

# Návrh a řízení autonomního robota

František Kráčmar

ČVUT v Praze, Fakulta strojní, Ústav přístrojové a řídicí techniky, Technická 4, 166 07 Praha 6, Česká republika

## Abstrakt

Cílem práce je realizace prototypu autonomního robota dle pravidel soutěže ARLISS, která má svými požadavky simulovat orbitální mise na planetě Mars. Součástí práce je přehled dosavadních přístupů k řešení soutěžního úkolu. Následně je popsán vývoj a výroba vlastního robota obsahující návrh šasi, řídicí elektroniky a algoritmu pro autonomní jízdu, dále komunikačního protokolu pro komunikaci se vzdáleným stanovištěm.

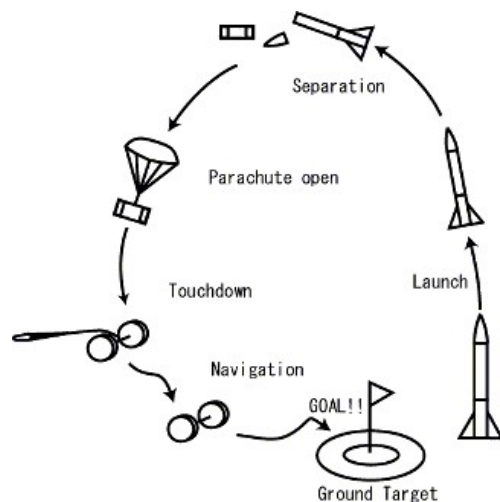
*Klíčová slova:* autonomní robot, navigace, GPS, diferenční řízení

## 1. Úvod

Člověk je prakticky denně obkloповán elektronickými výrobky. A i díky nepřebornému množství dostupných informací online se o ně nyní začíná zajímat podrobněji - jak fungují, nebo k jakému jinému účelu by se daly využít. Jisté vyústění těchto zájmů pak lze sledovat v pořádání soutěží zaměřených na robotiku, kterých se nyní účastní již středoškoláci. Tato práce si klade za cíl vycházet z pravidel jedné takovéto soutěže - ARLISS.

ARLISS<sup>1</sup> (A Rocket Launch for International Student Satellites) je mezinárodní soutěž sdružující univerzitní studenty a jejich učitele se záměrem nabídnout jim první reálné zkušenosti s aerospace odvětvím. Formou fiktivního zadání „orbitální mise na Marsu či Zemi“ studenty motivuje k tomu přijít s vlastním řešením k velmi komplikovanému zadání.

Tato mise je rozdělena do dvou podúkolů, čímž vznikají dvě kategorie soutěže - první, CanSat, se zaměřuje na vývoj miniaturních sond (tak aby se vešly do plechovky od nápoje), které během pádu ze 3000 metrů, jenž simuluje průlet po nízké oběžné dráze, mají za úkol sbírat experimentální data a ty následně zpracovat.



Obr. 1. Zadání soutěžního úkolu OpenClass [1]

Druhá kategorie, OpenClass, svým zadáním na-

<sup>1</sup><http://www.arliss.org>

\*Kontakt na autora: frantisek.kracmar@fs.cvut.cz

vazuje, a sice dává za úkol vyvinout robota, který se z výšky 3000 m plně autonomně dopraví na zadané GPS souřadnice v okruhu až 6 km. Schéma zadání lze vidět na obr. 1.

Soutěž tedy simuluje nároky splnění orbitální mise, zároveň však přidává inženýrský rozměr, když si studenti své prototypy zkouší na soutěžních letech nejen v Nevadské poušti. Organizace týmům poskytuje veškeré technické zázemí potřebné pro uskutečnění letů a s tím se pojí největší omezení - prostorové možnosti nákladového prostoru nosné rakety, uvedené v tab. 1.

Tabulka 1. Parametry nákladového prostoru

Kritérium	Přípustná hodnota
Průměr	146 mm
Výška	254 mm
Hmotnost	1,8 kg

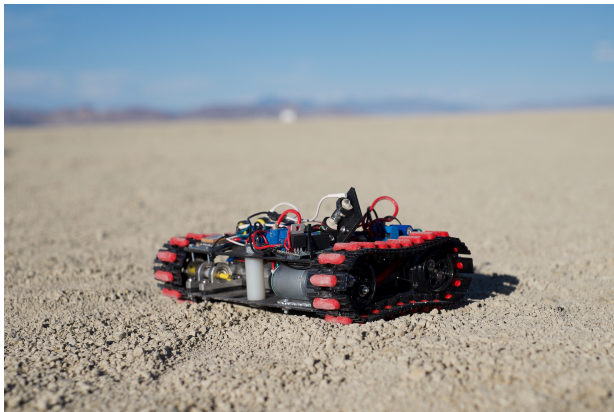
## 2. Přístupy k návrhu robota

Vzhledem k povaze zadání musí robot urazit část dráhy k cíli vzduchem. V současnosti k této skutečnosti existují dva přístupy - snést se k zemi a dále se pohybovat po ní, nebo doletět až na finální souřadnice s využitím nadmořské výšky poskytnuté nosnou raketou.

### 2.1. Cesta po zemi

Při tomto přístupu se k pohybu vzduchem přistupuje jako k nutnému zlu - robot se co nejbezpečněji a s co nejmenším vynaložením energie snese na zem na padáku, který se poté oddělí a začne se jednat pouze o úlohu přesunu na zemi. Tento přístup byl využit také například na roveru Opportunity na planetě Mars. Je ovšem nutné zmínit, že v této výzkumné misi nešlo pouze o samotný přesun. Vzhledem k její komplexnosti se řešení roveru jeví jednoznačně jako bezpečnější a energeticky méně náročné. O čemž svědčí fakt [2], že Opportunity na své patnáctileté směně zvládl na povrchu Marsu najezdit vzdálenost přesahující délku olympijského maratonu.

Zpravidla se zde volí konfigurace přední hnané nápravy se zadním vlečným prvkem zajišťujícím stabilitu. Avšak ne všichni účastníci soutěže se drželi pouze tohoto typu konstrukce a například University of Louisiana at Lafayette v roce 2017 soutěžila s robotem, který byl poháněn pásy, jak je vidět na obr. 2.



Obr. 2. Robot University of Louisiana at Lafayette z roku 2017 [3]

## 2.2. Cesta vzduchem

Alternativním přístupem k řešení soutěžního zadání je využití faktu, že se robot po vynesení nachází přibližně ve výšce 3 km a může vzálenost k cíli urazit letem za postupného klesání a tím minimalizovat výdej energie. Jen výjimečně se týmy rozhodnou navrhnout takový dron, a to především kvůli limitujícímu nákladnímu prostoru nosné rakety. V historii se však tým skládající se ze studentů FS ČVUT o takového robota pokoušel - viz obr. 3.



Obr. 3. Skládací kvadroptéra týmu z ČVUT [4]

## 2.3. Vlastní přístup

Po zvážení těchto variant jsem se rozhodl pro to, zaměřit se na robota pohybujícího se po zemi. Hlavním důvodem této volby jsou mé nulové zkušenosti se skutečnými podmínkami soutěžního prostředí. Pro úspěšný kompletní vývoj vlastní koptéry by bylo potřeba úspěšně zvládnout nesrovnatelně více překážek než při vývoji robota pozemního.

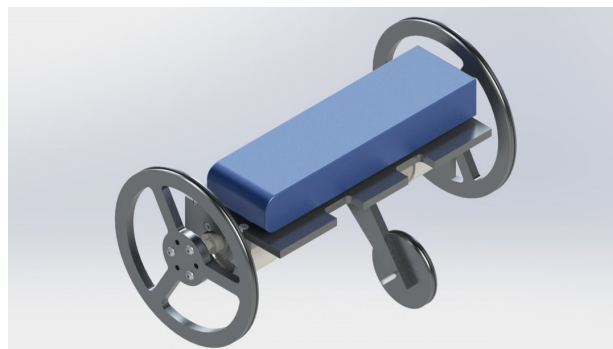
Dále si jsem vědom faktu, že je pro navržení a otestování plně funkčního prototypu nutné zvládnout problematiku návrhu a výroby jak šasi, tak elektroniky.

## 3. Vlastní robot

Ačkoliv se jedná pouze o prototyp, jeho vývoj v sobě spojuje komplexní vývojový cyklus. Vývoj začal definováním zadání a omezujících parametrů, ze kterých vznikl koncept celkového robota. Následně bylo nutné zvládnout problematiku návrhu i výroby šasi, kompletní volbu elektronických komponent pro požadované funkcionality, jejich usazení na desku plošných spojů a vývoj řídicího algoritmu.

### 3.1. Návrh kostry

Po seznámení se s omezujícími parametry začal vznikat virtuální návrh šasi roveru. Ten musel zohledňovat především předem dostupné komponenty, jejich vzájemné umístění a z toho vyplývající fyzikální vlastnosti celého robota. V rámci několika iterací se tak podařilo přijít s finálním návrhem - viz obr. 4, který byl odeslán do výroby a dodělán do konečného stavu. Jedná se o prototyp, proto mohou být některé části značně poddimenzované vzhledem k zatížení v soutěži. Toto se týká především kol. Na druhou stranu takto úzká kola mají při vývoji jednu značnou výhodu, a to jasně daný kontaktní bod s podložkou, což usnadňuje tvorbu matematického modelu.

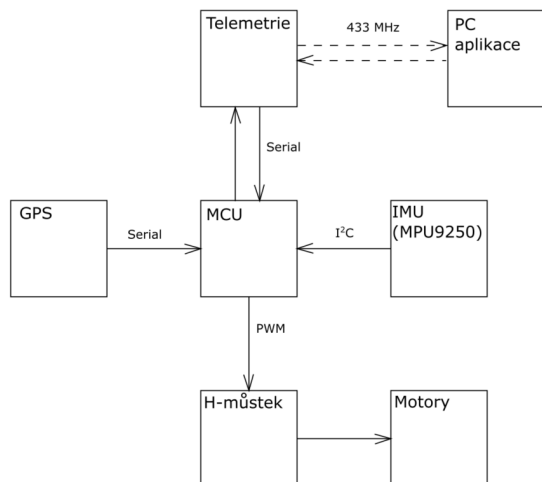


Obr. 4. Výsledný CAD návrh

### 3.2. Návrh elektroniky

Dalším důležitým bodem návrhu a realizace prototypu bylo připravit řídicí elektroniku pro plně autonomní řízení. Její blokové schéma lze vidět na obr. 5. Pro ovládání celého robota byl zvolen mikrokontrolér Teensy ve verzi 3.2 postavený na procesoru ARM, který svým výkonem násobně převyšuje velmi oblíbené Arduino. Zároveň si však Teensy z Arduina přebírá jeho nejsilnější zbraň, a sice možnost tvorby řídicího softwaru v Arduino IDE, kde je možné využívat již vytvořené knihovny jinými uživateli pro ovládání periférií. Mezi tyto periferie patří prvně H-můstek TB6612FNG zvolený především pro dostatečný stálý pracovní proud až 1,2 A na každý ze dvou kanálů. H-můstek spolu s DC motory Pololu 1104 zajišťuje kontrolovaný pohyb robota. Ze senzorů byl pro dobře zpracovanou dokumentaci a již vytvořenou SW knihovnou vybrán senzor pro měření polohy MPU9250, který obsahuje akcelerometr s rozsahem až  $\pm 16$  g, gyroskop s rozsahem až  $\pm 2000^\circ/\text{s}$  a magnetometr s měřicím rozsahem  $\pm 4800 \mu\text{T}$ . Ten je doplněný o GPS modul NEO-6M, jenž je díky své maximální přesnosti pozice 2,5 m plně dostačující pro splnění zadání. Pro možnost komunikace s robotem ze vzdáleného stanoviště byl přidán Telemetry modul pracující na frekvenci 433 MHz, a tedy zajišťující velkou komunikační

vzdálenost. Ten však bude muset být pro samotnou soutěž v USA vyměněn z legislativních důvodů. Na samotnou telemetrii se však nelze spolehnout ve všech případech a ani není úmyslem přes ni přenášet naprosto všechna aktuální data z robota, proto se do finální verze plánuje zakomponovat SD karta sloužící jako černá skříňka.



Obr. 5. Blokové schéma řídicí elektroniky

Pro kompaktnost potřebné elektroniky byla navržena deska plošných spojů za využití softwaru Autodesk Eagle. Deska prošla několika iteracemi - zprvu kvůli problémům se stabilitou napájení, další změny zohledňovaly vývojové požadavky - aktuální verze je již tvořena primárně SMD součástkami pro maximální úsporu místa na desce, byl doplněn senzor proudu pro průběžné měření stavu akumulátoru a v budoucnu také energetickou optimalizaci celého robota.

Robot je po celou dobu plnění úkolu napájen z akumulátoru, u kterého je stěžejní návrh jeho kapacity. Ta musí být dostatečná pro zvládnutí vzdálenosti k cíli, zároveň by však akumulátor měl být co nejmenší, protože svou velikostí se významně projevuje v návrhu šasi robota a navíc se jedná o nejtěžší komponentu. Z tohoto důvodu byla zvolena technologie LiPo, právě pro svou energetickou hustotu.

Při stanovení požadované kapacity se vycházelo z katalogových údajů použitých motorů, konkrétně z údaje o maximálních otáčkách, kdy Pololu 1104 dosahuje 200 rpm. Rovnicí (1) jsme schopni z tohoto údaje a průměru kola  $d$  vypočítat maximální teoretickou rychlost  $v$ .

$$v = f\pi d = 200 \cdot \pi \cdot 0,104 = 1,09 \text{ m/s} \quad (1)$$

Vezmeme-li jako vzdálenost  $l$ , kterou robot musí urazit, 6 km [4], čas  $t$  potřebný ke zvládnutí této cesty je dán rovnicí (2).

$$t = \frac{l}{v} = \frac{6000}{1,09} \approx 1,53 \text{ h} \quad (2)$$

Proudová zátěž akumulátoru byla určena jako součet údajů o spotřebě jednotlivých komponent dle jejich technických listů - zjištěné hodnoty jsou uvedeny v tab. 2. Celková proudová zátěž  $I$  byla stanovena s ohledem na vyšší bezpečnost na 1 A h. Potřebná kapacita akumulátoru tedy vychází z rovnice

(3), kde je konstantou  $k$  zohledněn fakt, že akumulátory typu LiPo jsou při reálných podmínkách využitelné ze 70 % své kapacity [5].

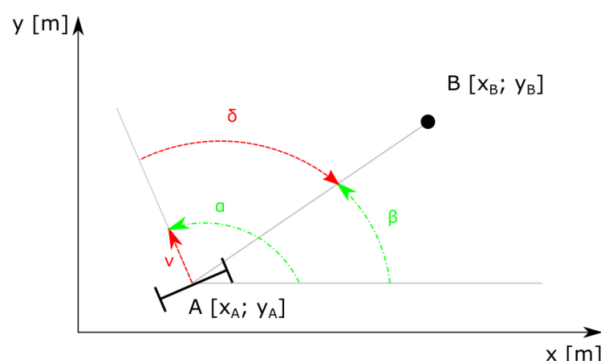
Tabulka 2. Proudový odběr jednotlivých komponent

Komponenta	Proudový odběr [mA]
Teensy	120
Motory	600
H-můstek	2,2
MPU9250	3,93
GPS NEO-6M	67
Telemetrie	125
<b>Celkem</b>	<b>919,13</b>

$$c = \frac{t \cdot I}{k} = \frac{1,53 \cdot 1}{0,7} = 2,2 \text{ A h} \quad (3)$$

### 3.3. Řídicí algoritmus

Při návrhu řídicího algoritmu se vycházelo z několika faktů. Úkolem je dostat se na zadané souřadnice, proto je také nezbytné, aby bylo možné zjistit aktuální pozici robota. Pro řízení směru pohybu je zapotřebí také určit aktuální směr (azimut), kterým se robot pohybuje. Pro řízení teoreticky stačí využívat pouze těchto veličin, neboť jak lze vidět na obr. 6, soutěžní povrch je planina, a proto se neuvažují překážky, které by zabraňovaly jízdě k cíli po přímce. Z těchto základních myšlenek pak vznikl souřadný systém vyobrazený na obr. 2. Jedná se o souřadný systém typu „Lokální sever-východ-nahoru“, jenž je variací na NED z [6], kdy je v blízkém okolí robota uvažována tangenciální rovina s povrchem země a její osy  $x$ ,  $y$  korespondují se zeměpisnými směry východ, respektive sever. Jednotkami os jsou metry vzniklé lineární vzdáleností ve stupních. Bod A označuje aktuální polohu robota určenou souřadnicemi  $x_A$  a  $y_A$ . Analogicky jsou pak značeny cílové souřadnice v bodě B  $x_B$  a  $y_B$ .



Obr. 6. Zavedený souřadný systém

Úhel  $\alpha$ , kterým je označován azimut robota, je jedním ze tří polohových úhlů - konkrétně tzv. yaw. Bylo tedy vhodné zvolit takový algoritmus, který s těmito úhly pracuje. V rámci vývoje byl proto zvolen Madgwickův filtr [7], který kombinuje právě data z akcelerometru, gyroskopu a magnetometru. Filtr využívá

principu fúzování dat, díky čemuž lze dosáhnout lepší stability polohových úhlů v porovnání se surovými daty ze senzorů. Upřednostněn byl před Kalmanovým filtrem, který je náročnější výpočtově a také citlivější na prvotní naladění konstant. Samotná vstupní data ovšem musí být sama o sobě validní, proto bylo potřeba provést kalibraci magnetometru kvůli lokálním deviacím magnetického pole.

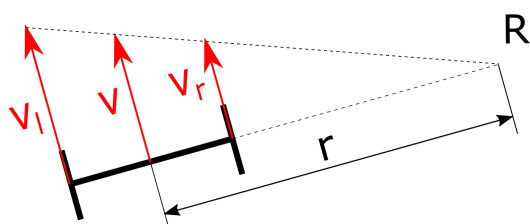
Požadovaný azimut směřující od aktuální pozice robota k cílovým souřadnicím je na obr. 6 označen jako  $\beta$  a je určen jednoduchým trigonometrickým výpočtem (4).

$$\beta(t) = \arctan \left( \frac{y_B - y_A(t)}{x_B - x_A(t)} \right) \quad (4)$$

Úhel  $\delta$  pak označuje regulační odchylku a vychází z rovnice (5).

$$\delta(t) = \alpha(t) - \beta(t) \quad (5)$$

Tento úhel je následně využit pro diferenční řízení, které je znázorněno na obr. 7. Výkony na jednotlivé motory jsou dány rovnicemi (6), kde  $v$  je rychlost robota předem určená konstantou a  $K_p$  je konstanta proporcionálního zesílení regulátoru. Z předpisu těchto rovnic je vidět, že se jedná o P regulátor, který na řízení tohoto systému postačuje, neboť tato soustava je integrační.



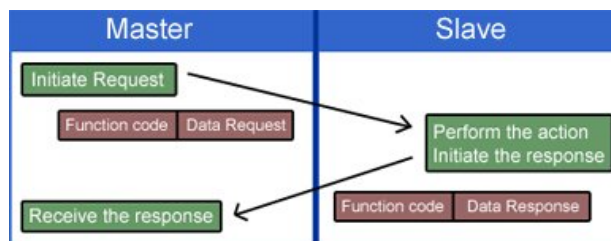
Obr. 7. Diferenční řízení

$$\begin{aligned} v_l(t) &= v(t) + K_p \cdot \delta(t) \\ v_r(t) &= v(t) - K_p \cdot \delta(t) \end{aligned} \quad (6)$$

Rozdílem v rychlostech jednotlivých kol je dán okamžitý střed otáčení  $R$  ve vzdálenosti  $r$  od středu robota, což umožňuje matematické vyjádření pohybu robota. Poloměr zatáčení  $r$  může nabývat hodnot od 0 m, kdy se robot otáčí na místě kolem svého středu, do  $\pm\infty$ , což znamená přímočarý pohyb vpřed/zpět. Takto zavedený systém řízení je vhodný pro pozdější vývoj, kdy lze mimo jiné ze zvoleného poloměru zatáčení zpětně dopočítávat požadované rychlosti kol.

### 3.4. Komunikace robota se vzdáleným stanovištěm

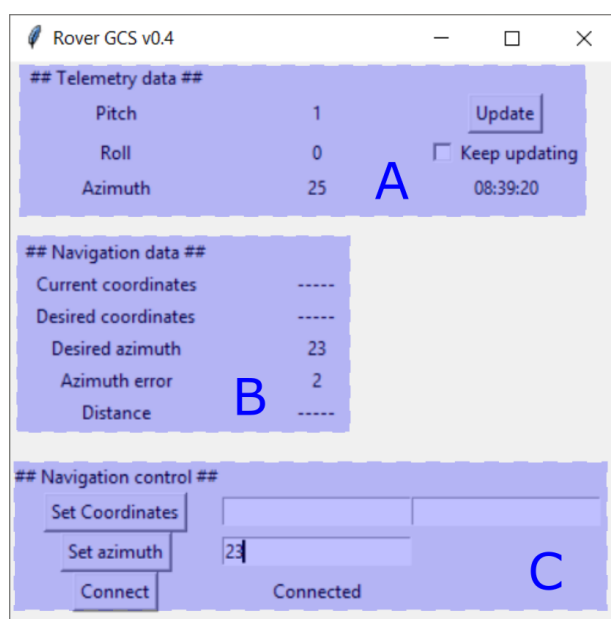
Kvůli možnosti vyčítat aktuální hodnoty z roveru a částečně ho ovládat byl navržen a naprogramován komunikační protokol. Svým principem odpovídá otevřenému protokolu MODBUS a přebírá z něj jeho aplikační schéma - viz obr. 8. V roli mastera zde figuruje PC aplikace (o té více dále), robot plní roli slave a odpovídá na dotazy zasílané masterem.



Obr. 8. Funkční schéma protokolu MODBUS [8]

Požadavky na robustnost tohoto protokolu nejsou nijak velké, neboť zprávy jsou zatím dlouhé maximálně 13 bytů, a proto nebyly použity již hotové knihovny (např. Mavlink<sup>2</sup>). Formát zprávy byl navržen tak, aby se v průběhu vývoje dal lehce upravit. Prvním znakem zprávy (na obr. 8 označen jako „Function code“) se určuje požadovaná akce (odeslat aktuální hodnotu, nastavit parametr), následují dobrovolná pole znaků obsahující konkrétní hodnoty. Ty slouží například pro přenos GPS souřadnic. Komunikace probíhá přes telemetry modul na 433 MHz a baudová rychlost je nastavena na 115200 baud.

Jak již bylo zmíněno dříve, byla vytvořena PC aplikace sloužící jako vzdálené ovládací stanoviště. Za použití volně dostupných knihoven zajišťuje uživatelsky přívětivý přehled o stavu robota - viz obr. 9. Prostorové úhly robota v sekci A, údaje o navigaci v sekci B a v sekci C je možné zadat požadované souřadnice či azimut, na které má robot dojet, a to bez nutnosti přehrát celý firmware mikrokontroléru.



Obr. 9. PC aplikace vzdáleného stanoviště

## 4. Budoucí vývoj

V dalším vývoji je v plánu zaměřit se především na robustnější řídicí algoritmus. Pro zpřesnění polohy bude použit Kalmanův filtr, který kombinuje data ze senzorů na základě statistiky. Alternativně k datům ze senzorů lze také doplnit matematický model pohybu robota z jeho popisu na obr. 7. Vstupními veličinami modelu budou úhlové rychlosti DC motorů, proto je

<sup>2</sup><https://mavlink.io>

nezbytná jejich identifikace - ideálně z experimentálně určené frekvenční charakteristiky.

V dlouhodobém horizontu pak lze vyzkoušet metodu řízení „model predictive control“ (MPC) pro optimalizaci ať již energetickou, tak především plánování trasy.

## 5. Závěr

Výsledkem práce je prototyp robota, který je schopen se plně autonomně dopravit na zadané GPS souřadnice. V potaz byly navíc vzaty specifické požadavky mezinárodní studentské soutěže ARLISS, aby bylo možné na tomto prototypu vytvořit již finální verzi robota schopného účasti v této soutěži. Po rešerši dosud využívaných konstrukcí v minulých ročnících byla zvolena tříkolová konstrukce se zadním vlečným kolečkem. Po provedeném CAD návrhu a následné výrobě byly zvoleny všechny potřebné elektronické komponenty. Pro jejich umístění na robota vznikl návrh desky plošných spojů, který byl odeslán do výroby a vlastnoručně osazen.

Pro orientaci v prostoru byl zaveden souřadný systém „lokální sever-východ-nahoru“, na základě něhož jsem zvolil proporcionální regulační obvod. Regulace probíhá na prostorových orientačních úhlech získaných kombinací dat z jednotlivých senzorů využitím Madgwickova filtru.

Za účelem snazšího ovládní nastavení robota byla dodatečně vyvinuta PC aplikace, která si přes mnou navržený a implementovaný komunikační protokol s prototypem vyměňuje všechny potřebné informace.

## Poděkování

Tato práce byla podpořena grantem Studentské grantové soutěže ČVUT č. SGS17/176/OHK2/3T/12.

## Literatura

- [1] Genya Ishigami. *ARLISS 2003 Project*. The Space Robotics Lab, Tohoku University, JAPAN. Dostupné z: <http://www.astro.mech.tohoku.ac.jp/~ishigami/ARLISS2003/> (cit. 18. 03. 2019).
- [2] NASA/JPL-Caltech. *Driving Distances on Mars and the Moon*. NASA. Dostupné z: <https://mars.nasa.gov/resources/6471/driving-distances-on-mars-and-the-moon/> (cit. 14. 03. 2019).
- [3] C.R.A.W.LAB at University of Louisiana at Lafayette. Dostupné z: <https://www.flickr.com/photos/crawlab/37292762755/in/album-72157677967941992> (cit. 14. 03. 2019).
- [4] Petr Homolka a Jaroslav Bušek. “Autonomní samoozkládací koptéra pro účel soutěže ARLISS/RescueBot”. In: *Sborník konference Nové metody a postupy 2017* (2017).
- [5] Larry H. Dostupné z: <https://www.helidirect.com/blog/getting-the-best-out-of-your-lipo-batteries.html> (cit. 16. 03. 2019).
- [6] Cai Guowei. *Unmanned Rotorcraft Systems*. Springer, 2011.
- [7] Sebastian O.H. Madgwick. *An efficient orientation filter for inertial and inertial/magnetic sensor arrays*. Dub. 2017.
- [8] Inc. Real Time Automation. *MODBUS RTU*. Real Time Automation, Inc. Dostupné z: <https://www.rtautomation.com/technologies/modbus-rtu/> (cit. 14. 03. 2019).