

České vysoké učení technické v Praze

Fakulta elektrotechnická



České vysoké učení technické v Praze  
Fakulta elektrotechnická

Diplomová práce

# Autopilot pro UAV letoun

*Bc. Petr Pahorecký*

Vedoucí práce: doc. Ing. Martin Hromčík, Ph.D.

Studijní program: Kybernetika a robotika

Obor: Letecké a kosmické systémy

2015



# ZADÁNÍ DIPLOMOVÉ PRÁCE

Student: **Bc. Petr Pahorecký**

Studijní program: **Kybernetika a robotika**  
Obor: **Letecké a kosmické systémy**

Název tématu česky: **Autopilot pro UAV letoun**

Název tématu anglicky: **Autopilot for UAV Aircraft**

## Pokyny pro vypracování:

1. Seznamte se s autopilotem vyvíjeným v rámci diplomové práce Bc. Jaroslava Halgašika na katedře řídicí techniky.
2. Seznamte se s řídicími algoritmy pro autopilota vyvíjeného v rámci diplomové práce Bc. Josefa Nováka na katedře řídicí techniky.
3. Implementujte vybrané řídicí algoritmy do HW autopilota.
4. Proveďte letové testování navrženého autopilota na UAV letounu vyvíjeného ve spolupráci katedry řídicí techniky s VZLÚ Praha.

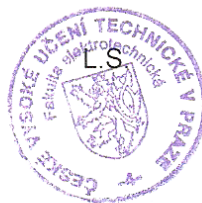
## Seznam odborné literatury:

- [1] Stevens, Lewis: Aircraft control and simulation, Wiley, 2001
- [2] Pech, Z., Věk, V.: Systémy řízení letu. Skripta ČVUT, Praha 1997.

Vedoucí diplomové práce: Ing. Martin Hromčík, Ph.D. (K13135)

Datum zadání diplomové práce: 7. ledna 2014

Platnost zadání do<sup>1</sup>: 31. srpna 2015



V Praze dne 7. 1. 2014

---

<sup>1</sup> Platnost zadání je omezena na dobu tří následujících semestrů.

### **Prohlášení autora práce**

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne .....

.....

Podpis autora práce

## **Poděkování**

Chtěl bych poděkovat vedoucímu práce, oponentovi i celému týmu za podporu celého projektu. Dále bych chtěl poděkovat rodině a přítelkyni za psychickou podporu během studia.

## **Anotace**

Tato práce pojednává o vytvoření autopilota do bezpilotního letounu. Jedná se o zkompletování tří diplomových prací zabývajících se výrobou letounu, jeho modelováním, návrhem a výrobou elektroniky a softwaru. Dále pojednává o řízení a následné naimplementování několika řídicích smyček. Nejnižší smyčka je postavena na LQ regulaci, vyšší smyčky používají maximálně PID regulátory. V práci je také obsažen výběr vhodných elektronických částí letounu, jejich zapojení a následné odzkoušení.

## **Anotation**

This document is about a creation of autopilot for unmanned aircraft vehicle. It join three diploma thesis, the mechanical creation of aircraft, modeling and simulink the aircraft and also design an production of PCB and software. It also say about controlling law and implementig several control loop. The lower loop used LQR – linear quadratic regulator. Another higher loop used maximal PID regulator. In thesis is also included the selection of suitable eletronic parts of the airfraft, their installation and subsequently tested.

# Obsah

Seznam obrázků .....	11
Seznam tabulek .....	13
1 Úvod.....	14
2 Popis letounu [3] .....	16
3 Model letounu a řízení.....	18
3.1 Model letounu .....	18
3.2 Řízení [1].....	19
3.2.1 Podélný pohyb .....	19
3.2.2 Stranový pohyb .....	21
4 Deska Autopilota [2] .....	24
4.1 Inerciální měřicí jednotka .....	24
4.2 Komunikační modul .....	25
4.3 Senzory .....	25
4.3.1 GPS .....	25
4.3.2 Analogové senzory .....	25
4.4 Připojení k debuggeru .....	26
4.5 Software .....	26
5 Výběr elektroniky do letounu.....	27
5.1 Motor .....	27
5.1.1 Požadavky .....	27
5.1.2 Výběr .....	29
5.2 Baterie .....	31
5.2.1 Hmotnost .....	31
5.2.2 Napětí .....	35
5.2.3 Proud .....	35
5.2.4 Kapacita .....	35



5.2.5	Závěr .....	35
5.3	Regulátor .....	37
5.3.1	Telemetrie.....	38
5.4	Senzory tlaku .....	38
5.4.1	Měření nízkých rychlostí.....	38
5.4.2	Požadavky .....	40
5.4.3	Diferenční senzor tlaku.....	41
5.5	Servomotory .....	44
6	Implementace regulátoru motoru a motoru. ....	46
6.1	Napájení .....	46
6.2	Řízení.....	46
6.3	Telemetrie .....	48
6.3.1	Topologie .....	48
6.3.2	Fyzická a linková vrstva.....	48
6.3.3	Protokol "Prostý text" .....	49
6.3.4	Implementace .....	49
6.4	Připojení motoru.....	50
6.5	Testování motoru s regulací .....	51
6.6	Shrnutí.....	54
7	Realizace dynamických senzorů tlaku.....	55
7.1	Zapojení.....	55
7.2	Implementace 1/2.....	58
7.3	Kalibrace .....	59
7.4	Implementace 2/2.....	64
8	Regulace.....	66
8.1	Módy .....	66

8.2	Tlumiče kmitů .....	67
8.3	Proporcionální regulace .....	67
8.4	Proporcionálně-integrační regulace.....	67
8.5	LQ regulace .....	67
8.5.1	Podélný pohyb .....	67
8.5.2	Stranový pohyb .....	69
8.6	GPS regulace .....	69
8.7	Kalibrační funkce.....	70
8.7.1	LQR podélné.....	70
8.7.2	LQR stranové .....	70
8.7.3	Tah motoru .....	70
8.7.4	Úhel stoupání .....	70
9	Propojení s řídicí jednotky autopilota s Matlabem .....	71
9.1	Komunikace Matlab – řídicí jednotka.....	71
9.2	Hardware in the loop simulation[29, 30] .....	72
9.2.1	Testování regulace na úhel stoupání.....	72
9.2.2	Testování regulace na úhel náklonu.....	74
9.2.3	Testování regulace na kurzový úhel .....	75
9.2.4	Testování podélné LQ regulace .....	77
9.2.5	Testování stranové LQ regulace .....	78
10	Oživení a ověření řízení na letounu .....	80
11	Závěr .....	81
12	Přílohy .....	83
12.1	Vybrané funkce .....	83
12.2	CD.....	88
13	Citace.....	89

## Seznam obrázků

Obrázek 1 - Návrh letounu.....	14
Obrázek 2 - Deska autopilota.....	15
Obrázek 3 - 3D model letounu.....	16
Obrázek 4 - Simulinkový model letounu.....	18
Obrázek 5 - Coef to 6DOF.....	19
Obrázek 6 - Podélné řízení LQR.....	20
Obrázek 7 - Regulace výšky.....	21
Obrázek 8 - Stranové řízení LQR.....	22
Obrázek 9 - Stranové řízení na kurs.....	23
Obrázek 10 - Deska plošných spojů autopilota.....	24
Obrázek 11 - Dvojlístá vrtule 11x7' [25].....	27
Obrázek 12 - Graf potřebného výkonu pro ustálený let v závislosti na rychlosti.....	28
Obrázek 13 - Rychlost stoupání v závislosti na rychlosti pro různé stoupání vrtulí.....	29
Obrázek 14 - Charakteristika motoru.....	30
Obrázek 15 - Graf bezrozměrného letového času v závislosti na hmotnostním poměru [14].....	33
Obrázek 16 - Graf popisující relativní letový čas, relativní hmotnost letounu a rozdíl.....	34
Obrázek 17 - Regulátor Jeti Spin Pro 66 [24].....	37
Obrázek 18 - Pitotova trubice [16].....	38
Obrázek 19 - Venturiho trubice [16].....	39
Obrázek 20 – Modelářská Pitotova trubice [26].....	40
Obrázek 21 - Stabilizátor napětí.....	41
Obrázek 22 - Senzor tlaku.....	42
Obrázek 23 - Závislost napětí na tlaku.....	43
Obrázek 24 - Úhel natočení podle délky pulsu [17].....	45
Obrázek 25 – Konektor [27]    Obrázek 26 – Svorkovnice [28].....	46
Obrázek 27 - Zobrazení pulsů [18].....	47
Obrázek 28 - Fyzické zapojení telemetrické linky.....	48
Obrázek 29 – Realizace zapojení motoru, regulátoru a baterií.....	50
Obrázek 30 - Schéma měřícího zařízení [3].....	52
Obrázek 31 - Závislost tahu na výkonu.....	53
Obrázek 32 - Prototyp diferenčního měření tlaku.....	55

Obrázek 33 - Návrh plošných spojů.....	56
Obrázek 34 - 3D model desky pro tlakové senzory .....	56
Obrázek 35 - Realizace desky pro tlakové senzory.....	57
Obrázek 36 - Realizace desky pro tlakové senzory 2.....	57
Obrázek 37 - Schéma měření Pitotovy trubice .....	59
Obrázek 38 - Porovnání rychlostí .....	61
Obrázek 39 - Odchylky rychlostí.....	62
Obrázek 40 - Odchylky rychlostí před a po korekci .....	63
Obrázek 41 - Porovnání LP s plovoucím průměrem a Kalmanovým filtrem .....	64
Obrázek 42 - Detail porovnání .....	65
Obrázek 43 - Porovnání regulace na úhel stoupání .....	73
Obrázek 44 - Porovnání výchylek výškovky při regulaci na úhel stoupání .....	73
Obrázek 45 - Porovnání regulace na úhel náklonu .....	74
Obrázek 46 - Porovnání výchylek křidélek při regulace na úhel stoupání.....	75
Obrázek 47 - Porovnání kurzové regulace .....	76
Obrázek 48 - Porovnání kurzové regulace - úhel náklonu .....	76
Obrázek 49 - Porovnání reakce křidélek na změnu kurzového úhlu .....	77
Obrázek 50 - Porovnání reakce křidélek při podélné LQ regulace .....	77
Obrázek 51 - Porovnání regulace na úhel náklonu .....	78
Obrázek 52 - Porovnání reakce křidélek.....	79
Obrázek 53 - Porovnání odchylek kormidla .....	79
Obrázek 54 - Schéma zapojení.....	80
Obrázek 55 - Testování výškovky .....	80

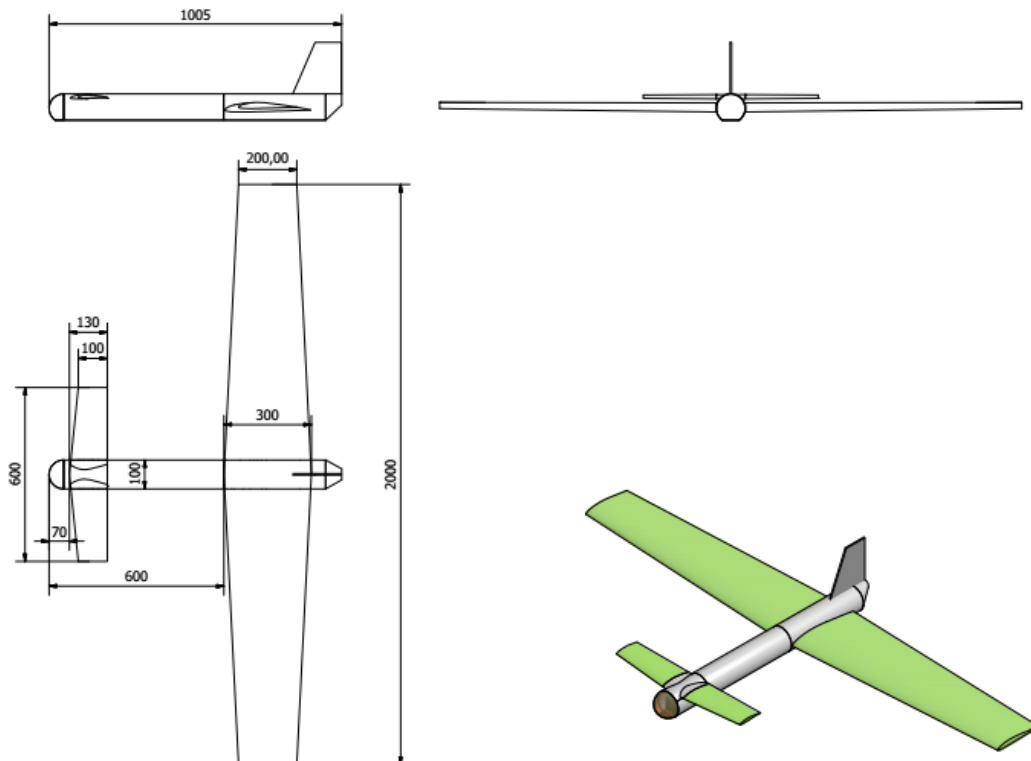
## Seznam tabulek

Tabulka 1 - Porovnání baterií.....	36
Tabulka 2 - Porovnání servomotorů.....	44
Tabulka 3 - Tah motoru v závislosti na nastaveném výkonu .....	53
Tabulka 4 - Naměřené hodnoty, Pitotova trubice.....	60
Tabulka 5 - Spočtené hodnoty, Pitotova trubice.....	61
Tabulka 6 - Nastavení sběrnice, Matlab - řídicí jednotka .....	71

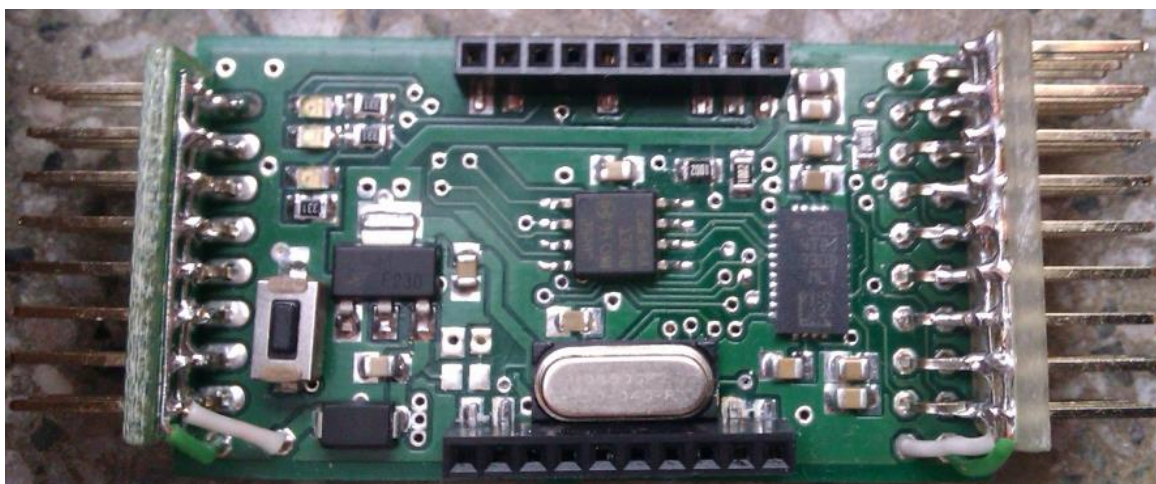
# 1 Úvod

Cílem mé diplomové práce bylo, na základě modelu letadla navrženého kolegou Josefem Novákem [1], implementovat řídicí algoritmy do jednotky autopilota vyvíjeného v rámci diplomové práce kolegy Jaroslava Halgašíka [2] a otestovat toto řízení na reálném letadle vyvíjené v rámci diplomové práce Petrem Adámkem [3].

V době kdy jsem se přidal k tomuto projektu, bylo letadlo pouze na papíře (Obrázek 1 - Návrh letounu) a hardware autopilota (Obrázek 2 - Deska autopilota) byl ve fázi osazování, testování a model letadla prozatím neexistoval.



Obrázek 1 - Návrh letounu



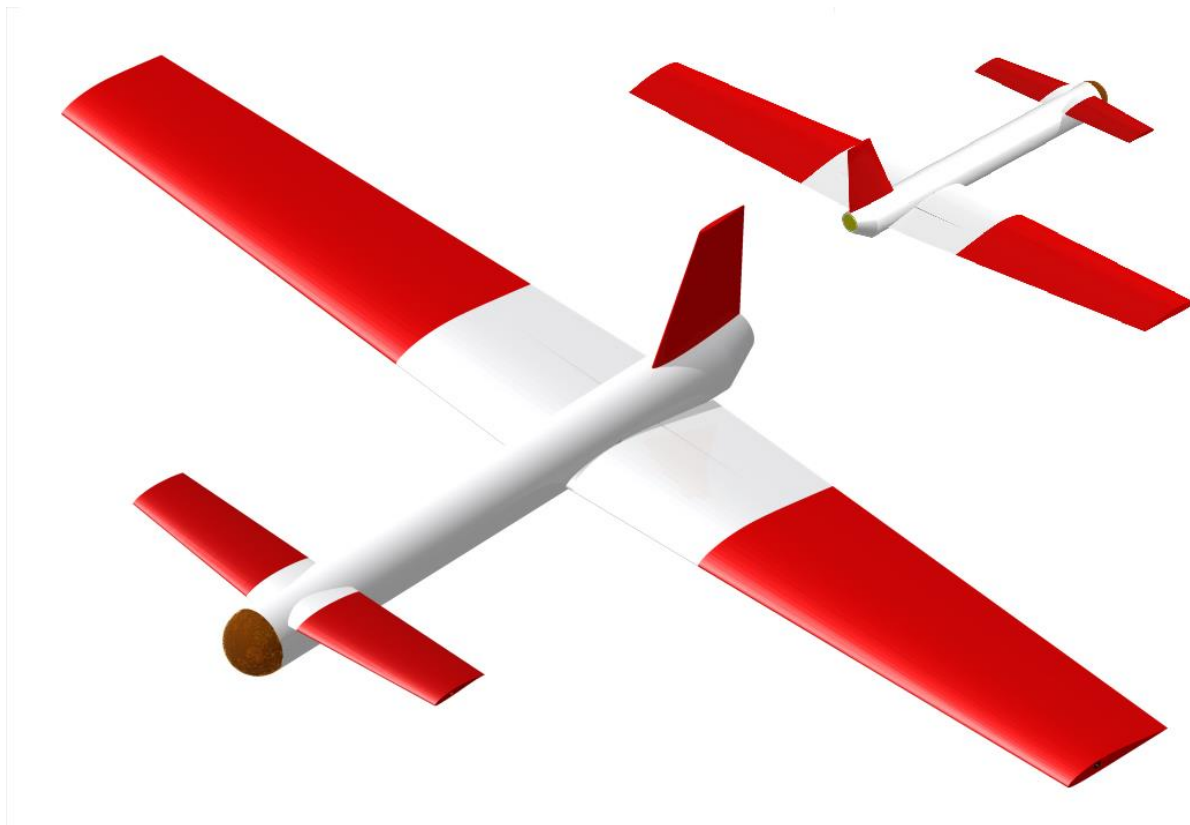
Obrázek 2 - Deska autopilota

Během práce na mě a kolegu Nováka také přibyly další “povinnosti” týkající se kompletního vybavení elektroniky pro tento letoun.

## 2 Popis letounu [3]

Letoun byl navržen na základě požadavků zadavatele, kde bylo stanoveno maximální rozpětí 2 m, návrhová hmotnost 4 kg, možnost umístění kamery, vytrvalost alespoň 30 min a dalším zásadním bodem zadání byla nekonvenční konfigurace.

Z důvodu požadavku na nekonvenční konfiguraci dostal návrh letadla kachní koncepci. Toto řešení přináší tu výhodu, že vztlak na kachní ploše působí na rozdíl od klasických ocasních ploch směrem nahoru. Tím pádem neubírá celkovému vztlaku na křídle. Zároveň je kachní konfigurace odolnější vůči pádu při malé rychlosti, neboť při přetažení dojde ke ztrátě vztlaku nejdříve na kachní ploše, to vede k automatickému potlačení a snížení úhlu náběhu. Nevýhodou však je ovlivnění proudění na křídle a menší prostor pro polohu těžiště než u klasické koncepce. V případě UAV to však problém není, protože payload<sup>1</sup> se v provozu bude měnit jen minimálně.



Obrázek 3 - 3D model letounu

---

<sup>1</sup> Nosná kapacita letadla neboli užitečné zatížení letadla



V přídi instalovaná kamera, resp. příprava pro její instalaci předurčuje tlačné uspořádání pohonné jednotky. Pohonná jednotka byla zvolena elektrická z důvodu absence jakýchkoliv provozních kapalin, není tedy nutné počítat s proměnou vahou letounu v závislosti na spotřebě paliva a také odpadá mazání pohyblivých částí motoru, tak s ohledem na instalaci kamery, protože elektromotor netrpí vibracemi tolik jako spalovací motor.

Jelikož má být letoun optimalizován na vytrvalost, byla pro návrh stanovena nízká cestovní rychlost 15 m/s a minimální rychlost letu 10 m/s.

Velké úsilí bylo vynaloženo na hledání optimálního profilu křídla, profilu a velikosti kachní plochy, tedy takový profil, který má největší součinitel vztlaku, ale zároveň nízký součinitel klopného momentu pro co nejmenší velikost kachní plochy. Byly porovnány různé profily křídla i kachní plochy spolu s velikostí a vybraná konfigurace je ta, která má nejmenší výchylku kachní plochy pro ustálený let.

Vrtule je zvolena tlačná a umístěna v horní části trupu. Tlačná je z důvodu možného umístění kamery v přídi. V horní části je umístěna, aby co nejméně zasahovala pod trup, což ovlivňuje výšku podvozku.

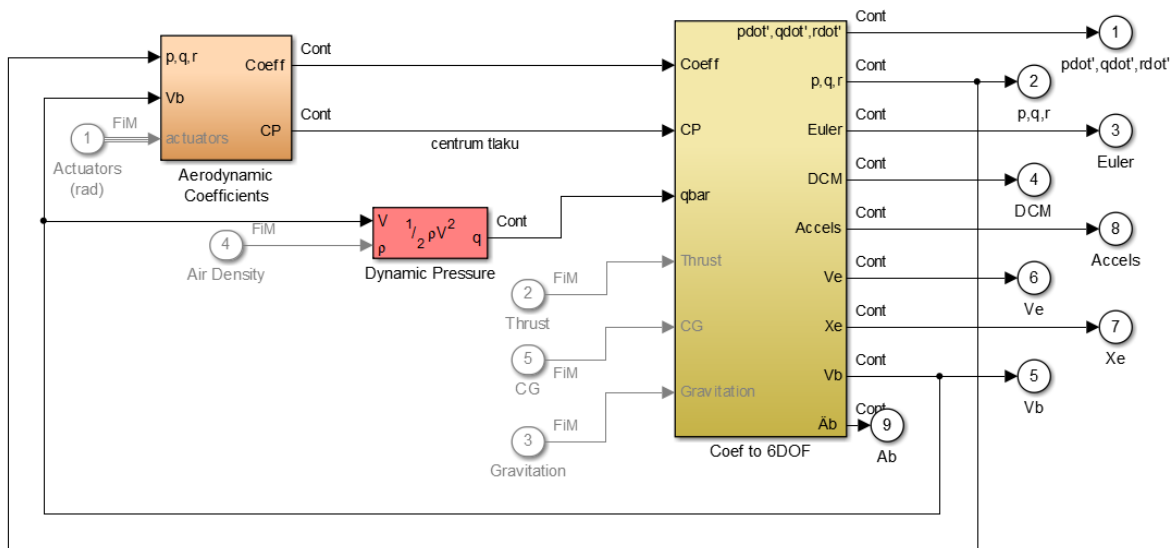
### 3 Model letounu a řízení

#### 3.1 Model letounu

Matematický model byl vytvořen v programu Matlab & Simulink, za použití Aerospace toolboxu [4].

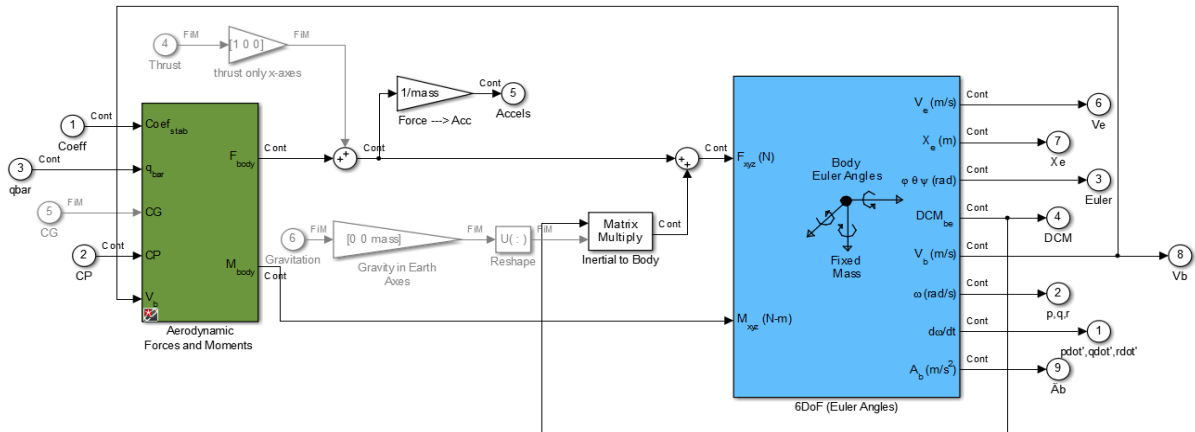
Obrázek 4 - Simulinkový model letounu - ukazuje vytvořený model letounu. Z důvodu, že se letoun bude pohybovat ve výšce v řádu několika stovek metrů, je hodnota hustoty vzduchu (Air Density) zvolena konstantní. Ze stejných důvodů je rovněž za konstantu zvolen vstup Gravitation neboli gravitační konstanta. Vstup CG je posun soustavy od těžiště, který je v našem případě [0 0 0], tedy počátek souřadné soustavy je umístěn v těžišti.

Blok "Aerodynamic Coefficients" počítá velikost koeficientů sil a momentů ve všech třech osách, v závislosti na výchylkách řídicích ploch (výškovka, křídélka, kormidlo) a také hodnotu centra tlaku, která je opět konstantní.



Obrázek 4 - Simulinkový model letounu

Blok “Coef to 6DOF” počítá pohyb a dynamiku letadla na základě spočítaných koeficientů.



Obrázek 5 - Coef to 6DOF

Blok “Aerodynamic Forces and Moments” je blok počítající síly a momenty letadla, které po připočtení tahu a gravitace se dále přepočtou pomocí bloku “6DoF (Euler Angles)” na polohu a rychlosti.

Oba zmíněné bloky jsou z knihovny Aerospace toolboxu [4].

## 3.2 Řízení [1]

V návrhu bylo uvažováno s LQ regulátorem, který reguluje stavové veličiny směrem k nule s co nejmenším akčním zásahem a rozkmitem těchto veličin. Model letadla byl rozšířen o dynamiku motoru a servomotorů.

### 3.2.1 Podélný pohyb

Úkolem řízení je navrhnout regulaci pro řízení výšky letu společně s rychlostí letu. Letoun sám o sobě je podélně stabilní, avšak obsahuje oscilující módy (phugoid mód<sup>2</sup>, short period mód<sup>3</sup> [22, 23]), které je cílem řízení co nejdříve utlumit.

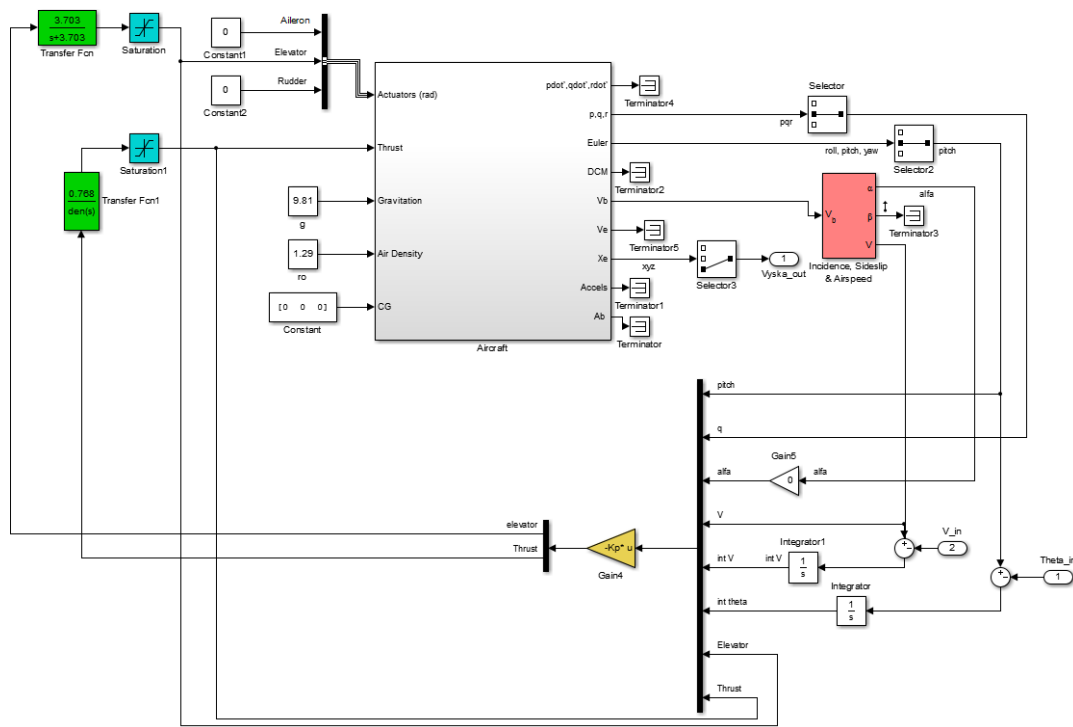
<sup>2</sup> Phugoidní mod - velmi pomalý 20-60s a nastává za změny vzdušné rychlosti, úhlu stoupání a výšky, avšak za konstantního úhlu náběhu. Při tomto modu nastává výměna kinetické a potenciální energie.

<sup>3</sup> Short period mod - jedná se o velmi rychlé oscilace v rámci několika sekund, kde letadlo kmitá okolo těžiště, mění se pouze úhel náběhu.

### 3.2.1.1 LQR

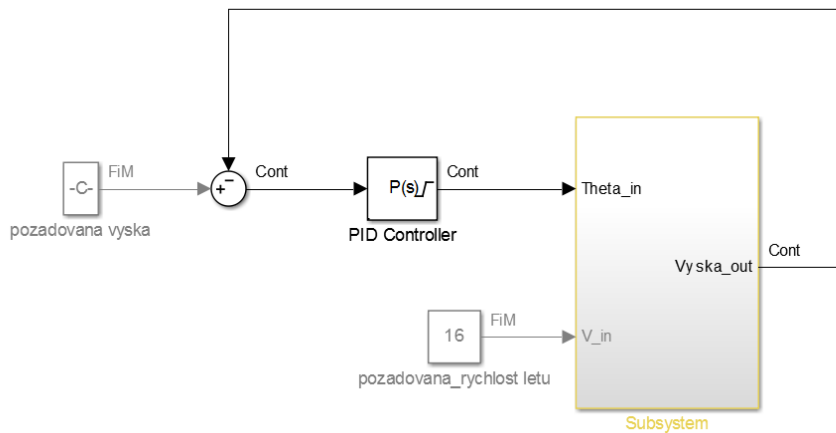
V regulaci pomocí LQR se jako reference bere požadovaný úhel stoupání a rychlost. Výstup z regulátoru je tah motoru a hodnota výchylky výškovky. Jak je ukazuje

Obrázek 6 - Podélné řízení LQR, kde do regulátoru vstupují veškeré stavy, které podélný pohyb ovlivňují, pitch (úhel stoupání letadla), q (úhlová rychlost stoupání letadla), alfa (úhel náběhu proudu vzduchu), V (vzdušná rychlost), int V (integrovaná vzdušná rychlost, od které je odečtena referenční hodnota rychlosti V\_in), int theta (integrovaný úhel náběhu, od kterého je odečtena reference theta\_in), výstupy regulátoru elevator (výškovka) a thrust (tah).



Obrázek 6 - Podélné řízení LQR

Další regulační smyčka je regulace výšky, jako vstup do regulátoru je rozdíl aktuální výšky od požadované výšky. Regulátor je typu P, tedy pouze násobení konstantou.



Obrázek 7 - Regulace výšky

### 3.2.2 Stranový pohyb

Úkolem stranového řízení je regulovat na požadovaný kurs a potlačovat vzruchy, které mohou vznikat okolním prostředím, jako náhlý porыв větru či změna hustoty vzduchu. Tyto vzruchy mohou vybudit stranové charakteristické módy nestabilní Spiral mod<sup>4</sup>, poté již stabilní Roll mod<sup>5</sup> a oscilující Dutch roll mod<sup>6</sup> [22, 23].

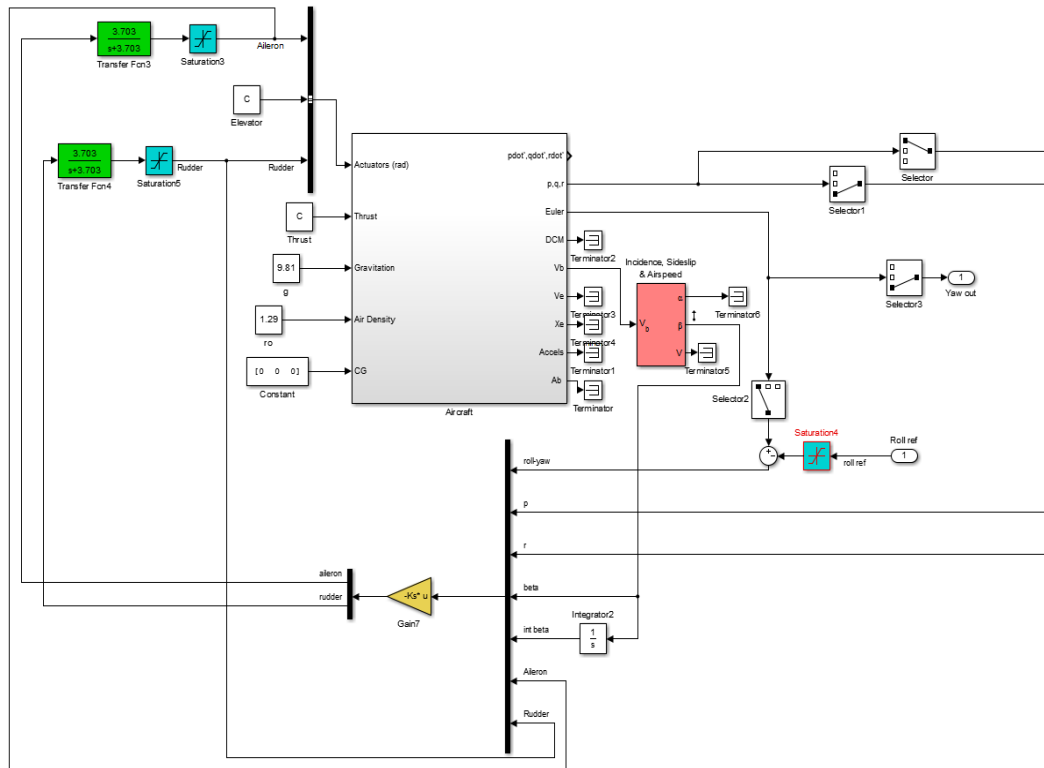
Navíc bylo zadavatelem uvedeno, aby byl proveditelný let po kružnici, tedy pod konstantním úhlem náklonu. Pro regulaci byla opět využita metoda pomocí LQ regulátoru.

Zapojení regulátoru je vidět na modelu (Obrázek 8 - Stranové řízení LQR), kde výstupy z regulátoru jsou výchylky křidélek a výchylka kormidla. Vstupy do regulátoru jsou aktuální stavy letadla, roll (úhel klopení), p (rychlost klopení), q (rychlost vybočení), beta (úhel vybočení), int beta (integrovaný úhel vybočení) a oba výstupy.

<sup>4</sup> Spiral mod – velmi pomalý nestabilní mód

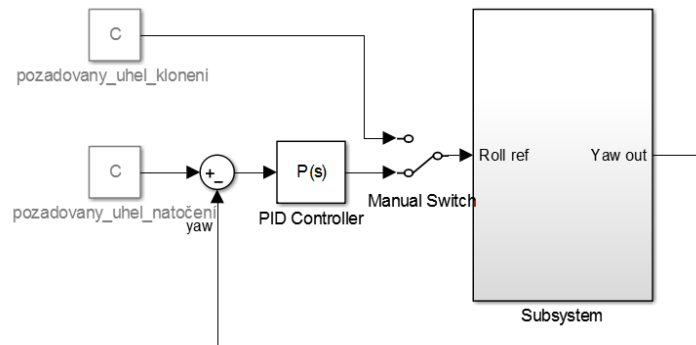
<sup>5</sup> Roll mod – exponenciálně tlumený klonivý mód

<sup>6</sup> Dutch roll mod- jedná se o kmitavý pohyb s poměrně malým tlumením.



**Obrázek 8 - Stranové řízení LQR**

Regulátor je řízen pomocí úhlu klopení. Zde je vytvořeno dvojí řešení, jednak pouze odečtením požadované výchylky, v tom případě letadlo letí po kružnici. Nebo provedením další regulační smyčky na kurs odečtením požadovaného kursu od aktuálního kursu a vynásobením konstantou, tím dostáváme opět referenci pro úhel klopení. Jedná se o P regulátor zobrazený na následujícím schématu.



**Obrázek 9 - Stranové řízení na kurs**

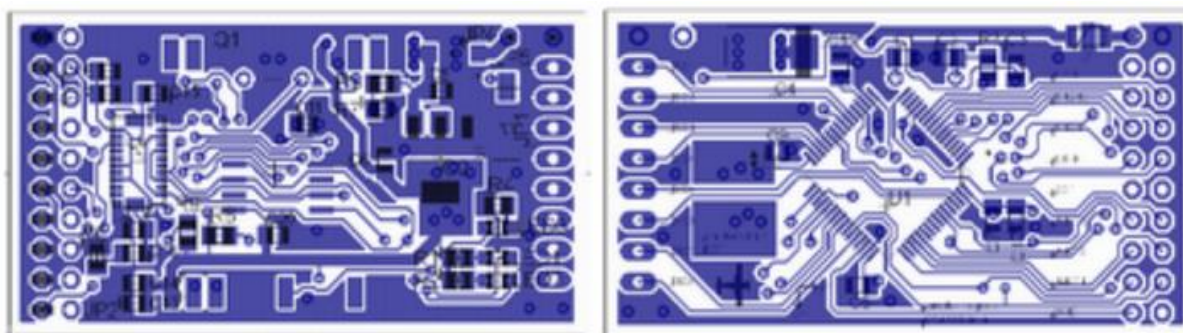
Další část stranového řízení je GPS navigace. Nastavením referenčního bodu v zeměpisných souřadnicích a odečtením od aktuální polohy získáme rozdíl zeměpisné délky a šířky. Dáme-li tyto hodnoty do poměru, získáme hodnotu rovnou tangentu úhlu. Tento úhel je náš požadovaný kurs.

## 4 Deska Autopilota [2]

Cílem vývoje této PCB<sup>7</sup> (dále jen „desky“) bylo vytvořit víceúčelovou řídicí jednotku pro zařízení typu RC modelu.

Napájení je řízeno stabilizátorem LD1117S33TR na nominální napětí 3.3 V se vstupem až 15 V, proud je limitován na 800 mA. Další důležitou informací je dropout voltage<sup>8</sup>, který je u tohoto regulátoru 1,1V (napětí 3,3V je zaručeno až při napětí větším nebo rovno 4,4 V).

Srdce desky je mikrokontrolér ARM Cortex M3 - STM32F100RB. Na desku je možné připojit komunikační modul Xbee. Dále obsahuje inerciální měřící jednotku komunikující přes I<sup>2</sup>C<sup>9</sup>, vstupní a výstupní porty, které zahrnují 8 PWM<sup>10</sup> výstupů pro ovládání servomotorů, 3 USART linky pro Xbee a GPS<sup>11</sup>. Na desce je rovněž připojena flash paměť pomocí SPI<sup>12</sup>. Deska je dvouvrstvá a její rozměry jsou 80 x 40 x 15 mm.



Obrázek 10 - Deska plošných spojů autopilota

### 4.1 Inerciální měřící jednotka

Tato jednotka kombinuje MEMS<sup>13</sup> akcelerometry, senzory pro měření úhlových rychlostí a magnetometr. Deska obsahuje senzor LSM330DL komunikující pomocí sběrnice I<sup>2</sup>C, který měří ve

---

<sup>7</sup> PCB - Printed Circuit Board, neboli deska plošných spojů [5]

<sup>8</sup> Propad napětí mezi vstupním a výstupním napětím

<sup>9</sup> Inter-Integrated Circuit, neboli multi-masterová sériová sběrnice [6]

<sup>10</sup> Pulse Width Modulation[7]

<sup>11</sup> Global Positioning System [8]

<sup>12</sup> Serial Peripheral Interface [9]

<sup>13</sup> Micro-Electro-Mechanical Systems [10]



třech osách zrychlení a úhlovou rychlost. Tento modul může být doplněn magnetometrem HMC5883, připojen rovněž přes I<sup>2</sup>C. Zpracování signálů ze sensorů je zdokumentováno v práci Complementary Filtering Approach to Orientation Estimation using Inertial Sensors Only od V. Koubelky a M. Reinsteina [13]. Výstupem jsou již polohové úhly a rychlosti. S připojeným magnetometrem i kalibrovaný směr k magnetickému severu.

## **4.2 Komunikační modul**

Pro komunikaci s pozemní stanicí a počítačem je použit bezdrátový modem s vysokým dosahem, který je závislý na použitém modulu. Na desce je slot, do kterého je možné zapojit Xbee Pro moduly. Použity mohou být moduly o různých frekvencích, 2,4 GHz, 868 MHz nebo 912 MHz [2]. Komunikace modulu s mikrokontrolérem je řešena pomocí UART 1. Pokud je to nutné, je možné zapojit i jiné moduly pomocí bočních konektorů.

## **4.3 Senzory**

Pro různé použití desky je možné připojit k tomu adekvátní senzory. Senzory se připojují na boční stranu desky, kde z jedné strany se připojují senzory a na druhé straně servomotory.

### **4.3.1 GPS**

GPS modul komunikuje s mikrokontrolérem pomocí UART 2 a je napájen z desky 3,3 V. K dispozici jsou informace o pozici, přesnosti i počtu satelitů.

### **4.3.2 Analogové senzory**

Na desku je možné připojit až 8 analogových vstupů, které jsou připojeny přímo na 12 bitový převodník. Rozsah převodníku je 0 - 3,3 V.

Deska je převážně určena pro zapojení tlakových sensorů a to pro měření vzdušné rychlosti, úhlu náběhu, úhlu vybočení či barometrické výšky. Ale je možné připojit jakýkoliv sensor splňující napěťové požadavky.

## 4.4 Připojení k debuggeru

Pro nahrání firmwaru do mikrokontroléru a možné debugování je použit STM32 MCU discovery kit [11], jež se připojí na desku pomocí SWD portu. Ten je umístěn na straně vedle UART 1 a připojen přímo na mikrokontrolér.

## 4.5 Software

Software na palubě je napsán tak, že je schopný provést automatické operace a to bez připojení k pozemní stanici. Je zde několik programových úrovní, které můžeme rozdělit do následujících skupin: inicializace mikrokontroléru a periférií, nízko úroňová komunikace se senzory, komunikace s pozemní stanicí a řídicí algoritmy.

Software je programován ve vývojovém prostředí Keil  $\mu$ Vision4 [12] ve volně šiřitelné verzi, která je omezená na debugování 32Kb, což je pro náš projekt dostačující.

Celý program je detailně popsán v diplomové práci Jaroslava Halgašíka[2]. Proto se ve své práci omezím pouze na části mnou upravované či přidané.

## 5 Výběr elektroniky do letounu

Prvním úkolem bylo vybrat vhodnou elektroniku, což zahrnovalo výběr vhodného senzoru tlaku pro měření rychlosti, výběr pohonu s regulátorem, servomotorů a v neposlední řadě zajištění dostatečného množství energie pro provoz. Následující odstavce popisují jednotlivé kroky vedoucí k výběru vhodných komponent.

### 5.1 Motor

Výběr motoru proběhl na základě potřebného výkonu pro vodorovný a stoupavý let.

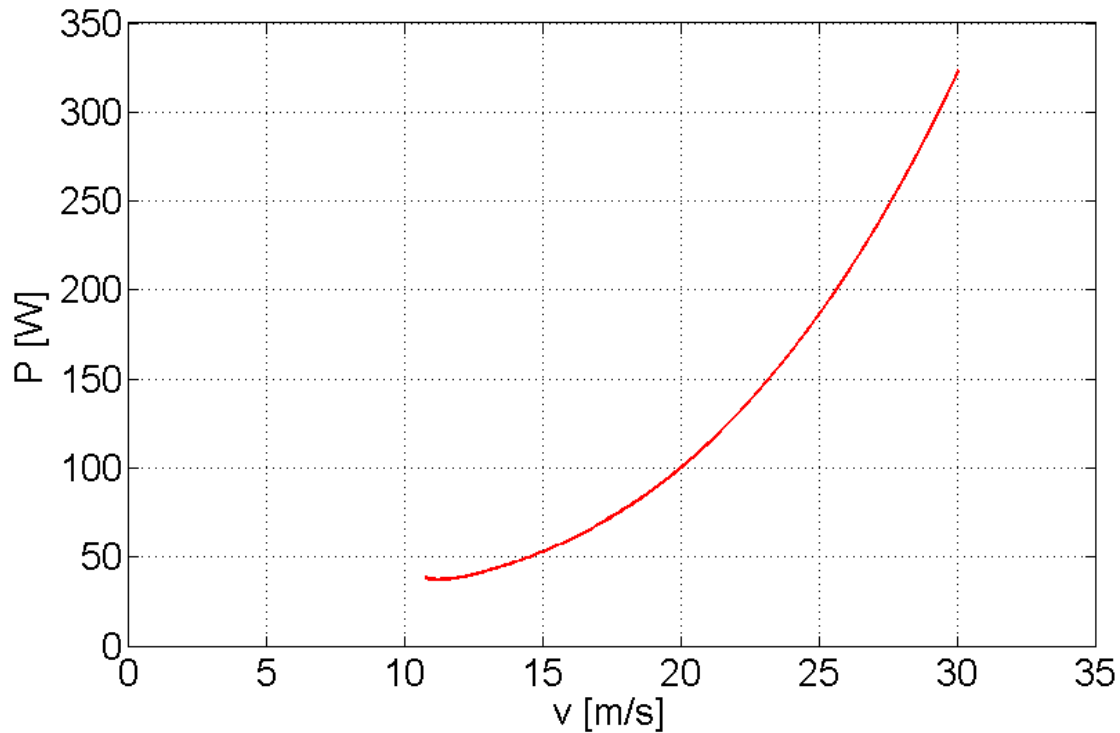
#### 5.1.1 Požadavky

K dispozici jsem měl graf potřebného výkonu pro navrhovanou vrtuli 11x7' (11' - průměr, 7 - stoupání), při 8000 rpm (rpm = *revolutions per minute*, *otáčky za minutu*). Průměr 11' vychází dle umístění vrtule, aby nebyla příliš velká a nepřekážela při startu a přistání, ale natolik velká, aby měla dostatečný tah.



Obrázek 11 - Dvojlistá vrtule 11x7' [25]

Z grafu (Obrázek 12 - Graf potřebného výkonu pro ustálený let v závislosti na rychlosti) je vidět, že pro návrhovou rychlost 15 m/s při vodorovném letu je dostačující výkon pouhých 55W.



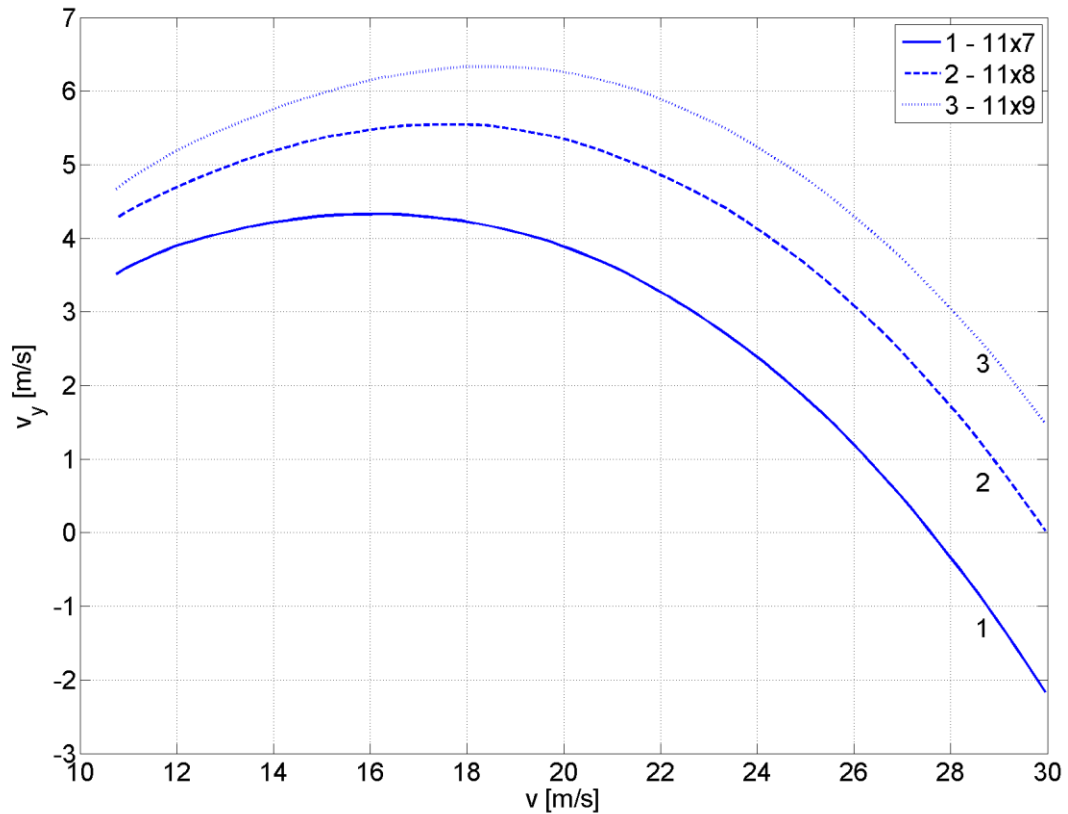
Obrázek 12 - Graf potřebného výkonu pro ustálený let v závislosti na rychlosti

Omezení pro motor bylo minimálně 55W trvalého výkonu pro vodorovný let.

Pro výpočet výkonu stoupavého letu byla použita stejná vrtule, avšak se zvýšením otáček na 10000 RPM, díky kterým vychází maximální rychlost stoupání 4,35 m/s a to při výkonu 280W, což udává další požadavky na motor [3].

Na následujícím grafu (Obrázek 13 - Rychlost stoupání v závislosti na rychlosti pro různé stoupání vrtulí) je vidět maximální možná rychlost stoupání pro vrtule 11' s rozdílným stoupáním

vrtulí.



Obrázek 13 - Rychlost stoupání v závislosti na rychlosti pro různé stoupání vrtulí

### 5.1.2 Výběr

Dle specifikovaných požadavků na výkon byl kolegou Adámkem jako první kandidát zvolen motor AXI 2820/10 GOLD LINE, který má při navrhované vrtuli 11x7 a napětí 10,42V výkon dle specifikace 292.9W při 8600 RPM.

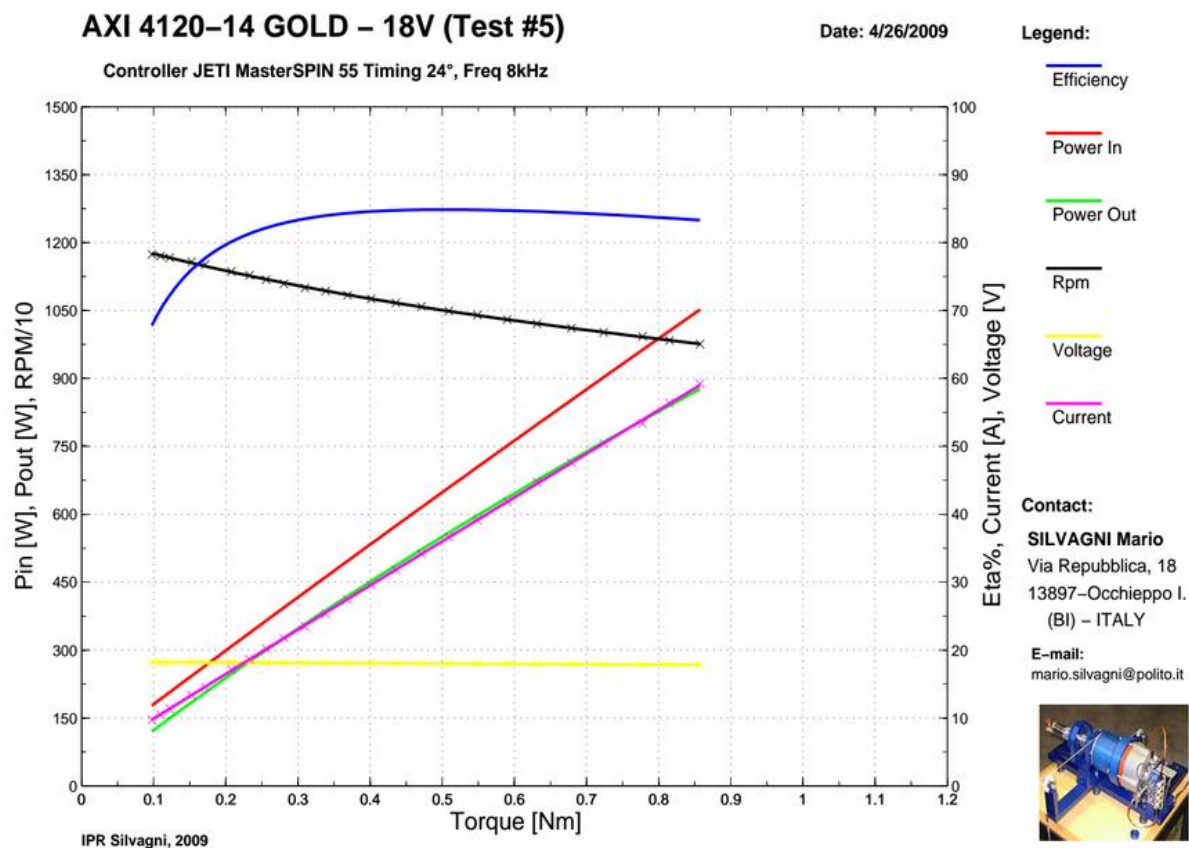
Výkon motoru by měl být dle výpočtů dostačující, i když pro stoupavý let vychází rezerva výkonu pouhých 12.9W a také nemá dostatečné otáčky. Navíc tento motor nesplňuje váhové požadavky, jelikož jak je uvedeno ve specifikaci, tento motor spadá do kategorie modelů do 2500g.

Vybral jsem tedy silnější bezkomutátorový motor s rotačním pláštěm AXI 4120/14 GOLD LINE. Tento motor je určen pro letadla od 3-5 kg, kam naše letadlo spadá.

Otáčky motoru jsou závislé na napětí a to vztahem 660 RPM na V, požadované otáčky 10000 RPM by měly být k dispozici již při 15.15V bez zátěže. Ve specifikaci motoru je uvedeno, že s vrtulí 12x8 dosáhne 10000 RPM při 17.8V.

Maximální trvalé proudy, v závislosti na chlazení, jsou 20-40A, kde také motor dosahuje účinnosti větší než 82%.

Jak je vidět z charakteristického grafu motoru, který byl dodán na požádání od výrobce, tak požadovaný výkon 280W je překonán při proudech pod 20A.



Obrázek 14 - Charakteristika motoru

## 5.2 Baterie

Baterie jsem vybíral podle několika kritérií - hmotnost, napětí, kapacita a maximální trvalý proud. Každý z následujících odstavců popisuje zmíněná kritéria.

### 5.2.1 Hmotnost

Při optimalizaci hmotnosti baterií jsem se inspiroval článkem Optimal Battery Capacity [14] zabývající se maximálním letovým časem dle hmotnosti baterií.

#### 5.2.1.1 Teoretický rozbor

Vychází z vodorovného letu, kde se musí hmotnost rovnat se vztlakem a tah s odporem, jak ukazují následující rovnice

$$m g = \frac{1}{2} C_L A \rho v^2, \quad (1)$$

$$T = \frac{1}{2} C_D A \rho v^2, \quad (2)$$

kde  $m$  je hmotnost letadla, gravitační zrychlení  $g = 9.81 \text{ m s}^{-2}$ , budeme uvažovat konstantní hustotu vzduchu  $\rho = 1.225 \text{ kg m}^{-3}$ ,  $v$  je vzdušná rychlost letadla,  $C_L$  a  $C_D$  jsou koeficienty vztlaku a odporu,  $A$  je plocha křídla a  $T$  je tah letadla.

Energie potřebná k letu v daném čase je dána rovnicí

$$E(t) = \int_0^t \frac{P}{\eta} d\tau \approx \frac{P}{\eta} t, \quad (3)$$

kde  $P$  je výkon potřebný pro udržení letové hladiny a je úměrný tahu s rychlostí,  $P = T v$ ,  $\eta$  je účinnost motoru.

Upravením rovnice (3), kde energii upravíme jako časově nezávislou a dosadíme do rovnic (1,2), tím získáme rovnici pro maximální možný čas  $t_{max}$  za maximální dodané elektrické energie  $E_{max}$

$$t_{max} = \frac{E_{max} \eta}{m^{\frac{3}{2}} \sqrt{\frac{2g^3 C_D}{\rho A^{\frac{3}{2}} C_L^2}}}. \quad (4)$$

Hmotnost letadla je závislá na energii, kterou letadlo s sebou nese, což můžeme zapsat touto jednoduchou lineární rovnicí

$$m = m_0 + \alpha E_{max}, \quad (5)$$

kde  $m_0$  je hmotnost letadla bez baterií a  $\alpha$  je převrácená hodnota hustoty energie. Po této úpravě dostáváme

$$t_{max} = \frac{E_{max}\eta}{(m_0 + \alpha E_{max})^{\frac{3}{2}}} \sqrt{\frac{2g^3 C_p}{\rho A C_L^{\frac{3}{2}}}}. \quad (6)$$

Zavedeme hmotnostní podíl  $\gamma = \alpha E_{max}/m_0$  a dosadíme do rovnice (6)

$$t_{max} = \frac{\gamma}{(1+\gamma)^{\frac{3}{2}}} \frac{\eta}{\alpha} \sqrt{\frac{\rho A C_L^{\frac{3}{2}}}{2 m_0 g^3 C_p}}. \quad (7)$$

Abychom byli schopni získat optimální hmotnost baterií, upravíme rovnici k získání bezrozměrného času  $\widehat{t}_{max}$ .

$$\widehat{t}_{max} = \frac{t_{max}}{\frac{\eta}{\alpha} \sqrt{\frac{\rho A C_L^{\frac{3}{2}}}{2 m_0 g^3 C_p}}}, \quad (8)$$

$$\text{tedy } \widehat{t}_{max} = \frac{\gamma}{(1+\gamma)^{\frac{3}{2}}}, \quad (9)$$

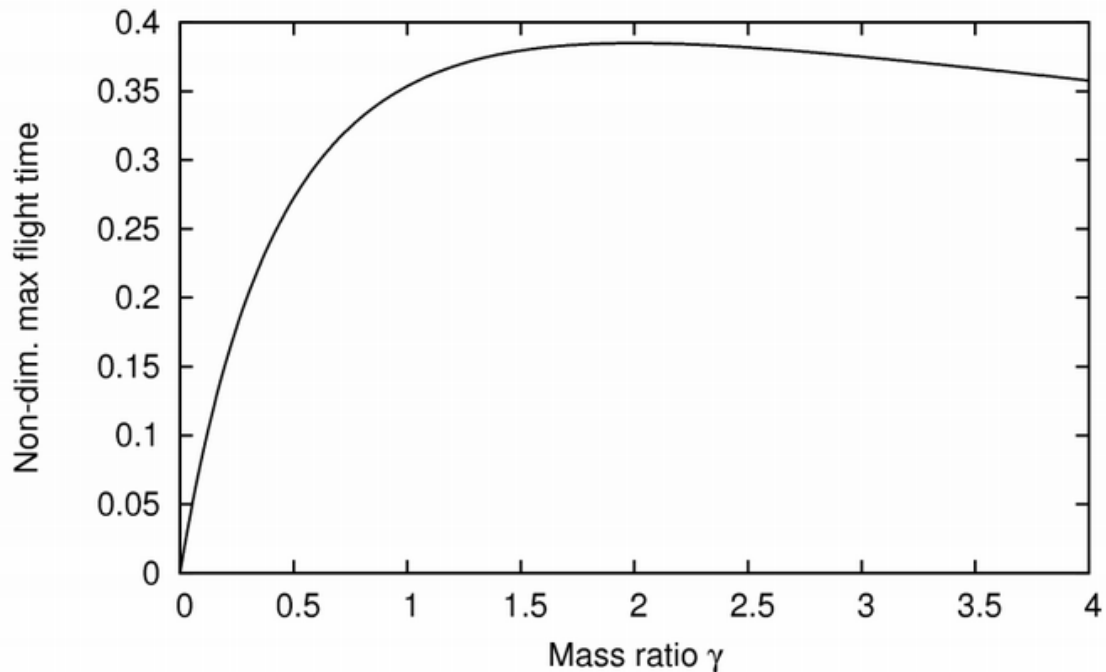
kde  $\widehat{t}_{max}$  závisí pouze na hmotnostním podílu.

Funkce (9) nabývá maxima v hodnotě  $\gamma = 2$ , z toho vychází jednoduchý závěr, který říká, že baterie by měly být ideálně dvakrát těžší než samotné letadlo bez baterií. Pokud se řídíme tímto výsledkem, dostaneme automaticky nejdelší možný letový čas v závislosti na hmotnosti baterií.

$$\widehat{t}_{max} = \frac{2}{(3)^{\frac{3}{2}}} \approx 0.385 \quad (10)$$

Tato hodnota udává maximální letový čas. Jak je vidět na grafu (Obrázek 15 - Graf bezrozměrného letového času v závislosti na hmotnostním poměru), tak zvýšením i snížením hmotnosti baterií klesne  $\widehat{t}_{max}$  pod zmíněnou hodnotu a tím klesne i letový čas.





Obrázek 15 - Graf bezrozměrného letového času v závislosti na hmotnostním poměru [14]

Tím jsme získali maximální letový čas, avšak dvojnásobná hmotnost baterií oproti hmotnosti letadla zvedne plošné zatížení a tím změní i dynamické vlastnosti letadla, což nemusí být vhodné.

Celková váha letadla je dána rovnicí, kde optimální hmotnost baterií je  $\gamma = 2$

$$m = (1 + \gamma)m_0, \text{ tedy } m = 3m_0. \quad (11)$$

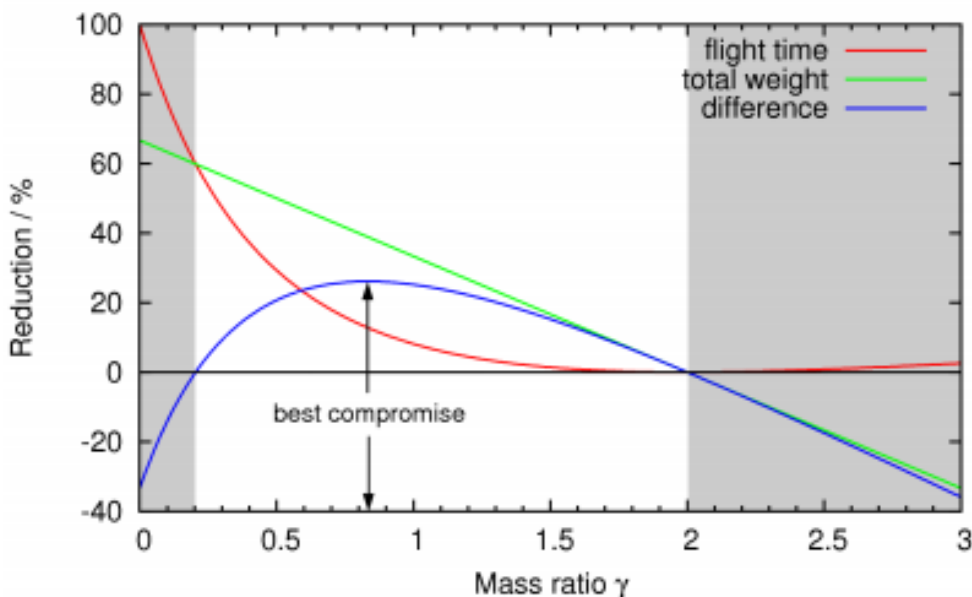
Pro nalezení největšího rozdílu mezi dobou letu a hmotností baterií je nutné vytvořit relativní změnu hmotnosti  $\Delta m$  a relativní změnu bezrozměrného času  $\Delta \widehat{t_{max}}$ .

$$\frac{\Delta m}{3m_0} = \frac{3m_m - (1+\gamma)m_m}{3m_m} = 1 - \frac{1}{3}(1 + \gamma), \quad (12)$$

$$\frac{\Delta \widehat{t_{max}}}{\widehat{t_{max}(2)}} = 1 - \frac{\sqrt{27}}{2} \frac{\gamma}{(1+\lambda)^{\frac{3}{2}}}. \quad (13)$$

Vyneseme tyto rovnice do grafu (Obrázek 16 - Graf popisující relativní letový čas, relativní hmotnost letounu a rozdíl) spolu s jejich rozdílem a vidíme, že hmotnostní podíl by se měl pohybovat mezi 0.2 až 2. V grafu také vidíme, že rozdíl relativní doby letu a relativní hmotnosti je

největší pro hmotnostní podíl  $\gamma \approx 0.833$ , pro tuto hodnotu je omezení doby letu vzhledem k váze nejlepší variantou (best compromise).



Obrázek 16 - Graf popisující relativní letový čas, relativní hmotnost letounu a rozdíl mezi hmotnostmi a relativním letovým časem [14]

Při hmotnostním podílu  $\gamma \approx 0.833$  si letadlo zanechá  $\approx 87\%$  letové času jak ukazuje rovnice (14) a hmotnost se redukuje na  $1.833 m_0$ , což je  $61\%$  oproti hmotnosti při maximálním letovém čase dle rovnice (15).

$$\widehat{t}_{max}(0.833) * \frac{100}{0.385} = \frac{0.833}{(1+0.833)^2} \frac{100}{0.385} \approx 87.18\% \approx 87\%, \quad (14)$$

$$m(0.833) * \frac{100}{3m_0} = (1 + 0.833)m_0 * \frac{100}{3m_0} = \frac{1.833}{3} * 100 = 61.1\% \approx 61\%. \quad (15)$$

### 5.2.1.2 Aplikace

Když vezmu v potaz hmotnostní podíl pro nejlepší kompromis 0.83, tak v našem případě kdy letadlo bez baterií váží 2.4 kg, by ideální baterie vážily 1.9 kg. Jelikož letadlo bylo navrženo pouze pro nosnost 4 kg, je nutné snížit váhu baterií na 1.6 kg, aby bylo splněno zadání. Po domluvě s ostatními členy projektu bylo ještě nutné upravit maximální hmotnost baterií, aby v budoucnu bylo možné implementovat do letadla kameru (uvedeno v zadání) a další elektronické zařízení. Musela být

vytvořena hmotnostní rezerva alespoň 0.2 kg, tím byla hmotnost baterií omezena na 1.4 kg. S touto rezervou se dostáváme na hmotnostní podíl  $\approx 0.58$ .

S hmotnostním podílem  $\gamma \approx 0.58$  se sníží letový čas na  $\approx 76\%$ , což je o  $11\%$  méně než při podílu hmotnosti nejlepšího kompromisu a hmotnost se redukuje na  $1.583 m_0$ , což je  $\approx 53\%$ .

### 5.2.2 Napětí

V závislosti na výběru motoru a jeho specifikaci bylo doporučeno 4-5 Li-Poly článků, tedy 14.8 V popř. 18.5 V. Jak je uvedeno ve specifikaci motoru, tak abychom dosáhli 10000 RPM je nutné použít 5 Li-Poly článků, tedy napětí 18.5 V. Maximální otáčky motoru by měly být 12210 RPM.

### 5.2.3 Proud

Dle specifikace motoru maximální trvalý proud v zátěži nepřekračuje 40A. Maximální proud, který motor zvládne po dobu 60s je 55A. Proud je možné zvednout v případě dostatečného chlazení. Baterie musí být schopné dodávat tyto proudy.

### 5.2.4 Kapacita

Požadavek na kapacitu byl jednoduchý a to co nejdelší možný vodorovný let při výše uvedené váze.

### 5.2.5 Závěr

Vybrané baterie, které splňovaly veškeré výše uvedené požadavky, byly:

Název	E-Flite LIPOL 18.5V 5000MAH 5ČL 30C EC5	FOXY G2 Li-Pol 5000mAh/18,5V 36/70C
Kapacita [mAh]	5000	5000
Napětí [V]	18.5	18.5
Nabíjecí proud [A]	5	5-10
Vybíjecí proud [A]	150 (30C)	180 (36C)
Max. vybíjecí proud [A]	200 (40C)	350 (70C)

Rozměry: v x š x d [mm]	46 x 44 x 164	43.5 x 49 x 155
Váha [g]	672	634
Cena [Kč]	3 770	2 490

Tabulka 1 - Porovnání baterií

Jako nejlepší výběr baterie jsem zvolil dvakrát paralelně zapojené - FOXY G2 Li-Pol 5000mAh/18,5V 36/70C, které jsou lehčí oproti E-Flite LIPOL 18.5 V a mohou dodávat a nabíjet vyššími proudy a přesto jsou levnější. Každá o hmotnosti 634 g, tedy celkem 1.268 kg, což je o 132 g méně než požadovaná hmotnost. Tím se upravil hmotnostní poměr na  $\approx 0.53$  a samozřejmě i letový čas  $\approx 0.73\%$  a hmotnost na 51%, navíc tímto důsledkem vznikla rezerva 0.3 kg.

Nominální napětí určující 5 Li-Pol článků dává 18.5 V. Minimální napětí baterie, pod které nesmí klesnout, jsou 3 V na článek, tedy 15 V. Maximální napětí baterie nesmí překročit 4.2 V na článek, což dává 21 V při plně nabitě baterii.

Vybíjecí proudy, které jsou tyto články schopny bezpečně zvládnout, jsou trvale 36C, tedy 180A na jednu baterii, což dostatečně převyšuje maximální proudy motoru. Maximální nabíjecí proud je 2 C, kde doporučená hodnota je 1 C, tedy 5 A. Z toho vychází i nejkratší dobíjecí čas 30 minut.

Nominální kapacita zmíněných baterií je 2x5000mAh, což odpovídá 185Wh.

Pro delší životnost článků se nedoporučuje odebírat vysoké proudy při kapacitě baterie nižší než 20-25%. Avšak v našem projektu toto nenastává a to z toho důvodu, že maximální trvalé proudy baterií jsou 360A a maximální trvalé proudy pro vybraný motor jsou okolo 40A, což je pouze 10% schopnosti baterií.

### 5.3 Regulátor

Regulátor byl vybrán na základě doporučení v modelářství, který je určen přímo pro tento motor. Byly navrženy 2 modely oba od firmy JETI, první byl s označením Advance 70 opto Pro, který ovšem nedisponuje stabilizátorem napětí pro napájení servomotorů a autopilota, takzvaný BEC obvod. Oproti tomu model spin 66 obsahuje BEC a navíc i telemetrický protokol a kupodivu je o cca 500Kč levnější. Oba regulátory jsou schopny pracovat při obdobném napětí 5-26 V (Advance 70 opto Pro 5-25.2 V) a to při trvalých proudech až 70A, plně dostačujících pro náš motor.

Na základě již zmíněných parametrů byl vybrán regulátor JETI spin 66.

Díky BEC obvodu máme k dispozici 5.5 V pro autopilota a až pro 8 servomotorů s celkovým maximálním odběrem až 5A.



Obrázek 17 - Regulátor Jeti Spin Pro 66 [24]

### 5.3.1 Telemetrie

Regulátor disponuje telemetrickými údaji jako aktuální výkon motoru v procentech, aktuální otáčky motoru, napětí baterií a teplota regulátoru.

Zmíněné údaje jsou zasílány pomocí telemetrického protokolu [15]. Tento protokol bylo nutné implementovat do destičky autopilota.

## 5.4 Senzory tlaku

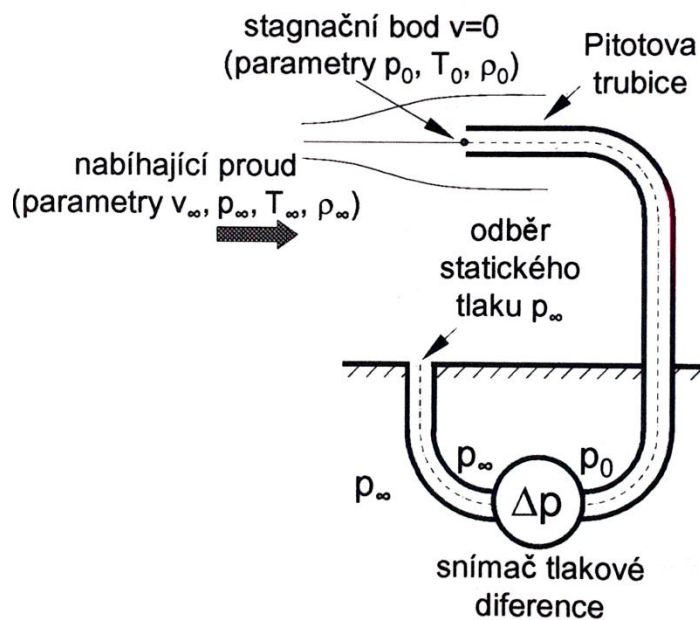
Senzory tlaku byly vybrány dle požadavků na rozsah, rozlišení, vstupní a výstupní napětí.

### 5.4.1 Měření nízkých rychlostí

Pro měření nízkých rychlostí se používají 2 typy trubice, Pitotova a Venturiho [16].

#### 5.4.1.1 Pitotova trubice

Rovina vstupního otvoru je kolmá na proud tj. rychlost v ústí trubice  $v=0$ , toto místo se nazývá stagnační bod. Odběr statického tlaku je v rovině, která je rovnoběžná s proudem, jak je ukazuje obrázek 18 - Pitotova trubice.



Obrázek 18 - Pitotova trubice [16]

“Aerodynamický” tvar Bernoulliho rovnice

$$\frac{1}{2}\rho v^2 + p = p_0 \quad (16)$$

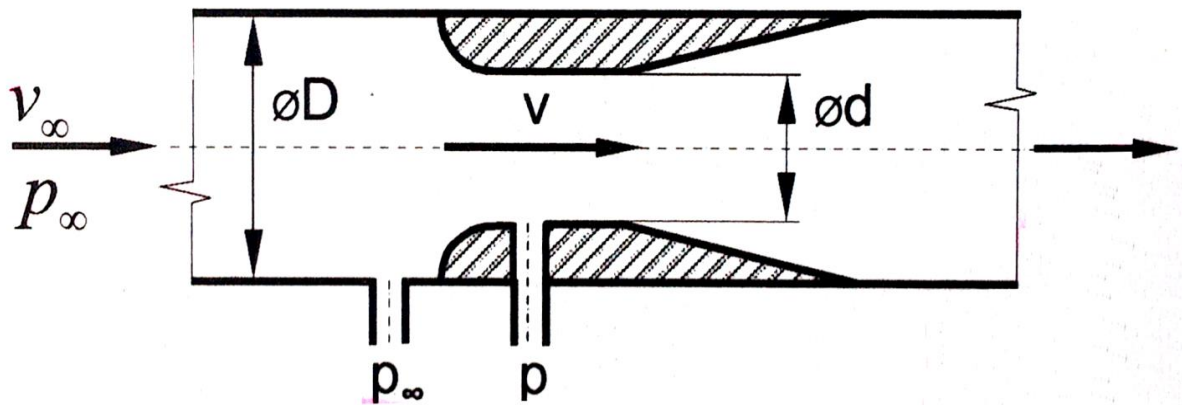
kinetický tlak + statický tlak = celkový (stagnační) tlak

z Bernoulliho rovnice odvodíme vztah pro rychlost.

$$v = \sqrt{\frac{2(p_1 - p_0)}{\rho}} = \sqrt{\frac{2\Delta p}{\rho}} \quad (17)$$

#### 5.4.1.2 Venturiho trubice

Ve Venturiho trubici probíhá odběr tlaku pouze staticky a to v zúženém místě a vně trubice, jak můžeme vidět na obrázku 19 – Venturiho trubice.



Obrázek 19 - Venturiho trubice [16]

Toto měření je výhodné a přesné pro malé rychlosti a to díky zmenšenému průměru v místě odběru tlaku, kde se zvýší rychlost. Avšak při vyšších rychlostech není možné touto trubicí měřit, jelikož v zúženém místě již brzy nastává rychlost zvuku.

Odvození rychlosti pro Venturiho trubici.

$$\frac{1}{2}\rho v_\infty^2 + p_\infty = \frac{1}{2}\rho v^2 + p, \quad (18)$$

$$\frac{v}{v_\infty} = \frac{A_D}{A_d} = \frac{\pi D^2}{\pi d^2}, \quad (19)$$

kde  $p$  je tlak,  $\rho$  je hustota,  $v$  je rychlost proudu, jak je ukázáno na obrázku 19 - Venturiho trubice. Vyjádřením  $v_\infty$  získáme vztah pro rychlost

$$\Delta p = \frac{1}{2} \rho v_\infty^2 \left[ \left( \frac{v}{v_\infty} \right)^2 - 1 \right] \Rightarrow v_\infty = \sqrt{\frac{2 \Delta p}{\rho \left[ \left( \frac{D}{d} \right)^4 - 1 \right]}} \quad (20)$$

#### 5.4.2 Požadavky

Zadavatelem byla zvolena modelářská Pitotova trubice z důvodu její jednoduché dostupnosti. V budoucnu je ovšem možné obě metody měření otestovat a popřípadě zaměnit.



Obrázek 20 – Modelářská Pitotova trubice [26]

Vstupní napětí bylo možné zvolit 3.3 V, které je k dispozici na destičce autopilota nebo 5.5 V, kterými je napájen i autopilot z regulátoru motoru.

Výstupní napětí je omezeno podle analogových vstupů na destičce autopilota, které je maximálně 3.3 V.

Rozsah jsem určil z navrhované maximální rychlosti letounu, která je 30 m/s a to za pomocí vztahu  $v = \sqrt{\frac{2(p_1 - p_0)}{\rho}} = \sqrt{\frac{2 \Delta p}{\rho}}$  (17), který jednoduchou úpravou převedeme na následující vztah

$$\Delta p = \frac{\rho v^2}{2}, \quad (21)$$



kde  $\rho = 1,225 \text{ kg m}^{-3}$  při  $15 \text{ C}^\circ$

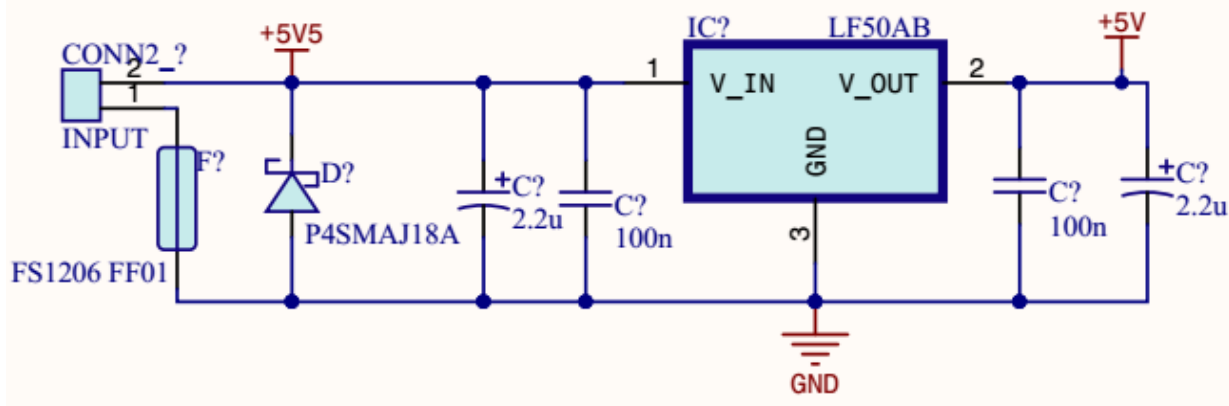
$$\Delta p = \frac{1,225 \cdot 30^2}{2} = 551.25 \text{ Pa.} \quad (22)$$

### 5.4.3 Diferenční senzor tlaku

Diferenční senzor tlaku MPXV7002DP byl vybrán na základě doporučení Ing. Pavla Pačese, Ph.D..

#### 5.4.3.1 Vstupní napětí

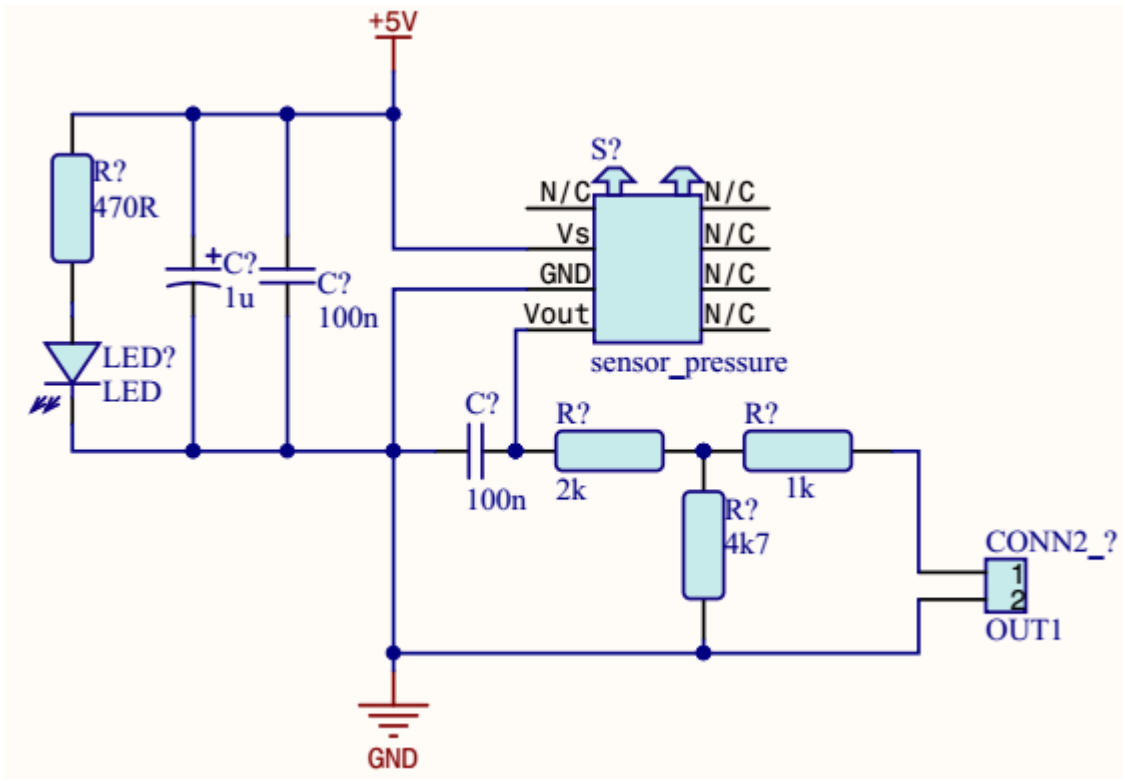
Vstupní napětí je v rozmezí 4.75 - 5.25V, jelikož z regulátoru dostáváme napětí 5.5 V a také pro udržení konstantního napětí v nezávislosti na odebíraném proudu servomotory jsem vložil stabilizátor napětí a pojistku s diodou proti přepólování.



Obrázek 21 - Stabilizátor napětí

### 5.4.3.2 Výstupní napětí

Výstupní napětí je v rozmezí od 2.25 - 4.75 V. Zde bylo nutné přidat dělič napětí na podporovaných 3.3 V, jak je vidět ve schématu zapojení. Filtrační a blokovací kondenzátory jsem zapojil na doporučení výrobce senzorů. Do obvodu byla přidána LED pro signalizaci napájení s předřadným odporem.



Obrázek 22 - Senzor tlaku

### 5.4.3.3 Rozsah

Rozsah rozdílů tlaků je až 2kPa, tedy do rychlosti

$$v \sqrt{\frac{2 \cdot 2000}{1,225}} = 57,14 \text{ ms}^{-1}, \quad (23)$$

což je téměř dvojnásobek navrhované maximální rychlosti.

### 5.4.3.4 Převodní charakteristika

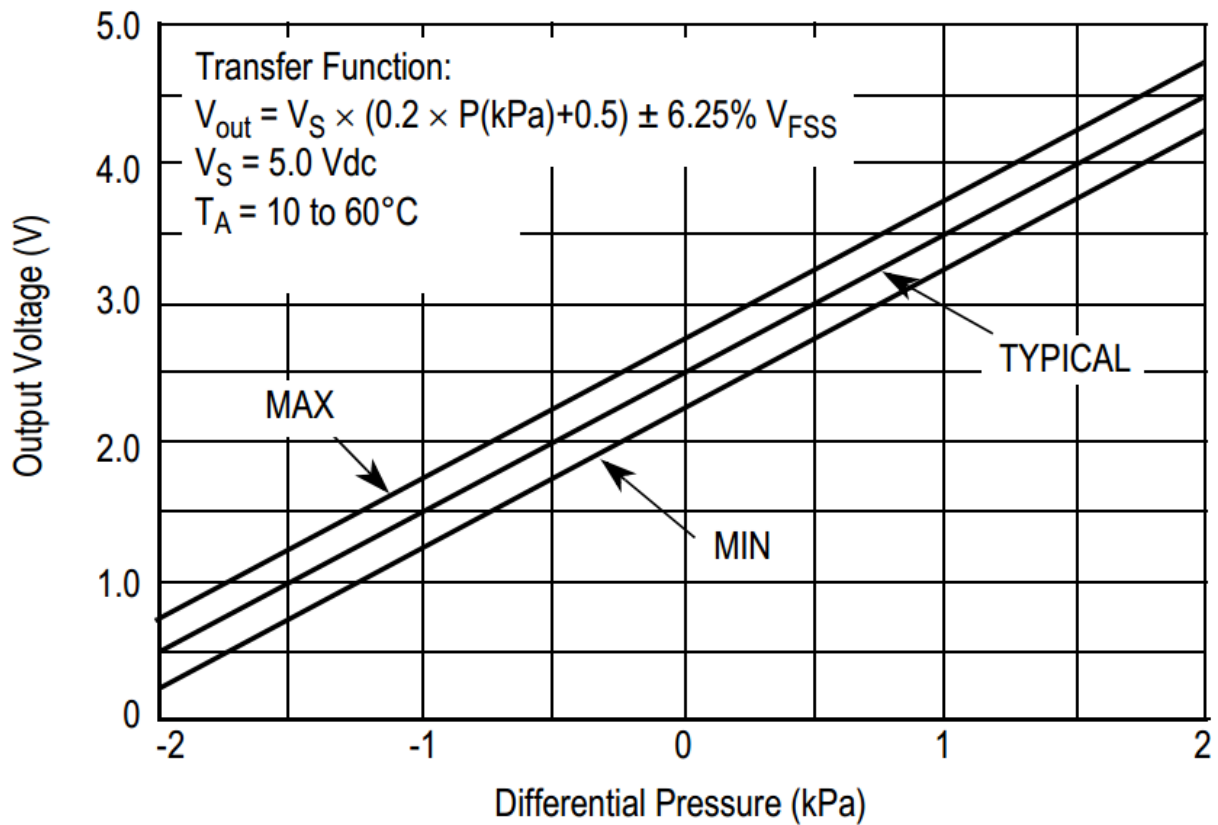
Převodová funkce tlaku na napětí nemá tvar.

$$V_{\text{out}} = V_s(0,2P + 0,5) + (2,5 - \text{offset}), \quad (24)$$

kde  $V_{out}$  je výstupní napětí senzoru,  $V_s$  je vstupní napětí senzoru,  $P$  je rozdíl tlaku v kPa a offset se pohybuje dle výrobce  $\pm 6,25\%$  od 2.5 V.

Upravím výstupní vztah dle použitého děliče a dosadím za výstupní napětí.

$$V_d = V_{out} \frac{4.7}{6.7} = \frac{4.7V_s}{6.7} (0,2P + 0.5) + \frac{4.7(2.5 - \text{offset})}{6.7} \quad (25)$$



Obrázek 23 - Závislost napětí na tlaku

## 5.5 Servomotory

Servomotory byly vybrány na základě hmotnosti letadla a rozpětí křídel na doporučení v modelářství, kde nám doporučili dva typy servomotorů HS-322HD [90] Standard Deluxe Karbonite Gear Servo a Deluxe HS-485HB[18] Karbonite Gear Servo, oba od výrobce Hitech. Oba servomotory pracují za stejného napětí a za stejných provozních teplot  $-20^{\circ}\text{C}$  až  $+60^{\circ}\text{C}$ . Rozměry jsou rovněž podobné a liší se pouze v jednotkách milimetrů. Oba servomotory již mají karbonové převody a model HS-485HB navíc disponuje kuličkovým ložiskem ve kterém je uložena výstupní hřídel, kdežto HS-322HD má pouze nylonové pouzdro. Spodní uložení je u obou typů stejné a to uložení do pouzdra. Hmotnost servomotoru HS-322HD je 43g, což je o 2 gramy méně než u druhého typu.

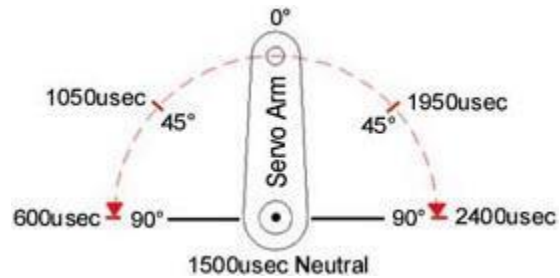
V následující tabulce je vidět porovnání obou servomotorů.

Označení	HS-322HD	HS-485HB
Pracovní napětí [V]	4.8 - 6	4.8 - 6
Rychlost bez zatížení pro 4.8 V [s/60°]	0.19	0.22
Rychlost bez zatížení pro 6 V [s/60°]	0.15	0.18
Točivý moment při 4.8 V [kg/cm]	3	4.8
Točivý moment při 6 V [kg/cm]	3.7	6
Hmotnost [g]	43	45
Převodovka	Karbonová	Karbonová
Horní uložení hřídele	Nylonové pouzdro	Kuličkové ložisko
Cena [Kč]	238	379

Tabulka 2 - Porovnání servomotorů

Po domluvě se zadavateli projektu byl vybrán servomotor HS-485HB z důvodu vyššího momentu a kvalitnějšího uložení hřídele.

Servomotor může operovat až 180° a to od -90° do 90°. Ovládání servomotoru je pulsem od 600 μs do 2400 μs.



Obrázek 24 - Úhel natočení podle délky pulsu [17]

Přepočet mezi úhlem a délkou pulsu ukazuje následující vztah.

$$t = (90 + \alpha) * 10 + 600, \quad (26)$$

kde  $t$  je délka pulsu [ $\mu$ s],  $\alpha$  je úhel od počáteční (nulové) hodnoty [ $^\circ$ ]  $\langle -90^\circ$  do  $90^\circ \rangle$ .

## 6 Implementace regulátoru motoru a motoru.

Implementace měla několik fází, první bylo zajištění napájení, další záleží na sepnutí on/off spínače, který spustí napájení řídicí elektroniky přes dvakrát 3-pinové konektory, černý zajišťující ovládání a druhý červený slouží pro komunikaci. Nakonec samotné připojení motoru s testováním komunikace, nastavení a výkonu motoru.

### 6.1 Napájení

K regulátoru jsem připojil vybrané baterie s nominálním napětím 18.5 V, na kterou bylo nutné připájet konektory (Obrázek 25 – Konektor) určené pro proud 40-50 A a pokryt je tepelně smrštiteľnou izolační trubičkou. Tyto konektory byly použity i pro prodlužovací kabely a následně spojeny s regulátorem za pomoci svorkovnice (Obrázek 26 – Svorkovnice) o maximálních trvalých proudech 63 A.

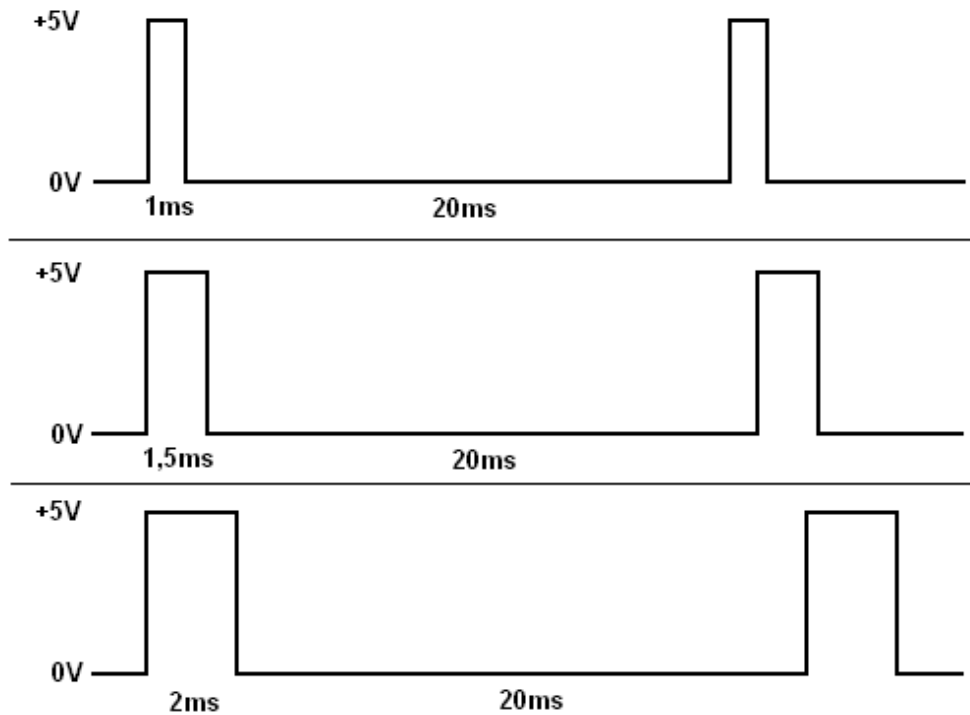


Obrázek 25 – Konektor [27]

Obrázek 26 – Svorkovnice [28]

### 6.2 Řízení

První černý konektor slouží pro ovládání výkonu motoru. Trojice kabelů představuje napájení pro řídicí jednotku 5.5 V (červený kabel) a GND (hnědý kabel). Na třetím kabelu (oranžový kabel) regulátor očekává klasický ovládací servomotorový signál, s tím, že při zapnutí je nutná inicializace maximální a minimální hodnoty. Minimální hodnota je impuls o délce 1 ms a maximální hodnota impuls o délce 2 ms. Některé servomotory mohou mít minimum i maximum s impulsem o jiné délce, ale perioda 20 ms vždy zůstává zachována.



Obrázek 27 - Zobrazení pulsů [18]

Inicializaci jsem naprogramoval, aby probíhala automaticky při spuštění, nalezneme ji v souboru autopilota, kde při spuštění je nastavena délka impulsu na 2 ms a po čase 2000 ms se nastaví na 1 ms, tím proběhne uložení hodnot do regulátoru a motor vydá dvě pípnutí o různých frekvencích. Kdyby nenastalo “stažení plynu” regulátor přejde do fáze nastavování, tedy nastanou 4 pípnutí a regulátor poté čeká “pípá” na nastavení módu stažením plynu. V našem případě regulátor běží v módu outrunner, neboli takzvaná obrácená koncepce, který je naprogramován pro motory AXI. Abychom nastavili mód outrunner musíme počkat, dokud motor nebude vydávat zvuk v podobě dvoj pípnutí, neboli mód 2 [20]. Nastavení módu jsem prováděl manuálně, automaticky se nastavují jen hodnoty plynu.

## 6.3 Telemetrie

Poslední červený 3 pinový konektor obsahuje rovněž 5.5 V (červený kabel) a GND (hnědý kabel), které jsou použity pro napájení destičky s tlakovými senzory, a jako třetí kabel (oranžový kabel) je telemetrický komunikační protokol JETI.

Po konzultaci s technickým pracovníkem firmy JETI model s.r.o., jsem se dozvěděl, že regulátor Spin Pro 66 používá pro komunikaci telemetrický protokol "prostý text". Jelikož na destičce autopilota byl nevyužitý UART3, nastavil jsem ho dle specifikace protokolu, jak je uvedeno v následujících odstavcích.

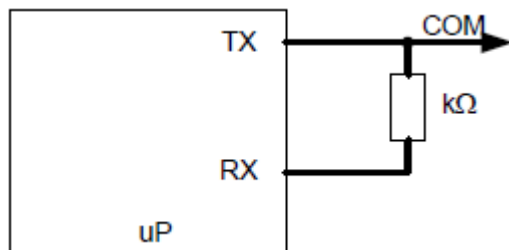
### 6.3.1 Topologie

Sběrnice má topologie typu "point to point", kde regulátor je typu master. Master inicializuje komunikaci odesláním textu v našem případě 5x SPIN PRO 66.

### 6.3.2 Fyzická a linková vrstva

Fyzická vrstva je realizována rozhraním UART v režimu half-duplex, které je fyzicky připojeno na UART 3. Rychlost komunikace je 9600 Baud<sup>14</sup> s počtem datových bitů 9, 2 stop bity a lichou paritou.

Komunikační linky RX a TX jsou fyzicky spojeny přes rezistor 2,4 k $\Omega$ .



Obrázek 28 - Fyzické zapojení telemetrické linky

<sup>14</sup> Jednotka modulační rychlosti (anglicky baud rate) udávající počet změn stavu přenosového média za jednu sekundu. V našem případě 1 baud = 1 bit/s.



Prvek, který začíná komunikovat je master. Ten odešle paket a uvolní sběrnici na dobu minimálně 20ms. V tuto dobu může slave v našem případě destička autopilota komunikovat. Odpověď od slavu není vyžadována, tedy slave nemusí komunikovat.

Pro obousměrnou komunikaci je nutné, aby řízení nebylo zapojeno, tedy aby oranžový kabel na černém konektoru nevysílal servomotorový signál. Tato funkce nebyla v našem projektu využita, protože potřebujeme pouze telemetrická data. Proto jsem jí nenaprogramoval a nebudu jí zde ani popisovat [15].

### 6.3.3 Protokol "Prostý text"

Protokol obsahuje oddělovače a prostý text ve formátu ASCII, oddělovače mají 9 bit nulový a data mají 9 bit nastaven na logickou jedničku. Vždy musí být odeslán celý 34B paket. První bajt neboli oddělovač začátku má hodnotu 0xfe, která je detekována a v tom okamžiku začíná přijímání dat až do znaku 0xFF neboli oddělovač konce.

### 6.3.4 Implementace

Po načtení celého paketu přijde na řadu rozdělení dat, které probíhá na základě pořadí, jež je dáno specifikací a to v pořadí status [B,], výkon [%], otáčky [rpm], napětí [V] a teplota [°C], dále se dopočítává kapacita baterie v procentech pro možnost vytvoření hlídání stavu baterie a jednoduššího zobrazení pro uživatele.

$$C = \left(1 - \frac{V_{\max} - V_{\text{actual}}}{V_{\max} - V_{\min}}\right) * 100, \quad (27)$$

kde C je kapacita baterie v %,  $V_{\max} = 21$  V je maximální napětí baterie, neboli 4.2 V na článek,  $V_{\text{actual}}$  je aktuální napětí baterie a  $V_{\min} = 15$  V minimální napětí baterie, neboli 3 V na článek.

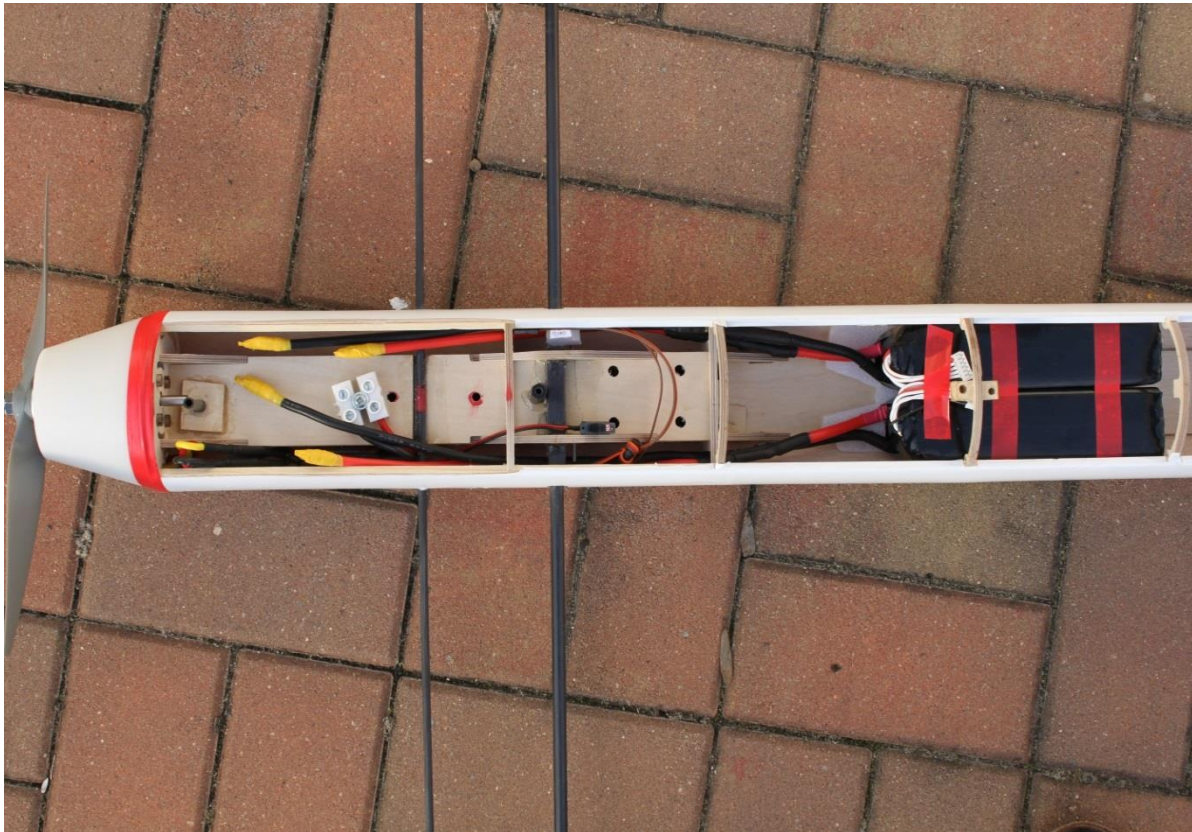
Navíc je hodnota  $V_{\text{cut}} = 15.5$  V, která z důvodu bezpečnosti článků vypne motor již při hodnotě článku 3,1 V, což je 8,3 % celkové kapacity baterie, ale autopilot a servomotory jsou stále napájeny a je možné doplachtit bezpečně na zem.

Toto je naimplementováno přímo v souboru „autopilot.c“ a napětí je možné změnit v hlavičce tohoto souboru, tedy „autopilot.h“, zde je hodnota uložena  $\text{MAX\_VOLTAGE} = V_{\max}$ , dále  $\text{MIN\_VOLTAGE} = V_{\min}$  a  $\text{CUT\_VOLTAGE} = 15.5$ . Z hodnot max a min se vypočítává aktuální kapacita,

jak ukazuje rovnice (27). Hodnota CUT\_VOLTAGE slouží pro odpojení motoru, jakmile dojde ke snížení pod tuto úroveň, začne se hodnota určující výkon motoru „přípust“ automaticky snižovat. Výkon motoru je snižován až do úplného zastavení.

## 6.4 Připojení motoru

Jelikož se jedná o tří fázový bezkomutátorový motor s pohyblivým pláštěm (statorem) a pevným rotorem, vedou k němu tři napájecí dráty, na které bylo nutno připájet konektory a přitavit izolaci, která byla dodávána spolu s motorem.



Obrázek 29 – Realizace zapojení motoru, regulátoru a baterií

Obrázek 29 – Realizace zapojení motoru, regulátoru a baterií – ukazuje výsledné zapojení motoru, regulátoru a baterií. Regulátor je umístěn u pravé stěny.

## 6.5 Testování motoru s regulací

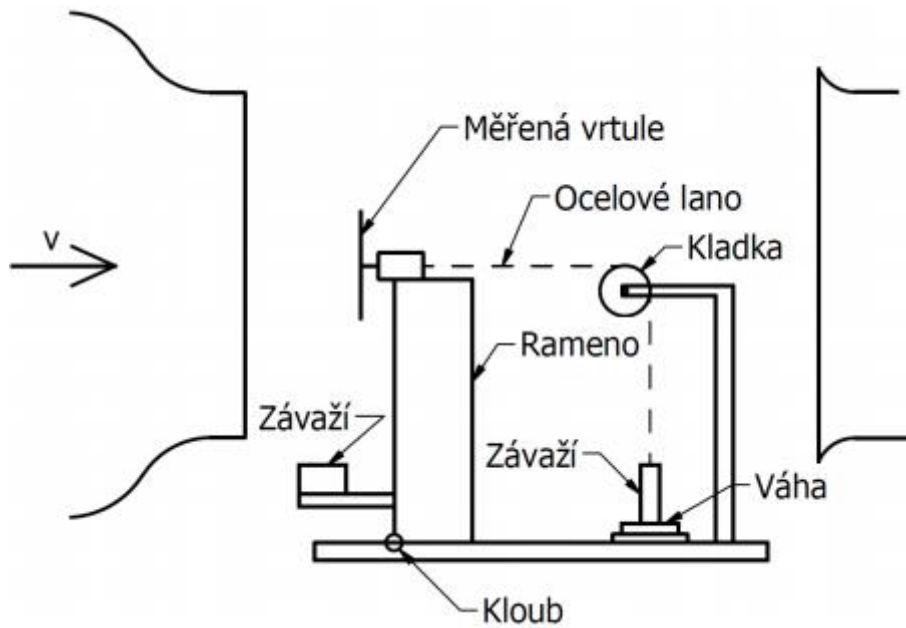
Testování motoru s regulací bylo spojeno s měřením motoru pro potvrzení teoretických hodnot uvedených v práci kolegy Adámka [3].

Pro řízení rychlosti, potažmo úhlu náklonu, je nutný znát tah, z toho důvodu bylo provedeno měření pro zjištění závislosti tahu na otáčkách motoru s návrhovou vrtulí 11x7'.

Na přípravek umístěný v měřicím tunelu byl upevněn motor s namontovanou vrtulí 11x7'. Motor je napojen na rameni, které je přes kloub spojeno se zemí. K rameni v úrovni motoru je uchyceno ocelové lanko, jež je přes kladku zatěžkáno závažím a položeno na váhu. Na druhém konci ramene je opět závaží, které vyrovnává váhy na obou stranách ramene. Tím je možné pomocí váhy dopočítat tah motoru.

Na rotační plášť motoru byla přilepena odrazová folie pro možné snímání otáček bezkontaktním senzorem otáček, pro kontrolu informace o otáčkách z regulátoru motoru.

Regulátor motoru byl připojen ke stabilnímu zdroji napětí 18.5V, který je schopen dodávat trvale až 40 A. Řízení regulátoru bylo realizováno pomocí desky autopilota, kde byla využita telemetrická data z regulátoru. Jelikož zdroj konstantního napětí měl analogový ukazatel s přesností 2 V, napětí bylo kontrolováno právě z telemetrických dat regulátoru.



Obrázek 30 - Schéma měřícího zařízení [3]

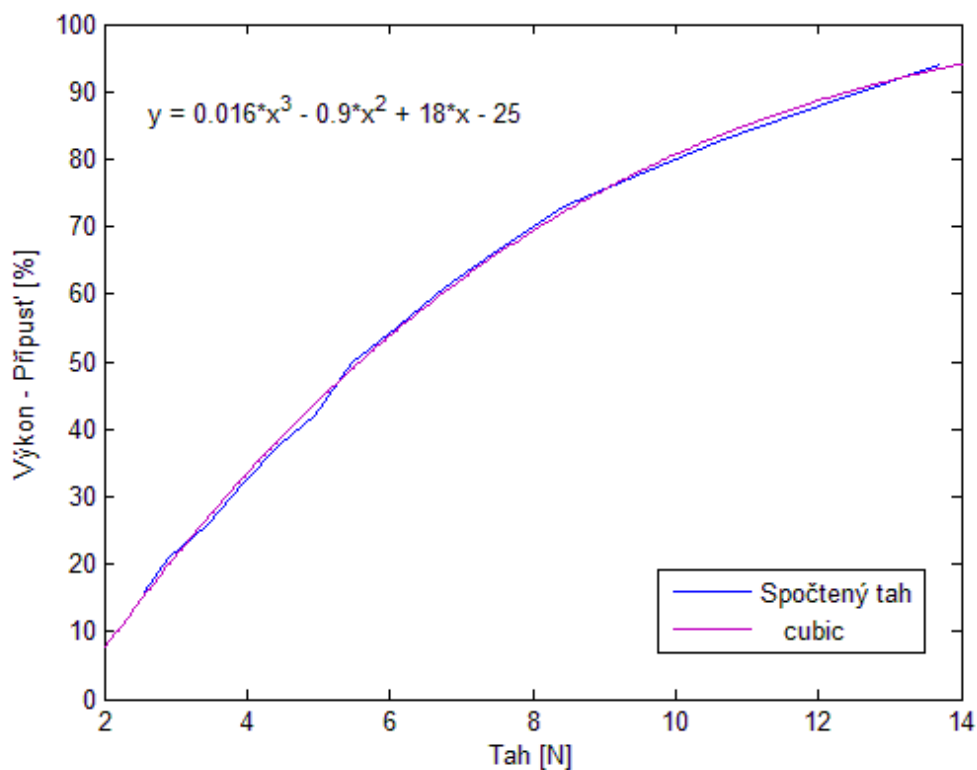
Měření proběhlo za třech rychlostí nabíhajícího proudu - staticky, poté za cestovní rychlosti  $16 \text{ ms}^{-1}$  a nakonec za rychlosti  $25 \text{ ms}^{-1}$ . Kompletní výpočty jsou uvedené v práci kolegy Adámka [3], proto se zde omezím pouze na vztah přípust' v závislosti na tahu. Následující tabulka se vztahuje pouze k cestovní rychlosti, jelikož letoun se bude pohybovat převážně v okolí této rychlosti.

Výkon [%]	15	16	21	26	32	38	42	50
Tah [N]	0	2.577	2.901	3.460	3.950	4.461	4.931	5.481

Výkon [%]	55	61	66	73	78	83	89	94
Tah [N]	6.109	6.77	7.443	8.424	9.542	10.670	12.338	13.702

Tabulka 3 - Tah motoru v závislosti na nastaveném výkonu

V tabulce je vidět, že není uvedena hodnota pro maximální výkon a to z toho důvodu, že nastalo proudové omezení zdroje. Proto není tato hodnota zpracovávána, jelikož nemá žádnou vypovídající hodnotu.



Obrázek 31 - Závislost tahu na výkonu

Obrázek 31 - Závislost tahu – zobrazuje proložení kubickou křivkou pro převod tahu na výkon. Necitlivost motoru byla zjištěna na 15%. Tedy již při výkonu 16% se motor roztočil.

Pro řízení modelu letounu je potřebný tah, avšak k dispozici máme ovládání motoru pouze procentuálně, vztah mezi nastaveným výkonem a tahem motoru s vrtulí 11x7' je dán kubickou křivkou.

$$p_{in} = 0.016 T^3 - 0.9 T^2 + 18 T - 25, \quad (28)$$

Kde  $p_{in}$  je procentuální přípust motoru, která je rovna nastavenému výkonu motoru a T je tah motoru.

Dále bylo potvrzeno, že je možné použít teoretické hodnoty koeficientu tahu  $C_t$ , jak je uvedeno v práci kolegy Adámka [3].

## 6.6 Shrnutí

Řízení letounu je navrženo na základě řízení tahu, avšak tah neznáme. Můžeme využít vztah závislosti otáček na tahu a dále řídit na otáčky, bohužel při testování motoru bylo zjištěno, že regulátor zobrazoval informace o otáčkách v zátěži (s připojenou vrtulí) až při otáčkách nad 10000 RPM bez zátěže (bez vrtule) zobrazoval hodnoty již od 0 RMP. Důvod se mi nepodařilo zjistit ani po konzultaci s technickou podporou výrobce regulátoru. Z tohoto důvodu bylo zavrhnuto řízení tahu na základě otáček. To by bylo možné až v případě namontování externího senzoru otáček. V úvahu by připadal například indukční senzor, například IA05BSF10POP<sup>15</sup>, který by bylo možné namontovat přímo k motoru. Z toho důvodu jsem používal výše uvedenou závislost přípust – tah. Důležité je, že tento vztah platí pouze pro napětí 18.5V a vrtuli 11x7'

---

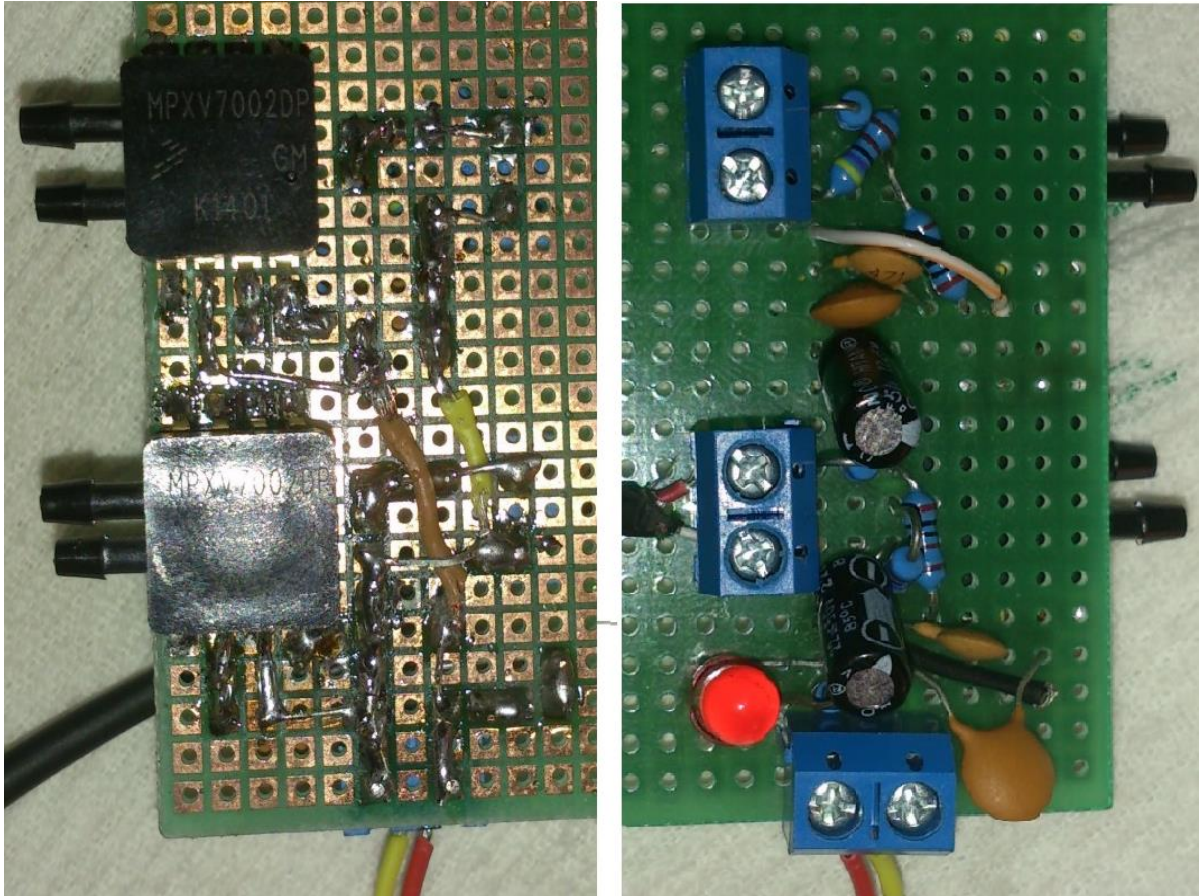
<sup>15</sup> Induktivní senzor přiblížení s krátkým tělem a hmotnosti 25g



## 7 Realizace dynamických senzorů tlaku

### 7.1 Zapojení

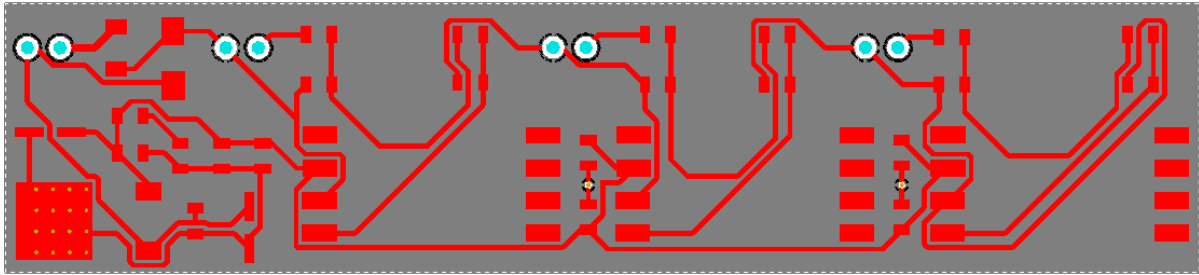
Senzory byly zapojeny dle schématu a to nejdříve v prototypové verzi (Obrázek 32 - Prototyp diferenčního měření tlaku), kde byla odzkoušena funkčnost při vstupním napětí 5 V, jelikož prototypová verze neobsahuje stabilizátor napětí, není možné zapojovat jiné napětí.



Obrázek 32 - Prototyp diferenčního měření tlaku

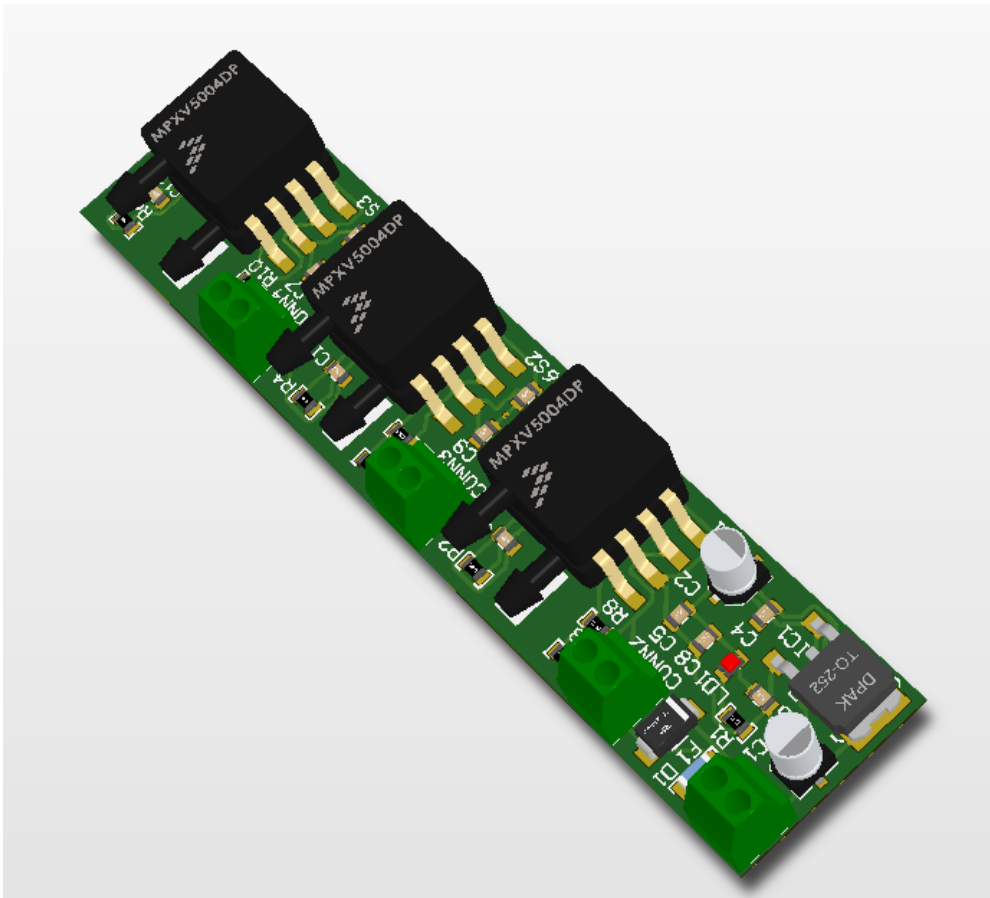
Spodní svorkovnice slouží pro napájení, které indukuje červená LED, a postranní svorkovnice jsou výstupní, kde kladná hodnota napětí je na levé straně svorkovnice.

Finální verze je již vyrobena na navrhovaném plošném spoji. Navíc obsahuje zmíněný stabilizátor a pojistku proti přepólování, kde návrh plošného spoje je zobrazen na následujícím schématu (Obrázek 33 - Návrh plošných spojů).



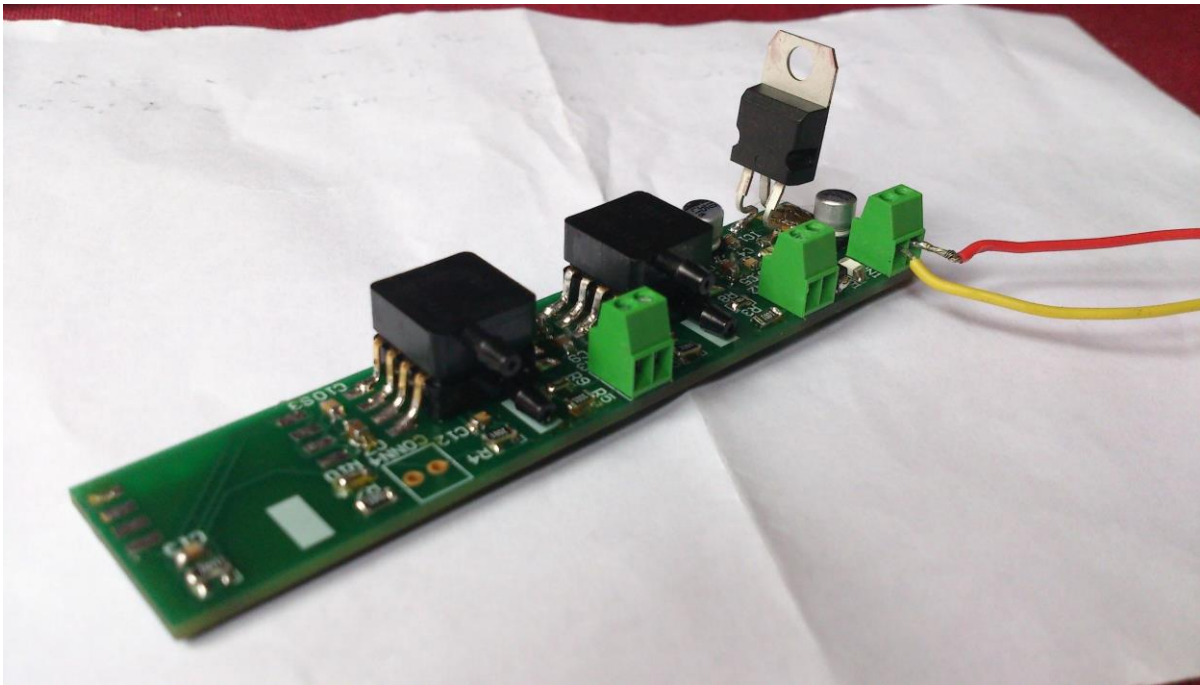
Obrázek 33 - Návrh plošných spojů

Obrázek 34 - 3D model desky pro tlakové senzory - zobrazuje výše uvedený návrh v celkovém provedení. Svorkovnice umístěná dole je napájecí vstupní napětí 5.45 - 30 V, další svorkovnice vždy u senzoru je jeho výstupem a to v rozmezí 0 - 3.3V, každý senzor je nejprve nutné kalibrovat.

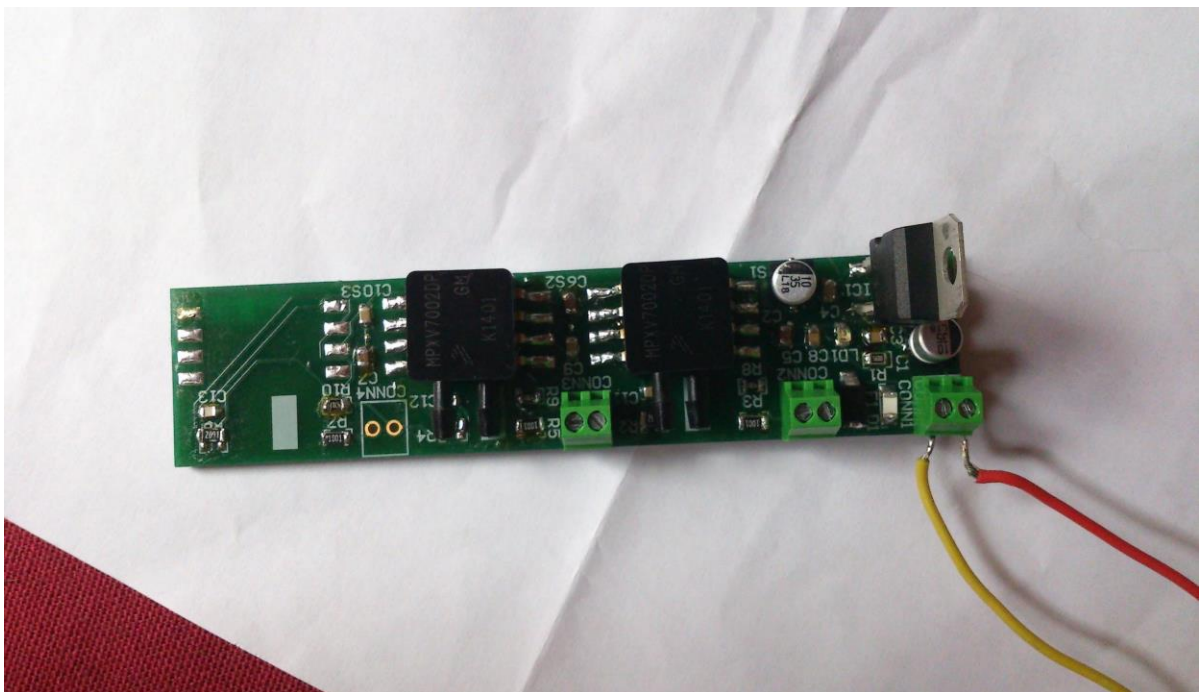


Obrázek 34 - 3D model desky pro tlakové senzory





Obrázek 35 - Realizace desky pro tlakové senzory



Obrázek 36 - Realizace desky pro tlakové senzory 2

Výsledná realizace je zobrazena na předchozích dvou obrázcích (Obrázek 35 - Realizace desky pro tlakové senzory a Obrázek 36 - Realizace desky pro tlakové senzory 2), kde je vidět, že jeden

senzor chybí a to z toho důvodu, že po odzkoušení byl nefunkční. Také zde není SMD stabilizátor, jelikož aktuálně nebyl dostupný, to však neovlivňuje funkčnost celého kitu, jelikož se aktuálně bude používat pouze jeden senzor tlaku a to pro měření vzdušné rychlosti.

## 7.2 Implementace 1/2

Senzory jsou k destičce autopilota připojeny na analogové vstupy, označeny AI<sup>16</sup> a číslem vstupu (až 8 vstupů: 4, 5, 8, 9, 10, 11, 12, 13).

Tyto hodnoty jsou přístupné po inicializaci daného vstupu a zavoláním funkce ADC1\_Read(), která vrací převedené analogové hodnoty pomocí 12 bitového převodníku z rozsahu 3.3V, tedy pro hodnotu 3.3V dostaneme 4096.

Čtení analogových vstupů jsem naimplementoval do souboru „analog.c“, kde je i popsáno čtení z jednotlivých senzorů a následně napětí přepočteno na tlak. Čtení ze všech analogových senzorů je připraveno po zavolání funkce „analog\_Read“. Aktuálně se čte pouze vzdušná rychlost, ale pomocí této funkce je možné jednoduše vytvořit obdobné funkce pro úhel náběhu i vybočení a jiné další analogové senzory.

Určení rychlosti probíhá pomocí funkce „airspeed“, která načte hodnotu z kanálu 4. Následně projde dolní propustí pro odstranění oscilací a napětí se přepočte na tlak a poté na rychlost pomocí následujících vztahů.

Ze vztahu (25) vyjádříme tlak  $P$  a za  $V_s$  dosadíme 5 V.

$$V_d = \frac{5 \cdot 4.7}{6.7} (0.2P + 0.5) + \frac{4.7 \cdot (2.5 - offset)}{6.7} \Rightarrow \quad (29)$$

$$P = \frac{67}{47} (V_d - 5 + offset) \quad (30)$$

Využijeme rovnice (17) pro výpočet rychlosti Pitotovy trubice

$$v = \sqrt{\frac{2P}{\rho}} = \sqrt{\frac{\frac{670}{235} (V_d - 5 + offset)}{\rho}}, \quad (31)$$

---

<sup>16</sup> analog input – analogový vstup

jak již bylo uvedeno hustota je konstantní a má hodnotu  $\rho = 1.225 \text{ kg m}^{-3}$ .

Jelikož uvažujeme pouze rychlosti ve směru Pitotovy trubice, tak je v programu omezena rovnice (29) pouze na kladná čísla.

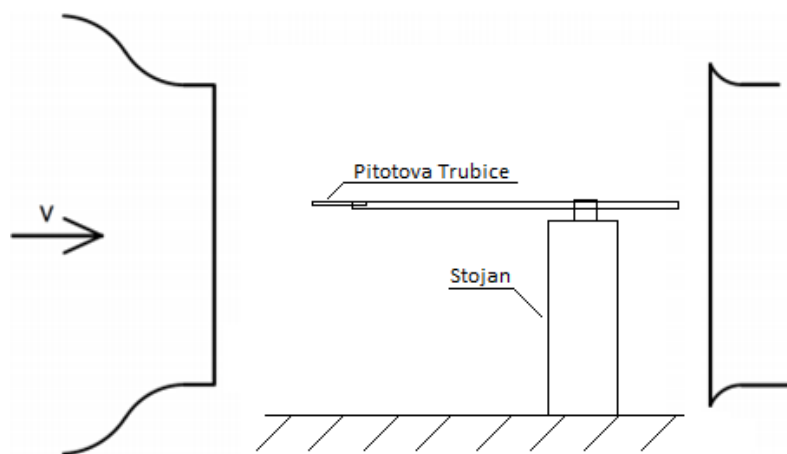
Dle datasheetu (Obrázek 23 - Závislost napětí na tlaku) má senzor lineární průběh, stačí nám ověřit pouze hodnotu napětí za nulového rozdílu tlaků, pro změření hodnoty *offset*.

### 7.3 Kalibrace

Pro ověření správnosti výpočtu a lineárního průběhu senzoru jsem udělal měření v tunelu ve VZLÚ<sup>17</sup>, kde byl senzor připojen na Pitotovu trubici a bylo snímáno napětí ze senzoru jak pomocí destičky autopilota v podobě 12 bitů, tak multimetrem PeakTech 2010 DMM, které sloužilo pouze pro ověření.

Nejprve byly změřeny odpory děliče, kde výsledný poměr je  $0.69 \pm 0.04 [-]$ . Rovněž budeme počítat s chybou stabilizátoru napětí,  $V_s = 5 \pm 0.05 \text{ V}$

Následující obrázek, ukazuje schéma umístění Pitotovy trubice, stojan byl umístěn záměrně v co možná nejdelší vzdálenosti od trubice z důvodu co nejmenšího ovlivnění měření.



Obrázek 37 - Schéma měření Pitotovy trubice

<sup>17</sup> Výzkumný a zkušební letecký ústav

Naměřené hodnoty jsem přepočítal pomocí vztahu 17, 29 a 30. Měření bylo provedeno dvakrát při zvyšování rychlosti a při snižování pro zjištění možné hystereze. Tato situace ovšem nenastala, hodnoty byly totožné, proto nebylo rovněž nutné provádět více měření. Hustota vzduchu byla v době měření  $1.16955 \text{ kg m}^{-3}$ . Měření je zobrazeno v následující tabulce, kde měřené napětí je napětí přímo zobrazené, přepočtené napětí je hodnota upravená poměrem děliče. ŘJ značí řídicí jednotku autopilota.

Rychlost v tunelu [ $\text{m s}^{-1}$ ]	Měřené napětí – multimetr [V]	Měřené napětí – ŘJ [V]	Přepočtené napětí – multimetr [V]	Přepočtené napětí – ŘJ [V]
0	$1.806 \pm 0.01$	1.807	$2.548 \pm 0.04$	2.549
5	$1.816 \pm 0.01$	1.814	$2.562 \pm 0.04$	2.559
10	$1.845 \pm 0.01$	1.850	$2.603 \pm 0.04$	2.595
15	$1.893 \pm 0.01$	1.891	$2.670 \pm 0.04$	2.668
20	$1.960 \pm 0.01$	1.957	$2.765 \pm 0.04$	2.760
25	$2.040 \pm 0.01$	2.046	$2.878 \pm 0.04$	2.887
30	$2.150 \pm 0.01$	2.152	$3.033 \pm 0.04$	3.035

Tabulka 4 - Naměřené hodnoty, Pitotova trubice

Napětí z desky autopilota oscillovalo, proto jsem vzal 240 hodnot, což odpovídá délce měření 4.8 vteřin a spočítal průměr těchto hodnot, který je uveden v tabulce. Jelikož největší chyba je způsobena právě měřením napětí na desce autopilota, nebudu již nadále chyby měření uvádět.

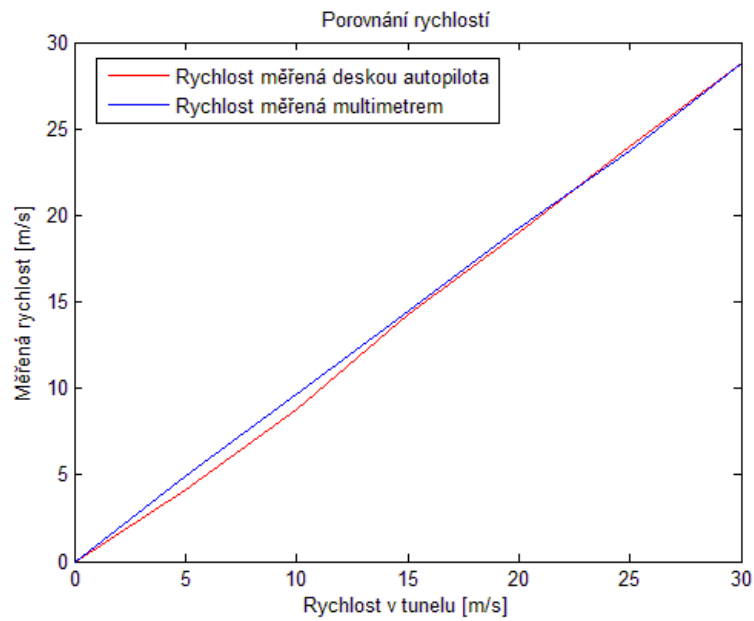
Hodnota *offset* byla určena při nulovém tlaku, která slouží pro přepočítávání na tlak.  $\Delta$  tlak je rozdíl mezi dynamickým a statickým tlakem. Byl spočten na základě změřeného napětí podle rovnice (30) s upravenou hodnotou děliče jak je popsáno ve druhém odstavci. Rychlosti byly dopočítány pomocí rovnice (31).

Rychlost v tunelu [ $\text{m s}^{-1}$ ]	$\Delta$ Tlak – tunel [Pa]	$\Delta$ Tlak – multimetr [Pa]	$\Delta$ Tlak – Deska autopilota [Pa]	Rychlost – multimetr [ $\text{m s}^{-1}$ ]	Rychlost - Deska autopilota
0	0	0	0	0	0
5	14.6	14.1	9.9	4.91	4.11
10	58.5	55.0	45.4	9.70	8.81

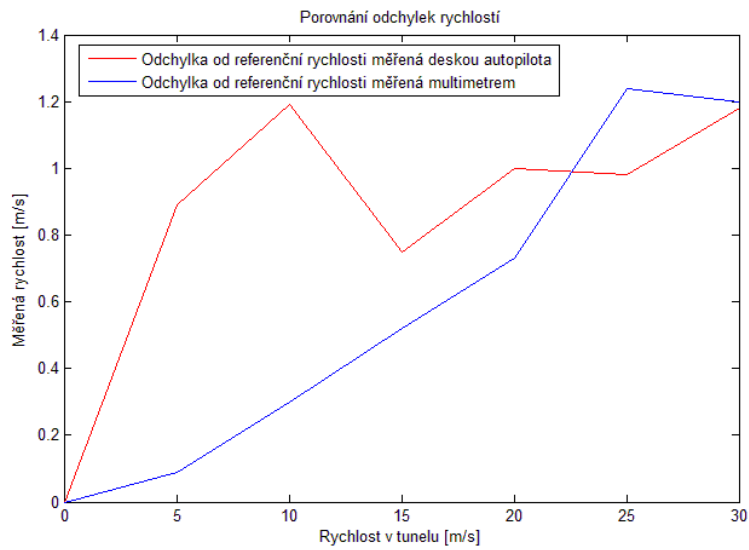
15	131.6	122.7	118.8	14.48	14.25
20	234.0	217.3	211.2	19.27	19.00
25	365.6	330.1	337.6	23.76	24.02
30	526.5	485.3	486.0	28.80	28.82

Tabulka 5 - Spočtené hodnoty, Pitotova trubice

Na grafu (Obrázek 38 - Porovnání rychlostí) je vidět porovnání rychlostí měřenými jak multimetrem, tak deskou autopilota, kde na x-ové ose je skutečná rychlost v tunelu a na y-ové ose rychlost měřená.



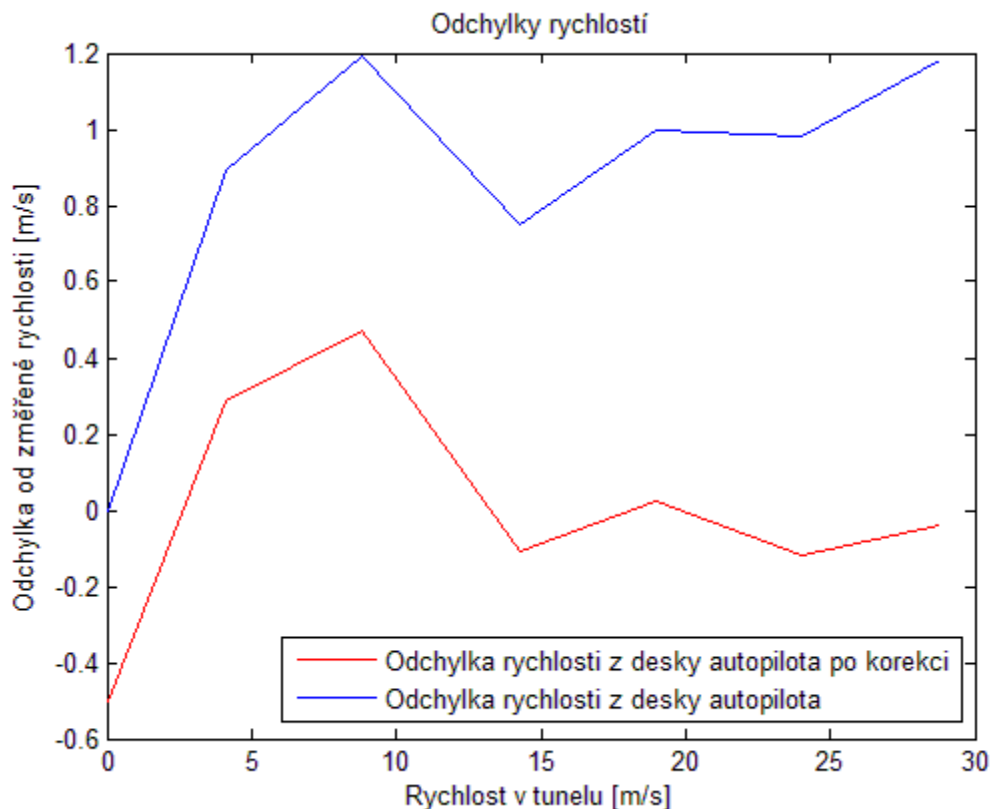
Obrázek 38 - Porovnání rychlostí



**Obrázek 39 - Odchytky rychlostí**

Odchylku od referenční hodnoty jsem proložil přímkou a zjistil korekční vztah, který je uveden v následující rovnici.

$$V_{skut} = 1.025 V_{měř} + 0.5 \quad (32)$$



**Obrázek 40 - Odchyly rychlostí před a po korekci**

Jak ukazuje graf (Obrázek 40 - Odchyly rychlostí před a po korekci), korekce nám snížila maximální odchytku ze  $1.2 \text{ ms}^{-1}$  na  $0.5 \text{ ms}^{-1}$  a to pouze při nízkých rychlostech. Jelikož letoun je navrhován na minimální letovou rychlost  $10 \text{ ms}^{-1}$ , bude k těmto rychlostem docházet pouze při vzletu a přistání.

Vyšší chyba měření při nízkých rychlostech je pravděpodobně způsobena konstrukcí Pitotovy trubice, která nemá dostatečně zakulacenou trysku. Také je nutné uvažovat, že kalibrace proběhla bez letounu, po upevnění na letoun se mohou naměřené hodnoty lišit.

Zvýšení přesnosti měření by bylo také možné, kdyby se výstup senzoru omezil pouze na kladný rozsah tlaků a následně opět využil celý rozsah řídicí jednotky autopilota.

## 7.4 Implementace 2/2

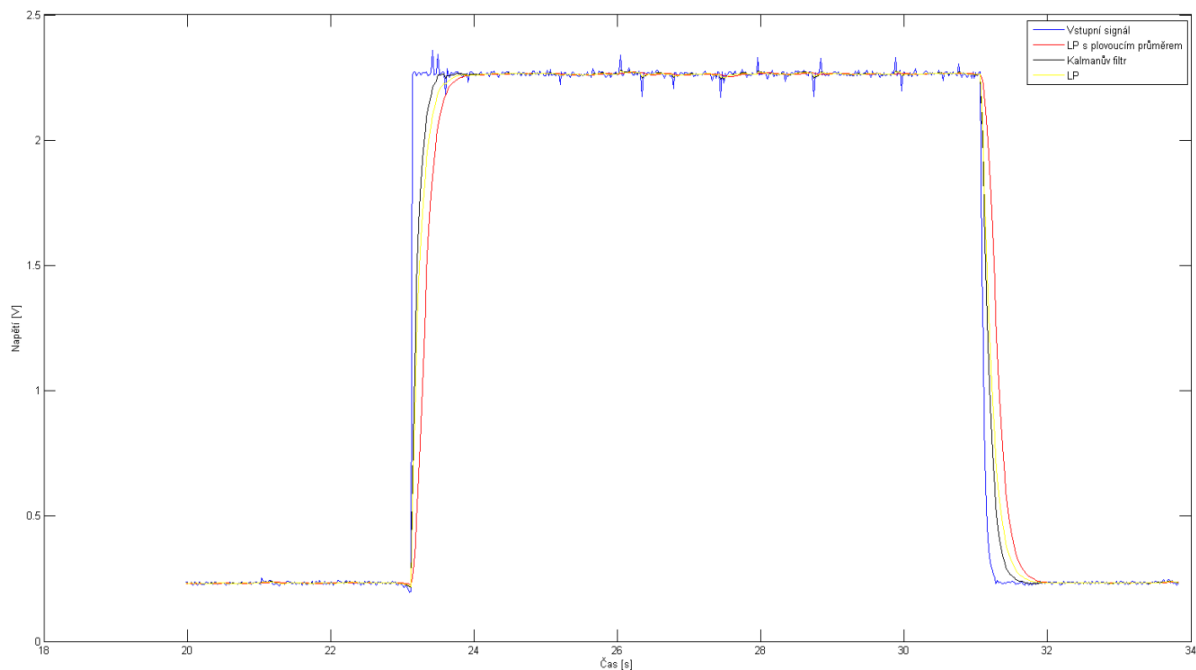
Po kalibrování jsem naimplementoval výsledný korekční vztah a doplnil čtení ze senzoru o plovoucí průměr [12.1 Vybrané funkce - 5 Airspeed].

Na začátku metody se nastaví čtení z AD převodníku, poté se zvolí kanál, inicializuje se a přečte se hodnota. Poté probíhá filtrace dolní propustí a plovoucí průměr o 10 hodnotách. Poslední podmínka je zda vychází správný výsledek neboli kladný tlak, který je kalibrován dle rovnice (32).

Dolní propust a plovoucí průměr je možné nahradit Kalmanovým filtrem [21], kde předpokládáme predikci rovnou minulému stavu [6 Airspeed\_kalman].

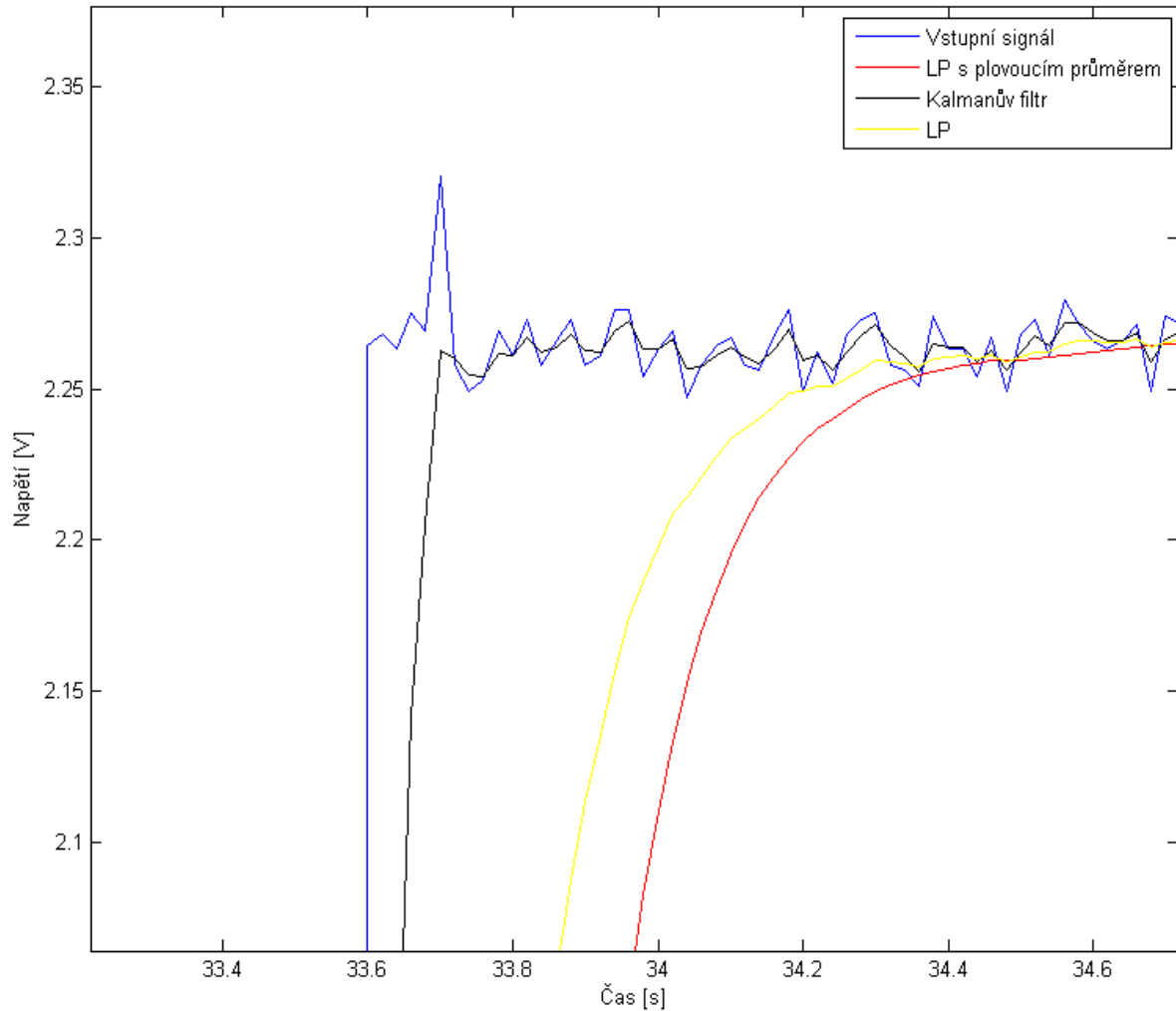
Filtry je možné měnit ve funkci „analog\_read“.

Na následujících grafech je vidět porovnání výše zmíněných dvou funkcí, kde jako vstupní hodnota je obdélník.



Obrázek 41 - Porovnání LP s plovoucím průměrem a Kalmanovým filtrem





Obrázek 42 - Detail porovnání

Na grafech je vidět, že nastavení dynamiky Kalmanova filtru je rychlejší než LP s plovoucím průměrem, což má za následek zkrácení doby náběhu, avšak zachycení i oscilace signálu jak je zobrazeno na detailu (Obrázek 42 - Detail porovnání). Pro využití Kalmanovy filtrace je nutné upravit konstanty filtru pro odstranění oscilací. Jelikož není nutné pro naše potřeby využívat Kalmanův filtr, rozhodl jsem se dále neladit konstanty filtrace a použít LP s plovoucím průměrem, neboť při letu nebudou nastávat tak rychlé změny rychlosti, aby nebyl LP s plovoucím průměrem dostačující.

## 8 Regulace

Regulátory jsou navrženy na základě Diplomové práce kolegy Nováka [1], pro jejichž implementaci bylo nejprve nutné vytvořit strukturu, která bude uchovávat požadované „referenční“ data. Jedná se o strukturu *“Reference\_data”* obsahující reference na požadovanou polohu, *Latitude* and *Longitude* (zeměpisné souřadnice [10]), *height* (výšku), *airspeed* (vzdušnou rychlost), *heading* (kurs), dále jsem přidal do struktury aktuálně nastavené ovládací páčky z pozemní stanice [*elevator* (výškovka), *aileron* ( *křídélka*), *rudder* (směrové kormido), *motor* (přípušť motoru)]. Výše zmíněná struktura je přístupná jak z *“main.c”*, kde je možné měnit reference z pozemní stanice během letu, tak v *“autopilot.c”*, který reguluje dle zmíněných referencí. Regulátory jsou naimplementovány v souboru *„autopilot.c”*, konkrétní kódy je možno vidět v příloze 12.1 Vybrané funkce.

### 8.1 Módy

Během letu je možné měnit až 4 módy letu. Po přistání je možné tyto módy měnit buď přímo nebo změnou parametru *paramcontrol* v souboru *„autopolit.c”*. Hodnoty parametru jsou 0-2, tedy 12 módů letu. První mód je vždy stejný, jedná se o mód, kdy je letoun řízen přímo pozemní stanicí, a veškeré regulace jsou vypnuty. V následující tabulce jsou popsány jednotlivé módy a jejich funkce.

Paramcontrol	Mód	Funkce
0	MAV_MODE_MANUAL	Přímé řízení – nastavuje servomotory dle ovládací páčky
0	MAV_MODE_STABILIZE	Připojení tlumičů na výškovku a křídélka
0	MAV_MODE_GUIDED	Řízení s tlumiči na úhel stoupání a úhel náklonu
0	MAV_MODE_AUTO	Řízení výšky a úhel náklonu
1	MAV_MODE_MANUAL	Přímé řízení – nastavuje servomotory dle ovládací páčky
1	MAV_MODE_STABILIZE	Řízení s tlumičem na úhel stoupání
1	MAV_MODE_GUIDED	Řízení s tlumiči na úhel stoupání a úhel náklonu
1	MAV_MODE_AUTO	Řízení s tlumiči na úhel stoupání a kurzový úhel
2	MAV_MODE_MANUAL	Přímé řízení – nastavuje servomotory dle ovládací páčky
2	MAV_MODE_STABILIZE	LQR podélná
2	MAV_MODE_GUIDED	LQR podélná i stranová
2	MAV_MODE_AUTO	LQR podélná i stranová, výšková regulace, GPS

## 8.2 Tlumiče kmitů

Funkce „damper“ má za úkol tlumit módy uvedené v kapitole 3.2 Řízení [1], jejich implementace spočívala ve vytvoření násobící funkce, do které vstupují 2 parametry aktuální rychlost změny úhlu a hodnota tlumení. Výstupem je jejich násobek.[12.1 Vybrané funkce -1 Tlumič]

## 8.3 Proporcionální regulace

Jedná se o funkci, do které vstupují 3 parametry, referenční hodnota, měřená hodnota a konstanta P. Výstupem je rozdíl referenční a požadované hodnoty roznásobený konstantou P. [12.1 Vybrané funkce -2 Proporcionální regulátor]

## 8.4 Proporcionálně-integrační regulace

Zde se již jedná o složitější funkci, jelikož obsahuje podfunkci windup, která omezuje maximální možnou integrační složku v závislosti na inicializační hodnotě.

Funkci můžeme rozdělit na dvě části, proporcionální a integrační. Proporcionální je vysvětlena výše, ke které přičteme hodnotu integrační. Tu získáme za pomoci funkce windup roznásobenou konstantou I. Do funkce windup vstupují 3 parametry, naintegrovaná složka již s přičteným aktuální přírůstkem (rozdíl referenční a aktuální hodnoty roznásobený dobou mezi touto a předchozí integrací, v našem případě 20ms), maximální hodnota (saturace) a inicializační hodnota. Výstupem je omezený součet integrační složky okolo inicializační hodnoty. [12.1 Vybrané funkce -3 Proporcionálně-integrační regulátor, 4 Windup]

## 8.5 LQ regulace

Vytvořil funkce pro LQ regulaci „LQR\_longitudinal“ a „LQR\_lateral“. Jak název napovídá, jedná se o regulaci podélného a stranového pohybu. Tyto funkce vypočítávají akční zásahy na základě matice Kp a Ks, které jsem obdržel od kolegy Nováka[1].

Rovněž jsou ve zmíněném souboru funkce pro udržení výšky „height\_regulation“, kursu „yaw\_regulation“ a pro dosažení bodu „gps\_regulation“.

### 8.5.1 Podélný pohyb

První funkce zajišťující regulační smyčku pro podélný pohyb je „LQR\_longitudinal“. Matice Kp je velikosti 2x8, převedena na 1x16, což jsou regulační konstanty. Veškeré úhlové vstupy jsou přepočteny na radiány. V této funkci [12.1 Vybrané funkce - 8 LQR\_longitudinal] je realizace

integračního členu pro integraci rychlosti se saturačním omezením. Jedná se o sčítání hodnoty spolu s odchylkou násobenou časovou změnou od předchozího výpočtu. Hodnota *INITIAL\_AIRSPEED* je počáteční hodnota integrátoru a hodnota *AIRSPEED\_WINDUP* je omezení integrátoru.

Další část funkce obsahuje výpočet akčního zásahu výškovky a tahu, opět se saturačním omezením na maximální výchylku. Vytvořil jsem strukturu pro větší přehlednost v řízení obsahující všechny čtyři možné řídicí vstupy - výškovka, křídélka, směrovky a tah s názvem *Actuators*. Obdobně jsem vytvořil strukturu s názvem *Reference*, která je dostupná po vložení hlavičky „reference.h“, obsahující informace o referenčních hodnotách. Hodnota D2R představuje přepočet stupeň na radián.

*Ins\_dataset* je struktura obsahující aktuální informace nejen o natočení letounu [*ins\_dataset.EUL* (x, y, z)]. Další struktura *Measurement* obsahuje aktuální informace o zrychlení a o úhlových rychlostech, tedy data z akcelerometru a senzoru úhlových rychlostí. Neméně důležitá struktura *Analog\_data* obsahuje informace z analogových senzorů, jak jsem popsal v kapitole 7.2 a 7.4. Struktura *last* je rovněž typu *Actuators* a obsahuje předchozí nastavené hodnoty.

Navíc jak je popsáno v kapitole 6.5 Testování motoru s regulací, kde bylo zjištěno, že regulátor poskytuje informaci o otáčkách pouze při RPM větší než 10 000, je nemožné regulovat pomocí otáček. Proto je pro převod mezi tahem a tahem v procentech (připustí motoru) vložena kubická křivka vystihující tento převod, který zobrazuje rovnice (28).

Regulaci výšky řeší funkce „*height\_regulation*“ . [12.1 Vybrané funkce - 7. Height\_regulation], ve které se vyskytuje struktura *Actual\_pos* obsahující aktuální informace z GPS, tedy polohu a výšku. Od aktuální výšky je odečtena referenční hodnota a tato odchylka je násobena konstantou. Jedná se tedy o P regulátor se saturací na úhel stoupání.

Veškeré saturace je možné měnit v souboru „autopilot.h“.

### 8.5.2 Stranový pohyb

Stranový pohyb letounu je regulován obdobně jako podélný, tedy opět pomocí LQR. O to se stará funkce „LQR\_lateral“ s maticí  $K_s$  s velikostí  $1 \times 14$ . Veškeré úhlové vstupy jsou opět přepočteny na radiány. [12.1 Vybrané funkce - 9 LQR\_lateral]

Ve funkci je realizace integrátoru na úhel vybočení se saturací, jelikož dle modelu je požadavek na úhel vybočení roven nule, není zde žádná referenční hodnota. Funkce rovněž obsahuje výpočet výchylky křidélek a kormidla. Od aktuálního úhlu náklonu je odečtena referenční hodnota, tento rozdíl je potom jediný možný námi ovlivnitelný člen, řízení je tedy pouze změnou hodnoty *reference.roll*.

Přes to je přidána další regulační smyčka pro řízení kursu, která je provedena odečtením aktuálního kursu od požadovaného a vynásobením konstantou. Jedná se tedy o P regulátor se saturací. Tato část je realizována funkcí „yaw\_regulation“ [12.1 Vybrané funkce - 10 Yaw\_regulation]

Veškeré saturace je možné měnit v souboru „autopilot.h“.

## 8.6 GPS regulace

GPS regulace nastavuje referenční hodnotu *reference.heading*. Funkce starající se o GPS navigaci se jmenuje „gps\_regulation“ [12.1 Vybrané funkce - 11 GPS regulation]. Vezme se rozdíl od požadovaných GPS souřadnic a za použití funkce atan2 se přepočte na požadovaný kurs.

Při použití této funkce se navíc dopočítá aktuální vzdušná vzdálenost od polohy letadla do referenčního bodu, uložená v proměnné *distance*.

## 8.7 Kalibrační funkce

Kalibrační funkce slouží pro výpočet počátečních podmínek integrátorů při změně módu.

[12.1 Vybrané funkce - 12 Kalibrace – dopočet integrační složky]

### 8.7.1 LQR podélné

Jedná se o výpočet počáteční podmínky pro klonění a rychlost. Z regulace LQ, z rovnice, vyjádříme proměnné  $pitch_{initial}$  a  $airspeed_{initial}$  a s parametry  $elevator$  a  $airspeed$ , dostaneme tím funkci  $f_p$

$$[pitch_{initial}, airspeed_{initial}] = f_p(elevator, airspeed). \quad (33)$$

Jelikož parametry  $elevator$  a  $airspeed$  jsou známé z aktuálního stavu letu, jedná se tedy o pouhé dosazení.

### 8.7.2 LQR stranové

Obdobně jako u podélného řešení jsou nutné u stranového dopočítat počáteční podmínky, zde je ovšem pouze jedna proměnná  $beta_{initial}$ , s parametry  $aileron$  a  $rudder$ , tím dostaneme funkci  $f_s$ .

$$[beta_{initial}] = f_s(aileron, rudder). \quad (34)$$

Parametry  $aileron$  a  $rudder$  jsou rovněž známy s aktuálního stavu letu.

### 8.7.3 Tah motoru

V tomto případě se jedná o PI regulaci, tedy dopočtení počáteční hodnoty integrační složky. Jedná se o funkci  $f_t$  o jedné neznámé  $airspeed_{initial}$  a jednom parametru  $airspeed$ .

Vzmemme-li standardní funkci pro výpočet PI regulace a vyjádříme počáteční integrální složku získáme zmíněnou funkci  $f_t$  ve tvaru:

$$[airspeed_{initial}] = f_t(airspeed) = \frac{tah - (airspeed_{referenční} - airspeed)(P-I)}{I}, \quad (35)$$

Kde P a I jsou konstanty regulátoru.

### 8.7.4 Úhel stoupání

Regulace pro úhel stoupání rovněž obsahuje integrační složku, kterou dopočteme obdobně jako u tahu motoru.

## 9 Propojení s řídicí jednotky autopilota s Matlabem

Pro ověření regulátorů jsem vytvořil obousměrnou komunikaci pro testování, kde řídicí jednotka zasílá aktuální informace o stavech jednotky. Je tedy možné porovnat implementované regulátory s regulátory simulovanými. Pro komunikaci je využito, sběrnice UART3.

### 9.1 Komunikace Matlab – řídicí jednotka

Pro navázání komunikace z Matlabu do řídicí jednotky jsem vytvořil protokol, který naplní referenční data a rovněž může měnit mód řízení. Pro zasílání dat z Matlabu je nutné připojit USB-TTL na UART3. Sběrnice je nastavena obdobně jako v případě komunikace s motorem, pro přepnutí mezi komunikací s motorem a Matlabem je nutné zaměnit funkci `parseRegData()` za `parseRefData()` v „main.c“,

BaudRate	9600
DataBits	9
StopBits	2
Parity	None
FlowControl	None

Tabulka 6 - Nastavení sběrnice, Matlab - řídicí jednotka

Data jsou ve formátu prostý text, kde rozlišení je na 30 desetinných míst, kromě prvního což je mód řízení, který nabývá pouze hodnot 1-4.

Nejprve znak odpovídající hodnotě `0xfe(254)` pro zahájení komunikace, poté mód letu a data v následujícím pořadí:

Referenční hodnoty (požadované hodnoty pro regulaci):

Úhel stoupání, úhel náklonu, kurz, rychlost, výška, longitude, latitude, tah, výškovka, křídélka, kormidlo

Stavová data (aktuální data letounu):

Eulerovy úhly - *theta*, *phi*, *psi*; úhlové rychlost – *p*, *q*, *r*; úhel náběhu, úhel skluzu a vzdušnou rychlost

Pro zasílání dat je využito funkce Serial Send který odesílá přímo ASCII, proto bylo nutné nejprve veškerá data převést, což bylo řešeno funkcí ASCII Decode, na konci datového balíčku zašleme stop

znak  $0xff(255)$ . Data se v jednotce zpracovávají okamžitě po přijetí stop znaku, proběhne rozdělení dat do struktury reference a mdata.

Řídící jednotka zasílá zpět velikosti akčních zásahů, tedy hodnoty nastavení křídélek, výškovky, kormidla a tahu motoru. Pro příjem dat je využita funkce Serial Receive, data jsou dále rozdělena dle délky zaslanych bytů, nejprve start byte  $0x00(0)$  data v rozlišení 10byt a poté stop znak  $0x0a(10)$ .

## 9.2 Hardware in the loop simulation<sup>18</sup>[29, 30]

V našem případě slouží HIL pro otestování reakce akčních ploch na změnu stavu letounu, tedy pro otestování navržených regulátorů, k tomu jsem použil komunikaci uvedenou výše a naimplementoval jsem ji do projektu od kolegy Nováka.

Důležitým prvkem je synchronizace, na hardwarové straně jsem to vyřešil změnou času (četnosti) výpočtu ten probíhal každých 20ms, nezávisle na přijatých datech. Vytvořil jsem tedy novou nekonečnou smyčku neomezenou časovým intervalem, která má za úkol kontrolovat přijatá data. Při přijetí celého paketu se spustí regulační smyčka a odešlou vypočtená data. Aby bylo dosaženo stejných podmínek jako při letu, tedy 20ms interval mezi výpočty, vyřešil jsem to na straně Simulinku, a to tak že simulace probíhá právě ve 20ms intervalech, vždy po výměně dat z řídicí jednotkou autopilota.

### 9.2.1 Testování regulace na úhel stoupání

Jako referenční hodnotou jsem nastavil  $10^\circ$ , počáteční hodnota  $0^\circ$ , a to za konstantního tahu. Regulace úhlu stoupání je řešena PI regulací, jak je popsáno v práci kolegy Nováka[1].

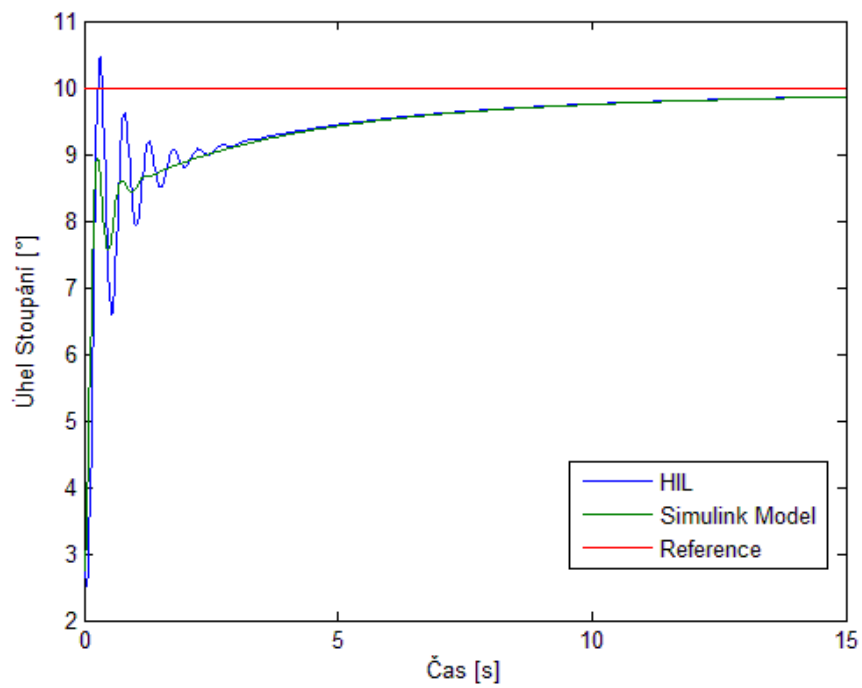
Na obrázcích níže jsou porovnány simulovaná regulace pomocí Matlabu s HIL simulací jak hodnoty výchylek výškovky, tak úhel stoupání.

Je zde větší překmit HIV oproti simulaci a rovněž delší doba kmitání, jelikož počáteční podmínky jsou shodné, může toto kmitání způsobovat převod ze spojité na diskrétní regulaci. Diskrétní vstupy výškovky jsou vidět na detailu v grafu níže (Obrázek 44 - Porovnání výchylek výškovky při regulaci na úhel stoupání).

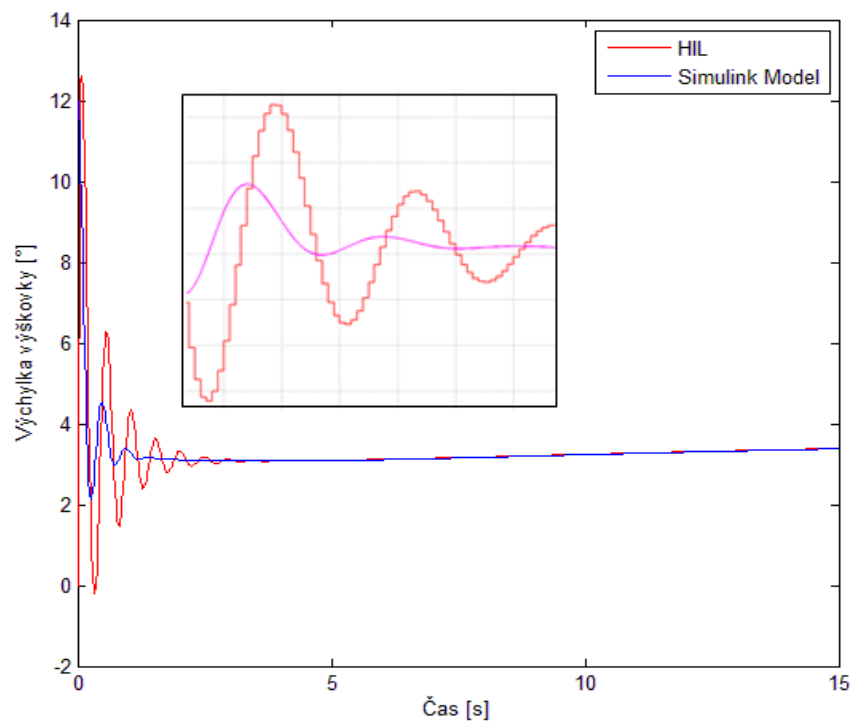
---

<sup>18</sup> [HIL] – Jedná se o reakci hardwaru se softwarovou simulací, reprezentující matematický model.





Obrázek 43 - Porovnání regulace na úhel stoupání



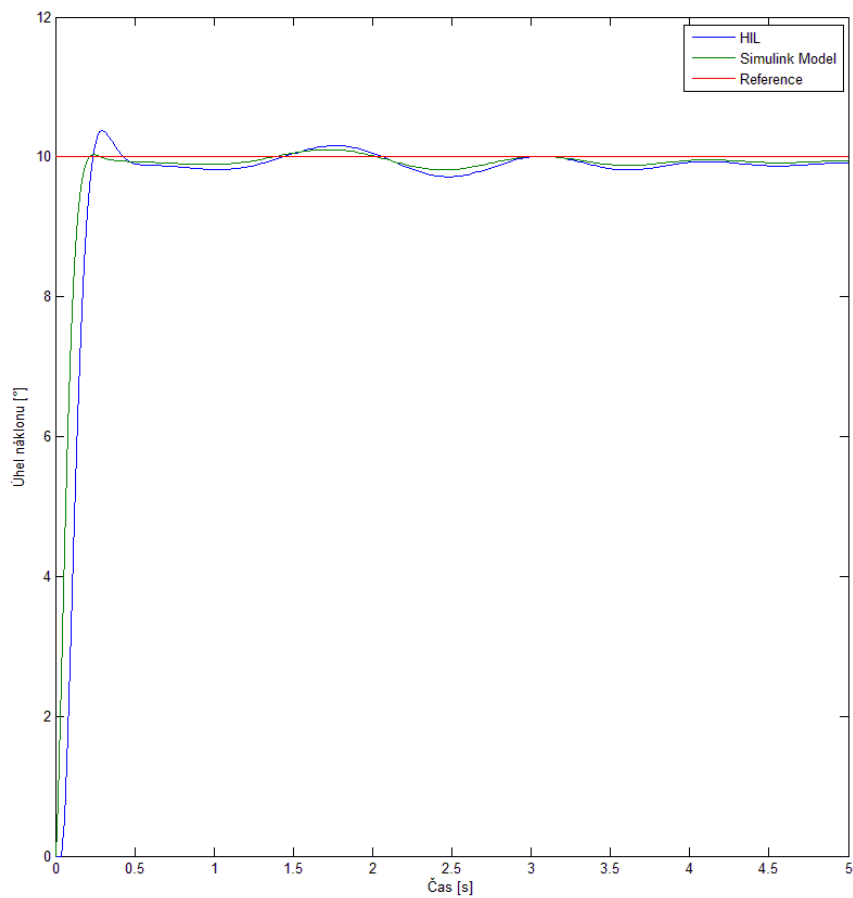
Obrázek 44 - Porovnání výchylek výškovky při regulaci na úhel stoupání

## 9.2.2 Testování regulace na úhel náklonu

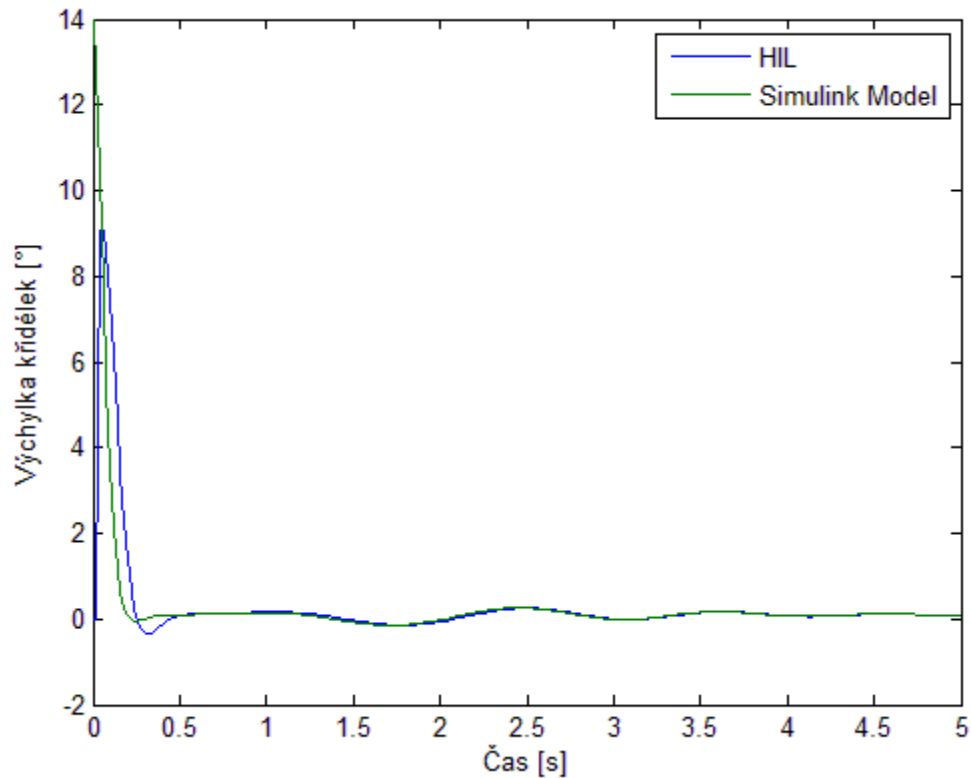
Jako referenční hodnotou jsem nastavil  $10^\circ$ , počáteční hodnota  $0^\circ$ . Regulace úhlu náklonu je řešena P regulátorem.

Obdobně jako při regulaci úhlu stoupání, můžeme vidět na obrázcích níže porovnání simulované regulace a HIL simulace pro úhel náklonu a výchylky křidélek.

Na obrázcích níže můžeme rovněž vidět zhoršení regulace při převodu na diskretní regulaci.



Obrázek 45 - Porovnání regulace na úhel náklonu



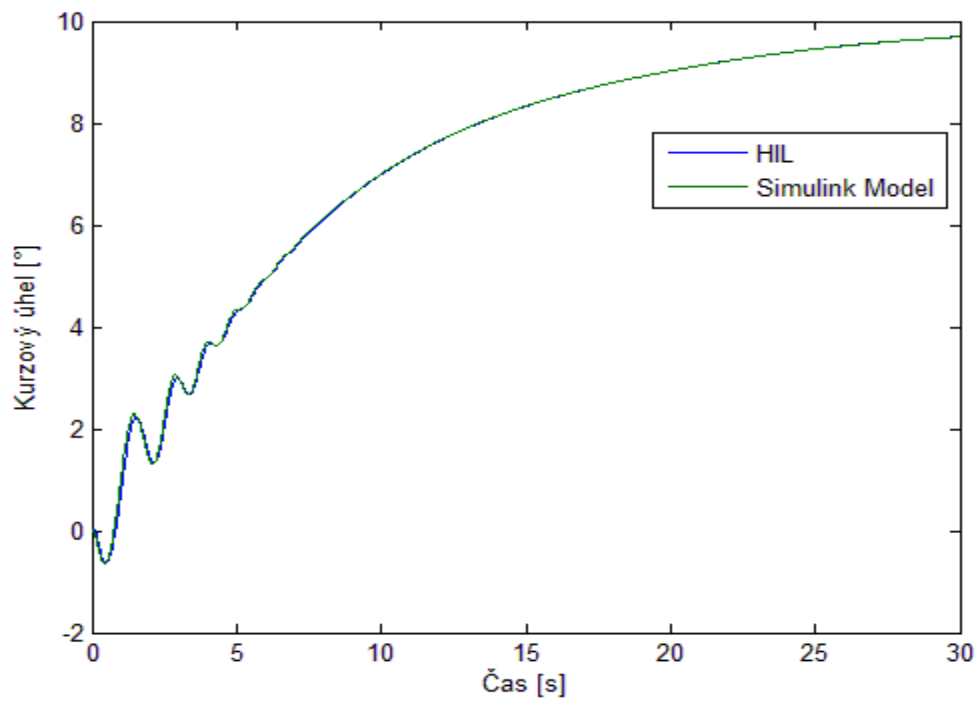
Obrázek 46 - Porovnání výchylek křídělek při regulace na úhel stoupání

### 9.2.3 Testování regulace na kurzový úhel

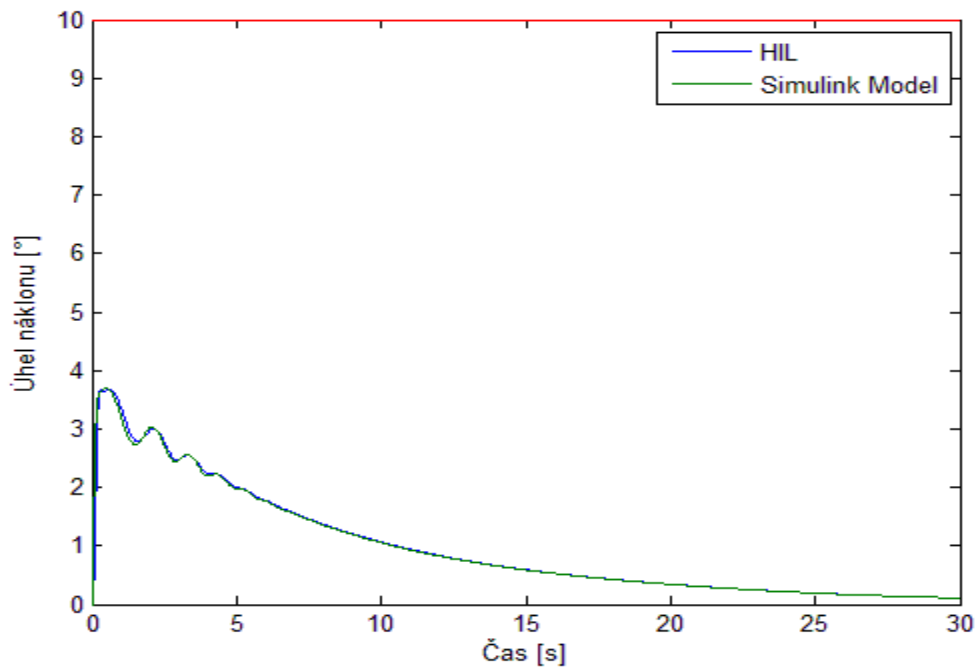
Jako referenční hodnotou jsem nastavil rovněž 10°, počáteční hodnota 0°. Regulace kurzového úhlu je řešena P regulátorem.

Obdobně jako při předchozích regulacích i zde je na následujících obrázcích ukázáno porovnání mezi simulovanou regulací a HIL simulací pro úhel náklonu, kurzový úhel a výchylky křídělek.

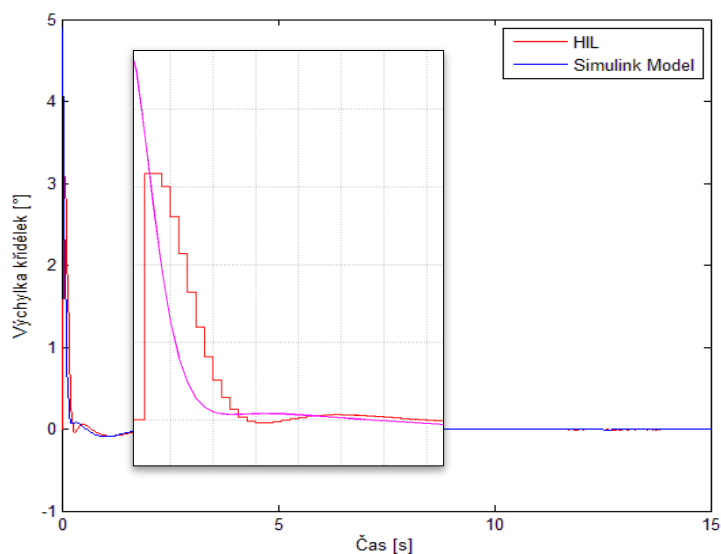
Na obrázcích je patrné že při změně kurzového úhlu diskrétní systém (HIL) téměř kopíruje systém spojitý (simulovaný model).



Obrázek 47 - Porovnání kurzové regulace



Obrázek 48 - Porovnání kurzové regulace - úhel náklonu

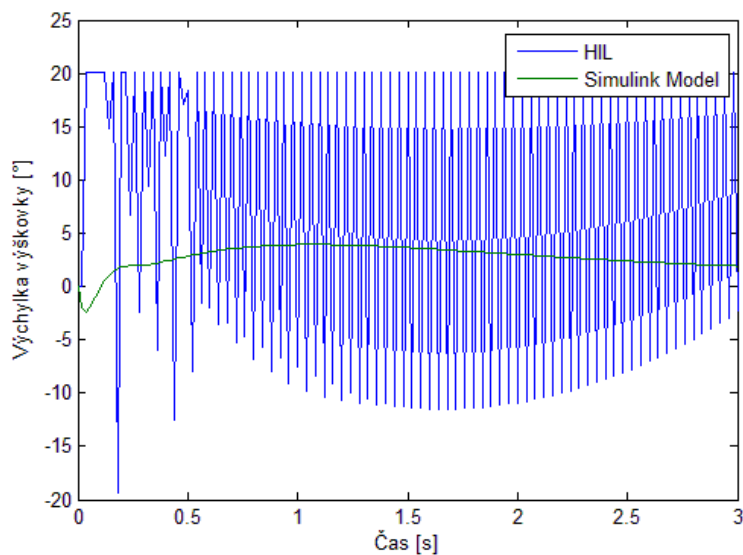


Obrázek 49 - Porovnání reakce křídledek na změnu kurzového úhlu

#### 9.2.4 Testování podélné LQ regulace

V tomto případě se jednalo o porovnání LQ regulace s referencí na  $0^\circ$  úhel stoupání.

Jak můžeme vidět v grafu, diskretizace systému značně rozkmitá výškovku, toto se však při reálném testování neprojevovalo, což můžeme vidět na videu „MOV\_1362.mp4“ v příloze. Rovněž simulace za konstantního kroku 20ms zneustabilnila i softwarově regulované regulace.

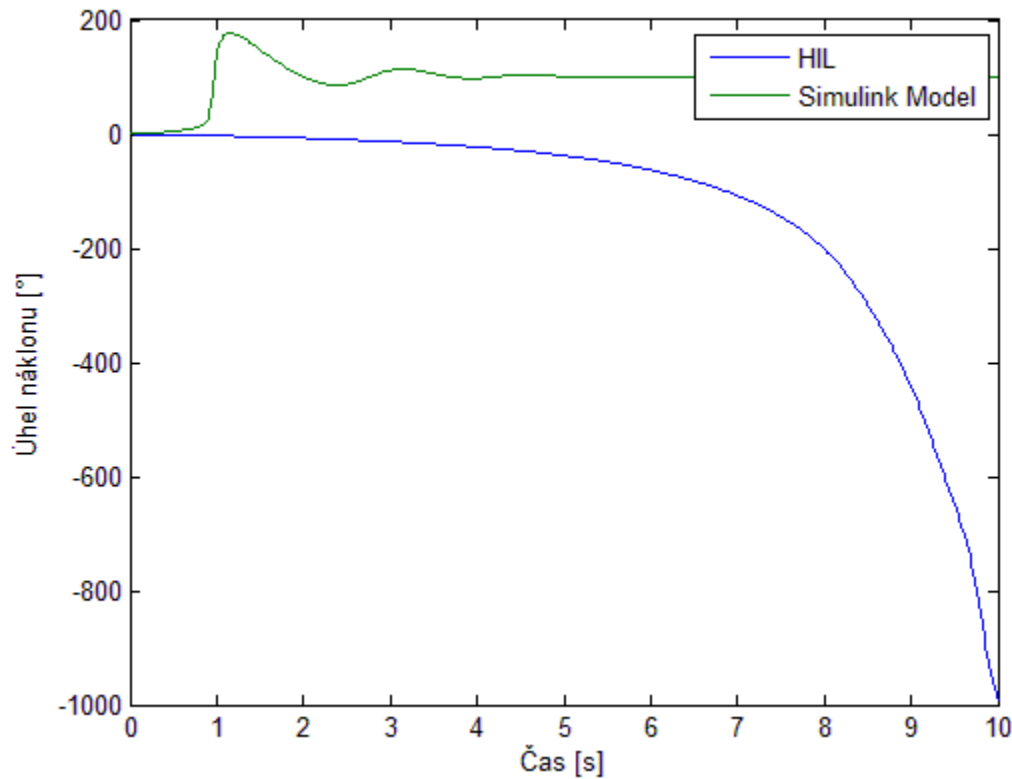


Obrázek 50 - Porovnání reakce křídledek při podélné LQ regulace

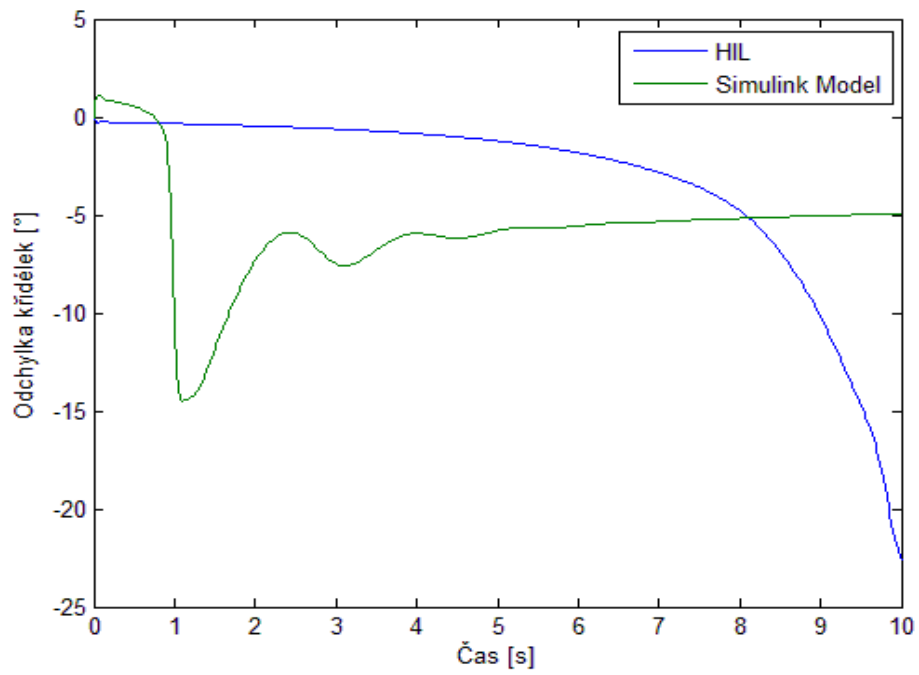
### 9.2.5 Testování stranové LQ regulace

Obdobně jako při podélné regulaci i zde se vlivem vzorkování simulace na 20ms znestabilnila jak softwarová, tak i hardwarová regulace pomocí HIL testování.

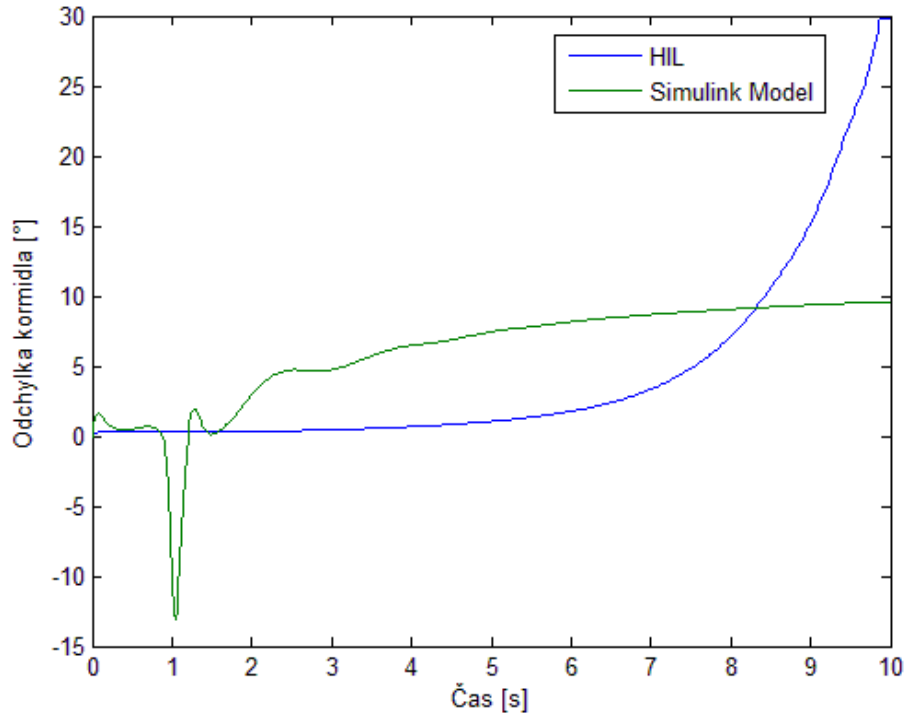
Opět jako u podélného pohybu se nestabilita regulátoru při reálném testování neprojevila, jak můžeme vidět na videu „MOV\_1359.mp4“. Na videu jsou reakce křídélek i kormidla v opačném směru.



Obrázek 51 - Porovnání regulace na úhel náklonu



Obrázek 52 - Porovnání reakce křídělek



Obrázek 53 - Porovnání odchylek kormidla

## 10 Oživení a ověření řízení na letounu

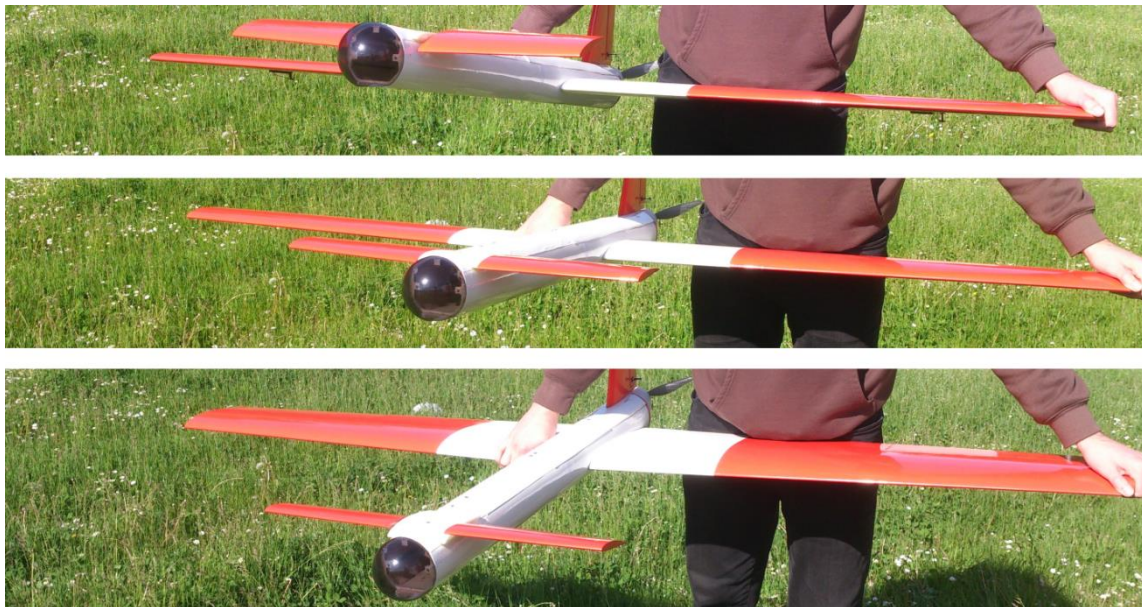
Veškeré součástky byly zapojeny dle následujícího schématu.



Obrázek 54 - Schéma zapojení

Ověření proběhlo pouze na zemi za vypnutého motoru a nastavené referenční rychlosti 15 m/s, hodnota měřené rychlosti byla simulována na 15m/s. Nejprve byly seřízeny servomotory pro nulové pozice do nulových výchylek, poté byly nastaveny počáteční hodnoty integrátorů pro ustálený let a zkontrolovány směry výchylek [1,3].

Obrázek 55 - Testování výškovky ukazuje změnu výškovky v závislosti na sklonu letadla v řídicí jednotce autopilota, kde je zapnut LQ podélný regulátor. Další fotografie v plném rozlišení spolu s videi jsou přiloženy v příloze.



Obrázek 55 - Testování výškovky



## 11 Závěr

Hlavní cíle této práce, což byla implementace regulátoru do desky autopilota a jak se během práce ukázalo, tak i samotný výběr vhodných komponent, tedy osazení letounu elektronikou, byly splněny a odzkoušeny. Velký důraz byl kladen na optimální součástky, především pohonnou jednotku a baterie pro zajištění dostatečné množství energie, dostatečnou robustnost, kvalitu a výkon vybraných komponent, avšak někdy na úkor váhy.

V práci jsem zdokumentoval komunikaci mezi jednotlivými komponenty, byla vytvořena komunikace mezi řídicí jednotkou letounu a řídicí jednotkou motoru, díky tomu jsou k dispozici motorová data, napětí baterií, teplota a jak se ukázalo tak nestabilní funkce pro měření otáček motoru. Vytvořil jsem obousměrnou komunikaci mezi řídicí jednotkou letounu a Matlabem pro možné testování.

Popsal jsem, jak jsem vytvořil PCB pro měření rozdílů tlaků, na jehož základě jsem kalibroval Pitotovu trubici pro měření vzdušné rychlosti. Deska je připojena k řídicí jednotce autopilota, kde jsem vytvořil funkce pro filtraci měřených dat.

Implementoval jsem regulátory dodané od kolegy Nováka, které jsem rozdělil do 3x4 módů, navíc jsem naprogramoval „kalibrační“ funkci, která při změně módu dopočte integrační členy, aby nenastala neočekávaná reakce akčních ploch.

Komunikaci s Matlabem jsem upravil a synchronizoval pro možné využití i v Simulinku, díky tomu jsem byl schopen porovnat modely se softwarovými regulátory spolu s hardwarovými implementovanými v řídicí jednotce autopilota za použití metody HIL. Zjistil jsem, že převod ze spojitě navrhovaných regulátorů na diskrétní, pro kaskádní regulaci sice zvýšil překmity a dobu ustálení u úhlu stoupání, ale systém zůstal stabilní. U stranového pohybu kaskádní regulace přílišné zhoršení chování systému nenastalo, zde byli překmity mnohonásobně menší než u podélného pohybu stejně tak doba ustálení zůstala obdobná. U LQ regulace je zjištění opačné, diskretizace systému zhoršila řídicí smyčky, v některých případech dokonce znestabilnila systém. Navzdory HIL testování LQR se při reálném testování tyto jevy neprojevíly.

V práci jsem rovněž navrhl hned několik možných vylepšení pro následovníky projektu jako např. přidat kvalitní senzor otáček nebo již zmíněnou kameru, dále by bylo možné vyzkoušet stream

video a telemetrických údajů pomocí GSM<sup>19</sup> sítě, což by umožnilo kontrolu nad zařízením i při ztrátě signálu s pozemní stanicí.

---

<sup>19</sup> Globální Systém pro Mobilní komunikaci,

## 12 Přílohy

### 12.1 Vybrané funkce

#### 1. Tlumič

```
float damper (float measure, float P){  
    return (measure * P);  
}
```

#### 2. Proporcionální regulátor

```
float regP (float reference, float measure, float P){  
    return (reference - measure) * P;  
}
```

#### 3. Proporcionálně-integrální regulátor

```
float regPI(float reference, float measure, float P, float I, float* sum, float satur,  
            float init){  
    *sum = windup(*sum + TIME_LOOP * (reference - measure), satur, init);  
    return (reference - measure) * P + *sum *I;  
}
```

#### 4. Windup

```
float windup (float data, float windup, float init){  
    if(data > (windup-init)){  
        data = (windup-init);  
    }  
    else if(data < (-windup+init)){  
        (data) = (-windup+init);  
    }  
    return data;  
}
```

## 5. Airspeed

```
void airspeed(void){
    ADC_DeInit(ADC1);
    ADC_channel = 4;
    ADC_Inicializace();
    ad_in3 = ADC1_Read();
    ad_in3_filt = K*ad_in3_filt + K1*ad_in3;
    napeti=3.3*ad_in3_filt/4096*((4.38+1.94)/4.38);
    j++;
    if(j>=10) j=0;
    rychlost[j]=napeti;
    for(i=0;i<10;i++){
        rychlost_sum = rychlost_sum+rychlost[i];
    }
    tlak = ((rychlost_sum/10) + AIRSPEED_OFFSET)-5;
    if(tlak<0) analog_data.airspeed= 0;
    else analog_data.airspeed = 1.025*sqrt(2*tlak/RO) + 0.5;
}
```

## 6. Airspeed\_kalman

```
void airspeed_kalman(void){
    ADC_DeInit(ADC1);
    ADC_channel = 4;
    ADC_Inicializace();
    ad_in3 = ADC1_Read();
    P = P + Q;
    average = average + K*((3.3*(float)(ad_in3)/4096)*((4.38+1.94)/4.38)-
average);
    K = P / (P + R);
    P = (1 - K)*P;
    tlak = (average + AIRSPEED_OFFSET)-5;
    if(tlak<0) analog_data.airspeed= 0;
    else analog_data.airspeed = 1.025*sqrt(2*tlak/RO) + 0.5;
}
```

## 7. Height\_regulation

```
void height_regulation(){
    reference.pitch = windup((actual_pos.height-
reference.height)*(h_reg[0]),PITCH_SATURATION);
}
```

## 8. LQR\_longitudinal

```
void LQR_longitudinal(){

    airspeed_sum = windup(airspeed_sum + TIME_LOOP * (analog_data.airspeed -
reference.airspeed), AIRSPEED_WINDUP);

    pitch_sum = windup(pitch_sum + TIME_LOOP * windup((ins_dataset.EUL.y -
reference.pitch) * (float) D2R, PITCH_SATURATION), PITCH_WINDUP);

    actuators.elevator = -((float)(ins_dataset.EUL.y) * Kp[0] * (float) D2R +
(float)(measurement.gyrs.y) * Kp[1] * (float) D2R +
(float)(analog_data.alfa) * (float) D2R * Kp[2]+
(float)(analog_data.airspeed) * Kp[3] +
(float)(airspeed_sum) * Kp[4] +
(float)(pitch_sum) * Kp[5] +
(float)(last.elevator) * (float) D2R * Kp[6] +
(float)(last.thrust) * Kp[7]);

    actuators.elevator = windup( actuators.elevator ,ELEVATOR_SATURATION);

    actuators.thrust = -(ins_dataset.EUL.y * D2R * Kp[8] + // úhel stoupání
measurement.gyrs.y * D2R * Kp[9] + // rychlost úhlu stoupání
analog_data.alfa * D2R * Kp[10] +
analog_data.airspeed * Kp[11] +
airspeed_sum * Kp[12] + //integrace rychlosti
pitch_sum * Kp[13] + //integrace uhlu klopeni
last.elevator * Kp[14] + //predchozi vychylka vyskovky
last.thrust * Kp[15]); // predchozi hodnota tahu

    actuators.thrust = windup(actuators.thrust,THRUST_SATURATION);

    actuators.thrustpercent =
(0.016*(actuators.thrust*actuators.thrust*actuators.thrust)-
0.9*(actuators.thrust*actuators.thrust)+18*actuators.thrust-25);
    actuators.rpm = sqrt(actuators.thrust/(RO*ct*D*D*D*D));
    last.elevator = actuators.elevator;
    last.thrust = actuators.thrust;

}
```

## 9. LQR\_lateral

```
void LQR_lateral(){
    beta_sum = windup(beta_sum + D2R * (analog_data.beta), BETA_WINDUP);
```

```

actuators.ailerons = windup(
-((ins_dataset.EUL.x-reference.roll) * D2R * Ks[0] + // úhel náklonu
measurement.gyrs.x * D2R * Ks[1] + // rychlost náklonu
measurement.gyrs.z * D2R * Ks[2] + // rychlost zmeny kursu
analog_data.beta * D2R * Ks[3] + // úhel vybočení
beta_sum * Ks[4] + // integrace vybočení
last.ailerons * Ks[5] + // predchozí hodnota kridélek
last.rudder * Ks[6]) // predchozí hodnota kormidla
,AILERON_SATURATION);

```

```

actuators.rudder = windup(
-((ins_dataset.EUL.x-reference.roll) * D2R * Ks[7] +
measurement.gyrs.x * D2R * Ks[8] +
measurement.gyrs.z * D2R * Ks[9] +
analog_data.beta * D2R * Ks[10] +
beta_sum * Ks[11] +
last.ailerons * Ks[12] +
last.rudder * Ks[13])
,RUDDER_SATURATION);

```

```

last.ailerons = actuators.ailerons;
last.rudder = actuators.rudder;

```

```

}

```

## 10. Yaw\_regulation

```

void yaw_regulation(){
    reference.roll = windup(
        (actual_pos.heading * D2R - reference.heading)*(yaw_reg[0])
        ,ROLL_SATURATION,0);
}

```

## 11. GPS regulation

```

void gps_regulation(){
    reference.heading = atan2(actual_pos.lon-reference.longitude,actual_pos.lat-
reference.latitude);
}

```

## 12. Kalibrace – dopočet integrační složky

```
//LQR podélný pohyb
a = ((float)(ins_dataset.EUL.y) * Kp[0] * (float) D2R +
      (float)(measurement.gyrs.y) * Kp[1] * (float) D2R +
      (float)(analog_data.alfa) * (float) D2R * Kp[2]+
      (float)(analog_data.airspeed) * Kp[3] +
      (float)(last.elevator) * (float) D2R * Kp[6] +
      (float)(last.thrust) * Kp[7]);

b = ((float)ins_dataset.EUL.y * D2R * Kp[8] +
      (float)measurement.gyrs.y * D2R * Kp[9] +
      (float)analog_data.alfa * D2R * Kp[10] +
      (float)analog_data.airspeed * Kp[11] +
      (float)last.elevator * Kp[14] +
      (float)last.thrust * Kp[15]);

airspeed_sum = windup(-(Kp[13]*(a+actuators.elevator) -
      Kp[5]*(actuators.thrust+b))/(Kp[4]*Kp[13]-
      Kp[5]*Kp[12]),AIRSPEED_WINDUP,0);
pitch_sum = windup(((a+actuators.elevator)*Kp[12]-
      (b+actuators.thrust)*Kp[4])/(Kp[4]*Kp[13]-
      Kp[5]*Kp[12]),PITCH_WINDUP,PITCH_INITIAL_INT);

//LQR stranový pohyb
c =
(float) (ins_dataset.EUL.x-reference.roll) * D2R * Ks[0] + // úhel náklonu
(float)measurement.gyrs.x * D2R * Ks[1] + // rychlost náklonu
(float)measurement.gyrs.z * D2R * Ks[2] + // rychlost zmeny kurzu
(float)analog_data.beta * D2R * Ks[3] + // úhel skluzu
(float)last.ailerons * Ks[5] + // predchozí hodnota kridélek
(float)last.rudder * Ks[6];

beta_sum = windup(-(actuators.ailerons + c)/Ks[4],BETA_WINDUP,0);
```

## 12.2 CD

1. Celý projekt z programu keil je přiložen pod názvem UAV\_Autopilot\_ins, který obsahuje veškeré zde zmíněné soubory i knihovny potřebné pro kompilaci.
2. Fotografie v plném rozlišení a video ukazující funkčnost regulace křidélek, výškovky a kormidla jsou uloženy ve složce Camera.
3. Ve složce Matlab je simulinkové schéma komunikace s hardwarem.
4. HIL složka obsahuje projekt pro testování hardwaru pomocí Matlabu.



## 13 Citace

1. NOVÁK, Josef. Sestavení matematického modelu a návrh řídicích algoritmů pro UAV letoun. Praha, 2014. Diplomová práce. ČVUT.
2. HALGAŠÍK, Jaroslav. Flight Control System Unit for Small UAV Aircraft. Praha, 2014. Diplomová práce. ČVUT.
3. ADÁMEK, Petr. Návrh bezpilotního prostředku. Praha, 2014. Diplomová práce. ČVUT.
4. Aerospace toolbox. [online]. [cit. 2014-04-10]. Dostupné z: <http://www.mathworks.com/products/aerotb/>
5. Plošný spoj. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2013 [cit. 2014-05-03]. Dostupné z: [http://cs.wikipedia.org/wiki/Plo%C5%A1n%C3%BD\\_spoj](http://cs.wikipedia.org/wiki/Plo%C5%A1n%C3%BD_spoj)
6. I<sup>2</sup>C: Inter-Integrated Circuit. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2014 [cit. 2014-04-02]. Dostupné z: <http://cs.wikipedia.org/wiki/I%C2%B2C>
7. Pulzně šířková modulace. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2013 [cit. 2014-05-03]. Dostupné z: [http://cs.wikipedia.org/wiki/Pulzn%C4%9B\\_%C5%A1%C3%AD%C5%99kov%C3%A1\\_modulace](http://cs.wikipedia.org/wiki/Pulzn%C4%9B_%C5%A1%C3%AD%C5%99kov%C3%A1_modulace)
8. Global Positioning System. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2014-04-22]. Dostupné z: [http://cs.wikipedia.org/wiki/Global\\_Positioning\\_System](http://cs.wikipedia.org/wiki/Global_Positioning_System)
9. Serial Peripheral Interface. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2014 [cit. 2014-04-13]. Dostupné z: [http://cs.wikipedia.org/wiki/Serial\\_Peripheral\\_Interface](http://cs.wikipedia.org/wiki/Serial_Peripheral_Interface)
10. MEMS. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2013 [cit. 2014-05-04]. Dostupné z: <http://cs.wikipedia.org/wiki/MEMS>
11. STM32 MCU Discovery Kits. STM32 MCU Discovery Kits - STMicroelectronics [online]. 2014 [cit. 2014-05-03]. Dostupné z: <http://www.st.com/stm32discovery>
12. Keil. Keil Embedded Development Tools for ARM [online]. 2013 [cit. 2014-05-03]. Dostupné z: <https://www.keil.com/>

13. KUBELKA, Vladimír a Michal REINSTEIN. Complementary Filtering Approach to Orientation Estimation using Inertial Sensors Only. *2012 IEEE International Conference on Robotics and Automation: RiverCentre, Saint Paul, Minnesota, USA*. 2012, May 14-18.
14. ANDRÉS. Optimal Battery Capacity. April 12, 2011.
15. JETI Telemetry Protocol CZ V1.06. In: Telemetrický komunikační protokol JETI [online]. verze: 1.06, 2012 [cit. 2014-05-01]. Dostupné z: <http://www.jetimodel.com/cs/Protokol-telemetrie/>
16. PETR, Kočárník. Aerodynamika a mechanika letu: Měření nízkých rychlostí. Praha, 2012/13.
17. HS-322HB Standart Deluxe. In: HS-322HB Standart Deluxe [online]. 1999-2014 [cit. 2014-05-02]. Dostupné z: [https://www.servocity.com/html/hs-322hd\\_standard\\_deluxe.html#.U2O4I\\_I\\_t0Z](https://www.servocity.com/html/hs-322hd_standard_deluxe.html#.U2O4I_I_t0Z)
18. HS-485HB Servo. In: HS-485HB Servo [online]. 1999-2014 [cit. 2014-05-02]. Dostupné z: [https://www.servocity.com/html/hs-485hb\\_servo.html#.U2O3\\_Pl\\_t0Y](https://www.servocity.com/html/hs-485hb_servo.html#.U2O3_Pl_t0Y)
19. REINER, 2011
20. *SPIN Pro 66: User Manual* [online]. 2012 [cit. 2014-05-03]. Dostupné z: <http://www.jetimodel.com/en/show-file/125/>
21. Měření rychlosti: Kalmanův filtr. *Robotika.cz* [online]. 2005 [cit. 2014-05-09]. Dostupné z: <http://robotika.cz/guide/filtering/en>
22. PECH, Zdislav a Vratislav VĚK. Systémy řízení letu. Vyd. 1. Praha: Česká technika - nakladatelství ČVUT, 2006, 114 s. ISBN 80-010-3374-0.
23. Flight dynamics. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2014 [cit. 2014-05-05]. Dostupné z: [http://en.wikipedia.org/wiki/Flight\\_dynamics\\_\(fixed-wing\\_aircraft\)](http://en.wikipedia.org/wiki/Flight_dynamics_(fixed-wing_aircraft))
24. Spin Pro 66: JETI model s.r.o. *Spin PRO 66* [online]. 2014 [cit. 2014-05-13]. Dostupné z: <http://www.jetimodel.com/cs/katalog/Regulatory-otacek/@produkt/SPIN-Pro-66/>
25. XOAR 11x7 dřevěná vrtule EP. *Topmodel CZ* [online]. 2014 [cit. 2014-05-13]. Dostupné z: [http://www.topmodelcz.cz/index.php?&desktop\\_back=eshop&action\\_back=&id\\_back=&desktop=eshop&action=zbozi\\_detail&id=11186](http://www.topmodelcz.cz/index.php?&desktop_back=eshop&action_back=&id_back=&desktop=eshop&action=zbozi_detail&id=11186)
26. Spektrum telemetrie Air: Pitotova trubice. *RCM.cz* [online]. 2014 [cit. 2014-05-13]. Dostupné z: <http://www.rcm.cz/produkt/Spektrum-telemetrie-Air---Pitotova-trubice-i21477>

27. Přehled modelářských silových konektorů. *RC.305* [online]. 2012 [cit. 2014-05-13]. Dostupné z: <http://rc.305.cz/view.php?cisloclanku=2012030005>
28. Svorkovnice SVL 16,0. *GES-ELECTRONIC* [online]. 2014 [cit. 2014-05-13]. Dostupné z: <http://www.ges.cz/cz/svl-16-0-GES06600652.html>
29. Hardware-in-the-loop simulation. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2014 [cit. 2015-01-08]. Dostupné z: [http://en.wikipedia.org/wiki/Hardware-in-the-loop\\_simulation](http://en.wikipedia.org/wiki/Hardware-in-the-loop_simulation)
30. *Hardware In The Loop Simulator in UAV Rapid Development Life Cycle* [online]. 2007 [cit. 2015-01-08]. ISSN ICIUS2007-A006. Dostupné z: <http://arxiv.org/ftp/arxiv/papers/0804/0804.3874.pdf>