



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION



DEPARTMENT OF BIOMEDICAL ENGINEERING

# SYSTÉM ŘÍZENÍ NÁSTROJE ROBOTICKÉ RUKY Z OPERAČNÍHO ROBOTA DA VINCI V LABVIEW

CONTROLLER OF THE DA VINCI ROBOTIC ARM INSTRUMENT USING LABVIEW

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

Hana Vítová

AUTHOR

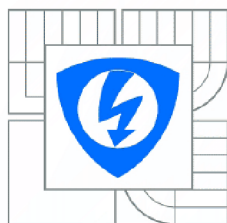
VEDOUCÍ PRÁCE

Ing. Jiří Sekora

SUPERVISOR

BRNO, 2010





VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav biomedicínského inženýrství

# Bakalářská práce

bakalářský studijní obor

Biomedicínská technika a bioinformatika

**Studentka:** Hana Vítová

**ID:** 101799

**Ročník:** 3

**Akademický rok:** 2009/2010

## NÁZEV TÉMATU:

**System řízení nástroje robotické ruky z operačního robota Da Vinci v LabVIEW**

## POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s funkcí robotické ruky operačního robota Da Vinci a navrhňte program v LabVIEW s ohledem na zjištěné stupně volnosti. Navrhňte výstupní komunikační protokol pro komunikaci se servomotory řízenými mikrokontrolérem. Zvažte i možné zabezpečení protokolu.

V prostředí LabVIEW vytvořte virtuální stroj pro řízení nástroje robotické ruky pomocí jednoduchého uživatelského prostředí. Program bude využívat standardní výstup počítače pro komunikaci se servomotory pomocí navrženého protokolu, řízení nástroje realizujete pomocí game padu.

## DOPORUČENÁ LITERATURA:

[1] Servo motor controller using labVIEW [online]. Dostupný z <<http://forums.ni.com/>>.

[2] Intuitive Surgical, The da Vinci Surgical System [online]. Dostupný z <<http://www.intuitivesurgical.com/>>

**Termín zadání:** 8.2.2010

**Termín odevzdání:** 31.5.2010

**Vedoucí práce:** Ing. Jiří Sekora

**prof. Ing. Ivo Provazník, Ph.D.**

*Předseda oborové rady*

## UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **Abstrakt:**

Práce popisuje tvorbu ovládacího rozhraní pro nástroj ruky operačního robota Da Vinci firmy Intuitive Surgical. Prvotním cílem je základní seznámení s funkcemi operačního robota, kterým je věnována první část. V dalších částech následuje návrh komunikace, jejímž základem je ergonomické ovládací zařízení - gamepad, které pomocí páček umožňuje kontrolu nad dvěma nezávislými čepelemi na operačním nástroji. Přenos pohybu hlavy nástroje ruky zajišťuje program sestavený v LabVIEW. Vyhodnocuje vstupní data z ovladače a na základě předem definovaných mezí odesílá bitové instrukce. Komunikace je sestavena v závěrečné části programu v LabVIEW a data jsou prostřednictvím sériového protokolu odeslána na cílové zařízení. Program poskytuje také omezené možnosti nastavení pro snadnou přenositelnost na jiné PC a modulovatelnost. Díky uživatelskému prostředí má uživatel plnou vizuální kontrolu nad vstupními hodnotami. V návrhu rozhraní je počítáno s reálnými možnostmi ovládní a mechanického návrhu supportu za pomoci krokových motorků.

## **Abstract:**

Bachelor work describes creation of interface for surgical robotic hand Da Vinci made by Intuitive Surgical. First goal was to get familiarized with its functions. This is the contents of the first few chapters. The rest of the work focuses on communication design between computer and Da Vinci arm instrument. On the user side, there is a controlling device connected, a gamepad with levers, which allows user to move two independent blades on the instrument. Transmission of this movement to the robotic arm instrument is task of a software programmed in LabVIEW. It evaluates entry data from the gamepad and after comparing them to predefined bounds, it sends bit instructions to Da Vinci. The communication is opened in the last part of the software solution and the data (2 bytes) are sent through serial protocol to target device. Program provides also a limited options, which can be set by the user. These settings are primarily used for easy portability between computers. Via graphical interface has user complete visual control over input values. The whole design is based on the real possibilities of arm instrument manoeuvrability and mechanical support design using stepping motors.

## **Klíčová slova:**

Da Vinci, robot, Intuitive Surgical, LabVIEW, komunikační protokol, sériový port

## **Key words:**

Da Vinci, robot, Intuitive Surgical, LabVIEW, communication protokol, serial port



VÍTOVÁ, H. *Systém řízení nástroje robotické ruky z operačního robota Da Vinci v LabVIEW*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2010. 42 s. Vedoucí bakalářské práce Ing. Jiří Sekora.

## Prohlášení

Prohlašuji, že svou bakalářskou práci na téma Systém řízení nástroje robotické ruky z operačního robota Da Vinci v LabVIEW jsem vypracovala samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autorka uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením tohoto projektu jsem neporušila autorská práva třetích osob, zejména jsem nezasáhla nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědoma následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne 31. května 2010

.....  
podpis autora

## Poděkování

Děkuji panu Ing. Jiřímu Sekorovi za pomoc, připomínky, nadšený přístup, vstřícnost, ochotu a odborné rady při zpracování bakalářské práce.

V Brně dne 31. května 2010

.....  
podpis autora

# Obsah:

Seznam obrázků .....	8
Úvod .....	9
1. Historie operačních robotů.....	11
1.1 Operační robot Da Vinci v České republice .....	12
1.2 Popis operačního robota Da Vinci .....	13
1.3 Určování stupňů volnosti .....	14
2. Nástroj ruky robota Da Vinci.....	16
2.1 Anatomie nástroje ruky robota Da Vinci.....	16
2.2 Možnosti zakončení nástroje ruky .....	17
2.3 Pohyby nástroje ruky.....	18
3. LabVIEW .....	19
3.1 Program – blokové schéma .....	19
3.2 Uživatelské prostředí.....	20
3.3 Nastavení a parametry VISA.....	22
3.4 Kontrola systémových požadavků.....	24
3.5 Softwarové řešení v LabVIEW .....	24
3.6 Návrh komunikačního protokolu .....	27
4. Návrh ovládacího zařízení .....	30
4.1 Popis gamepadu .....	30
4.2 Ovládání nástroje ruky pomocí gamepadu .....	31
4.3 Krokové motorky .....	31
4.4 Mechanické ovládání.....	33
5. Závěr.....	34
Použitá literatura.....	35
Seznam zkratk .....	37
Seznam příloh.....	38

## Seznam obrázků

Obrázek 1 První cystoskop sestrojený Phillipem Bozzinim .....	11
Obrázek 2 Sestava Da Vinci konzole, operační robot, řídicí jednotka (zleva) .....	13
Obrázek 3 Ovládání nástroje ruky v chirurgově konzole .....	14
Obrázek 4 Popis nástroje ruky.....	16
Obrázek 5 Varianty zakončení.....	17
Obrázek 6 Pohyby instrumentu.....	18
Obrázek 7 Blokové schéma ovládacího programu v LabVIEW.....	20
Obrázek 8 Záložka základního ovládání uživatelského prostředí.....	21
Obrázek 9 Záložka pokročilého nastavení uživatelského prostředí .....	22
Obrázek 10 Řazení bitů.....	26
Obrázek 11 VISA bez bloku pro hlášení chyb .....	26
Obrázek 12 Nastavení mezí v programu pro určení směru (rotace z a osa z) .....	27
Obrázek 13 Návrh komunikačního protokolu .....	29
Obrázek 14 XOR.....	29
Obrázek 15 Logitech Gamepad Dual Action s popisem tlačítek .....	30
Obrázek 16 Charakteristiky krokových motorů .....	32
Obrázek 17 Technický výkres motorů .....	33

# Úvod

*„Mladý Rossum, slečno, to byl nový věk. Věk výroby po věku poznání. Když si okoukl anatomii člověka, viděl hned, že je to příliš složité a že by to dobrý inženýr udělal jednodušeji... člověk, to je něco, co, dejme tomu, cítí radost, hraje na housle, chce jít na procházku a vůbec potřebuje dělat spoustu věcí – které jsou vlastně zbytečné... Mladý Rossum vynalezl dělníka s nejmenším počtem potřeb. Musel ho zjednodušit. Vyhodil všechno, co neslouží přímo práci. Tím vlastně vyhodil člověka a udělal Robota... Roboti nejsou lidé. Jsou mechanicky dokonalejší než my, mají úžasnou rozumovou inteligenci, ale nemají duši... Výrobek inženýra je technicky vytríbenější než výrobek přírody.“ \**

Karel Čapek ve svém dramatu R.U.R. vydaném v roce 1920 poprvé použil slovo robot. Ve skutečnosti toto pojmenování vymyslel jeho bratr Josef jako odvozeninu od starého českého výrazu robota, která označovala povinnou práci sedláků a rolníků na panství. Slovo robot se dnes používá běžně ve všech světových jazycích. Každým dnem dostává nový a nový význam, protože pokrok v tomto oboru je jedním z nejrychlejších. Kolébkou robotiky se bohužel nestala Česká republika, ale je jí Země vycházejícího slunce – Japonsko, která udává závratné tempo ve vývoji robotů.

Roboti se stali základním kamenem pro oblíbený sci-fi žánr. Zde už se pomalu začali odlišovat roboti a roboty. Roboty byly pouhé stroje, které sloužily lidem, neoplývaly žádnou umělou inteligencí a usnadňovaly práci v továrnách, jako například různé manipulátory nebo podavače. Zatímco roboti jsou úplně jinou skupinou. Jejich cílem je co nejvíce napodobit člověka, ať už v jeho inteligenci nebo po vizuální stránce. Mají schopnosti učit se, komunikovat a dokonce v určitém směru nahrazovat člověka jako bytost - robotický kamarád pro děti či inteligentní sluha pro starší lidi.

Isaac Asimov, profesor biochemie, v roce 1942 v povídce Hra na honěnou, která je uvedena ve sbírce povídek Já, robot, stanovil jednoznačná pravidla, podle kterých se musí každý robot řídit. Tyto zákony byly později přijaty i odbornou veřejností pod názvem Zákony robotiky, ačkoliv se původně jednalo o beletristické dílo:

1. Robot nesmí ublížit člověku nebo svou nečinností dopustit, aby mu bylo ublíženo.
2. Robot musí poslechnout člověka, kromě případů, kdy je to v rozporu s prvním zákonem.
3. Robot se musí chránit před poškozením, kromě případů, kdy je to v rozporu s prvním nebo druhým zákonem. \*\*

Dnes se běžně setkáváme s roboty, aniž bychom si to příliš uvědomovali. Výrobní linky v továrnách tíhnou k absolutní automatizaci, dětské interaktivní hračky nás také v mnohém dokážou překvapit. Pomalu se začínají dostávat roboti i do domácnosti, i když momentálně jsou výsadou několika stovek bohatých lidí. Je ale otázkou času, kdy se dostanou mezi obyčejnou populaci a budou se nám snažit ulehčovat práci.

Samostatnou kapitolou je použití robotiky ve vojenských organizacích nebo při dobývání vesmíru. Těžko si představit, jak by NASA bez robotů Spirit a Opportunity dobývala Mars. Naštěstí se celá robotizace společnosti nevyhnula ani tak důležitému odvětví, jakým

---

\* ČAPEK, Karel. *R.U.R.*, Praha: Nakladatelství ARTUR, 2004. 98 s., ISBN 80-86216-46-2 cit. 14-15s

\*\* ASIMOV, Isaac. *Já, robot* 2. vydání, Praha: Triton, 2004. 323 s., ISBN 80-7254-560-4

je lékařství. Dnes se již běžně používají roboty na operačních sálech. Jsou to zatím sice jen obrovské operační nástroje bez vlastní inteligence, ale díky nim chirurgické zákroky postoupily o obrovský skok kupředu. Minimálně invazivní operace přinášejí pacientovi rychlejší rekonvalescenci a méně pooperačních problémů. Samozřejmě se vývoj těchto strojů nezastavil, současným cílem je nahrazení chirurga samostatným robotem, který bude umět pohotově vyhodnotit situaci tak, jak to dosud umí jen člověk.

Americká společnost Intuitive Surgical vyrobila v roce 1997 operačního robota Da Vinci. Své jméno dostal po renesančním umělci Leonardu Da Vincim, člověku různorodých zájmů. Jednalo se o velkého inovátora, vědce, malíře, sochaře, historika a přírodovědce. Leonardo se narodil roku 1452 v Anchianu u Vinci, v poměrně chudých poměrech, což však nezabránilo tomu, aby se z něj stal jeden z nejznámějších lidí, kteří kdy žili. Leonardo se nebál přijít s něčím novým, vymýšlet šílené vynálezy, které se v průběhu několika dalších století staly skutečností.

# 1. Historie operačních robotů

Samostatným odvětvím robotiky se stalo používání robotů v lékařství. Zatím se ale jedná pouze o mechanické sestavy, které jsou plně obsluhované operujícím chirurgem. Řídí se z elektronického rozhraní, které ve většině případů obsahuje joysticky na ovládání pohybu a různé typy okulárů na vizuální dohled. Operativní výkony se staly minimálně invazivní pro pacienty, a tak po nich hrozí menší pooperační komplikace než u běžných operací. Chirurgické zákroky pomocí robotů jsou poměrně mladým odvětvím v lékařství, první robotická operace byla provedena až v roce 1997.

K prvním předchůdcům zobrazovacích video pomůcek patří cystoskop, který v roce 1806 sestrojil Phillip Bozzini. Jednalo se o tzv. Lichtleiter (viz obrázek 1), jehož základní součástí bylo kukátko se zrcadly, které osvětlovala zapálená svíčka. Hlavním cílem bylo dosáhnout vizuálního kontaktu oka doktora se sledovaným vnitřním orgánem. V průběhu 19. století se spousta výzkumníků včetně Antiona Desormeauxa (zv. otec endoskopie) snažila vylepšit systém cystoskopu zejména z hlediska nepřímého osvětlení a jeho konstrukce. Charles Preston na konci století vyvinul studenou nízkonapětovou lampu, která nepotřebovala chlazení. Od té doby se uvedený zdroj světla používal v běžné lékařské praxi.



*Obrázek 1 První cystoskop sestrojený Phillipem Bozzinim*  
[2]

Na začátku 20. století se, obvykle v gynekologii a postupem času i v chirurgii, začaly používat endoskopy. Laparoskopické operace se po vynálezu profukovacího přístroje začlenily mezi klasické chirurgické postupy. V roce 1966 se uskutečnila první gynekologická laparoskopická procedura, kterou povedl Kurt Semm. V 70. letech se pak laparoskopie začala používat v diagnostice. Po roce 1980 chirurgové pokročili v používání tzv. minimálně invazivních operací břicha. Počátkem 90. let se objevily tzv. VATS, zákroky prováděné s využitím videotechniky.

Vznik chirurgických robotů měl tři hlavní příčiny: operace na dálku, vojenskou medicínu a snahu o co nejméně invazivní výkony. V 70. a 80. letech byl prováděn výzkum americké armády a NASA, který vyústil ve vývoj systému Satava, umožňujícího operace bez nutné

přítomnosti chirurga na sále. Satava byl prvním prototypem robotického systému. Lékař seděl u počítače na druhé straně místnosti a ovládal operační nástroje. Původní záměr byl použít tohoto robota pro vojenské účely – léčení vojáků na bojišti kdekoli po celém světě. K jeho realizaci ale nedošlo.

S příchodem výpočetní techniky mezi běžnou populaci v 90. letech začalo být jasné, že počítače vyřeší omezení konvenčních video endoskopických (dále VE) technik. Současné VE techniky mají několik zásadních limitů: zobrazení ve 2D negativně ovlivňuje vizualizaci, nástroje jsou velmi dlouhé a mají omezenou pohyblivost. Proto se výzkumníci uchýlili k propojení robota s počítačem. První takto vylepšený operační robot nesl název RoboDoc a vytvořil ho tým Integrated Surgical Systems v Sacramentu v roce 1992. RoboDoc byl při operacích používán hlavně ortopedy a umožňoval vyvrtání přesné díry ve stehně. V roce 1994 byl představen v Santa Barbaře AESOP, robot týmu Computer Motion Inc., který umožnil přesnou kontrolu VE technik. Poskytoval stabilní pole vidění a byl řízen hlasovými pokyny chirurga. Na konci 90. let byly představeny dva robotické operační systémy. Robot Da Vinci (Intuitive Surgical, Sunnyvale) v roce 1997, kdy s jeho pomocí provedli v Bruselu první klinickou robotickou proceduru - cholecystektomii. První operace srdce a první bypass věnčité tepny s robotem Da Vinci následovaly v roce 1998. Poté byl v roce 2000 Da Vinci oficiálně schválen pro obecné operativní aplikace. Druhým, velmi podobným robotickým systémem, byl Zeus od společnosti Motion, Santa Barbara. Představen byl v roce 1998, ale v současnosti již není pro klinické použití dostupný.

Oba robotické operační systémy (Da Vinci, Zeus) jsou koncepčně podobné. Chirurgova konzole je připojena pomocí elektronického rozhraní k robotické ruce, která se používá k manipulaci endoskopu a chirurgických nástrojů. Zeus má otevřenou konzoli, díky které chirurg přímo vidí na místo operace. Robot Zeus se skládá ze tří oddělených rukou nezávisle upevněných na stole v operační místnosti a každá z nich má 5 stupňů volnosti. Da Vinci používá konzoli se dvěma nezávislými monitory, na kterých chirurg sleduje operaci. Da Vinci systém využívá tři robotické ruce se šesti stupni volnosti a 360 stupňovou rotací zápěstí nástroje. Obě sestavy umožňují chirurgovi kompletní kontrolu nad operačními nástroji a kamerou [6].

## **1.1 Operační robot Da Vinci v České republice**

Do České republiky se poprvé dostal robotický systém Da Vinci v roce 2005 [4] do Nemocnice Na Homolce, poměrně záhy po oficiálním uvedení na celosvětový trh. K roku 2009 je operační robot Da Vinci celkem v 6 nemocnicích, na 8 operačních sálech [15],[18]: v Praze (Nemocnici Na Homolce – celkem dva systémy, Ústřední vojenské nemocnici ve Střešovicích), Brně (Fakultní nemocnice u Svaté Anny), Ústí nad Labem (Masarykova nemocnice – celkem dva systémy), Novém Jičíně (nemocnice s poliklinikou), Mostišti u Velkého Meziříčí (nemocnice sv. Zdislavy) a Olomouci (nemocnice).

Da Vinci je používán nejvíce v oborech, jakými jsou urologie (radikální operace prostaty, partiální resekce ledviny, plastiky ledvinné pánvičky a nefrektomie), břišní chirurgie (resekce žaludku, střev, konečníku, jater), cévní chirurgie (rekonstrukce břišní aorty a pánevních tepen) a kardiochirurgie [14].

Největší výhodou zákroku prováděného robotem je minimální invazivnost operace (MIO) a velká šetrnost k pacientovi. Při MIO má pacient minimální krevní ztráty i potřebu léků. Nutná



doba hospitalizace následující po operaci se významně zkracuje a pacient je v brzké době schopen přejít do normálního života [18].

## 1.2 Popis operačního robota Da Vinci

Operační robot Da Vinci od společnosti Intuitive Surgical je v současné době jediný svého druhu (viz část 1. Historie operačních robotů). Celý robot má dvě hlavní části. První je kontrolní konzole, pomocí které chirurg ovládá druhou část robota – chirurgický vozík s nástroji (viz obrázek 2).

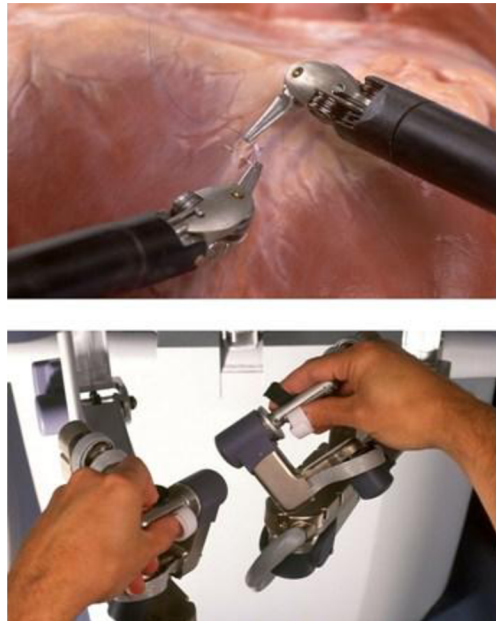


*Obrázek 2 Sestava Da Vinci konzole, operační robot, řídicí jednotka (zleva)*  
[8]

V chirurgově konzoli nalezneme hlavní ovládací páky (viz obrázek 3), které se chovají jako vysoce rozlišovací vstupní zařízení, jež dokážou přečíst přesnou pozici a orientaci. Nachází se zde zobrazovací systém, díky kterému má chirurg maximální kontrolu nad svými pohyby. Obraz z kamery nad pacientem je zobrazován na stereoskopický displej, jenž využívá CRT monitor, na který chirurg nahlíží přes binokulární brýle. Rameno pohybuje pomocí elektronického ovládacího systému. Celé toto zařízení je umístěno mimo sterilní oblast na operačním sále. Uživatelské prostředí se skládá z nášlapných pedálů a tlačítek umožňující chirurgovi kontrolu nad danou operační procedurou a dává mu možnost měnit jeho nastavení. Díky tomuto rozhraní dokáže chirurg ze vzdálenější konzole kontrolovat pozici robotických rukou, ale i obsluhovat všechny nezbytné funkce nástrojů.

Důležitou součástí celé konzole, v níž sedí chirurg, je elektronické ovládání. Na něm je možné nastavit rychlost, spolehlivost a speciální ochranný systém. Díky němu dokáže plně kontrolovat všech 48 stupňů volnosti s obnovovací frekvencí 1000 cyklů za vteřinu. Chirurg během operace sedí v konzoli, která je umístěna za sterilní částí operačního sálu. Pohyby nástrojů kontroluje skrze stereoskopické brýle s 3D zobrazovacím systémem a operační výkon provádí pohyby páky.

Druhá část celého systému Da Vinci je samotnému chirurgovi vzdálenější, protože je vždy umístěna nad lůžkem operovaného pacienta. Všechny části tohoto komplexu musí být sterilní. Sterilizace se provádí před každým zákrokem. Robot má celkem čtyři pracovní ramena. Na jednom z nich je na konci umístěna kamera, která přenáší v reálném čase obraz chirurgovi do zobrazovacího systému. Ostatní tři ramena jsou vždy zakončena vyměnitelnými nástroji. Nachází se zde také rozhraní pro asistenty hlavního chirurga, který operaci řídí z konzole. Všechna ramena s nástroji plně ovládá chirurg.



*Obrázek 3 Ovládání nástroje ruky v chirurgově konzole*  
[1]

Kamera a nástroje na ramenech jsou umístěny na chirurgickém vozíku, který se pohybuje pomocí koleček a lze s ním volně manévrovat kdekoli v chirurgickém sále. Před samotnou operací je umístěn nad pacientem. Nedávno byl představen systém Da Vinci S, který je elektricky poháněn, má robotické ruce s nižším profilem a nově i druhý monitor připevněný přímo na operační stůl. Na každém vozíku je rameno zakončené kamerou, na zbylých třech jsou nástroje. Robotické paže imitují lidské paže s rameny, lokty a zápěstími. Výměnné nástroje jsou umístěny na robotické ruce. Nástroje musí být sterilizovány, protože se nachází nejbližší pacientovi, na jejich konci jsou řazeny instrumenty od skalpelů po různé uchopovače (viz část 2.2 Možnosti zakončení ruky). Zobrazování v endoskopech využívá dva nezávislé optické kanály, které zakončuje dvojice 3 CCD kamer [6],[9].

### **1.3 Určování stupňů volnosti**

Stupeň volnosti je obecná kinematická veličina, která určuje jakým počtem nezávislých souřadnic je určena poloha tuhého tělesa a udává jeho pohybové možnosti. Tuhé těleso je soustava hmotných bodů, které nemění při pohybu svoji vzájemnou polohu. Volné těleso v rovině má tři stupně volnosti (dva translační a jeden rotační), v prostoru šest, z nichž jsou tři rotační (orientace tělesa vzhledem ke vztažné soustavě) a tři translační (udávají polohu

hmotného středu tělesa). Translační pohyb (TP) je pohyb, při kterém je přímka proložená objektem neustále rovnoběžná se směrem pohybu, rychlost a zrychlení všech bodů zůstávají stejné. Pokud daným tělesem proložíme přímku – osu otáčení (všechny body na ní ležící zůstávají v klidu), a začneme jí kolem této přímky rotovat, bude se jednat o pohyb rotační (RP). Trajektorií všech bodů jsou souosé kružnice ležící kolmo k ose otáčení.

Pohyb tělesa může být omezován vazbami, které odebírají určitý počet stupňů volnosti podle svého typu. Vazby můžeme rozdělit do dvou skupin, v závislosti na oblasti pohybu, a to na rovinné a prostorové. Rovinných vazeb je celkem šest, v závorce je uveden odebraný počet stupňů volnosti: rotační (2 TP), posuvná (1 RP, 1 TP), valivá (1 TP), posuvný kloub (1 TP), obecná (1RP nebo 1TP) a vetknutí (1RP, 2TP). Například kloub může odebrat jeden stupeň, pokud bude posuvný nebo dva, v případě, že se bude jednat o pevný. V prostoru je tento počet možných vazeb velký, pro příklad uvádím: šroubová vazba, sférický kloub [10].

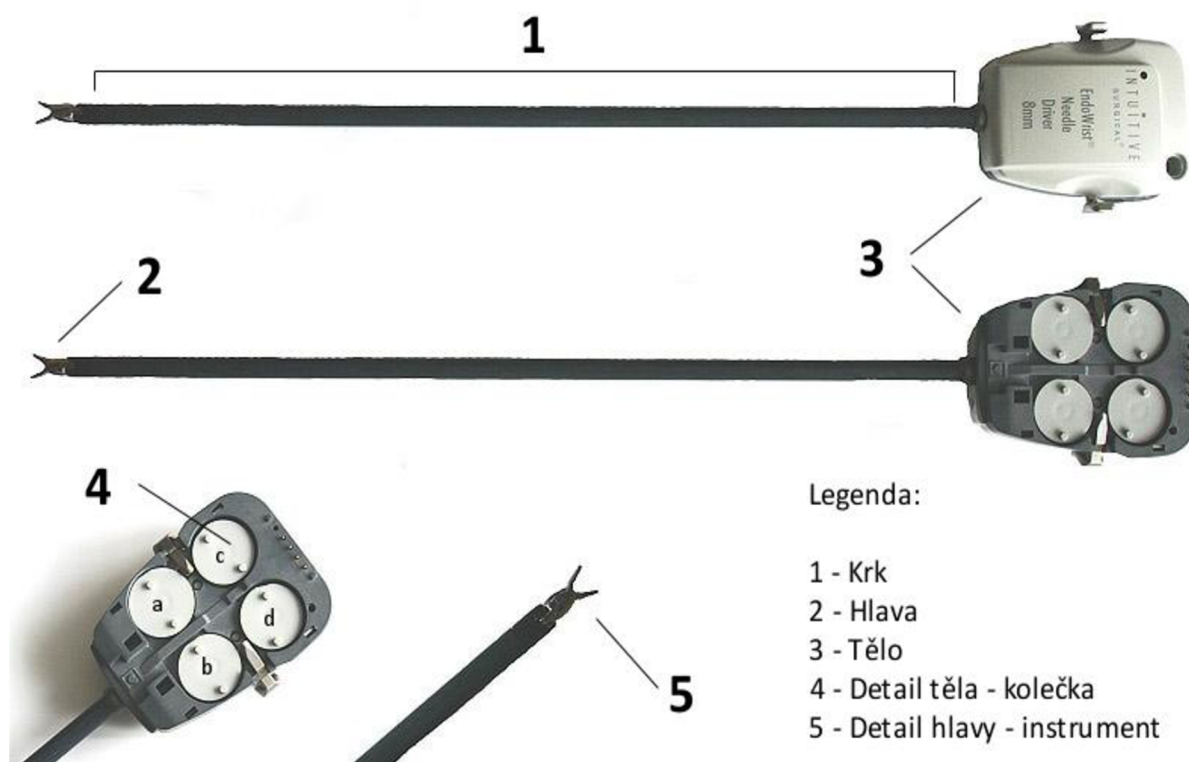
## 2. Nástroj ruky robota Da Vinci

Operační robot Da Vinci od firmy Intuitive Surgical má celkem tři ramena, na jejichž koncích se nacházejí tzv. nástroje ruky. V současnosti je možné zakoupit i nový typ robota, který je vybaven jedním ramenem navíc. Každé rameno má na svém konci specifické zakončení, pokud je třeba vyměnit daný instrument za jiný, mění se celý nástroj ruky. Možná zakončení nástroji jsou popsána v části 2.2 Možnosti zakončení nástroje ruky. Právě k tomuto nástroji ruky, k dispozici je s nůžkami a kleštěmi, je navrženo ovládací zařízení s programem v LabVIEW.

### 2.1 Anatomie nástroje ruky robota Da Vinci

Společnost Intuitive Surgical dodává k robotu Da Vinci nástroje ruky s názvem EndoWrist. Ty se skládají ze tří hlavních částí – těla, krku a hlavy (viz obrázek 4). Každý díl plní specifickou funkci a spolupracuje s ostatními. Nejsložitější je tělo, které tvoří tvarovaná krabička s rozměry 120 x 60 mm. Přesný model celé ruky je zobrazen v příloze B. Na hřbetu těla se nachází označení a specifikace. Druhá strana je z hlediska řízení zajímavější. Obsahuje čtyři otočná kolečka, která se při použití nástroje ruky přikládají na protikus ležící na rameni operačního robota. Dojde tak k přesnému spojení obou součástí.

Každé z koleček nástroje má svoji přesnou funkci, prostřednictvím dutého krku ovládá hlavu ruky. Kolečko *a* (viz obrázek 4) rotuje hlavou ruky. Kolečko *b* umožňuje náklon hlavy v prostoru. Kolečka *c* a *d* mají podobnou funkci, rozevírají a zavírají jednotlivá křídla hlavy. Pomocí těchto pohybů je možné uchopování resp. stříhání v závislosti na použitém instrumentu.



Obrázek 4 Popis nástroje ruky

Při ovládní hlavy různými kombinacemi pohybů koleček dosáhneme velmi přesných pohybů hlavy nástroje ruky. Rozsah pohybu je definován mechanickými mezemi hlavy. Hlava má celkem 4 stupně volnosti. Každé kolečko určuje jeden z pohybů.

Propojovacím článkem mezi tělem a hlavou nástroje ruky je krk. Jedná se o 395 mm dlouhou dutou neohebnou trubičku, která uvnitř vede ocelová lanka, jež jsou plně kontrolována kolečky z těla. Jejich otáčením je lanko prodlužováno resp. zkracováno a dochází tak k ovládní hlavy. Krk při chirurgických zákrocích slouží jako propojovací článek mezi okolím a vnitřním prostředím pacienta.

Poslední část nástroje ruky je jediným prvkem, který je na každé ruce jiný. Možná provedení hlavy jsou podrobněji popsány v části 2.2 Možnosti zakončení nástroje ruky. Zakončovací prvek nástroje ruky s převodními kladkami, na kterých jsou natažena lanka, se v anatomii robotické ruky nazývá hlava.

## 2.2 Možnosti zakončení nástroje ruky

Robotický nástroj ruky má možnost různých zakončení na hlavě (viz část 2.1 Anatomie nástroje ruky robota Da Vinci). Pracovnímu konci se říká instrument a volí se dle potřeb pro danou operaci. Každý instrument má odlišnou funkci. Nejpoužívanější jsou nůžky, uchopovače nebo skalpely. Umožňují provádění minimálně invazivních zákroků. Před samotnou operací je nutné zvolit všechny potřebné instrumenty, které se po důkladné sterilizaci uchycují na ramena operačního robota. Jedno rameno robota má vždy na konci nástroje ruky kameru, která zprostředkovává vizuální obraz chirurgovi na konzoli. Tak může plně kontrolovat pohyby nástrojů uvnitř těla pacienta. Na obrázku 5 jsou vyobrazeny varianty nástrojů, které je možné k robotu zakoupit. Každý nástroj má stejné možnosti pohybu v ose x, y, z a lze s ním i rotovat (viz obrázek 6). Stejně tělo u všech typů ruky umožňuje univerzálnost připojení jakéhokoliv nástroje na libovolné rameno robota.



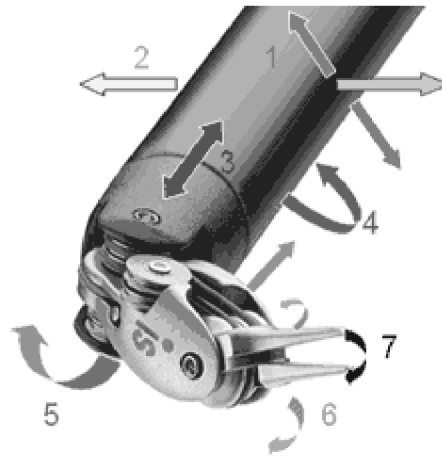
Obrázek 5 Varianty zakončení  
[8]

Pokud je třeba vyměnit instrument, nemění se pouze hlava, ale celý nástroj ruky. Instrument tedy každou ruku omezuje jen na určité typy použití. Pro snadnější orientaci

a rychlejší výměnu jsou na jejím hřbetu vypsány potřebné informace o nástroji. Například EndoWrist Needle Driver 8 mm – Robotická ruka EndoWrist, zakončení vroubkovanými kleštěmi s velikostí funkční pracovní plochy (FPP) 8mm. FPP určuje velikost plochy, kterou je možné při operaci použít.

### 2.3 Pohyby nástroje ruky

Pohyb ruky jako celku umožňují ramena, která ovládají hloubku zanoření ruky, úhlové natočení a její obecnou orientaci v prostoru. V návrhu ovládání není s těmito pohyby počítáno. Projekt je zaměřen na pohyby samostatného nástroje ruky resp. na hlavu s instrumentem. Toto řešení má spoustu omezení, jako je například nemožnost pohybu celé ruky v prostoru. Ruka bude pouze staticky upevněna na mechanickém supportu, který ovládá kolečka v těle a jimi definuje směr natočení nástroje na konci ruky. Celá mechanická sestava musí být pevně uchycena k podložce, aby nedocházelo k nechtěným pohybům, například zatažením za komunikační kabel.



Obrázek 6 Pohyby instrumentu  
[21]

Na obrázku 6 jsou znázorněny hlavní pohyby instrumentu v prostoru. Základní posuny ve směru x a y odpovídají číslům 1 a 2 na obrázku 6. Číslo 4 označuje rotaci, která je zprostředkována krkem ruky. Rozevírání a zavírání nástroje reprezentuje číslo 7. Ostatní označení na obrázku jsou kombinací výše uvedených pohybů. V našem případě nebudeme používat pohyb č. 1, 2 a 3, který je uskutečňován pomocí ramene robota. Z výše uvedených důvodů je pohyb v těchto osách pominut.

Základním pohybem je otevírání a zavírání čepelí. Hlava (kromě nástroje skalpelu) se skládá ze dvou identických křídel, která se navzájem ovlivňují. Výchozí poloha celého nástroje je rovnoběžné zarovnání hrotu nástroje s krkem. V tento okamžik můžeme pohybovat křídly doleva či doprava. Pokud se rozhodnu pohnout pravým na pravou stranu, nijak tím neovlivním druhé. V případě, že budu chtít pravým křídlem na stranu levou, stojí mi v cestě překážka v podobě levého křídla. Pokud je ruka volně (není umístěna na žádném supportu), lze tuto druhou čepel lehce přetlačit. Samotné zabezpečení tohoto pohybu není v softwarové části možné bez zpětné vazby s rukou, proto je s ním počítáno do hardwarové realizace.

## 3. LabVIEW

LabVIEW je grafický vývojový program, který dokáže reálně simulovat měřicí nebo automatizační techniku. Základní programátorské prostředí je tvořeno dvěma okny. V prvním (tzv. Block Diagram) je možné program navrhnout za využití blokových schémat. Tato část programu je později uživateli skryta, ale je ji možno měnit. Druhou částí je uživatelské prostředí (tzv. Front Panel), v níž lze pomocí změny konstant či přepínání tlačítek nastavovat program na hodnoty, které jsou k daným výpočtům nebo operacím třeba. V LabVIEW se dají dobře sestavit i přístroje, které v tzv. virtuálním prostředí mohou měřit a simulovat děje stejným způsobem, jako by šlo o reálné zařízení. Program je vytvořen v LabVIEW verze 8.2.

### 3.1 Program – blokové schéma

Samotný program se skládá ze dvou základních částí. První okno (Front Panel) je určeno pro uživatele, v druhém (Block Diagram) se nachází jádro celého programu (viz příloha C a D).

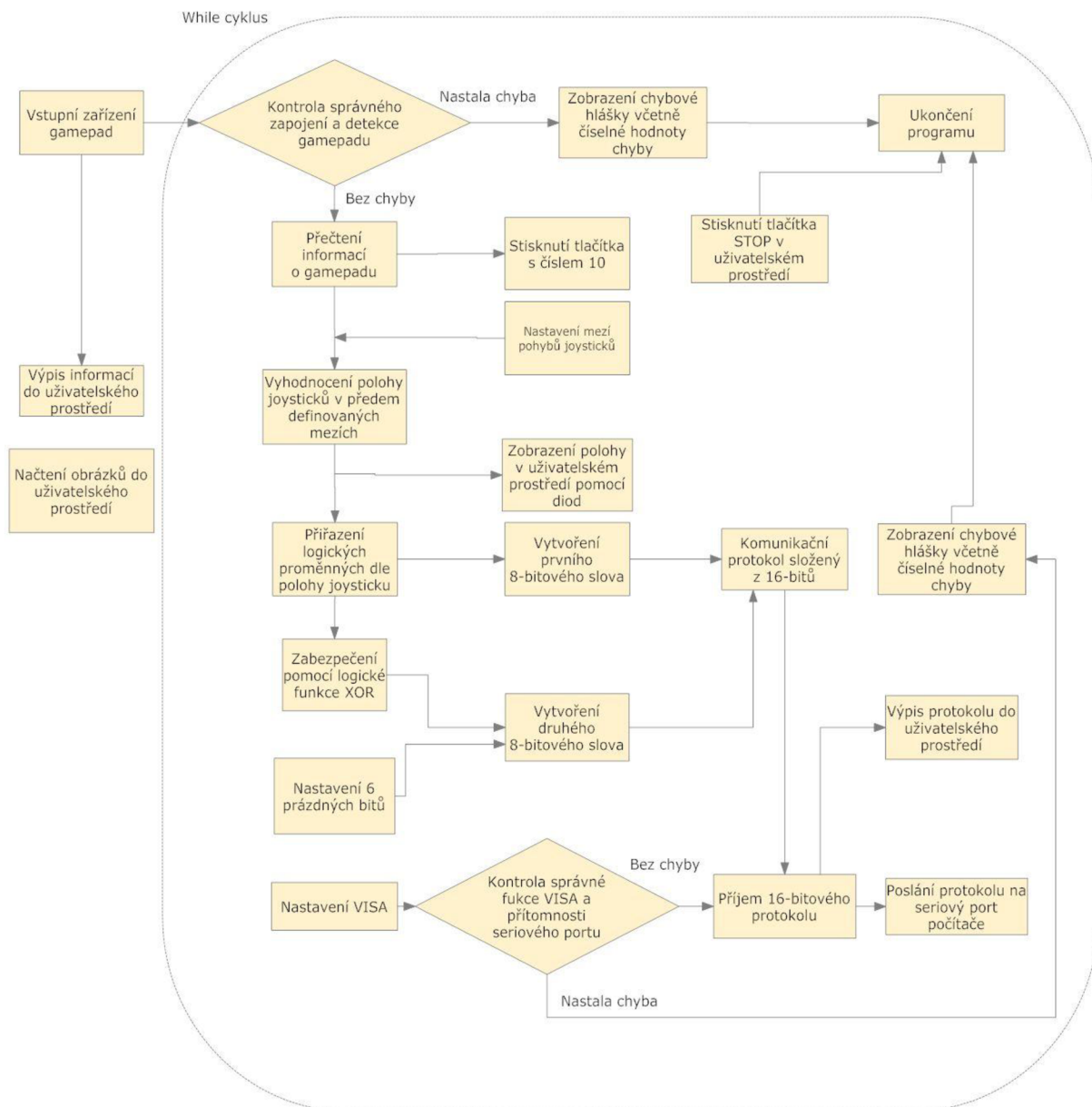
V realizaci řízení nástroje robotické ruky je kladen důraz na jednoduchost, tato skutečnost se odráží i v návrhu uživatelského prostředí. Uživateli je skryta podstatná část nastavení, která jsou využívána v druhém okně (Block Diagram). Tyto konstanty není cílem měnit, protože zajišťují bezchybný chod programu. Uživatel může svými úkony v programu měnit pouze nastavení, které mohou ovlivnit funkci programu po přenosu na jiný počítač.

Na obrázku 7 je zobrazeno blokové schéma programu. Ze vstupního zařízení (viz část 4. Návrh ovládacího zařízení) připojeného k PC pomocí USB sběrnice jsou získávány vstupní informace. Jedná se o standardní hardwarové zařízení – gamepad.

V okamžiku spuštění programu jsou načteny obrázky z úložiště do paměti a jsou zobrazeny v uživatelském rozhraní. Vstupní proměnná má podobu souřadnic polohy joysticků, nebo je reprezentována jako logická proměnná při stisknutí tlačítka. Výstupní proměnnou je 16bitový protokol (viz část 3.6 Návrh komunikačního protokolu), po kterém jsou data přenesena prostřednictvím sériového portu do mikrokontroléru typu Motorola 68HC16, disponujícího dostatečnou rychlostí.

Informace z gamepadu jsou zpracovávány kontinuálně pomocí *while* cyklu, který umožňuje nekonečné opakování smyčky příkazů. Celý program (resp. *while* smyčku) je možné přerušit stisknutím definovaného tlačítka na gamepadu (viz část 4.2 Ovládání nástroje ruky pomocí gamepadu) nebo stisknutím tlačítka *STOP* z uživatelského panelu programu.





Obrázek 7 Blokové schéma ovládacího programu v LabVIEW

### 3.2 Uživatelské prostředí

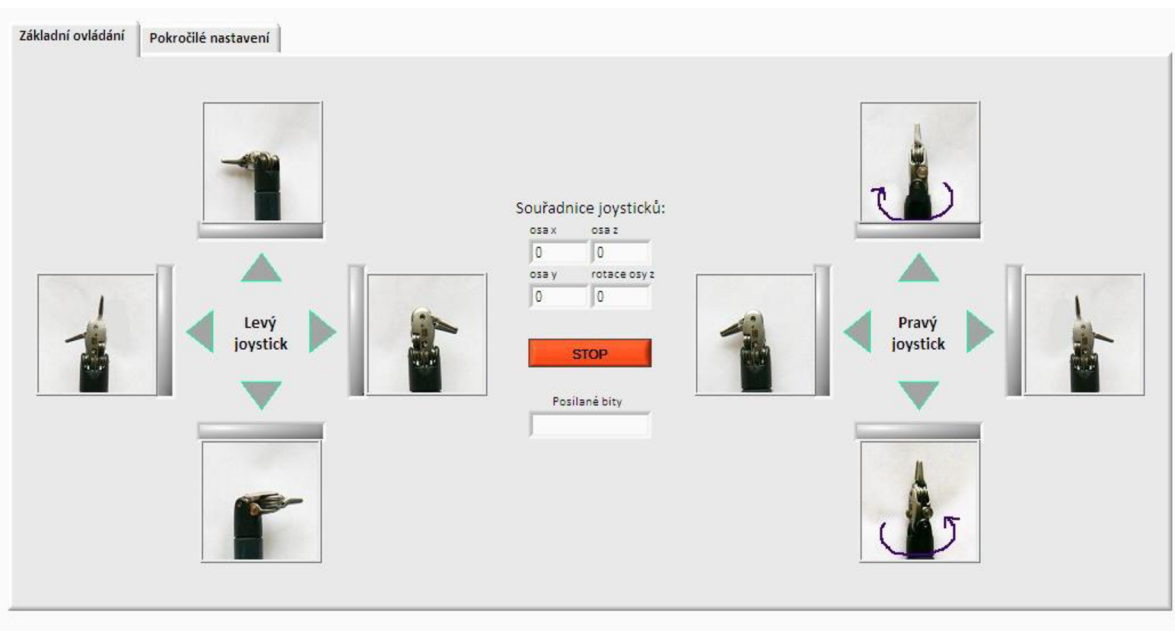
Uživatelské prostředí je navrženo tak, aby bylo přehledné a zobrazovalo na první pohled všechny důležité informace. Skládá se celkem ze dvou záložek, mezi kterými lze snadno přepínat. Celé řízení nástroje je uskutečněno přes vstupní zařízení gamepad. Uživatelské prostředí proto slouží ke spouštění resp. zastavení programu. Při pohybu joysticků gamepadu dochází ke světelné indikaci, která uživateli ukazuje, že přenos pohybu páček probíhá v pořádku. Na panelu lze také kontrolovat hodnotu posílaných dat nebo nastavit vlastní parametry VISA (viz část 3.3 Nastavení a parametry VISA).

V první záložce nesoucí název Základní ovládání se nachází zjednodušené uživatelské prostředí, které poskytuje nejzákladnější informace. Záložku můžeme vizuálně rozdělit na tři hlavní části (viz obrázek 8). V prvním oddílu (nejvíce vlevo) se nachází virtuální zobrazení



levého joysticku gamepadu. Uprostřed jsou uvedeny základní informace. V poslední části (nejvíce vpravo) je opět virtuálně naznačen pravý joystick gamepadu.

Celá tato záložka má poskytovat základní a nezbytné informace uživateli, který z nich snadno vyhodnotí, zda program funguje a reaguje na jeho pokyny. Hlavním úkolem je jednoznačné signalizování pohybů joysticku, které je realizováno pomocí zelených kontrolkek. Jejich poloha v panelu odpovídá reálnému směru pohybu joysticků na gamepadu. Tyto diody se rozsvítí v případě shody a budou zhasnuty, pokud dojde k překročení mezí. Meze jsou nastaveny ve druhém okně programu (Block Diagram) a bude k nim více řečeno v části 3.5 Softwarové řešení programu.



Obrázek 8 Záložka základního ovládání uživatelského prostředí

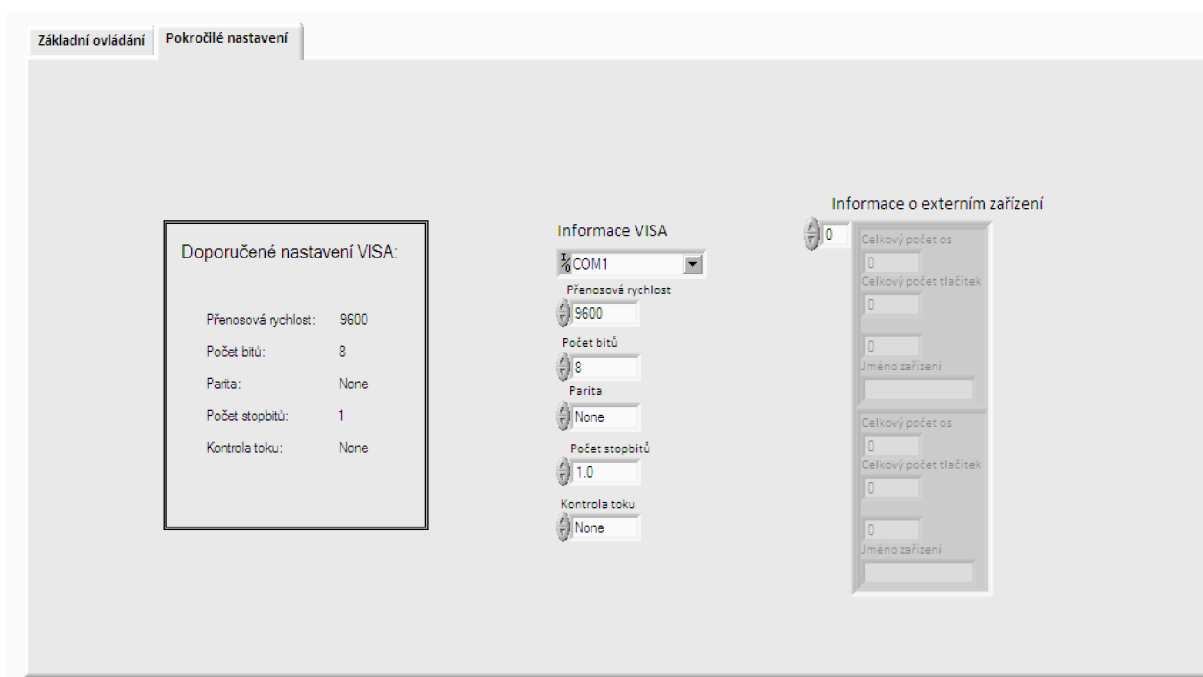
Ke každé kontrolce je přiřazen obrázek hlavy nástroje. Na obrázcích jsou vždy zachyceny krajní polohy nástrojů. Například pokud by uživatel pohnul pouze levým joystickem doleva, otevřelo by se levé křídlo. Při delším podržení páčky v poloze vlevo by se křídlo po určité době, v závislosti na počtu kroků motorků, dostalo do pozice uvedené na obrázku. Dva obrázky jsou doplněny šipkou, která naznačuje rotaci hlavy nástroje ve směru, respektive v protisměru hodinových ručiček. Obrázky jsou nahrávány z adresáře, ke kterému je nastavena relativní cesta. Tento způsob nahrávání obrázků byl zvolen z důvodu snadné přenositelnosti mezi jinými počítači. Cestu k obrázkům nemůže uživatel z uživatelského rozhraní měnit, musel by zasáhnout přímo do zdrojového kódu v Block Diagram.

Obrázky hlavy nástroje robotické ruky slouží pro lepší orientaci uživatele v programu. Může si snadno představit, jaký pohyb bude nástroj ruky vykonávat při určitém pohybu joysticku. Uživatel by měl s ovládáním nástroje ruky vždy začínat ve výchozím postavení hlavy vůči nástroji, to znamená, že hlava by měla být rovnoběžně zarovnána s krkem nástroje ruky.

K zastavení programu slouží červené tlačítko *STOP*, nacházející se uprostřed panelu. Ve středu je také zobrazován v textových polích číselně vyjádřený pohyb joysticku,

resp. jeho aktuální souřadnice. Pro vizuální kontrolu je pod tlačítkem *STOP* zobrazován i výsledný 16bitový komunikační protokol, který je zasílán na výstup sériového portu. Program lze zastavit i pomocí gamepadu, tlačítkem s číslem 10, které leží těsně nad levým joystickem.

V případě nutnosti získání podrobnějších informací o stavu zařízení uživatel přepne na druhou záložku s názvem Pokročilé nastavení (viz obrázek 9). V této záložce lze také nastavit parametry VISA, které přímo ovlivňují průběh přenosu komunikačního protokolu. Může zde nastavit rychlost přenosu v baudech, počet bitů, paritu, počet stop bitů a kontrolu toku. V textovém rámečku jsou uvedeny doporučené hodnoty pro nastavení VISA (Podrobně v části 3.3 Nastavení a parametry VISA).



Obrázek 9 Záložka pokročilého nastavení uživatelského prostředí

Poslední částí uživatelského prostředí jsou informace o externím zařízení v záložce pokročilého nastavení. V tabulce je uveden počet os a počet tlačítek, která jsou na zařízení k dispozici. V posledním textovém poli je uveden název zařízení.

Celý program se spouští přímo z prostředí LabVIEW. Program lze spustit z obou oken, ať už Block Diagram nebo Front Panel, pomocí bílé šipky.

### 3.3 Nastavení a parametry VISA

VISA, softwarová architektura virtuálního instrumentu, je široce používaný vstup nebo výstup (I/O API), který se používá pro komunikaci mezi instrumenty a počítačem. Dokáže komunikovat přes standardní počítačové protokoly, jako je například sériový port nebo USB [11].

V návrhu řízení je VISA využívána pro nastavení parametrů posílaného komunikačního protokolu a pro komunikaci se sériovým portem. V uživatelském panelu s pokročilým nastavením je možné nastavit základní parametry VISA, jako je přenosová rychlost, počet bitů, parita, počet stop bitů a kontrola toku.

Přenosová rychlost je nastavena jako výchozí na 9600 Bd. Jednotka 1 Baud datového komunikačního systému je počet symbolů přenesených za 1 vteřinu. Symbol může mít více než dva stavy a může tedy reprezentovat více než jeden binární bit (binární bit vždy zastupuje přesně dva stavy - zapnuto nebo vypnuto). Přenosová rychlost v baudech není tedy totožná s přenosovou rychlostí v bitech. V našem případě to znamená přenos 9600 symbolů za vteřinu.

S daty může být přenášen také bit parametru parity. Ve výchozím nastavení je vybrána hodnota none - žádná. Bit parity umožňuje jednoduchou kontrolu proti vzniklým chybám, aby bylo možné detekovat korupci dat, která mohla vzniknout důsledkem přenosu. Lze vybrat lichou paritu, sudou paritu nebo vůbec žádnou.

V případě, že je použita sudá či lichá parita, pak je v každém datovém bajtu spočítán počet značek (logických bitů s hodnotou 1) a je přenesen jeden bit s příznakem, jestli počet bitů s hodnotou 1 v přenášeném bajtu je lichý či sudý. Například když je vybrána sudá parita, pak má paritový bit hodnotu 0, pokud počet jedniček v datovém slově je sudé číslo. Lichá parita je přesným opakem, tedy paritový bit je 0, pokud počet jedniček v předchozím slově je liché číslo. Paritu v našem programu není třeba nastavovat, protože tento krok je přímo zabudován v druhém slovu komunikačního protokolu pomocí kontrolního součtu XOR (viz část 3.6 Návrh komunikačního programu).

Parametr počet bitů určuje množství hodnot, které budou přenášeny, jakmile je odeslán startovací bit (start bit). V tu chvíli vysílač odesílá skutečné datové bity. Může být 5, 6, 7 nebo 8 datových bitů. Jejich počet závisí na čísle, které bylo zvoleno v nastavení. Jak vysílač, tak přijímač se musí před začátkem vysílání dohodnout na počtu data bitů za slovo a také na přenosové rychlosti v baudech. Téměř všechna zařízení odesílají data pomocí buď 7 nebo 8 datových bitů.

Datové bity jsou přenášeny v pořadí od LSB (least significant bit) po MSB (most significant bit). Toto pořadí znamená, že na začátku přenosu je odeslán nejméně důležitý bit, bit 0, po němž následuje 4 až 7 bitů zbylých dat, výsledkem tedy může být 5 až 8 bitů dat.

Jednotlivé bity jsou fyzicky reprezentovány úrovní napětí ve vodiči po určitý časový úsek. Záporným napětím je identifikována logická 1, kladným pak logická 0. Rozsahy napětí se obvykle pohybují mezi -5V a -15V, respektive mezi +5V a +15V. Trvání je pro všechny datové bity stejné od počáteční synchronizace pomocí start bitu. V případě, že je využito pouze 7 datových bitů, lze odeslat pouze ASCII hodnoty nižší než 128, zatímco znaky s vyšším kódem odeslat nelze. Nejnižší číslo (tedy 5 datových bitů) limituje přenos na ASCII hodnotu 31. Jakmile byla data přenesena, odešle vysílač stop bit.

V nastavení počtu stop bitů lze ovlivnit délku trvání ukončujícího signálu. Tento příznak má vždy hodnotu 1 a lze jej detekovat správně i v případě, že předchozí datový bit měl také hodnotu 1. Je to způsobeno trváním stop bitů, které je odlišné od standardní délky datového bitu a může trvat v rámci násobků 1, 1.5 nebo 2 časové intervaly klasického bitového přenosu [20].

### 3.4 Kontrola systémových požadavků

Pro správné fungování programu je třeba mít nainstalován ovladač sériového rozhraní NI-VISA 4.62 z oficiálních stránek firmy National Instruments [13]. Při instalaci není třeba žádného speciálního nastavení, k funkci plně dostačuje standardní instalace.

Dále je třeba zajistit, aby využívaný počítač měl v sobě integrován sériový port typu RS-232. V případě použití notebooku, do kterých se tento port neintegruje, je možné využít převodník USB – Serial. Ten umožní připojení sériového zařízení k portu USB, který bude toto zařízení správně detekovat. V případě počítače existují také rozšiřující karty pro sériový port do sběrnice PCI, které stačí jen fyzicky nainstalovat do počítače. Pokud uživatel nekontroluje nebo nezajistí, aby počítač měl tento port, bude na tuto skutečnost upozorněn chybovou hláškou při startu programu v LabVIEW. Ta mu neumožní s programem dále pracovat do té doby, dokud tato skutečnost nebude odstraněna.

Pokud by v počítači bylo více sériových portů (dále jen COM), pak se může stát, že výchozí nastavení portu COM1 v programu v LabVIEW nebude správné. V takovém případě lze zjistit konkrétní identifikaci portu například v programu Measurement & Automation Explorer (dále jen MAX) instalovaným standardně s programem LabVIEW. V položce My System, Devices and Interfaces se nachází rozbalovací seznam se sériovými a paralelními porty, na nichž MAX dokáže nadetekovat připojená zařízení a snadno tak určit správné pořadové číslo používaného COMu.

### 3.5 Softwarové řešení v LabVIEW

Po připojení gamepadu je v programu provedena kontrola správného zapojení a detekce tohoto ovládacího zařízení. V případě nesprávného připojení nebo jiné chyby při komunikaci s hardwarem vyzve program uživatele ke zkontrolování zapojení. Chybové okno zobrazí číselný kód chyby a vypíše její přesný popis. Číslo chyby umožňuje uživateli najít její přesnou charakteristiku a možné řešení přímo u vydavatele softwaru LabVIEW. Veškerá chybová hlášení jsou přeložena do češtiny. Uživatel může na základě této informace přerušit běh aplikace stiskem tlačítka ukončit, které je součástí chybového okna. Při bezchybném připojení program pokračuje dál, bez jakékoli uživatelské odezvy.

Výpis chybových informací je v programu realizován pomocí rozhodovací struktury *case*, ve které jsou vypsány všechny známé chyby, k nimž může dojít při komunikaci s externím ovládacím zařízením. U každé možné chyby je vytvořeno dialogové okno s chybovou hláškou a tlačítkem pro ukončení programu. Číselná hodnota chyb je získávána pomocí chybového svazku.

Stejná struktura včetně stejného seznamu chyb je v programu zopakována ještě jednou, tentokrát má za úkol kontrolovat průběh *while* smyčky. V případě jakéhokoliv problému se zapojením ovládacího zařízení vyvolá dialogové okno a upozorní uživatele na chybu. K této situaci může dojít například při odpojení joysticku.

V případě bezchybného průchodu následuje programové získání dat o poloze obou joysticků nebo o stisknutí tlačítek. Tyto informace jsou posílány do *while* smyčky, kde je program dále zpracovává.

Pomocí bloku *joystickAcquire* jsou přímo získávány veškeré informace o gamepadu, údaje o osách a tlačítkách. Tlačítka, pro která není nadefinována žádná funkce, při zmáčknutí nijak nereagují a nezpůsobují v programu žádnou odezvu. V další části programu jsou z celkového svazku informací rozděleny na základní typy os – osa x, y, z a rotace z. Z tlačítek je využíváno pouze číslo 10.

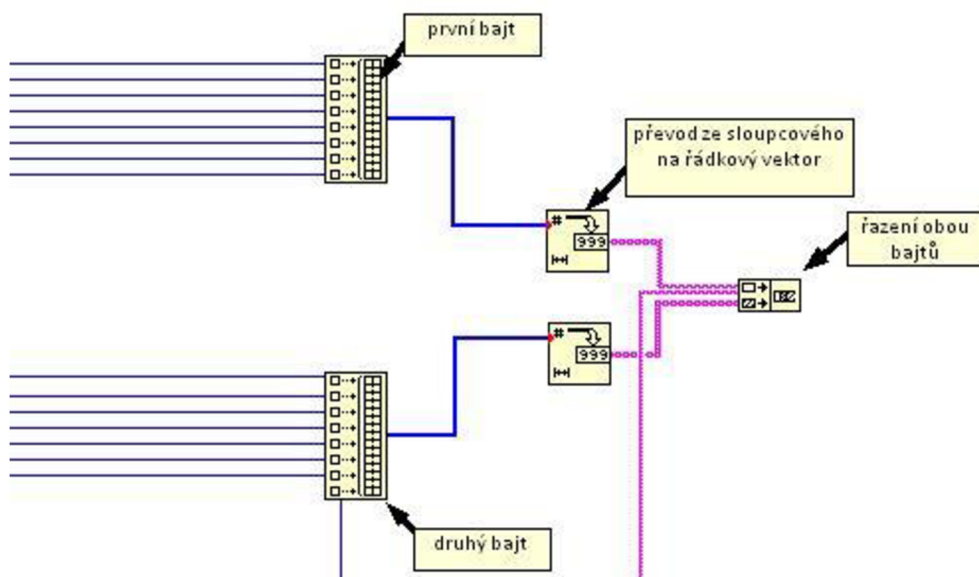
Pro správné vyhodnocování směrů joysticků jsou stanoveny meze, které uživatel nemůže z uživatelského prostředí měnit. V přednastavených intervalech je ponechána určitá tolerance pro klidový stav joysticků, protože ne vždy se podaří docílit přesné polohy obou páček se souřadnicemi [0,0]. Interval klidu, tzn. pozice páček, ve které není přenášen na instrument žádný pohyb, byl stanoven na absolutní hodnoty čtvercového pole o rozměru 4096 x 4096. Pokud hodnota poslaná z joysticku na vstup programu nepřesáhne výše uvedené souřadnicové pole, program tento fakt vyhodnotí tak, že se joystick nepohnul.

V případě, že hodnota tuto hranici přesáhne, program zahájí vyhodnocování směru joysticku. Mezní rozlišovací hodnota pole joysticku je stanovena na hodnotu +/- 32768, tato hodnota vychází z mechanických možností zařízení. Jedná se o poměrně velké rozlišovací číslo, které poskytuje zařízení vysokou citlivost. Vždy se joystick pohybuje buď v záporných, nebo v kladných souřadnicích.

V obou případech dochází k dalšímu zpracování informace. Pokud souřadnice joysticku leží ve stanoveném úseku klidu, je vrácena logická proměnná FALSE, která je převedena na 0 pomocí převodníku boolean to integer. V případě vyhovění nastavenému intervalu je výslednou hodnotou TRUE, který se převede na 1. Pokud se jedná o odezvu TRUE, je tato hodnota zároveň vyvedena na vstup příslušné diody, která je rozsvěcována v uživatelském prostředí. Při změně hodnoty z TRUE na FALSE, kontrolka okamžitě zhasne, aby znázornila uživateli opuštění aktivní oblasti. Z výsledných hodnot ze všech 4 směrů pro oba pákové ovladače je sestaven 1bajtový komunikační protokol ve tvaru sloupcového vektoru.

V další fázi je do komunikačního protokolu přiřazeno druhé slovo o velikosti jednoho bajtu, sloužící primárně k zabezpečení komunikačního protokolu. Prvních 6 bitů je volných. Tato rezerva je ponechána pro budoucí potřeby aplikace, například pro ovládání dalších nástrojů. Poslední dva bity jsou určeny pro kontrolní součet pomocí logické funkce XOR. Zmiňovaný druh zabezpečení je podrobněji vysvětlen v části 3.6 Návrh komunikačního protokolu.

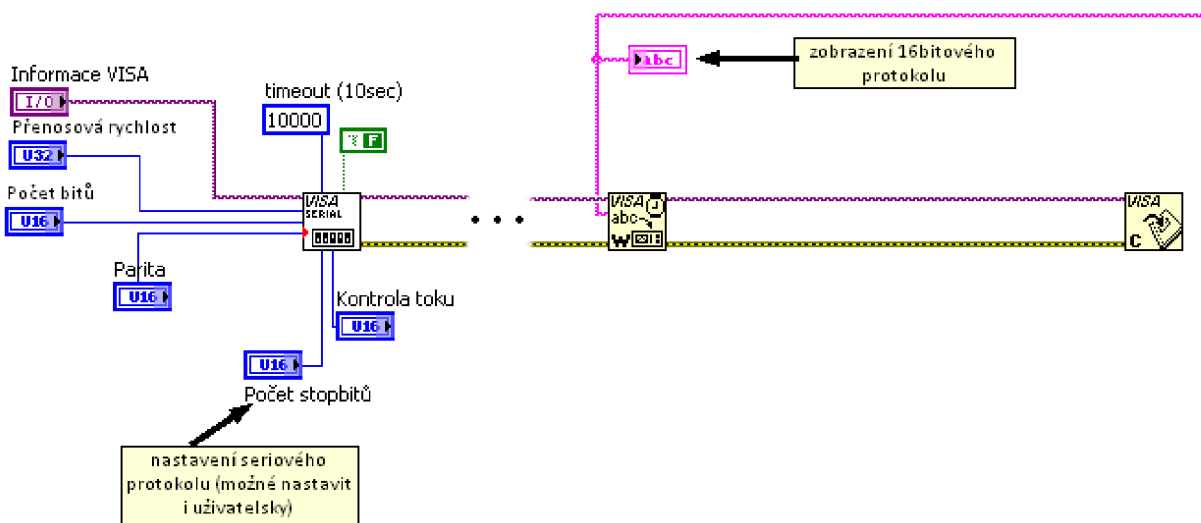
Po naplnění obou slov hodnotami 1 a 0 jsou oba sloupcové vektory o velikosti 1x8 převedeny na vektory řádkové a následně seřazeny za sebe (viz obrázek 10). Vznikne tak základ pro komunikační protokol o velikosti 16 bitů. Prvních 8 bitů patří porovnání směru. Tyto bity slouží pro seřazené čtyři dvojice logických hodnot, které reprezentují aktuální pozice pákových ovladačů gamepadu. Druhá polovina posílaných informací je tvořena šesti nevyužitými bity a dvěma speciálními příznaky pro zabezpečení.



Obrázek 10 Řazení bitů

Tento 16bitový protokol je posílán na vstup bloku *VISA write*, který jej následně odešle na sériový port počítače. V bloku *VISA Configure Serial Port* jsou nastaveny parametry pro odesílání dat komunikačního protokolu. Tyto parametry může uživatel měnit v záložce pokročilého nastavení v uživatelském prostředí.

Komunikační protokol jako takový je obousměrným procesem. Program je možné také rozšířit o programový blok *VISA read*, který umožňuje číst zpětnou vazbu z připojených zařízení. Tento blok není v programu uveden (viz obrázek 11), aby docházelo k jeho bezproblémovému chodu. Program přidáním bloku může být připraven k budoucímu rozšíření o zpětnou vazbu z mikrokontroléru, motorků nebo z robotické ruky. Na základě takových informací by bylo pak možné například sledovat přesnou polohu jednotlivých křídel instrumentu.



Obrázek 11 VISA bez bloku pro hlášení chyb



V případě, že by došlo k jakékoliv chybě, která by mohla ovlivnit průběh komunikace mezi zařízeními skrze protokol, je uživatel upozorněn chybovým hlášením [12]. Analýza jednotlivých chyb je realizována pomocí struktury *case*, která obsahuje seznam všech známých chyb. V případě, že dojde k neočekávané chybě, program vyhledá odpovídající popis na základě identifikátoru chyby a zobrazí dialogové okno obsahující popis chyby a její jedinečné identifikační číslo.

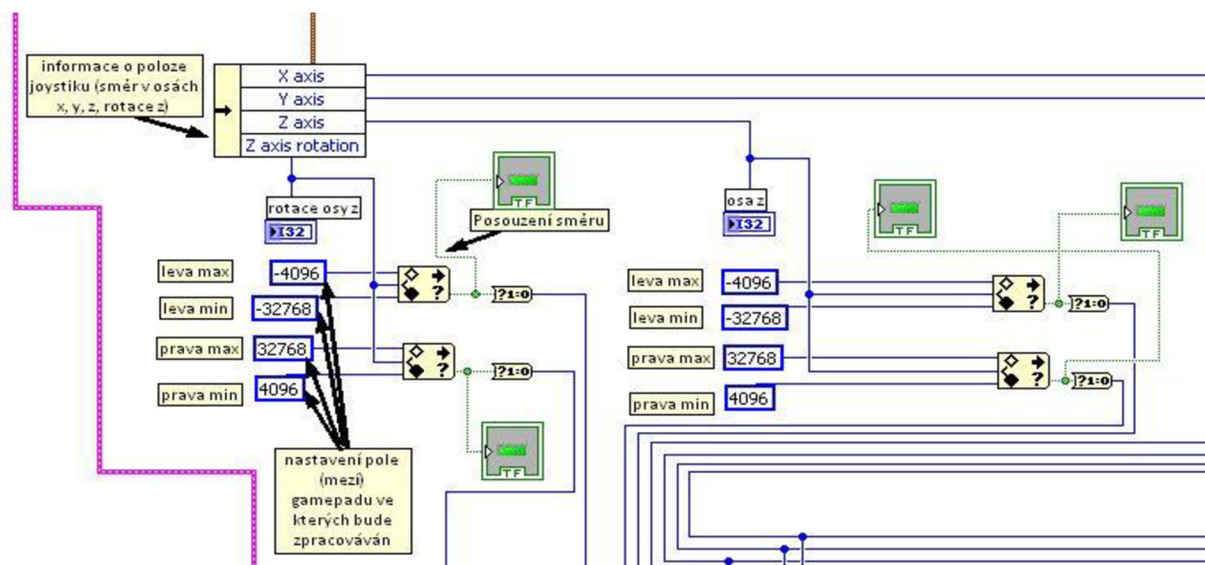
Uživatel v takovém případě bude nucen stisknout tlačítko ukončit, jež je součástí chybového okna a pokusit se chybu napravit na základě informací získaných z jejího popisu. Nejčastější chybou může být nenainstalování ovladače nebo pomnutí faktu, že daný počítač nemá sériový port.

### 3.6 Návrh komunikačního protokolu

Komunikační protokoly slouží ke komunikaci mezi počítačem a externím zařízením. Většinou jsou využívány standardní typy komunikace, například přes TCP/IP. V případě potřeby je možné komunikační protokol pro aktuální potřeby uživatele navrhnout. Je ale třeba pro usnadnění počítat s klasickými počítačovými výstupy. Dnes se nejčastěji používá rozhraní USB, pro naše potřeby je však dostačující sériové. Komunikačním rozhraním pro sériový protokol je RS 232, umožňující oboustranné propojení a komunikaci dvou zařízení [16].

Pro interakci mezi manuálním ovládáním zprostředkovaným programem v LabVIEW a nástrojem robotické ruky je třeba sestavit komunikační protokol. Program v LabVIEW zaznamenává pohyby obou joysticků, které jsou dány externím hardwarovým zařízením, v našem případě se jedná o gamepad značky Logitech (viz část 4.1 Popis gamepadu).

Z bloku pro rozpoznávání zařízení jsou získány potřebné informace o směru, rotaci a jejich přesné číselné hodnoty. Tyto číselné hodnoty jsou porovnány s předem nadefinovanými hodnotami v LabVIEW, které určují, jestli se má ruka pohnout například ve směru osy x v kladném nebo v záporném směru. Výchozí rozsahy joysticků jsou získány pomocí ovládacího panelu herního zařízení ve Windows XP, kde lze dané zařízení i správně nakalibrovat.



Obrázek 12 Nastavení mezí v programu pro určení směru (rotace z a osa z)

Aby bylo možné rozpoznat, zda se joystick pohybuje v kladném nebo záporném směru, je nastaveno rozmezí hodnot pro každý směr. Za nimi následuje rozhodovací blok (viz obrázek 12), vracející příznak TRUE, pokud se proměnné získané z gamepadu pohybují v předdefinovaných hodnotách a FALSE pokud leží mimo ně. Převodním blokem jsou změněna z datového typu boolean na integer, kde TRUE = 1 a FALSE = 0. Z každého směru lze získat čtyři kombinace logických nul a jedniček (viz tabulka 1).

Komunikační protokol bude sloužit pro ovládání krokových motorků. V uvedené tabulce jsou zobrazeny možnosti pro jeden krokový motor. Tato čísla budou základem komunikačního protokolu a obsadí celkem dva bity.

Může dojít celkem ke 4 číselným kombinacím, které ovlivňují pohyby servomotorku:

- 00 zastavení motorků, ale je s nimi možné volně otáček
- 01 motorek se otáčí doprava
- 10 motorek se otáčí doleva
- 11 motorek se zastaví a nebude s ním možné otočit.

*Tabulka č. 1 Kombinace z jednoho směru*

L	0	0	1	1
R	0	1	0	1

Z každého směru získáme tedy různé kombinace 2 bitů, ze všech dohromady 8 bitů. Každý směr reprezentuje jeden motorek, který ovládá tentýž směrový pohyb nástroje ruky pomocí kolečka na těle. Například pokud budeme chtít pohnout pravým křídlem hlavy doprava, zapne se motorek č. 1 a otočí kolečkem doprava. V komunikačním programu se tato skutečnost projeví číselnou kombinací 01 na 5. a 6. bitu komunikačního protokolu.

Na obrázku 13 je zobrazen teoretický návrh komunikačního protokolu. První jednobajtové slovo naplníme pomocí hodnot získaných z joysticků gamepadu a jim přiřazeným logickým hodnotám 0 a 1. První dva bity náležejí pohybu levého joysticku doprava a doleva (pohyb levého křídla), další dva bity pohybu levého joysticku nahoru a dolů (náklonu hlavy v prostoru). Další 4 bity jsou ovládány pravým joystickem, kde 5. a 6. bit určuje pohyb doprava a doleva (pohyb pravého křídla) a poslední dva bity pro pohyb nahoru a dolů (rotace hlavy).

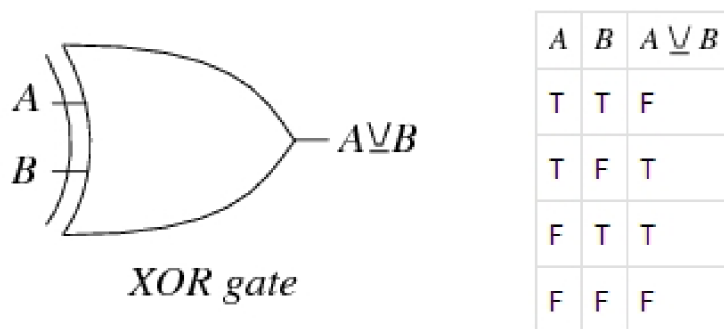
Druhá část protokolu je prozatím naplněna pouze 2 bity, které slouží pro zabezpečení. Ty obsadily poslední dvě místa v celém protokolu. Ostatní bity jsou pro současnou aplikaci ponechány prázdné. Je jim nastavena výchozí neměnná hodnota 0. Volné bity jsou plánovány pro možné další aplikace, například pro ovládání dalších nástrojů robotické ruky.





Obrázek 13 Návrh komunikačního protokolu

Pro zabezpečení komunikačního protokolu byla použita logická funkce XOR [5], kontrolní součet. Matematicky se jedná o tzv. exkluzivní disjunci. Na obrázku 14 je uvedena tabulka možných kombinací se schematickou značkou. Pouze v případě neshody – kombinaci TRUE-FALSE vrací TRUE, ostatní případy jsou vyhodnoceny jako FALSE. V programu je XOR realizován kaskádou XORů, které vzájemně porovnávají vždy sudé a liché bity resp. bity levé - L a pravé - R. Dochází i k zabezpečení 6 bitů naplněných nulami. Proto je XOR L a XOR R řazen na samotném konci komunikačního protokolu. Další zabezpečení není třeba, protože uživatel má nad robotickou rukou vizuální kontrolu.



Obrázek 14 XOR  
[5]

## 4. Návrh ovládacího zařízení

Aby bylo možné robotickou ruku snadno ovládat, byl zvolen hardwarový způsob řízení. Pro běžného uživatele počítače je poměrně jednoduché sžít se s ovladači určenými pro hraní her. Jsou ve většině případů velmi intuitivní a ergonomické, proto není nutné dlouhé seznamování či školení, aby mohl daný člověk s tímto zařízením pracovat. V úvahu byly brány herní joysticky a gamepady. Samostatný joystick se ukázal jako nevhodný pro potřeby ovládní, protože s jeho pomocí by nebylo možné snadno ovládat pohyby nástroje ruky při zavírání, otevírání, rotaci a natáčení v ose z současně. Bylo by třeba použít dva tyto ovladače zároveň. Proto byl zvolen gamepad, u kterého pomocí dvou joysticků ovládanými palci dosáhneme výše zmiňovaných směrových pohybů ruky.

### 4.1 Popis gamepadu

Při výběru byla zohledněna primárně univerzálnost použití. Zejména připojení na jakýkoliv typ PC nebo Mac, bez nutnosti instalování ovladačů a komunikaci pomocí rozhraní USB. V Microsoft Windows je tento hardware automaticky rozpoznatelný pomocí funkce Plug and Play. Zároveň zde není problémem jeho přesná kalibrace.

Ze široké nabídky gamepadů byl vybrán Logitech Gamepad Dual Action (obrázek 15). Hlavními ovládacími prvky jsou dva analogové joysticky, jejichž velkou výhodou je plynulé přenášení okamžité polohy [3]. Pomocí obou páček bude ruka ovládána v základních směrech osy x, y, z a rotaci z (vše v kladných i záporných hodnotách). Kombinace různých poloh obou pákových ovladačů přenáší pohyby na robotickou ruku s velkou přesností. Na čelní straně se dále nachází digitální křížové tlačítko, které zjednodušeně umožňuje nadefinované posuny v osmi směrech (0°, 45°, ..., 315°). Akčních tlačítek (na obrázku 15 pod číslem 3) je celkem sedm. Jedno z nich nese označení mode (mód) a umožňuje přepínání různého nastavení gamepadu. Při zvolení druhého módu se rozsvítí červená kontrolka. Posledním typem jsou boční tlačítka - spouště.



Obrázek 15 Logitech Gamepad Dual Action s popisem tlačítek [7]

## 4.2 Ovládání nástroje ruky pomocí gamepadu

Na obrázku 15 je zobrazen Gamepad Dual Action s popisky prvků. Číslem jedna jsou označeny dva analogové joysticky, které tvoří hlavní část pro ovládání robotického nástroje ruky v základních směrech. Levou páčkou je možné ovládat nástroj ruky ve dvou směrech: x a y. Pohybem joysticku doleva a doprava určujeme orientaci pohybu v kladné i záporné části osy x. Pokud pohneme páčkovým ovladačem nahoru a dolů budeme určovat hodnoty na ose y. Kombinací těchto pohybů dosáhneme 2D plochy, ve které můžeme definovat jakýkoliv bod z rozsahu.

Abychom mohli ruku ovládat v trojrozměrné dimenzi – prostoru, je nutné použít pravý joystick, pomocí kterého definujeme osu z. Z-tová souřadnice je určena pohybem doleva a doprava. Navíc je zde možnost určení rotace ruky v ose z a to pomocí pohybu joysticku nahoru a dolů. Rozsah obou joysticků je od 0, kdy jsou v daném směru ve střední, klidové poloze, až po hodnotu  $\pm 32768$ . Tento uvedený celočíselný rozsah lze definovat ve všech dimenzích (osa x, y, z a rotace z), velikost intervalu představuje dostatečnou přesnost řízení pohybu. Pohyby joysticků jsou vizuálně zobrazeny v uživatelském prostředí pomocí světelných kontrollek.

Stisk tlačítka s číslem 10 je definován pro ukončení programu v LabVIEW. Jak již bylo výše uvedeno ostatní akční tlačítka s čísly 1-4 nejsou využita, ale je s nimi počítáno pro další možná ovládání, například pro ovládání dalších nástrojů ruky. Pomocí jednoho gamepadu by bylo možné řídit všechna, bohužel však ne v současný okamžik, ale vždy jedno po druhém.

## 4.3 Krokové motorky

Pro precizní pohyby robotické ruky musí být použity krokové motorky, které zaručí nastavení přesné polohy a její udržení. S krokovými motorky je počítáno i v části návrhu komunikačního protokolu (viz část 3.6 Návrh komunikačního protokolu). Bez navržení mechanického ovládání by nebylo možné sestavit funkční protokol. Motorek se skládá ze dvou základních částí: ze statoru (sada cívek) a rotoru (hřídel na kuličkových ložiscích).

Princip krokového motoru: proud procházející cívkou statoru indukuje magnetické pole, které přitáhne opačný pól magnetu rotoru. Otáčení rotoru pomocí rotujícího magnetického pole dosáhneme vhodným zapojením cívek. Krokové motorky můžeme charakterizovat různými typy řízení.

1. Unipolární a bipolární řízení. U unipolárního prochází proud v jednom okamžiku jen jednou cívkou, tento motor má největší krouticí moment. Výhodou je jednoduché zapojení řídicí elektroniky. U bipolárního řízení prochází proud vždy dvěma protilehlými, navzájem opačně orientovanými, cívkami v magnetickém poli.
2. Jednofázové a dvoufázové řízení motorků. Při jednofázovém řízení magnetické pole generuje pouze jedna cívka. Při dvoufázovém dvě sousedící cívky generují shodně orientované magnetické pole.
3. Řízení s plným a polovičním krokem. Řízení s plným krokem - na jednu otáčku je potřeba přesně tolik kroků, kolik zubů má stator daného motoru. Řízením

s polovičním krokem dosáhneme dvojnásobné přesnosti. Technicky se jedná o střídání kroků s jedno- a dvoufázovým řízením [19].

Počet stabilních poloh rotoru určuje počet kroků, které je motorek schopen udělat během jednoho otočení o 360°. Grafy točivého momentu viz příloha A. Abychom mohli krokový motor řídit, potřebujeme ovladač - mikrokontrolér (viz část 3.1 Program – blokové schéma). K dispozici pro účely této bakalářské práce jsou motorky řady PK. Jedná se o bipolární dvoufázové krokové motorky od firmy Vexta. Velikost kroku u motorku je 1,8°. Na obrázku 17 je znázorněna mechanická struktura a v příloze A charakteristické průběhy motorků. Níže jsou uvedena označení všech 4 motorků včetně základních popisů, přesné specifikace jsou na obrázku 16.

PX245-03A-C10	12V	-	0,4A
PX245-03A-C10	12V	-	0,4A
PK245-02A-C23	6V	-	0,8A
PK245-01A-C111	4V	-	1,2A

Alfanumerický název má svůj vlastní systém označení, každé písmeno (cifra) má určitý informativní význam. Pro vysvětlení je uveden tento kód:

### PK 245 – OXA

PK	... označení série motoru
2	... počet fází
4	... rozměr motoru pro 4 = 42mm
5	... délka pouzdra motoru
OXA	... unipolární

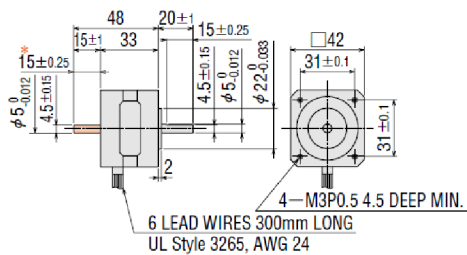
Z výše popsaného označení motorku vyplývá, že v našem případě je pro případnou realizaci počítáno s dvoufázovými unipolárními motorky s rozměrem strany základny 42 mm. Motorky však svými proporcemi přesahují velikost těla robotické ruky, proto pro ovládání ze supportu je nutné vymyslet převodní systém pomocí ozubených koleček.

Model Single Shaft Double Shaft	Connection Type	Holding Torque	Current per Phase	Voltage	Resistance per Phase	Inductance	Rotor Inertia J	Lead Wires (Pin)	Connection Diagram (see page B-197)
		N·m	A/phase	V DC	$\Omega$ /phase	mH/phase	kg·m <sup>2</sup>		
PK243-01A	Bipolar (Series)	0.2	0.67	5.6	8.4	10	35×10 <sup>-7</sup>	6	[3]
	Unipolar	0.16	0.95	4	4.2	2.5			[2]
PK243-02A	Bipolar (Series)	0.2	0.28	13	48	60	35×10 <sup>-7</sup>	6	[3]
	Unipolar	0.16	0.4	9.6	24	15			[2]
PK243-03A	Bipolar (Series)	0.2	0.22	17	77	84	35×10 <sup>-7</sup>	6	[3]
	Unipolar	0.16	0.31	12	38.5	21			[2]
PK244-01A	Bipolar (Series)	0.33	0.85	5.6	6.6	12.8	54×10 <sup>-7</sup>	6	[3]
	Unipolar	0.26	1.2	4	3.3	3.2			[2]
PK244-02A	Bipolar (Series)	0.33	0.57	8.6	15	26.8	54×10 <sup>-7</sup>	6	[3]
	Unipolar	0.26	0.8	6	7.5	6.7			[2]
PK244-03A	Bipolar (Series)	0.33	0.28	17	60	120	54×10 <sup>-7</sup>	6	[3]
	Unipolar	0.26	0.4	12	30	30			[2]
PK245-01A	Bipolar (Series)	0.43	0.85	5.6	6.6	11.2	68×10 <sup>-7</sup>	6	[3]
	Unipolar	0.32	1.2	4	3.3	2.8			[2]
PK245-02A	Bipolar (Series)	0.43	0.57	8.6	15	28.4	68×10 <sup>-7</sup>	6	[3]
	Unipolar	0.32	0.8	6	7.5	7.1			[2]
PK245-03A	Bipolar (Series)	0.43	0.28	17	60	100	68×10 <sup>-7</sup>	6	[3]
	Unipolar	0.32	0.4	12	30	25			[2]

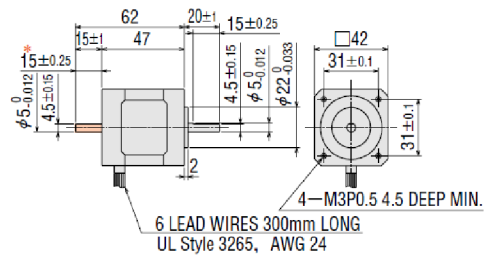
•Degree of Protection: IP30

Obrázek 16 Charakteristiky krokových motorů  
převzato z [17]

- PK243-0□A (Single Shaft) Mass 0.21 kg
- PK243-0□B (Double Shaft) Mass 0.21 kg



- PK245-0□A (Single Shaft) Mass 0.35 kg
- PK245-0□B (Double Shaft) Mass 0.35 kg



Obrázek 17 Technický výkres motorů  
Vlevo motor PK243, vpravo motor PK245 [17]

#### 4.4 Mechanické ovládání

Aby mohl být nástroj ruky ovládán pomocí motorků, je třeba sestavit support, v němž budou uloženy veškeré mechanické součásti. Základní součástí jsou protikusy ke kolečkům na těle ruky, které ovládají její pohyby. Je důležité dosáhnout co největší přesnosti, aby zde vznikaly co nejmenší vůle. Kolečka jsou přímo od výrobce vyrobena z umělé hmoty, proto je třeba zvážit možné opotřebení, pro případ, že by byl protikus vyfrézován ze železa.

Za protikusem bude následovat převodní systém ozubených kol, které musí přenášet stejný počet kroků, jež jsou schopny motorky udělat. Samotné umístění posledního kola bude přímo na hřídeli motoru. Součástí supportu je mikrokontrolér a napájení s transformátorem, který převede běžné síťové napětí 230 V na 12 V napětí. Tento návrh realizace však vůbec nezahrnuje možnost pohybu celé ruky jako celku, kterou v originálním provedení zajišťují ramena. Ruka bude tedy moci vykonávat pouze pohyby hlavou - instrumentem, který je na jejím konci.

Datový přenos 2 bajtovými slovy z programu v LabVIEW bude zprostředkovávat mikrokontrolér. Data jsou posílána pomocí sériového komunikačního protokolu s rozhraním RS 232. Ze supportu budou za tímto účelem vyvedeny dva kabely. Jeden, který bude zajišťovat napájení celého komplexu s robotickou rukou a druhý, který umožní komunikaci gamepadu s rukou pomocí počítače a mikrokontroléru.

## 5. Závěr

V práci se mi podařilo zanalyzovat základní potřeby konečného projektu a najít k nim patřičná řešení. Získala jsem také nezbytné znalosti ve všech potřebných oblastech, především pak ohledně funkce samotného operačního robota. V závislosti na cílech práce jsem zvolila řešení, které se mi dosud zdá optimální - ovládací zařízení gamepad. Ze snadno dostupných ovládacích zařízení nabízí gamepad nejlepší poměr ekonomičnosti a intuitivního řízení, především díky integraci dvou joysticků. V blokovém diagramu jsem navrhla možnost základního provedení ovládacího programu. Podle něj jsem vytvořila program v LabVIEW, který vyhodnocuje polohy joysticků. Program z těchto dat sestavuje komunikační protokol, který odesílá na sériový port počítače. V programu je také počítáno se všemi možnými chybami, které mohou nastat při připojení a detekci gamepadu a při nastavení sériového portu přes VISA. Samotné uživatelské prostředí je jednoduché a umožňuje i možnost uživatelského nastavení VISA. Zabezpečení komunikačního programu je provedeno pomocí kontrolního součtu pomocí logické funkce XOR. Na konci práce je popsáno možné další pokračování řešení týkající se realizace supportu a mechanického ovládní nástroje robotické ruky. V práci se mi podařilo splnit všechny zadané cíle.

## Použitá literatura

- [1] All On Robots: Surgeon robot [online] [cit. 11. 10. 2009]. Dostupný z URL: <[http://www.allonrobots.com/images/surgeon\\_robot.jpg](http://www.allonrobots.com/images/surgeon_robot.jpg)>
- [2] American Medical Association. *Cystoscope Philippe Bozzini* [online] [cit. 15. 3. 2010]. Dostupný z URL: <<http://www.ama-assn.org>>
- [3] BACHIOCHI, Jeff. *Joystick technology: Joystick anatomy* [online]. Vydáno březen 2005, Circuit Cellar, [cit. 15. 11. 2009]. Dostupný z URL: <<http://www.circuitcellar.com/library/print/0305/Bachiochi176/2.htm>>
- [4] DOLEŽALOVÁ, Jana. Reflex. *Robote Da Vinci, nástup* [online]. Vydáno 26. 10. 2006, Reflex 43/2006, [cit. 28. 11. 2009]. Dostupný z URL: <<http://reflex.cz/clanek25338.htm>>, ISSN: 0862-6634
- [5] GERMUNDSSON Roger. *Wolfram Mathworld: XOR* [online]. Poslední aktualizace 4. 12. 2009, [cit. 4. 12. 2009]. Dostupný z URL: <<http://mathworld.wolfram.com/XOR.html>>
- [6] GHARAGOZLOO, Farid, FARZAD, Najam: *Robotic surgery*, New York: Nakladatelství McGraw-Hill Medical, 2009. 418 s., ISBN 978-0-07-145912-9
- [7] Heureka.cz Logitech Dual Action [online, upraveno] [cit. 20. 11. 2009]. Dostupný z URL: <<http://gamepady.heureka.cz/logitech-dual-action>>
- [8] Intuitive Surgical. *Da Vinci Components* [online] [cit. 1. 10. 2009]. Dostupný z URL: <<http://www.intuitivesurgical.com/corporate/newsroom/mediakit>>
- [9] Intuitive Surgical: The da Vinci Surgical System [online]. [cit. 28. 11. 2009]. Dostupný z URL: <[http://www.intuitivesurgical.com/products/davinci\\_surgicalsystm](http://www.intuitivesurgical.com/products/davinci_surgicalsystm)>
- [10] MALENOVSKÝ, E. *Dynamika*. (přednáška) Brno: FSI VUT zimní semestr 2008
- [11] National Instrument. *National Instrument VISA* [online] [cit. 10. 2. 2010]. Dostupný z URL: <<http://www.ni.com/visa>>
- [12] National Instrument Support: Input Device Control VI Error Codes. [online] Vydáno 1. 6. 2010 [cit. 15. 4. 2010]. Dostupný z URL: <<http://zone.ni.com/reference/en-XX/help/371361F-01/lverror/inputdevicecontrolerrorcodes/>>
- [13] National Instrument Support. *NI-VISA 4.6.2 - Pocket PC 2003, Real-Time OS, Windows 2000/7/7 64 bit/XP/Server 2003/Server 2008 x64/Vista/Vista x64/Mobile*. [online] [cit. 10. 2. 2010]. Dostupný z URL: <<http://joule.ni.com/nidu/cds/view/p/id/1605/lang/en>>
- [14] Nemocnice sv. Zdislavy. *Centrum robotické chirurgie Vysočina* [online]. [cit. 28. 11. 2009]. Dostupný z URL: <[http://www.nszmostiste.cz/index.asp?levl=158005802&menu\\_id=016&lang\\_=CZ](http://www.nszmostiste.cz/index.asp?levl=158005802&menu_id=016&lang_=CZ)>

- [15] NOSKA, Martin. Computerworld. *Robotická chirurgie přinesla revoluci i do českého zdravotnictví* [online]. Vydáno 16. 5. 2009, [cit. 28. 11. 2009]. Dostupný z URL: <<http://computerworld.cz/aktuality/Roboticka-chirurgie-prinesla-revoluci-i-do-ceskeho-zdravotnictvi-4036>>, CZ-ISSN 1210-9924
- [16] OLMR, Vít. *Hw.cz: HW server představuje – sériová linka RS – 232* [online]. Poslední aktualizace 12. 12. 2005, [cit. 13. 4. 2009]. Dostupný z URL:< <http://hw.cz/rs-232>>
- [17] OPIS Engineering. *PK – series, 2 phase*. [online] [cit. 12. 12. 2009]. Dostupný z URL: <[http://www.opis.cz/orientalmotor/pdf/pk-series\\_2-phase.pdf](http://www.opis.cz/orientalmotor/pdf/pk-series_2-phase.pdf)>
- [18] PROCHÁZKA, Jan. Olomouc.cz. *V olomoucké nemocnici operuje prostatu robot da Vinci* [online]. Vydáno 1. 10. 2009, [cit. 28. 11. 2009]. Dostupný z URL: <<http://zpravodajstvi.olomouc.cz/view.php?uname=V-olomoucke-nemocnici-operuje-prostatu-robot-da-Vinci-12306>>
- [19] ŘEZÁČ, Kamil. *Krokové motory* [online]. Vydáno 28. 10. 2002 [cit. 17. 11. 2009]. Dostupný z URL: <<http://robotika.cz/articles/steppers/cs>>
- [20] PCCOM - Solution Provider for RS232/422/485 Serial Communication. *Baud Rate* [online]. Vydáno 9. 5. 2008 [cit. 27. 3. 2010]. Dostupný z URL: <[http://www.pccompci.com/Baud\\_Rate.html](http://www.pccompci.com/Baud_Rate.html)>
- [21] Urology Specialists, P.C. Robot [online] Dostupný z URL: <<http://www.urologyspecialistskc.com>>



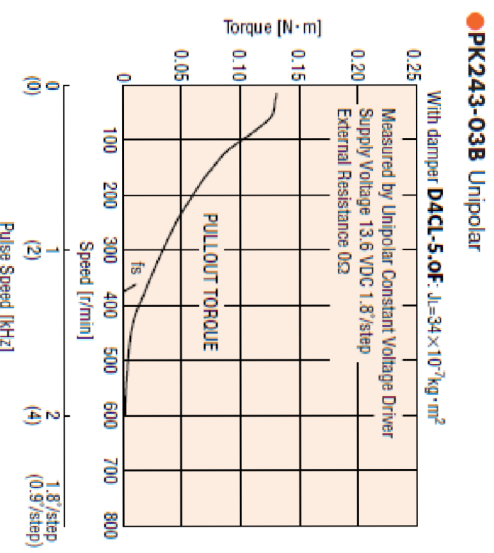
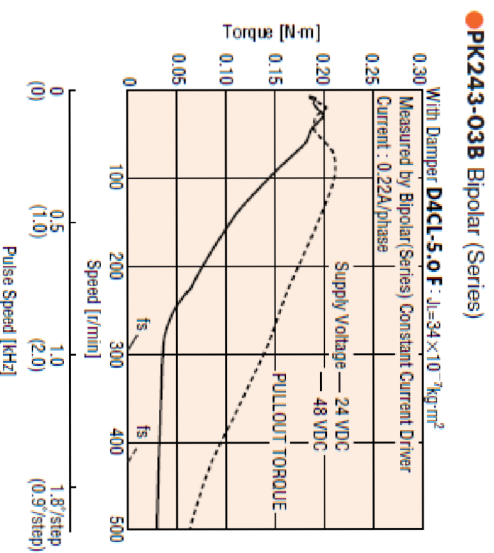
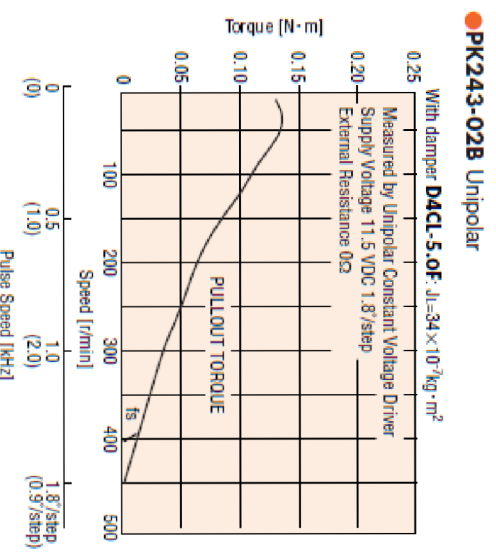
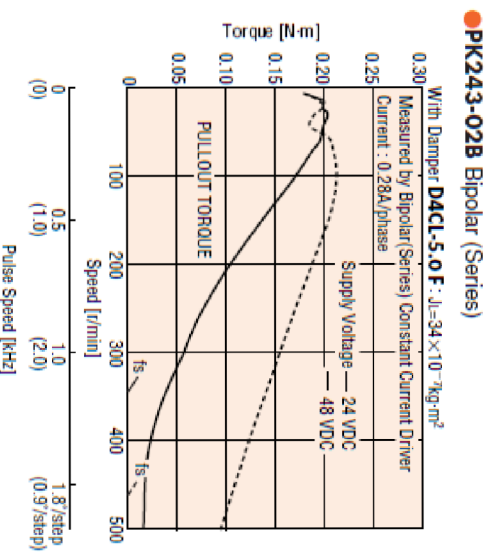
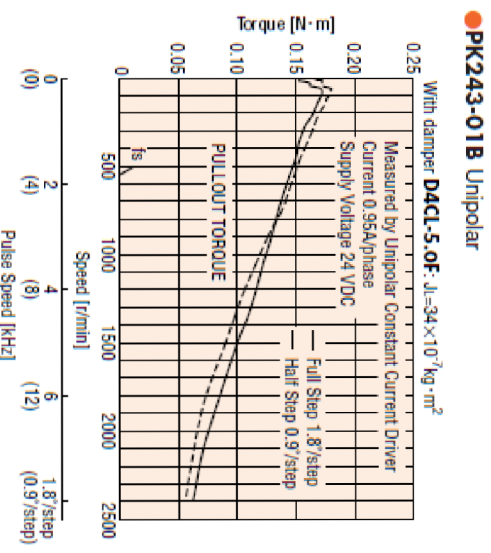
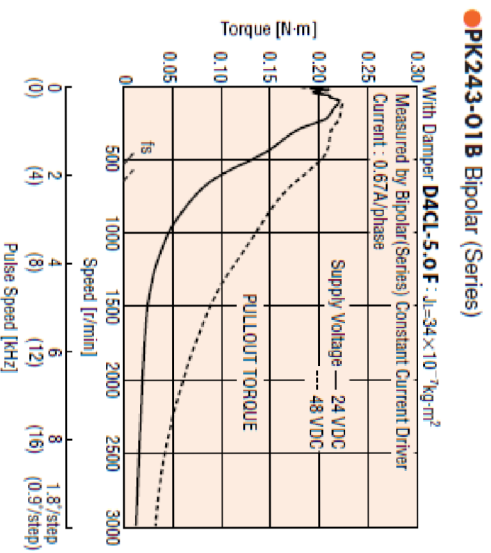
## Seznam zkratek

2D	-	Dvě dimenze
3D	-	Tři dimenze
AESOP	-	Automated Endoscopic System for Operative Positioning
API	-	Application program interface
CCD	-	Chip charge Coupled Device
CRT	-	Cathode Ray Tube
F	-	FALSE - nepravda
FPP	-	Funkční pracovní plocha
I/O	-	Input / Output (vstup/výstup)
L	-	Left - vlevo
Log.	-	Logická
LCD	-	Liquid Crystal Display
Mac	-	Macintosh
MIO	-	Minimálně Invazivní Operace
NASA	-	National Aeronautic and Space Administration
PC	-	Personal Computer
PCI	-	Peripheral Component Interconnect
R	-	Right - vpravo
RP	-	Rotační pohyb
RS 232	-	Recommended standard 232
R. U. R.-	-	Rossum's Universal Robots
T	-	TRUE - pravda
TCP/IP	-	Transmission control protocol / internet protocol
TP	-	Translační pohyb
USB	-	Universal Serial Bus
VATS	-	Video-Assisted Thoracic Surgery
VE	-	Video endoskopický
VISA	-	Virtual Instrument Software Architecture
Xor L	-	Xor levý
Xor R	-	Xor pravý

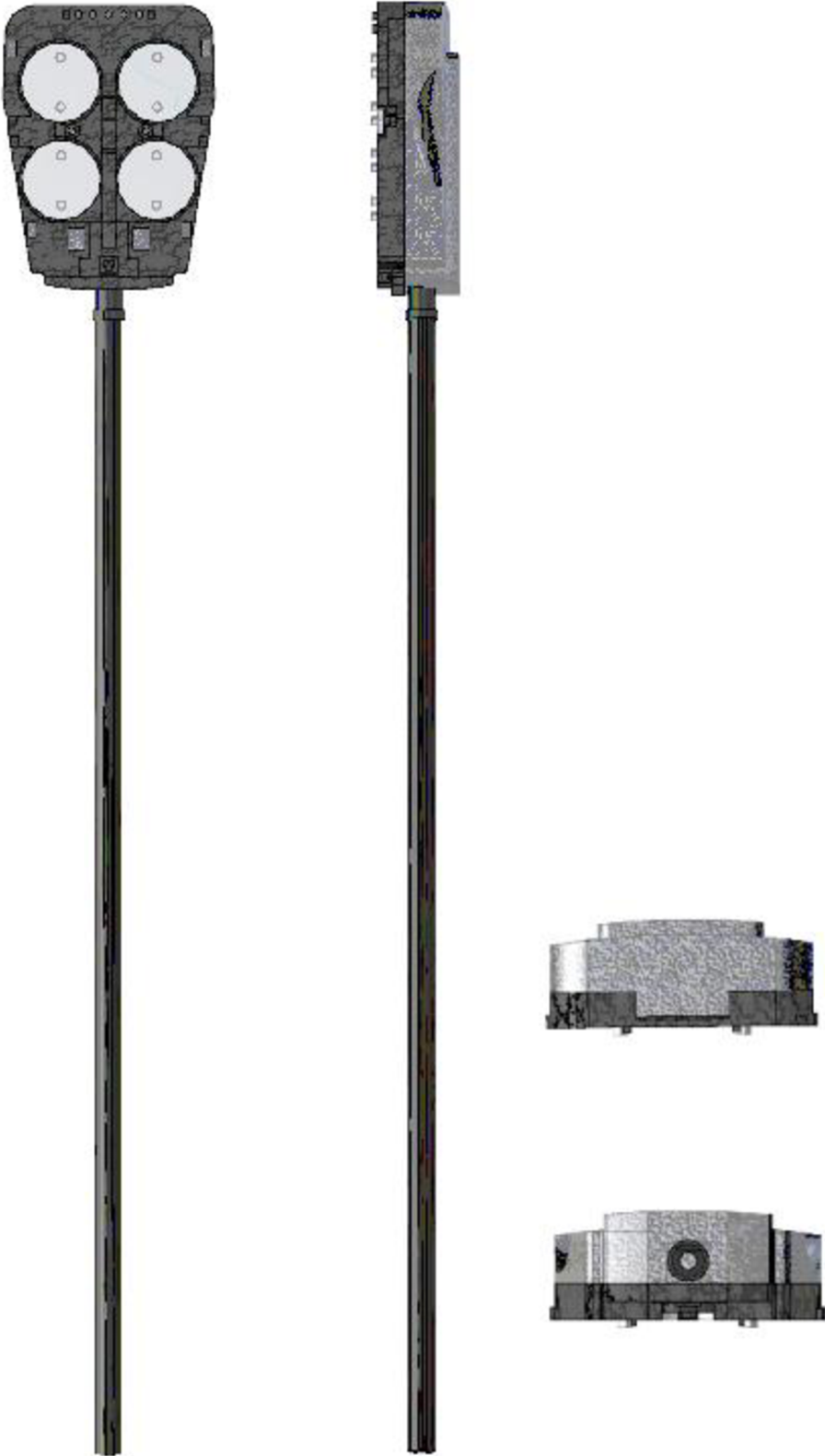
## Seznam příloh

Příloha A .....	39
Příloha B .....	40
Příloha C .....	41
Příloha D .....	42

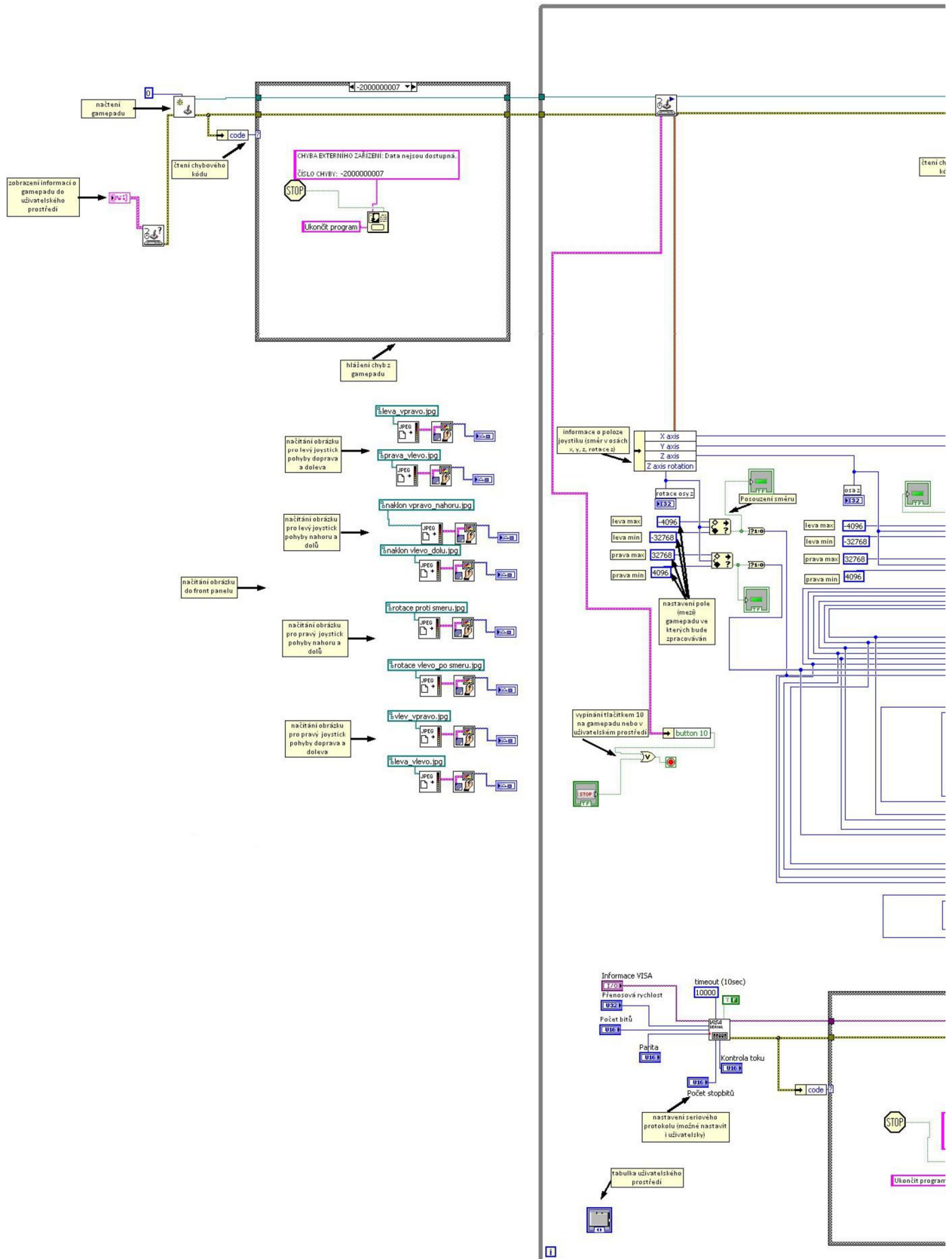
# Príloha A: grafy točivých momentů motorů série PK



# Příloha B: Nástroj robotické ruky - Inventor



# Příloha C: první část blokového schématu programu



# Příloha D: druhá část blokového schématu programu

