



ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA BIOMEDICÍNSKÉHO INŽENÝRSTVÍ
Katedra biomedicínské informatiky

Rozhraní člověk – stroj pro ovládání robotické paže

Human-machine interface for controlling the robotic arm

Bakalářská práce

Studijní program: Biomedicínská a klinická technika

Studijní obor: Biomedicínská informatika

Autor bakalářské práce: David Jirsa

Vedoucí bakalářské práce: Ing. Jan Kauler, Ph.D.

Konzultant práce: Mgr. Radim Krupička, Ph.D.

Kladno 2017

Katedra biomedicínské informatiky

Akademický rok: 2016/2017

Z a d á n í b a k a l á ř s k é p r á c e

Student: **David Jirsa**
Obor: Biomedicínská informatika
Téma: **Rozhraní člověk - stroj pro ovládání robotické paže.**
Téma anglicky: Human-machine interface for controlling the robotic arm.

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

Využijte miopotenciálového náramku od firmy MYO s dodanou knihovnou za účelem ovládání robotické paže (předloketní protézy). Implementujte své programové vybavení k poskytnuté robotické paži s kontroléry EPOS pro ovládání pohonů a jejich programovým vybavením. Vámi vyvinuté a implementované programové vybavení umožní ovládat jednotlivé kinematické dvojice robotické paže pomocí miopotenciálových aktivit svalových skupin na vaší paži.

Realizujte dvě paradigmatu ovládání :

1. Podle zvolených gest (předdefinovaná v Myo knihovně) s využitím pozičního módu mikrokontrolérů EPOS. Konkrétnímu gestu bude odpovídat jednoznačná konfigurace předloketní protézy.
2. Podle miopotenciálové obálky rozpoznávaných svalových skupin s využitím momentového módu mikrokontrolérů EPOS. Při kontrakci daného svalu se začne určený pohon protézy pohybovat, rychlost bude úměrná síle zatnutí svalu. Pro přepínání pohonů jednotlivých kinematických dvojic protézy můžete využít i integrovaných akcelerometrů v MYO náramku.

Seznam odborné literatury:

- [1] SCIAVICCO Lorenzo, SICILIANO Bruno, Modelling and control of robot manipulators, ed. 1, Springer - Verlag London Limited, 2001, ISBN 1-85233-221-2
[2] , Sachseln (Switzerland), [<http://www.maxonmotor.ch/ch/en/index.htm>], 2011, [Citováno 2011], Přístupné z: Online
[3] BEKEY, George A, Autonomous robots: from biological inspiration to implementation and control, ed. 1, The MIT Press, 560 s., ISBN 978-0262025782

Zadání platné do: 11.09.2018

Vedoucí: Ing. Jan Kauler, Ph.D.

Konzultant: Mgr. Radim Krupička, Ph.D.

.....
vedoucí katedry / pracoviště

.....
děkan

V Kladně dne 20.02.2017

PROHLÁŠENÍ

Prohlašuji, že jsem bakalářskou práci s názvem *Rozhraní člověk – stroj pro ovládání robotické paže* vypracoval samostatně pouze s použitím pramenů, které uvádím v seznamu přiloženém k bakalářské práci.

Nemám závažný důvod proti užití tohoto školního díla ve smyslu § 60 zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů.

V Kladně dne

.....

David Jirsa

PODĚKOVÁNÍ

Rád bych touto cestou poděkoval vedoucímu bakalářské práce Ing. Janu Kaulerovi, Ph.D., za poskytnuté konzultace ohledně řešené problematiky a užitečné podněty při tvorbě práce. Dále bych rád poděkoval konzultantovi bakalářské práce Mgr. Radim Krupička, Ph.D. za ochotu a vstřícný přístup řešení problému, především se softwarovou částí bakalářské práce.

ABSTRAKT

Rozhraní člověk – stroj pro ovládání robotické paže

Tato práce předkládá návržení a vyhotovení ovládání robotické paže, která se nachází v prostorách fakulty. Ovládání robotické paže je uskutečněno pomocí náramku Myo, který snímá myopotenciálovou aktivitu svalů a dle zvoleného módu ovládání (užití gest dlaně nebo EMG obálky) dochází k rozpohybování paže. Na základě experimentů bylo zjištěno, že vytvořené řešení vyhovuje potřebám projektu.

Klíčová slova

Robotická paže, Myo náramek, Protéza paže, EPOS kontrolér

ABSTRACT

Human-machine interface for controlling the robotic arm

This thesis presents design and creation controlling robotic arm, which is located on the premises of the Faculty. Control of the robotic arm is accomplished by Myo armband, which monitor mio potencial aktivty of the muscles of the arm and according to selected control method (use hand gestures or EMG envelope) occurs into arm motion. On the basis of the experiments it was found that the proposed solution meets the needs of the project.

Keywords

Robotic arm, Myo armband, Prostetic arm, EPOS controller

Obsah

Seznam zkratk	5
1 Úvod	6
1.1 Struktura práce	7
2 Přehled současného stavu	8
2.1 Protézy horních končetin	8
2.1.1 Tahové protézy	9
2.1.2 Myoelektrické protézy	10
2.2 Hardware řešení projektu	11
2.2.1 Náramek MYO	11
2.2.2 Robotická stavebnice Rascal	13
2.2.3 Robotická paže ze součástek maxon motor	14
2.3 Robotická paže ovládána myšlenkami	16
3 Cíle práce	17
4 Návrh aplikace	18
4.1 Analýza požadavků	18
4.2 Technické specifikace	18
4.3 Funkční specifikace	19
5 Implementace	20
5.1 Myo SDK MATLAB MEX Wrapper	20
5.2 C# Wrapper for Myo Armband	22
5.3 Vlastní implementace	22
5.4 Vícevláknové aplikace	23
5.4.1 Teorie	23
5.4.2 Využití vláken	24
5.4.3 Nevýhody vláken	24
5.4.4 Řešená situace	24
5.5 Návrhové vzory	26
5.5.1 Návrhový vzor MVC	26
6 Uživatelská dokumentace	28

6.1	Připojení a kalibrace náramku Myo	28
6.2	Připojení robotické paže	32
7	Testování	34
8	Diskuse	36
9	Závěr.....	37
	Seznam použité literatury	38
	Seznam obrázků.....	41
	Seznam tabulek	43
	Obsah na přiloženém CD	44
	Přílohy.....	45
	Třída MyoConnector	45

Seznam zkratek

Zkratka	Význam
EMG	Elektromyografie
API	Application Programming Interface
EPOS	Modulární, digitální poziční kontrolér
DI	Digitální vstup

1 Úvod

Lidské tělo je neskutečně složitý mechanismus, ve kterém jdou funkční prvky velmi složitě nahrazovat. Teprve v posledních dekáдах došlo k velmi významnému technologickému pokroku a nově získané technologie se promítli do mnoha dalších průmyslových odvětvích ale také do medicíny či farmakologie.

Díky mohutnému rozvoji na poli robotiky a kybernetiky dochází v posledních letech ke konstrukci funkčně skoro dokonalých strojů, které jsou sestrojovány, aby usnadnili lidem práci a zvyšovali životní komfort, avšak vytvořit stroj tak aby byl v souladu s lidským organismem je velkou výzvou pro inženýry.

Horní končetina, paže, je funkční celek, pro jehož bezchybný provoz je zapotřebí aby bylo dosaženo dokonalé souhry 27 kostí, 39 svalů a 36 kloubů, nervů a šlach. Všechny tyto části se starají o každodenní činnosti a žádný zdravý jedinec se nedokáže představit jaké by to bylo, kdyby ony činnosti nemohl z důvodu absence horní končetiny provádět. Důsledkem může být pacientovo vyčlenění ze společnosti a vážné psychologické problémy.

Protézy se snaží navrátit alespoň základní funkčnost chybějící končetiny. V posledních letech dochází k postupnému nahrazování standartních tahových protéz modernějšími myoelektrickými protézami, které umožní pacientovi znovu využívat aktivní úchop a tvarem připomínají více horní končetinu. Nevýhodou těchto protéz je, že pro její obsluhu je potřeba dlouhá rehabilitace svalů v amputovaném pahýlu a teprve až ke konci rehabilitace, která trvá zhruba jeden rok, je možné zhodnotit, zda pacient může myoelektrickou protézu využívat. Dalším důležitým aspektem u komerčně nabízených protéz je cena, která se může pohybovat v řádech statisíců až milionů korun a pojišťovny mají přísná kritéria pro jejich proplácení.

Cílem této práce je snaha prozkoumat možnosti myopotenciálového náramku Myo pro obsluhu robotické paže. Výhodou tohoto řešení je to, že náramek nepotřebuje EMG implantáty a cenově je více dostupný, v porovnání s komerčně nabízenými řešeními. Dále samotné řešení není striktně vázáno na protézy, ale mohlo by být využito i v průmyslu, kde by se vzdáleně ovládal stroj v rizikovém prostředí, popřípadě by řešení mohlo být využito při realizaci exoskeletů.

1.1 Struktura práce

Úvodní první kapitola představuje obecný úvod do problematiky, a nastiňuje za jakým účelem byla tato práce vytvořena. Druhá kapitola detailněji pojednává o problematice protéz a popisuje v jaký byl výchozí stav projektu. Třetí kapitola jasně vymezuje cíle práce. Ve čtvrté kapitole jsou rozebrány požadavky, které mají být během řešení dodrženy. Pátá kapitola detailně popisuje postup řešení projektu, popisuje využití metody a software a též pojednává o problémech, které se během tvorby řešení vyskytli. Šestá kapitola slouží jako manuál pro správně spuštění a nastavení programu. V sedmé kapitole je popsáno, jakým způsobem byla vytvořená aplikace testována. Osmá kapitola je věnována diskuzi a možnosti návaznosti na tuto práci. V deváté kapitole je vyneseno rozhodnutí, zda projekt splnil požadavky zadání.

2 Přehled současného stavu

V této kapitole jsou popsány dnes používané protézy horních končetin a použité hardwarové vybavení využito k vypracování zadání práce.

2.1 Protézy horních končetin

V současné klinické praxi jsou využívány výhradně tři druhy protéz, a to protézy kosmetické, tahové a myoelektrické. V následujících tabulkách jsou podotknuty jednotlivé výhody a nevýhody zmíněných protéz. [1]

Protéza	Hmotnost	Aktivita	Úchop	Cena
kosmetická	srovnatelná s lidskou	pasivní	nemožný	nízká
tahová	mírně těžší	částečně aktivní	možný	střední
myoelektrická	znatelně těžší	plně aktivní	přirozený	vysoká

Tabulka 1: Porovnání výhod a nevýhod stávajících protéz

Protéza	Vzhled	Nácvik	Nasazení	Údržba	Poruchovost
kosmetická	dobry	není	snadné	minimální	žádná
tahová	špatný	snadný	obtížné	jednoduchá	nízká
myoelektrická	dokonalý	obtížný	snadné	nákladná	vysoká

Tabulka 2: Hlubší porovnání výhod a nevýhod stávajících protéz

2.1.1 Tahové protézy

Též známé jako pasivní funkční protézy. Konstrukčně jsou řešeny podobně jako kosmetické protézy, ale liší se použitým jádrem. U kosmetické protézy je použito jádro z pěnové hmoty, ale u tahové protézy je použit odlitek z ušlechtilé oceli. Protéza je zakončena, buď terminální kosmetickou rukavicí, která je vybavena hákovým mechanismem pro úchop, nebo dnes oblíbeným pracovním nástavcem, který lze využít pro úchop nejrůznějších nástrojů. Toto zakončení je v oblibě především u mužské části populace, která pracuje s nástroji jako šroubovák atd. V případě, že je protéza vybavena hákovým mechanismem pro úchop, je rozevírání zajištěno druhou rukou.

Pohyb protézy je zajišťován soustavou tahových zařízení a lanek, tažených podél zádové části a ukotvených na nosný systém. Nedostatkem tohoto ukotvení je, že při dlouhodobém aktivním využívání protézy může dojít ke vniku poranění, či poškození míst, kde je protéza připevněna k tělu pacienta. Protéza má také částečně omezený manipulační prostor, který je závislý na koordinaci pohybu ostatních částí těla.

Z celkového hlediska se jedná o mnohonásobné zlepšení funkčnosti oproti kosmetickým protézám. Následující tabulka 3 předkládá hlavní klady a zápory tahových protéz. [1; 2]

Klady	Zápory
nepatrně vyšší hmotnost	potřeba tahových bandáží
cena protézy do 80 000 Kč	vznik poranění při dlouhém použití
vysoká spolehlivost	složitě nasazení
snadný servis	nepřirozený vzhled
vysoká odolnost běžného použití	limitovaný manipulační prostor
snadná obsluha	

Tabulka 3: Klady a zápory tahových protéz

2.1.2 Myoelektrické protézy

V současnosti se jedná o patrně nejdokonalejší ve větším množství vyráběné protézy. Z konstrukčního hlediska jsou velmi podobné se standartními protézami, ale při porovnání řízení pohybu, jsou myoelektrické protézy ohromným krokem kupředu. U těchto protéz je úchop řešen pomocí klešťovitého mechanismu, způsob úchopu je palec s druhým, popř. třetím prstem. Další pohyby, které protéza nabízí, je rotace v zápěstí a alternativní pohyb paží v oblasti lokte (flexe a extenze). Větší estetičnosti vzhledu je docíleno kosmetickým PVC krytem.

Pohyb protézy je zajištěn elektrickou energií, která je uchovávána v systému Li – Ion baterie. K rozpohybování pohonů dochází v závislosti na impulzech vysílaných z amputačního pahýlu, kde se nacházejí implantované elektrody pro snímání EMG. Elektrody jsou vyrobeny ze stříbra o čistotě 99 % při rozměrech 1 cm x 1 cm. Snímání EMG je velmi citlivé na aktivitu okolních svalových skupin, které často ovlivňují požadovaný výsledek. Jedná se o sumační signál v řádu μV maximálně jednotek mV. Snímané EMG je následně zesíleno operačních zesilovačem INA 121, který vyrábí firma Texas Instruments, a přivedeno k aktuátoru, který vykoná požadovaný pohyb.

Z důvodu sumace signálu je zapotřebí, aby pacient cíleně rehabilitoval a izolovaně posiloval zvolené svaly, kterými následně ovládá protézu. Rehabilitace probíhá buď v ambulantní intenzivní péči, či v rehabilitačním ústavu, přibližně po dobu jednoho roku. Avšak i přes usilovný trénink není zaručen úspěch, jelikož cílené svaly nemusí mít dostatečně silný myopotenciál. Teprve až po ukončení rehabilitace je možné posoudit, zda je využití myoelektrické protézy vhodné. Protéza je indikována pouze případech, kdy je pacient po oboustranné amputaci, trpí malformací obou horních končetin, nebo má pouze jednu ruku s vážným omezením úchopu.

Následující tabulka 4 předkládá hlavní klady a zápory tahových protéz. [1]

Klady	Zápory
nezávislé ovládání	potřeba dobíjet baterie
šetrná k pacientovu tělu	složitá příprava pro obsluhu
snadná manipulace	vysoká hmotnost
nezávislost uživatele	vysoká cena (250 000 Kč a více)
přesnost pohybů	poruchovost a drahý servis

Tabulka 4: Klady a zápory myoelektrických protéz

2.2 Hardware řešení projektu

V této části je představen hardware, se kterým jsem měl možnost během celé doby řešení zadání projektu pracovat. U nevyhovujícího hardwaru je vysvětleno, proč byl z možných řešení zamítnut.

2.2.1 Náramek MYO

Náramek byl vyvinut kanadskou společností Thalmic Labs, která byla založena v roce 2012 a náramek uvedla do prodeje v roce 2014. Dnes je možné jej zakoupit v oficiálním obchodě výrobce za cenu 199 USD (cena platná k 16. 5. 2017). Balení obsahuje:

- Náramek Myo (černé, nebo bílé provedení)
- Standartní Micro-USB kabel
- USB Bluetooth® adaptér pro komunikaci náramku s Windows PC nebo počítačem MAC
- Svorčky pro přizpůsobení velikosti náramku



Obrázek 1: Náramek MYO

Náramek je vybaven moderními senzory, jako je osm povrchových EMG senzorů, které jsou vyrobeny z nerezové ocele medicínské kvality. Dále je náramek vybaven senzitivním, 9-osým IMU snímačem, skládá se z tříosého akcelerometru, tříosého magnetometru a tříosého gyroskopu. Pro indikaci stavu náramku je využita dvojice LED diod. Celý náramek je poháněn procesorem ARM Cortex M4. Nabíjení vestavěné lithiové baterie je uskutečňováno pomocí Micro-USB portu a náramek vydrží fungovat na jedno nabití jeden den plného využívání. [3]

IMU snímač se stará o přesné zaznamenání polohy a zrychlení náramku v reálném čase. Na druhou stranu se EMG senzory starají o snímání myopotenciálů jednotlivých svalů kolem celého předloktí a náramek tak dokáže rozeznat prováděné gesto dle předem definovaný gest výrobcem, konkrétně rozpozná:

- sevřená dlaň (Fist)
- mávnutí doleva (Wave Left)
- mávnutí doprava (Wave right)
- roztažené prsty (Fingers Spread)
- dvojklik (Double Tap)

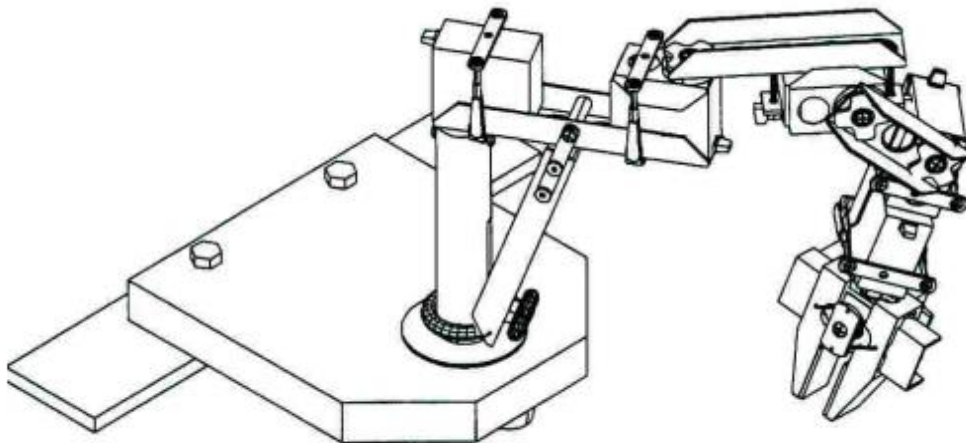


Obrázek 2: Vizualizace podporovaných gest

O propojení náramku s dalšími zařízeními se stará Bluetooth® 4.0 LE a je možné jej propojit s většinou moderních zařízeních jako jsou iPhone a iPad se systémem IOS 7 a výše, zařízení s operačním systémem Android 4.3 a výše. Pro komunikaci náramku s osobními počítači je potřeba použít přibalený USB Bluetooth® adaptér a mít nainstalovaný program „Myo connect“ (ke stažení na stránkách výrobce), tento program lze nainstalovat na osobní počítače s operačním systémem Windows 7 a novější, či Mac s operačním systémem OS X 10.8 a novější.

2.2.2 Robotická stavebnice Rascal

Stavebnici vyrábí a prodává americká společnost Robix a je určena pro studenty. Jedná se jednoduchou stavebnici skládající se z kontroléru, servomotorů a spojovacích prvků konstrukce. V balení se dále nachází potřebná dokumentace pro sestavení deseti funkčních projektů, pro potřeby řešení mého projektu, byla zvolena konstrukce „Chemist“ dle přiloženého návodu. Tato stavebnice byla použita v raném stádiu návrhu možného řešení, ale postupem času se ukázalo, že její technologická vybavenost nevyhovuje potřebám řešení. [4]



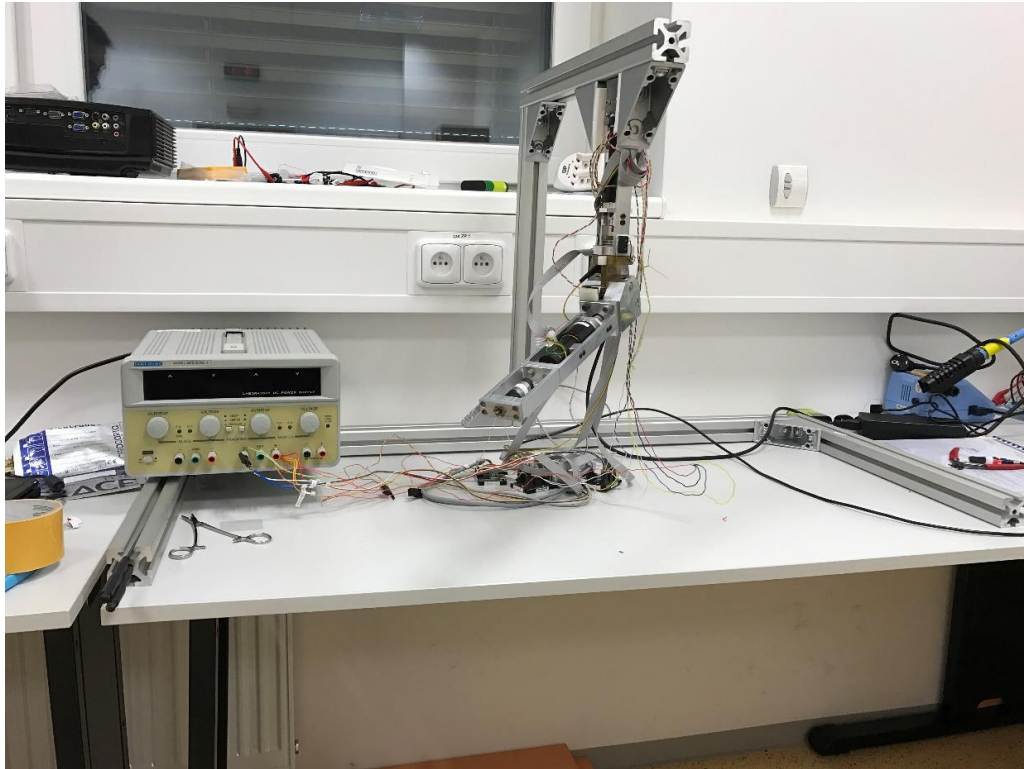
Obrázek 3: Schéma sestavení robotické ruky

Pro zhotovení vybrané konstrukce bylo použito šest servomotorů Hitec HS422. Čtyři servo motory na paži umožňují pohyb ve třech stupních volnosti, tedy v ose x (pohyb doleva a doprava), pro tento pohyb je k dispozici dvojice servo motorů pro dosažení větší flexibility. Další dva servo motory umožňují pohyb v ose y. Zbývající dva servomotory se starají o pohyb konce ruky, a to, o rotaci a sevření.

Důvod pro nahrazení této konstrukce byl ten, že k jejímu sestavení byly použity analogové servomotory, které dokáží vykonávat pohyby pouze sériově (pokyny jsou vykonávány v pořadí ve kterém dorazí). Pro „přirozený“ pohyb byla potřeba, aby se pohyby vykonávali paralelně (konstrukce dokáže pohybovat více směry současně) a k tomu bylo potřeba využít digitální servomotory, kterými je vybavena konstrukce použitá ve finální verzi řešení.

2.2.3 Robotická paže ze součástek maxon motor

Robotická paže se nachází v prostorách fakulty v laboratoři A114 a byla zhotovena v rámci diplomové práce Bc Jana Froňka na zakázku švýcarské společnosti maxon motor. Tato paže byla zvolena po konzultaci s vedoucím projektu, po zjištění, že dříve zvolená stavebnice Rascal nedokáže splnit požadavky projektu. [5]



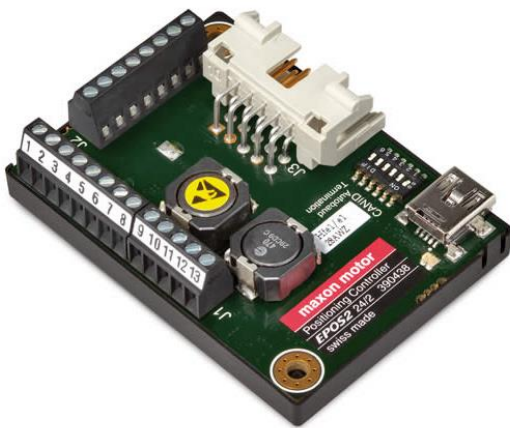
Obrázek 4: Robotická paže ze součástek maxon motor

Paže je tvořena třemi servomotory nesoucích katalogové označení 339152 (2x) a 110075, jejichž kombinací lze dosáhnout plnohodnotného pohybu v prostoru v rozsahu 180°. Pohyb ruky je omezen zapojenými kabely, které se při větším natočení zamotávají a mohlo by docházet k poškození konstrukce, proto jsou na ruce nainstalovány koncové spínače, jejichž signál je veden jako DI do jednotky EPOS, která pohyb zastaví, aby nedošlo k poškození paže.

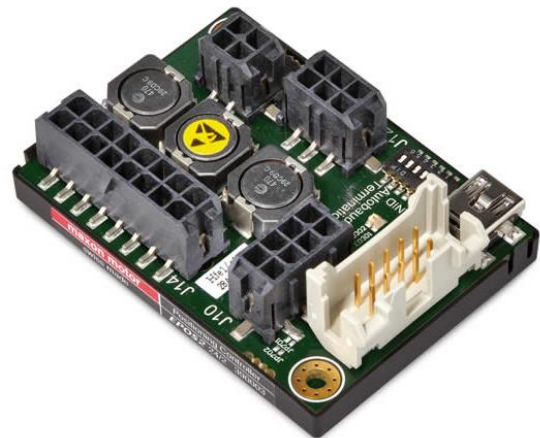
Paže se ovládá pomocí programu běžícího na připojeném osobním počítači skrze rozhraní USB. Pro správnou funkčnost komunikace počítače a EPOS kontrolérů je zapotřebí mít nainstalované ovladače, které lze stáhnout ze stránek výrobce.

O správný pohyb paže se stará trojice modulárních, pozičních digitálních EPOS kontrolérů, které jsou vhodné pro magneticky aktivované motory i enkodéry se stálým výstupem proudu v rozmezí 1 až 750 W. Velká možnost operačních módů, a také mnoha odlišných příkazů použitého rozhraní, dělá z těchto kontrolérů všestranné použití v mnoha rozdílných řídicích systémech na poli automatizovaných technologií a mechatroniky. [6]

Pro vytvoření použité konstrukce, byla použita trojice EPOS kontrolérů, konkrétně jeden hlavní kontrolér EPOS 24/2 DC a dva kontroléry EPOS 24/2 DC/EC. Kontrolér EPOS 24/2 DC je modulární, digitální poziční kontrolér, fungující s kartáčovými DC motory s enkodéry až do 48 wattů. Kontrolér EPOS 24/2 DC/EC je také modulární, digitální, poziční kontrolér, fungující s enkodéry nebo bezkartáčovými EC motory s Hallovými senzory a enkodéry až do 48 wattů



Obrázek 5: Kontrolér EPOS 24/2 DC



Obrázek 6: Kontrolér EPOS 24/2 DC/EC

2.3 Robotická paže ovládána myšlenkami

V této kapitole je popsán existující případ, kdy bylo docíleno sestrojení funkční robotické paže s využitím náramku Myo. [7; 8]

V roce 2005 byl Johnny Matheny diagnostikován s rakovinou levého předloktí a v roce 2008 mu byla amputována levá paže až do poloviny humeru. Posléze v roce 2015 podstoupil operaci v Johns Hopkins Hospital a následně začal spolupracovat s Johns Hopkins Applied Physics Laboratory. [9]

Johnny podstoupil jako první operaci cílenou reinervaci svalů (tzv. TRM), při které došlo k přeřazení nervů ruky, které dříve kontrolovaly paži a ruku, aby je bylo možné využít k ovládání pokročilých protéz. Následovala další operace, nazývaná Osseointegrace, během které mu byl do dřene kosti amputačního pahýlu implantováno kompresní zařízení, čímž bylo docíleno silného spojení zařízení a kosti. V druhé fázi došlo k nastavení implantátu skrze kůži, čímž bylo zajištěno možnosti napojení protézy. Vývod napojení protézy mimo tělo pacienta vyřešil nejdůležitější otázku napojení protézy k tělu. V případě, že protéza nepasuje pacientovi dokonale na amputační pahýl, můžou se vyskytnout problémy jako je bolest, odřeniny, vředy, puchýře a další komplikace.

Protéza vyhotovená v rámci této spolupráce je momentálně asi nedokonalejší myoelektrická protéza, která existuje.



Obrázek 7: Johnny Matheny s nasazenou protézou

3 Cíle práce

Jako hlavní cíle bakalářské práce, byli identifikovány:

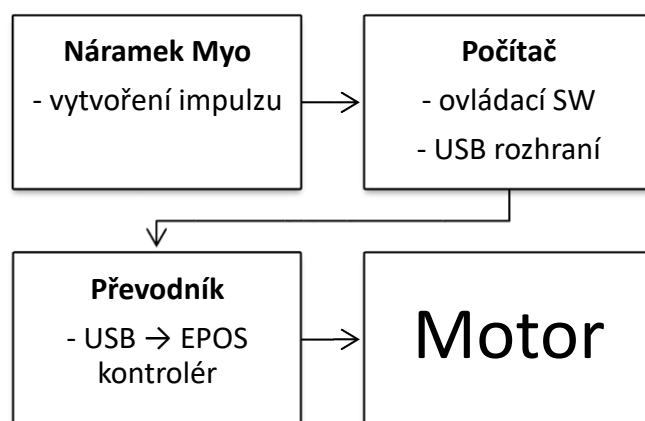
- Seznámením se s poskytnutým hardwarem, náramek Myo a robotické paže, a nastudování existujících software řešení, a jejich dokumentací
- Navržení možných způsobů ovládání a zhodnocení jejich praktického nasazení, zdůvodnění výběru
- Implementace komunikace náramku Myo s programem pro ovládání robotické paže
- Vytvoření dvou způsobů ovládání robotické paže s využitím dat získaných z náramku. Konečné ovládání bude uskutečňováno pomocí předdefinovaných gest či EMG obálky

Na závěr je potřeba zhodnotit výsledky, zda vyhovují zadání projektu.

4 Návrh aplikace

4.1 Analýza požadavků

Nejdůležitějším požadavkem na vytvářenou aplikaci bylo, aby vyhovovala potřebám projektu a to tak, že po vytvoření impulsu (aktivita svalů) dojde k rozpohybování robotické paže. Následující diagram naznačuje průběh celého procesu funkčnosti aplikace.



Obrázek 8: Schéma komunikace mezi jednotlivými prvky

Bylo zapotřebí vytvořit komunikaci mezi náramkem Myo a počítačem. Následná komunikace mezi počítačem a kontroléry EPOS probíhá přes rozhraní USB a je řešena v dodaném demo programu, který byl dodán společně s robotickou paží, pro bakalářskou práci poskytl vedoucí.

4.2 Technické specifikace

Pro potřeby projektu byl použito hardwarové vybavení, které bylo blíže popsáno v kapitolách 2.2.1. a 2.2.3. Napájení paže je zajištěno laboratorním zdrojem energie MPS-3005L-3 od společnosti Matrix, při nastaveném napětí 12 V.

Pro implementaci propojení náramku Myo a PC, bylo použito vývojové prostředí od společnosti Microsoft „Visual Studio 2015“ a programovací jazyk C#. Pro vizuální kontrolu získaných dat bylo použito prostředí Matlab.

4.3 Funkční specifikace

Během začátků seznamování se s hardwarem vnikala velká diskuze, které technologie by byli pro ovládání robotické paže nejvhodnější, tak aby výsledný produkt mohl fungovat jako plnohodnotná myoelektrická protéza.

Jak již bylo popsáno v kapitole 2.2.1. náramek Myo disponuje mnoha moderními technologiemi, a všechny by mohli být použity ve finálním řešení projektu. Proto se vyskytla otázka, které bude nejlepší použít.

V počátcích jsem pracoval s plánem využít data získaná z gyroskopu a využít poziční mód kontrolérů EPOS. V takovémto případě by bylo potřeba transformovat souřadný systém robotické paže vůči souřadnému systému náramku. Byla zde myšlenka vycházet z bakalářské práce Václava Šípka, který řešil obdobný problém s kamerou Kinect. Toto řešení bylo nakonec zavrhnuto, jelikož výchozí souřadný systém náramku je velmi proměnlivý. Nastavuje se vždy při synchronizaci, kterou je uživatel nucen provést vždy při nasazení náramku a k dosažení konstantních výsledků bylo zapotřebí mít vždy náramek nasazený přesně (stejně natočení kolem ruky) a současně ruka musela být vždy ve stejné poloze. Toto bylo otestováno skrze virtuální simulaci paže, kterou umožňuje Myo SDK MATLAB MEX wrapper, se kterým se během práce pracovalo. Další důvod, proč bylo toto řešení zavrhnuto, byl ten, že program pro ovládání robotické paže je napsán v jazyce C# a výpočty by probíhali v prostředí Matlab, konečná komunikace mezi oběma prostředím by sice byla možná, ale složitá a nemusela by vést k uspokojivému výsledku, proto bylo toto ovládání zamítnuto.

Následně se řešení přesunulo možnosti využít EMG senzory. Díky volbě tohoto řešení zanikla nutnost přepočítávat souřadné systémy, a tak jsem veškeré řešení vytvářel v jazyce C#. Po učinění tohoto rozhodnutí a diskuzi s vedoucím bakalářské práce jsme došli k závěru, že konečným řešením projektu budou dva možné způsoby ovládání, které jsou detailně popsány v zadání této bakalářské práce.

V případě využití možnosti ovládat náramek gesty, je nastavena konstantní rychlost pohonu, dokud gesto trvá, ale pokud je využito řízení pomocí EMG obálky, jsou gesta využívána pro zvolení pohodu, a následně kontrakcí svalu je pohon rozpořehován rychlostí, která je přímo úměrná síle zatnutí svalu.

5 Implementace

V této kapitole je popsáno, jakým způsobem jsem postupoval při vypracování zadání bakalářské práce.

Vedoucí projektu mi poskytl základní program pro ovládání robotické paže, který bylo pro potřeby projektu nutno upravit z pozičního módu do módu rychlostního / momentového, toho jsem docílil po nastudování dokumentace, kterou jsem též měl k dispozici a konzultaci s konzultantem práce. Po této úpravě jsem se posunul k návrhu propojení programu s náramkem Myo. [10; 11; 12]

Společnost Thalmic Labs, výrobce náramku, poskytuje na svých stránkách volně stažitelnou knihovnu funkcí, určenou pro vytváření vlastních programů pro náramek Myo. Knihovna je napsána v programovacím jazyce C++. Preferuji programovací jazyky C# a Matlab, se kterými jsem se setkal během studia, a proto jsem hledal existující řešení, která by mi usnadnila práci na vyřešení projektu.

Po hledání na vývojářském fóru výrobce a odkazů na webovou repositář „www.github.com“ jsem našel několik existujících řešení, která dle dostupné dokumentace vyhovovali požadavkům. Zaměřil jsem se především na Wrapper knihovny. Tyto knihovny se skládají z tenké vrstvy kódu (tzv. shim), který překládá existující rozhraní knihovny (v mém případě z jazyka C++) do podporovaných rozhraní (mým požadavkem byl programovací jazyk, se kterým jsem již měl zkušenosti). [10]

5.1 Myo SDK MATLAB MEX Wrapper

První wrapper, na který jsem narazil obsahuje metody, které vhodně přeloží knihovnu náramku Myo do prostředí programu Matlab. [11]

Balíček obsahuje zjednodušenou třídu MyoMex, která umožňuje uživateli využívat streamování dat z jednoho nebo dvou náramků při frekvenci 50 Hz (sledování IMU dat), případně při frekvenci 200 Hz (sledování EMG dat). Sledování EMG dat je možné pouze v případě použití jednoho náramku, z důvodu hardwarových / softwarových omezení. Veškerá tato funkcionalita je docílena pouze jedním příkazem.


```

mm = MyoMex(); // Konstruktor, začátek sběru dat myoData
mm = mm.myoData; // Získání objektu myoData

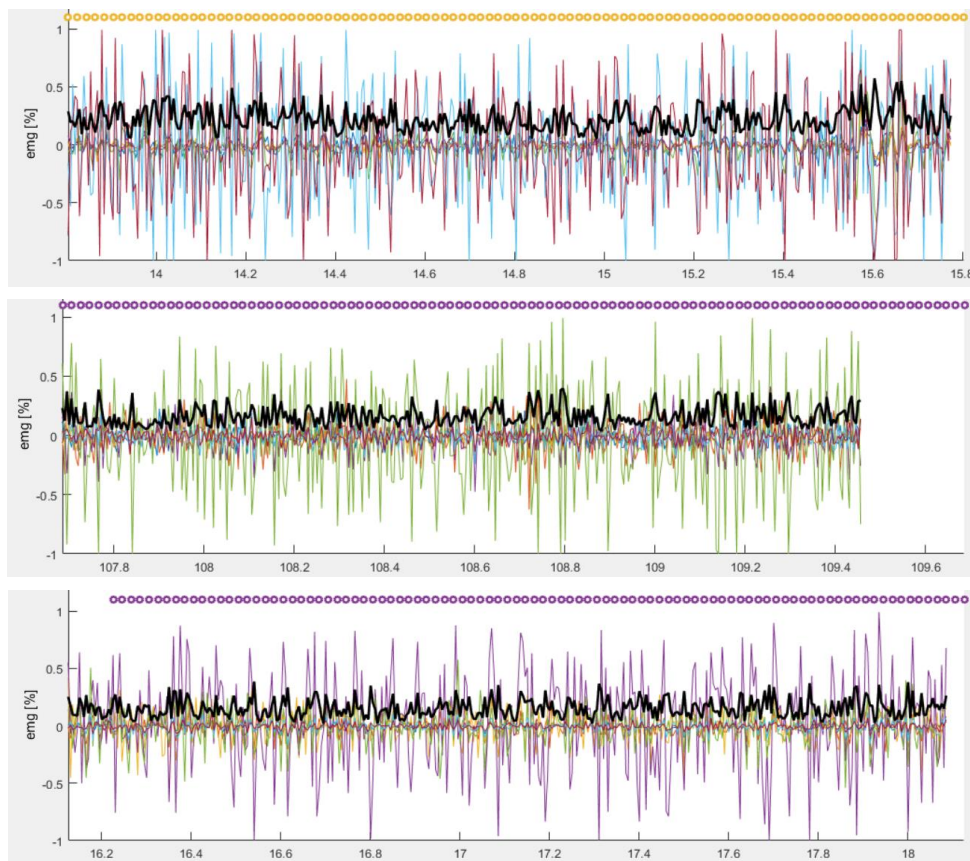
// Data jsou zaznamenána do logu proměnné m (př. quat_log,
// gyro_log, accel_log, emg_log atd.)

```

Obrázek 9: Konstrukce / zobrazení dat MATLAB wrapperu

Data získaná z IMU se skládají z odhadovaných kvaternionů (určení polohy v prostoru), tříosého gyroskopu (získání úhlové rychlosti) a tříosého akcelerometru (získání lineárního zrychlení). EMG senzory poskytují osm kanálů dat (každá elektroda vysílá vlastní kanál) a navíc vestavěnou detekci gest ruky.

Wrapper nebyl ve finálním řešení použit, ale používal jsem ho během celého procesu řešení, jelikož v něm jsou přehledně zobrazená data z EMG senzorů. Zobrazení bylo použito pro další plánování finálního ovládání.



Obrázek 10, 11, 12: Demonstrace MATLAB wrapperu pro vizualizaci dat

Na obrázcích 10, 11 a 12 je zachyceno jedno řešení, kdy jsem se úspěšně snažil rozlišit jednotlivé svaly a evokovat vždy jen jednu skupinu svalů pro podráždění jednoho EMG senzoru, kdy následný pohyb slouží pro řízení pomocí myopotenciálové obálky.

5.2 C# Wrapper for Myo Armband

Druhý wrapper, jež vyhovoval mým potřebám a následně byl u použit do konečného řešení projektu. [12]

Wrapper vytváří objektově orientované API na vysoké úrovni v programovacím jazyce C# a je kompatibilní s .NET 2.0+. Využívá technologii Code Contract, díky níž je docíleno čistějšího API pro uživatele. Součástí wrapperu jsou tři ukázkové úlohy, na kterých jsou demonstrovány základní funkcionality funkcí. [13; 14]

5.3 Vlastní implementace

Jak již bylo popsáno v předešlých kapitolách, tak jsem pro svůj způsob řešení využil mnou upravený demo program do jiného pohybového módu a přeloženou knihovnu náramku Myo.

Začal jsem studovat funkce knihovny náramku a prohledával jsem vývojářské fórum, pro možné reference. Narazil jsem na obdobný problém, ohledně začlenění komunikace náramku do Windows form aplikace, který řešil jiný uživatel fóra. Na jeho problém odpověděl uživatel „tobyuct“, který nahrál část kódu, konkrétně funkce pro napojení náramku do WPF aplikace. Tento kód jsem převzal, uživatel vyjádřil souhlas s využitím kódu jinými uživateli, a patřičně jsem jej rozšířil tak aby vyhovoval mému zadání. [15]

Během tvorby mé problematiky jsem z důvodu použitého wrapperu z kapitoly 5.2, vytvořil více vláknovou aplikaci, tento pojem jsem měl v povědomí, ale během studia jsem jej nemusel nikdy řešit, a tak jsem začal studovat tvorbu více vláknových aplikací. Tato problematika je stručně popsána v kapitole 5.4. Po vyřešení problému s více vlákny jsem aplikaci začal výrazně upravovat tak, aby vyhovovala dnešnímu programátorskému standartu. Pro toto řešení jsem použil návrhový vzor MVC. Stručný popis návrhových vzorů a konkrétně mnou použitého vzoru je v kapitole 5.5.

5.4 Více vláknové aplikace

Tato kapitola popisuje problematiku více vláknových aplikací [16]

5.4.1 Teorie

Vlákno v informatice označuje vlákno výpočtu neboli samotný výpočetní tok, tedy posloupnost po sobě jdoucích operací. Každá spuštěná aplikace má alespoň jedno vlákno, ve kterém počítá. Dříve platilo, že proces měl je jedno vlákno (přesněji, nebyl důvod tyto pojmy odlišovat). Dnes je stále více programů více vláknových, tedy uvnitř jednoho procesu (a v jednom adresovém prostoru, tedy se sdílenou pamětí může zároveň běžet více vláken.

Rozdíly mezi více vláknovým a multiprocessorovým multitaskingem jsou kromě sdílené paměti a režie při přepínání:

- Přepnutí mezi vlákny bývá výrazně rychlejší
- V některých případech není třeba při přepínání vláken volat jádro OS
- Rychlejší může být i vytváření a rušení vlákna
- Vlákno spotřebuje méně paměti (důležité pro aplikace se stovkami a více vláken)

Vlákna je možné vytvářet i čistě aplikačně bez OS (například pokud podporu více vláken nemá). Takto vzniklá vlákna je poté možné spouštět postupně v jednom vlákně operačního systému nebo takzvaně m:n, tedy v několika vláknech OS současně spouštět větší počet aplikačních vláken. Toto řešení sice není tak dobré jako řešení s podporou operačního systému ale pro některé úlohy může být stále rychlejší.

Typickými aplikacemi, kde dochází k využití více vláken jsou:

- Servery – obsluha více klientů najednou
- Výpočetní aplikace – výpočet lze urychlit rozdělením úlohy na více procesů
- Aplikace reálného času – lze využít specifických rozvrhovačů. Více vláknová aplikace je výkonnější než složité asynchronní programování, neboť vlákno čeká na příslušnou událost namísto přerušování vykonávání kódu a přepínání kontextu

5.4.2 Využití vláken

Vlákna je výhodné použít, pokud aplikace splňuje některé následující kritérií:

- Je složena z nezávislých úloh
- Může být blokována po dlouhou dobu
- Obsahuje výpočetně náročnou část
- Musí reagovat na asynchronní události
- Obsahuje úlohy s nižší nebo vyšší prioritou než zbytek aplikace

5.4.3 Nevýhody vláken

Využití vláken by mělo být použito jen v případech, kde to je opravdu nutné, protože tvorba vláken je na určitých platformách poněkud zdoluhavá operace. Každé vlákno vytváří v paměti vlastní zásobník, do kterého jsou ukládány mezivýsledky, stavy proměnných, adresy apod. Maximální počet vláken je omezen platformou. Při využití vláken dochází ke zvýšení složitosti kódu a je daleko složitější sledovat tok programu. K řízení sdílení prostředků se používá synchronizace, ta zaručuje, že stav dat nebude změněn z více vláken současně.

5.4.4 Řešená situace

Podstata vzniklého problému, kvůli které bylo potřeba podrobněji nastudovat problematiku více vláknového programování, spočívala v tom, že při zpuštění programu pro ovládání paže se vytvořilo jedno vlákno akcí, ale použitý wrapper z kapitoly 5.2 při volání svých funkcí vytváří nové vlákno. Tudíž došlo ke kolizi, a byla potřeba vytvořit takovou metodu, aby spolu vlákna komunikovali, konkrétně, aby vlákno náramku bylo schopné poskytovat hodnoty do vlákna paže. Pro toto řešení jsem vytvořil metody dle referencí k Visual studiu. Vytvořená metody jsou zobrazeny na obrázku 13.

```

// Nastavení hodnot pro motůrek a rozpořybování
public void invokerMove(string NodeNum, string speedValue)
{
    view.Invoke((MethodInvoker)delegate ()
    {
        view.getBoxNodeId().Text = NodeNum;
        view.getBoxTargetPosition().Text = speedValue;

        buttonEnable_Click();
        buttonMove_Click();
    });
}

// Vypnutí motůrtku
public void invokerDisable()
{
    view.Invoke((MethodInvoker)delegate ()
    {
        buttonDisable_Click();
    });
}

```

Obrázek 13: Metody invoker pro komunikaci mezi vlákny

5.5 Návrhové vzory

Při objektově orientovaném programování často dochází k nutnosti řešit problematiku obecnou metodikou tak, aby i jiní programátoři měli snadnější cestu k pochopení kódu, tyto metodiky lze označit jako Best practise, tedy takové postupy, které jsou lety prověřeny a které zajistí, jak lepší čitelnost kódu, tak i to, že pro některé době nebudeme muset svůj kód přepisovat z důvodu chyb v návrhu. Jedná se o návrhové vzory. [17]

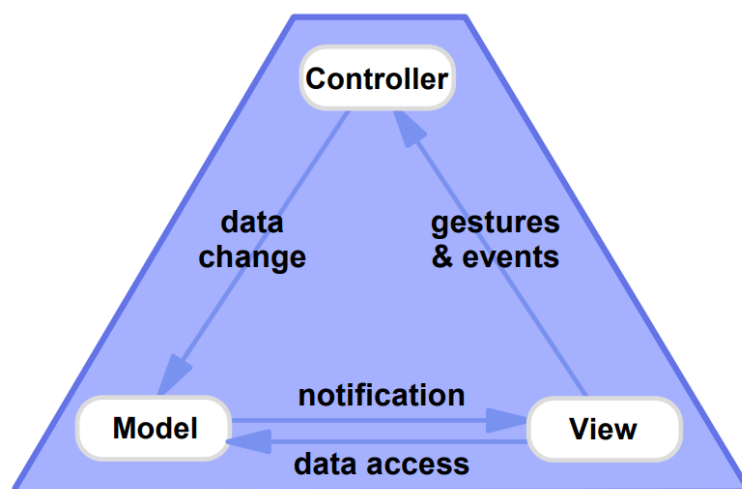
Existuje velké množství návrhových vzorů a neustále vznikají další. Pro úpravu vytvořené aplikace byl použit návrhový vzor MVC.

5.5.1 Návrhový vzor MVC

Návrhový vzor MVC (Model – View – Controller) je spíše softwarová architektura, která rozděluje aplikaci do tří vrstev: datový model, uživatelské rozhraní a řídicí logika. Přičemž modifikace některé z nich má pouze minimální vliv na ostatní. [18; 19]

Každá část kódu aplikace spadá do jedné z kategorií MVC vzoru, který říká, že je nutné tyto části oddělit od samotných komponent nebo modulů. V praxi jde o velmi užitečný přístup, kdy je použití tohoto vzoru nezbytné pro udržení přehledného kódu.

Na obrázku 14 je znázorněno schéma vztahu jednotlivých komponent.

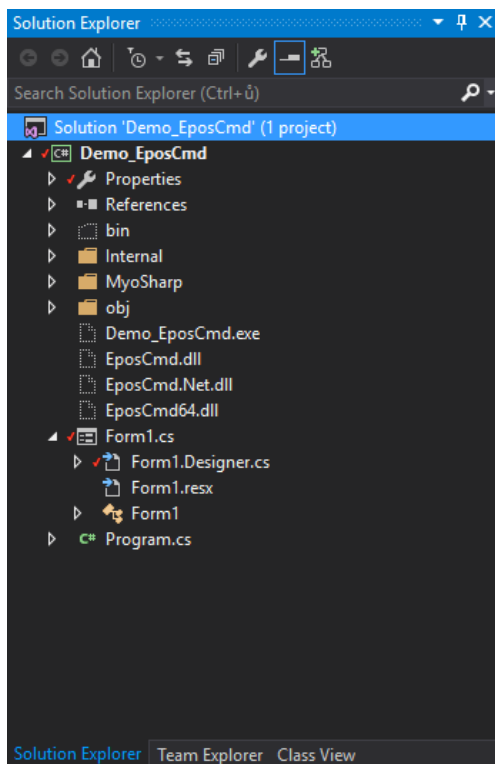


Obrázek 14: Schéma MVC vzoru

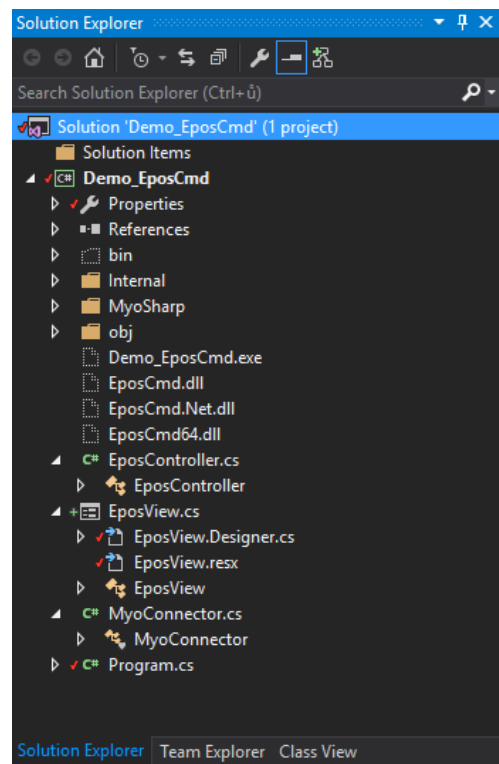
Jednotlivé komponenty jsou:

- Model – zajišťuje přístup k datům a manipulaci s nimi
- View (pohled) – převádí data reprezentovaná modelem do podoby vhodné k prezentaci uživatele
- Controller (řadič) – reaguje na události pocházející od uživatele a zajišťuje změny modelu nebo pohledu

Na obrázcích 15 a 16 je zobrazeno, jak vypadalo zobrazení Solution exploreru vytvářené aplikace před a po aplikování MVC modelu.



Obrázek 15: Solution explorer aplikace bez MVC vzoru



Obrázek 16: Solution explorer aplikace s MVC vzorem

Na přiložených obrázcích je použitý model reprezentován třídami EposController (Controller), EposView (View) a MyoConnector (Model). Pro lepší porozumění je potřeba prostudovat samotný kód, který je patřičně okomentován a uložen na přiloženém CD. Ukázka připojení náramku je ukázána v kapitole „Přílohy“, kde je popsáno, kde je možné měnit kód pro zachování funkčnosti.

6 Uživatelská dokumentace

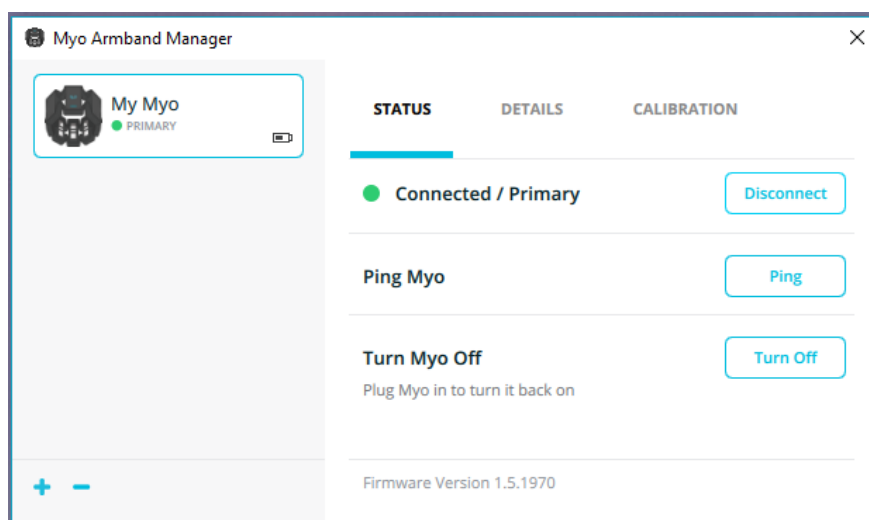
Pro otestování vytvořených programů je potřeba mít nejprve nainstalované příslušné ovladače:

- Myo Connect
- EPOS USB driver

Zmíněné ovladače jsou volně přístupné na stránkách výrobců a stačí je jen nainstalovat. Ovladače jsou uloženy na přiloženém CD ve složce „Ovladače“. Dále je potřeba mít nainstalovaný některý program, ve kterém bude možnost otevřít projekt a zkompilovat vytvořený program v mém případě je použito dříve zmíněné Visual Studio 2015 32-bit.

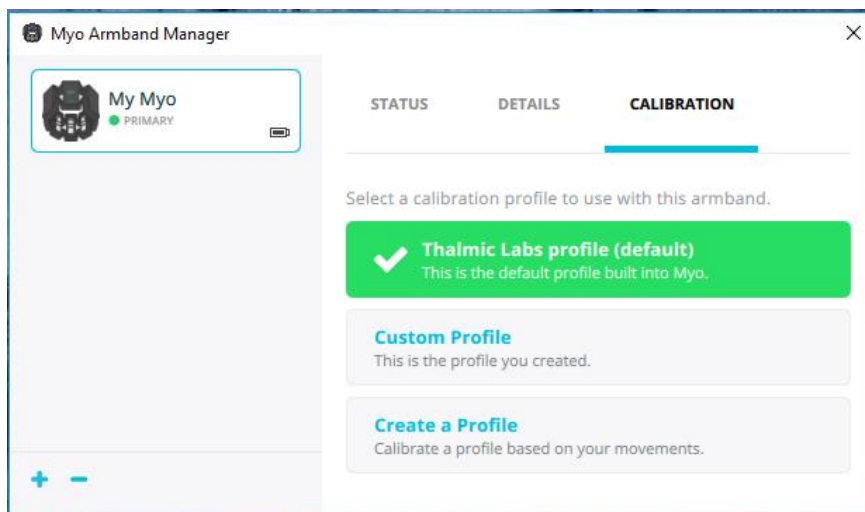
6.1 Připojení a kalibrace náramku Myo

Při prvním připojení náramku k osobnímu počítači je uživatel proveden kalibrací náramku, kdy mu jsou vyobrazena přednastavená gesta a uživatel je provádí a následně ukládá, tento krok je možné přeskočit a začít využít přednastavený základní profil, který dodaný software nabízí, ale je doporučeno si všechna gesta personalizovat pro docílení lepší funkčnosti náramku. Postup kalibrace je popsán na následujících obrázcích.



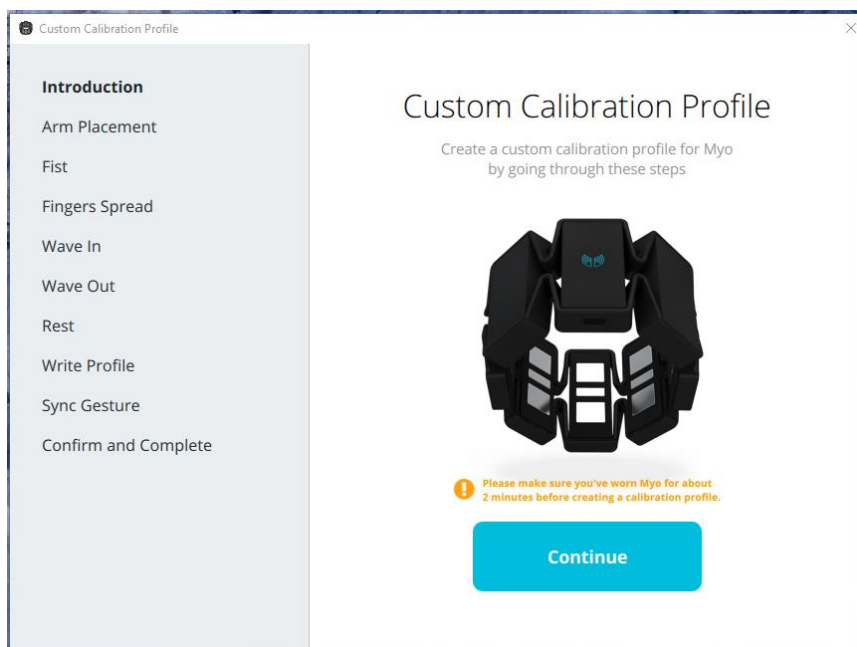
Obrázek 17: Myo Connect – UI

Po zapnutí programu Myo connect se zobrazí okno, které je zobrazeno na obrázku 17. Pro kalibraci je nutné přejít za záložku „Calibration“, jak je ukázáno na obrázku 18. Pro tvorbu nového profilu je třeba kliknout na položku „Create a Profile“.

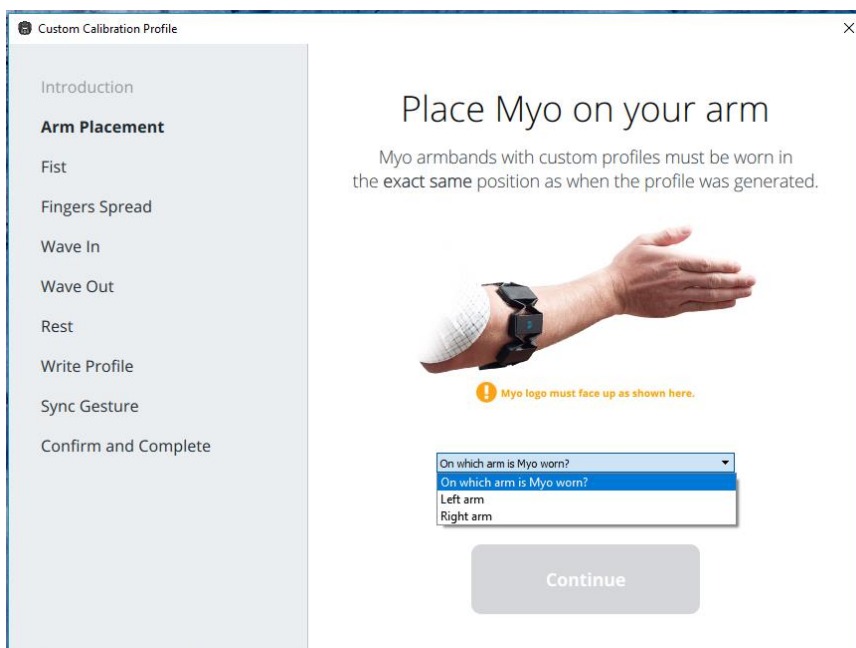


Obrázek 18: Myo Connect – záložka kalibrace

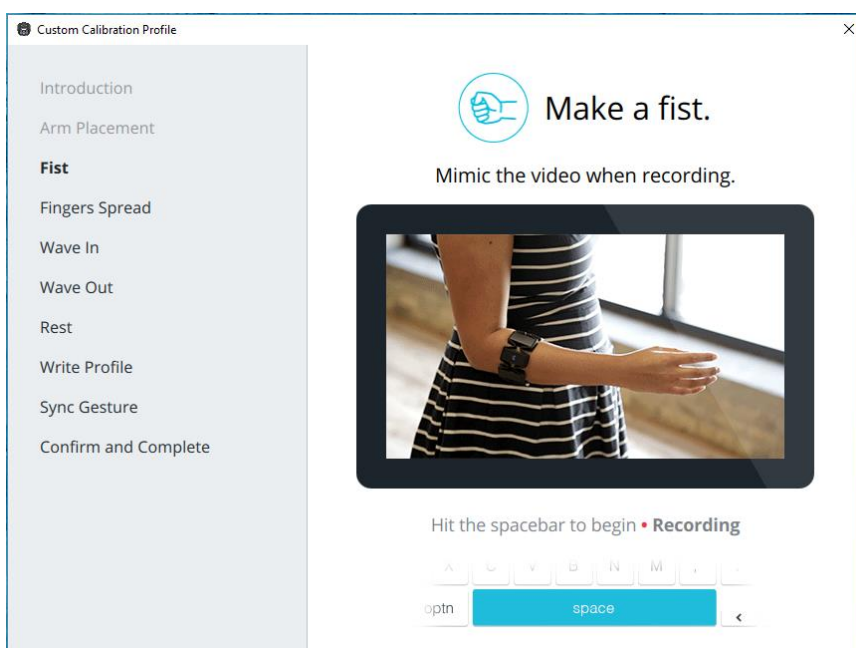
Následně je otevřeno nové okno, které je zobrazeno na obrázku 19. Nyní je uživatel proveden velmi intuitivním návodem, jak nastavit náramek. Tento postup je zobrazen na následujících obrázcích.



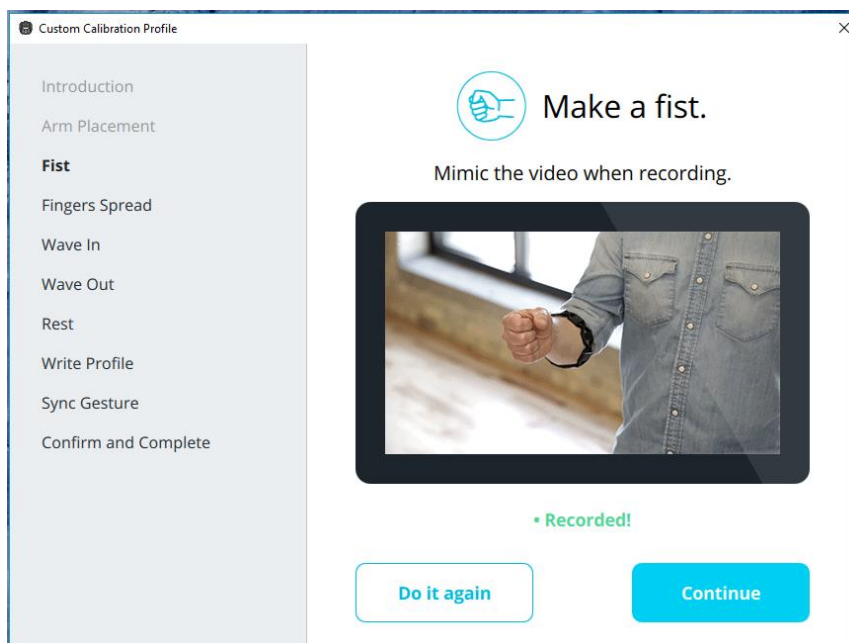
Obrázek 19: Myo Connect – začátek kalibrace



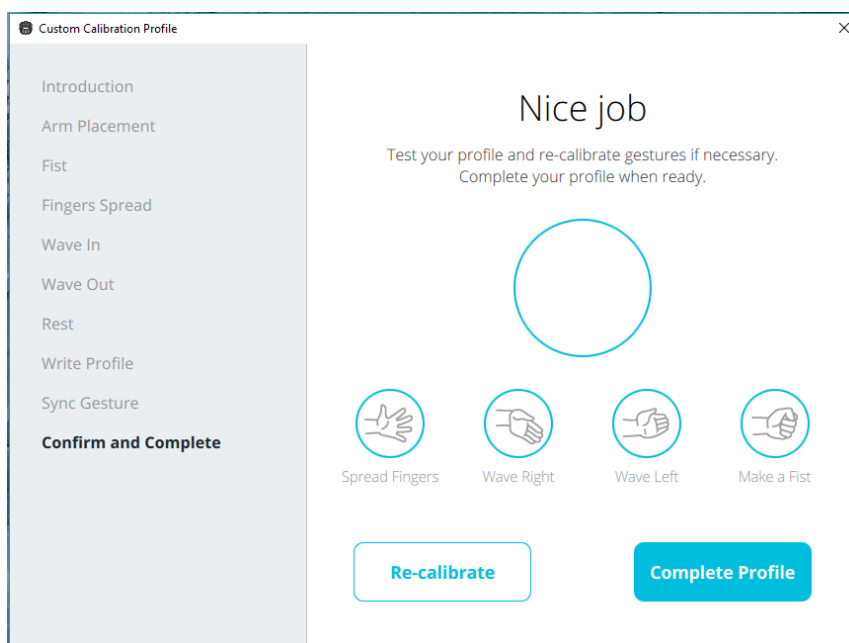
Obrázek 20: Myo Connect – výběr paže pro kalibraci



Obrázek 21: Myo Connect – Výzva pro uchopení gesta



Obrázek 22: Myo Connect – splnění záznamu gesta



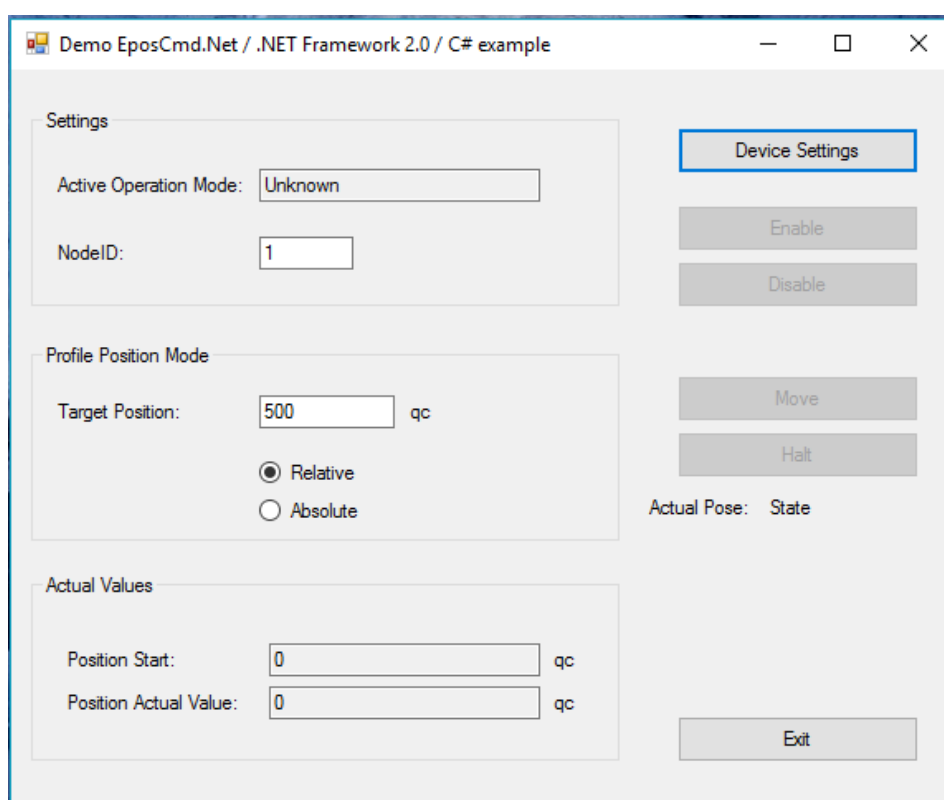
Obrázek 23: Myo Connect – dokončení tvorby profilu

Po dokončení tvorby profilu se zobrazí okno, kde si uživatel může nahraná gesta ozkoušet a v případě nesrovnalostí provést kalibraci od začátku.

6.2 Připojení robotické paže

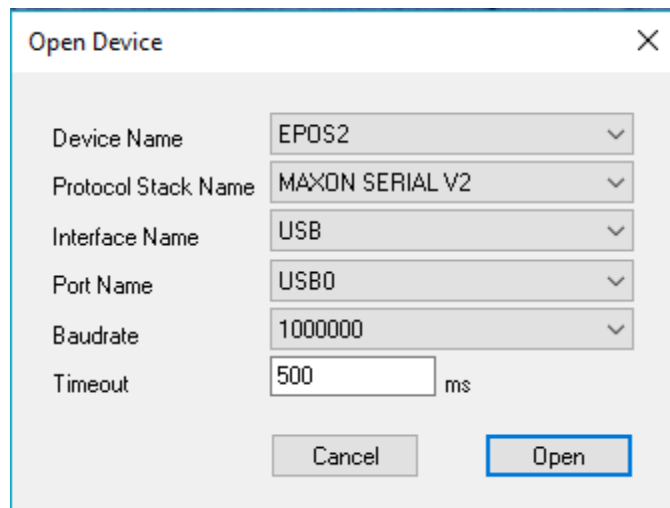
Robotická paže komunikuje s počítačem, na kterém běží program, skrze rozhraní USB. Před spuštěním programu je potřeba mít již nasazený náramek, jelikož během kompilace je vyslán příkaz pro jeho odemčení, aby bylo docíleno co nejnižší prodlevě mezi gestem a následným rozpořbováním motoru paže.

Po spuštění programu je potřeba povolit robotickou paži kliknutím na tlačítko „Device Setting“, zobrazí se okno, které automaticky vyplní potřebné informace v případě, že je paže připojená. Po odsouhlasení paže je možné ji okamžitě začít ovládat pomocí náramku.

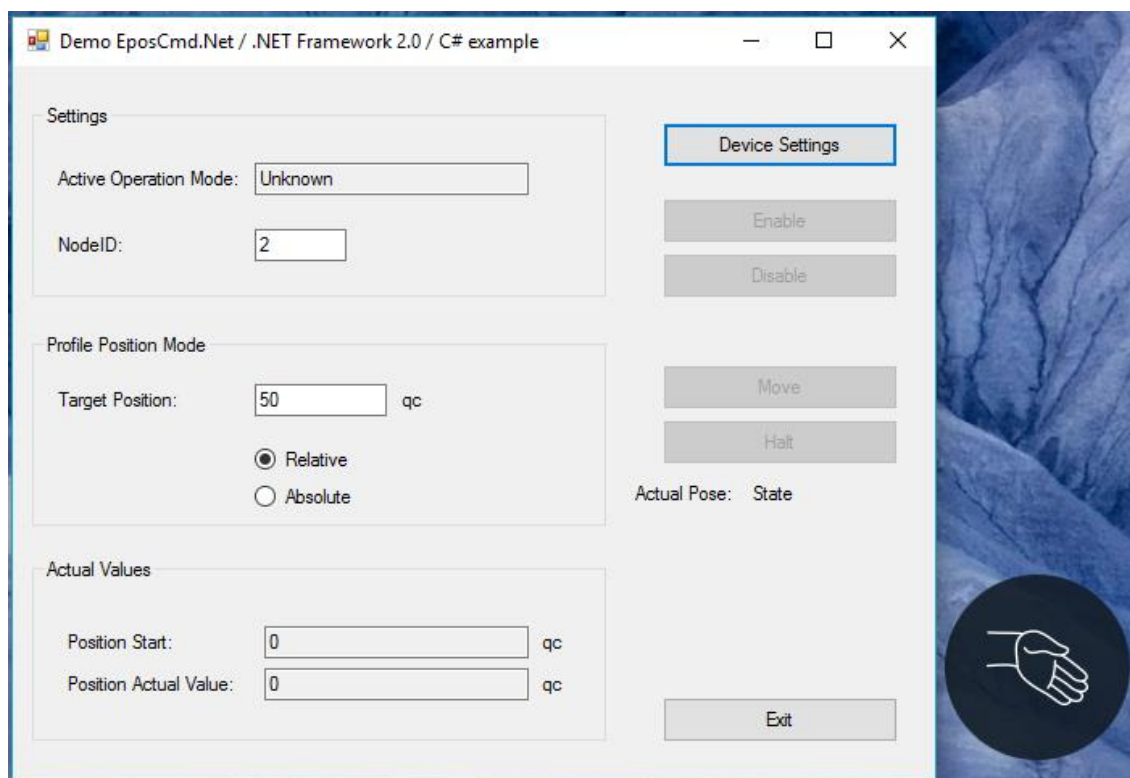


Obrázek 24: Spuštěný program – před povolením paže

Data z náramku přepisují hodnoty v polích „NodeID“ (zvolení požadovaného motůrku a „Target Position“ (nastavení rychlosti).



Obrázek 25: Spuštěný program – výzva pro připojení paže



Obrázek 26: Spuštěný program – reakce programu na data z náramku

Na obrázku 26 je zobrazena reakce programu na vytvořené gesto. Bohužel obrázek není vhodný pro zobrazení real time procesu, kdy se gesta dají měnit velmi rychle.

7 Testování

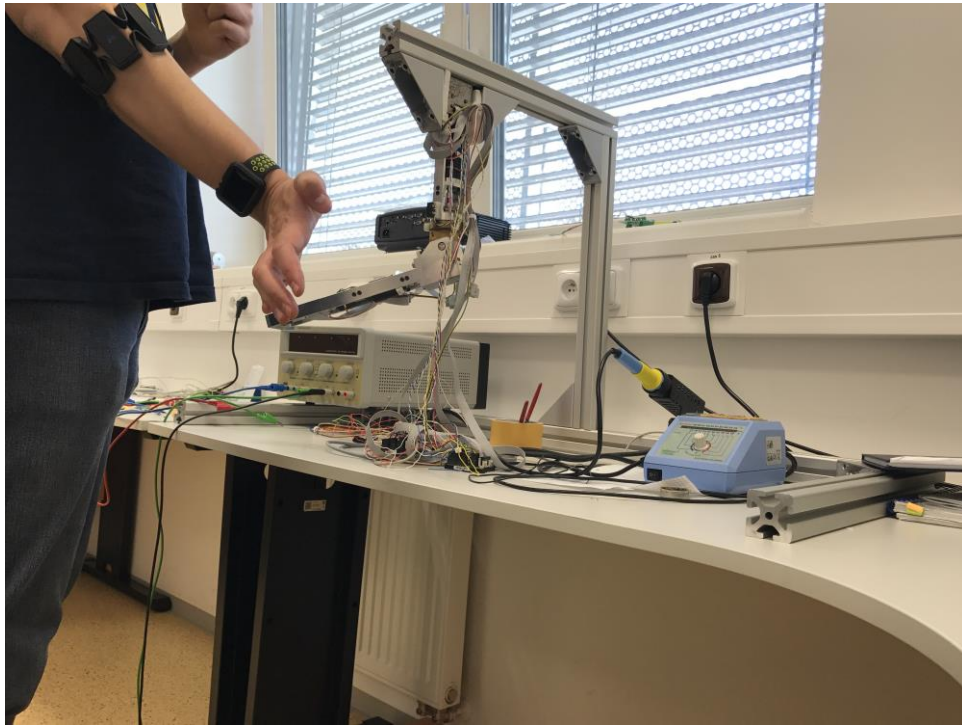
Vzhledem k rozměrům robotické paže nebylo možné provést jiné než laboratorní pokusy, kterým jsem podrobil sebe a své spolužáky. Po spuštění aplikace dochází k bezprostřední reakci na prováděná gesta, kdy využívané pohony jsou přiřazeny dle následující tabulky 5. Takto nastavené hodnoty jsou platné pro oba způsoby ovládání. V případě využití možnosti ovládat paži za pomoci EMG obálky jsou gesta využita pro výběr pohonu a následné rozpohybování regulována dle svalových kontrakcí.

Gesto dle knihovny	Prováděné gesto	Pohon
Double Tap	Dvojklik	Změna směru pohybu
Fist	Sevřená dlaň	1
Wave In	Mávnutí doleva	2
Fingers Spread	Roztažené prsty	3

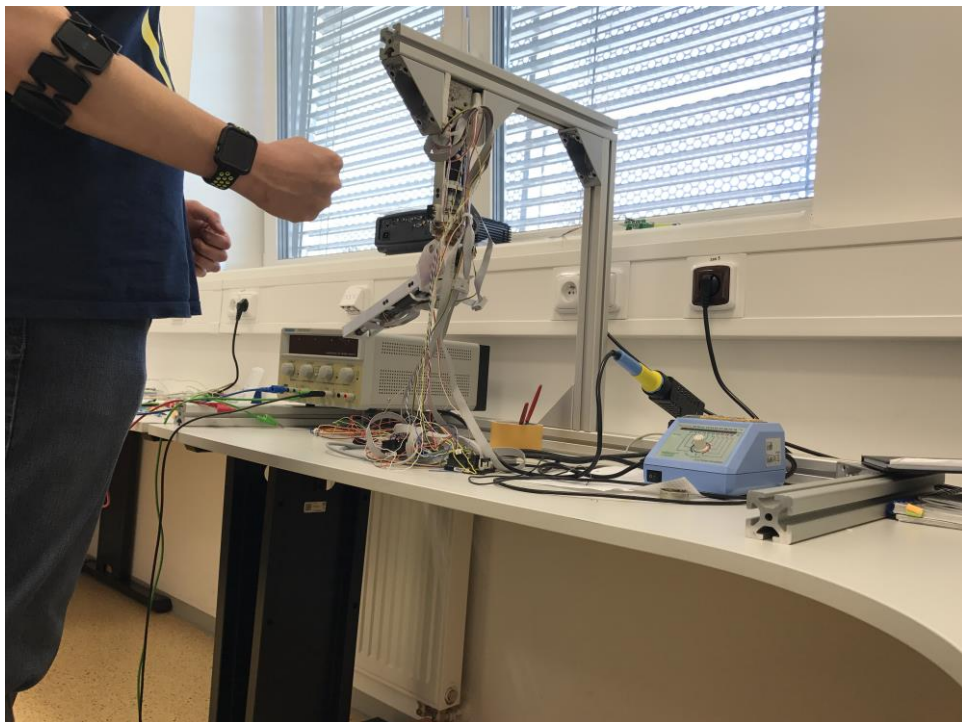
Tabulka 5: Přiřazení gest knihovny náramku k pohonům robotické paže

Vytvořená aplikace funguje spolehlivě a doba odezvy robotické paže (změna pohonu, zastavení pohonu atd.) je velmi rychlá, což je skvělý výsledek pro oblast, na kterou byl během tvorby projektu brán zřetel.

Ve složce „Videa“ na přiloženém CD jsou nahrány záznamy, na kterých je demonstrováno ovládání paže.



Obrázek 27: Záznam testování – gesto 1



Obrázek 28: Záznam testování – gesto 2

8 Diskuse

Po otestování vytvořené aplikace jsem došel k závěru, že náramek Myo je možné využít pro realizaci ovládání robotické paže.

V práci se skrývá velký potenciál pro další směřování ať už bakalářských či diplomových prací. Navazující studenti by mohli rozšířit paži o konstrukci dlaně a správnou detekcí svalových kontrakcí by mohla být vytvořena plnohodnotná myoelektrická paže, která má v porovnání s komerčně nabízenými protézami o mnoho nižší náklady a ovládání nevyžaduje chirurgický zákrok. Dále by práce mohla být rozšířena ve směru napájení, kdy by se pro napájení paže využil akumulátor a robotické paži by se přidala větší mobilita.

Jako menší nedostatek náramku Myo vidím to, že nejsou přítomny žádná fyzická tlačítka, která by umožňovala náramek manuálně zapnout či vypnout. K zapnutí dochází při sebemenším pohybu s náramkem a často se mi stávalo že se náramek vybíjel během přenášení v batohu / zavazadle. Jelikož náramek nedisponuje baterií s příliš velkou kapacitou bylo by v běžné praxi potřeba denní dobíjení, ale tento nedostatek vzhledem k poskytnutí efektivnímu ovládání protéz zanedbatelný a dal by se vyřešit dobíjením z akumulátoru protézy během nošení.

9 Závěr

Cílem této bakalářské práce bylo vytvořit programové vybavení, které by umožňovalo ovládat robotickou paži pomocí dat získaných z náramku Myo a umožnit uživateli dva způsoby ovládání (pomocí gest a EMG obálky), vytvořené ovládání bylo podrobena testování.

Na základě otestování vytvořeného softwaru jsem došel k závěru, že vytvořený software vyhovuje zadání projektu a náramek představuje dobrou a cenově dostupnou možnost, jak ovládat robotické protézy a vytvořit tak zcela nové myoelektrické protézy se vnějším snímáním myopotenciálové aktivity.

Kódy obsažené v této práci jsou distribuovány pod licencí MIT, jejíž plné znění se nachází na přiloženém CD v souboru „Licence.txt“.

Seznam použité literatury

- [1] FRONĚK, Jan. *Kinematický a silový model protézy ruky*. Kladno: ČVUT, 2010. Bakalářská práce, ČVUT, Fakulta biomedicínského inženýrství. Katedra biomedicínské techniky.
- [2] CALIFORNIA INSTITUTE OF TECHNOLOGY, *Mobile dexterous siren degree of freedom robot arm with real-time control system* [online]. Inventor> SERAJI, Homayoun, LIM, David, LEE, Thomas. US 5737500 (A). Int. Cl. B25J9/16; B25J9/18. Appl. No. US19950431361. United States and Trademark Office. Dostupné z:
https://worldwide.espacenet.com/publicationDetails/biblio?DB=worldwide.espacenet.com&II=0&ND=3&adjacent=true&locale=en_EP&FT=D&date=19980407&CC=US&NR=5737500A&KC=A
- [3] Myo Armband product page [online]. [cit. 2017-05-16]. Dostupné z:
<https://www.myo.com>
- [4] Rascal Project Book. [online]. [cit. 2017-05-16]. Dostupné z:
http://www.robix.com/Rascal_Project_Book.pdf
- [5] FRONĚK, Jan. *Analýza a konstrukce protézy paže s využitím inteligentních pohonů*. Kladno: ČVUT, 2012. Diplomová práce, ČVUT, Fakulta biomedicínského inženýrství. Katedra přírodovědných oborů.
- [6] EPOS controller. [online]. [cit. 2017-05-16]. Dostupné z:
<http://www.maxonmotor.com/maxon/view/content/EPOS-Detailsite>
- [7] APL's Modular Prosthetic Limb Reaches New Levels of Operability. [online]. [cit. 2017-05-16]. Dostupné z:
<http://www.jhuapl.edu/newscenter/pressreleases/2016/160112.asp>
- [8] Meet the Man With a Myo-Controlled Robotic Arm. [online]. [cit. 2017-05-16]. Dostupné z:
<http://developerblog.myo.com/meet-the-man-with-a-myo-controlled-robotic-arm/>

- [9] Meet the Man With a Thought-Controlled Robotic Arm. [online].
[cit. 2017-05-16]. Dostupné z:

<https://www.bloomberg.com/news/videos/2015-11-12/meet-the-man-with-a-thought-controlled-robotic-arm>
- [10] EPOS2 Positing Controllers Firmware Specification. [online].
[cit. 2017-05-16]. Dostupné z:

http://www.cetoni.de/fileadmin/user_upload/Documents/Manuals/EPOS2-Firmware-Specification-En.pdf
- [11] EPOS2 Positing Controllers Getting Started. [online]. [cit. 2017-05-16].
Dostupné z:

http://www.maxonmotor.com/medias/sys_master/root/8822983196702/EPOS2-24-2-Getting-Started-En.pdf
- [12] EPOS2 Positing Controllers USB Driver Installation. [online].
[cit. 2017-05-16]. Dostupné z:

http://academy.maxonjapan.co.jp/wp-content/uploads/2012/02/EPOS2_USB_Driver_Installation_E.pdf
- [13] Myo developer forum. [online]. [cit. 2017-05-16]. Dostupné z:

<https://developer.thalmic.com/forums/>
- [14] Myo SDK MATLAB MEX Wrapper. [online]. [cit. 2017-05-16]. Dostupné z:

<https://www.mathworks.com/matlabcentral/fileexchange/55817-myo-sdk-matlab-mex-wrapper>
- [15] C# Wrapper for Myo Armband. [online]. [cit. 2017-05-16]. Dostupné z:

<https://github.com/tayfuzun/MyoSharp>
- [16] Code Contract Download. [online]. [cit. 2017-05-16]. Dostupné z:

<https://marketplace.visualstudio.com/items?itemName=RiSEResearchinSoftwareEngineering.CodeContractsforNET>
- [17] Code Contract. [online]. [cit. 2017-05-16]. Dostupné z:

[https://msdn.microsoft.com/en-us/library/dd264808\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/dd264808(v=vs.110).aspx)

- [18] MyoSharp (ConsoleExample into Windows Form). [online]. [cit. 2017-05-16]. Dostupné z:
<https://developer.thalamic.com/forums/topic/2745/?page=1#post-10632>
- [19] DPG Vlákna. [online]. [cit. 2017-05-16]. Dostupné z:
<http://www.nti.tul.cz/cz/images/3/31/DPG-3.pdf>
- [20] Návrhové vzory. [online]. [cit. 2017-05-16]. Dostupné z:
<https://www.algoritmy.net/article/51224/Navrhove-vzory>
- [21] Model-View-Presenter (MVP). [online]. [cit. 2017-05-16]. Dostupné z:
<https://doc.nette.org/cs/0.9/model-view-presenter>
- [22] MVP: Model-View-Presenter. The Taligent Programming Model for C++ and Java. [online]. [cit. 2017-05-16]. Dostupné z:
<http://www.wildcrest.com/Potel/Portfolio/mvp.pdf>
- [23] SCIAVICCO Lorenzo, SICILIANO Bruno, Modelling and control of robot manipulators, ed. 1, Springer – Verlag London Limited, 2001, ISBN 1-85233-221-2
- [24] Sachseln (Switzerland), [online], 2011, [citováno 2011], Dostupné z:
<http://www.Maxonmotor.ch/ch/en/index/htm>
- [25] B EKEY, George A, Autonomous robots: from biological inspiration to implementation and control, ed. 1, The MIT Press, 560 s., ISBN 978-0262025782

Seznam obrázků

Obrázek 1: Náramek Myo [3]	11
Obrázek 2: Vizualizace podporovaných gest [3].....	12
Obrázek 3: Schéma sestrojení robotické ruky [4]	13
Obrázek 4: Robotická paže ze součástek maxon motor	14
Obrázek 5: Kontrolér EPOS 24/2 DC [6]	15
Obrázek 6: Kontrolér EPOS 24/2 DC/EC [6].....	15
Obrázek 7: Johnny Matheny s nasazenou protézou [7].....	16
Obrázek 8: Schéma komunikace mezi jednotlivými prvky	18
Obrázek 9: Konstrukce / zobrazení dat MATLAB wrapperu.....	21
Obrázek 10: Demontrace MATLAB wrapperu pro vizualizaci dat	21
Obrázek 11: Demontrace MATLAB wrapperu pro vizualizaci dat	21
Obrázek 12: Demontrace MATLAB wrapperu pro vizualizaci dat	21
Obrázek 13: Metody invoker pro komunikaci mezi vlákny	25
Obrázek 14: Schéma MVC vzoru [22]	26
Obrázek 15: Solution explorer aplikace bez MVC vzoru	27
Obrázek 16: Solution explorer aplikace s MVC vzorem	27
Obrázek 17: Myo Connect – UI	28
Obrázek 18: Myo Connect – záložka kalibrace.....	29
Obrázek 19: Myo Connect – začátek kalibrace	29
Obrázek 20: Myo Connect – výběr paže pro kalibraci.....	30
Obrázek 21: Myo Connect – Výzva pro uchopení gesta	30
Obrázek 22: Myo Connect – splnění záznamu gesta	31

Obrázek 23: Myo Connect – dokončení tvorby profilu	31
Obrázek 24: Spuštěný program – před povolením paže.....	32
Obrázek 25: Spuštěný program – výzva pro připojení paže	33
Obrázek 26: Spuštěný program – reakce programu na data z náramku.....	33
Obrázek 27: Záznam testování – gesto 1	33
Obrázek 28: Záznam testování – gesto 2	33

Seznam tabulek

Tabulka 1: Porovnání výhod a nevýhod stávajících protéz.....	8
Tabulka 2: Hlubší porovnání výhod a nevýhod stávajících protéz.....	8
Tabulka 3: Klady a zápory tahových protéz	9
Tabulka 4: Klady a zápory myoelektrických protéz	10
Tabulka 5: Přiřazení gest knihovny náramku k pohonům robotické paže	34

Obsah na přiloženém CD

Ovladače.....	Složka obsahuje ovladače nutné pro plnou funkcionalitu
Programy.....	Finální verze ovládání robotické paže – 2 módy
Videa.....	Vytvořená video dokumentace během testování
Key_Words.pdf.....	Klíčová slova
Abstrakt_CZ.pdf.....	Abstrakt v českém jazyce
Abstrakt_ENG.pdf.....	Abstrakt v anglickém jazyce
BP_jirsa.pdf.....	Text bakalářské práce
Licence.txt.....	Znění licence MIT
Zadani_BP.pdf.....	Zadání bakalářské práce

Přílohy

Třída MyoConnector

```
using System;
using System.Windows.Forms;
using EposCmd.Net;
using EposCmd.Net.DeviceCmdSet.Operation;

using MyoSharp.Communication;
using MyoSharp.Device;
using MyoSharp.Poses;
using MyoSharp.Exceptions;

namespace Demo_EposCmd
{
    class MyoConnector
    {
        /*
         * Vytvoření proměnných
         */

        private EposController eposController;
        private DeviceManager deviceManager;
        private Device device;
        private IChannel channel;
        private IHub hub;
        private int current = 1;
        private int speed = 200;

        /*
         * Navázání komunikace s náramkem
         */

        public MyoConnector(EposController controller)
        {
            this.eposController = controller;
            channel =
Channel.Create(ChannelDriver.Create(ChannelBridge.Create(),
MyoErrorHandlerDriver.Create(MyoErrorHandlerBridge.Create()))
);
            hub = Hub.Create(channel);

            hub.MyoConnected += _hub_MyoConnected;
            hub.MyoDisconnected += _hub_MyoDisconnected;

            channel.StartListening();
        }
    }
}
```

```

void _hub_MyoDisconnected(object sender, MyoEventArgs e)
{
    e.Myo.PoseChanged -= Myo_PoseChanged;
}

/*
 * Odemčení náramku pro rychlejší změny gest, zaznamenávání
 změny póz
 */

void _hub_MyoConnected(object sender, MyoEventArgs e)
{
    eposController.setMyoState("Myo connected");
    e.Myo.PoseChanged += Myo_PoseChanged;
    e.Myo.SetEmgStreaming(true);
    e.Myo.Unlock(UnlockType.Hold);
}

/*
 * Gettery a settery
 */

public DeviceManager getDeviceManager()
{
    return deviceManager;
}

public Device getDevice()
{
    return device;
}

public void setDeviceManager(DeviceManager value)
{
    deviceManager = value;
}

public void setDevice(Device value)
{
    device = value;
}
}

```

```

/*
 * Switch rozhodující o tom, co se při jakém gestu vykoná.
 * Zde je možné provádět úpravy v případě potřeby
 */

void Myo_PoseChanged(object sender, PoseEventArgs e)
{
    switch (e.Pose)
    {
        case MyoSharp.Poses.Pose.DoubleTap:
            current = current * (-1);
            break;

        case MyoSharp.Poses.Pose.FingersSpread:

            speed = 500 * current;
            eposController.invokerMove("3", speed.ToString());
            break;
        case MyoSharp.Poses.Pose.Fist:
            speed = 75 * current;
            eposController.invokerMove("1", speed.ToString());
            break;
        case MyoSharp.Poses.Pose.Rest:
            eposController.invokerDisable();
            break;
        case MyoSharp.Poses.Pose.Unknown:
            eposController.invokerDisable();
            break;
        case MyoSharp.Poses.Pose.WaveIn:
            speed = 50 * current;
            eposController.invokerMove("2", speed.ToString());
            break;
        default:
            break;
    }
}
}

```