



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV MECHANIKY TĚLES, MECHATRONIKY A
BIOMECHANIKY

FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF SOLID MECHANICS, MECHATRONICS AND
BIOMECHANICS

ZPRACOVÁNÍ A VIZUALIZACE DAT Z HMOTNOSTNÍHO SPEKTROMETRU TYPU TOF-MALDI

DATA PROCESSING AND VISUALIZATION FROM THE TOF-MALDI MASS SPECTROMETER

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. PAVEL KUBA

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. PAVEL HOUŠKA, Ph.D.

BRNO 2012

Vysoké učení technické v Brně, Fakulta strojního inženýrství

Ústav mechaniky těles, mechatroniky a biomechaniky

Akademický rok: 2011/2012

ZADÁNÍ DIPLOMOVÉ PRÁCE

student(ka): Bc. Pavel Kuba

který/která studuje v **magisterském navazujícím studijním programu**

obor: **Mechatronika (3906T001)**

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

Zpracování a vizualizace dat z hmotnostního spektrometru typu TOF-MALDI

v anglickém jazyce:

Data processing and visualization from the TOF-MALDI mass spectrometer

Stručná charakteristika problematiky úkolu:

Cílem této práce je řešit problematiku zpracování, archivace a vizualizace naměřených dat měřených hmotnostním spektrometrem typu TOF-MALDI. Jedná se především o problematiku kalibrace měření, odstranění vlivu matrice a pozadí z měřených dat, vizualizace a reprezentace ve 2D a 3D.

Cíle diplomové práce:

1. Seznamte se s problematikou hmotnostní spektrometrie a s existujícím software pro jeho řízení,
2. Řešte efektivitu získávání dat z digitizéru a efektivitu filtrace signálu s ohledem na rychlost a paměťové nároky algoritmu,
3. Řešte problematiku kalibrace měření, odstranění vlivu matrice a pozadí z měřených dat,
4. Řešte efektivitu 2D skenování a možnosti zobrazení výsledků ve 3D,
5. Realizovaná řešení zhodnoťte, popište výhody a nevýhody.

Seznam odborné literatury:

- [1] Ertl, L.: SOFTWARE PRO ŘÍZENÍ HMOTNOSTNÍHO SPEKTROMETRU A NANÁŠECÍHO ZAŘÍZENÍ, diplomová práce, Brno 2011, dostupné z URL:<www.vutbr.cz>
- [2] MOSKOVETS, E, et al. High-Throughput Axial MALDI-TOF MS Using a 2-kHz Repetition Rate Laser. Analytical chemistry. 2006, 78, s. 912–919.
- [3] PREISLER, J. Jan Preisler [online]. 2008 [cit. 2011-05-18]. Studijní materiál ke stažení. Dostupné z WWW: <<http://bart.chemi.muni.cz/>>.
- [4] National Instruments: LabVIEW User Manual[online], June 2003, 320999E-01, dostupné z URL:<<http://www.ni.com>>

Vedoucí diplomové práce: Ing. Pavel Houška, Ph.D.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2011/2012.

V Brně, dne 23.11.2011

L.S.

prof. Ing. Jindřich Petruška, CSc.
Ředitel ústavu

prof. RNDr. Miroslav Doupovec, CSc., dr. h. c.
Děkan fakulty

ABSTRAKT

Diplomová práce se zabývá vývojem řídicích aplikací pro nanášecí zařízení a hmotnostní spektrometr. Práce obsahuje popis principu činnosti zařízení a jejich hardwarových specifikací. Práce dále obsahuje popis jednotlivých aplikací. Funkčnost aplikací byla ověřena na reálných měřeních.

SUMMARY

This thesis describes the development of control applications for the deposition machine and mass spectrometer. Thesis describes operation principles of both devices and their hardware specifications. Thesis also describes the design of developed applications. Functionality was tested on series of real measurements.

KLÍČOVÁ SLOVA

Hmotnostní spektrometrie, hmotnostní spektrometr, TOF-MALDI, nanášecí zařízení, LabVIEW, stavový stroj, XControl

KEYWORDS

Mass spectrometry, mass spectrometer, TOF-MALDI, deposition machine, LabVIEW, state machine, XControl

PODĚKOVÁNÍ

Na tomto místě chci poděkovat vedoucímu mé diplomové práce Ing. Pavlu Houškovi, Ph.D. za odborné konzultace a poskytnuté rady. Dále chci poděkovat Mgr. Antonínu Bednaříkovi a doc. Mgr. Janu Preislerovi, Ph.D. za cenné informace a příjemnou pracovní atmosféru.

PROHLÁŠENÍ

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně na základě svých vědomostí, rad a pokynů vedoucího diplomové práce, poskytnutých materiálů a odborných konzultací.

V Brně, dne 23. května 2012

Pavel Kuba

BIBLIOGRAFICKÁ CITACE

KUBA, P. *Zpracování a vizualizace dat z hmotnostního spektrometru typu TOF-MALDI*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2012. XY s. Vedoucí diplomové práce Ing. Pavel Houška, Ph.D.

OBSAH

| | | |
|---------|--|----|
| 1 | Úvod..... | 9 |
| 2 | Hmotnostní spektrometrie | 10 |
| 2.1 | Techniky ionizace | 10 |
| 2.2 | Metody hmotnostní analýzy | 11 |
| 2.3 | Hmotnostní spektrometr typu MALDI-TOF | 11 |
| 3 | Popis řízených zařízení a požadavky zadavatele | 13 |
| 3.1 | Nanášecí zařízení | 13 |
| 3.1.1 | Popis současné funkcionality | 13 |
| 3.1.2 | Hardware | 13 |
| 3.1.3 | Stávající software..... | 15 |
| 3.1.4 | Požadavky zadavatele na zlepšení | 15 |
| 3.2 | Hmotnostní spektrometr | 16 |
| 3.2.1 | Popis současné funkcionality | 16 |
| 3.2.2 | Hardware | 17 |
| 3.2.2.1 | Laser, optická soustava | 17 |
| 3.2.2.2 | Polohovací stolek..... | 18 |
| 3.2.2.3 | Videokarta | 18 |
| 3.2.2.4 | Měřicí karta..... | 18 |
| 3.2.2.5 | Řídicí PC | 19 |
| 3.2.3 | Stávající software..... | 19 |
| 3.2.4 | Požadavky zadavatele na zlepšení | 20 |
| 4 | Formulace cílů | 21 |
| 4.1 | Nanášecí zařízení | 22 |
| 4.2 | Hmotnostní spektrometr | 23 |
| 5 | Řešení stávajících chyb..... | 24 |
| 5.1 | Řešení problémů nanášecího zařízení..... | 24 |
| 5.2 | Řešení problémů hmotnostního spektrometru | 24 |
| 6 | Popis vyvíjených aplikací | 27 |
| 6.1 | Obecný popis struktury aplikace | 27 |
| 6.2 | Popis aplikace pro nanášecí zařízení | 28 |
| 6.2.1 | Event struktura..... | 28 |
| 6.2.1.1 | Nastavení dráhy pohybu..... | 30 |
| 6.2.1.2 | Nastavení pulzu pro piezopipetor..... | 30 |
| 6.2.1.3 | Nastavení režimu „dropping“ | 31 |
| 6.2.1.4 | Nastavení režimu „fluorescence“ | 31 |

| | | |
|---------|---|----|
| 6.2.1.5 | Manuální ovládání..... | 31 |
| 6.2.1.6 | Automatické ovládání | 31 |
| 6.2.2 | Řídicí proces..... | 31 |
| 6.2.2.1 | Stav „manual“ | 32 |
| 6.2.2.2 | Stav „automatic“ | 33 |
| 6.2.3 | Proces posuvů | 34 |
| 6.2.4 | Proces efektoru..... | 35 |
| 6.2.4.1 | Režim „dropping“ | 35 |
| 6.2.4.2 | Režim „piezo“ | 36 |
| 6.2.5 | Proces kamery..... | 37 |
| 6.2.6 | Inicializace a ukončení procesů..... | 37 |
| 6.3 | Hmotnostní spektrometr | 38 |
| 6.3.1 | Uživatelské rozhraní, event struktura | 38 |
| 6.3.1.1 | Nastavení dráhy pohybu..... | 39 |
| 6.3.1.2 | Nastavení parametrů měření..... | 39 |
| 6.3.1.3 | Manuální ovládání..... | 40 |
| 6.3.1.4 | Automatické ovládání..... | 40 |
| 6.3.1.5 | Ovládání kamery | 40 |
| 6.3.1.6 | Zobrazení měřených dat..... | 41 |
| 6.3.2 | Hlavní řídicí proces | 41 |
| 6.3.2.1 | Stav „automatic“ | 42 |
| 6.3.3 | Proces posuvů | 43 |
| 6.3.4 | Proces měření..... | 44 |
| 6.3.5 | Proces zpracování dat | 45 |
| 6.3.6 | Proces kamery..... | 46 |
| 6.3.7 | Inicializace a ukončení procesů..... | 46 |
| 6.4 | Čtení dat z hmotnostního spektrometru. | 46 |
| 7 | Ověření funkčnosti..... | 48 |
| 8 | Závěr | 50 |

1 Úvod

Tato diplomová práce se zabývá vývojem řídicích aplikací pro hmotnostní spektrometr a zařízení pro nanášení vzorků, které jsou v tomto spektrometru analyzovány. Zadavatelem práce je Ústavu chemie Přírodovědecké fakulty Masarykovy univerzity, který obě zařízení vlastní a věnuje se jejich vývoji.

Práce svým obsahem navazuje na předchozí řešení Ing. Lukáše Ertla, který se problémem zabýval také v rámci své diplomové práce.

První část práce se zabývá popisem metod hmotnostní spektrometrie a charakteristikou jednotlivých zařízení.

Druhá část se již pak zabývá popisem jednotlivých aplikací, které byly v rámci práce vyvíjeny. Vývoj aplikací probíhal v prostředí NI LabVIEW.

Vytvořené aplikace byly použity pro analýzu skutečných vzorků.

2 Hmotnostní spektrometrie

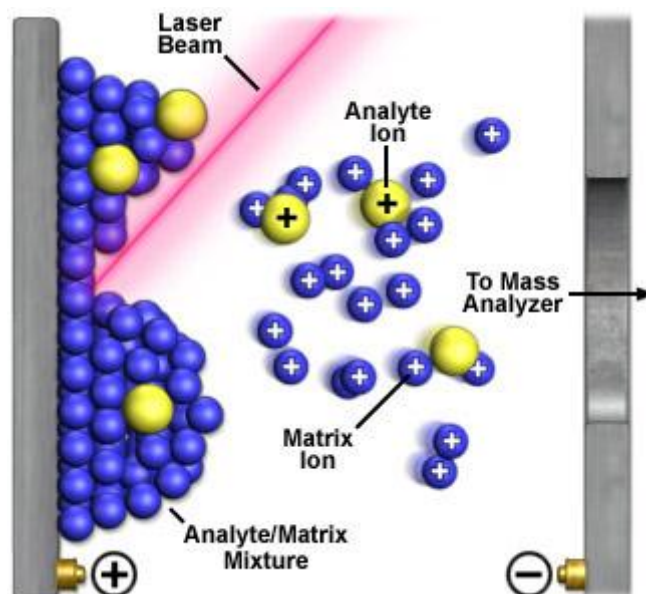
Hmotnostní spektrometrie (MS z angl. mass spectrometry) je jednou z metod analytické chemie. Princip metody spočívá v ionizaci zkoumané chemické látky na nabitě částice a následným měřením poměru hmotnosti k náboji (m/z), který je pro každou částici jedinečný. Tato informace pak slouží k určení zastoupení částic ve zkoumaném vzorku, a tím i k identifikaci jeho složení.

Historie moderní hmotnostní spektrometrie sahá až do roku 1886, kdy Eugen Goldstein pozoroval chování plynu s nízkým tlakem. Goldstein zaznamenal záření, které probíhalo od anody skrz kanálky v katodě. O tři roky později Wildhelm Wien zjistil, že silné elektrické či magnetické pole odchyluje záření pozorované Goldsteinem. Následně sestrojil zařízení s paralelním elektrickým a magnetickým polem, které separovalo částice záření podle jejich m/z poměru.

Různé druhy hmotnostních spektrometrů se od sebe navzájem liší způsobem ionizace, rozdělení částic podle hmotností a detekce těchto částic. V současnosti se setkáváme s následujícími způsoby ionizace a hmotnostní analýzy. [1]

2.1 Techniky ionizace

Vzorky mohou být ionizovány různými způsoby. Metoda „Electron Impact Ionisation“ (EI) spočívá ve vystřelování elektronů na vzorek žhavenou katodou. Elektron po nárazu „utrhne“ z vazby vlastní elektron molekuly, a tím ionizuje. V případě metody „Chemical Ionisation“ (CI) je ionizace dosaženo použitím reaktivního plynu s kombinací EI. Vzorek může být také ionizován metodou „Fast Atom Bombardment“ (FAB), která spočívá v ostřelování vzorku atomy argonu. Další používanou technikou je „Electrospray Ionization“ (ESI), ve které se rozpuštěné vzorky vstříknou do elektrického pole. Poslední popisovanou metodou je „Matrix Assisted Laser Desorption/Ionization“ (MALDI). Princip metody spočívá v desorpci analytu paprskem laseru. Velkou výhodou této metody je, možnost archivace vzorku. Se stejným vzorkem lze provést ještě jedno nebo více měření. [2]



Obrázek 1 - Metoda MALDI [3]

2.2 Metody hmotnostní analýzy

Ionizovaný vzorek se analyzuje v hmotnostním analyzátoru, který může být několika druhů, v závislosti na použité metodě. Metoda „Quadrupole“ spočívá v separování částic pomocí dvou párů tyčí pod napětím, které v závislosti na frekvenci změny napětí propustí jen částice o určité hmotnosti. „Magnetic Sector Field“ je metoda založená na odklánění různě hmotných částic pomocí silného magnetického pole. Částice s nižší hmotností se odkloní více, než částice s vyšší, a tím dojde k separaci. Tato metoda lze rozšířit o dvojici zakřivených desek, na které je připojeno napětí a v závislosti na velikosti elektrického pole, generovaného těmito deskami, prochází jen částice o určité hmotnosti. Poslední popisovanou metodou je „Time of Flight“ (TOF). Ionizované vzorky jsou urychlovány pomocí vysokého napětí a vyslány do letové trubice. Různě hmotné částice mají po průchodu elektrickým polem různou kinetickou energii a tedy i různou rychlost. Měří se doba letu částic a z ní se vypočítají jednotlivé hmotnosti. [2]

2.3 Hmotnostní spektrometr typu MALDI-TOF

Jak je zřejmé z názvu, jedná se o hmotnostní spektrometr, který kombinuje MALDI metodu ionizace a TOF metodu hmotnostní analýzy. Jeho podrobnější popis je zde uveden z toho důvodu, že je předmětem diplomové práce.

Vzorek určený k analýze je složený z analytu a matrice. Krátkými laserovými pulsy (běžně v jednotkách nanosekund) se ozařuje, a tím předává matrici energii. Absorpce energie vede k desorpci matrice, která plní několik funkcí. Chrání analyt před destrukcí laserovým paprskem, ionizuje analyt při

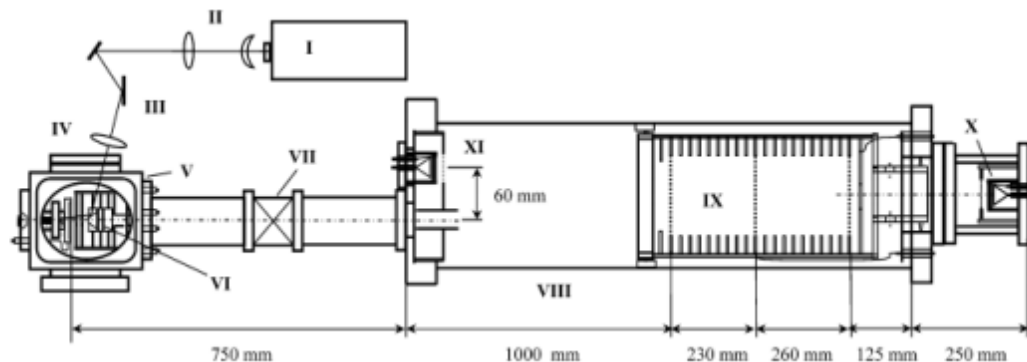
desorpci a strhává ho s sebou, čímž vzniká „oblak“ iontů. Vniklý „oblak“ je urychlen do letové trubice dvojicí extrakčních mřížek, na které je přivedeno vysoké napětí.

Princip průletového analyzátoru spočívá v měření času, který iont potřebuje k překonání vzdálenosti mezi zdrojem a detektorem. Výhodou je především krátká doba analýzy, jeden puls generuje jedno spektrum. Přepočítání do hmotnostní domény se provádí podle vztahu (1). Kde m je hmotnost částice, z je náboj, e je elementární náboj, U je urychlovací napětí, t je doba průletu a L je délka trubice.

$$\frac{m}{z} = 2 \cdot e \cdot U \cdot \frac{t^2}{L^2} \quad (1)$$

Ke zvýšení rozlišení hmotností se používají různé metody. Jednou z nich je pulzní (zpožděná) extrakce, která upravuje nerovnoměrné rozložení kinetických energií mezi částicemi v okamžiku po vypaření laserem. Principem je zpoždění urychlovacího napětí o určitý čas po desorpci. Tím se vyrovnají kinetické energie rychlejších částic, které se ve chvíli aplikace napětí nacházejí dále od zdroje s těmi, co jsou zdroji blíže a mají nižší energii.

Druhou metodou zvýšení rozlišení je použití iontového zrcadla na konci letové trubice. Reflektor je tvořen několika prstencovými elektrodami, na něž je přivedeno napětí o stejné polaritě jako urychlovací, ale opačné prostorové orientace. Částice s nižší energií opustí reflektor dříve, než částice s vyšší energií, které do reflektoru proniknou hlouběji, všechny tak dopadnou na detektor současně. [4]



Obrázek 2 - Schéma spektrometru MALDI TOF:

I, II, III) laser a optická soustava, IV, V, VI) soustava posuvů a extrakčních mřížek, VII) bránový ventil, VIII) letová trubice, IX) reflektor (iontové zrcadlo), X, XI) MCP detektory [5]

3 Popis řízených zařízení a požadavky zadavatele

Předmětem této diplomové práce je pokračování vývoje software pro řízení hmotnostního spektrometru typu MALDI-TOF a nanášecího zařízení pro nanášení vzorků, určených k analýze v hmotnostní spektrometrii. Obě zařízení jsou vyvíjena na Ústavu chemie Přírodovědecké fakulty Masarykovy univerzity. Hlavním cílem práce je doplnit nové požadované funkce, zvýšit rychlosti měření spektrometru, zvýšit rychlost nanášecího zařízení a zvýšit spolehlivosti softwaru.

Obě zařízení byla již dříve zadavatelem sestavena a využívána, a proto nebylo možné zasahovat do specifikací či výběru komponent.

3.1 Nanášecí zařízení

Nanášecí zařízení je určeno k přípravě vzorků pro analýzu v hmotnostním spektrometru. Konkrétně se jedná o přípravu vzorků pro vyvíjený spektrometr a pro komerční spektrometry v majetku zadávajícího ústavu. Nanášecí zařízení je přenosné, řízené počítačem prostřednictvím USB rozhraní. Řídicí software je realizován v prostředí NI LabVIEW.

3.1.1 Popis současné funkcionality

Zařízení pracuje ve dvou nanášecích režimech, které používají dvě zcela odlišné metody. První z nich je metoda otisknutí kapky, která se vytvoří na konci kapiláry, na plochu desky. Tato metoda je dále označována jako „dropping“. Druhá metoda nanášení spočívá ve vstřikování mikroskopických kapek pomocí piezopipetoru. Dále označováno jako „piezo“ metoda. Zařízení je možno rozšířit o snímač „fluorescence“ již nanesených vzorků. Tento režim však není plně implementován.

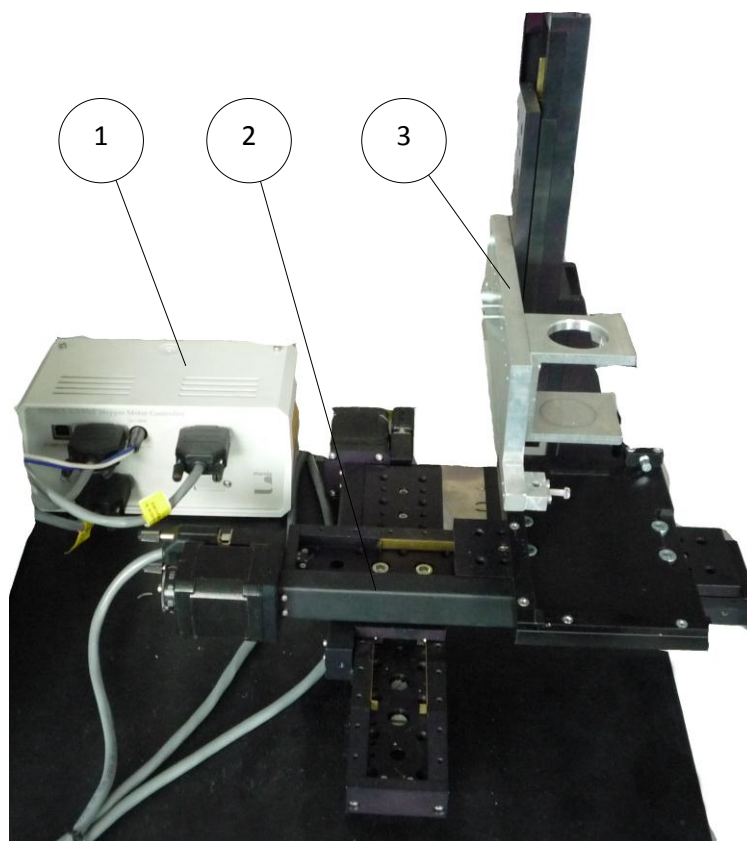
Protože výrobci hmotnostních spektrometrů nepoužívají jeden standardizovaný typ desky, je nanášecí zařízení vybaveno adaptérem, který dovoluje použití tří druhů těchto desek. Tím lze nanášecí zařízení použít pro přípravu vzorků pro všechny hmotnostní spektrometry, které zadavatel vlastní. [6]

3.1.2 Hardware

Zařízení je možné popsat jako trojosý kartézský manipulátor, jehož dva lineární posuvy jsou použity pro pohyb desky v rovině xy a jeden pro pohyb ramene v ose z . Pohybem ramene se tedy v módu „dropping“ otiskují vzniklé kapky a v módu „piezo“ a „fluorescence“ se udržuje konstantní výška zdvihu. Lineární posuvy jsou zde zastoupeny modelem 8MT175-100 od výrobce STANDA. Jedná se o spojení krokového motoru, pohybového šroubu a lineárního vedení. Poloha

posuvu se měří nepřímo, počtem pulsů, které vyše kontrolér. Posuvy jsou vybaveny koncovými spínači, které slouží i jako referenční hodnoty. Ovládání všech tří os je zajištěno kontrolérem 8SMC1-USBhF od stejného výrobce. Tato řídicí jednotka je ovládána z připojeného PC přes USB. K ovládání piezopipetoru se používá zařízení vyvinuté Ing. Krásenským z Masarykovy Univerzity, které je s PC taktéž spojeno přes USB. Následují hardwarové specifikace nanášecího zařízení:

- rozlišení: 2,5 μm na jeden celý krok
- počet kroků na otáčku: 200
- krokování: 1/1, 1/2, 1/4, 1/8
- stoupání závitu: 0,5 mm
- maximální rychlost: 10 mm/s
- „soft“-start/stop
- maximální zatížení: horizontální 80N, vertikální 30N
- ovládání kontroléru motorů i piezopipetoru z PC pomocí USB [7] [8]



Obrázek 3 - Nanášecí zařízení: 1) kontrolér, 2) posuvy xy, 3) posuv z s ramenem

3.1.3 Stávající software

Stávající software byl vyvíjen předchozím řešitelem Ing. Lukášem Ertlem v prostředí Labview od společnosti National Instruments. Struktura aplikace je postavena na principu stavového stroje, blíže v kapitole 6.1. Rozdělení do několika *while* smyček s nezávislým časováním je dosaženo možné paralelizace procesů. Konkrétně se jedná o rozdělení na *event* strukturu, která zachytává uživatelské akce a vkládá jednotlivé příkazy do fronty příkazů, ty se zpracovávají v samostatném procesu. Tím je řídicí proces, který určuje na základě stavu zařízení další příkazy k vykonání. Součástí řídicího procesu je obsluha uživatelského rozhraní, ovládání piezopipetoru a výpočet dráhy, po které se mají pohybovat jednotlivé osy. Samostatně stojící proces motorů komunikuje s kontrolérem, posílá do něj příkazy a vyčítá stav, ve kterém se právě posuvy nachází. Samostatný je také proces ovládání a zobrazování obrazu z kamery.

Z pohledu uživatele je aplikace rozdělena do tří hlavních režimů: manuálního, automatického a „piezo“ módu. V manuálním režimu je posuv všech tří os nanášecího zařízení ovládán pomocí směrových tlačítek nebo zadáním relativních souřadnic. Součástí nastavení pozice je i rychlost motorů a velikost krokování. Automatický režim umožňuje nastavení mřížky bodů, které budou nanášeny metodou „dropping“. Po spuštění sekvence se postupně nanášou všechny nastavené body. Uživatel do procesu již nemusí, a ani nemůže zasahovat. V posledním režimu, „piezo“ módu, se k předešlému nastavení mřížky bodů přidává nastavení generovaného pulsu pro piezopipetor. Takto nastavený puls se převede na sekvenci příkazů a pošle do kontroléru. „Piezo“ mód obsahuje, na rozdíl od ostatních, display obrazu z kamery. Před započítím nanášení je totiž potřeba zkontrolovat, zde se generují kapky správného tvaru, směru a velikosti. [6]

3.1.4 Požadavky zadavatele na zlepšení

Zadavatel přišel s novými požadavky na funkcionalitu a na opravu stávajících závad, které komplikovaly práci. Mezi požadavky bylo zachování dvou nanášecích režimů (dropping, piezo) s možností rozšíření o režim fluorescence. Dále byl kladen důraz na zvýšení rychlosti na maximum a celkovou stabilizaci procesu. Navíc by měla být aplikace spustitelná i na méně výkonných zařízeních. Ze zkušeností obsluhy vyplynulo, že při nanášení často dochází k výpadkům pohybu některých os. V tom případě je pak nutné proces zastavit a s novým nastavením pokračovat. Obsluha se tedy nemůže na zařízení spolehnout a musí ho kontrolovat, což je v rozporu s požadavky zadavatele. Kolísavá rychlost a

nespolehlivost mohou vést ke znehodnocení nanášených vzorků, a tím k degradaci celého nanášecího procesu.

Spolu se stávajícím stavem software byly požadavky zhodnoceny a z nich vyvozené závěry vedly k formulaci cílů diplomové práce.

3.2 Hmotnostní spektrometr

Obecný popis hmotnostní spektrometrie a princip činnosti jednotlivých typů spektrometrů se nachází v kapitole 2. Tato kapitola tedy pojednává o konkrétním spektrometru typu MALDI-TOF, jež nese označení „Drobeček“ a nachází se na Ústavu chemie Přírodovědecké fakulty Masarykovy univerzity. Podle zadavatele byla motivace pro vývoj vlastního spektrometru následující:

V době výroby spektrometru byla rychlost komerčních zařízení velmi nízká, analýzy trvaly dlouho. Z tohoto důvodu byl navržen vývoj vlastního spektrometru, který by prováděl stejné analýzy rychleji. Hlavní příčinou pomalé analýzy komerčních spektrometrů byly použité lasery. Ty pracovaly s maximální frekvencí 50 Hz. K výraznému zrychlení došlo díky použití laseru o frekvenci pulzování 2 kHz. [5]

Nyní je znovu snaha urychlit proces měření a neztratit při tom na kvalitě naměřených dat. K tomu účelu je použito zrcátko, které má kompenzovat pohyb stolku v jedné ose. Předpokládaná doba měření na vyvíjeném spektrometru by tak měla být několikrát kratší než u momentálně nejrychlejšího komerčního spektrometru.

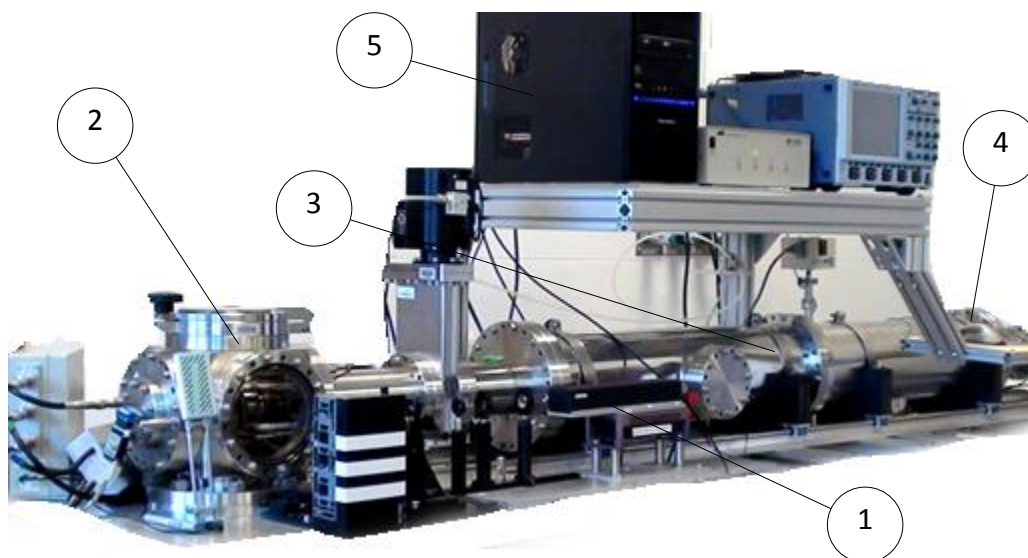
Dalšími výhodami vývoje vlastního spektrometru je možnost opravy obsluhou, velká flexibilita co se týče rozšíření a výrazně nižší cena.

3.2.1 Popis současné funkcionality

Zařízení je schopné analýzy ve třech režimech, „imaging“, „small plate“ a „oscilloscope“. První uvedený režim analyzuje určenou plochu bod po bodu podle zadaného kroku. Vytváří se tak rastr, kde každý jednotlivý bod obsahuje naměřené spektrum. Takto naměřená data se dají zpracovat do podoby mapy intenzity. Režim „small plate“ je možné popsat jako režim známých bodů. V tomto režimu se používají desky s označenými body a obsluha si pouze vybírá, který bod se má analyzovat. Výsledkem je pak vždy jedno spektrum. V posledním režimu je zařízení používáno jako osciloskop. Analýza probíhá neustále a obsluha může libovolně měnit polohu stolku, a tak proměřovat libovolné body.

3.2.2 Hardware

Hardware hmotnostního spektrometru podléhá vysokým nárokům na spolehlivost, přesnost a čistotu. Komponenty zařízení pracují v hlubokém vakuu ($1,33 \times 10^{-6}$ Pa) a v prostředí s vysokým napětím ($14 \div 15$ kV). Protože se jedná o typ MALDI-TOF, jsou vzorky vypařovány laserem a následně prolétají tubusem dlouhým 2,6 m. Odpovídající schéma je na obrázcích Obrázek 2 a Obrázek 4.



Obrázek 4 - Hmotnostní spektrometr: 1) laser, 2) stolek a extrakční mřížky, 3) letová trubice, 4) detektor, 5) řídicí PC

3.2.2.1 Laser, optická soustava

Prvním krokem k analýze vzorku je jeho ionizace. Vzorek je skrz optickou soustavu odpařen laserem typu Nd:YAG s těmito specifikacemi:

- vlnová délka: 355 nm
- frekvence pulzování: 2 kHz (maximální 10 kHz)
- energie pulzu: 8.6 μ J/puls

Během posuvu stolku na další souřadnici musí být laser odkloněn, aby zbytečně nevypaloval vzorek. K tomuto účelu slouží zrcátko s proměnným natočením závislým na napětí. Model 6810P (Cambridge Technology) mění polohu se zpožděním menším jak 1 ms. K ovládání zrcátka se používá zařízení myDAQ od společnosti National Instruments. Jedná se o cenově dostupný převodník a měřicí zařízení. Obsahuje jak digitální, tak analogové vstupy a výstupy. [9] [10] [11]

3.2.2.2 Polohovací stolek

Polohovací stolek je sestaven ze dvou lineárních posuvů (modely MM-3-M-F a MM-4-M-Ex) od firmy National Aperture. Tyto posuvy jsou uzpůsobeny pro práci ve vakuu. Jedná se o kombinaci DC motorů přes planetovou převodovku spojenými s kuličkovými šrouby, které zajišťují posuv po lineárním vedení.

- rychlost: až 5 mm/s
- přesnost: 3 μ m

Polohy posuvů jsou měřeny nepřímou přes enkodér na hřídeli motoru. Polohu je tedy nutné přepočítat z převodového poměru a stoupání šroubu (pro každou osu jině). Posuvy jsou vybaveny koncovými spínači, které slouží i jako referenční hodnoty.

Řízení posuvů zajišťuje kontrolér MC-4SA od firmy National Aperture. Ten je s PC propojen přes řídicí kartu NI PCI-7344 od firmy National Instruments. V LabVIEW je ovládání karty dostupné přes knihovnu „Vision and Motion“. Konfigurace je možná pohodlně přes „Measurement & Automation Explorer“ (dále jen MAX). V této aplikaci lze naladit PID regulátory, přiřazovat osy do vektorových prostorů a konfigurovat další parametry os. [12] [13] [14] [15]

3.2.2.3 Videokarta

Kvůli požadavkům na zobrazování právě vypalovaného vzorku je součástí hmotnostního spektrometru i USB karta (AVerMedia DVD EZMaker 7) na digitalizaci videa z analogové kamery.

- rozlišení: 720 x 480 px
- snímková frekvence: 30 fps

Protože zde obraz z kamery slouží pouze jako kontrola, není zaznamenáván, jsou výše uvedené specifikace pro aplikaci dostačující. Naopak nižší obnovovací frekvence nevyužívá tolik systémových prostředků. [16]

3.2.2.4 Měřicí karta

K měření dat z detektoru (typ MCP detektoru od firmy Photonis) se používá karta AP 200 od společnosti Acquiris (nyní Agilent). Jedná se o osciloskopickou kartu s vlastním průměrováním. Vlastní *averager* dovoluje zpracovat velký objem dat bez zatížení řídicího PC. Proces analýzy vzorku může pracovat, až na frekvenci 2 kHz, což je frekvence laseru, při které má ještě dostatečný výkon. Karta také

disponuje vlastním programovatelným hradlovým polem. Následují specifikace měřící karty:

- vzorkování: až 2 GHz
- paměť: 4 × 256 KB [17]

3.2.2.5 Řídicí PC

Řídicí PC musí splňovat požadavky na vysokou propustnost sběrnic a vysoký výpočetní výkon. Z měřící karty je vyčítáno velké množství dat a v reálném čase se tato data zpracovávají a ukládají. Specifikace důležitých komponent jsou následující:

- základní deska: Gigabyte P55-UD3L
- procesor: Intel Core i5-760 Quad-Core
- paměti: Kingston 4GB DDR3 1333MHz

3.2.3 Stávající software

Stávající software byl, stejně jako software pro nanášecí zařízení, vyvinut v prostředí NI LabVIEW. Struktura je i v tomto případě založena na principu „stavového stroje“, včetně paralelizace jednotlivých procesů. Opět je zde tedy možné pozorovat rozdělení na *event* strukturu, řídicí proces, proces ovládání motorů, proces ovládání měřící karty a proces ovládání kamery.

Z pohledu uživatele je aplikace rozdělena na několik jednotlivých režimů. Jedná se o režim „imaging“, při kterém je z desky vypalována souvislá plocha vzorku. Tímto vzniká síť bodů s jednotlivými zprůměrovanými spektry. Následnou analýzou je možné dosáhnout zobrazení do mapy rozdělení intenzity. Druhým režimem je režim „small plate“. V tomto režimu se používá mírně odlišný typ desek, než u režimu „imaging“. Na desce jsou vyfrézované body, do kterých se nanášejí vzorky. Tím se zajistí jejich ohraničení a určí přesná poloha při analýze. Dalším režimem je určování analyzovaného bodu pomocí obrazu z kamery. Posledním měřícím režimem je režim osciloskopu. V tomto režimu laser nepřetržitě vypaluje vzorek a karta získává data, která se následně zobrazují. Slouží především demonstračním účelům a pro kontrolu nanášených vzorků. Aplikace obsahuje taktéž režim zobrazování naměřených dat. V něm je možné vybrat z několika druhů zobrazení, od jednotlivých spekter po zpracované zobrazení prostorových map intenzity.

Pohyb stolku je zde ovládán buď ručním, nebo plně automatickým procesem. Podobně jako u aplikace pro nanášecí zařízení je možné vytvořit mřížku

bodů, které budou analyzovány. V této aplikaci uživatel zadá parametry, se kterými se má měření uskutečnit a po spuštění procesu již do něj nemusí, a ani nemůže zasahovat. Naměřená data se ukládají do binárního souboru, protože objem ukládaných dat je velký. Tento soubor pak lze otevřít a analyzovat výše zmiňovanými metodami. [6]

3.2.4 Požadavky zadavatele na zlepšení

Požadavkem je v první řadě zrychlení celého procesu a zaměření se na režim „imaging“. Zachování ostatních režimů je volitelné, ale struktura programu musí být uzpůsobena jejich jednoduché implementaci. Zrychlení v režimu „imaging“ by mělo být dosaženo použitím zrcátka jako kompenzace pohybu stolku v ose x. Požadavkem na analýzu naměřených dat je funkční kalibrace dat. Jedná se o přizpůsobení hmotnostní domény naměřených dat skutečným hodnotám. Naměřené hodnoty se vždy liší od skutečných, pomocí sady kalibračních látek, lze hmotnostní doménu přepočítat.

4 Formulace cílů

Jak již bylo uvedeno v předchozí kapitole, zadavatel přišel s novými požadavky na funkcionalitu a na opravení stávajících závad, které komplikovaly práci. Po analýze kódů aplikací a konzultacích s obsluhou bylo překročeno k vytvoření zcela nového kódu, který by splňoval všechny požadavky zadavatele. Současný stav software to neumožňoval, protože struktura nebyla od počátku tvořena jako modulární. Struktura byla vytvořena pro jednoúčelovou aplikaci, a s přibývajícimi požadavky na nové funkce, se příliš rozrostla, přestala být přehledná a každý zásah do ní by představoval velké změny v kódu. Stejně tak se muselo změnit uživatelské rozhraní, které bylo ve stávající podobě málo přehledné a obsahovalo mnoho duplicitních nebo pro uživatele nepotřebných prvků.

Vzhledem k odlišným požadavkům zadavatele a nutnosti řešení pozorovaných chyb nejsou v seznamu cílů obsaženy všechny body zadání diplomové práce. Jedná se o filtraci signálu a o odstranění vlivu matrice a pozadí z naměřených dat. Po konzultaci se zadavatelem bylo zjištěno, že výše zmíněné body jsou pro něj nepotřebné nebo by se musely řešit převážně změnou hardwaru, což není v tuto chvíli prvořadé. Tyto změny se plánují do budoucna.

4.1 Nanášecí zařízení

Požadavky na změny jsou uvedeny v předchozí kapitole. Z nich formulované cíle práce jsou seřazeny níže:

- Zcela manuální ovládání procesu nanášení a to jak v režimu „dropping“, tak v režimu „piezo“. Obsluha má mít možnost pomocí směrových šipek vybrat bod k nanesení a podle vybraného režimu nanést vzorek. Buď otisknutím, nebo vygenerováním proudu kapek. V prvním režimu má mít možnost nastavit výšku zdvihu, o který se posune osa z. V druhém režimu má mít možnost nastavit tvar generovaného pulsu, a počet opakování. Obsluha může vybrat mezi jedním pulsem, dávkou pulsů nebo nepřetržitým dávkováním s určitou frekvencí.
- Automatický režim, který podporuje oba režimy nanášení. Podle zadaných parametrů se vypočte mřížka bodů, které se mají nanést. Obsluha má mít možnost měnit počáteční a koncové body, včetně velikosti rozestupu. Posuvy pak pohybují s deskou podle vypočtené dráhy a na každém bodě se v závislosti na režimu spustí daná sekvence. V režimu „dropping“ se k možnosti nastavení zdvihu přidává možnost nastavení doby čekání do dalšího otisknutí. Čekání je zde nutné z důvodu používané metody separace vzorků. Tím pádem obsluha nemusí měnit rychlosti posuvů a tím i délku čekání, což byl způsob používaný dříve. V režimu „piezo“, zůstává nastavení totožné, jen se pulsy generují automaticky po dojetí na souřadnice.
- Dostupnost obrazu z připojené kamery v „piezo“ režimu. Obsluha má mít možnost vizuálně zkontrolovat kvalitu generovaných kapek.
- Vytvořit možnost rozšíření o režim „fluorescence“, pro který prozatím není hardwarové vybavení.
- Vytvořit logickou a modulární strukturu.
- Zabránit výskytům pozorovaných chyb

4.2 Hmotnostní spektrometr

Z požadavků uvedených v předchozí kapitole byly formulovány následující cíle diplomové práce:

- Manuální režim. V tomto režimu má mít obsluha možnost se pohybovat s deskou pomocí kurzorových šipek a nastavený krok. V jakémkoli místě desky (v mezích softwarových limitů) pak může spustit sekvenci na získání a uložení dat.
- Automatický režim. V tomto režimu obsluha nastaví všechny potřebné parametry měřící karty a dráhu, po které se má pohybovat stolek. Po spuštění se celé měření provede automaticky.
- Režim „imaging“. Tento režim odpovídá automatickému režimu. Přes určitou plochu vložené desky je provedeno postupné měření a ukládání spekter do binárního souboru. Tento režim může pracovat ve dvou módech chování zrcátka, které příklání paprsek laseru na desku. První mód je známý z původního programu, po dojetí stolku na bod se zrcátko přikloní, laser vypálí vzorek a zrcátko se znovu odkloní. V druhém módu se zrcátko využívá jako kompenzace osy x . To by mělo vést k výraznému zvýšení rychlosti procesu. Rozsah kompenzace je možné navolit a tím i změnit množství bodů, na které stolek nemusí najet.
- Ukládání souborů dat a parametrů. Oddělení parametrů měření od binárního souboru spekter na jednu stranu zvýší počet souborů, na druhou stranu však povede ke zjednodušení programu. Parametry měření se uloží ve formátu XML, který je v LabVIEW nativně podporován. Struktura je však lehce modifikovatelná na jiný formát ukládání dat.
- Vytvořit možnost rozšíření o další režimy.
- Vytvořit logickou a modulární strukturu.
- Zabránit výskytům pozorovaných chyb
- Vytvořit aplikaci, která bude zobrazovat naměřená data ve 2D a 3D. Nemusí mít modulární strukturu, protože se počítá s jejím využitím jen do doby, než se začne používat nějaký otevřený formát ukládání dat a analýza tedy bude moci probíhat ve specializované aplikaci třetí strany.

5 Řešení stávajících chyb

Vzhledem k rozdílným problémům obou zařízení bylo nutné pro každé z nich přistoupit na jiný postup řešení. U nanášecího zařízení bylo nejdříve potřeba prostudovat systém, se kterým se ovládají posuvy, a zda je v tomto ohledu možné něco zlepšit. U hmotnostního spektrometru byl hardware v zásadě funkční, ale bylo třeba kalibrovat polohu stolku a laseru.

5.1 Řešení problémů nanášecího zařízení

Hlavní problém nanášecího zařízení bylo to, že při nanášení se občas přestala pohybovat jedna nebo více os. Problém byl přičítán složitému kódu a faktu, že veškerá komunikace mezi PC a kontrolérem probíhala změnou globálních proměnných v LabVIEW. Po bližším prozkoumání se zjistilo, že ke kontroléru je dodáván software, který umožňuje nastavovat různé parametry a ukládat je do profilů podle používaného modelu zařízení. V předešlém kódu tyto profily nebyly použity a pracovalo se se základním nastavením (default profile). Pouhou změnou tohoto profilu se odstranily některé závady. Jedná se o chybné nastavení délky posuvů a koncových spínačů. V LabVIEW se však musely uzpůsobit orientace os, protože dodávaný software tuto funkci sice podporuje, ale ne zcela. Změnou orientace osy se zaměnila pouze poloha koncových spínačů, logika otáčení motorů zůstala stejná.

Ač došlo k výraznému zlepšení chování nanášecího zařízení, odpadla například potřeba přepočítávat polohu a kontrolovat, zda již není posuv na koncovém spínači, při kterém stále docházelo k výpadkům pohybu posuvů. Bylo zjištěno, že po určité době se motory samy vypínají. Nebyla zjištěna žádná závislost na teplotě motorů, počtu ujetých kroků nebo době zapnutí. V dodávaném softwaru je sice možnost automatické vypínání motorů vypnout, ale ani po vypnutí této funkce problémy neustaly. Bylo přikročeno k řešení přímo v LabVIEW. Před každým příkazem na změnu polohy je kontrolováno, zda je motor zapnutý, pokud není, vyšle se příkaz na jeho zapnutí.

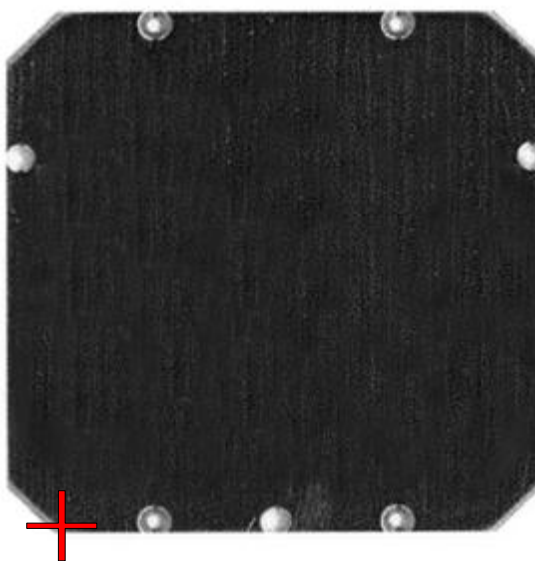
K ovládání kontroleru slouží v LabVIEW sada funkčních bloků, které jsou dodávány výrobcem posuvů. Všechny funkční parametry jsou však v globálních proměnných, což může být i nadále zdrojem problémů. [8]

5.2 Řešení problémů hmotnostního spektrometru

Problém hmotnostního spektrometru spočíval především v kalibraci polohy laseru a posuvů, tak aby bylo vždy jasně určeno, od kterého bodu se odvozuje souřadný systém. Vzhledem k omezeným možnostem pohybu v ose x (používá se

posuv s kratší vzdáleností, než je rozměr desky) byl za počáteční bod vybrán levý dolní roh. Na obrázku Obrázek 5 označeno červeným křížem. Vertikální poloha laseru se upravila mechanickým natočením zrcátka, horizontální pak nastavením určité nulové polohy napětí přiváděného na svorky zrcátka. Poté se nastavila poloha stolku pomocí manuálního režimu ovládání. Počáteční poloha se dá vždy znovu obnovit pomocí vytvořené funkce „homing“, která nalezne koncové spínače posuvů a najede na definované souřadnice.

Druhým problémem posuvů byly špatně fungující koncové spínače, které se spínaly, i když sepnuté nebyly. Tento problém se dal vyřešit buď výměnou spínačů za jiné, nebo softwarově v LabVIEW. Koncové spínače se v minulosti již několikrát měnily a vzhledem k jejich vysoké ceně bylo zvoleno druhé, softwarové, řešení. Byla proto vytvořena funkce, která kontroluje, zda požadované souřadnice nepřekročí nastavenou hranici. Tím pádem mohly být hardwarové koncové spínače vypnuty.



Obrázek 5 - Umístění počátku souřadného systému

6 Popis vyvíjených aplikací

Veškerá funkcionální se v NI LabVIEW vytváří programováním funkčních bloků, které si lze představit jako funkce se vstupy a výstupy. Každý funkční blok má v systému svou grafickou reprezentaci. Rozsáhlé funkce a se skládají do podoby tzv. *SubVI*. Prvky uživatelského rozhraní jsou tvořeny vstupy tzv. *controly* a výstupy tzv. *indicatory*, které mohou být různých datových typů: čísla, text, grafy, atd. Zvláštní postavení, mezi prvky uživatelského rozhraní, mají tzv. *XControly*, které mohou plnit funkci jak *controlů*, tak *indicatorů*. *XControl* se zároveň chová jako samostatné *SubVI*, ve kterém se zpracovávají vstupy a převádí na výstupy. Těchto prvků je ve vyvíjených aplikacích s úspěchem využito.

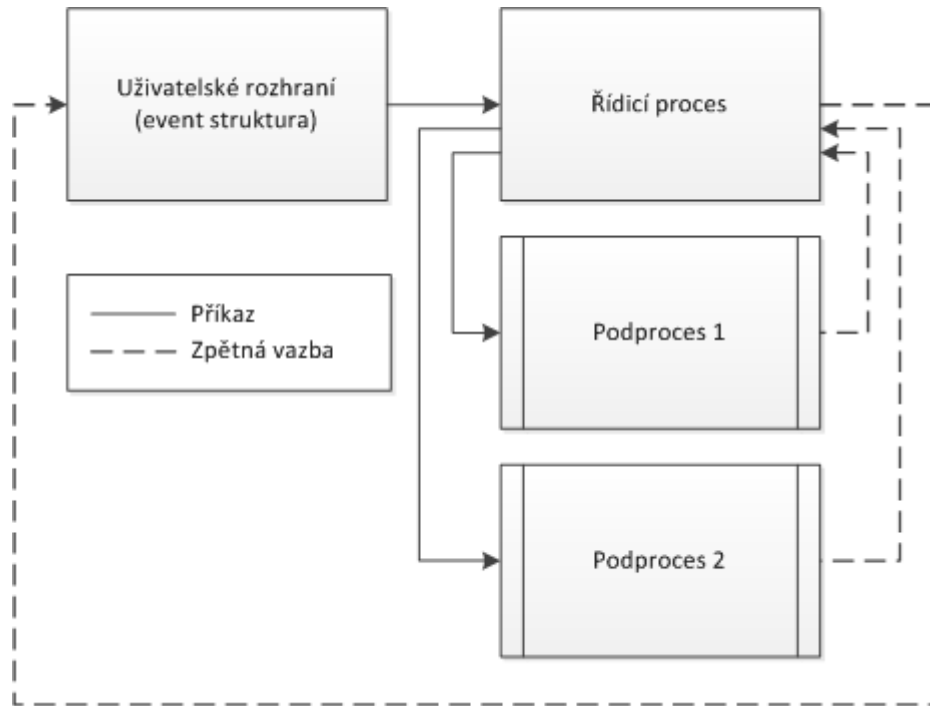
Struktura aplikací obou zařízení byla navržena stejná. Aplikace byly navrhovány současně „zevnitř“, z pohledu programátora a struktury kódu, i z „vnějšku“, z pohledu uživatele.

6.1 Obecný popis struktury aplikace

Vzhledem k požadavkům na modulární program byla vybrána struktura principem odpovídající stavovému stroji s paralelními procesy (vlákny). Chování aplikace je tedy řízeno stavy, které se řeší v řídicím procesu. Řídicí proces reaguje na příkazy z uživatelského rozhraní a podle stavu, ve kterém se nachází, rozesílá příkazy do ostatních procesů. Jedná se tedy o stejnou strukturu, jaká byla použita v původní aplikaci. Oproti původní aplikaci byl však program striktně rozdělen na řídicí a řízenou část. Pouze řídicí proces posílá příkazy ostatním (podřízeným) procesům a čeká na jejich reakci.

Dalším rozdílem je také to, že veškeré vstupy od uživatele se řeší na jednom místě pomocí *event* struktury. Tato struktura zachytává události z uživatelského rozhraní. To, že jsou všechny vstupy na jednom místě, sice komplikuje předávání informací do procesu, pro který je informace určena, nicméně má to i své výhody. V řídicím procesu je možné tyto informace rozdistribuovat pomocí systému *front*.

Pomocí *front* se z řídicího procesu posílají příkazy a parametry potřebné v daném procesu a nazpět se vrací jednoprvková fronta stavů tzv. *notifier*. Díky zpětné vazbě z *notifieru*, má řídicí proces přehled o tom, zda byl, či nebyl proveden příkaz a zda již je podřízený proces připraven přijmout další příkazy. [18]



Obrázek 6 - Struktura aplikace

6.2 Popis aplikace pro nanášecí zařízení

Aplikace pro ovládání nanášecího zařízení byla vyvíjena jako první v pořadí a při jejím vývoji bylo zjištěno mnoho cenných poznatků, kterých bylo využito ve vývoji dalších aplikací. Struktura se, s mírnými úpravami, drží návrhu z kapitoly 6.1. První popisovanou částí je *event* struktura, která řeší vstupy z uživatelského rozhraní.

6.2.1 Event struktura

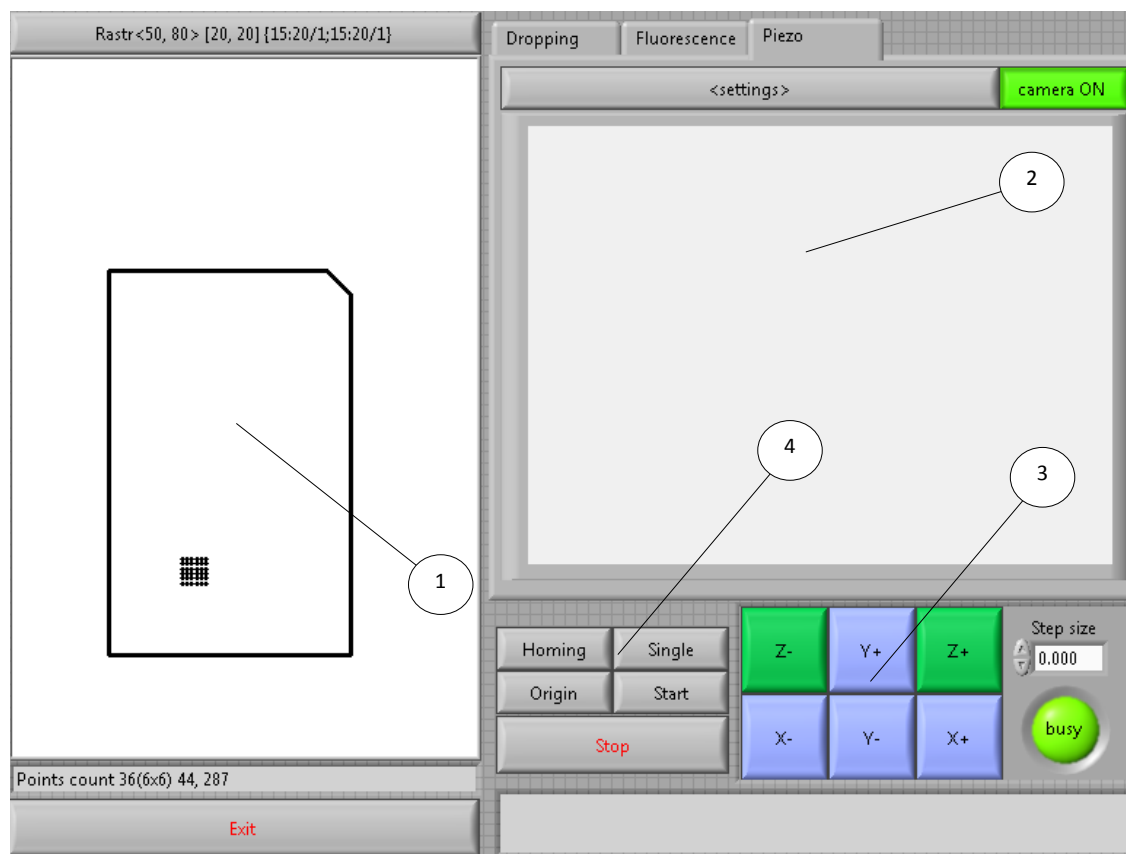
Přes poměrně velký počet ovladačů, které může uživatel používat a nastavovat, je *event* struktura v celku jednoduchá. Toho bylo dosaženo pomocí rozdělení uživatelského rozhraní na několik jednotlivých *XControlů*. Vzhledem k možnosti předzpracování lze *XControl* využít jako jediný výstup. Tím se výrazně sníží počet prvků, které je nutno v blokovém diagramu zapojit.

Podle použitého *XControlu* se do fronty stavů vloží určitá informace pro řídicí proces. Jedná se konkrétně o tři různé možnosti: příkazy manuálního ovládání, příkazy automatického ovládání a příkaz k ukončení aplikace. Součástí příkazu manuálního ovládání je směr, ve kterém se mají posuvy pohybovat, a relativní souřadnice pohybu. Příkaz automatického ovládání sestává z následujících informací: informace o režimu, ve kterém se uživatel nachází (dropping, piezo), parametry konkrétního režimu (délka čekání, zdvih, nastavení pulzu), informace o dráze pohybu a konkrétní příkaz (start sekvence, ukončení,

najetí na počáteční bod aj.). Příkaz na ukončení aplikace pouze vloží tento požadavek do fronty a řídicí proces rozdistribuuje příkaz do všech podprocesů.

Z pohledu uživatele je tedy aplikace rozdělena do několika částí, podle způsobu ovládání a nastavení. Jedná se o jednotlivé *XControl*y: Nastavení dráhy pohybu, automatické a manuální ovládání, stavový řádek a kontrolka zaneprázdnění a nakonec sada záložek s ovládáním různých režimů.

Vzhledem k tomu, že práce navazuje na předchozí řešení, jsou některé funkce, které byly shledány jako vyhovující, totožné. Zde se jedná o nastavení pulzu pro piezopipetor, které bylo upraveno a vloženo do samostatného *XControlu*. Naprogramování tohoto *XControlu* je navíc náplní práce jiného řešitele. Jedná se tedy o syntézu několika řešení. Stejně platí pro *XControl* řešící nastavení dráhy pohybu stolku. Součástí těchto nastavení je možnost vytvářet různé profily, které se ukládají ve formě XML v definovaném adresáři.



Obrázek 7 - Uživatelské rozhraní aplikace pro nanášecí zařízení:

- 1) nastavení dráhy pohybu, 2) nastavení jednotlivých režimů, 3) manuální ovládání,
4) automatické ovládání

6.2.1.1 Nastavení dráhy pohybu

XControl nastavení dráhy pohybu sestává ze zobrazení nastavené mřížky a tlačítka, které otevírá vlastní nastavení. Uživatel má možnost nastavit rozsah, ve kterém se bude mřížka bodů vytvářet, a nastavit odstup jednotlivých bodů. Obojí ve dvou osách, x a y .

Výpočet dráhy funguje následovně. Spočítá se rozdíl mezi počátečním a koncovým bodem v každé ose a podělí se velikostí kroku. Takto získaný zaokrouhlený počet bodů je znovu vynásoben velikostí kroku a každý jednotlivý bod je přičten k počátku. Takto vzniknou pro obě osy pole bodů, ve kterých se má stolek zastavit. V dalším kroku se z hodnot těchto dvou polí vytvoří dráha pohybu, podobná „hadu“. To znamená, že po dosažení maximální souřadnice v jedné ose, se stolek posune v druhé ose. Toho je dosaženo použitím dvou vnořených smyček *for*, do kterých vstupují počty bodů v jednotlivých osách. Nejdříve se dosáhne maximální hodnoty pro osu x , a následně dojde k posunu na další hodnotu v ose y . Pro každý lichý řádek osy y se obrací pořadí bodů v ose x . Takto vzniklá dráha, vytváří hadovitý pohyb stolku. Výstupem je pole *clusterů*, které obsahují souřadnici x a y . Počet prvků pole je dán počtem bodů, které se mají nanést.

Zobrazení mřížky probíhá v samostatné smyčce, ve které se k původnímu obrázku prázdné desky postupně přidávají body z pole. Souřadnice x a y obrázku jsou totožné se souřadnicemi, které se mají nanést. Provádí se tak kontrola nanášení.

Samotné zadávání parametrů obsluhou je upraveno pomocí funkčních bloků „in range and coerce“, které v případě zadání hodnot mimo rozsah, automaticky zaokrouhlují data na definované hranice. Nemůže se tak stát, že by se vzorky měly nanést mimo plochu desky, nebo že by posuvy dorazily až na koncové spínače.

6.2.1.2 Nastavení pulzu pro piezopipetor

XControl pro nastavení pulzů má stejně jako předchozí jeden zobrazovací prvek a tlačítko na vyvolání samotného nastavení parametrů. Zobrazovacím prvkem je zde display obrazu připojené kamery, s jehož pomocí se vizuálně kontroluje správnost nastavení pulzů. Obraz z kamery lze tlačítkem buď zapnout, nebo vypnout. V případě správného nastavení pulzů již může být kamera vypnuta a nezatěžovat tak řídicí počítač. Samotné zpracování obrazu probíhá v jiném

Druhým prvkem je tlačítko, které vyvolá okno s nastavením pulzu. Nastavené parametry pulzu se zobrazí do grafu a ve formě *clusteru* pošlou do fronty příkazů. Zpracování těchto parametrů probíhá až v procesu, který ovládá

kontrolér piezopipetoru. Uživatel tak zvolí tvar pulzu a způsob jakým má být generován. Jedná se o generování jednotlivého pulzu na bod, dávky pulzů nebo dávkování s konstantní frekvencí.

6.2.1.3 Nastavení režimu „dropping“

Pro nastavení je také použit *XControl*, ale jeho výhody nejsou plně využity. Ovládací prvky jsou dva: délka sekvence a sestup ramene. Délka sekvence odpovídá době mezi dvěma sestupy ramene. Je tak docíleno plynulého nanášení s jasně definovaným trváním, které je potřebné u některých metod přípravy vzorků. Zobrazovací prvek tohoto *XControlu* je v tuto chvíli nevyužitý, ale zůstává možnost ho upravit tak, aby ukazoval, kolik času zbývá do dalšího otisknutí kapky, která se tvoří na konci kapiláry, nebo jiná informace.

6.2.1.4 Nastavení režimu „fluorescence“

Přestože režim fluorescence není funkční, není dostupné potřebné vybavení, je pro něj vytvořen *XControl*, který dovoluje nastavit dobu fluorescence a výšku, ve které se má pohybovat rameno s čidlem. Zobrazovacím prvkem je graf, do kterého se zobrazuje naměřená intenzita fluorescence.

6.2.1.5 Manuální ovládání

XControl pro manuální ovládání nabývá podoby šesti tlačítek a nastavení velikosti kroku relativního posunutí. Pro každou osu jsou vyhrazena dvě tlačítka, pohyb v kladném a záporném směru. Výstupem je pak příkaz a délka kroku, které vstupují do fronty příkazů přímo.

6.2.1.6 Automatické ovládání

XControl pro automatický režim je tvořen tlačítky, které reprezentují jednotlivé příkazy, které chceme vykonat. V závislosti na zvoleném příkazu se do fronty vkládá určitá kombinace parametrů. Při zastavování procesu tlačítkem „stop“, například není potřebná informace o dráze pohybu. O jednotlivých příkazech je pojednáno v další kapitole.

6.2.2 Řídicí proces

Řídicí proces je jádrem celé aplikace, v ní se vyčítají vstupující požadavky uživatele, dále se zpracovávají a následně distribuují příkazy do jednotlivých podprocesů. V každém průchodu se kontroluje, zda nejsou ve frontě nové příkazy. Pokud ano, kontroluje se, zda se řídicí proces nachází ve stavu, ve kterém může přijímat nové příkazy (ready), pokud se v něm nenachází, jsou příchozí příkazy

ignorovány a proces setrvává v nezměněném stavu. Tímto způsobem je zaručeno, že se automatické nanášení nepřerušuje po stisknutí tlačítka pro manuální ovládání nebo obráceně. Pro dosažení této funkcionality musí být řídicí proces rozšířen o paměť. Paměť je nejen v řídicím procesu, ale i ve všech ostatních procesech zastoupena *shift registrem*. V *shift registru* se uchovávají data z předchozího průchodu procesem. To znamená, že pokud se během iterace data obsažená v *shift registru* nezmění, zůstává proces ve stejném stavu. Paměť řídicího procesu obsahuje kromě informace o stavu řídicího procesu, i informace o stavech jednotlivých režimů a další. Na základě rozhodnutí, ve kterém stavu se bude proces nacházet, se spustí *SubVI* náležící danému stavu.

Pokud se proces nachází ve stavu „ready“, je připravena přijímat další požadavky od uživatele, jinak je neaktivní. Do stavu „init“ se řídicí proces dostává pouze po spuštění aplikace a provádí se tak inicializační sekvence, které v tomto případě znamenají pouze zjištění polohy posuvů. Stav „stop“ zajišťuje zastavení všech právě probíhajících sekvencí a přechod řídicího procesu do „ready“ stavu. Stav „exit“ pak znamená ukončení všech procesů. Stav „manual“ a „automatic“ jsou popsány v následujících podkapitolách.

Informace o tom, ve kterém stavu se zrovna nachází řídicí proces a informace o poloze, se kontrolují v každém průchodu procesem a následně zobrazují ve stavovém řádku. K tomu slouží samostatné *SubVI*.

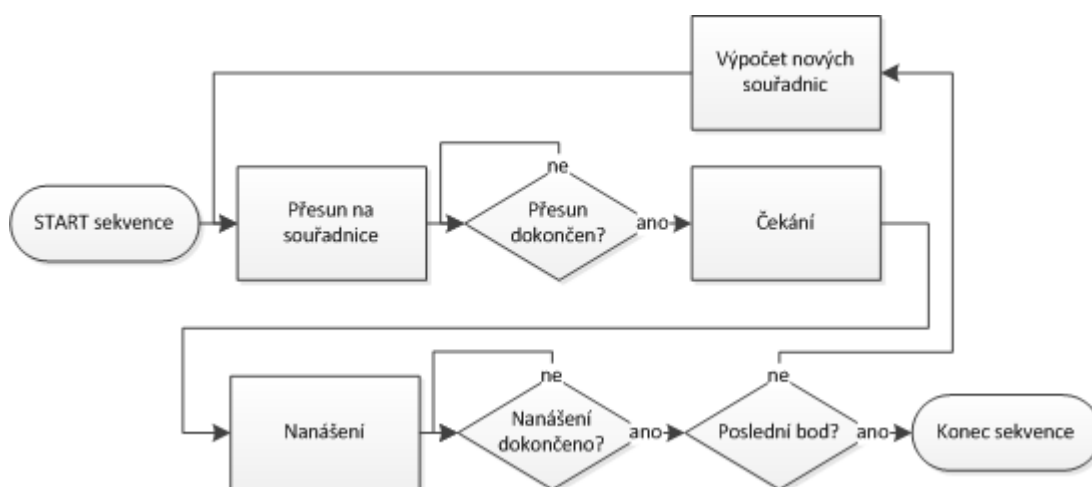
6.2.2.1 Stav „manual“

Pokud se řídicí proces nachází ve stavu „manual“ provádí se následující sekvence stavů. V prvním kroku se z informací z *event* struktury vytvoří příkazy pro proces ovládající pohyb stolku. Přiřazují se tedy osy a podle aktuální pozice se vypočítává cílová, již v absolutních souřadnicích. Tento příkaz se následně vloží do fronty a režim „manual“ přejde do stavu, ve kterém čeká na odezvu z procesu ovládajícího stolek. V tomto kroku se vyčítá *notifier*. *Notifier* v tomto případě obsahuje informace o tom, zda jsou všechny osy zastaveny a jakou mají pozici. Pokud *notifier* není vyčten nebo nejsou všechny osy zastaveny, setrvává proces ve stejném stavu a změní se pouze údaj o polohách os. Ve chvíli, kdy jsou všechny osy zastaveny, přechází režim do „ready“ stavu, ve kterém se i stav řídicího procesu přepíše na „ready“, a tím je možné přijmout další příkaz. Tímto způsobem ovládá obsluha nanášecího zařízení zcela manuálně.

6.2.2.2 Stav „automatic“

Ve chvíli, kdy uživatel stiskne jedno z následujících tlačítek: „start“, „origin“, „homing“; přejde řídicí proces do stavu „automatic“. Stejně jako manuální režim má i režim automatický svoje vlastní stavy, ve kterých pracuje. První z nich je stav „init“, který podle vstupujícího příkazu rozhoduje o tom, která sekvence se spustí. Příkaz „homing“ spouští samostatnou sekvenci v procesu posuvů a následně přechází do stavu, ve kterém se čeká na odezvu z tohoto procesu. Čekání na odezvu probíhá stejně, jako v předchozí podkapitole. Příkaz „origin“ načítá první bod z pole souřadnic vypočtené dráhy a předává tuto souřadnici do fronty procesu ovládání stolku. Následně se opět mění stav na čekání na odezvu.

Nejvýznamnějším příkazem je však příkaz „start“. Tento příkaz spouští sekvenci, která postupně najede na všechny body a podle zvoleného způsobu nanese vzorky. Sekvence začíná vyčtením polohy aktuálního bodu a přidáním daného příkazu do fronty posuvů. Informace o aktuálním bodu se udržuje v paměti řídicího procesu. Následuje čekání na odezvu, které v kladném případě vyústí v přechod do stavu „wait for time“. Tento stav řeší, zda se má čekat před přechodem na další souřadnici, používá se v módu „dropping“. Dalším krokem je vložení příkazu, který spouští sekvenci efektoru, do fronty efektoru. Do této fronty vstupuje nastavení módu, ve kterém chce uživatel nanášet. Stejně jako u příkazů pro proces posuvů, i zde následuje stav, ve kterém se čeká na odezvu z procesu efektoru. Posledním krokem je kontrola, zda je aktuální bod posledním, který se má nanést a pokud ne dojde k přepočtu aktuálního bodu. Pokud je aktuální bod posledním, dojde k přechodu řídicího procesu do stavu „ready“. Tímto způsobem projde řídicí proces všemi body nanášení, bez toho aniž by se do běhu dalo zasáhnout jiným způsobem, než tlačítkem „stop“.



Obrázek 8 - Sekvence automatického nanášení

6.2.3 Proces posuvů

Jak bylo popsáno v předchozí podkapitole, řídicí proces posílá ostatním podprocesům příkazy, které spouštějí vlastní sekvence příkazů, v daném podprocesu. V procesu posuvů jsou to příkazy určené k ovládní pohybu stolku a v některých případech i ramene. Proces posuvů běží nepřetržitě, dokud z řídicího procesu nepřijde příkaz „exit“, který proces ukončí. V každém průchodu se kontroluje, zda není ve frontě nějaký příkaz, čekající na zpracování, pokud je, provede se, pokud není, setrvává proces v předchozím stavu. Proces posuvů tedy opět obsahuje paměť ve formě *shift registru*. V tomto případě se přenáší *cluster* stavu, obsahující informaci o příkazu a informace o osách, a pole *VISA resources*, což jsou reference na zařízení, která pro komunikaci využívají standardu *VISA* (GPIO, Ethernet, UART). Zde jsou to tři karty kontroléru motorů. Jednotlivé příkazy odpovídají stavům, ve kterých se proces může nalézat. Každý stav obsahuje jedno nebo více *SubVI*, které se postupně vykonávají.

Do „ready“ stavu se proces dostane, buď příkazem z řídicího procesu, nebo po průchodu procesu ve stavu „homing“. V „ready“ stavu je proces nečinný a nespouští se žádná další *SubVI*.

Ve stavu „init“ se spouští inicializační sekvence. Součástí těchto sekvencí je: přiřazení identifikátorů jednotlivým osám podle jejich sériových čísel, načtení parametrů pro jednotlivé osy nanášecího zařízení a nakonec vynulování čítače polohy. Přiřazování identifikátorů podle sériových čísel je nutné, protože na různých zařízeních, kde se aplikace spouští, je z neznámých důvodů různé pořadí zapojení os. Druhým krokem je tedy načtení parametrů. To se provádí načtením profilu osy, který je uložen v definovaném adresáři. Tento profil se vytváří pomocí software, který se dodává spolu se zařízením. V předchozí verzi aplikace nebylo těchto profilů využito a tím pádem se s posuvy zacházelo jako s neznámým zařízením se základním (defaultním) nastavením. Nyní je tedy jasně definováno, s jakým modelem se pracuje, čímž je dosaženo práce v reálných jednotkách, dříve bylo nutné přepočítávat počet kroků motoru. Navíc nebyl nastaven rozsah a logika posuvu, docházelo tak k potenciálně nebezpečným situacím, kdy posuv dojel na koncový spínač, který neseplnul. V jednom případě špatně definovaného pohybu tak došlo k přetržení pružné spojky hřídele motoru a šroubu. Všechny nastavované parametry se zapisují do globálních proměnných, které slouží jako paměť a současně jako indikátor stavu kontroléru motorů. Tento způsob ovládní vytvořil výrobce a přesto, že se i do budoucna může stát zdrojem problémů, nebyl v rámci práce měněn, neboť není prioritou zadavatele. Tou bylo hlavně rychlé opravení stávajících chyb, což vývoj nových ovladačů vylučuje.

Pokud z řídicího procesu přijde příkaz „homing“, přejde proces posuvů do stavu, ve kterém se resetuje poloha os. Toto zajišťuje samostatné *SubVI*, které neplní jinou funkci. Nejprve jsou zadány souřadnice bodů mimo pracovní prostor, tedy až za koncovými spínači a je vykonán příkaz k pohybu os. Následně se kontroluje, zda jsou všechny tři osy na některém limitu. Pokud ano, jsou nastaveny hodnoty poloh koncových spínačů. Pro každou osu je tato hodnota jiná, protože každá osa má jinou počáteční polohu. Následně je zadán příkaz na pohyb do polohy nula. Nyní se čeká, až všechny posuvy, dojedou na požadovanou nulovou polohu a tím je kalibrace dokončena.

Proces posuvů přechází do stavu „move“ ve chvíli, kdy je z řídicího procesu zadán příkaz na změnu pozice, to se děje jak v manuálním, tak automatickém režimu. Příkaz má tvar pole jednoho až tří *clusterů*, podle počtu os, které se mají posunout na novou pozici. Zpracování příkazu probíhá v samostatném *SubVI*, ve kterém se přepisují hodnoty „Destination Position“ pro jednotlivé osy v globální proměnné nastavení kontroléru. Příkazem „Start“, z knihovny funkčních bloků vytvořené výrobcem, se předává příkaz přímo do kontroléru.

Na předchozí stav navazuje stav „wait“, ve kterém se vyčítá stav jednotlivých os, konkrétně rychlost a pozice posuvu, a po zpracování se odesílá ve formě *notifier* zpět do řídicího procesu. Toto se děje dokud z řídicího procesu nepříjde příkaz na přechod do stavu „ready“.

Posledním stavem procesu posuvů je stav „exit“. V něm se uzavírají reference na zařízení *VISA* a dochází k zastavení smyčky *while*, ve které se celý proces nachází.

6.2.4 Proces efektoru

Poslední proces, který má přímo vliv na chování zařízení, ovládá kontrolér pro piezopipetor a kontrolér posuvů. Přichystáno je i rozšíření pro ovládání zařízení měřící fluorescenci. Proces efektoru je rozdělen do dvou vrstev, podobným způsobem jako hlavní řídicí proces. V první vrstvě se řeší, ve kterém režimu má proces pracovat, zda „dropping“ či „piezo“. Druhá vrstva již řeší konkrétní stavy daných režimů. Jednotlivé režimy jsou zastoupeny formou *SubVI*. Nastavení obou zařízení probíhá v samostatném režimu „none“, který slouží i k ukončení referencí a současně jako „ready“ stav celého procesu.

6.2.4.1 Režim „dropping“

V režimu „dropping“ se vzorky nanáší otisknutím kapky, vytvořené na konci kapiláry. Toho je docíleno sjezdem ramene, které je připojeno na posuv osy z. V tomto režimu je tedy znovu ovládán kontrolér posuvů. Rozdělení do dvou

různých procesů má svoje opodstatnění, odděluje se tak posuv stolku a ramene a navíc se všechny současné i připravované režimy nanášení řídí z jednoho místa. Nanášení v režimu „dropping“ prochází několika následujícími stavy.

Stav „ready“ je zde zastoupen jen jako doplněk, nemělo by k němu dojít.

Ve stavu „init“ se načítají parametry nanášení, tedy aktuální pozice osy z a velikost sjezdu. Doba čekání na další nanášení se řeší v hlavním řídicím procesu. Tyto dva parametry se uchovávají v „shift“ registru, aby byly dostupné dalším stavům. Tento stav se provádí pouze na začátku samotného nanášení.

Ve stavu „move“ se spouští *SubVI*, které vyšle příkaz do kontroléru na posun o zadaný krok dolů a následně zpět. Součástí tohoto *SubVI* je kontrola stavu osy z (rychlost a poloha).

Na předchozí stav navazuje stav „wait“, ve kterém se odesílá *notifier* s informací o tom, že je ukončen pohyb ramene.

Posledním stavem je „stop“ stav, ve kterém se maže *shift registr* a čekající *notifier*, pokud takový existuje. Tím je dosaženo čistého nastavení pro další nanášení.

6.2.4.2 Režim „piezo“

V režimu „piezo“ se vzorky nanáší vystřikováním kapiček pomocí piezopipetoru. Kontrolér piezopipetoru generuje, na základě zadaných parametrů, elektrické pulzy, které budí piezokrystal. Nastavení kontroléru se provádí posláním sekvence znaků ve formátu *string* přes *VISA*. Stejným způsobem se do něj posílají všechny příkazy na započítí a ukončení generování pulzů. Nanášení v režimu „piezo“ prochází několika následujícími stavy.

Stav „ready“ je zde znovu zastoupen jen doplňkově. V případě, že by do procesu přišel nerozpoznaný příkaz, přejde do toho stavu.

Stav „init“ vkládá do *shift registru* informace o nastavení kontroléru. Současně se toto nastavení posílá do kontroléru samotného pomocí bloku „*VISA write*“. Tento funkční blok se používá ke vkládání všech příkazů. Předtím, než se však nějaké příkazy pošlou, musí se parametry převést na sekvenci znaků. To zajišťuje samostatné *SubVI*.

Ve stavu „move“ se na základě režimu vstřikování rozhoduje, jaké sekvence znaků se odešlou. Na výběr je režim jednotlivého pulzu, dávky a frekvence.

Stav „wait“ navazuje na předchozí a čeká se v něm na dokončení nanášení. Pokud se jedná o nanášení jednoho pulzu nebo nanášení s konstantní frekvencí, nečeká se a okamžitě se posílá *notifier*. Pokud se ovšem jedná o nanášení dávkou,

čeká se pomocí funkčního bloku „Wait (ms)“ definovaný počet milisekund. Délka čekání se odvíjí od počtu pulzů a periody, se kterou se mají generovat.

Stav „stop“ opět vymazává čekající *notifier*, pokud takový existuje a do kontroléru vkládá sekvenci ukončovacích znaků.

6.2.5 Proces kamery

Proces ovládání kamery je umístěn ve vlastní smyčce. Proces kamery není na rozdíl od ostatních procesů, řízen z hlavního řídicího procesu, ale přes výše uvedený *XControl* pro nastavení režimu „piezo“. Vlastní *SubVI* ovládání kamery se může nacházet ve čtyřech stavech.

První z nich je stav „ready“, ve kterém se kontroluje, zda uživatel nestiskl tlačítko na zapnutí kamery. Pokud ano, změní se do dalšího průchodu stav procesu na „CamStart“.

Pokud se proces nachází v tomto stavu, spouští se funkční blok „Start Aquisition“, který podle předem definovaných nastavení spouští získávání obrazu z kamery. Stav procesu se současně mění na „CamRunning“.

V tomto stavu se vyčítá získaný obraz a posílá se do výše zmiňovaného *XControlu*, ve kterém je display na zobrazování tohoto obrazu. Současně se kontroluje, zda je stále požadována akvizice obrazu. Pokud ano, stav zůstává stejný a v dalším průchodu se vyčte další snímek. Pokud se má získávání dat zastavit, přejde proces do stavu „CamStop“.

V případě, že se proces nachází v tomto stavu, spustí se funkční blok „Stop Aquisition“, který zastaví získávání dat z kamery a proces opět přejde do stavu „ready“.

6.2.6 Inicializace a ukončení procesů

Jak již bylo zmíněno, posílání příkazů probíhá pomocí systému front, které vedou z hlavního procesu. Tento systém front musí být po startu aplikace, před vstupem do jednotlivých procesů, inicializován. Inicializace probíhá pomocí funkčního bloku „Obtain Queue“. Jako vstupující parametry slouží datový typ vytvářené fronty a její název. Stejným způsobem je inicializován systém *notifierů*, které slouží jako odezvy z jednotlivých podprocesů.

Po řádném ukončení všech procesů, je nutné ukončit i systém front a *notifierů*. To zabraňuje chybám, které by mohly vzniknout. Pokud není aplikace ukončena touto cestou, mohou v paměti zůstat informace o frontách a o prvcích v nich obsažených.

Výjimkou je proces kamery, který nepoužívá systém front ani *notifierů*. Jeho inicializace probíhá před vstupem programu do vlastní smyčky procesu.

6.3 Hmotnostní spektrometr

Původní předpoklad, že budou obě dvě aplikace vyvíjeny současně, nedošel naplnění. Zadavatel vyžadoval přednostní práci na nanášecím zařízení, protože na jeho zprovoznění čekala řada uživatelů. Tento fakt přinesl jisté výhody, při vývoji první aplikace byly vyzkoušeny jisté postupy, které se s postupem času ukázaly jako méně výhodné a při vývoji druhé aplikace od nich bylo upuštěno. Nevýhodou však bylo to, že bylo nutné přepsat některé části kódu, které byly původně zamýšleny k použití v obou aplikacích. Struktura aplikace pro hmotnostní spektrometr se však neliší od té, která je popsána v kapitole 6.1. První popisovanou částí je opět *event* struktura, která řeší vstupy z uživatelského rozhraní.

6.3.1 Uživatelské rozhraní, event struktura

Stejně jako v aplikaci pro nanášecí zařízení, je i zde použito *XControls*, což výrazně zjednodušuje *event* strukturu. V závislosti na uživatelem právě použitým *XControlu*, se generují v *event* struktuře různé příkazy pro hlavní řídicí proces. *Event* struktura se dělí na pět různých případů. Jednotlivé *XControly* budou popsány po upřesnění uživatelských akcí.

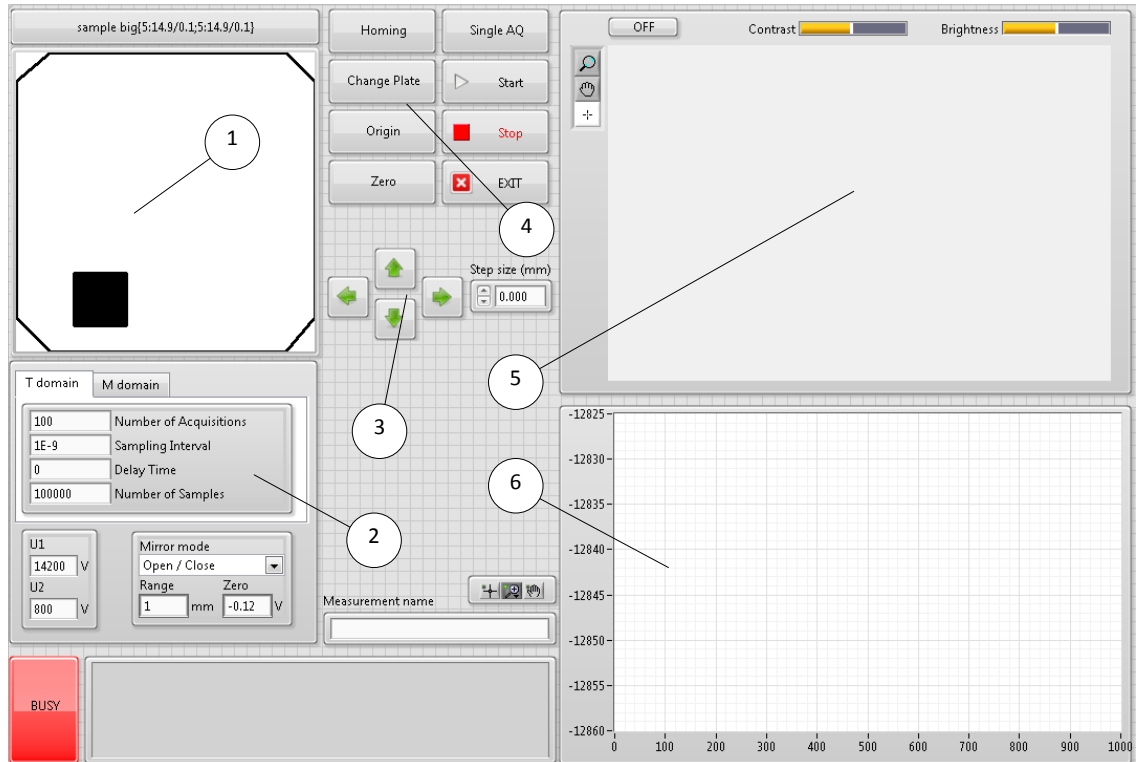
První z nich je „timeout“, tedy vypršení určitého nastaveného času. Pokud se po dobu 100 ms nezjistí žádná uživatelská akce, přechází *event* struktura do tohoto stavu. V něm se do stavového řádku posílají informace o stavu jednotlivých procesů a současně se mění hodnota kontrolky „busy“, která slouží k rozeznání, zda je program zaneprázdněn, či ne.

Druhou definovanou uživatelskou akcí je změna hodnoty „filename“. Tato hodnota slouží jako součást pojmenování souborů, které se během měření ukládají. K její změně je určeno textové pole v grafickém rozhraní.

Třetím stavem je identifikace uživatelem stisknutých kláves. V závislosti na tom, které klávesy, nebo jejich kombinace, uživatel stiskne, se vygeneruje příkaz pro hlavní řídicí proces. Jedná se o směrové klávesy, klávesu *shift* a klávesu *ctrl*. Pomocí těchto kláves může uživatel manuálně ovládat pohyb stolku s deskou. Základní velikost kroku se definuje v *XControlu* manuálního ovládání, v kombinaci se stisknutou klávesou *shift* se krok zdvojnásobí, v kombinaci s klávesou *ctrl* se naopak udělá krok poloviční. Pokud je stisknuta jakákoli jiná klávesa, žádný příkaz se negeneruje.

Čtvrtou definovanou akcí je změna hodnoty *XControlu* manuálního ovládání. Tou může být změna velikosti kroku nebo jeden ze čtyř směrů, kterým se může stolek pohybovat. Generuje se příkaz pro hlavní řídicí proces a vkládá se do fronty.

Posledním popisovaným stavem je změna hodnoty *XControlu* automatického ovládání. Podle hodnoty, která se v tomto *XControlu* vygeneruje, se do fronty příkazů pro hlavní řídicí proces pošle určitý příkaz. K tomu je zapotřebí celkově tři *XControlů*. *XControl* automatického ovládání, nastavení dráhy pohybu a parametrů měření. Následuje popis jednotlivých *XControlů*.



Obrázek 9 - Uživatelské rozhraní aplikace pro hmotnostní spektrometr:

- 1) nastavení dráhy pohybu, 2) nastavení parametrů měření, 3) manuální ovládání,
4) automatické ovládání, 5) obraz z kamery, 6) graf naměřených dat

6.3.1.1 Nastavení dráhy pohybu

Tento *XControl* je téměř totožný s tím, který je popsán v podkapitole 6.2.1.1. Následuje proto pouze popis odlišností. K výpočtu dráhy se využívá stejného algoritmu, ale vzhledem k tomu, že k analýze se dají použít jen desky jednoho typu, neobsahuje nastavení definice desky. Zobrazení analyzovaných bodů je také uzpůsobeno tomuto typu desek.

6.3.1.2 Nastavení parametrů měření

V tomto *XControlu* se nastavují různé parametry pro nastavení měřicí karty, velikosti urychlovacích napětí a nastavení chování zrcátka.

Nastavení měřící karty se může provádět ve dvou doménách, časové a hmotnostní. Měřící karta ve skutečnosti pracuje jen v doméně časové, a proto se nastavení zadané uživatelem v hmotnostní doméně, přepočítá do domény časové.

Jednotlivé parametry nastavení jsou počet měření, perioda vzorkování, počáteční hmotnost (zpoždění) a koncová hmotnost (počet vzorků).

Napětí, která uživatel zadává, odpovídají těm, která jsou nastavena na hardware hmotnostního spektrometru. Tato informace je důležitá pro další analýzu naměřených dat, konkrétně pro správný přepočet dat do hmotnostní domény.

Posledním možným nastavením je chování zrcátka. Zde si uživatel zvolí, zda chce, aby zrcátko pracovalo jako kompenzace osy x nebo běžným způsobem, tedy se pomocí zrcátka při každém naměřeném spektru laserový paprsek přiklonil a odklonil. Součástí nastavení je pak rozsah pohybu zrcátka v ose x a jeho nulová poloha ve voltech.

6.3.1.3 Manuální ovládání

K manuálnímu ovládání polohy stolku je určen samostatný *XControl*. Sestává ze čtyř tlačítek, každé pro jeden směr, ve kterém se může stolek pohybovat, a nastavením délky kroku, o který se má stolek pohnout. Jako výstup z *XControlu* je příkaz obsahující informaci o směru a velikosti kroku.

6.3.1.4 Automatické ovládání

K volbě automatických režimů slouží následující *XControl*, který obsahuje osm tlačítek, každé pro určitý příkaz. Jsou to tlačítka „homing“, pro reset polohy stolku, „change plate“, které slouží k výměně desky se vzorky, „origin“, pro najetí stolkem na první bod dráhy pohybu, „zero“, pro najetí na výchozí polohu, „single aq“, po jehož stisknutí se provede a uloží jedno měření, „start“, které započíná nastavenou sekvenci měření, „stop“, které tuto sekvenci ukončuje a „exit“, které vypíná všechny procesy a zastavuje běh aplikace.

6.3.1.5 Ovládání kamery

XControl pro ovládání kamery zde plní informativní funkci, uživatel může během měření kontrolovat, zda se vzorky vypalují správným způsobem. K ovládání kamery slouží tlačítka „on/off“, na zapnutí a vypnutí snímání, dále posuvníky „contrast“ a „brightness“ na změnu parametrů kontrastu a světlosti. Obraz z kamery lze přiblížit pomocí funkce „zoom“, posunout pomocí funkce „pan“ a lze v něm označit bod, do kterého míří paprsek laseru. Přestože byla provedena

kalibrace stolků a laserového paprsku, není možné zabránit malým výchylkám způsobených různými otřesy. Tato varianta se osvědčila více, než statický kříž.

6.3.1.6 Zobrazení měřených dat

Posledním prvkem uživatelského rozhraní je graf, ve kterém se zobrazují surová naměřená data. Tento prvek není součástí žádného *XControlu*. Z důvodu snížení zátěže se neprovádí žádný přepočítání zobrazených os nebo filtrace signálu, ta navíc podle zadavatele není nutná.

6.3.2 Hlavní řídicí proces

V hlavním řídicím procesu se vyčítají příchozí příkazy z uživatelského rozhraní, zpracovávají se a dále distribuují do ostatních podprocesů. Struktura procesu je stejná jako v případě aplikace pro nanášecí zařízení. Rozdíl je v tom, že stavy jednotlivých procesů se posílají ve formě *notifieru* do stavu „timeout“ v *event* struktuře, stejně tak se na konci každé iterace rozhoduje, zda má proces pokračovat, zda se nachází v jiném stavu než „exit“. Ve zmíněné aplikaci pro nanášecí zařízení je toto řešeno v samostatném *SubVI*.

Pokud se proces nachází ve stavu „ready“, je připraven přijímat další příkazy z uživatelského rozhraní. V opačném případě setrvává ve stávajícím stavu.

Ve stavu „init“ se spouští počáteční sekvence příkazů. Distribuuje se příkaz na inicializaci zařízení a čeká se na odezvu z podprocesu posuvů. Touto odezvou je získání aktuálních souřadnic posuvů.

Stav „stop“ zastavuje probíhající měření a resetuje paměť hlavního procesu. Současně se zjišťuje poslední pozice posuvů, aby bylo možné plynule navázat. Zastavením měření dojde k uzavření referencí na soubory, které se během měření vytváří, data však zůstanou zachována.

Ve stavu „exit“ dochází k rozeslání příkazu na ukončení všech podprocesů. Ukončí se i samotný hlavní řídicí proces.

Stav „manual“ má v podstatě stejnou podobu, jako v aplikaci pro nanášecí zařízení. Hlavní rozdíl se nachází v kroku „wait“, kdy se ve vyčítaném *notifier* nachází informace o poloze a o tom, zda je dokončen pohyb. Řídicí proces však již neposílá příkaz, k ukončení vyčítání stavu v podprocesu posuvů, přechod do „ready“ stavu si řídí podproces posuvů sám.

Poslední stav, který může nastat je stav „automatic“. Jeho popis je v následující podkapitole.

6.3.2.1 Stav „automatic“

Stejně jako předchozí stavy, i tento má svoje vlastní *SubVI*, ve kterém se provádí definované sekvence stavů. Program nejdříve vstupuje do stavu „init“, ve kterém se, podle příkazu příchozího z uživatelského rozhraní, řeší, jaká sekvence se spustí.

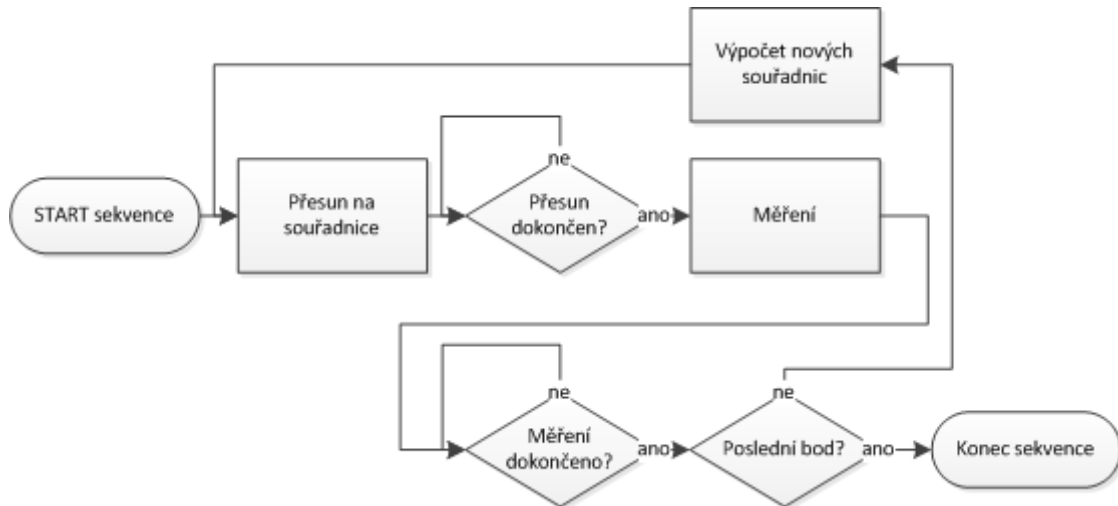
Pokud uživatel stiskne tlačítko „change plate“, vyšle se z *event* struktury příkaz „move to limit“, který zapříčiní distribuci tohoto příkazu do podprocesu posuvů. To má za následek, že posuvy najedou na koncové spínače a zastaví se. Obsluha má pak možnost vyměnit desku se vzorky. Po druhém stisku tlačítka „change plate“, se vyšle z *event* struktury příkaz „move from limit“. Změnu těchto příkazů řeší příslušný *XControl*. Příkaz „move from limit“ zadá procesu posuvů pokyn na přesun do nulových souřadnic. Následně se změní stav obou procesů na „wait“, ve kterém se čeká na dojetí do požadovaných souřadnic.

Příkaz „homing“ provádí výše popsanou sekvenci, jen automaticky a bez zastavení na výměnu desky. Do fronty příkazů jsou tak vloženy dva výše zmíněné.

Tlačítko „origin“ započne sekvenci stavů, které vedou k najetí posuvů na souřadnice prvního bodu vygenerované dráhy pohybu. Ve stavu „motors start“ se přiřadí tato první souřadnice a pošle se jako příkaz do fronty pro podproces posuvů. Následně přejde automatický režim do stavu „motors wait“, ve kterém se čeká na odezvu z podprocesu posuvů.

Stiskem tlačítka „single aq“ se pak ve stavu „init“ řeší otevírání referencí na soubory s daty a parametry a zaslání příkazu na nastavení parametrů do podprocesu měření. Příkaz obsahuje informace o počtu analyzovaných bodů, nastavení chování zrcátka a vlastní nastavení měřící karty. Počet bodů, nastavení měřící karty a použitá napětí se ukládají do souboru ve formátu XML, který slouží jako hlavičkový soubor pro naměřená data. Následuje přidání příkazu na započítání akvizice do fronty pro podproces měření a změnu stavu na „acqui wait“. V tomto stavu se čeká na odezvu z podprocesu, zda měření proběhlo.

Spojením dvou předešlých sekvencí vzniká sekvence, která proběhne při stisknutí tlačítka „start“. V prvním kroku se načtou parametry měření a vytvoří nové soubory s parametry a daty. Následuje zaslání příkazu na najetí do souřadnic vygenerované dráhy, v závislosti na čísle kroku. V dalším stavu se čeká na odezvu z podprocesu posuvů. Následuje vyslání příkazu na započítání měření do fronty podprocesu měření. Poté se čeká na odezvu z podprocesu měření. Pokud měření proběhlo, je v další části sekvence navýšena hodnota „step“, která určuje, který bod se má momentálně analyzovat. Jestliže se jedná o poslední bod analýzy, přechází řídicí proces v dalším průchodu do stavu „ready“.



Obrázek 10 - Sekvence automatického měření

6.3.3 Proces posuvů

Příkazy posílané z řídicího procesu do fronty podprocesu posuvů, jsou vyčítány s definovaným zpožděním. Pokud je toto zpoždění překročeno, nastává „timeout“ stav, ve kterém nejsou vyčteny žádné příkazy a nemění se stav procesu. V každém průchodu je současně kontrolováno, zda nepřišel příkaz „exit“, který ukončí běh procesu. Informace o předchozím stavu jsou udržovány v *shift registru*. Každý jednotlivý stav odpovídá sekvenci příkazů, které se mají vykonat.

Ve stavu „ready“ se proces nachází ve chvíli, kdy není plněn žádný příkaz. Přechod do tohoto stavu řídí sám proces posuvů, nikoli hlavní řídicí proces, jak je tomu u aplikace pro nanášecí zařízení. V „ready“ stavu je proces nečinný a nezatěžuje tak řídicí PC.

Stav „init“ je první, který proběhne. V něm se provádí inicializace zařízení a jeho nastavení. Parametry nastavení jsou uloženy v globální proměnné, která se nemění, je z ní vždy jen vyčítáno. K nastavení jednotlivých parametrů se používá funkčních bloků knihovny „NI 73XX“, což je součást knihovny „Vision and Motion“. Po zadání parametrů je proveden příkaz na reset polohy os a odeslání informací o stavu posuvů do hlavního procesu.

Funkce stavu „move to limit“ byla již částečně popsána v předchozí kapitole. Jedná se o příkaz, který hledá koncové spínače obou posuvů za účelem, buď výměny desky se vzorky, nebo kalibrace počáteční polohy. Tento stav je také jediným, ve kterém jsou zohledněny signály z hardwarových koncových spínačů. Ostatní funkce posuvů používají softwarové limity.

Kontrola softwarových limitů je prováděna ve vlastním *SubVI*, které porovnává aktuální polohu s nastavenými hodnotami limitů. Jsou použity funkční

bloky „in range and coerce“, vstupující požadovaná poloha je v případě překročení nastavených limitů zaokrouhlena. Tak je zabráněno vyjetí posuvů mimo ohraničený pracovní prostor.

Dalším již částečně popisovaným stavem je „move from limit“, který resetuje aktuální polohu posuvů na nulu, vyšle příkaz k přesunutí na výchozí souřadnice a znovu resetuje aktuální polohu. V tomto stavu nejsou použity ani hardwarové ani softwarové limity. Informace o stavu posuvů jsou pak posílány do hlavního řídicího procesu.

Pokud z hlavního procesu přijde příkaz „move“, je nejdříve zkontrolováno, zda se požadované souřadnice nevyskytují mimo povolenou hranici. Následně je příkaz zadán do kontroléru a proces posuvů přechází do stavu „wait“.

V tomto stavu se kontroluje, zda byly dokončeny pohyby obou posuvů (rychlost je nulová). Pokud ano, přechází proces posuvů do stavu „ready“, pokud ne, setrvává ve stavu „wait“. V každém průchodu je pomocí *notifieru* odeslán stav posuvů do hlavního procesu, kde se dále zpracovává.

Posledním možným stavem, který může nastat, je stav „stop“. V tomto stavu jsou okamžitě ukončeny pohyby obou posuvů a přechází se do stavu „wait“.

6.3.4 Proces měření

Stejně jako v procesu posuvů, je i zde vyčítání příkazů z hlavního procesu vykonáváno s určitým zpožděním a struktura tedy odpovídá, té popsané výše. Rozdíly lze nalézt v paměti procesu, která obsahuje informace jak o stavu, ve kterém se proces nachází, tak i o parametrech, které jsou důležité pro vyčítání dat z měřicí karty. Součástí paměti je i reference na *DAQmx task* zrcátka, což je způsob, kterým LabVIEW komunikuje se zařízeními od National Instruments. Proces měření tedy obsluhuje jak měřicí kartu, tak zrcátka, které zde slouží k odklápění paprsku laseru nebo jako kompenzace pohybu v ose x .

Inicializace obou zařízení se provádí ve stavu „init“. Co se týče nastavení zrcátka, to je možné provést v „MAX“. Nastavení měřicí karty je však nutné provést v samotném LabVIEW pomocí funkčních bloků z knihovny „AqDx“, která se dodává spolu s měřicí kartou. Nastavuje se hlavně režim, ve kterém karta pracuje (averager), a *trigger*.

Ve stavu „set param“ se nastavují parametry měření a vyčítání z karty, podle nastavení z uživatelského rozhraní. Tento stav předchází samotnému měření. Po přepsání paměti se proces měření znovu vrátí do stavu „ready“.

Stav „data aq“ se provádí ve chvíli, kdy z hlavního procesu dojde příkaz na započítání měření. Z paměti je vyčtena informace o nastavení zrcátka a podle toho,

který bod se má analyzovat, se zrcátko přikloní. Následuje měření dat, kdy měřící karta čeká na příchozí *trigger* z ovladače laseru. Když se *trigger* nedostaví do určité doby, je o tom obsluha informována. Následuje vyčítání dat. Data se vyčítají ve tvaru pole datového typu *int32* a posílají do fronty dat. Tato fronta je na jiném místě znovu vyčtena a zpracována. Tak dochází k urychlení procesu měření. Proces pak přechází do stavu „done“.

V tomto stavu se v závislosti na režimu zrcátka zpracovává stav procesu. V režimu „open/close“, tedy přiklání a odklání zrcátka po každém bodu, se pouze odesílá *notifier* do hlavního procesu s informací o tom, že měření bylo dokončeno a stav procesu měření se mění na „ready“. V případě, že je zrcátko v režimu „X axis compensation“, tedy kompenzace pohybu posuvů v ose x, posílá se *notifier* jen tehdy, až jsou provedena všechna měření v rozsahu zrcátka. To jaký bod se má momentálně analyzovat řeší *SubVI*, které vytváří pomyslný pohyb stolku. V každém kroku se mění hodnota aktuální natočení zrcátka podle toho, v kterém bodu se měření nachází. Pokud aktuální bod dosáhne hranice rozsahu zrcátka, je resetován a vyšle se *notifier* do hlavního procesu s informací o tom, že měření proběhlo. Pokud není dosaženo konce rozsahu, přechází proces znovu do stavu „data aq“. Pokud se tímto způsobem dosáhne hranice nastavení mřížky, nepřičítá se k aktuálnímu bodu, ale odečítá a celý proces se opakuje. Používají se informace o aktuálním bodu z rozsahu zrcátka, aktuálním bodu z rozsahu nastavení mřížky, počtu bodů v mřížce a informace o tom, zda se má k aktuálnímu bodu přičítat nebo odečítat. Vyhodnocování řeší struktura logických operátorů.

Posledním stavem procesu je stav „stop“, ve kterém se maže obsah fronty příkazů a odklání se zrcátko.

6.3.5 Proces zpracování dat

Proces zpracování a ukládání dat je nejjednodušší ze všech popisovaných procesů. Sestává pouze z *while* smyčky, ve které se vyčítají data z fronty naměřených dat. Data se vyčítají pouze ve chvíli, kdy jsou nějaká ve frontě obsažena. Pokud je fronta prázdná, je proces nečinný.

Data se před uložením zpracovávají tím způsobem, že se prvních deset a posledních dvacet hodnot nahradí hodnotou jedenáctou. Data se mění z toho důvodu, že obsahují „nulové“ hodnoty, které by zkreslily výsledné spektrum.

Spektra se ukládají do binárního souboru ve formátu *little-endian* v datovém typu *int32*. Vzhledem k těmto faktům je dosaženo snadného zpracování a menšího obsazeného prostoru na harddisku.

Pro kontrolu správnosti měření se data zobrazují v grafu. Pro zvýšení rychlosti zobrazování jsou data převedena na datový typ *int16*. Snížením rozlišení se sníží zátěž procesoru.

6.3.6 Proces kamery

Proces ovládání a zobrazování obrazu z kamery je téměř totožný s tím, který je popsán v kapitole 6.2.5. Rozdíl se nachází v možnosti měnit kontrast a světlost obrazu a v převrácení obrazu z kamery, optická soustava totiž obsahuje prvky, které zrcadlí obraz.

6.3.7 Inicializace a ukončení procesů

Inicializace a ukončení systému front jsou popsány v kapitole 6.2.6. Před ukončením fronty posuvů a měření je potřeba ukončit reference na používaná zařízení a posunout stolek na nulové souřadnice. Ukončení procesu ukládání dat probíhá ve chvíli, kdy se v procesu vyskytne chyba. Chyba se zaručeně vyskytne ve chvíli, kdy je zrušena fronta dat a přitom je požadováno vyčítání z této fronty. Ukončení procesu kamery je řešeno pomocí lokální proměnné.

6.4 Čtení dat z hmotnostního spektrometru.

Struktura aplikace na čtení dat z hmotnostního spektrometru se částečně liší od té, která byla doteď popisována. Program se dělí na dvě části, na *event* strukturu, obsluhující uživatelské akce a na řízený proces. Tato jednoduchá struktura je určena pro jednocelové aplikace, jako je tato. Tato aplikace slouží jako provizorní řešení, není zamýšleno pokračování jejího vývoje. Data z hmotnostního spektrometru by měla být v budoucnu ukládána do některého z otevřených formátů pro hmotnostní spektrometrii. Pokud by i přesto bylo potřeba aplikaci dále vyvíjet, není obtížné změnit strukturu na tu, která je použita u druhých dvou aplikací. Část funkcí je převzatá od minulého řešitele, aby byla zachována návaznost, není nutné vytvářet znovu funkční algoritmy.

Aplikace umožňuje načítání dat v použitém formátu (binární data, parametry v XML) a jejich zpracování. Jedná se o zobrazení mapy intenzity naměřeného vzorku ve 2D a 3D, zobrazení jednotlivých spekter a výpočet statistických hodnot.

V *event* struktuře se generují příkazy pro řízený proces, využívá se systému fronty. V řízeném procesu se pak spouští jednotlivé *SubVI*. Proces obsahuje paměť ve formě *shift registru*, která obsahuje cestu k otevíraným souborům, vypočítanou matici mapy intenzity a dvě pole časové a hmotnostní domény.

Prvním stavem je „init“, ve kterém se nastavují všechny důležité parametry na počáteční hodnoty.

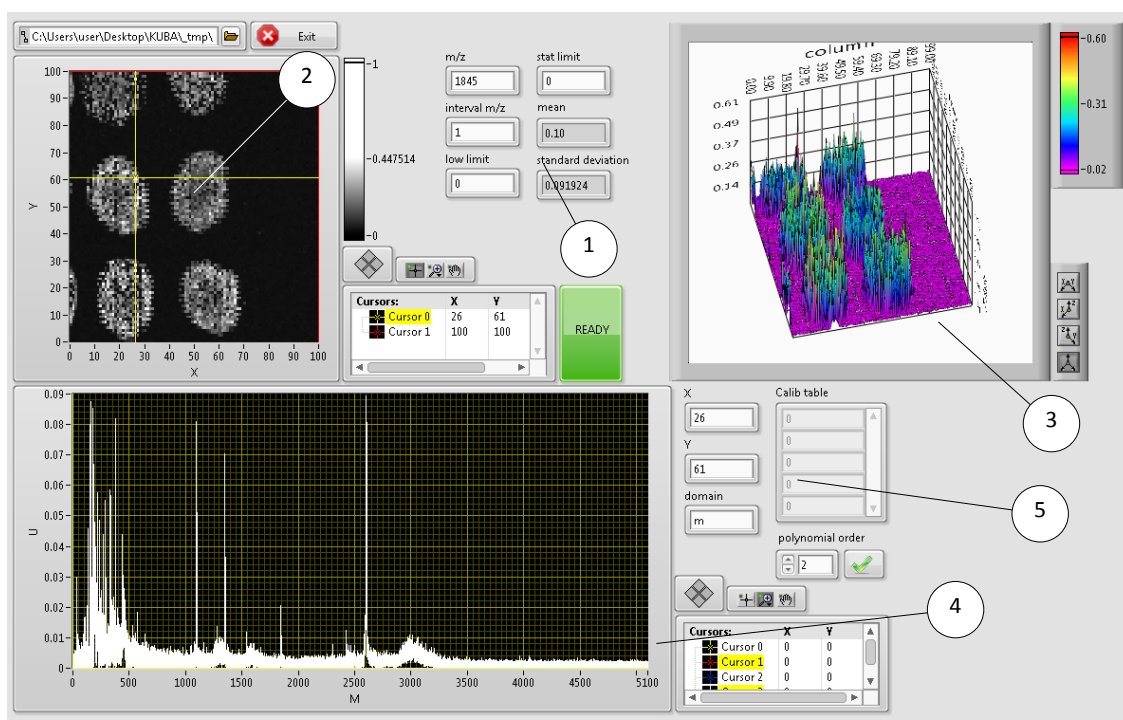
Příkaz „open“ otevírá reference na soubory a uživatel je současně vyzván, zda se má načíst i soubor kalibrace, pokud takový existuje. Pokud je načten soubor kalibrace, je kalibrace provedena hned na počátku.

Příkaz „map“ spouští sekvenci, která vytváří mapu intenzity. K tomu je potřeba vědět, jaká hmotnost je vyžadována, rozptyl kolem této hmotnosti a limit minimální hodnoty. Data se postupně vyčítají a zpracovávají. Výsledná mapa intenzity se zobrazuje současně ve 2D i 3D.

Příkaz „spectrum“ vyčítá ze souboru jednotlivé spektrum, které se určí buď pomocí kurzoru z grafu intenzity, nebo vepsáním souřadnic.

Příkaz „calib“ značí kalibraci, která se provádí následovně. Obsluha vepíše známé hodnoty peptidů a poté je označí pomocí kurzorů v grafu spektra. Je možné zvolit řád polynomu, kterým se prokládají označené body. Kalibrační koeficienty jsou následně aplikovány na hmotnostní doménu. V případě, že obsluha použije kalibraci, uloží se nastavení této kalibrace do souboru, který je možné později načíst.

Stavy „ready“ a „exit“ jsou zahrnuty, ale nejsou využity. Mohou být použity v případě, že by byla potřebná změna struktury.



Obrázek 11 - Uživatelské rozhraní aplikace pro čtení dat:

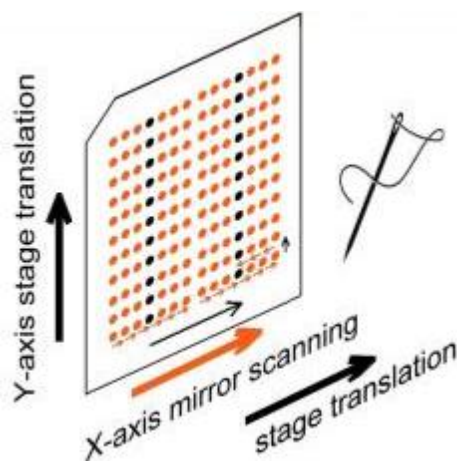
1) nastavení map intenzity, 2) 2D mapa, 3) 3D mapa, 4) spektrum, 5) kalibrace

7 Ověření funkčnosti

Funkčnost všech tří aplikací byla ověřena nanesením zkušební vzorku, jeho měřením a analýzou. Zkušebním vzorkem je kalibrační směs „calmix“, která obsahuje čtyři peptidy o známých hmotnostech.

Pro nanášení byl zvolen režim „dropping“, z toho důvodu, že se vytvoří kapka určitého tvaru, který pak bude možné pozorovat i při měření. V průběhu nanášení nebyly pozorovány chyby, které se objevovaly dříve, jako jsou zastavení posuvu nebo vynechání bodu. Režim „piezo“ byl ověřen se vzorky určenými k analýze v komerčních spektrometrech. Nebyly pozorovány žádné chyby při nanášení.

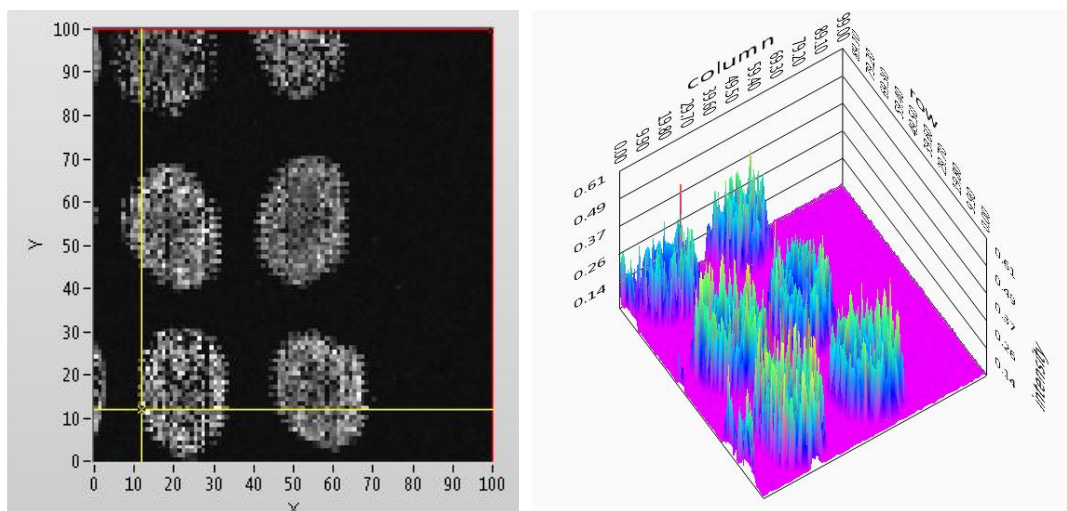
Měření vzorku probíhalo formou „stitched imaging“. Rozsah byl nastaven na 100 x 100 bodů po 0,1 mm. Tím vznikla plocha 1 cm² veliká o celkovém počtu 10 000 bodů. Jako režim chování zrcátka byla zvolena kompenzace pohybu v ose x. Rozsah byl nastaven na 1 mm, tím pádem vykonaly posuvy desetkrát méně kroků. Laser byl nastaven na frekvenci 2 kHz. Každý bod byl průměrem sta měření o 100 000 vzorcích.



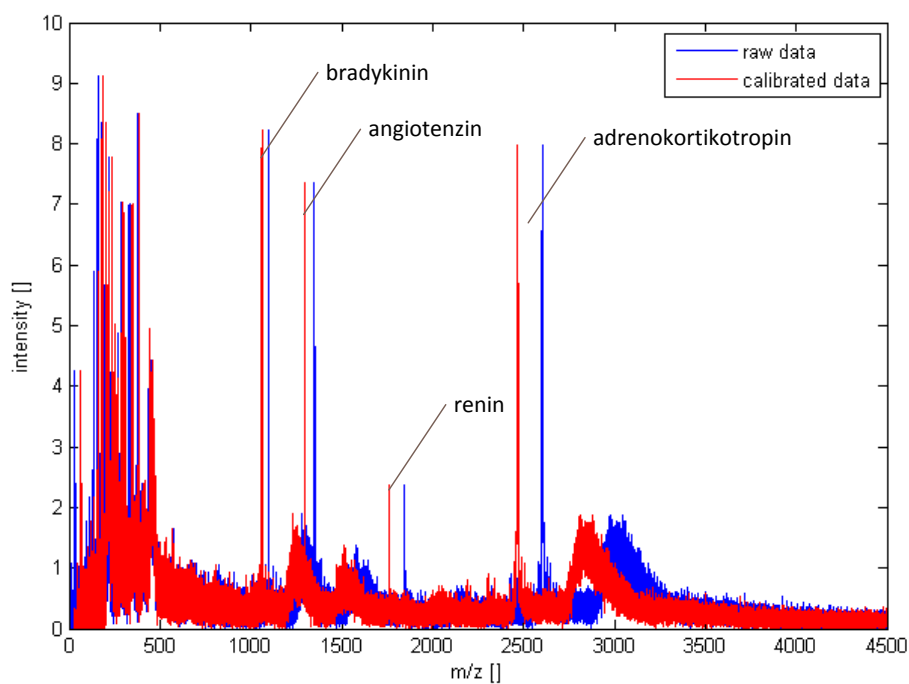
Obrázek 12 - Schéma režimu "stitched imaging" [19]

Měření proběhlo úspěšně a navíc velmi rychle. V porovnání s komerčním spektrometrem, proběhlo téměř desetkrát rychleji (Tabulka 1), bez výrazné ztráty kvality naměřených spekter. To se potvrdilo při analýze v aplikaci pro čtení dat.

Z naměřených dat byly vytvořeny mapy intenzity ve 2D a 3D (viz Obrázek 11). Z uvedeného spektra lze vyčíst obsažené látky: bradykinin, angiotenzin, renin a adrenokortikotropin. Na tomto spektru byla ověřena i funkční kalibrace (Obrázek 14).



Obrázek 13 - Srovnání 2D a 3D mapy intenzity



Obrázek 14 - Naměřená spektra: surová a kalibrovaná data

| metoda | f (kHz) | t_t (ms) | $t_a + t_p$ (ms) | t_{spot} (ms) | t_{total} (min) |
|-------------------------|---------|------------|------------------|-----------------|-------------------|
| Autoflex Speed | 1 | | | 750 | 125 |
| Drobeček (pouze posuvy) | 1 | 95 | 100 + 10 | 206 | 34.5 |
| Drobeček (zrcátko) | 1 | 220 | 100 + 10 | 130 | 21.9 |
| Drobeček (zrcátko) | 2 | 220 | 51 + 10 | 83.5 | 13.1 |

Tabulka 1 - Srovnání rychlostí:

t_t – posuvy, t_a – měření dat, t_p – vyčítání dat aj,
 t_{spot} – čas měření jednoho bodu, t_{total} – celkový čas měření [19]

8 Závěr

Práce se zabývala problematikou vývoje řídicích aplikací pro nanášecí zařízení a hmotnostní spektrometr. Na základě rozboru charakteristiky zařízení, stavu předchozích aplikací a konzultací se zadavatelem, bylo upuštěno od pokračování vývoje původních aplikací a rozhodnuto o vytvoření zcela nových aplikací, které by splňovaly požadavky na novou funkcionalitu. Z toho důvodu také řešení neobsahuje všechny prvky zadání práce.

Pro vývoj aplikací byla použita nová struktura, které splňuje požadavky na modularitu. Aplikace nyní obsahují všechny požadované režimy a současně jsou lehce rozšiřitelné. Vyvíjené aplikace obsahují části, které byly v rámci spolupráce vytvořeny jinými řešiteli v rámci jejich absolventských prací.

Obě vyvíjené aplikace jsou nyní používány k ovládání daných zařízení v laboratořích zadavatele. V rámci dalšího vývoje se počítá s rozšířením o další režimy a funkce. U aplikace nanášecího zařízení se jedná o režim měření fluorescence nanesených vzorků a načítání parametrů nanášení z externího souboru. U aplikace hmotnostního spektrometru by měl další vývoj směřovat rozšířením o kalibraci před samotným měřením či změnu používaného formátu na jeden z otevřených formátů určených pro hmotnostní spektrometrii. Naměřená data by pak mohla být analyzována v sofistikovanějším programu, než je vyvinutá aplikace pro čtení naměřených dat. Tato aplikace byla z tohoto důvodu vyvinuta pouze jako dočasné řešení.

Části práce byly zahrnuty v příspěvku zadavatele do letošního ročníku konference AMSM ve Vancouveru v Kanadě. Příspěvek „High throughput MS imaging using a fast scanning mirror“ prezentuje výsledky práce na režimu měření „stitched imaging“ hmotnostního spektrometru.

ZDROJE

1. **Grayson, M.** *Measuring mass from positive rays to proteins*. Philadelphia : Chemical Heritage Press, 2002. Dostupné z: <http://www.clinchem.org/content/49/2/342.2.full>.
2. **Van Bramer, S. E.** *An Introduction to Mass Spectrometry*. Chester : Widener University, 1997. Dostupné z: <http://science.widener.edu/~svanbram/>.
3. **National High Magnetic Field Laboratory.** *MALDI*. [Online] [Citace: 22. květen 2012.] Dostupné z: http://www.magnet.fsu.edu/education/tutorials/tools/ionization_maldi.html.
4. **Preisler, J.** Jan Preisler. *Studijní text MALDI TOF MS*. [Online] [Citace: 22. květen 2012.] Dostupné z: <http://bart.chemi.muni.cz/courses/Lab%20Cv%20MALDI%202008%20CZ.doc>.
5. **Moskovets, E.** High-Throughput Axial MALDI-TOF MS Using a 2-kHz Repetition Rate Laser. *Analytical Chemistry*. 2006, Sv. 78, 912 - 919.
6. **Ertl, L.** *Software pro řízení hmotnostního spektrometru a nanášecího zařízení*. Brno : Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2011.
7. **Standa.** *8MT175 Datasheet*. [Online] [Citace: 22. květen 2012.] Dostupné z: <http://www.standa.lt/files/8MT175.pdf>.
8. —. *8SMC1-USBhf Manual*. [Online] [Citace: 2012. květen 22.] Dostupné z: <http://www.standa.lt/files/usb/8SMC1-USBhf%20User%20Manual.pdf>.
9. **355nm.** *DTL-374QT Laser System Specifications*. [Online] [Citace: 22. květen 2012.] Dostupné z: http://www.355nm.com/LaserInnovations_355nm_Laser-Export_DTL-374QT.htm.
10. **Cambridge Technology.** *6810P Specifications*. [Online] [Citace: 22. květen 2012.] Dostupné z: <http://www.camtech.com/products/6800/6810p.html>.
11. **National Instruments.** *myDAQ User Guide and Specifications*. [Online] [Citace: 22. květen 2012.] Dostupné z: www.ni.com/pdf/manuals/373060e.pdf.
12. **National Aperture.** *MM-3-M-F Specifications*. [Online] [Citace: 22. květen 2012.] Dostupné z: <http://www.naimotion.com/mm3mf.htm>.
13. —. *MM-4-M-Ex Specifications*. [Online] [Citace: 22. květen 2012.] Dostupné z: <http://www.naimotion.com/mm4mex.htm>.
14. —. *MC-4SA Specification*. [Online] [Citace: 22. květen 2012.] Dostupné z: <http://www.naimotion.com/mc4sa.htm>.
15. **National Instruments.** *PCI-7344 Specifications*. [Online] [Citace: 22. květen 2012.] Dostupné z: <http://www.ni.com/pdf/products/us/4mo636-637.pdf>.
16. **AverMedia.** *DVD EZMaker 7*. [Online] [Citace: 22. květen 2012.] Dostupné z: <http://www.avermedia.eu/avertv/cz/product/ProductDetail.aspx?Id=444>.
17. **Acqiris.** *AP 200 Programmers Guide*. [Online] [Citace: 22. květen 2012.] Dostupné z: <http://pen.phys.virginia.edu/manuals/acqiris/UserManualAveragers.pdf>.
18. **National Instruments.** *LabVIEW User Manual*. [Online] [Citace: 22. květen 2012.] Dostupné z: www.ni.com/pdf/manuals/320999e.pdf.
19. **Bednařík, A.** *High throughput MS imaging using a fast scanning mirror*. Brno : Masaryk University, 2012.