného datového bloku ve Stepu7
- exportujeme zdrojový kód jako soubor .AWL
- otevřeme zdrojový kód v Excelu
- provedeme požadované úpravy hodnot DB a opět uložíme jako soubor .AWL

- importujeme upravený zdrojový kód jako soubor .AWL
- ve Stepu 7 komplujeme importovaný zdrojový kód
- po úspěšném zkompilování se přepíší staré hodnoty DB novými



The screenshot shows a Microsoft Excel spreadsheet titled "db6 - Microsoft Excel". The table has columns A through E. Row 1 contains the header "END\_DATA\_BLOCK". Rows 2 through 19 contain the AWL code for defining a data block:

```

1 DATA_BLOCK DB 6
2 TITLE =
3 VERSION : 0.1
4
5
6 STRUCT
7 prvni : INT := 1; //first
8 druhý : INT := 2; //second
9 ctvrty : REAL := 4.400000e+000; //fourth
10 paty : REAL := 5.500000e+000; //fifth
11 sedmy : BOOL; //seventh
12 END_STRUCT;
13 BEGIN
14 prvni := 11;
15 druhý := 22;
16 ctvrty := 4.400000e+000;
17 paty := 5.500000e+000;
18 sedmy := TRUE;
19 END DATA_BLOCK

```

Obr. 1 Výsledek exportovaného datového bloku jako .AWL

### **2.1.2 Funkční bloky AG SEND a AG RECV**

#### **Způsob připojení:**

Ethernet

#### **Popis:**

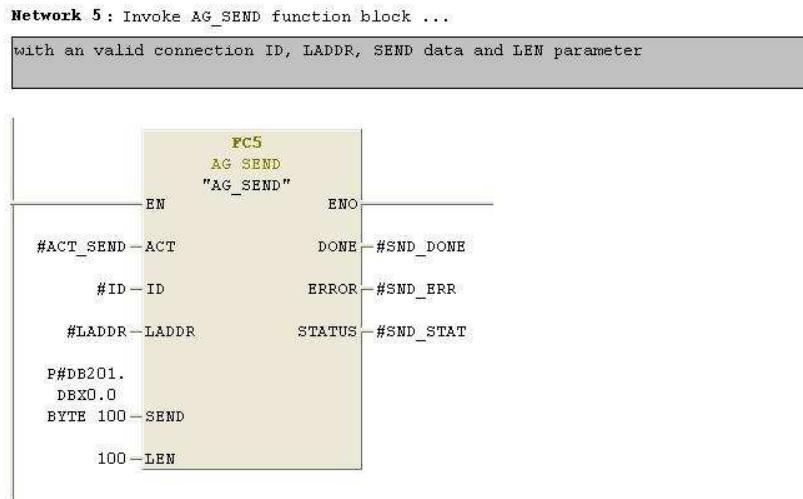
Použijeme funkční bloky AG\_SEND a AG\_RECV ve Stepu 7 (Obr. 2 a 3). Bloky lze uživatelsky například nastavit tak, aby po připojení klienta, v našem případě technik s notebookem, došlo k dávkovému přenosu předem definovaného rámce dat [3]. Tato data by byla reprezentována polem bytů. Přenos by byl realizován na základě ISO modelu v protokolu TCP, čímž by bylo zaručeno pořadí přenášených dat. Uživatel by na straně PC přijal data v podobě socket streamu. Socket bychom přijali Excellem nebo jiným programovacím prostředím k dalšímu zpracování. Na základě znalosti rozložení dat v datovém bloku bychom zrekonstruovali data z přijatého pole bytů.

Import dat do PLC by byl obdobný. Námi reprezentovaná data v některém z programátorských prostředí bychom převedli na pole bytů reprezentujících daný datový blok a v rámci soketu s definovanou adresou poslali přes ethernet do PLC.

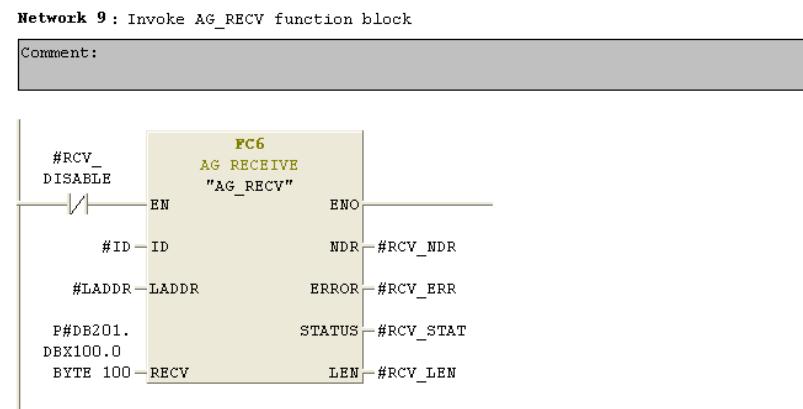
#### **Nástin provedení:**

- technik se připojí k PLC, tím inicializuje blok AG\_SEND a ten odešle dávkově DB
- v Excelu nebo jiném programu se zrekonstruuje struktura přijatého DB
- provedeme požadované úpravy hodnot DB

- v Excelu nebo jiném programu nakonfigurujeme ethernetové spojení
- pomocí bloku AG\_RECV přjmeme dávkový přenos DB zpět do PLC



Obr. 2 Konfigurace bloku AG\_SEND ve Stepu 7



Obr. 3 Konfigurace bloku AG\_RECV ve Stepu 7

### 2.1.3 Knihovna libnodave.dll

**Způsob připojení:**

MPI, Ethernet

**Popis:**

Použitím této bezplatné komunikační knihovny získáme k dispozici její veškeré funkce. Ty nám umožní navázat komunikaci s PLC v rámci programování maker v Excelu. Dále pomocí funkcí této knihovny budeme schopni přistoupit k požadovaným datům v PLC a ty dále zpracovat v Excelu. Funkce knihovny umožňují i zápis dat do PLC. Knihovna podporuje komunikaci s PLC přes MPI sběrnici nebo protokol TCP/IP v ethernetové síti (Obr. 4).

Tato knihovna vznikla jako reakce na licencovanou siemenovskou knihovnu prodave.dll. Libnodave podporuje i několik dalších programovacích jazyků [4].

#### Nástin provedení:

- technik v Excelu vyplní parametry požadované k navázání komunikace s PLC přes MPI nebo Ethernet
- po stisknutí tlačítka dojde k exportu DB
- provedeme požadované úpravy hodnot DB
- po stisknutí tlačítka dojde k importu DB do PLC

	A	B	C	D
1				
2				
3	Port:	102		<input type="button" value="START"/>
4	IP:	172.16.28.230		
5	Rack:	0		<input type="button" value="lesen"/>
6	Slot:	2		
7	I-DB:	1		<input type="button" value="DB löschen"/>
8	MD:	10		
9	Raster (ms):	100		
10	Dauer (ms):	20000		

Obr. 4 Aplikace využívající funkcí knihovny libnodave

#### 2.1.4 Open Communication Wizard (OC Wizard)

##### Způsob připojení:

Ethernet

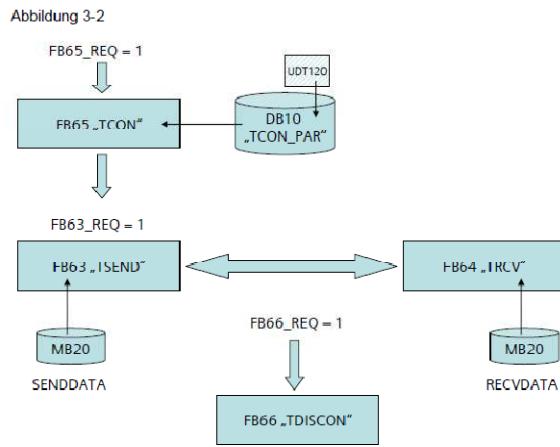
##### Popis:

Všechny nové Simatic S7-CPU obsahují integrované rozhraní Profinetu, které umožňuje TCP/IP komunikaci. Ta je možná pouze při použití funkčních bloků určených pro otevřenou komunikaci. Použijeme tedy funkční bloky FB63 TSEND a FB64 TRCV (Obr. 5). Abychom zajistili správnou funkčnost komunikace, musíme nastavit specifické parametry těchto bloků. Pro snadné a jasné nastavení potřebných specifikací přenosu slouží freeware nástroj OC Wizard (Obr. 6). Nástroj je po instalaci automaticky integrován do prostředí Simatic Manageru [5]. S jeho pomocí můžeme exportovat uživatelsky definovaný blok UDB nebo DB se všemi parametry uvedenými v nastavení komunikace. Součástí instalace OC Wizardu je i podrobný návod, jak správně nastavit požadovanou komunikaci. TSEND a TRCV fungují obdobně jako AG\_SEND a AG\_RECV viz 1.1.2 Funkční bloky AG\_SEND a AG\_RECV.

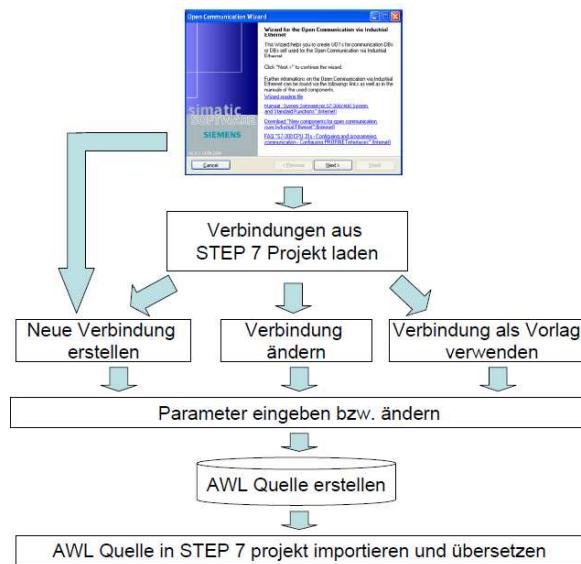
##### Nástin provedení:

- technik specifikuje pomocí OC Wizardu druh komunikace, který budou FB TSEND a TRCV používat
- technik se připojí k PLC, tím inicializuje blok TSEND a ten odešle dávkově DB
- v Excelu nebo jiném programu se zrekonstruuje struktura přijatého DB

- provedeme požadované úpravy hodnot DB
- v Excelu nebo jiném programu nakonfigurujeme ethernetové spojení
- pomocí bloku TRCV přjmeme dávkový přenos DB zpět do PLC



Obr. 5 Schéma průběhu přenosu při použití OC Wizardu



Obr. 6 Schéma funkcionality OC Wizardu

## 2.1.5 S7Backup

**Způsob připojení:**

Ethernet

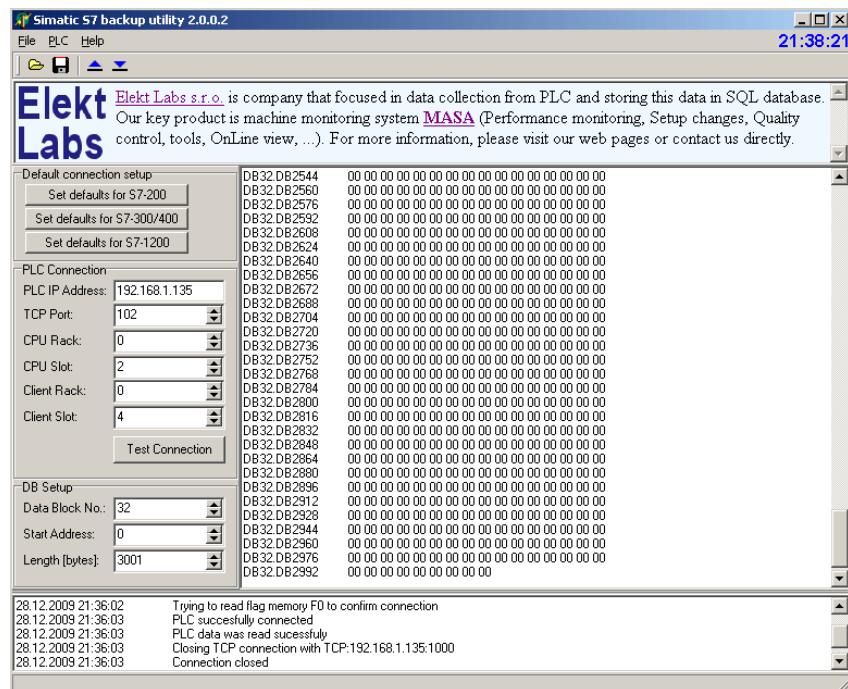
**Popis:**

Tato aplikace umožňuje zálohovat data z datového bloku přes ethernet pro PLC S7-200, 300, 400 a 1200. Aplikace může být využita pro snadné zálohování dat do

souboru a pro pozdější zpětné obnovení těchto dat do Step7 [6]. S7Backup využívá výše zmíněné freeware knihovny libnodave.dll. Abychom mohli S7Backup použít pro naše účely, museli bychom pracovat se zálohovanými daty datového bloku v Excelu. Upravené hodnoty opět přidat do S7Backup a s jeho pomocí je obnovit ve Step7.

### Nástin provedení:

- program od ElektLabs
- technik zadá parametry k navázání spojení s PLC v programu S7Backup
- exportovaný datový blok se zobrazí v bytové podobě, viz Obr. 7
- zálohovaná data z S7Backup převedeme manuálně do Excelu a zpracujeme
- provedeme požadované úpravy hodnot DB
- data v Excelu převedeme do formátu S7Backup
- pomocí programu S7Backup zapíšeme upravený zálohovaný DB do PLC



Obr. 7 Aplikace S7Backup

## 2.2 Placené možnosti

### 2.2.1 INDIAN PLC COM

#### Způsob připojení:

MPI, Ethernet

### **Popis:**

PLCCom S7 je knihovna s vysoce optimalizovanými komponentami pro vývojáře softwaru. Poskytuje možnosti čtení a zápisu dat z a do S7 (200s, 300s, 400s). Knihovna je vytvořena na základě .NET Framework a nebo Javy [7].

Možnosti komunikace v závislosti na CPU:

Sériový MPI adaptér, sériový PPI adaptér, MPI adaptér, USB (přes virtuální sériový port), Ethernet TCP/IP (přes CP243, CP343, CP443)

Funkčnost:

- Čtení a zápis operací následujících datových typů: Bool, Bit, Byte, Word, DWord, Float, Unicode Char, INT, DINT, 16Bit BCD
- Čtení datumu a času
- Čtení vstupů a výstupů, datových bloků, časovačů, čítačů

### **Nástin provedení:**

- program od Indi.an
- technik v Excelu vyplní parametry požadované k navázání komunikace s PLC přes MPI nebo Ethernet
- po stisknutí tlačítka dojde k importu DB
- provedeme požadované úpravy hodnot DB
- po stisknutí tlačítka dojde k exportu DB do PLC

**Cena:**

PLCCom S7 100% Pure .Net	-	399,00 € (10 300,- Kč)
PLCCom S7 100% Pure Java	-	399,00 € (10 300,- Kč)
PLCCom S7 Suite (.NET+Java)	-	599,00 € (15 500,- Kč)

## **2.2.2 ACCON S7 EasyLog**

**Způsob připojení:**

MPI, Ethernet

### **Popis:**

Program ACCON S7 EasyLog je software pro monitorování a archivaci procesních dat z PLC Siemens Simatic S7-200/-300/-400 do souborů formátu .CSV (Obr. 8). EasyLog je vhodná aplikace umožňující jednoduchý zápis dat z PLC Siemens Simatic S7 do CSV souboru čitelného v MS Excel (Obr. 9). Soubory formátu MS Excel může EasyLog archivovat každý den. Soubory jsou uloženy do předem určeného adresáře a pojmenovány dle data vytvoření [8].

## Funkčnost:

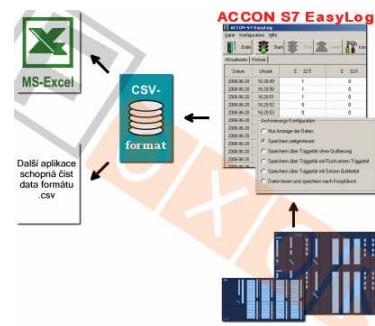
- čtení vstupů, výstupů, merkerů, časovačů, čítačů
- zápis proměnných do souboru formátu .XLS - možno dále zpracovat v MS Excel
- nevyžaduje žádné programování PLC
- podpora komunikace s PLC Siemens Simatic S7-300 / S7-400 přes adaptéry:
  - ACCON NetLink PRO (Ethernet, PROFIBUS) - také PLC S7-200
  - eWON 500MPI (Ethernet)
  - ACCON MPI-USB (USB)
  - ACCON MPI-RS232 (Sériový port)
  - PC/CP nastavení STEP 7

## Nástin provedení:

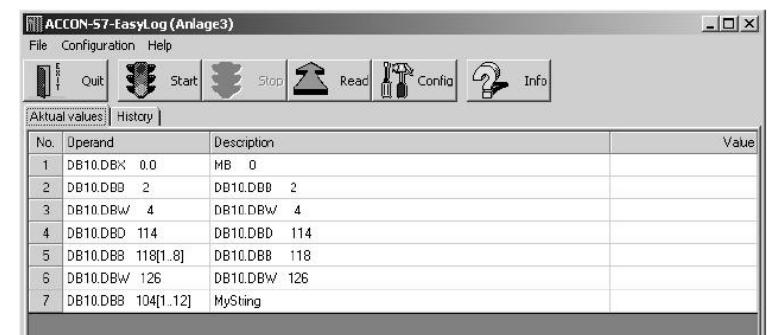
- program od Deltalogic Software
- technik nastaví archivaci DB v ACCON S7 EasyLog
- vygeneruje z ACCON S7 EasyLog CSV soubor s daty DB
- v Excelu provedeme požadované úpravy hodnot DB
- program nepodporuje import DB zpět do PLC – nutno zvolit jiný způsob importu

## Cena:

ACCON S7 EasyLog - 13 900,- Kč s DPH



Obr. 8 Schéma ACCON S7 EasyLog



Obr. 9 Aplikace ACCON S7 EasyLog

### **2.2.3 AGLink S7 komunikační .DLL**

**Způsob připojení:**

MPI, Ethernet

**Popis:**

AGLink S7 komunikační knihovna je založena na funkcích DLL knihoven, které umožňují programátorovi bez znalosti různých protokolů vytvořit program komunikující s PLC S5 a S7 přes sériové rozhraní počítače MPI/PROFIBUS nebo Ethernet [9].

K procesním datům můžete přistupovat bez nutnosti úprav programu v PLC. Programátor PC nepotřebuje žádné detailní znalosti o probíhajícím PLC programu. Jediné co musí znát, je, na kterém místě jsou data, která mají být aplikací zpracovávána (Obr. 10).

Funkčnost:

- Čtení AS-Info (MLFB, výstupní soubor, firmware)
- Diagnóza bufferu
- Čtení provozního stavu (RUN/STOP)
- Čtení a zápis datového, vstupního a výstupního bytu
- Čtení a zápis merkerů, časovačů, čítačů
- Konvertovací funkce
- Chybová hlášení
- Čtení a zápis více operandů po blocích

Podporovaná programátorská prostředí:

- Borland Delphi
- Borland C++-Builder
- Visual C/C++
- Visual Basic
- C# (C-Sharp)
- Visual Basic .Net
- LabVIEW

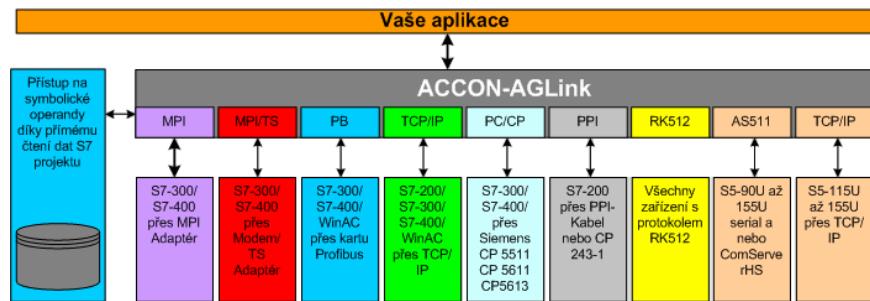
**Nástin provedení:**

- program od Deltalogic Software
- technik ve vlastní aplikaci v Excelu vyplní parametry požadované k navázání komunikace s PLC přes MPI nebo Ethernet
- po stisknutí tlačítka dojde k importu DB
- provedeme požadované úpravy hodnot DB
- po stisknutí tlačítka dojde k exportu DB do PLC

## Cena:

AGLink S7 komunikační .DLL

- od 50 000,- Kč s DPH (v závislosti na komunikaci)



Obr. 10 Schéma možností přístupu k datům pomocí AGLink S7 komunikační .DLL

### 2.2.4 ACCON PG S7 Power Tools

#### Způsob připojení:

MPI, Ethernet

#### Popis:

Program S7 Power Tools je komplet čtyř softwarových modulů, které umožňují práci s daty, programy a projekty Siemens Simatic PLC S7-300 a S7-400 (Obr.11) [10]. Moduly obsahují nástroje na zabezpečení, parametrizování, diagnostiku nebo programování dat PLC S7. Všechny moduly programu jsou v německém jazyce. Projekty S7, které byly vytvořeny programem Step 7, můžete otevírat, upravovat, nahrávat a ukládat přímo v programu ACCON PG S7.

#### Moduly ACCON PG S7:

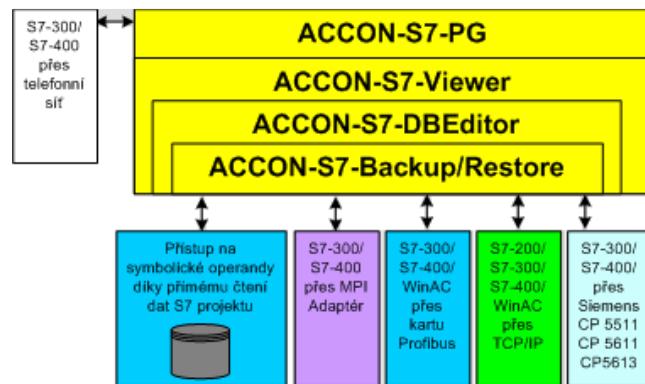
- S7 Backup/Restore – dokáže zálohovat celý program z PLC do S7-projektu a naopak
- S7 DB Editor – pro zobrazení a editaci DB přímo v datových modulech projektu S7
- S7 AWL Viewer – umožňuje zobrazit všechny moduly včetně komentářů, symbolů a proměnných
- S7 PG AWL Editor – podporuje integrované AWL-příkazy Stepu 7 pro nepřímé adresování a přístup k polím, komentářům a symbolům

#### Nástin provedení:

- program od Deltalogic Software
- technik pomocí ACCON PG S7 Power Tools importuje buď DB, nebo celý program
- provedeme požadované úpravy v jednom z nabízených modulů přímo
- upravený DB importujeme do PLC

**Cena:**

ACCON PG S7 Power Tools - 13 400,- Kč s DPH



Obr. 11 Schéma možností přístupu k datům v rámci ACCON PG S7 modulů

## 2.2.5 I/O server eWON

**Způsob připojení:**

MPI, Ethernet

**Popis:**

Produkty eWon v sobě zahrnují modem a bránu pro MPI nebo průmyslový Ethernet s IP routrem (Obr. 12 a 13). Tím nabízejí zajímavou možnost připojení na PLC Simatic prakticky odkudkoliv. Výhodou eWon produktů je přímá integrace do prostředí Simatic Step 7. Nejsou potřeba žádné změny v PLC programu. Lze spravovat více PLC najednou [11].

**Funkčnost:**

### Vzdálená správa PLC

- Získávání, předávání a zaznamenávání dat.
- Správa alarmů a událostí, uživatelské programování chování modemu
- Hlášení prostřednictvím SMS, e-mailů, FTP souborů, trap SNMP, atd.

### Vzdálené řízení

- Intranetový nebo internetový přístup z Vašeho PLC software vývojového prostředí

**Nástin provedení:**

- program od eWON Software
- technik připojí eWon server k PLC přes MPI a přes Ethernet k PC
- v programu od eWon nastaví parametry pro navázání komunikace PLC s PC
- eWon aplikace umožňuje vlastní správu dat

**Cena:**

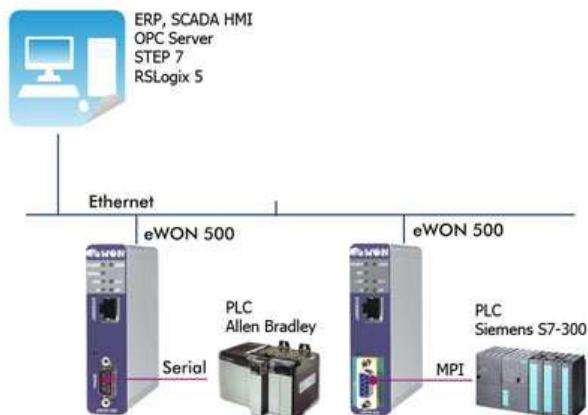
eWON 500/4001 - serial <=> Ethernet gateway - 11 700,- Kč s DPH

**Doplňující informace:**

V případě zájmu nabízí firma eWon software a hardware zdarma k zapůjčení na vyzkoušení na 7 dní.



Obr. 12 Ukázka eWon serverů



Obr. 13 Možnost řešení spojení pomocí eWon serveru

## 2.2.6 S7 DB TAGS

**Způsob připojení:**

MPI

**Popis:**

Runmode S7DBTAGS je 32-bitová aplikace pro Microsoft Windows 95/98/ME/NT/2000/XP/VISTA. Aplikace podporuje pouze projekty Siemens Simatic S7-300 a S7-400. S7DBTAGS operuje na základě exportovaného zdrojového AWL kódu a využívá dále exportovanou symbolickou tabulkou ve formátu ASC (Obr. 14 a 15). Oba tyto soubory je možné generovat v rámci Simatic Manageru.

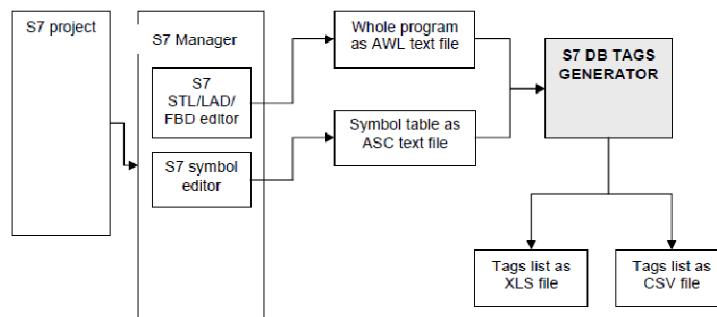
Aplikace podporuje jednoduché i složené datové typy. Není vyžadována instalace. Výstupem S7DBTAGS je Excelovská tabulka ve formátu XLS nebo CSV. Program je možné využít také v příkazové řádce [12].

### Nástin provedení:

- program od Runmode
- technik v Simatic Manageru vygeneruje zdrojový kód DB ve formátu AWL a také symbolickou tabulkou ve formátu ASC
- program S7 DB Tags vytvoří excelovskou reprezentaci DB na základě obou souborů
- provedeme požadované úpravy hodnot DB
- program nepodporuje import DB zpět do Step7 – nutno zvolit jiný způsob importu

### Cena:

S7 DB TAGS - 270,00 € (7 000,- Kč)



Obr. 14 Schéma generování výsledné tabulky pomocí S7DBTAGS

### Sample outputs

#### DB with mixed data types

DataBlock	Tag	Type	Address	Description	Initial value	Actual value
Constants	Ax11LowerLimitSwitch	DINT	DB5.DBDO	micron	L#135000	L#135000
Constants	Ax11UpperLimitSwitch	DINT	DB5.DBDO4	micron	L#2620000	L#2620000
Constants	Ax12LowerLimitSwitch	DINT	DB5.DBDO8	degrees *1000	L#0	L#0
Constants	Ax12UpperLimitSwitch	DINT	DB5.DBDO12	degrees *1000	L#189000	L#189000
Constants	Ax13LowerLimitSwitch	DINT	DB5.DBDO16	micron	L#0	L#0
Constants	Ax13UpperLimitSwitch	DINT	DB5.DBDO20	micron	L#0	L#0
Constants	Ax14LowerLimitSwitch	DINT	DB5.DBDO24	micron	L#0	L#0
Constants	Ax14UpperLimitSwitch	DINT	DB5.DBDO28	micron	L#0	L#0
Constants	Ax21LowerLimitSwitch	DINT	DB5.DBDO32	degrees *1000	L#2000	L#2000
Constants	Ax21UpperLimitSwitch	DINT	DB5.DBDO36	degrees *1000	L#272000	L#272000

Obr. 15 Ukázka výsledné tabulky generované S7DBTAGS

### 2.2.7 PC-MPI-S7-LINK

#### Způsob připojení:

MPI

## **Popis:**

Použitím jednoduchých funkcí v C, C++, Delphi, Visual Basicu nebo také v Excelu je uživatel schopen rychle přistoupit k datům v PLC. Spojení je realizováno pouze přes adresu MPI portu daného PLC. Uživatel může okamžitě číst a zapisovat do merkerů, časovačů, čítačů, vstupů, výstupů a datových bloků (Obr. 16). Není nutná žádná úprava programu v PLC.

Aplikace podporuje MS Windows 95, 98, 2000 nebo NT 4.0. Uživatel zkopíruje dll knihovnu pro výměnu dat mezi PC a Simaticem S7 do složky Windows.

Spojení je realizováno jako RS232 na PC-MPI-Adaptéru. Podporovány jsou PLC řady S7 300/400 s MPI rozhraním [13].

## **Nástin provedení:**

- program od firmy Traeger
- technik v Excelu vyplní parametry požadované k navázání komunikace s PLC přes MPI
- po stisknutí tlačítka dojde k importu DB
- provedeme požadované úpravy hodnot DB
- po stisknutí tlačítka dojde k exportu DB do PLC

## **Cena:**

PC-MPI-S7-LINK-S7-200/300/400 pro Win95 / 98 / NT / 2000 / XP / Vista

- jednotková	- 511,00 € (13 300,- Kč)
- firemní	- 2.044 € (53 100,- Kč)
- zdrojový kód	- 9.203 € (234 600,- Kč)
- školení	- 251,00 € (6 500,- Kč)

The screenshot shows a Microsoft Excel spreadsheet titled "PC-S7-LINK". The title bar includes "(Easy-Version-Demo)". The main content area has the following structure:

- Kopieren Sie "PCS7EASY.DLL" in ihren Windows-Pfad!**  
**Copy "PCS7EASY.DLL" to the Windows system directory!**
- Dieses Beispiel liest einen Datenbaustein ab DW0 10 Worte aus und stellt diese in der Tabelle dar.**  
**This example selects a DB starting from DW0 of 10 words and represents these in the table:**
- A table with columns "ComNr/ComNo.", "Werte: / value:", "Baudrate:", and "SPS-MPI-Adr:". The first row contains:

ComNr/ComNo.	1	Baudrate:	19200
local MPI-Adr	0	SPS-MPI-Adr:	2
SPS-Typ (200/300/400)	300	Ergebnis:/Result:	DB Lesen betätigen / press DB read
DBNr:/DBNo:	10		
- A button labeled "DB lesen / read DB" is located next to the table.
- TRAEGER**  
GmbH Industry Components  
Am Steigacker 26, D-92694 Eitzendorf  
Tel: 0961/48230-0 Fax: 0961/48230-20  
<http://www.traeger.de>

Obr. 16 Vzorová aplikace v Excelu s použitím PC-MPI-S7-LINK

## **2.2.8 IP-S7-LINK**

**Způsob připojení:**

Ethernet

**Popis:**

Uživatel si přidá do složky Windows speciální dll knihovnu, která slouží k výměně dat mezi PC a Simaticem S7 přes průmyslový Ethernet. Podporovány jsou systémy MS-WINDOWS (95/98/2000/NT/XP/Vista) a Linux.

Potřebný hardware tvoří standardní ethernetová karta v PC a v PLC S7 CP-343, CP343-1 IT, CP-443-1, CP443-1 IT, CP-343 LEAN, Profi-Net nebo S7-LAN.

Komunikace je realizována přes protokol TCP/IP (Obr. 17). Ke správnému připojení je potřeba IP adresa a číslo zásuvného modulu v racku. Uživatel může okamžitě číst a zapisovat do merkerů, časovačů, čítačů, vstupů, výstupů a datových bloků. Není nutná žádná úprava programu v PLC.

Podporovanými programovacími jazyky jsou C, C++, Delphi, Visual Basic, Excel, MS Word, MS Access [14].

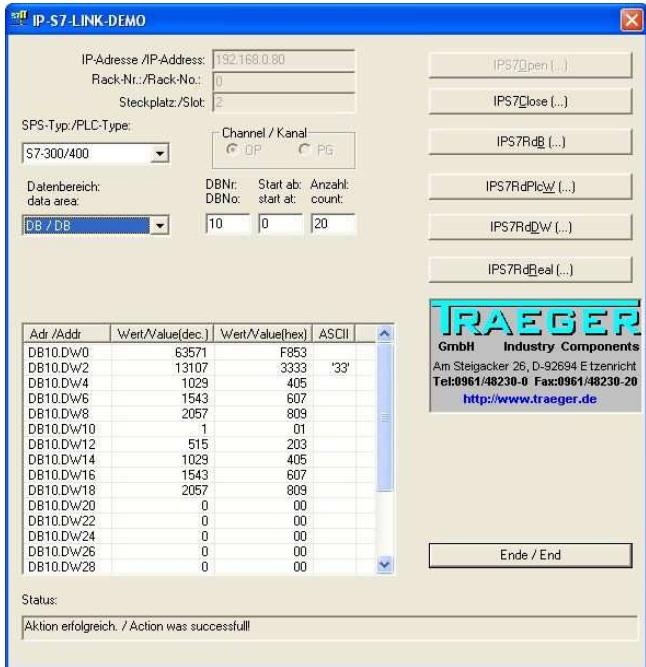
**Nástin provedení:**

- program od firmy Traeger
- technik v Excelu vyplní parametry požadované k navázání komunikace s PLC přes Ethernet
- po stisknutí tlačítka dojde k importu DB
- provedeme požadované úpravy hodnot DB
- po stisknutí tlačítka dojde k exportu DB do PLC

**Cena:**

IP-S7-LINK-S7-200/300/400 Win95 / 98 / NT / 2000 / XP / Vista

- jednotková	- 505,00 € (13 100,- Kč)
- firemní	- 3.057 € (79 500,- Kč)
- zdrojový kód	- 9.203 € (234 600,- Kč)
- školení	- 251,00 € (6 500,- Kč)



Obr. 17 Vzorová aplikace s použitím IP-S7-LINK

## 2.3 Výsledné doporučení

Z bezplatných možností lze doporučit knihovnu libnodave.dll (2.1.3 *Knihovna libnodave.dll*). Využitím jejích možností získáme přístup k datům v PLC přes rozhraní MPI nebo ethernet. Není potřeba modifikovat vlastní program v PLC. Technikovi stačí importovat knihovnu mezi knihovny ve Windows a tím získá podporu všech nabízených funkcí knihovny. Funkce lze využít při psaní maker definujících export a import DB. Dále stačí celý přenos obsluhovat z Excelu

Dále bych z bezplatných možností doporučil metodu exportu DB do souboru AWL (2.1.1 *Generovaný soubor AWL*). Na rozdíl od knihovny libnodave.dll zde musí technik disponovat programem Step7 a verzí programu nahraného v PLC. Výhodou je, že má technik během každého kroku přehled, co se s daty děje. Dále má možnost provést různé sady úprav dat DB a jednotlivé soubory AWL s různými úpravami importovat do Step7 a až zde se rozhodnout, kterou sadu úprav si vybere.

Z placených možností bych doporučil využít služeb firmy Traeger. Ta nabízí licenci na program pro přenos přes MPI nebo Ethernet (2.2.7 *PC-MPI-S7-LINK*, 2.2.8 *IP-S7-LINK*). Výhodou je hotová aplikace v Excelu, kde technik zadá parametry potřebné pro přenos. Po úspěšném navázání komunikace s PLC již může číst a zapisovat DB.

Jako další v úvahu připadá komunikační knihovna AGLink S7 (2.2.3 *AGLink S7 komunikační .DLL*). Použijeme-li implementované funkce knihovny, můžeme přistupovat

k datům v PLC, aniž bychom museli modifikovat vlastní program nahraný v PLC. Nevýhodou je fakt, že před vhodným uvedením do provozu, musí programátor za pomocí komunikační knihovny vytvořit vlastní aplikaci. Technik již pak bez nutnosti znalosti kódu aplikace může číst a zapisovat data z a do PLC. Tato varianta je cenově shodná s předchozí, kde uživatel získá již hotovou aplikaci. Ovšem zde ji zase může přizpůsobit přesně vlastním požadavkům.

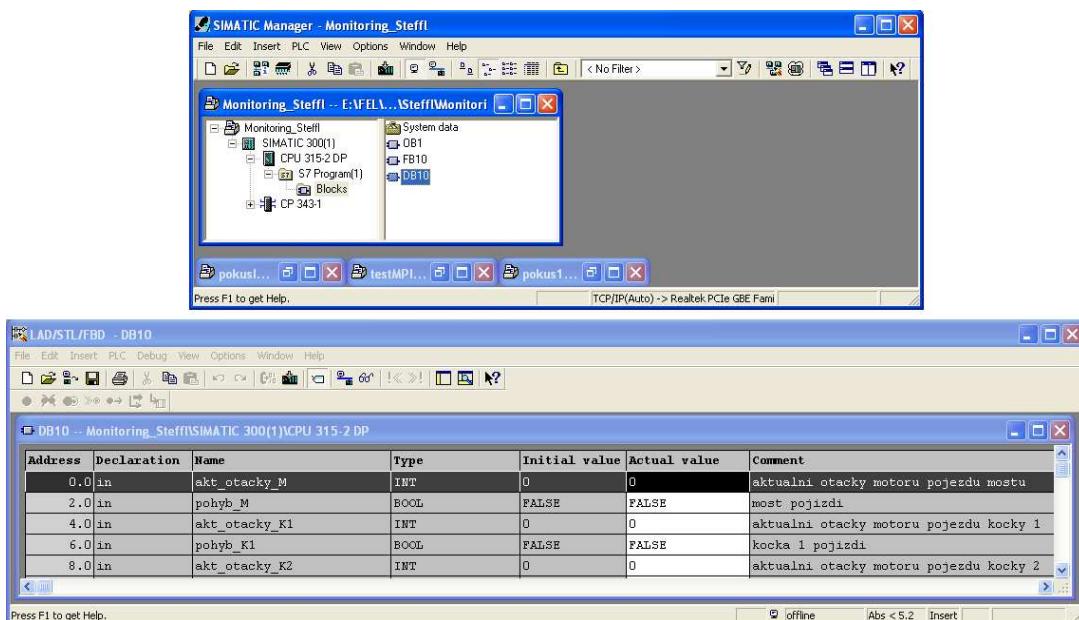
## 2.4 Otestování doporučených možností přenosu dat

### 2.4.1 Knihovna libnodave.dll

První uváděnou bezplatnou možnost, knihovnu libnodave.dll, jsem vybral jako nejvhodnější pro implementaci aplikace umožňující obousměrný dávkový přenos dat. Bude podrobně popsána v kapitole 3 *Knihovna libnodave.dll detailně*.

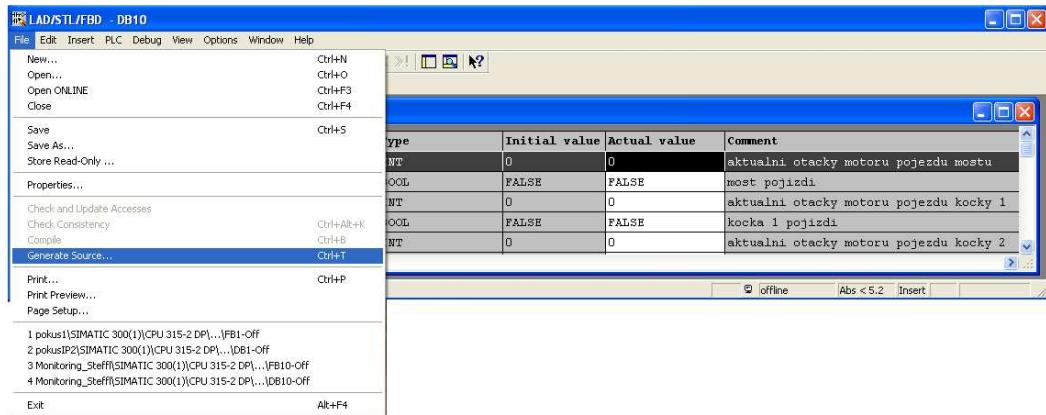
### 2.4.2 Generovaný soubor AWL

Druhá bezplatná možnost vyžaduje, aby měl uživatel k dispozici licencovaný program Simatic Manager od firmy Siemens AG. V Simatic Manageru otevřeme vybraný projekt obsahující datový blok, který požadujeme přenést do MS Excel pro další zpracování a archivaci (Obr. 18).



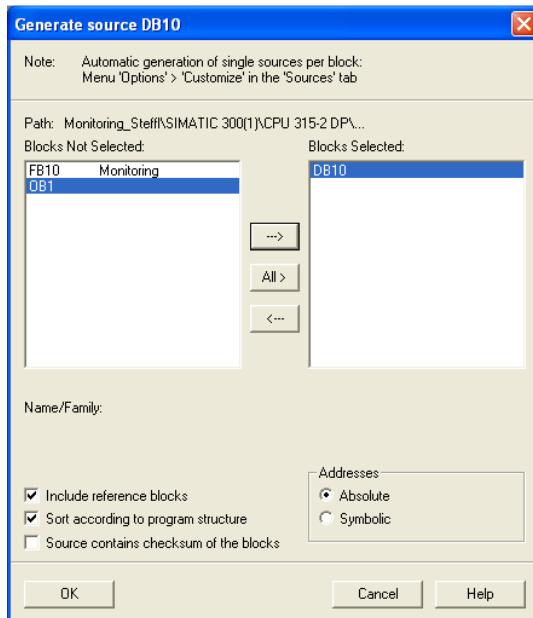
Obr. 18 Simatic Manager a prohlížeč DB

Dále vygenerujeme zdrojový kód vybraného datového bloku DB. V prohlížeči DB v možnosti „File“ vybereme položku „Generate Source...“ (Obr. 19). Zobrazí se nám okno, ve kterém musíme jako cílovou složku pro uložení zdrojového kódu DB vybrat složku „Sources“.



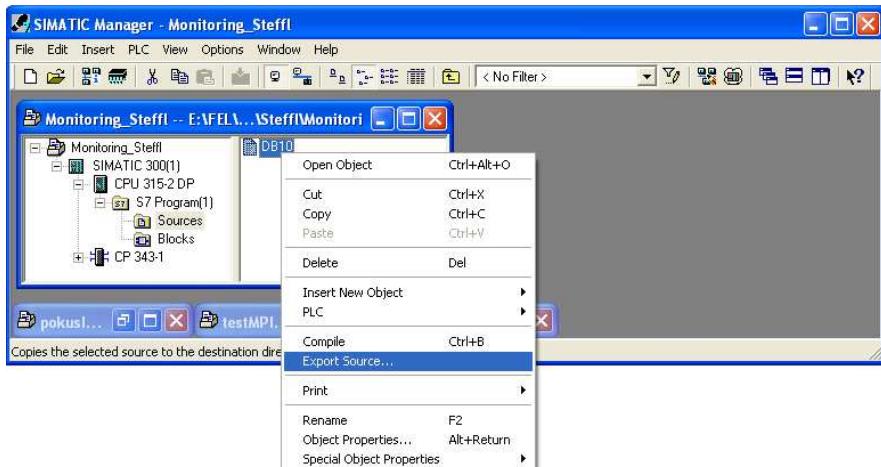
Obr. 19 Generování zdrojového kódu DB

Po zadání názvu souboru se zdrojovým kódem se otevře nové okno, kde máme možnost zvolit, který DB daného projektu si přejeme vygenerovat. Můžeme si vybrat, co všechno se v generovaném souboru má zobrazit (Obr. 20).



Obr. 20 Přiřazení dat pro vygenerování zdrojového kódu

Soubor se zdrojovým kódem zobrazíme výběrem podsložky „Sources“. Klikneme pravým tlačítkem myši na soubor a vybereme možnost „Export Source...“ (Obr. 21). Z nabízených formátů pro uložení souboru vybereme následující „STL Source (\*.awl)“.



Obr. 21 Exportování zdrojového kódu

Exportovaný soubor s příponou awl můžeme nyní otevřít v MS Excel. Nevýhodou tohoto způsobu je skutečnost, že se všechny údaje exportovaného datového bloku zobrazí pouze ve dvou sloupcích. Druhý sloupec tvoří komentáře. Pro další úpravy bychom tedy museli zobrazená data pomocí maker v MS Excel rozdělit na jednotlivé celky a dále pracovat s vyseparovanými hodnotami proměnných dle vlastních potřeb.

Upravená data požadujeme importovat zpět do datového bloku. Musíme tedy opět nejlépe pomocí maker obnovit podobu dat, kterou jsme získali exportem do souboru awl. Takto upravený excelovský soubor uložíme jako awl soubor a pomocí Simatic Manageru přes „Insert“ a „External Source...“ (Obr. 22) uložíme do podsložky „Sources“.



Obr. 22 Vložení externího souboru

Nyní musíme importovaný soubor otevřít a zkompilovat. Otevřeme možnost „File“ a vybereme „Compile“. Proběhne-li komplikace v pořádku, změní se data uložená v DB na nová(2).

Tento způsob přes vygenerovaný soubor typu awl je i přes poměrně velký počet kroků vedoucích k přenosu a následné archivaci dat snadný. Náročnou částí realizace aplikace založené na tomto způsobu by bylo napsání maker ve VBA, která by zajistila separaci a další

úpravu dat ze souboru awl. Další nevýhodu této metody představuje skutečnost, že by koncový uživatel musel vlastnit Simatic Manager a musel by s ním umět i pracovat.

#### **2.4.3 IP-S7-LINK**

Německá firma Traeger nabízí demoverzi svého programu jak pro přenos přes ethernet tak přes sběrnici MPI. Mohl jsem vyzkoušet pouze ethernetové připojení, jelikož demoverze pro přenos přes MPI sběrnici podporovala pouze sériové rozhraní, které jsem neměl k dispozici. Přidal jsem do systémového adresáře knihovnu výrobce IPS7LNK.dll a nastavil IP adresu ethernetového modulu CP 343-1 IT, na kterém pracuji. Poté jsem zadal číslo datového bloku a zahájil čtení (Obr. 23). Připojení proběhlo úspěšně a demoverze programu zobrazila prvních deset hodnot z datového bloku. Tím jsem vyčerpal funkce poskytované demoverzí.

	A	B	C	D	E
1					
2					
3					
4					
5					
6					
7					
8					
9					
10					
11					
12					
13		IP-Adresse: 192.168.0.1	Ergebnis: Lesen OK		
14		DBNr: 10			
15		Werte:			
16		500			
17		0			
18		1000			
19		0			
20		6006			
21		0			
22		745			
23		0			
24		0			
25		0			
26					

Obr. 23 Demoverze programu firmy Traeger pro připojení pře ethernet

#### **2.4.4 AGLink S7 komunikační .DLL**

Tuto komunikační knihovnu od německé firmy Deltalogic jsem nebyl bohužel schopen otestovat. Výrobce neposkytuje žádnou testovací demoverzi. Uvedu proto několik poznatků z jejich firemní licenční politiky. Pro tento produkt existují dvě možné licence. S jednoúčelovou licencí získá uživatel možnost pracovat pouze na jednom počítači. S vývojovou licencí získá uživatel možnost nasadit knihovnu v rámci celého pracoviště spolu s neomezenou délkou použití. Cena jednoúčelové licence se pohybuje kolem 10 000 Kč a cena za vývojovou licenci kolem 78 000 Kč [9].

### 3 Knihovna libnodave.dll detailně

V následujících podkapitolách uvádím detailní popis knihovny libnodave. Podávám informace o autorovi a licenční politice knihovny. Dále popisuji krok za krokem postup, jak lze realizovat spojení mezi PLC a PC. Jednotlivé kroky jsou zdokumentovány pro snadné pochopení.

#### 3.1 Úvod a licenční politika

Knihovna libnodave.dll je komunikační knihovna umožňující přenos dat mezi počítačem a programovatelnou řídicí jednotkou od firmy Siemens [4]. Knihovna podléhá licenční politice na základě podmínek GNU Library General Public License - GNU GPL (Všeobecná veřejná licence GNU) a je zcela zdarma ke stažení z webové stránky <http://libnodave.sourceforge.net/>. Podmínky GNU představují projekt svobodného softwaru. GPL licence je tzv. copyleftová licence. Copyleft je opakem copyrightu a zajišťuje ochranu svobody softwaru a i jeho případných modifikací, které na základě této licence opět podléhají GPL licenci [17].

Autorem knihovny je německý programátor Thomas Hergenhahn. Jeho původním cílem bylo poskytnout bezplatnou knihovnu napsanou v programovacím jazyku C. První verzi knihovny napsal v Linuxu a pro Linux. Tato verze libnodave-0.1 vyšla koncem roku 2002. Nejnovější dostupná verze knihovny libnodave-0.8.4.5 je z konce roku 2009. Jednotlivé verze představují aplikaci balíčky, které obsahují kromě vlastní knihovny libnodave.dll i funkční ukázky ve všech aktuálně podporovaných programovacích jazycích. Součástí balíčku nejnovější verze libnodave-0.8.4.5 jsou i podpůrné testovací skripty.

Knihovna představuje mocný a konkurenceschopný nástroj pro práci s daty v PLC. Konkurenční produkty podléhající copyrightové licenční politice mají společné vysoké pořizovací náklady, které jsou spojené s dalšími výdaji při obnovování licencí. Jako příklad zmíním funkčně podobnou knihovnu prodave.dll přímo od firmy Siemens AG nebo knihovny zmíněné v kapitolách 2.2.1 *INDI.AN PLC COM* a 2.2.3 *AGLink S7 komunikační DLL* od firem Deltalogic a Indi.an. Knihovna libnodave samozřejmě nenabízí stejný startovní komfort, jako když si uživatel koupí již hotový produkt připravený k použití. Pro práci s libnodave musí uživatel vynaložit počáteční programátorské úsilí. Výhodou je pak na míru implementovaná aplikace poskytující stejné možnosti jako konkurenční produkty.

## 3.2 Podporované programovací jazyky a způsoby připojení

Jak jsem již zmínil v minulé podkapitole, knihovna libnodave byla původně vytvořena a podporována pouze linuxovými operačními systémy. Od verze libnodave-0.5 se použitelnost knihovny rozšířila i na systémy Windows. Knihovnu je možno použít s následujícími programovacími jazyky: C, C++, C#, Delphi, Pascal, Perl a VBA. V přípravě je i podpora Javy, již dnes je k dispozici testovací třída. Vzhledem k tomu, že jsem vyvinul funkční aplikaci v MS Excel za použití VBA, budu se dále zmiňovat o možnostech knihovny při použití operačního systému Windows a programování ve VBA.

Knihovna podporuje přenos dat přes ISO model TCP/IP protokolu za použití ethernetu. Podporovány jsou procesory řady 243, 343 a 443 a i jejich -IT verze. Knihovna podporuje komunikační siemensovské knihovny a drivery, které umožňují práci s daty přes MPI adaptéry CP5611, CP5511 a USB-MPI firmy Siemens. Knihovna by měla podporovat i ostatní MPI a PPI adaptéry přes sériové rozhraní. Autor knihovny Thomas Hergenhahn dokonce nabízí možnost při nefunkčnosti některého z adaptérů, zaslat mu debugovaný výpis pokusu o připojení.

## 3.3 Práce s knihovnou

První krok k úspěšnému použití knihovny je zkopírovat libnodave.dll do systémového adresáře operačního systému, tedy do \Windows\System32\. Abychom mohli s knihovnou pracovat ve VBA, musíme do našeho projektu importovat modul obsahující veškeré konstanty, procedury a funkce, které kooperují s knihovnou libnodave.dll. Modul je základní programátorský nástroj pro práci s makry ve VBA. Ve verzi libnodave-0.8.4.5 se modul nachází v podložce s ukázkovým kódem pro VBA. Modul nese označení Modul12. Součástí modulu je i ukázkový kód pro práci s knihovnou. Uživatel s jeho pomocí může vyzkoušet, zdali dokáže s pomocí adaptéru, který má k dispozici, či ethernetového připojení navázat komunikaci s PLC. V následujících podkapitolách představím základní postupy při realizaci přenosu dat. Používám kód, který jsem implementoval ve své aplikaci. Jsou tudíž změněny některé názvy parametrů volání funkcí oproti původnímu modulu Modul12. Názvy proměnných jsou řešeny tak, aby byl kód přehlednější a čitelnější.

### 3.3.1 Navázání spojení s PLC

Navázání spojení obsluhuje funkce initialize. Funkce má tři parametry. Ty slouží jako návratové proměnné dalších tří funkcí, které jsou v rámci funkce initialize volány podle zvoleného způsobu připojení. Funkce initialize vrací hodnotu odkazující na úspěšné připojení.

## Ukázka volání funkce initialize:

```
Private Function initialize(ByRef ph As Long, ByRef di As Long, ByRef dc As Long)  
ph = 0  
di = 0  
dc = 0  
  
initialize = -1  
  
ph = openSocket(port, peer)  
  
If (ph > 0) Then  
    di = daveNewInterface(ph, ph, "IF1", 0, daveProtoISOTCP, daveSpeed187k)  
  
    res = daveInitAdapter(di)  
  
    If res = 0 Then  
        dc = daveNewConnection(di, 0, Rack, Slot)  
        res = daveConnectPLC(dc)  
  
        If res = 0 Then  
            initialize = 0  
        End If  
    End If  
End If  
End Function
```

Proměnná **ph** obsluhuje port v případě spojení přes sběrnici MPI a socket v případě ethernetového připojení. Její hodnota musí být větší než nula, v případě chyby nabývá záporných hodnot. Proměnná **di** obsluhuje rozhraní použitého protokolu a její hodnota je dále použita k inicializaci adaptéru. Návratová hodnota procesu inicializace musí být rovna nule, která odpovídá úspěšné inicializaci. Proměnná **dc** slouží k vytvoření spojení s PLC a její hodnota je použita jako parametr volání funkce pro připojení PLC a návratová hodnota musí být rovna nule. Obecně při práci s knihovnou libnodave jsou požadovány nulové návratové hodnoty funkcí nebo hodnoty kladné. Záporné hodnoty indikují problém v přenosové cestě ze strany adaptéru. Proměnné **ph**, **di** a **dc** jsou v základním modulu lokálními proměnnými, chce-li uživatel provádět více operací najednou, změní je na globální a ponechá je nastavené až do požadovaného zrušení spojení s PLC.

## Nastavení komunikačního kanálu:

- $ph = \text{setPort}(\text{serialport}, \text{baudrate}, \text{Asc}(\text{Left}$(\text{parity}, 1)))$

Použijeme-li MPI adaptér přes sériové rozhraní, jsou parametry volání funkce setPort název sériového portu (COM1 apod.), vnitřní znaková rychlosť sběrnice a paritný bit, ten je implicitně nastaven na hodnotu „O“ vyjadřující lichý paritní součet.

- $ph = \text{openS7online}(\text{serialport}, \text{Null})$

Pro MPI adaptér od firmy Siemens použijeme jako parametry volání název přístupového místa aplikace nastaveného v PG/PC rozhraní, standardně „S7ONLINE“, a ukazatel na handler chybového hlášení. Možnost připojení přes openS7online je podle slov autora knihovny libnodave nejnovější přírůstek. Během modifikací volání funkce v nových verzích knihovny došlo k přidání vstupního parametru funkce, ukazatele na handler, a volání funkce ve vzorovém modulu nebylo upraveno. Hodnota Null, kterou používám je tedy nepodložená.

- $ph = \text{openSocket}(\text{portNumber}, \text{IPaddress})$

Pro ethernetové připojení se musí otevřít socket. Parametry volání funkce jsou číslo portu, které je pro protokol TCP/IP rovno 102 a IP adresa PLC. Tu může uživatel libovolně nastavit v hardwarovém nastavení PLC.

## Inicializace adaptéru:

Funkce daveNewInterface reprezentuje fyzické spojení s PLC a nebo jeho vnitřní sítí. Spojení je vytvořeno na základě údajů o použité lokální adrese adaptéru, rychlosti sítě a protokolu použitého v dané síti. První dva parametry představují příchozí a odchozí komunikační kanály, třetí parametr odpovídá typu připojeného terminálu. První tři parametry uživatel nenastavuje.

- $di = \text{daveNewInterface}(ph, ph, "IF1", \text{mpiAddressOfAdapter}, \text{protocolType}, \text{speed})$

Pro MPI adaptér přes sériové rozhraní odpovídá volání výše uvedenému popisu. V knihovně jsou implementovány čtyři různé MPI protokoly. Rozdíly mezi nimi nejsou bohužel specifikovány.

- $di = \text{daveNewInterface}(ph, ph, "IF1", 0, 50, \text{speed})$

Pro MPI adaptér od firmy Siemens není potřeba zadávat lokální adresu adaptéru. Použijeme jeden jediný implementovaný protokol, jehož konstanta je 50. Dále zadáme požadovanou rychlosť sítě.

- **di**= daveNewInterface(**ph, ph, "IF1", 0, daveProtoSOTCP, daveSpeed187k)**

Ethernetové připojení také nemá lokální adresu adaptéru. Existují dva různé protokoly. Jeden pro řadu 300 a 400. Druhý pro řadu 200.

Hodnota vytvořeného fyzického spojení s PLC je zároveň parametr volání funkce inicializace adaptéru. Návratová hodnota musí být 0. Ve stručném popisu této funkce autor uvádí, že pouze MPI adaptéry potřebují inicializaci a že u ostatních je návratová hodnota vždy nulová. Uživatel má možnost nastavit delší timeout pro pomalá spojení. V méém případě se jedná o údaj zadaný v [ms], vstupní parametr procedury je v [ns].

```
Call daveSetTimeout(di, timeout* 1000)
```

```
res = davelInitAdapter(di)
```

### Připojení PLC:

Funkce daveNewConnection reprezentuje připojení PLC. Funkce potřebuje znát lokální adresu adaptéru, číslo použitého racku a slotu v PLC. Volání je pro všechny způsoby připojení stejné. Návratová hodnota slouží jako vstupní parametr funkce obsluhující vlastní připojení PLC. Hodnota musí být opět 0. Proběhne-li připojení úspěšně, je i návratová hodnota inicializační funkce rovna nule a PLC je připojeno s PC a připraveno na přenos dat.

```
dc = daveNewConnection(di, mpiAddressOfAdapter, rackNo, slotNo)
res = daveConnectPLC(dc)
```

```
If res = 0 Then
    initialize = 0
End If
End Function
```

### 3.3.2 Zrušení spojení s PLC

Spojení je zrušeno voláním funkce cleanUp. Nejdříve je odpojeno PLC, pak je odpojen adaptér. V případě MPI adaptéru přes sériové rozhraní je odpojen sériový port. U MPI adaptéru je zavřeno přístupové místo aplikace a u ethernetového spojení je zavřen socket.

```
Private Sub cleanUp(ByRef ph As Long, ByRef di As Long, ByRef dc As Long)
```

```
If dc <> 0 Then
    res = daveDisconnectPLC(dc)
```

```

Call daveFree(dc)
dc = 0
End If

If di<> 0 Then
    res = daveDisconnectAdapter(di)
    Call daveFree(di)
    di = 0
End If

If ph<> 0 Then

    res = closePort(ph) / res = closeS7online(ph) / res = closeSocket(ph)
End If
ph = 0

End If
End Sub

```

### 3.3.3 Čtení dat z PLC

Knihovna libnodave nabízí několik možností přístupu k datům uložených v PLC. Některé jsou označeny jako pouze implementované a jiné jako implementované a testované. Já se zde budu primárně zabývat postupy, které jsem použil při vytváření aplikace a následně se zmíním i o dalších implementovaných/testovaných možnostech. Funkce daveReadBytes se stará o načtení požadovaných dat, zároveň její návratová hodnota vypovídá o úspěchu operace.

- **res2**= daveReadBytes(**dc, daveDB, Db**, 0, 18, 0)

**Dc** je ukazatel na strukturu reprezentující navázané spojení s PLC. **DaveDB** je jedna z možných konstant specifikujících oblast paměti PLC. **Db** je číslo datového bloku. Dále následuje počáteční adresa, od které se data načtou. Další parametr udává počet čtených bytů a poslední parametr představuje buffer, ukazatel na místo v paměti, kde si přejeme, aby výsledky poslední čtecí operace byly uloženy. Návratová hodnota funkce daveReadBytes musí být nula, kladné návratové hodnoty představují problém na straně PLC a záporné jsou chyby detekované knihovnou libnodave.

Uvedené volání funkce čte z datového bloku **Db** 18 bytů od adresy 0 datového bloku. Jedná se například o 2 hodnoty typu REAL (2\*4 byty), 2 hodnoty typu DINT (2\*4 byty) a 1 hodnotu typu INT (1\*2 byty).

```
If res2= 0 Then
    v0= daveGetFloat(dc)
    v1= daveGetFloat(dc)
    v2= daveGetS32(dc)
    v3= daveGetS32(dc)
    v4= daveGetS16(dc)
```

```
Cells(1, 1) = v0
Cells(2, 1) = v1
Cells(3, 1) = v2
Cells(4, 1) = v3
Cells(5, 1) = v4
```

```
Else
    e$= daveStrError(res)
End If
```

Přiřazení jednotlivých načtených hodnot do proměnných nebo rovnou jejich výpis se provádí voláním funkcí na základě typu načtených proměnných. Uživatel musí přesně vědět, na kterých adresách se nachází která proměnná a jakého je typu. Volání funkce daveReadBytes nastaví vnitřní pozici bufferu na nulu a každé volání funkce konvertující byty na číslo tuto pozici inkrementuje v závislosti na typu proměnné.

Autor uvádí, že je rychlejší číst byty po blocích i za cenu přečtení několika nepotřebných bytů, než například přečíst 4 byty z určité adresy do bufferu a pak další 4 byty do inkrementovaného bufferu viz následující příklad.

```
daveReadBytes(dc, daveDB, 5, 68, 4, Buffer);
daveReadBytes(dc, daveDB, 5, 78, 4, Buffer+4);
```

Knihovna nabízí podporu dalších konstant specifikujících oblast paměti. Uvádím zde vybrané konstanty:

- davelnputs - vstupní byty
- daveOutputs - výstupní byty
- daveCounter - čítače
- daveTimer - časovače

Mezi podporované konvertovací funkce načtených bytů z bufferu patří :

- daveGetS8 a daveGetU8 – pro signed nebo unsigned BYTE
- daveGetS16 a daveGetU16 – pro signed nebo unsigned INT

daveGetS32 a daveGetU32 – pro signed nebo unsigned DINT

daveGetFloat – pro typ REAL

Funkce daveReadBytes umožňuje přečíst bloky o maximální velikosti 220 Bytů. Požadujeme-li čtení větších oblastí paměti, můžeme použít funkci daveReadManyBytes.

### 3.3.4 Zápis dat do PLC

Zápis dat do PLC je realizován obráceným postupem než čtení. Nejdříve naplníme buffer požadovanými hodnotami převedenými pomocí konvertovacích funkcí na jednotlivé byty. Opět platí, že je rychlejší nejdříve naplnit buffer a ten celý zapsat do PLC než zapisovat každou hodnotu zvlášť. Zápis umožňuje funkce daveWriteBytes.

- **pom\_dint**= davePutFloat(**buffer**(0), Cells(1, 1))  
**res2**= daveWriteBytes(**dc, daveDB, Db**, 38, 4, **buffer**(0))  
**e\$**= daveStrError(**res2**)

Nejdříve načteme požadovanou hodnotu a převedeme ji na základě jejího typu na bytovou reprezentaci, která je uložena do bufferu. Dále předáme buffer spolu s číslem datového bloku, adresou a zapisovanou délkou bytů z bufferu jako parametry funkci daveWriteBytes. Návratová hodnota funkce zápisu slouží jako vstupní parametr funkce daveStrError, která vrací řetězec „ok“ při úspěšném zápisu a řetězec „error“ při neúspěchu.

Knihovna podporuje stejné oblasti paměti pro zápis jako pro čtení. Mezi podporované a zároveň testované konvertovací funkce načítaných hodnot do bufferu patří:

davePut8 – pro typ BYTE  
davePut16 – pro typ INT  
davePut32 – pro typ DINT  
davePutFloat – pro typ REAL

### 3.3.5 Další možnosti

Jelikož ke knihovně libnodave neexistuje žádný manuál a ne všechny její možnosti jsou zdokumentovány, pokusím se krátce přiblížit i některé z dalších testovaných možností knihovny. Libnodave obsahuje funkce umožňující nastavit PLC do run a stop módu. Dále umožňuje načíst a zapsat vnitřní čas PLC. Máme možnost nechat si zobrazit seznam dostupných partnerů pro daný protokol. Dále můžeme nastavit a smazat bit. Knihovna dokonce podporuje čtení názvu datového bloku. Knihovna poskytuje i mnoho dalších implementovaných funkcí a procedur. Zájemce se nejdříve musí zamyslet, co od knihovny požaduje a poté jednotlivé postupy řešení vyzkoušet. Zájemce o použití knihovny libnodave

a všech jejích dalších nabízených možností bych tímto odkázal na německé fórum <http://sps-foren.de>, kde je členem i samotný autor knihovny, pro získání dalších cenných rad a informací [18].

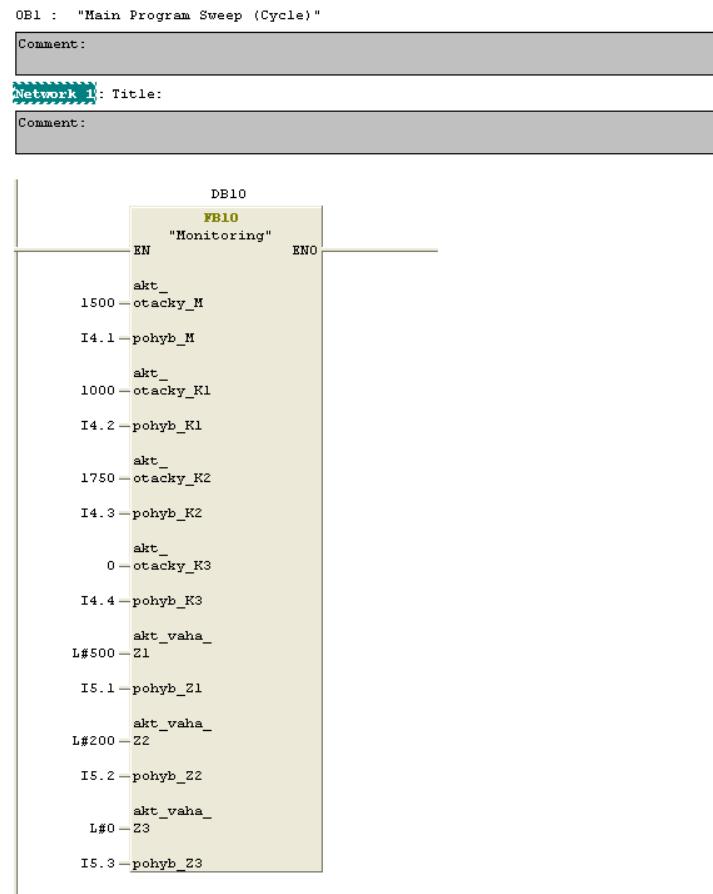
## 4 Funkční a datový blok

Monitorování provozu mostového jeřábu je realizováno pomocí funkčního bloku nahraného do řídicího PLC Simatic S7. Tato kapitola popisuje, jak funkční blok zpracovává vstupní data od senzorů mostového jeřábu a jak jsou data uložena v datovém bloku.

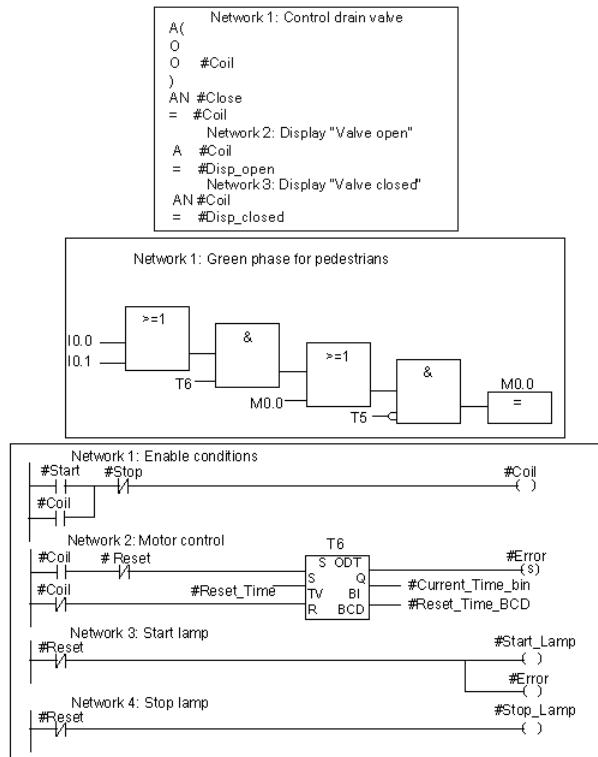
### 4.1 Funkční blok

#### 4.1.1 Obecný popis

Rozsáhlé řídicí aplikace naprogramované v prostředí Step 7 se pro větší přehlednost a usnadněnou kontrolu správné funkčnosti rozdělují do více logických celků obsahujících část programu. Programátor může vytvářet funkce FC bez statické paměti nebo funkční bloky FB se statickou pamětí. Funkce a funkční bloky zastřešuje v programátorské hierarchii Step 7 organizační blok OB. Ten probíhá cyklicky a vykonává v sobě obsažený kód. Existuje více druhů OB v závislosti na časovém odstupu vykonávaných cyklů tzv. scan cyklů. Základní je OB1 probíhající v nekonečné smyčce. Funkce a funkční bloky mohou být v organizačním bloku volány vícekrát na různých místech v závislosti na jejich potřebě. (Obr. 24). Programátor má možnost psát FB ve třech různých programátorských jazycích: STL, LAD a FBD (Obr. 25). STL (Statement List Programming Language) je založen na textové podobě programování. FBD (Function Block Diagram Programming Language) je založen na grafické a symbolické logice známe z boolské algebry. LAD (Ladder Logic Programming Language) je založen na obvodové reprezentaci. LAD je též znám pod názvem „žebříčkový diagram“. Pro realizaci monitorovacího FB jsem zvolil LAD [15], [16].



Obr. 24 Příklad volání funkčního bloku Monitoring v OB1



Obr. 25 Příklad programátorůských jazyků STL (nahoře), FBD (uprostřed) a LAD (dole)

#### **4.1.2 Monitorovací funkční blok**

V našem případě se jedná o vstupní funkční blok. FB dostává na vstup data od senzorů měřících provoz mostového jeřábu (Obr. 26). Jeřáb se skládá ze tří hlavních částí. První část tvoří samotný most. Druhou část tvoří pracovní kabina mostu s možností pohybu v kolmém směru na směr pohybu mostu. Její pracovní označení je kočka. Třetí část představuje zdvižný mechanismus, na který se upevňuje převážené závaží. Pracovní název je zdvih. Dle požadavků na zpracování monitorovacího funkčního bloku vychází z možnosti nasazení až tří koček a až tří zdvihů v libovolné kombinaci na jednom mostě. Hlavním úkolem FB je vstupní data zpracovat do výsledné podoby zadané firmou SIMO-CZ s.r.o. Zpracovaná data budou uložena v datovém bloku DB, se kterým funkční blok pracuje a který je určen k přenosu dat do aplikace v MS Excel pro další zpracování a archivaci. V následujících dvou podkapitolách uvádím teoretický popis zpracování dat. V případě kočky a zdvihu pro zjednodušení uvádím vždy jen jednotné číslo, jelikož úkony spojené s výpočtem dat pro kočku a zdvihu jsou shodné pro všechny tři kočky a zdvihy.



Obr. 26 Ukázka dvou mostových jeřábů s elektroinstalací firmy SIMO-CZ s.r.o.

#### **Most a kočka:**

Most a kočka mají shodné vstupní i statické veličiny. Vstupními veličinami jsou aktuální otáčky pojazdu mostu a kočky a proměnná typu BOOL udávající, jestli jsou most nebo kočka právě v pohybu. Vstupní veličiny typu BOOL mají řídící význam. Pro účely zpracování těchto dat musíme ještě znát nominální rychlosti pojazdu mostu a kočky a také hodnotu otáček, při kterých je nominální rychlosti pojazdu mostu a kočky dosahováno, viz Tab. 1. Tyto údaje tvoří statické veličiny a musí být specifikovány předem.

Mezi statické veličiny mostu a kočky také patří funkčním blokem zpracovávané výše zmíněné údaje, které se dále využívají k výpočtu provozních veličin. Funkční blok k výpočtu využívá i potřebné dočasné veličiny. Ty nejsou uloženy v DB. Cílem monitorovacího bloku je držet přehled o provozních hodinách, redukovaných provozních hodinách, ujeté dráze a počtu sepnutí pro pojezd mostu a kočky.

Provozní hodiny se počítají jako součet dílčích časových úseků vnitřního časovače funkčního bloku. Ten se opakuje s periodou 200 ms. Časovač se spustí a běží v nekonečné smyčce, právě když boolská proměnná udávající pohyb je na hodnotě true. Zároveň je při každém přechodu této proměnné z false na true inkrementován čítač počtu sepnutí.

Pro výpočet redukovaných provozních hodin je nutné znát koeficient zatížení. Ten v případě mostu odpovídá součtu zátěže na všech nasazených zdvihách ku nosnosti mostu. V případě kočky odpovídá součtu zátěže zdvihů na kočce zavěšených ku součtu jejich nosností. Redukované provozní hodiny počítám stejně jako provozní hodiny, ale před přičtením časového intervalu vnitřního časovače tento časový úsek vynásobím aktuálním koeficientem zatížení.

Abych byl schopen vypočítat ujetou dráhu pojezdu mostu a kočky, potřebuji znát koeficient rychlosti. Ten odpovídá podobně jako koeficient zatížení časovému rozložení jednotlivých nabývaných hodnot otáček motoru mostu a pojezdu kočky ku otáčkám, při kterých je dosahováno nominální rychlosti. Koeficient rychlosti pak obdobně jako koeficient zatížení slouží k počítání pomocných redukovaných hodin rychlosti. Ze znalosti nominální rychlosti pojezdu mostu a kočky jsem pak schopen vypočítat ujetou dráhu. Ta je rovna součinu pomocných redukovaných hodin a nominální rychlosti. Výpočet ujeté dráhy provádím vždy, když se motor mostu nebo pojezdu kočky zastaví. Tedy vždy když boolská proměnná udávající pohyb přejde z hodnoty true na false.

### Zdvih:

Vstupní veličiny zdvihu jsou aktuální váha břemene a opět proměnná typu BOOL udávající při hodnotě true, že zdvih spouští nebo zdvihá. Statické veličiny specifikované předem tvoří celková nosnost jeřábu, nosnost zdvihu, hmotnost prázdného háku a umístění zdvihu, viz Tab. 1. Hmotnost prázdného háku odpovídá hmotnosti, do které se hák považuje za prázdný. Umístění zdvihu udává, na které kočce je který zdvih zavěšen. Může se jednat o libovolnou kombinaci. Přípustná je i varianta, kdy jsou na jedné kočce zavěšeny tři zdvihy.

Další statické veličiny tvoří funkčním blokem zpracovávaná data. Monitorovací blok má za úkol spočítat provozní a redukované provozní hodiny zdvihu, počet sepnutí a počet cyklů zdvihu, zatěžovací diagram a počet přetížení zdvihu nad 110%. Jeden cyklus u zdvihu znamená naložení, převoz a vyložení břemene. Zatěžovací diagram udává procentuelní rozložení převážené zátěže během provozu s krokem deseti procent. Uvažujeme i možnost přetížení zdvihu až do hodnoty 120%.

Výpočet provozních a redukovaných provozních hodin je shodný jako u mostu a kočky. Rozdíl je ve výpočtu koeficientu zatížení zdvihu. Koeficient odpovídá poměru hmotnosti aktuálně převážené zátěže a nosnosti zdvihu. Prázdný hák sám o sobě něco váží a je proto vhodné pro správné vypočtení koeficientu zatížení tuto hmotnost odečítat jak od aktuální váhy, tak od nosnosti zdvihu. Díky odečtení vlastní hmotnosti háku lze pokrýt celé zatěžové spektrum od 0% do 120%.

Na základě aktuálně spočítaného koeficientu zatížení jsou pro jednotlivé intervaly odpovídající kroku po deseti procentech vypočítány časové přírůstky provozních hodin. Tím získám základ pro výpočet zátěžového diagramu. Provozní hodiny zdvihu odpovídají sumě všech časů jednotlivých intervalů koeficientu zatížení. Když boolská proměnná zdvihu přejde na hodnotu false, spočítám vlastní zátěžový diagram z poměru časů jednotlivých intervalů ku provozním hodinám. Výpočet se provádí až po ukončení provozu zdvihu, aby nebyl scan cyklu organizačního bloku zatížen výpočtem při každém průchodu.

Počet sepnutí zdvihu se opět inkrementuje při každém přechodu boolské proměnné z hodnoty false na true. Výpočet cyklů zdvihu je podmíněn převezením a vyložením zátěže. Je-li aktuální hmotnost zdvihu rovna hmotnosti prázdného háku, nastavím pomocnou proměnnou a tím zahajím hlídání cyklu. Termín nastavit znamená nastavit z hodnoty false na true. Jakmile dojde k přemístění háku a zároveň je nasetována pomocná proměnná, zbývá počkat na vyprázdnění zdvihu, které nastane, je-li opět aktuální hmotnost zdvihu rovna hmotnosti prázdného háku.

Počet přetížení zdvihu nad 110% je realizováno pomocným bitem. Je-li koeficient zatížení větší než 110% nastavím pomocný bit a jakmile boolská hodnota zdvihu přejde na false, inkrementuji čítač přetížení.

Tab. 1 Přehled vstupních a statických provozních veličin jeřábu

	<b>most</b>	<b>kočka</b>	<b>zdvh</b>
vstupní veličiny	aktuální otáčky	aktuální otáčky	aktuální váha
	pohyb	pohyb	pohyb
statické veličiny	nominální rychlosť pojazdu	nominální rychlosť pojazdu	nosnost jeřábu
	hodnota otáček pri nominálnej rychlosťi	hodnota otáček pri nominálnej rychlosťi	nosnosť zdvihu
			hmotnosť prázdného háku
			umiestnenie zdvihu
zpracované statické veličiny	provozní hodiny	provozní hodiny	provozní hodiny
	redukované provozní hodiny	redukované provozní hodiny	redukované provozní hodiny
	ujetá dráha	ujetá dráha	počet sepnutí
	počet sepnutí pojazdu	počet sepnutí pojazdu	počet cyklů
			zátěžový diagram
			počet přetížení nad 110 %

## 4.2 Datový blok

### 4.2.1 Obecný popis

Datový blok neobsahuje žádnou část vlastního kódu. Je používán jako datové úložiště proměnných, se kterými pracují funkční bloky, funkce a organizační bloky. Označení používané ve Step 7 je sdílený datový blok DB, to proto, že na data uložená v jednom

datovém bloku mohou přistupovat i jiné funkční bloky. Programátor má možnost DB libovolně strukturovat. Datový blok obsahuje přehled o původních nastavených hodnotách a aktuálních hodnotách uložených v DB po volání FB.

#### **4.2.2 Monitorovací datový blok**

Datový blok obsahující všechna provozní data o provozu mostového jeřábu, která jsou uvedena v předchozí podkapitole 3.1.2 *Monitorovací funkční blok*, generuje automaticky ve funkčním bloku. Takto vytvořený DB obsahuje všechny vstupní a statické proměnné daného funkčního bloku (Obr. 27). Jedním z hlavních cílů této práce je vytvořit aplikaci, která umožní data DB zálohovat. To znamená, že aplikace uživateli umožní data DB přenést a uložit. Od aplikace požadujeme i možnost zápisu dat do DB. Data z DB jsou přenášena po bytech. Jednotlivé byty jsou v DB adresovány postupně. Přidělený počet adres záleží na typu proměnné. Vlastní realizace přenosu bude popsána v následující kapitole 4 *Knihovna libnodave.dll detailně*.

10.0.in	pohyb_K2	BOOL	FALSE	FALSE	kocka 2 pojizdi
12.0.in	akt_otacky_K3	INT	0	0	aktuální otacky motoru pojezdů kocky 3
14.0.in	pohyb_K3	BOOL	FALSE	FALSE	kocka 3 pojizdi
16.0.in	akt_vaha_Z1	DINT	L#0	L#0	aktuální vaha bremene na zdvihu 1
20.0.in	pohyb_Z1	BOOL	FALSE	FALSE	zdvih 1 zdviha nebo spusti
22.0.in	akt_vaha_Z2	DINT	L#0	L#0	aktuální vaha bremene na zdvihu 2
26.0.in	pohyb_Z2	BOOL	FALSE	FALSE	zdvih 2 zdviha nebo spusti
28.0.in	akt_vaha_Z3	DINT	L#0	L#0	aktuální vaha bremene na zdvihu 3
32.0.in	pohyb_Z3	BOOL	FALSE	FALSE	zdvih 3 zdviha nebo spusti
34.0.stat	nom_pojezd_M	INT	0	0	nominalní rychlosť pojezdu mostu
36.0.stat	mot_pojezd_M	INT	0	0	otacky motoru, pri ktorých je dosahovaná nomi
38.0.stat	prov_hodiny_M	REAL	0.000000e+000	0.000000e+000	provozní hodiny pojezdu mostu
42.0.stat	red_prov_hodiny_M	REAL	0.000000e+000	0.000000e+000	redukované provozní hodiny pojezdu mostu
46.0.stat	red_hod_rychlosti_M	REAL	0.000000e+000	0.000000e+000	redukované hodiny rychlosť pojezdu mostu
50.0.stat	draha_M	REAL	0.000000e+000	0.000000e+000	účtená draha pojezdu mostu

Obr. 27 Ukázka části monitorovacího datového bloku

## 5 Vlastní aplikace v MS Excel

Aplikaci jsem vytvářel v Microsoft Office Excel 2007 a Microsoft Visual Basic 6.5, který je součástí verze Excelu 2007 [19]. Základem aplikace jsou tři pracovní listy. První s názvem „Nastavení“ umožňuje uživateli vybrat jeden ze dvou způsobů připojení k PLC. Uživatel má na výběr připojení přes sběrnici MPI nebo přes protokol TCP/IP v ethernetové síti. Po nastavení požadovaných parametrů pracuje uživatel dále ve druhém pracovním listu s názvem „Přenos“. V tomto listu se nachází přehledné tabulky pro zpracování a vizualizaci přenášených dat z PLC. Uživatel zde ovládá připojení k PLC a přenos dat. Vybraná data reprezentující zátěžové diagramy jednotlivých zdvihů slouží ke generování grafického znázornění diagramů. Uživatel má možnost přenesená data a vygenerované grafy automaticky zálohovat do nového listu. Oba zmíněné listy a všechny jejich funkce podrobně popíší v dalších podkapitolách.

Třetím pracovním listem je list „Vzor DB“ (Obr. 28). Zde je zobrazen kompletní datový blok referující na monitorovací funkční blok. Zobrazení je kopíí reprezentace datového bloku v Simatic Manageru. Upravil jsem pouze vzhled a rozdělil data na logické celky do tří přehledných tabulek. První tabulka reprezentuje data mostu, druhá tabulka data kočky 1, 2 a 3. Třetí tabulka zobrazuje data zdvihu 1, 2 a 3. Tento list s datovým blokem slouží jako vzorová záložní kopie. Uživatel má neustále a i bez přístupu k Simatic Manageru možnost podívat se na použité typy proměnných, jejich adresy a pracovní jména.

MOST						
Address	Declaration	Name	Type	Initial value	Actual value	Comment
5,0	in	akt_otacky_M	INT	0	0	[ot/min] aktuální otacky motoru pojedzu mostu
6,0	in	pohyb_M	BOOL	FALSE	FALSE	[F,T] most pojizdi
7,0	stat	nom_pojezd_M	INT	0	0	[m/min] nominální rychlosť pojedzu mostu
8,0	stat	mot_pojezd_M	INT	0	0	[ot/min] otacky motoru, pri ktorých je dosahovaná nominálna rychlosť pojedzu mostu
9,0	stat	prov_hodiny_M	REAL	0,00E+00	0,00E+00	[min] provozní hodiny pojedzu mostu
10,0	stat	red_prov_hodiny_M	REAL	0,00E+00	0,00E+00	[min] redukovane provozni hodiny pojedzu mostu
11,0	stat	red_hod_rychlosťi_M	REAL	0,00E+00	0,00E+00	[min] redukovane hodiny rychlosťi pojedzu mostu
12,0	stat	draha_M	REAL	0,00E+00	0,00E+00	[m] ujeda draha pojedzu mostu
13,0	stat	pocet_sepnuti_M	DINT	L#0	L#0	[...] pocet sepnuti pojedzu mostu

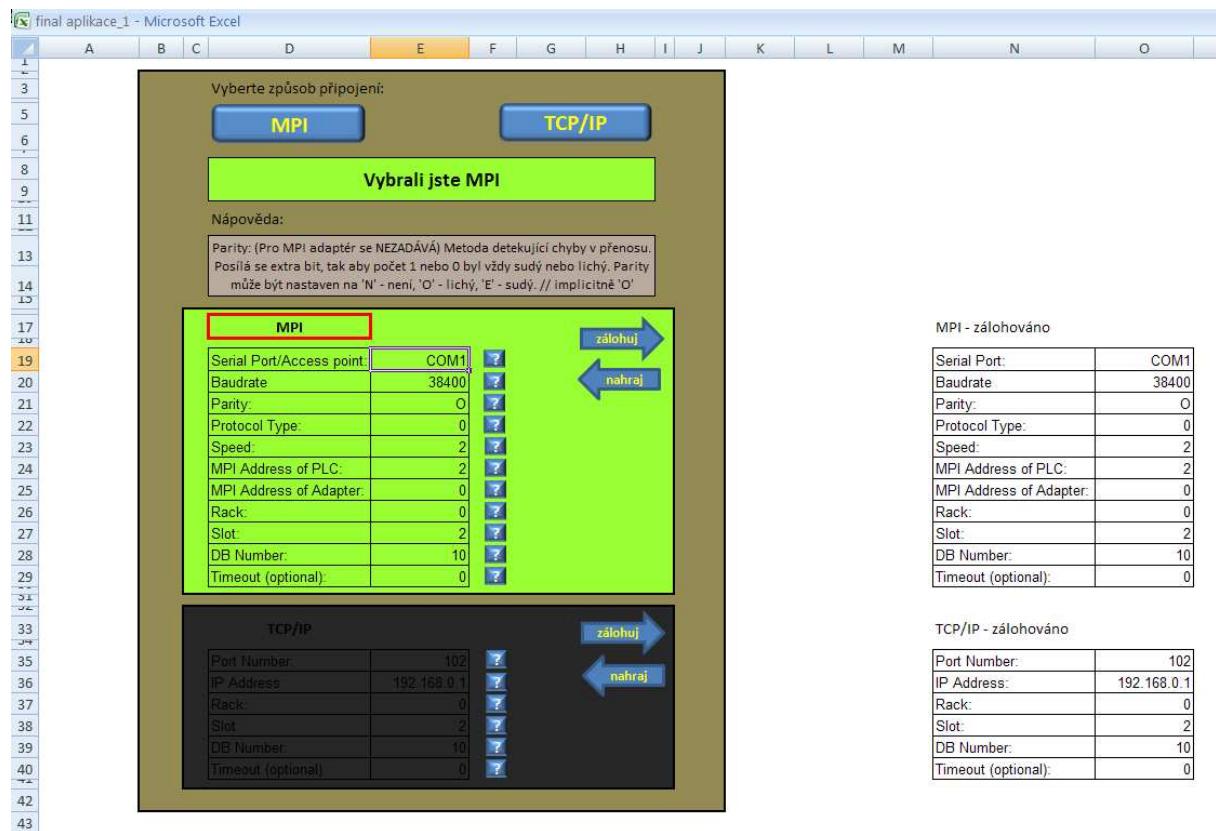
KOČKA 1, 2, 3						
Address	Declaration	Name	Type	Initial value	Actual value	Comment
19,0	in	akt_otacky_K1	INT	0	0	[ot/min] aktuální otacky motoru pojedzu kočky 1
20,0	in	pohyb_K1	BOOL	FALSE	FALSE	[F,T] kočka 1 pojizdi
21,0	in	akt_otacky_K2	INT	0	0	[ot/min] aktuální otacky motoru pojedzu kočky 2
22,0	in	pohyb_K2	BOOL	FALSE	FALSE	[F,T] kočka 2 pojizdi
23,0	in	akt_otacky_K3	INT	0,00E+00	0	[ot/min] aktuální otacky motoru pojedzu kočky 3
24,0	in	pohyb_K3	BOOL	FALSE	FALSE	[F,T] kočka 3 pojizdi
25,0	stat	nom_pojezd_K1	INT	0,00E+00	0	[m/min] nominálna rychlosť pojedzu kočky 1
26,0	stat	mot_pojezd_K1	INT	0,00E+00	0	[ot/min] otacky motoru, pri ktorých je dosahovaná nominálna rychlosť pojedzu kočky 1
27,0	stat	prov_hodiny_K1	REAL	0	0,00E+00	[min] provozní hodiny pojedzu kočky 1
28,0	stat	red_prov_hodiny_K1	REAL	0	0,00E+00	[min] redukovane provozni hodiny pojedzu kočky 1
29,0	stat	red_hod_rychlosťi_K1	REAL	0	0,00E+00	[min] redukovane hodiny rychlosťi pojedzu kočky 1
30,0	stat	draha_K1	REAL	0	0,00E+00	[m] ujeda draha pojedzu kočky 1
31,0	stat	pocet_sepnuti_K1	DINT	L#0	L#0	[...] pocet sepnuti pojedzu kočky 1
32,0	stat	nom_pojezd_K2	INT	0,00E+00	0	[m/min] nominálna rychlosť pojedzu kočky 2

Obr. 28 Pracovní list „Vzor DB“ – ukázka datového bloku

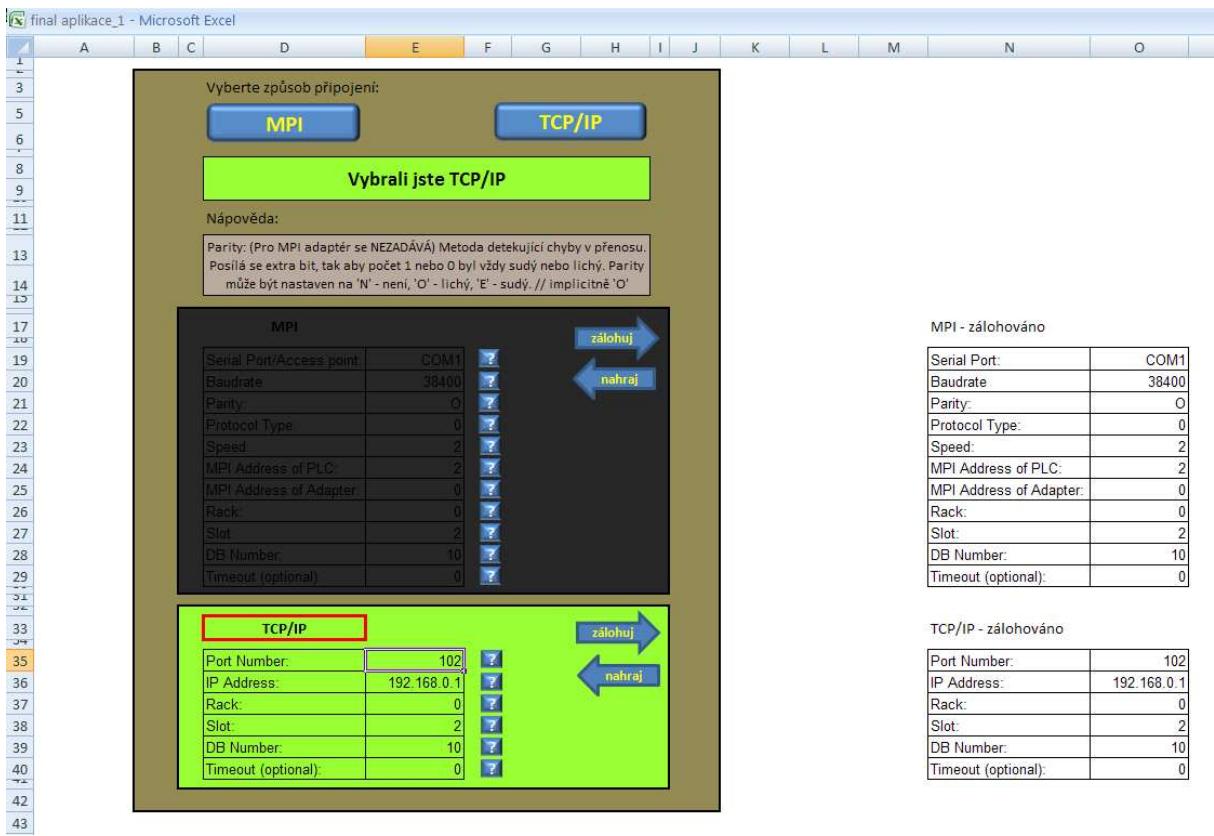
## 5.1 Pracovní list Nastavení

Tento list slouží výhradně k výběru a nastavení parametrů připojení. List obsahuje interaktivní formulář, kde uživatel pomocí dvou tlačítek ovládá výběr způsobu připojení. První tlačítko slouží pro výběr sběrnice MPI (Obr. 29) a druhé pro výběr protokolu TCP/IP v ethernetové síti (Obr. 30). Uživatel je o zvoleném způsobu informován v okénku pod tlačítky. Výběr je též graficky znázorněn pomocí zeleného podbarvení indikujícího aktivní tabulkou pro zadání parametrů připojení. Součástí obou tabulek nastavení parametrů je možnost nastavené hodnoty připojení zálohovat nebo nahradit. Zálohovací tabulky jsou součástí listu „Přenos“ a nepřímou součástí interaktivního formuláře.

Formulář obsahuje také okno nápovědy, kde je možné zobrazit nápovědu ke každému ze zadávaných parametrů připojení. Součástí nápovědy jsou i implicitní hodnoty umožňující první hrubou konfiguraci. Nápovědu pro jednotlivé parametry uživatel zobrazí stisknutím tlačítka nápovědy na konci každého parametru. Po opětovném stisknutí nápověda zmizí. Interaktivní formulář neumožňuje zálohovat/nahrávat nebo zobrazovat nápovědu k parametrům neaktivního způsobu připojení.



Obr. 29 Pracovní list „Nastavení“ – vybraný způsob MPI



Obr. 30 Pracovní list „Nastavení“ – vybraný způsob TCP/IP

## 5.2 Pracovní list Přenos

Pracovní list „Přenos“ tvoří hlavní část aplikace, je složen ze dvou částí (Obr. 31). Ovládací panel na levé straně listu tvoří jednu část. Druhou část tvoří přehledné tabulky připravené pro práci a archivaci provozních dat mostového jeřábu. Jsou vytvořené na míru dle požadavků firmy SIMO-CZ s.r.o.

### 5.2.1 Práce s PLC

Celkem se v listu nachází tři tabulky. První tabulka sdružuje provozní data mostu, druhá kočky 1, 2 a 3 a třetí tabulka obsahuje provozní data zdvihu 1, 2 a 3. Všechny tři tabulky se skládají ze dvou graficky odlišených skupin dat. První skupinu tvoří vstupní statické hodnoty, které jsou nastaveny provozovatelem jeřábu již v samotném programu před nahráním do programovatelné řídicí jednotky. V tabulkách jsou uváděny s ohledem na jejich výpovodní hodnotu. Jsou důležité pro archivaci, jelikož jsou základem pro výpočet všech provozních dat obsažených ve druhé skupině dat. Aplikace umožňuje první skupinu dat pouze číst. Čtení slouží ke kontrole správně nastavených hodnot. Zápis není povolen z důvodu provozní bezpečnosti. Druhou skupinu dat tvoří samotné provozní hodnoty. Aplikace umožňuje tyto hodnoty číst a zapisovat. Tabulky sestávají z popisu proměnných a přidružených aktuálních hodnot včetně uvedených jednotek.

Po výběru a nastavení správných hodnot připojení uživatel stiskem tlačítka „Připoj PLC“ spustí makro v pozadí aplikace, které se pokusí připojit aplikaci k PLC s pomocí knihovny libnodave.dll. V případě, že uživatel nezadal všechny parametry, bude aplikací vyzván, aby tak učinil. Aplikace nehlídá způsob zadání parametrů a tudíž, zadá-li uživatel parametry pro připojení špatně, nedojde k úspěšnému připojení. Uživatel je během práce s aplikací neustále informován o právě probíhající operaci a jejím stavu. Panel se stavovými informacemi je součástí ovládacího panelu.

Dojde-li k úspěšnému připojení aplikace s PLC, může uživatel načíst data z datového bloku do aplikace pomocí stisknutí tlačítka „Načti DB z PLC“. Další tlačítko v pořadí „Vynuluj DB“ nuluje všechny data v tabulkách. Uživatel může aplikaci použít i k zapsání dat do PLC, k tomu slouží tlačítko „Zapiš DB do PLC“. Zápis dat do PLC je vhodný pro nulování některých provozních hodnot přímo v PLC, aniž by provozní technik musel do PLC nahrávat novou verzi řídicího programu. Zápis a čtení dat jsou možné pouze tehdy, je-li PLC úspěšně připojeno. V opačném případě je uživatel informován, že PLC není připojeno. Pro ukončení práce s daty v PLC uživatel použije tlačítko „Odpoj PLC“, které uzavře spojení s PLC a vynuluje řídicí proměnné spojení používané knihovnou libnodave.dll.

### **5.2.2 Práce s daty v Excelu**

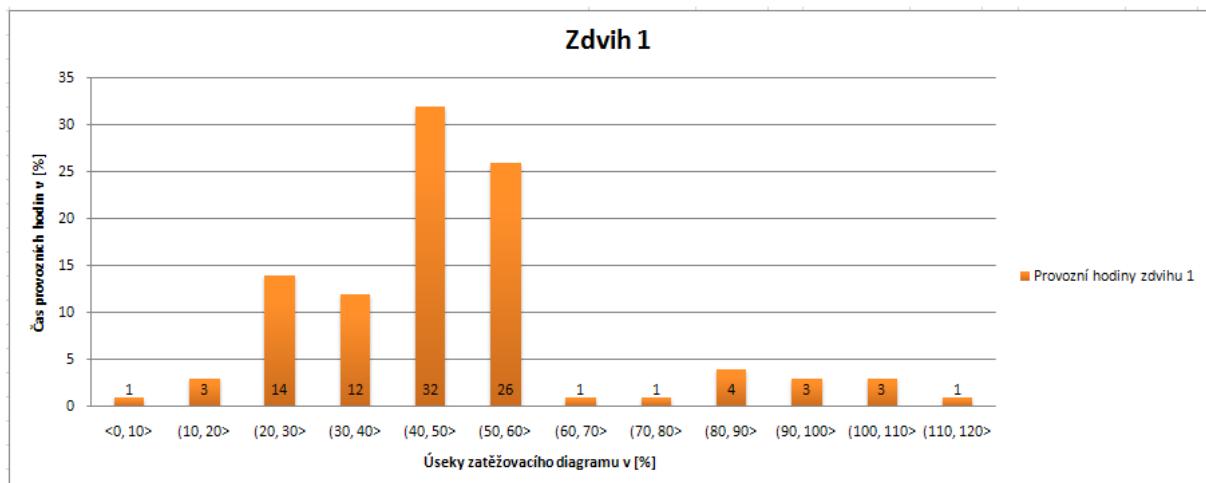
Součástí zpracovávaných dat zdvihu 1, 2 a 3 jsou i zátěžové diagramy (Obr. 32). Data představují procentuální rozložení zátěže zdvihů v čase a jsou rozdělena na dvanáct intervalů s krokem po 10 procentech. Je tedy počítáno i s přetížením až o 20 percent. Součástí ovládacího panelu jsou i dvě tlačítka „Generuj grafy“ a „Smaž grafy“. Uživatel má možnost nechat si automaticky vygenerovat zátěžové diagramy nebo je smazat, aby mohl vygenerovat nové. Grafy jsou po vygenerování umístěny pod tabulkou hodnot zdvihu.

Poslední poskytovanou funkcí aplikace je zálohování dat. Po kliknutí na tlačítko „Zálohuj“ se vytvoří nový list, který bude pojmenován na základě aktuálního kalendářního data a přidá se za pracovní list „Vzor DB“. Záleží na uživateli, jestli součástí archivačních dat budou i vygenerované grafy zátěžových diagramů. Aplikace je vytvořena pro dlouhodobé archivační účely a počítá tedy s maximálně jednou zálohou na den. Pokud by chtěl uživatel během jednoho dne vytvořit více různých záloh, musel by přistoupit k manuálnímu přejmenování generovaných listů. Pokud je již pro aktuální den záloha vytvořena a uživatel si přeje zálohu přepsat, jednoduše opětovně zálohuje, aplikace oznámí, že je daná záloha již přítomna a že bude přepsána.

final aplikace\_1 - Microsoft Excel

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1															
2															
3															
4															
5															
6															
7															
8															
9															
10															
11															
12															
13															
14															
15															
16															
17															
18															
19															
20															
21															
22															
23															
24															
25															
26															
27															
28															
29															
30															
31															
32															
33															
34															
35															
36															
37															
38															
39															
40															
41															
42															
43															
44															
45															
46															
47															
48															
49															

Obr. 31 Pracovní list „Přenos“ – ovládací panel a tabulky hodnot



Obr. 32 Pracovní list „Přenos“ – ukázka grafu zatěžovacího diagramu

## 6 Závěr

V rámci této bakalářské práce jsem ve spolupráci s firmou SIMO-CZ s.r.o. řešil problematiku spojenou s monitorováním provozu mostového jeřábu. Mým úkolem bylo vytvořit v programátorském prostředí Step 7 od firmy Siemens AG samostatný funkční blok, který zajišťuje výpočet provozních veličin na základě vstupních hodnot definujících provozní podmínky mostového jeřábu. Monitorovací funkční blok spolu s datovým blokem jsem úspěšně navrhl.

Dalším cílem práce bylo najít vhodný způsob, na základě kterého by byl realizovatelný obousměrný přenos dat mezi PLC a PC. Nalezení takového způsobu znamenalo projít velké množství cizojazyčných fór s danou tématikou. Nejvíce důvěryhodných informací bylo v němčině, v mateřském jazyce výrobce. Zdokumentoval jsem všechny nalezené vhodné způsoby a vyzkoušel ty, které jsem doporučil k realizaci přenosu.

Pro realizaci aplikace umožňující obousměrný přenos dat mezi počítačem a programovatelnou řídicí jednotkou jsem zvolil knihovnu libnodave.dll. Tato knihovna se ukázala být velmi vhodným řešením pro splnění všech požadavků firmy SIMO-CZ s.r.o. na přenos dat.

Aplikaci jsem testoval na procesoru 315-2DP a ethernetovém modulu CP 343-1 IT. Testy navázání komunikace a přenosu dat přes ethernet proběhly úspěšně. Měl jsem dále k dispozici adaptér CP5611 pro připojení k procesoru 315-2DP přes MPI sběrnici. Testoval jsem tedy nejnovější přenosovou funkci knihovny libnodave openS7online. Aplikace se úspěšně připojí k PLC, avšak přenos dat zatím nefunguje.

Knihovnu libnodave.dll bych doporučil k dalšímu průzkumu jejího využití. Možnosti použití knihovny jsem rádně zdokumentoval a doplnil ukázkami kódu z implementované aplikace. Splnil jsem všechny zadané požadavky firmy SIMO-CZ s.r.o. týkající se funkcionality aplikace vytvořené v MS Excel.

## 7 Literatura

1. ČSN ISO 12482-1 - Náhled:. Česká technická norma. [Online] <http://www.cni-normy.cz/hledat.php>.
2. Siemens - Industry Automation and Drive Technologies - Service& Support - Automation Service, Automation Support, Simatic Service, Simatic Support, Technical Support, Technical Consulting:. *Siemens*. [Online] Siemens AG, 6. 10 2006.  
<http://support.automation.siemens.com/WW/lbisapi.dll?func=cslib.csinfo&objId=837001&objAction=csOpen&ehbid=24352635&query=import+data+block+to+excel&lang=en&siteid=cseus&aktprim=0&extranet=standard&viewreg=WW>.
3. Siemens - Industry Automation and Drive Technologies - Service& Support - Automation Service, Automation Support, Simatic Service, Simatic Support, Technical Support, Technical Consulting:. *Siemens*. [Online] Siemens AG, 3. 7 2008.  
<http://support.automation.siemens.com/WW/lbisapi.dll/17853532?func=ll&objId=24675522&objAction=csView&lang=en&siteid=cseus&aktprim=0&extranet=standard&viewreg=WW&load=treecontent>.
4. **Hergenhahn, Thomas.** LIBNODAVE, a free communication library for Simatic S7 PLCs. *LIBNODAVE -- Exchange data with Siemens PLCs*. [Online] 10. 5 2010.  
<http://libnodave.sourceforge.net/>.
5. Siemens - Industry Automation and Drive Technologies - Service& Support - Automation Service, Automation Support, Simatic Service, Simatic Support, Technical Support, Technical Consulting:. *Siemens*. [Online] Siemens AG, 6. 7 2009.  
<http://support.automation.siemens.com/WW/lbisapi.dll?query=OC+Wizard&func=cslib.cssearch&content=adsearch%2Fadsearch.aspx&lang=en&siteid=cseus&objaction=cssearch&searchinprim=0&nodeid0=27103175&x=0&y=0>.
6. Elekt Labs s.r.o. [Online] Elekt Labs s.r.o.  
<http://www.elektlabs.cz/products/s7backup/index.en.html>.
7. INDI.AN. [Online] Indi.An GmbH & Co. KG, 2009. <http://www.indi-an.com/>.
8. EasyLog - pro zápis dat z Simatic S7-200/-300/-400 do Excelu. *FOXON*. [Online] FOXON s.r.o. [http://www.foxon.cz/easylog-pro-zapis-dat-z-simatic-s7200300400-do-excelu-p-278.html?cPath=142\\_185\\_221](http://www.foxon.cz/easylog-pro-zapis-dat-z-simatic-s7200300400-do-excelu-p-278.html?cPath=142_185_221).
9. AGLink S7 komunikační .DLL. *FOXON*. [Online] FOXON s.r.o. [http://www.foxon.cz/s7-software-aglink-s7-komunikacni-dll-c-142\\_185\\_9.html](http://www.foxon.cz/s7-software-aglink-s7-komunikacni-dll-c-142_185_9.html).

10. ACCON S7 PG, editor S7-300/-400 AWL - software licence. *FOXON*. [Online] FOXON s.r.o. <http://www.foxon.cz/accon-s7-pg-editor-s7300400-awl-software-liscence-p-170.html?zenid=f1ac01a71a1112c517346e9022f6eb04>.
11. eWON 500/4001 - serial <=> Ethernet gateway. *FOXON*. [Online] FOXON s.r.o. [http://www.foxon.cz/ewon-5004001-serial-ethernet-gateway-p-801.html?cPath=339\\_340&zenid=f1ac01a71a1112c517346e9022f6eb04](http://www.foxon.cz/ewon-5004001-serial-ethernet-gateway-p-801.html?cPath=339_340&zenid=f1ac01a71a1112c517346e9022f6eb04).
12. S7 DB TAGS generator for S7-300/-400 PLCs. *Runmode programming Siemens PLCs*. [Online] Runmode. <http://www.runmode.com/s7dbtags.html>.
13. PC-S7-LINK - DLL/Lib für Kommunikation PC über PC/MPI-Adapter zur SIMATIC S7 . *Traeger industry components*. [Online] Traeger industry components GmbH. <http://www.traeger.de/softwaretreiber-simatic-s7/pc-s7-link-dll-lib-fuer-kommunikation-pc-ueber-pc-mpi-adapter-zur-simatic-s7.html>.
14. IP-S7-LINK. *Traeger industry components*. [Online] Traeger industry components GmbH. <http://www.traeger.de/softwaretreiber-simatic-s7/ip-s7-link.html>.
15. **Berger, Hans.** *Automating with STEP 7 in STL and SCL*. Erlangen : Publicis Corporate Publishing, 2005. ISBN 3-89578-243-2.
16. **Berger, Hans.** *Automating with SIMATIC*. Erlangen : Publicis Corporate Publishing, 2003. ISBN 3-89578-223-8.
17. GNU. *gnu free documentation license*. [Online] [http://gl.pandapedia.com/wiki/gnu\\_free\\_documentation\\_license](http://gl.pandapedia.com/wiki/gnu_free_documentation_license).
18. SPS-Forum - Powered by vBulletin:. *SPS-Forum.de -> Wissen ist das einzige Gut das sich vermehrt wenn man es teilt!* [Online] Jelsoft Enterprises Ltd. <http://sps-foren.de/>.
19. **Černý, Jaroslav.** *Excel 2000-2007 záznam, úprava a programování maker - 2. aktualizované vydání*. Praha : Grada Publishing, 2008. ISBN 978-80-247-2305-1.