

# Aplikácia DDE a OPC komunikačných protokolov v identifikácii a riadení reálneho systému

Štefan JAJČIŠIN, Anna JADLOVSKÁ

Katedra kybernetiky a umelej inteligencie, FEI TU v Košiciach, Slovenská republika

stefan.jajcisin@tuke.sk, anna.jadlovaska@tuke.sk

**Abstrakt** — Článok sa zaoberá opisom reálneho hydraulického systému z hľadiska technického a programového vybavenia, spôsobom jeho ovládania, ako aj komunikáciou s počítačom, na ktorom je spustený algoritmus identifikácie alebo riadenia v prostredí Matlab. V rámci spojenia reálneho systému s počítačom je bližšie rozobraný spôsob vytvorenia a realizácie komunikácie s využitím DDE a OPC protokolov. V rámci článku sú uvedené aj zjednodušené vývojové diagramy základných programov, ktoré sme vytvorili v prostredí Matlab za účelom systémovej identifikácie a riadenia reálneho hydraulického systému, ktoré využívajú DDE a OPC protokoly.

**KLúčové slová** — komunikačný protokol DDE, komunikačný protokol OPC, PLC automat, reálny hydraulický systém, simulačný jazyk Matlab.

## I. ÚVOD

V článku sa zaoberáme dvoma rôznymi spôsobmi komunikácie reálneho systému s Matlabom, a to pomocou DDE a OPC protokolov.

V prvej časti sa venujeme opisu samotného reálneho systému. Uvádžeme technické vybavenie systému a vzájomné prepojenia jednotlivých častí, ale aj programovú výbavu, ktorá je súčasťou reálneho systému. V ďalšej časti sa zaoberáme komunikáciou reálneho systému s počítačom, na ktorom je spustený identifikačný alebo riadiaci algoritmus vytvorený v prostredí Matlab. Zaoberáme sa základným postupom pri vytváraní a udržiavaní komunikácie prostredníctvom DDE a OPC protokolov. V rámci článku sa tiež venujeme implementácii vytvorených komunikačných programov pri identifikácii a riadení reálneho hydraulického systému.

Súčasťou článku je vývojový diagram programu pre zber potrebných dát pri identifikácii, ako aj vývojový diagram programu pri uskutočňovaní riadenia hydraulického systému pomocou PSD algoritmov.

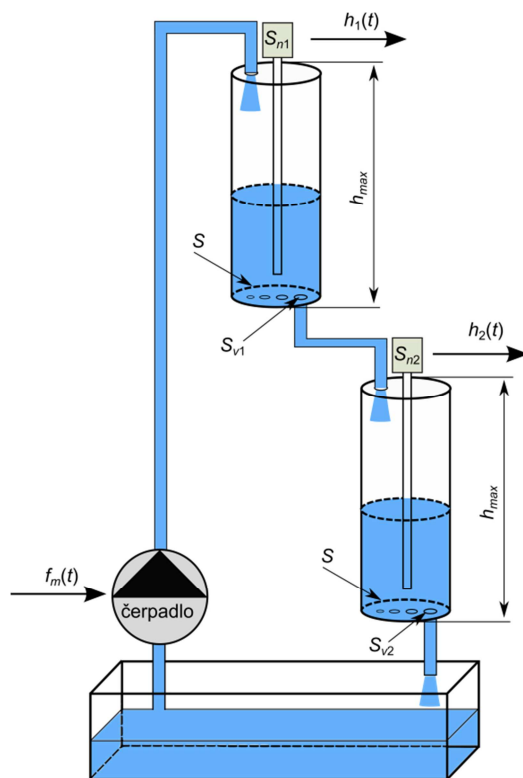
## II. REÁLNY MODEL HYDRAULICKÉHO SYSTÉMU

Reálny model hydraulického systému, ktorý vlastní Katedra kybernetiky a umelej inteligencie je tvorený dvoma valcovými nádobami umiestnenými pod sebou. Kvapalina, v našom prípade voda, je do prvej nádoby privádzaná prostredníctvom čerpadla, ktoré ju vytláča z vaničky cez prírodnú trubicu do vstupného otvoru prvej nádoby. Nádoby sú prepojené hadičkou, ktorou kvapalina vytekajúca z výstupného otvoru v prvej nádoby preteká cez vstupný otvor do druhej nádoby. Z výstupného otvoru druhej nádoby kvapalina vyteká pod vplyvom hydrostatického tlaku späť do vaničky. Veľkosť prierezu výstupných otvorov je možné vybrať zo štyroch rôznych hodnôt pomocou škrtiacich skrutiek.

Tento reálny systém predstavuje hydraulický systém dvoch nádob bez vzájomnej interakcie.

Schematické znázornenie fyzikálnej podstaty reálneho hydraulického systému je na Obr. 1, pričom jednotlivé symboly majú nasledujúci význam:

- $S$  - prierez nádob,
- $S_{v1}, S_{v2}$  - prierez výstupných ventilov prvej, resp. druhej nádoby,
- $h_{max}$  - výška nádob (maximálna výška hladiny),
- $S_{n1}, S_{n2}$  - snímače merajúce aktuálnu výšku hladiny v oboch nádobách,
- $f_m(t)$  - frekvencia motora čerpadla generovaná frekvenčným meničom,
- $h_1(t), h_2(t)$  - aktuálna výška hladiny kvapaliny v nádobách.



Obr. 1 Hydraulický systém – fyzikálny princíp

#### A. Technické vybavenie

Z hľadiska technického vybavenia sú, okrem už spomenutých častí, súčasťou tohto systému aj:

- membránové čerpadlo *Spring* [1];
- frekvenčný menič *Allen-Bradley PowerFlex 40* [2], ktorý na základe hodnoty vstupného napätia v rozsahu 0 – 10V mení frekvenciu napätia motora čerpadla v definovanom rozsahu;
- programovateľný logický automat (PLC) *Allen-Bradley Micrologix 1200* so vstupno-výstupnou analógovou kartou [3];
- kapacitné snímače *Dinel CLM-36* [4], ktoré sú umiestnené v oboch nádobách.

#### B. Programové vybavenie

Programové vybavenie, ktoré je súčasťou reálneho hydraulického systému je softvér v rámci frekvenčného meniča, ktorým sa nastavujú jednotlivé parametre, napríklad:

- doba nábehu motora na požadovanú frekvenciu,
- minimálna a maximálna frekvencia motora,
- parameter určujúci, ktorý zo vstupov frekvenčného meniča má byť aktívny (prúdový, napäťový, potenciometer, atď.).

Podrobné informácie o jednotlivých parametroch a ich nastavovaní sú uvedené v [2].

Za programové vybavenie reálneho systému sa dá tiež považovať aj program spustený v PLC, keďže naším cieľom je návrh riadiacich algoritmov, ktoré budú fungovať v Matlabe. Avšak program spustený v PLC, ktorý zabezpečuje zápis na výstupy PLC, načítavanie príslušných údajov zo vstupov PLC a ošetrenie havarijných stavov sme museli taktiež navrhnuť a naprogramovať pomocou rebríkových schém v prostredí RSLogix 500 [5].

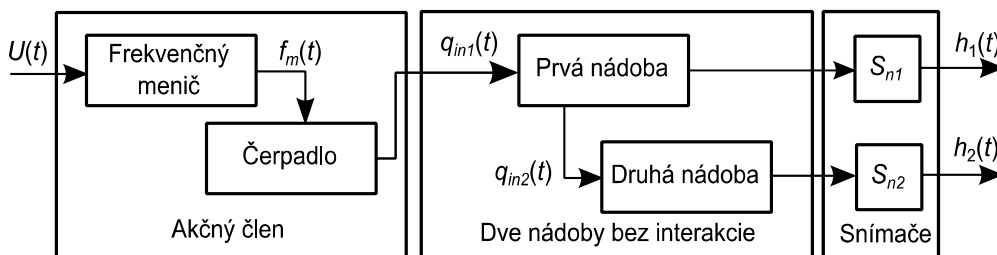
#### C. Princíp ovládania reálneho hydraulického systému

Z hľadiska systémového prístupu predstavuje frekvencia motora čerpadla  $f_m(t)$  vstup do systému. Frekvenciu  $f_m(t)$  je možné meniť rôznymi spôsobmi, napr. prúdovým signálom v rozsahu 4 – 20mA, napäťovým signálom v rozsahu 0 – 10V, pomocou potenciometra na paneli meniča alebo ďalšími inými spôsobmi. Ktorý z uvedených spôsobov sa má použiť je potrebné nastaviť vo frekvenčnom meniči. V našom prípade využívame napäťový vstup, na ktorý privádzame napätie  $U(t)$  generované v rozsahu 0 – 10V PLC automatom prostredníctvom výstupov vstupno-výstupnej analógovej karty.

Dynamiku hydraulického systému predstavuje meniaci sa výška hladiny kvapaliny v oboch nádobách  $\dot{h}_1(t)$  a  $\dot{h}_2(t)$ .

Výstupmi systému sú aktuálne výšky hladín kvapaliny v oboch nádobách  $h_1(t)$  a  $h_2(t)$ , ktoré sú snímané kapacitnými snímačmi  $S_{n1}$  a  $S_{n2}$ . Údaje namerané snímačmi sú prostredníctvom vstupov analógovej karty spracovávané v PLC, t.j. preškáľované a prostredníctvom komunikačného kanála odoslané do programu spustenom v Matlabe na osobitnom počítači. Po spracovaní týchto údajov je v Matlabe na základe príslušného algoritmu vygenerovaný vstup (akčný zásah) hydraulického systému, ktorý je opäť odoslaný do PLC, kde sa údaje znova preškáľujú a odošlú prostredníctvom analógovej karty na vstup frekvenčného meniča.

Systémový rozbor hydraulického systému je na Obr. 2, kde okrem už spomenutých veličín je  $q_{in1}(t)$ , resp.  $q_{in2}(t)$  prítok do prvej, resp. druhej nádoby.

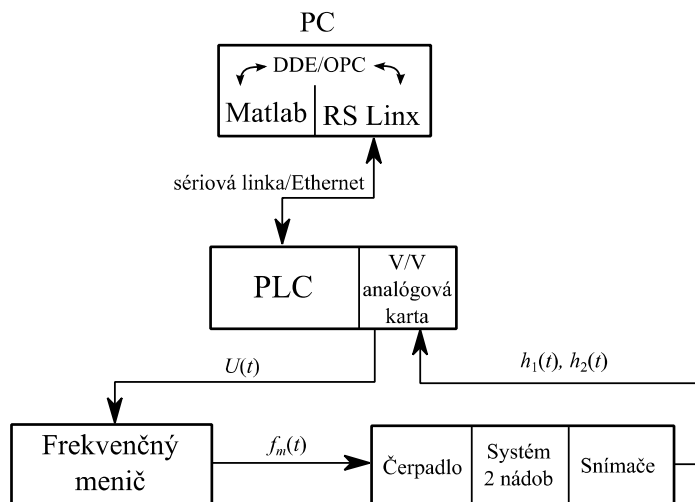


Obr. 2 Hydraulický systém – systémový pohľad s jedným vstupom a dvomi výstupmi

### III. KOMUNIKAČNÉ PROTOKOLY DDE A OPC

Komunikáciu medzi PLC a počítačom zabezpečuje v našom prípade softvérový produkt firmy Rockwell Automation RSLinx [6], ktorý je potrebné do počítača vopred nainštalovať. RSLinx vykonáva úlohu DDE, resp. OPC servera, pričom DDE, resp. OPC klientom je Matlab. Takáto konštelácia nám umožňuje pristupovať k jednotlivým premenným v PLC pomocou DDE a OPC nástrojov priamo z Matlabu.

Na Obr. 3 je znázornená kompletná štruktúra reálneho hydraulického systému s pripojením na počítač. V ďalšom sa zameriame na stručný opis a spôsob uskutočnenia komunikácie medzi RSLinx-om a Matlabom na báze DDE a OPC protokolov.



Obr. 3 Kompletná štruktúra reálneho hydraulického systému s pripojením na počítač

#### A. DDE protokol

DDE (*Dynamic Data Exchange*) je komunikačný protokol založený na architektúre typu *klient - server*. Protokol bol vytvorený firmou *Microsoft* a umožňuje výmenu údajov medzi aplikáciami spustenými v operačnom systéme *Windows*. Aplikácia podporujúca DDE protokol môže slúžiť ako server pre viac klientov, klient pre viac serverov alebo byť zároveň klientom aj serverom.

Základné prvky, ktoré musia byť v procese komunikácie známe sú:

1. *Service* - meno aplikácie DDE servera (väčšinou názov aplikácie alebo služby),
2. *Topic* - téma rozhovoru (zvyčajne názov súboru, cesta),
3. *Item* - identifikátor údajov, napr. bunka (rozsah buniek) tabuľky, názov premennej.

Je potrebné zdôrazniť, že názvy pre *Service*, *Topic* a *Item* závisia od komunikujúcich aplikácií a ich presné označenie si je vždy potrebné overiť.

DDE protokol umožňuje uskutočniť tri typy komunikácie medzi DDE serverom a klientom:

1. *Hot link* - pri zmene hodnoty údajov dochádza k ich automatickému prenosu,
2. *Warm link* - pri zmene hodnoty údajov je najprv serverom odoslaná informácia o tom, že došlo k zmene, prenos údajov sa vykoná až na základe potvrdenia prenosu klientom,
3. *Cold link* - prenos údajov sa vykonáva iba na základe požiadavky klienta.

#### B. OPC protokol

OPC (*OLE for Process Control*) protokol je založený na COM (*Component Object Model*) technológii, kde komponenty, ako sú aplikácie, dátové objekty, ovládacie prvky, služby a pod. môžu navzájom komunikovať ako objekty. Taktiež má architektúru typu *klient – server*.

OPC je štandardný protokol pri komunikácii rôznych dátových zdrojov s klientskymi aplikáciami riadiacich systémov.

Na úrovni OPC servera sú dáta definované v skupinách (*Group*), ktoré obsahujú jednotlivé položky (*Item*). Položky, podobne ako v DDE komunikácii, predstavujú konkrétne bunky alebo premenné dátových zdrojov. Skupiny a ich položky môžu byť verejné alebo lokálne. Verejné môžu byť zdieľané viacerými klientmi, pričom lokálne sú určené iba pre konkrétneho klienta. V každej skupine musí byť definovaná minimálne jedna položka. Dôležité si je uvedomiť, že položky nie sú dátovými zdrojmi, iba ich reprezentujú, adresujú.

Okrem toho, OPC špecifikácia dovoľuje uskutočňovať komunikáciu aj na základe vzniknutých udalostí alebo výnimiek, nielen v presne stanovených intervaloch [7].

### IV. APLIKÁCIA KOMUNIKAČNÝCH PROTOKOLOV V IDENTIFIKÁCIÍ A RIADENÍ REÁLNEHO SYSTÉMU

V tejto časti bližšie popíšeme postup vytvorenia a uskutočňovania komunikácie medzi RSLinx-om (server) a Matlabom (klient). V oboch prípadoch, či už s použitím DDE alebo OPC protokolu, pozostáva komunikácia z troch základných krokov:

1. vytvorenie komunikačného kanála, resp. klienta,
2. čítanie/zápis údajov cez komunikačný kanál,
3. zrušenie komunikačného kanála.

V RSLinx-e nie je potrebné nastavovať žiadne parametre, automaticky plní úlohu DDE, resp. OPC servera. Za zmienku snáď stojí možnosť overenia aktívnych tém komunikácie (*topic-ov*) prostredníctvom položky *DDE/OPC* v hlavnom menu okna RSLinx-u. Čo sa týka nastavení Matlabu, uvádzame syntax príkazov, ktoré umožňujú realizovať vyššie uvedené kroky. V prípade OPC komunikácie je potrebné mať nainštalovaný *OPC Toolbox* v prostredí Matlab.

#### A. DDE komunikácia

##### 1. Vytvorenie komunikačného kanála

```
service = 'rslinx';  
topic = 'riadenie'; (Topic musí byť zhodný s názvom súboru, programu, ktorý je nahraný  
v PLC, teda s ktorým je nadviazaná komunikácia.)
```

```
kanal = ddeinit(service,topic);
```

##### 2. Načítanie hodnoty premennej PLC na adrese F8:1 do premennej Matlabu h2

```
h2_adr = 'F8:1';  
h2 = ddereq(kanal,h2_adr);
```

##### Zápis hodnoty premennej Matlabu u do premennej PLC na adrese F8:2

```
u_adr = 'F8:2';  
u = 5;  
b = ddepoke(kanal,u_adr,u); (Premenná b nadobudne hodnotu 0/1 podľa toho, či bol zápis  
neúspešný/úspešný.)
```

##### 3. Zrušenie komunikačného kanála

```
b = ddeterm(kanal);
```

#### B. OPC komunikácia

##### 1. Vytvorenie OPC klienta s príslušnými skupinami pre vstupy, výstupy a ich položkami

```
opc_client = opcda('localhost', 'RSLinx OPC Server');  
gInputs = addgroup(opc_client, 'Inputs');  
gOutputs = addgroup(opc_client, 'Outputs');
```

```

itemI = additem(gInputs,{'[riadenie]F8:2'}); (V hranatých zátvorkách je názov súboru,
      programu, ktorý je nahraný v PLC, teda s ktorým je nadviazaná komunikácia.)
itemO = additem(gOutputs,{'[riadenie]F8:0','[riadenie]F8:1'});

```

Okrem zadenovania skupín a ich položiek je potrebné nastaviť aj ich dátové typy, napr. `set(itemO(1),'DataType','single');`  
 Taktiež je možné nastaviť aj rôzne parametre skupín a ich položiek. Okrem iného napr. `set(gInputs,'UpdateRate',1);`  
`set(itemO(1),'Tag','h1');`  
 Detailnejší popis a možnosti nastavenia jednotlivých parametrov skupín a ich položiek je uvedený v [8].

### Nadviazanie komunikácie

```
connect(opc_client);
```

### 2. Načítanie hodnoty premennej PLC na adrese F8:1 do premennej Matlabu h2

```
h2_struct = read(itemO(2));
```

```
h2 = h2_struct.Value;
```

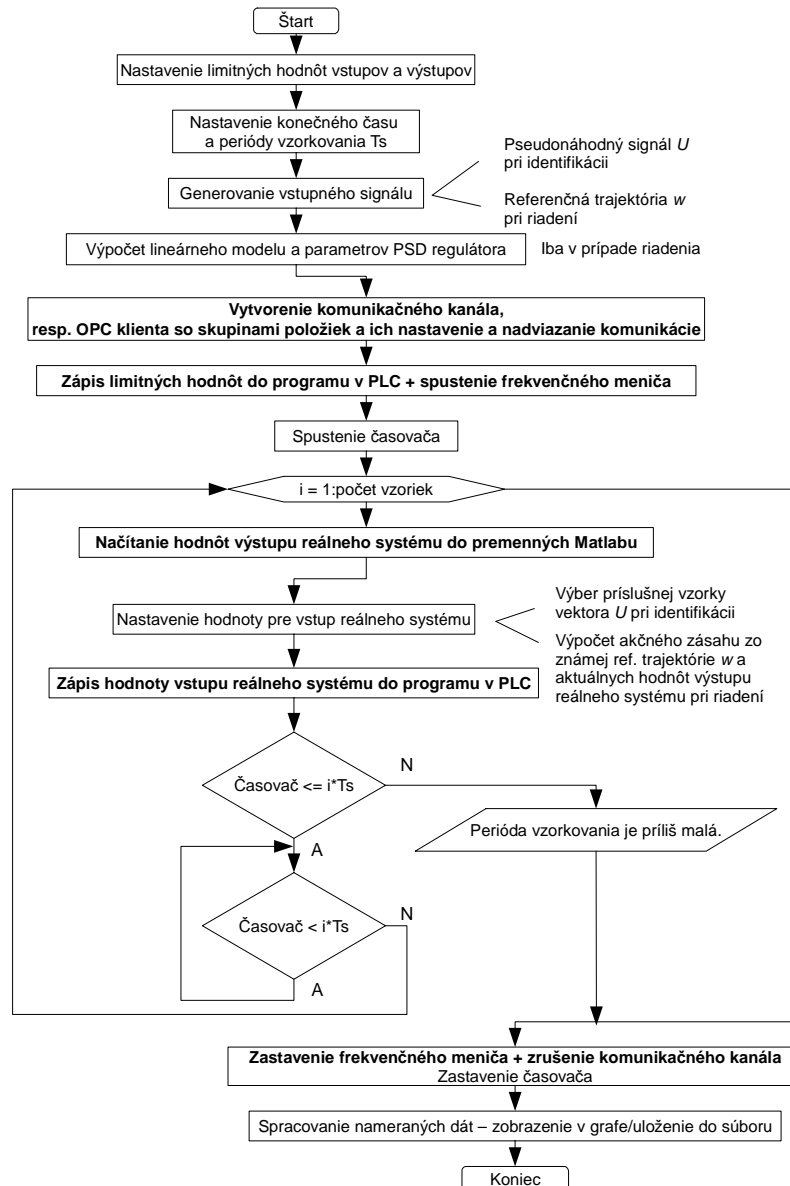
### Zápis hodnoty premennej Matlabu u do premennej PLC na adrese F8:2

```
u = 5;
```

```
write(itemI(1),u);
```

### 3. Zrušenie komunikačného kanála

```
disconnect(opc_client);
```



Obr. 4 Vývojový diagram aplikácie DDE a OPC komunikačných protokolov v identifikácii a riadení reálneho systému

Uvedené príkazy jazyka Matlab sme aplikovali na všetky premenné programu v PLC, ktoré sú potrebné pre správnu funkčnosť reálneho hydraulického systému. Okrem vstupného napätia  $U$  a výstupných hodnôt  $h_1$  a  $h_2$  sú to aj dva bity nastavujúce povolenie a štart frekvenčného meniča, ako aj hodnoty maximálnej výšky hladiny v oboch nádobách.

Príkazy realizujúce komunikáciu sme potom implementovali do algoritmov identifikácie a riadenia reálneho systému v súlade s vývojovým diagramom na Obr. 4. Takto doplnené algoritmy umožňujú posielat' na vstup reálneho systému hodnotu napätia akčného zásahu, či už ako vstupný signál v prípade identifikácie alebo akčný zásah generovaný riadiacim algoritmom. Zároveň sú v rámci algoritmov načítavané hodnoty výstupných veličín hydraulického systému, ktoré v prípade riadiaceho algoritmu slúžia pre výpočet akčného zásahu v ďalšom kroku.

## V. ZHODNOTENIE

V článku sme prezentovali reálny hydraulický systém z pohľadu fyzikálnej štruktúry, technického a programového vybavenia, ktorý vlastní Katedra kybernetiky a umelej inteligencie. Uviedli sme stručný opis DDE a OPC komunikačných protokolov, ako aj ich použitie v komunikácii medzi počítačom a PLC ovládajúcim reálny hydraulický systém v rámci algoritmu identifikácie a riadenia reálneho systému.

Hoci realizácia komunikácie na báze DDE protokolu je jednoduchšia, v podstate vyžaduje znalosť a použitie štyroch jednoduchých príkazov jazyka Matlab, OPC toolbox predstavuje výhodnejší komunikačný prostriedok. Zložitejšie vytváranie OPC klienta a definovanie skupín s príslušnými položkami je zdanlivo nevýhoda, avšak v rámci OPC toolboxu existuje grafické používateľské prostredie uľahčujúce túto procedúru. Navyše tento prístup ponúka oveľa viac možností ako nastaviť vlastností jednotlivých položiek a komunikácie, okrem iného možnosť reagovať na udalosti a definované výnimky v rámci komunikácie. OPC toolbox Taktiež poskytuje nástroje pre kompletne vytvorenie a realizáciu komunikácie v prostredí Simulink, čo umožňuje vyhnúť sa písaniu akéhokoľvek zdrojového kódu v jazyku Matlab. Ďalšou výhodou OPC protokolu je jeho otvorenosť a fakt, že predstavuje medzinárodný štandard v komunikácii riadiacich systémov.

Z hľadiska časovej náročnosti počas prijímania a odosielania dát boli v našom prípade oba protokoly vyhovujúce. Je však potrebné zároveň podotknúť, že pri identifikácii a riadení reálneho hydraulického systému úplne postačuje perióda vzorkovania 1s. Zaujímavé by bolo porovnať tieto dva protokoly pri nasadení na systémy s rýchlo sa meniacou dynamikou, kde perióda vzorkovanie je rádovo ms.

## POĎAKOVANIE

Tento článok bol vytvorený realizáciou projektu Rozvoj Centra informačných a komunikačných technológií pre znalostné systémy (ITMS kód: 26220120030) na základe podpory operačného programu Výskum a vývoj financovaného z Európskeho fondu regionálneho rozvoja.

## LITERATÚRA

- [1] Bia-Verder, *SPRING membránové provedení*, technická dokumentace. Dostupné na internete: [http://bia-verder.verder.ams3 footsteps.nl/Cerpadla/Davkovaci\\_cerpadla/Mechanicka/Seko\\_SPRING\\_MS](http://bia-verder.verder.ams3 footsteps.nl/Cerpadla/Davkovaci_cerpadla/Mechanicka/Seko_SPRING_MS).
- [2] Rockwell Automation, *Allen-Bradley PowerFlex 40 Adjustable Frequency AC Drive*, User Manual. Dostupné na internete: <http://www.ab.com/support/abdrives/webupdate/literature.html>.
- [3] Rockwell Automation, *MicroLogix 1200 Programmable Controllers*, User Manual. Dostupné na internete: [http://literature.rockwellautomation.com/idc/groups/literature/documents/um/1762-um001\\_-en-p.pdf](http://literature.rockwellautomation.com/idc/groups/literature/documents/um/1762-um001_-en-p.pdf).
- [4] Dinel, *Kapacitní hladinoměry CLM-36*, Návod k použití. Dostupné na internete: <http://www.dinel.cz/uploads/pdf/090916081606-clm-36-nav-cz.pdf>.
- [5] Rockwell Automation, *RSLogix 500*, Product Profile. Dostupné na internete: [http://literature.rockwellautomation.com/idc/groups/literature/documents/pp/lg500-pp001\\_-en-e.pdf](http://literature.rockwellautomation.com/idc/groups/literature/documents/pp/lg500-pp001_-en-e.pdf).
- [6] Rockwell Automation, *RSLinx Classic*, Getting Results Guide. Dostupné na internete: [http://literature.rockwellautomation.com/idc/groups/literature/documents/gr/linx-gr001\\_-en-e.pdf](http://literature.rockwellautomation.com/idc/groups/literature/documents/gr/linx-gr001_-en-e.pdf).
- [7] Michal Blaho, „OPC – štandard priemyselnej komunikácie II“ v *Posterus.sk*, Ročník 2, číslo 5, ISSN 1338-0087.
- [8] Mathworks, *OPC Toolbox 2*, User Guide. Dostupné na internete: [http://www.mathworks.com/help/pdf\\_doc/opc/opc.pdf](http://www.mathworks.com/help/pdf_doc/opc/opc.pdf).