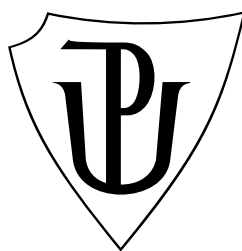


UNIVERZITA PALACKÉHO V OLMOUCI

PŘÍRODOVĚDECKÁ FAKULTA

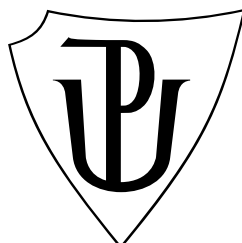


DIPLOMOVÁ PRÁCE

2009

Bc. Petr Hamal

UNIVERZITA PALACKÉHO V OLMOUCI
PŘÍRODOVĚDECKÁ FAKULTA
Katedra experimentální fyziky



Automatizace měření otěru kyčelního
implantátu pomocí optické 3D skenovací
profilometrie

diplomová práce

Autor: Bc. Petr Hamal

Program studia: N1701 - Fyzika

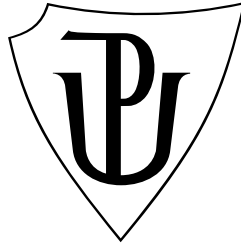
Obor studia: Aplikovaná Fyzika

Forma studia: Prezenční

Vedoucí diplomové práce: Mgr. Dušan Mandát

Termín odevzdání: 29. května 2009

PALACKÝ UNIVERSITY, OLOMOUC
FACULTY OF NATURE
Department of Experimental Physics



Automation of Total Hip Arthroplasty Wear
Measurement Using Optical 3D Scanning
Profilometry

Diploma Thesis

Author: Bc. Petr Hamal

Program of study: N1701 - Physics

Specialization of study: Application of Physics

Form of study: Attendance

Supervisor: Mgr. Dusan Mandat

Date of submit: 2009, 29 May

ABSTRACT

This thesis deals with a solution of automation of total hip arthroplasty wear measurement using optical 3D scanning profilometry. It shows one way to approach this questions. In the measurement process automation solutions, it using the development environment of LabVIEW, which is now commonly used for industrial automation.

BIBLIOGRAFICKÁ IDENTIFIKACE

Autor: Bc. Petr Hamal

Název práce: Automatizace měření otěru kyčelního implantátu pomocí optické 3D skenovací profilometrie

Typ práce: Diplomová práce

Pracoviště: Katedra experimentální fyziky

Vedoucí diplomové práce: Mgr. Dušan Mandát

Rok obhajoby: 2009

Abstrakt: Tato práce se zabývá řešením automatizace měření otěru kyčelního implantátu pomocí optické 3D skenovací profilometrie. Ukazuje jednu z možností, jak k této problematice přistupovat. Při řešení automatizace měřicího procesu využívá programového prostředí LabVIEW, které je v dnešní době velmi rozšířené právě pro vývoj automatizačních systémů.

Klíčová slova: automatizace měření, optická 3D skenovací profilometrie, LabVIEW, MATLAB, kyčelní implantát

Počet stran: 80

Počet příloh: 1 - CDROM

Jazyk: Český

BIBLIOGRAFIC IDENTIFICATION

Author: Bc. Petr Hamal

Thesis name: Automation of Total Hip Arthroplasty Wear Measurement Using Optical 3D Scanning Profilometry

Type of thesis: Diploma Thesis

Department: Department of Experimental Physics

Supervisor: Mgr. Dusan Mandat

Year of presentation: 2009

Abstract: This thesis deals with a solution of automation of total hip arthroplasty wear measurement using optical 3D scanning profilometry. It shows one way to approach this questions. In the measurement process automation solutions, it using the development environment of LabVIEW, which is now commonly used for industrial automation.

Keywords: Automation Measurement, Optical 3D Scanning Profilometry, LabVIEW, MATLAB, Total Hip Arthroplasty

Number of pages: 80

Number of appendices: 1 - CDROM

Language: Czech

Obsah

1	Úvod	15
2	Optická 3D skenovací profilometrie	16
2.1	Teorie	16
2.1.1	Geometrická analýza metody	16
2.1.2	Citlivost měřicího systému	18
2.1.3	Nejistoty měření	19
2.1.4	Analýza stopy	19
2.1.5	Průběh kalibrace	20
3	Labview, Matlab	21
3.1	Obecný popis	21
3.1.1	LabVIEW	21
3.1.2	MATLAB	22
3.2	Vzájemná komunikace	22
4	Použitý hardware	23
4.1	Digitální kamera	23
4.1.1	Kamera Lu120M firmy Lumenera Corporation	24
4.2	Krokový motor	27
4.2.1	Rotační krokový motor Newport PR50CC	29
4.2.2	Lineární krokový motor Newport ILS150CCL	31
4.2.3	Řídící jednotka Newport SMC100CC	33
4.3	Projektor stopy	35
5	Použitý software	37
5.1	Digitální kamera	37
5.1.1	National Instruments LabVIEW Plug-in	38
5.2	Krokový motor	42
5.3	Synchronizace	48
5.4	Uživatelské rozhraní	50
5.4.1	Nastavení kamery	53
5.4.2	Kalibrace sestavy	55
5.4.3	Vlastní měření	57

6	Experiment	59
6.1	Kyčelní implantát	61
6.1.1	Nová jamka	61
6.1.2	Vzorek použité jamky	63
7	Závěr	66
8	Přílohy	68
8.1	XML soubor s nastavením parametrů digitální kamery	68
8.2	MATLAB skript	72
8.3	CD-ROM	77

Seznam obrázků

1	Geometrické uspořádání optické 3D skenovací profilometrie. . .	17
2	(a) rozložení intenzit kolem proužku, (b) aproximace proužku . .	20
3	funkce „MATLAB Script Node“ - způsob vložení MATLAB skriptu do LabVIEW.	22
4	Schéma zpracování signálu ze CCD senzoru, včetně obslužné elektroniky [19].	24
5	Schéma zpracování signálu z CMOS senzoru, včetně obslužné elektroniky [19].	25
6	CMOS kamera - model Lu120M firmy Lumenera Corporation [17].	26
7	Obecné schéma krokového motoru s aktivním rotorem [8]. . . .	28
8	Obecné schéma krokového motoru s pasivním rotorem [8]. . . .	29
9	Obecné schéma hybridního krokového motoru [8].	30
10	Krokový motor - model PR50CC firmy Newport Corporation [11].	31
11	Krokový motor - model ILS150CCL firmy Newport Corporation [13].	31
12	Řídící jednotka krokového motoru - model SMC100CC firmy Newport Corporation [15].	33
13	V-A charakteristika polovodičové diody [23].	36
14	Hlavní okno programu „Lucam Software“.	37
15	LuCamOpen.vi - slouží k otevření portu pro komunikaci s kamerou.	38
16	LuCamClose.vi - slouží k uzavření portu pro komunikaci s kamerou.	38
17	Funkce „IMAQ create“ vytvoří objekt „obrázek“.	39
18	Funkce „IMAQ write bmp file“ uloží obrázek.	39
19	Funkce „IMAQ Dispose“ zruší objekt „obrázek“.	39
20	LuGetFormat.vi - určen k zjištění hodnot a vlastností digitální kamery.	40
21	LuEnableFastFrames.vi - umožní získání snímku z kamery. . . .	40
22	LuDisableFastFrames.vi - ukončí mód získání snímku z kamery.	40
23	LuFastSnap.vi - získá snímek z digitální kamery.	40
24	LuStartStream.vi - spustí režim videa.	41
25	LuStopStream.vi - ukončí režim videa.	41

26	LuTakeVideo.vi - zobrazí živé vysílání z digitální kamery.	41
27	LuPropRange.vi - vrací minimální, maximální a standardní hodnotu požadované vlastnosti.	42
28	LuSetProperty.vi - slouží k nastavení hodnoty zvolené vlastnosti.	42
29	Stavový diagram řídicí jednotky SMC100CC.	43
30	Modul „Enter Leave disable state.vi“ - přepne řídicí jednotku do/z stavu DISABLE.	44
31	Obecný předpis syntaxe pro řídicí jednotku SMC100CC.	44
32	Modul „port init.vi“ - otevře port pro komunikaci s řídicí jednotkou krokového motoru.	45
33	Modul „close.vi“ - uzavře port pro komunikaci s řídicí jednotkou krokového motoru.	45
34	Modul „Enter Leave configuration state.vi“.	46
35	Modul „Set HOME type.vi“ - nastaví počáteční pozici krokového motoru.	46
36	Modul „Home.vi“ - přesune krokový motor do počáteční pozice.	46
37	Modul „Enter Leave disable state.vi“.	47
38	Modul „move absolute.vi - posune krokovým motorem na zadanou pozici“.	47
39	Modul „move relative.vi“ - posune krokovým motorem o zadaný krok.	47
40	Modul „get absolute position.vi“ - vrátí hodnotu aktuální absolutní pozice.	47
41	Modul „get positive position limit.vi“ - vrátí maximální dosažitelnou pozici v kladném směru.	48
42	Modul „get negative position limit.vi“ - vrátí maximální dosažitelnou pozici v záporném směru.	48
43	Funkce „Flat Sequence Structure“ pro sekvenční vykonávání příkazů.	49
44	Modul „tell time for relative motion.vi“ - vrátí čas potřebný k vykonání pohybu krokového motoru.	49
45	Hlavní panel programu	51
46	Inicializační část programu - programový kód. Slouží k otevření komunikačních portů digitální kamery a krokových motorů.	51
47	Hlavní cyklus programu - programový kód.	52

48	Nastavení kamery - přední panel. Tato část slouží k nastavení parametrů digitální kamery.	53
49	Nastavení kamery - programový kód, který obsluhuje nastavení jednotlivých parametrů digitální kamery a zobrazuje živý náhled z kamery.	54
50	Kalibrace sestavy - přední panel. Slouží k nastavení parametrů kalibrace a následného získání kalibračních dat.	55
51	Kalibrace sestavy - blokový diagram. Programový kód, který obsluhuje a řídí proces kalibrace.	56
52	Vlastní měření - přední panel. Slouží k zadání informací pro vlastní měření vzorku.	57
53	Vlastní měření - blokový diagram. Programový kód, který obsluhuje a řídí proces měření vzorku.	58
54	Experimentální zapojení - digitální kamera, světelný zdroj, měřený vzorek uchycený v držáku rotačního krokového motoru (zleva doprava). Schéma experimentálního zapojení (ve výřezu).	59
55	Jamka kyčelního implantátu - nová.	60
56	Jamka kyčelního implantátu - opotřebená.	60
57	Snímek z měření - snímek světelné stopy kopírující povrch nové jamky kyčelního implantátu.	61
58	3D profil měřené nové jamky kyčelního implantátu [pix].	62
59	Vrstevnicový graf [pix] (Contour Plot) - nová jamka kyčelního implantátu.	62
60	Snímek z měření - snímek světelné stopy kopírující povrch opotřebené jamky kyčelního implantátu.	63
61	3D profil měřené opotřebené jamky kyčelního implantátu [pix].	64
62	Vrstevnicový graf [pix] (Contour Plot) - opotřebená jamka kyčelního implantátu.	64
63	Image Init.	73
64	Image UnInit.	73
65	Motor Port Init.	73
66	Motor Port UnInit.	74
67	Camera settings.	74
68	Camera settings 1 init.	74
69	Camera settings 2 init.	75
70	Camera settings 1 and 2.	75

71	Read Lucam Snapshot Settings.	75
72	Write Lucam Snapshot Settings.	75
73	Modul Posuv.vi	76
74	Maximální interval nastavení pozic.	76
75	Nastavení počáteční pozice pro kalibraci.	76
76	Modul pro vlastní měření předmětu.	77
77	„Matlab image script“ pro zpracování získaných obrázků.	77

Seznam tabulek

1	Shrnutí technologií CCD a CMOS [19].	25
2	Technické specifikace modelu Lu120M	26
3	Detaily modelu PR50CC	30
4	Technické specifikace modelu PR50CC	30
5	Detaily modelu ILS150CCL	32
6	Technické specifikace modelu ILS150CCL	32
7	Technické specifikace modelu SMC100CC	34

Prohlášení

Prohlašuji, že jsem práci na téma **Automatizace měření otěru kyčelního implantátu pomocí optické 3D skenovací profilometrie** vypracoval samostatně za použití uvedené literatury.

Podpis autora: _____

Poděkování

Rád bych na tomto místě poděkoval všem, kteří mi pomohli slovem i skutkem při práci na tomto textu. Poděkování si zaslouží vedoucí mé práce Mgr. Dušan Mandát za cenné rady a konzultace při tvorbě diplomové práce, za poskytnutí částí odborné literatury a umožnění práce v laboratoři.

Poděkování též zaslouží pacienti jejichž kyčelní implantáty byly použity za účelem studia poškození a opotřebení implantátů. Bez nich by tato práce nikdy nevznikla.

1 Úvod

Společná laboratoř Optiky Univerzity Palackého v Olomouci a Fyzikálního Ústavu Akademie Věd ČR (dále jen SLO) byla zapojena do projektu monitorujícího stav kyčelních implantátů pacientů ortopedického oddělení Fakultní nemocnice v Olomouci. Životnost kyčelních implantátů je závislá na mnoha faktorech. Mezi faktory ovlivňující životnost implantátů patří věk, hmotnost, pohlaví pacienta, tělesné aktivity, apod. U pacienta je v průběhu užívání kyčelní náhrady pravidelně sledován její stav (zpravidla pomocí rentgenových technik) a v případě velkého opotřebení či jiného poškození je kyčelní náhrada vyměněna za novou.

Opotřebená jamka kyčelního implantátu je následně podrobena řadě různých testů. Výsledky testů pak slouží k nalezení závislostí materiálu, tvaru, usazení náhrady v těle pacienta, atd. na výše uvedených faktorech. Testům je v současné době podroben velký počet vzorků (u kyčelních endoprotéz více jak 200).

Mezi důležité parametry, které se z testů zjišťují, patří tvar a velikost opotřebené jamky kyčelního implantátu. Tyto testy se provádí na pracovišti SLO, kde se k měření těchto parametrů využívá metody optické 3D skenovací profilometrie. Princip měření je založen na přepočtu změny posunu Δu polohy proužku vlivem změny polohy Δr zkoumaného povrchu vzhledem k uživatelsky definované referenční rovině určené kalibrací [3].

Základem měření je získat sadu snímků kyčelní jamky s projekční stopou, která kopíruje zkoumaný povrch jamky. Každý snímek představuje polohu jamky po otočení o konstantní, předem daný, úhel. Z měření na jednom vzorku jamky se tak získá sada snímků, která je následně podrobena dalšímu zpracování a vyhodnocení výsledků. Pokud se má proměřit malý počet vzorků, je možné využít software dodávaného výrobcem k digitální kameře a krokovým motorům. Měření lze provést manuálně tak, že se pomocí jednoho software vyšle krokovému motoru požadavek o pootočení vzorku a pomocí druhého software se digitální kamerou vyfotí obraz světelné stopy na vzorku a následně se uloží. Tento postup se opakuje po požadovaný počet kroků. Poté jsou získané snímky zpracovány ve specializovaném software, který zároveň vyhodnotí výsledky.

Tento postup však nelze uplatňovat na velký soubor vzorků, už jen z hlediska časové a manuální náročnosti. Proto na řadu přichází automatizace měření, která má za úkol co nejvíce zjednodušit práci a ušetřit čas labora-

torního pracovníka při daném měření.

Cílem diplomové práce je vytvoření programu, který bude obsluhovat a řídit hardware použitý v experimentu a zároveň na konci měření poskytne data v digitální formě připravené pro další vyhodnocení. Celá práce laboratorního pracovníka by měla spočívat jen ve výměně vzorků a nastavení potřebných parametrů v obslužném programu. Průběh měření by pak měl být plně v režii obslužného programu, který veškeré kroky spojené s pohybem vzorku, snímáním obrazu a prvotní analýzou získaných snímků provede automaticky.

2 Optická 3D skenovací profilometrie

2.1 Teorie

Optická 3D skenovací profilometrie je bezkontaktní metoda, pomocí které je možné získat informace o topografii povrchu zkoumaného předmětu. Díky bezkontaktnímu měření je tato metoda vhodná k využití např. i v medicínské praxi.

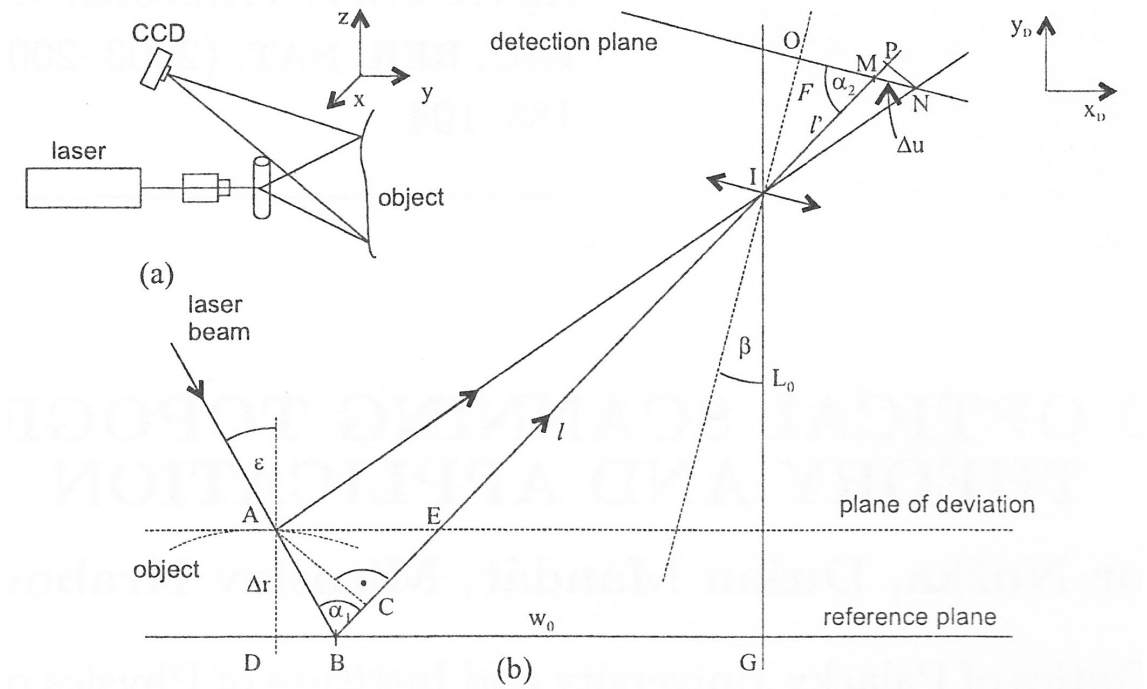
Neboť je použitý kyčelní implantát podroben řadě různých testů, je vhodné využívat měřicích metod, které nepoškodí zkoumaný vzorek. A tuto vlastnost má např. optická 3D skenovací profilometrie. Zároveň je tato metoda, díky využití optiky, dostatečně přesná. I proto byla tato metoda vybrána pro měření otěru kyčelního implantátu.

Princip měření je založen na přepočtu změny posunu Δu polohy proužku vlivem změny polohy Δr zkoumaného povrchu vzhledem k uživatelsky definované referenční rovině určené kalibrací [3].

2.1.1 Geometrická analýza metody

Geometrické uspořádání optické 3D skenovací profilometrie je zobrazeno na Obr. 1a. Světelný svazek ze zdroje (laser) prochází optickou soustavou s válcovou čočkou. Tato optická soustava upraví světelný paprsek na úzký světelný proužek zaostřený do tangenciální roviny blízké rovině referenční, ke které je vztažen zkoumaný předmět.

Upravený světelný paprsek, úzký světelný proužek, dopadá na povrch předmětu. Odražený a deformovaný proužek od povrchu předmětu je zachycen pomocí digitální kamery. Ze znalosti tvaru proužku v detekční rovině (rovina senzoru digitální kamery) lze určit tvar povrchu měřeného předmětu. K odvození



Obrázek 1: Geometrické uspořádání optické 3D skenovací profilometrie.

algoritmu pro výpočet tvaru zkoumaného předmětu pomocí zaznamenaných proužků je použito geometrické schéma na Obr. 1b.

Bod B je bodem průniku projekční stopy a referenční roviny. Tento bod je v rovině detekce pozorován v bodě M . Pozorovací systém je umístěn od referenční roviny ve vzdálenosti $L_0 = \overline{IG}$ a od bodu B ve vzdálenosti $w_0 = \overline{BG}$. Úhel β je úhel, který svírá optická osa digitální kamery s normálou referenční roviny. Při měření předmětu dopadne projekční stopa do bodu A . Tento bod je zobrazen do bodu N ležícího v detekční rovině. Ze znalosti vzdálenosti $\Delta u = \overline{NM}$ a podobnosti trojúhelníků ΔAIC a ΔNIP lze určit hledanou vzdálenost $\Delta r = \overline{AD}$

$$\frac{l - \Delta r \cos \alpha_1 / \cos \epsilon}{\Delta r \sin \alpha_1 / \cos \epsilon} = \frac{l' + \Delta u \cos \alpha_2}{\Delta u \sin \alpha_2}. \quad (1)$$

Po zavedení parametrů $a(z)$ a $b(z)$, kde

$$a(z) = \frac{\sin \alpha_1 (\cot \alpha_1 + \cot \alpha_2)}{l \cot \epsilon} [1/\text{mm}] \quad (2)$$

a

$$b(z) = \frac{l' \sin \alpha_1}{l \sin \epsilon \sin \alpha_2} [-], \quad (3)$$

dostaneme zjednodušený tvar Rovnice 1

$$\frac{1}{\Delta r} = a(z) + \frac{b(z)}{\Delta u}. \quad (4)$$

Parametry $a(z)$ a $b(z)$ jsou odvozeny od parametrů $\epsilon, \alpha_1, \alpha_2, l$ a l' , proto lze jejich hodnoty získat z kalibrace. Pro účely měření je vhodné brát hodnoty Δu v jednotkách pixelů a tudíž parametr b v [pix/mm]. Rovnice 4 může být přepsána do tvaru

$$\Delta r = \frac{\Delta u}{b(z) + a(z)\Delta u}. \quad (5)$$

S využitím rozvoje do McLaurenovi řady lze psát

$$\Delta r \approx \frac{1}{b}\Delta u - \frac{a}{b}(\Delta u)^2 = \frac{\Delta u}{b}(1 - \Delta u), \quad (6)$$

nebo

$$\Delta r = c_1\Delta u - c_2(\Delta u)^2, \quad (7)$$

kde

$$c_1 = \frac{1}{b}, \quad c_2 = \frac{a}{b}. \quad (8)$$

2.1.2 Citlivost měřicího systému

S využitím Rovnice 5 lze určit citlivost měřicího systému, tj. minimální změnu profilu c k posunu zaznamenaného proužku pixelem o velikosti p_s . Substitucí $\Delta r = c$ a $\Delta u = p_s$ a s využitím Rovnice 2 a Rovnice 3 dostáváme

$$c = \frac{l \cos \epsilon \sin \alpha p_s}{l' \sin \alpha_1 + \sin(\alpha_1 + \alpha_2)p_s}, \quad (9)$$

po zjednodušení

$$c = \frac{1}{a + b}. \quad (10)$$

Citlivost měřicího systému c vyjádřena pomocí měřitelných veličin L_0, w_0, F, ϵ a β (viz Obr. 1b):

$$c = \frac{p_s \sqrt{L_0^2 + w_0^2} \cos \epsilon \sin(\beta + \Theta_2)}{F \frac{\sin(\epsilon + \Theta_1)}{\sin(\beta + \Theta_2)} - p_s \cos \epsilon}, \quad (11)$$

kde $\Theta_1 = \arctan \frac{w_0}{L_0}$ a $\Theta_2 = \arctan \frac{L_0}{w_0}$.

2.1.3 Nejistoty měření

Nejistotu měření odvodíme z Rovnice 7. Dostaneme diferenciál

$$\delta(\Delta r) = |c_1 - 2c_2\Delta u|\delta(\Delta u) + |\Delta u|\delta c_1 + (\Delta u)^2\delta c_2, \quad (12)$$

kde je $\delta(\Delta u)$ v jednotkách pixelů. Odhad variance $s_{\Delta u}^2$ hodnoty Δu je

$$s_{\Delta u}^2 = \frac{(x_{\text{Di}} - u)^T H^{-1} (x_{\text{Di}} - u)}{n' - 1}, \quad (13)$$

kde H je diagonální matice a $H_{ii} = h(x_i) / \sum h(x_i)$ a n' je počet nenulových hodnot v jednom řádku. Potom je odhad střední kvadratické chyby $\delta(\Delta u)$

$$\delta(\Delta u) = \sqrt{s_{\Delta u}^2}. \quad (14)$$

Parametry c_1 a c_2 jsou vypočteny regresní analýzou z dat získaných z počáteční kalibrace měřicí soustavy. Z n měření veličin Δu a Δr dostaneme následující odhad kovariantní matice

$$\mathbf{S}_c = \frac{1}{n-2} (\mathbf{U}^T \mathbf{U})^{-1} \sum_{i=1}^n (\Delta r_i - \overline{\Delta r_i})^2, \quad (15)$$

kde \mathbf{S}_c je matice typu 2x2, \mathbf{U} je matice typu nx2 hodnot Δu a $(\Delta u)^2$. Potom jsou odhady středních kvadratických chyb parametrů c_1 a c_2

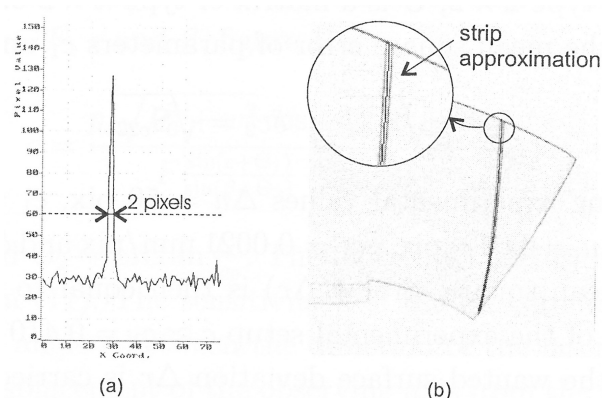
$$\delta c_1 = \sqrt{\mathbf{S}_{c11}}, \quad \delta c_2 = \sqrt{\mathbf{S}_{c22}}. \quad (16)$$

2.1.4 Analýza stopy

Po získání záznamu obrazu povrchu zkoumaného předmětu osvětleného projekční stopou je provedena analýza projekční stopy (proužku). Cílem analýzy je oddělit proužek od samotného obrazu a převést jej na množinu bodů pro pozdější zpracování. Tato analýza se skládá ze tří základních kroků: prahování obrazu, filtrace obrazu, výpočet středu proužku.

Prahování obrazu se provádí k odfiltrování oblastí mimo proužek (pozadí). Toto filtrování je možné použít, neboť intenzita proužku je výrazně vyšší než intenzita pozadí, viz Obr. 2a. Výběr vhodné hodnoty prahování je závislý na faktorech jako je intenzita světla na pozadí, expoziční doba digitální kamery a druh zkoumaného povrchu.

Po prahování obrazu může stále v obraze zůstat několik výraznějších zrn, která mohou zanést chyby do konečného výpočtu. Proto je třeba v této fázi



Obrázek 2: (a) rozložení intenzit kolem proužku, (b) aproximace proužku

tato zrna nalézt a odfiltrovat. Analýza nežádoucích zrn probíhá tak, že se prohledává blízké okolí nenulových pixelů (blízké okolí proužku). Jestliže je počet sousedících nenulových pixelů menší než 2, je tento pixel označen jako zrno a následně smazán. V opačném případě je pixel označen za součást proužku. Toto filtrování má i ten efekt, že se hrany proužku stanou hladší, výraznější.

Posledním krokem analýzy je určení středu proužku, tj. určení bodů proužku s nejvyšší intenzitou Obr. 2b. Proužek je následně nahrazen souborem hodnot centrálních bodů. K výpočtu střední hodnoty každého řádku proužku se využívá váhového průměrování. Pro střední hodnotu $\bar{x}_D(y_D)$ proužku na řádku y_D platí

$$\bar{x}_D(y_D) = \frac{\sum h(x_D, y_D)x_D}{\sum h(x_D, y_D)}. \quad (17)$$

kde $h(x_D, y_D)$ je intenzita pixelu o souřadnicích (x_D, y_D) . Vektor bodů $(\bar{x}_D(y_D), y_D)$ získaný tímto výpočtem představuje výsledný proužek.

Pokud je experimentální zapojení kalibrováno, je vhodné pro definici referenčních rovin použít rovinu obdélníkového tvaru. V tomto případě můžeme nahradit proužek polynomem prvního řádu, $y_D = \lambda_0 + \lambda_1 x_D$, kde λ_0 a λ_1 jsou parametry získané z regresní analýzy. Aproximace polynomem prvního řádu nám zanechává do výpočtu další filtrování dat a umožňuje nám přesnější zpracování dat [2].

2.1.5 Průběh kalibrace

Kalibrace experimentálního sestavení spočívá v získání sady snímků uživatelsky definovaných referenčních rovin. Tato sada snímků je následně podrobena

analýze pozic stopy na vzorku, jejíž výsledkem je výpočet kalibračních konstant a a b z Rovnice 2 resp. z Rovnice 3. Tyto kalibrační konstanty jsou později použity při zpracování výsledků vlastního měření předmětu a slouží k převodu jednotek [pix] na jednotky [mm] (Rovnice 4).

3 Labview, Matlab

Při tvorbě software pro automatizaci měření otěru kyčelního implantátu pomocí optické 3D skenovací profilometrie bylo využito programovacího prostředí LabVIEW. Části kódu, které mají za úkol zpracování dat získaných z obrázků, jsou napsány pro Matlab. Z prostředí LabVIEW se následně spustí kernel Matlabu a provede se příslušný výpočet.

3.1 Obecný popis

3.1.1 LabVIEW

LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) je platforma a vývojové prostředí pro vizuální programovací jazyk „G“. LabVIEW je vyvíjeno firmou National Instruments. Jeho využití je v dnešní době velmi široké. Běžně se používá pro získávání dat a jejich zpracování, pro vývoj měřicích, řídicích a automatizačních systémů [4].

Programovací jazyk v prostředí LabVIEW je označován jako jazyk „G“. Na rozdíl od „klasického“ programování pomocí textových jazyků, LabVIEW pracuje s grafickými prvky (ikonami, moduly), které představují jednotlivé funkce. LabVIEW využívá programování na principu datového toku, kde tok dat procházející uzly v blokovém diagramu určuje posloupnost provádění funkcí (data flow model) [5].

Tvorba software v LabVIEW se skládá ze dvou částí. První částí je vytvoření grafického uživatelského rozhraní programu pomocí ovládacích prvků (tlačítka, stupnice, apod.) a indikátorů (různé typy grafů, LED diody, tabulky, apod.). Tato činnost se provádí na čelním panelu (Front Panel). Druhou částí je tvorba programového kódu, který propojí jednotlivé prvky z čelního panelu a přidělí jim požadované operace. Programový kód se tvoří v části zvané blokový diagram (Block Diagram).

Prostředí LabVIEW podporuje velké množství hardware, komunikaci pomocí různých sběrnic (GPIB, RS-232, RS-485, aj.) a protokolů (TCP, Blue-

tooth, aj.) a umožňuje do programového kódu vložit např. skripty z prostředí Matlab, vlastní dll knihovny, apod. Podporuje též moderní vývojovou platformu .NET.

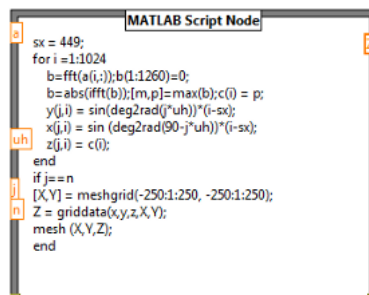
3.1.2 MATLAB

MATLAB je programové prostředí a skriptovací programovací jazyk. Je vyvíjen firmou The MathWorks. MATLAB je využíván pro vědeckotechnické numerické výpočty, modelování, návrhy algoritmů, počítačové simulace, analýzu a prezentaci dat, měření a zpracování signálů, návrhy řídicích a komunikačních systémů, atd. Nastavbou Matlabu je Simulink. Jedná se o program určený pro simulaci a modelování dynamických systémů. Simulink využívá algoritmy Matlabu pro numerické řešení především nelineárních diferenciálních rovnic [6].

Data jsou v MATLABu uložena ve formě matic. Matice jsou tam klíčovou datovou strukturou. A právě od této datové struktury je odvozen i název MATLAB, který vznikl zkrácením slov MATrix LABORatory. Programovací jazyk MATLABu je odvozen z jazyka Fortran [6].

3.2 Vzájemná komunikace

Grafickém vývojové prostředí LabVIEW umožňuje do programového kódu vložit MATLAB skript. Obr. 3 ukazuje příklad využití MATLAB skriptu. Do programové části LabVIEW vložíme funkci „MATLAB Script Node“. Do těla této funkce následně vložíme MATLAB skript a definujeme vstupní a výstupní hodnoty. Při vyhodnocení této funkce LabVIEW předá potřebné parametry jádru MATLABu, ten požadavek zpracuje a výsledné hodnoty vrátí LabVIEW.



```

a)
sx = 449;
for i = 1:1024
    b = fft(a(i:)); b(1:1260) = 0;
    b = abs(fft(b)); [m,p] = max(b); c(i) = p;
    y(i,i) = sin(deg2rad(j*uh))^(i-sx);
    x(i,i) = sin(deg2rad(90-j*uh))^(i-sx);
    z(i,i) = c(i);
end
uh)
end
if j == n
j)
[X,Y] = meshgrid(-250:1:250, -250:1:250);
n)
Z = griddata(x,y,z,X,Y);
mesh(X,Y,Z);
end
z)

```

Obrázek 3: funkce „MATLAB Script Node“ - způsob vložení MATLAB skriptu do LabVIEW.

4 Použitý hardware

4.1 Digitální kamera

K výrobě obrazových snímačů pro digitální kamery se využívají dvě technologie, CCD (charge coupled device) a CMOS (complementary metal oxide semiconductor). Obě technologie umožňují zaznamenat obraz pomocí elektrických signálů. Princip záznamu spočívá v konverzi světelného záření, dopadajícího na jednotlivé obrazové elementy (zvané pixely) světlocitlivého senzoru, na elektrický náboj. Počet lokálně generovaných elektronů přitom odpovídá intenzitě dopadajícího světelného záření. Rozdíl mezi těmito technologiemi je prakticky jen ve způsobu sběru signálu z buněk a v logice ovládání celého senzoru [18].

CCD technologie

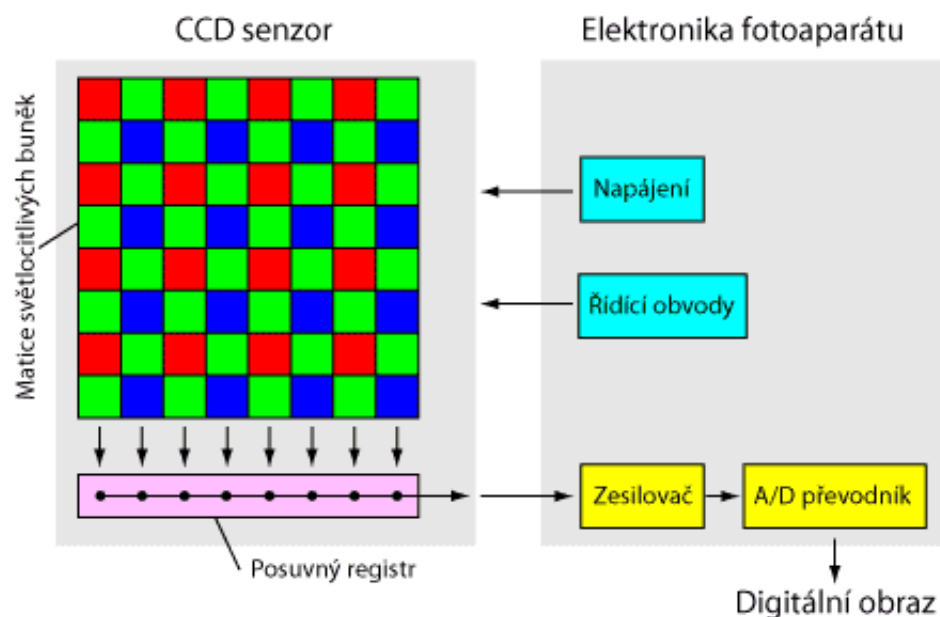
CCD senzor (Charged Coupled Device) je založen na schopnosti transportovat signál z buněk skrze jiné buňky, aniž by tím utrpěla kvalita signálu. Signál se postupně posouvá až k okraji, kde se nachází posuvný registr, který následně předá signál jeden po druhém do zesilovače a A/D převodníku. Posun signálu buňkami je zprostředkován pomocí nábojové vazby buněk - Charged Coupled (svázány nábojem).

Na rozdíl od ostatních integrovaných obvodů, CCD senzor používá odlišnou metodu výroby. Ta neumožňuje integrovat do CCD senzoru jinou elektroniku a tak většina řídicích obvodů včetně zesilovačů a A/D převodníků je mimo senzor. Proto se výsledného digitálního obrazu dosáhne až pomocí dalších integrovaných obvodů. Soubor integrovaných obvodů jako celek má potom i výrazně vyšší spotřebu ve srovnání s technologií CMOS a i výroba CCD senzoru je finančně nákladnější [19].

CMOS technologie

Výroba CMOS (Complimentary Metal Oxide Semiconductor) senzorů je relativně jednoduchá a nevyžaduje žádné specifické postupy. Je totiž založena na technologii CMOS, která se používá i pro výrobu ostatních integrovaných obvodů, jako např. pamětí a procesorů. Tento fakt se odráží na výrobní ceně, která je oproti výrobní ceně CCD senzorů podstatně levnější.

Díky technologii CMOS je možné přímo do senzoru integrovat pomocné elektronické obvody. Potom má každá buňka svoji vlastní elektronickou část



Obrázek 4: Schéma zpracování signálu ze CCD senzoru, včetně obslužné elektroniky [19].

včetně vlastního zesilovače. Díky této elektronice může být každá buňka adresována a čtena přímo, oproti CCD senzorům, kde je potřeba nejprve vyčíst celou matici hodnot. U technologie CCD jsou buňky čteny postupně, u CMOS je každá buňka adresována samostatně pomocí jejich souřadnic a proto může být i vyčtena samostatně bez nutnosti vyčtení celé matice hodnot [19].

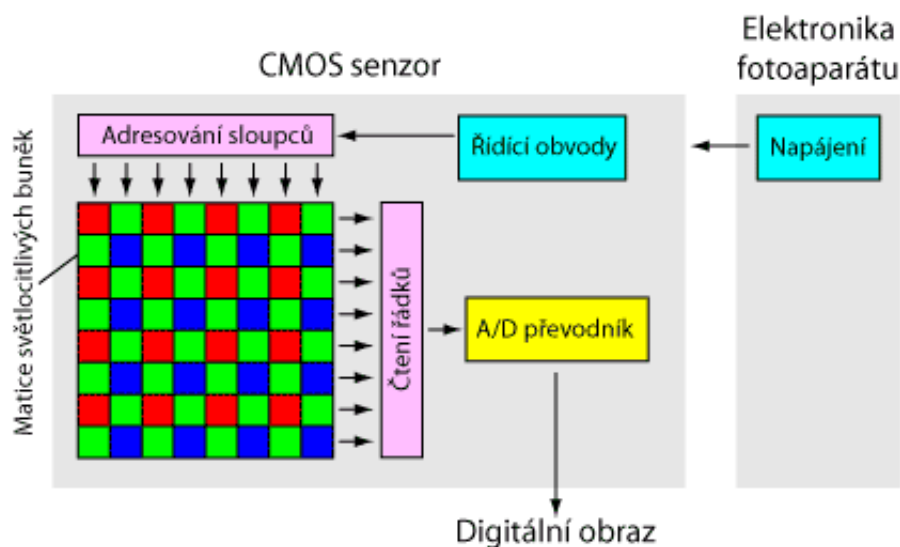
Shrnutí technologií CCD a CMOS

Tabulka Tab. 1 shrnuje rozdíly mezi technologiemi CCD a CMOS použitých u senzorů v digitálních kamerách a fotoaparátech. Zaměřuje se na základní vlastnosti jako je výrobní cena senzorů, funkčnost, složitost apod.

4.1.1 Kamera Lu120M firmy Lumenera Corporation

Lu120M je monochromatická průmyslová kamera od firmy Lumenera Corporation, která obsahuje 2/3" CMOS snímač. Komunikace s jednotkou PC probíhá pomocí rozhraní USB 2.0. Maximální rozlišení této 1,3 megapixelové kamery je 1280x1024 pixelů. Při maximálním rozlišení poskytuje rychlost až 15 snímků za sekundu (fps), při rozlišení 640x480 pixelů lze dosáhnout snímkovací frekvence 60 fps.

Technické specifikace modelu Lu120 jsou shrnuty v Tab. 2 nebo v [16].



Obrázek 5: Schéma zpracování signálu z CMOS senzoru, včetně obslužné elektroniky [19].

	CCD	CMOS
Cena	Drahý, protože výrobní linka musí být specializovaná na tento typ technologie.	Levný, protože využívá standardní technologii pro "běžné" integrované obvody (paměti, procesory atd.).
Odběr	Odebírá cca 50x více energie než CMOS.	Malý.
Šum	Malý, velká kvalita obrazu.	Díky menšímu fill faktoru a potřebě mikroobjektívů je šum větší.
Složitost	Velká, mnoho obvodů je mimo senzor.	Malá, vše potřebné je přímo v obvodu senzoru.
Funkčnost	Omezená, komplikovaný transport obsahu buněk.	Vysoká, maticově adresovatelné buňky.
Rychlost	Nízká, sekvenční čtení.	Vysoká, adresné čtení.

Tabulka 1: Shrnutí technologií CCD a CMOS [19].

Image Sensor	2/3" format, monochromatic 8.6 mm x 6.9 mm array
Effective Pixels	1280 x 1024, 6.7 μ square pixels
Frame Rate	15 fps at 1280x1024, faster with ROI
Sensitivity	Moderate
Dynamic Range	56 dB Rolling Shutter 52 dB Global Shutter +100 dB in multi-slope integration mode
White Balance	Auto / Manual
Power Requirement	USB 2.0 bus power, or external 6 VDC, 500 mA
Power Consumption	2.5 Watts
Operating Temperature	-10 °C to +50 °C
Operating Humidity	20 % - 80 %, Non-condensing
Interface Connector	Standard USB 2.0

Tabulka 2: Technické specifikace modelu Lu120M



Obrázek 6: CMOS kamera - model Lu120M firmy Lumenera Corporation [17].

4.2 Krokový motor

Krokový motor je zařízení, které převádí elektrické impulzy do diskretních mechanických pohybů. Jedná se o synchronní stroje, které mají přesně rozdělený plný úhel na daný počet kroků.

Jsou-li do krokového motoru posílány elektrické impulzy ve správném pořadí, hřídel krokového motoru se začne otáčet v diskretních přírůstkových krocích. Tyto kroky přímo souvisejí se vstupními elektrickými impulzy. Velikost kroku motoru souvisí s množstvím vstupních elektrických impulzů, rychlost závisí na frekvenci vstupních impulzů a směr otáčení krokového motoru je závislý na sekvenci vstupních impulzů elektrického signálu [8].

Díky rozdělení plného úhlu na daný počet kroků, lze u krokového motoru deterministicky dosáhnout libovolného stavu. Stačí jen znát stav minulý a stav po provedení n -kroků.

Následující seznam shrnuje hlavní výhody a nevýhody použití krokových motorů. Výhody krokového motoru:

- úhel rotace motoru je úměrný vstupnímu impulzu,
- posun motoru o jeden krok je velmi přesný, odchylka od zadané hodnoty se zpravidla pohybuje maximálně do 3% zadaného kroku,
- rychlá odezva na startovací, ukončovací a nebo reverzní impulz,
- dlouhá životnost krokového motoru závislá pouze na životnosti ložisek. Jádru motoru neobsahuje žádné mechanické části (např. kontaktní kartáče), které by se používáním rychle opotřebovávaly,
- odezva motoru na digitální impulzy poskytuje možnost "řízení v otevřené smyčce", což činí motory jednodušší a méně náročné na ovládání,
- rychlostí otáčení je úměrná frekvenci vstupních impulzů

Nevýhody krokového motoru:

- sklon k mechanickému zakmitávání, které může vést k nestabilitě při pohybu,
- vyšší náročnost řízení krokového motoru při vysokých rychlostech.

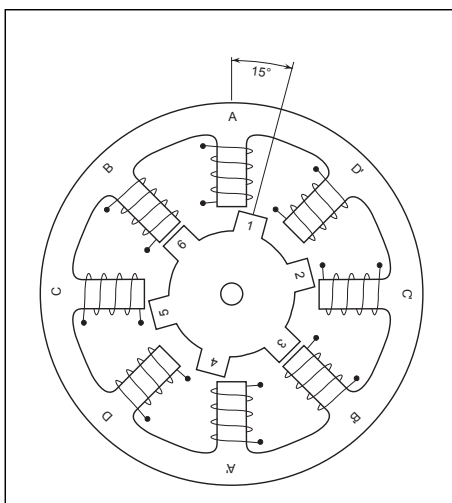
Krokové motory lze podle konstrukčního provedení rozdělit do tří základních skupin [9]:

- krokový motor s aktivním rotorem (permanentní Magnet Steppek Motor - MP),
- krokový motor s pasivním rotorem (variabilní Reluktance Steppek Motor - BR),
- hybridní krokový motor (hybrid Steppek Motor - HB).

Krokový motor s aktivním rotorem

Rotor krokového motoru s aktivním rotorem (Permanent Magnet Stepper Motor) je tvořen permanentním magnetem. Permanentní magnet je zmagnetizován tak, že se na něm střídají pruhy severního a pruhy jižního magnetického pólu. Magnetické póly rotoru tak poskytují vyšší hodnoty intenzity magnetického toku a proto v porovnání s VR motorem vykazuje motor PM lepší vlastnosti točivého momentu.

Obecné schéma krokového motoru s aktivním rotorem je zobrazeno na Obr. 7.



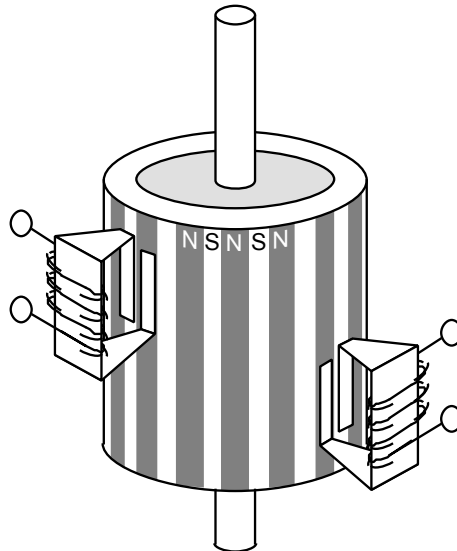
Obrázek 7: Obecné schéma krokového motoru s aktivním rotorem [8].

Krokový motor s pasivním rotorem

Krokový motor s pasivním rotorem (Variable Reluctance Stepper Motor) se skládá z multi-pólového železného rotoru a statoru. Využívá rozdílné mag-

netické vodivosti v příčné i podélné ose a rozdílného počtu pólů na statoru a rotoru.

Obecné schéma krokového motoru s pasivním rotorem je zobrazeno na Obr. 10.



Obrázek 8: Obecné schéma krokového motoru s pasivním rotorem [8].

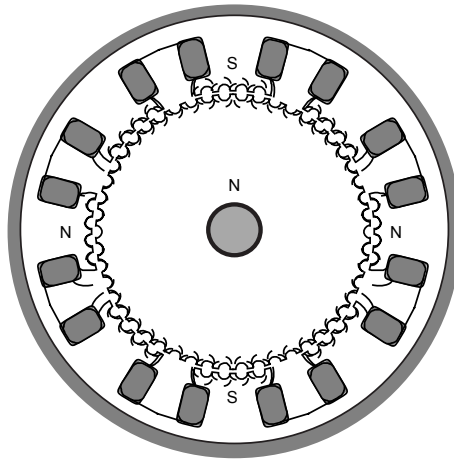
Hybridní krokový motor

Hybridní krokový motor (Hybrid Stepper Motor) kombinuje nejlepší vlastnosti z obou předchozích modelů, PM a VR krokových motorů. Rotor je multipólový, podobně jako u VR krokového motoru, a zároveň kolem hřídele je umístěn axiálně zmagetizovaný soustředný magnet, podobně jako u PM krokového motoru.

Obecné schéma hybridního krokového motoru je zobrazeno na Obr. 9.

4.2.1 Rotační krokový motor Newport PR50CC

K rotaci měřeného vzorku (implantát kloubní jamky kyčelního kloubu) je použit krokový motor PR50CC firmy Newport Corporation. Technické specifikace tohoto produktu jsou uvedeny níže v Tab. 3 a Tab. 4, nebo v [10].



Obrázek 9: Obecné schéma hybridního krokového motoru [8].

Base Material	Aluminum
Bearings	Ball bearings
Drive Mechanism	Ground worm gear
Worm Gear Ratio	1:63
Reduction Gear	11.8642:1
Motor	DC servo motor UE17CC, 1 Full step = 0.02°

Tabulka 3: Detaily modelu PR50CC

Travel Range (°)	360 continuous
Resolution (°)	0.01
Minimum Incremental Motion (°)	0.02
Uni-directional Repeatability (°)	0.02 typical, 0.05 guaranteed
Reversal Value (Hysteresis) (°)	0.03 typical, 0.1 guaranteed
Absolute Accuracy (°)	0.05 typical, 0.1 guaranteed
Maximum Speed (°/s)	20
Wobble (μ rad)	40 typical, 100 guaranteed

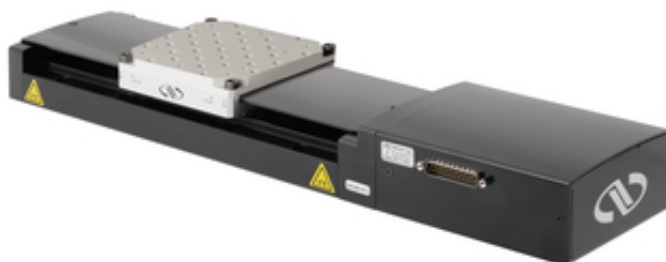
Tabulka 4: Technické specifikace modelu PR50CC



Obrázek 10: Krokový motor - model PR50CC firmy Newport Corporation [11].

4.2.2 Lineární krokový motor Newport ILS150CCL

K lineárnímu posuvu měřeného vzorku je použit lineární krokový motor ILS150CCL firmy Newport Corporation. Technické specifikace produktu jsou uvedeny níže v Tab. 5 a Tab. 6, nebo v [12].



Obrázek 11: Krokový motor - model ILS150CCL firmy Newport Corporation [13].

Base Material	Extruded Aluminum
Bearings	Double-row recirculating ball bearings
Drive Mechanism	Backlash-free ball screw
Drive Screw Pitch (mm)	2
Feedback	Screw mounted rotary encoder, 4,000 pts/rev, index pulse
Limit Switches	Optical
Origin	Optical, at center of travel, including mechanical zero signal
Motor	DC servo motor UE404S2

Tabulka 5: Detaily modelu ILS150CCL

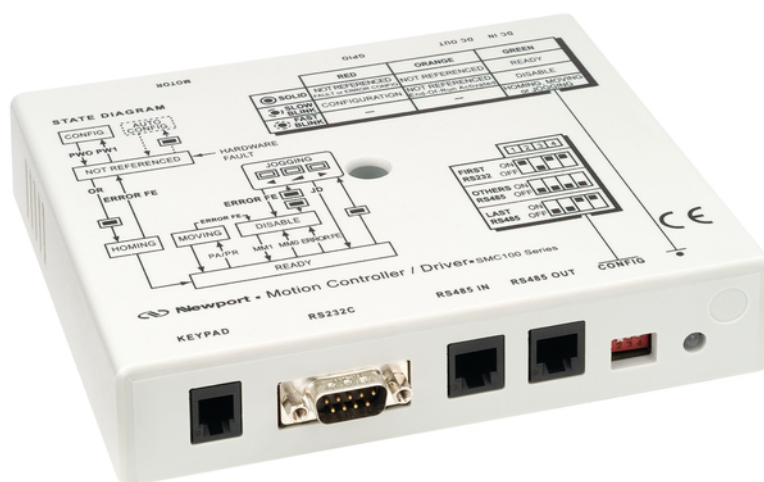
ILS1650CCL	Typical	Guaranteed
Travel Range (mm)	50 - 250	
Resolution (μm)	0.5	
Uni-directional Repeatability (μm)	0.7	1.5
Reversal Value (Hysteresis) (μm)	0.4	1
On-Axis Accuracy ¹ (μm)	2.5	5
Maximum Speed (mm/s)	50	
Pitch ⁽¹⁾ (μrad)	40	100
Yaw ⁽¹⁾ (μrad)	55	100

Tabulka 6: Technické specifikace modelu ILS150CCL

4.2.3 Řídící jednotka Newport SMC100CC

SMC100CC firmy Newport Corporation (viz Obr. 12) je jednoosá řídicí jednotka pro krokové motory. Komunikace s PC je zprostředkována pomocí integrovaného sériového rozhraní RS-232-C nebo pomocí rozhraní USB (toto zapojení vyžaduje externí adaptér SMC-USB). Díky integrovanému rozhraní RS-485 je možné vzájemné propojení více řídicích jednotek SMC100CC a tím tak dosáhnout víceosého zapojení krokových motorů. Sériové rozhraní RS-485 integrované do řídicí jednotky SMC100CC poskytuje plně duplexní režim, který zjednodušuje komunikaci mezi řídicími jednotkami a ovládání jednotlivých řídicích jednotek prostřednictvím softwaru [14].

Technické specifikace řídicí jednotky SMC100CC jsou shrnuty v Tab. 7.



Obrázek 12: Řídící jednotka krokového motoru - model SMC100CC firmy Newport Corporation [15].

Description	Single-axis motion controller/driver for DC servo motors (SMC100CC) or stepper motors (SMC100PP)
Controller Capability	DC servo motors in open or closed loop operation
Motor Output Power	48 VDC @ 1.5 A rms, 3 A peak, 100 kHz PWM switching frequency
Control Loop	Floating point digital PID loop with velocity and friction feedforward, 2kHz servo rate
Motion	Point-to-point motion with s-gamma motion profiler allowing acceleration and jerk time control; Backlash and Hysteresis compensation mode
Computer Interface	RS-232-C with 57,600 baud rate USB compatible with external adapter SMC-USB (requires Windows™ operating system) RS-485 internal link for chaining up to 31 controllers from the same COM port
Programming	40+ intuitive, 2 letter ASCII commands Command set includes software limits, user units, synchronized motion start, stop all
I/O	4 TTL out (open collector) 4 TTL in (2.21 kΩ pull up to 5 V) 1 analog input, ±10 V, 8-Bit
Dedicated Inputs	RS-422 differential encoder input for A, B, and I, max. 2 MHz rate Forward and reverse limit, home switch and index pulse
Dedicated Outputs	1 open-collector output for “InMotion” 1 open collector output for “NotReferenced”
Display	Two color LED
Internal Safety Feature	Watchdog timer
Software	Windows Utility program LabView™ drivers

Tabulka 7: Technické specifikace modelu SMC100CC

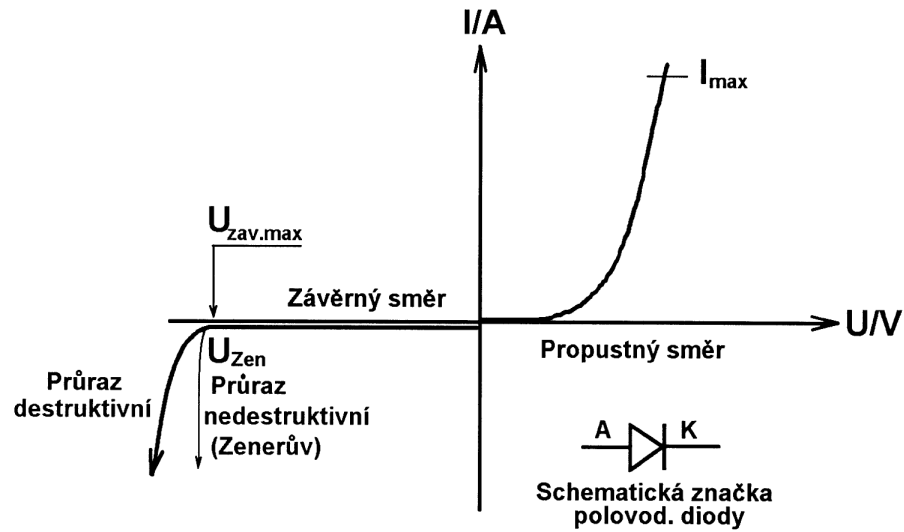
4.3 Projektor stopy

Projektorem stopy je červená LED dioda. Světlo vyzařované touto diodou je usměrněno pomocí válcové čočky. Ta vytvoří svislou tenkou stopu, kterou je následně osvětlen zkoumaný předmět. Dioda pracuje v podprahovém režimu. V tomto režimu je světlo produkované diodou nekoherentní, průběh intenzity má gaussovský tvar a světelná stopa na povrchu předmětu nevytváří koherenční zrnitost (tzv. speckle), která by byla v daném případě nežádoucí a ovlivňovala by výslednou přesnost měření. V nadprahovém režimu dioda produkuje koherentní záření, které na povrchu předmětu vytváří koherenční zrnitost. Proto použitím diody v podprahovém režimu, získáme přesnější výsledky než při použití diody v nadprahovém režimu.

Polovodičová dioda

Polovodičová dioda je elektronická součástka, která obsahuje zpravidla jeden PN přechod. Dioda se skládá ze dvou typů příměsových polovodičů, z polovodiče typu N a polovodiče typu P. Polovodič typu N vznikne přidáním pětímocného prvku (např. arsen As, antimon Sb) do čistého čtyřmocného prvku (např. křemík Si). V polovodiči typu N převládá počet elektronů nad počtem děr. Elektrony se stávají majoritními nosiči elektrického náboje. Polovodič typu P naopak vznikne přidáním trojmocného prvku (např. hliník Al, gallium Ga) do prvku čtyřmocného (např. křemík Si). V polovodiči typu P převládá počet děr nad počtem elektronů a ty se stávají majoritním nosičem elektrického náboje [21].

Na Obr. 13 je znázorněna volt-ampérová (V-A) charakteristika polovodičové diody. V propustném směru začne proud procházet po překonání potenciálové bariéry, která se pro polovodičové diody z křemíku pohybuje na hranici 0,65 V, pro diody z germania je tato hranice mezi 0,2 V až 0,3 V. Proud I_{\max} je maximální proud v propustném směru, po jehož překročení dochází ke zničení polovodičové diody. V závěrném směru protéká diodou malý závěrný proud tvořený minoritními nosiči náboje. Po překročení maximálního závěrného napětí $U_{\text{zav.max}}$ dochází lavinovitému nárůstu proudu. Dochází k destruktivnímu průrazu polovodičové diody [23].



Obrázek 13: V-A charakteristika polovodičové diody [23].

PN přechod

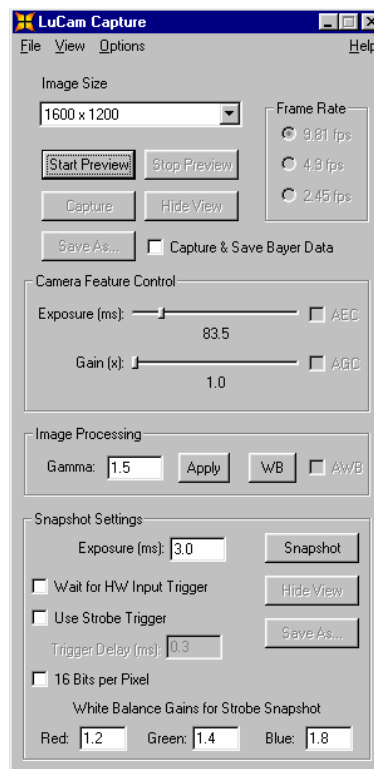
Na rozhraní spoje polovodiče typu P a polovodiče typu N vznikne oblast, tzv. PN přechod. PN přechod vzniká difuzí polovodiče typu P do polovodiče typu N za vysokých teplot. Vlastností přechodu PN je, že propouští elektrický proud pouze jedním směrem. Může být proto zapojen buď v propustném směru a nebo ve směru závěrném [22]. Přechod PN je využit v polovodičových součástkách jako jsou diody, tranzistory a další.

Pokud se k polovodiči typu P připojí kladný pól elektrického zdroje a k polovodiči typu N záporný pól zdroje, PN přechod je zapojen v propustném směru. Naopak, zapojí-li se kladný pól zdroje k polovodiči typu N a záporný pól zdroje k polovodiči typu P, PN přechod bude zapojen v závěrném směru.

5 Použitý software

5.1 Digitální kamera

K digitální kameře Lu120M dodává firma Lumenera Corporation relativně širokou nabídku software a ovladačů. Uživatel produktu řady Lu120 od firmy Lumenera Corporation může pro běžná použití využít software "LuCam Software" (viz Obr. 14), který je dodáván přímo s kamerou, příp. je dostupný na webu firmy Lumenera. Pro uživatele, kteří chtějí ovládání kamery začlenit do vlastní aplikace, nabízí firma Lumenera Software Development Kit (SDK). SDK již ale není volně ke stažení a je třeba si jej u firmy Lumenera objednat. Poslední skupinou, která může kameru začlenit do svých aplikací, jsou uživatelé software Matlab (od firmy MathWorks) a LabVIEW (od firmy National Instruments). Potřebné knihovny jsou dodávány společně s kamerou, příp. jsou dostupné z webu firmy Lumenera, viz [20].



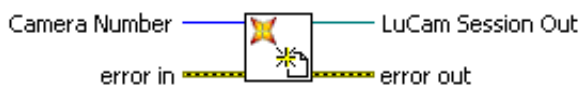
Obrázek 14: Hlavní okno programu "Lucam Software".

5.1.1 National Instruments LabVIEW Plug-in

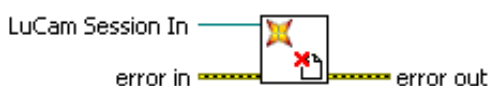
V následující kapitole bude blíže popsána obsluha digitální kamery Lu120M v prostředí LabVIEW. Firma Lumenera k tomuto produktu nabízí přímo plug-in, který umožní začlenit obsluhu kamery do projektů tvořených v prostředí LabVIEW.

V následujícím textu budou popsány moduly, které jsou využity přímo v projektu automatizace měření.

Na začátku práce s kamerou je potřeba vytvořit spojení kamery a LabView. K tomuto účelu slouží modul LuCamOpen.vi (viz Obr. 15). Na vstupu je potřeba zadat číslo kamery. Číslování začíná od čísla 1. Pokud je v systému zapojena pouze jedna Lumenera kamera, stačí na vstup připojit číslo 1. Na výstupu daného modulu dostaneme referenci na právě otevřenou relaci. Tato relace pak slouží jako vstupní hodnota k ostatním modulům pro práci s Lumenera kamerou a zároveň je i výstupní hodnotou daných modulů. Na konci práce s kamerou je potřeba ukončit danou relaci (spojení). To se provede použitím modulu LuCamClose.vi (Obr. 16).



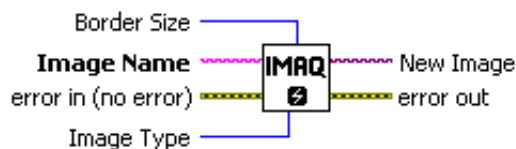
Obrázek 15: LuCamOpen.vi - slouží k otevření portu pro komunikaci s kamerou.



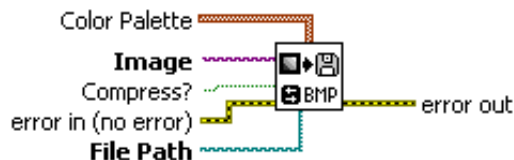
Obrázek 16: LuCamClose.vi - slouží k uzavření portu pro komunikaci s kamerou.

Neboť výstupem z kamery jsou data ve formě snímků, je vhodné na začátku vytvořit objekt obrázek. Tento krok se provede pomocí vestavěné funkce „IMAQ Create“ (Obr. 17). Výstupem této funkce je obrázek zvoleného typu (např. Grayscale-U8, RGB-U32 a další).

Data, která se nachází v objektu obrázek lze uložit do různých formátů. V našem případě se využil formát BMP. Jedná se o formát pro ukládání rastrové grafiky. Data z objektu obrázek lze v LabView uložit do formátu BMP s využitím funkce „IMAQ write bmp file“ (Obr. 18).

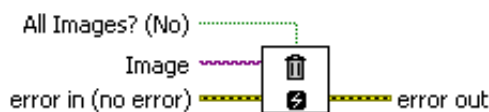


Obrázek 17: Funkce „IMAQ create“ vytvoří objekt „obrázek“.



Obrázek 18: Funkce „IMAQ write bmp file“ uloží obrázek.

Po ukončení práce s objektem „obrázek“ vytvořeným pomocí funkce „IMAQ Create“ je potřeba tento objekt řádně ukončit. K ukončení slouží další ve-stavěná funkce „IMAQ Dispose“ (Obr. 19). Zpravidla se volá před ukončením programu.

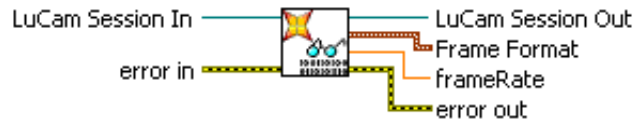


Obrázek 19: Funkce „IMAQ Dispose“ zruší objekt „obrázek“.

Abychom mohli pracovat s nastavením kamery, tj. měnit různé parametry jako světlost, kontrast, dobu expozice a další, je vhodné po otevření relace použít modul LuGetFormat.vi (Obr. 20). Výstupem je klastr vlastností a hodnot, které lze na kameře nastavovat.

K jednoduchému získání snímku z kamery slouží následující 3 moduly, LuEnableFastFrames.vi, LuFastSnap.vi a LuDisableFastFrames.vi. Nejdříve je potřeba pomocí modulu LuEnableFastFrames.vi (Obr. 21) umožnit získání snímku z kamery. Následně se využije modulu LuFastSnap.vi (Obr. 23), který získaný snímek z kamery převede s využitím NI-IMAQ do podoby obrázku. Tento modul musí vždy následovat po modulu LuEnableFastFrames.vi a zároveň před modulem LuDisableFastFrames.vi. Po získání snímku je třeba řádně ukončit mód získání snímku. K tomuto kroku slouží modul LuDisableFastFrames.vi (Obr. 22).

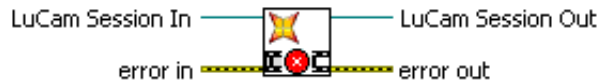
Digitální kameru Lu120M od firmy Lumenera lze využít i v režimu vi-



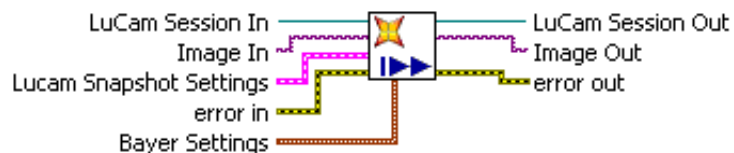
Obrázek 20: LuGetFormat.vi - určen k zjištění hodnot a vlastností digitální kamery.



Obrázek 21: LuEnableFastFrames.vi - umožní získání snímku z kamery.

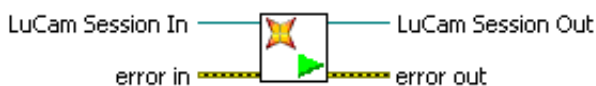


Obrázek 22: LuDisableFastFrames.vi - ukončí mód získání snímku z kamery.



Obrázek 23: LuFastSnap.vi - získá snímek z digitální kamery.

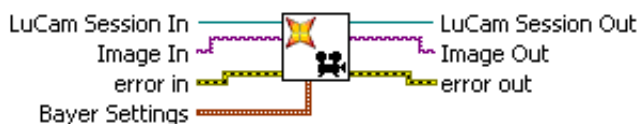
dea. Pro tuto funkci slouží následující 3 moduly, LuStartStream.vi, LuTakeVideo.vi a LuStopStream.vi. Spuštění videa se provádí pomocí modulu LuStartStream.vi (Obr. 24). K získání jednotlivých snímků z kamery a k jejich následnému zpracování slouží modul LuTakeVideo.vi (Obr. 26). Tento modul musí vždy následovat po modulu LuStartStream.vi a zároveň musí být umístěn před modulem LuStopStream.vi. Režim videa je ukončen modulem LuStopStream.vi (Obr. 25).



Obrázek 24: LuStartStream.vi - spustí režim videa.

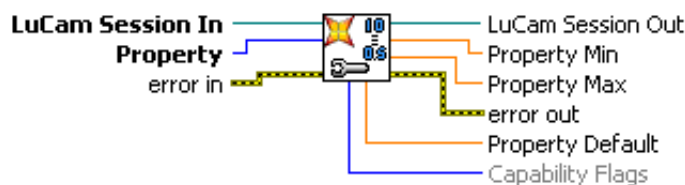


Obrázek 25: LuStopStream.vi - ukončí režim videa.

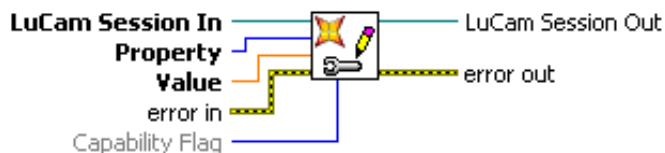


Obrázek 26: LuTakeVideo.vi - zobrazí živé vysílání z digitální kamery.

Poslední dva použité moduly v projektu automatizace měření jsou modul LuPropRange.vi (Obr. 27) a modul Lu SetProperty.vi (Obr. 28). Modul LuPropRange.vi vrací na výstupu minimální, maximální a standardní hodnotu požadované vlastnosti (např. světlost, kontrast, doba expozice, a další). Modul Lu SetProperty.vi slouží k nastavení hodnoty zvolené vlastnosti. Tento modul se často využívá v kombinaci s modulem LuPropRange.vi.



Obrázek 27: LuPropRange.vi - vrací minimální, maximální a standardní hodnotu požadované vlastnosti.



Obrázek 28: LuSetProperty.vi - slouží k nastavení hodnoty zvolené vlastnosti.

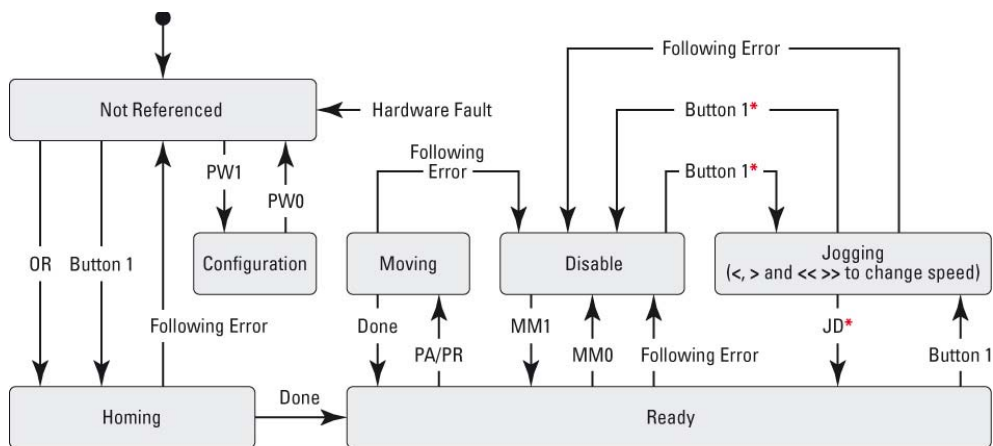
5.2 Krokový motor

Krokové motory firmy Newport se ovládají skrze příslušné řídicí jednotky. Námi použité krokové motory PR50CC a ILS150CCL lze řídit pomocí jednotky SMC100CC (Obr. 12).

SMC100CC

Ovládání řídicí jednotky SMC100CC je rozděleno do několika provozních stavů. Toto rozdělení má zaručit bezpečné a konzistentní operace. Stavový diagram řídicí jednotky SMC100CC je zobrazen na Obr. 29. Je vidět, že jednotka obsahuje 7 provozních stavů - NOT REFERENCE, HOMING, READY, CONFIGURATION, MOVING, DISABLE a JOGGING. Každý stav dovoluje jen omezené ovládání řídicí jednotky. Po zapojení SMC100CC k napájení přejde řídicí jednotka do stavu NOT REFERENCE. Z tohoto stavu je možné přejít do stavu CONFIGURATION. Stav CONFIGURATION umožňuje nastavení různých parametrů týkajících se řídicí jednotky a krokového motoru. Ze stavu NOT REFERENCE do stavu CONFIGURATION lze přejít pomocí příkazu PW1, pro opačný směr se využije příkazu PW0.

Po ukončení konfigurace se řídicí jednotka vrátí do stavu NOT REFERENCE. Budeme-li chtít řídicí jednotce poslat příkaz, který vykoná pohyb krokového motoru, je prvně potřeba dostat řídicí jednotku ze stavu NOT REFERENCE do stavu READY. Řídicí jednotce se pošle příkaz OR, v prostředí



* No action, when jogging speed is different than zero, e.g. one of the keys "<", ">" or "<<>>" is pressed.

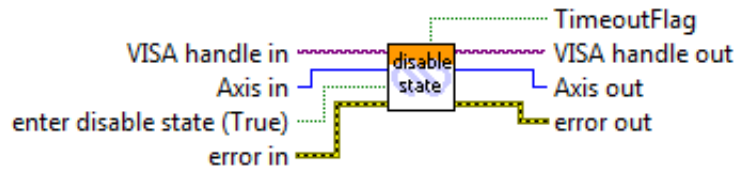
Obrázek 29: Stavový diagram řídicí jednotky SMC100CC.

LabView se využije modulu „Home.vi“ (viz Obr. 36), kterým se spustí vyhledávání počáteční pozice krokového motoru. Během této operace je řídicí jednotka ve stavu HOMING. Pokud je vyhledání počáteční pozice úspěšné, krokový motor se přesune do této pozice a stav řídicí jednotky se automaticky změní na READY. V tomto stavu je krokový motor napájen elektrickou energií.

Je-li řídicí jednotka ve stavu READY, může přijímat příkazy, které souvisejí s pohybem krokového motoru. Jedná se o příkazy PA a PR (move absolute a move relative). V prostředí LabView jsou tyto funkce volány moduly „Move absolute.vi“ (Obr. 38) a „Move relative.vi“ (Obr. 39). Při provádění jednoho z příkazů PA nebo PR se řídicí jednotka přepne do stavu MOVING (viz Obr. 29). Po úspěšném provedení příkazu se navrátí zpět do stavu READY. Vyskytnou-li se v průběhu zpracování příkazu PA nebo PR problémy, řídicí jednotka přejde ze stavu MOVING do stavu DISABLE.

Dostane-li se řídicí jednotka do stavu DISABLE, napájení krokového motoru se vypne. Do stavu DISABLE se lze dostat i přes příkaz MM0 nebo v prostředí LabView pomocí modulu „Enter Leave disable state.vi“ (Obr. 30), zpět do stavu READY příkazem MM1 nebo pomocí modulu „Enter Leave disable state.vi“.

Posledním provozním stavem řídicí jednotky je stav JOGGING. Tento stav lze využít jen s použitím přídatné SMC-RC klávesnice, která umožňuje ovládat řídicí jednotku s krokovým motorem i bez použití počítače. Do stavu JOGGING se lze dostat ze stavů DISABLE a READY a to jen stisknutím



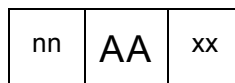
Obrázek 30: Modul „Enter Leave disable state.vi“ - přepne řídicí jednotku do/z stavu DISABLE.

příslušného tlačítka na SMC-RC klávesnici [24].

Firma Newport nabízí dvě možnosti ovládání řídicí jednotky, pomocí jednoduchých ASCII příkazů nebo přes programovatelné rozhraní LabView.

ASCII příkazy

První možností je využití jednoduchých ASCII příkazů posílaných po sériové lince. Základem syntaxe je dvou písmenný příkaz, kterému předchází a následují další volitelné parametry. Syntaxe výsledného příkazu je znázorněna na Obr. 31.



Obrázek 31: Obecný předpis syntaxe pro řídicí jednotku SMC100CC.

První část syntaxe, nn, je rezervována pro adresu příslušné řídicí jednotky. Tato část je volitelná. Používá-li se pouze jedna řídicí jednotka, je možné tuto část syntaxe vynechat. Prostřední část syntaxe na Obr. 31, AA, reprezentuje hlavní a tudíž povinnou část. Pomocí této části provádíme na krokovém motoru potřebné úkony (např. pro zjištění aktuální pozice slouží příkaz TP). Poslední část syntaxe na Obr. 31, xx, je opět volitelná a slouží pro nastavení hodnoty daného příkazu. Pro dotaz na aktuální hodnotu daného příkazu se „xx“ nahradí symbolem „?“.

Příklad: pro zjištění aktuálně nastavené rychlosti pohybu krokového motoru, který je ovládán řídicí jednotkou s adresou 1 se na řídicí jednotku pošle příkaz „1VA?“. Pro přenastavení rychlosti pohybu téhož motoru na hodnotu 50 unit/s by příkaz vypadal následovně - „1VA50“.

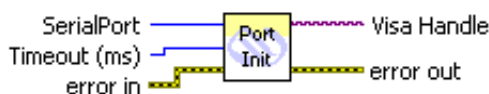
Podrobnější informace o ASCII příkazech, které je možno využít k řízení řídicí jednotky a krokového motoru, se nacházejí v manuálu [24].

LabVIEW plug-in

Druhou možností, jak komunikovat s řídicí jednotkou krokového motoru, je využít plug-in pro prostředí LabView, který firma Newport dodává k řídicím jednotkám. Tento plug-in je též volně ke stažení na stránkách firmy Newport Corporation [15] v sekci „Downloads“.

Funkce pro komunikaci s řídicí jednotkou krokového motoru jsou do prostředí LabView přidány ve formě modulů. V následující části textu budou popsány moduly, které jsou využity přímo v projektu automatizace měření.

Pro komunikaci s řídicí jednotkou krokového motoru je nejdříve potřeba otevřít příslušný komunikační port linky RS-232. K otevření portu slouží modul „Port Init.vi“ (viz Obr. 32). Jako povinná vstupní hodnota slouží číslo sériového portu, které se má pro komunikaci otevřít. Bude-li probíhat komunikace na portu COM1, na vstupu „SerialPort“ bude číslo 1, pro COM2 by bylo číslo 2, atd. Některé parametry komunikace po RS-232 jsou pevně dané. Patří mezi ně přenosová rychlost (hodnota 57600), datové bity (8), parita (žádná), stop bit (1), řízení toku (XON/XOFF) a ukončovací znak (CR/LF).



Obrázek 32: Modul „port init.vi“ - otevře port pro komunikaci s řídicí jednotkou krokového motoru.

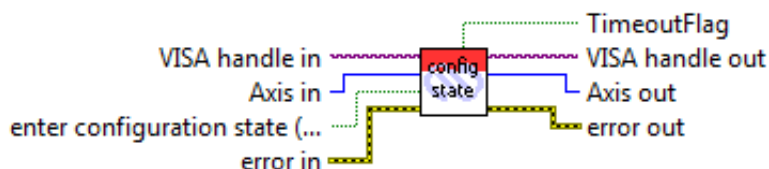
Po ukončení práce s řídicí jednotkou je potřeba řádně ukončit i komunikaci na daném sériovém portu. Tato operace se provede za pomoci modulu „Close.vi“ (Obr. 33).



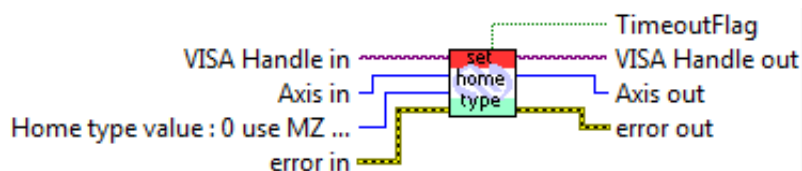
Obrázek 33: Modul „close.vi“ - uzavře port pro komunikaci s řídicí jednotkou krokového motoru.

Po zapnutí napájení řídicí jednotky je řídicí jednotka v režimu NOT REFERENCE. Aby bylo možné používat řídicí jednotku k ovládání pohybu krokového motoru, je potřeba přejít ze stavu NOT REFERENCE do stavu READY. Podle Obr. 29 se do stavu READY dostaneme pomocí mezistavu HOMING.

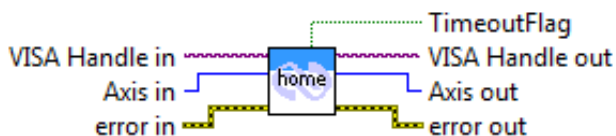
Pokud je řídicí jednotka ve stavu NOT REFERENCE, pomocí modulu „Set HOME type.vi“ (Obr. 35) se řídicí jednotce pošle příkaz na nastavení typu počáteční pozice a poté se pomocí modulu „Home.vi“ (Obr. 36) pošle příkaz k přesunu na počáteční pozici. Řídicí jednotka přijme příkaz k přesunu mezi stavy a začne provádět mezistav HOMING. Pokud tento mezistav skončí bez chyb, mezistav HOMING automaticky přejde do stavu READY.



Obrázek 34: Modul „Enter Leave configuration state.vi“.



Obrázek 35: Modul „Set HOME type.vi“ - nastaví počáteční pozici krokového motoru.

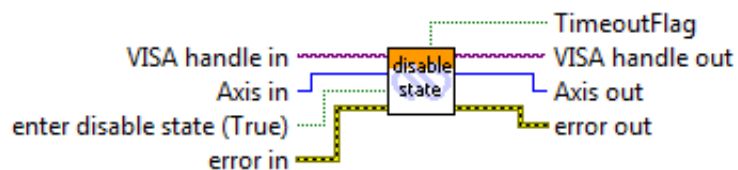


Obrázek 36: Modul „Home.vi“ - přesune krokový motor do počáteční pozice.

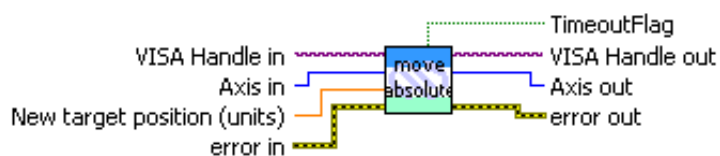
K pohybu krokového motoru slouží dva moduly. Modul „Move absolute.vi“ (Obr. 38) a modul „Move relative.vi“ (Obr. 39). Modul „Move absolute.vi“ slouží k posunu krokového motoru na zadanou pozici vzhledem k počáteční nulové pozici. Naopak modul „Move relative.vi“ slouží k posunu krokového motoru o zadanou hodnotu vzhledem k poslední dosažené pozici.

Pro zjištění hodnoty absolutní pozice vzhledem k počáteční nulové hodnotě slouží modul „Get absolute position.vi“ (Obr. 40).

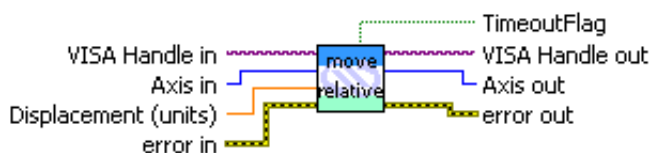
Dalšími užitečnými moduly jsou „Get negative position limit.vi“ (Obr. 42) a „Get positive position limit.vi“ (Obr. 41). Hodnoty získané danými moduly



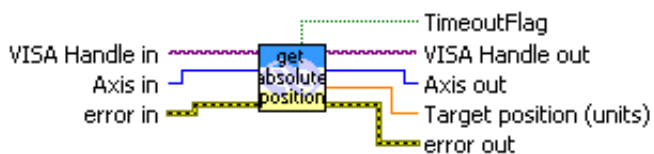
Obrázek 37: Modul „Enter Leave disable state.vi“.



Obrázek 38: Modul „move absolute.vi - posune krokovým motorem na zadanou pozici“.

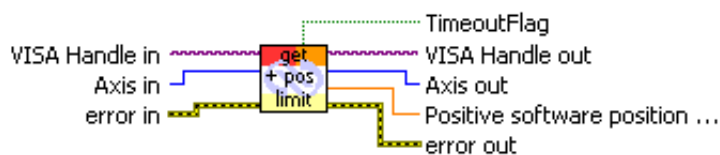


Obrázek 39: Modul „move relative.vi“ - posune krokovým motorem o zadaný krok.

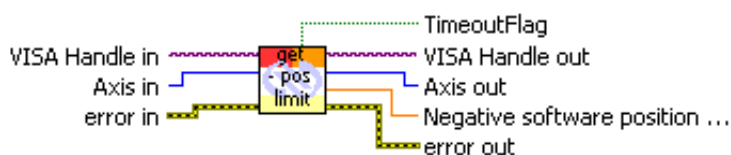


Obrázek 40: Modul „get absolute position.vi“ - vrátí hodnotu aktuální absolutní pozice.

určují maximální interval hodnot, ve kterém lze realizovat posuv krokového motoru.



Obrázek 41: Modul „get positive position limit.vi“ - vrátí maximální dosažitelnou pozici v kladném směru.



Obrázek 42: Modul „get negative position limit.vi“ - vrátí maximální dosažitelnou pozici v záporném směru.

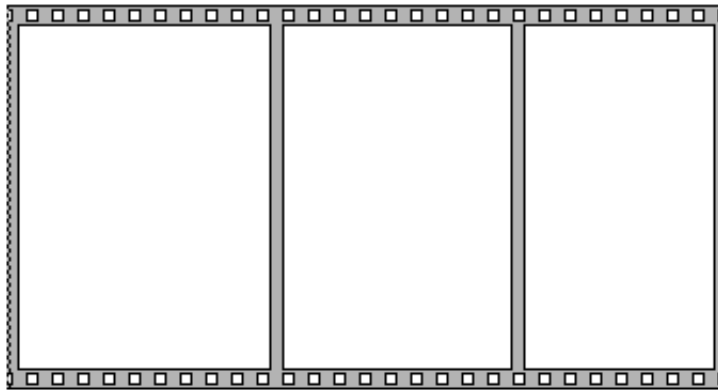
5.3 Synchronizace

Jednou z důležitých částí programu sloužícího pro automatizaci měření otěru jamky kyčelního implantátu je samotná synchronizace. Program komunikuje s různým hardware (krokové motory, digitální kamera) a je potřeba, aby byla tato komunikace správně sladěna.

Vezměme si vlastní měření kyčelní jamky. Každý cyklus tohoto měření sestává ze tří kroků. Prvním krokem je pořízení snímku projekční stopy osvětlující zkoumaný předmět. Ve druhém kroku se předmětem pootočí o daný úhel. Posledním krokem je analýza stopy ze získaného snímku. Aby vše fungovalo, je potřeba vhodně sesynchronizovat komunikaci programu s krokovým motorem, digitální kamerou a zpracováním obrazu pomocí MATLAB skriptu.

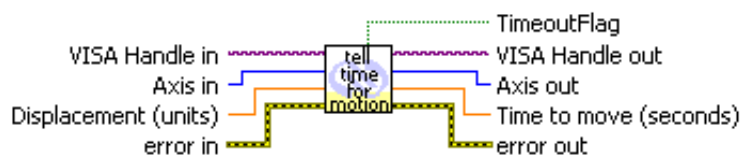
Je mnoho způsobů, jak vyřešit problém synchronizace. Jednou z možností je využití „flat sequence structure“ (viz Obr. 43).

Tento „flat sequence“ vykonává dané příkazy sekvenčně. To znamená, že dokud se nevykoná požadavek v aktuální části „flat sequence“, následující část „flat sequence“ nebude spuštěna. Tento postup se dá dobře využít pro části získání snímku z digitální kamery (a současně jeho uložení) a analýzou stopy



Obrázek 43: Funkce „Flat Sequence Structure“ pro sekvenční vykonávání příkazů.

z daného snímku (pomocí MATLAB skriptu). U pootočení předmětu pomocí krokového motoru však může nastat problém. Tato část cyklu je označena jako splněná v okamžiku, kdy je na řídicí jednotku krokového motoru odeslán požadavek na pootočení předmětem. Následně je vykonána další část „flat sequence“. Proto se může stát, že krokový motor bude stále v pohybu a přitom už bude digitální kamera pořizovat snímek. Aby této situaci zabránilo, následuje po části obsluhující krokový motor dodatečná sekvence, která pouze pozdrží vykonávání programu. V této sekvenci se nalézá funkce „Tell time for relative motion“ (viz Obr. 44), jejíž výstupem je doba potřebná pro vykonání pohybu krokového motoru o zadané velikosti kroku.



Obrázek 44: Modul „tell time for relative motion.vi“ - vrátí čas potřebný k vykonání pohybu krokového motoru.

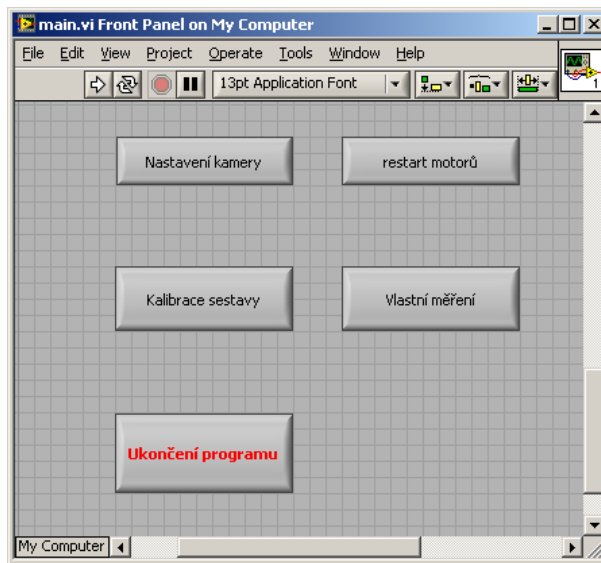
Samozřejmě, že tento způsob synchronizace jednotlivých částí měřicího zařízení není jediný použitelný a určitě by se dal vylepšit. Ale pro tento případ je dostačující a funkční.

5.4 Uživatelské rozhraní

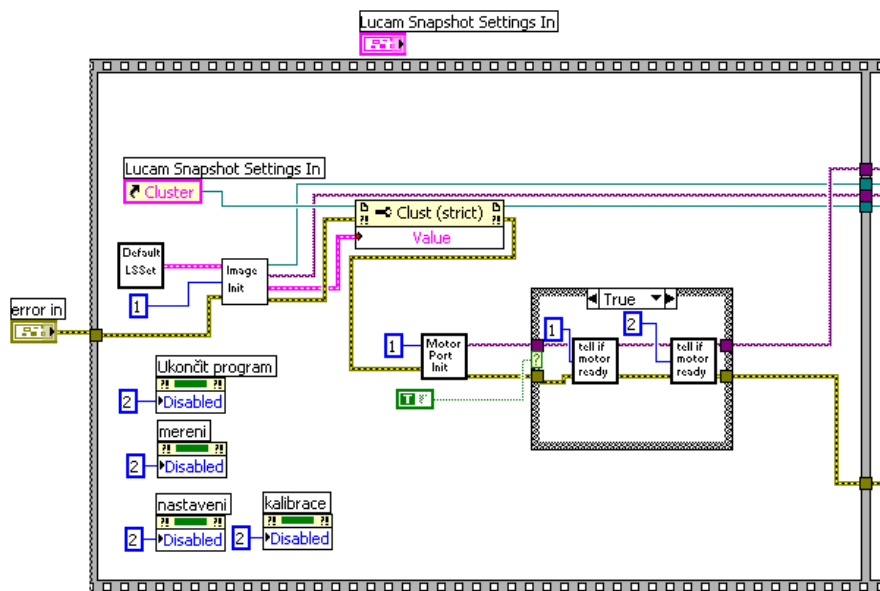
K tvorbě programu pro automatizaci měření otěru kyčelního implantátu bylo zvoleno grafické programovací prostředí LabVIEW. Jednou z výhod je relativně jednoduchá a intuitivní tvorba programu včetně grafického uživatelského rozhraní. Díky nespočetnému množství zabudovaných funkcí a využití programovacího jazyka "G" je tvorba programového kodu uživatelsky přívětivá. Dalším faktorem, které ovlivnilo výběr prostředí LabVIEW byla dostupnost ovladačů a knihoven pro práci s krokovými motory a digitální kamerou i pro prostředí LabVIEW.

Grafické rozhraní programu je jednoduché (Obr. 45), obsahuje pouze potřebné ovládací prvky (nastavení kamery, reset motorů, kalibrace sestavy, vlastní měření, ukončení programu). Kód programu je rozdělen do tří částí. První je částí inicializační (Obr. 46). V této části se provedou potřebné úkony pro komunikaci s digitální kamerou a krokovými motory, tj. otevřou se příslušné porty pro komunikaci a řídicí jednotky krokových motorů se nastaví do stavu „Ready“. Druhou částí je hlavní smyčka programu (Obr. 47), která následuje po inicializační části. V této části programu je pak možné provádět nastavení vlastností digitální kamery, je možné provést kalibraci experimentální sestavy a provést vlastní měření na daném předmětu, kterým je jamka kyčelního implantátu. Ve třetí části programu, která je vyvolána ukončením programu, jsou řádně ukončeny veškeré komunikace s krokovými motory a digitální kamerou a příslušné porty jsou uzavřeny.

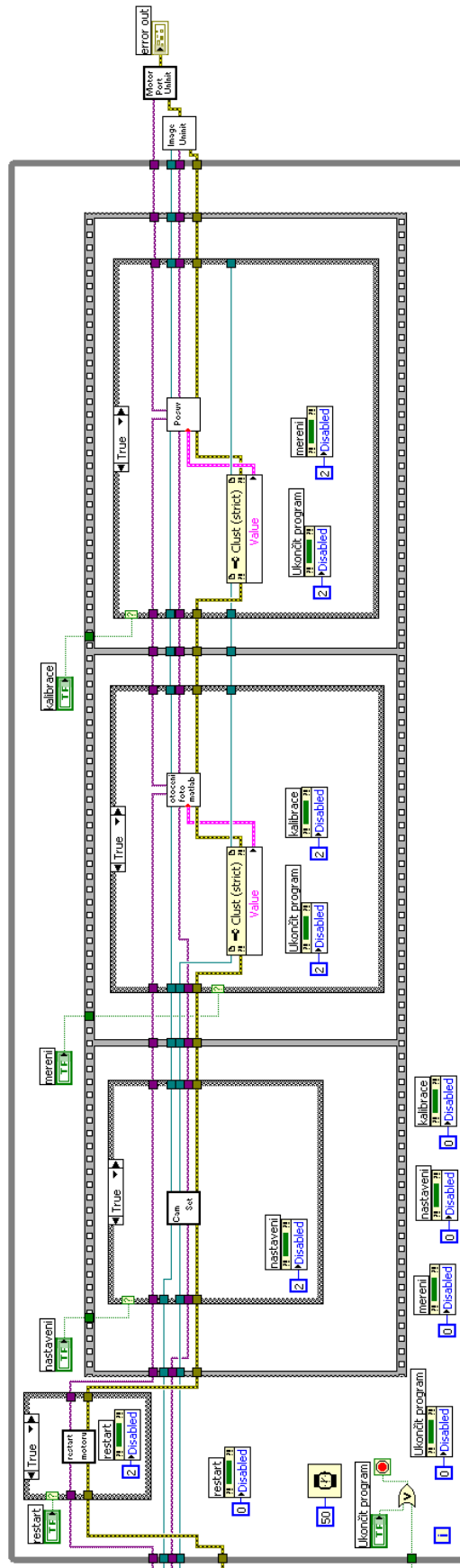
Hlavní smyčka programu obsahuje kód pro vykonání nastavení kamery, resetu řídicích jednotek krokových motorů, kalibrace sestavy a vlastního měření. Každá tato část je uložena jako samostatný „vi“ soubor, který je v případě potřeby zavolán a vykonán.



Obrázek 45: Hlavní panel programu



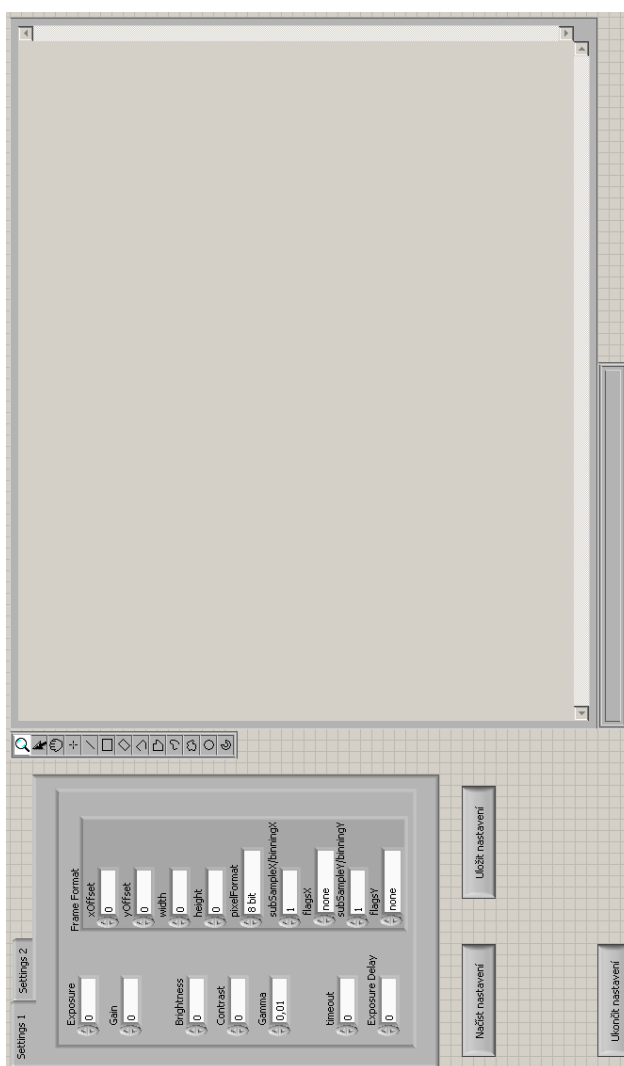
Obrázek 46: Inicializační část programu - programový kód. Slouží k otevření komunikačních portů digitální kamery a krokových motorů.



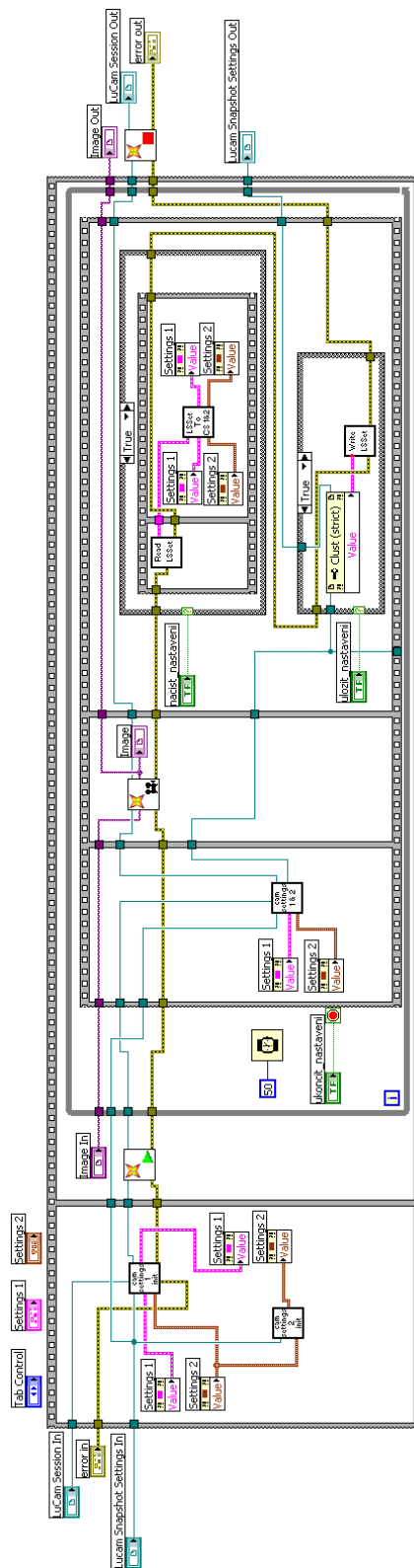
Obrázek 47: Hlavní cyklus programu - programový kód.

5.4.1 Nastavení kamery

Tato část programu má na starosti nastavení různých vlastností digitální kamery jako světlost, kontrast obrazu, expoziční doba a další. Na počátku je načteno standardní nastavení digitální kamery. To je možné podle potřeb upravit. Aktuální nastavení je možné průběžně sledovat, neboť v pravé části okna „nastavení kamery“ běží živé vysílání. Po nastavení příslušných parametrů je možné si toto nastavení uložit pro pozdější použití. Data jsou ukládána do formátu XML.



Obrázek 48: Nastavení kamery - přední panel. Tato část slouží k nastavení parametrů digitální kamery.

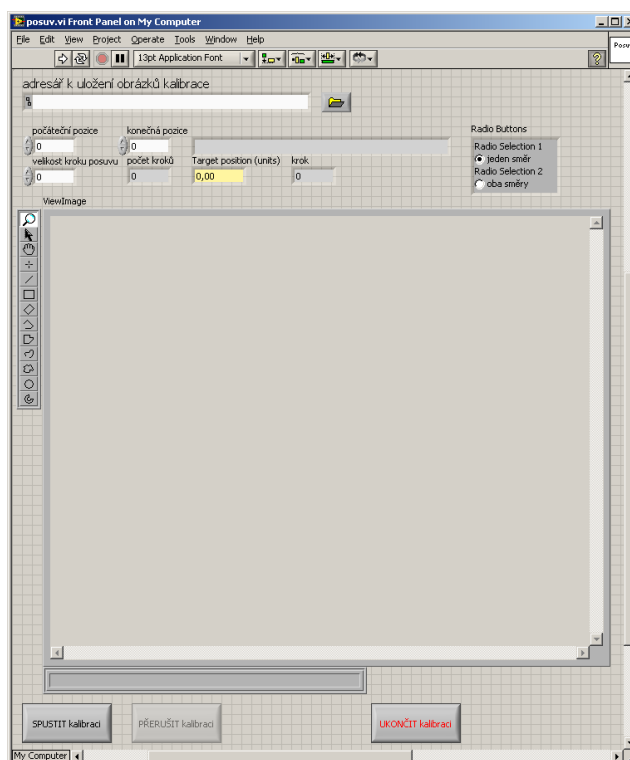


Obrázek 49: Nastavení kamery - programový kód, který obsluhuje nastavení jednotlivých parametrů digitální kamery a zobrazuje živý náhled z kamery.

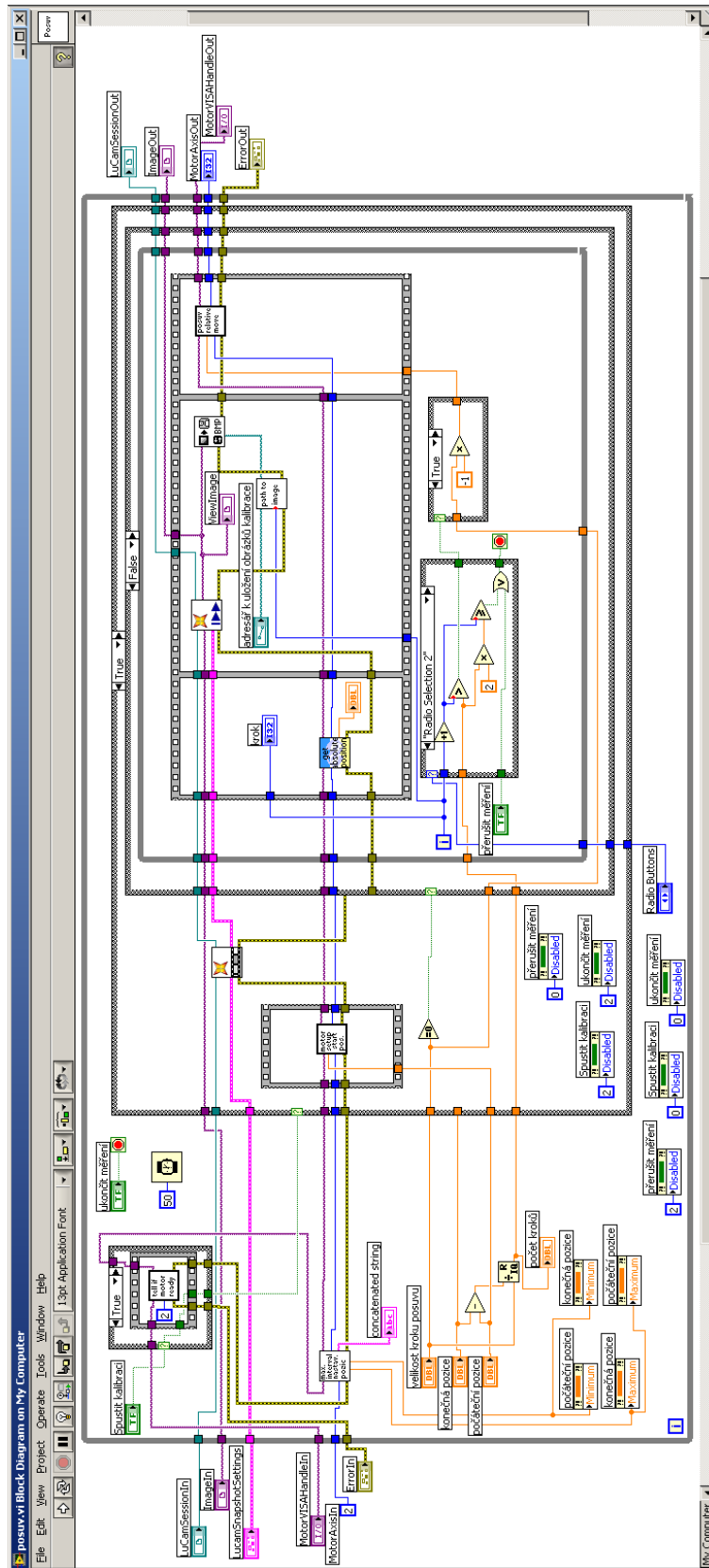
5.4.2 Kalibrace sestavy

Část programu „Kalibrace sestavy“ je momentálně určena pro posuv lineárního krokového motoru, který posouvá předmětem vpřed a vzad vůči poloze světelného zdroje. Tímto způsobem se získá sada referenčních rovin, které se vyhodnotí a získaná data se použijí pro přepočítání jednotek pixelů [pix] na délkovou jednotku v milimetrech [mm] u výsledných dat získaných z vlastního měření příslušné jamky kyčelního implantátu.

Po otevření okna si program zjistí interval hodnot, ve kterých je možné posouvat s předmětem. Uživatel pouze zadá počáteční a konečnou pozici posuvu, velikost kroku posuvu, místo uložení získaných snímků a zda-li se má posuv provést v jednom či obou směrech.



Obrázek 50: Kalibrace sestavy - přední panel. Slouží k nastavení parametrů kalibrace a následného získání kalibračních dat.

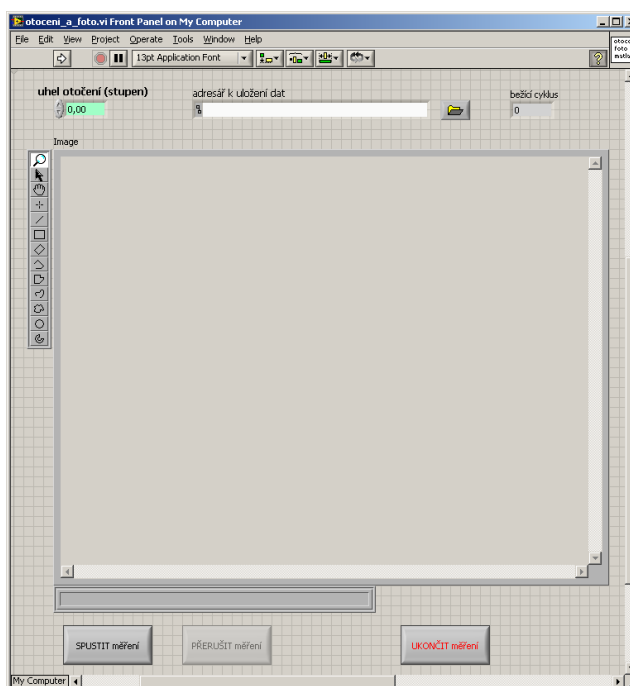


Obrázek 51: Kalibrace sestavy - blokový diagram. Programový kód, který obsluhuje a řídí proces kalibrace.

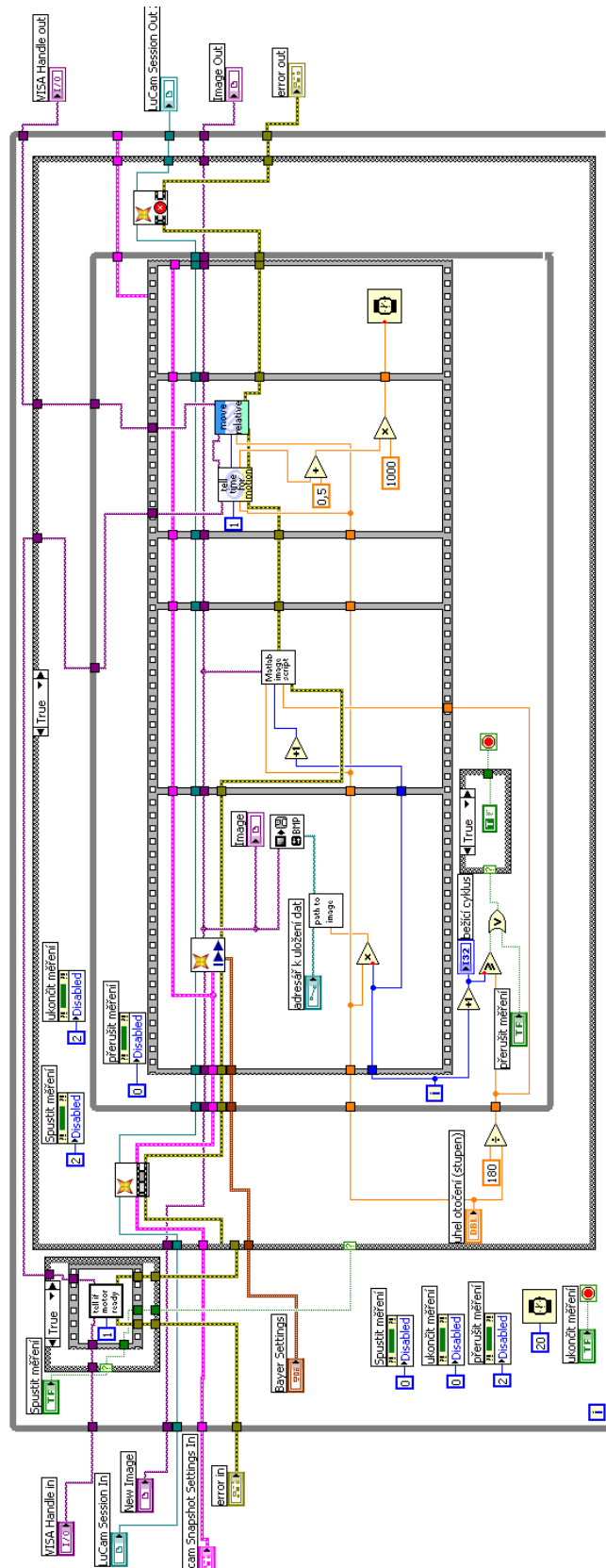
5.4.3 Vlastní měření

Po stisku tlačítka „Vlastní měření“ v hlavní nabídce programu se otevře okno (Obr. 52), které slouží k vlastnímu měření příslušného předmětu. Pro spuštění měření stačí nastavit velikost úhlu pootočení měřeného vzorku a místo, kam se mají získané obrázky z měření uložit. Poté již stačí spustit měření stiskem příslušného tlačítka.

V blokovém diagramu (Obr. 53) je cyklus, který obstarává vlastní měření, rozdělen na tři části. Na začátku každého cyklu se digitální kamerou vyfotí zkoumaný předmět a obraz se uloží, po uložení obrazu následuje pootočení krokového motoru o předem zadaný úhel. Třetí část má na starosti zpracování získaného obrázku. Je zavolán příslušný MATLAB skript, který provede požadované výpočty. Tyto tři úkony se opakují dokud se nezíská stanovený počet dat. Počet dat se odvíjí od velikosti krkoku pootočení a platí $n_{mereni} = 180[^\circ]/\omega[^\circ]$, kde n_{mereni} je počet měření na jednoho předmětu a ω je zadaný úhel pootočení. Probíha-li poslední cyklus, MATLAB skript kromě zpracování příslušného obrázku provede i zkompletování získaných dat a vykreslí 3D model předmětu. Použitý skript je zobrazen na Obr. 3.



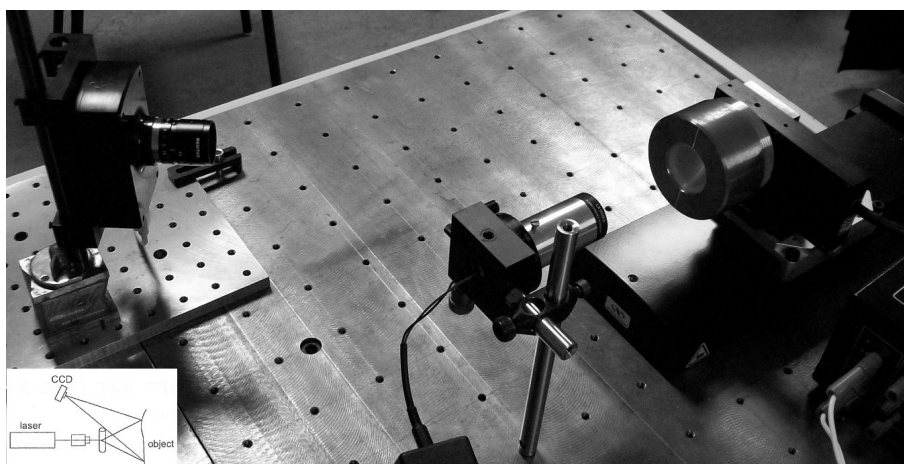
Obrázek 52: Vlastní měření - přední panel. Slouží k zadání informací pro vlastní měření vzorku.



Obrázek 53: Vlastní měření - blokový diagram. Programový kód, který obsluhuje a řídí proces měření vzorku.

6 Experiment

Sestavení experimentu je zobrazeno na Obr. 54. Experimentální uspořádání obsahuje zdroj světelného záření (LED dioda + válcová čočka), vzorek (jamka kyčelního implantátu) uchycený na rotačním krokovém motoru, posuvný lineární krokový motor, na kterém je umístěn rotační krokový motor se vzorkem, a digitální kamera.



Obrázek 54: Experimentální zapojení - digitální kamera, světelný zdroj, měřený vzorek uchycený v držáku rotačního krokového motoru (zleva doprava). Schéma experimentálního zapojení (ve výřezu).

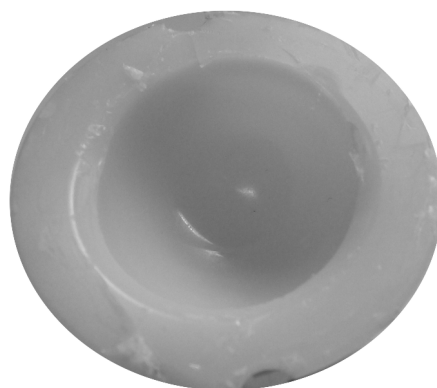
Vlastní experiment probíhal na dvou vzorcích kyčelních implantátů. Prvním vzorkem byla nová jamka kyčelního implantátu (viz Obr. 55). Druhým vzorkem byla již použitá jamka kyčelního implantátu (Obr. 56). Již pouhým srovnáním Obr. 55 a Obr. 56 je vidět silné opotřebení druhé jamky.

Postup měření byl v obou případech totožný. Bylo potřeba získat sadu snímků kyčelní jamky s projekční stopou. Každý snímek představoval polohu jamky po otočení o konstantní, předem daný, úhel. V těchto dvou případech byl volen úhel pootočení $\omega = 10^\circ$. Po získání obrázku se, pomocí MATLAB skriptu vloženého do programu v LabVIEW, provedla příslušná analýza. Na konci měření skript vyhodnotil získaná data a vykreslil příslušný 3D graf. Tato data se uložila pro případné další zpracování.

Pokud se před vlastním měřením provedla kalibrace, výsledné hodnoty byly přepočítány do délkových jednotek v [mm].



Obrázek 55: Jamka kyčelního implantátu - nová.

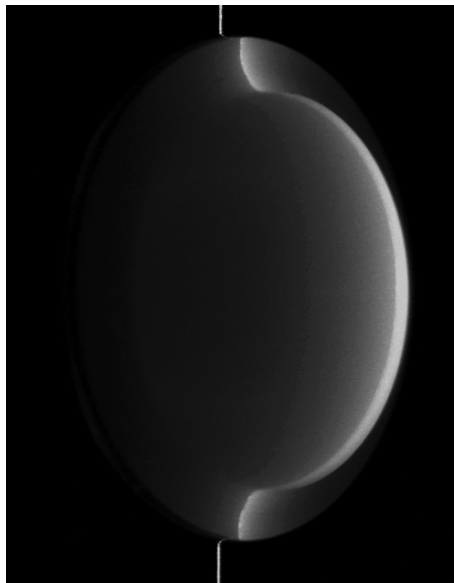


Obrázek 56: Jamka kyčelního implantátu - opotřebená.

6.1 Kyčelní implantát

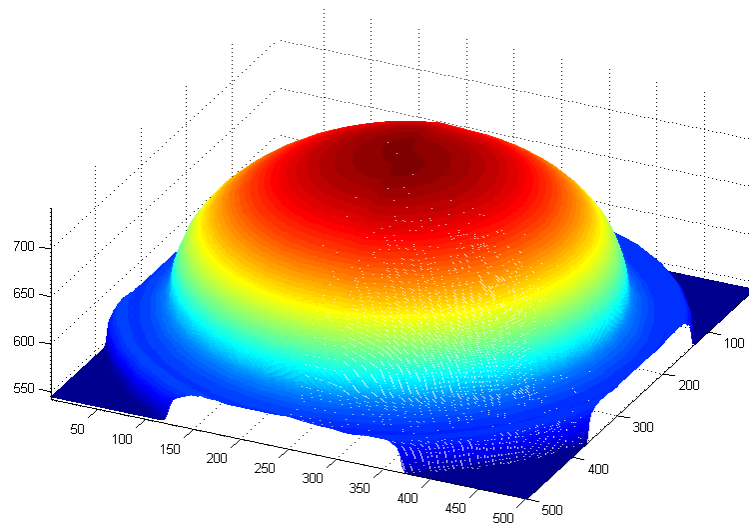
6.1.1 Nová jamka

Jako první se měřila nová jamka kyčelního implantátu. Jak je vidět z Obr. 55, jamka je sférického typu, půdorys jamky je kruhového tvaru. Jeden ze snímků získaných digitální kamerou při měření jamky je zobrazen na Obr. 57. Jak je vidět ze snímku, měření probíhalo za snížených světelných podmínek, abychom docílili výrazného rozdílu mezi intenzitou pozadí a intenzitou projekční stopy.

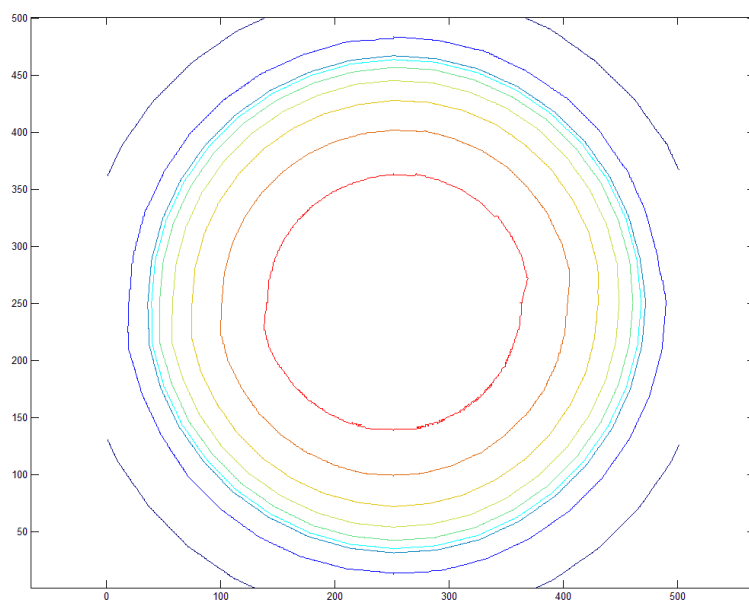


Obrázek 57: Snímek z měření - snímek světelné stopy kopírující povrch nové jamky kyčelního implantátu.

Po získání série snímků podobných Obr. 57, tj. na konci měření, MATLAB skript vypočítal a zároveň zobrazil 3D model měřené jamky (viz Obr. 58). 3D profil modelu odpovídá reálné situaci nové jamky. Aby byla lépe vidět symetrie nové jamky, byl vykreslen též vrstevnicový graf (tzv. Contour Plot), viz Obr. 59. Jednotlivé křivky v grafu spojují místa se stejnou topografickou výchytkou.



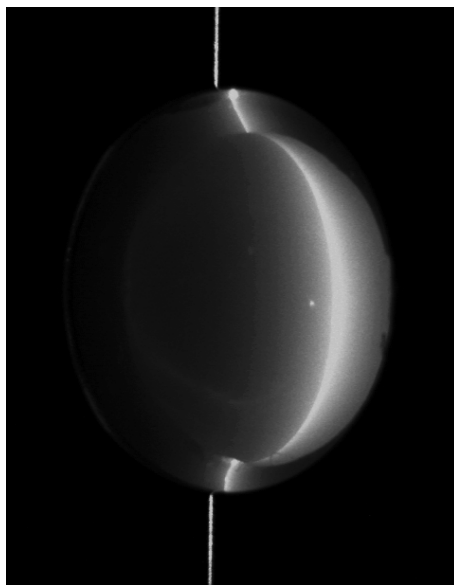
Obrázek 58: 3D profil měřené nové jamky kyčelního implantátu [pix].



Obrázek 59: Vrstevnicový graf [pix] (Contour Plot) - nová jamka kyčelního implantátu.

6.1.2 Vzorek použité jamky

Ve druhé části experimentu byla proměřována použitá jamka kyčelního implantátu (viz. Obr. 56). Již při prvním prozkoumání pouhým pohledem je vidět značné opotřebení jamky. Tento fakt je pozorovatelný i na výsledném 3D modelu jamky na Obr. 61, kdy se sférický tvar původní jamky mění na část elipsoidu.

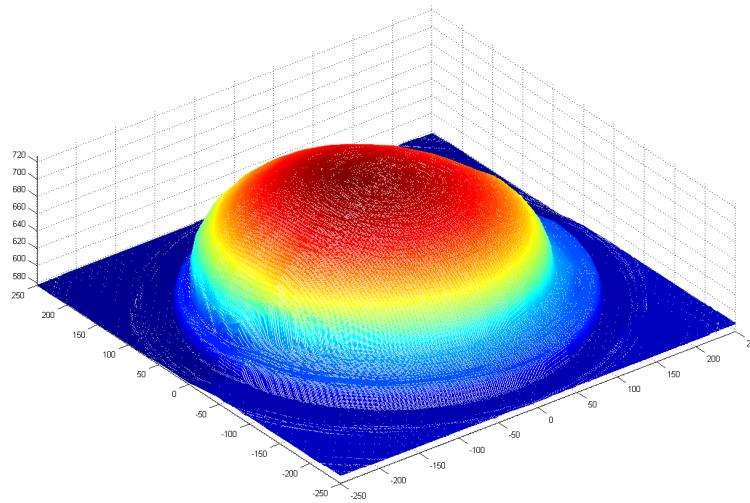


Obrázek 60: Snímek z měření - snímek světelné stopy kopírující povrch opotřeбенé jamky kyčelního implantátu.

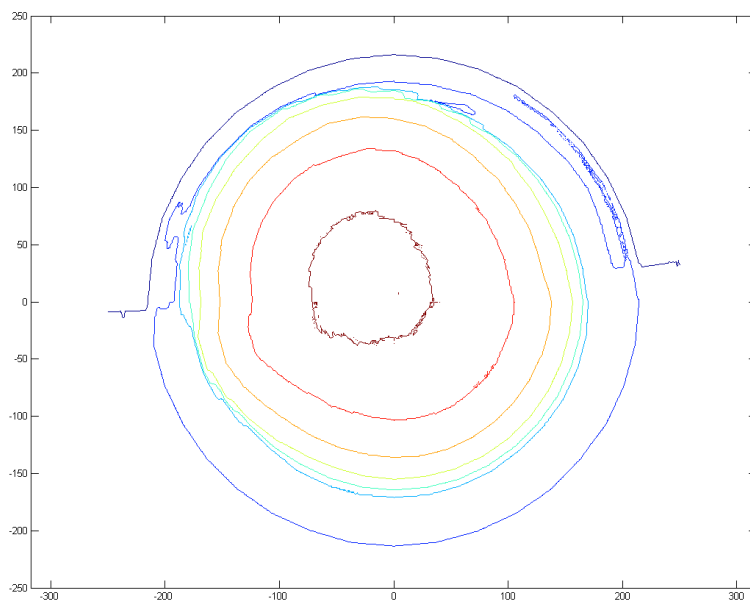
Ještě více je tato změna vidět na vrstevnicovém grafu na Obr. 62, kde se původně kruhové vrstevnice začínají protahovat a zaujímat eliptický tvar.

Data získaná z měření opotřebovaných jamek kyčelních implantátů jsou uložena a následně podrobena dalším výpočtům a vyhodnocením již podle konkrétních potřeb. Náplní této práce není popis dalšího zpracování a vyhodnocení naměřených dat. Tato problematika už je kapitolou sama pro sebe a hodně záleží na požadavcích, které jsou kladeny na dodatečné zpracování. Kapitola **Experiment** měla pouze ukázat získání dat z měření, jejich prvotní zpracování do podoby, ve které jsou připravena pro další použití, a jednoduchou vizualizaci získaných dat. K vizualizaci dat se použil jak 3D model měřeného předmětu, tak i vrstevnicový graf, který nám velice názorně a přehledně zobrazuje místa se stejnou topografickou výšilkou měřeného předmětu.

Aby bylo možné porovnat data získaná z opotřebované jamky kyčelního implantátu, byla jako referenční vzorek zvolena nová kyčelní jamka. Ke zvýraznění



Obrázek 61: 3D profil měřené opotřebené jamky kyčelního implantátu [pix].



Obrázek 62: Vrstevnicový graf [pix] (Contour Plot) - opotřebená jamka kyčelního implantátu.

rozdílů byla vybrána jedna z více opotřebovaných jamek. Rychlost a velikost opotřebení (tj. celková životnost) jamky kyčelního implantátu závisí na celé řadě faktorů. Mezi tyto faktory lze zahrnout věk, pohlaví, hmotnost, tělesnou aktivitu pacienta atd.

7 Závěr

Hlavním cílem diplomové práce bylo vytvořit software, který by z větší části zautomatizoval měření opotřebenění jamek kyčelních implantátů. K tomuto účelu bylo jako vývojového prostředí zvoleno LabVIEW firmy National Instruments. Snímky jsou pořizovány digitální CMOS kamerou Lu120M firmy Lumenera Corporation, rotační posuv předmětu je řešen pomocí rotačního krokového motoru PR50CC firmy Newport, pro lineární posuv předmětu je využit lineární krokový motor ILS150CCL téže firmy a jako projektor stopy slouží LED dioda, jejíž světelný svazek je usměrněn válcovou čočkou.

Primární požadavek na software v podobě automatizace měřicího procesu při vlastním proměřování jamky se podařilo naplnit. Mimo vlastní měření předmětu lze software využít i pro základní kalibraci experimentálního sestavení, která spočívá v získání sady snímků uživatelsky definovaných referenčních rovin. Po zpracování této sady snímků jsou získány kalibrační konstanty a a b z Rovnice 2 resp. z Rovnice 3, které slouží k převodu jednotek [pix] na jednotky [mm] (Rovnice 4).

Program též umožňuje nastavit, příp. doladit hodnoty parametrů digitální kamery. Toto nastavení může být následně uloženo do souboru a při dalších měřeních znovu načteno.

Vývoj programu však nekončí, nadále se bude pracovat na jeho vylepšení a optimalizaci. Později přibudou další funkce pro podrobnější zpracování dat a uchování výsledků v podobě databáze, jejich zpětného vyčtení z databáze, příp. exportu do různých datových formátů, apod. Též bude rozšířena sekce kalibrace o další funkce a možnost provedení komplexnější kalibrace experimentálního sestavení.

Jako ukázka funkce vytvořeného programu slouží Sekce 6 - Experiment. V této části práce jsou ukázány výsledky měření na dvou vzorcích jamek kyčelního implantátu. Prvním vzorkem je nová jamka (viz Obr. 55), jako druhý vzorek slouží již opotřebená jamka pacienta (Obr. 56). Výsledky jsou prezentovány pomocí 3D grafů (Obr. 58 resp. Obr. 61) znázorňující 3D profil zkoumaných jamek a 2D vrstevnicových grafů (Obr. 59 resp. Obr. 62), které názorně podávají informace o topografii jednotlivých jamek.

K vyhodnocení celkového poškození a opotřebenění jamky kyčelního implantátu je použita optická 3D skenovací profilometrie. Výhodou této metody je, že se jedná o bezkontaktní metodu, která nepoškozuje povrch zkoumaného

předmětu. Předmět tak může být podroben dalšímu zkoumání. Zároveň tato metoda poskytuje dostatečně přesné výsledky měření.

8 Přílohy

8.1 XML soubor s nastavením parametrů digitální kamery

Nastavené hodnoty parametrů digitální kamery se ukládají do textového souboru ve formátu XML. Obsah XML souboru je zobrazen níže. Struktura XML souboru je vygenerována přímo prostředím LabVIEW. XML soubor začíná elementem definujícím základní vlastnosti souboru

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
```

a je následován kořenovým elementem. V tomto případě je to element `<LVData></LVData>`. Element `<Version></Version>` nese informaci o použité verzi prostředí LabVIEW.

Každá proměnná je charakterizována svým datovým typem, názvem a hodnotou, případně seznamem možných hodnot. Proměnná je obalena elementem datového typu (např. `<SGL></SGL>`, `<U32></U32>`, apod.). Uvnitř tohoto elementu jsou pak informace o názvu (`<Name></Name>`) a hodnotě (`<Val></Val>`), příp. seznamu možných hodnot (`<Choice></Choice>`).

Pokud jsou některé parametry (názvy a proměnné) umístěny v „klastru“ („klastr“ spojuje více proměnných, polí, objektů či jiných „klastřů“ do jednoho objektu), je tato informace zanesena i do souboru XML použitím elementu `<Cluster></Cluster>`. `<NumElts></NumElts>` nese informaci o počtu elementů v klastru. Uvnitř elementu `<Cluster></Cluster>` se potom nacházejí příslušné proměnné.

Výpis XML souboru s informacemi o nastavení parametrů digitální kamery:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<LVData>
  <Version>8.2</Version>
  <Cluster>
    <Name>Lucam Snapshot Settings In</Name>
    <NumElts>17</NumElts>
    <SGL>
      <Name>Exposure</Name>
```

```
<Val>17.00000</Val>
</SGL>
<SGL>
  <Name>Gain</Name>
  <Val>1.00000</Val>
</SGL>
<SGL>
  <Name>Gain Red/Mag</Name>
  <Val>1.00000</Val>
</SGL>
<SGL>
  <Name>Gain Blue/Cyan</Name>
  <Val>1.00000</Val>
</SGL>
<SGL>
  <Name>Gain Grn1/Yel1</Name>
  <Val>1.00000</Val>
</SGL>
<SGL>
  <Name>Gain Grn2/Yel2</Name>
  <Val>1.00000</Val>
</SGL>
<U32>
  <Name>Strobe Flags</Name>
  <Val>0</Val>
</U32>
<SGL>
  <Name>Strobe Delay</Name>
  <Val>0.00000</Val>
</SGL>
<EL>
  <Name>HW Trigger?</Name>
  <Choice>Off</Choice>
  <Choice>On</Choice>
  <Val>0</Val>
</EL>
```

```

<SGL>
<Name>timeout</Name>
<Val>1000.00000</Val>
</SGL>
  <Cluster>
    <Name>Frame Format</Name>
    <NumElts>9</NumElts>
    <U32>
      <Name>xOffset</Name>
      <Val>0</Val>
    </U32>
    <U32>
      <Name>yOffset</Name>
      <Val>0</Val>
    </U32>
    <U32>
      <Name>width</Name>
      <Val>1280</Val>
    </U32>
    <U32>
      <Name>height</Name>
      <Val>1024</Val>
    </U32>
    <EL>
      <Name>pixelFormat</Name>
      <Choice>8 bit</Choice>
      <Choice>16 bit</Choice>
      <Choice>24 bit RGB</Choice>
      <Choice>YUV 422</Choice>
      <Choice>Count</Choice>
      <Choice>Filter</Choice>
      <Choice>32 Bit RGB</Choice>
      <Choice>48 Bit RGB</Choice>
      <Val>0</Val>
    </EL>
  <U16>

```

```

        <Name>subSampleX/binningX</Name>
        <Val>1</Val>
    </U16>
    <U16>
        <Name>flagsX</Name>
        <Val>0</Val>
    </U16>
    <U16>
        <Name>subSampleY/binningY</Name>
        <Val>1</Val>
    </U16>
    <U16>
        <Name>flagsY</Name>
        <Val>0</Val>
    </U16>
</Cluster>
<U32>
    <Name>Shutter Type</Name>
    <Val>0</Val>
</U32>
<SGL>
    <Name>Exposure Delay</Name>
    <Val>0.00000</Val>
</SGL>
<EW>
    <Name>Buffer Last Frame?</Name>
    <Choice>Off</Choice>
    <Choice>On</Choice>
    <Val>0</Val>
</EW>
<EL>
    <Name>ulReserved2</Name>
    <Choice>Do Not Change</Choice>
    <Val>0</Val>
</EL>
<SGL>

```



```

        <Name>flReserved1</Name>
        <Val>0.00000</Val>
    </SGL>
    <SGL>
        <Name>flReserved2</Name>
        <Val>0.00000</Val>
    </SGL>
</Cluster>
</LVData>

```

8.2 MATLAB skript

Níže je uveden výpis zjednodušené verze MATLAB skriptu určeného pro zpracování snímků a vyhodnocení 3D profilu měřené jamky. Proměnná sx určuje střed jamky v ose x , j je j -tý snímek měření, n je celkový počet snímků z měření, $fft()$ je funkce rychlé Fourierovy transformace. Tato funkce je použita právě k analýze projekční stopy, $x(j, i)$, $y(j, i)$, $z(j, i)$ jsou souřadnice daného bodu v prostoru. Výsledný 3D profil předmětu je vykreslen pomocí funkce $mesh()$.

```

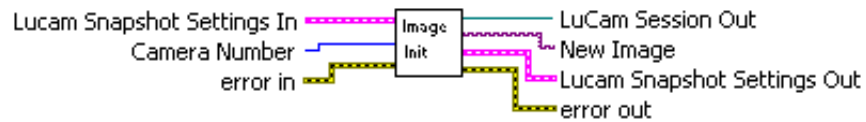
sx = 449;
for i =1:1024
    b=fft(a(i,:));b(1:1260)=0;
    b=abs(ifft(b));[m,p]=max(b);c(i) = p;
    y(j,i) = sin(deg2rad(j*uh))*(i-sx);
    x(j,i) = sin (deg2rad(90-j*uh))*(i-sx);
    z(j,i) = c(i);
end
if j==n
    [X,Y] = meshgrid(-250:1:250, -250:1:250);
    Z = griddata(x,y,z,X,Y);
    mesh (X,Y,Z);
end

```

VI moduly vytvořené pro program

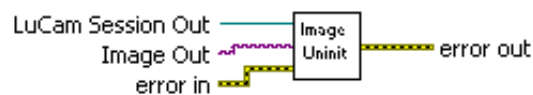
Inicializace

Modul Image_init.vi (Obr. 63) je určen k inicializaci práce s digitální kamerou a obrázky.



Obrázek 63: Image Init.

Opačnou funkci k Image_init.vi zastává modul Image_Uninit.vi (Obr. 63), který řádně ukončuje práci s digitální kamerou a obrázky.



Obrázek 64: Image UnInit.

Modul motor_Port_Init.vi (Obr. 65) otevírá komunikační port, který je určen ke komunikaci programu s řídicí jednotkou krokového motoru.



Obrázek 65: Motor Port Init.

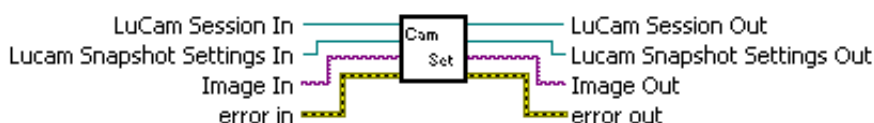
K uzavření portu slouží modul motor_Port_UnInit.vi (Obr. 66).



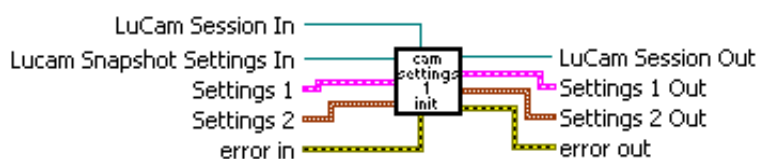
Obrázek 66: Motor Port UnInit.

Nastavení kamery

Celý modul ovládající nastavení kamery je Cam_Settings.vi (viz Obr. 67). Tento „vi“ modul využívá dalších modulů, jako Cam_Settings_1_init.vi (Obr. 68), Cam_Settings_2_init.vi (Obr. 69).



Obrázek 67: Camera settings.

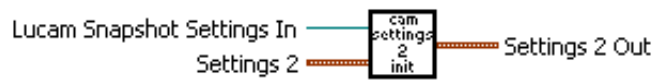


Obrázek 68: Camera settings 1 init.

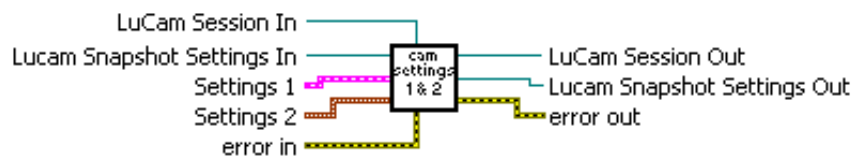
Modul Cam_Settings_1_and_2.vi (Obr. 70) slouží k převodu jednotlivých parametrů nastavení kamery do jediného formátu kompatibilního s formátem využívaným moduly od firmy Lumenera.

Modul Read_Lucam_Snapshot_Settings.vi (Obr. 71) slouží k načtení parametrů nastavení kamery z uloženého XML souboru. V případě problémů s načtením souboru bude použito standardní nastavení.

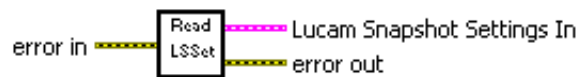
Naopak modul Write_Lucam_Snapshot_Settings.vi (Obr. 72) je určen k uložení parametrů nastavení kamery.



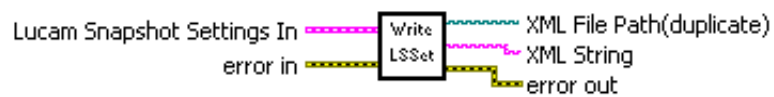
Obrázek 69: Camera settings 2 init.



Obrázek 70: Camera settings 1 and 2.



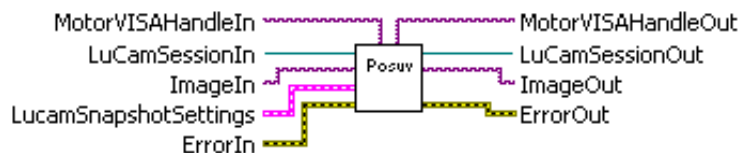
Obrázek 71: Read Lucam Snapshot Settings.



Obrázek 72: Write Lucam Snapshot Settings.

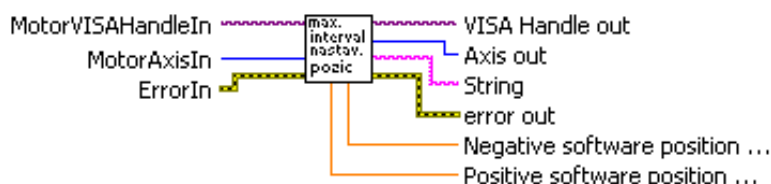
Kalibrace systému

Hlavním modulem pro kalibraci systému je posuv.vi (Obr. 73), který řídí proces kalibrace.



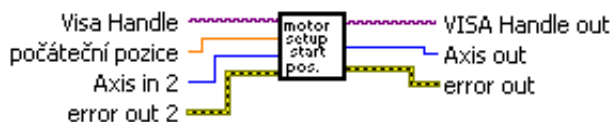
Obrázek 73: Modul Posuv.vi

Modul max_interval_pro_nastaveni_pozic.vi (Obr. 74) slouží ke zjištění maximálního intervalu pozic, do kterých lze přesunout krokový motor. Zjištěné hodnoty vrací jako informaci v textové podobě.



Obrázek 74: Maximální interval nastavení pozic.

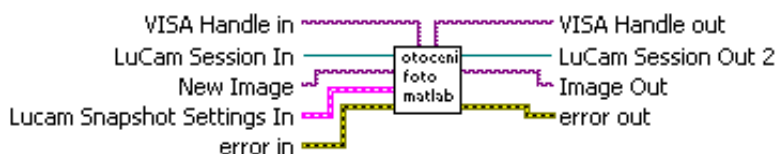
Modul posuv_SetUp_Start_Pos.vi (Obr. 75) nastaví krokový motor do uživatelem zadané pozice, která je výchozím bodem kalibrace.



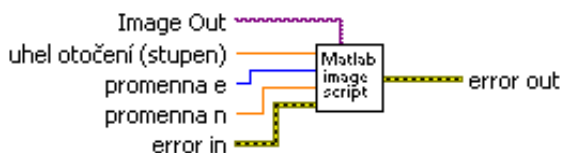
Obrázek 75: Nastavení počáteční pozice pro kalibraci.

Vlastní měření

Modul `otoceni_a_foto.vi` (Obr. 76) obstarává vlastní měření zkoumaného předmětu. Po zadání počátečních údajů provede měření a prvotní vyhodnocení získaných dat. Data jsou zpracovávána vloženým MATLAB skriptem do prostředí LabVIEW. Tuto část má na starosti modul `zpracovani_obrazku_Matlab.vi` (Obr. 77).



Obrázek 76: Modul pro vlastní měření předmětu.



Obrázek 77: „Matlab image script“ pro zpracování získaných obrázků.

8.3 CD-ROM

Na cd-rom se nachází zdrojové kódy programu, ovladače a pluginy digitální kamery a krokových motorů pro vývojové prostředí LabVIEW .

Použitá literatura

- [1] Mlčoch, J., Rössler, T. *Úvod do fyzikálního měření*. Olomouc: Univerzita Palackého, 2005.
- [2] Nožka L., Mandát D., Hrabovský M. *The 3D Optical Scanning Topography, Theory And Application*. Acta Univ. Palacki. Olomuc., Fac. Rer. Nat. (2003-2004), Physica 42-43, p. 185-194
- [3] Mandát D., Rössler T., Hrabovský M., Gallo J. *Aplikace optických topografických metod v medicíně*. Acta Mechanica Slovaca, Košice 2006. s. 327 - 332.
- [4] *LabVIEW* [online]. [cit. 2009-05-06].
URL: <http://en.wikipedia.org/wiki/Labview>
- [5] Mayer T., Porganszki E., Stewart R. *Grafické programovací prostředí LabVIEW pomohlo zkrátit čas vývoje z roků na měsíce* [online]. [cit. 2009-05-06].
URL: http://digital.ni.com/worldwide/bwcontent.nsf/web/all/D5E5360894D965358625757D003EBAE3?OpenDocument&node=35413_cs
- [6] *Matlab* [online]. [cit. 2009-05-06].
URL: <http://cs.wikipedia.org/wiki/MATLAB>
- [7] *Stepper Motor* [online]. [cit. 2009-02-24].
URL: http://en.wikipedia.org/wiki/Stepper_motor
- [8] Industrial Circuits Application Note, *Stepper Motor Basics* [online]. [cit. 2009-02-26].
URL: <http://www.solarbotics.net/library/pdflib/pdf/motorbas.pdf>
- [9] *Pohony s krokovými motorky* [online]. [cit. 2009-04-10].
URL: http://fei1.vsb.cz/kat453/www453/soubory/texty/ucebni_texty/se/cast_C_el_pohony/se_eph_c1_krokac_02_teorie.pdf
- [10] *PR50 & SR50 Series Compact Rotation Stages* [online]. [cit. 2009-04-11].
URL: <http://www.newport.com/store/genproduct.aspx?id=582178&lang=1033&Section=Spec>

- [11] *PR50 rotation stage, DC motor* [online]. [cit. 2009-04-11].
URL: <http://search.newport.com/?sku=PR50CC>
- [12] *ILS Series High-Performance Mid-Range Travel Linear Stages* [online]. [cit. 2009-04-11]. URL: <http://www.newport.com/ILS-Series-High-Performance-Mid-Range-Travel-Line/140110/1033/catalog.aspx?Section=Spec>
- [13] *ILS150CCL linear stage, DC motor* [online]. [cit. 2009-04-11].
URL: <http://search.newport.com/?sku=ILS150CCL>
- [14] *SMC100 Single-Axis Motor Controller/Driver - Product Detail* [online]. [cit. 2009-04-11].
URL: <http://www.newport.com/store/genproduct.aspx?id=400968&lang=1033&Section=Detail#>
- [15] *PR50 rotation stage, DC motor* [online]. [cit. 2009-04-11].
URL: <http://search.newport.com/?sku=SMC100CC>
- [16] *Lu120* [online]. [cit. 2009-04-12].
URL: <http://www.lumenera.com/industrial/lu120.php>
- [17] *Lu120 industrial camera* [online]. [cit. 2009-04-12].
URL: <http://www.lumenera.com/images/gallery/industrial/products/Industrial-Board-1.jpg>
- [18] ZMEŠKAL Oldřich, BUCHNÍČEK Miroslav, SEDLÁK Ondřej. *Principy digitálního záznamu obrazu* [online]. Vysoké učení technické v Brně, Brno 2001. [cit. 2009-04-12].
URL: http://www.fch.vutbr.cz/lectures/imagesci/download/stud02_hrakra01.pdf
- [19] *Obrazové problémy digitální fotografie I.- Senzor* [online]. [cit. 2009-04-12]. URL: http://www.fotografovani.cz/art/fotech_df/rom_trouble1.html
- [20] *Download centre of Lumenera Corporation* [online]. [cit. 2009-04-22].
URL: <http://www.lumenera.com/support/download.php>
- [21] Brzobohatý J., Musil V., Bajer A., Boušek J., Prokop R. *Elektronické součástky*. Fakulta elektrotechniky a komunikačních technologií VUT v Brně, Brno 2002.

- [22] *Přechod P-N* [online]. [cit. 2009-04-28]. URL: http://cs.wikipedia.org/wiki/P%C5%99echod_P-N
- [23] Kubínek, R. *Elektrina a magnetismus* [online]. [cit. 2009-04-28]. URL: <http://apfyz.upol.cz/ucebnice/down/elmag.pdf>
- [24] *Single-Axis Motion Controller/Driver for DC or Stepper Motor*. User's manual for SMC100CC & SMC100PP, Newport Corp. 2008