

**ZÁPADOČESKÁ UNIVERZITA V PLZNI  
FAKULTA ELEKTROTECHNICKÁ**

**KATEDRA TECHNOLOGIÍ A MĚŘENÍ**

# **BAKALÁŘSKÁ PRÁCE**

**Ovládací software pro regulátory průtoku**

**ZADÁNÍ BAKALÁŘSKÉ PRÁCE**  
(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Karel ŠÍMA**  
Osobní číslo: **E11B0166P**  
Studijní program: **B2612 Elektrotechnika a informatika**  
Studijní obor: **Komerční elektrotechnika**  
Název tématu: **Ovládací software pro regulátory průtoku**  
Zadávací katedra: **Katedra technologií a měření**

Z á s a d y   p r o   v y p r a c o v á n í :

1. Prostudujte a popište principy funkce plynových regulátorů průtoku
2. Seznamte se s dokumentací plynových regulátorů od firmy Sierra Instruments a popište jejich základní funkce
3. Dle požadavků naprogramujte ovládací software pro definovanou přípravu plyných směsí
4. Proveďte testovací měření a zhodnoťte výsledky



Rozsah grafických prací: podle doporučení vedoucího  
Rozsah pracovní zprávy: 20 - 30 stran  
Forma zpracování bakalářské práce: tištěná/elektronická  
Seznam odborné literatury:


1. Watson, B.: C# 4.0: řešení praktických programátorských úloh, ISBN:978-80-7413-094-6
2. Troelsen, A.: C# a .NET 2.0 profesionálně, ISBN:80-86815-42-0
3. Dokumentace od firmy Sierra Instruments (dostupná u vedoucího práce)
4. <http://www.sierrainstruments.com/>
5. <http://msdn.microsoft.com/en-us/library>

Vedoucí bakalářské práce: Ing. Petr Kuberský  
Regionální inovační centrum elektrotechniky

Datum zadání bakalářské práce: 14. října 2013  
Termín odevzdání bakalářské práce: 9. června 2014

  
Doc. Ing. Jiří Hammerbauer, Ph.D.  
děkan



  
Doc. Ing. Vlastimil Skočil, CSc.  
vedoucí katedry

V Plzni dne 14. října 2013

## **Abstrakt**

Předkládaná bakalářská práce je zaměřena na řešení problematiky ovládání regulátorů průtoku od společnosti Sierra Instruments. V úvodu práce je popsán obecný princip činnosti hmotnostních regulátorů průtoku a jsou zde uvedeny základní parametry konkrétních přístrojů. Praktická část práce je zaměřena na návrh softwarového projektu a jeho následnou realizaci. Hlavním výstupem této práce je software v jazyce C# nad platformou MS .NET Framework 4.0., který byl naprogramovaný za účelem řízení regulátorů průtoku pro laboratorní použití.

## **Klíčová slova**

Sierra Instruments, kalorimetrický hmotnostní regulátor, SmatrTrak 100, SideTrak 840, C#, MCV, MVVM.

## **Abstract**

The bachelor thesis is focused on development of a software tool for mass flow controllers from Sierra Instruments Company. The first part of the thesis contains description of the operating principle of mass flow controllers and their technical parameters. In this part, electrical specifications and communication protocol were discussed. The main output of the thesis is the software project that was designed and subsequently programmed in programming language C# above the platform MS.NET Framework 4.0.

## **Key words**

Sierra Instruments, calorimetric mass controller, SmatrTrak 100, SideTrak 840, C#, MCV, MVVM.

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně, s použitím odborné literatury a pramenů uvedených v seznamu, který je součástí této bakalářské práce.

Dále prohlašuji, že veškerý software, použitý při řešení této bakalářské práce, je legální.

.....

podpis

V Plzni dne 2.6.2014

Karel Šíma

## **Poděkování**

Tímto bych rád poděkoval vedoucímu bakalářské práce Ing. Petru Kuberskému za cenné profesionální rady, připomínky a metodické vedení práce.

Dále bych také chtěl poděkovat celé mojí rodině, která mě po celou dobu vypracovávání této práce a projektu podporovala a měla se mnou trpělivost.

# Obsah

<b>OBSAH</b> .....	<b>7</b>
<b>SEZNAM SYMBOLŮ A ZKRATEK</b> .....	<b>8</b>
<b>ÚVOD</b> .....	<b>9</b>
<b>1 APARATURA NA MÍCHÁNÍ PLYNNÝCH SMĚSÍ</b> .....	<b>10</b>
1.1 INFORMACE O SIERRA SMARTTRAK 100.....	11
1.1.1 Kalorimetrický hmotnostní průtokoměr .....	12
1.1.2 Technické parametry Sierra SmartTrak 100.....	13
1.1.3 Komunikace Sierra SmartTrak 100.....	14
1.1.4 Připojení zařízení k PC.....	15
1.2 INFORMACE O SIERRA SIDETRAK 840.....	16
1.2.1 Technické parametry Sierra SideTrak 840.....	16
<b>2 PROGRAMÁTORSKÁ DOKUMENTACE</b> .....	<b>20</b>
2.1 ZADÁNÍ PROBLÉMU.....	20
2.2 OBECNÝ PROBLÉM.....	20
2.3 ANALÝZA PROBLÉMU .....	21
2.3.1 Zvolení vývojové platformy.....	21
2.3.2 Datové struktury.....	22
2.4 KOMUNIKACE .....	22
2.5 KOMUNIKAČNÍ PŘÍKAZY ZAŘÍZENÍ SMARTTRAK 100.....	23
2.5.1 Struktura komunikačních příkazů .....	23
2.5.2 Přehled komunikačních příkazů.....	24
2.6 KOMUNIKACE SIDETRAK 840.....	30
2.6.1 Struktura komunikace .....	30
2.6.2 Komunikační příkazy .....	31
2.7 NÁVRH APLIKACE .....	32
2.7.1 Třída SerialComu.....	32
2.7.2 Třída FTDIManager .....	33
2.7.3 Třída FlowDriver.....	33
2.8 ARCHITEKTURA .....	33
2.8.1 Jádro - CORE .....	34
2.8.2 Grafické rozhraní - GUI .....	35
2.9 CHOD PROGRAMU .....	35
<b>3 MĚŘENÍ</b> .....	<b>38</b>
3.1 OVĚŘENÍ FUNKCIONALITY – TEST 1 .....	38
3.2 TESTOVÁNÍ NA VZORCÍCH – TEST 2.....	39
<b>ZÁVĚR</b> .....	<b>41</b>
<b>SEZNAM LITERATURY A INFORMAČNÍCH ZDROJŮ</b> .....	<b>42</b>
<b>PŘÍLOHY</b> .....	<b>1</b>



## Seznam symbolů a zkratek

CORE .....	Jádro aplikace
CR .....	Carriage Return – znaková hodnota
CRC .....	Cyklický redundantní součet (z angl. Spojení Cyclic Redundancy Check)
CRL.....	Common Language Runtime – virtuální stroj MS .NET Frameworku
DLL.....	Dynamic-link library
D-SUB .....	D-subminiature – kategorie konektorů
FTDI.....	Future Technology Devices International – společnost vyrábějící čipy
GUI .....	Grafické uživatelské rozhraní
LFE .....	Laminární člen
LSB .....	Least significant bit – „nejméně“ důležitý bit
MCV .....	Model Control View – architektonický vzor
MSB.....	Most significant bit – „nejvíce“ důležitý bit
MVVM.....	Model View ViewModel – architektonický vzor
OS .....	Operační Systém
POCO.....	Plain Old CLR Object – objektová konstrukce
RS-232 .....	Sériový port
RxD.....	Receive data – příjem dat
RTD .....	Odporový teplotní senzor
cc/m .....	Standartní kubický centimetr
TxD.....	Transmit data – vysílání dat
USB.....	Universal Serial Bus
WPF .....	Windows Presentation Foundation

## Úvod

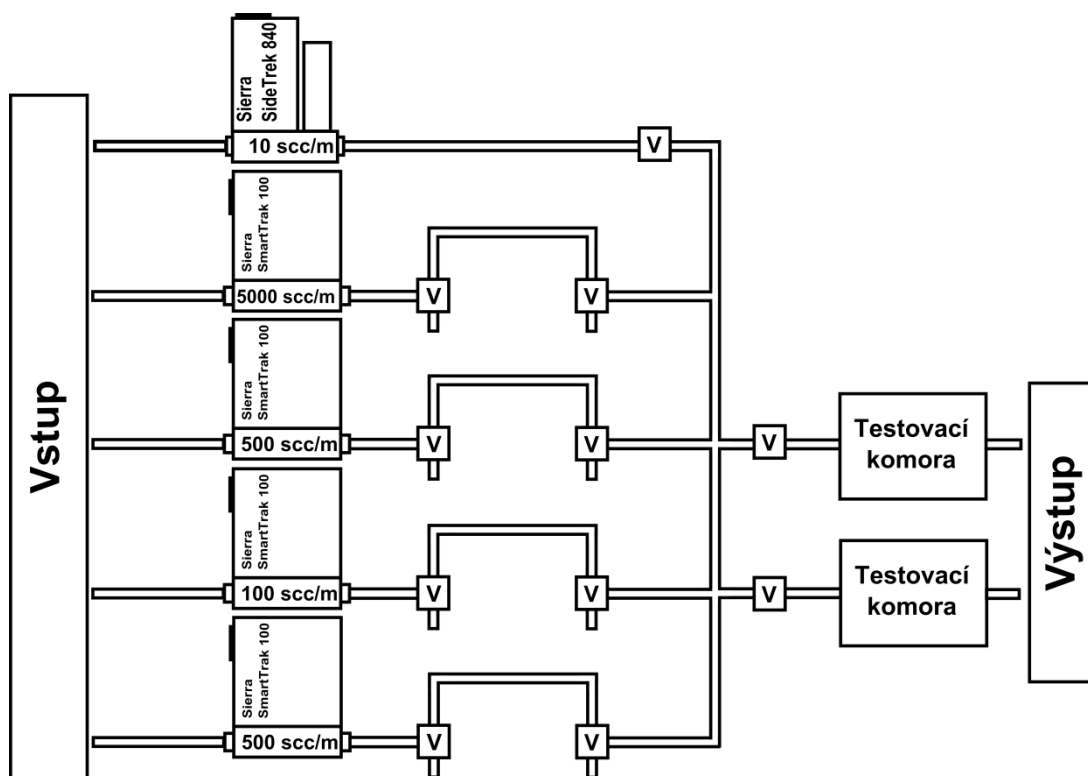
V dnešní době je automatizace prakticky nedílnou součástí všech průmyslových odvětví, kde výrazně přispívá k efektivitě provozu, výroby či testování. Testování a automatizace měřicích postupů je úzce spojena nejen s průmyslovou ale i akademickou sférou. Systematické testování širokého spektra nových materiálů a efektivní zpracování dat se stává nezbytnou součástí pro úspěšné uplatnění výzkumných týmů v daných oblastech.

V dnešní době existuje řada platforem pro automatizaci měřicích přístrojů, které jsou podporovány řadou výrobců. Tím umožňují poměrně snadnou automatizaci měřicích přístrojů pomocí grafického programového prostředí. Příkladem za všechny je Labview. Úskalím podobných prostředí typu Labview může být podpora zařízení, která je dána právě výrobcem přístrojů anebo standardizovaným komunikačním protokolem. Ovšem u specializovaných přístrojů nemusí být jednoduché aplikovat standardizované konvence z důvodu jedinečnosti daných přístrojů. Dalším úskalím může být požadavek na specifické funkce či specifické chování programu. Pro univerzální prostředí pak jednoduchý a zároveň specifický požadavek může znamenat komplikované a zdlouhavé vytváření řešení. U specializovaných přístrojů může být dále problém s nedostačujícím či příliš jednoduchým softwarem od výrobce, jenž je často pro specifické účely v akademické sféře prakticky nepoužitelný.

Příkladem podobného problému bylo budování aparatury určené na míchání směsí plynů, která byla založena na hmotnostních regulátorech průtoku. K hmotnostním regulátorům byl výrobcem dodán základní software, který ale nevyhovoval potřebnému použití. V případě podobných situací je nutno navrhnout a realizovat projekt pro konkrétní aplikaci, jako je tomu v případě této bakalářské práce. Cílem této bakalářské práce je realizovat ovládací software pro aparaturu určenou na míchání plyných směsí a zefektivnit tak testování nových plynových senzorů vyvíjených na Fakultě elektrotechnické Západočeské univerzity v Plzni.

## 1 Aparatura na míchání plyných směsí

Aparatura slouží k přípravě plyných směsí v různých koncentracích. Tyto směsi plynů jsou následně používány k testování vyvíjených senzorů. U nově vyvíjených senzorů se testuje řada parametrů, z nichž nejzákladnějším je citlivost na různé plynné látky. Základ aparatury tvoří pět hmotnostních regulátorů průtoku od společnosti Sierra Instruments. Konkrétně se jedná o čtyři přístroje Sierra SmartTrak 100 a jeden přístroj Sierra SideTrak 840. Tyto přístroje jsou hlavními prvky aparatury na míchání plyných směsí. K přístrojům je připojeno nerezové potrubí, na kterém jsou umístěné přepínací ventily. Tyto ventily jsou manuální a umožňují rychlou a snadnou modifikaci míchací aparatury. Potrubí, která vedou z jednotlivých přístrojů, jsou na konci spojena do dvou výstupních uzlů. Oba výstupní uzly jsou připojeny k testovacím komorám. Tyto komory jsou utěsněny a obsahují pouze vstupní a výstupní otvor a dále vývody kontaktů pro testované vzorky. Na *Obr. 1.1* je vidět schématický náčrt míchací aparatury, písmenem V jsou označeny přepínací/zavírací ventily.

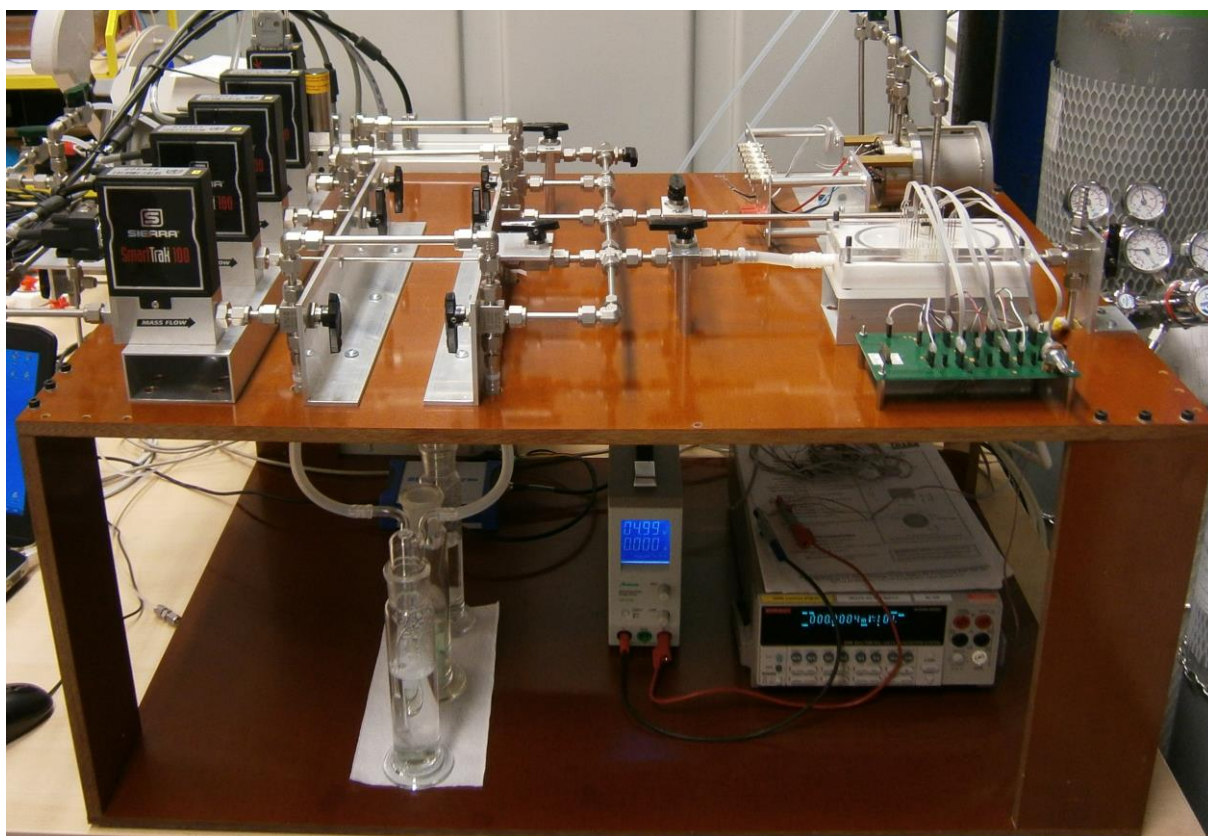


Obr 1.1 Schématický náčrt aparatury pro míchání plyných směsí

Ze schématického náčrtu je patrné umístění jednotlivých hmotnostních regulátorů Sierra SmartTrak 100. Dva přístroje mají maximální regulovaný průtok 500 scc/m (standard cubic centimetr per minute – standartní kubický centimetr za minutu), jeden přístroj má maximální

regulovaný průtok 5000 scc/m a poslední přístroj má maximální regulovaný průtok 100 scc/m. Dále je z nákresu patrné, že v mixovací aparatuře je instalován jeden přístroj Sierra SideTrak 840 s maximálním průtokem 10 scc/m. Přístroj SideTrak 840 s maximálním průtokem 10 scc/m a přístroj SmartTrak 100 s maximálním průtokem 100 scc/m jsou v aparatuře začleněny z důvodu používání agresivních a toxických plynů, jako jsou například  $\text{NO}_2$ ,  $\text{NH}_3$  a  $\text{Cl}$ . Právě z tohoto důvodu, že tyto přístroje budou pracovat s agresivními plyny, mají nízké maximální regulované průtoky. Zbylé přístroje budou pracovat s běžnými neagresivními plyny, jako jsou:  $\text{O}_2$ ,  $\text{N}_2$ ,  $\text{CO}_2$ ,  $\text{CO}$  apod.

Pro bližší představu je na *Obr. 1.2* zachyceno reálné uspořádání hmotnostních regulátorů v realizované aparatuře.



*Obr 1.2 Aparatura na míchání plyných směsí*

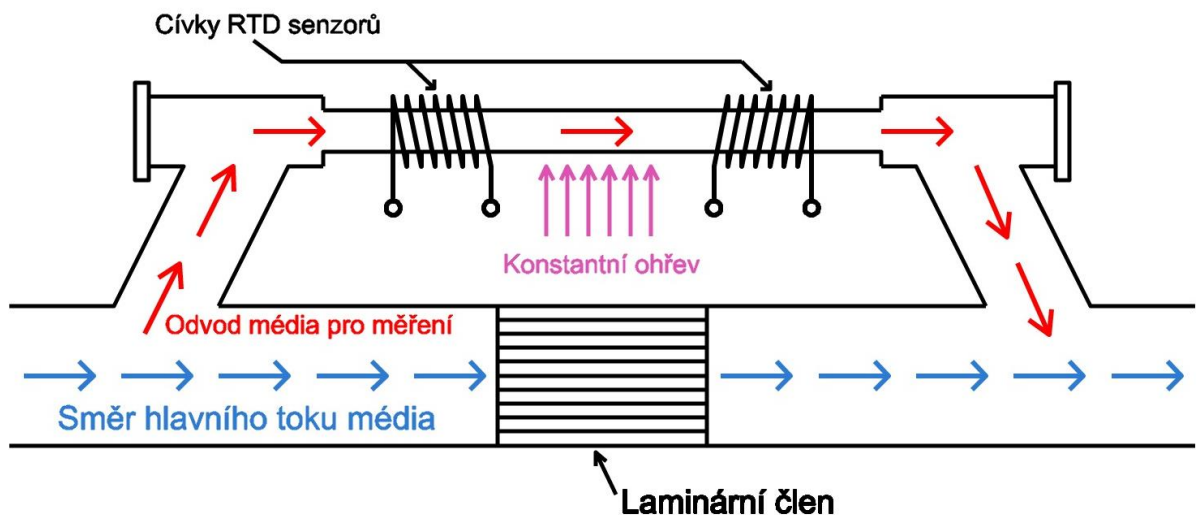
## 1.1 Informace o Sierra SmartTrak 100

Přístroje společnosti Sierra Instruments typové řady SmartTrak 100 jsou regulátory průtoku. Obsahují v sobě měřicí a akční člen. Měřicí člen je v tomto přístroji realizován jako kalorimetrický hmotnostní průtokoměr. Jako akční člen výrobce použil elektronicky řízený

ventil. Vše řídí a reguluje vnitřní elektronika přístroje, která umožňuje také ovládání tohoto regulátoru pomocí komunikačního portu. [1]

### 1.1.1 Kalorimetrický hmotnostní průtokoměr

Na Obr. 1.3 je zobrazen princip kalorimetrického hmotnostního průtokoměru.



Obr 1.3 Schéma kalorimetrického hmotnostního průtokoměru [1]

Z Obr. 1.3 je vidět, že na vstupu měřicího členu je z hlavního potrubí odvedena část protékajícího média do malé kapiláry. Velikost kapiláry je v určitém poměru k hlavnímu potrubí, což zajišťuje poměrné rozdělení průtoku. Dále je využito laminárního členu (LFE – Laminar Flow Element), aby byl poměr konstantní. Tento laminární člen je těleso složené z několika přepážek, které pomáhají upravit přesně poměr průtoků mezi hlavním potrubím a kapilárou. [1][2]

Po oddělení průtoku v poměrné části do kapiláry, dochází na kapiláře k vlastnímu měření průtoku. Měření průtoku je realizováno dvěma teplotními senzory RTD (RTD – Resistance Temperature Detector). Mezi těmito senzory je umístěn vyhřívací prvek, který zahřívá médium v kapiláře na konstantní hodnotu. Hodnota teploty, na kterou se médium zahřívá, je dána výrobcem. Měření začíná změřením hodnoty teploty na začátku kapiláry, pomocí prvního RTD senzoru. Poté následuje ohřev a na konci kapiláry se změří hodnota teploty pomocí druhého RTD senzoru. Teplota média v celé kapiláře má určitou úměrnost ke změně rychlosti proudění tohoto média. Pokud dojde ke změně rychlosti průtoku, dojde také ke

změně odebíraného tepla procházejícím médiem. Tuto změnu tepla sleduje druhý RTD senzor. Z rozdílu hodnoty konstantního ohřevu a hodnot prvního a druhého RTD senzoru dokáže následně řídicí elektronika přístroje spočítat velikost průtoku. Přesnost tohoto principu závisí na použitých senzorech teploty, přesnosti vyhřívacího prvku a strojního zpracování potrubí a kapiláry. Dále je přesnost závislá na přesném určení konstant a korekčních hodnot pro výpočet průtoku.[1]

### 1.1.2 Technické parametry Sierra SmartTrak 100

Přístroje Sierra ovšem obsahují ve své vnitřní programové paměti dalších devět korekčních koeficientů a konstant. To umožňuje provoz regulátoru i s jiným plynným médiem než na jaký byl kalibrován. Sierra SmartTrak 100 umožňuje pracovat s těmito plyny:

- Vzduch
- Argon (Ar)
- Oxid uhličitý (CO<sub>2</sub>)
- Oxid uhelnatý (CO)
- Hélium (He)
- Vodík (H)
- Metan (CH<sub>4</sub>)
- Dusík (N)
- Oxid dusný (N<sub>2</sub>O)
- Kyslík (O<sub>2</sub>)

Přístroje je možno využívat i s jinými plynnými médii za předpokladu, že vypočteme korekční koeficient zvaný K-faktor. [1]

Upevnění přístroje k potrubí je technicky řešeno pomocí šroubení 1/4“ a svěrných spojek SWAGELOK. Toto upnutí zajišťuje utěsnění vstupního i výstupního spoje až do tlaku 220 barů. [1]

Pro správnou funkci je nutné splnit požadovaný typ napájení. U přístrojů SmartTrak se jedná o zdroj dodávající 24 V s tolerancí  $\pm 10\%$  a proudovým odběrem až 700 mA. [1]

### 1.1.3 Komunikace Sierra SmartTrak 100

Komunikace je realizována pomocí konektoru DE-15, což je 15 pinový konektor D-SUB velikosti E. Na těle přístroje je také umístěn konektor RJ-45. Tento konektor by mohl vytvářet mylnou myšlenku, že zařízení umí komunikovat i pomocí síťového rozhraní. Avšak přístroj touto funkcí nedisponuje. Konektor RJ-45 zde neslouží ke komunikaci, ale výrobce pomocí něj připojuje k základnímu přístroji zobrazovací modul s klávesnicí pro manuální ovládání. Pro komunikaci s ovládacím softwarem se využívá konektor DE-15, který umožňuje dva typy komunikace. Na *Obr. 1.4* je schéma zapojení pinů v konektoru DE-15 společně s popisem jejich funkce. [1]



Obr 1.4 Zapojení pinů v konektoru DE-15 [1]

Jak je patrné z výčtu funkcí jednotlivých pinů, konektor DE-15 umožňuje ovládat zařízení pomocí sériového rozhraní. Sériové rozhraní je zajištěno pomocí pinů číslo 7 a 13. Pin číslo 7 – RS - 232 Transmit je vysílání dat do softwaru. Pin číslo 13 – RS - 232 Receive je přijímání dat ze softwaru. Tyto piny poskytují datové signály oproti zemnímu potenciálu, který je zapojen na piny 1, 3 a 10. Konkrétní použití pinů zemního potenciálu není nijak stanoveno. Výrobce pouze uvádí, že piny zemního potenciálu jsou spojeny do jednoho uzlu uvnitř přístroje. Z toho plyne, že není vhodné je vně přístroje opět spojovat do dalšího uzlu, aby nedošlo k vytvoření smyček mezi vnějším a vnitřním uzlem. Sériové rozhraní je tedy prvním způsobem komunikace s přístrojem. Po sériové lince se přijímají data a zároveň posílají nastavovací příkazy. Všechna nastavení a čtení tedy probíhají digitálně. Přístroj zasílá data po sériovém rozhraní v několika režimech. První režim (Stream On) neustále zasílá

nastavené informace. Druhý režim (Stream Off) zasílá data pouze na dotázání. Třetí režim (Stream Echo) zasílá data na dotazování a po zadání nastavovacího příkazu jako zpětnou vazbu. [1]

Další možností jak komunikovat s přístrojem je čistě analogově. U tohoto přístroje jsou dvě možnosti analogové komunikace a to pomocí napěťové hladiny nebo pomocí proudové smyčky. Pomocí napěťové hladiny nebo proudové smyčky lze nastavovat a zpětně také získávat hodnotu regulovaného průtoku. Napěťová hladina pro získání hodnoty průtoku je realizována pinem 2 proti jednomu pinu zemního potenciálu (pin 1, 3, 10). Nastavovací napěťová hladina je realizována pinem 8 oproti pinu zemního potenciálu (pin 1, 3, 10). Proudová smyčka pro získání hodnoty průtoku je vyvedena na pin 14 a opět je nutné jí použít proti jednomu pinu zemního potenciálu (pin 1, 3, 10). Nastavování pomocí proudové smyčky je možné na základě požadavku při objednávce zařízení. Zařízení má pak změněný pin 10 ze zemního potenciálu na vstup pomocí proudové smyčky. Nastavovací proudová smyčka je tedy realizována pinem 10 a jedním pinem zemního potenciálu (pin 1,3). Dále lze analogově ovládat nastavení režimu ventilu. Ventil umožňuje tři režimy, jsou to režimy „Auto“, „Close“ a „Purge“. Přístroj ve výchozím stavu pracuje v režimu „Auto“, to znamená automatické držení hladiny podle nastaveného průtoku. V případě, že přivedeme referenční napětí z pinu 11 na pin 12, stane se, že ventil se uzavře a je v režimu „Close“. Po odpojení referenčního napětí se zpět přístroj vrací do režimu „Auto“. Touto cestou jsme schopni analogově ovládat otevření a zavření ventilu. Pokud ovšem přivedeme referenční napětí z pinu 11 na pin 4, přístroj začne pracovat v režimu „Purge“. Režim „Purge“ je jednoduše řečeno propláchnutí. Během tohoto režimu dojde k maximálnímu otevření ventilu a přístrojem prochází maximální průtok bez jakéhokoliv řízení. Průtok je tedy vyšší než deklarovaný maximální regulovaný průtok. Výrobce ve své dokumentaci uvádí, že režim „Purge“ by se neměl používat příliš často a po dlouhé intervaly. Je to z důvodu, aby nedošlo k poškození vnitřních mechanických částí přístroje. Po odpojení referenčního napětí z pinu 4 se opět přístroj vrací zpět do režimu „Auto“. [1]

#### 1.1.4 Připojení zařízení k PC

Přístroj umožňuje digitální komunikaci a řízení pouze přes sériové rozhraní, což by mohl být v současné době zdánlivě problém. Protože prakticky vymizely sériové porty ze základní výbavy počítačů a notebooků. Tento problém je ovšem snadno řešitelný pomocí RS232/USB převodníků. Pro převodníky připojující přístroje míchací aparatury bylo využito čipů firmy



FTDI. Pro míchací aparaturu byl konkrétně použit čip FT232. Jedná se o čip, který je velmi často využívaný k řešení tohoto problému. Převodníky využívané pro komunikaci s přístroji dále obsahují čip MAXIM MAX232, což je čip, který upravuje napěťové hladiny na standard sériového rozhraní RS-232. Jelikož se jedná však o připojení celkem čtyřech zařízení Sierra SmartTrak 100, bylo tedy využito čtyř převodníků připojených do USB hubu, který se následně připojuje pomocí USB do řídicího počítače či notebooku. Tímto řešením se výrazně zjednodušila manipulace a připojování měřicí aparatury k počítačovému vybavení.

## 1.2 Informace o Sierra SideTrak 840

Přístroj Sierra SideTrak 840 je typově podobné zařízení jako Sierra SmartTrak 100. Přístroj Sierra SideTrak 840 je ale uzpůsoben pro průtok agresivních a toxických plynů. To znamená především zvýšenou odolnost proti plynům, které jsou vysoce korozivní. Přístroj je k provozu s těmito plyny uzpůsoben konstrukčně pomocí použitých materiálů. Vnitřní komponenty jsou vyrobeny z nerezové oceli a dále je konstrukce osazena teflonovými prvky a teflonovou výstelkou ventilu. [3]

Přístroj Sierra SideTrak 840 má také funkci regulátoru průtoku plynu jako přístroj Sierra SmartTrak 100. Jedná se tedy rovněž o přístroj kombinující měřicí a akční člen. Měřicí člen je shodně jako u SmartTrak 100 realizován kalorimetrickým hmotnostním průtokoměrem. Akční člen je shodně realizován elektronicky ovládaným ventilem. Hlavní předností přístroje Sierra SideTrak 840 je uzpůsobení průtokoměru a ventilu na velmi malé průtoky. Ventil je na bázi elektromagnetického řízení. V míchací aparatuře je začleněn přístroj Sierra SideTrak 840 s rozsahem průtoku 0 – 10 scc/m. [3]

### 1.2.1 Technické parametry Sierra SideTrak 840

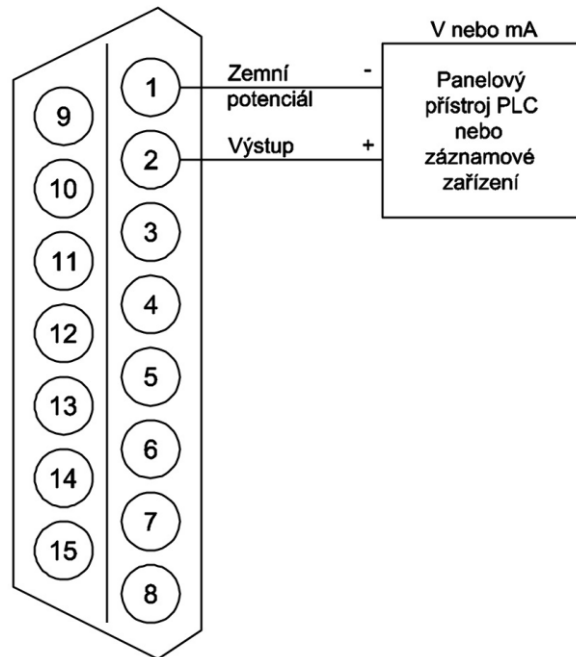
Přístroj, který je instalován v mixovací aparatuře, byl kalibrován na plyn NO<sub>2</sub> o koncentraci 100 ppm. Z důvodu velmi malého rozsahu a nutnosti vysoké přesnosti při měření i regulaci tento přístroj neumožňuje pracovat s více plynnými médii jako Sierra SmartTrak 100. [3]

Přístroj je též upevněn v aparatuře k potrubí pomocí svěrných svorek SWAGELOK.



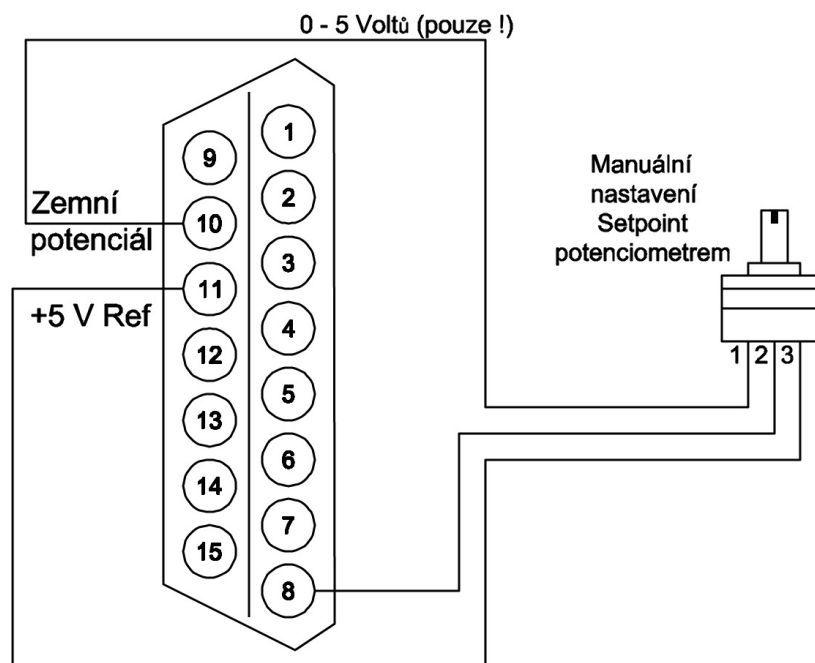
Obr 1.5 Konektor přístroje Sierra SideTrak 840 [3]

Přístroj Sierra SideTrak disponuje pouze analogovým ovládáním. To je podobné analogovému ovládání použitého u zařízení Sierra SmartTrak 100. Připojení ovládání je realizováno pomocí konektoru rodiny D-SUB velikosti DA-15. Připojení pinů a jejich funkce je zobrazena v *Obr. 1.5*. Výstup z průtokoměru je hodnota hladiny napětí, která udává procentuální hodnotu průtoku v scc/m z maximálního regulovatelného průtoku. Tato hodnota je realizována za pomoci pinu 2 – „Výstupní signál“ a pinu 1 – „Signálová zem“. Výstupem je napětí 0 – 5 V. Hodnota průtoku pak odpovídá procentuální hodnotě 0 - 100% pro 0 – 5 V. Další možností je proudový výstup. Proudový výstup lze vytvořit připojením vhodné odporové zátěže a odečítáním změn proudu ve větvi zátěže. Výrobce je doporučován jako referenční zátěž odpor s hodnotou 1 k $\Omega$ . Takto vznikne měřicí proudová smyčka a výstupní proud se poté měří v rozsahu 4 - 20mA. Na *Obr. 1.6* je schématické znázornění zapojení pro vyčítání hodnoty proudu nebo napětí odpovídající danému průtoku. [3]



Obr 1.6 Zapojení pinů pro výstup na přístroji Sierra SideTrak 840 [3]

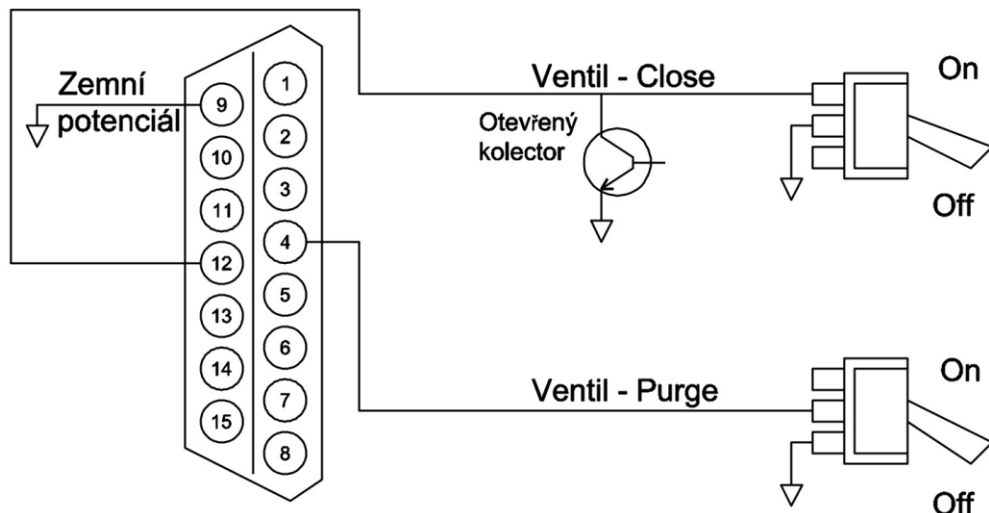
Pro nastavování napěťové hladiny průtoku (nastavení hodnoty průtoku) se využívá pinu 8 – „SetPoint“ oproti pinu 10 – „Signálová zem“. Dále se využívá referenční hodnoty +5 V pro rozsah 0 - 5 V z pinu 11. Schéma zapojení je zobrazeno na *Obr. 1.7*. [3]



Obr 1.7 Zapojení pinů pro vstup na přístroji Sierra SideTrak 840 [3]

Ovládání režimů ventilu je realizované pomocí pinů 4 a 12. Režim „Close“ – uzavření ventilu se aktivuje přivedením napájecího napětí na pin 12 – „Valve close“ a spojení

pinu 9 - „Zem napájení“ se zemním potenciálem. Režim „Purge“ – průplach se aktivuje podobným způsobem, na pin 4 se připojí napájecí napětí a pin 9 je spojen se zemním potenciálem. Pokud jsou piny 4 a 12 odpojeny, přístroj pracuje ve výchozím režimu „Auto“. Schéma zapojení je na *Obr. 1.8*. [3]



*Obr 1.8 Zapojení pinů pro zapnutí Close a Purge na přístroji Sierra SideTrak 840 [3]*

Výše bylo uvedeno, jakým způsobem se ovládají přístroje z hlediska hardwaru. Přístroj není ovládán přímo hardwarově za pomoci voltmetru, přepínače a potenciometru, ale je ovládán pomocí modulu s mikropočítačem PIC a softwaru. Tento modul obsahuje jeden D/A převodník pro nastavování průtoku. Tento převodník je 8 bitový. Dále modul obsahuje jeden A/D převodník, který je 12 bitový. Převodník slouží k odečítání hodnot z výstupu regulátoru. Dále jeden FTDI čip, který je připojen k sériovému rozhraní mikropočítače PIC a zajišťuje komunikaci s PC. Připojení k PC je podobné jako u přístrojů Sierra SmartTrak 100, tedy za pomoci RS-232/USB převodníku reprezentovaným čipem FTDI FT232. [3]

## 2 Programátorská dokumentace

### 2.1 Zadání problému

Předmětem zadání této práce bylo vytvoření uživatelského ovládacího softwaru pro hmotnostní regulátory průtoku popsané v kapitolách 1.1 a 1.2. Vzhledem k tomu, že aparatura na míchání plyných směsí (viz *Obr. 1.1* v kapitole 1) bude sloužit k vývoji nových typů senzorů, uživatelský software měl být navržen s důrazem na maximální univerzálnost, jednoduchost a přehlednost ovládacích prvků. Software bude zpravidla pracovat pod operačním systémem Windows 7, nicméně v rámci univerzálnosti byla rovněž požadována kompatibilita s Windows XP. V rámci této práce byly stanoveny následující požadavky, které by měly být poskytovány prostřednictvím hlavního okna grafického rozhraní softwaru:

1. Možnost aktivace/deaktivace jednotlivých hmotnostních regulátorů.
2. Možnost sledování aktuálních průtoků u aktivovaných regulátorů (formou zobrazeného číselného údaje a případně i grafu).
3. Možnost přímého nastavení hodnoty průtoku pro každý zvolený regulátor.
4. Možnost přepínání mezi třemi základními režimy činnosti regulátoru průtoku (AUTO / CLOSE / PURGE).
5. Možnost nastavení předdefinovaného časového profilu průtoku pro jednotlivé regulátory (načtení dat ze souboru a případně jednoduchý generátor časových funkcí).
6. Průběžné ukládání hodnot průtokoměrů od každého regulátoru na persistentním médiu (ideálně soubory).

### 2.2 Obecný problém

Důvodem vzniku požadavku na vývoj tohoto softwaru byl nekomfortní a příliš jednoduchý software dodávaný výrobcem. Výrobce ke svému zařízení dodává program, který umožňuje jen manuální nastavování. Tento přístup by byl použitelný pouze pro nastavení jedné časově neproměnlivé koncentrace mixované směsi. Aparatura ale byla vytvořena za účelem automatizace a zefektivnění postupu při testování nově vyvíjených organických

senzorů citlivých na plynné látky. Toto laboratorní nasazení vyžaduje časově proměnlivé koncentrace různých plynů případně směsí plynů. Při použití softwaru dodávaného výrobcem by to znamenalo, že by jeden pracovník musel neustále sedět u ovládacího PC a měnit hodnoty v požadovaném čase. Tento přístup je ovšem nejen neefektivní, ale také neproveditelný. V současné době je aparatura osazena celkově pěti přístroji a každý přístroj by se ovládal jedním programem. Jeden pracovník by tudíž v ideálním stavu musel ovládat pět programů zároveň. Dalším problémem je, že obsluhující pracovník by musel u ovládacího PC strávit například 5, 6 nebo v případě dlouhodobých testů i více než 10 hodin, což je velmi nepraktické. Neustálá přítomnost by byla rovněž nutná, kdyby byly požadovány malé změny v průtocích v malých časových intervalech apod.

Hlavními problémy tedy byly: nemožnost automatického nastavování průtoku u jednotlivých zařízení v závislosti na čase, ovládání jednoho přístroje pomocí jednoho programu a absence záznamu dat o průběhu nastavené hodnoty průtoku. Problém s nemožností ovládat více průtokoměrů z jednoho programu má za následek spuštění více programů pro řízení průtoku celé aparatury a velkou nepřehlednost.

## **2.3 Analýza problému**

### **2.3.1 Zvolení vývojové platformy**

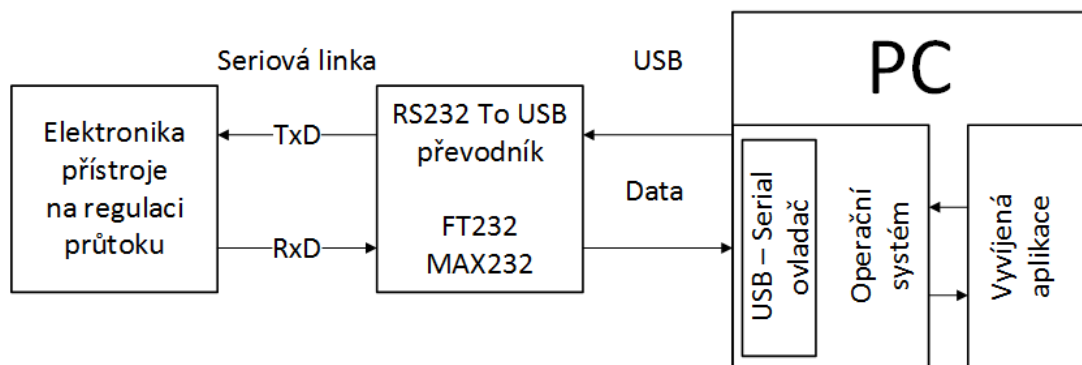
V předchozí kapitole byl definován hlavní problém a v závislosti na požadavcích na softwaru uvedených v kapitole 2.1 byla provedena analýza problému a volba použitých technologií. Jako vývojová platforma byla zvolena platforma Microsoft .NET Framework a jazyk pro vývoj C#. Platforma Microsoft .NET C# byla zvolena z důvodu chodu aplikace nad systémy Microsoft Windows NT 5.1 (Windows XP) a vyšší. Platforma Microsoft .NET C# má hlavní výhodu, že má plnou podporu a propojení se systémy Windows NT. Kompatibilita se systémy Windows NT 5.1 byla zajištěna využitím starší verze .NET Framework 4.0. Použití této verze zajistilo kompatibilitu se systémy NT 5.1. Bohužel toto také znemožnilo využívat nové programovací techniky a metody. Tyto techniky musely být vynechány a funkcionality musela být zajištěna staršími přístupy programování a jinými programátorskými strukturami.

### 2.3.2 Datové struktury

Pro základní návrh aplikace byl použit objektivě orientovaný přístup. Z důvodu objektivého návrhu aplikace a využití architektonických vzorů byly zvoleny POCO třídy pro reprezentaci dat v paměti. POCO je anglická zkratka pro Plain Old CLR Object. Jedná se o třídu, která obsahuje jen proměnné a zapouzdření je provedeno pomocí „properties“. POCO třídy zpravidla neobsahují žádné funkční metody. POCO třídy v tomto projektu reprezentují jednotlivé objekty potřebné pro chod programu. Tyto třídy obsahují jen proměnné hodnotových datových a referenčních datových typů. Mezi referenčními datovými typy převažují standartní kolekce a vytvořené POCO objekty. Všechny kolekce, které byly použity, jsou generické, což znamená silnou typovou kontrolu vstupních dat. V softwaru se jedná především o kolekce „List“ a „LinkedList“. U kolekcí „List“ byly z převážné většiny použity jako datové typy struktury uchovávající jednoduchá dvou a tří hodnotová data. Struktury byly zvoleny s ohledem na jednoduchost těchto dat. Tyto data jsou totiž hodnoty typu int a double. Pro hodnotové datové typy je lepší, když jsou uloženy přímo ve „stacku“ a nikoliv ve „stacku“ a „heap“. Uložení do „stacku“ a „heap“ je totiž vytváření reference na referenci hodnoty. Toto vzniká při použití objektivého datového typu obsahující hodnotové datové typy.

### 2.4 Komunikace

Komunikace u zařízení SmartTrak 100 je stanovena výrobcem. U zařízení SideTrak 840 byla komunikace domluvena a naprogramována při vývoji komunikační desky. Komunikace SideTrak byla stanovena na bázi komunikace SmartTrak 100 s ohledem na dostupnou funkcionalitu zařízení. Přesné komunikační příkazy jsou popsány v následujících kapitolách 2.5.2 a 2.6.2.



Obr. 2.1 Schéma komunikace mezi aplikací a přístrojem

Na obrázku diagramu *Obr. 2.1* je vidět princip komunikace mezi zařízením a ovládacím PC. Princip komunikace je stejný jak u zařízení typu SmartTrak 100, tak i u zařízení typu SideTrak 840. Komunikace probíhá z aplikace přes OS (Operační Systém), přičemž OS musí mít nainstalován ovladač pro čip FT232. Pokud OS tento ovladač detekuje a je k PC připojen RS232/USB převodník založený na čipu FT232, OS ho detekuje a umožní na úrovni OS komunikovat s tímto zařízením jako se sériovým portem. Tuto obsluhu si vykonává OS sám a dále se pracuje s regulátory přes sériovou linku, která je vytvořena pomocí připojeného převodníku.

## 2.5 Komunikační příkazy zařízení SmartTrak 100

### 2.5.1 Struktura komunikačních příkazů

Níže uvedené komunikační příkazy pro sériovou komunikaci jsou pro verzi firmwaru v2.044. Sériová komunikace je prováděna pomocí komunikačních příkazů, které mají přesně definovanou strukturu. Tato struktura je zobrazena v rovnici ( 2.1 ). [1]

$$\begin{aligned} \text{Příkaz} = \{ & 1 \text{ byte pro rozlišení dotazu a zápisu} + 4 \text{ příkazové byt} \\ & + \text{proměnlivý počet bytů pro hodnotu} + 2 \text{ byty CRC} + \text{CR} \} \end{aligned} \quad (2.1)$$

Dále je výrobcem specifikovaná maximální délka příkazu. Ta je stanovena maximálně na 26 bytů. Výše uvedená zkratka CRC reprezentuje „Cyclic redundancy check“ což znamená cyklický redundantní součet. Jedná se o jednoduchý, ale přesto velice účinný a rychlý způsob určení datové integrity přenášených dat a příkazů po sériové lince. Výrobce ve své dokumentaci předepisuje výpočet CRC a tento předpis výpočtu je přiložen v příloze B. Celý příkaz je ukončen CR, což je znaková hodnota zvaná „Carriage Return“ (jedná se o jeden znak). [1]

Příchozí komunikace je stejná, jen je vynechán první byte (znak) pro rozlišení dotazu a zápisu, jelikož tento znak by byl zbytečný.

Výrobce k zařízení poskytuje dokumentaci k příkazům, která obsahuje tzv. „bezpečné příkazy“. Bezpečné příkazy jsou níže vyjmenovány v kapitole 2.5.2, jsou to příkazy pro základní obsluhu, které nemohou poškodit přístroj či provést uživatelsky neměnné přenastavení přístroje. Dále přístroje umí obsluhovat mnoho dalších příkazů. Ovšem výrobce



o nich neinformuje právě z důvodu ochrany funkcionality a zachování nastavení chodu a kalibračních hodnot průtokoměru.

## 2.5.2 Přehled komunikačních příkazů

Příkazy jsou popisovány pro verzi softwaru 2.044 přístrojů SmartTrak 100 verze 2. Starší přístroje verze 1 mají jiné komunikační příkazy a jinou strukturu příkazů. Ty ale nebyly používány. [4]

- FLOW

FLOW je příkaz, který vrací hodnotu průtoku. Nejedná se o nastavovací příkaz. Příkaz je vrácen zařízením ve STREAM režimu. STREAM režim představuje neustálé zasílání dat z přístroje přes sériovou komunikaci. [4]

Přijímaný příkaz:

“Flow + hodnota + CRC + CR“

- SETPOINT

SETPOINT nastavuje hodnotu udržovaného průtoku. Hodnota se v zařízení ukládá do paměti typu flash. Uložení do paměti flash zajišťuje možnost použít uloženou hodnotu i po odpojení a opětovném připojení napájení. Tento příkaz ovšem není doporučován pro přístroje od verze 2 a vyšší. Není doporučován z důvodu jiné práce s paměťmi uvnitř přístroje. Z tohoto důvodu byl nahrazen příkazem SETPOINT\_FLASH. Své využití při komunikaci stále má a to jako součást výčtu dat pomocí čtecího příkazu SYNC, který postupně vrací dostupné nastavené parametry z paměti přístroje. [4]

Přijímaný příkaz:

“Sinv + hodnota + CRC + CR “

- UNIT INDEX

UNIT INDEX vrací a nastavuje typ jednotky, ve kterých vrací přístroj údaje. Návrátová hodnota je v rozsahu 1 až 30. Tabulka s přehledem

podporovaných jednotek je uvedena v příloze A. Podle nastavené jednotky vrací přístroj všechny hodnoty týkající se průtoku v nastavené jednotce. [4]

Příkaz pro čtení:

“?Unti + CRC + CR“

Příkaz pro zápis:

“!Unti + index jednotky + CRC + CR“

Přijímaný příkaz:

“Unti + index jednotky + CRC + CR“

- SETPOINT\_FLASH

SETPOINT\_FLASH je příkaz, který v nových verzích přístrojů nahrazuje příkaz SETPOINT. Tento příkaz po zaslání čtecího příkazu vrací aktuální hodnotu pro nastavování průtoku uloženou v paměti flash. Pomocí zápisového příkazu se hodnota pro nastavování průtoku nastaví a aktivuje. [4]

Příkaz pro čtení:

“?Setf + CRC + CR“

Příkaz pro zápis:

“!Setf + hodnota + CRC + CR“

Přijímaný příkaz:

“Setf + hodnota + CRC + CR“

- SETPOINT RAM

SETPOINT RAM vrací a nastavuje aktuální hodnotu nastavovaného průtoku v RAM paměti. Pokud nastavíme tímto příkazem SETPOINT hodnotu nastavovaného průtoku a dojde k přerušení napájení a následně k jeho obnově, nedojde-li již k obnově nastavované hodnoty, která byla nastavena pomocí tohoto příkazu. Přístroj po připojení napájení totiž vyčte hodnotu uloženou v paměti FLASH. [4]

Příkaz pro čtení:

“?Setr + CRC + CR“

Příkaz pro zápis:

“!Setr + hodnota + CRC + CR“

Přijímaný příkaz po čtecím příkazu:

“Setr + hodnota + CRC + CR“

Přijímaný příkaz po zapisovacím příkazu:

“Sinv + hodnota + CRC + CR“

- **STREAM**

STREAM je příkaz pro nastavení zasílání dat, který umožňuje/podporuje tři základní režimy. [4]

- On
  - Přístroj zasílá neustále data o průtoku (Flow) a dále „echo“ o nastavené hodnotě po jakémkoliv zapisovacím příkazu (zpětná vazba o nastavení).
- Off
  - Přístroj odpovídá pouze na žádost čtecího příkazu.
- Echo
  - Přístroj odpovídá na žádost čtecího příkazu a dále pak odpovídá po jakémkoliv zapisovacím příkazu (zpětná vazba o nastavení).

Příkaz pro čtení:

“?Strm + CRC + CR“

Příkaz pro zápis:

“!Strm + název režimu + CRC + CR“

Přijímaný příkaz:

“Strm + název režimu + CRC + CR“

Pro upřesnění, názvy režimů jsou: „On“, „Off“ a „Echo“.

- **GAS INDEX**

GAS INDEX je příkaz pro nastavení typu plynného média, které protéká přístrojem. Podle nastaveného plynu přístroj vrací patřičně přepočtené hodnoty. Přehled plynů a jejich indexů je uveden v *Tab. 2.1*. [4]

*Tab. 2.1 Přehled plynů*

index	chemická značka/vzorec	název plynu
1	-	vzduch
2	Ar	argon
3	CO <sub>2</sub>	oxid uhličitý
4	CO	oxid uhelnatý
5	He	helium
6	H	vodík
7	CH <sub>4</sub>	metan
8	N	dusík
9	N <sub>2</sub> O	oxid dusný
10	O	kyslík

Příkaz pro čtení:

“?Gasi + CRC + CR“

Příkaz pro zápis:

“!Gasi + index plynu + CRC + CR“

Přijímaný příkaz:

“Gasi + index plynu + CRC + CR“

- **VERSION NUMBER**

VERSION NUMBER vrací verzi firmwaru, který je v přístroji nahrán. Tento příkaz může být důležitý z důvodu možného použití rozdílných verzí firmwaru. Firmware pro řadu přístrojů 1 obsahuje jiné příkazy než firmware

přístrojů řady 2. Prakticky to znamená, že v různých firmwarech mají příkazy jinou strukturu anebo dokonce nejsou podporovány.

Příkaz pro čtení:

“?Vern + CRC + CR“

Přijímaný příkaz:

“Vern + verze firmwaru + CRC + CR“

- SERIAL NUMBER

SERIAL NUMBER vrací sériové výrobní číslo přístroje. Tohoto příkazu se dá využít k identifikaci přístroje, pokud by to případná problematika situace vyžadovala. [4]

Příkaz pro čtení:

“?Srnmm + CRC + CR“

Přijímaný příkaz:

“Srnmm + sériové číslo + CRC + CR“

- VALVE INDEX

VALVE INDEX je příkaz pro nastavení stavu nastavovacího ventilu. Nastavovací ventil může pracovat ve třech režimech:

- 1 - „Automatic“
  - Režim, při kterém je elektronicky ovládán a je udržována nastavená hodnota průtoku.
- 2 - „Closed“
  - Režim, při kterém je ventil uzavřen a průtok je nulový.
- 3 - „Purge“
  - Režim, při kterém je ventil zcela otevřen, není elektronicky regulovaný. Průtok, tedy může být vyšší než maximální průtok uvedený na štítku přístroje. Maximální průtok je daný zdrojem proudícího média. [4]

Příkaz pro čtení:

“?Vlvi + CRC + CR“

Příkaz pro zápis:

“!Valve + index režimu + CRC + CR“

Přijímaný příkaz:

“Valve + index režimu + CRC + CR“

- SYNC

SYNC – synchronizace, příkaz pro získání dostupných dat z přístroje. Vrací mnoho hodnot, hodnoty jsou níže vyjmenované. Návrátové hodnoty jsou vráceny jako souvislý textový řetězec. Jednotlivé hodnoty odděluje znak ‘\n’ (new line znak – znak nové řádky). [4]

Příkaz pro čtení:

“?Sync + CRC + CR“

Přijímaný příkaz:

„Návrátový řetězec“

Celý návrátový řetězec obsahuje tyto data:

- sync
  - sync příkaz slouží k identifikaci začátku zaslání synchronizačních dat
- verze firmwaru
  - sériové číslo
- „DataEvent“
  - nespecifikováno dokumentací, využívá ho software výrobce
- „ManufactureNumberEvent“
  - nespecifikováno dokumentací, využívá ho software výrobce
- „TypeEvent“
  - nespecifikováno dokumentací, využívá ho software výrobce

- „PassWordEvent“
  - nespecifikováno dokumentací, využívá ho software výrobce
- setpoint
  - nastavená hodnota udržované hladiny průtoku
- fullscale - maximální regulovaný průtok
  - nastavený plyn
- „GasSpanEvent“
  - nespecifikováno dokumentací, využívá ho software výrobce
- „SetPointIndexEvent“
  - nespecifikováno dokumentací, využívá ho software výrobce
- „OutputIndexEvent“
  - nespecifikováno dokumentací, využívá ho software výrobce
- unit
  - nastavená jednotka pro odesílání
- valve
  - aktuální nastavení režimu ventilu
- stream
  - nastavení režimu zasílání dat
- příkaz Sync
  - sync příkaz na konci slouží k signalizaci konce zasílání synchronizačních dat

## 2.6 Komunikace SideTrak 840

### 2.6.1 Struktura komunikace

Přístroj SideTrak 840 je analogovým přístrojem. Z tohoto důvodu byl na katedře technologií a měření vytvořen ovládací modul, založený na mikrokontroléru PIC. Tento modul zprostředkovává převod mezi digitální komunikací z PC realizovanou pomocí sériové linky a analogového ovládání přístroje SideTrak 840. Tento modul byl přizpůsoben pro základní činnost přístroje. Tento komunikační protokol obsahuje jen tři příkazy. Příkazy mají podobnou strukturu jako je struktura příkazů u přístrojů SmartTrak 100. Ve struktuře příkazu komunikačního modulu byl pouze vypuštěn CRC. Struktura příkazu SideTrak 840 je zobrazena v rovnici ( 2.2 ).

$$\text{Příkaz} = \{ 1 \text{ byte pro rozlišení dotazu a zápisu} + 4 \text{ příkazové byty} + \\ + 2 \text{ byty reprezentující max 16 bitovou hodnotu (MSB a LSB)} + CR \} \quad (2.2)$$

## 2.6.2 Komunikační příkazy

- Průtok

FLOW je příkaz, který vrací hodnotu průtoku. Nejedná se o nastavovací příkaz. Příkaz je vrácen zařízením neustále a zasilá informaci o aktuální hodnotě průtoku.

Přijímaný příkaz:

“Flow + hodnota + CR“

- Nastavení hladiny průtoku

SETF je příkaz pro zaslání čtecího příkazu, který vrací aktuální hodnotu pro nastavování průtoku uloženou v paměti mikrokontroléru PIC. Pomocí zápisového příkazu se nastaví hodnota pro hladinu průtoku.

Příkaz pro čtení:

“?Setf + CRC + CR“

Přijímaný příkaz čtení:

“Setf + hodnota + CRC + CR“

Příkaz pro zápis:

“!Setf + hodnota + CRC + CR“

Přijímaný příkaz pro zápis:

- není

- Nastavení režimu ventilu

VLVI je příkaz pro nastavení stavu nastavovacího ventilu. Ventil může pracovat ve třech režimech:

- 1 - „Automatic“
  - Režim, při kterém je elektronicky ovládán a je udržována nastavená hodnota průtoku.
- 2 - „Closed“



- Režim, při kterém je ventil uzavřen a průtok je nulový.
- 3 - „Purge“
  - Režim, při kterém je ventil zcela otevřen, není elektronicky regulovaný. Průtok, tedy může být vyšší než maximální průtok uvedený na štítku přístroje. Maximální průtok je daný zdrojem proudícího média.

Příkaz pro čtení:

“?Vlvi + CRC + CR“

Přijímaný příkaz pro čtení:

“Valve + index režimu + CRC + CR“

Příkaz pro zápis:

“!Valve + index režimu + CRC + CR“

Přijímaný příkaz pro zápis:

- není

## 2.7 Návrh aplikace

Problém byl dekomponován jako vícevláknová aplikace. Pro vícevláknovou dekompozici problému bylo rozhodnuto z důvodu obsluhy pěti přístrojů v reálném čase a zobrazování okamžitých dat. Další věcí, která přispěla ke zvolení vícevláknové dekompozice byl fakt, že se jedná o asynchronně zasílaná data.

Celý projekt obsahuje mnoho tříd, což je patrné z diagramů tříd, které jsou zobrazeny v přílohách C a D. Příloha C zobrazuje diagram tříd pro jádro a příloha D zobrazuje diagramy tříd pro grafická rozhraní. Z důvodu velkého množství tříd budou zmíněny ty nejdůležitější a bude přiblížen jejich význam. Všechny třídy jsou okomentovány dokumentačními komentáři a na přiloženém DVD je nahrána programátorská dokumentace ve formátu kompletní HTML stránky s navigací.

### 2.7.1 Třída SerialComu

SerialComu je třídou pro vytvoření komunikace. Při vytvoření instance dojde ke kompletnímu vytvoření datových objektů reprezentující přístroj, se kterým bude daná instance

této třídy spolupracovat. Tato třída je připravena pro spuštění ve vláknech. To znamená, že obsahuje jednu metodu, která je k tomuto účelu určena. Metoda, která se spouští ve vláknech je pojmenována „Run“. Jedná se o bezparametrickou metodu, ve které je naprogramovaná smyčka, která obsluhuje sériový port. Tato metoda zajišťuje čtení a zápis dat po sériové komunikaci na základě změn parametrů třídy. Dále jsou v této třídě metody pro analýzu příchozích dat. Metody analyzující příchozí data dále přidávají do svých událostí příchozí data a tím umožňují zobrazování dat vně vlákna.

### 2.7.2 Třída FTDIManager

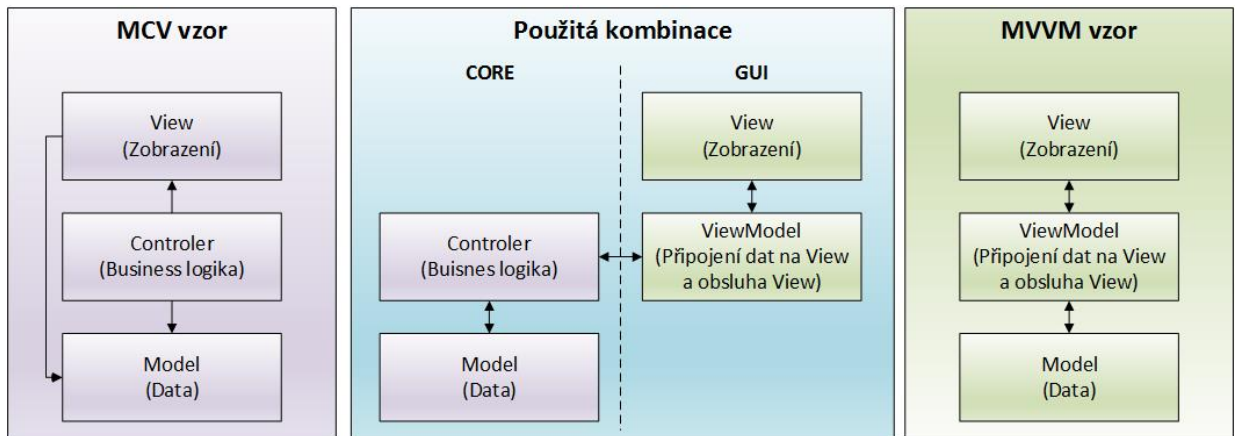
Tato třída využívá DLL od výrobce čipů FTDI. Tato třída umožňuje vyčíst z připojených čipů FT232 DescriptionName. DescriptionName je řetězcová hodnota, který v tomto problému umožňuje identifikovat zařízení. Identifikace zařízení je zajištěna pomocí názvů „Ventil1“, „Ventil2“ apod.

### 2.7.3 Třída FlowDriver

Tato třída je určena pro propojení komunikace a grafického rozhraní. Zajišťuje přebrání informací z hlavního okna, validaci a správné převádění vstupních dat. Dále zajišťuje napojení událostí na data z komunikačního vlákna a poskytnutí těchto dat výstupním kontrolkám.

## 2.8 Architektura

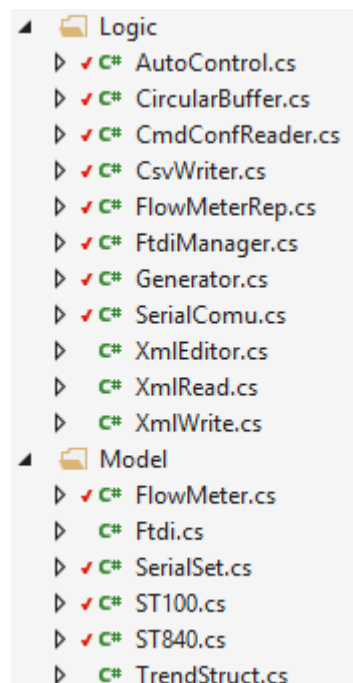
Software byl navržen pomocí třívrstvé architektury, což znamená oddělení dat, logiky a grafického rozhraní. Dále bylo využito kombinace architektonických vzorů MCV (Model Control View) a MVVM (Model View ViewModel). Ze vzoru MCV software jsou využity bloky Model a Control. Tato část je naprogramována jako DLL knihovna, která obsahuje jádro softwaru. Vzor MVVM byl použit pro grafické rozhraní k softwaru. Z MVVM architektury byly použity bloky View a ViewModel. Na *Obr. 2.2* je grafické znázornění využití softwarových architektur.



Obr. 2.2 Grafické znázornění použitých vzorů

### 2.8.1 Jádro - CORE

Jak bylo uvedeno výše, jedná se o částečnou implementaci vzoru MVC. V modelu jsou POCO třídy reprezentující potřebné objekty a v bloku Control je umístěna business logika. Z bloku Control jsou pak data vně jádra poskytována pomocí událostí, na které se můžou pověsit posluchači, což umožňuje napojení na grafické rozhraní. Hlavní logika jádra byla už od počátku navržena pro vláknové použití a to hlavně u komunikační třídy SerialComu. Uspořádání a přehled tříd je zobrazen v *Obr. 2.3*.

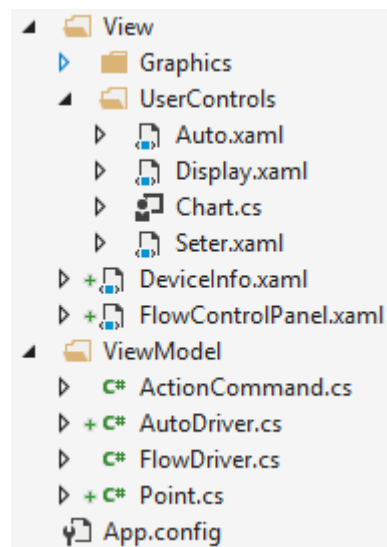


Obr. 2.3 Uspořádání tříd v aplikačním jádře

## 2.8.2 Grafické rozhraní - GUI

V části View jsou umístěny všechny uživatelské kontrolky a okna. ViewModel v tomto návrhu neplní úplně přesně obecnou funkci ViewModelu. Toto vzniklo vícevláknovou dekompozicí problému. ViewModel v tomto řešení funguje spíše jako „Driver“ – ovladač, který umožňuje propojit ve vláknech běžící komunikaci a grafické rozhraní. Toto řešení se nakonec ukázalo jako nejlepší pro danou problematiku. Byl totiž zkoušen i jiný přístup, který byl testován v přípravných fázích vývoje. Bylo zkoušeno přistupování k datům přes komplikované delegování metod napříč vlákny měnící hodnoty v GUI.

Využitím třívrstvé architektury, oddělení jádra od GUI a aplikováním vzorů se zvýšila udržitelnost kódu a zároveň se snížila cyklomatická složitost celého softwaru. Uspořádání a přehled tříd v projektu WPF GUI (hlavní okno aplikace) je znázorněno na *Obr. 2.4*.



*Obr. 2.4 Uspořádání tříd v GUI*

## 2.9 Chod programu

Po spuštění programu dojde k vyčtení dat z konfiguračního souboru a k detekci FTDI převodníků s konkrétními DescriptionName. DescriptionName je řetězcová hodnota, která přesně identifikuje jednotlivé přístroje. Po základní inicializaci dat a GUI provede program zobrazení hlavního okna aplikace, které je vidět na *Obr. 2.5*. Aplikace zobrazuje aktuální hodnoty průtoku a identifikaci přístroje v podobě jména. Aplikace vykresluje grafy

s náhledem posledních tří minut a zobrazuje nastavovací oblast. V nastavovací oblasti lze přepnout mezi manuálním a automatizovaným řízením.



Obr. 2.5 Hlavní okno aplikace

Program umožňuje deaktivovat jednotlivé zobrazené sloupce pro přístroje a tím je možno zpřehlednit zobrazení pro dané měření. Další součástí aplikace je samostatný editor pro tvorbu automatizačních skriptů. Okno editoru je zobrazeno na Obr. 2.6.

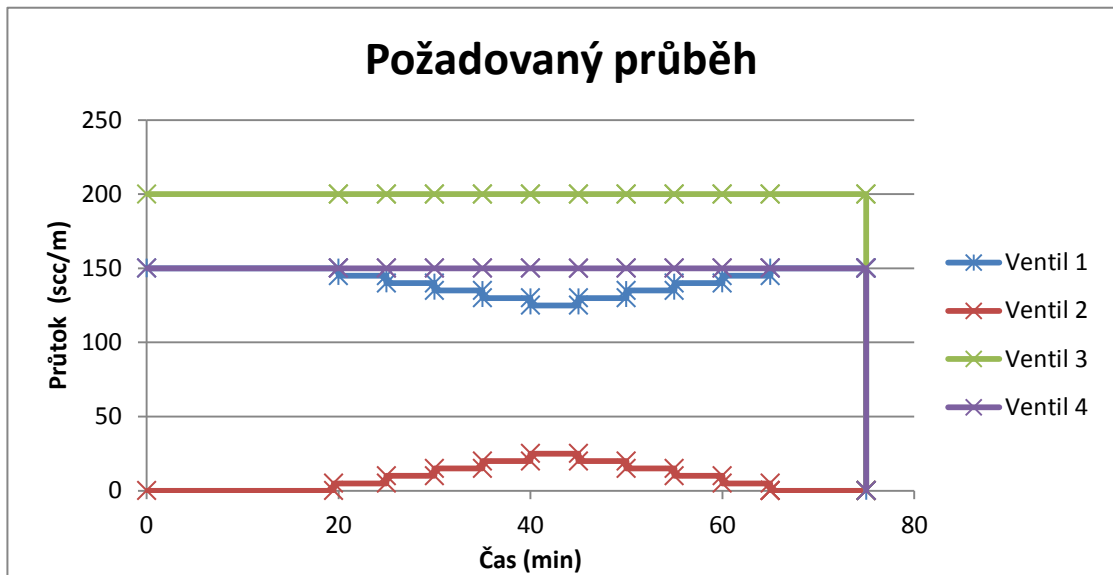


Obr. 2.5 Grafické rozhraní editoru skriptů

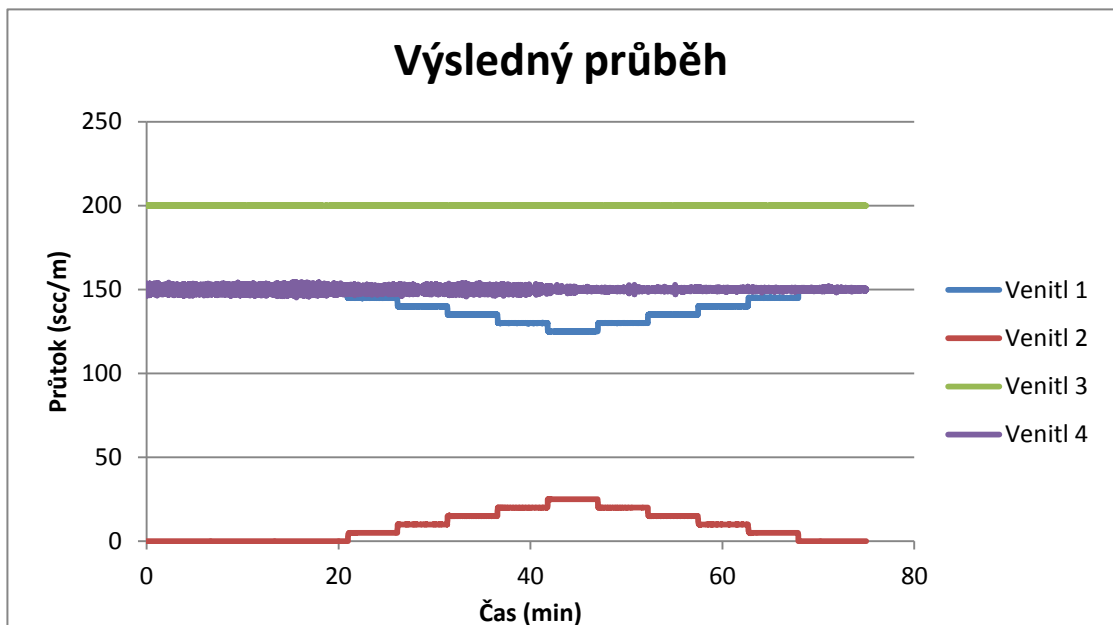
Hlavním cílem editoru je vkládání předpřipravených textových vstupů a následné převedení dat do automatizačního skriptu. Dále editor umožňuje generovat časové funkce se zadanými parametry, čímž umožňuje rychlou přípravu jednoduchých měřicích cyklů.

### 3 Měření

#### 3.1 Ověření funkcionality – test 1.



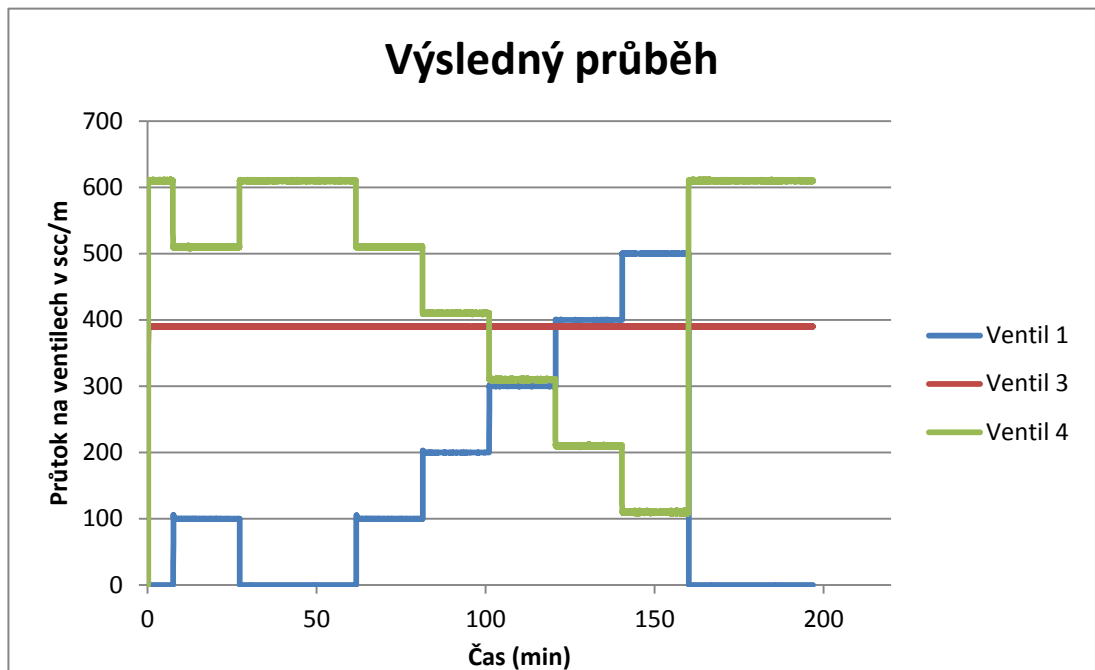
Obrázek 3.1 Požadovaný průběh testu 1.



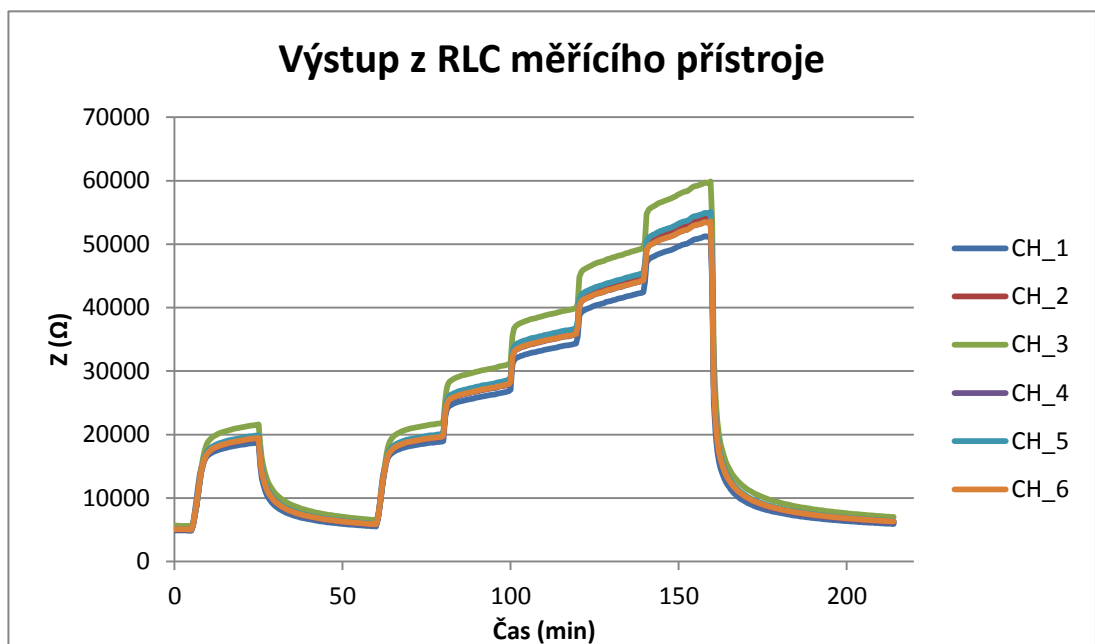
Obrázek 3.2 Výsledný průběh testu 1.

Z výše uvedených Grafů 3.1 a 3.2 vyplývá, že test 1. proběhl úspěšně. Ve výsledném průběhu je viditelné kmitání. To je způsobeno spuštěním průtokoměru s maximálním regulovaným průtokem 5000 scc/m na nízký průtok – konkrétně na 150 scc/m. Toto kmitání je přirozené a v pořádku kvůli vyšší odchylce spodní chyby kalibrace průtokoměru.

### 3.2 Testování na vzorcích – test 2.



Graf 3.1 Výsledný průběh testu 2.



Graf 3.2 Výstup z RLC měřicího přístroje během testu 2.

Ve výše uvedených grafech 3.3 a 3.4 je znázorněn průběh laboratorního měření. Graf 3.3 znázorňuje grafickou interpretaci dat získaných z přístrojů SmatrTrack. Z grafu 3.3 je patrné, že byly použity pouze průtokoměry 1, 3 a 4. Průtok byl nastavován bez kmitání a jednotlivé schodové změny jsou dobře z grafu patrné. Na grafu 3.4 je zobrazena grafická



interpretace dat z RLC měřiče, který měl na sondy připojeny testované vzorky. Výstupem RLC měřiče je hodnota komplexní impedance. Při pohledu na graf 3.3 a 3.4 je patrná též časová souběžnost obou grafů a závislost nastavovaného průtoku a na měnící se impedance testovaných senzorů. Při testu se měnily koncentrace 50, 100, 150, 200 a 250 ppm plynu NH<sub>3</sub>.

## Závěr

Hlavním cílem předložené bakalářské práce bylo vyvinutí softwaru pro přístroje společnosti Sierra Instruments SmartTrack 100 a SideTrack 840.

V první části této práce byly představeny výše zmíněné přístroje a aparatura, ve které jsou instalovány regulátory průtoku. Z dokumentace dodané výrobcem k přístrojům byly získány informace o komunikaci a o komunikačním protokolu. Toto seznámení s těmito informacemi bylo klíčové pro vývoj výsledného softwaru.

V druhé části byly zmíněny podrobnosti k vývoji softwaru a jeho hlavní rysy. Hlavními rysy jsou vícevláknová architektura a kombinace architektonických vzorů MCV a MVVM. Software byl vyvíjen v jazyce C# nad platformou Microsoft .NET 4.0. Při vývoji byly zohledněny specifické požadavky, které vycházely z laboratorního použití aparatury. Hlavním požadavkem zadání bylo automatizované řízení průtoku v čase. Tento hlavní požadavek byl splněn i se všemi ostatními požadavky, jako jsou například: manuální řízení, ovládání režimu ventilu, editor pro úpravu a přípravu skriptů a další.

V třetí části této práce jsou předloženy grafy testovacích sekvencí, které byly použity pro ověření správnosti nastavování. Ze získaných výsledků vyplývá, že činnost vyvinutého softwaru je správná.

Svojí práci bych zhodnotil jako zdařilou, jelikož se během řešení tohoto problému podařilo splnit všechny požadavky, které byly kladeny na koncový software. Toto hodnocení je také tvrzeno již několika měsíční provoz softwaru v laboratoři bez chyb.

## Seznam literatury a informačních zdrojů

- [1] SIERRA INSTRUMENTS. Smart-Trak® 2 Series 100 instruction manual [online]. C.2. Monterey, 2011 [cit. 2014-02-23]. Dostupné z: [http://www.sierrainstruments.cz/down/100st2\\_101\\_Instruction\\_manual\\_revC2.pdf](http://www.sierrainstruments.cz/down/100st2_101_Instruction_manual_revC2.pdf)
- [2] ĎAĎO, Stanislav, Ludvík BEJČEK a Antonín PLATIL. Měření průtoku a výšky hladiny. 1. vyd. Praha: BEN - technická literatura, 2005, 447 s. ISBN 80-730-0156-X.
- [3] SIERRA INSTRUMENTS. Sierra Series 830/840/860 Side-Trak™ and Auto-Trak™ Mass Flow Meters and Controllers [online]. G .1 - 2-06. Monterey [cit. 2014-02-23]. Dostupné z: [http://www.sierrainstruments.com/techsupport/manuals/im\\_83.pdf](http://www.sierrainstruments.com/techsupport/manuals/im_83.pdf)
- [4] SIERRA INSTRUMENTS. Most Commonly Used Commands on the SmartTrak ® Series [online]. v1e Monterey, 2013 [cit. 2014-02-23]. Dostupné z: [http://www.sierrainstruments.com/userfiles/file/manuals/SmartTrak-100-Series-Basic-Command-Set\\_v1d\\_21013.pdf](http://www.sierrainstruments.com/userfiles/file/manuals/SmartTrak-100-Series-Basic-Command-Set_v1d_21013.pdf)

### Literatura a informační zdroje použité při realizaci softwarového projektu

- [5] TROELSEN, Andrew. C# a .NET 2.0 profesionálně. Vyd. 1. Brno: Zoner Press, 1197 s. ISBN 80-868-1542-0.
- [6] BISHOP, J. M. C#: návrhové vzory. Vyd. 1. Brno: Zoner Press, 2010. 328 s. ISBN 978-80-7413-076-2.
- [7] Sharp, John. Microsoft Visual C# 2010 : krok za krokem. Vyd. 1. Brno: Computer Press, 2010. 696 s. ISBN 978-80-251-3147-3
- [8] Sells Chris. C# a WinForms : programování formulářů Windows: Microsoft .NET development series. Vyd. 1. Brno: Zoner Press, 2005. 648 s. ISBN 80-86815-25-0
- [9] Watson Ben. C# 4.0 : řešení praktických programátorských úloh. Vyd. 1. Brno : Zoner Press, 2010. 656 s. ISBN 978-80-7413-094-6
- [10] MICROSOFT. Microsoft Developer Network [online]. 2014 [cit. 2014-05-29]. Dostupné z: <http://msdn.microsoft.com/en-us/library>
- [11] STACK EXCHANGE INC. Stack Overflow [online]. 2014 [cit. 2014-05-29]. Dostupné z: <http://stackoverflow.com/>

## Přílohy

### Příloha A – Tabulka jednotek podporovaných přístroji SmartTrak 100 [1]

index	zkratka	název jednotky
1	scc/s	standartní kubický centimetr za sekundu
2	scc/m	standartní kubický centimetr za minutu
3	scc/h	standartní kubický centimetr za hodinu
4	ncc/s	normální kubický centimetr za sekundu
5	ncc/m	normální kubický centimetr za minutu
6	ncc/H	normální kubický centimetr za hodinu
7	SCF/s	standartní kubická stopa za sekundu
8	SCF/m	standartní kubická stopa za minutu
9	SCF/H	standartní kubická stopa za hodinu
10	NM3/s	normální kubický metr za sekundu
11	NM3/m	normální kubický metr za minutu
12	NM3/H	normální kubický metr za hodinu
13	SM3/s	standartní kubický metr za sekundu
14	SM3/m	standartní kubický metr za minutu
15	SM3/H	standartní kubický metr za hodinu
16	sl/s	standartní litr za sekundu
17	sl/m	standartní litr za minutu
18	sl/H	standartní litr za hodinu
19	NL/s	normální litr za sekundu
20	NL/m	normální litr za minutu
21	NL/H	Normální litr za sekundu
22	g/s	gram za sekundu
23	g/m	gram za minutu
24	g/H	gram za hodinu
25	kg/s	kilogram za sekundu
26	kg/m	kilogram za minutu
27	kg/H	kilogram za hodinu
28	lb/s	libra za sekundu
29	lb/m	libra za minutu
30	lb/H	libra za hodinu

**Příloha B Metoda výrobce Sierra v jazyce C# pro výpočet CRC [4]**

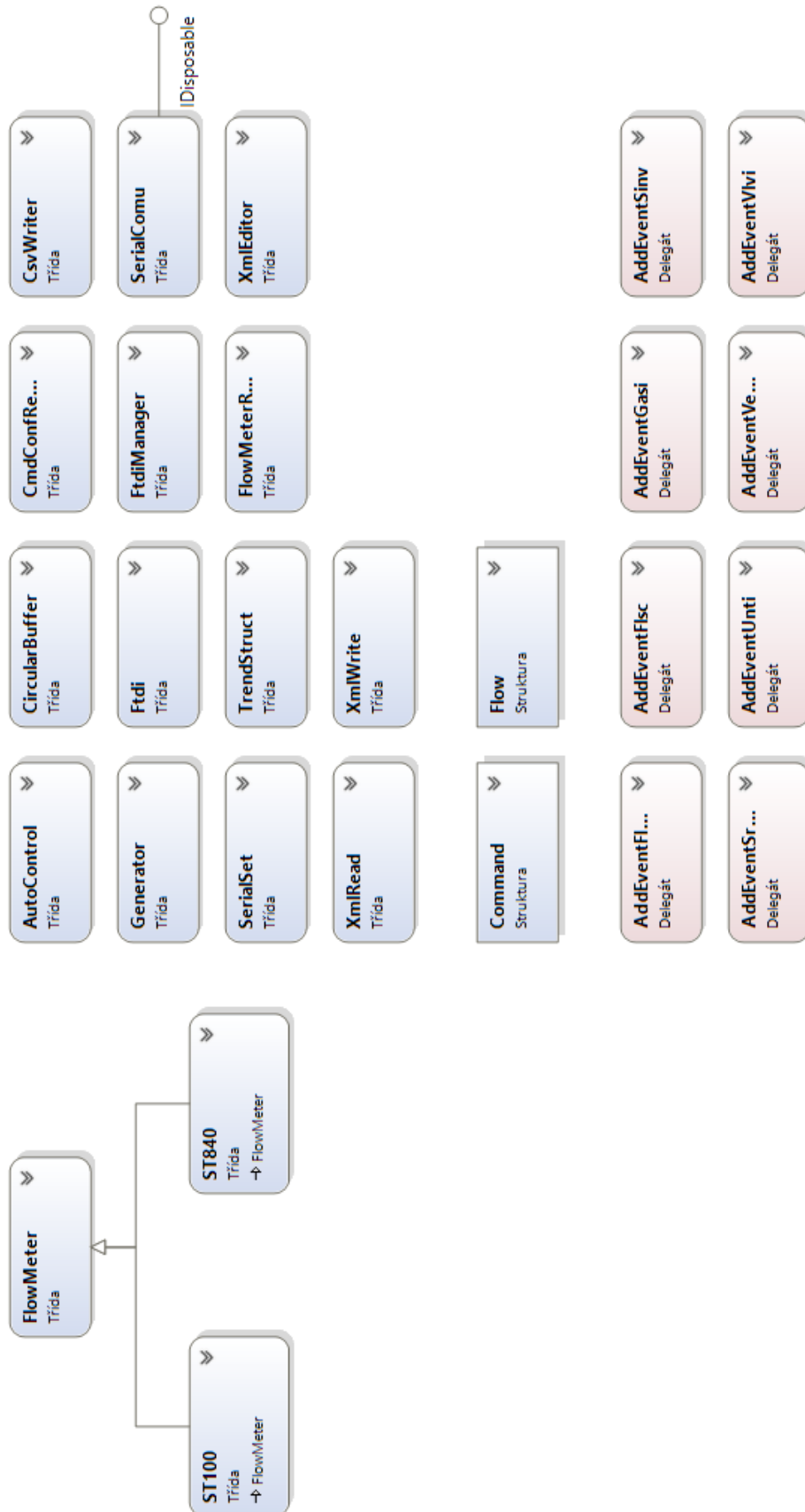
```
private static uint CalcCRC(byte[] cmd)
{
    int i,j;
    uint crc;

    crc=0xffff;
    for (i=0; i<cmd.Length; i++)
    {
        crc=crc^((uint)(cmd[i]*0x0100));
        for(j=0; j<8; j++)
        {
            if((crc&0x8000)==0x8000)
                crc=((crc<<1)^0x1021)&0xffff;
            else
                crc=(crc<<1)&0xffff;
        }
    }

    if((crc&0xff00)==0x0d00) crc +=0x0100;
    if((crc&0x00ff)==0x000d) crc +=0x0001;
    if((crc&0xff00)==0x0000) crc +=0x0100;
    if((crc&0x00ff)==0x0000) crc +=0x0001;

    return crc;
}
```

### Příloha C Diagram tříd jádra aplikace



## Příloha D Diagram tříd grafického rozhraní aplikace

## Grafické rozhraní hlavního okna



## Grafické rozhraní okna editoru

