



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

AUTOMATICKÉ UDRŽOVÁNÍ LETU RC MODELU

AUTOPILOT OF RC PLANE

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

FILIP DENK

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. VOJTĚCH MRÁZEK,

BRNO 2016

Abstrakt

Práce se zabývá stabilizací letu dálkově ovládaného modelu letadla. Cílem je vytvořit systém, který se v průběhu letu sám postará o to, aby letadlo bylo schopné (bez zásahu pilota) udržovat směr letu. Systém využívá 3-osý gyroskop pro získání polohy letadla. Pro řízení byla použita jednotka Arduino Uno.

Abstract

This thesis is focused on stabilization of the radio-controlled aircraft model. The aim is to create a system that could control the direction of flight (without pilot intervention) itself. This system also includes determination of the aircraft's position using a 3-axis gyroscope. The Arduino Uno microcontroller was used as a control unit.

Klíčová slova

stabilizace, autopilot, Arduino, MPU-6050, gyroskop, IIC, RC letadlo, letectví, regulace, PID, fuzzy, PWM

Keywords

stabilization, autopilot, Arduino, MPU-6050, gyroscope, IIC, RC plane, aviation, regulation, PID, fuzzy, PWM

Citace

DENK, Filip. *Automatické udržování letu RC modelu*. Brno, 2016. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Mrázek Vojtěch.

Automatické udržování letu RC modelu

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Mrázka. Další informace mi poskytl Zdeněk Kopeček. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Filip Denk
16. května 2016

Poděkování

Děkuji vedoucímu práce Ing. Vojtěchu Mrázkovi za umožnění práce na vlastně navrhnutém tématu, odborné vedení a užitečné rady do teoretické a praktické části. Dále děkuji modeláři Zdeňku Kopečkovi za velice užitečné rady a také děkuji rodičům za podporu při studiu.

© Filip Denk, 2016.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1	Úvod	3
2	Základní parametry modelu letadla	4
2.1	Typy RC letadel	4
2.2	Materiál	6
2.3	Řídící prvky letadla	7
2.3.1	Vrtule	7
2.3.2	Vodorovná ocasní plocha	7
2.3.3	Křídélka	8
2.3.4	Svislá ocasní plocha	8
2.3.5	Klapky	8
2.4	Princip letu	8
2.5	Výběr vhodného letadla	9
2.5.1	Beta 1400	10
2.5.2	GAMA	10
3	Metody elektronického ovládání	11
3.1	RC souprava	11
3.1.1	Vysílač	11
3.1.2	Přijímač	12
3.2	Modelářské frekvence	12
3.3	Telemetrie	14
4	Elektronika v letadle	16
4.1	Schéma zapojení prvků	16
4.2	Popis jednotlivých prvků	17
4.2.1	Motor	17
4.2.2	Baterie	19
4.2.3	Regulátor	19
4.2.4	Servomotor	20
4.2.5	Výpočetní jednotka	20
4.2.6	Inerciální měřicí jednotka	22
4.3	Komunikace mezi prvky	23
4.3.1	PWM modulace (Pulse-width modulation)	23
4.3.2	AD převodník	24

5	Stabilizační systém	26
5.1	Možnosti regulace	26
5.1.1	PID regulace	26
5.1.2	Fuzzy regulace	27
5.2	Návrh systému	28
5.2.1	Funkce a použití systému	29
5.3	Implementace systému	30
5.3.1	Hardware	30
5.3.2	Software	31
5.3.3	Komplikace při implementaci	35
6	Výsledky	37
7	Závěr	40
	Literatura	41
	Přílohy	44
	Seznam příloh	45
A	Obsah CD	46

Kapitola 1

Úvod

Cílem této práce je navrhnout a vytvořit stabilizační jednotku, která za letu při aktivaci začne (bez zásahu pilota) řídit letadlo a bude se snažit udržet jeho směr. Stabilizační jednotka je důležitou součástí letounu. Využívá se v mnoha případech - např. když je samotná pilotáž letadla náročná (bojové letouny), nebo pro bezpečnost při převážení pasažérů (dopravní letadla). V oblasti radiově řízených - RC (anglicky *Radio control*) letadel - je stabilizace méně využívaná. Je to dáno tím, že u těchto modelů letadel není kladen takový důraz na bezpečnost a navíc piloti - modeláři - chtějí mít letadlo plně pod svojí kontrolou. Využití však stabilizace najde při natáčení videa z paluby letadla, nebo pro výzkumné účely.

Na začátku práce jsou vysvětleny základní pojmy, které pomohou objasnit principy ovládní letadla a jeho konstrukci. Poté následuje popis toho, jak a čím se letadlo ovládá, tzn. popis elektroniky v letadle a přenosu signálu. Po této teorii je popsán návrh stabilizačního systému, který byl dále implementován a taky otestován v reálném použití. V závěru je shrnut výsledek mého řešení.

Veškeré vlastnosti letadel, které v této práci popisují, se vztahují na RC letadla, nikoliv na letadla větších rozměrů, která jsou řízená pilotem zevnitř kabiny (např. dopravní atd.). Pravdou však je, že většina vlastností a principů fungují stejně u všech letounů.

Kapitola 2

Základní parametry modelu letadla

Tato kapitola slouží pro seznámení s různými druhy letadel a jejich konstrukcí. Všechna letadla se skládají alespoň z těchto částí: trup, křídla, kormidla a ovládací elektronika. Tyto části se však na každém letadle liší (velikost, hmotnost, umístění na letadle atd.) a právě díky těmto odlišnostem existují různé druhy letadel.

2.1 Typy RC letadel

Některé modely mohou létat v uzavřených místnostech, některé zase potřebují velký otevřený prostor (letiště). Svoji roli hraje také cena, zkušenost pilota atd. Vzhledem ke konstrukci a účelu se ale RC letadla dají rozdělit do následujících základních skupin:

- **Trénovací modely** - jsou určeny pro piloty, kteří nemají s řízením zkušenosti. Tyto letadla jsou konstrukčně navržena tak, aby byla stabilní - tzn. hornoplošník. Na výrobu těchto modelů se často používá extrudovaný polypropylén (EPP) připomínající tvrdý polystyrén, který je odolný vůči nárazům a tvrdým přistáním. Někdy se používá tlačný pohon, protože při přistání je motor s vrtulí více chráněný.



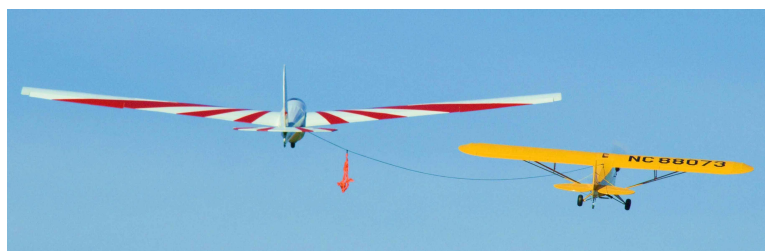
Obrázek 2.1: Cvičný model Wild Hawk [15] vyrobený z odolného EPP materiálu.

- **Park-flyery** - zde patří letadla menších rozměrů, která nevyžadují moc prostoru a je možné s nimi létat např. v hale, na zahradě či na poli. Jsou levná a konstrukčně velice jednoduchá. Tyto modely jsou určeny jak pro začátečníky, tak pro zkušenější piloty.



Obrázek 2.2: Park-flyer Aero Bird 3 [19].

- **Větroně (kluzáky)** - tyto modely se vyznačují vysokým rozpětím křídel - pohybuje se cca od 1.5m do 6m. Nejsou tak pohyblivé jako akrobatické modely, jsou však za letu velice stabilní. Větroně jsou určeny pro lety ve vysokých výškách - jsou dobře viditelné a snadno se udrží v termických stoupavých proudech. Obecně mají větroně nejdelší dobu letu (např. 1 až 2 hodiny).
 - **Bezmotorové** - velice jednoduchá forma letadla, protože kvůli absenci motoru nevyžadují tolik vybavení jako motorové větroně. S těmito modely se často startuje na svazích (anglicky se tomuto stylu létání říká *slope soaring*).
 - **Motorové** - tyto větroně nemusí (díky motoru) být házeny ze svahu, nebo startovat pomocí jiného letadla. Vrtule je většinou sklopná - při vypnutí motoru se vrtule sklopí a nevytváří tím zbytečný odpor.



Obrázek 2.3: Jeden z možných startů bezmotorového větroně vypadá následovně - letadlo s motorem vytáhne větroň pomocí lana do požadované výšky, poté se lano odpojí a větroň může plachtit (tažné letadlo mezitím přistane) [10].

- **Akrobaty** - jsou určené pro zkušené piloty, protože akrobatické modely jsou "agresivní" - velice citlivé a rychlé. Pohonné jednotky takových letadel jsou výkonnější, než je potřeba k normálnímu letu. Je to kvůli akrobatickým prvkům, ke kterým jsou tyto letadla určeny, např. let po boku, nebo "visení na vrtuli". S tím souvisí další důležitý parametr akrobatických letadel a to pevnost draku - kvůli přetížení jednotlivých částí letadla při akrobacii.



Obrázek 2.4: Model Katana. Konkrétně tento model vyrábí firma VA-models [9] z Rožnova pod Radhoštěm, kde vyřezávají z EPP materiálu celé konstrukce letadla.

2.2 Materiál

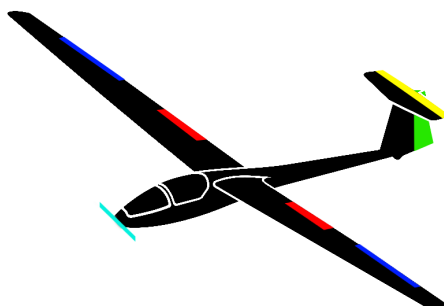
Při výběru materiálu jsou nejdůležitějšími parametry hmotnost, pevnost, pružnost, cena a náročnost zpracování. Materiály, které tyto nároky splňují (nejčastěji se používají) budou nyní stručně popsány [24]:

- **Dřevo**
 - **Balsa** - asi nejpoužívanější dřevěný materiál mezi modeláři. Je velice lehká, pevná, dostupná a dobře se s ní pracuje. Balsa je hojně využívaná jako konstrukce křídel (žebra), nebo trupu.
 - **Smrk, borovice** - pevná dřeva, která jsou však trochu těžší. Používají se nejvíce na výrobu nosníků. Pokud jsou nosníky symetrické části modelu, je třeba dát pozor na stejnou hustotu let (hmotnost).
 - **Buk, habr** - velmi tvrdá dřeva, která se hodí na silně namáhaná místa - motorové lože, nosníky, překližky. Logicky se nedají aplikovat všude, protože jsou těžká. Zejména habr se používá na výrobu vrtule.
- **Extrudovaný polypropylén (EPP)** - tento materiál připomíná běžný polystyrén. Je lehký, hodně odolný a jednoduše se s ním pracuje (oprava, vyřezávání, lepení). Je velice vhodný pro stavbu cvičných modelů - celá konstrukce letadla je vyřezána z EPP.
- **Lamináty** - tímto souhrnným názvem lze nazvat hmoty, které vznikají prosycením skelných vláken nejčastěji potystyrenovými nebo epoxydovými pryskyřicemi. V současné době si již běžní modeláři sami nevyrábí laminátové trupy, ale hotové trupy si kupují. Předností je nízká cena, nevýhodou malá pružnost. Používají se např. jako potahy silně namáhaných částí.

- **Uhlíkové vlákno** - uhlíková vlákna mají vysokou pevnost, pružnost a nízkou hmotnost. Jsou však málo odolné proti nárazu - jsou křehké. Často se používají například pro konstrukci F3K modelů (bezmotorové), které vyžadují nízkou hmotnost, nebo na zpevnění EPP konstrukce jako nosníky.

2.3 Řídící prvky letadla

Jsou to části letadla, které se za letu dokáží pohybovat (vychylovat od výchozí pozice), a tím jsou schopny měnit dráhu/rychlost letadla. Umístění jednotlivých částí na letadle lze vidět na obrázku 2.5.



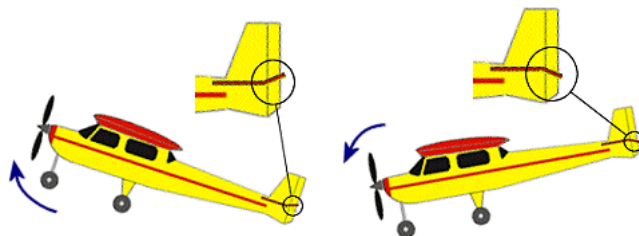
Obrázek 2.5: Barevně vyznačené řídicí prvky letadla: vrtule - světle-modrá, křídélka - tmavě-modrá, výškovka - žlutá, směrovka - zelená, klapky - červená.

2.3.1 Vrtule

Je připevněna na hřídel motoru a vytváří tah, potřebný pro pohyb letadla směrem vpřed. Jedny z nejdůležitějších vlastností vrtule jsou: počet listů, materiál, průměr, profil. Výběr vrtule úzce souvisí s výkonem motoru.

2.3.2 Vodorovná ocasní plocha

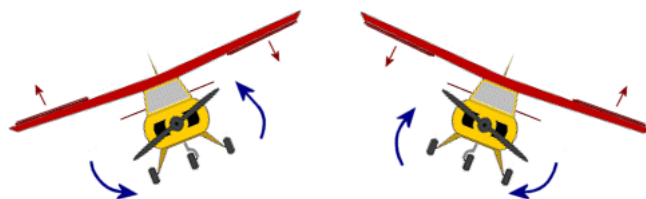
Vodorovná ocasní plocha (neboli "výškovka") se nachází na ocasu letadla a je vodorovná s křídly. Její pohyb určuje, pod jakým úhlem má model stoupat či klesat (anglicky osa *pitch*). Když se výškovka vychýlí směrem nahoru, tak se začne přední část letadla (kabina) vlivem proudění vzduchu zvedat směrem nahoru - letadlo začne stoupat - a naopak.



Obrázek 2.6: Vliv výškovky na úhel stoupání/klesání letadla. Obrázek převzat z [7].

2.3.3 Křídélka

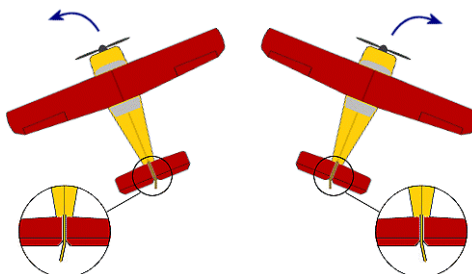
Křídélka jsou umístěna na odtokové hraně křídel. Pomocí křidélek můžeme model natáčet libovolně vzhledem k podélné ose (anglicky osa *roll*). Když chceme letadlo naklonit na nějakou stranu, tak se jedno křídélko musí vychýlit směrem nahoru a druhé dolů - to proto, že křídélko, které se vychýlí směrem dolů generuje více vztlaku na křídle a naopak. Pokud jsou na křídle také vztlakové klapky, tak jsou křídélka umístěny vždy dál (směrem od trupu) než klapky a to kvůli většímu vlivu na natočení.



Obrázek 2.7: Vliv křidélek na klonění letadla. Obrázek převzat z [7].

2.3.4 Svislá ocasní plocha

Svislá ocasní plocha (neboli "směrovka") se stejně jako výškovka nachází na ocasu letadla a je na ni kolmo upevněna. Jejím pohybem doprava nebo doleva určuje, jak se bude letadlo pohybovat po svislé ose (anglicky osa *yaw*).



Obrázek 2.8: Vliv směrovky na zatáčení letadla. Obrázek převzat z [7].

2.3.5 Klapky

Nachází se na odtokové hraně křídel, avšak blíže trupu, než křídélka. Nejčastěji se klapky využívají k brždění (snížení pádové rychlosti) nebo např. k mixu s výškovkou. Když se před přistáním klapky vyklopí dolů, umožní letadlu letět (mít dostatek vztlaku) i při nižších rychlostech. Narozdíl od křidélek, se klapky pohybují obě dvě buď nahoru nebo obě dolů.

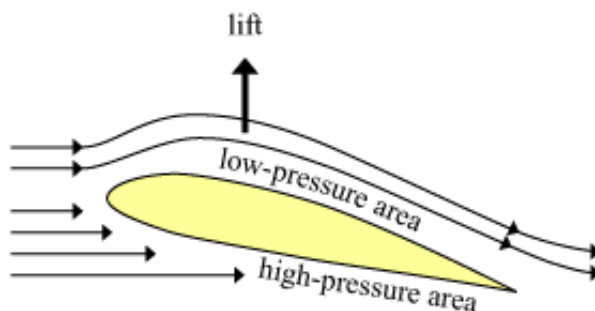
2.4 Princip letu

Aby letadlo letělo ustáleným letem (tedy rovně, bez klesání, stoupání či změny rychlosti), musí být v rovnováze několik sil. Vztlak, který směřuje nahoru (drží letadlo ve vzduchu),

musí být v rovnováze se silou tíhy letadla, která směřuje dolů. Zároveň ale tah motoru vyvolovaný vrtulí, který směřuje dopředu, musí být v rovnováze s odporem, které způsobuje tvar letadla a tření vzduchu o jeho nerovný povrch a řada dalších věcí. Z toho vyplývá, že je velice důležité, jaký profil má křídlo (na kterém vzniká vztlak) [21].

Princip vzniku vztlaku na křídlech je znázorněn na obrázku 2.9: na něm lze vidět proudnice vzduchu, které obtékají křídlo. Částice vzduchu, které proudí v tomto případě zleva doprava, narážejí na náběžnou hranu křídla. Část vzduchu proudí tedy pod křídlem, část nad křídlem. Jelikož se musí částice vzduchu na konci křídla (na odtokové hraně) opět setkat, tak musí obtéct křídlo za stejný čas. Dráha, kterou ale musí 'uletět' částice nad křídlem, je delší, než dráha pod křídlem. Logicky si tedy dokážeme odvodit, že vzduch nad křídlem proudí rychleji, než pod křídlem. A právě díky tomu vzniká nad křídlem podtlak a pod křídlem přetlak. Oba tyto jevy se podílí na síle, která křídla zdvihá směrem nahoru (vztlak) zhruba v poměru 1:3 (přetlak:podtlak) [28].

Vztlak je možné za letu zvýšit a to zvětšením prohnutí profilu nebo zvětšením plochy křídla. Jak je to za letu možné na letadle s pevnou nosnou plochou? Slouží k tomu prostředky ke zvýšení vztlaku, což jsou na většině letadel klapky - plošky umístěné na zadní straně křídla u jeho kořene, tedy blíž k trupu. Vysunutím nebo odklopením klapek se zvětší zakřivení křídla a u některých typů klapky také nosná plocha. Používají se při přistání, kdy je dosažení co nejmenší rychlosti, kdy je letadlo ještě schopné letět, žádoucí a také při vzletu, kdy je zase dobré mít vztlak co nejdříve [28].



Obrázek 2.9: Obtékání částic vzduchu kolem křídla. Pod křídlem je přetlak a nad křídlem podtlak - šipka směřující nahoru znázorňuje vztlak. Obrázek převzat z [3].

2.5 Výběr vhodného letadla

Model, který bude použit pro práci by měl splňovat následující požadavky:

- **Odolnost** - při experimentování je potřeba počítat se selháním stabilizačního systému a proto by následující oprava škod měla být jednoduchá.
- **Dosatek výkonu** - pro letadlo by neměl být problém "unést" navíc elektroniku, která bude při práci potřeba. Jsou to zejména:

- Stabilizační jednotka cca 40 g
 - Gyroskop cca 15 g
 - kabeláž cca 10 g
- **Cena** - limit jsem stanovil na 5000 Kč - cena zahrnuje celý drak modelu, motor, regulátor, servomotory a akumulátor. Není zahrnuta RC souprava.

Jako vhodné modely, které splňují tyto požadavky jsem vybral model Beta 1400 a GAMA 2100 Sport (dále jen "GAMA"). Podrobnější specifikace obou modelů jsou shrnuty v následující tabulce, výhody a nevýhody jsou pak popsány zvlášť.

Tabulka 2.1: Porovnání modelů.

	Beta 1400 [26]	GAMA [27]
Rozpětí [cm]	142	210
Délka [cm]	94	123
Hmotnost (letová) [g]	650-720	1300-1400
Plocha křídla [dm ²]	24.5	37.3
Motor	CD2622 - 1900KV	C2835 - 1000KV
Regulátor [A]	20	25
Akumulátor	11.1 V, 1300mAh	11.1 V, 2200mAh
Vrtule	5x4 plastová	10x7 sklopná, plastová
Serva	4x9g (V, S, Kř.)	2x11g (V, S) + 4x9g (Kl., Kř.)
Cena [CZK]	3240	4440
Materiál	EPP	EPP

2.5.1 Beta 1400

Beta je cvičný model. Díky motoru, který je uložený na pylonu za křídlem, snese i tvrdší přistání. Ovládat se dají směrovka, výškovka a křidélka. Model je stabilní (hornoplošník) a jednoduchý na ovládání. Beta je často používána na pořizování videí, protože motor nevadí ve výhledu.

2.5.2 GAMA

Toto letadlo se konstrukcí řadí mezi sportovní větroně, který je určený pro začátečníky. Narozdíl od Bety má navíc ovládané vztlakové klapky na křídlech. Z toho důvodu je potřeba myslet na to, že RC sestava by měla mít navíc alespoň jeden kanál. S tímto modelem je možné provádět základní akrobatické prvky.

Parametry obou letadel jsou srovnatelné. Vybraný model je nakonec GAMA a to díky volnému prostoru v kabině, který umožní jednodušeji umístit stabilizační jednotku a další elektroniku do kabiny letadla.

Kapitola 3

Metody elektronického ovládání

V této kapitole jsou popsána zařízení, pomocí kterých se standardně ovládají RC modely. Stručně jsou popsány jejich vlastnosti a taky frekvenční pásma, ve kterých komunikují.

3.1 RC souprava

Pro elektronické ovládání letadla se používá RC souprava. Tato souprava je tvořená vysílačem a přijímačem. RC set nemusí být nutně od jednoho výrobce. Je možné vysílač a přijímač různě kombinovat, ale je potřeba zajistit, aby tyto dvě zařízení byla kompatibilní - např. způsob kódování signálu, řízení přepínání kanálů atd.

3.1.1 Vysílač

Je zařízení, které obsahuje kniply a přepínače na ovládání letadla. Pohyb jednotlivých kniplů a přepínačů vysílač přemění na vysokofrekvenční signál, jenž prostřednictvím antény vyzařuje do okolí. Vysílače se dnes rozdělují podle mnoha kritérií, jako například:

- **Podle pásma** - 2.4GHz , 35MHz a další (více v sekci *Modelářské frekvence* 3.2).
- **Podle počtu kanálů** - základ jsou 4 kanály - motor, výškovka, směrovka, křídélka (2 serva jsou spojeny do jednoho kabelu - nevýhoda trimování). Jsou však vysílače, které mají kanálů mnohem více např. 9, 12, atd.
- **Podle způsobu držení vysílače** - palcové a pultové.

Vysílač použitý k této práci (obr. 3.1) [12] je 6-ti kanálový a pracuje v pásmu 2.4GHz . Obsahuje dva hlavní řídicí kniply, které jsou 2-osé. Na jednom se ovládá plyn a křídélka, na druhém výškovka a směrovka. Toto řízení se nazývá proporciální. Naopak neproporciální prvky představují dva 2-polohové přepínače. Také je na ovladači jeden 3-polohový přepínač, který se používá např. pro přepnutí módu na přistání, nebo létání v termice. Další vlastnosti vysílače:

- **Trimování** (subtrim)
- **EPA** - nastavení rozmezí pohybu pro každé servo - pomáhá např. proti přemáhání serva v koncových polohách.
- **Exponenciální výchylky**

- **Dual rate** - dvojité výchylky.
- **Paměť** na 10 modelů (akrobat, větroň, helikoptéra).
- **Mixy** klapky-výškovka, směrovka-křídélka, brždění apod.
- **Ovládací mód** - určuje, co se bude jakým kniplem ovládat - např. MÓD 1 - na pravém kniplu se ovládá plyn + křídélka, na levém směrovka + výškovka.



Obrázek 3.1: Vysílač použitý pro bakalářskou práci - Hitec Optic 6 Sport. Obrázek převzat z [12].

3.1.2 Příjímač

Příjímač přijímá prostřednictvím antény signál vyzařovaný vysílačem a přemění jej na digitální impulsy sloužící pro ovládání jednotlivých serv, elektronického regulátoru otáček motoru atd. Příjímač je pochopitelně umístěn na palubě letadla.

Konkrétně přijímač Hitec Optima 6 [13], který je použit pro bakalářskou práci, se dodává v setu s vysílačem Hitec Optic 6 Sport - je s ním kompatibilní. Má rovněž 6 kanálů. Pro každý kanál jsou vyhrazeny 3 piny - plus, minus a signál PWM (více v sekci *PWM modulace* 4.3.1).

3.2 Modelářské frekvence

Každý vysílač generuje rádiový signál, do kterého se za pomoci modulace zakóduje stav ovládacích prvků na ovladači. Frekvence tohoto signálu je pevně daná a během provozu se nemění (s výjimkou 2.4GHz). Aby bylo možné model ovládat, musí být nastaven vysílač i přijímač na shodnou frekvenci.



Obrázek 3.2: Přijímač Hitec Optima 6. Obrázek převzat z [13].

V České republice jsou ze zákona [20] pro ovládání RC modelů letadel povolena tyto frekvenční pásma: 13MHz , 27MHz , 35MHz , 40MHz a 2.4GHz . Ne všechna pásma jsou přímo určená pro modely letadel - např. frekvence 40MHz je určena pro modely letadel, aut i lodí. Mezi nejpoužívanější patří 2.4GHz a 35MHz (kvůli spolehlivosti se postupně přechází na 2.4GHz). Naopak k nejméně používaným patří 13MHz , 27MHz a 40MHz . Zde je popis zmíněných frekvenčních pásem a také vysvětlení pojmu *krystal*:

- **27MHz** - toto pásmo používají staré soupravy a je možné ovládat s nimi všechny druhy modelů (letadel, aut i lodí). Nevýhodou je možnost rušení, protože toto frekvenční pásmo využívají také občanské radiostanice, či hlasové vysílačky [35].
- **35MHz** - stále používané frekvenční pásmo. Má tu výhodu, že je určené výhradně pro ovládání modelů letadel.
- **40MHz** - pásmo, které je povoleno pro všechny druhy modelů (letadla, auta, lodě). Lze se také setkat s RC sety, které pracují na frekvenci 41MHz - u nás v ČR však tato frekvence není ze zákona povolena [35].
- **Krystal** - Aby mohlo létat několik modelů současně, musí vysílače vysílat na různých kmitočtech. V přiděleném kmitočtovém pásmu je určena řada kmitočtů a těm se říká kanály. Kanály jsou očíslovány pořadovými čísly tak, aby se mohli modeláři mezi sebou snadno dohodnout, na jakém kmitočtu jejich rádio pracuje. Jednotlivým kmitočtům se říká kanály. Rozdělení kmitočtů je platné celosvětově. Např. kanál 70 má přiřazenou frekvenci 35.100MHz a kanál 75 má frekvenci 35.150MHz .

Kmitočet rádia zajišťuje krystal ve vysílači a v přijímači. Změnu kmitočtu - kanálu - zajistí výměna krystalu ve vysílači a přijímači za jiný pár krystalů. Přijímače jsou schopny pracovat jen v jednom kmitočtovém pásmu. Není tedy možné dát do přijímače pro 27MHz krystal pro pásmo 35MHz . Stejně tak u vysílače. Krystal se používá jen u zařízení, která pracují ve frekvenčním pásmu řádu MHz [33].

- **2.4GHz** - je to volné - bezlicenční - pásmo, které je v dnešní době stále používanější. Mimo řízení modelů se toto pásmo využívá např. pro bezdrátový přenos dat (Wi-Fi). Pro vytvoření spojení mezi vysílačem a přijímačem se v tomto případě nepoužívají pevně laděné krystaly, ale stačí provést párování obou dvou zařízení (způsob závisí na výrobci).

Pro přenos dat mezi vysílačem a přijímačem se používá modulační metoda FHSS

(Frequency Hopping Spread Spectrum). Její princip je následující: při přenosu dat dochází k rychlým pseudonáhodným změnám frekvence podle určitého algoritmu, který zná jak vysílač, tak přijímač. FHSS je definováno ve frekvenčním pásmu 2.4 GHz , přesněji od 2.402 GHz do 2.480 GHz . Při přenosu se celá tato šířka pásma rozdělí po 1 MHz do 79 frekvencí, mezi kterými se přepíná [25].

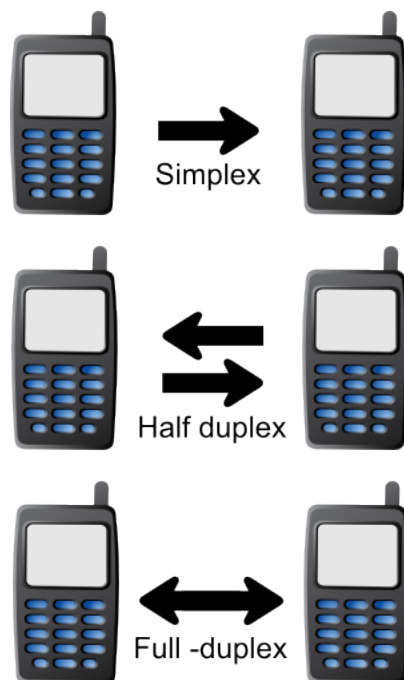
Největší výhody tohoto přenosu jsou:

- Spolehlivost
- Odolnost vůči rušení
- Absence krystalů
- Kratší anténa na vysílači i přijímači

3.3 Telemetrie

V některých případech je užitečné, když má pilot za letu k dispozici podrobnější informace o stavu letadla. K tomu se používá telemetrie. Je několik způsobů, jak se data z paluby letadla dostanou k pilotovi na zem. Princip je takový, že sensory v letadle naměří veličinu a tato hodnota se posílá vysílači. Logicky tedy vyplývá, že vysílač dokáže vysílat i přijímat signál a naopak (anglicky se takovýmto zařízením říká *transceiver*).

Podle směru komunikace mezi vysílačem a přijímačem se telemetrie klasifikuje jako full-duplex komunikace. Znamená to, že vysílač i přijímač dokáží najednou vysílat i přijímat signál. Dalšími druhy komunikace jsou half-duplex a simplex.

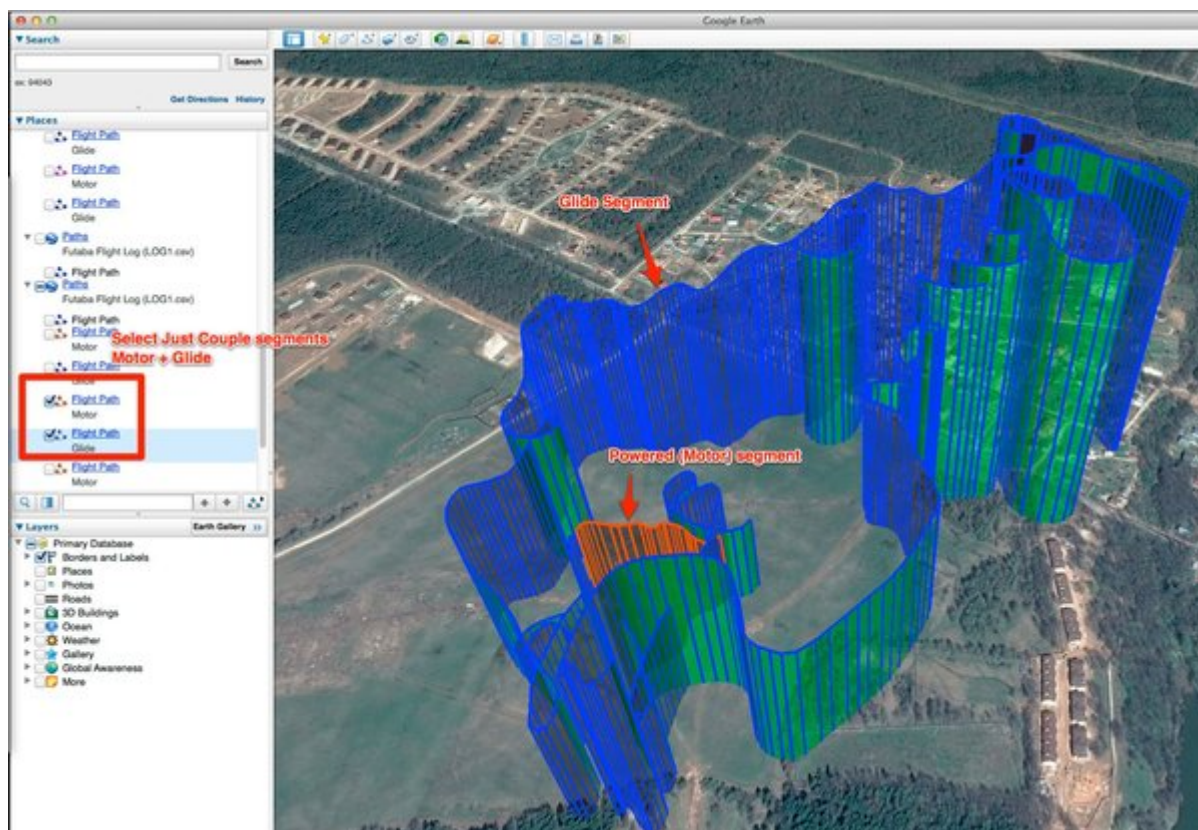


Obrázek 3.3: Tři druhy přenosu rozdělené podle směru komunikace. Obrázek převzat z [4].

Telemetrií se dají sledovat například tyto veličiny:

- Napětí pohonného akumulátoru
- Stoupání/klesání (variometr)
- Výška
- Rychlost (pitotovou trubicí)
- Otáčky spalovacích i elektromotorů
- Zrychlení

Naměřené hodnoty si pilot může zobrazit rovnou na přijímači nebo na přídatném displeji. Taky je možné informace ukládat do paměti přijímače a později si je nechat zobrazit softwarovými nástroji. Konkrétně u tohoto obrázku (3.4) je vidět kombinace GPS, odběru motoru a výškoměru. Čím oranžovější je zbarvení, tím větší proudový odběr měl elektromotor.



Obrázek 3.4: Trasa zaznamenaná pomocí telemetrie. Obrázek převzat z [8].

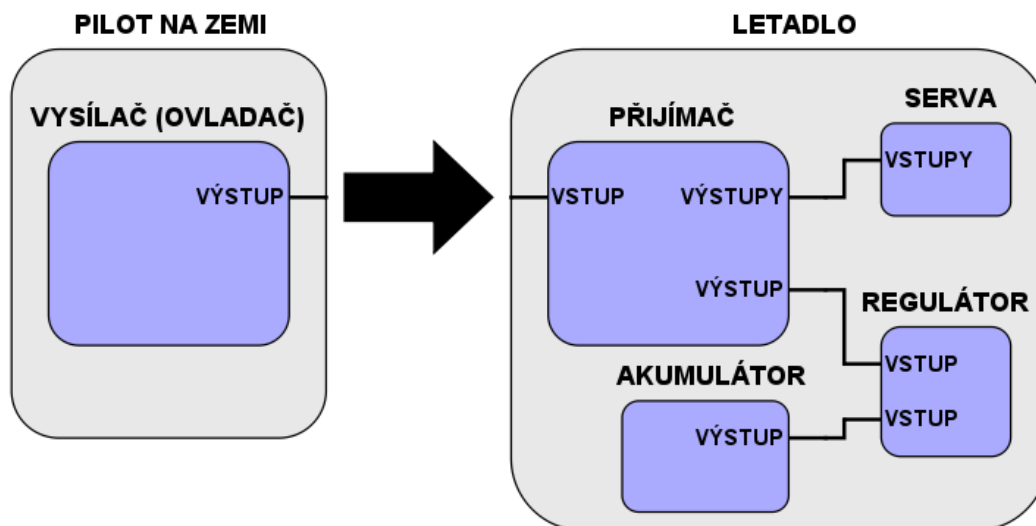
Kapitola 4

Elektronika v letadle

Kapitola popisuje elektrotechnická zařízení, která jsou nutná pro let a stabilizaci letadla. Existuje velké množství různých zařízení, která se v letadlech používají jako např. telemetrie nebo video systémy, ale to je už nad rámec této práce. Na konci kapitoly je popsán druh modulace signálu, který se používá pro komunikaci mezi elektrotechnickými prvky v letadle.

4.1 Schéma zapojení prvků

Na obrázku číslo (4.1) je zobrazené klasické zapojení prvků v letadle. Vysílač resp. ovladač vysílá neustále signál. V letadle je přijímač (popsaný již v sekci *RC set (3.1.2)*), který tyto signály zachytává a dále je rozesílá např. do jednotlivých servomotorů nebo do regulátoru motoru. Pro zjednodušení jsou na schématu serva vyobrazena jako jeden prvek. Reálně je jich více např. 6.

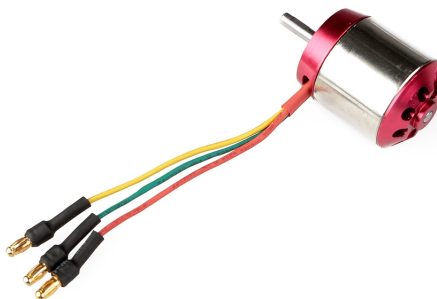


Obrázek 4.1: Standardní způsob zapojení prvků.

4.2 Popis jednotlivých prvků

4.2.1 Motor

Motor je jedna z nejdůležitějších částí letadla (když pomineme bezmotorové létání), protože vytváří tah, potřebný pro generování vztlaku. V této kapitole je uveden rozdíl mezi dvěma základními typy pohonných jednotek modelů - elektromotoru a spalovacího motoru. Zaměříme se zejména na jejich parametry a na způsob regulace.



Obrázek 4.2: Elektromotor C2835 použitý v letadle GAMA. Obrázek převzat z [5].

Motory v ovládacích letadlech se dělí hlavně na:

- **Elektromotory**
- **Spalovací motory**

Spalovací motor

Spalovací motory nacházejí využití hlavně v letadlech, které potřebují dostatek výkonu - akrobatická letadla, těžká letadla apod. Spalovací motor se liší od elektromotoru hlavně tím, že není poháněn akumulátorem, ale spaluje kapalinu (např. směs metylalkoholu a ricinového oleje). Jednoduché rozdělení spalovacích motorů může vypadat následovně:

- **Benzinové**
 - **Jedno-válcové**
 - **Více-válcové**
- **Žhavicí**
 - **Dvou-taktové**
 - **Čtyř-taktové**

Výkon spalovacího motoru řídí karburátor. Ten je součástí motoru, v níž dochází ke smíšení paliva a vzduchu. Samotnou regulaci výkonu motoru řídí servo, které podle úhlu natočení zvětšuje/zmenšuje přísun vzduchu do karburátoru. Když tedy pilot na vysílači stahuje plyn

dolů, tak servo uzavírá otvor karburátoru a naopak [32].

Často je žádoucí, aby motor - při stáhnutém plynu na minimum - dokázal běžet na volnoběh a nevypl se. Ovládání servem pracuje tak, že i při snížení plynu se přívod vzduchu neuzavře úplně. Díky tomu motor pracuje na volnoběh. Když pak bude pilot chtít motor vypnout, tak buď pomocí tlačítka (anglicky *cut-off*), nebo stáhne trim (s plynem) na minimum.

Hlavní parametry spalovacího motoru jsou:

- **Obsah (objem)**
- **Počet válců**
- **Spotřeba**
- **Druh paliva**
- **Hmotnost**

Výhody spalovacího motoru jsou hlavně výkon a fakt, že při spálení paliva v nádrži stačí kapalinu dotankovat a letadlo může znovu létat. Nevýhody jsou naopak údržba, hlasitý provoz motoru a také hmotnost (která je vyšší než u elektromotorů).

Elektromotor

Elektromotor je využíván u všech druhů modelů letadel. Hlavní rozdíl oproti spalovacímu motoru je, že elektromotor nespaluje žádné palivo - je poháněn akumulátorem. Regulaci výkonu provádí regulátor (4.2.3), který je zapojený mezi motorem a přijímačem. Elektromotory se dělí na [29]:

- **Kartáčové (DC)** - regulace je postavena na bázi regulace napětí. DC motory ztrácí výkon a rychleji se opotřebovávají, protože uhlíkové kartáče se obroušují.
- **Bezkartáčové (BLDC)** - bezkartáčové mají složitější regulaci, protože je nutné řešit frekvenci přepínání.

Hlavní parametry elektromotoru BLDC jsou:

- **Výkon** - vyjádřený ve Watech. Je to jeden z hlavních parametrů, podle kterého se člověk musí při výběru motoru řídit, protože je potřeba, aby motor utáhl letovou hmotnost letadla (včetně baterie atd.). Podle následující tabulky se dá zjistit, jak výkonný motor je potřeba, když víme, jaká je letová hmotnost letounu:

Tabulka 4.1: Výkon na Kg pro jednotlivé typy letadel [30]

Historické modely a pomalolety bez akrobatických ambicí	100-140 W/kg
Trenéry, větroně a cvičné hornoplošníky	140-200 W/kg
Sportovní modely s možností základní akrobacie	200-250 W/kg
Polomakety a makety vojenských letadel (warbirdy)	250-300 W/kg
3D akrobaty	300-400 W/kg a víc

- **Otáčky na volt** - tato vlastnost je značena zkratkou *KV*, neznamená to však kilo-Volt, ale počet otáček za minutu na 1 *V*.
- **Špičkový odběr** [*A*] - maximální proudový odběr motoru, úzce souvisí s výběrem regulátoru.
- **Hmotnost**
- **Rozměry**

Výhody elektromotoru jsou jednoznačně nízká cena a žádná údržba. Dalšími výhodami může být tichý provoz či nízká hmotnost. Nevýhoda je naopak potřeba dobíjení (nebo více akumulátorů) a menší výkon než u spalovacích motorů.

4.2.2 Baterie

V dnešní době převládají v modelech akumulátory typu LiPo. Existují ještě např. NiMh či NiCd, které se stále používají např. v ovladačích. Hlavní parametry LiPo akumulátorů jsou:

- **Počet článků** - značí se jako 1S, 2S, 3S atd. Od počtu článků se odvíjí napětí - provozní napětí jednoho článku je 3.7V. Nemělo by klesnout pod hodnotu 2.5V, protože potom už je baterie podvybitá a ne vždy se podaří "oživit"[34].
- **Kapacita** - udávaná v *mAh*. Čím větší kapacita, tím déle je letadlo schopno provozu.
- **Parametr "C"** - je to konstanta, která udává, jak velký proud je baterie schopna dodat/přijmout. Např. používá-li se akumulátor o kapacitě 2200*mAh*, s parametrem 30C, pak je schopna dodat proud:

$$2,2Ah * 30 = 66A$$

Pokud by motor odebíral větší proud, než je tato hodnota, tak hrozí až vyhoření baterie. S tímto souvisí taky regulátor, který je popsán v další sekci.

Výstupy akumulátoru tvoří zdroj stejnosměrného napětí (+ -) a balanční konektor, pomocí kterého je při nabíjení kontrolováno napětí každého článku. Některé akumulátory obsahují také ukazatel stavu nabití přímo na obalu. Stav baterie jde taky zjistit za letu pomocí telemetrie.

4.2.3 Regulátor

Často označován jako ESC (Electronic Speed Controller) - je zařízení, které přijímá signál z přijímače a podle něj řídí otáčky motoru. Je napojený jak k motoru, tak k akumulátoru a přijímači (který také napájí). Regulátor může plnit taky další funkce jako např. BEC (stabilizátor napětí) nebo brzda - dovolí sklopné vrtuli u modelu letadla zastavit se a sklopit. Hlavní parametr je proud, který dokáže regulátor přenést do motoru.

Mějme příklad: elektromotor, jenž má maximální odběr 35A, je napájen baterií, která je schopná dodat proud až 66A. Pokud je regulátor, zapojený mezi motorem a baterií, dimenzovaný na 40A, je vše v pořádku. Pokud by ale regulátor zvládl odběr stejný nebo nižší, hrozí opět vyhoření. Vždy je lepší počítat s rezervou [30].



Obrázek 4.3: Regulátor RAY R-50SB. Obrázek převzat z [14].

4.2.4 Servomotor

Servomotor (dále "servo") je velmi důležitá součást každého modelu. Je to malá elektromechanická součástka, která má v sobě motor, který je řízen se zpětnou vazbou. Má přesně definovaný rozsah. Je řízen integrovaným regulátorem, jehož vstupem je PWM signál. Typické rozměry serva jsou např.: 24mm x 11mm x 22mm (závisí na tom, jak výkonné musí servo být apod.).

Úkolem serva je hýbat s ovládací plochou letadla (2.3) podle PWM signálu z přijímače. Aby tohle bylo možné, tak musí být servo propojené s pohyblivou plochou letadla něčím, co pohyb serva přenese. Takovému prvku se říká táhlo/bowden. Např. když se na ovladači posune knipl směrovky doprava, tak vysílač odešle přijímači tento pokyn. Přijímač přepośle pokyn na odpovídající servo a to se podle signálu natočí. Vlivem tohoto se pohne směrovka, protože je se servem propojena tahýlkem.



Obrázek 4.4: Servo, které je propojené tahýlkem ke křídélku. Obrázek převzat z [18].

4.2.5 Výpočetní jednotka

Pro realizaci stabilizačního asistenta letu, je nutné použít výpočetní jednotku, která bude přijímat data z gyroskopu a na základě toho řídit serva na letadle. Důležitým požadavkem na tuto jednotku je, aby dokázala pracovat v reálném čase (anglicky *real-time*). Jasným kandidátem pro tuto úlohu byla platforma Arduino UNO, protože již s ní mám jednu dobrou

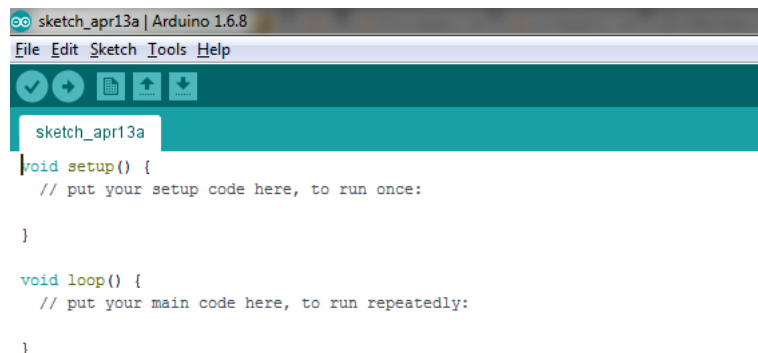
zkušenost a je to hojně podporované zařízení, které je jednoduše rozšiřitelné o různé moduly a senzory jako např. MPU-6050 (gyroskop + akcelerometr).

Arduino UNO je otevřená elektornická platforma založená na mikroprocesoru ATmega328P od firmy Atmel. Slouží jak pro začátečníky, kteří se mohou naučit základy elektroniky spojené s programováním, tak pro pokročilé, kteří dokáží pomocí této platformy vytvořit interaktivní systém, např. vlastní robot, tepelné čidlo atd. Velká výhoda platformy Arduino je, že na internetu je množství návodů, rad a tipů, jak řešit různé úlohy - rozsáhlá podpora komunity.

Tabulka 4.2: Parametry vývojového kitu Arduino UNO

Mikrokontrolér	ATmega328P
Frekvence procesoru	16MHz
Digitální I/O piny	14x (6x PWM)
Provozní napětí	5V
Flash paměť	32kB
SRAM paměť	2kB
EEPROM paměť	1kB
Délka	68.6mm
Šířka	53.4mm
Hmotnost	25g

Mikroprocesor se programuje v multiplatformním vývojovém prostředí pomocí kombinace jazyků *Wiring* a *Processing*. Vychází z jazyka C++ a obsahuje navíc např. vestavěné funkce pro práci se vstupy/výstupy, načtení hodnoty signálu apod. [1].



```
sketch_apr13a | Arduino 1.6.8
File Edit Sketch Tools Help
sketch_apr13a
void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}
```

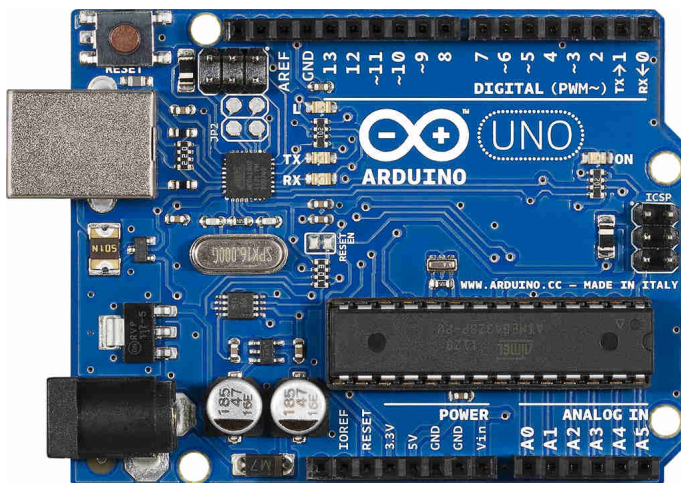
Obrázek 4.5: Vývojové prostředí Arduino.

Jak lze vidět na obrázku (4.5), tak struktura programu (který se překládá a nahrává do mikroprocesoru) se skládá ze dvou hlavních částí - funkcí:

- ***void setup()*** - tato funkce je prováděna jako první, a celkem proběhne právě 1x. Jak

napovídá jméno, funkce je určena pro nastavení. Konkrétně jde například o inicializaci registrů, proměnných, zařízení připojených k desce, nebo nastavení komunikace prostřednictvím terminálu.

- ***void loop()*** - logika programu je umístěna v této funkci. Tělo funkce je prováděno v nekonečné smyčce.



Obrázek 4.6: Arduino UNO Rev3. Obrázek převzat z [1].

4.2.6 Inerciální měřicí jednotka

Inerciální měřicí jednotka (dále "IMU") je elektronické zařízení, které využívá kombinaci akcelerometru a gyroskopu pro měření úhlové rychlosti, pozice tělesa, odstředivé síly či detekci vibrace - je to základní prvek stabilizační jednotky. Zařízení tohoto typu nachází uplatnění např. v mobilním, leteckém nebo automobilovém průmyslu. Platforma Arduino UNO nemá IMU integrovanou, ale lze tuto (kompatibilní) jednotku dokoupit - což jsem zvolil jako vhodné řešení.

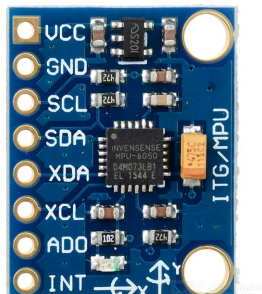
GY-521

Senzor GY-521 tvoří deska plošných spojů (anglicky *PCB*) osazená hlavní komponentou - čipem MPU-6050. Tento čip obsahuje MEMS akcelerometr a MEMS gyroskop. Je velmi přesný, protože používá 16-bitový analogově/digitální převodník pro každý kanál. Snímá osu X, Y a Z zároveň (ve stejný čas) [38].

V GY-521 je zabudovaná jednotka DMP (anglicky *Digital Motion Processor*), která provádí komplexní výpočty s daty z gyroskopu či akcelerometru. Má to tu výhodu, že Arduino jenom čte data, která senzor posílá a nemusí se tedy zdržovat dalšími výpočty. Komunikace s Arduinem je realizována pomocí sběrnice IIC rychlostí $400kHz$.

Princip komunikace je následující: senzor získává data a zpracovává je. Když má připravená data, tak je uloží na svůj zásobník (1024B FIFO zásobník) a Arduino pošle žádost o přerušování (anglicky *interrupt*), díky které Arduino pozná, že může číst data.

Protože by bylo obtížné naimplementovat celou komunikaci přes sběrnici IIC, použil jsem knihovnu *i2cdevlib* [37], jejíž autorem je *Jeff Rowberg*. Tato knihovna obsahuje implementaci rozhraní pro některá zařízení, jež komunikují přes IIC sběrnici. Mezi podporovaná zařízení patří taky GY-521.



Obrázek 4.7: Senzor GY-521. Obrázek převzat z [6].

4.3 Komunikace mezi prvky

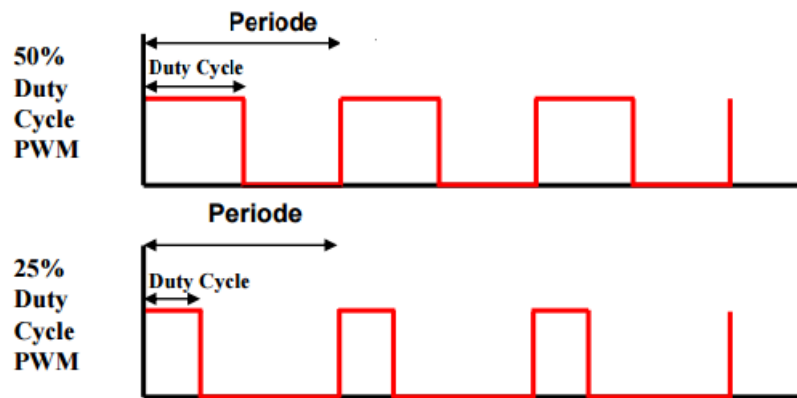
Aby mohly mezi sebou prvky v letadle komunikovat, musí rozumět signálu, který si posílají (kompatibilita). Tímto signálem je standardně PWM signál, jehož princip je vysvětlen v této sekci.

4.3.1 PWM modulace (Pulse-width modulation)

Je to modulace, používaná při přenosu signálu mezi zařízeními uvnitř RC letadla. Tento diskrétní signál se přenáší v jednotlivých pulzech, jejichž perioda je standardně $20ms$. Hodnota signálu je zakódována střídou - poměr mezi stavy 1 a 0 v jedné periodě. Šířka (délka) stavu 1 se pohybuje standardně od $1ms$ do $2ms$. Např. střední poloha serva tzn. natočení 0° , se dosáhne 50% střídou - krajní pozice 0% a 100% střída [31].

S PWM modulací pracují například tato zařízení:

- Přijímač
- Servo
- Regulátor motoru
- RC přepínače
- Osvětlení
- Stabilizační jednotky



Obrázek 4.8: Průběh PWM signálu. Vrchní - 50% střída, spodní - 25% střída. Obrázek převzat z [31]

4.3.2 AD převodník

Data, která přijímá Arduino z externích senzorů přichází ve formě analogového signálu. Pro další zpracování (např. posílání dat z Arduina do serva) těchto dat je potřeba analogový signál převést na digitální. Elektronická součástka, která je pro tento převod určená se nazývá analogově-digitální převodník (zkratka A/D, nebo anglicky *ADC*).

Arduino UNO obsahuje 6 analogových pinů (vstupů), které jsou označeny A0-A5, avšak jen jeden 10-bitový A/D převodník. Jelikož převodník dokáže zpracovávat jen jeden analogový signál najednou, je potřeba tok dat řídit pomocí multiplexoru (na schématu 4.9 je označen jako *INPUT MUX*) [2].

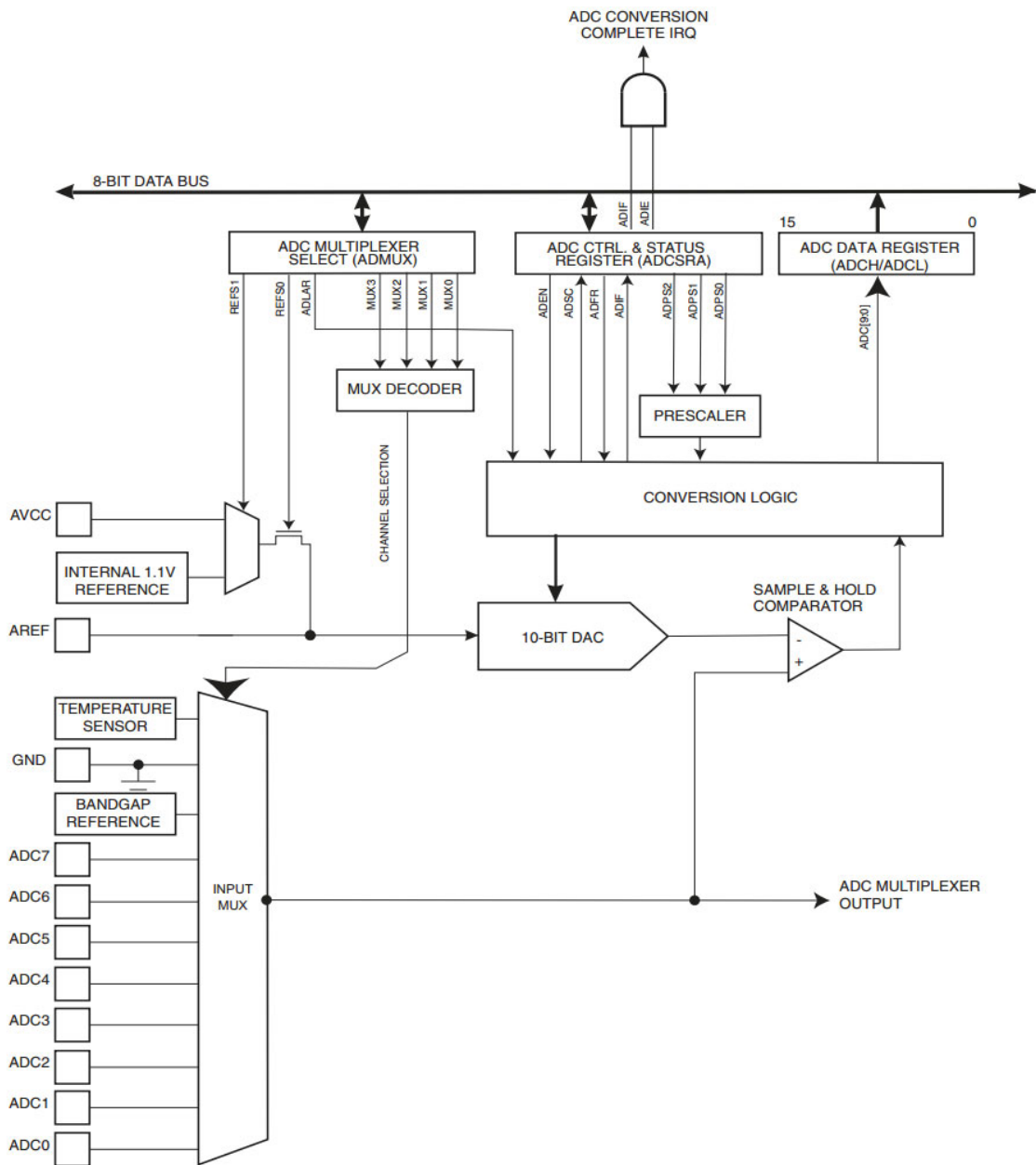
Na příkladu níže můžeme vidět, jak jednoduše lze napsat program pro opakované čtení a vypisování hodnoty z pinu A3 pomocí funkce *int analogRead()*:

```

1  int analogPin = 3;
2  int val = 0;
3
4  void setup()
5  {
6  Serial.begin(9600);
7  }
8
9  void loop()
10 {
11 val = analogRead(analogPin);
12 Serial.println(val);
13 }

```

Výsledná hodnota, ukládaná do proměnné *val* může nabývat hodnot v rozmezí 0-1023, protože převodník je 10-bitový (2^{10}). Když je potřeba vypočítat přesnost převodu (jaký rozsah měřené veličiny pokrývá 1 jednotka namapovaná do výsledné proměnné), postupuje se následovně:



Obrázek 4.9: Schéma 10-bitového A/D převodníku uvnitř mikrokontroléru ATMEGA328P. Obrázek převzat z [2].

$$(IN_h - IN_l) / 2^b = RANGE$$

- IN_h horní hranice vstupní proměnné
- IN_l dolní hranice vstupní proměnné
- b počet bitů převodníku (v případě Arduino UNO = 10)
- $RANGE$ rozsah vstupní veličiny, kterou pokrývá 1 jednotka výstupní proměnné

Kapitola 5

Stabilizační systém

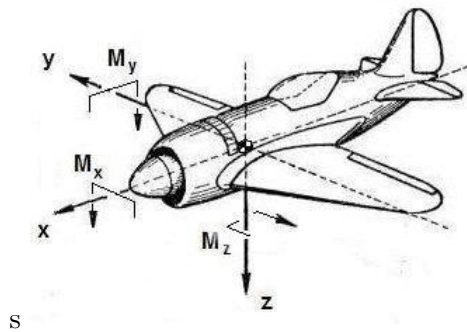
Stabilizační systém je souhrn prvků - kombinace hardware a software, které mezi sebou spolupracují a zajišťují stabilní let letadla. První část této kapitoly je věnována možným způsobům regulace, kterými jsem se zabýval. Další dvě části popisují návrh stabilizačního systému a jeho implementaci.

5.1 Možnosti regulace

Regulátor je zařízení, které ovlivňuje regulovaný systém tak, aby dosáhl požadovaného stavu. Možné způsoby regulace, které jsem zvolil jako vhodné jsou:

- PID regulátor
- Fuzzy regulátor

Veličiny, které bude potřeba regulovat v této práci jsou celkem 3 - jsou to 3 polohy letadla - poloha vzhledem k ose X, Y a Z v souřadnicové soustavě letadla (viz. obrázek 5.1).



Obrázek 5.1: Souřadnicová soustava letadla. Obrázek převzat z [16].

5.1.1 PID regulace

PID regulátor je jeden z nejčastěji používaných regulátorů. Název PID vznikl spojením tří částí, ze kterých se regulátor PID skládá - proporciální, integrační a derivační část. Každá z těchto částí ovlivňuje chování regulátoru.

Od toho se také odvíjí tři parametry, které u tohoto regulátoru lze nastavit [39]:

- **Parametr P** - proporciální regulátor násobí rozdíl mezi požadovanou a změřenou hodnotou (odchylkou) s proporciální konstantou. Jako primitivní regulaci lze použít jen P složku (složka I a D je nulová), ale nevýhoda použití jen proporciální části je, že systém se nikdy neustálí a pořád kmitá okolo požadované hodnoty.
- **Parametr I** - integrační regulátor vynásobí odchylku s integrační konstantou a přičte si ji ke své složce. Znamená to, že pokud bude změřená hodnota nižší než požadovaná, integrační složka se bude zvyšovat a naopak. Po ustálení teploty na požadované hodnotě bude integrační složka nastavená na výkon, který je třeba pro udržení ustálené teploty.
- **Parametr D** - derivační regulátor násobí rychlost změny odchylky s derivační konstantou. Čím rychleji se měřená hodnota odchyluje, tím víc bude derivační regulátor působit a naopak.

Matematicky je PID regulátor definován takto:

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{d}{dt} e(t)$$

Obrázek 5.2: Vzorec PID regulátoru. Obrázek převzat z [17].

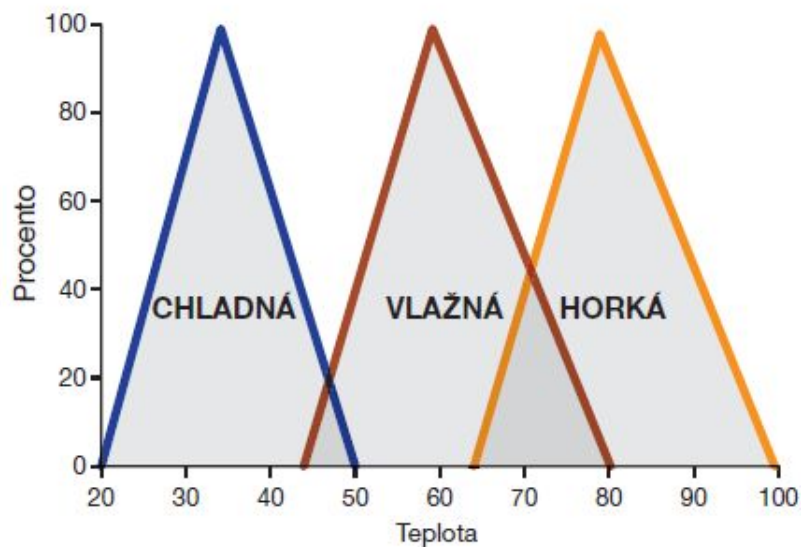
Správné nastavení těchto tří paramterů je složitá záležitost - nastavit se dají experimentálně, nebo taky pomocí metod, díky kterým je nastavení jednodušší např. Ziegler-Nicholsova metoda.

5.1.2 Fuzzy regulace

Fuzzy regulátor je méně častý způsob regulace. Je to rozšíření booleovské logiky - nevyhodnocuje se jestli prvek do dané množiny patří, ale vyhodnocuje se jak moc prvek do dané množiny patří - stupeň příslušnosti (hodnota z intervalu $\langle 0; 1 \rangle$). Používá se zejména tam, kde není znám matematický model řízené soustavy nebo je obtížně detekovatelný. Výhodnou fuzzy regulátorů je relativní jednoduchost a rychlost jejich návrhu [36].

Fuzzy regulátor se skládá ze tří hlavních výpočetních fází:

- **Fuzzifikace** - nejprve se zjistí stupně příslušnosti vstupních veličin pro jednotlivé množiny použité v pravidlech.
- **Inference** - pro každé pravidlo je vyhodnocena premisa (levá strana pravidla) jako konjunkce jednotlivých výrazů. Výsledek stanoví míru významnosti pravidla v dané situaci.
- **Defuzzifikace** - výpočet numerické hodnoty výstupu vážením výsledných funkcí příslušnosti jednotlivých pravidel.

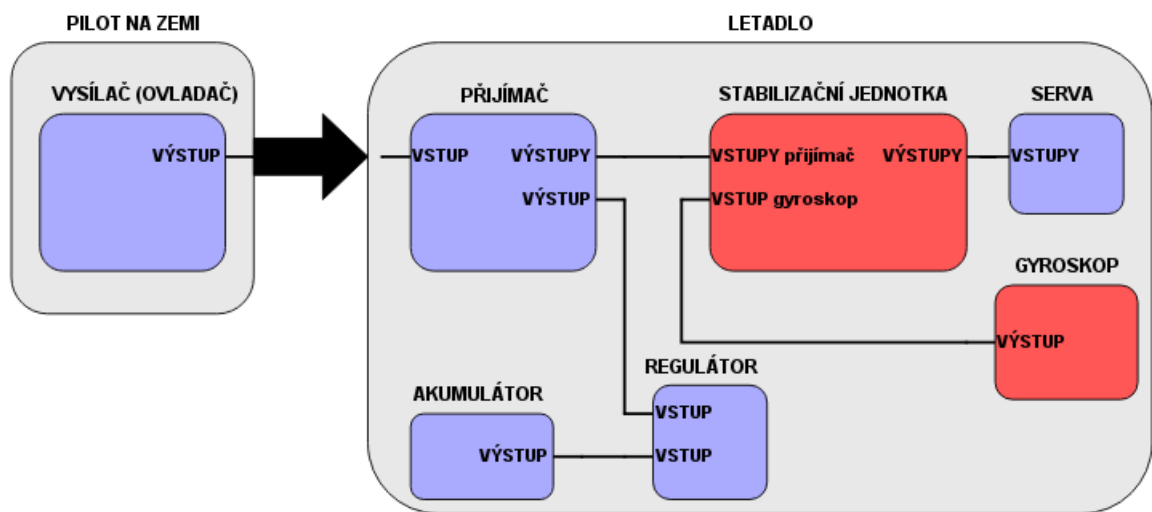


Obrázek 5.3: Příklad rozdělení fuzzy množin při regulaci teploty kapaliny. Obrázek převzat z [11].

Původně jsem se rozhodl pro účel této práce implementovat PID regulátor, ale nakonec jsem použil fuzzy regulátor - důvod volby je popsán níže v sekci *Problémy* (5.3.3).

5.2 Návrh systému

Cílem této práce je vytvořit stabilizační systém. To znamená, že musí být v letadle nějaký prvek, který bude mezi přijímačem a řídicími prvky letadla (serva a motor). Takový prvek je zvolená platforma Arduino UNO (4.2.5). Rozdíl oproti standardnímu zapojení elektroniky v letadle (4.1) je, že přijímač neposílá signál přímo k řídicím prvkům letadla, ale do Arduina, které tento signál čte a zároveň jej modifikuje podle dat z IMU (4.2.6). Nakonec Arduino posílá takto upravený - regulovaný - signál do řídicích prvků letadla. Upravené schéma je na obrázku číslo 5.4.



Obrázek 5.4: Upravené schéma, které navíc od standardního zapojení obsahuje prvky stabilizačního systému (vyznačeny červeně). Pro zjednodušení jsou na schématu serva vyobrazena jako jeden prvek.

5.2.1 Funkce a použití systému

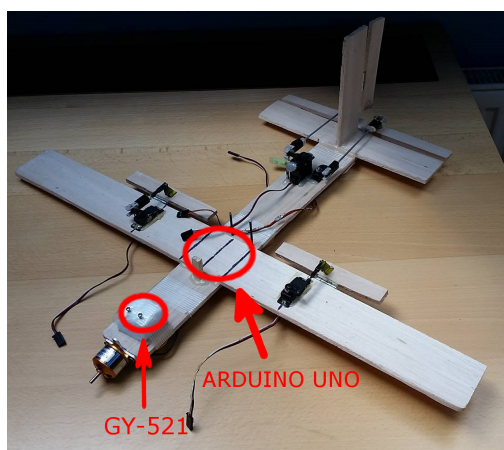
Stabilizační systém letounu jsem navrhnul jako asistenta letu - tzn. stabilizace není zaplá pořád, ale pilot si může za letu stabilizaci vypnout/zapnout - toho je dosaženo díky jednomu volnému kanálu z RC setu (celkem 6 kanálů), který je nevyužitý. Činnost stabilizační jednotky se dělí do dvou hlavních stavů, jejichž průběh je popsán následovně:

- **Stav VYPNUTO** - v tomto stavu výpočetní jednotka pouze přeposílá signál z přijímače na ovládané prvky letadla. Funguje tedy pouze jako spoj mezi přijímačem a řízenými prvky letadla.
 1. Na vstupu stabilizační jednotky jsou načítány data z přijímače.
 2. Data nejsou nikterak modifikována, ale přímo se posílají na odpovídající výstup. Toto přeposílání trvá, dokud se nepřeklopí stav kanálu, který je určený na vypnutí/zapnutí (na stav ZAPNUTO).
- **Stav ZAPNUTO** - tento stav přebírá kontrolu nad letadlem a ovládá jeho prvky tak, aby se kurz letu co nejvíce blížil kurzu letu, při kterém byla stabilizace aktivována.
 1. Při zapnutí stabilizační jednotky je pomocí dat z IMU zjištěna aktuální poloha letadla v souřadnicové soustavě letadla (osa X, Y, Z), která se uloží do paměti mikrokontroléru - toto se provede jen jednorázově při každém aktivování (zapnutí) stabilizace.
 2. Všechny signály (kanály), které vstupují do stabilizační jednotky z přijímače jsou ignorovány, až na jeden - kanál určený pro zapnutí/vypnutí stabilizace. Stav tohoto kanálu je neustále načítán a jakmile se překlopí do stavu VYPNUTO, tak výpočetní jednotka začne signál pouze přeposílat.

3. Probíhá načítání dat z IMU. Na základě těchto dat je vypočítán a poslán na výstup signál, který řídí serva letadla - probíhá regulace. Tento krok běží v nekonečné smyčce.

5.3 Implementace systému

Jako první věc, která se ukázala velice přínosná, byla stavba menšího modelu - cca 70cm x 70cm - na kterém jsem prováděl experimenty. Tento simulační model nebyl určený k letu, ale velice mi ulehčil práci (ušetřil čas) a to nejen proto, že se s ním dalo jednoduše manipulovat - naklánění ve 3 osách, ale taky byl mnohem více skladnější a přenositelnější, než letadlo GAMA (které má 3x větší rozpětí křídel).



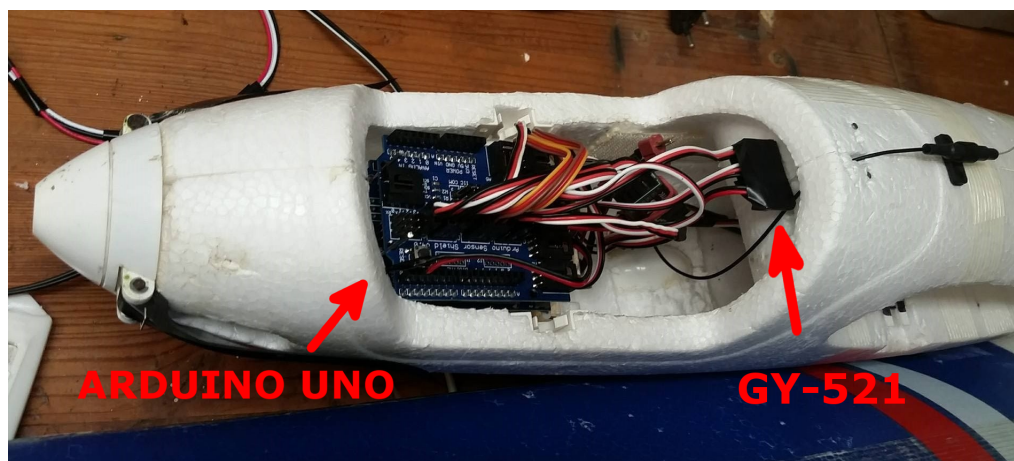
Obrázek 5.5: Model pro testování na zemi.

5.3.1 Hardware

Tato část vyžadovala správné umístění a zapojení všech elektrotechnických součástek do letadla. Jak je zmíněno v popisu GAMY, prostoru je v kokpitu dost, a proto byly provedeny jediné úpravy: vyrobení lůžka pro Arduino a vyřezání otvoru pro IMU a jeho upevnění. Konkrétní fyzické prvky, kterými jsem tedy osadil letadlo, jsou tyto:

- **Přijímač** - Hitec Optima 6 (v RC setu s vysílačem)
- **Stabilizační jednotka** - Arduino UNO
- **Serva** - více druhů, např. výšková a směrovka jsou řízeny servem Turnigy TSS-10HM
- **IMU** - GY-521
- **Akumulátor** - FOXY G2 - LC LI-POL 2200MAH/11,1V 40/80C
- **Regulátor** - RAY-50SB

Jak vypadá výsledné zapojení při pohledu na kokpit lze vidět na tomto obrázku:



Obrázek 5.6: Propojení Arduina se senzorem, přijímačem a servy v letadle.

Z důvodu nízké spolehlivosti elektroniky, kterou prodejce GAMY tento model osadil - konkrétně serva - jsem asi po 6 nalétaných hodinách musel koupit nová serva s kovovými převody.

5.3.2 Software

Pro celou rodinu Arduino produktů existuje vývojové prostředí *Arduino IDE*, ve kterém stačí vybrat druh desky, která se bude programovat. Datový kabel, který propojuje počítač a Arduino je USB 2.0 typu A-B.

Vývojové prostředí je nezávislé na operačním systému (anglicky *cross-platform*), a tak byla moje první volba programování v UNIXovém prostředí - konkrétně v distribuci *Fedora 22 x64*. Objevila se však nepříjemná chyba a to výpadky spojení mezi PC a Arduinem, v důsledku čehož bylo potřeba celé prostředí restartovat (někdy i 2x). Tuto chybu se mi nepovedlo vyřešit, a proto jsem celý vývoj přesunul na OS Microsoft Windows 7 x64, kde problém s konektivitou nebyl.

Logika programu

Implementovaný program má následující strukturu:

1. **Přidání knihoven** - standardní a externí knihovny.
2. **Deklarace proměnných** - proměnné pro jednotlivá serva, piny, vektory atd.
3. **Funkce *setup()***
 - (a) Inicializace MPU-6050
 - (b) Inicializace servomotorů
 - (c) Povolení přerušování na vybraných pinech
4. **Funkce *loop()***

- (a) Hlavní větev celého programu - podmínka kontroluje, zda je stabilizační systém zapnutý či vypnutý.
- Když je **vypnutý**, tak jsou povolena přerušení na pinech, kterými se ovládá výškovka, směrovka a křídélka (signál je jen přeposlán).
 - Když je **zapnutý**, tak jsou zakázána všechna přerušení z přijímače až na jeden, kterým se zapíná/vypíná stabilizace. Poté se čtou data z IMU, které zpracovává fuzzy regulátor. Zpracované signály jsou poslány k ovládacím prvkům letadla.

Celý implementovaný program, který je ve smyčce vykonáván stabilizační jednotkou, je dlouhý cca 450 řádků, proto popíši jen stěžejní části.

Zapnutí/vypnutí stabilizace

Detekce zapnutí nebo vypnutí stabilizace je prováděna Arduinem pomocí čtení PWM vstupu na pinu číslo 4. Tento pin je propojený s kanálem číslo 5 na přijímači. Ovladač tento kanál ovládá pomocí 2-polohového spínače. Prakticky to znamená, že když je poloha spínače na vysílači v poloze 0, tak je PWM střída 0% a když je v poloze 1, tak je PWM střída 100%. Hodnotu střídy tedy Arduino přečte a nastaví booleovskou proměnnou na *TRUE* nebo *FALSE*. Část kódu [22], která detekuje zapnutí/vypnutí:

```
1  if(bUpdateFlags & PILOT_FLAG)
2  {
3      if(unPilotInShared > LIMIT_PILOT)
4      {
5          unPilotBoolIn = true;
6      }
7      else
8      {
9          unPilotBoolIn = false;
10     }
11 }
```

Systém jsem navrhnul tak, aby při každém zapnutí jednorázově zjistil polohu letadla a tu se snažil udržet. Toto se ukázalo jako veliký problém, protože při aktivaci se načetla poloha z gyroskopu - 3 hodnoty, pro každou osu jedna - a v důsledku vibrací/poryvů větru za letu se často stávalo, že načtená hodnota pro každou osu byla úplně jiná, než jakou by pilot očekával. Prakticky to znamenalo, že když např. letadlo letělo rovně (nezatáčelo, nestoupalo ani neklesalo), tak při zapnutí stabilizace by pilot očekával, že se bude snažit letět dále rovně, ale letadlo mělo tendenci stoupat.

Snažil jsem se vyřešit tento problém načtením prvních 10 hodnot, ze kterých se vypočítal pro každou osu průměr. Výsledek byl lepší, ale i přes zvýšený počet načtených hodnot výsledek nebyl uspokojivý. Počet hodnot se však nedal zvyšovat do nekonečna kvůli časovému zpoždění. Proto jsem tento způsob zavrhl a systém funguje tak, že cílová poloha je napevno nastavená v kódu. Když jsem tuto polohu chtěl měnit, tak jsem si nechal vypisovat aktuální polohu z IMU v letadle (na zemi) a když jsem s letadlem posouval, otáčel a nastavil jej do požadované polohy, tak jsem aktuální hodnoty, které se četly z IMU, napevno přepsal v kódu [37] následovně:

```
1 mpu.setXGyroOffset(220);
2 mpu.setYGyroOffset(76);
3 mpu.setZGyroOffset(-85);
```

Čtení dat z přijímače

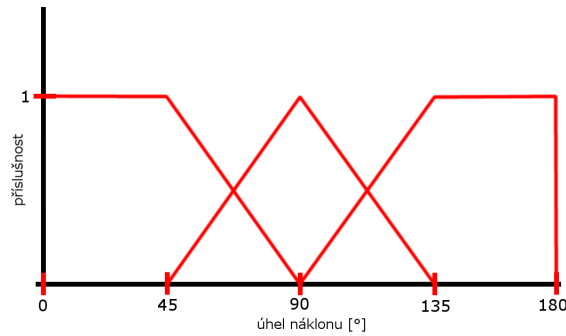
Pro realizaci čtení signálu z přijímače letadla na vstupu Arduina jsem se rozhodl využít standardní funkci na toto určenou - *unsigned long pulseIn(pin, value)*. Funkce přečte hodnotu střídy z PWM signálu a s tou pak může pracovat. Funkce vykonává přesně to co se od ní očekává, ale při čtení více signálů, se celý systém spožduje, protože tato funkce si po celou dobu načítání střídy "vyhradí" procesor pouze pro sebe. To má pak nepříjemné následky jako např. překmitávání serv, přetečení bufferu na GY-521 atd.

Řešením je zjišťovat hodnotu jednotlivých signálů pomocí přerušení. Ke každému signálu se povolí přerušení (na odpovídajícím pinu), které se generuje při každé změně PWM signálu. Cílem těchto přerušení je změřit délku hodnoty 1 v jednom PWM cyklu. To zajišťují takzvané funkce obsluhy přerušení (zkratka ISR, anglicky *Interrupt Service Routine*). Takové funkce se provedou při každé změně signálu a když je hrana signálu vzestupná, tak se do proměnné uloží čas vyvolání přerušení. Když je naopak hrana sestupná, tak se od aktuálního času odečte čas, který je uložený v proměnné z předchozího přerušení - takto se dá získat hodnota signálu PWM pomocí přerušení, což ušetří hodně času procesoru. Příklad [22] čtení hodnoty výškovky:

```
1 // povoleni preruseni
2 PCintPort::attachInterrupt(IN_ELEVATOR, calcElevator, CHANGE);
3
4 // ISR - obsluzna funkce pro nacteni hodnoty vyskovky
5 void calcElevator()
6 {
7     if(digitalRead(IN_ELEVATOR) == HIGH)
8     {
9         uLElevatorStart = micros();
10    }
11    else
12    {
13        unElevatorInShared = (uint16_t)(micros() - uLElevatorStart);
14        bUpdateFlagsShared |= ELEVATOR_FLAG;
15    }
16 }
```

Fuzzy regulace

Jednoduchý fuzzy regulátor jsem si celý naimplementoval sám, protože externí knihovny pro fuzzy byly zbytečně složité pro řešení, které jsem potřeboval. Hlavní tři požadavky jsou tyto: funkčnost, jednoduchá použitelnost, rychlost výpočtu. Rozložení fuzzy množin jsem si navrhnul následovně:



Obrázek 5.7:

Váhy bočních lichoběžníků jsou větší, než váha (koeficient) středního trojúhelníku, aby byla reakce "agresivnější" v mezních hodnotách. Na příkladu kódu uvedu, jak se počítá funkce příslušnosti (anglicky pojem *membership function*) pro lichoběžník:

```

1  if (*input >= trapezoidA[j] && *input <= trapezoidD[j]) // is in interval
2  {
3      if (*input >= trapezoidA[j] && *input < trapezoidB[j])
4      {
5          partMember[j+1] = (*input-trapezoidA[j])/(trapezoidB[j]-trapezoidA[j]);
6      }
7      else if (*input >= trapezoidB[j] && *input < trapezoidC[j])
8      {
9          partMember[j+1] = 1.0;
10     }
11     else // between C and D
12     {
13         partMember[j+1] = (trapezoidD[j]-*input)/(trapezoidD[j]-trapezoidC[j]);
14     }
15 }
16 else
17 {
18     partMember[j+1] = 0.0;
19 }

```

Matematicky jsou funkce příslušnosti pro trojúhelník a lichoběžník definovány následovně:

$$trojuhelnik(x, a, b, c) = \begin{cases} 0 & x \leq a \\ \frac{x-a}{b-a} & a \leq x \leq b \\ \frac{c-x}{c-b} & b \leq x \leq c \\ 0 & c \leq x \end{cases}$$

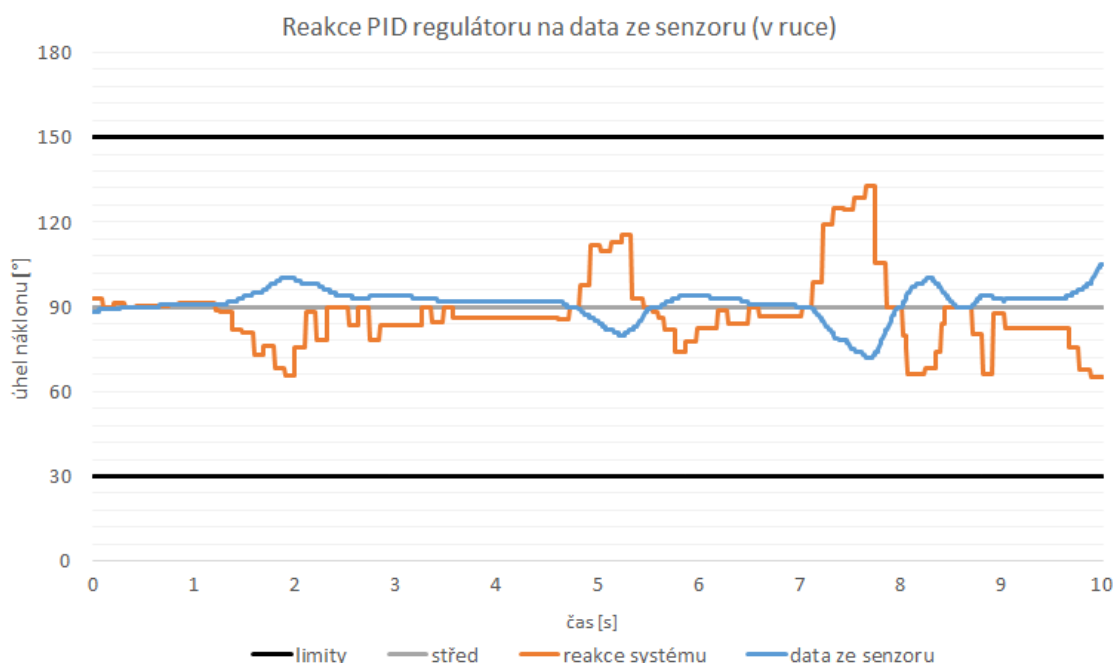
$$lichobeznik(x, a, b, c, d) = \begin{cases} 0 & x \leq a \\ \frac{x-a}{b-a} & a \leq x \leq b \\ 1 & b \leq x \leq c \\ \frac{d-x}{d-c} & c \leq x \leq d \\ 0 & d \leq x \end{cases}$$

Při testování jsem zjistil, že reakce fuzzy regulátoru při prudké odchylce nebyla dostatečná (byla slabá), a proto jsem se rozhodl do regulátoru implementovat derivační složku. Funguje

tak, že počítá rozdíl mezi každými 10 prvky, a když tento rozdíl překročí určenou hranici, tak se reakce regulátoru násobí konstantou, která je definována v kódu. Samozřejmě může nastat stav, kdy rozdíl překročí hranici v situaci, kdy se letadlo přibližuje požadované pozici - v tomto případě se reakce regulátoru nenásobí, protože by to naopak stabilizaci zhoršilo. Hranice pro překročení i konstanta byly určeny experimentálně.

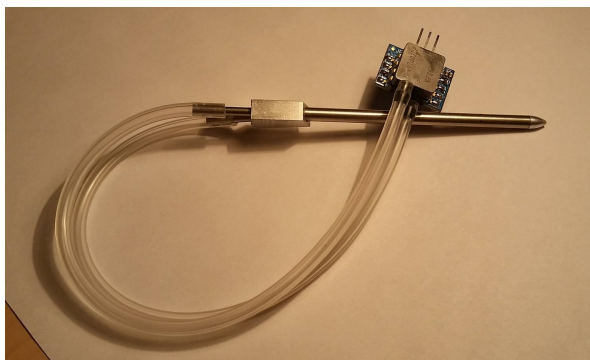
5.3.3 Komplikace při implementaci

- **PID regulace** - podle původního návrhu systému jsem implementoval PID regulátor. Použil jsem externí knihovnu *Arduino PID Library* [23]. Ze začátku jsem vyzkoušel jenom P složku, která fungovala tak, jak se očekává - vždy kolem svého cíle kmitala. Postupně jsem se snažil experimentálně nastavit integrační a derivační složky, ale nepovedlo se mi to nastavit do použitelného stavu. Navíc jsem narazil na problém, který je zřejmý z grafu 5.8 - reakce měly skokový průběh. Následující graf znázorňuje reakci PID regulátoru na data ze senzoru při regulaci jednoho kanálu - výškovky. Z tohoto důvodu jsem implementoval fuzzy regulátor, který měl lepší výsledky.



Obrázek 5.8: Reakce PID regulátoru.

- **Čtení signálů z přijímače** - problém se týká funkce *pulseIn()*, která způsobovala zpoždění celého systému, řešení je již popsáno v sekci *Čtení dat z přijímače* (5.3.2).
- **Regulace rychlosti** - do stabilizačního systému jsem plánoval integrovat regulaci rychlosti pomocí pitotovy trubice, která slouží k měření relativní rychlosti letadla. Na obrázku 5.9 lze vidět pitotovu trubici s čipem, který měří rozdíl tlaků. Z čipu se mi povedlo načíst data tzn. tlak, pomocí kterého se dá vypočítat rychlost. Problém byl v tom, že hodnoty, které čip posílal, byly absolutně neadekvátní reálné rychlosti vůči



Obrázek 5.9: Pitotova trubice s čipem MPXV7002DP.

větru. Jednoduchý test probíhal tak, že jsem v uzavřeném prostoru v bezvětrí držel pitotovu trubici v klidu, potom jsem ji držel čelem před zapnutým ventilátorem, a poté jsem ventilátor vypnul. Sledoval jsem hodnoty, které čip posílal a v klidném stavu byla rychlost cca 0 km/h , před větrákem se hodnota pohybovala mezi $10 - 20 \text{ km/h}$ a když jsem ventilátor vypnul, tak se rychlost pohybovala stále ve stejném rozmezí. Problém se mi nepodařilo vyřešit, a tak jsem pitotovu trubici vůbec v práci nepoužil.

- **Spolehlivost celého systému** - při prvních testech se stávalo, že se celá stabilizační jednotka se např. po 15 vteřinách běhu zastavila a přestala reagovat na příchozí signály jak z přijímače tak z IMU. Myslel jsem si, že servomotory nemají dostatečné napájení, ale to se neukázalo jako příčina. Když jsem po několika pokusech úplně odstranil všechny pomocné výpisy, které se posílaly přes sériový port USB, tak už se systém nezastavoval.

Kapitola 6

Výsledky

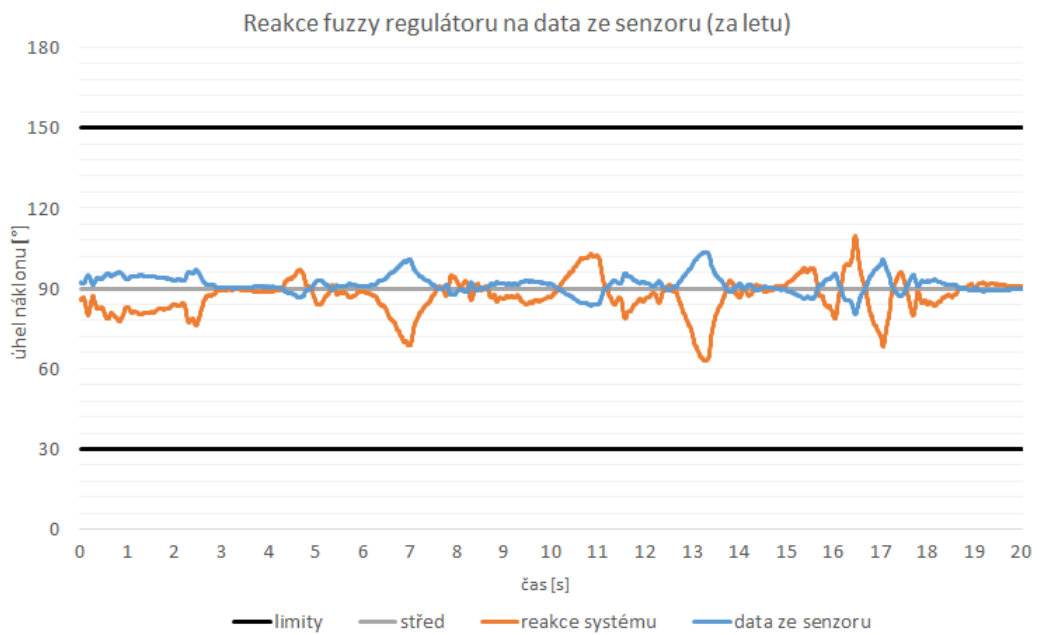
Výsledky jsem získával posloupností těchto kroků:

1. Nastavení výchozí polohy IMU na zemi - letadlo jsem položil na zem a vyvážil jej tak, aby bylo v poloze jako letadlo na obrázku číslo 5.1 - aby leželo na zemi "rovně".
2. Vzlet a vystoupení do bezpečné výšky (cca 20 m).
3. Snaha o přímý let - bez zatačení, stoupání či klesání.
4. Zapnutí stabilizace.
5. Vypnutí stabilizace.
6. Přistání.
7. Stáhnutí logovacího souboru z SD karty.

V průběhu zapnuté stabilizace byl výkon elektromotoru nastavený cca na 30 %. Letadlo drželo směr, který byl na zemi nastavený a odchylovalo se minimálně. Stabilizaci jsem musel vypínat z toho důvodu, že jsem letadlo ztrácel z dohledu, ne proto, že by model mířil k zemi.

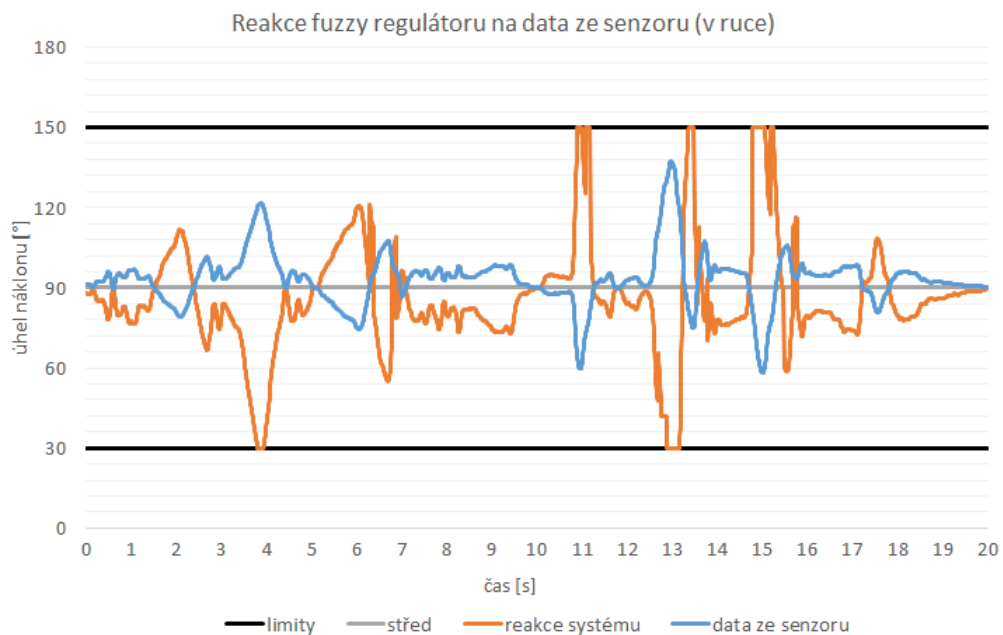
Rozumných výsledků jsem dosáhl až po velikém počtu pokusů. Chtěl bych podotknout, že volba materiálu EPP byla vzhledem k několika tvrdým přistáním správná volba (jednoduchá a levná oprava).

Zde jsou tedy záznamy reakce fuzzy regulátoru na data z IMU. Z důvodu přehlednosti je na grafech vyobrazená regulace pouze jednoho kanálu - výškovky:



Obrázek 6.1: Reakce fuzzy regulátoru - např. na intervalu od 15-20 s jde vidět, jak systém reagoval na poryv větru a po malých překmitech se poloha ustálila.

Záznam reakce při testování v ruce:



Obrázek 6.2: Reakce Fuzzy regulátoru - lze vidět přítomnost derivační složky, když letadlo začalo ostře klesat/stoupat, tak regulátor reagoval mnohem agresivněji.

Při experimentování se stav letadla nezhoršil tak, aby model potřeboval zásadní opravy, ale chtěl bych zmínit jeden pokus, po které jsem málem přišel o celé letadlo a veškerou elektroniku potřebnou k bakalářské práci. Byl jsem ve fázi, kdy byla funkční reakce na gyroskop (regulace), ale ještě nebylo funkční ovládání letadla pomocí vysílače. Bylo to způsobené problémem se čtením dat z přijímače, který jsem ještě neměl vyřešený - *Čtení dat z přijímače (5.3.2)*. Chtěl jsem vyzkoušet jak zhruba stabilizační jednotka reaguje na data z gyroskopu v reálném provozu. Nastavil jsem tedy výchozí polohu a letadlo jsem lehce hodil na menším kopečku, kousek od mého bydliště. Po mém hodě mířil model k zemi, ale těsně před pádem stabilizační jednotka přitáhla výškovku a tak letadlo nepřistálo ale letělo dál. K mému neštěstí jsem nastavil výchozí polohu tak, že letadlo mírně klesalo, ale klesalo přesně tak, jaký byl spád kopce. Letělo rychlostí cca 30 km/h a jelikož je to větroň, tak letadlo bez motoru plachtilo, drželo směr díky stabilizaci a pomalu se mi ztrácelo z dohledu. Já jsem s tím nic nemohl dělat a jenom jsem se díval. Na jednu stranu jsem se radoval - stabilizace funguje, na druhou stranu jsem nevěděl, kde můj model je. Byl podvečer, takže letadlo jsme s kamarády šli hledat až další den. Neměli jsme představu, kam mohlo letadlo doletět, ale když jsem si na mapě vyznačil koridor, kterým letadlo směřovalo (pomocí gyroskopu), tak jsme opravdu v tomto směru letadlo našli na pozemku (konkrétně na stromě), který se nacházel až 1300 metrů od místa hodů.

Kapitola 7

Závěr

Systém byl implementován a otestován. Myslím si, že hlavního cíle práce (i když nebyl kompletně splněn návrh) jsem dosáhl - při letu stačí zapnout stabilizaci pomocí spínače na ovladači a letadlo stabilně letí. Nasazení systému ve stavu, v jakém nyní je, sice není úplně praktické, ale mám v plánu tyto nedostatky implementovat a zjednodušit tak používání systému. Nicméně systém je použitelný např. pro natáčení videa, u kterého je stabilní poloha velice důležitá, nebo pro přistávání při silném větru.

Na celém stabilizačním systému jsem začal pracovat již v září 2015, tedy na začátku zimního semestru, z důvodu potřeby experimentování s letadlem (což jsem v Brně neměl možnost provádět).

Na trhu se již prodává množství stabilizačních jednotek, které mají stejný cíl jako tato práce, ale od začátku jsem plánoval udělat stabilizaci jako základní funkci jednotky, kterou budu moct do budoucna rozšířit o další funkce jako např.:

- Použití 3-polohového přepínače na ovladači pro volby více funkcí.
- Regulace rychlosti pomocí pitotovy trubice.
- Režim přistání, kroužení.
- Automatický výkrut, loop atd.
- Použití GPS - návrat domů (potřeba také výškoměr).

Jelikož jsem amatérský modelář, tak tato práce pro mě nebyla jenom zábavou, ale také zdrojem mnoha nových informací a zkušeností.

Literatura

- [1] Arduino UNO. [online]. [cit. 2016-05-14].
<http://www.czechduino.cz/?17,arduino-uno>.
- [2] Arduino's analog-to-digital converter: how it works. [online]. [cit. 2016-05-16]. <http://apcmag.com/arduino-analog-to-digital-converter-how-it-works.htm/>.
- [3] Basic Theory of Flight. [online]. [cit. 2016-05-10].
<http://www.avionale.com/basic-theory-of-flight/>.
- [4] Control By Radio Signal. [online]. [cit. 2016-05-12]. <http://www.robotoid.com/apnotes/electronics-control-by-radio-signal.html>.
- [5] GAMA střídavý motor. [online]. [cit. 2016-05-13].
<http://www.pelikandaniel.com/?sec=product&id=61150>.
- [6] GY-521 MPU6050 3-Axis Acceleration Gyroskop. [online]. [cit. 2016-05-16].
<http://www.dx.com/cs/p/gy-521-mpu6050-3-axis-acceleration-gyroscope-6dof-module-blue-154602#.Vw0lJf196M8>.
- [7] How and Why do airplanes fly? – Find Out Here. [online]. [cit. 2016-05-10].
<http://tinoshare.com/how-and-why-do-airplanes-fly/>.
- [8] How To Get Into Hobby RC: Telemetry Systems. [online]. [cit. 2016-05-13].
<http://www.tested.com/tech/532834-how-get-hobby-rc-telemetry-systems/>.
- [9] Katana - zelená ARF. [online]. [cit. 2016-05-11].
<http://www.va-models.com/eshop1/katanazelenaarf-p-549.html>.
- [10] Launching RC gliders. [online]. [cit. 2016-05-11].
<http://www.rc-airplane-world.com/launching-rc-gliders.html>.
- [11] Objasnění pojmu fuzzy logika. [online]. [cit. 2016-05-16].
<http://www.controlengcesko.com/hlavni-menu/artykuly/artykul/article/objasneni-pojmu-fuzzy-logika/>.
- [12] OPTIC 6 SPORT 2,4 GHz (mode 1), přijímač MINIMA 6S. [online]. [cit. 2016-05-12].
<http://www.peckamodel.cz/produkt/rc-modely-a-prislusenstvi/rc-soupravy-a-prislusenstvi/rc-soupravy/rc-soupravy-hitec/1hi1095-optic-6-sport-2-4-ghz>.

- [13] Optima 6 - 6 Channel 2.4GHz Receiver. [online]. [cit. 2016-05-12]. <http://hitecrd.com/products/aircraft-radios-receivers-and-accessories/2.4ghz-aircraft-receivers-modules/optima-6-6-channel-2.4ghz-receiver/product>.
- [14] RAY R-50SB střídavý regulátor 50A SBEC. [online]. [cit. 2016-05-14]. <http://www.pelikandaniel.com/?sec=product&id=69370>.
- [15] RTF Wild Hawk. [online]. [cit. 2016-05-10]. <http://www.nitroplanes.com/rtfwiha54el3.html>.
- [16] Souřadnicové soustavy používané v letectví. [online]. [cit. 2016-05-15]. http://www.slavetind.cz/stavba/mechanika/sour_soustavy.aspx.
- [17] Stavíme kvadrokoptéru: bezpečnost, PID regulátor. [online]. [cit. 2016-05-16]. <http://www.root.cz/clanky/stavime-kvadrokopteru-bezpecnost-pid-regulator/>.
- [18] STINGER II. [online]. [cit. 2016-05-14]. <http://www.greatplanes.com/lanier/gpma1010.html>.
- [19] Types of RC Airplanes. [online]. [cit. 2016-05-10]. <http://www.hooked-on-rc-airplanes.com/types-of-rc-airplanes.html>.
- [20] Český telekomunikační úřad: Všeobecné oprávnění č. VO-R/15/08.2005-27 k využívání rádiových kmitočtů a k provozování zařízení pro dálkové ovládání modelů v pásmech 13 MHz až 40 MHz. [online]. [cit. 2016-05-05]. http://www.ctu.cz/1/download/0patreni%20obecne%20povahy/VO_R_15_08_2005_27.pdf.
- [21] Airlines, C.: Základy letu. [online]. [cit. 2016-05-05]. <http://www.csavirtual.cz/cs/vyukove-centrum/skola-pilotu/146-zaklady-letu?catid=44%3Astudium>.
- [22] B, D.: How To Read Multiple RC Channels. [online]. [cit. 2016-05-11]. <http://rcarduino.blogspot.cz/2012/04/how-to-read-multiple-rc-channels-draft.html>.
- [23] Beauregard, B.: Arduino PID library. [online]. [cit. 2016-05-11]. <http://playground.arduino.cc/Code/PIDLibrary>.
- [24] Blabol, J.: Manuál stavby RC modelů. [online]. [cit. 2016-05-05]. http://www.akroforum.cz/download/manuals/manual_stavby_rc_modelu.doc.
- [25] Chvalovský, K.: Která bezdrátová technologie je nejlepší? [online]. [cit. 2016-05-05]. <http://www.lupa.cz/clanky/ktera-bezdratova-technologie-je-nejlepsi/>.
- [26] DANIEL, F. P.: BETA 1400. [online]. [cit. 2016-05-05]. <http://www.pelikandaniel.com/?sec=product&id=46822>.
- [27] DANIEL, F. P.: GAMA 2100 Sport - ARF brushless, vztlak klapky, 6 serv. [online]. [cit. 2016-05-05]. <http://www.pelikandaniel.com/?sec=product&id=60971>.
- [28] Fialková, M.: Proč letadlo letí - polopaticky. [online]. [cit. 2016-05-05]. <http://www.laazatec.cz/clanek448.html>.

- [29] Homolka, J.: Modelářské elektromotory - 2. díl. [online]. [cit. 2016-05-05].
[://www.rc-zoom.cz/modelarske-elektromotory-2-dil/](http://www.rc-zoom.cz/modelarske-elektromotory-2-dil/).
- [30] Homolka, J.: Modelářské elektromotory - 4. díl. [online]. [cit. 2016-05-05].
<http://www.rc-zoom.cz/modelarske-elektromotory-4-dil/>.
- [31] Josef Schwarz, J. S., Richard Růžička: IMP - studijní opora.
- [32] Karas, T.: VŠE O SPALOVACÍCH (ŽHAVICÍCH) MOTORECH. [online]. [cit. 2016-05-05].
<http://rchouby.cz/index.php?cl=1/spalovaci>.
- [33] Kolman, L.: Krystaly. [online]. [cit. 2016-05-05].
<http://www.kolmanl.info/index.php?show=KRISTALY>.
- [34] Kolman, L.: Li-Pol. [online]. [cit. 2016-05-05].
http://www.kolmanl.info/index.php?show=aku_lipol.
- [35] Modelarina.cz: Modelářský slovníček. [online]. [cit. 2016-05-05].
<http://www.modelarina.cz/page.html?chapter=0&id=3>.
- [36] Peringer, D. I. P.: IMS - přednášky.
- [37] Rowberg, J.: 2C device library collection for AVR/Arduino or other C++-based MCUs. [online]. [cit. 2016-05-11].
<https://github.com/jrowberg/i2cdevlib/>.
- [38] Software, A.: MPU-6050 Accelerometer + Gyro. [online]. [cit. 2016-05-11].
<http://playground.arduino.cc/Main/MPU-6050>.
- [39] Valter, J.: PID regulátor. [online]. [cit. 2016-05-11].
<http://valter.by1.cz/plynula-regulace-pid>.

Přílohy

Seznam příloh

A Obsah CD

46

Příloha A

Obsah CD

Adresářová struktura CD:

- *src/* - zdrojové soubory, demoaplikace
- *lib/* - použité externí knihovny
- *xdenkf00.pdf* - technická zpráva
- *README.txt*