

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV MECHANIKY TĚLES, MECHATRONIKY A BIOMECHANIKY
FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF SOLID MECHANICS, MECHATRONICS AND BIOMECHANICS

NÁVRH Trenažéru přistání na padákovém kluzáku DESIGN OF PARAGLIDER LANDING SIMULATOR

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. PAVEL WERTHEIMER

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. JIŘÍ KREJSA, Ph.D.

BRNO 2011

Abstrakt

Trénink přistávání na padákovém kluzáku je velice neefektivní, protože každý let potřebuje dlouhou přípravu. Školy paraglidingu proto hledají alternativu pro výcvik začínajících pilotů. Řešením je trenažér létání na padákovém kluzáku, návrhem takového zařízení se tato diplomová práce zabývá. Návrh zahrnuje mechanickou část – konstrukce zavěšení pilota stejně jako na reálném padáku; elektronickou část – snímání akčních zásahů pilota a řízení síly v řídicích šňůrách a softwarovou část – návrh firmwaru pro řídicí mikrokontrolér. V rámci práce je také vyroben funkční prototyp.

Summary

Practice of landing with paraglider is very inefficient since every flight requires lengthy preparation. Therefore paragliding schools look for a more efficient alternative how to train new pilots. A paraglider flying simulator, which is the subject of this Master's Thesis, could be that alternative. The design of the simulator includes three parts: a mechanical part – a frame to hang a pilot, that is the same as in real paraglider; an electronic part – scanning of pilot's control actions and control of force in breaks of a paraglider; and a software part – a design of firmware for a microcontroller. Within the Master's thesis a function prototype has been made.

Klíčová slova

trenažér, padákový kluzák, stejnosměrný motor, H-můstek, ATxmega, digitální řízení

Keywords

simulator, paraglider, DC motor, H-bridge, ATxmega, digital control

WERTHEIMER, P. *Návrh trenažéru přistání na padákovém kluzáku*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2011. 53 s. Vedoucí Ing. Jiří Krejsa, Ph.D.

Čestně prohlašuji, že jsem diplomovou práci na téma *Návrh trenažéru přistání na padákovém kluzáku* vytvořil samostatně pod vedením svého vedoucího diplomové práce s využitím odborné literatury, kterou jsem všechnu uvedl v seznamu literatury.

Bc. Pavel Wertheimer

Rád bych tímto poděkoval Ing. Jiřímu Krejsovi, Ph.D. za odborné vedení při přípravě této diplomové práce. Děkuji také Stanislavu Slámovi, Bc. Miroslavu Zezulovi a Bc. Daliboru Šulcovi za pomoc při realizaci prototypu trenažéru.

Bc. Pavel Wertheimer

Obsah

1	Úvod	8
2	Formulace problémů a cílů řešení	9
2.1	Padákový kluzák	10
2.2	Řídící prvky padákového kluzáku	12
2.2.1	Řídící šňůry (brzdy)	12
2.2.2	Řízení pomocí náklonu v sedačce	13
2.2.3	Speed systém	13
3	Mechanická část	14
3.1	Konstrukční řešení zavěšení sedačky	14
3.1.1	Speed systém	14
3.1.2	Upravená konstrukce zavěšení sedačky	15
3.1.3	Volba a umístění snímačů	16
3.2	Provedení řídicích šňůr	17
3.2.1	Možné způsoby provedení tahu	17
3.2.2	Proměnný tah pomocí DC motoru	19
3.2.3	Snímání síly	21
4	Elektronika	23
4.1	Výkonová elektronika	23
4.1.1	H-můstky	24
4.1.2	Brzdný odpor	26
4.1.3	Zdroje	27
4.2	Řídící elektronika	28
4.2.1	Mikrokontrolerový modul	28
4.2.2	Zesilovač pro senzor zatížení	31
4.2.3	Akcelerometr	31
4.2.4	Enkodéry	32
5	Programová obsluha	34
5.1	Obsluha senzorů	35
5.1.1	Analogové senzory	35
5.1.2	Digitální senzory	38
5.2	Řízení motorů	39
5.2.1	Generování PWM	40
5.3	Komunikace s PC	43
5.3.1	USART	43

5.3.2	Komunikační protokol	44
6	Závěr	47
	Seznam použité literatury	48
	Seznam použitých zkratk a symbolů	50
	Seznam příloh	53

1. Úvod

Létání na padákovém kluzáku láká v dnešní době čím dál více lidí nadšených myšlenkou jeho jednoduchosti. Pro jeho základní zvládnutí není potřeba několikaletého studia a mnoha zkoušek. Naopak postačuje (přeje-li alespoň trochu počasí) jednotýdenní výcvik, při kterém poznáte základní ovládání a aerodynamiku padákového kluzáku a samozřejmě pro létání velice důležité základy meteorologie. Po absolvování takového výcviku a následném vyřízení pilotního průkazu u Letecké amatérské asociace ČR můžete svobodně vyletět nad krajinu. Dokonce i vybavení pro paragliding nepotřebuje hangáry či dobrou příjezdovou cestu na místo startu. Vše se dá jednoduše sbalit do batohu a odnést na start po vlastních. Všechny tyto důvody dávají člověku často zmiňovaný pocit volnosti a nezávislosti.

Tím však výhody padákového létání končí. Velkou nevýhodou je především značná závislost na povětrnostních podmínkách. Létat je možno pouze pokud neprší a rychlost větru je v „rozumných mezích“ (tuto mez si určuje každý pilot sám), avšak všeobecně v rozmezí $0 \div 8 \text{ m}\cdot\text{s}^{-1}$. Další nevýhodou je finanční náročnost, i přesto že se jedná o nejlevnější letecký sport. Například ve školách paraglidingu je třeba vynaložit mnoho nákladů při hledání vhodného leteckého terénu a čekání na vhodné povětrnostní podmínky, a to se v těchto školách uskutečňují pouze krátké, na podmínky ne příliš náročné lety. Další velká nevýhoda je nebezpečnost. Dnes sice padáky (obzvláště ty školní) mají vysokou úroveň bezpečnosti, nicméně létání je stále nevyzpytatelné a obzvláště u začátečníků může snadno dojít ke zranění.

Paraglidingové školy proto hledají alternativy pro trénink letu i při nepříznivém počasí, pro uskutečnění prvních pokusů leteckých nováčků, a nebo pro prostý trénink standardních letových situací a přistání. Vytvoření takového zařízení pro společnost El Speedo s.r.o. je cílem této práce.

Navrhované zařízení musí mít veškeré standardní ovládací prvky, které má i padákový kluzák - tj. řídicí šňůry, řízení pomocí náklonu v sedačce, speed systém. V řídicích šňůrách by měla být generována odpovídající silová odezva (v závislosti na letovém stavu).

Jedná se o poměrně unikátní zařízení, na internetu lze jen těžko najít zmínku o podobném, natož o jeho vlastnostech.

2. Formulace problémů a cílů řešení

Pod pojmem trenažér máme namysli zařízení, které bude připojeno k počítači podobně jako joystick. A stejně tak mu bude zprostředkovávat hodnoty z různých senzorů, udávající akční zásah uživatele. Nejde však o stolní zařízení, ale o zařízení, které dokáže napodobit pozici pilota při letu, tzn. zařízení, na které se může pilot v postroji zavěsit. Trenažér musí nejen napodobit chod všech řídicích prvků, ale také vytvořit jejich silové odezvy shodné s těmi u letícího padákového kluzáku. Na počítači, ke kterému bude trenažér připojen, poběží virtuální realita letu a její vizualizace bude promítána na plátno před pilota.

Návrh paraglidingového trenažéru lze rozdělit na několik podúloh. První je návrh hardwarové části, tedy samotná konstrukce s ohledem na napodobení připojení na letící padákový kluzák a s ohledem na umístění snímačů a motorů. Dále je potřeba navrhnout řídicí elektroniku pro pohony, pro obsluhu senzorů a pro komunikaci s nadřazeným PC. Poté pro tuto elektroniku navrhnout firmware. Posledním úkolem je návrh softwaru pro PC, který probíhá paralelně v rámci jiné diplomové práce. Výstupem práce je i funkční prototyp.

Než však přistoupíme k samotnému návrhu, projdeme si základní prvky padákového kluzáku nutné pro pochopení problematiky.

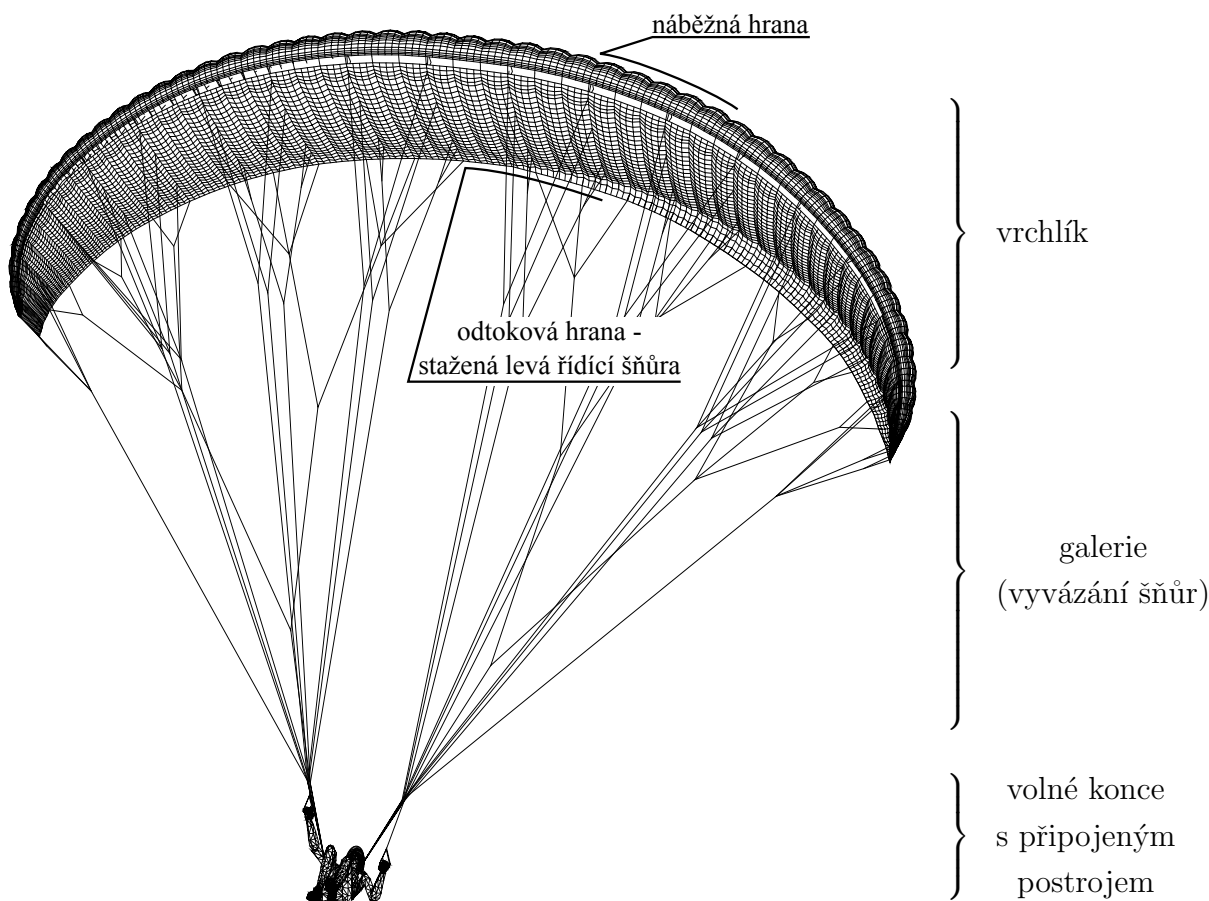


Obrázek 2.1: Létání na padákovém kluzáku. (převzato z [5])

2.1. Padákový kluzák

Vymezení kategorie podle zákona vyhlášky 108/1997 Sb. § 24, odstavec 2:

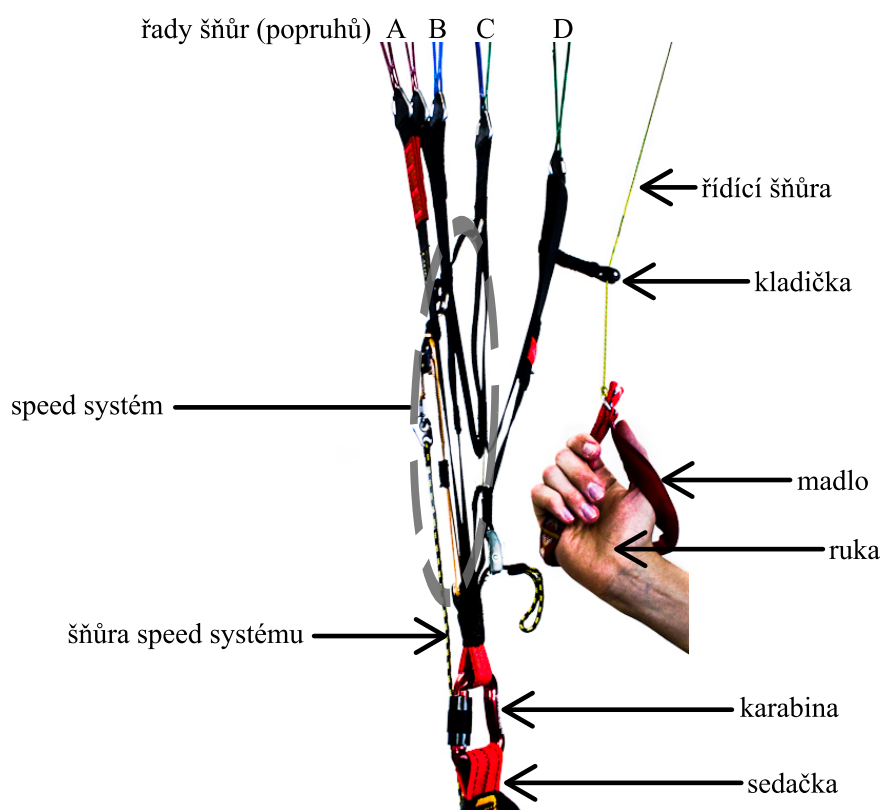
Padákový kluzák je bezmotorové letadlo těžší vzduchu, které je konstruováno maximálně pro dvě osoby a jehož vzlet se uskutečňuje rozběhem pilota, aerovletem nebo navijákem, a jehož charakter nosné plochy není určován tuhou konstrukcí.



Obrázek 2.2: Schéma padákového kluzáku. (zdroj [8])

Padákový kluzák je jedním z nejjednodušších zařízení, které umožňuje let, není určen pro seskoky z letadla (nesnese takové dynamické zatížení - není pro ně ani konstruován). Padákový kluzák je určený primárně pro let a jeho konstrukční parametry jsou tomu podřízeny – snaha o co nejlepší poměr dopředné rychlosti ku rychlosti klesání (tzv. klouzavost). U moderních padáků se tento poměr, v závislosti zda jde o školní a nebo závodní křídlo, pohybuje od 7,5 do 9,5. Za dobrých povětrnostních podmínek tak dovolují podniknout tzv. přelety, tedy za využití termických či svahových stoupavých proudů nastoupat do větších výšek, dovolujících přelet do dalšího regionu se stoupavými proudy. Každý pilot kluzáku by tedy měl mít dobré povědomí o meteorologii a proudění vzduchu v krajině. Takový přelet při dobrých podmínkách může trvat v závislosti na pilotových schopnostech

třeba celý den (v noci by se létat nemělo). Vzdálenosti uražené při přeletu pak mohou být desítky až stovky kilometrů, evropský rekord činí 335,94 km (pilot Karel Vejchodský, 28.7.2008) [13].



Obrázek 2.3: Popruhy padákového kluzáku.

Padákový kluzák se skládá (viz obr. 2.2) z vrchlíku, galerie (vyvázání šňůrami) a z popruhů, které jsou spojeny do tzv. volných konců. Do ok na volných koncích se pomocí karabin připojuje postroj pilota (někdy též sedačka), ve kterém je umístěn záložní padák.

Vrchlík je vyroben ze speciální tkaniny (lehké a takřka nepropustné pro vzduch), sešité tak aby vytvořila „kapsu“, která se po nafouknutí vytvaruje v letecký profil. Galerii tvoří systém šňůr vyvázaných v několika řadách, každé řadě odpovídá jeden popruh volného konce. Na jednom padáku je použito až 350 m šňůry ze speciálního materiálu s dobrou stálostí délky a pevností. Popruhy (obr. 2.3) jsou při letu pro pilota nejdůležitější částí padáku, také jedinou na kterou dosáhne. Jsou na nich umístěny veškeré ovládací prvky. Každý volný konec je složen ze 3 až 4 řad popruhů (odpředu dozadu označeny A až D a rozlišeny barvami).

Padákové kluzáky se vyrábí v několika velikostech (S až XL), každá velikost odpovídá určitému rozsahu letové hmotnosti. Krom velikosti kluzáku se jeho konstrukce také liší podle určení – školní křídlo, křídlo pro začínající, pro pokročilé, závodní a tandemové.

2.2. Řídící prvky padákového kluzáku

Nyní si podrobněji projdeme základní řídicí prvky padákového kluzáku a jejich charakteristiky chodu, tak abychom je byli schopni napodobit při návrhu simulátoru.

2.2.1. Řídící šňůry (brzdy)

Řídící šňůry (někdy také nazývané brzdy) jsou hlavním ovládacím prvkem padákového kluzáku. Jsou umístěny ještě za popruhy řady D a tvoří tak nejzadnější řadu šňůr (obr. 2.3). Vyvázaný jsou zvláště pro levou a pro pravou polovinu kluzáku na samotnou odtokovou hranu, kterou při stahování deformují (viz obr. 2.2) a mění tak geometrii leteckého profilu na příslušné straně kluzáku. Na druhém konci jsou šňůry přivázány k madlům, za která již přímo tahá pilot.

Šňůry jsou nad madlem protaženy kladičkou, která usměrňuje jejich chod. Kladička také určuje základní tzv. trimovou polohu řízení, je to poloha úplně vypuštěného řízení (plně odbrzděno) a kluzák je tak při aplikaci řízení (pomineme-li speed systém viz dále) schopen pouze zpomalovat. Pro naše účely si můžeme trimovou polohu nadefinovat jako nulovou - tj. nulové, tedy žádné stažení řídicí šňůry.

Pilot může z trimové pozice řídicí šňůru stahovat až po začátek jejího větvení, tj. v závislosti na typu a velikosti padáku cca $200 \div 300 \text{ cm}$. V praxi se však takto hluboké stažení řídicích šňůr nevyužívá, naopak při hlubokém stažení za letu může dojít k odtržení proudnic od leteckého profilu a kolapsu celého vrchlíku. Při standardních režimech letu a přistání je obvyklé maximální stažení kolem 120 cm .

Při návrhu simulátoru je také potřeba počítat se silou v řídicích šňůrách, vznikající odporem vzduchu při obtékání deformovaného profilu a samotných řídicích šňůr vyvázaných na odtokovou hranu. Průběh velikosti této síly primárně závisí na hloubce stažení řízení, ne však lineárně. Při letu však může docházet k různým přídatným dynamickým účinkům, které sílu výrazně ovlivní, např. po přetažení¹ kluzáku síla v řídicích šňůrách prudce poklesne. Veškerou silovou odezvu v řízení by měl simulátor být schopen napodobit. K simulaci nestandardních letových režimů, jako je odtržení proudnic, není simulátor primárně určen, avšak pro účel výuky by je měl alespoň v malé míře zvládat. Přesnější hodnoty a průběhy síly nebyly zatím proměřeny, avšak dle odhadů instruktora létání na padákovém kluzáku (zadavatel práce) a zkoušek dosahuje maximálních hodnot kolem 150 N .

Pro účely návrhu také odhadneme obvyklou rychlost stahování nebo spíše povolování řídicí šňůry. Tato hodnota se pohybuje okolo $1,5 \text{ m/s}$.

¹přetažení – velké stažení brzd pod bod, kdy deformace vrchlíku je natolik velká, že dochází k odtržení proudnic od leteckého profilu

2.2.2. Řízení pomocí náklonu v sedačce

Další způsob řízení padákového kluzáku je náklonem do stran. Vykloněním pilot přenáší váhu z jedné poloviny kluzáku na druhou, čímž stoupá plošné zatížení jedné poloviny a zároveň klesá u druhé poloviny křídla. Padák tak vlivem aerodynamických sil má tendenci zatačet na více zatíženou stranu.

2.2.3. Speed systém

Posledním základním ovládacím prvkem je tzv. speed systém, tedy systém, který provádí změnu délky jednotlivých řad popruhů volných konců a upravuje tak geometrii leteckého profilu, resp. snižuje úhel náběhu (úhel pod kterým je letecký profil obtékán, více v [9]). Po aplikování speedu dojde ke zvýšení dopředné rychlosti kluzáku (zmenší se totiž jeho čelní odpor) na úkor většího opadání, tj. rychlosti klesání (zároveň totiž klesne vztlaková síla).

Ovládání speed systému se nachází na sedačce, jedná se o 2 až 3 „hrazdičky“ umístěné pod sedákem, pilot tak může jednoduše ovládat speed systém nohama. Chod speed systému při sešlapování je poměrně těžký, pilot totiž pro změnu délky popruhů musí přes kladičku jakoby zvedat svou vlastní váhu.

3. Mechanická část

Pod pojmem návrhu mechanické části se skrývá návrh samotného zavěšení pro pilota v postroji a umístění motorů a senzorů tak, aby byly schopny simulovat řídicí prvky padáku a získávat potřebné hodnoty jejich stavů.

3.1. Konstruktivní řešení zavěšení sedačky

Konstrukce zavěšení musí vyřešit dobré napodobení pro řízení padáku náklonem a také speed systém.

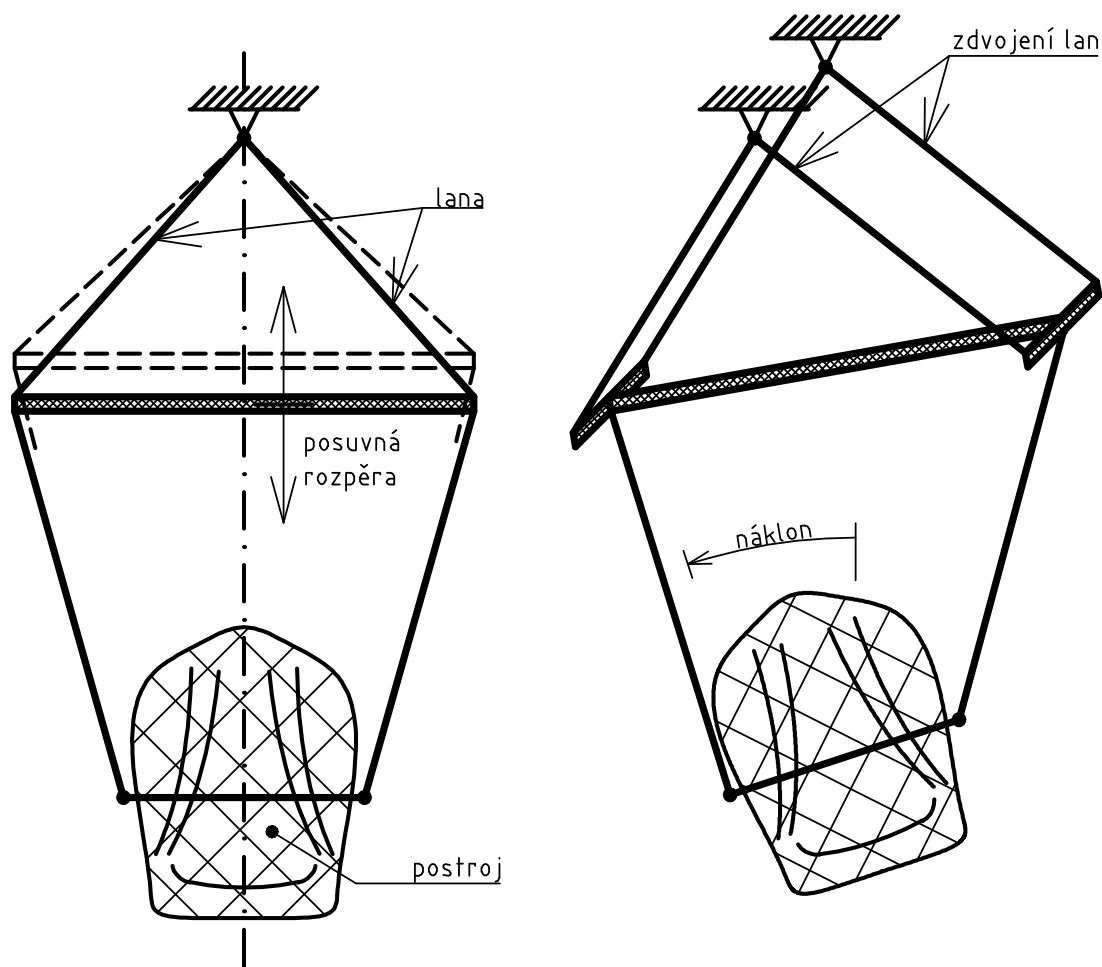
Vycházíme proto z konstrukce hojně využívané pro simulaci zavěšení na padák za účelem individuálního nastavení sedačky každého pilota. Kinematicky se jedná o soustavu znázorněnou na obr. 3.1 vlevo. Konstrukce se skládá ze dvou závěsných lan, na jednom konci přichycených na nosnou konstrukci (např. strop) - na druhém konci mají oka, do kterých je možno zacvaknout karabiny padákové sedačky. Důležitou součástí je rozpěrná tyč vložená mezi závěsná lana, tyč je možno mezi lany výškově posouvat a takto nastavovat „živost“ řízení padáku náklonem. Pro naše účely bude konstrukci potřeba mírně upravit.

První úpravou konstrukce zavěšení je zdvojení lan nad rozpěrou, obr. 3.1 vpravo. Samotnou rozpěru opatříme na koncích kratšími příčnickami, tvar rozpěry připomíná široké písmeno H. Takto je zavěšení odolné vůči rotaci kolem vlastní svislé osy, kterou by mohl způsobit například pilot vlastními pohyby v průběhu simulace letu. V případě, že bude vizualizace simulovaného letu promítána na plátno před pilota, je taková rotace nepřijatelná.

3.1.1. Speed systém

Pro dosažení realistického chodu speed systému bychom mohli použít například dostatečně tuhé pružiny, uchycené k závěsným lanům. Výhodnější variantou je však využití volných konců z padáku, které již na sobě speed systém přímo mají. Použití volných konců místo závěsných lan také dále zvyšuje míru podoby s reálným padákem. Umožňuje připojení postroje i ovládání speed systému obvyklými karabinami. Takto si pilot může vzít pro simulaci letu vlastní postroj, na který je zvyklý a má jej pro sebe nastaven. Jednoduše jej připojí ke konstrukci simulátoru standardně jako k padákovému kluzáku. Větší podoba s reálným padákem dává i reálnější pocit ze simulace.

Abychom mohli uvázat volné konce k závěsné konstrukci tak, aby zůstala plně zachována funkčnost speed systému, bude třeba druhé změny proti původní konstrukci. Jednotlivé řady popruhů vyvážíme podél příčnic, s obdobnými rozestupy jako je tomu u reálného padáku. Aplikace speed systému vyvolá poměrné zkrácení popruhů a tedy sklopení H-rozpěry. Aby mohlo ke sklopení dojít, musí být nosná lana připojena k nosné konstrukci přes kyvnou tyč, viz obr. 3.2.

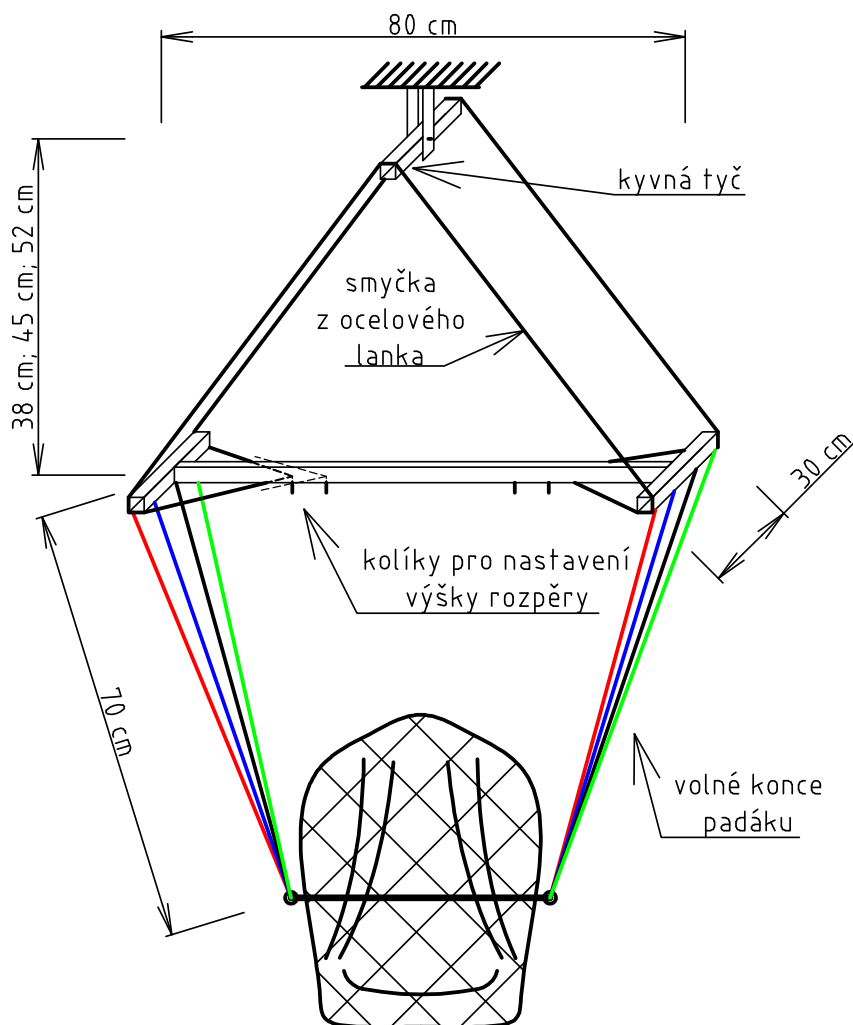


Obrázek 3.1: Schema závěsu sedačky původní (vlevo) a upravené (vpravo).

3.1.2. Upravená konstrukce zavěšení sedačky

Zvolené konstrukční řešení je znázorněno na obr. 3.2. Lano je použito klasické ocelové průměru 2,5 mm. Rozpěra tvaru H je svařovaná z ocelových uzavřených profilů se čtvercovým průřezem, standardního hutního materiálu. Volné konce jsou přivázány k příčnícím, jejich poloha je zajištěna šrouby. Před mechanickým poškozením v místě přivázání jsou chráněny PVC bužírkou. Ocelové lano je spojeno do velké smyčky, a proto není potřeba řešit jeho upevnování ke konstrukci. Možnost výškového nastavení rozpěry je realizována zaháknutím smyčky na rozpěře v různých pozicích (viz obr. 3.2 – kolíky pro nastavení výšky rozpěry). Délka lana nad rozpěrou se zkracuje (resp. prodlužuje) a rozpěra stoupá (resp. klesá).

Veškeré rozměry jsou stanoveny experimentálně s důrazem na napodobení chování reálného padáku. Celá konstrukce je velmi jednoduchá a levná, složená ze standardních polotovarů.



Obrázek 3.2: Schema upravené konstrukce zavěšení.

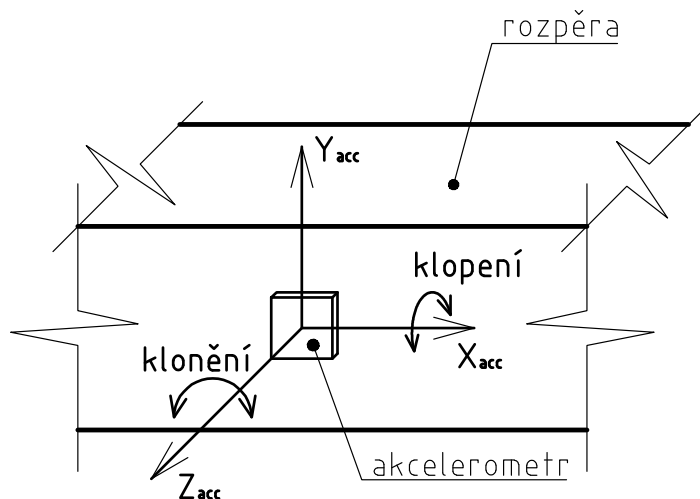
3.1.3. Volba a umístění snímačů

Konstrukce je díky použitým lanům velice jednoduchá, o to problematičtější je zabudování nějakého mechanického senzoru pro měření náklonu v sedačce. Řešení takového problému proto obejdeme využitím dnes již velice dostupných MEMS akcelerometrů.

Využijeme toho, že při vyklánění v sedačce se zároveň v jisté míře kloní i rozpěra. Navíc při použití speed systému se rozpěra naklápí. Referencí pro měření je samozřejmě směr gravitačního zrychlení. Akcelerometrický senzor umístíme na rozpěru, poblíž středu. Přesné umístění není důležité, protože hodnoty ze senzoru budou upravovány filtrem typu dolní propust s velkou časovou konstantou a dynamická zrychlení při přechodových jevech se tedy do měření příliš neprojeví. Problémem by však mohlo být houpání konstrukce, které má poměrně velkou periodu, veškeré tření u pohybujících se částí je tedy výhodou.

Důležitější je orientace senzoru, respektive jeho tří os, ve kterých zrychlení měří. Podle jejich natočení vzhledem k rozpěře je potřeba pro výpočet klonění a klopení použít hod-

noty z příslušných os. Pro jednodušší výpočet úhlů proto volíme orientaci tak, aby vždy jedna z os byla rovnoběžná s osou otáčení. To znamená, že úhel pak počítáme z ostatních dvou os kolmých. Lépe situaci objasňují obrázek 3.3 a rovnice pro výpočet úhlů klonění 3.1 a klopení 3.2.



Obrázek 3.3: Nákres orientace akcelerometru.

$$\textit{klonění} = \arctan \frac{x_{acc}}{y_{acc}} \quad (3.1)$$

$$\textit{klopení} = \arctan \frac{z_{acc}}{y_{acc}} \quad (3.2)$$

Proměnné x_{acc} , y_{acc} a z_{acc} jsou hodnoty z jednotlivých os akcelerometru. Úhel klonění odpovídá míře vyklonění v sedačce a klopení odpovídá míře aplikace speed systému.

3.2. Provedení řídicích šňůr

Jak už bylo řečeno na volných koncích se krom speed systému nacházejí také řídicí šňůry. Jejich provedení je tedy již předem dáno. Větším problémem ovšem bude generování silové odezvy v řídicí šňůře v průběhu letu. Také je potřeba vyřešit měření úrovně stažení řídicích šňůr.

3.2.1. Možné způsoby provedení tahu

Možností provedení tahu připadá do úvahy víc, každá má samozřejmě své výhody a své nevýhody.

Soustava gumových provazců

Několik gumových vláken (dostatečně silných např. modelářských) je spojeno paralelně, tak aby ze začátku stahování řízení zabíralo nejprve jen jedno a postupně s čím dál tím větším stažením se přidávaly další. Takto lze poměrně dobře (po malém experimentování a upravování) nastavit sílu a její průběh adekvátně k chování reálného padáku. Dokonce i přetažení padáku, tedy náhlý pokles síly v řízení na konci jeho rozsahu, by mohlo být realizováno například silnými magnety ze vzácných zemin. Při překročení kritické síly by došlo k odtržení magnetu od podložky, gumy připojené na magnet by se stáhly a síla v řízení by pominula.

Velkou výhodou tohoto řešení je jednoduchost a nízká cena, také malá setrvačnost je velikým plusem. Veškeré setrvačné hmoty v řízení padáku totiž představují jen madlo řízení, kus tkaniny (odtoková hrana) padáku a vzduch v ní ukrytý. Celkově tedy velice nízká setrvačnost by měla být zachována i na simulátoru.

Nevýhodou je nemožnost měnit velikost síly v průběhu simulovaného letu - v závislosti na aktuálním stavu a vůbec složitější nastavování velikosti síly. Také parametry gumy se mohou s časem měnit a ovlivňovat tak chování řízení.

Lineární motor

Další možností je využití lineárního motoru. Řídící šňůra by byla přichycena přímo na jeho pohyblivou část.

Výhodou je zachování poměrně malé setrvačnosti řízení padáku a možnost nastavení velikosti síly tahu v reálném čase.

Záporem elektronického řízení síly tahu v řídicí šňůře je nutnost sílu i měřit. Velkou nevýhodou tohoto řešení je však i vysoká cena lineárního motoru a jeho malá síla, pokud by nebylo využito převodů, avšak tímto vše nabývá na složitosti.

Klasický elektrický motor

Využití klasického rotačního motoru vyžaduje výrobu bubnu, na který bude řídicí šňůra navíjena. K motoru však můžeme připojit standardní senzor natočení (např. enkodér) pro jednoduché měření hodnoty stažení řízení. Pro naše účely by elektrický stroj měl mít co největší poměr kroutící moment/hmotnost (kvůli co nejnižší setrvačnosti), zároveň by však měl zvládat i dostatečnou rychlost potřebnou při navíjení řídicí šňůry; to nás přivádí k užití synchronního nebo stejnosměrného stroje s permanentními magnety ze vzácných zemin. Z důvodů lepší dostupnosti a jednodušší říditelnosti bude vhodnější užít stejnosměrný motor.

Výhody tohoto provedení jsou poměrně nízká cena (v závislosti na kvalitě motoru) a opět elektronické řízení síly.

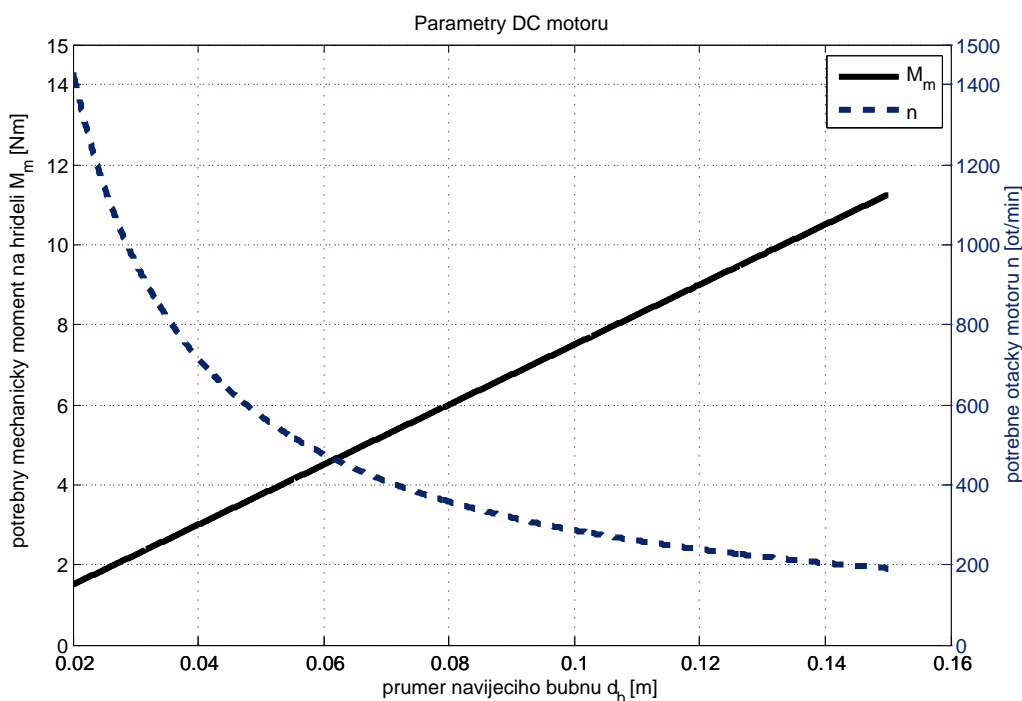
Nevýhodou může být navíjení šňůry na bubnu, tedy riziko jejího zauzlení, a opět nutnost sílu i měřit.

3.2.2. Proměnný tah pomocí DC motoru

Parametry stejnosměrného motoru nutné pro splnění odhadnutých hodnot rychlosti $v_{max} = 1,5 \text{ m/s}$ a síly $F_{max} = 150 \text{ N}$ v řízení padákového kluzáku jsou vypočítány podle jednoduchých vzorců 3.3 a 3.4 a v závislosti na průměru navíjecího bubnu znázorněny v grafu (obr. 3.4). Symbol M_m označuje mechanický moment motoru nutný pro splnění hodnot síly, n jsou otáčky motoru nutné pro splnění hodnot rychlosti a r_b je poloměr navíjecího bubnu.

$$M_m = F_{max} \cdot r_b \text{ [Nm]} \quad (3.3)$$

$$n = \frac{v_{max}}{r_b} \frac{60}{2\pi} \text{ [ot/min]} \quad (3.4)$$



Obrázek 3.4: Graf momentu a rychlosti DC motoru v závislosti na průměru navíjecího bubnu.

Podle vypočtených parametrů motoru je jasné, že k dosažení síly je potřeba použít motor s převodovkou. Převodový poměr však volíme nejmenší možný, aby nebyla omezena rychlost navíjení.

Z důvodů rychlé dostupnosti a především ceny je použit starší stejnosměrný motor s permanentními magnety. Jediné údaje otiskované na motoru jsou elektrické – $24 \text{ V}/3 \text{ A}$. Ostatní parametry máme naměřeny pomocí dynamometru na výstupní hřídeli. Převodový

nominální napětí	U_n	24 V
nominální proud	I_n	3 A
příkon	P_1	72 W
konstanta motoru	$c\phi$	$\approx 0,8 \text{ V}\cdot\text{s}/\text{rad}$
převodový poměr		$\approx 12,31$
nominální otáčky	n_n	240 ot/min
nominální moment	M_n	2,5 Nm
záběrný moment	M_z	$\approx 10 \text{ Nm}$
odpor vinutí	R_a	1,5 Ω
indukčnost vinutí	L_a	$\approx 6 \text{ mH}$

Tabulka 3.1: Parametry použitého motoru.

poměr dvoustupňové převodovky je $\frac{517}{42} \approx 12,31$. Veškeré zjištěné parametry motoru jsou shrnuty v tabulce 3.1.

Příkon motoru P_1 dle rovnice 3.5 je v porovnání se zdánlivě potřebným výkonem mechanickým P_{mech} (rovnice 3.6) - vypočteným z odhadnutých parametrů síly F_{max} a rychlosti v_{max} řízení, velice malý. Musíme si však uvědomit, že maximální hodnoty síly a rychlosti ve skutečnosti nikdy nepůsobí zároveň, navíc můžeme motor krátkodobě přetěžovat. Také je dobré brát v potaz, že maximální hodnoty síly a rychlosti jsou jen odhadnuté a že simulátor je vyvíjen jako prototyp, který pomůže lépe určit hodnoty reálné. Symboly U_n a I_n označují nominální hodnoty napětí a proudu DC motoru.

$$P_1 = U_n \cdot I_n = 24 \text{ V} \cdot 3 \text{ A} = 72 \text{ W} \quad (3.5)$$

$$P_{mech} = F_{max} \cdot v_{max} = 150 \text{ N} \cdot 1,5 \text{ m}\cdot\text{s}^{-1} = 225 \text{ W} \quad (3.6)$$

Z parametrů motoru je jasné, že pro docílení odhadnuté síly a rychlosti řídicí šňůry bychom museli volit pokaždé jiný průměr navíjecího bubnu. Kompromisní hodnota průměru, s přihlédnutím na dostupné polotovary, je $d_b = 68 \text{ mm}$. Teoreticky dosažitelné hodnoty síly a rychlosti řídicí šňůry za nominálního napájení motoru jsou $v_{max} \approx 0,85 \text{ m/s}$ při $F_{max} \approx 70 \text{ N}$.

Použitý motor nemá zabudovaný snímač natočení, využili jsme proto nutnosti vyrobit navíjecí buben. Součástí bubnu jsou vyfrézované zuby použitelné jako clonky pro optický enkodér. Z hlediska použité technologie výroby je počet zubů stanoven na 18. S použitím klasického dvoukanalového snímání můžeme dosáhnout rozlišení 72 pulsů na otáčku. Se stávajícím průměrem bubnu můžeme rozlišit přibližně 3 mm na řídicí šňůře, což je postačující.

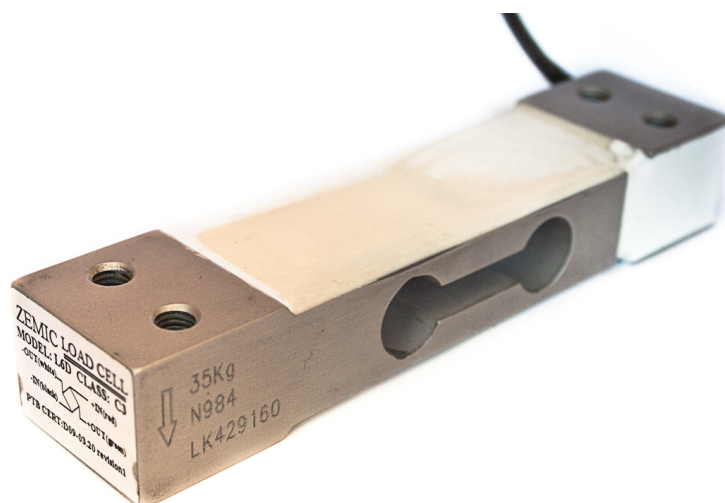
Celý motor je pomocí příruby uchycen k zadní části příčnicku rozpěry. Řídicí šňůra prochází vodícím okem a je přivázána k bubnu.

3.2.3. Snímání síly

Sílu tahu v řídicí šňůře můžeme díky použití DC motoru libovolně měnit, abychom však mohli sílu řídit, musíme ji i měřit.

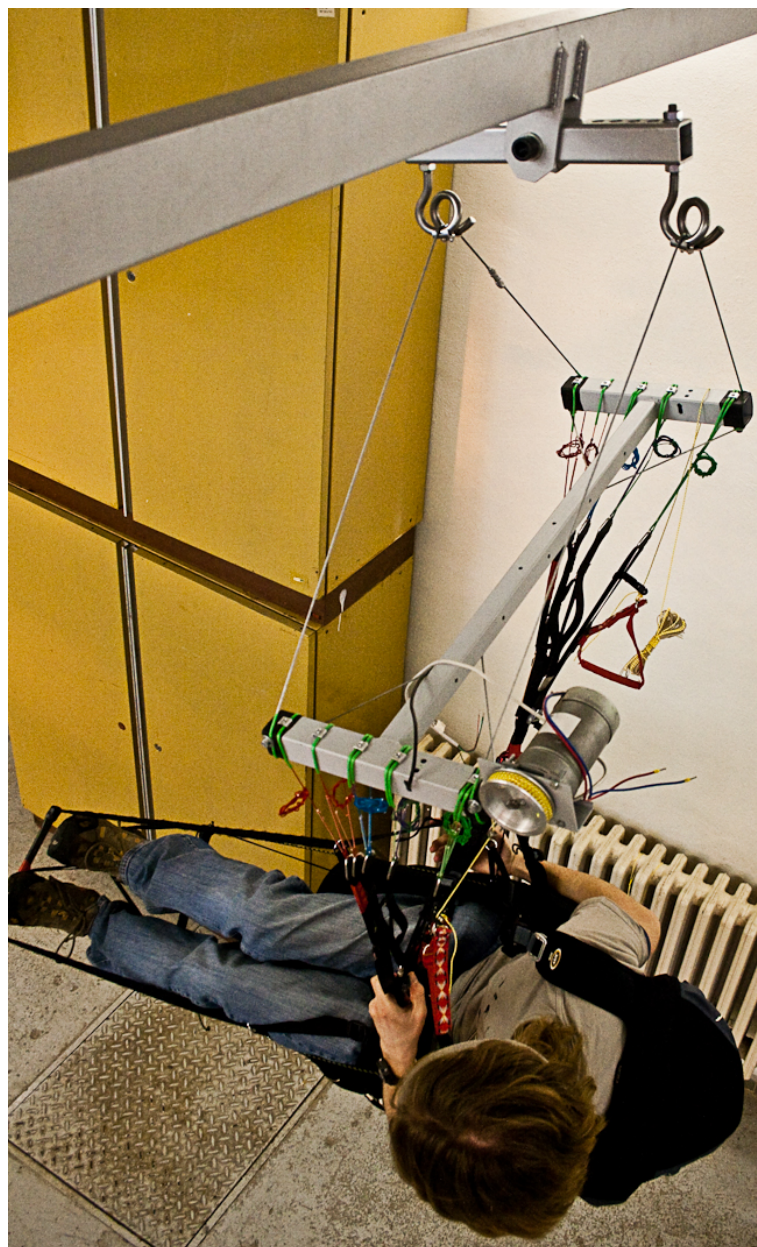
Díky použití DC motoru lze sílu nepřímo odhadovat z proudu protékajícího vinutím, získaná hodnota síly by však byla poměrně nepřesná. Nepřesnost dále zvyšuje použití převodovky – zanášení dalších třecích momentů a setrvačností. Tyto neduhy pomůže odstranit přímé snímání tahu v řízení.

Použité tenzometrické snímače jsou dnes již díky masové výrobě pro digitální váhy levné a jednoduché k použití. Jedná se o tzv. jednobodový snímač zatížení, obrázek 3.5. Na vážícím profilu z hliníkové slitiny jsou nalepeny čtyři tenzometry zapojené do měřícího mostu. Snímač firmy Zemic je schopen vážit do 35 kg a snést přetížení bezpečně na 52,5 kg, popřípadě extrémně až 105 kg. Jednobodové snímače zatížení svojí konstrukcí nejsou citlivé na mimoosové zatěžování (krut).



Obrázek 3.5: Jednobodový snímač zatížení.

Snímač je vložen mezi přírubu motoru a rozpěru (uchycen vždy dvojicí šroubů M6), tělo snímače je zasunuto do profilu příčníku pro zajištění lepší mechanické ochrany.

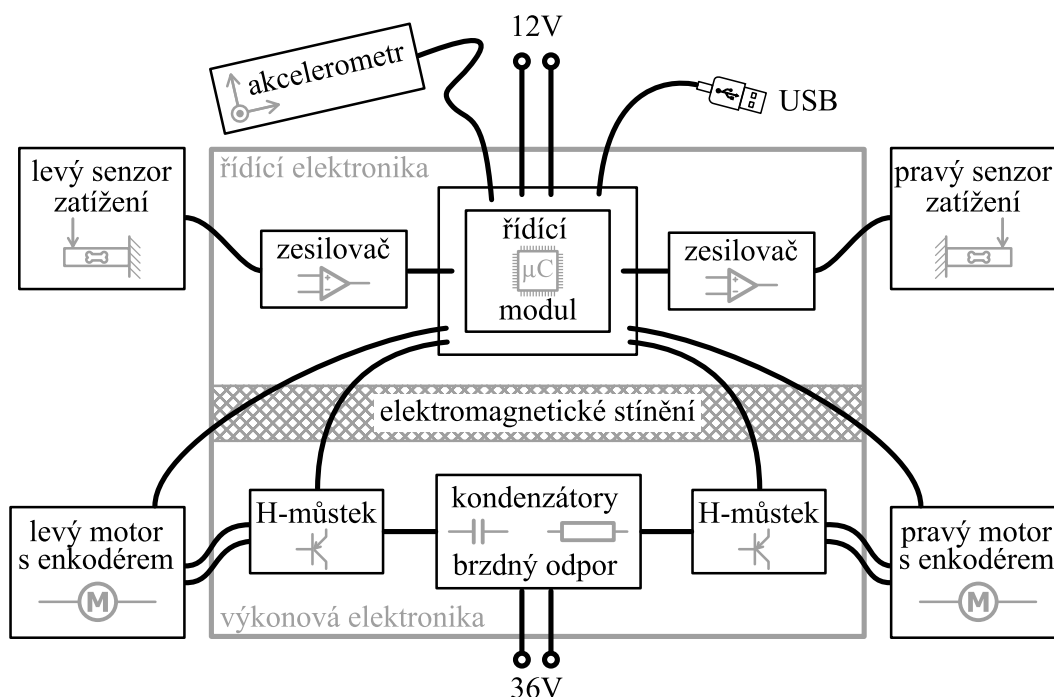


Obrázek 3.6: Fotografie navržené konstrukce.

4. Elektronika

Abychom mohli řídit motory a měřit hodnoty ze senzorů, potřebujeme pro ně obsluhu v podobě elektronických obvodů. Veškerá elektronika simulátoru se dá rozdělit do dvou hlavních okruhů - výkonová a řídicí. Všechny obvody jsou také navrhovány v podobě modulů, které jsou připojeny k jednomu hlavnímu (řídicímu), který provádí veškerou jejich obsluhu a komunikuje s nadřazeným PC.

V rámci diplomové práce jsou všechny obvody také vyrobeny. Plošné spoje jsou navrženy v programu EAGLE, soubory jsou součástí přílohy.



Obrázek 4.1: Schéma rozvržení elektroniky.

Fyzické umístění jednotlivých prvků elektroniky ukazuje schéma (obr. 4.1). Řídicí elektronika spolu s H-můstky a brzdovým odporem je umístěna na střed rozpěry, nachází se tak v pomysleném středu veškerých elektronických periferií umístěných na konstrukci padákového simulátoru. Jediná elektronika umístěná mimo samotný trenažér jsou zdroje. Napětí od zdrojů je přiváděno pomocí kabelů, vedených vrchem přes závěsnou konstrukci, tak aby neomezovaly pohyb konstrukce simulátoru.

4.1. Výkonová elektronika

Výkonovou elektronikou v našem případě rozumíme zdroje a elektroniku spojenou s napájením motorů. Výkon aktuálně použitého motoru není nijak velký, nicméně motory budou dva

a s přihlédnutím na jejich přetěžování se dostáváme k proudovému odběru řádově deseti ampér. Elektronika by také měla být pokud možno připravena na použití motorů silnějších.

4.1.1. H-můstky

Při návrhu vycházíme z [10]. Jelikož potřebujeme elektrický stroj používat ve třech ze čtyř pracovních kvadrantů, je nutné použít pro napájení čtyřkvadrantový měnič, tzv. H-most. Řízen bude signály PWM z mikrokontroléru.

Pro návrh H-můstku je rozhodující napětí a proud, které musí být schopen dodávat. Podle toho jsou pak vybrány vhodné spínací prvky. V našem případě z důvodu použití nízkého napětí nejspíše půjde o spínací prvky typu MOSFET, tedy unipolární tranzistory.

Nominální napětí motorů je 24 V , pokud chceme motor krátkodobě přetěžovat, bude lépe použít napětí vyšší. S vyšším napětím budeme schopni, podle známého vztahu pro indukované napětí u_i DC motoru 4.1, dosahovat vyšších otáček (resp. úhlových rychlostí ω). Abychom mohli motor přetěžovat i do oblastí vyššího momentu na hřídeli M , rovnice 4.2, bude třeba zdroj, který dokáže dodat vyšší než nominální hodnotu proudu i (více než 3 A). Symbol $c\phi$ označuje napěťovou a momentovou konstantu stejnosměrného motoru s permanentními magnety.

$$u_i = c\phi \cdot \omega \quad (4.1)$$

$$M = c\phi \cdot i \quad (4.2)$$

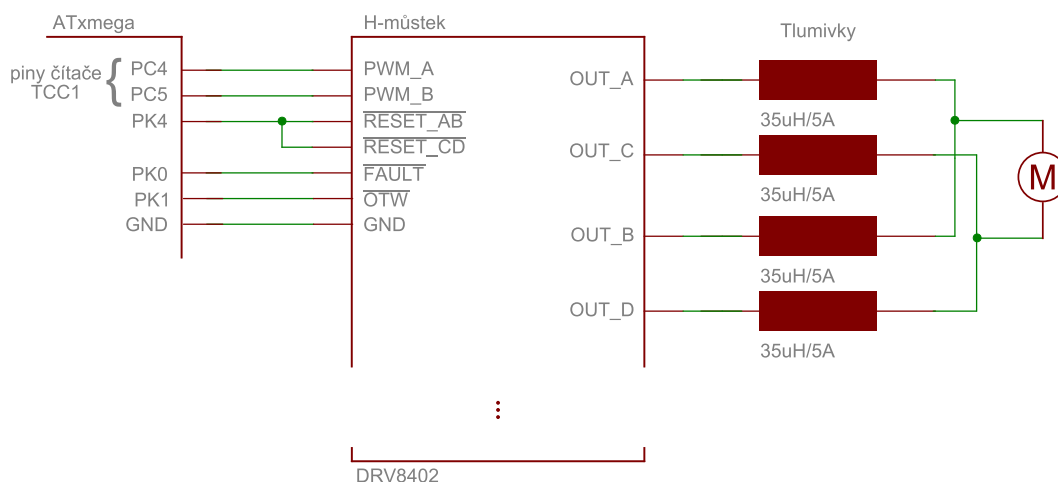
Při přetěžování vyšším napětím hrozí motoru pouze elektrické proražení izolace vinutí. Použité napětí bude stále dostatečně nízké a průraz izolace nehrozí. Horší situace plyne z proudového přetěžování. Průchodem elektrického proudu vinutím motoru se samotné vinutí ohřívá Jouleovými ztrátami a při překročení určité teploty může dojít k tepelnému zničení izolace vinutí. Přetěžování motoru proudově je možné jen v krátkých časových úsecích, kdy větší Jouleovy ztráty pojme tepelná kapacita vinutí.

Pokud nechceme napájecí zdroj vyrábět, bude lepší užít standardní hodnotu napětí 36 V – což je nejbližší vyšší k nominálním 24 V . Odebíraný proud předimenzovaný kvůli přetěžování a násobený dvěma (používáme 2 motory) činí kolem 8 A . Celkový výkon výkonového zdroje by tedy měl být asi 300 W .

Pro samotný měnič je nejlepší, vzhledem ke zvoleným parametrům napájení, použít jako spínací prvky unipolární tranzistory. Jsou pro uvažované hodnoty napětí a proudů dobře dostupné, levné a disponují nejlepšími vlastnostmi, jako rychlostí sepnutí/rozepnutí a odporem v sepnutém stavu. Při návrhu měniče je dobré mít na mysli jejich náchylnost na přepětí a volit součástky dimenzované alespoň na dvojnásobek spínaného napětí. Také spínaný proud může, například při rozjezdu nebo reverzaci motoru, narůst na několikanásobně vyšší hodnoty nežli nominální.

Pro naše účely bude jednodušší využít integrovaný obvod DRV8402 firmy Texas Instruments. Jedná se o dvojitý H-můstek složený z tranzistorů MOSFET s budiči a několika ochranami. Můstek je schopen spínat napětí až 50 V a proud 5 A (špičková hodnota 12 A) jednou větví a pracovat na frekvenci až 500 kHz.

Obvod DRV8402 je zapojen v režimu, kdy oba H-můstky pracují paralelně, můžeme takto spínat 10 A (špičkově 24 A). Integrovaný obvod potřebuje pro buzení tranzistorů napájení 12 V. Má automatickou ochranu proti přehřátí, které signalizuje při 125 °C a při 150 °C odstavuje všechny tranzistory. Dále automaticky detekuje podpětí při napájecím napětí pod 9,8 V a odstavuje všechny tranzistory, při nedostatečně vysokém napětí pro buzení tranzistorů by nemuselo docházet k jejich plnému sepnutí. Poslední ochranou, kterou DRV8402 disponuje, je automatické omezování proudu, hodnota proudu, při kterém začne omezování pracovat, je uživatelsky nastavitelná. V našem případě je nastavena na 9,3 A. Pro signalizaci možných poruch má integrovaný obvod dva logické výstupy, piny \overline{FAULT} a \overline{OTW} ¹. DRV8402 automaticky řeší i tzv. dead-time², ovládání je velice jednoduché – H-můstek řídíme pouze dvěma signály PWM (pro každou větev), piny PWM_A a PWM_B . Logická jednička znamená sepnuté horní tranzistory, logická nula sepnuté dolní. Dále jsou k dispozici dva signály RESET (pro každou větev zvlášť – piny $\overline{RESET_AB}$ a $\overline{RESET_CD}$), které vyřadí z provozu tranzistory příslušné větve.



Obrázek 4.2: Zjednodušené schéma zapojení H-můstku – připojení k mikrokontroléru.

Elektronické zapojení je poměrně jednoduché, DRV8402 nepotřebuje mnoho externích prvků. Většina jsou pouze blokovací kondenzátory. Protože tranzistory H-můstku jsou velice kvalitní a dosahují veliké strmosti spínaného napětí:

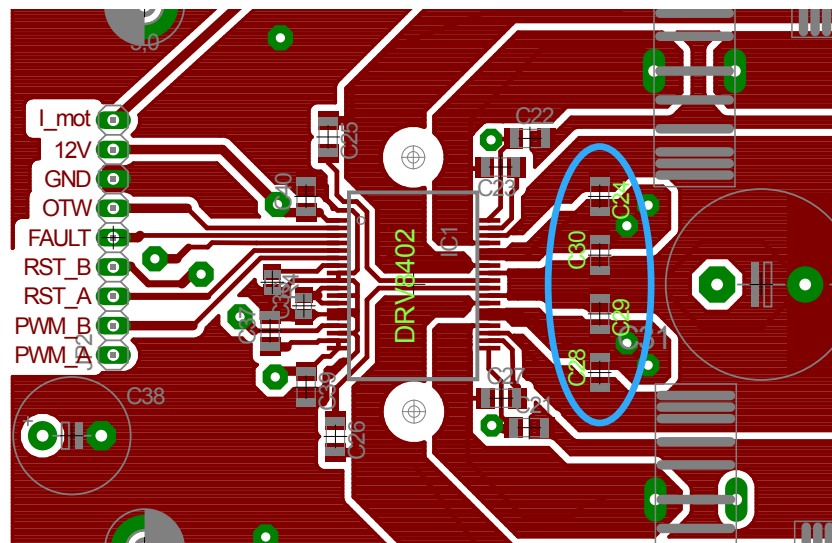
$$\frac{dU}{dt} \approx 4 \text{ kV}/\mu\text{s}$$

¹OverTemperature warning

²Dead-time je časová prodleva mezi sepnutím horního a spodního tranzistoru jedné větve.

je důležité, pro snížení kapacitních proudů a rušení, zapojit tlumivku do série ke každému výkonovému výstupu. Tlumivky musí být dostatečně dimenzovány k procházejícímu proudu, aby nedocházelo k jejich přesycování a ztrátě funkce. Elektrické zapojení je díky integrovanému obvodu DRV8402 velice jednoduché, náročnější je však jednotlivé výkonové prvky správně umístit na desku plošného spoje. Popsané parametry jsou převzaty z [11].

Obrázek 4.3 ukazuje výřez desky plošných spojů kolem integrovaného obvodu DRV8402. Důležité je fyzicky umístit keramické blokovací kondenzátory $C24$ a $C28$ až $C30$ co nejbližší napájecím výkonovým vývodům dvojitého H-můstku. Paralelně je připojen i velký elektrolytický kondenzátor s nízkým vnitřním odporem.



Obrázek 4.3: Výřez vrchní vrstvy DPS kolem integrovaného obvodu DRV8402. Označené blokovací kondenzátory je třeba umístit co nejbližší H-můstku.

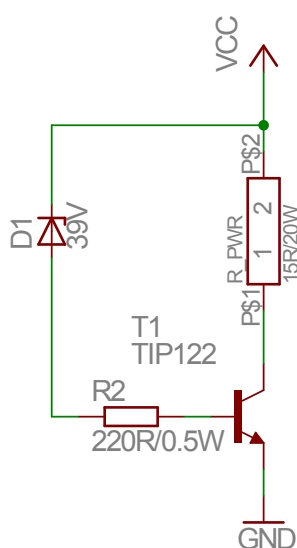
Kromě H-můstku je na stejném plošném spoji i senzor proudu ACS712 firmy Allegro MicroSystems [1]. Je připojen do série na jeden z výstupů H-mostu pro DC motor. Senzor je založen na principu Hallovy sondy, díky tomu je analogový výstup snímače od výkonového obvodu galvanicky oddělen. Napájen je 5 V , které pro něj vyrábí z 12 V úrovně lineární napěťový regulátor 78L05. Snímač má rozsah $\pm 20\text{ A}$ a citlivost 100 mV/A . Výstup snímače pro hodnotu proudu 0 A je polovina napájecího napětí, čili $2,5\text{ V}$. Výstupní signál se pohybuje v poměrně vysokých hodnotách napětí, takže před přivedením na A/D převodník jej bude nutno snížit napěťovým děličem.

4.1.2. Brzdny odpor

Použité DC motory jsou již poměrně výkonné a mají nezanedbatelný moment setrvačnosti. Proto například při reverzaci nebo při chodu motoru v generátorovém režimu (rychlé stahování řídicí šňůry proti silovému působení motoru) mohou v jisté míře vyrábět elek-

trickou energii a zvyšovat takto napětí (na kondenzátorech) meziobvodu. Pokud nebude vyrobená energie někde využita, může způsobit takový nárůst napětí, který v krajním případě poškodí prvky elektrického obvodu a nebo vyřadí z činnosti celý výkonový zdroj.

Zabránit přepětí meziobvodu má tzv. brzdny odpor. Jedná se o výkonový odpor, který je analogovými prvky automaticky připojován mezi napájecí svorky meziobvodu a „zkratováva“ je. Elektrické schéma zapojení odporu ukazuje obrázek 4.4. Napětí meziobvodu kontroluje Zenerova dioda $D1$ s průrazným napětím $U_Z = 39\text{ V}$, při překročení Zenerova napětí začíná vést proud, který vstupuje do báze Darlingtonova tranzistoru $T1$. Vybuzený tranzistor s velkým zesílením dovoluje průchod kolektorovému proudu, čímž přes brzdny odpor R_{PWR} zkratováva výkonové svorky meziobvodu. Před bází Darlingtonova tranzistoru je připojen odpor $R2$, který má za úkol omezovat bázový proud v případě vyššího nárůstu napětí meziobvodu.



Obrázek 4.4: Elektrické schéma zapojení brzdnyho odporu.

Kromě této automatické hardwarové ochrany před přepětím je napětí obvodu přivedeno k řídicímu modulu. Mikrokontrolér může napětí monitorovat a v případě potřeby provést další softwarové zásahy (např. odstavení H-můstku).

Na DPS s brzdnyho odporem je umístěno také pole elektrolytických kondenzátorů, které mají za úkol blokovat napětí výkonového zdroje blízko pulsního proudového odběru H-můstku. Odstraňují tak parazitní indukčnost delší kabeláže. Kvůli zvýšení proudové propustnosti je lépe použít více menších kondenzátorů zapojených paralelně, v našem případě je zapojeno 10 elektrolytických kondenzátorů, každý o kapacitě $330\ \mu\text{F}$.

4.1.3. Zdroje

Neméně důležitá část výkonové elektroniky jsou zdroje. Pro výkonovou část jsou parametry zdroje již jasné, viz podkapitola 4.1.1. Výroba výkonového AC/DC zdroje by byla

sama o sobě těžká úloha, jednodušší bude využít snadno dostupné průmyslové zdroje stejnosměrného napětí. Použitý zdroj má jmenovitou hodnotu napětí 37 V a maximální proudový odběr 8 A .

Zdroj však potřebuje i řídicí elektronika. Samotný mikrokontrolér a jeho periferie potřebují pro svůj chod $3,3\text{ V}$, avšak pro napájení H-můstků a senzorů zatížení (viz podkapitola 4.2.2) potřebujeme napětí 12 V . Nabízí se možnost získávat 12 V z již usměrněného napětí 37 V , při použití lineárního napěťového regulátoru by však vznikaly neúnosně velké výkonové ztráty. Řešením je použití spínaného zdroje, ať už síťového nebo DC/DC napojeného na výkonový zdroj. Proudový odběr celé řídicí elektroniky by neměl překročit $0,5\text{ A}$, což odpovídá výkonu zdroje nad 5 W . Získat nižší hladiny napětí pro mikrokontrolér atd. již jednoduchou lineární regulací není problém. Jako prozatímní řešení používáme 12 V větev PC zdroje.

4.2. Řídicí elektronika

Pojem řídicí elektronika v našem případě označuje krom řídicího mikrokontroléru také elektrické obvody obsluhující senzory (či senzory samotné) a obvody pro komunikaci s nadřazeným PC.

Z důvodu náchylnosti řídicích obvodů na rušení plynoucí z jejich umístění blízko výkonové elektroniky (tím spíše že je využívána pulsně šířková modulace) je dobré zajistit elektromagnetické stínění. Nejjednodušší je vložit mezi výkonovou a řídicí část plech vodivý magneticky i elektricky, např. běžný železný (resp. ocelový). Musí být dostatečně tlustý, aby nedocházelo k jeho magnetickému přesycování a elektricky uzemněný.

Na prototypu máme elektroniku umístěnou na desce přišroubované k ocelové rozpěře. Rozpěra rozděluje elektroniku na řídicí a výkonovou a sama tak působí jako stínění. Při rozmístění na desce jsou navíc obě části od sebe poměrně daleko.

4.2.1. Mikrokontrolérový modul

Řídicí mikrokontrolér musí zvládat několikero činností. Konkrétně se jedná o:

- získávání dat ze signálů senzorů, jak analogových tak digitálních
- PWM modulaci pro dva DC motory (resp. pro dvojici H-můstků)
- komunikaci s PC pomocí USB
- obsluhu signalizačních LED atp.
- výpočty regulace pro dva DC motory
- výpočty filtrů či jiného zpracování dat ze senzorů.

Zvolený mikrokontrolér by měl, kvůli jednodušší a spolehlivější implementaci, mít pro tyto činnosti hardwarovou podporu a samozřejmě dostatečný výpočetní výkon.

Všechny požadavky dobře splňují mikrokontroléry z rodiny ATxmega firmy Atmel [4]. Přesto, že mikrokontroléry mají velké množství modulů a periférií, jsou postaveny na jednoduchých 8-bitových procesorech. Jediné, čím mikrokontroléry nedisponují, je hardwarová podpora USB. Mají však podporu pro USART, který lze pomocí specializovaných integrovaných obvodů jednoduše na USB převést. Pro použití je vybrán typ ATxmega128A1, skupina ATxmega A1 je totiž z rodiny nejstarší a má již dostatek podpory z hlediska dostupných programátorů či dostupnosti mikrokontrolérů samotných. V prototypu je použita jednoduchá vývojová deska osazená pouze samotným mikrokontrolérem s nezbytnými blokovacími kondenzátory u napájecích pinů; krystalovými oscilátory o hodnotách 8 MHz (pro taktování jádra procesoru) a $32,768\text{ kHz}$ (pro čítač reálného času); a napěťovým regulátorem pro napájení mikrokontroléru (díky němu můžeme pro napájení použít napětí z širokého rozsahu od nominálního $3,3\text{ V}$ až po 15 V). Na vývojové destičce jsou na standardní kolíky vyvedeny všechny piny mikrokontroléru a ještě zvlášť piny pro programátor. Využitím vývojové desky odpadá problém s výrobou vlastní desky a osazením integrovaného obvodu s malou roztečí pinů, a vůbec ulehčuje manipulaci při vývoji prototypu.

K vývojové desce s mikrokontrolérem je navržena rozvodná DPS, zajišťující vyvedení napájecích napětí a signálů k a od mikrokontroléru. Krom toho jsou na rozvodné desce umístěny také filtry, převodník USART \Leftrightarrow USB, napěťová reference A/D převodníku a signalizační LED.

Napájecí napětí $3,3\text{ V}$ pro mikrokontrolér, ale i připojované moduly, zajišťuje lineární napěťový regulátor na vývojové desce.

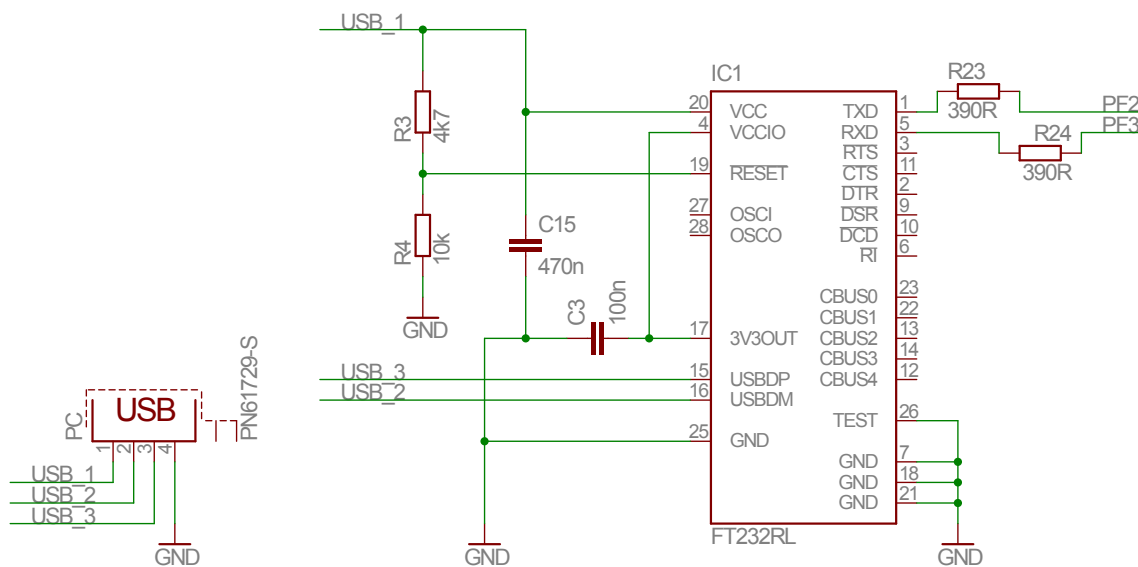
Moduly jsou k rozvodné desce připojovány pomocí kolíkových konektorů se zámkem, mají tak jednoznačnou možnost zasunutí a ochranu proti samovolnému vysunutí. Signály z jednotlivých modulů jsou přivedeny na patřičné piny, které dané využití umožňují, jak je popsáno v [3].

Analogové signály, tj. ze senzorů zatížení, z proudových senzorů, z akcelerometru a napětí meziobvodu, jsou přivedeny na piny „analogových“ portů A a B. Veškeré analogové vstupy prochází low-pass filtrem, kvůli vysokofrekvenčnímu odrušení. Keramické kondenzátory filtrů jsou umístěny vždy co nejbližší vstupním pinům A/D převodníku. Pro A/D převodník je zajištěno externí referenční napětí $2,5\text{ V}$ pomocí integrovaného obvodu REF2925 firmy Texas Instruments. S touto hodnotou referenčního napětí má 12-bitový převodník rozlišení $0,61\text{ mV}$. Analogové signály snímačů musí být upraveny tak, aby při měření nedocházelo k saturaci A/D převodníku, tedy aby maximální hodnoty přiváděného signálu byly do $2,5\text{ V}$. Zároveň by však také neměly být příliš nízké, aby využily celý rozsah.

Digitální signály jsou přiváděny na vstupně-výstupní piny mikrokontroléru, které jsou napojeny na dobře propracovaný systém událostí (event systém) s hardwarovou podporou. Kromě generování různých přerušení dokáže systém např. zpracovávat kvadraturní signál z enkodérů. Aby však kvadraturní signál z enkodérů mohl event systém mikrokontroléru zpracovávat musí být dva kanály enkodéru přivedeny na po sobě jdoucí piny jednoho portu. Enkodéry máme připojeny na piny PD0 až PD3. Dále jsou přivedeny signály chybových hlášení z H-můstku na piny PK0 až PK3 a výstupní signály pro reset H-můstků, piny PK4 a PK5. Připojený máme ještě jeden důležitý vstup (na pin PE2), určený pro stop tlačítko – pro potřeby nouzového odstavení výkonové elektroniky. Na piny PJ2 až PJ5 máme připojeny stavové LED, pro potřeby ladění a pro signalizaci stavů simulátoru.

Signály PWM pro řízení motorů generuje mikrokontrolér pomocí čítačů, každému přísluší určité piny mikrokontroléru – H-můstky máme připojeny na piny PC4, PC5 (čítač TCC1) a PE4, PE5 (čítač TCE1).

Vyvedeny máme také 2 sériové linky USART, na pinech PF2, PF3 (USARTF0) a PF6, PF7 (USARTF1). Piny USARTF1 jsou vyvedeny přímo na kolíky konektoru a jsou určeny pro komunikaci pomocí sériového portu počítače pro potřeby ladění aplikace, popřípadě pro kontrolní výpisy. USARTF0 slouží jako hlavní linka pro komunikaci s nadřazeným PC, na kterém poběží simulace letu. Kvůli kompatibilitě s moderními PC je sériová linka převedena na USB pomocí integrovaného obvodu FT232R firmy FTDI. Jedná se o jednu z velice jednoduchých metod získání plné kompatibility zařízení s rozhraním USB. Obrázek 4.5 znázorňuje jeho elektrické zapojení podle [6].



Obrázek 4.5: Elektrické schéma zapojení převodníku USB.

4.2.2. Zesilovač pro senzor zatížení

Použitý jednobodový snímač zatížení firmy Zemic má podle katalogového listu [14] zesílení tenzometrického mostu při plném zatížení 2 mV/V . Doporučovanému budicímu napětí 10 V odpovídá při zatížení 35 kg napětí na výstupu tenzometrického mostu 20 mV . Abychom byli schopni pomocí A/D převodníku v řídicím mikrokontroléru získávat rozumné hodnoty, musíme signál ze snímače zatížení zesílit.

Firma Texas Instruments vyrábí integrovaný obvod přímo určený pro práci s elektrickými odporovými mosty. Jedná se o obvod INA125, kromě zesilovače s diferenciálním vstupem v sobě ukrývá i regulovaný referenční zdroj napětí pro buzení mostu. Napájení zesilovače může být i nesymetrické³, což je výhodné, protože nemusíme řešit zdroj záporného potenciálu, navíc vzhledem k tomu že výstupní signál bude přiveden na A/D převodník schopný zvládat jen kladná napětí, není ani záporný výstup žádoucí.

Zesilovač umožňuje zesílení v rozsahu $4 \div 10000$ krát, dále umožňuje nastavit napěťový offset výstupu⁴ a posouvat tak nulu výstupu zesilovače, právě pro případy nesymetrického napájení. INA125 má čtyři úrovně referenčního napětí pro napájení mostu $1, 2, 4, 5, 10\text{ V}$. Zesilovač má také široké pásmo napájecího napětí – pro nesymetrické $2, 7 \div 36\text{ V}$. Napájení by však mělo být vždy alespoň o $1, 2\text{ V}$ vyšší než případné výstupní napětí a o $1, 25\text{ V}$ vyšší než použité referenční (budicí) napětí. Frekvenční propustnost zesilovače je závislá na velikosti zesílení a pohybuje se v rozsahu asi 150 kHz (při zesílení 4 krát) až po $0, 9\text{ kHz}$ (při zesílení 500 krát). Veškeré technické údaje převzaty z [12].

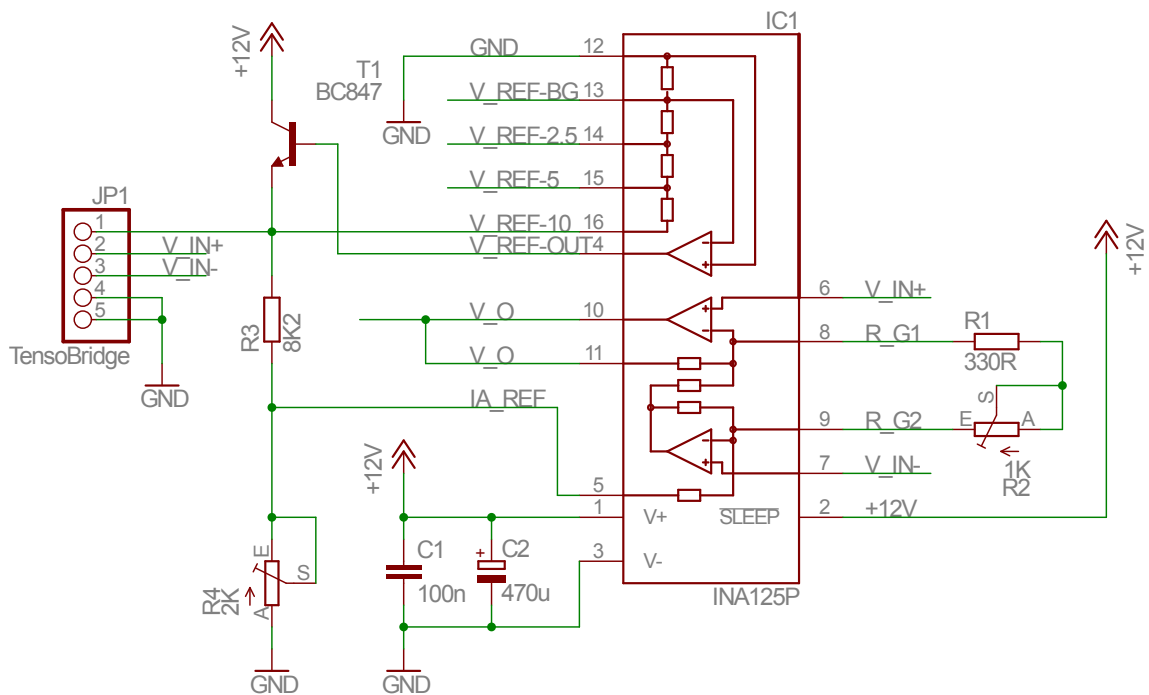
Na obrázku 4.6 je schéma zapojení integrovaného obvodu INA125. Jako budicí napětí tenzometrického mostu je použito výrobcem doporučených 10 V . Dobrý regulovaný zdroj poskytuje referenční napětí obvodu INA125, abychom jej však mohli použít musí být samotný INA125 napájen 12 V . Referenční napětí 10 V je proudově posíleno pomocí tranzistoru $T1$, krom napájení tenzometrického mostu je totiž také použito (po snížení nastavitelným odporovým děličem) jako offsetové napětí – přivedeno na pin 5 (signál $I_{A_{REF}}$). Proměnným odporem $R4$ můžeme offsetové napětí nastavovat v rozmezí přibližně $0 \div 1, 96\text{ V}$. Zesílení diferenciálního zesilovače lze nastavovat pomocí proměnného odporu $R2$, v rozmezí přibližně $49 \div 186$ krát.

4.2.3. Akcelerometr

Je použit akcelerometr firmy Analog Devices, typ ADXL327 (viz [2]). Jedná se o malý akcelerometr v SMD pouzdře se třemi osami a rozsahem $\pm 2g$. Díky malému rozsahu má poměrně velkou citlivost 420 mV/g a dobře se tak hodí pro snímání náklonu.

³Pro napájení stačí kladné napětí vůči zemi, na rozdíl od symetrického napájení, kde je potřeba napětí kladné i záporné. Nesymetricky napájený zesilovač však také nedokáže řídit výstupní napětí i pod nulový potenciál (zem).

⁴Hodnota napětí přičteného k výstupu zesilovače.



Obrázek 4.6: Elektrické schéma zapojení zesilovače (INA125).

Elektrické zapojení je velice jednoduché, ADXL327 totiž nepotřebuje žádné externí elektronické prvky kromě blokovacích kondenzátorů na napájení, které má hodnotu $3,3\text{ V}$. Výstupy jsou tři analogové signály odpovídající naměřenému zrychlení ze tří os, analogové hodnoty mají offset polovinu napájecího napětí. Na každém analogovém výstupu jsou interní sériové odpory velikosti přibližně $32\text{ k}\Omega$, low-pass filtry ke každému signálu jsou řešeny už pouze paralelním kondenzátorem. Kapacita kondenzátoru určuje šířku pásma výstupního signálu, v našem případě používáme kondenzátory o velikosti 470 nF , což zajišťuje šíři pásma propustnosti do 10 Hz (pro pokles o 3 dB). Kondenzátory však nejsou umístěny na DPS s akcelerometrem, ale (z důvodu lepší odolnosti proti rušení) až u řídicího mikrokontroléru co nejbližší pinům A/D převodníku.

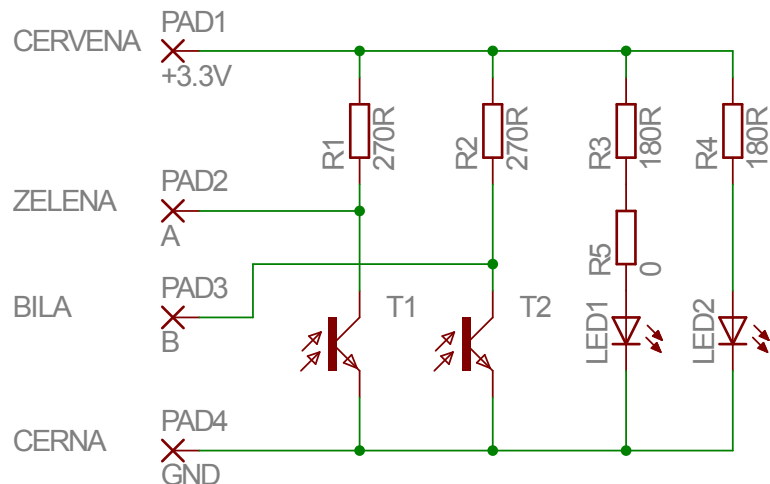
Podle nastavení A/D převodníku (viz podkapitola 4.2.1) můžeme teoreticky rozlišit až 688 dílků na 1 g , měření je však zatíženo šumem. Prakticky naměřená data po zpracování (tedy ve formě úhlu náklonu) a po softwarové filtraci dávají použitelné hodnoty s rozlišením $0,5^\circ$. Pro potřeby paraglidingového trenažéru je rozlišení dostatečné, zajišťuje kolem 50 úrovní v celém rozsahu vyklánění v sedačce a přibližně 30 úrovní pro průběh speed systému.

4.2.4. Enkodéry

Enkodéry jsou použity pro měření úhlové polohy motoru, čili přímo odpovídají hodnotě stažení řídicí šňůry. Nejsou použity komerční výrobky, které jsou poměrně drahé a těžko by se u nich řešilo mechanické připojení k motorům. Enkodéry jsou poskládány z diskretních

součástek, viz elektrické schéma obr. 4.7. Stínící clonky enkodéru jsou součástí navíjecího bubnu jak je popsáno v podkapitole 3.2.2.

Hodnoty odporů $R3$ a $R4$ omezujících proud infračervenými LED a pull-up odpory $R1$ a $R2$ u fototranzistorů jsou stanoveny tak, aby elektronika enkodéru byla kompatibilní s $3,3V$ napájením i logikou mikrokontroléru.



Obrázek 4.7: Elektrické schéma fotoelektrického enkodéru.

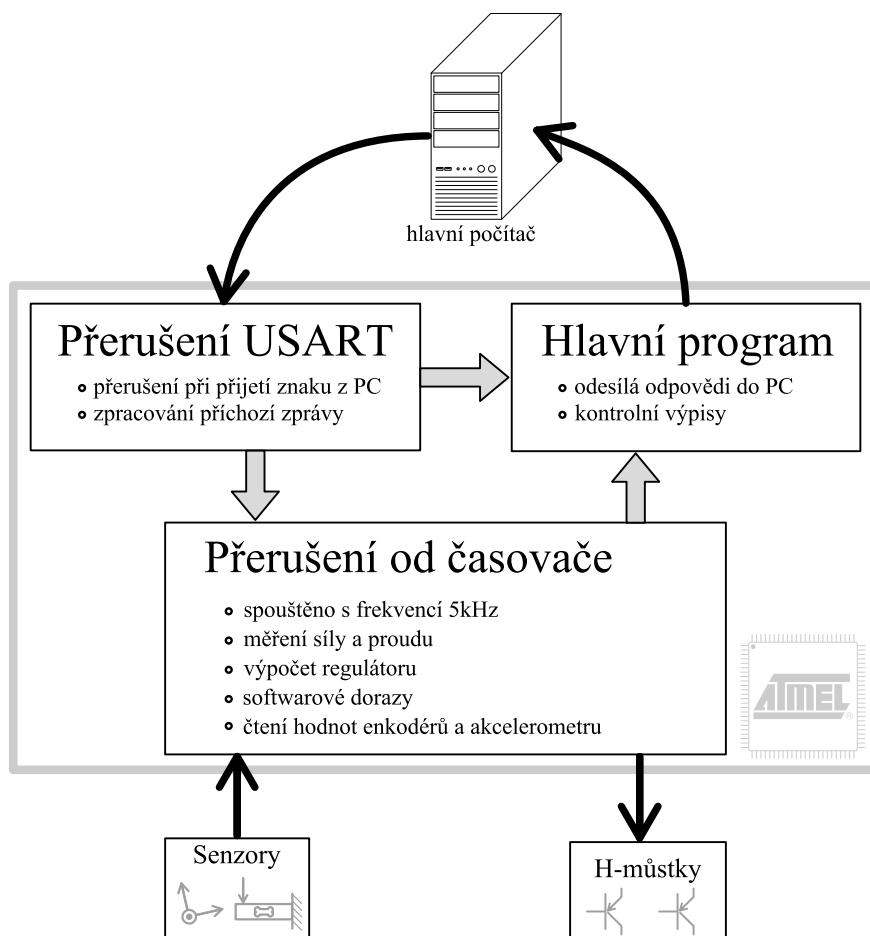
Oba výstupy enkodéru jsou ještě vyhlazeny filtrem typu dolní propust, který je z důvodu lepší odolnosti proti rušení umístěn až co nejdříve vstupním pinům mikrokontroléru. Filtř je složen ze sériového odporu o hodnotě $1\text{ k}\Omega$ a paralelního keramického kondenzátoru o kapacitě 100 nF . Časová konstanta filtru τ má hodnotu:

$$\tau = 1\text{ k}\Omega \cdot 100\text{ nF} = 0,1\text{ ms}.$$

5. Programová obsluha

Poslední prvek, který je nutno navrhnout, je program (někdy též nazývaný firmware) pro řídicí mikrokontrolér. Firmware je závislý od typu použitého mikrokontroléru. Musí být složen jen z příkazů dostupných v instrukční sadě procesoru. Odlišovat se může také podle bitové šířky slova procesoru.

Firmware je psaný v jazyku C, čímž je alespoň v jisté míře zajištěna jeho přenositelnost pro případ použití jiného mikrokontroléru. Zdrojový kód je zpracován pomocí překladače a assembleru vývojového prostředí IAR Embedded Workbench. Výstupem je soubor typu Intel-Hex, který je možno přímo pomocí programátoru nahrát do programové paměti mikrokontroléru. Zdrojový kód i zkompilovaný výstup jsou v elektronické příloze.



Obrázek 5.1: Schéma běhu programu.

Na obrázku 5.1 je znázorněno schéma běhu programu. Úlohy jsou rozděleny mezi hlavní program, obsluhu přerušování od čítače a obsluhu přerušování při příchodu znaku z PC. V obsluze přerušování od čítače, která je spouštěna s frekvencí 5 kHz provádíme ty části výpočtu, u kterých je nutno zajistit přesné časování a to v následujícím sledu:

- měření hodnot sil v řídicích šňůrách a proudů protékajících motory

- výpočet regulátoru síly
- čtení hodnot z enkodérů
- zajištění softwarových dorazů¹ motorů
- odečítání hodnot z akcelerometru

V obsluze přerušení při příchodu znaku z PC je příjmová část komunikačního protokolu (po přijetí celé zprávy dochází k nastavení příznaku pro odeslání odpovědi). A konečně v hlavním programu běží procesy, u kterých nezáleží na přesném časování a procesy zabírající delší časové úseky. Konkrétně se jedná o odesílací část komunikačního protokolu a případné odesílání kontrolních zpráv, např. pro ladění softwaru.

5.1. Obsluha senzorů

Jedním z hlavních úkolů programu je získání, zpracování a poté odeslání dat ze senzorů. Na simulátoru jsou dvě skupiny senzorů. V první skupině jsou senzory ovládacích prvků, což jsou enkodéry u řídicích šňůr a akcelerometr pro zjištění náklonu. Ve druhé skupině jsou senzory používané pro řízení motorů, tedy snímače zatížení (síly v řídicích šňůrách), senzory proudu procházejících motory a měření napětí meziobvodu. Za řídicí senzor můžeme považovat také bezpečnostní stop tlačítko.

Ačkoliv pro funkci simulátoru jsou nejdůležitější data ze senzorů ovládacích prvků, pro chod samotné hardwarové části jsou mnohem důležitější data z řídicích senzorů. Pro zajištění kvalitní regulace motorů je potřeba snímat řídicí data v přesných intervalech na poměrně vysoké frekvenci.

Senzory můžeme také rozdělit podle jejich výstupu na analogové a digitální. Zpracování jejich signálu se podle toho značně liší.

5.1.1. Analogové senzory

Senzory analogové jsou přivedeny na vstupy analogově digitálních převodníků. ATxmega128A1 má převodníky dva, každý z nich má 4 multiplexované kanály. Na A/D převodník A jsou připojeny senzory síly a senzory proudu, na převodník B jsou přivedeny všechny tři osy akcelerometru a napětí meziobvodu.

Nastavení převodníku provádíme pomocí ovládacích registrů. Oba převodníky máme nastaveny stejně:

- referenční napětí 2,5 V na pinu AREFA
- hodinový zdroj pro běh převodníku (maximální možný) 2 MHz

¹Podle hodnot z enkodérů zamezuje dalšímu navíjení, resp. odvíjení řídicí šňůry.

- převod probíhá jednorázově (na vyžádání)
- rozlišení převodníku 12-bitů
- neznaménkový převod (převádíme pouze kladná napětí)

Pomocí registrů také přiřazujeme piny k jednotlivým kanálům převodníku.

Příklad nastavení převodníku A, díky využití od výrobce dobře propracovaného systému maker je nastavení patrné i ze samotného zdrojového kódu:

```

1 // ADCA - mereni sily a proudu
2 ADCA.REFCTRL = ADC_REFSEL_AREFA_gc; // nastaveni reference
3 ADCA.PRESCALER = ADC_PRESCALER_DIV16_gc; // nastaveni preddelicky hodin
4 ADCA.CTRLA = ADC_ENABLE_bm; // povoleni A/D prevodniku
5 ADCA.CHO.CTRL = ADC_CH_INPUTMODE_SINGLEENDED_gc; // neznamenkovy prevod
6 ADCA.CHO.MUXCTRL = ADC_CH_MUXPOS_PIN1_gc; // prirazeni pinu ke kanalu 0
7 ...

```

Pro spuštění převodu je nadefinována funkce *convert_A()*, která spouští převod všech 4 kanálů převodníku a vyčká na jeho dokončení. Funkce nemá žádné vstupy ani výstupy, výsledné hodnoty můžeme libovolně číst z výsledkových registrů příslušného kanálu, např. *ADCA.CHORES*.

Senzory řízených veličin

Hodnoty nutné pro řízení motorů měříme vždy před výpočtem regulátoru. Kvůli potlačení šumu měřených veličin je každá naměřená hodnota průměrem z osmi po sobě jdoucích vzorků. Od naměřených hodnot jsou ještě odečteny offsety, tak abychom získali znaménkovou hodnotu odpovídající měřené veličině.

Zdrojový kód získávání hodnot síly v levé řídicí šňůře a proudu v levém motoru, klíčové slovo *s16* označuje celočíselný 16-bitový znaménkový (signed) datový typ:

```

1 // mereni sily a proudu - prumer osmi vzorku
2 s16 suma = 0; // deklarace souctovych promennych
3 s16 suma2 = 0;
4 for(u8 i=8; i; i--) //ziskani 8 vzorku
5 {
6     convert_A();
7     suma += SILA_1; // pricteni prave namereneho vzorku
8     suma2 += PROUD_1;
9 }
10 force = suma>>3; // deleni osmi (bitovym posunem)
11 force = force - 2000; // odedcteni offsetu sily

```

```

12 | current = suma2>>3; // deleni osmi
13 | current = current - 2250; // odedcteni offsetu proudu

```

Akcelerometr

Hodnoty z akcelerometru měříme na konci zpracování přerušení od časovače. Protože naklání a naklápění akcelerometru neprobíhá při provozu nijak rychle a malé dopravní zpoždění hodnot z akcelerometrů nevadí, můžeme si dovolit použít jako filtr setrvačný člen prvního řádu s relativně velkou časovou konstantou. Kromě šumu se tak zbavíme i vlivu „parazitních“ zrychlení vznikajících pohybem konstrukce.

Filtrování digitálním setrvačným článkem probíhá podle rovnice 5.1 vycházející z [7]. Symbol y_{k+1} označuje výstupní hodnotu z filtru, y_k hodnotu z předchozího kroku filtrace, x_k aktuálně naměřenou hodnotu a C_f filtrační konstantu. Konstantu C_f lze vypočítat podle vzorce 5.2, je závislá na periodě (délce kroku) digitálního filtru T_f a požadované časové konstantě filtru τ .

$$y_{k+1} = y_k - C_f \cdot y_k + C_f \cdot x_k \quad (5.1)$$

$$C_f = \frac{T_f}{\tau} \quad (5.2)$$

Protože výpočet filtru probíhá s frekvencí 5 kHz a časovou konstantu filtru volíme přibližně $0,5\text{ s}$, vychází konstanta filtru

$$C_f = \frac{1}{2500}.$$

Mikrokontrolér nemá hardwarovou podporu pro dělení, čili takovýto výpočet běh programu neúměrně zpomaluje. Z důvodu implementace upravíme hodnotu konstanty na

$$C_f = \frac{1}{2048},$$

číslo 2048 je totiž celočíselnou mocninou čísla 2 a v takovýchto případech můžeme dělení v mikroprocesoru nahradit jednoduchým bitovým posunem. Časová konstanta filtru se sice mírně posune, ale na její přesné hodnotě nám příliš nezáleží. Nová hodnota časové konstanty je

$$\tau = 0,4096\text{ s}.$$

Aby při dělení nedocházelo k velkým zaokrouhlovacím chybám provádíme výpočet s 32-bitovými proměnnými. Až při čtení výsledku převádíme hodnotu zpět na 16-bitovou.

Implementace filtru v jazyce C, klíčové slovo `s32` označuje celočíselný 32-bitový znaménkový (signed) datový typ:

```

1 | // filtr akcelerometru
2 | s32 akcel_x = 0; // promena pro prubezne ukladani hodnoty

```

```

3 | convert_B(); // spusteni A/D prevodu
4 | // vypocet low-pass filtru
5 | akcel_x -= akcel_x>>11;
6 | akcel_x += ((s32)AKCEL_X<<16)>>11;
7 |
8 | s16 akcel_x_res = akcel_x>>16; // vysledna filtrovana hodnota

```

5.1.2. Digitální senzory

Typů digitálních signálů je dlouhá řada a v poslední době se stále rozrůstá. Každý typ si žádá patřičné zpracování, s porozuměním jeho podstaty. V našem případě používáme typy dva. Kvadrurní signál z enkodérů a logický signál ze stop tlačítka.

Enkodéry

Signál enkodéru se skládá ze dvou kanálů. Při otáčení hřídelky enkodéru v obou kanálech probíhají podobné obdélkové signály, vzájemně jsou však fázově posunuty o 90°. Podle jejich pořadí můžeme určit směr otáčení a každá logická změna v kterémkoli kanálu odpovídá otočení o jeden dílek enkodéru.

ATxmega disponuje hardwarovým zpracováním kvadrurních signálů. Zvyšuje tak spolehlivost a zjednodušuje práci s nimi. Nevýhodou může být pouze 16-bitový čítač pulsů enkodéru, pro naše účely však dostačuje, jelikož máme enkodéry s nízkým počtem pulsů a do pracovní oblasti se vejde přibližně sedm otáček. Dekódování probíhá pomocí jednoho kanálu event systému, vstupem do něj jsou piny připojené na enkodér. Na výstup tohoto kanálu je připojen jeden z 16-bitových čítačů. Hodnota čítače přímo odpovídá měřenému úhlu natočení.

Postup nastavení registrů ATxmegy pro zpracování kvadrurního signálu je následující. Dva po sobě jdoucí piny jednoho vstupně-výstupního portu, na které jsou přivedeny kanály enkodéru, musí být nastaveny jako vstupní snímající nízkou úroveň. První z pinů musí být namapován na jeden z kanálů event systému. Příslušnému kanálu musíme povolit dekodování kvadrurního signálu a nastavit počet vzorků digitálního filtru kanálu, od jednoho až po osm. Dále musíme nastavit zdroj čítače na dekodování kvadrurního signálu a namapovat na použitý kanál event systému. Nakonec čítač spustíme.

Nastavení registrů pro dekodování enkodéru na levém motoru:

```

1 | // nastaveni pinu na snimani nizke urovne
2 | PORTD.PINCTRL |= PORT_ISC_LEVEL_gc;
3 | PORTD.PIN1CTRL |= PORT_ISC_LEVEL_gc;
4 | // prirazeni pinu na kanal 0 event systemu
5 | EVSYS.CHMUX = EVSYS_CHMUX_PORTD_PIN0_gc;
6 | // kanal 0 ev. sys. snimani kvadratury a dig. filt. z~8 vzorku

```

```

7 | EVSYS.CHCTRL = EVSYS_QDEN_bm | EVSYS_DIGFILT_8SAMPLES_gc;
8 | // citame kvadraturu a vstup z~kanalu 0 ev. sys.
9 | TCDO.CTRLD = TC_EVACT_QDEC_gc | TC_EVSEL_CHO_gc;
10 | // spusteni citace TCDO
11 | TCDO.CTRLA = TC_CLKSEL_DIV1_gc;

```

Stop tlačítko

Bezpečnostní tlačítko slouží pro nouzové zastavení motorů, respektive odstavení H-můstků. Po jeho stisknutí mikrokontrolér bezprostředně uvede H-můstky do stavu RESET, čímž všechny výkonové tranzistory přejdou do stavu vysoké impedance, tj. stav podobný rozpojení elektrického obvodu motoru. Činnost obvodu s brzdým odporem však zůstává zachována.

Signál stop tlačítka je přiveden na pin mikrokontroléru schopný generovat přerušení. Aby byla zajištěna rychlá odezva nastavíme nejvyšší prioritu přerušení.

5.2. Řízení motorů

Stejnoseměrné motory ovládáme pomocí napětí přiloženého na jeho svorky, jedná se o nejefektivnější způsob. Díky modulaci PWM můžeme napětí měnit plynule v rozmezí $-37 \div 37 V$. Motory řídíme tak, abychom dosáhli požadované hodnoty síly v řídicí šňůře. Regulátor si můžeme představit jako blok, do kterého vstupuje aktuálně naměřená hodnota a požadovaná hodnota síly tahu v řídicí šňůře, a na výstup posílá hodnotu napětí, které má být přiloženo na svorky motoru.

Požadovanou hodnotu síly nám posílá hlavní řídicí PC. Vzhledem k tomu, že simulace neběží na PC s real-time operačním systémem, časování příchozích zpráv není z podstaty pravidelné. Za aktuálních podmínek získáváme požadovanou hodnotu přibližně 65-krát za sekundu. Regulátor však běží na frekvenci $5 kHz$, průběh hodnoty požadované síly tak může obsahovat relativně velké skoky. Tyto skoky se mohou projevit i na výstupu z regulátoru a způsobit nepřírozené pulsace síly v řídicí šňůře. Z tohoto důvodu nefiltrujeme hodnotu měřené síly tahu, ale až hodnotu regulační odchylky, tedy hodnotu rozdílu požadované a aktuálně naměřené hodnoty síly. Takto jedinou filtrací omezíme šum snímače síly i skoky požadované hodnoty síly. Používáme filtr typu setrvačný článek 1. řádu, stejný jaký je popsán v podkapitole 5.1.1 v odstavci o akcelerometru, pouze s menší časovou konstantou:

$$\tau = 0,0512 s.$$

Hodnota je opět volena s důrazem na jednoduchý (rychlý) výpočet v mikrokontroléru – dělení pouze čísly rovnými celočíselné mocnině dvojky.

Odzkoušeny máme regulátory P a PI. Z nichž se lépe osvědčil jednodušší P regulátor. Zaváděním integrační složky se sice regulace (převážně pro ustálené hodnoty) zpřesňuje, avšak zvyšuje se setrvačnost pohybu řídicí šňůry. Dále budeme zkoušet PD regulátor, jehož implementace může být kvůli číslíkové derivaci složitější.

Aktuálně používaný P regulátor je laděn metodou pokus-omyl, hodnota zesílení

$$K_P = 1,$$

čili hodnota regulační odchylky je v nezměněné podobě posílána na výstup regulátoru. Soustava s tímto regulátorem nekmitá, hůře však zvládá nižší hodnoty sil, respektive rychlost navíjení šňůry při nižších hodnotách požadované síly je malá.

Implementace použitého regulátoru je natolik triviální, že nebudeme zdrojový kód uvádět, v případě zájmu si jej lze prohlédnout v elektronické příloze.

K řízení motorů patří i tzv. softwarové dorazy, které nahrazují funkci koncových spínačů. Na základě hodnot z enkodéru příslušného motoru zabráníme dalšímu navíjení či odvíjení šňůry. Při překročení nastavených mezí dovolíme pohyb pouze ve směru, který vrací motor do pracovní oblasti.

Implementace v jazyce C, pro lepší pochopení vysvětleme, že hodnota z enkodéru se při odvíjení šňůry z počáteční polohy pohybuje od 0 do záporných čísel a kladná hodnota napětí na motoru způsobuje navíjení:

```

1 // softwarove dorazy motoru_1
2 s16 enkoder = ENKODER_1; // nacteni hodnoty z enkoderu
3 if(enkoder >= 0) // vrchni doraz
4 {
5     if(voltage > 0)
6         voltage = 0;
7 }
8 else
9     if(enkoder <= -700) // spodni doraz
10        if(voltage < 0)
11            voltage = 0;
12 // vystup z regulatoru
13 motor_1(voltage);

```

Pracovní oblast je $(0; -700)$ v pulsech enkodéru. Funkce `motor_1()` je blíže popsána v následující podkapitole 5.2.1.

5.2.1. Generování PWM

Modulace PWM má několik parametrů. Předně je však dobré podotknout, že využijeme unipolární řízení H-můstků, které má dvojnásobnou frekvenci výstupního PWM vzhle-

dem k frekvenci spínání tranzistorů, čímž zajišťuje nižší zvlnění proudu procházejícího motorem. S vyšší frekvencí se také výrazně snižuje akustický hluk. [10]

ATxmega má hardwarovou podporu PWM, založenou na čítači a porovnávacích registrech. Čítáním pulsů taktovací frekvence vzniká virtuální trojúhelníkový signál, jehož hodnota je neustále porovnávána s hodnotami zapsanými v porovnávacích registrech. Logické hodnoty výstupů z těchto porovnávacích jsou vyvedeny na piny čítače, které tak tvoří jednotlivé kanály pulsně šířkové modulace. Mikrokontrolér má také hardwarovou podporu pro správně načasované obnovování hodnot v porovnávacích registrech, vše zajišťuje automaticky.

Rozhodujícím parametrem pulsně šířkové modulace je frekvence spínání tranzistorů. Vzhledem k implementaci do mikrokontroléru jsme omezeni velikostí nejvyšší frekvence, kterou můžeme použít jako zdroj čítače, což je v našem případě frekvence jádra mikroprocesoru o hodnotě $f_{\mu P} = 32 \text{ MHz}$. Dále nás omezuje rozlišení výstupní hodnoty napětí, jaké potřebujeme dosáhnout. Musíme tedy zvolit kompromis, při kterém je dostačující frekvence i rozlišení. Začneme rozlišením, pro zajištění plynulé regulace je taktéž potřebná i plynule se měnící hodnota výstupního napětí. Dostatečné by mohlo být 10-bitové rozlišení, které nám na celém rozsahu $\langle -37 \text{ V}; 37 \text{ V} \rangle$ dává teoretickou možnost rozlišení výstupního napětí:

$$\Delta U = \frac{37 \text{ V} - (-37 \text{ V})}{2^{10}} \approx 0,07 \text{ V}.$$

Při této hodnotě rozlišení a použití symetrického trojúhelníkového signálu jsme schopni dosáhnout frekvence spínání tranzistorů:

$$f_s = \frac{32 \text{ MHz}}{2 \cdot 2^{10}} = 15625 \text{ Hz}.$$

Výsledná hodnota frekvence PWM bude $f_{PWM} = 2 \cdot 15625 \text{ Hz} = 31250 \text{ Hz}$, tato hodnota je dokonce nad slyšitelným spektrem, takže akustický hluk měniče je minimální. Velikost zvlnění proudu procházejícího motorem můžeme odhadnout z velikosti elektrické časové konstanty motoru vypočítané z parametrů vinutí motoru

$$\tau_e = \frac{L_a}{R_a} = \frac{6 \text{ mH}}{1,5 \Omega} = 4 \text{ ms}.$$

Protože vinutí motoru představuje setrvačný členek 1. řádu, můžeme vypočítat i šíři pásma propustnosti, resp. kritickou frekvenci f_c (při poklesu o 3 dB) za kterou má setrvačný členek útlum 20 dB/dek

$$f_c = \frac{1}{2 \cdot \pi \cdot \tau} = \frac{1}{2 \cdot \pi \cdot 4 \text{ ms}} \approx 39,8 \text{ Hz}.$$

Frekvence PWM je přibližně o tři řády větší než frekvence kritická. To znamená, že útlum proudových pulsací je přibližně 60 dB . Hodnota útlumu je vysoká a frekvence pulsně šířkové modulace tak postačuje.

Nastavení registrů mikrokontroléru pro zajištění výše popsané PWM je následující. Předně je nutné nastavit piny kanálů PWM jako výstupní. Dále už nastavujeme registry čítače, nejprve povolíme, aby hardware čítače převzal kontrolu nad výstupními piny. Dále nastavíme mód čítání na symetrický trojúhelníkový s obnovováním porovnávacích registrů při průchodu nulou. Zapišeme hodnotu rozlišení, resp. hodnotu horní úvrati čítače. Do porovnávacích registrů obou kanálů zapišeme shodnou inicializační hodnotu (aby na výstupu H-můstku bylo 0 V) a čítač spustíme.

Konkrétní nastavení registrů pro levý motor na pinech čítače TCC1:

```

1 // nastaveni TCC1 - PWM pro motor_1
2 PORTC.DIRSET = (1<<4) | (1<<5); // vystupni piny
3 // TCC1 ovlada vystupni piny a dual-slope mod
4 TCC1.CTRLB = TC1_CCBEN_bm | TC1_CCAEN_bm | TC_WGMODE_DS_T_gc;
5 TCC1.PER = 1024-1; // rozliseni 10 bitu (hodnota TOP)
6 TCC1.CCA = 511; // inicializace stridy
7 TCC1.CCB = 511;
8 TCC1.CTRLA = TC_CLKSEL_DIV1_gc; // spusteni citace

```

Pro ovládání motorů je dále nadefinována funkce `motor(voltage)` s parametrem `voltage`. Funkce ovládá střídu PWM, což odpovídá výstupnímu napětí H-můstku. Přípustná hodnota parametru vzhledem k rozlišení modulace je v rozmezí $\langle -511; 511 \rangle$, funkce je však proti hodnotám mimo definiční obor ošetřena – hodnotu vstupního parametru saturuje na -511 (příp. 511). Pro změnu hodnoty v porovnávacích registrech pulsně šířkové modulace používáme zásobníky, do kterých novou hodnotu uložíme, mikrokontrolér automaticky zajistí přepsání hodnoty ze zásobníku do porovnávacího registru v momentě, kdy čítač dosáhne hodnoty 0. Před zápisem nových hodnot do zásobníků zakážeme automatické přepisování hodnot do porovnávacích registrů, zabráníme tak použití ne zcela aktualizovaných (chybných) dat.

Zdrojový kód funkce `motor(voltage)` pro levý motor, makro `SATURATE` provede saturaci proměnné (první parametr) do zadaných intervalů (2. a 3. parametr):

```

1 void motor(s16 voltage)
2 {
3     SATURATE(voltage,-511,511); // saturace vstupni hodnoty
4     // zakaz aktualizace porovnavacich registru
5     TCC1.CTRLFSET = TC1_LUPD_bm;
6     // zapis nove hodnoty do zasobniku
7     TCC1.CCABUF = 511+voltage; // unipolarni PWM
8     TCC1.CCBBUF = 511-voltage;
9     // povoleni aktualizace porovnavacich registru

```

```
10 | TCC1.CTRLFCLR = TC1_LUPD_bm;  
11 | }
```

5.3. Komunikace s PC

Komunikace hardwaru trenažéru s počítačem, na kterém simulace běží probíhá pomocí sériového portu. Na dnešních komerčních PC se však konektor sériové linky (RS-232) víceméně nevyskytuje. Proto je použit integrovaný obvod FT232R (viz kapitola 4.2.1), který hardwarově emuluje USB. Výsledný efekt je, že komunikujeme po jednoduché sériové lince, ale fyzicky jsme připojeni přes USB.

5.3.1. USART

Modul USART mikrokontroléru ATxmega128A1 zajišťuje hardwarovou podporu pro komunikaci po sériové lince. Sériová komunikace může, jak už název Universal Synchronous/Asynchronous Receiver and Transmitter napovídá, probíhat ve více variantách. Využíváme variantu asynchronní, pro kterou je integrovaný obvod FT232R určen. [6]

Nastavení modulu provádíme opět pomocí registrů mikrokontroléru. Parametry asynchronní komunikace nastavujeme na hodnoty:

- rychlost komunikace 115200 *bit/s*
- osm datových bitů
- jeden stop bit
- žádná kontrola parity

Takovéto nastavení dovoluje odesílat/přijímat 11520 *Byte/s*. Postup nastavení mikrokontroléru je následný. Pin přijímací (tzv. RxD) nastavíme jako vstupní s interním pull-up rezistorem. Vysílací pin (tzv. TxD) nastavíme jako výstupní. Poté již následuje konfigurace samotného modulu USART. Nastavení rychlosti komunikace (tzv. Baud Rate) je složitější, více v [4]. Zde jen tolik, že pro nastavení požadované rychlosti musíme do registrů BAUDCTRLA a BAUDCTRLB uložit hodnoty vypočtené podle vzorců v [4]. Dále povolíme přerušení při příchodu znaku a nastavíme mu nízkou prioritu, komunikace je totiž ve srovnání s během ostatních částí programu pomalá a nezáleží tolik na přesném načasování zpracování. Poté už stačí nastavit jen počet datových bitů a povolit příjem i odesílání. Ostatní zdánlivě nenastavené parametry jsou výchozí konfigurací mikrokontroléru.

Zdrojový kód nastavující parametry modulu USART:

```

1 | PORTF.PIN6CTRL = PORT_OPC_PULLUP_gc; // pull-up na RXD pinu
2 | PORTF.DIRSET = 1<<3; // nastavime TXD pin jako vystupni
3 | // nastaveni Baud Rate na 115.2kbaud
4 | USARTF0.BAUDCTRLA = 131;
5 | USARTF0.BAUDCTRLB = (u8)((-3) << USART_BSCALE_gp);
6 | // generuj preruseni pri prijmu znaku
7 | USARTF0.CTRLA = USART_RXCINTLVL_LO_gc;
8 | USARTF0.CTRLB = USART_CHSIZE_8BIT_gc; // 8 datovych bitu
9 | // povolime prijem i vysilani
10 | USARTF0.CTRLB = USART_RXEN_bm | USART_TXEN_bm;

```

Poznámka: Stejně je nastaven i druhý USART použitý pro kontrolní výpisy, pouze negeneruje přerušování při příjmu znaku.

Odeslání zprávy obsluhuje funkce `putchar(c)`, parametr `c` je 8-bitová proměnná – tj. osm datových bitů poslaných v jedné zprávě. Odesílané datové bity se uloží do registru `DATA` a mikrokontrolér automaticky zajistí jejich odeslání. Zdrojový kód:

```

1 | void putchar(u8 c)
2 | {
3 |     // počkame na uvolneni registru DATA
4 |     while(!(USARTF0.STATUS & USART_DREIF_bm));
5 |     // zapiseme zpravu k odeslani
6 |     USARTF0.DATA = c;
7 | }

```

Pro jednodušší práci s odesílanými daty máme nadefinováno funkcí více. Například pro odesílání proměnné v zápisu šestnáctkové soustavy `puthex16(h)` a mnoho dalších. Všechny však využívají jako základ funkci `putchar(c)` a jsou k nahlédnutí v příloženém zdrojovém kódu firmwaru.

Přijaté zprávy jsou ukládány do hardwarových zásobníků, číst je můžeme opět z registru `DATA`. Moduly USART mají registr pro příchozí i odchozí data společný, liší se pouze funkcí při čtení nebo při zápisu z/do něj.

5.3.2. Komunikační protokol

Komunikace s nadřazeným PC probíhá podle domluveného protokolu formou otázky a odpovědi. Komunikaci vždy začíná nadřazené PC zasláním zprávy v určitém tvaru, mikrokontrolér odpovídá.

Zpráva přicházející z PC obsahuje aktuální požadované síly v pravé a levé řídicí šňůře. A zpráva posílaná z mikrokontroléru nazpátek obsahuje hodnoty ze sensorů, tj. velikost stažení řídicích šňůr a filtrované hodnoty zrychlení ze všech tří os akcelerometru.

Formát zpráv je v obou případech stejný. Zpráva začíná startovacím znakem 'x', za kterým následují přenášené hodnoty. Hodnoty jsou posílány ve formě znaků čísel šestnáctkové soustavy, tzn. že jedna 16-bitová proměnná odpovídá čtyřem přenášeným znakům. Jednotlivé hodnoty nejsou ve zprávě nijak odděleny, ani nemají žádné identifikátory. Komunikační protokol tedy přesně definuje pořadí jednotlivých hodnot ve zprávě i jejich velikost (v bytech).

Příklad zprávy přicházející z hlavního PC s označením jejího kódování,

$$\underbrace{x}_{start} \underbrace{01af}_{F_L} \underbrace{a5}_{F_P} \underbrace{b6}_{F_P} \underbrace{ab}_{status}$$

hodnoty F_L a F_P jsou velikosti požadovaných sil v levé a pravé řídicí šňůře – obě 16-bitové proměnné. Byte status je zatím nevyužívaný, nachystán je pro posílání příkazů typu „spust řízení motorů“ (při zahájení simulace) atp. Vícebytové hodnoty jsou posílány od vyššího bytu (MSB) k nižšímu bytu (LSB).

Příklad zprávy posílané jako odpověď z mikrokontroléru,

$$\underbrace{x}_{start} \underbrace{26d1}_{l_L} \underbrace{a5b6}_{l_P} \underbrace{a35c}_{x_{acc}} \underbrace{9e0d}_{y_{acc}} \underbrace{dd91}_{z_{acc}}$$

l_L a l_P jsou hodnoty stažení levé a pravé řídicí šňůry a x_{acc} , y_{acc} a z_{acc} jsou hodnoty zrychlení v osách akcelerometru – všechny 16-bitové proměnné.

Zpracování příchozí zprávy provádí funkce přerušení při příjmu znaku z PC. Po přijetí startovacího znaku 'x' začne funkce ukládat příchozí znaky do softwarového zásobníku. Před vložením nově příchozího znaku do zásobníku ověří, zda je z množiny číslic šestnáctkové soustavy, tj. '0' ÷ '9' a 'a' ÷ 'f'². Pokud není, zahazuje celý zásobník a čeká na nové přijetí startovacího znaku. Pokud vše proběhne v pořádku, tedy přijme 10 správných znaků, rozloží a převede celou zprávu na jednotlivé hodnoty a zapíše příznak pro odeslání odpovědi. Zdrojový kód zpracování příjmu je poměrně dlouhý, zájemce může nahlédnout do přiloženého zdrojového kódu firmwaru.

Pokud hlavní program zjistí, že je nastaven příznak pro odeslání odpovědi, neprodleně tak učiní. Ze sensorů čteme data s frekvencí 5 kHz, takže odesílané hodnoty jsou vždy aktuální. Pro odeslání 16-bitové proměnné využíváme funkci `puthex16(h)` (viz podkapitola 5.3.1), která se již sama postará o převod čísla na znaky čísel šestnáctkové soustavy.

Komunikace může pomocí tohoto protokolu a za stávajícího nastavení sériové linky, běžet s frekvencí přibližně 500 cyklů³ za sekundu. Skutečná rychlost se však může měnit

²Všecké znaky (vyjma číslic) musí být minuskule, neboli malá písmena.

³Jeden cyklus je příjem zprávy z PC a odeslání odpovědi.

a závisí na řídicím počítači, s jakou frekvencí si data bude žádat. Protože převádíme binární data na znaky čísel šestnáctkové soustavy⁴, musíme odesílat dvakrát více bytů než je obsaženo v přenášených proměnných. Avšak rychlost komunikace je i tak dostatečná. Navíc získáme jistou dávku ochrany před chybami při přenosu dat, a také jednoduchost a variabilitu posílaných zpráv.

⁴Jednomu bytu odpovídají dvě číslice šestnáctkové soustavy a jeden odesílaný znak je byte.

6. Závěr

Úkol navrhnout treňažer přistání na padákovém kluzáku byl splněn. Navržená konstrukce je jednoduchá, rozebíratelná a umožňuje nastavení výšky rozpěry¹. Zavěšení sedačky na konstrukci je stejné jako na reálném padáku. Umožňuje nácvik řízení padáku pomocí všech základních ovládacích prvků. Což je kromě řídicích šňůr a speed systému také řízení pomocí náklonu v postroji. Nepříjemnou vlastností je „houpání“ konstrukce při pohybu pilota v postroji, není však nijak markantní.

Elektronika simulátoru zahrnuje výkonovou část, která zajišťuje napájení motorů pomocí H-můstků i řídicí část postavenou na mikrokontroléru ATxmega128A1. Kromě obvodů jsou navrženy i veškeré DPS. Jejich funkčnost je odzkoušena na prototypu. Elektronika je dostatečně nadimenzována, aby se dala použít i s výkonnějšími motory. DPS pro finální výrobek by se měly kvůli vyšší spolehlivosti nechat vyrobit průmyslově.

Program pro řídicí mikrokontrolér je psán v jazyce C a obsahuje inicializace používaných modulů a periférií a základní obslužné funkce pro práci s nimi. Navržena je i jednoduchá regulace síly v řídicích šňůrách. Stávající rozvržení umožňuje snadné zkoušení a ladění aplikace.

Ovládací prvky navrženého zařízení jsou velice podobné s těmi reálného padáku. Elektronické řízení tahu v řídicích šňůrách zajišťuje vysokou variabilitu, tudíž můžeme jednoduše simulovat řízení různých typů padáků a nebo různé letové situace (nejen přistávání). Treňažer tak může sloužit i pro první přiblížení při nácviku nebezpečných letových režimů.

Další postup bude testování vyrobeného prototypu s instruktorem padákového létání a co nejlepší naladění řízení síly tahu. Pomocí senzorů treňažeru můžeme také lépe odhadnout síly a rychlosti pohybu řídicích šňůr reálného padákového kluzáku. A konečně na základě všech získaných dat a zkušeností z prototypu vyrobit finální zařízení.

¹Nastavení výšky rozpěry určuje (jak je popsáno v kapitole 3.1) „živost“ řízení padáku náklonem.

Seznam použité literatury

- [1] Allegro MicroSystems, Inc. *ACS712 : Fully Integrated, Hall Effect-Based Linear Current Sensor IC with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor* [online]. Rev. 13. [s.l.] : [s.n.], [200?] [cit. 2011-05-16]. Dostupné z WWW: <http://www.allegromicro.com/en/Products/Part_Numbers/0712/index.asp>.
- [2] Analog Devices, Inc. *ADXL327 : Small, Low Power, 3-Axis ± 2 g Accelerometer* [online]. Rev 0. [s.l.] : [s.n.], 08/2009 [cit. 2011-05-14]. Dostupné z WWW: <<http://www.analog.com/en/mems-sensors/inertial-sensors/adxl327/products/product.html>>.
- [3] Atmel Corporation. *XMEGA A1 Microcontroller : ATx-mega64A1/128A1/192A1/256A1/384A1 Preliminary* [online]. revision M. [s.l.] : [s.n.], 2008, updated 9/10 [cit. 2011-05-14]. Dostupné z WWW: <http://www.atmel.com/dyn/products/product_docs.asp?category_id=163&family_id=607&subfamily_id=1965&part_id=4298>.
- [4] Atmel Corporation. *XMEGA A Microcontroller : XMEGA A Manual Preliminary* [online]. revision H. [s.l.] : [s.n.], 2008, updated 12/09 [cit. 2011-05-14]. Dostupné z WWW: <http://www.atmel.com/dyn/products/product_docs.asp?category_id=163&family_id=607&subfamily_id=1965&part_id=4298>.
- [5] El Speedo s.r.o. *El Speedo* [online]. c2002 [cit. 2011-04-28]. Dostupné z WWW: <<http://www.elspeedo.cz/>>.
- [6] Future Technology Devices International Ltd. *FT232R : USB UART IC* [online]. Version 2.09. [s.l.] : [s.n.], August 2005, April 2011 [cit. 2011-05-16]. Dostupné z WWW: <<http://www.ftdichip.com/Products/ICs/FT232R.htm>>.
- [7] KLÍMA, Bohumil. *Mikroprocesorové řízení elektrických pohonů* Brno : skripta VUT, 2007. 43 s.
- [8] NOVA Vertriebsges.m.b.H. *Nova International* [online]. [cit. 2011-04-28]. Dostupné z WWW: <<http://nova-wings.com/>>.
- [9] PLOS, Richard, a kol. *Paragliding*. Cheb: Svět křídel, 2010. 232s. ISBN 978-80-86808-78-9.
- [10] SKALICKÝ, Jiří. *Elektrické regulované pohony*. Brno : skripta VUT, 2007. 123 s.
- [11] Texas Instruments Incorporated. *DRV8402 : Dual Full Bridge PWM Motor Driver* [online]. [s.l.] : [s.n.], Feb 2009 [cit. 2011-05-10]. Dostupné z WWW: <<http://www.ti.com/lit/gpn/drv8402>>.

- [12] Texas Instruments Incorporated. *INA125 : Instrumentation Amplifier* [online]. [s.l.] : [s.n.], 1997, 27 Sep 2000 [cit. 2011-05-14]. Dostupné z WWW: <<http://focus.ti.com/docs/prod/folders/print/ina125.html>>.
- [13] *World XContest 2011 : XContest — biggest worldwide online paragliding & hang-gliding contest* [online]. c2006 [cit. 2011-04-27]. Dostupné z WWW: <<http://www.xcontest.org/>>.
- [14] Zemic Europe B.V. *L6D load cell* [online]. Rev6. [s.l.] : [s.n.], 2011 [cit. 2011-05-14]. Dostupné z WWW: <<http://www.zemic.nl/Products-Single-Point-load-cells/product-l6d.html>>.

Seznam použitých zkratek a symbolů

A/D	analogově digitální (převodník)
DC	z angličtiny Direct Current (stejnoseměrný proud)
DPS	deska plošných spojů
LSB	z angličtiny Least Significant Byte (nejnižší byte)
MEMS	z angličtiny Micro-Electro-Mechanical Systems (mikroelektromechanické systémy)
MSB	z angličtiny Most Significant Byte (nejvyšší byte)
P	proporční (regulátor)
PC	z angličtiny Personal Computer (osobní počítač)
PD	proporčně-derivační (regulátor)
PI	proporčně-integrační (regulátor)
PWM	z angličtiny Pulse Width Modulation (pulsně šířková modulace)
USART	z angličtiny Universal Synchronous / Asynchronous Receiver and Transmitter (Synchronní / asynchronní sériové rozhraní)
USB	z angličtiny Universal Serial Bus (univerzální sériová sběrnice)

τ [s]	časová konstanta filtru typu dolní propust
τ_e [s]	elektrická časová konstanta motoru
ω [rad/s]	úhlová rychlost motoru
$c\phi$ [V.s/rad]	napěťová a momentová konstanta DC motoru s permanentními magnety
C_f [1]	filtrační konstanta digitálního filtru typu dolní propust
d_b [m]	průměr navíjecího bubnu
$f_{\mu P}$ [Hz]	taktovací frekvence mikroprocesoru
f_c [Hz]	kritická frekvence filtru typu dolní propust
F_L, F_P [N]	tahová síla v řídicí šňůře levé a pravé
F_{max} [N]	maximální tahová síla v řídicí šňůře
f_s [Hz]	frekvence spínání tranzistoru H-můstku
f_{PWM} [Hz]	frekvence PWM pro motory
i [A]	proud motorem
I_n [A]	nominální proud motorem
K_P [1]	konstanta (zesílení) P regulátoru
L_a [H]	indukčnost vinutí motoru
l_L, l_P [m]	délka odvinuté řídicí šňůry levé a pravé
M [Nm]	moment na hřídeli motoru
M_m [Nm]	mechanický moment motoru
M_n [Nm]	nominální moment motoru
M_z [Nm]	záběrný moment motoru
n [ot/min]	otáčky motoru
n_n [ot/min]	nominální otáčky motoru
P_1 [W]	příkon motoru
P_{mech} [W]	mechanický výkon motoru

R_a [Ω]	odpor vinutí motoru
r_b [m]	poloměr navíjecího bubnu
t [s]	čas
T_f [s]	délka kroku digitálního filtru
U [V]	napětí
u_i [V]	indukované napětí motoru
U_n [V]	nominální napětí motoru
U_Z [V]	průrazné napětí Zenerovy diody
v_{max} [m/s]	maximální rychlost pohybu řídicí šňůry
$x_{acc}, y_{acc}, z_{acc}$ [m/s^2]	změřené hodnoty z jednotlivých os akcelerometru
x_k	aktuálně naměřená hodnota k filtraci
y_{k+1}	výstupní hodnota z digitálního filtru
y_k	hodnota z předchozího kroku filtrace digitálního filtru

Seznam příloh

K diplomové práci je přiložen kompaktní disk s následujícími dokumenty v elektronické podobě.

El. schemata a DPS

Veškerá elektronická schemata zapojení a navržené DPS, ve formátu projektů programu EAGLE.

Firmware

Ve formátu projektu vývojového prostředí IAR Embedded Workbench. Obsahuje zdrojový kód v jazyce C i zkompilovaný program.

Diplomová práce

Elektronická podoba práce ve formátu pdf a její zdrojový kód v \LaTeX .