

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA STROJNÍ



BAKALÁŘSKÁ PRÁCE

2017

**Systemy SCADA a nástroje pro sběr,
vizualizaci a analýzu průmyslových dat**

Tomáš

Primus

Anotační list

Jméno autora: Tomáš Primus

Fakulta: Fakulta strojní

Zadávací ústav: Ústav přístrojové a řídicí techniky

Studijní program: Teoretický základ strojního inženýrství

Studijní obor: bez oboru

Rok: 2017

Název bakalářské práce: Systémy SCADA a nástroje pro sběr, vizualizaci a analýzu průmyslových dat

Název bakalářské práce anglicky: SCADA Systems for acquisition, visualisation, and analysis of industrial process data

Vedoucí práce: doc. Ing. Ivo Bukovský, Ph.D.

Bibliografické údaje: Počet stran: 55

Počet obrázků: 24

Klíčová slova: SCADA, HMI, PLC, Python, komunikace, mySCADA

Key words: SCADA, HMI, PLC, Python, communication, mySCADA

Anotace: Práce se zabývá popisem systémů SCADA, ukázáním jejich možností a popsáním jednotlivých prvků systémů. Dále práce obsahuje rešerši hlavních výrobců SCADA ovlivňujících český trh. V druhé části je napsáno ukázáno navázání komunikace mezi SCADA a PLC. A poslední část této práce je o využití programovacího jazyka Python pro analýzu dat v systému mySCADA

Abstract: The bachelor thesis describes SCADA SW systems and the potentials of their use. It describes individual elements in SCADA hierarchy. Further, this thesis reviews main SCADA systems being used, and some also developed, in Czech Republic. The second part of this thesis is about the method of

communication between particular system mySCADA and PLC. At the end, the Python use data analysis within mySCADA system is described and fundamental data analysis is demonstrated on data obtained in a laboratory with mySCADA system.

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně s tím, že její výsledky mohou být dále použity podle uvážení vedoucího bakalářské práce pana doc. Ing. Iva Bukovského Ph.D. Souhlasím také s případnou publikací výsledků bakalářské práce nebo její podstatné části, pokud budu uveden jako její spoluautor.

Dne.....

Podpis ...

Poděkování:

Děkuji panu doc. Ing. Ivovi Bukovskému Ph.D. za pomoc při vedení bakalářské práce. Mé poděkování patří též všem z mySCADA za spolupráci při získávání údajů pro praktickou část práce.

Obsah

1	Úvod	8
2	Nástroje pro vizualizaci dat v průmyslu – systémy SCADA	10
3	Nástroje pro sběr dat, prvky SCADA systémů	12
3.1	Vstupně-výstupní hardware	12
3.2	HMI	12
3.3	Komunikace	12
3.3.1	Průmyslové sběrnice	13
3.3.2	Komunikační protokoly	14
3.3.3	Typy počítačových komunikací	14
3.4	OPC server	15
3.5	RTU (Remote terminal units)	16
3.6	PLC	16
3.7	Bezpečnost SCADA systémů a způsoby útoků	18
4	Výrobci SCADA software na českém trhu	21
4.1	InTouch software	21
4.2	Control Web	22
4.3	Simatic WinCC	22
4.4	Iconics	24
4.5	TIRS.NET	24
4.6	Promotic	25
4.7	Reliance	26
4.7.1	Nástroje pro vizualizaci v Reliance	27
4.7.2	Nástroje pro analýzu dat v Reliance	28
4.8	mySCADA	31
4.8.1	Nástroje pro vizualizaci dat v mySCADA	31
4.8.2	Nástroje pro analýzu dat v mySCADA	32
5	Praktické ověření a popis komunikace řídicího systému PLC a mySCADA	33
5.1	Použitý software a hardware	33
5.2	Popis komunikace	33
6	Využití Python v mySCADA	41
7	Závěr	48
8	Zdroje	50

Seznam použitých zkratk

SCADA – Supervisory control and data acquisition

PLC – Programmable logic controller

RTU – Remote terminal unit

HMI- Human -machine interface

1 Úvod

Průmysl po celém světě prochází čtvrtou průmyslovou revolucí a stále více se klade nárok na zautomatizování procesů. Lidé jsou nahrazováni stroji a kontrolu nad výrobním procesem přebírají počítače. Proto je stále více a více důležitější vizualizace, sběr dat a jejich vyhodnocování, aby bylo možné převzít kontrolu nad řízením. K tomuto účelu byl navržen software zvaný SCADA [1], který informace nejen shromažďuje, ale také podporuje vzdálenou regulaci a řízení, a měl by prostupovat až do řízení celého podniku. Modernizace všech odvětví umožňuje aplikaci SCADA i do nevýrobních procesů například do řízení a regulace rodinných domů, elektráren, nádraží a dalších.

Tato práce se zabývá popisem SCADA software, jeho prvky a způsobem komunikace mezi těmito prvky. Cílem bylo vytvořit rešerši hlavních SCADA systémů na českém trhu v oboru průmyslové vizualizace a vzdáleného dozorování v průmyslových aplikacích, ale i v běžném použití, a seznámit se s principem fungování těchto systémů se zaměřením na analýzu procesních dat. Je zde popsáno, co firmy nabízejí a jaká je jejich dostupnost. Jelikož se problematikou SCADA zabývá čím dál tím více firem jsou v této práci popsány jen základní a nejhlavnější výrobci. Více se pak tato práce zabývá softwarem od firmy mySCADA a Reliance. Tito výrobci nezaujímají majoritní postavení na trhu, ale jejich SW je cenově dostupný a mají velký potenciál.

Vlastní práci autora a cíle, které byly stanoveny, jsou navázání komunikace mezi PLC a programy od mySCADA a popsání komunikačního protokolu. Komunikace byla prakticky ověřena a vyzkoušena na jednoduché simulaci výrobního procesu. Jedná se o hydraulický podavač, který je řízen přes PLC Siemens S7 200. Data jsou přes Ethernet posílána a vizualizována runtime moduly mySCADA. Tento příklad slouží jako názorná simulace výrobního procesu v průmyslu. Dále pak bylo týmem ČVUT [2] vytvořeno rozhraní v Pythonu, které dokáže nahradit webovou aplikaci, díky tomu, že umí číst komunikaci mezi serverem a webovým prohlížečem. Proto dokáže zobrazit data a už přímo v Pythonu se s nimi dá pracovat. Tento modul se pomalu aplikuje do programů

mySCADA. V dnešní době je vytvořeno rozšíření pro Python přímo v okně designer aplikace, kde se může psát vlastní Python skript, a tedy je možnost i vlastního programování aplikace. Cílem v této práci bylo ukázat a popsat způsob, jakým Python v mySCADA funguje a komunikace, která zde probíhá. Vše je ověřeno na příkladu a ukázce jednoduché vizualizace, sběru dat a jejich analýze s grafickým výstupem. Programovací jazyk Python byl zvolen z důvodu volné licence a schopnosti rychlého vytváření aplikací. Ve srovnání například Node.js, což je aplikace pro spouštění webového JavaScriptu, kterou mySCADA používá, je vytváření aplikací v Pythonu mnohem jednodušší a rychlejší. Je zde i efektivnější vykreslování grafů a další matematické výpočty. Python oplývá také spoustou knihoven pro statistiku a matematickou analýzu. Při nedostatku knihoven, nebo když je zapotřebí přidat další funkce, je poměrně snadné si novou knihovnu vytvořit a přesně ji nadefinovat. Díky knihovnám a objektovému principu je Python ideální pro SW prototyping širokého spektra aplikací. Dále je také ceněn způsob, jakým Python pracuje s pamětí, díky tomu, že pro proměnné preferuje ukazatele v paměti, pokud alokování fyzické paměti není nutné. Dále se liší například od Matlabu tím, že neimportuje všechny knihovny a moduly pokud není potřeba.

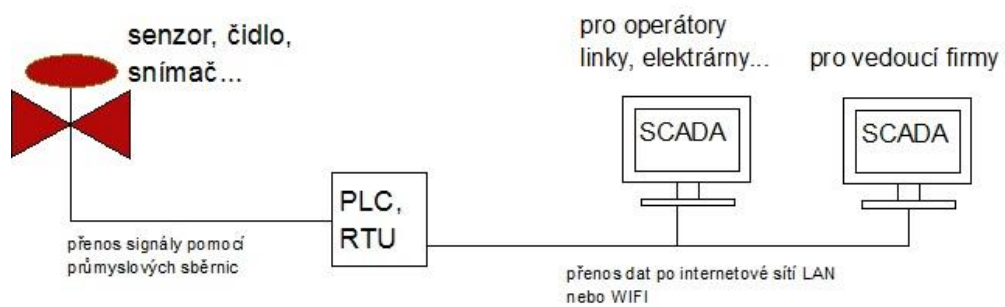
2 Nástroje pro vizualizaci dat v průmyslu – systémy SCADA

Vizualizace průmyslových dat je velmi důležitá. Umožňuje nám snadno vidět data v grafické podobě, a nejen spoustu čísel. Dříve byla tato vizualizace reprezentována pomocí velkých blikajících panelů dosahujících obřích rozměrů. V dnešní době jsou tyto panely nahrazovány SCADA systémy, které běží na jednom nebo více počítačích.

SCADA (Supervisory Control and Data Acquisition) [3], je do češtiny volně přeloženo jako Dispečerské řízení, sběr dat a jejich vizualizace. Jde o software používaný v průmyslové automatizaci. Jedná se o hromadný sběr technických dat (například z výrobní linky) a jejich vyhodnocování. Výsledky jsou pak zobrazeny přes webové aplikace ve webovém prohlížeči. Díky tomuto systému je online vidět vše, co se ve sledovaném zařízení děje, jaké hodnoty jsou aktuálně nastaveny a jaké jsou aktuální, navíc také možné dálkové řízení tohoto zařízení nebo objektu. Kromě možnosti dálkového řízení a nastavování vstupních parametrů, je možné regulovat například otáčky turbíny, nebo rychlost výroby na výrobní lince. Mezi další možnosti patří zjišťování trendů ve výrobě a určování nutnosti seřízení výrobních strojů, plánování výroby apod.

Samotné fungování SCADA systému začíná od sběru dat pomocí vstupních objektů, tj. senzorů a snímačů. Mohou být, jak analogové, tak binární. Zajišťují pro automatizaci procesu vstupní informace. Jsou umístěny přímo na námi zkoumaném mechanismu a přenášejí nám aktuální data z procesu. Ty jsou sbírána pomocí PLC nebo RTU jednotek. Tyto programovatelné jednotky předávají data dále, a to už samotnému SCADA softwaru. Ten se může dále dělit na operátorské stanoviště a dále na vizualizační stránku pro management firmy. Data jsou dále posílána do datových úložišť, nejčastěji do databází, odkud mohou být zpětně získána. [3–5] Pro názornou ukázkou je přiložen obrázek hierarchie systémů (Obrázek 1)

Hierarchie systémů SCADA



Obrázek 1-Hierarchie systémů SCADA

3 Nástroje pro sběr dat, prvky SCADA systémů

Mezi základní prvky SCADA obecně patří vstupně-výstupní hardware, regulátory, HMI, komunikace (sítě), software a databáze.

3.1 Vstupně-výstupní hardware

Pojmem vstupně-výstupní hardware se rozumí zařízení, která odesílají informace dovnitř do systému. Jedná se tedy o senzory, snímače, čidla, měřidla a o hardware podobného charakteru. V opačném směru jsou výstupní zařízení, která naopak předávají informace z řídicích počítačů do ovládacích prvků procesu.

3.2 HMI

HMI-human machine interface, česky přeloženo – rozhraní člověk/stroj. V dnešní době velmi často diskutované a s průmyslem 4.0 pevně spjaté spojení. HMI představuje rozhraní pro dva druhy aplikací-samostatně stojící a integrované. Samostatně stojící aplikace jsou ty, které jsou přímo na strojích nebo na menších výrobních linkách, na rozdíl od aplikací integrovaných, které jsou spojeny s řídicím nebo hostitelským počítačem a HMI softwarem. HMI aplikace mohou fungovat na bázi PC, panelového PC, statického terminálu, anebo na mobilním zařízení. Je to software zobrazující operátorovi informace o procesu a jeho stavu. Což je vlastně i principem SCADA systémů, proto se v literatuře často vyskytuje pojem SCADA/HMI dohromady. [5, 6]

3.3 Komunikace

Komunikace ve SCADA hierarchii je velmi složitá, a pro různé výrobce i odlišná. Může být rozdělena do dvou úrovní – fyzická vrstva, a komunikační vrstva. Na úrovni fyzické vrstvy jsou průmyslové sběrnice. Na úrovni senzorů a PLC (RTU) se ještě stále používají sběrnice RS-232, RS-485. [7] V případě RS-485 jde o kroucenou dvoulinku s vysokým dosahem až 1,2 km. RS-485 se vyrábí ve dvou variantách. První je již zmíněná dvojlinka, druhým typem jsou dvě dvojlinky. Pro první variantu je komunikace vedena v obou směrech, proto je nezbytné, aby vysílač signálu věděl, jakým směrem a kdy může vysílat. Pro druhý způsob, tj.

dvě dvojlínky je komunikace zajištěna v každém směru zvlášť. V případě RS 232 jde o dvojlínku reprezentovanou napětovými úrovněmi vzhledem k zemi tzn. pro každý signál je jeden vodič a jeden společný zemní vodič, vůči kterému jsou napětové úrovně vztaženy. Jelikož zapojení signálů vyžaduje společnou zem, dochází k omezení délky vodičů na max. 60 metrů. Toto spojení bylo vyvinuto hlavně pro komunikaci lokálních zařízení, což nám pro připojení PLC-snímače stačí. Tyto typy linek se také používají z důvodu nízké náchylnosti k rušení v porovnání např. s Ethernetem. Již zmíněný průmyslový Ethernet nachází svoji úlohu o úroveň výše ve SCADA hierarchii tzn. komunikace mezi PLC-PC a dále pak mezi databázemi a dalšími zařízeními v síti. Od vzniku standardu průmyslový Ethernet, bylo vyvinuto více variant řešení jeho fyzické vrstvy. Původně koaxiální kabely byly nahrazeny kroucenými páry s vyšší přenosovou rychlostí, až 1000 Mb/s. Téměř většina níže popsaných komunikačních protokolů využívá standardu Ethernet. [8, 9]

3.3.1 Průmyslové sběrnice

3.3.1.1 Profinet

Sběrnice PROFINET, založená na standardu Ethernet, použitelná pro všechny oblasti výroby a automatizace procesů. V běžné automatizaci komunikuje přes protokol TCP/IP. [10] Pro komunikaci v reálném čase je používán protokol RT s přenosovou rychlostí 10ms a pro nejnáročnější průmyslové aplikace probíhá komunikace přes protokol s označením IRT s rychlostí odezvy 1ms. [11]

3.3.1.2 Interbus

Interbus je průmyslová sběrnice, která slouží k propojení I/O zařízení (PLC) a field zařízení. Fyzické adresování je založeno na fyzickém umístění zařízení v síti, přenos dat je prováděn pomocí sériového přenosu bitů a je založen na přístupu ke sběrnici typu Master – slave. [12]

3.3.2 Komunikační protokoly

3.3.2.1 Modbus

Jde o komunikační protokol na úrovni aplikační vrstvy, je podporována většina komunikačních médií, ať už jde o sériové linky typu RS-232, RS-485 nebo jde o síť Ethernet s využitím komunikačního protokolu TCP/IP, o kterém je krátce napsáno později. Je zde také podporován přístup Master/slave. [13]

3.3.2.2 TCP/IP

Dalo by se říci, že je dnes nejpoužívanějším síťovým protokolem. Jeho největší zastoupení je v Internetu, a jelikož je stále více SCADA systémů připojených přes internet a ethernet, je třeba se o tomto protokolu lehce zmínit. Komunikace probíhá v několika vrstvách, kde každá vrstva má svůj úkol podle ISO/OSI modelu (Open System Interconnection-propojení otevřených systémů) [8]. Referenční model ISO/OSI je sedmivrstvý. První čtyři vrstvy modelu (fyzická, linková, síťová a transportní vrstva) jsou vrstvy zaměřené na vlastní komunikační síť a zbylé vrstvy (relační, prezentační a aplikační) jsou orientovány na jejich aplikaci. Za zmínku v této práci stojí síťová a transportní vrstva. Úkolem transportní vrstvy je spolehlivý přenos dat v dané kvalitě a přidávání zdrojových dat do hlaviček každého paketu. Úkolem síťové vrstvy je zajištění síťové adresace a správné nasměrování dat. Zkratka TCP/IP [10] znamená: IP (internet protocol) – protokol přijímá datové segmenty a přidává do nich svoji hlavičku a odešle data na adresy, které jsou cílové. IP protokol ale nekontroluje, zda data, která odeslal došla v pořádku. Proto je spolu s IP protokolem i protokol TCP (transmission control protocol), který kontroluje, zda odeslaná data došla v pořádku. [14]

3.3.3 Typy počítačových komunikací

3.3.3.1 Co je přístup Master – Slave

Jde o model počítačové komunikace, kde první zařízení (master) kontroluje jedno nebo více zařízení zapojených do procesu (slave). Ve SCADA systémech

je v roli master SCADA systém, který sbírá data, v roli slave je pak PLC, které data posílá. U mnoho systémů se ale tento druh přístupu nahrazuje architekturou klient-server, která je popsána níže v této práci.[15]

3.3.3.2 Peer to peer (P2P)

Jde o označení typu počítačové sítě, ve které jsou si všechna zařízení rovna. Mohou tak sloužit současně jako klienti i jako servery. Ve SCADA hierarchii jsou jako peer to peer zapojena PLC nebo RTU, o stupeň výše jsou v P2P zapojeny řídicí počítače. [16]

3.3.3.3 Klient – server

Na rozdíl od typu P2P je architektura sítě klient – server, jak už název říká, složena z nerovnocenných zařízení. Jde o vztah mezi dvěma zařízeními, z nichž jedno (klient) žádá a o služby druhý program (server). Jako příklad je architektura klient-server ve SCADA reprezentována OPC servery a OPC klienty. [17]

3.4 OPC server

Druhy OPC serverů jsou různé a liší se podle výrobců SCADA systémů.

OPC (Open proces control)- Jde o software, který běží na PC, a který díky svým hardwarovým nástrojům vytváří spojení a jednotné rozhraní pro výměnu dat mezi aplikacemi (klienty) a koncovými zařízeními prostřednictvím OPC serveru. Jde tedy o komunikační architekturu typu klient server, na rozdíl od průmyslových sběrnic, které jsou typu master-slave. Koncové zařízení např. PLC pak čte informace, a i zpětně je může zapisovat. PLC mu tyto informace poskytuje, pro OPC server v čitelné podobě. Dle OPC Foundation je samotné OPC rozdělené na několik skupin dalších specifikací. [18] Pro SCADA systémy jsou nejdůležitější takzvané OPC Data Acces. Tyto standardy podrobně upravují datovou komunikaci mezi koncovými zařízeními (PLC) a SCADA uživatelskými rozhraními. Jde o komunikaci v reálném čase, což je pro okamžitou vizualizaci

procesu nezbytné. Za zmínku také stojí standart typu OPC Alarms & Events, který sdílí výstrahy a události mezi koncovými zařízeními a klientskými aplikacemi. Novou verzí od verze OPC Clasic je OPC UA (unified architecture), která vznikla roku 2008. Tato nová verze na rozdíl od té starší má velký pokrok v bezpečnosti. Jde o certifikování a autentifikace na úrovni komunikačních protokolů, používají se dokonalejší šifry a mohou mít různou odolnost vůči prolomení v závislosti na důležitosti komunikace. [18–21]

3.5 RTU (Remote terminal units)

Spolu s PLC se jedná o průmyslové počítače určené k přímému připojení vstupů z akčních členů. Jinak také nazvány I/O členy – čtou vstupy I (inputs) a posílají hodnoty na výstupy O (outputs). Data z výstupů dále předávají do řídicích počítačů buď s OPC klient-server modulem nebo v menších aplikacích přes Ethernet přímo do řídicího počítače s nainstalovanou SCADOU.

Rozdíl mezi PLC a RTU je v poslední době dosti stírán a rozdíly jsou minimální. Stále ale platí, že RTU se používají spíše v celosvětovém měřítku, díky jejich lepší komunikace na větší vzdálenosti, která může probíhat i bezdrátově. [22] Další nevýhodou, která je v dnešní době už také dost potlačena je horší možnost programování. Starší modely RTU byli bez operačního systému, a proto jen četly vstupy a posílaly hodnoty na výstupy. [23]

3.6 PLC

PLC (Programmable Logic Controller) je průmyslový počítač řízený vlastním operačním systémem, který je konkrétně specializovaný a umožňuje nám řešit události v reálném čase, s co nejkratší dobou odezvy. Má velký počet vstupů a výstupů pro komunikaci s okolím. Na vstupní periférie jsou přivedeny signály z řízeného procesu. Můžou být binární ve stavech (zapnuto/vypnuto), nebo v podobě spojitých analogových signálů ukazující nám průběh měřené veličiny v čase. Na výstupních perifériích jsou připojeny akční prvky řízeného procesu, opět buď v binárním stavu, nebo ve stavu spojitého řízení analogové veličiny. Což znamená, že výstupem řídíme například zapnutí a vypnutí motoru, tlak v potrubí, anebo průtok. Celý proces řízení vstupů a výstupů je řízen pomocí

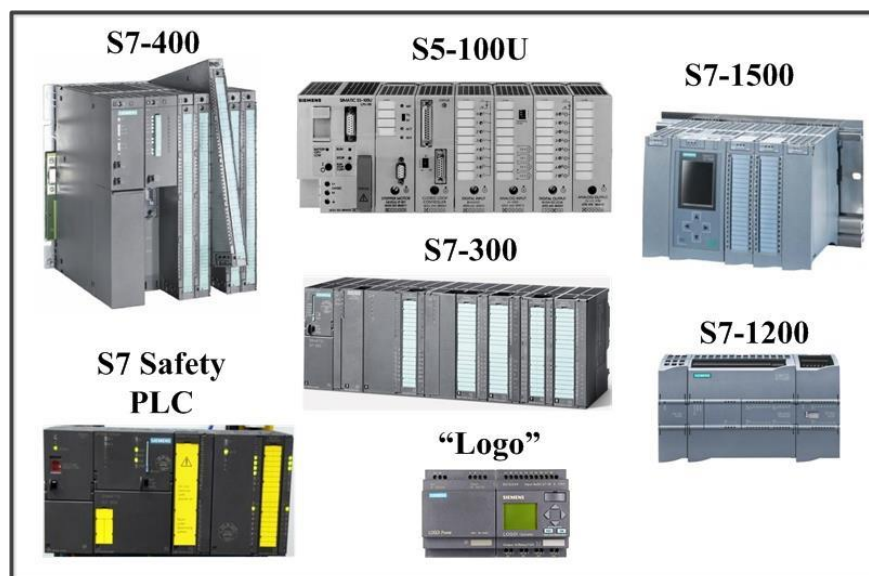
CPU, (central processing unit) [24], [25] centrální procesorovou jednotkou. Tato procesorová jednotka je integrovaným obvodem, reprezentovaným pomocí mikroprocesoru. CPU je předem naprogramováno a je v něm uložený programový algoritmus. Díky němu se rozhoduje, jak budou výstupy reagovat na změny hodnot vstupů. Celý program je prováděn cyklicky, a proto může PLC reagovat okamžitě na nečekané změny ve výrobním procesu. PLC má i další rozhraní, než jen vstupy a výstupy. Je vybaveno také rozhraním pro komunikaci s programátorem, dále také pro připojení k dalším PLC, anebo k jiným systémům v hierarchii SCADA.

Pro programování PLC se používají tři základní jazyky. První z nich, je jazyk liniových, resp. kontaktoých schémat (Ladder Diagram). Program je vytvořen postupným zapojením fiktivních prvků obvodu v tzv. proudových cestách. Dalším programovacím jazykem je jazyk blokových schémat logické funkce (Control System Flowchart). Vytvoření programu spočívá v zapojení jednotlivých bloků, které reprezentují činnosti, které má PLC vykonat. Příkladem jsou bloky NAND, NOR, NEGACE atd. Třetím programovacím jazykem je jazyk symbolických instrukcí (Statement List), který používá pro svůj zápis textovou formu. [26]

Programovatelné logické jednotky mohou být rozděleny do dvou typů, kompaktní a modulární. Kompaktní PLC je typ, který v jednom modulu obsahuje CPU, i určitý počet vstupů a výstupů. Poměr vstupů a výstupů je určen výrobcem. Výhodou kompaktních PLC bývá cena i rychlost zpracování signálů, protože signály nemusí procházet přes řadič sběrnice. Nevýhodou však může být to, že tyto PLC mají menší paměť pro program a data. Používají se proto pro malé řídicí systémy, převážně jako centrální systém.

Modulární PLC je tvořeno více moduly sestavenými do jednoho celku. Sestava je složena z hlavního řídicího modulu obsahujícího CPU a k němu připojené periferní moduly. Periferní moduly mohou být digitální, analogové, s různým typem komunikačních schopností, s různým počtem vstupů a výstupů atd. Jelikož modulární systémy slouží ke zpracování většího množství dat, jsou vybaveny větší kapacitou pro program i přijímaná data. Modulární PLC sestava

může být dále rozšiřována o další moduly. Speciálním modulem je bezpečnostní systém. Základním bezpečnostním systémem soustavy PLC je poruchově bezpečný systém tvořený dvěma samostatnými sestavami PLC, které jsou navzájem propojeny. Každá sestava má svoji CPU, které mezi sebou komunikují a vyměňují si informace. Je v nich nahrán stejný kód. Proto pokud dojde ke změně stavu, která je neočekávaná dojde k centrálnímu STOP stavu na obou PLC, což je obě odpojí od zdroje. [27–29]



Obrázek 2-Typy PLC od firmy Siemens V moderních průmyslových automatizacích se vyskytují PLC Siemens S7-1200 a vyšší, díky jejich rychlosti a možnosti programování
Převzato z: <http://automationprimer.com/2014/03/16/siemens-plcs/>.

3.7 Bezpečnost SCADA systémů a způsoby útoků

Průmyslová špionáž tu vždy byla a vždy bude. Ale u SCADA systémů se nejedná jen o průmyslové zneužití, ale může být ohrožena i národní bezpečnost, například při cíleném útoku na elektrické rozvodné sítě nebo na nemocniční zařízení. Proto se bezpečností SCADA začínají zabývat nejen výrobci softwaru, ale i světové vlády. Je třeba ale podotknout, že se tyto útoky zatím tolik nevyskytují. Což je způsobeno nejspíše specifikou těchto systémů, a tím, že je jejich struktura odlišná pro každého výrobce. Dále také, že si většina firem vytváří svůj vlastní komunikační protokol. Pravdou zůstává, že systémy SCADA

mají stále spoustu zranitelností a nejsou pravidelně aktualizované. [30] Což z bezpečnostního hlediska není dobrý krok.

Způsoby útoků na SCADA software se liší podle cíle. Tím nejčastějším scénářem útoku je phishing [31]. Útoky jsou vedeny prostřednictvím e-mailu, který míří na operátory linek a spoléhají na jejich nepozornost. Daná osoba otevře e-mail, který vypadá důvěryhodně či s odkazem na někoho z firmy, často mívá velmi podobnou strukturu a design. Obsahuje pak přílohu nebo odkaz, který vypadá zcela normálně, ale na pozadí stahuje, nebo přímo spouští program, který začne škodit. Cílem škodlivého programu může být získání a následné odeslání dat útočníkovi. Ukradená data mohou být předmětem vydírání, nebo mohou být prodána a zneužita konkurencí. Dalším z častých záměrů je napadení výrobního procesu a oběti tak napáchat velké škody, především ve výrobních podnicích. Útok může být veden i přímou infiltrací útočníka k SCADA hardwaru a následného připojení média obsahující malware. Tento způsob útoků je méně častý, protože vyžaduje vysokou znalost prostředí a s tím související fyzickou bezpečnost firmy. Jelikož SCADA neběží už jen na LAN sítích, ale často jsou připojeny ke stanicím s přístupem na Internet. Vzniká zde riziko přímých útoků z Internetu. Hackeři svými útoky cílí na nezabezpečené porty a chyby v komunikačních protokolech. Posílají nevině vypadající příkaz typu „ping“ a při objevení slabiny své útoky zintenzivní a snaží se o nakažení celého systému. Pokračování už je stejné jako při předchozích typech útoků – získat data, nebo narušit proces.

Ochrana proti útočníkům je různá. Základem je antivirus, aktualizovaný SW, odstranění zjištěných slabin systémů, často sám výrobce publikuje opravné balíky pro svůj SW. Dále pravidelné vynucení změny hesel a nastavení jejich komplexity (tj. minimální délka, složení z různých skupin znaků atp.). V dnešní době je řada i specializovaných SW a služeb, které mohou firmy na svoji ochranu využívat. Logovací systémy, které dokážou na základě zadaných pravidel vyhodnocovat a přímo blokovat akce na síti, či zasílat upozornění na příslušného pracovníka o nestandardních akcích. A v neposlední řadě také nejsilnější obrana, kterou je bezpečnostní vzdělávání a testování. Vzdělaná

obsluha systémů často dokáže sama a rychleji detekovat snahu o útok a díky své znalosti systému může přímo zasáhnout. [32]

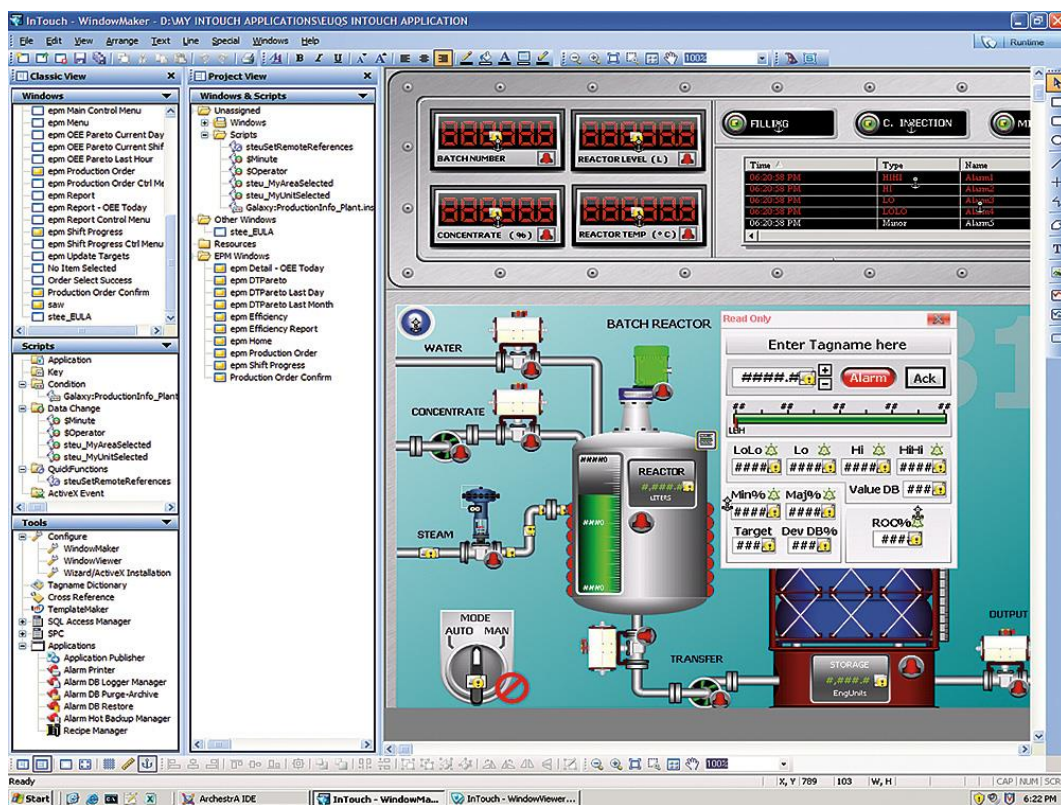
4 Výrobci SCADA software na českém trhu

Je mnoho firem a softwarů, které se vizualizací a sběrem dat zabývají. V průmyslu 4.0 je vizualizace procesu nezbytná, což otevírá cesty pro další výrobce.

4.1 InTouch software

Systém InTouch je produktem skupiny Wonderware s názvem FactorySuite. Podle informací v [33] jde o software, který používá jedna třetina všech průmyslových světových zařízení pracujících se systémy typu SCADA.

Jde o uživatelsky přívětivé prostředí s velkým množstvím grafických prvků a jejich snadnou implementací do provozu. Běží na všech platformách od firmy Windows od Windows 7, včetně MS Windows Server 2008 a vyšší, a je možné jej připojit k jakémukoliv průmyslovému automatizačnímu řídicímu zařízení. [5]



Obrázek 3-schéma boileru v programu InTouch. Převzato z:

<http://www.pantek.cz/produkty/intouch/>

Ze screenshotu programu (Obrázek 3) InTouch je možné vidět, jak vypadá samotná vizualizace ve SCADA systémech. Jde o jednoduché schéma měřené soustavy doplněné o grafické ukazatele a měřiče, které ukazují aktuální hodnoty. V horní polovině jsou vidět grafy, které slouží k chronologické vizualizaci dat.

Co se týče komunikace, disponuje InTouch rozsáhlou nabídkou propojení s OPC servery, PLC a RTU od skoro všech největších distributorů na světě.

Nevýhodou softwaru InTouch je pouze to, že jde o placený software, který si mohou dovolit jen velké firmy. [33]

4.2 Control Web

SCADA systém od firmy Moravské přístroje a.s. Jde o firmu s dlouholetou tradicí v oblasti elektroniky a průmyslového vybavení. V loňském roce představili novou verzi Control Web v7. Tato verze disponuje jednoduchým prostředím pro vývoj aplikačních programů a současně se také snaží o snadný přístup k automatizační technologii. Síla designérových programů je v detailnosti vykreslování procesu, i ve 3D. Jsou tedy názorněji vidět například poruchy a alarmy. Dále jsou podporovány internetové protokoly IPv6 a IPv4. Co se zdokonalilo je nástroj, který umožňuje volbu mezi dvěma odlišnými způsoby běhu aplikace. První aplikace jsou aplikace reálného času, což je hlavní princip SCADA. Další jsou ale aplikace, které jsou řízené změnou dat (data driven). V data driven aplikaci systém sám aktivuje přístroje, pokud se změní datové elementy, které přístroj používá. [34]

4.3 Simatic WinCC

Firma Siemens se nezabývá jen výrobou hardwaru, ale mají i svůj SCADA software. WinCC je založeno na kooperaci s databází (MS SQL Server), v níž jsou uložena konfigurační i archivní data. Tato koncepce umožňuje přístup ke zmíněným datům metodami ODBC (Open Data-Base Connectivity) a SQL (Structured Query Language). Jako skriptovací jazyk je zde použit jazyk C, nově doplněný o VisualBasic. Hlavní součástí SCADA od Siemens je TIA Portal, což je

designerové okno, nejen pro aplikace, ale i pro programování PLC. Což je obrovskou výhodou firem, které se zabývají výrobou PLC i SCADA.

Novinkou je nadstavba WinCC/WebUX, která nabízí možnost vzdáleného přístupu k aplikaci vytvořené ve WinCC. Zpřístupnění aplikace na dálku je využito u mobilních telefonů, tabletů, ale i PC. Výhodou je, že není třeba aplikaci instalovat, pracuje totiž prostřednictvím webového prohlížeče využívajícího HTML5.

Z hlediska sledování efektivity výroby byla vytvořena nadstavba zvaná WinCC/Performance Monitor. Tato nadstavba umožňuje flexibilní nastavení výpočtu specifických prvků systému, a z nich vypovídající analýzu pomocí Ganttova diagramu nebo tabulkového zobrazení. Na základě tohoto zobrazení je pro uživatele jednodušší analyzovat slabá místa, porozumět odchylkám a následně maximalizovat produktivitu a kvalitu finálních výrobků nebo celého procesu. Jak už bylo v této práci zmíněno, trendem současné doby je potřeba zpracovávat velké množství historických dat. Na základě tohoto zpracování vznikají analýzy, které vedou k zásadním rozhodnutím o řízené technologii. Aby mohla být data dlouhodobě uložena, přišla společnost Siemens s nadstavbou Simatic Process Historian, což je vlastně databáze přímo vytvořená od Siemens. Tato nadstavba umožňuje sběr obrovského množství dat z několika projektů WinCC najednou a současně jejich dlouhodobé uložení. Data jsou na stanici se serverem zasílána okamžitě při svém vzniku. Sestavy, které vznikají analýzou historických dat uložených ve WinCC, nebo v databázi Simatic Process Historian, je možné vytvořit a spravovat pomocí webového rozhraní. Pro zobrazení lze využít MS Excel, MS Word a MS PowerPoint. Je možné nastavit automatické zasílání těchto sestav e-mailem konkrétním osobám. Pod pojmem konkrétní osoby se rozumí daná vrstva uživatelů SCADA. Například alarmy pro techniky nepřijdou řediteli společnosti.

Co nové verze WinCC umějí je simulace všech komponent procesu. Veškeré procesy lze simulovat najednou, a tím řešit všechny možné situace, bez nutnosti připojení se k řízenému technologickému zařízení a mít veškerý

potřebný hardware. Podrobná simulace pak urychlí odstávku skutečného systému a předejde možným chybám. [35, 36]

4.4 Iconics

Iconics je americký výrobce automatizačního softwaru se sídlem i v České republice. V oblasti SCADA vyrábějí software pod názvem GENESIS64. Jeho síla spočívá ve velmi dobré grafické stránce vizualizačních oken. Je zde podporována grafika ve 2D i ve 3D, což umožňuje pohled na sledovaný problém ze všech stran. Iconics také nabízí balíčky pro analýzu dat, pro sledování alarmů a další standartní balíčky. Pro firmy celosvětového formátu je nabízeno rozšíření zvané EarthWorX™, které poskytuje real-time vizualizaci zařízení rozprostřených po celém světě. Je k dispozici mapa světa se závody a po přiblížení se zobrazí aktuální alarmy a stavy procesů. Pokročilým balíčkem je rozšíření zvané FDDWorX™. Aplikace je určena pro analýzu dat pomocí prediktivní analýzy. To znamená, že program čte aktuální data, má v databázi data minulá a na základě porovnání toho, co se v historii dělo, a jak se vyvíjí současné dění, může odhadnout budoucí chování systému. Tato analýza slouží k zabránění poškození zařízení nebo k ušetření výrobních nákladů. Když už k poruše systému dojde, je situace srovnána se situacemi, které nastaly v minulosti, což umožňuje rychlejší opravu chyb.

Díky úzké spolupráci s Microsoft a síti uživatelů po celém světě, patří firma Iconics mezi hlavní distributory SCADA na světě. [37, 38]

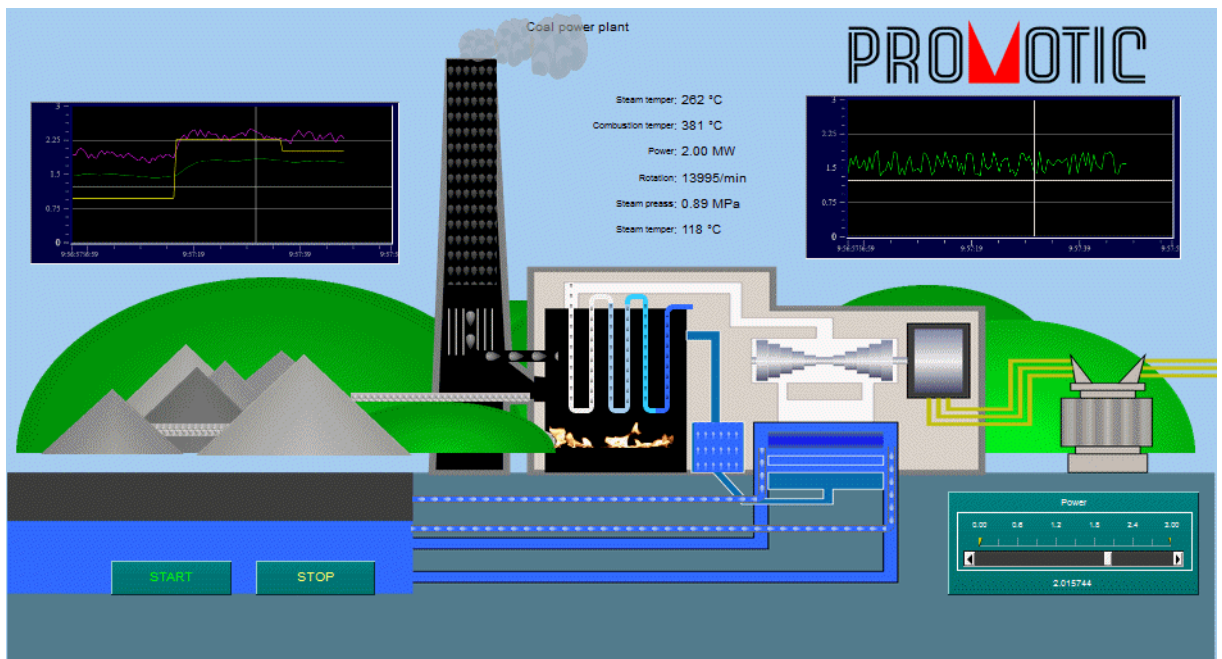
4.5 TIRS.NET

SCADA systém od české firmy CORAL s.r.o. je hojně využíván napříč Českou republikou. Mezi hlavní uživatele patří České Radiokomunikace, a.s., Povodí Labe nebo Elektrárny Opatovice, a.s. Umožňuje tradiční funkce SCADA, které již zde byly popsány, jako je komunikace a sledování technologie, získávání dat, zpětné zapisování a řízení technologie. Také je zde uživatelsky přívětivá vizualizace, zobrazování alarmů a trendů. Samozřejmostí je ukládání dat do databází a export obrázků, grafů a tabulek do nejrůznějších formátů. Na rozdíl od jiných SCADA systémů má systém TIRS.NET vlastní komunikační protokol

s příslušnou technologií implementována přímo v sobě. Tzn. veškeré komunikační možnosti a funkce jsou plně v režii tvůrců SCADA systému. Což umožňuje přizpůsobení komunikace neobvyklým situacím a průběžné doplňování další funkčností. [39]

4.6 Promotic

Promotic je komplexní SCADA objektový softwarový nástroj pro tvorbu aplikací, které monitorují, řídí a zobrazují technologické procesy v nejrůznějších oblastech průmyslu.[40] Jde o volně šiřitelný software od českého výrobce. Je založen na platformě Windows a podporuje komunikaci v rámci většiny PLC a vstupně/výstupních zařízení. V editoru obrazů je knihovna předdefinovaných grafických objektů, které se dají snadno použít. Podporována je také tvorba vlastních složených grafických prvků a jejich import/export do XML. Promotic (jako i ostatní SCADA software) umožňuje vzdálené prohlížení aplikace pomocí webových prohlížečů. Je možné v reálném čase v internetovém prohlížeči vidět průběhy trendů, zobrazovat alarmy a další aktuální události aplikace. Výhodou je také napojení na databáze, kam se mohou data ukládat a být později použita. [41]Také bych chtěl kladně hodnotit tutoriály na webu Promotic, [42] jsou přehledné a poměrně jednoduché. Nevýhodou může být, že pokud nemáte hardwarový klíč, tak se bude program spouštět jen v demo verzi a o klíč musíte požádat výrobce.



Obrázek 4- Grafické prostředí softwaru PROMOTIC pro vizualizaci uhelné elektrárny
 Dostupné z: <http://www.promotic.eu/cz/appl/img/imgCoalPlant.gif>

4.7 Reliance

Reliance je český SCADA software od společnosti GEOVAP, spol. s r.o. Od roku 2002 je členem OPC Foundation. Z tohoto hlediska je to výhodou pro zákazníky, protože jsou ke každé verzi dostupné OPC klienty, které se mohou připojit k jakémukoliv serveru OPC a PLC/HW zařízení. Systém Reliance se skládá z několika modulů, mezi ty nejzákladnější patří: Design modul, který je určen pro vývoj vizualizačního projektu. Runtime a Server modul sloužící ke spuštění vizualizace procesu. A v neposlední řadě moduly Mobile Client a Web Client pro samotnou vizualizaci procesu na webu a v mobilním zařízení.

Co má Reliance navíc jsou drivery pro interakci s uživateli pomocí SMS. To jsou například dotazy na stavy technologie, automatické reporty, povely nebo alarmy. [43, 44]

4.7.1 Nástroje pro vizualizaci v Reliance

Mezi zajímavé schopnosti Reliance patří technologie zvaná Postmort. Tato technologie zaznamenává průběh sledovaného procesu v reálném čase s možností následovného přehrávání. Jak uvádí firma distribuující Relianci na svém webu [43], je funkce Postmort denně využívána například v obslužném pracovišti metra ve stanici Florenc. Operátoři mohou sledovat aktuální změny stavů návěstidel, kolejových obvodu, výhybek a další, včetně průjezdů vlakových souprav stanicí. A v případě nestandardního chování si mohou zpětně přehrát celou situaci a diagnostikovat tak snáze chyby. [44]

Na *Obrázek 5* je znázorněna vizualizace vytápění a cirkulace vzduchu v rodinném domku. Reliance umožňuje vizualizaci projektu ve 2D, nikoliv však ve 3D, jak se může zdát z *Obrázek 5*, jde pouze o efekt. Kdyby toto nebyla pouze demo verze, byl by projekt otevřen v runtime modulu zvaném Reliance 4 Control. Tento program umožňuje zobrazení vizualizační obrazovky s aktuálními daty. Zobrazení trendů a grafů. K dispozici je také diagnostika běžící aplikace, která umožňuje detekce příčin poruch. Vizualizační obrazovka je naprogramována Javě. Dalším modulem, na kterém může výsledná vizualizace projektu běžet je Reliance 4 View, který má stejné funkce jako program Reliance 4 Control, ale rozdíl je v tom, že neumožňuje ovládání vizualizovaného technologického procesu. Je určeno jako nahlížecí pracoviště, nejčastěji pro manažery a vedoucí pracovníky firmy. Další možností vzdáleného přístupu k procesu poskytuje program Reliance 4- Web Client, který spouští vizualizaci u vzdálených uživatelů prostřednictvím sítě Internet. Což je například případ zde ukázaný. Reliance 4 Web Client je založený na platformě Java, je tedy nezávislý na operačním systému nebo na webovém prohlížeči. Data jsou čerpána z datových serverů Reliance 4 Control Server a Reliance 4 Server. Web Client má pouze omezené funkce runtime modulů. Nejmodernějším programem pro vizualizaci je Reliance 4 Smart Client, který poskytuje aktuální obraz z procesu v mobilním telefonu nebo tabletu. Vizualizační okna jsou generována datovými servery jako webové stránky ve

formátu HTML5. Jsou vytvořena JavaScriptových frameworkem jQuery Mobile, který umožňuje správné zobrazení v cílovém zařízení. [44]



Obrázek 5- Ukázka vizualizace rodinného domu v Reliance – Převzato z: <https://www.reliance.cz/cs/products/reliance-demo-applications/building-automation-demo>

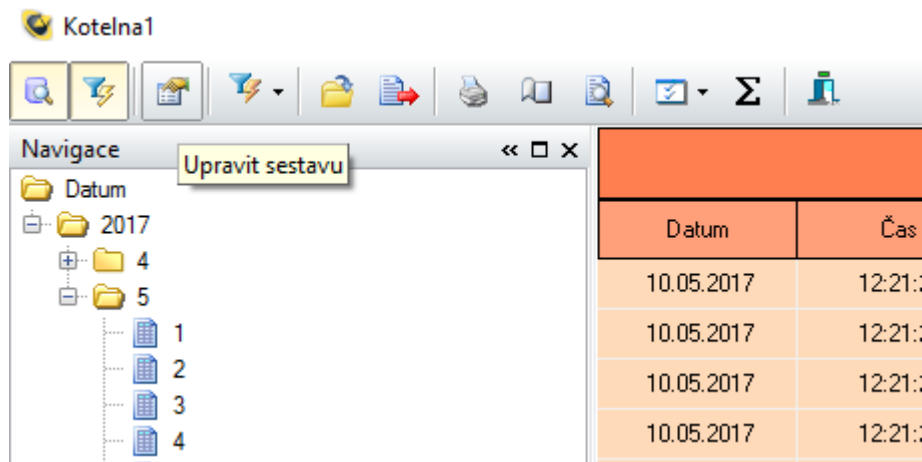
4.7.2 Nástroje pro analýzu dat v Reliance

Analýzou dat se zabývají speciální firmy, ale proč by se analýza zadávala firmám, které si účtují miliony, když jsou moderní SCADA systémy tak vyspělé, že data analyzují samy.

V Reliance je mnoho možností, jak naložit s nasbíranými daty. Přímo v runtime projektu je možné zobrazovat trendy a grafy. Pro případ rodinného domu je to například graf průběhu teplot, který analyzuje teplotu za posledních 18 hodin. Graf je možné přibližovat, posouvat, zkoumat předchozí hodnoty. Lze také přepnout do 3D grafu, nastavit osy nebo změnit vykreslovanou veličinu. Reliance podporuje externí export měřených dat, a to do formátů HTML, PDF,

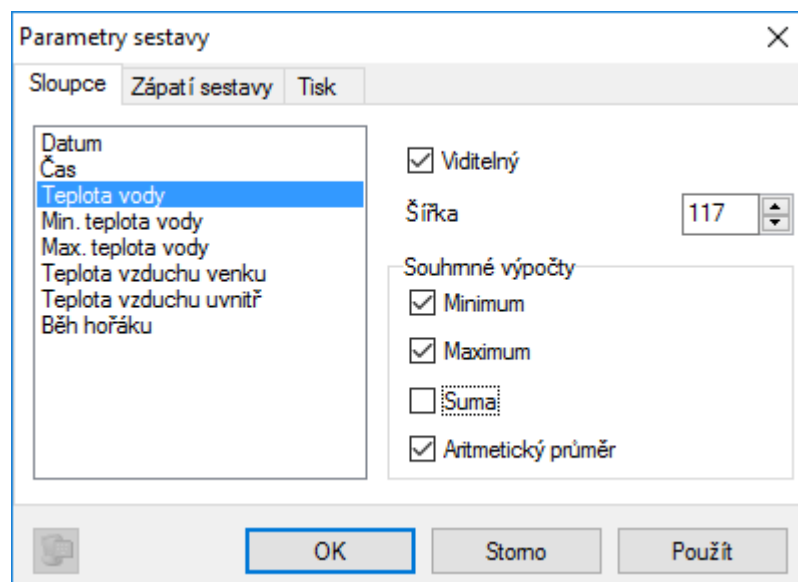
CSV, XLS, XLSX, XML a do CSV souborů. Tyto data se pak dají dále analyzovat pomocí MS Excel nebo externích aplikací, mezi které patří i Python.

Další možností, jak data analyzovat je v databázových tabulkách, a to pomocí tabulkových sestav. Po zobrazení tabulkové sestavy v runtime modulu je možné příkazem "Upravit sestavu" zobrazit dialog "Parametry sestavy".



Obrázek 6-Upravení sestavy vzorového příkladu a nastavení parametrů

V tomto dialogu je možné pro jednotlivé položky (sloupce) sestavy zapnout souhrnné výpočty (Minimum, Maximum, Suma, Aritmetický průměr).



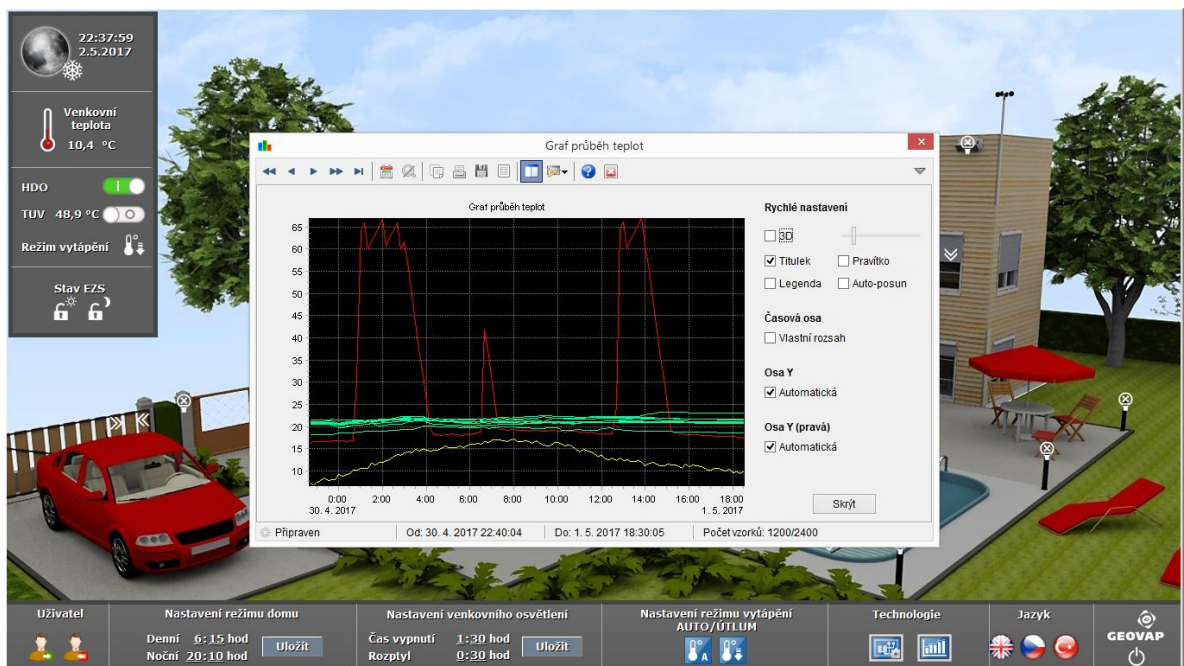
Obrázek 7-Nastavení parametrů sestavy pro vyhodnocení dat

V tomto okně se navolí parametry sestavy, pro které se vyhodnotí „souhrnné výpočty“. Kliknutím na tlačítko "Výsledky souhrnných výpočtů" se zobrazí okno s výsledky.

	Teplota vody	Min. teplota vody	Max. teplota vody	Teplota vzduchu venku	Teplota vzduchu uvnitř
Minimum	15	65	75	5	15
Maximum	76	65	75	5	15
Aritmetický průměr	39.17	65	75	5	15

Obrázek 8- Výsledky souhrnných výpočtů na vzorovém příkladu

Včasnou analýzou měřených dat je možné předejít poškození systému nebo neočekávanému chování, například hledáním maxim a minim. U automatických linek je pravidelný pohled na aktuální trend důležitý k udržení chodu linky a předpovědění budoucích událostí. Slouží také k hledání optimálních výrobních časů a množství vyrobených kusů. [44]



Obrázek 9-Graf průběhu teplot zobrazené v Reliance– Převzato z:
<https://www.reliance.cz/cs/products/reliance-demo-applications/building-automation-demo>

4.8 mySCADA

Softwaru od firmy mySCADA [45] je věnován zbytek této práce. Tato SCADA byla zvolena z důvodu navázání spolupráce mezi skupinou ASPICC na ČVUT [2] a týmem z mySCADA. Proto jsem dostal možnost jejich software používat bezplatně a vyzkoušet si ho. MySCADA je naprogramovaná v C++. Výhodou toho, že jsou aplikace naprogramovány v C++ je možné spustit mySCADA na všech operačních systémech včetně Android nebo Linux. Což znamená, že SCADA komunikuje napřímo s PLC (používá plný runtime modul), a nejen přes webový prohlížeč. To umožňuje ovládat výrobní procesy napřímo, ne přes tenkého klienta ve webovém prohlížeči. Navíc spousta firem stále více zvažuje bezplatné platformy jako je například Linux, a proto je pro ně mySCADA, která na tomto OS běží, rozumnou volbou. Pro spouštění skriptu na straně serveru mySCADA používá Node.js. A právě na tuto úroveň byl přidán Python. Lze tedy využívat stávající možnosti Node.js, ale také použít knihovny Python. Výhodnou skriptování na straně serveru je, že je požadavek uživatele vykonán webovým serverem, tudíž Node.js, a ne přímo prohlížečem. Webový prohlížeč je zde jen jako koncový zobrazovací prostředek. Pro archivaci dat používá mySCADA vlastní databáze, se kterou se komunikuje vlastním druhem komunikace. Mezi významné zákazníky této malé pražské firmy patří například MND, firma Weber nebo Skupina ČEZ. [45]

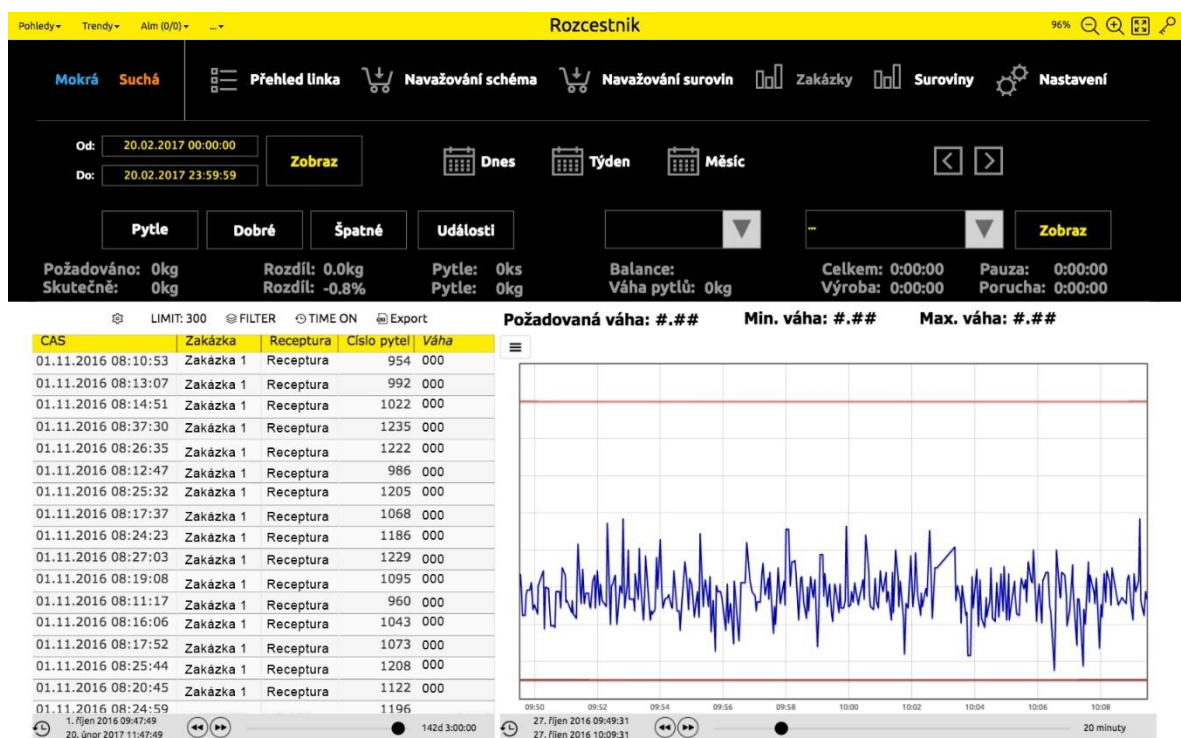
4.8.1 Nástroje pro vizualizaci dat v mySCADA

Vizualizační okno se otevře ve webovém prohlížeči jako normální webová stránka. Samozřejmě vyžaduje přihlášení, aby se zamezilo neoprávněné manipulaci. Také je možné použít aplikaci myVIEW, která je určena také na prohlížení stávajícího projektu. Celý projekt je spouštěn pomocí runtime aplikace, která je zde nazvána myPRO. Design okna se tvoří v myDESIGNER, který má základní grafické možnosti a umožňuje také import obrázků různých typů. V myDESIGNER se také nastavují proměnné, které náležejí určitým textovým polím nebo obrázkům atd. Také se zde nastavuje komunikace s PLC, a také ukládání do databází a do Alarmů. V sekci datalogy se nastavuje, za jaký čas se data zalogují a kam se mají uložit. Jak už bylo zmíněno mySCADA podporuje OS

od různých výrobců, proto může vizualizační okno běžet na telefonu nebo tabletu. Z toho okna je možné nejen proces sledovat, ale i ho řídit.

4.8.2 Nástroje pro analýzu dat v mySCADA

Rychlá analýza dat probíhá přímo v okně myVIEW, nebo ve webovém prohlížeči. Je zde možnost si nechat vykreslit aktuální grafy sledované veličiny. Pro trochu pokročilejší analýzu používá mySCADA balíček zvaný myREPORTS, který si stáhne data a zpracuje je. Výsledkem jsou grafy, vypsání minim, maxim a další základní matematické operace.



Obrázek 10- Příklad analýzy dat pro firmu Weber- Saint-Gobain, převzato z [45],

Pro ilustraci (Obrázek 10) byla vybrána demoverze z [45], na které je ukázáno, jak vypadá zobrazení alarmů, vykreslení dat a aktuální nastavení parametrů. Firma Weber se zabývá výrobou mixů a směsí pro stavebnictví, proto jsou na obrázku vidět požadované váhy jednotlivých surovin, počty pytlů a o jaký typ receptury jde.

5 Praktické ověření a popis komunikace řídicího systému PLC a mySCADA.

5.1 Použitý software a hardware

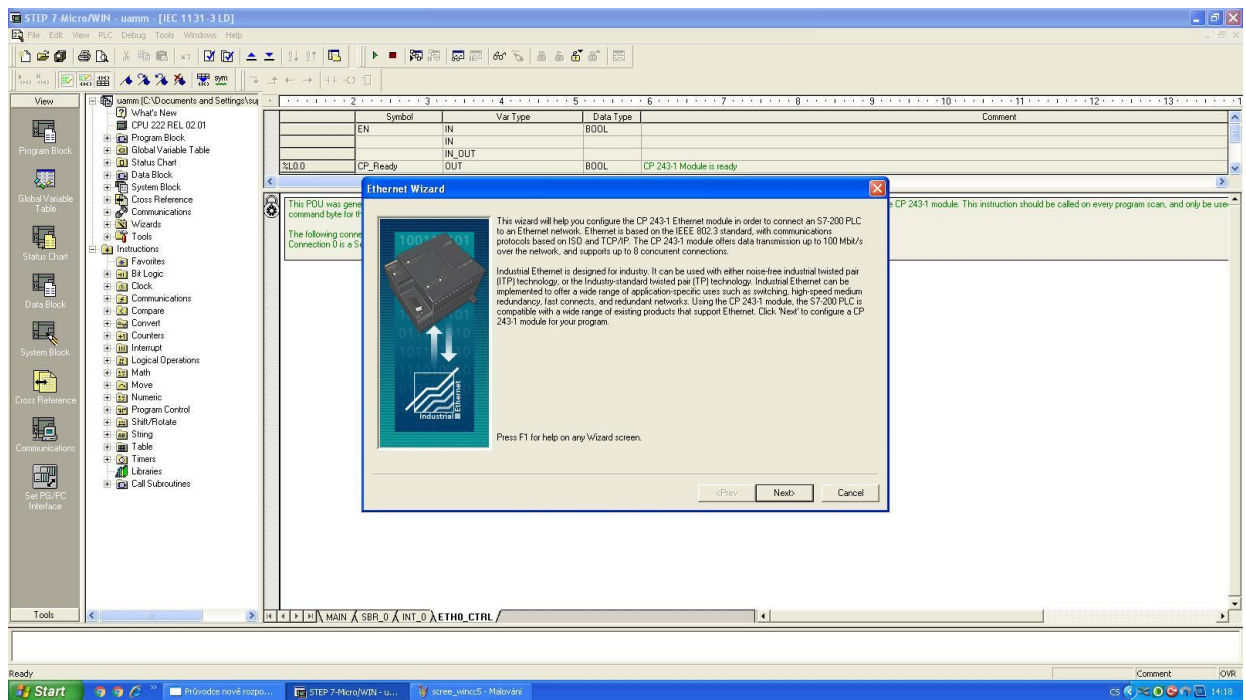
Pro praktickou ukázkou popsání komunikace mezi PLC a SCADA systémem jsem zvolil PLC Siemens S7 200. Jako SCADA systém jsem vybral software od firmy mySCADA. Používal jsem program myDESIGNER pro vytvoření projektu, dále pak myPRO, který slouží jako runtime. Pak stačí jen napsat IP adresu svého počítače do webového prohlížeče a zobrazí se to samé jako v myDESIGNER. Je možné také použít myVIEW, který slouží jako prohlížeč SCADA projektu. Zobrazí tedy to samé jako webový prohlížeč, když je zadána IP adresa počítače. Pro programování PLC byl použit program Step 7- Micro/WIN 3.1.

5.2 Popis komunikace

Komunikace bude vedena přes síť Ethernet, a to konkrétně ve školní síti v laboratoři 109.

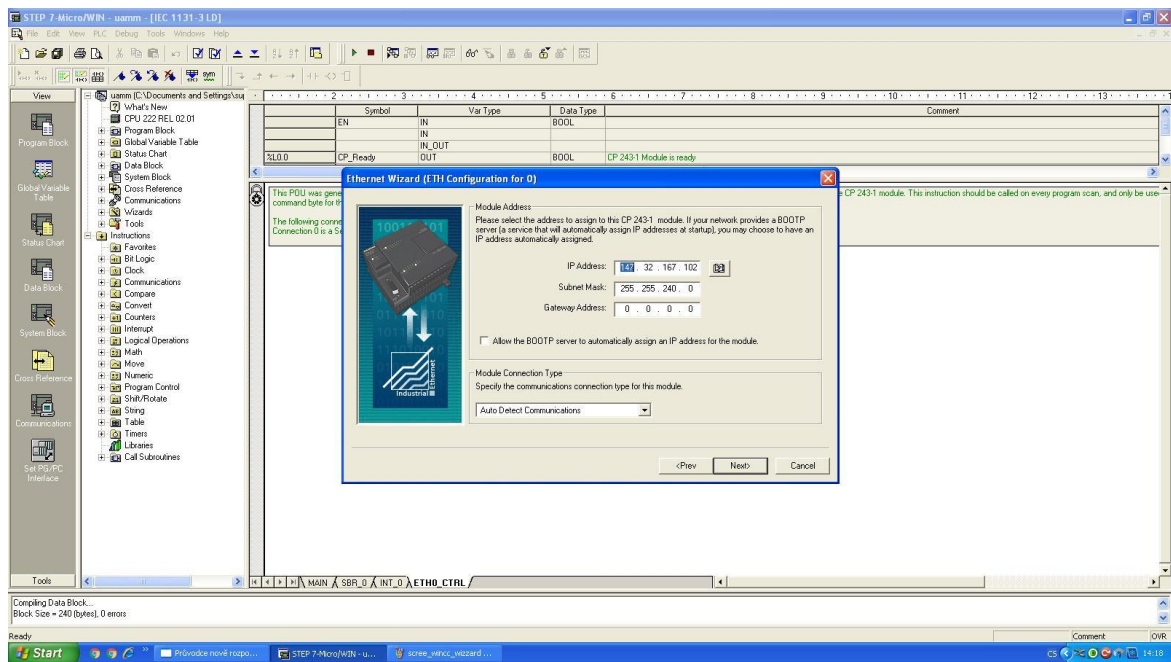
Pro navázání komunikace používá mySCADA přímý přístup do paměti PLC, je tedy schopna číst přímo proměnné nebo vstupy a nemusí být vázána jen na výstupy, jak tomu je u některých jiných SCADA softwarů.

Začne se kliknutím na PLC, kde je možnost Ethernet Wizard a objeví se okno jako je na (Obrázek 11). První okno je pouze informační, proto hned můžeme dát „next“ (dále), a pokračovat k dalšímu kroku.



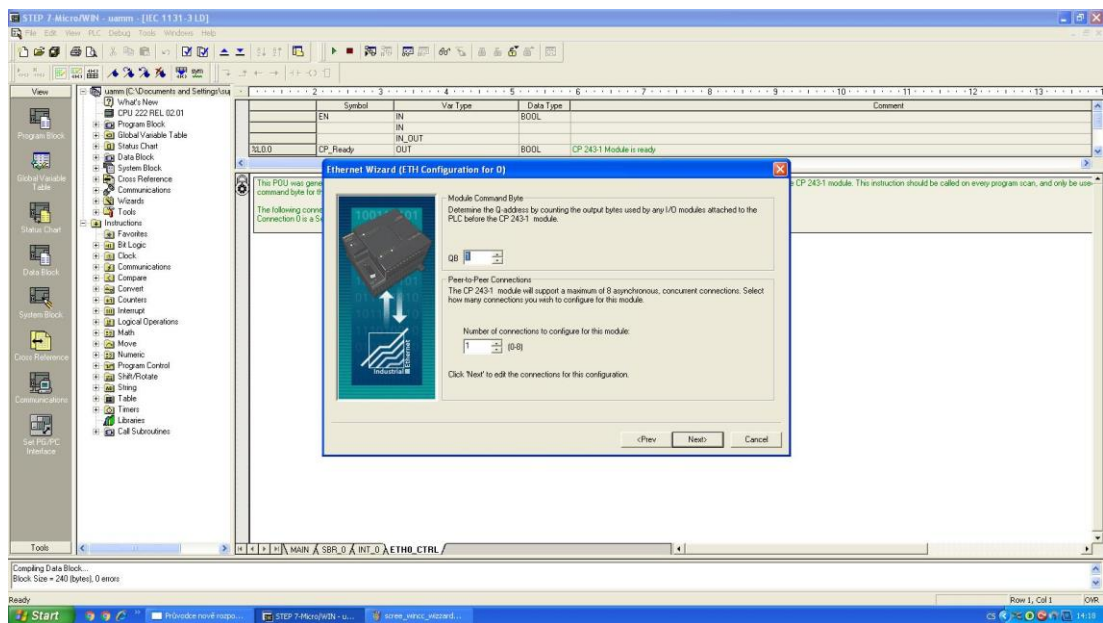
Obrázek 11- Ethernet Wizard, první okno

V dalším kroku se nastaví IP adresa PLC, které používáme. Pod touto adresou je maska podsítě. V mém případě je to maska školní podsítě, která je trochu netradičně neceločíselná. Hodnota „Gateway adres“ budou samé nuly. Pak je důležité nezaškrtnout volbu „allow the BOOTP server to automatically assign an IP address for the module“. A v posledním bodě „module connection type“ nechat, nebo nastavit „Auto Detect communications“. Po nastavení parametrů se může přejít k dalšímu kroku, tlačítko „next“. Na (Obrázek 12) je vidět, jak jsem postupoval při vytváření vlastní komunikace.



Obrázek 12- Nastavení IP adresy a masky podsítě

V dalším kroku se nastavuje počet přístupů. Nastavuje se zde počet přístupů k peer to peer komunikaci. V našem případě máme pouze jedno zařízení, které bude komunikovat s daným PLC. Pokud bychom chtěli zobrazovat měřené veličiny na více zařízeních musel by se jejich počet zde nastavit. Maximální hodnota může být však 8. Po nastavení volby pokračovat dále.

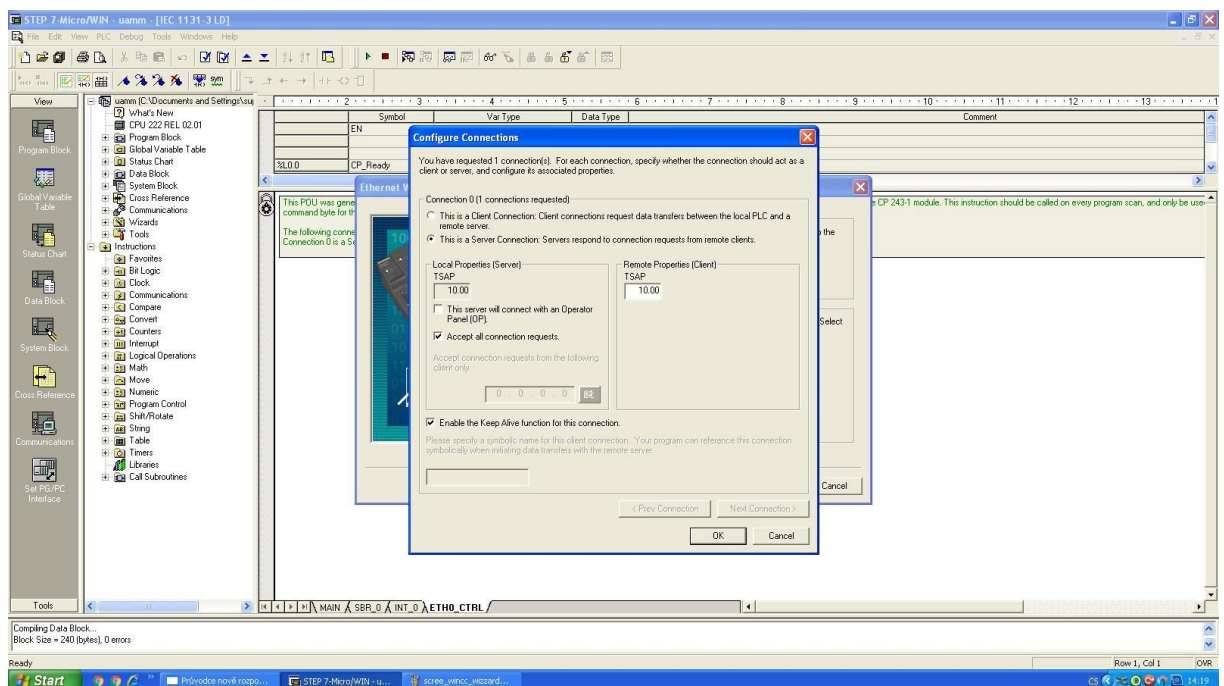


Obrázek 13- Nastavení počtu přístupů

V dalším, a asi nejdůležitějším kroku, se nastavuje způsob komunikace a komunikační kanál. V první části okna zaškrtneme druhou možnost, a to že jde o „Server connection:“. PLC je tedy v roli serveru a SCADA v roli klienta, který se připojuje.

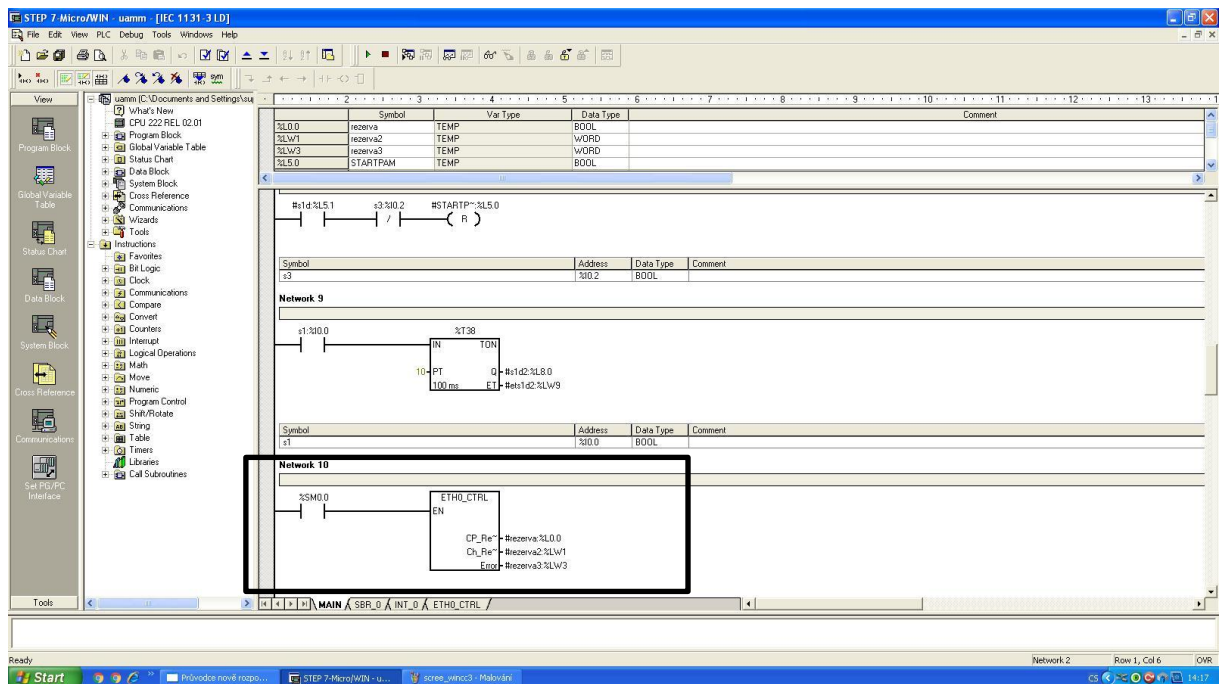
TSAP – (Transport service access point) [46] je transportní přístupový bod služby. Jedná se o nastavení transportní vrstvy, která zajišťuje přenos dat s požadovanou kvalitou. Local Properties je transportní adresa serveru, která se nastaví na automaticky na 10.00. Client adresu nastavíme na stejné hodnoty a to na 10.00. TSAP 10 znamená, že Ethernetový modul a PLC jsou spojeny vedle sebe. Druhá adresa 00 znamená, že je použit první z 8 komunikačních kanálů.

Dále se zaškrtneme volby „accept all connection request“ a „enable the keep alive functions for the connections“, které nám povolí přístup pro všechny systémy, které se budou chtít připojit.



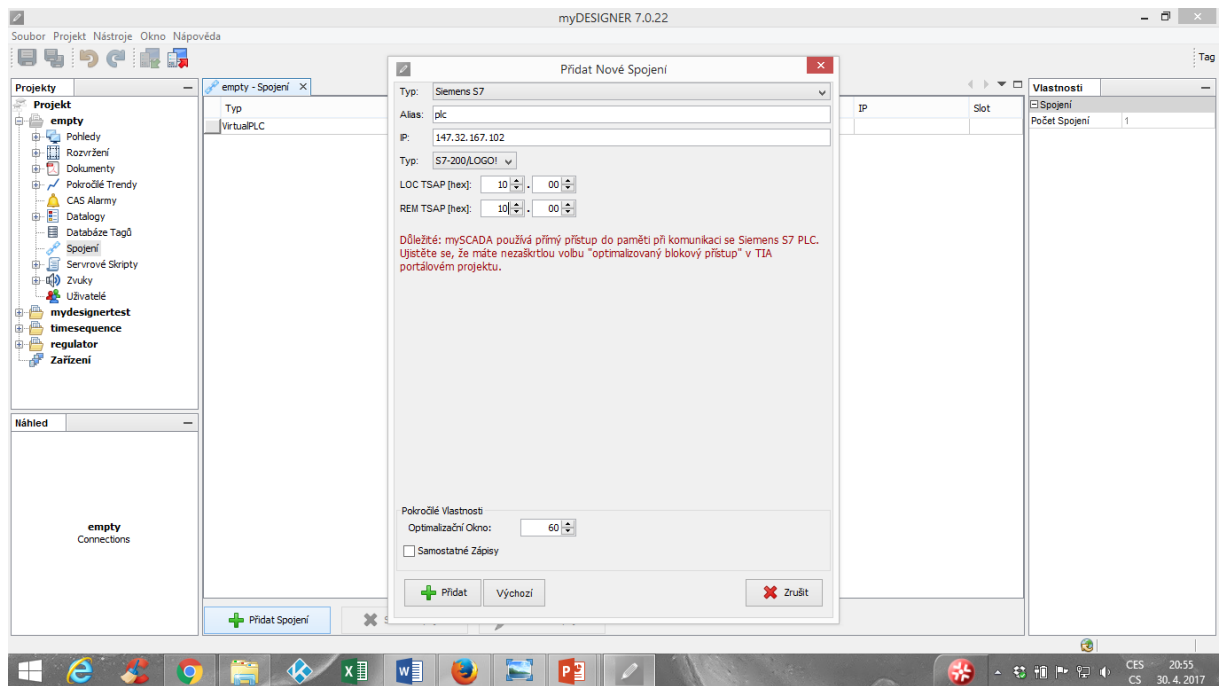
Obrázek 14- Nastavení typu spojení a transportního kanálu

Dále je třeba nastavit „ethernet controller“, vložením přímo do objektu. Důležité je, nepoužívat označené proměnné u „ethernet controller“ jinde v objektu.



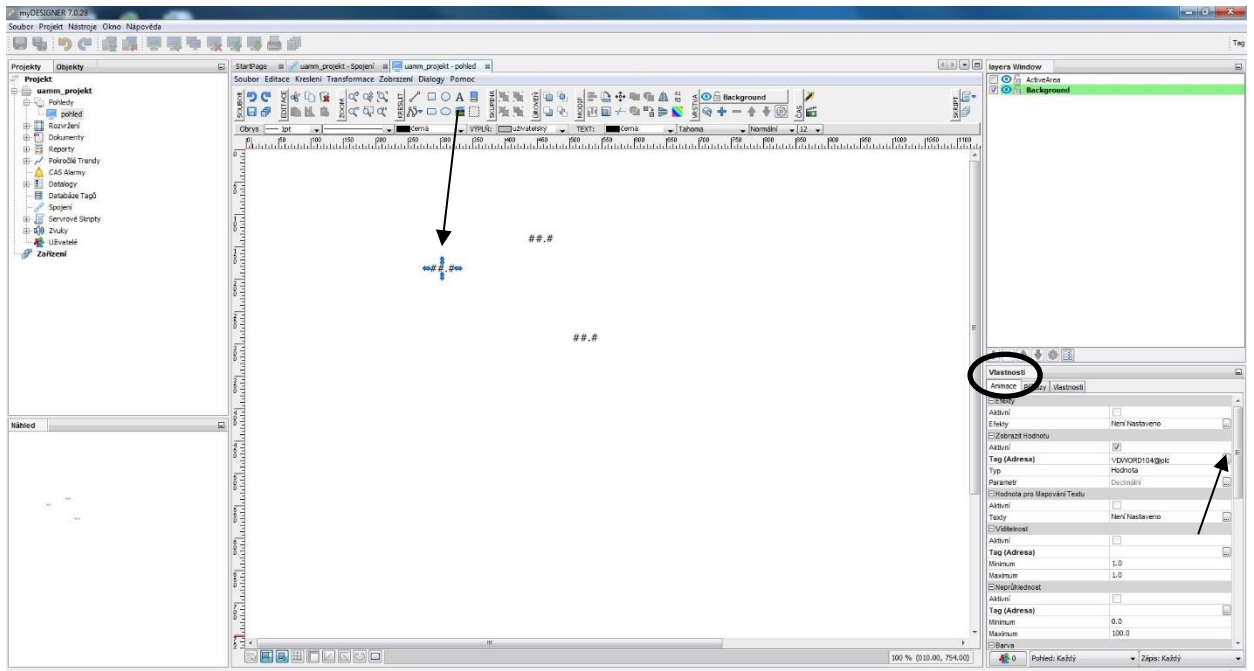
Obrázek 15- Nastavení ethernet controller

Pokud je vše nastaveno, může se přejít k nastavení komunikace přímo v mySCADA (Chyba! Nenalezen zdroj odkazů.). [47] Začneme otevřením stávajícího nebo vytvořením nového projektu. Dále v levém sloupci spojení dáme-přidat spojení. Vybereme, že chceme navázat spojení s PLC, pak vybereme, že to bude PLC značky „Siemens S7“. Dále se nastaví „alias“, jak budeme PLC nazývat. Pod alias napíšeme IP adresu PLC, se kterým se bude komunikovat. Vybereme typ PLC, v mém případě to je PLC S7 200 (pozn. pro LOGO! se nastavují stejné transportní vrstvy). Následuje napsání hodnot transportních vrstev, které jsem zvolil ve Step 7-micro/WIN a to v mém případě 10.00. Pak už stačí jen „přidat“.



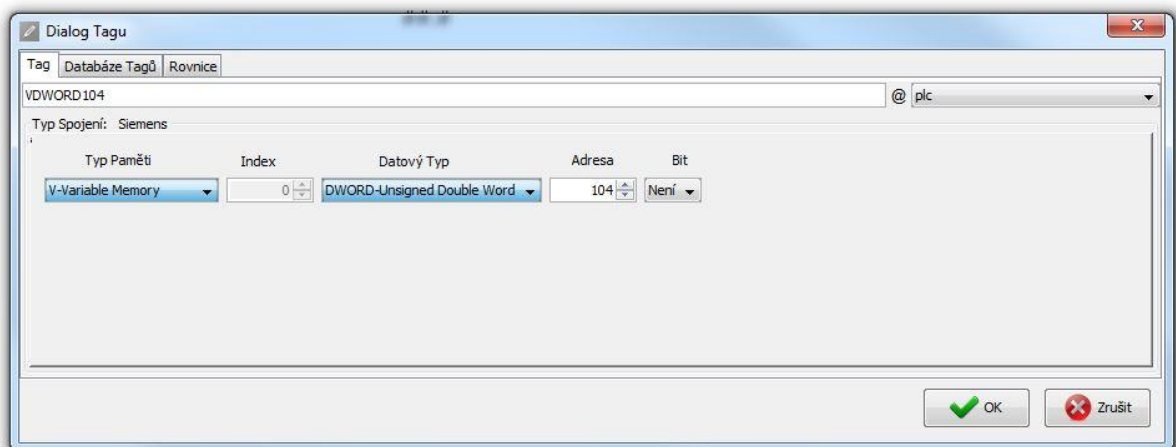
Obrázek 16- Přidání PLC do mySCADA a nastavení konkrétních parametrů

Ted' máme přidané zařízení do myDESIGNER, můžeme takhle postupovat i pro další zařízení. Abychom pak konkrétní hodnotu, kterou chceme vizualizovat viděli, kde chceme, přesuneme se v levé liště do „pohledy“ a pravým tlačítkem myši vytvořím „nový pohled.“ Zobrazí se mi okno jako je na (Obrázek 167). V tomto okně se tvoří budoucí design, který se zobrazí ve webovém prohlížeči. Začnu vložením textového pole a nastavením jeho vlastností. V pravém okénku „vlastnosti“ a v rámečku „zobrazit hodnotu“, kliknutím na tři tečky otevřu okno, kde se nastaví adresace přímo z PLC viz (Obrázek 18).



Obrázek 17- Okno pro design aplikace- myDESIGNER

V levém sloupci nastavuji „typ paměti“ tj. přímý přístup například k timerům, vstupům, výstupům a proměnným. V mém případě se jedná o „variable memory“, tedy o proměnnou. V „datovém typu“ se nastavuje datový typ proměnné, v mém případě se jedná o double word. Datový typ složený ze dvou Wordů, tedy ze 32 bitů v paměťové oblasti. Dále se nastavuje „adresa“ a „bit“. Datový typ i adresa jsou zapsané v programu pro programování PLC, je tedy nutné, aby byly adresy stejné, jinak ke spojení nedojde.

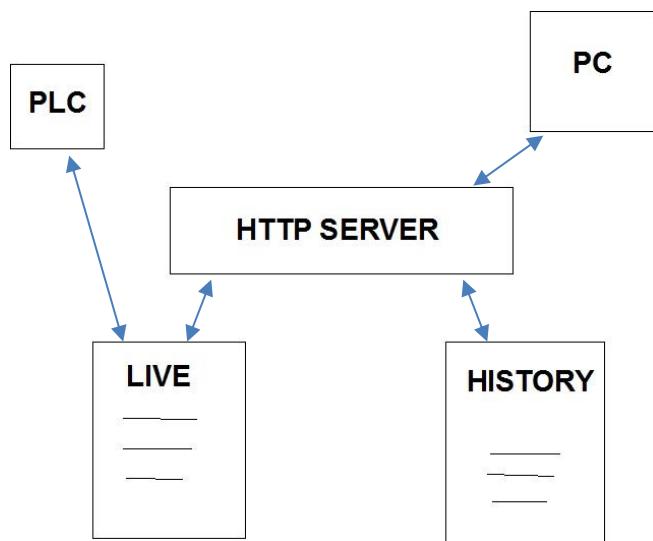


Obrázek 18- Přidání proměnných do mySCADA

Stačí už jen potvrdit, nahrát do zařízení (v mém případě je to „můj počítač“, k tomu je důležité mít nainstalované myPRO, jak jsem zmínil na začátku této kapitoly) a spojení pro komunikaci je dokončeno. Při otevření webového prohlížeče nebo myVIEW uvidíme aktuální hodnotu měřené veličiny. A takto SCADA systémy fungují.

6 Využití Python v mySCADA

Modul vytvořený v Pythonu byl navržen, aby dokázal nahradit webovou aplikaci a umožnil data vizualizovaná touto aplikací použít pro vlastní účely. MySCADA má sice nástroje pro analýzu dat – jako jsou grafy, určení minim, maxim, průměrů a odchylek [45]. Ale tyto nástroje mohou být úspěšně využívány i v Pythonu.



Obrázek 19- Přístup Pythonu k datům z mySCADA

Data nasbíraná pomocí PLC jsou nasbírána a evidována v „LIVE“ databázi. K té je možný aktuální přístup přes http server. Po nějakém zadaném čase už tyto data nejsou live, ale přesunují se do databáze s názvem „HISTORY“. Přesun informací mezi databází a serverem probíhá přes zazipované soubory napsané pro node.js, což je software napsaný v JavaScriptu. Aby Python mohl nahradit webovou aplikaci, je třeba, abychom se dostali k těmto zip souborům a převedli je do řeči Pythonu. Proto bylo za potřebí vytvořit v Pythonu knihovnu, která si stáhne zip soubor, rozbalí ho. A další funkci, která soubor přeloží do textu, se kterým už můžeme pracovat:

```
import MyScada #import knihovny mySCADA
```

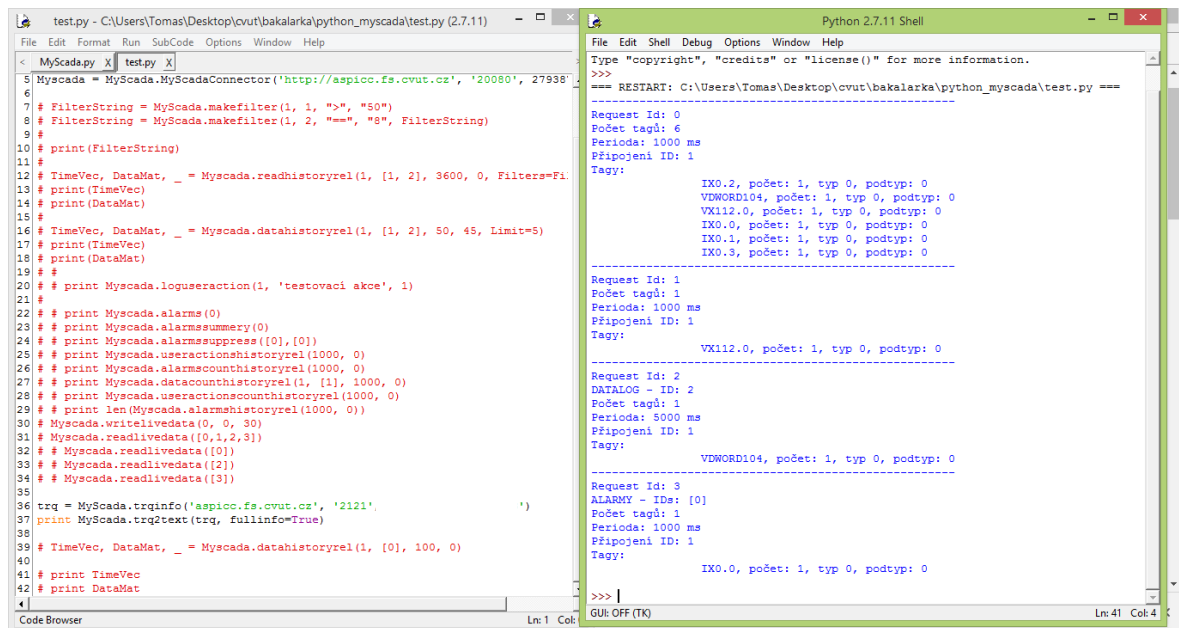
```
Myscada = MyScada.MyScadaConnector('webova adresa', 'project ID') #pripojeni se na server a nacteni informaci
```

#project ID je specifické číslo každého projektu, které se zjistí buď v myDESIGNERU a nebo přímo ve webovém rozhraní

```
trq = MyScada.trqinfo('webova adresa', 'project ID', 'jmeno', 'heslo') #rozzipovani souboru
```

```
print MyScada.trq2text(trq, fullinfo=True) #prevedeni do textového formátu
```

Na obrázku 12 je na levé obrazovce vidět program, který se dotazuje a v pravé části je způsob odpovědi. Je zde vypsáno, kolik je používaných portů, kolik je připojeno ID, jakého typu jsou sbírané proměnné a jaká je vzorkovací perioda.



Obrázek 20- Vlevo skript pro vypsání základních parametrů, vpravo spuštěný skript

Když už byla vytvořena tato základní knihovna, je na čase podívat se na způsob komunikace v mySCADA. Server se databáze „HISTORY“ dotazuje pomocí příkazů typu „t=30&prj=126768“. Kde t=30 znamená *type*, 30= *vyčtení alarmů* a *prj* je číslo projektu. Dotaz serveru je také doplněn o *start*- čas začátku ve vteřinách, *end*- čas konce, *user*-název uživatele, *limit*- omezení počtu, *seq*- sekvenční číslo, které se vrací o jednu hodnotu vyšší pokud je zadáno, jinak nic. Jako příklad uvedu otázku serveru na prvních 100 alarmů. Otázka je ve formátu: *t=30&prj=126768&limit=100* , kde 126768 je ID projektu. Odpověď může přijít ve formátu typu:

```
{ "r":30, "seq":123457, "d":({ "id":0, "cell":{ "msg": "alarm1", "area": "1", "dev": "device1", "stat": 1, "atm": (123456778, 0203), "dtm": (0, 0), "acktm": [123456778, 0203], "v": 12.321
```

,"av":12.421,"dv":12.51,"ackv":3.12,"f":"#.##","sv":3,"user":"admin"}},{...}),"err":"CH YBOVA_HLASKA"}. Kde jednotlivé prvky znamenají:

r - číslo funkce na kterou se odpovídá

d – blok s daty

id - unikátní číslo vypisovaného řádku

msg – zprava zobrazena při aktivaci alarmu

area – číslo oblasti

dev – název zařízení

stat – status alarmu – maskovaný char bit0 = 1 – ACT, bit1 = 1 – confirmed, bit2 = 1 – undefined (tag se nevyčítá), bit7 = 1 – tag nebyl vyčten ani jednou. Tzn. *stat*= 0 – tag vše čte a alarm neaktivní

atm – activation time (UTC Timestamp, popřípadě null, pokud nebyl alarm aktivován) (s,us)

dtm – deactivation time (UTC Timestamp, popřípadě null, pokud nebyl alarm deaktivován) (s,us)

acktm – acknowledge time (UTC Timestamp, popřípadě null, pokud nebyl alarm potvrzen)(s,us)

v – value aktuální hodnota tagu

av – hodnota při poslední aktivaci alarmu

dv – hodnota při poslední deaktivaci alarmu

f - formát pro zobrazování čísel

err: chybová hláška (pokud chyba nenastala, tato položka se neposílá)

Se základní znalostí principu dotazování a odpovědi, je možné Pythonem poslat stejnou otázku, bude se tedy vydávat za server a čekat na odpověď. Dotaz provedeme nově nadefinovanou funkcí.

```

def datahistory(self, datalogid, dataids, start, end, **kwargs):

myscadarequest = {'t': '30',
                  'prj': self.prj,
                  'start': start,
                  'end': end,
                  'limit': limit,
                  'limitEnd': limitend,
                  'minSev': minseverity,
                  'maxSev': maxseverity,
                  'seq': seqnumber
                  }

```

Funkce myscadarequest se dotazuje databáze na stejném principu, jako byl v této práci popsán výše.

```

response = self.sendrequesthistory(myscadarequest)

if response(u'seq') != seqnumber + 1:
    print('MySCADA warning: %s' % 'Sequential number is not corespond to request.')
    return 0

return response(u'd')

```

Odpověď je také typově stejná jako je popsáno v předchozí části. Server zašle odpověď a na obrazovce se nám vypíše dotazovaná data.

Příklad dotazu: *request ID pro Alarmy limit*

datalog *proměnná* *použitá funkce* *seqnumber*

```

dl=MyScada.datahistoryrel (2,[1],3600,0)

```

Výsledkem je pole o dvou vektorech (jak je vidět na obrázku), kde první část pole jsou časy v UNIX formátu a druhá část jsou data, ze kterých časy jsou.

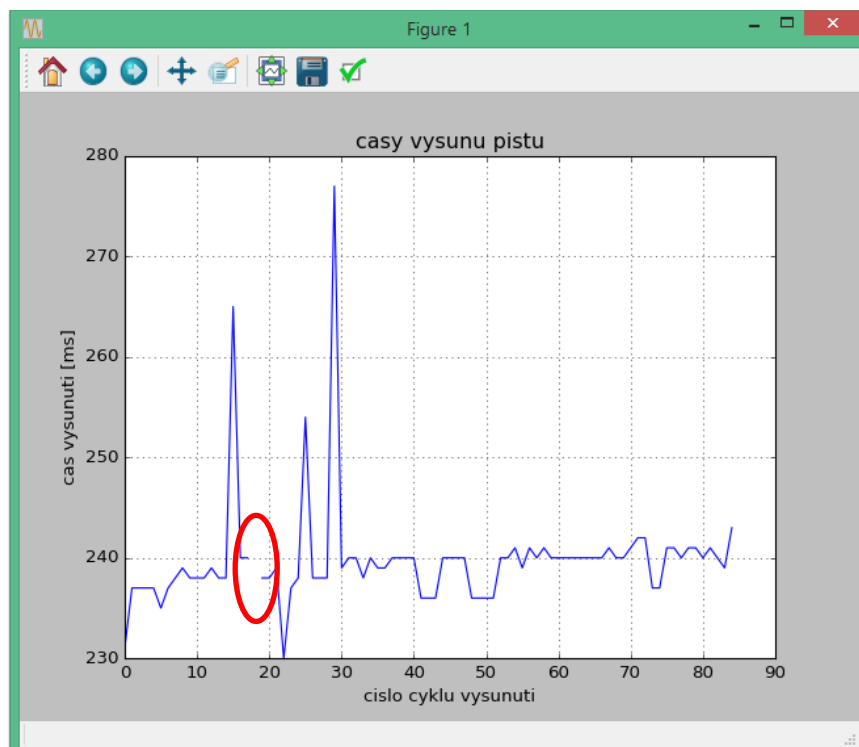
```

Python 2.7.11 Shell
File Edit Shell Debug Options Window Help
Python 2.7.11 [Anaconda 4.0.0 (64-bit)] (default, Feb 16 2016, 09:58:36) [MSC v.1500 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
== RESTART: C:\Users\Tomas\Desktop\cvut\bakalarKa\python_mySCADA\analiza.py ==
[[1494836267.3677, 1494836272.31131, 1494836277.26102, 1494836301.9305, 1494836307.73317, 1494836346.56698, 1494836351.51314, 1494836366.5067, 1494836371.43225, 149483
6376.34618, 1494836391.22671, 1494836396.12323, 1494836401.08637, 1494836406.74535, 1494836410.92012, 1494836415.83645, 1494836420.7504, 1494836425.70691, 1494836497.4
117, 1494836501.82152, 1494836511.75331, 1494836517.50537, 1494836522.4753, 1494836527.43263, 1494836532.34262, 1494836537.33796, 1494836542.29046, 1494836547.26922, 1
494836553.16998, 1494836571.93779, 1494836576.84161, 1494836581.846, 1494836586.82779, 1494836591.75315, 1494836596.63189, 1494836601.60018, 1494836616.52592, 14948366
21.47332, 1494836626.40662, 1494836631.37956, 1494836666.03066, 1494836671.70263, 1494836676.6118, 1494836681.52677, 1494836691.3883, 1494836696.34646, 1494836734.5200
3, 1494836784.08374, 1494836789.06999, 1494836798.9879, 1494836805.33555, 1494836808.86063, 1494836813.77603, 1494836843.5132, 1494836848.46544, 1494837003.67337, 1494
837007.15434, 1494837061.3638, 1494837066.62693, 1494837071.35633, 1494837076.19577, 1494837081.07437, 1494837270.06469, 1494837275.11433, 1494837314.64665, 1494837319
.71476, 1494837339.45485, 1494837344.4005, 1494837349.37248, 1494837354.34933, 1494837374.21694, 1494837379.06217, 1494837399.74822, 1494837403.81557, 1494837499.55959
, 1494837504.53614, 1494837509.55416, 1494837514.57402, 1494837519.51501, 1494837559.13277, 1494837564.07715, 1494837598.94414, 1494837603.90078, 1494837643.66669, 149
4837648.60372], [[231, 237, 237, 237, 237, 235, 237, 238, 239, 238, 238, 238, 239, 238, 238, 265, 240, 240, None, 238, 238, 239, 230, 237, 238, 254, 238, 238, 238, 277
, 239, 240, 240, 238, 240, 239, 239, 240, 240, 240, 240, 236, 236, 236, 240, 240, 240, 240, 236, 236, 236, 236, 240, 241, 239, 241, 240, 241, 240, 240, 240, 240, 240,
240, 240, 240, 241, 240, 240, 241, 242, 242, 237, 237, 241, 241, 240, 241, 241, 240, 241, 240, 239, 243]]]
>>> |

```

Obrázek 21- Pole naměřených hodnot vypsaných v Python

S těmito daty se pak může dále pracovat, já jsem si data uložil zvlášť, do souboru zvaného pickle (něco jako zazipovaný soubor), protože, kdybych se dotazoval pokaždé přímo mySCADA, ukazovaly by se mi stále nové hodnoty. Pak už stačilo jen pickle rozbalit a nakládat s ním jako s maticí. Bylo tedy možné zjistit maximální hodnotu, minimální hodnotu, průměrnou hodnotu a vykreslit grafy. První graf, co jsem vytvořil je graf časů výsunu pístu v závislosti na počtu vysunutí.



Obrázek 22- Graf časů výsunu pístu

Zajímavostí v tomto grafu je červeně vyznačená oblast. Zde nejsou žádná data dostupná, proto zde graf nebyl vytvořen. Možností, proč se něco takového událo

je více. První z nich je, že došlo k nějakému výpadku buď na síti, nebo mezi PLC a pístem, nebo mezi SCADA a PLC. Další možností může být, že SCADA zaznamenala UNIXový čas, který ale neměl žádnou hodnotu proměnné v paměti. Graf také velmi dobře ukazuje na maxima a minima, což nám hned umožňuje zkoumat, co se v daný okamžik mohlo stát. Na první pohled je vidět také čas, kolem kterého se výsuny pístu pohybují, je tedy zřejmé, že při potížích, se časy budou zpoždovat a celá křivka se posune směrem nahoru.

```

Python 2.7.11 Shell
Python 2.7.11 [Anaconda 4.0.0 (64-bit)] (default, Feb 16 2016, 09:58:36) [MSC v
.1500 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
=> RESTART: C:\Users\Tomas\Desktop\cvut\bakalarka\python_myscada\histogram.py =
>>> max (y)
277
>>> min (y)
230
>>> np.median (y)
240.0
>>> np.mean (y)
239.79761904761904
>>>

Python 2.7.11 Shell
File Edit Shell Debug Options Window Help
File Edit Format Run SubCode Options Window Help
MyScada.py X analiza.py X histogram.py X
1 import pickle
2 from numpy import *
3 import matplotlib.pyplot as plt
4
5 input = open('data.pkl', 'rb')
6
7 data = pickle.load(input)
8
9 #input.close()
10 #print data[1]
11 #print data
12 y = data[1]
13 y = array (y)
14 y = y.squeeze ()
15 #print data
16 plt.plot (y)
17
18 plt.xlabel('cislo cyklu vysunuti')
19 plt.ylabel('cas vysunuti [ms]')
20 plt.title('cas vyvunu pistu');plt.grid()
21 plt.show ()
22
23
GUI: OFF (TK) Ln: 13 Col: 4 Code Browser Ln: 21 Col: 11

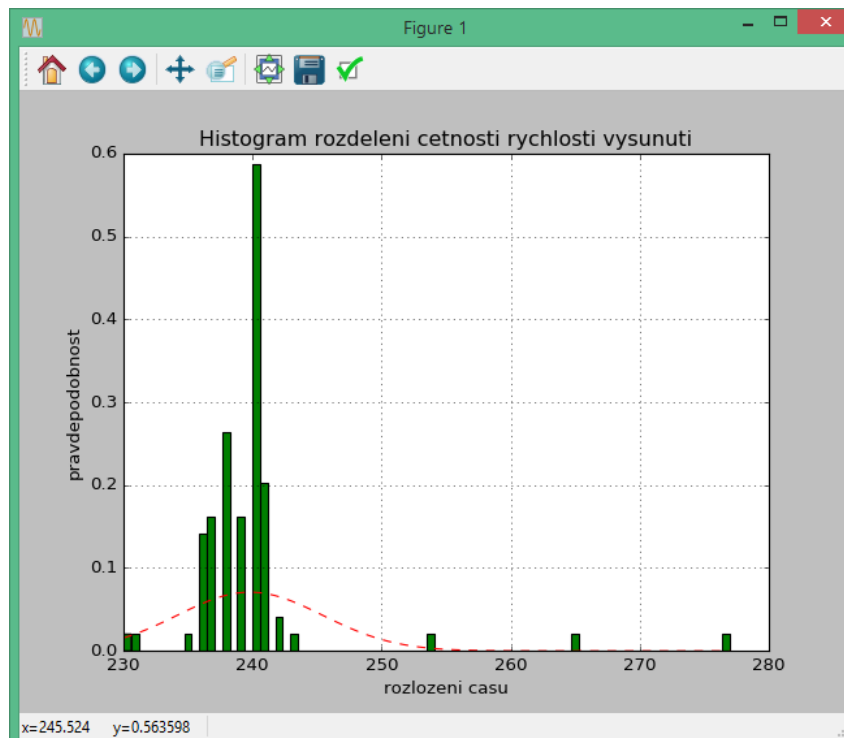
```

Obrázek 23-vpravo nástroj pro rychlou analýzu, vlevo výpočet základních parametrů

V pravém okně je napsaný skript pro otevření „pickle“ a pro vykreslení grafu. V levém okně jsou pak hodnoty minima, maxima, mediánu a průměru z časů vysunutí pístu. Tento skript ukazuje na jednoduchost a matematickou schopnost Pythonu při rychlé analýze dat.

Dalším grafem je histogram, který ukazuje na pravděpodobnost rozložení časů. Křivka, která tento histogram kopíruje je Gaussova křivka. V nejvyšším bodě této křivky je čas, za který se píst vysunul nejvíc krát. Podle chování této křivky je možné sledovat, jak se vyvíjí časy. Kdyby se například křivka začala posunovat směrem doprava, znamenalo by to, že se proces začal zpomalovat. Na základě toho, je tedy možné situaci vyhodnotit a zkoumat, jaké jsou příčiny problému.

V mém případě to může být např. snížení tlaku, nebo chyby v komunikaci mezi PLC a pístem.



Obrázek 24- Histogram rozdělení četností rychlosti vysunutí pístu včetně poruch s dlouhými intervaly vysunutí.

Ve skutečném provozu taková analýza může ušetřit až několik tisíc korun, protože při ohromném množství vyrobených kusů je poznat každé zpomalení na lince. Průběžnou analýzou a zkoumáním histogramů dokážeme ve správný čas vyslat technika, aby linku prověřil nebo jen seřídil, čímž se udrží výrobní čas procesu na stejné hodnotě.

7 Závěr

V této práci byly představeny systémy SCADA, jak fungují, kde v průmyslu, ale i mimo něj, nacházejí své místo a z jakých prvků a komponent jsou složeny. Byl zmíněn i způsob komunikace, typy komunikačních protokolů, a také byla nastíněna otázka bezpečnosti, která je v poslední době žhavým tématem. Na představení SCADA systémů navazovala řešerše výrobců tohoto softwaru na českém trhu. Byly zde shrnuty možnosti a velikosti firem, jejich vizualizační balíčky a schopnosti analyzovat data.

V další části této bakalářské práce jsem se zaměřil na zprůchodnění komunikace mezi PLC a mySCADA. Komunikace byla úspěšně vytvořena, a proto jsem mohl sbírat data a ovládat výsuny pístu, což simuluje skutečný výrobní proces. Software od mySCADA jsem si vybral z důvodu přehlednosti jejich aplikací, dále také pro navázání spolupráce, a v neposlední řadě také díky jejich ochotě se rozdělit o jejich know-how v komunikaci mezi serverem a klientem-webovým prohlížečem.

Poslední část byla zaměřena na analýzu nasbíraných dat. Pomocí programu vytvořeného skupinou ASPICC jsem získal data, poté jsem vytvořil program, který data připraví pro analýzu. Následovalo pak vykreslení grafu a zjištění maxim, minim průměrů a odchylek. Poté jsem vše zanesl do histogramu, který ukazuje na četnosti vysunutí pístu pro časové hodnoty. Poté jsem histogram proložil křivkou, která částečně ukazuje na Gaussovo rozložení. Aby byla Gaussova křivka přesnější muselo by se provést více měření. V průmyslu se na takovémto rozložení a jeho případném posunutí dá zkoumat zpomalování nebo naopak zrychlování výrobního procesu.

Práce může sloužit také jako výukový materiál v otázce SCADA systémů a navazování komunikace a také jako stručný přehled aktuálního trhu se systémy SCADA. Dále práce souhrnně popisuje možnost využití Pythonu pro pokročilou analýzu dat pro mySCADA systém, i Pythonu jako možného front-endu místo webové aplikace, což otevírá nové možnosti SCADA systémů. Je zde prostor pro vylepšení modulu v Pythonu pro mySCADA a jeho použití na úrovni

webového zobrazení, s tím rozdílem, že je zde omezena stránka vizualizace, ale na druhou stranu snadnější a okamžité matematické operace s daty. Tento krok by mohl optimalizovat výrobu, uspořit energii potřebnou k výrobě a přispět k lepšímu sledování kvality.

8 Zdroje

- [1] *What is SCADA? Supervisory Control and Data Acquisition* [online]. [vid. 2017-05-17]. Dostupné z: <https://inductiveautomation.com/what-is-scada>
- [2] *About ASPICC | ASPICC, CTU FME* [online]. [vid. 2017-06-06]. Dostupné z: <http://aspicc.fs.cvut.cz/>
- [3] *SCADA* [online]. 2015 [vid. 2016-12-05]. Dostupné z: <https://cs.wikipedia.org/w/index.php?title=SCADA&oldid=12872702>
- [4] *SCADA* [online]. 2016 [vid. 2017-05-15]. Dostupné z: <https://en.wikipedia.org/w/index.php?title=SCADA&oldid=751057995>
- [5] *4 SCADA/HMI systémy* [online]. [vid. 2017-02-02]. Dostupné z: http://homel.vsb.cz/~rep75/Predmety/Distrib_sys/Vytah/04.htm
- [6] *Skříně pro rozhraní člověk-stroj | Schneider Electric Česká republika* [online]. [vid. 2017-06-06]. Dostupné z: [/cs/product-range/60124-skrine-pro-rozhrani-clovek-stroj/](https://www.schneider-electric.com/cz/cs/product-range/60124-skrine-pro-rozhrani-clovek-stroj/)
- [7] *RS 485 & 422 | Vývoj.HW.cz* [online]. [vid. 2017-06-10]. Dostupné z: <http://vyvoj.hw.cz/teorie-a-praxe/dokumentace/rs-485-422.html>
- [8] PROF. ING. FRANTIŠEK ZEZULKA, CSC. Průmyslový Ethernet. *Automa* [online]. nedatováno, **2005**(4). Dostupné z: http://automa.cz/cz/casopis-clanky/prumyslovny-ethernet-2005_04_30417_493/
- [9] *Základní informace o RS-485 a RS-422 pro každého | Automatizace.HW.cz* [online]. [vid. 2017-06-10]. Dostupné z: <http://automatizace.hw.cz/zakladni-informace-o-rs-485-rs-422-pro-kazdeho>
- [10] PETERKA, Jiri. *Jiří Peterka: Síťový model TCP/IP* [online]. [vid. 2017-06-06]. Dostupné z: <http://www.earchiv.cz/a92/a231c110.php3>
- [11] PRÖTZSCH, Matthias. *PROFINET* [online]. [vid. 2017-06-06]. Dostupné z: <https://www.siemens.com/global/en/home/products/automation/industrial-communication/profinet.html>
- [12] *INTERBUS* [online]. 2017 [vid. 2017-05-29]. Dostupné z: <https://en.wikipedia.org/w/index.php?title=INTERBUS&oldid=767846551>
- [13] *MODBUS | Automatizace.HW.cz* [online]. [vid. 2017-05-29]. Dostupné z: <http://automatizace.hw.cz/clanek/2004070701>
- [14] *Sylaby pro Internet a sítě, Programování aplikací pro Internet* [online]. [vid. 2017-05-15]. Dostupné z: http://www.352.vsb.cz/uc_texty/Internet_site/ias-protokoly.htm

- [15] *master/slave - Slovník počítačové informatiky a sítí* [online]. [vid. 2017-05-29]. Dostupné z: <http://www.svetsiti.cz/slovník.asp?hid=masterslave>
- [16] VACHTL, Pavel. *Sdílení souborů na Internetu a síť P2P - základní technologický přehled* [online]. 25. listopad 2009 [vid. 2017-06-06]. Dostupné z: <http://pcworld.cz/internet/sdileni-souboru-na-internetu-a-site-p2p-zakladni-technologicky-prehled-8350>
- [17] *klient-server :: počítačové sítě* [online]. [vid. 2017-06-06]. Dostupné z: <http://site-pc.webnode.cz/klient-server/>
- [18] What is OPC? *OPC Foundation* [online]. [vid. 2017-05-15]. Dostupné z: <https://opcfoundation.org/about/what-is-opc/>
- [19] NOSEK JIŘÍ. *ADMINISTRACE POČÍTAČOVÝCH SÍTÍ OPC Server Funkce a využití v průmyslové automatizaci* [online]. 2011. Dostupné z: http://www1.fs.cvut.cz/cz/u121110/site/Nosek-OPC_server.pdf
- [20] *OPC servery pro PLC Simatic S7 - přehled trhu | Automatizace.HW.cz* [online]. [vid. 2017-05-15]. Dostupné z: <http://automatizace.hw.cz/opc-servery-pro-plc-simatic-s7-prehled-trhu>
- [21] *Unified Architecture - OPC Foundation* [online]. [vid. 2017-05-29]. Dostupné z: <https://opcfoundation.org/about/opc-technologies/opc-ua/>
- [22] *Difference Between PLC and RTU | Difference Between* [online]. [vid. 2017-06-06]. Dostupné z: <http://www.differencebetween.net/technology/industrial/difference-between-plc-and-rtu/>
- [23] INDIRA KUNDU. Basics of Automation, PLC and SCADA. In: [online]. Engineering. B.m. 03:42:31 UTC [vid. 2017-02-07]. Dostupné z: http://www.slideshare.net/indira_kundu/basics-of-automation-plc-and-scada
- [24] *PLC - sestava* [online]. [vid. 2017-06-06]. Dostupné z: <http://www.plc-automatizace.cz/knihovna/plc/plc-hw-sestava.htm>
- [25] *Centrální procesorová jednotka* [online]. 2017 [vid. 2017-05-15]. Dostupné z: https://cs.wikipedia.org/w/index.php?title=Centr%C3%A1ln%C3%AD_processorov%C3%A1_jednotka&oldid=14613869
- [26] HOFREITER, Milan, ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE a STROJNÍ FAKULTA. *Základy automatického řízení*. V Praze: České vysoké učení technické, 2012. ISBN 978-80-01-05007-1.
- [27] *PROMOTIC a PLC Simatic S7 raz-dva | Automatizace.HW.cz* [online]. [vid. 2017-02-08]. Dostupné z: <http://automatizace.hw.cz/promotic-a-plc-simatic-s7-raz-dva>
- [28] *PLC - Programovatelný logický automat* [online]. [vid. 2017-02-07]. Dostupné z: <http://www.plc-automatizace.cz/knihovna/plc.htm>

- [29] MIREK. *Co to je PLC a k čemu?* [online]. [vid. 2017-05-29]. Dostupné z: <http://elektrika.cz/data/clanky/co-to-je-plc-a-k-cemu>
- [30] *Postřehy z bezpečnosti: zranitelnosti SCADA/ICS zkoumány experty z EU, NATO i USA - Root.cz* [online]. [vid. 2017-05-06]. Dostupné z: <https://www.root.cz/clanky/postrehy-z-bezpecnosti-zranitelnosti-scada-ics-zkoumany-experty-z-eu-nato-i-usa/>
- [31] *HOAX | Phishing | Co je to phishing* [online]. [vid. 2017-06-06]. Dostupné z: <http://www.hoax.cz/phishing/co-je-to-phishing>
- [32] *Bezpečnost na internetu* [online]. 2017 [vid. 2017-06-06]. Dostupné z: https://cs.wikipedia.org/w/index.php?title=Bezpe%C4%8Dnost_na_internetu&oldid=14911703
- [33] *Wonderware InTouch HMI* [online]. [vid. 2017-02-02]. Dostupné z: <http://global.wonderware.com/DE/Pages/WonderwareInTouchHMI.aspx>
- [34] *Představili jsme finální podobu nového systému Control Web v7* [online]. [vid. 2017-05-13]. Dostupné z: <http://www.mii.cz/art?id=822&cat=146&lang=405>
- [35] *SIMATIC WinCC V13 - Digital Factory & Process Industries and Drives - Siemens* [online]. [vid. 2017-05-15]. Dostupné z: <http://www1.siemens.cz/ad/current/index.php?vw=0&ctxnh=d8746d77f6&ctxp=home>
- [36] PLACHÝ LADISLAV. Siemens představuje vizualizační software WinCC v rámci nové verze TIA Portal V14. *Automa*. nedatováno, **16**(11).
- [37] *Iconics* [online]. 2017 [vid. 2017-05-14]. Dostupné z: <https://en.wikipedia.org/w/index.php?title=Iconics&oldid=771492417>
- [38] *Výrobní informační systémy & Obchodní vizualizační systémy* [online]. [vid. 2017-05-18]. Dostupné z: <http://www.iconics.com/Home.aspx?lang=cs-CZ>
- [39] TIRS.NET 6 - profesionální SCADA/HMI systém. *CORAL s.r.o.* [online]. [vid. 2017-05-17]. Dostupné z: <http://www.coral.cz/scada-hmi-systemy-tirs/tirs-net-6-profesionalni-scadahmi-system/>
- [40] *Co je PROMOTIC* [online]. [vid. 2017-02-07]. Dostupné z: <http://www.promotic.eu/cz/pmdoc/WhatIsPromotic/WhatIsPromotic.htm>
- [41] *SCADA/HMI systém PROMOTIC* [online]. [vid. 2017-05-18]. Dostupné z: <https://www.promotic.eu/cz/index.htm>
- [42] *Co je PROMOTIC* [online]. [vid. 2017-05-15]. Dostupné z: <https://www.promotic.eu/cz/pmdoc/WhatIsPromotic/WhatIsPromotic.htm>

- [43] *SCADA/HMI systém Reliance 4* [online]. [vid. 2017-05-18]. Dostupné z: <https://www.reliance.cz/cs/products/reliance4-scada-hmi-system#page=structure>
- [44] *Vývojové prostředí Reliance 4 Design* [online]. [vid. 2017-05-18]. Dostupné z: <https://www.reliance.cz/cs/products/reliance4/development-environment>
- [45] APP, mySCADA. mySCADA Technologies- professional HMI/SCADA Solution. *mySCADA Technologies* [online]. [vid. 2017-05-28]. Dostupné z: <https://www.myscada.org/>
- [46] *TSAP meaning in a multi-PLC Ethernet Network - Entries - Forum - Industry Support - Siemens* [online]. [vid. 2017-05-15]. Dostupné z: <https://support.industry.siemens.com/tf/WW/en/posts/tsap-meaning-in-a-multi-plc-ethernet-network/9614?page=0&pageSize=10>
- [47] APP, mySCADA. Getting Started Tutorial. *mySCADA Technologies* [online]. 13. března 2015 [vid. 2017-06-06]. Dostupné z: <https://www.myscada.org/getting-started-tutorial/>

Seznam obrázků

Obrázek 1-Hierarchie systémů SCADA.....	11
Obrázek 2-Typy PLC od firmy Siemens V moderních průmyslových automatizacích se vyskytují PLC Siemens S7-1200 a vyšší, díky jejich rychlosti a možnosti programování Převzato z: http://automationprimer.com/2014/03/16/siemens-plcs/	18
Obrázek 3-schéma boileru v programu InTouch. Převzato z: http://www.pantek.cz/produkty/intouch/	21
Obrázek 4- Grafické prostředí softwaru PROMOTIC pro vizualizaci uhelné elektrárny Dostupné z: http://www.promotic.eu/cz/appl/img/imgCoalPlant.gif	26
Obrázek 5- Ukázka vizualizace rodinného domu v Reliance – Převzato z: https://www.reliance.cz/cs/products/reliance-demo-applications/building-automation-demo	28
Obrázek 6-Upravení sestavy vzorového příkladu a nastavení parametrů.....	29
Obrázek 7-Nastavení parametrů sestavy pro vyhodnocení dat	29
Obrázek 8- Výsledky souhrnných výpočtů na vzorovém příkladu	30
Obrázek 9-Graf průběhu teplot zobrazené v Reliance– Převzato z: https://www.reliance.cz/cs/products/reliance-demo-applications/building-automation-demo	30
Obrázek 10- Příklad analýzy dat pro firmu Weber- Saint-Gobain, převzato z [45],.....	32
Obrázek 11- Ethernet Wizard, první okno	34
Obrázek 12- Nastavení IP adresy a masky podsítě	35
Obrázek 13- Nastavení počtu přístupů.....	35
Obrázek 14- Nastavení typu spojení a transportního kanálu	36
Obrázek 15- Nastavení ethernet controller.....	37
Obrázek 16- Přidání PLC do mySCADA a nastavení konkrétních parametrů	38
Obrázek 17- Okno pro design aplikace- myDESIGNER.....	39
Obrázek 18- Přidání proměnných do mySCADA.....	39
Obrázek 19- Přístup Pythonu k datům z mySCADA.....	41
Obrázek 20- Vlevo skript pro vypsání základních parametrů, vpravo spuštěný skript.....	42
Obrázek 21- Pole naměřených hodnot vypsanych v Python.....	45
Obrázek 22- Graf časů výsunu pístu.....	45
Obrázek 23-vpravo nástroj pro rychlou analýzu, vlevo výpočet základních parametrů	46
Obrázek 24- Histogram rozdělení četností rychlosti vysunutí pístu včetně poruch s dlouhými intervaly vysunutí.....	47