

**UNIVERZITA PARDUBICE  
FAKULTA ELEKTROTECHNIKY  
A INFORMATIKY**

**DIPLOMOVÁ PRÁCE**

2011

Bc. Pavel Chmelař

**UNIVERZITA PARDUBICE  
FAKULTA ELEKTROTECHNIKY A INFORMATIKY**

**Stabilizace polohy létajícího objektu pomocí  
inerciálních senzorů**

**Bc. Pavel Chmelař**

Diplomová práce

2011

## ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Pavel CHMELAŘ**  
Osobní číslo: **I09399**  
Studijní program: **N2612 Elektrotechnika a informatika**  
Studijní obor: **Komunikační a řídicí technologie**  
Název tématu: **Stabilizace polohy létajícího objektu pomocí inerciálních senzorů**  
Zadávací katedra: **Katedra elektrotechniky**

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

Práce se bude zabývat určením změny polohy předmětu se 6 stupni volnosti pomocí inerciálních senzorů a zpětnovazebním řízením polohy, ve smyslu stabilizace polohy volného modelu. Létající objekty jako například vrtulníky, obzvláště pokud jsou menší hmotnosti, jsou velice citlivá na změny okolního prostředí a nepřesnosti v řízení či mechanické konstrukci a mají tendenci se chovat nestabilně měnit náhle polohu, rotovat atp. Kompenzace těchto výchylek uživatelem, zvláště u modelů s malým momentem setrvačnosti, je velice namáhavé ne-li prakticky nemožné. U některých modelů se situace částečně řeší umělým zvyšováním momentu setrvačnosti pomocí gyroskopického momentu rotujících setrvačnicků, což vede ke zvýšení stability na úkor ovladatelnosti. S rozvojem miniaturních inerciálních čidel se přímo nabízí možnost jejich využití jako součásti řízení pro kompenzaci těchto vlivů. Data z inerciálních čidel bývají značně zašuměná a jejich použití je zatíženo velkou chybou, proto je potřeba signály filtrovat a upravit před dalším použitím.

Náplní teoretické části bude přehled principů inerciálních prvků a jejich použití pro výpočet polohy. Popis možností úpravy a filtrace signálů z čidel (Kalmánův filtr atp.) Úvod do problematiky řízení vybraného modelu s ohledem na typ měřených dat, přehled použitelných regulačních pochodů.

Náplní praktické části bude vytvoření / výběr modulu s potřebnými inerciálními čidly, nastavení kalmánova filtru a algoritmu pro výpočet polohy s ohledem na chování modelu. Dále pak práce bude obsahovat návrh regulačního pochodu pro stabilizaci modelu s využitím měřených dat. Po simulovaném ověření například v prostředí matlab bude prakticky ověřen na reálném modelu, implementace měření a řízení musí proběhnout s ohledem na omezenou nosnost modelu.

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

**RIPKA, P.; TIPEK, A. Master Book of Sensors. Praha: BEN, 2003. ISBN 0-12-752184-4**

**Kalman filter [online] Dostupný z WWW:  
[http://en.wikipedia.org/wiki/Kalman\\_filter](http://en.wikipedia.org/wiki/Kalman_filter)**

**BALÁTĚ J.; Automatické řízení, BEN, ISBN 978-80-7300-148-3**

**ŠULC B.; VÍTEČKOVÁ M.; Teorie a praxe návrhů regulačních obvodů, ČVUT, ISBN 80-01-03007-5**

Vedoucí diplomové práce:

**Ing. Pavel Rozsival**  
Katedra elektrotechniky

Datum zadání diplomové práce: **17. prosince 2010**

Termín odevzdání diplomové práce: **20. května 2011**



prof. Ing. Simeon Karamazov, Dr.

děkan



L.S.



Ing. Zdeněk Němec, Ph.D.

vedoucí katedry

V Pardubicích dne 12. listopadu 2010

## Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně Univerzity Pardubice.

V Pardubicích dne 20.5. 2011

Pavel Chmelař

## Poděkování

Tímto bych rád poděkoval vedoucímu diplomové práce, Ing. Pavlu Rozsívalovi, za poskytnuté odborné rady a připomínky během zpracování této práce a všem ostatním lidem, kteří mě podporovali. Především bych tímto rád poděkoval rodičům, za podporu a financování po celou dobu studia.

## SOUHRN

Cílem této práce bylo navrhnout zařízení pro stabilizaci polohy létajícího objektu použitím inerciálních senzorů (akcelerometr, gyroskop). Funkci celého systému řídí mikrokontroler, který získaná data z jednotky měření inerciálních veličin zpracovává a předává PID regulátoru, jehož úkolem je dle zvolené požadované hodnoty polohy setrvat v tomto stavu. Mikrokontroler s inerciálním senzorem a podpůrnými obvody, jsou připevněny na létajícím objektu quadcopter. Quadcopter setrvává v klidu nad zemí a za pomoci regulátoru je schopen vyrovnávat vnější vlivy okolí, které působí na jeho stabilitu v prostoru.

## KLÍČOVÁ SLOVA

jednotka měření inerciálních veličin; akcelerometr; gyroskop; mikrokontroler; quadcopter, PID regulátor

## TITLE

The position stabilization of a flying object using inertial sensors

## ABSTRACT

The aim of this work was to develop device for the position stabilization of a flying object using inertial sensors (accelerometer, gyroscope). Function of the system manages microcontroller, which collected data form IMU sensor, proposed and pass data to PID regulator, whose objective it's according to set required value of position keep on this position. Microcontroller with IMU and subcircuits are mounted on flying object quadcopter. Quadcopter is in steady state at position relative to Earth and with help of regulator balance influence external environment, which effect on stability in space.

## KEYWORDS

IMU (Inertial Measurement Unit); akcelerometr; gyroscope; microcontroler; quadcopter, PID controller

## Obsah:

Seznam zkratk:	10
Seznam obrázků:	12
Seznam tabulek:	13
Úvod	14
1. Akcelerometry	15
1.1. Způsob vyhodnocení zrychlení	15
1.2. Skutečná gravitační síla Země	17
1.3. Princip 3D MEMS akcelerometru	18
1.3.1. Mechanická struktura	18
1.3.2. Elektrická struktura	21
1.4. Závislost na teplotě	22
1.5. Možnosti použití 3D MEMS akcelerometrů	23
1.6. Základní dělení akcelerometrů	24
2. Gyroskopy	25
2.1. Princip funkce klasického rotačního gyroskopu	26
2.2. Dělení gyroskopů	28
2.3. Princip vibračního gyroskopu	28
3. Získávání dat	30
3.1. Průměr	30
3.2. Klouzavý průměr	31
3.2.1. Klouzavý průměr – jednoduchý (SMA)	32
3.2.2. Klouzavý průměr – exponenciální (EMA)	33
3.3. Problémy výpočtu klouzavých průměrů	35
3.4. Kalmanův filtr	37
3.4.1. Základní části Kalmanova filtru:	37
3.4.1.1. Stavový vektor	37
3.4.1.2. Dynamický model	37
3.4.1.3. Model pozorování	38
3.4.2. Algoritmus Kalmanova filtru:	38
3.4.2.1. Predikce	38
3.4.2.2. Korekce	39
3.5. Rozšířený Kalmanův filtr	40
3.5.1. Predikce	41
3.5.2. Korekce	41
4. Orientace v prostoru	43
5. Regulace	45
5.1. Dělení regulátorů	46
5.1.1. Proporcionální regulátor P	46
5.1.2. Integrační regulátor I	47
5.1.3. Derivační regulátor D	47
5.1.4. Kombinace P, I, D	48
5.2. Nastavování parametrů regulátoru PID	49
6. Praktické řešení	51
6.1. IMU senzor CHR-6D	51
6.1.1. Hlavní vlastnosti modulu	52
6.1.2. Základní funkce	53
6.1.3. Kalibrace	53
6.1.4. Komunikace s IMU	54



6.1.4.1.	CHR-6D RX pakety.....	54
6.1.4.2.	CHR-6D TX pakety.....	55
6.2.	Mikrokontroler – kit STM32 VL Discovery .....	55
6.2.1.	LCD displej.....	57
6.2.1.1.	Zapojení LCD displeje a ovládání .....	58
6.2.2.	Komunikace .....	59
6.3.	Konstrukce quadcopteru.....	59
6.3.1.	Motory a regulátory.....	61
6.3.2.	Nastavení regulátorů .....	62
6.3.3.	Řízení motorů pomocí PWM .....	63
6.3.4.	Deska plošného spoje.....	64
6.3.5.	Kompletní sestava .....	65
6.4.	Softwarové řešení .....	67
6.4.1.	Vývojový diagram aplikace .....	68
6.4.2.	Regulace.....	69
6.4.3.	Řízení PWM.....	70
6.4.4.	Obsluha LCD displeje .....	70
6.4.5.	Způsob komunikace .....	72
6.4.6.	Software pro PC .....	73
6.4.7.	Software pro mobilní telefon .....	74
7.	Závěr .....	76
	Seznam použité literatury:.....	77
	Přílohy:.....	79

## Seznam zkratek:

akcelerometr.....	senzor zrychlení
ASIC.....	(Application-Specific Integrated Circuit) – integrované obvody pro určité použití
MEMS.....	(Micro-Electro-Mechanical Systems) – integrace mechanických elementů, senzorů, akčních členů a elektroniky na křemíkový substrát
Coriolisova síla.....	zvláštní odstředivá síla uplatňující se při relativním pohybu hmotného bodu v nesetřvačné otáčející se soustavě
Coriolisovo zrychlení.....	složka celkového zrychlení hmotného bodu pohybujícího se v otáčející se soustavě
precese.....	fyzikální jev, při němž se působením vnější dvojice sil mění orientace osy setřvačnicku v prostoru
gyroskop.....	setřvačnick
ESA.....	Evropská kosmická agentura
GPS.....	(Global Position System) – systém pro určování polohy
IMU.....	(Inertial Measurement Unit) – jednotka pro určení základní polohy objektu v prostoru za pomoci akcelerometru a gyroskopu
PID regulátor.....	běžný regulační prvek pro řízení systému. Zahrnuje v sobě proporcionální, integrační a derivační složku
TTL.....	(Transistor-Transistor-Logic) – standard používaný pro implementaci digitálních integrovaných obvodů. Běžné napájecí napětí je 5 V, z čehož vyplývá pro „log 1“ napětí přibližně 5 V
USART.....	(Universal Synchronous and Asynchronous Receiver/Transmitter) – přijímač, vysílač sériového přenosu dat
Bd.....	(Baud) – jednotka modulační rychlosti
mikrokontroler (MCU).....	většinou bývá v podobě monolitického integrovaného obvodu obsahující kompletní mikropočítač
AVR.....	označení pro rodinu osmibitových mikročipů typu RISC s harvardskou architekturou od firmy Atmel

ARM.....	(Advanced RISC Machine) – označení architektury MCU
embedded.....	vestavěný jednoúčelový systém, ve kterém je počítač zabudován přímo do zařízení
assembler.....	jazyk symbolických adres
POR.....	(Power On Reset) – periférie mikrokontroleru, která způsobí restartování po připojení napájecího napětí, pro zajištění definovaného vnitřního stavu
PDR.....	(Power Down Reset) – periférie mikrokontroleru, která způsobí restartování po odpojení napájecího napětí, zajistí řízené vypnutí
LVD.....	(Low Voltage Detect) – periférie pro detekci poklesu napájecího napětí pod referenční úroveň a následně generuje signál reset
WDT.....	(Watchdog timer) – periférie pro reset systému při zastavení činnosti z nežádoucí příčiny
SPP.....	(Serial Port Profile) – emulace virtuální sériové linky prostřednictvím bluetooth
ESC.....	(Electronic Speed Control) – elektronický obvod s účelem ovládní otáček motoru
Li-Pol.....	lithium polymerové nabíjecí baterie
LCD.....	(Liquid Crystal Display) – displej z tekutých krystalů
TFT.....	(Thin Film Transistor) – tenkovrstvý tranzistor, varianta provedení LCD displeje
RGB.....	barevný model, aditivní způsob míchání barev
bmp.....	(Windows Bitmap) – přípona souboru obrázku
SDK.....	(Software Development Kit) – vývojové nástroje pro tvorbu aplikací na dané platformě nebo hardware
Perl.....	interpretovaný programovací jazyk

## Seznam obrázků:

Obr. 1.01 – Příklad připojení akcelerometru.....	15
Obr. 1.02 – Příklady snímání zrychlení – (2).....	16
Obr. 1.03 – Skutečná gravitační síla Země – (3).....	17
Obr. 1.04 – Základní princip MEMS akcelerometru – (4).....	18
Obr. 1.05 – Znázornění mechanické MEMS struktury akcelerometru – (4) .....	19
Obr. 1.06 – Detailní záběry na reálné provedení jednotlivých částí struktury - pevných a pohyblivých elektrod (electrodes), detekční hmoty (mass), pružiny (spring) a samotestovací strukturu (self test) – (4) .....	20
Obr. 1.07 – 3D struktura akcelerometru (měření zrychlení v osách X, Y, Z) – (3)...	20
Obr. 1.08 – Struktura spojení snímače a ASIC elektroniky – (4) .....	21
Obr. 1.09 – Blokové schéma elektrické struktury 3D akcelerometru – (4) .....	21
Obr. 1.10 – Vliv teploty na nulové body „g“ v osách XYZ – (2).....	22
Obr. 1.11 – Použití akcelerometrů dle snímacího rozsahu – (4).....	23
Obr. 2.01 – Představa pro lepší porozumění principu gyroskopu – (5).....	26
Obr. 2.02 – Představa působení síly – (5) .....	26
Obr. 2.03 – Precesní pohyb – (5) .....	27
Obr. 2.04 – Kompozice vibračního poměru gyroskopu – (1) .....	28
Obr. 2.05 – Corilissova síla F – (1) .....	29
Obr. 3.01 – Grafické znázornění průměru .....	31
Obr. 3.02 – Grafické znázornění klouzavého průměru pro $k=20$ (SMA) .....	33
Obr. 3.03 – Grafické znázornění klouzavého průměru (EMA) .....	34
Obr. 3.04 – Grafické znázornění klouzavého průměru pro různá $k$ .....	35
Obr. 3.05 – Grafické znázornění rychle se měnícího průběhu.....	36
Obr. 3.06 – Grafické znázornění jednoduchého Kalmanova filtru .....	42
Obr. 4.01 – Znázornění souřadnicových systémů (10) .....	43
Obr. 5.01 – Jednoduchý regulační obvod .....	45
Obr. 5.02 – Impulzní odezva a přenosová funkce regulátoru P (11) .....	46
Obr. 5.03 – Impulzní odezva a přenosová funkce regulátoru I (11) .....	47
Obr. 5.04 – Impulzní odezva a přenosová funkce regulátoru (11).....	48
Obr. 5.05 – PID regulátor.....	48
Obr. 5.06 – Přenosová funkce PID.....	49
Obr. 5.07 – Přehled metod nastavení PID regulátoru (12).....	50
Obr. 6.01 – CHR-6D a jeho skutečná velikost.....	51
Obr. 6.02 – Orientace modulu (13) .....	52
Obr. 6.03 – STM32VLDISCOVERY .....	56
Obr. 6.04 – Ukázka výstupu dat LCD S65 .....	57
Obr. 6.05 – Zadní strana a označení připojení LCD S65 ( <sup>1</sup> str.70).....	58
Obr. 6.06 – Způsob vykreslování ( <sup>1</sup> str.70).....	58
Obr. 6.07 – Bluetooth modul – Parallax (15).....	59
Obr. 6.08 – Princip funkce quadcopteru .....	60
Obr. 6.09 – Princip funkce rameno 1 .....	60
Obr. 6.10 – Princip funkce rameno 2 .....	60
Obr. 6.11 – Motor Turnigy CDRUM C3530-1400 a regulátor HK45A ESC.....	62
Obr. 6.12 – Přechodová charakteristika .....	62
Obr. 6.13 – Zjištění kritické periody z průběhu výkonu motoru 1 .....	63
Obr. 6.14 – Řízení signálem PWM .....	64
Obr. 6.15 – Deska plošného spoje quadcopteru.....	64
Obr. 6.16 – Quadcopter.....	65

Obr. 6.17 – Zapojení MCU, CHR–6D a bluetooth .....	66
Obr. 6.18 – Vývojový diagram main a regulační smyčky .....	68
Obr. 6.19 – Výstup testovacího programu pro převod čísel double na pole znaků ...	72
Obr. 6.20 – SW pro PC .....	73
Obr. 6.21 – Základní ovládání quadcopteru.....	74
Obr. 6.22 – Aplikace v mobilním telefonu .....	75

Seznam tabulek:

Tab. 1.01 – Základní dělení akcelerometrů.....	24
Tab. 5.01 – Nastavení parametrů metodou Ziegler-Nichols .....	50
Tab. 6.01 – Popis pinů modulu IMU.....	52
Tab. 6.02 – Struktura datového rámce RX.....	54
Tab. 6.03 – Popis datového rámce RX.....	54
Tab. 6.04 – Struktura paketu GET_DATA .....	54
Tab. 6.05 – Struktura datového rámce RX.....	55
Tab. 6.06 – Popis datového rámce RX.....	55
Tab. 6.07 – Struktura paketu SENSOR_DATA.....	55
Tab. 6.08 – Struktura paketu komunikace mezi PC a kvadcopterem .....	72
Tab. 6.09 – RX pakety přijímané MCU.....	72
Tab. 6.10 – TX pakety vysílané MCU .....	73

## Úvod

Úkolem této diplomové práce bylo navrhnout systém pro stabilizaci létajícího objektu. Návrh systému spočíval ve výběru vhodného senzoru inerciálních veličin a mikrokontroleru, který bude data zpracovávat a řídit kompletní systém včetně algoritmu stabilizace polohy. Regulaci zajišťuje softwarový PID regulátor, který řídí výkony motorů quadrotoru ve správném poměru vzhledem k jeho poloze.

První dvě kapitoly popisují akcelerometry a gyroskopy, několik slov na úvod do problematiky, principy jejich funkce, popis základních vlastností a možnosti využití. Zmíněny jsou i problémy způsobující nepřesnost ve vyhodnocení získaných dat. Třetí kapitola je zaměřená na filtraci dat od nejjednodušších způsobů, jako je použití aritmetického průměru, až po rozbor Kalmanova filtru. U všech filtrů je názorná ukázka aplikovaná na data získaná z modulu CHR-6D. Čtvrtá kapitola popisuje orientaci objektu v prostoru. Zejména jsou popsány důležité úhly, které je nutné stabilizovat. Pátá kapitola popisuje regulaci. Je v ní obsažen popis základních regulátorů, dále jejich kombinování s uvedenými výhodami a nevýhodami. Zahrnuje i přehled metod nastavování regulátorů s praktickou ukázkou často využívané metody a jejím postupem. V šesté kapitole je popsáno praktické řešení. Zabývá se modulem CHR-6D, principem funkce modulu, jeho obsluhy a kalibrace. Poté je rozebírán popis kitu STM32VLDISCOVERY, jeho výhody, možnosti použití, způsob programování a vlastnosti MCU na desce kitu. Z důvodu sledování stavu je použit LCD displej, u kterého je rovněž rozbor jeho ovládání a komunikace s mikrokontrolerem. V další části je popsána komunikace s modelem quadcopteru. Podstatnou částí kapitoly je rozbor konstrukce quadcopteru a jeho způsobu regulace s nastavováním konstant regulátoru. Následuje popis funkce a výhod použitých motorů včetně jejich regulace a způsobu řízení otáček. V části popisující funkci software jsou naznačeny problémy, které se mohou vyskytnout při vývoji na použitém mikrokontroleru, dále řešení posílání dat po sériové lince, vývojový diagram programu běžící na MCU se zaměřením na způsob řešení regulace. Zmíněna je také obsluha časovače pro řízení PWM signálu pro motory. Ke konci kapitoly je popsán software pro ovládání quadcopteru s možnostmi jeho ovládání včetně řešení komunikace mezi MCU a ovládacím zařízením pomocí paketů. Na závěr kapitoly je popsáno ovládání prostřednictvím aplikace v mobilním telefonu. Poslední, sedmá kapitola je shrnutí dosažených výsledků a závěr.

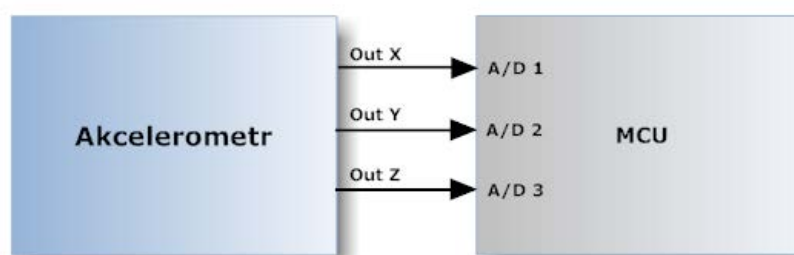
# 1. Akcelerometry

Akcelerometry se v dnešním robotizovaném světě využívají stále více. Proto nikdo nechce v této oblasti zůstat pozadu a společnosti zabývající se jejich výrobou neustále vyvíjejí nové nebo vylepšují stávající funkční struktury pro vylepšení vlastností senzorů. Díky vylepšování výrobního procesu se snižuje jejich spotřeba i vlastní velikost a cena. S tímto souvisí i obliba implementace do různých mobilních nebo embedded zařízení. V posledních dvaceti letech speciálně díky optice a mikrotechnologijím byl enormní pokrok v preciznosti, linearitě a stabilitě.

Podle základních mechanických principů akcelerace vychází ze vztahu mezi silou a hmotou. Akcelerometry využívají těchto fyzikálních jevů a jejich pomocí získávají elektrický signál. Umožňují přímé měření signálu (piezoelektrický senzor) nebo nepřímé, zde se však může vyskytnout chyba v převodu mezi veličinami a tím klesá schopnost určení přesné hodnoty senzoru. Tyto senzory můžeme hodnotit porovnáváním vlastností, poté se dle kritérií na přesnost určí vhodnost použití senzoru. Přesnost je limitována fyzikálním principem, s kterým pracuje konkrétní akcelerometr a jakou technologií je vyroben.

V různých odvětvích, jako je automobilní, letového provozu, přístrojové techniky, lékařství a automatizace se vlastnosti akcelerometrů výrazně liší, tomu odpovídá adekvátně cena.

## 1.1. Způsob vyhodnocení zrychlení

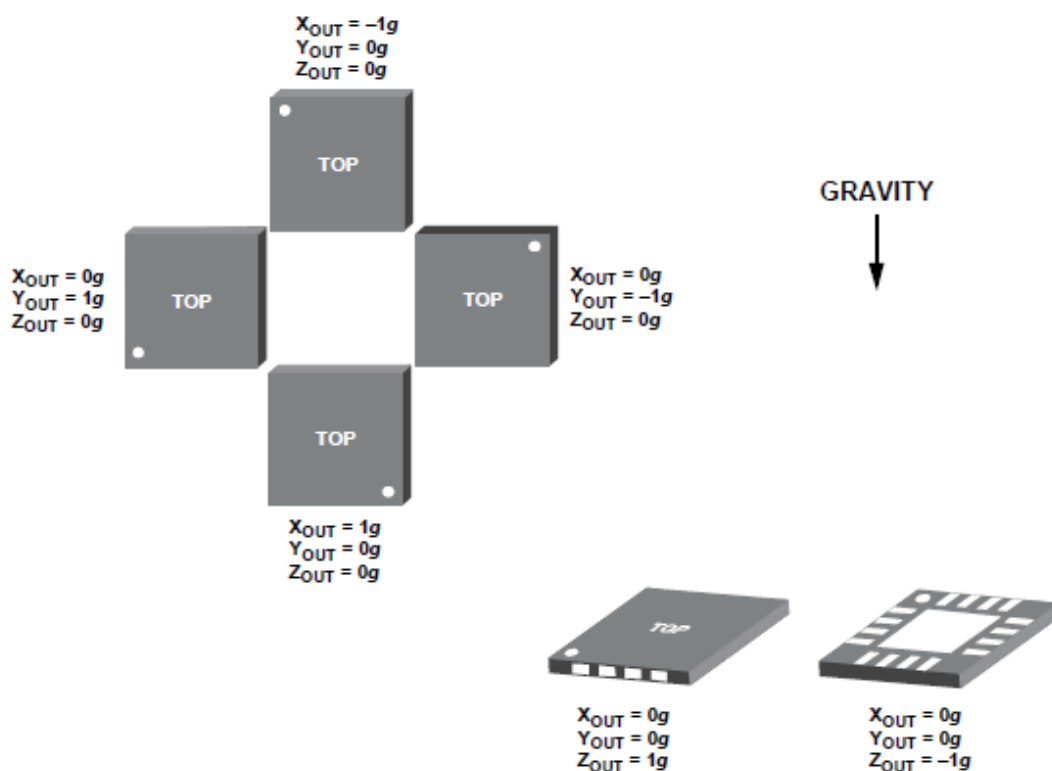


Obr. 1.01 – Příklad připojení akcelerometru

Výstupy jednotlivých os (XYZ) akcelerometru jsou připojeny k A/D převodníku (například A/D v MCU (Obr. 1.01)). Úroveň napětí výstupu akcelerometru a MCU musí být pro správné vyhodnocení požadovaných hodnot stejné. Podle parametru počtu bitů A/D převodníku se určí, v kolika úrovních napětí se mohou vyskytovat výstupy akcelerometru. Citlivost akcelerometru po přepočtu udává maximální velikost zrychlení, kterého je schopen akcelerometr dosáhnout, a odpovídá maximál-

ní hodnotě výstupního napětí konkrétní osy. Záporné zrychlení neboli zpomalení má maximální hodnotu při úrovni výstupu konkrétní osy na hodnotě nula voltů. Nulové zrychlení má akcelerometr v polovině rozsahu souboru hodnot A/D převodníku (bývá označována jako „bias“) respektive v polovině referenčního napětí přivedeného na převodník. Tato úroveň má důležitý význam pro přesné vyhodnocení polohy v prostoru.

Následující ukázka (Obr. 1.02) znázorňuje výstupy jednotlivých os akcelerometru. Pravá část ukazuje působení gravitace na akcelerometr v případě, kdy je rovnoběžně se zemským povrchem. Taková orientace způsobí v ose Z právě zrychlení ( $1g$ ) odpovídající hodnotě gravitačního zrychlení Země ( $g = 9,81 m \cdot s^{-2}$ ). V ostatních osách nepůsobí žádné zrychlení. V levé části jsou uvedeny další příklady polohy akcelerometru vzhledem k povrchu Země s odpovídajícími výstupy v jednotlivých osách.



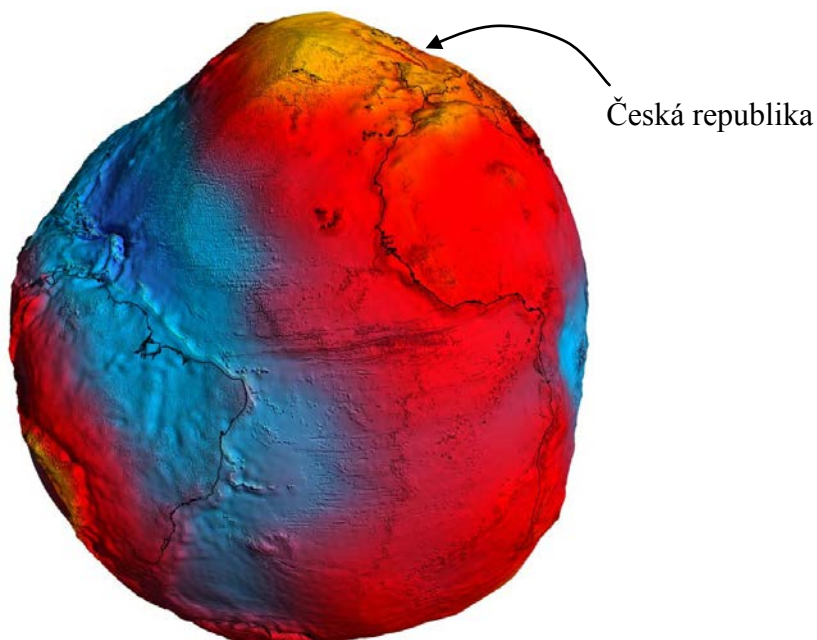
Obr. 1.02 – Příklady snímání zrychlení – (2)



## 1.2. Skutečná gravitační síla Země

Evropská kosmická agentura (ESA) v článku publikovaném 31. března 2011 uvedla výsledky projektu GOCE (Gravity Field and Steady-State Ocean Circulation Explorer), který měl za úkol zkoumat gravitační pole Země. Sonda, pojmenovaná podle názvu projektu měla za úkol sbírat data po dobu dvanácti měsíců. Vypuštěna byla v březnu roku 2009. Palivo pro svou misi má do konce roku 2012 a stále může sbírat data a jejich pomocí korigovat chyby v měření.

Sonda obíhá planetu na nízké orbitě necelých 260 km nad zemským povrchem. Hlavním prvkem sondy je gradiometr zahrnující šest vysocecitlivých akcelerometrů měřící gravitaci ve 3D. Jde o první takovéto využití ve vesmíru. Část z kompletního 3D modelu je zobrazena na následujícím obrázku (Obr. 1.03).



Obr. 1.03 – Skutečná gravitační síla Země – (3)

Planeta Země nemá tvar koule a gravitační zrychlení je v rozsahu od  $9,78 \text{ m} \cdot \text{s}^{-2}$  do  $9,82 \text{ m} \cdot \text{s}^{-2}$ . Na gravitačním modelu (Obr. 1.03) modrá místa znázorňují oblasti s nejmenším gravitačním zrychlením a žlutá místa naopak nejvyšší. Vědci nyní mají k dispozici podrobný model gravitačního pole a mohou tak lépe pochopit chování planety, jako měření proudění oceánů, změny hladin moří, chování ledovců při změně klimatu nebo vulkanickou aktivitu a zemětřesení.

Nepřesnost v určení správné hodnoty gravitačního zrychlení může vést k chybnému vyhodnocení výsledku aktuálního zrychlení a v případě použití akcelerometru při měření překonané vzdálenosti za pomoci dvojitě integrace vzniká v systému chyba.

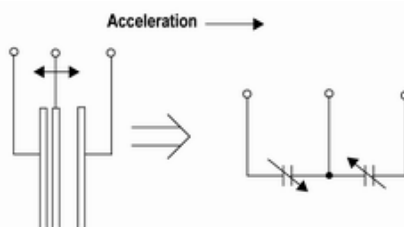
### 1.3. Princip 3D MEMS akcelerometru

#### 1.3.1. Mechanická struktura

Struktura a funkce MEMS akcelerometru je založena na proměnné kapacitě tříelektrového vzduchového kondenzátoru. Využívá se zde známé nelineární závislosti kapacity  $C$  na vzdálenosti elektrod kondenzátoru  $d$  (velikosti vzduchové mezery) dle vzorce (1.1):

$$C = \varepsilon \cdot \frac{S}{d} [F] \quad (1.1)$$

( $\varepsilon$  – permitivita,  $S$  – plocha elektrod). Pokud je tedy jedna elektroda pohyblivá a její pohyb bude závislý na působícím zrychlení, vzniká kapacitní akcelerometr. Struktura pohyblivých nosítků (elektrod) je snadno realizovatelná pomocí technologie MEMS, proto mají tyto akcelerometry název podle výrobního procesu.



**Obr. 1.04 – Základní princip MEMS akcelerometru – (4)**

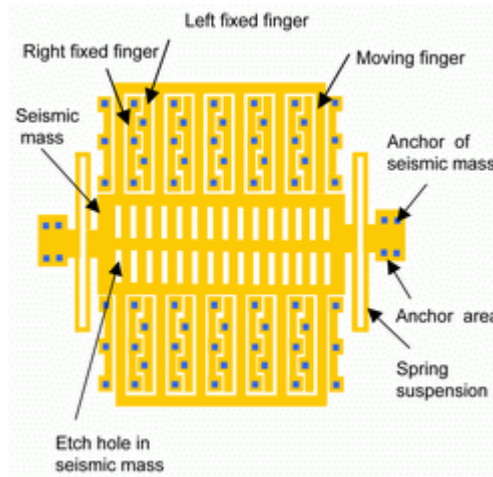
Výsledná funkční struktura ovšem není až tak jednoduchá, jak se na první pohled zdá. Důležité je právě zajistit lineární a dostatečně citlivý převod zrychlení na mechanický posuvný pohyb. Ten totiž určuje samotný měřicí rozsah senzoru, tj. maximální a minimální měřitelné zrychlení [4].

Zde se vychází ze základního vztahu druhého Newtonova zákona, upraveného (1.2) pro působení síly při zrychlení:

$$F = m \cdot a [N] \quad (1.2)$$

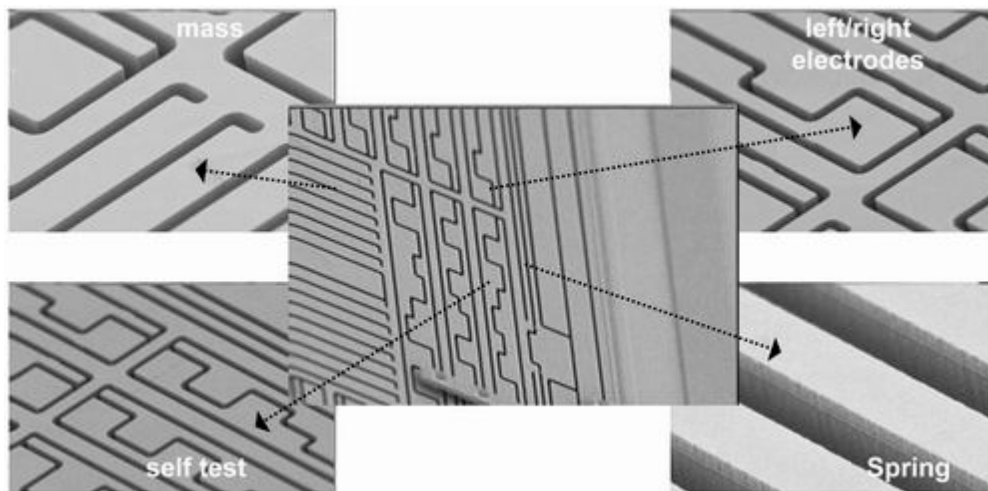
Kde  $F$  je síla vzniklá působením zrychlení  $a$  na hmotu  $m$  (Seismic mass). Síla se pak přes pružiny (Spring suspension) převádí na posuv nosíčku (Seismic mass),

jejíž některé části tvoří pohyblivé elektrody vzduchového kondenzátoru (Moving finger). Jejich pozice vůči levým pevným elektrodám (Left fixed fingers) a pravým pevným elektrodám (Right fixed fingers) určuje elektronicky měřenou hodnotu kapacity takto vzniklého kondenzátoru.[4]



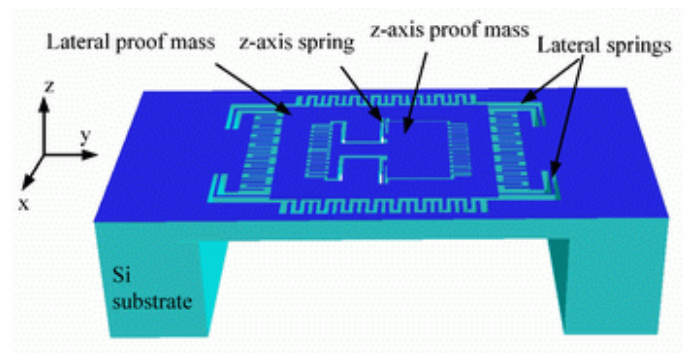
**Obr. 1.05 – Znázornění mechanické MEMS struktury akcelerometru – (4)**

Nosníková struktura (Seismic mass) a pružiny se vyrábí leptáním polykrystalického křemíku (polysilicon), přičemž s postupným vývojem dochází k postupnému protahování pohyblivých elektrod ve směru kolmém na měřenou osu z dřívějších jednotek mikrometrů až na desítky mikrometrů. Díky tomu je možné několikanásobně protáhnout délku elektrod a tím získat lepší odstup signál/šum, menší křížovou citlivost (např. vliv zrychlení v ose  $X$  na osu  $Y$ ) a hlavně odezvu na změnu velikosti zrychlení. Do budoucna se již také počítá nahrazením polySi krystalickým křemíkem.[4]



**Obr. 1.06 – Detailní záběry na reálné provedení jednotlivých částí struktury - pevných a pohyblivých elektrod (electrodes), detekční hmoty (mass), pružiny (spring) a samotestovací strukturu (self test) – (4)**

Výše uvedená a popsaná struktura však umožňuje měření zrychlení jen v jednom směru kolmém na pohyblivé elektrody = 1D akcelerometry. Technologicky vcelku není problém na chipu přidat další stejnou strukturu pouze proti té předchozí pootočenou o 90°. Vznikne tak 2D akcelerometr, který například měří v osách XY nebo XZ, dle natočení senzoru. Složitější je vytvořit jednochipový 3D akcelerometr, protože se musí přidat výškově pohyblivá struktura v ose Z.

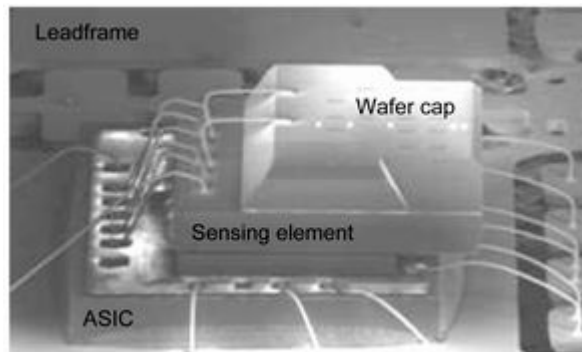


**Obr. 1.07 – 3D struktura akcelerometru (měření zrychlení v osách X, Y, Z) – (3)**

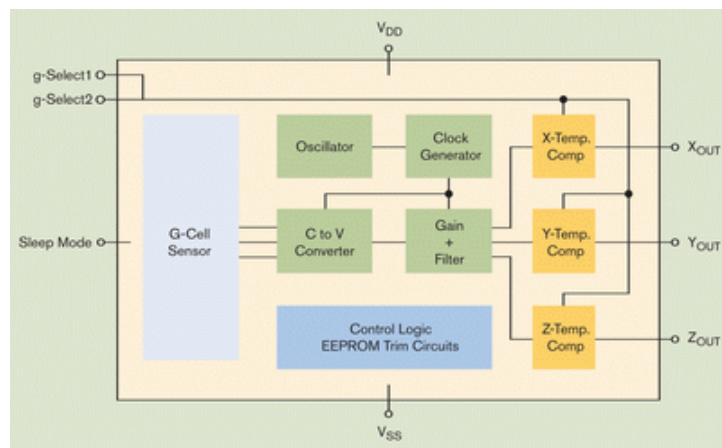
Zdokonalováním výrobního procesu není obtížné 3D strukturu realizovat a v současné době jsou tyto akcelerometry nabízeny s různými parametry od mnoha výrobců.

### 1.3.2. Elektrická struktura

Celá ASIC elektronika, která měří změnu kapacity, převádí ji na změnu napětí a zpracovává takto získaný signál na standardní lineární napět'ový výstup, je implantována pod snímacím elementem (Sensing element) překrytého krycí „kopulí“ (Wafer cap). K měření kapacity se využívá metody spínaných kondenzátorů řízené číslicovou logikou a generátorem spínacího hodinového signálu. Dochází tak k převodu změny kapacity na změnu napětí. To je následně linearizováno a filtrováno opět obvody se spínanými kondenzátory a nakonec se provádí kompenzace vlivu teploty. Výsledkem je lineární, zesílený a kompenzovaný napět'ový signál s definovanou převodní konstantou - citlivostí podávající informaci o kolik se musí změnit hodnota měřeného zrychlení, aby došlo ke změně výstupního napětí o 1 V (hodnota g/V). Vše se provádí zvlášť pro každou osu snímání (kanál).[4]



Obr. 1.08 – Struktura spojení snímače a ASIC elektroniky – (4)

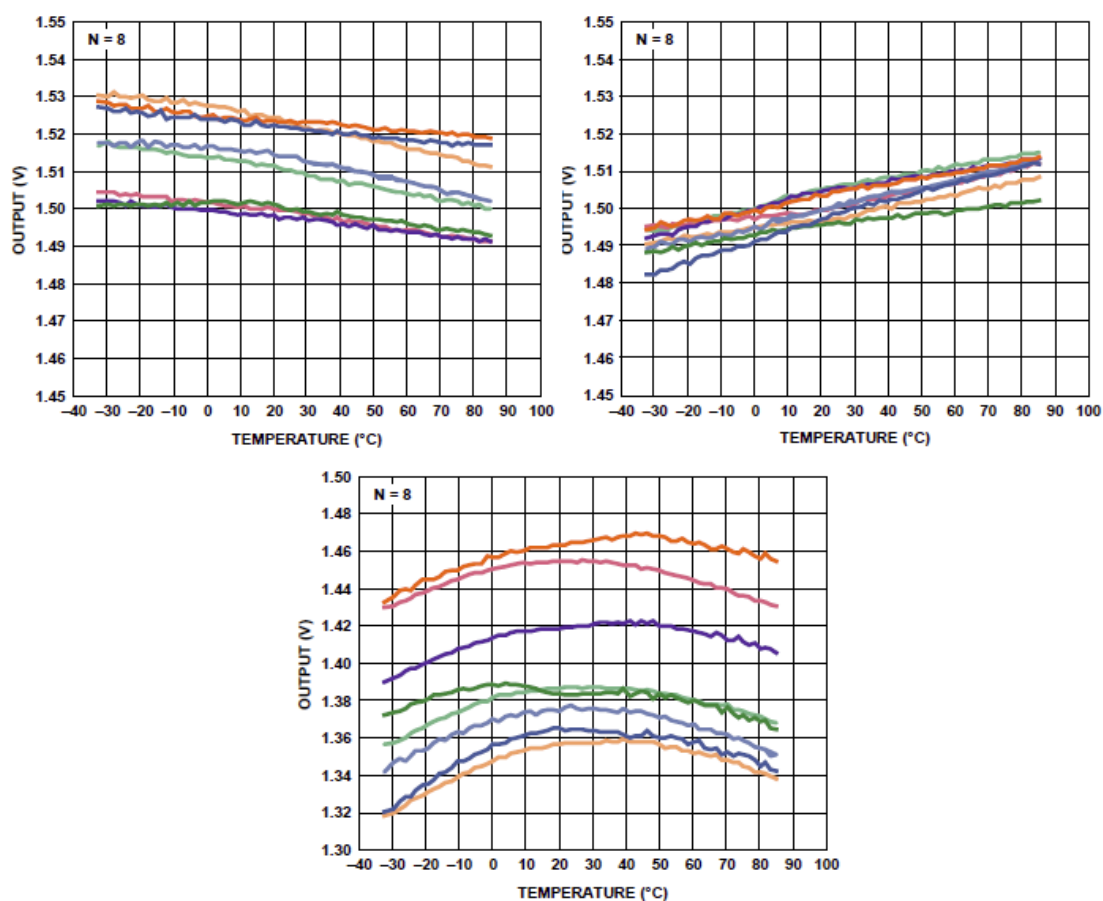


Obr. 1.09 – Blokové schéma elektrické struktury 3D akcelerometru – (4)

## 1.4. Závislost na teplotě

I přes všechna opatření kompenzující vliv při změnách teplot působící na akcelerometr dochází ke změnám parametrů, zejména se mění napěťová úroveň bodu  $0g$ , kdy není akcelerometr v pohybu v osách, ve kterých nepůsobí gravitační síla.

Grafy pod odstavcem (Obr. 1.10) znázorňují vliv teploty působící na jednotlivé osy ( $XYZ$ ). Pro každou osu je v grafu znázorněno osm křivek představující náhodný výběr z osmi desek plošných spojů se zapájenými akcelerometry stejného typu. Při konstantní teplotě, kolem pokojové ( $25\text{ }^{\circ}\text{C}$ ), a napětí pro A/D převodník  $3\text{V}$  je dle principu popsaném v kapitole (1.1.) očekávaná hodnota  $1,5\text{V}$  na výstupech z akcelerometru pro osy  $X$  a  $Y$ . V případě osy  $Z$  je tato hodnota v průměru o  $0,1\text{V}$  menší (konstrukční vlastnost).



Obr. 1.10 – Vliv teploty na nulové body „g“ v osách XYZ – (2)

Pro každou osu mají křivky různý trend a tato skutečnost představuje problém ve zpracování dat z akcelerometru. Vliv změn teplot působí na osy různě a při návrhu preciznějších systémů je nutné tuto vlastnost zahrnout. Odchylka od očekávaných hodnot v celém rozsahu provozních teplot není příliš veliká, ale při použití akcelerometru pro navigaci se chyba při jiné teplotě než je standardní pokojová lineárně ča-

sem zvyšuje, když vliv teploty není zahrnut do algoritmu výpočtu. Při návrhu plošného spoje je také nutné dbát, aby akcelerometr nebyl teplotně ovlivňován okolními aktivními, případně pasivními prvky nacházející se na desce.

### 1.5. Možnosti použití 3D MEMS akcelerometrů

- Měření sklonu či naklonění ploch a předmětů
- Monitorování pohybu objektů při přepravě
- Zabezpečovací zařízení
- Detekce a monitorování nárazů a vibrací
- Měření zrychlení
- Měření brždění
- Detekce a měření pádu
- Měření otřesů
- Měření a předpovídání seismické aktivity
- Trakční a bezpečnostní systémy automobilů



Obr. 1.11 – Použití akcelerometrů dle snímacího rozsahu – (4)

Volba citlivosti akcelerometru je závislá na prostředí, kde se bude využívat. Výrobci se snaží všemožně nalákat zákazníky na přidanou hodnotu jejich výrobků a v dnešní době není problém sehnat akcelerometr s nastavitelným rozsahem citlivosti od 1g až 16g. V prostředí, kde je potřeba zaznamenat sebemenší chvění, například při měření seismické aktivity, jsou používány akcelerometry s nízkým g od několika

tisícin do pár jednotek, jsou schopny poměrně přesně určit i sebemenší záchvěvy. Zařízení pro osobní užití (notebook, PDA, ...) v případech detekce pádu, změna natočení displeje dle orientace zařízení a podobně postačí velikost  $g$  v řádu jednotek podobně jako v prvním případě.

V případech, kde působí mnohem větší zrychlení a vibrace vyšší intenzity je zapotřebí použít akcelerometr s nižší citlivostí neboli vyšším  $g$ . Při měření nárazu, srážek automobilů se volí hodnoty kolem padesáti  $g$ . Pro tyto aplikace se vyrábí akcelerometry s velkým rozsahem nastavení citlivosti.

## 1.6. Základní dělení akcelerometrů

Tab. 1.01 – Základní dělení akcelerometrů

Měření	Typ akcelerometru
<b>síla nebo napětí</b>	<ul style="list-style-type: none"> <li>• piezoelektrický akcelerometr</li> <li>• piezorezistivní akcelerometr</li> <li>• rezonátory</li> </ul>
<b>posunutí</b>	<ul style="list-style-type: none"> <li>• potenciometrický akcelerometr</li> <li>• kapacitní akcelerometr</li> <li>• indukční akcelerometr</li> <li>• servo-řízený akcelerometr</li> <li>• optický akcelerometr</li> </ul>



## 2. Gyroskopy

Na počátku dvacátého století vznikaly první rádiové navigace. Použití této navigace znamenalo enormní rozvoj v určování polohy a námořnictví. Hrubé určení polohy, ke které sloužil kompas a lodní zápisky, se používaly po staletí. Teprve před několika desetiletími se začal používat gyrokompas (gyroskop), ten nahradil vše potřebné. Gyroskop sám prošel intenzivním vývojem. Schopnost rychlého rotačního pohybu stabilizovat a fixovat řízení v prostoru, byla známá už v 19. století. První experiment byl proveden už v roce 1865 díky využití pohonu elektrickým motorem. První gyrokompas patentoval Dr. H. Anschütz-Kämpfe roku 1904 a byl použit roku 1908.

Gyroskopy musí být schopny detekovat kolísající rotační rychlost od spojitě do  $100^\circ/\text{s}$ , to si žádá vyšší kvalitu výroby. Například posun o 2 nm mezi těžištěm a vektorem tlaku na něj působící vytvoří odchylku  $(10^{-9})^\circ \cdot h^{-1}$ . Tuto časovou nestabilitu je nutné co nejvíce omezit.

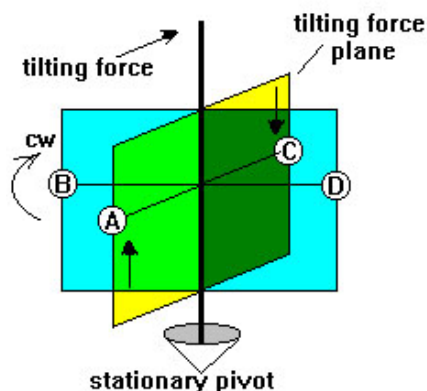
Pro lodě, jejichž plavba může trvat několik týdnů, se chyba akumuluje a je nutné provádět periodickou korekci polohy. V porovnání s letadly, kde let trvá jenom pár hodin nebo střela, která letí jen několik sekund, se tato akumulovaná chyba neuvažuje. Běžně má gyroskop odchylku  $(10^{-2})^\circ \cdot h^{-1}$ , to znamená chybu v rychlosti  $1,6 \text{ km} \cdot h^{-1}$ .

Inerciální navigace se vyvinula souběžně v posledních desetiletích, první zařízení s kombinací akcelerometru a gyroskopu byla vyrobena v roce 1923, první 3D zařízení vyrobena v roce 1924, první provozuschopné zařízení bylo vypuštěno v roce 1940, na raketě V2 a první výkonná stanice testovaná za letu ve Francii byla v roce 1961. Akcelerometry a gyroskopy jsou čím dál více oblíbenější a proto se používají v různých zařízeních.

V současnosti díky ceně, dostupnosti a stále lepší přesnosti není obtížné zkonstruovat vlastní systém pro lokální inerciální navigaci. Výhodou takového řešení je lepší přesnost, než jakou poskytuje systém GPS. Ke své funkci nepotřebuje signál ze satelitů, a proto nachází uplatnění i v prostorách budov. Při globální navigaci jde použít spíše jako doplněk z důvodu uvedených nežádoucích vlastností akcelerometru a gyroskopu. S pravidelnou korekcí pozice pomocí signálu GPS je toto použití vhodné například dojde-li ke ztrátě signálu z družic při vjetí dopravního prostředku do tunelu a v dalších podobných případech.

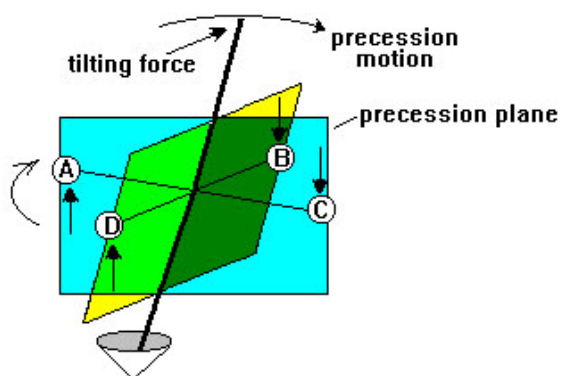
## 2.1. Princip funkce klasického rotačního gyroskopu

Obvykle mají gyroskopy tvar disku, ale může jím být jakýkoliv objekt, který má povahu setrvačníku. Součásti tvořící gyroskop by měly být ve stejné vzdálenosti od středu. Obvykle se používá tvar disku s velkou obručí. Když se gyroskop neotáčí, chová se jako každý jiný objekt, nicméně když se gyroskop uvede do pohybu na své vlastní ose vysokou rychlostí, odolává vlivu působení sil v určitých směrech.



Obr. 2.01 – Představa pro lepší porozumění principu gyroskopu – (5)

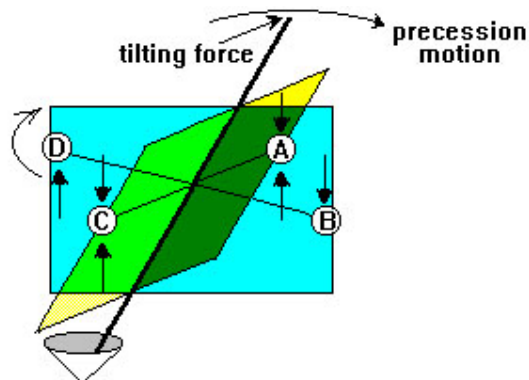
Namísto obruče slouží pro představu čtyři bodové síly A, B, C, D, které zastupují velmi důležité části pro představu funkce gyroskopu (Obr. 2.01). Dolní osa je stacionární a může se otáčet na čepu všemi směry. Když vychylovací síla (tilting force) působí shora na osu, bod A se pohybuje vzhůru a bod C opačným směrem (Obr. 2.02). Mezitím gyroskop rotuje podle směru hodinových ručiček, bod A se stává bodem B, když se gyroskop otočí o  $90^\circ$ . To samé se stane s bodem C a D.



Obr. 2.02 – Představa působení síly – (5)

Bod A se stále směřuje vzhůru, až se otočí na pozici o  $90^\circ$ , jako je vidět na (Obr. 2.02) a bod C směřuje dolů. Kombinace pohybů bodů A a C způsobuje rotaci osy v tzv. „precesní rovině“ (precession plane) doprava. Tento jev se nazývá precese.

Gyroskopická osa se pohybuje v pravém úhlu na rotační pohyb. V tomto případě doprava. Pokud se gyroskop otáčí proti směru hodinových ručiček, osa se bude pohybovat v precesní rovině doleva. Působí-li ve směru hodinových ručiček například vychylovací síla tahem místo tlakem, precese bude doleva.



Obr. 2.03 – Precesní pohyb – (5)

Když se gyroskop otočí o dalších  $90^\circ$  (Obr. 2.03), bod C je tam, kde byl bod A když vychylovací síla začala působit. Směřování bodu C dolů je nyní centrováno vychylovací silou a osa se nevychýlí. Čím silněji tlačí vychylovací síla na osu, tím více obruč z druhé strany tlačí osu zpět při opsání  $180^\circ$ .

Vlastnost precese gyroskopu se využívá k držení (monorail trains – metro nad zemí jedoucí po jedné úzké koleji) roviny při najíždění do zatáčky. Hydraulický válec působí silou (dle aktuálního stavu) na jednu osu mohutného gyroskopu. Občas je precese nežádoucí, když dva proti sobě se otáčející gyroskopy jsou použity na stejnou osu. Potom musí být použito výkyvné uložení.

Z poslední doby je známé použití gyroskopu jako rehabilitační či zábavný prostředek zvaný Powerball. Podstatou je roztočit v ruce gyroskop a čím rychleji se gyroskop točí, tím má větší odporovou sílu, která se pohybuje kolem 18 kg při otáčkách blížících se patnácti tisícům. Tímto cvičením se zpevní svaly ruky a zápěstí, což je velice důležité v dnešním světě počítačů, kdy lidé u nich tráví podstatnou část dne.

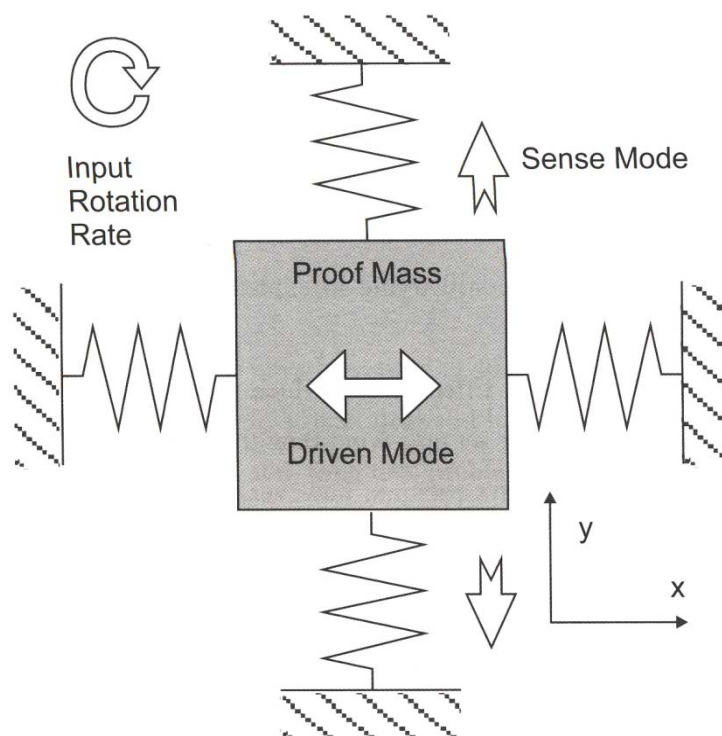
## 2.2. Dělení gyroskopů

Podle způsobu měření můžeme gyroskopy rozdělit do tří základních skupin:

- 1) **rotační gyroskopy** – představují nejstarší typy gyroskopů
- 2) **vibrační gyroskopy** – dnes jsou nejvíce používány a jejich vývoj stále pokračuje
- 3) **optické gyroskopy** – patří mezi dvě předcházející skupiny díky svým specifickým vlastnostem

## 2.3. Princip vibračního gyroskopu

Princip spočívá v elektrodynamickém závěsu na pružinový systém. U tohoto typu se nejedná o otáčivý pohyb, ale tělísko označené jako „Proof Mass“ vibruje dopředu a dozadu v postupném pohybu.



Obr. 2.04 – Kompozice vibračního poměru gyroskopu – (1)

Základní funkční princip všech vibračních gyroskopů vychází z detekce Coriolisova zrychlení. Jednoduchý vibrační gyroskop je složený z elektrodynamicky upevněných pružin, které dovolují pohyb ve dvou ortogonálních směrech. V gyroskopické dynamice je zřetelný pohyb všemi třemi směry ortogonálních os (Obr. 2.04)

Tělísko (proof-mass), je umístěno do oscilačního pohybu v ose  $X$  („drive axis“ – osa pohybu) paralelně se substrátem. Pohyb tělíška je citlivý na úhlové změny indukované v substrátu, když rotuje kolem osy  $Z$  kolmo na substrát. Vstupní rozsah indikuje Coriolisovo zrychlení v ose  $Y$  („sense axis“ – osa senzitivity), která je kolmá na osu  $X$  (osa pohybu) a osu  $Z$  (vstupní rozsah). Toto Coriolisovo zrychlení představuje Coriolisův pohybový rozkmit úměrný úhlové rychlosti substrátu. Detekce Coriolisova pohybu vyvolaná Coriolisovým zrychlením umožňuje odvodit původní rotaci gyroskopu.

Coriolisův efekt lze vysvětlit na praktické ukázce rezonátoru (případně testovacího prvku). Každá částice o hmotnosti ( $m$ ) je chápána jako elastická struktura. Když je částice uvedena do pohybu a pohybuje se kolem osy  $X$ , pohybuje se rychlostí rotace  $\Omega$ , potom síla  $F$  (2) známá jako Coriolisova je generována podél osy  $Y$ , kolmá na osu  $X$ [1].

$$F = 2 \cdot m \cdot v \cdot \Omega \cdot \cos(\omega \cdot t) [N] \quad (2)$$

$F$  – Coriolisova síla

$m$  – hmotnost tělesa

$v$  – rychlost pohybu v ose  $x$

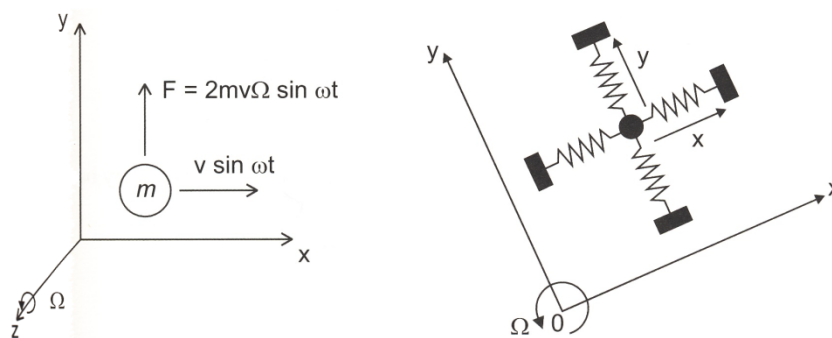
$\Omega$  – úhlová rychlost

$\omega$  – vibrační frekvence tělesa o hmotnosti  $m$

Výsledek vibrace má složky  $X$  a  $Y$ :

- vibrace ve směru  $X$
- vibrace související s Coriolisovou silou ve směru  $Y$

Měření následné vibrace (podél osy  $Y$ ) je použito na určení počtu otáček.



Obr. 2.05 – Coriolisova síla  $F$  – (1)

### 3. Získávání dat

Senzor převádí měřenou veličinu na elektrické napětí. Toto napětí je z výstupu dále převedeno z analogové formy na číselnou. Data v podobě čísel jsou zpracována počítačem nebo mikrokontrolerem. Každé takto provedené měření je zatíženo chybou nebo šumem. Problémem na straně senzoru je především měření veličiny, které je ovlivňované náhodnými chybami. Další nežádoucí vlastností je také vlastní chvění mikromechanizovaného systému snímání. V části za senzorem vzniká nejvyšší ovlivnění převodem dat z analogové formy na digitální, které doprovází kvantizační šum. Nelze zanedbat i okolní teplotu ovlivňující mechanickou část senzoru.

I přes všechna vylepšení senzorů eliminující výše uvedené problémy jsou data takto znehodnocena. Pro správnou interpretaci dat je nutné použití filtrace pro zlepšení odhadu měřené veličiny. Teprve správně vyfiltrovaná data jsou použitelná pro řízení daného systému. Možností jak znehodnocená data filtrovat je vícero. V této kapitole je několik způsobů popsáno od nejjednodušších po složitější. Data, která poslouží jako experimentální ukázka pro jednotlivé filtrace, byla získána zaznamenáním výstupu osy  $Z$  akcelerometru.

Myšlenku filtrace zpracovávaných dat nejlépe vystihuje následující citát: „Co neměříš, to neřídíš.“[6]

#### 3.1. Průměr

Nejjednodušší představu o hodnotě naměřených dat se získá použitím aritmetického průměru. Výpočet průměru vyjadřuje vztah (3.1):

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (3.1)$$

Hodnota  $n$  ve vzorci představuje počet prvků, ze kterých je tvořen průměr a  $x_i$  představuje  $i$ -tý prvek souboru naměřených dat.

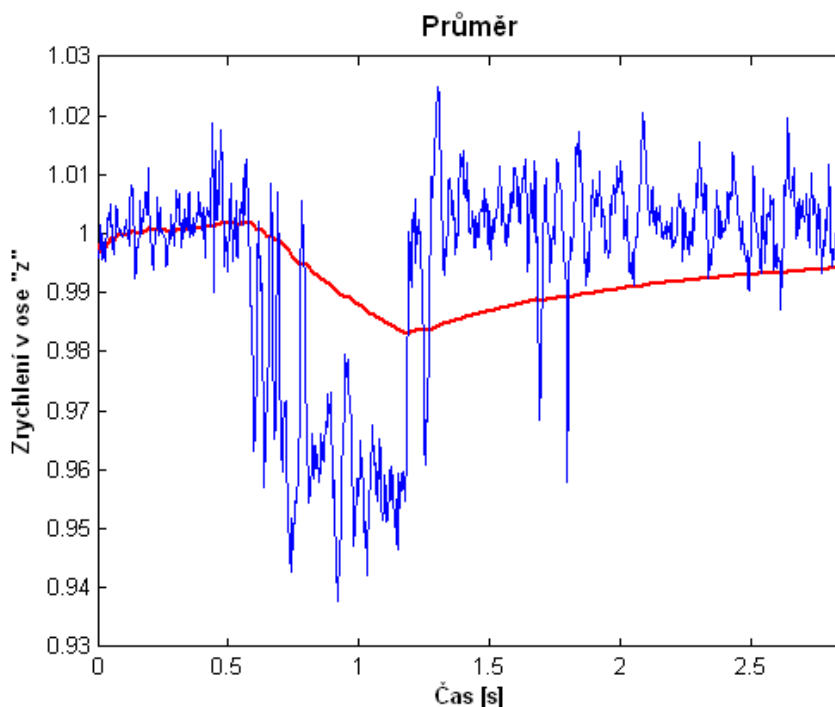
K výpočtu průměru dle uvedeného vztahu je potřeba znát celý soubor dat, což může představovat při vysokých hodnotách  $n$  velký problém při uchování všech dat. Lze ale průměr počítat rekurzivním způsobem dle vztahu (3.2):

$$\bar{x}_{n+1} = \frac{\bar{x}_n \cdot (n - 1) + x[n]}{n} \quad (3.2)$$

Veličina  $\bar{x}_n$  představuje hodnotu minulého průměru,  $x[n]$  je aktuální vzorek signálu. Vzorec lze upravit na jednodušší vztah (3.3):

$$\bar{x}_{n+1} = \bar{x}_n + \frac{x[n] - \bar{x}_n}{n} \quad (3.3)$$

Tímto postupem se docílí menší paměťové nároky. Nyní si stačí pamatovat hodnotu předchozího průměru a aktuální hodnotu vzorku sledovaného průběhu. Grafické znázornění aritmetického průměru:



Obr. 3.01 – Grafické znázornění průměru

Ze získaného průběhu je možné vidět, že obyčejný aritmetický průměr, který se s každou novou hodnotou akumuluje, neposkytuje uspokojivé výsledky. Při náhlé změně hodnoty není schopen pružně reagovat na změnu veličiny a výsledný průběh neodpovídá vůbec realitě. Průměr je vhodné použít spíše tam, kde máme jistotu, že je měřená hodnota konstantní, to ale není případ pro řešený problém.

### 3.2. Klouzavý průměr

Hlavní myšlenkou klouzavého průměru je postupné průměrování několika posledních hodnot. Neprůměruje se celý souboru dat, jako je to u aritmetického průměru. Touto metodou výpočtu je možné postihnout změnu ve sledovaném průběhu.

### 3.2.1. Klouzavý průměr – jednoduchý (SMA)

Pro jeho výpočet jsou známy všechny data souboru několika posledních hodnot o velikosti  $k$ . Výpočet hodnoty klouzavého průměru udává vztah (3.4):

$$\bar{x}_p[n] = \frac{1}{k} \sum_{i=n-k}^{n-1} x_i \quad (3.4)$$

Hodnota  $n$  představuje aktuální pozici prvku, který je počítán. Hodnota  $k$  udává velikost průměrovacího okna a  $x_i$  představuje  $i$ -tý prvek souboru dat v průměrovacím okně. Délka a složitost výpočtu jednoho souboru dat o velikosti  $k$  je lineárně závislá na hodnotě  $k$ . I v tomto případě lze výpočet přepsat na rekurzivní formu, na vztah (3.5):

$$\bar{x}_{p+1}[n] = \frac{\bar{x}_p \cdot k + x[n] - x[n-k]}{k} \quad (3.5)$$

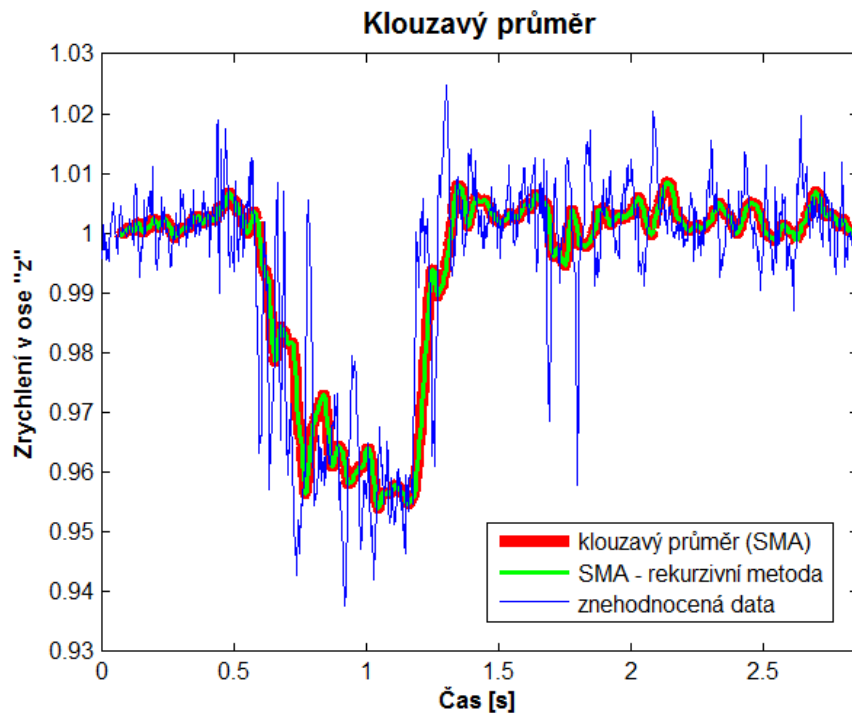
Ve vztahu se vyskytuje výraz  $\bar{x}_p \cdot k$ , který představuje součet posledních  $k$  hodnot klouzavého průměru v minulém kroku. Dále vztah obsahuje hodnotu  $x[n]$  představující aktuální vzorek  $n$  sledovaného průběhu (hodnota na konci průměrovacího okna) a hodnota  $x[n-k]$  představuje hodnotu průběhu posunutou o index  $k$  doleva (hodnota na začátku průměrovacího okna). Vzorec lze i zde zjednodušit na tvar (3.6):

$$\bar{x}_{p+1}[n] = \bar{x}_p + \frac{x[n] - x[n-k]}{k} \quad (3.6)$$

Použitím vztahu pro rekurzivní výpočet klouzavého průměru bylo dosaženo snížení náročnosti výpočtu. Výpočet jedné hodnoty souboru  $k$  hodnot se provede v jenom interačním kroku.



Grafické znázornění jednoduchého klouzavého průměru:



Obr. 3.02 – Grafické znázornění klouzavého průměru pro  $k=20$  (SMA)

Průběh znázorňuje aplikaci klouzavého průměru na znehodnocená data. Jeho křivka nyní mnohem lépe popisuje získaná data. Na grafu jsou zaznamenány obě metody výpočtu (klasická i rekurzivní), obě podávají stejné výsledky. U rekurzivní metody je důležitý první odhad průměru, který je spočítán stejným způsobem jako je tomu u klasické metody výpočtu, tedy průměr prvních  $k$  vzorků.

### 3.2.2. Klouzavý průměr – exponenciální (EMA)

Pokud není možné, například z paměťového hlediska, nebo si pozorovatel nechce pamatovat všechny vzorky z aktuálního průměrovacího okna, je nutné tyto vzorky nějak odhadnout. Nejlepší se odhad získá, když se použije aktuální hodnota zjišťovaného průměru. Vztah pro výpočet přejde do tvaru (3.7):

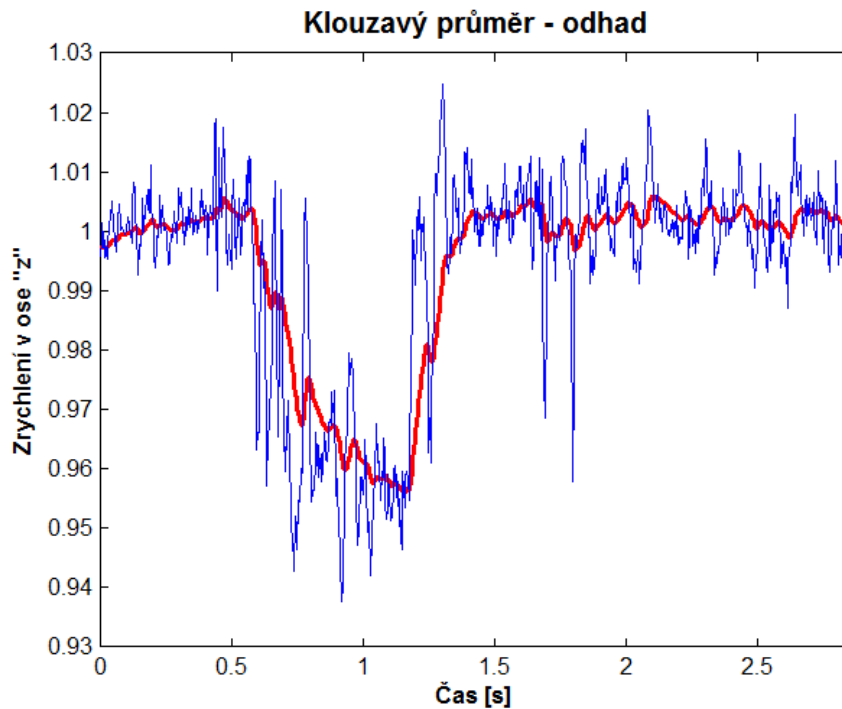
$$\bar{x}_{p+1}[n] = \frac{\bar{x}_p \cdot k + x[n] - \bar{x}_p}{k} \quad (3.7)$$

Lze upravit na přehlednější tvar (3.8):

$$\bar{x}_{p+1}[n] = \bar{x}_p + \frac{x[n] - \bar{x}_p}{k} \quad (3.8)$$

Pro tento způsob výpočtu si opět stačí pamatovat aktuální vzorek  $x[n]$  a hodnotu dosavadního průměru  $\bar{x}_p$ .

Grafické znázornění exponenciálního klouzavého průměru:



Obr. 3.03 – Grafické znázornění klouzavého průměru (EMA)

V porovnání s jednoduchým klouzavým průměrem bylo zjištěno, že průběh je více vyhlazen a bylo dosaženo lepších výsledků při menší paměťové náročnosti, nedochází ani k opoždění o  $k$  vzorků. Ve vztahu se vyskytují jen aktuální známé hodnoty. Použitím aktuálního průměru, při výpočtu, místo prvního prvku ze souboru  $k$  hodnot způsobí, že počítaný průměr obsahuje hodnotu měření v bodě  $x[0]$ . Tato hodnota má váhu (3.9):

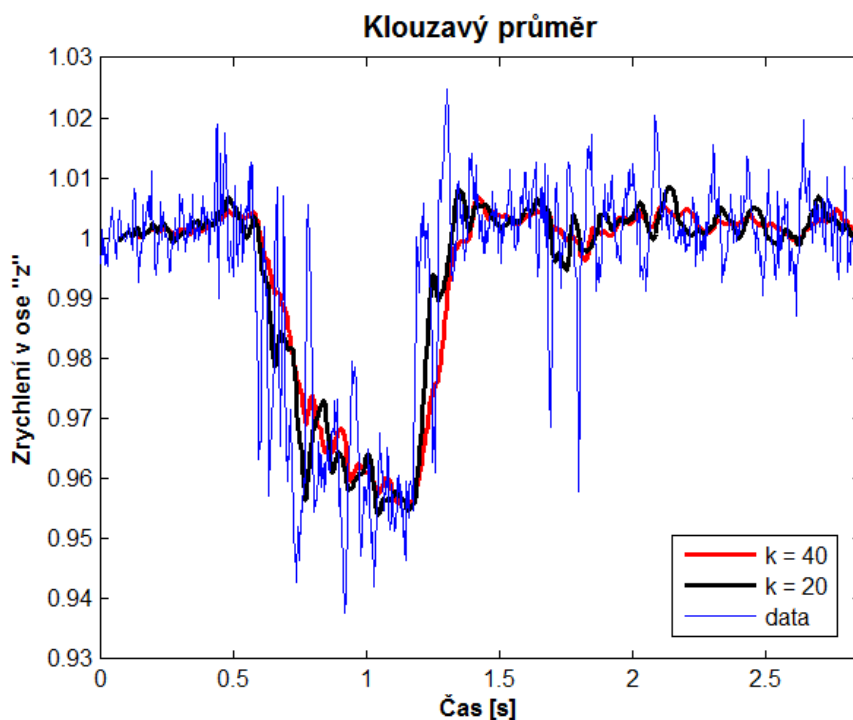
$$\left(\frac{k-1}{k}\right)^n \quad (3.9)$$

Z tohoto důvodu se tento výpočet klouzavého průměru nazývá exponenciálním. U tohoto průměru není ani nutné zajistit přesnější odhad první hodnoty průměru, protože výpočet je schopen (oproti výpočtu klasického klouzavého průměru) dokonkovat na správnou hodnotu.

### 3.3. Problémy výpočtu klouzavých průměrů

Z předešlých vztahů a grafů je zřejmá závislost na hodnotě  $k$ . S větší hodnotou  $k$  lze dosáhnout většího vyhlazení, ale vzniká problém pomalejší odezvy na změnu sledovaného průběhu. Na následujícím grafu je tento jev znázorněn.

Grafické znázornění klouzavého průměru pro různá  $k$ :

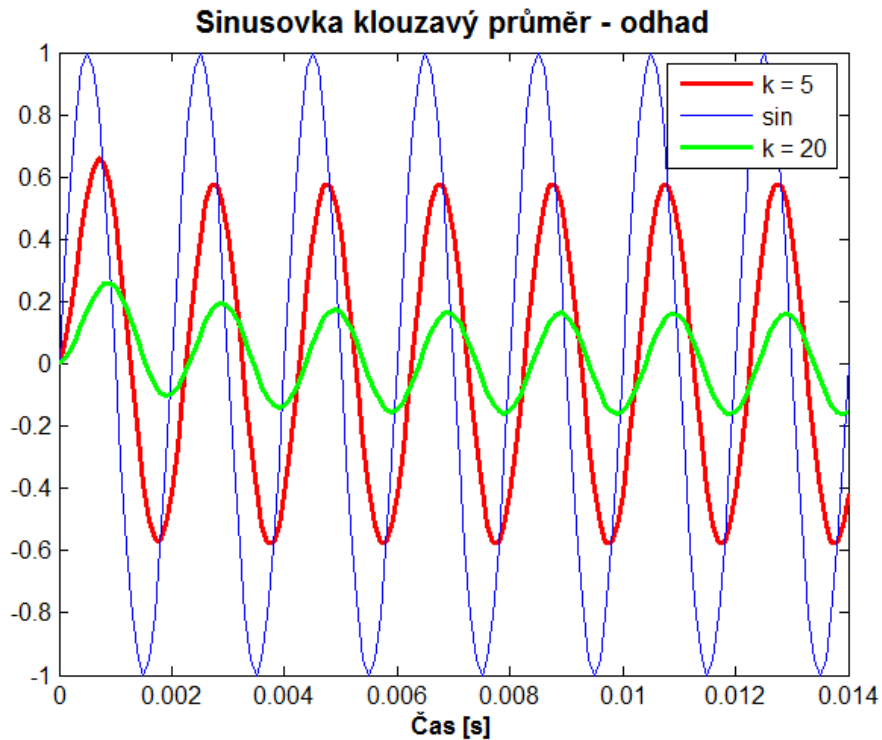


Obr. 3.04 – Grafické znázornění klouzavého průměru pro různá  $k$

Černý průběh označuje krok  $k = 20$ , je více zvlněný, ale stíhá rychleji reagovat na změny ve sledovaných datech. Oproti tomu červenou barvou je znázorněn krok  $k = 40$ , zde se chová naopak oproti černému průběhu. Volba  $k$  závisí na konkrétní aplikaci a požadavcích, které jsou kladeny na systém.

Podstatnější problém je zpoždění a zkreslení signálu rychle se měnícího. Závisí značně na  $k$ , jak ukazuje další graf.

Grafické znázornění klouzavého průměru pro rychle měnící se data:



Obr. 3.05 – Grafické znázornění rychle se měnícího průběhu

Tato nežádoucí vlastnost je způsobena předpokladem plovoucího průměru, který počítá s konstantností hodnoty pro celou délku okna a tím vzniká nežádoucí zpoždění.

K odstranění problému je nutné použít složitějšího modelu odhadu, kde by se hodnota odhadu parametru měnila. Například lze použít model přímky, která má rovnici (3.10):

$$y = ax + b \quad (3.10)$$

U této rovnice by se odhadovaly parametry dva a to parametr  $a$  a  $b$ .

### 3.4. Kalmanův filtr

Vynálezce Rudolf Emil Kalman se narodil v Budapešti, 19. 5. 1930. Myšlenka Kalmanova filtru vznikla v roce 1958. V letech 1960 a 1961 publikoval články o Kalmanově filtru a ihned způsobil revoluci v teorii odhadu.

Jedná se o rekurzivní filtr (tzn., vychází z předchozího stavu). Kroky Kalmanova filtru jsou predikce v následujícím čase a korekce v následujícím čase. Dynamický systém bývá zatížen šumem nejčastěji tzv. „bílým“. Ke zlepšení odhadu filtru je použito, jak měření související se stavem, tak i nesouvisející. Zjednodušeně je možné považovat Kalmanův filtr za „vylepšený“ odhad plovoucího průměru.[6]

#### 3.4.1. Základní části Kalmanova filtru:

##### 3.4.1.1. Stavový vektor

Představuje popis stavu dynamického systému (stupně volnosti). Proměnné stavového vektoru nelze měřit přímo, ale mohou být odvozeny z měřených hodnot. Elementy stavového vektoru tvoří např. pozice, rychlost, orientační úhel a podobně. Jednoduchým příkladem může být vlak jedoucí konstantní rychlostí po přímé koleji. V případě vlaku se jedná o dva stupně volnosti a to dosažená vzdálenost „s“ a rychlost jakou se pohybuje „v“:

$$\underline{x} = \begin{bmatrix} s \\ v \end{bmatrix} \quad (3.4.1)$$

Stav je popsán v časových intervalech dvěma hodnotami a to současný „a priori“ (před úpravou) a hodnoty po úpravě (korekci) „a posteriori“. Součastné hodnoty se označují  $x^-$ , následující  $x^+$

##### 3.4.1.2. Dynamický model

Popisuje transformace stavového vektoru v čase. Obvykle bývá popsán diferenciálními rovnicemi.

$$\dot{\underline{x}}(t) = \frac{d\underline{x}(t)}{dt} = f(\underline{x}(t), \underline{m}(t)) \quad (3.4.2)$$

Pro lineární systém lze přepsat na:

$$\dot{\underline{x}} = F \cdot \underline{x}(t) + \underline{n}(t) \quad (3.4.3)$$

$F$  je dynamická matice a je konstantní,  $x(t)$  je stavový vektor a  $n(t)$  je dynamický šum, který je chápán jako bílý šum a má kovarianční matici  $Q(t)$ .

Existují systémy, jež nelze popsat diferenciálními rovnicemi, ale takové případy nejsou předmětem zkoumání.

### 3.4.1.3. Model pozorování

Představuje vztah mezi stavem a měřením. V lin. systémech lze popsat měření systémem lineárních rovnic, které závisí na stavových veličinách. Pozorování se obvykle děje v diskrétních časech  $t_i$ :

$$\underline{l}(i_i) = h(\underline{x}(t_i), \underline{v}(t_i)) \quad (3.4.4)$$

Vektor takového systému je:

$$\underline{l}(i_i) = H \cdot \underline{x}(t_i) + \underline{v}(t_i) \quad (3.4.5)$$

$l(t_i)$  – vektor pozorování v časech  $i$ ,  $H$  – matice pozorování (v lin. sys. konstantní),  $w_i$  – šum měřicího procesu s kovarianční maticí  $R(t_i)$ .

## 3.4.2. Algoritmus Kalmanova filtru:

### 3.4.2.1. Predikce

Jak již bylo zmíněno, predikce je první krok Kalmanova filtru. Předpovídaný stav, lépe současný, je počítán se zanedbáním šumu dynamického systému tedy:

$$\underline{\dot{x}}^- = F \cdot \underline{x}^-(t) \quad (3.4.6)$$

Stavový vektor v čase  $t$  lze vyjádřit Taylorovým polynomem ve snaze se přiblížit stavu  $\underline{x}^-(t_0)$ :

$$\underline{x}^-(t) = \underline{x}^-(t_0) + \underline{\dot{x}}^-(t_0)(t - t_0) + \frac{1}{2}\underline{\ddot{x}}^-(t_0)(t - t_0)^2 + \dots \quad (3.4.7)$$

Pomocí předchozí rovnice (3.4.6) lze přepsat vztah jako:

$$\underline{x}^-(t) = \underline{x}^-(t_0) + F \cdot \underline{x}^-(t_0)(t - t_0) + \frac{1}{2}F^2 \underline{x}^-(t_0) \cdot (t - t_0)^2 + \dots \quad (3.4.8)$$

Tedy vztah  $\underline{x}^-(t)$  z diferenciálních rovnic. Jinými slovy aktuální předpovídaný stav je lineární kombinace původního stavu  $\underline{x}^-(t_0)$ .

$$\underline{x}^-(t) = \Phi_0^t \cdot \underline{x}^-(t_0) \quad (3.4.9)$$

$\Phi_0^t$  je nazývána *stavovou přechodovou maticí*, která transformuje jakýkoliv počáteční stav  $\underline{x}(t_0)$  do odpovídajícího stavu  $\underline{x}(t)$  v čase  $t$ .

Z rovnic (3.4.6) a (3.4.9):

$$\dot{\underline{x}}^- = F \cdot \underline{x}^-(t) = F \cdot \Phi_0^t \cdot \underline{x}^-(t_0) \quad (3.4.10)$$

Opětovným použitím (3.4.9) se vztah upraví na:

$$\dot{\underline{x}}^-(t) = \frac{d}{dt} \cdot \underline{x}^-(t) = \frac{d}{dt} \cdot [\Phi_0^t \cdot \underline{x}^-(t_0)] = \left[ \frac{d}{dt} \cdot \Phi_0^t \right] \cdot \underline{x}^-(t_0) \quad (3.4.11)$$

Z porovnání (3.4.10) a (3.4.11) je patrné:

$$\frac{d}{dt} \cdot \Phi_0^t = F \cdot \Phi_0^t \quad (3.4.12)$$

s výchozí maticí  $\Phi_0^0 = I$ , protože  $\underline{x}(t_0) = I \cdot \underline{x}(t_0)$ .

Dále kovarianční matice  $P^-(t_i)$  předpovídaného vektoru stavu je získána pomocí zákona přenosu chyb:

$$P^-(t_i) = \Phi_{t_{i-1}}^{t_i} \cdot P(t_{i-1}) \cdot (\Phi_{t_{i-1}}^{t_i})^T + Q \quad (3.4.13)$$

V obecnější formě je kovarianční matice šumu  $Q$  funkcí času, poté se vyjádří takto.

$$P^-(t_i) = \Phi_{t_{i-1}}^{t_i} \cdot P(t_{i-1}) \cdot (\Phi_{t_{i-1}}^{t_i})^T + \int_{t_{i-1}}^{t_i} Q(t) dt \quad (3.4.14)$$

### 3.4.2.2. Korekce

V korekčním kroku předpovídaný stavový vektor  $\underline{x}^-(t_i)$  je „vylepšen“ pozorováním v čase  $t_i$  takže stav po odkončení má formu:

$$\underline{x}^+(t_i) = \underline{x}^-(t_i) + \Delta \cdot \underline{x}(t_i) \quad (3.4.15)$$

s kovarianční maticí:

$$\underline{P}^+(t_i) = \underline{P}^-(t_i) + \Delta \cdot \underline{P}(t_i) \quad (3.4.16)$$

Kalmanův filtr je optimalizovaný filtr, to znamená, že stavové veličiny ve stavové kovarianční matici  $P^+$  jsou minimalizovány. Jelikož je  $P^-$  známé z predikčního kroku, z toho vyplývá, že  $\Delta P$  je minimalizované.

$$\Delta P(t_i) = E[\Delta \underline{x}(t_i) \Delta \underline{x}(t_i)^T] \quad (3.4.17)$$

Tato podmínka vyhoví při:

$$\begin{aligned} \Delta \underline{x}(t_i) &= P^- H^T (HP^- H^T + R(t_i))^{-1} \cdot (\underline{l}(t_i) - H \cdot \underline{x}^-(t_i)) \\ \Rightarrow \Delta \underline{x}(t_i) &= K(t_i) \cdot (\underline{l}(t_i) - \underline{l}^-(t_i)) \end{aligned}$$

s

$$K(t) = P^- H^T (HP^- H^T + R(t_i))^{-1} \quad (3.4.18)$$

$K$  se nazývá *matice zesílení*. Výraz  $(\underline{l}(t_i) - \underline{l}^-(t_i))$  představuje reziduum měření.

Udává rozdíl mezi predikovaným měřením  $\underline{l}^-(t_i) = H \underline{x}^-(t_i)$  a aktuálním měřením  $\underline{l}(t_i)$ .

Konečná úprava stavu je:

$$\underline{x}^+(t_i) = \underline{x}^-(t_i) + K(t_i) \cdot (\underline{l}(t_i) - H \cdot \underline{x}^-(t_i)) \quad (3.4.19)$$

V této rovnici jsou odhadovaný stav a měření váhovány. Zesílení udává, kolik procent z vypočítané predikce se použije na úpravu stavu. Použitím 100% korekce znamená, že filtr není aktivní a proto musí mít Kalmanovo zesílení hodnotu menší jak 1. Kovariance měřených hodnot je menší než u predikovaných. Větší váhu při výpočtu mají právě měřená data, menší predikované stavy, a tak je redukována odchylka.

Kovarianční matice stavu po je vyjádřena zákonem přenosu chyb:

$$P^+(t_i) = P^-(t_i) - K(t_i) H P^-(t_i) = (I - K(t_i) H) P^-(t_i) \quad (3.4.20)$$

### 3.5. Rozšířený Kalmanův filtr

Prozatím byly uvažovány jen lineární systémy, ale v praxi se vyskytují i nelineární. Kalmanův filtr pro nelineární systémy má název Extended Kalman Filter (EKF). Filtr linearizuje odhadovaný stav, to znamená, že systém musí mít spojitě diferencovatelné funkce.

Nevýhodou tohoto filtru je časově náročnější výpočet. Implementace pro lineární systém je jednodušší a efektivnější, když je předpočítána dynamická matice  $F$ , stavová přechodová matice a pozorovací matice  $H$ . V případě nelineárních systémů se tyto matice mění v každém časovém kroku a nelze je považovat za konstantní hodnoty.



### 3.5.1. Predikce

Pro nelineární systémy je dynamická matice  $F$  funkcí stavu a musí se odhadnout. Předpokládaný stav je výsledkem řešení diferenciálních rovnic ve tvaru:

$$\dot{\underline{x}}^- = f(\underline{x}^-(t)) \quad (3.5.1)$$

Vyjádření této rovnice Taylorovou řadou v okolí  $x$  jako predikovaného stavu  $\underline{x}^-(t_i)$  a zanedbáním podmínky vyššího řádu, lze poté dynamickou matici  $F$  vypočítat takto:

$$F(t_i) = \left. \frac{\partial f(\underline{x})}{\partial \underline{x}} \right|_{\underline{x}=\underline{x}^-(t_i)} \quad (3.5.2)$$

Následující kroky se počítají stejně jako u klasického Kalmanova filtru dle vztahů (3.4.12) a (3.4.14), ale nemohou být takto vyjádřeny. V tomto případě nejsou matice konstantní, ale jsou závislé na časovém kroku:

$$\frac{d}{dt} \cdot \Phi_{t_{i-1}}^{t_i} = F \cdot \Phi_{t_{i-1}}^{t_i} \quad (3.5.3)$$

$$P^-(t_i) = \Phi_{t_{i-1}}^{t_i} \cdot P(t_{i-1}) \cdot (\Phi_{t_{i-1}}^{t_i})^T + \int_{t_{i-1}}^{t_i} Q(t) dt \quad (3.5.4)$$

### 3.5.2. Korekce

Tak jako diferenciální rovnice v predikčním kroku odpovídají nelineárním, je také nutná linearizace Taylorovou řadou v okolí predikovaného stavu  $\underline{x}^-(t_i)$  a vyšší řády jsou i v tomto případě zanedbány. Aproximační matice se získá:

$$H(t_i) = \left. \frac{\partial h(\underline{x})}{\partial \underline{x}} \right|_{\underline{x}=\underline{x}^-(t_i)} \quad (3.5.5)$$

V případě předpokládaného měření  $\underline{l}^-(t_i)$  pro výpočet rozdílu je  $\underline{l}^-(t_i) = h(\underline{x}^-(t_i))$

Pro další výpočty platí stejné vztahy jako v případě klasického Kalmanova filtru s rozdílem, že jsou opět závislé na čase:

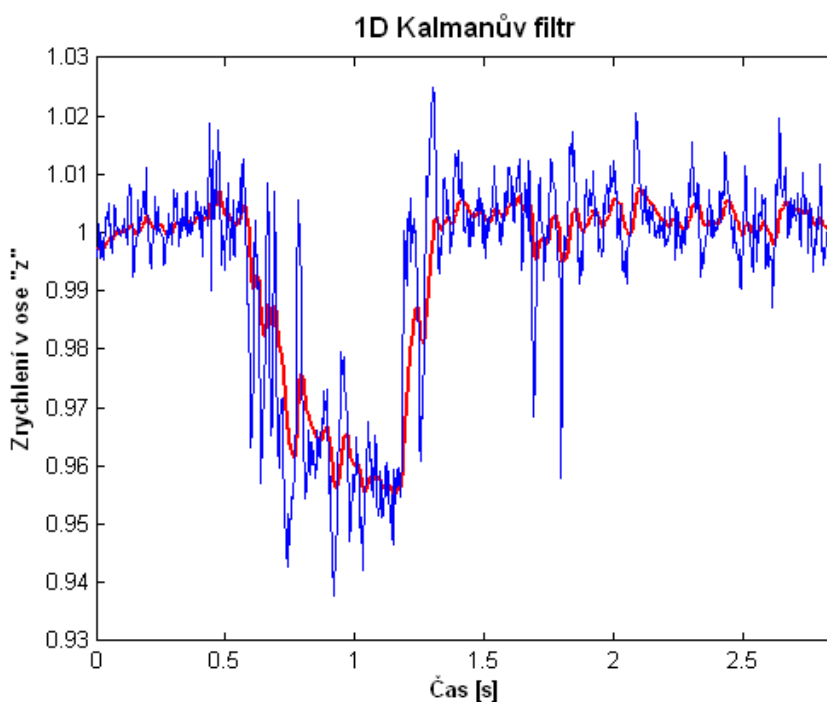
$$\underline{x}^+(t_i) = \underline{x}^-(t_i) + K \left( l(i) - H \cdot \underline{x}^-(t_i) \right)$$

$$P^+(t_i) = (I - K(t_i)H(t_i))P^-(t_i)$$

s

$$K(t_i) = P^-H(t_i)^T (H(t_i)P^-H(t_i)^T + R(t_i))^{-1}$$

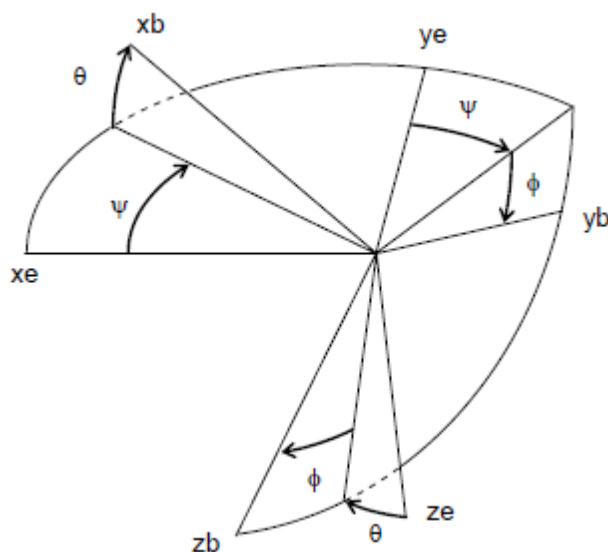
Grafické znázornění jednoduchého Kalmanova filtru na signál obsahující šum:



Obr. 3.06 – Grafické znázornění jednoduchého Kalmanova filtru

## 4. Orientace v prostoru

Pro popis létajícího objektu je nutné definovat souřadnicový systém. Ve většině případů se spojených s létáním se používají dva souřadnicové systémy. Jedním z nich je souřadnicový systém fixně vzhledem k Zemi a může posloužit k posouzení analýzy letu inerciálními veličinami. Ostatní souřadnicové systémy jsou vázány na konstrukci letícího objektu. Následující obrázek (Obr. 4.01) znázorňuje oba souřadnicové systémy, jak vůči Zemi (symboly  $x_e$ ,  $y_e$  a  $z_e$ ), tak souřadnicový systém vycházející z konstrukce letícího objektu (symboly  $x_b$ ,  $y_b$  a  $z_b$ ).



Obr. 4.01 – Znázornění souřadnicových systémů (10)

Orientace objektu v prostoru je nejčastěji popsána třemi následujícími natočeními kolem os, jejichž uspořádání je podstatné. Úhlové natočení jsou tzv. Eulerovy úhly. Orientace konstrukce objektu vzhledem k Zemi si lze představit pomocí následujících natočení:

- 1) Natočení konstrukce vzhledem k ose  $z_b$  se nazývá yaw a označení má  $\Psi$
- 2) Natočení konstrukce vzhledem k ose  $y_b$  se nazývá pitch a označení má  $\Theta$
- 3) Natočení konstrukce vzhledem k ose  $x_b$  se nazývá roll a označení má  $\phi$

$$\text{pitch}(\theta) = \frac{180}{\pi} \cdot \text{atan}\left(\frac{\text{akcelerace}Y}{\text{akcelerace}Z}\right) [^\circ] \quad (4.1)$$

$$\text{roll}(\phi) = \frac{180}{\pi} \cdot \text{atan}\left(\frac{\text{akcelerace}X}{\text{akcelerace}Z}\right) [^\circ] \quad (4.2)$$

Další možností, jak určit úhly náklonu, je využití integrace dat z gyroskopu dle následujících vztahů:

$$\theta = \theta + gyroY \cdot \Delta t [^\circ] \quad (4.3)$$

$$\Phi = \Phi + gyroX \cdot \Delta t [^\circ] \quad (4.4)$$

Data získaná pomocí gyroskopu udávají úhlovou rychlost. Časový okamžik  $\Delta t$  je časový odstup dvou vzorků a dle známého vztahu pro výpočet dráhy (4.5):

$$s = v \cdot t [m] \quad (4.5)$$

je možné podobně spočítat úhly. Vztahy (4.3) a (4.4) tedy popisují opsanou úhlovou dráhu.

## 5. Regulace

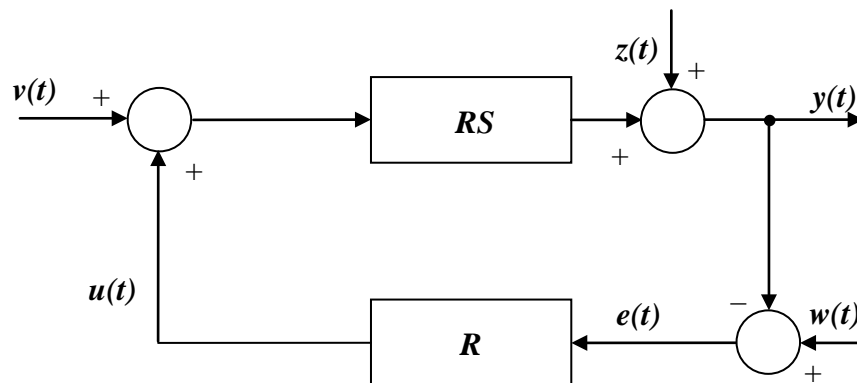
Pro stabilizaci polohy je důležitá věrohodnost dat ze senzorů. Použitím vhodné filtrace se zajistí jejich správná interpretace a data jsou připravená k dalšímu použití. K samotné stabilizaci polohy je zapotřebí regulace. Regulací se rozumí přinutit systém, aby se ustálil na požadované hodnotě, která je od systému vyžadována. V případě stabilizace polohy létajícího objektu je zapotřebí, aby objekt setrval na místě v rovnovážné poloze při působení vnějších vlivů okolního prostředí, v důsledku jeho nevyváženosti nebo různého silového tahu motorů.

Systém, který je zapotřebí regulovat se nazývá regulační soustava. Její chování lze popsat diferenciálními rovnicemi. U stavového modelu zápis v maticové podobě vypadá takto:

$$\frac{dx}{dt} = A \cdot x + B \cdot u \quad (5.1)$$

$$y = C \cdot x + D \cdot u \quad (5.2)$$

Regulace systému se vykonává za pomoci regulačního obvodu. Jeho obecné schéma je na dalším obrázku (Obr. 5.01):



Obr. 5.01 – Jednoduchý regulační obvod

Do regulované soustavy je přivedena vstupní hodnota  $v(t)$ . Cílem regulace je zajištění požadované výstupní hodnoty  $y(t)$  regulované soustavy  $RS$ . Výstupní hodnota je snímána a přivedena do rozdílového členu, kde se vyhodnotí rozdíl řídicí veličiny  $w(t)$  (požadovaná hodnota) a aktuální výstupní hodnoty. Při odlišných hodnotách vzniká rozdíl, který se nazývá regulační odchylka a má označení  $e(t)$ . S touto odchylkou poté pracuje regulátor  $R$ , který představuje hlavní člen regulačního obvodu. Podle velikosti odchylky se nastaví akční veličina  $u(t)$ , která působí na regulovanou soustavu a zajistí ustálení systému na požadované hodnotě.

Takto definovaný systém je ideální, ale v praxi je potřeba uvažovat, že bývá často ovlivněn vnějším, popřípadě vnitřními vlivy. Proto je do regulačního obvodu ještě zahrnuta poruchová veličina  $z(t)$  představující šum nebo jiné negativní stavy systému, jimiž může být i vlastní závada.

## 5.1. Dělení regulátorů

Regulátor bývá popsán podobným matematickým vyjádřením, jako je to u regulované soustavy (vztahy (5.1)(5.2)) s tím rozdílem, že regulovaná soustava má danou strukturu i parametry. Regulátor může mít strukturu i parametry libovolné a je možné je získat například výsledkem matematického řešení regulátoru nebo pozorováním regulovaného systému. Do základních skupin regulátorů řadíme:

- **P** – proporcionální regulátor
- **I** – integrační regulátor
- **D** – derivační regulátor

### 5.1.1. Proporcionální regulátor P

Zahrnuje v sobě jenom násobení konstantou, která se nazývá zesílení.

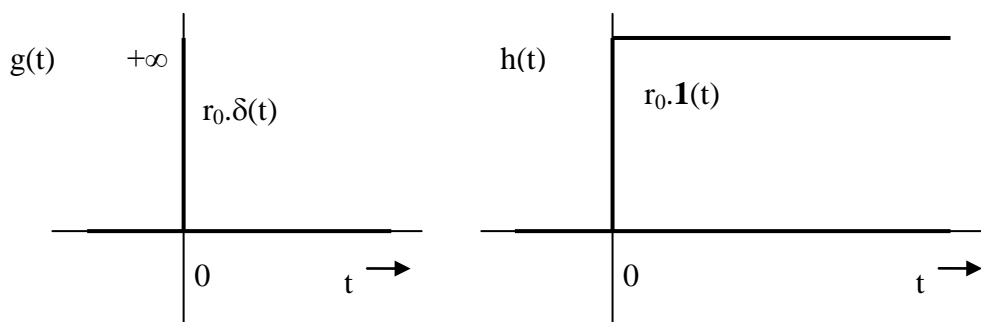
Přenosová funkce je dána:

$$R(s) = r_0$$

Akční veličina je vyjádřena jako:

$$u(t) = r_0 \cdot e(t)$$

Impulzní odezva  $g(t)$  a přenosová funkce  $h(t)$ :



Obr. 5.02 – Impulzní odezva a přenosová funkce regulátoru P (11)

Obsahuje-li regulační obvod dle (Obr. 5.01) pouze regulátor typu P libovolného řádu, nikdy nebude regulační odchylka zcela vyregulována. Její ustálená hodnota sice s proporcionálním zesílením regulátoru  $r_0$  klesá, ale nikdy nebude nulová. Navíc  $r_0$  nelze libovolně zvětšovat s ohledem na stabilitu obvodu.

### 5.1.2. Integrační regulátor I

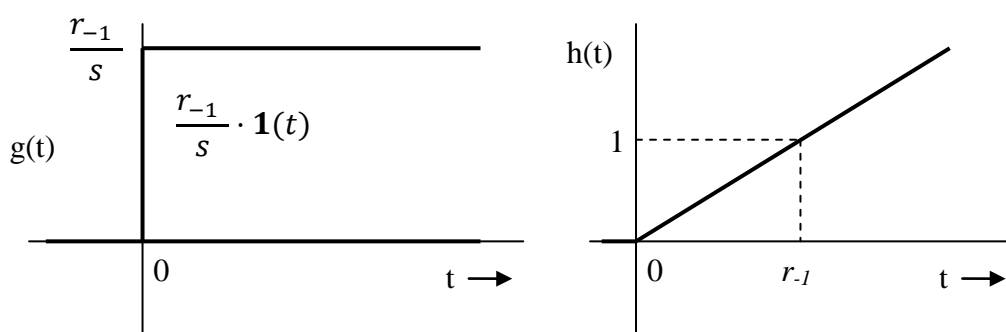
Přenosová funkce je dána:

$$R(s) = \frac{1}{r_{-1} \cdot s}$$

Akční veličina je vyjádřena jako:

$$u(t) = r_{-1} \cdot \int_0^t e(\tau) \cdot d\tau$$

Impulzní odezva  $g(t)$  a přenosová funkce  $h(t)$ :



Obr. 5.03 – Impulzní odezva a přenosová funkce regulátoru I (11)

Pro úplné odstranění regulační odchylky v obvodu dle (Obr.5.01) je zapotřebí regulátor, jehož výstupní signál (akční veličina) má integrační složku.  $I$  regulátor zcela odstraňuje regulační odchylku. Při návrhu je nutné dbát na velikost integrační konstanty. Při její špatné volbě by mohlo dojít k nestabilitě regulátoru.

### 5.1.3. Derivační regulátor D

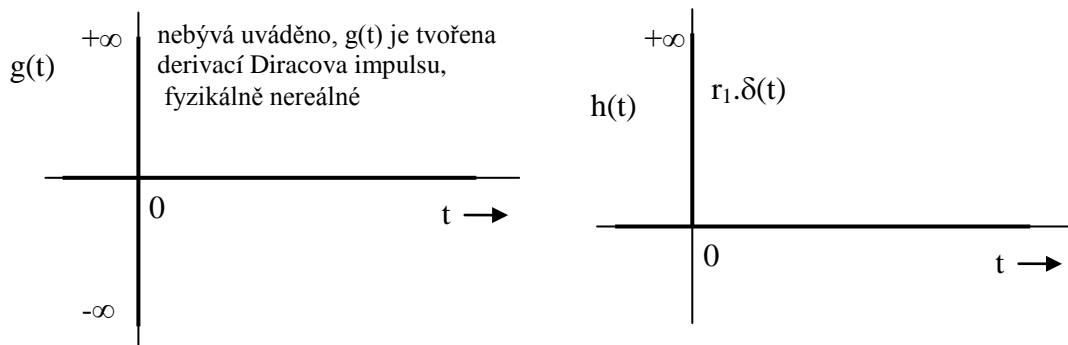
Přenosová funkce je dána:

$$R(s) = r_1 \cdot s$$

Akční veličina je vyjádřena jako:

$$u(t) = r_1 \cdot \frac{de(t)}{dt}$$

Impulzní odezva  $g(t)$  a přenosová funkce  $h(t)$ :



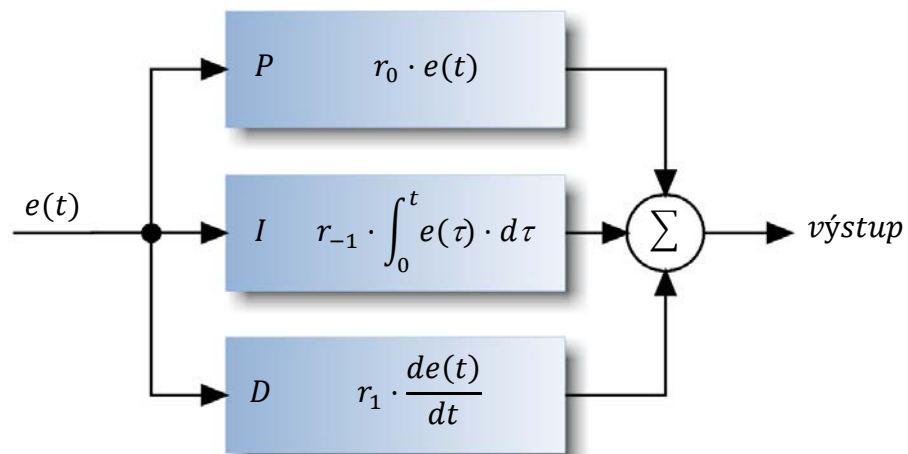
Obr. 5.04 – Impulzní odezva a přenosová funkce regulátoru (11)

U derivačního regulátoru nastává problém realizovatelnosti z důvodu nemožnosti derivovat Diracův impuls, proto derivační regulátor samostatně použit nelze, neboť reaguje pouze na derivaci, tj. na změny regulační odchylky a statickou hodnotu této veličiny neregistruje, není schopen ji odstranit.

#### 5.1.4. Kombinace P, I, D

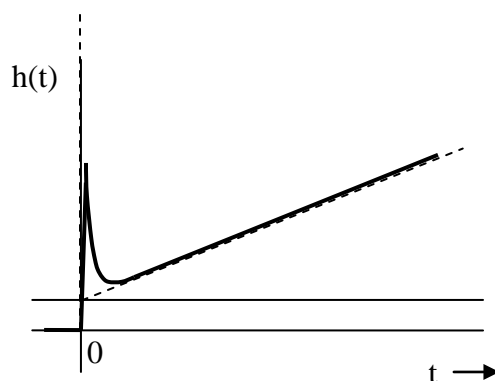
Z uvedených základních typů regulátoru v předchozím přehledu, tj.  $P$ ,  $I$ ,  $D$  se v praxi samostatně využívají dva typy regulátorů, jsou jimi regulátor  $P$  a  $I$ . V některých systémech takovéto řešení postačuje a poměrně jednoduše se realizují.

Pro zlepšení průběhů regulačních pochodů se velmi často používají tzv. kombinované regulátory, které vzniknou součinností regulátorů základních strukturálních typů. Do této skupiny patří regulátory typů  $PD$  (proporcionálně derivační),  $PI$  (proporcionálně integrační) a  $PID$  (proporcionálně integračně derivační). Derivační regulátor samostatně nepoužitelný se zde v kombinaci s ostatními typy efektivně využívá pro vytvoření derivační složky akčního signálu a tím i ke zkvalitnění procesu regulace (příznivější přechodový děj). Blokové schéma PID regulátoru je na dalším obrázku (Obr. 5.05).



Obr. 5.05 – PID regulátor





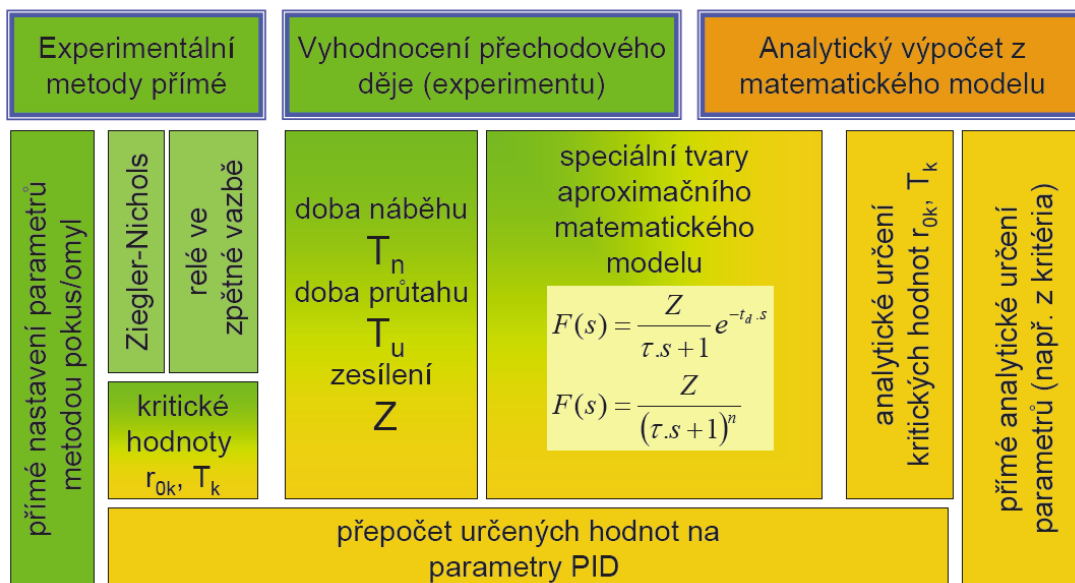
Obr. 5.06 – Přenosová funkce PID

Přenosová funkce PID regulátoru (Obr. 5.06) názorně ukazuje vliv každé složky regulátoru. Zesílení  $r_0$  způsobí zvýšení úrovně. Derivační složka  $r_1$  má za následek prudký pokles průběhu akční veličiny a integrační  $r_{-1}$  růst v čase. Po skokové změně regulační odchylky nejprve převládá vliv derivační složky a poté se v čase začne uplatňovat integrační část.

Proporcionální složka  $P$  odráží současnou odchylku, integrační složka  $I$  zahrnuje odchylku v minulosti a derivační složka  $D$  je odhad následující odchylky. Na regulované soustavě, které nejsou dostatečně popsány, je použití PID regulátoru vhodné. Tento typ regulátoru nemusí vždy představovat nejlepší možnost řízení, ale při správném nastavení parametrů je schopen odolávat nestabilitě. Horší nastavení parametrů má za následek delší dobu regulace vzniklé odchylky, proto je vhodné věnovat nastavení patřičnou pozornost.

## 5.2. Nastavování parametrů regulátoru PID

Metod nastavování regulátoru je velké množství. Lze postupovat od těch nejprimitivnějších pouhým experimentálním nastavením až po metody založené na analytickém výpočtu z matematického modelu regulované soustavy. Pomyslný střed tvoří metody kombinované využívající experimentálního přístupu a následného matematického řešení. Následující přehled (Obr. 5.07) názorně ukazuje, jaké jsou možnosti nastavení regulátoru, kde zelená barva označuje experimentální metody a žlutá barva znamená určování dle matematických modelů.



Obr. 5.07 – Přehled metod nastavení PID regulátoru (12)

Jednoduchou a často využívanou metodou je metoda Ziegler-Nichols. Dle tabulky (Tab. 5.01) je potřeba zjistit dva důležité parametry  $r_{0k}$  – kritické zesílení a  $T_k$  – kritická perioda.

Tab. 5.01 – Nastavení parametrů metodou Ziegler-Nichols

Typ regulátoru	$r_0$	$r_{-1}$	$r_1$
<i>P</i>	$0,5 \cdot r_{0k}$		
<i>PI</i>	$0,45 \cdot r_{0k}$	$0,85 \cdot T_k$	
<i>PID</i>	$0,6 \cdot r_{0k}$	$0,5 \cdot T_k$	$0,125 \cdot T_k$

Kritické zesílení se zjišťuje při nulovém vlivu integrační a derivační složky. Proporcionální složka regulátoru se postupně zvyšuje, až je dosaženo kmitavého průběhu na mezi stability, kdy výstup konstantně kmitá. Ze zaznamenaného průběhu se poté určí kritická perioda odpovídající časovému úseku jednoho kmitu. Nakonec se dle tabulky dopočítají parametry a podle nich nastaví jednotlivé složky regulátoru.

Rychlejšího zjištění kritického zesílení se dosáhne použitím zapojení relé do zpětné vazby, které diskretně kmitá a tvoří obdélkový průběh. Hodnota kritického zesílení se určí dle vztahu (5.2.1):

$$r_{0k} = \frac{4 \cdot R}{\pi \cdot e_{max}} \quad (5.2.1)$$

Kde hodnota  $R$  označuje velikost amplitudy relé a  $e_{max}$  je určena jako polovina špičkové hodnoty vzniklého výstupního průběhu vlivem kmitání relé.

## 6. Praktické řešení

V bakalářské práci na téma „Navigace robota pomocí akcelerometru a gyroskopu“ jsem navrhl a sestrojil systém pro inerciální navigaci. Návrh spočíval ve vytvoření kompletní desky plošného spoje obsahující akcelerometr a gyroskop. Pro potřeby diplomové práce jsem se rozhodl použít hotové řešení senzoru inerciálních veličin a kitu mikrokontroleru s vyvedenými porty.

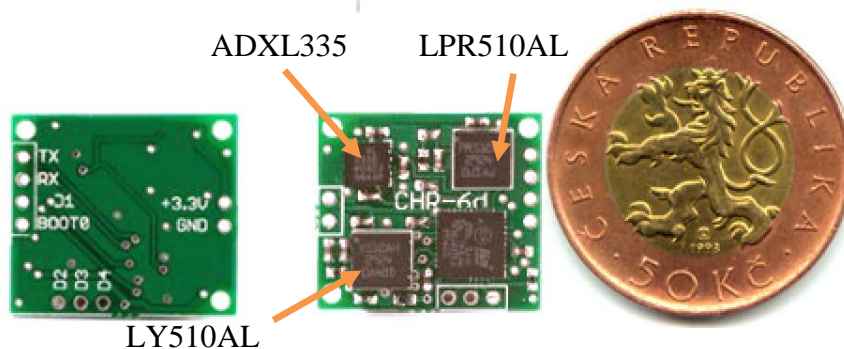
### 6.1. IMU senzor CHR-6D

Výběr senzoru, pomocí něhož bude stabilizován létající objekt, spočíval v základním požadavku, aby obsahoval tříosý akcelerometr a tříosý gyroskop. Modul pro měření inerciálních veličin (IMU) od společnosti CH-Robotics splňuje tyto požadavky a je navržen s ohledem na nízkou cenu a jednoduchost použití.

CHR-6D představuje šestiosou inerciální jednotku s měřením rotace ve třech osách a měřením akcelerace ve třech osách. Modul má minimální rozměry (porovnání na Obr. 6.01) a váží jen 1,5 gramů.

Má vestavěn mikrokontroler ARM Cortex běžící na frekvenci 64 MHz. Všechny kanály senzoru jsou převzorkovány a rozšířeny na rozlišení 16 bit a pomocí dolnofrekvenčního FIR filtru s Parks-McClellan oknem k odstranění nežádoucího šumu produkovaného senzory. Zlomová frekvence filtru je nastavitelná v rozsahu 10 Hz až 140 Hz s krokem 10 Hz pro každý kanál zvlášť. Počet sekcí filtru je nastavitelná na 8, 13, 32 a 64 sekcí nebo je možné filtr vyřadit pokud není vyžadován. Firmware jednotky zahrnuje i rozšířený Kalmanův filtr pro úhly pitch a roll.

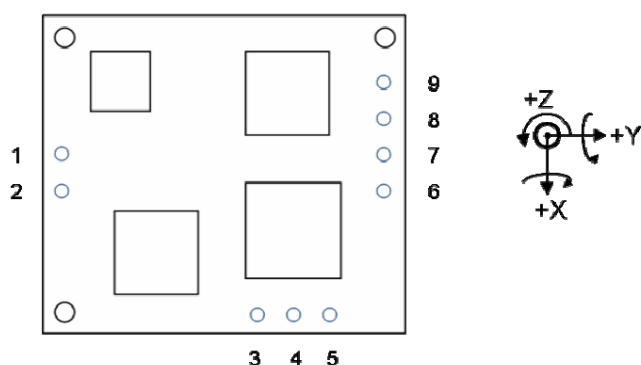
Komunikace je řešena prostřednictvím periferie UART o rychlosti 115200 Baudů s úrovní TTL. Přenos dat lze nakonfigurovat na automatické posílání dat z modulu nebo na vyžádání uživatele zasláním příslušného paketu na vyžádání.



Obr. 6.01 – CHR-6D a jeho skutečná velikost

### 6.1.1. Hlavní vlastnosti modulu

- Vestavěný rozšířený Kalmanův filtr (EKF) pro úhly pitch a roll
- + 3.3 V napájecí napětí
- 3–osy gyroskopu (osa X,Y – pitch/roll rychlost – LPR510AL , osa Z – yaw rychlost – LY510AL) oba gyroskopy od společnosti STMicroelectronics
- 3–osy akcelerometru (ADXL335) od společnosti Analog Devices
- Vestavěný 32-bit ARM Cortex procesor – STMicroelectronic
- TTL UART komunikace
- Nastavitelný FIR filtr s Parks-McClellan oknem pro každý kanál
- 16-bit efektivní rozlišení dat
- Vysoká stabilita vlivem teplot
- $\pm 400$  °/s rozsah gyroskopu
- $\pm 3g$  rozsah akcelerometru



Obr. 6.02 – Orientace modulu (13)

Tab. 6.01 – Popis pinů modulu IMU

Popis pinů modulu		
č. pinu	označení	popis funkce
1	Vcc	3.3 V napájecí napětí
2	GND	Zem napájení
3	D4	GPIO pin 4 (pro nahrání firmware)
4	D3	GPIO pin 3 (pro nahrání firmware)
5	D2	GPIO pin 2 (pro nahrání firmware)
6	Boot0	Bootloader pin. Pro normální režim nezapojeno
7	D1	GPIO pin 1 (pro nahrání firmware)
8	RX	RX (vstupní) pin pro UART
9	TX	TX (výstupní) pin pro UART

### 6.1.2. Základání funkce

Modul pracuje ve dvou módech komunikace. V „Broadcast“ módu jednotka posílá data dle frekvence zvolené uživatelem mezi 20Hz a 300Hz. V módu „Silent“ se posílají data jen na vyžádání uživatele po poslání paketu „GET\_DATA“. Bez ohledu na frekvenci posílání dat jsou registry IMU aktualizovány s frekvencí 500 Hz. Posílaná data „SENSOR\_DATA“ paketem jsou vždy poslední nejaktuálnější. Uživatel může volit, ze kterých kanálů chce přijímat data. Toto nastavení se do modulu zapíše vysláním paketu „SET\_ACTIVE\_CHANNELS“.

Data z modulu jsou vysílána v 16 bit formátu se znaménkem. Pro jejich správnou interpretaci je zapotřebí přijatá data vynásobit měřítkem. Pro gyroskop je měřítko 0,02014 a akcelerometr 0,0001678. Pitch a roll úhly mají rozsah od -360° až 360° a platí pro ně měřítko 0,0109863. Veškerá nastavení modulu pro filtr, bias, frekvence posílání dat a podobně, je možné uložit do paměti vysláním paketu „WRITE\_TO\_FLASH“.

Modul umožňuje ověření správné funkce všech kanálů vysláním datového paketu „SELF\_TEST“. Jsou-li všechny kanály v pořádku, vyšle modul paket „STATUS\_REPORT“, který obsahuje stav jednotlivých kanálů.

### 6.1.3. Kalibrace

V kapitole (1.1.) byl popsán princip vyhodnocení dat výstupu z akcelerometru, obdobně platí i pro výstupní data gyroskopu. Modul CHR-6D má v sobě zabudované nastavení biasu pro každou osu a vysílaná data po UART představují skutečnou hodnotu zrychlení, zpomalení. Hodnotu bias je možné libovolně nastavit. Data z A/D převodníku upravená biasem vypadají dle (6.1).

$$data = ADCdata - bias \quad (6.1)$$

Možnost nastavení biasu je vhodná pro stabilizaci objektu, kdy je pevně nastavena orientace modulu. Běžně se výchozí poloha zařízení nastavuje právě tím, že po zapnutí probíhá kalibrace dle aktuální polohy.

### 6.1.4. Komunikace s IMU

Komunikace s IMU probíhá, jak již bylo popsáno výše pomocí UART s přenosovou rychlostí 115200 Baudů. Komunikace je paketově orientovaná.

#### 6.1.4.1. CHR-6D RX pakety

RX pakety jsou datové rámce, které přijímá modul CHR-6D. Každý paket přijatý IMU musí začínat tříbytovou sekvencí dat „*snp*“ označující začátek nového paketu. Čtvrtý byte datového rámce je identifikátor typu paketu (*PT*) vyjadřující jaký typ paketu byl vyslán do modulu. Podle něho se řídí další zpracování. Pátý byte z rámce značí z kolika (*N*) datových bytů je složen paket. Poslední dva byty datového rámce jsou kontrolním součtem všech bytů v paketu. Suma umožňuje ověření integrity přijatých dat. Při nerozeznání typu paketu je automaticky vyslán zpět „UNRECOGNIZED\_COMMAND“ paket.

Tab. 6.02 – Struktura datového rámce RX

Funkce	's'	'n'	'p'	PT	N	d1	...	d <sub>N</sub>	CHK	
Byte	1	2	3	4	5	6	...	N+5	N+6	N+7

Tab. 6.03 – Popis datového rámce RX

1 – 3	Každý paket přijatý IMU musí začínat tříbitovou sekvencí dat „ <i>snp</i> “ označující začátek nového paketu
4	<b>PT</b> identifikátor typu paketu
5	<b>N</b> očekávaný počet datových bytů
6 – (N+5)	<b>d1</b> až <b>d<sub>N</sub></b> obsahuje <b>N</b> datových bytů paketu
(N+6)–(N+7)	<b>CHK</b> je dvoubitová suma všech dat v jednou rámci

Celková velikost paketu s *N* datovými byty je (*N*+7) bytů. Po příjmu kompletního paketu je kontrolován součet všech bytů v datovém rámci, jestliže souhlasí s posledními dvěma byty kontrolního součtu. Pokud suma nesouhlasí je vyslán zpět „BAD\_CHECKSUM“ paket jako odpověď.

Ukázka paketu GET\_DATA:

Tab. 6.04 – Struktura paketu GET\_DATA

Byte	1	2	3	4	5	6
data	's'	'n'	'p'	0x01	CHK_H	CHK_L

První tři byty označují začátek paketu, čtvrtý byte má hodnotu 0x01 (typ paketu „GET\_DATA“), a protože žádost na vyslání dat neobsahuje žádné byty dat, jako pátý a šestý byte je vyslána kontrolní suma bytů paketu. Přenos dat po UART je osmibitový, a proto je kontrolní 16 bit suma rozdělena na dvě části. Po příjmu je zapotřebí pomocí logického posuvu a součtu sumu spojit dohromady.

### 6.1.4.2. CHR-6D TX pakety

TX pakety představují datové rámce, které IMU posílá. Struktura paketu je totožná se strukturou paketu RX.

Tab. 6.05 – Struktura datového rámce RX

Funkce	's'	'n'	'p'	PT	N	d1	...	d <sub>N</sub>	CHK
Byte	1	2	3	4	5	6	...	N+5	N+6 N+7

Tab. 6.06 – Popis datového rámce RX

1 – 3	Každý paket přijatý IMU musí začínat tříbitovou sekvencí dat „snp“ označující začátek nového paketu
4	PT identifikátor typu paketu
5	N očekávaný počet datových bytů
6 – (N+5)	d1 až d <sub>N</sub> obsahuje N datových bytů paketu
(N+6)–(N+7)	CHK je dvoubitová suma všech dat v jedné rámci

Celková velikost paketu s  $N$  datovými byty je i v tomto případě  $(N+7)$  bytů.

Ukázka paketu „SENSOR\_DATA“:

Tab. 6.07 – Struktura paketu SENSOR\_DATA

Byte	1	2	3	4	5	6	7	8	...	5	6
data	's'	'n'	'p'	0xB7	N	CH_A	1.CH_A_H	1.CH_A_L	...	CHK_H	CHK_L

Datový rámec začíná stejně jako ostatní tříbytovou sekvencí „snp“. Čtvrtý byte má hodnotu 0xB7 a označuje SENSOR\_DATA paket. Pátý byte označuje počet datových bytů paketu a má hodnotu  $N = 1 + 2 \cdot (\text{počet aktivních kanálů})$ . Následuje šestý byte CH\_A jehož spodních 6 bitů značí, které kanály jsou vyslány. Má-li bit hodnotu logickou jedničku je daný kanál aktivní, další dvojice datových bytů obsahují data jednotlivých aktivních kanálů od prvního do posledního. Poslední dva byty datového rámce je kontrolní součet všech bitů datového rámce.

## 6.2. Mikrokontroler – kit STM32 VL Discovery

Přijímaná data z IMU mají 16 bit formát. Pro algoritmus regulátoru a nastavování jeho konstant je vhodné použití formátu double. Na základě těchto faktů je zapotřebí použít mikrokontroler, který tento datový formát dokáže zpracovat v rozumném čase. Kit STM32VLDiscovery od firmy STMicroelectronics obsahuje 32 bit mikroprocesor ARM Cortex–M3 STM32F100RB běžící na maximálním kmitočtu 24 MHz. Podpora 32 bit instrukcí je pro řešený problém ideální. MCU obsahuje navíc mnoho užitečných periférií a jeho cena je nízká v porovnání s jeho výbavou.



Obr. 6.03 – STM32VLDiscovery

Vlastnosti kitu:

- STM32F100RBT6B mikrokontroler, 128 kB Flash, 8 kB RAM
- ST-Link s výběrem použití jako nezávislý ST-Link (SWD konektor pro programování a ladění externích zařízení)
- Možnost napájení z externího zdroje
- Možnost napájení externích obvodu napětím 5V a 3,3V
- Dvě uživatelské LED diody(zelená a modrá)
- Uživatelská tlačítka(uživatelské a reset)
- Výstup všech I/O pinů MCU pro rychlé připojení

Vlastnosti mikrokontroleru:

- Ochranné prvky: vestavěná POR, PDR, LVD a WDT ochrana
- RC oscilátor 8 MHz a 40 kHz pro RTC a watchdog
- Maximální provozní frekvence 24 MHz
- 2x12bit D/A a 1x12bit A/D převodníky s podporu DMA
- 7xPWM 16bit čítačů/časovačů s 26 nezávislými kanály
- 2xI2C, 2xSPI (12 MB), 3x USART 3Mb/s
- 7 kanálů DMA pro periferie i paměť

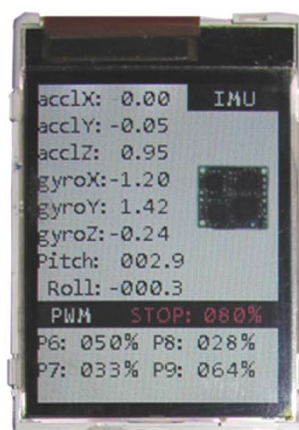
Programování a ladění software probíhalo ve vývojovém studiu Atollic TrueSTUDIO® STM32 Lite 2.0.1 od společnosti Atollic. Lite verze není omezena na délku napsaného kódu, ale uživatel nemá možnost nahlížet do aktuálního stavu registrů a má k dispozici jen jeden „breakpoint“. Nepříjemné je stále nabízení plné verze k zakoupení.



### 6.2.1. LCD displej

Data zpracovávaná mikrokontrolerem je vhodné zobrazit pro lepší přehled stavu MCU. Zejména pro regulaci je zapotřebí zobrazit aktuální stav úhlů pitch a roll i výkon jednotlivých motorů. Pro zobrazení aktuálních dat, přijatých z IMU, a jednotlivých výkonů motorů byl vybrán LCD displej, který je použit v mobilním telefonu Siemens S65, jehož výrobcem je firma Sharp a má typové označení LS020xxx.

Jedná se o 16 bit barevný TFT displej s rozlišením 132x176 pixelů a počtem barev 65 tisíc. Jeho předností je snadná ovladatelnost a díky početné komunitě uživatelů je pěkně popsáné nastavování a řízení displeje zejména pro mikrokontrolery Atmel AVR. K tomuto MCU jsou dostupné zdrojové kódy pro řízení, zobrazování textu a kreslení základních geometrických objektů nebo obrázků. Aby bylo možné zdrojové kódy použít bylo zapotřebí přepracovat a uzpůsobit zdrojový kód pro STM32VLDISCOVERY kit. Display má řídicí napětí 3,3V stejně jako MCU na kitu, tím je umožněno přímé připojení LCD k MCU bez dalších podpůrných obvodů a uzpůsobení výstupů. Na obrázku (Obr. 6.04) je ukázka výstupu dat na displeji.



Obr. 6.04 – Ukázka výstupu dat LCD S65

V sekci označené IMU je zobrazen aktuální stav posledních přijatých dat prostřednictvím modulu CHR-6D. Pro regulaci jsou nejdůležitější úhly náklonu v ose X úhel pitch a v ose Y úhel roll. Hodnoty těchto úhlů jsou aktuálně blízké nule, z toho vyplývá, že se modul nachází v rovnovážné poloze. Tento stav potvrzuje i hodnota osy Z akcelerometru blízká jedničce. Oddíl označený PWM zobrazuje jednotlivé výkony motorů quadcopteru. Zde jsou data náhodně zvolená pro ukázkou. PWM je generována časovačem a jeho výstupy jsou přivedeny na porty, konkrétně port B. Tomu odpovídá i označení motorů symbolem „P“ + číslo pinu portu. Červený nápis „STOP:“ znamená omezení výkonu motorů z důvodu vývoje a bezpečnosti.

### 6.2.1.1. Zapojení LCD displeje a ovládání

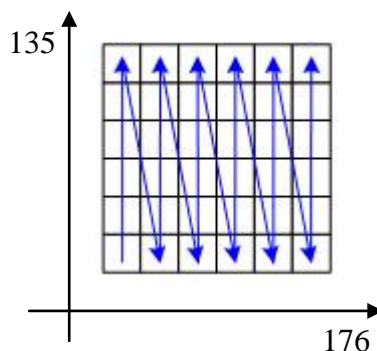


Obr. 6.05 – Zadní strana a označení připojení LCD S65 (1 str.70)

LCD display komunikuje prostřednictvím sběrnice SPI. Jako master je MCU, slave představuje LCD. Dle terminologie SPI, vývod označený „DAT“ odpovídá vodiči MOSI, vývod „CLK“ označuje SCLK, „CS“ je  $\overline{SS}$ . „RESET“ představuje klasický reset LCD displeje a vývody označené LED slouží jako podsvětlení.

Speciální funkci zastává vývod označený „RS“, který určuje, zda jsou do LCD vysílána data nebo příkazy. Nastavením vodiče RS na hodnotu logické jedničky displej očekává přijetí příkazu, opačně se zapisují data, která se mají zobrazit. Komunikace mezi MCU a LCD probíhá rychlostí 12 Mbit/s a uživatel si může zvolit režim komunikace mezi 16 bit a 8 bit, která je o poznání rychlejší.

Vykreslování LCD (Obr. 6.06) probíhá od levého horního pixelu doprava, po vykreslení řádku pokračuje vykreslování dalších, až dokonce po zaplnění celého displeje. Jeho orientaci je možné zvolit na šířku nebo na výšku. Pro vykreslování byla zvolena orientace, jako je naznačená na obrázku pod odstavcem. Pokud chce uživatel překreslit jen část obrazu, není nutné překreslovat displej celý a tím překreslení části LCD je velmi rychlé.

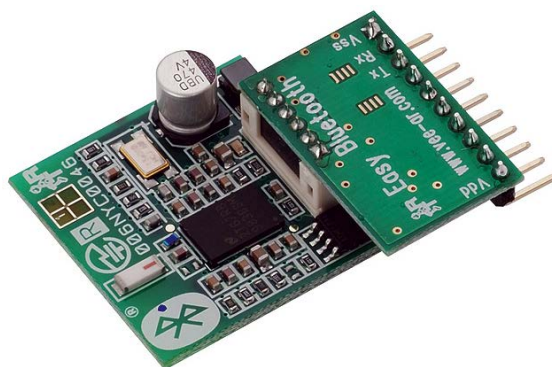


Obr. 6.06 – Způsob vykreslování (1 str.70)

### 6.2.2. Komunikace

Modul CHR-6D komunikuje prostřednictvím UART, vodiče RX a TX jsou připojeny ke kitu na příslušné piny periferie USART1. Pro nastavování regulátoru a ovládání quadcopteru bylo vybráno také UART rozhraní, díky svojí jednoduchosti. Rychlosti přenosu jsou pro účely ovládání dostačující, obzvláště kdy komunikace neprobíhá neustále. Pokyn je zapotřebí vyslat jen občas.

Jelikož se jedná o létající objekt, použití vodičů k nějakému ovládání nepřichází v úvahu, je nutné zvolit způsob bezdrátové komunikace. Pro zachování předností UART rozhraní se nabízí použít modul bluetooth. Od specifikace 2.0 v sobě zahrnuje protokol SPP emulující sériovou linku. V ovládacím zařízení se objeví virtuální sériová linka a uživatel komunikuje standardně jako při spojení vodiči. Modul bluetooth (Obr. 6.07) byl použit od firmy Parallax.



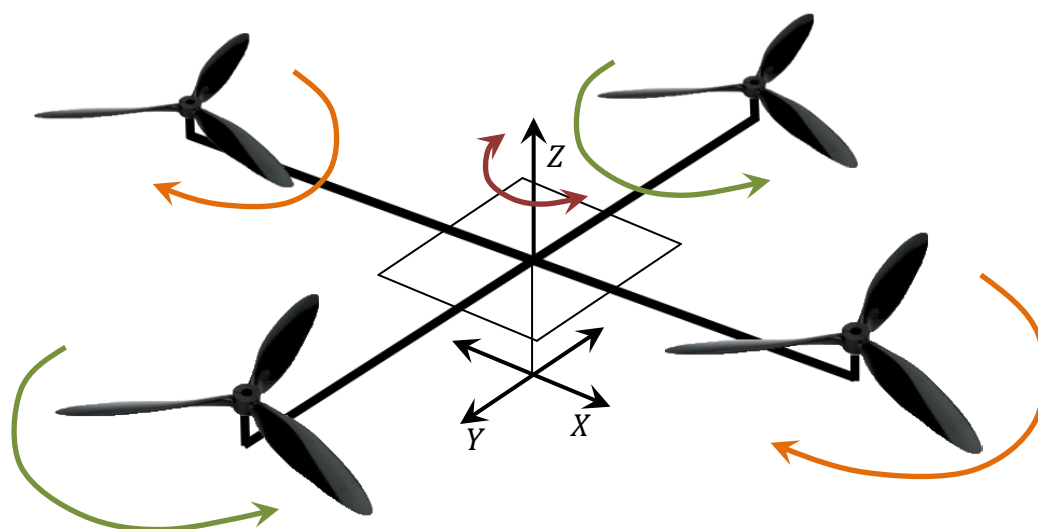
Obr. 6.07 – Bluetooth modul – Parallax (15)

Napájení modulu je zajištěno zdrojem 5V. Vodiče RX a TX jsou připojeny na jednotku USART2. Výhodou tohoto řešení je možnost ovládání jakýmkoliv zařízením s podporou bluetooth a uživatel není omezen jen na použití PC a jeho realizace může být provedena i na mobilních telefonech a podobných zařízeních.

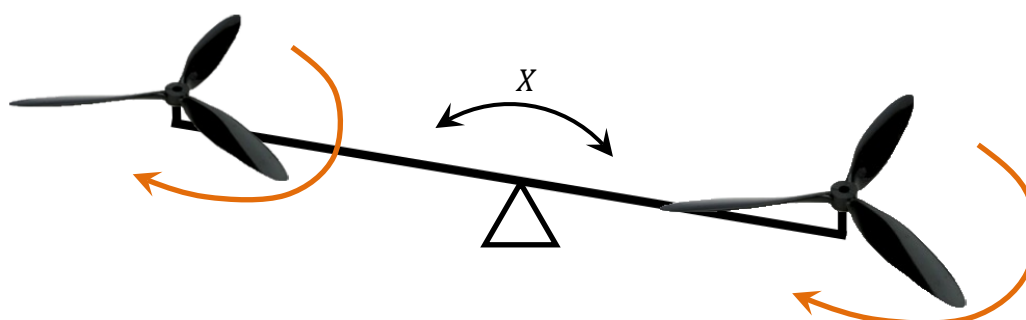
Kompletní zapojení a podrobnější popis komunikace se nachází v dalších kapitolách.

### 6.3. Konstrukce quadcopteru

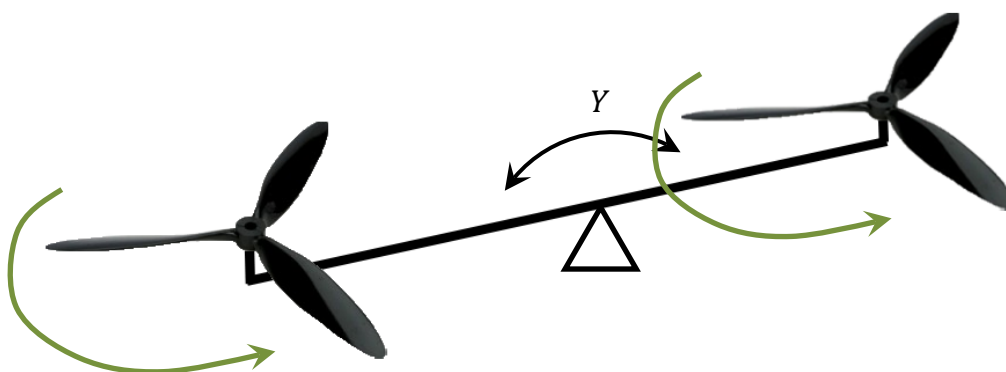
Quadcopter (Obr.6.08) je létající objekt, který se vznáší pomocí čtyř vrtulí, proto takové pojmenování. Každé protilehlé vrtule se otáčí stejným směrem, tím je zajištěn vztlak, aby mohl objekt vzlétnout. První dvojice protilehlých vrtulí (Obr. 6.09) se točí jedním směrem pro vyvážení tahu v ose  $X$ . Druhá dvojice protilehlých vrtulí se otáčí opačně pro vyvážení v ose  $Y$ . Hlavní důvod opačného otáčení protilehlých dvojic je zamezení rotace quadcopteru podle osy  $Z$ .



Obr. 6.08 – Princip funkce quadcopteru



Obr. 3.09 – Princip funkce rameno 1



Obr. 6.10 – Princip funkce rameno 2

Z pohledu regulace je vhodné systém dekomponovat na tři nezávislé části. Vhodným řešením je regulace v ose  $X$ , kde se řeší rovnováha na páce (Obr. 6.09), podobně i v ose  $Y$  (Obr. 6.10) a poslední regulací je zamezení rotace v ose  $Z$  (Obr. 6.08).

Kříž quadcopteru, na jehož koncích jsou upevněny motory, tvoří uhlíkové hranoly. Na středu kříže je přidělena z vrchu i zespodu karbonová destička pro uchycení desky plošného spoje na distanční sloupky na horní plochu. Na spodek quad-

copteru je připevněna Li-Pol baterie sloužící k celkovému napájení motorů a komponent desky plošného spoje. U každého motoru je na konstrukci kříže ještě přidělán regulátor otáček. Důvodem volby konstrukčních prvků na bázi uhlíku je celková hmotnost zařízení. Při větší hmotnosti je zapotřebí větší tah motorů a tím se zkracuje doba letu, která i tak trvá kolem patnácti minut do vybití baterie.

V praxi je málo věcí ideálních a i přes poměrně stabilní konstrukci quadcopteru, k tomu aby mohl kontrolovaně létat, je zapotřebí regulace.

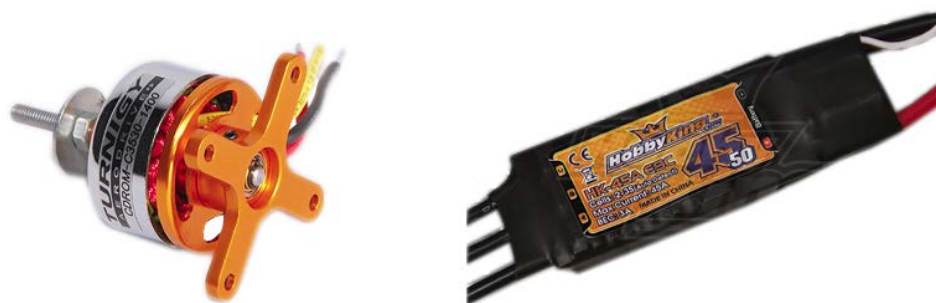
### **6.3.1. Motory a regulátory**

Aby mohl quadcopter vzlétnout je zapotřebí poměrně vysoký tah motorů. K tomuto účelu je vhodné použití střídavých bezkartáčových (brushless – bez uhlíku) motorů. Tah motorů lze plynule ovládat prostřednictvím PWM pulzů, ale ke správnému chodu je vyžadován regulátor otáčení. Pro regulátor je důležité „zjistit“ v jaké pozici je rotor motoru. Je možné použití motoru s vestavěnými senzory, ale bývá problém s kompatibilitou motoru a regulátoru. V současné době se vyrábí motory bez senzoru (sensorless) a pro zjištění polohy rotoru je využito následující metody. Velmi zjednodušeně řečeno, regulátor do vinutí motoru vysílá vysokofrekvenční pulsy a vyhodnocuje jejich indukovaný zpětný odraz, který závisí na polaritě magnetu, jenž je cívce nejbližší.[16]

Výhodou tohoto řešení je levnější motor, neobsahuje snímač polohy a není zapotřebí použití více drátů pro regulátor, protože sensorové motory mají ještě pět dalších drátů vedoucích k regulátoru od snímače. Regulátory tak mají pouze tři vodiče a záměnou kterýchkoliv dvou vodičů mezi motorem a regulátorem se změní směr otáčení rotoru.

Konkrétně pro létání jsou nejvhodnější motory s rotačním pláštěm. Zásadní výhodou této konstrukce je, že motor má velmi vysoký kroučící moment, což dovoluje pohánět přímo vrtule velkých průměrů a stoupání, bez použití převodovky. Střídavé motory jsou oproti stejnosměrným se stejným výkonem vždy o 30-50% lehčí a mají až o polovinu vyšší účinnost. Významnou výhodou střídavých elektromotorů při použití přímého náhonu je mnohem příznivější průběh kroučícího momentu. Zatímco stejnosměrný motor dobře „táhne“ jen při otáčkách blížících se maximu, se střídavým motorem bude dobře navržený model schopen horizontálního letu na 20 nebo i méně procent plného výkonu. To vše znamená výrazné snížení hmotnosti modelu, zvýšení výkonů a prodloužení doby letu.[16].

Ukázka motoru a regulátoru použitých na quadcopteru (Obr. 6.11):

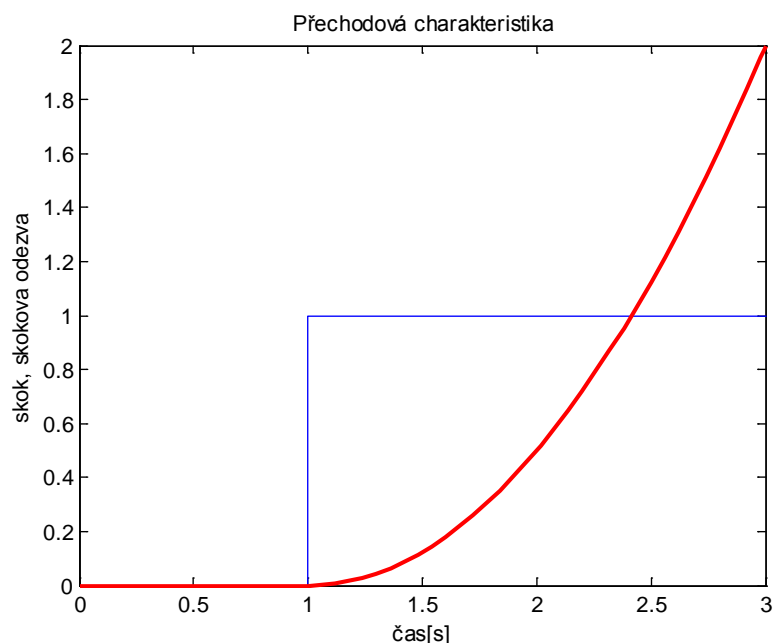


**Obr. 6.11 – Motor Turnigy CDROM C3530-1400 a regulátor HK45A ESC**

Parametr C35 označuje průměr 35 mm a hodnota 1400 vyjadřuje počet otáček na volt. ESC regulátor má maximální zatížení proudem 45 A a podporuje motory s otáčkami do 2200/V. Pro tento typ motoru je dostačující i z hlediska dlouhodobého využívání díky jeho dimenzování.

### 6.3.2. Nastavení regulátorů

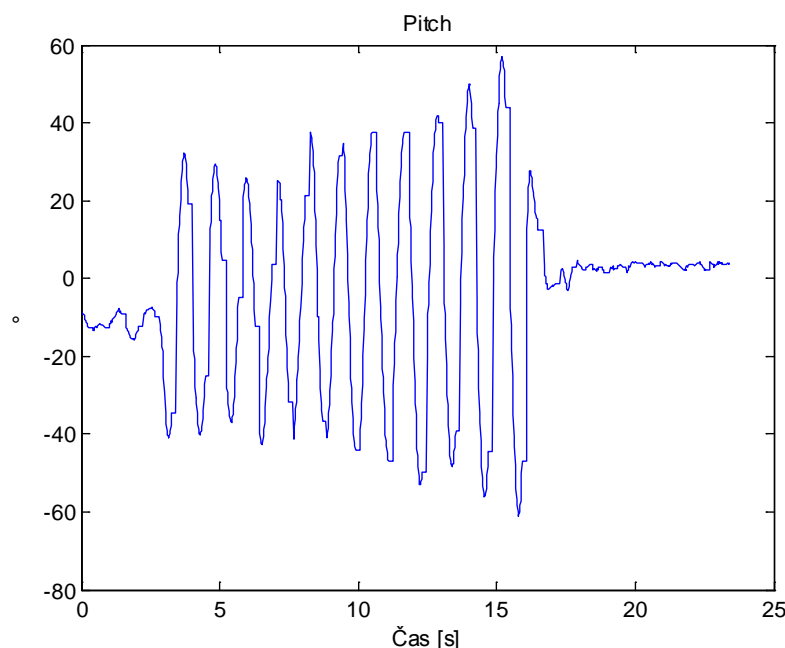
V případě regulace quadcopteru se jedná o integrační soustavu, která má přechodovou funkci stále integrující do nekonečna (Obr. 6.12) a pro nastavení konstant regulátoru bylo nutné použití experimentálních metod, konkrétně Ziegler-Nichols.



**Obr. 6.12 – Přechodová charakteristika**

Nastavení probíhalo vyřazením integrační a derivační složky z regulace a po malých krocích se nastavovala proporcionalní složka, která způsobila pravidelné kmitání quadcopteru kolem rovnovážné polohy, a tím bylo zjištěné kritické zesílení.

Kmitavý průběh se zaznamenal do textového souboru a byl nainportován do programu Matlab. Data průběhu úhlu pitch (Obr. 6.13) posloužila k nalezení kritické periody výběrem jednoho kmitu a určení doby jeho trvání.



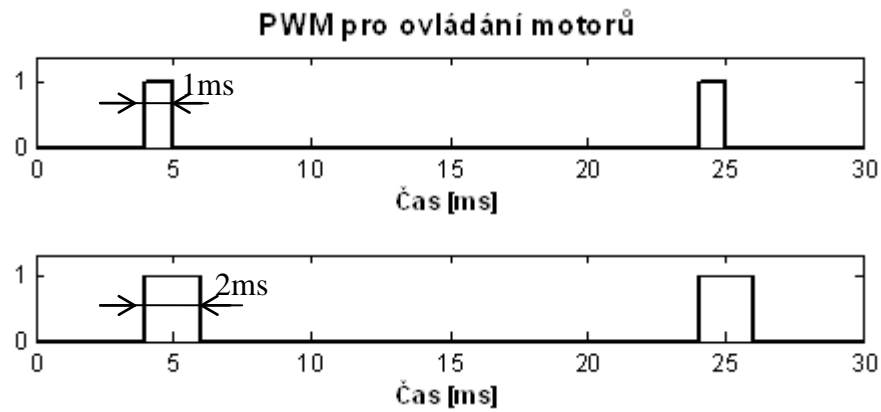
**Obr. 6.13 – Zjištění kritické periody z průběhu výkonu motoru 1**

Postupným nastavováním bylo zjištěno kritické zesílení  $r_{0k} = 0,0012$  a dle průběhu úhlu náklonu (pitch) kritická perioda  $T_k = 1,17$  s.

### 6.3.3. Řízení motorů pomocí PWM

Generované signály PWM pro motory jsou z výstupu kitu s MCU přivedeny na regulátory jednotlivých motorů. Způsob ovládání střídavých motorů vychází z principu ovládání servomotorů. Základní signál má hodnotu logické jedničky po dobu 1 ms a periodou 20 ms.

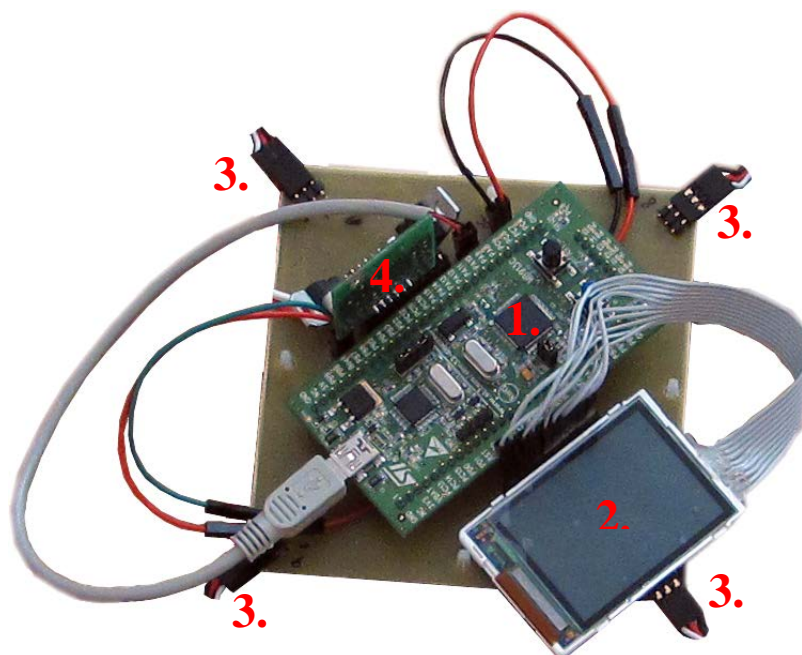
Po zapnutí quadcopteru musí mít impuls PWM šířku 1 ms, který představuje zastavený motor (Obr. 6.14 nahoře). Při větší šířce počátečního impulsu regulátor motor nedovolí roztočit z bezpečnostních důvodů. Pro zaběhnutí motoru je nutné mít vždy šířku pulzu 1 ms. Plynulým zvyšováním šířky impulsu z 1 ms výš se začínají motory roztáčet. Horní hranice šířky impulsu je 2 ms (Obr. 6.14 dole), ale při této hodnotě se motor točil až příliš a bylo nutné snížit maximální hodnotu na 1,7 ms. Tím bylo umožněno plynulé ovládání rychlosti otáčení vrtulí při nastavování procent výkonu. Právě plynulá změna rychlosti je pro stabilizaci důležitá.



Obr. 6.14 – Řízení signálem PWM

### 6.3.4. Deska plošného spoje

Deska byla vytvořena pomocí programu PADS a je umístěna na vrchní straně karbonové destičky quadcopteru. Na ní jsou rozmístěny všechny komponenty napájení, připojení LCD, modul IMU, bluetooth a samotný kit s MCU. Modul CHR-6D je umístěn pod kitem na středu konstrukčního kříže. Toto místo nejlépe vystihuje orientaci quadcopteru. Jednotlivé kanály PWM jsou vyvedeny v rozích desky, pro jejich snadnější připojení.



Obr. 6.15 – Deska plošného spoje quadcopteru

Zapojení komponent DPS:

- 1) Kit STM32VLDISCOVERY
- 2) LCD displej (S65)
- 3) Jednotlivé vývody PWM k motorům



#### 4) Modul bluetooth

Dále je na desce umístěn pod kitem s MCU (1)) modul CHR-6D . Napájení zajišťuje Li-Pol baterie. Její napětí je přivedeno na stabilizátory 5V, který slouží k napájení (1) a 4)) a 3V pro napájení IMU modulu a (2)).

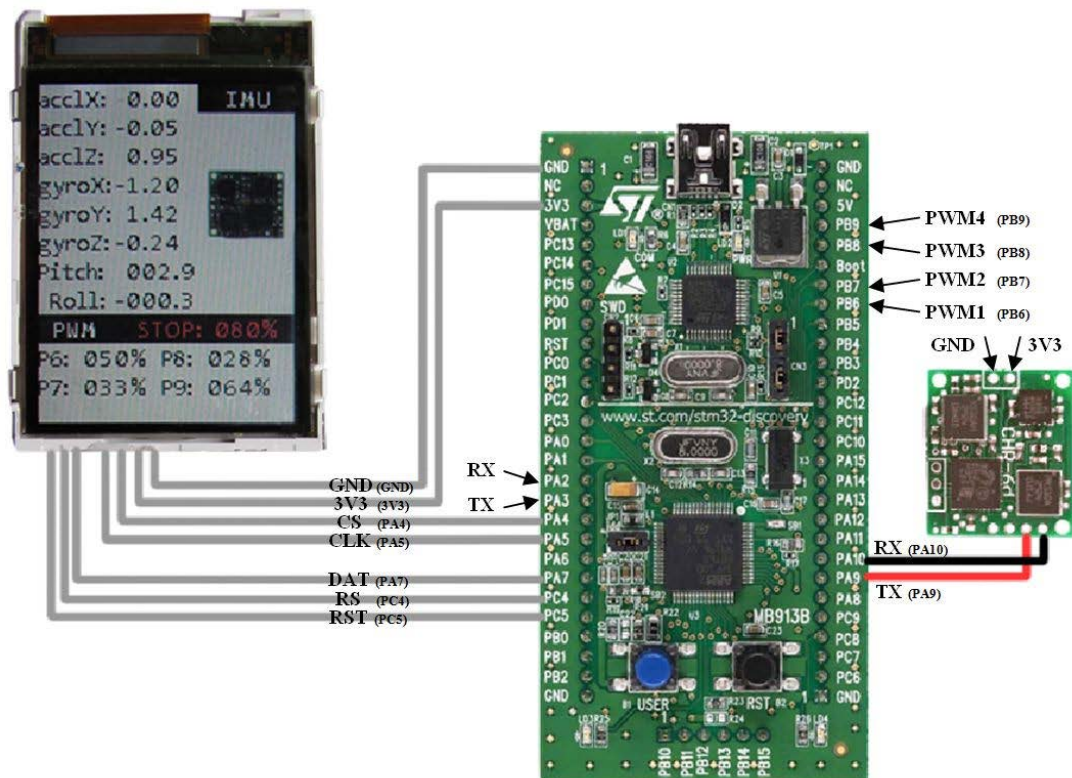
#### 6.3.5. Kompletní sestava



Obr. 6.16 – Quadcopter

Na začátku kapitoly (6.3.) je popsána konstrukce quadcopteru a kompletní sestava na obrázku výše (Obr. 6.16). Propojení jednotlivých částí s MCU je znázorněno na následujícím obrázku (Obr. 6.17). MCU na kitu je napájeno napětím 3,3 V. Regulátory ESC vyžadují signál PWM o napěťové úrovni 5 V a proto bylo zapotřebí převést napětí 3,3V translatorem na hodnotu 5 V. K tomuto účelu byl využit obvod od Texas Instruments s označením TXB0104D. Na jedné straně obvodu se připojí napětí 3,3V a vstupní vodiče, které je třeba převést na vyšší napětí. Z druhé strany se připojí napětí 5V a výstupní vodiče mají již vyšší napěťovou úroveň. Převod napětí neumožňuje zatížit výstupy větším proudem, ale pro potřeby signálu PWM dostačuje. Jednotlivé ESC regulátory jsou tímto způsobem připojeny k MCU na piny, dle obrázku na vývody PWM1 až 4, jež představují výstup časovače generující PWM.

Bluetooth modul pro ovládání je připojen dle obrázku na piny RX, TX bez žádných přivedených vodičů. Pro ovládání je použita periferie USART2. Na piny RX a TX označené černým a červeným vodičem je připojen modul CHR-6D využívající periferii USART1. LCD displej je napájen z kity výstupním napětím 3,3V. Vodiče označené DAT, CLK a CS jsou připojeny na vývody SPI1 sběrnice. Zbývající vodiče pro řízení jsou ovládány klasickými porty MCU.



Obr. 6.17 – Zapojení MCU, CHR-6D a bluetooth

## 6.4. Softwarové řešení

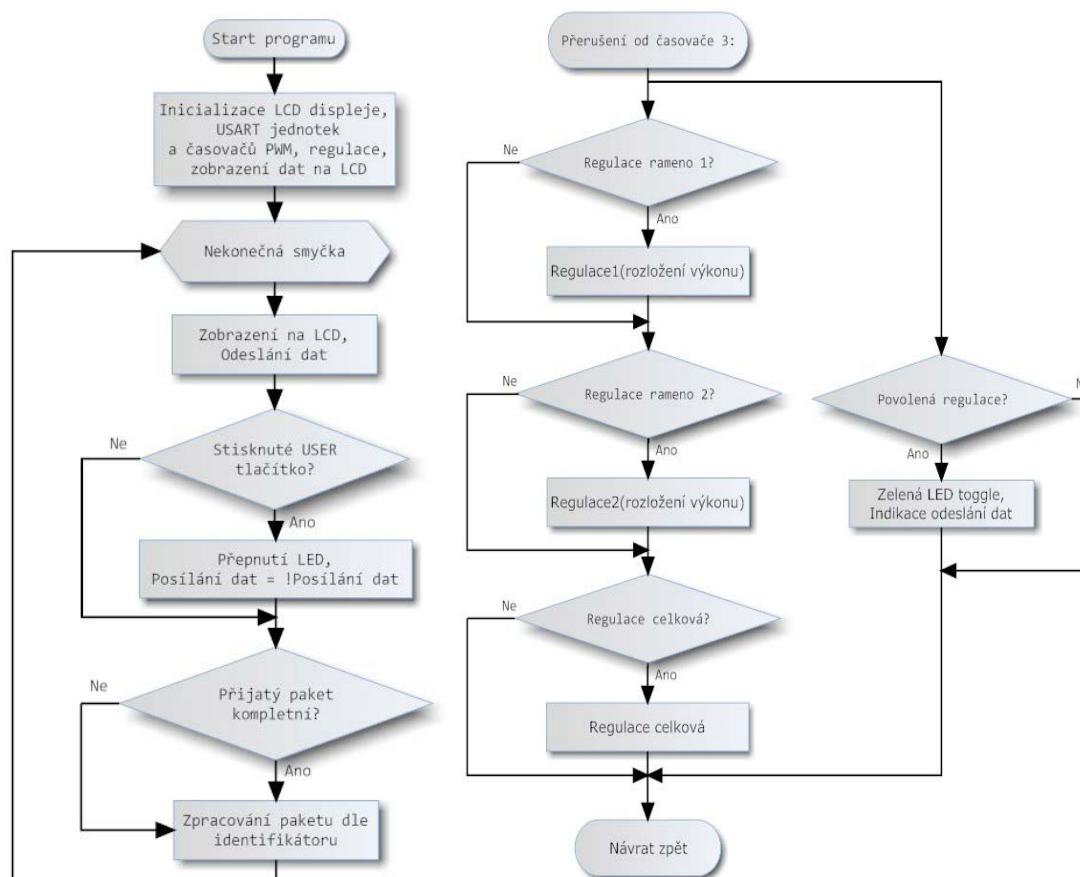
Hlavním cílem programu v mikrokontroleru je spuštění regulační smyčky v pravidelných časových okamžicích. Na základě přijímaných dat z IMU jednotky vyhodnotit polohu v prostoru a jestliže je zjištěna odchylka od požadované stabilní polohy provést opatření v podobě změny tahu jednotlivých motorů vedoucí ke stabilizaci polohy v prostoru.

Data vysílaná IMU jednotkou prostřednictvím periferie USART1 přichází ve formě datových paketů, jaké byly popsány v kapitole o modulu CHR-6D (6.1.). Jednotlivé pakety jsou ukládány do bufferu, nad kterým nepřetržitě při každém přijatém bytu běží funkce *seriovyPortPrijemDat*. Funkce má za úkol v bufferu nacházet označení začátku paketu (znaky *snp*). Po jejich nalezení je ze struktury paketu známé, že následuje identifikátor typu, o jaký paket se jedná, dále velikost pole dat a vlastní data. Po „vyčtení“ jmenovaných bytů a uložení datových bytů do pole přichází na řadu funkce *zparcovaniPaketu* s parametry získaných hodnot. Kód funkce dle typu paketu zvolí příslušné zpracování pole dat jestliže není nulové, nebo provede příkaz odpovídající typu paketu.

Data jednotlivých kanálů senzoru mají 16 bit formát ve dvojkovém doplňku a tím, že jsou vysílány prostřednictvím USART ve standardním formátu 8N1 (osm bitů datových a jeden stop bit) je nutné vyslaná data rozdělit na dvě části, horních osm bitů a spodních osm bitů. Na přijímací straně v MCU je zapotřebí horní část posunout bitovým posuvem o osm bitů doleva a logickým součtem sečíst bity horní části s bity spodní části, a tím je zajištěna rekonstrukce jednotlivých kanálů.

Architektura mikrokontrolerů typu ARM je navržena pro nízké energetické požadavky tzn., není-li v MCU naprogramovaný žádný zdrojový kód, tak celý systém je v „režimu spánku“ a vše působí jako vypnuté. Postupným tvořením funkčního kódu a nutnosti použití nějaké periferie mikrokontroleru nebo nějakého I/O portu, je nutné před jejich použitím přivést hodinový signál do periferie i do samotného portu, aby jí bylo možné vůbec použít. Nastavování hodinových signálů se děje pomocí registrů MCU. ARM má poměrně složité nastavování registrů, je jich velké množství, a proto k nastavení lze použít předepsaných funkcí. Při vývoji softwaru je nutné dát pozor na přivedení kmitočtu do periferie nebo portu, jinak vznikají nepříjemné chyby i v případech správného nastavení periferie.

### 6.4.1. Vývojový diagram aplikace



Obr. 6.18 – Vývojový diagram main a regulační smyčky

Zjednodušený vývojový diagram (Obr. 6.18) popisuje funkci základního programu (levá část) a přerušení od časovače 3 sloužící jako regulační smyčka (pravá část). Po připojení napájecího napětí ke kitu STM32VLDiscovey nebo po stisku tlačítka RESET proběhne nejprve inicializace LCD displeje a USART jednotek, zároveň s ní i základní inicializace časovačů PWM, regulace a zobrazení na LCD.

V nekonečné smyčce programu se kontroluje stisk tlačítka, které povoluje/zakazuje posílání aktuálních úhlů náklonu do PC, indikace je zajištěna pomocí modré LED diody. Po kontrole stisku tlačítka probíhá zpracování přijatých ovládacích paketů. Je-li přijatý paket kompletní, dojde k jeho zpracování podle indikátorů uvedené v podkapitole (6.4.5.) a cyklus nekonečné smyčky se opakuje.

Přerušení od časovače 3 představuje regulační smyčku. Jednotlivé regulační pochody popsané v kapitole (6.3.) lze aktivovat/deaktivovat a tato smyčka zpracovává jen aktivní. Zelená LED dioda bliká v případě, když je alespoň jedna regulace aktivní. Při každém regulačním cyklu změní svůj stav a tím informuje o rychlosti regulační smyčky, která je dvojnásobná než rychlost blikání. S aktivní regulací se

nastaví příznak odeslání dat v podobě úhlů náklonu a v hlavní smyčce programu jsou odeslána prostřednictvím sériové linky zajišťující ovládání.

Program zahrnuje ještě další přerušení od periférií USART a dalšího časovače. V obsluze přerušení od jednotky USART1 se zpracovávají aktuální přijaté pakety z IMU modulu nebo se odesílají požadavky, případně konfigurační data, do modulu. Podobným způsobem je řešeno i přerušení USART2 sloužící k ovládání. Přerušení od časovače 2 slouží k nastavení příznaku zobrazení dat na LCD, které se vykoná v hlavní smyčce. Interval časovače 2 představuje frekvenci zobrazení dat na LCD displeji, podobně jako interval regulační smyčky udává frekvenci posílání dat.

#### 6.4.2. Regulace

Pomocí časovače 3 mikrokontroleru, v obsluze jeho přerušení, běží po povolení regulační smyčky. Pro stabilizaci polohy v prostoru je zapotřebí použít tři regulační smyčky. Jedna slouží na regulaci tahu protilehlých motorů v jedné ose. Druhá vykonává stejnou funkci pro druhou osu a třetí slouží ke stabilizaci rotace quadcopteru. Ovládacím softwarem je možné jakoukoliv regulační smyčku vyřadit z provozu nebo opět spustit, rovněž je možné nastavit i periodu spouštění aktivních regulačních smyček.

Regulační chod smyček regulující náklon v jednotlivých osách je rozdělen do dvou částí. V první části běží klasický algoritmus PID regulátoru, jehož výstup je omezen na hodnoty mezi  $\pm 1$  a představuje poměr výkonu mezi motory. V regulátoru je ještě zahrnuto omezení integrační složky, kdy vlivem dlouhodobějšího chybového stavu se naintegruje velká hodnota a není se schopna v rozumném čase odintegrovat zpět.

Druhá část řeší rozdělení výkonu tahu vrtulí. Do funkce *nastavRameno* vstupuje jako jeden parametr poměr výkonu z PID regulátoru, druhý parametr je rozdělení celkového výkonu mezi dva motory. Vztahy (6.2) a (6.3) uvádí výpočet poměru výkonu mezi dvěma motory.

$$výkon\ motoru\ 1 = \frac{poměr + 1}{2} \cdot výkon \quad (6.2)$$

$$výkon\ motoru\ 2 = \left(1 - \frac{poměr + 1}{2}\right) \cdot výkon \quad (6.3)$$

Ve výpočtu se objevuje jednička přičtená k poměru výkonu motorů z PID regulátoru. Jejím účelem je zamezení možnosti výpočtu záporného výkonu motoru. Hodnota „1 –“ ve vzorci (6.3) značí, že motor dva je doplněk k motoru jedna. Tímto způsobem fungují obě smyčky stabilizující quadcopter v osách  $X$  a  $Y$ .

### 6.4.3. Řízení PWM

Mikrokontroler umístěný na kitu obsahuje velké množství časovačů. Pro potřeby generování signálu PWM byl vybrán časovač 4 vzhledem k dobrému umístění výstupních pinů. Časovač má 4 výstupní nezávislé kanály, u kterých je možné libovolně měnit šířku pulzu PWM. Nastavení časovače spočívá ve volbě periody průběhu a určení délky časového intervalu trvání kladného impulsu.

MCU pracuje na frekvenci 24 MHz, tomu odpovídá doba jednoho hodinového impulsu necelých 42 ns. Šířky pulzů generované pomocí PWM jsou uvedeny v podkapitole (6.3.2.). K dosažení šířky impulsu 1 ms by čítač musel napočítat ohromné číslo, nemluvě o periodě opakování pulzu a v signálu by byla zanesena nepřesnost postupným načítáním. MCU obsahuje předděličku hodinových impulsů. Na rozdíl od jednodušších mikrokontrolerů libovolně nastavitelnou. Její hodnota byla zvolena 100. Použitím předděličky je nejmenší časový úsek časovače nyní 4,2  $\mu$ s a pro dosažení intervalu 1ms stačí načítat 240 „tiků“ časovače. Pro periodu opakování průběhu se načítá 4800. Počet načítaných impulsů je na rozumné úrovni a po ověření osciloskopem šířky impulsů odpovídají požadovaným.

### 6.4.4. Obsluha LCD displeje

Po připojení LCD displeje je zapotřebí provést jeho inicializaci. Ta spočívá ve vyslání sekvence konfiguračních bytů do jeho řídicí jednotky. Po tomto procesu je možné zahájit zápis dat.

Na výběr byly dvě softwarové knihovny LCD S65, první od Christiana Kranze<sup>1</sup>, psaná v assembleru, a druhá později objevená od Steffena Buchnera<sup>2</sup>, psaná programovacím jazykem C. Pomocí první knihovny byla do kitu STM32VLDISCOVERY přepsána inicializace displeje a jeho základní ovládání posílání příkazů a dat do řídicí jednotky. Druhá jmenovaná knihovna byla implementována pro kreslení objektů na displej. Obsahuje funkce umožňující vykreslení základních geometrických útvarů

<sup>1</sup> [http://www.superkranz.de/christian/S65\\_Display/DisplayIndex.html](http://www.superkranz.de/christian/S65_Display/DisplayIndex.html)

<sup>2</sup> <http://www.my404.de/index.php?&main=atmusic&sub=lcd-library.htm&lang=de>

(obdélníky, úsečky), psaní textu, zobrazování obrázků a další. Každému kreslenému objektu lze navolit barvu dle standardu RGB.

Obrázky zobrazené na displeji musí být uloženy ve formě datového pole v paměti na MCU. S knihovnou je dodávána utilita *Imageconverter*, která na předložený obrázek ve formátu bmp s bitovou hloubkou 24 bitů vytvoří *.h* soubor obsahující pole s daty obrázku. Pomocí parametrů konvertoru obrázků lze navolit, jakou barevnou hloubku bude mít obrázek na displeji. Paměť MCU je poměrně velká, ale obrázek velikosti celé plochy displeje se do paměti 128 kB nevejde a z důvodu úspory paměti je vhodnější použít menší obrázek a 8 bit barevnou hloubku. Vytvořený soubor stačí vložit do programu a lze ho ihned použít.

Autor první LCD knihovny vytvořil šikovný program s názvem *Font* pro tvorbu fontů textu zobrazovaných na displeji a je použitelný i pro druhou knihovnu. Uživatel si může s pomocí této utility vytvořit vlastní styl písma nebo použít kterýkoliv oblíbený styl písma uložený v počítači a program sám převede znaky zvoleného stylu písma do formátu využívaného pro zobrazení na displeji. Jednotlivé znaky jsou převedeny na pixely do políčka o rozměrech zadaných uživatelem. Tyto rozměry slouží jako nastavení velikosti písma. Podporována je i volba řezu písma (tučný, kurzíva,...). Dále lze navolit výběr použitých znaků daného stylu včetně diakritiky fontu. Zvolený styl písma a jeho vzhled musí velikostně odpovídat možnostem displeje, zejména jeho rozlišení a velikosti políčka pixelů pro vytvoření znaku, aby se dostatečně detailně vykreslil.

Výstupem programu je i v tomto případě vygenerovaný *.h* soubor s daným stylem písma, který se vloží do projektu a je připraven k použití. Algoritmus vykreslení znaku na LCD postupuje dle dat obsažených ve vygenerovaném souboru, kde jednotlivé znaky jsou uloženy ve formě pozic, vzhledem ke zvolenému místu vykreslení, pixelů tvořící znak. Funkce knihovny umožňují psát do jejich parametrů přímo text ve formě *string* nebo pole znaků.

Pro zobrazení čísel uložených v MCU bylo zapotřebí napsat jednoduchou funkci pro převod čísla ve formátu *double* se znaménkem do pole znaků *char*. Funkce umožňuje libovolné nastavení počtu číslic před a za desetinou tečkou. Výstup programu se nachází na následujícím příkladu (Obr. 6.19). V levé části je převedeno kladné číslo se odpovídajícím počtem míst před a za desetinou tečkou, vpravo záporné číslo s počty míst o jedno větší než je zapotřebí.

```

Zadejte číslo ve formátu double: 327.41
Zadejte rozumný počet číslic před desetinou čárkou: 3
Zadejte rozumný počet číslic za desetinou čárkou: 2
Prevek pole [0] =
Prevek pole [1] = 3
Prevek pole [2] = 2
Prevek pole [3] = 7
Prevek pole [4] = .
Prevek pole [5] = 4
Prevek pole [6] = 1
Pokračujte stisknutím libovolné klávesy...

Zadejte číslo ve formátu double: -189.53
Zadejte rozumný počet číslic před desetinou čárkou: 4
Zadejte rozumný počet číslic za desetinou čárkou: 3
Prevek pole [0] = -
Prevek pole [1] = 0
Prevek pole [2] = 1
Prevek pole [3] = 8
Prevek pole [4] = 9
Prevek pole [5] = .
Prevek pole [6] = 5
Prevek pole [7] = 3
Prevek pole [8] = 0
Pokračujte stisknutím libovolné klávesy...

```

**Obr. 6.19 – Výstup testovacího programu pro převod čísel double na pole znaků**

### 6.4.5. Způsob komunikace

Komunikace mezi PC a kvadcopterem probíhá po virtuální sériové lince prostřednictvím bluetooth využívající protokol SPP a je orientovaná pomocí paketů. Každý paket (Tab. 6.08) začíná identifikátorem, další byty tvoří data obsažená v paketu, pro kontrolu úspěšného přijetí paketu následuje kontrolní součet bytů obsažených v paketu a jako poslední je vyslán ukončovací znak.

**Tab. 6.08 – Struktura paketu komunikace mezi PC a kvadcopterem**

identifikátor	data paketu	kontrolní součet bytů v paketu	;
---------------	-------------	--------------------------------	---

Při příjmu jednotlivých bytů v MCU se vyvolá přerušení od jednotky USART2. Jednotlivé bity se zapisují do bufferu a po přijetí ukončovacího znaku paketu je nastaven indikátor přijetí kompletního paketu a v hlavní smyčce běžícího programu se dle identifikátoru vybere příslušné zpracování dat obsažených v paketu.

U zpracování dat probíhá nejprve ověření integrity dat paketu. Součet všech přijatých bytů se porovná se součtem obsaženým v paketu a dojde-li ke shodě, jsou teprve data zpracována. Po zpracování je odeslána zpráva o vykonané činnosti nebo je potvrzena správnost přijatých dat. Kontrola porovnáním součtu bytů je zavedena z bezpečnostních důvodů, aby nedošlo ke zranění například mylným nastavením výkonu motoru. Je-li zjištěn rozdíl mezi součty je odeslána nazpět zpráva s chybovým hlášením.

V následujícím přehledu (Tab. 6.09) (Tab. 6.10) je uveden význam identifikátorů paketu přijímanými MCU (RX pakety) nebo vyslanými MCU (TX pakety).

**Tab. 6.09 – RX pakety přijímané MCU**

Identifikátor	Význam
PWM	Nastavení výkonu motoru (jednoho nebo více)
STP	Nastavení minimální a maximální šířky PWM
POS	Nastavení pojistky
REG	Nastavení konstant regulátoru
STR	Nastavení periody regulační smyčky
SRZ	Nastavení žádané hodnoty regulátoru
SRE	Zapínání/vypínání regulačních smyček



Tab. 6.09 – RX pakety přijímané MCU - pokračování

Identifikátor	Význam
SRP	Nastavení výkonu ramen quadcopteru
RKO	Vynulování konstant regulátoru

Tab. 6.10 – TX pakety vysílané MCU

Identifikátor	Význam
SDP	Odeslání dat IMU do PC (pitch, roll)
SDM	Vyslání nastavení motorů
SPC	Odeslání dat IMU do PC (pitch, roll) + motory
SOK	Potvrzení úspěšného nastavení
CRC	Indikace chyby kontrolního součtu + motor stop
PWM	Potvrzení nastavení výkonu motoru
STP	Potvrzení nastavení max/min šířky impulzu PWM
POS	Potvrzení nastavení pojistky
ERR	Indikace špatného kontrolního součtu příkazu

#### 6.4.6. Software pro PC

Pro vývoj a nastavení regulace stability polohy bylo nutné vytvořit program, kterým je možné ovládat nastavení týkající se výkonů motorů quadcopteru a hlavně nastavovat a ovládat jednotlivé regulační smyčky.

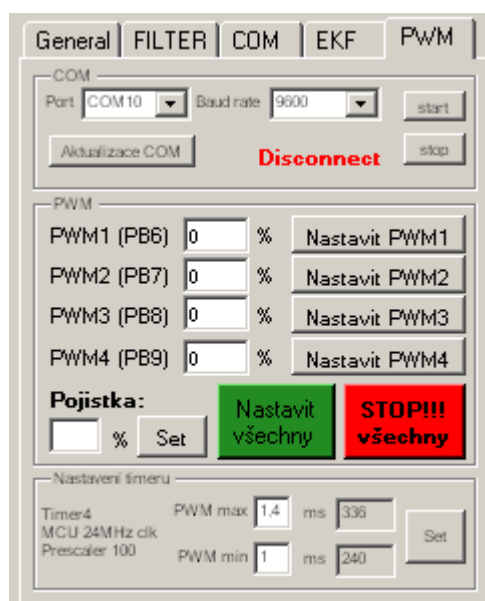
Jádro programu bylo převzato z aplikace pro modul CHR-6D. S tímto programem je umožněno měnit nastavení IMU modulu, otestovat jeho funkce a výstupní data přehledně zobrazit pomocí grafů (Obr. 6.20).



Obr. 6.20 – SW pro PC

Do programu byly implementovány funkce pro nastavení PWM (Obr. 6.21) a ovládání regulace, pravá část hlavního okna programu. Pro každý regulátor lze nezávisle nastavit jeho konstanty, žádanou hodnotu a rozložení výkonu mezi protilehlé motory pro regulační smyčky stabilizující rovnováhu v osách. Nechybí ani užitečné nastavení časové periody regulace a nulování stavu regulátoru. Uživatel má možnost ukládat aktuální data do souboru a jejich pomocí, ze zjištěného chování quadcopteru, nastavit konstanty regulátoru dle kapitoly (5.).

Po kliknutí tlačítkem myši na záložku PWM se zobrazí možnosti nastavení (Obr. 6.21). Komunikace s quadcopterem se zahájí tlačítkem „start“. Do políček jednotlivých PWM lze vyplnit výkony motorů v procentech. Jednotlivé motory lze vyzkoušet individuálně nebo poslat nastavení do všech najednou. Pro nouzové zastavení je přítomné tlačítko STOP, které vypne regulaci a zastaví všechny motory. Políčko pojistka dovolí omezit maximální výkon motorů z důvodu bezpečnosti. Nastavením časovače lze ovlivnit minimální a maximální šířku pulzu PWM. Změnou tohoto nastavení lze nalézt nejlepší využití procentuálního rozsahu výkonu.



Obr. 6.21 – Základní ovládání quadcopteru

#### 6.4.7. Software pro mobilní telefon

Dnešní mobilní telefony až na několik málo výjimek disponují technologií bluetooth a podporují aplikace psané v jazyce Java, konkrétně v J2ME určenou pro tato zařízení. Na základě této skutečnosti se nabízí elegantní využití mobilního telefonu jako dálkového ovládání quadcopteru.

Prostřednictvím internetu byl nalezen zdrojový kód<sup>3</sup> pro využití bluetooth prostřednictvím Java aplikace jako virtuálního sériového portu (SPP). Do aplikace bylo nutné implementovat ovládání quadcopteru. Její vývoj probíhal v programovacím prostředí NetBeans IDE 7.0, které je na vývoj aplikací pro mobilní telefony uzpůsobeno. K tomu, aby aplikace fungovala na konkrétním zařízení, je nutné nainstalovat jeho SDK. Pro vývoj na mobilním telefonu Nokia E65 byl stažen od výrobce softwarový balíček „S60 3rd Edition SDK Feature Pack 2 SDK“, který se implementoval do vývojového prostředí. SDK ke své činnosti vyžaduje nainstalovanou podporu jazyka Perl. Do projektu bylo zapotřebí ještě implementovat knihovnu „RXTX JAVA Communication API“, která umožňuje komunikaci po sériové lince.

Ukázka vzhledu aplikace (Obr. 6.22) v mobilním telefonu:



Obr. 6.22 – Aplikace v mobilním telefonu

Nalevo je ukázka spustitelného programu v mobilním telefonu, po spuštění aplikace (prostřední obrázek) se zobrazí hlavní menu, kde je možnost vyhledání zařízení bluetooth. Po dokončeném vyhledání uživatel zvolí nalezené zařízení quadcopteru a od tohoto okamžiku se aktivuje komunikace mezi quadcopterem a mobilním telefonem. Obrázek napravo ukazuje ovládací prostředí. Příkazy jsou vysílány stiskem jednotlivých tlačítek.

<sup>3</sup> [http://m.pandatron.cz/?740robot\\_ovladany\\_pres\\_bluetooth](http://m.pandatron.cz/?740robot_ovladany_pres_bluetooth)

## 7. Závěr

Na létajícím objektu quadcopteru byla vyzkoušena stabilizace polohy. Pro nastavení regulátoru byla použita metoda Ziegler-Nichols, kdy se postupně zvyšovala velikost proporcionální složky, která zajistila pravidelné kmitání quadcopteru kolem rovnovážné polohy. Zvyšování probíhalo do meze stability, aby bylo zajištěné stálé kmitání. Maximální hodnota proporcionální složky na mezi stability představuje kritické zesílení. Ze zaznamenaného průběhu byla určena kritická perioda. Podle přepočtení tabulky pro nastavení konstant regulátoru byl inicializován regulátor, ale výsledky nebyly uspokojivé. Při spuštěné regulaci je možné pozorovat zákmity. Rozdíl mezi realitou a simulací je znatelný.

Pro ovládání otáček motorů se vycházelo z principu funkce klasického relé a minimální šířka PWM impulsu byla nastavena na 1 ms a maximální na 2 ms. Vyzkoušením se ukázalo, že při hodnotě 2 ms má motor až příliš velké otáčky a pro vznesení stačí menší. Maximální šířka impulsu PWM byla redukována na 1,7 ms. Snížení bylo nutné provést i z důvodu regulace kdy skok o jedno procento výkonu motoru znamenal velkou změnu jeho otáček.

Ovládání regulátoru otáček motoru (ESC) vyžaduje řídicí pulzy o napěťové úrovni 5 V. Použitý MCU má výstupní úroveň 3,3 V. Motory se i s menším napětím podařilo roztočit, ale byla zpozorována nestabilita otáček. Přidáním napěťového translátoru byla zajištěna požadovaná úroveň napětí a chování motorů je již v normálu.

Způsob výpočtu úhlů náklonu pitch a roll je řešen pomocí akcelerometru. Při pohybu quadcopteru působí zrychlení v jednotlivých osách a tím vzniká odchylka od skutečných hodnot. V případě, kdy se jedná o stabilitu objektu na místě, tak nežádoucí zrychlení nejsou přítomna a tuto metodu výpočtu úhlů pomocí výstupních hodnot akcelerometru je možné použít. Nežádoucí je i chvění quadcopteru, kde hlavním zdrojem jsou jeho motory. Při dostatečně velkých otáčkách nehrozí ovlivnění IMU modulu.

## Seznam použité literatury:

- [1] **Ripka, Pavel a Típek, Alois. 2003.** *Master Book of Sensors*. Praha : BEN, 2003. str. 1020 (2 části). ISBN 0-12-752184-4.
- [2] **Devices, Analog. 2009.** Datasheet ADXL 335. [Online] 2009. [Citace: 14. 04. 2011.] <[http://www.analog.com/static/imported-files/data\\_sheets/ADXL335.pdf](http://www.analog.com/static/imported-files/data_sheets/ADXL335.pdf)>.
- [3] **Goce, ESA. 2011.** ESA - GOCE - Earth's gravity revealed in unprecedented detail. *ESA Portal*. [Online] 31. 03. 2011. [Citace: 29. 04. 2011.] <[http://www.esa.int/SPECIALS/GOCE/SEM1AK6UPLG\\_0.html](http://www.esa.int/SPECIALS/GOCE/SEM1AK6UPLG_0.html)>.
- [4] **Vojáček, Antonín. 2007.** Jak pracují nové 3D MEMS akcelerometry Freescale ? | HW.cz. *HW.cz / Vše o elektronice a programování*. [Online] 30. 04. 2007. [Citace: 23. 03. 2011.] <<http://hw.cz/Produkty/Nove-soucastky/ART1875-Jak-pracuji-nove-3D-MEMS-akcelerometry-Freescale-.html>>.
- [5] **Cef (Terry), Pearson. 1999.** *Gyroscopes - Everything you needed to know*. [Online] 07. 1999. [Citace: 25. 03. 2011.] <<http://www.gyroscopes.org/how%5Chagwa4.pdf>>.
- [6] **Winkler, Zbyněk, 2005.** Měření rychlosti (Robotika.cz > Guide). *Robotika.cz*. [Online] 11. 10. 2005. [Citace: 19. 02. 2011.] <<http://robotika.cz/guide/filtering/en>>.
- [7] **Kleinbauer, Rachael. 2004.** Kalman Filtering Implementation with Matlab. *OPUS-Datenbank*. [Online] 2004. [Citace: 26. 02. 2011.] <<http://elib.uni-stuttgart.de/opus/volltexte/2005/2183/pdf/kleinbauer.pdf>>.
- [8] **Welch, Greg a Gary, Bishop. 2006.** An Introduction to the Kalman Filter. *The Kalman Filter*. [Online] 24. 07. 2006. [Citace: 19. 02. 2011.] <[http://www.cs.unc.edu/~welch/media/pdf/kalman\\_intro.pdf](http://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf)>.
- [9] **Welch, Greg a Gary, Bishop. 1997.** The Kalman Filter. [Online] 24. 03. 1997. [Citace: 12. 03. 2011.] <<http://www.cs.unc.edu/~welch/kalman/>>.
- [10] **Premerlani, William a Bizard, Paul. 2009.** Direction Cosine Matrix IMU: Theory. *gentlenav - Firmware for Bill Premerlani's IMU based UAV Dev Board autopilots*. [Online] 17. 05. 2009. [Citace: 05. 05. 2011.] <<http://gentlenav.googlecode.com/files/DCMDraft2.pdf>>.

- [11] **Ing. Losenický, Miroslav CSc. 2009.** *Soustavy, regulátory-UPA – přednášky.* Pardubice : autor neznámý, 2009. str. 78.
- [12] **Doc. Ing. Dušek, František CSc. 2008.** *Teorie řídicích systémů - přednášky.* 2008. str. 70.
- [13] **CHR-6d IMU/Inclinometer** : Datasheet. *CH Robotics, Better Sensors.*  
[Online] [Citace: 16. 03. 2011.]  
<[http://www.chrobotics.com/docs/chr6d\\_datasheet.pdf](http://www.chrobotics.com/docs/chr6d_datasheet.pdf)>.
- [14] **Datasheet STM32VLDISCOVERY.** *STMicroelectronics.* [Online] [Citace: 07. 03. 2011.]  
<[http://www.st.com/internet/com/TECHNICAL\\_RESOURCES/TECHNICAL\\_LITERATURE/USER\\_MANUAL/CD00267113.pdf](http://www.st.com/internet/com/TECHNICAL_RESOURCES/TECHNICAL_LITERATURE/USER_MANUAL/CD00267113.pdf)>.
- [15] **Datasheet – Easy bluetooth module.** [Online] 14. 04. 2010.  
<<http://www.parallax.com/Portals/0/Downloads/docs/prod/comm/30085-EasyBluetooth-v1.3.pdf>>
- [16] **Pelikán, Daniel.** *RCM Pelikán.* [Online] [Citace: 10. 05. 2011.]  
<<http://www.rcm-pelikan.cz/index.php?sec=list&storage=21>>.
- [17] **Chmelař, Pavel. 2009.** *Navigace robota pomocí akcelerometru a gyroskopu.* Pardubice : Univerzita Pardubice, 2009. str. 45.
- [18] **TAUFER, Ivan, KOTYK, Josef a JAVŮREK, Milan. 2009.** *Jak psát a obhajovat závěrečnou práci bakalářskou, diplomovou, rigorózní, disertační, habilitační.* Pardubice : Univerzita Pardubice, 2009. str. 40.  
ISBN 978-80-7395-157-3.

Přílohy:

**Příloha 1** – CD:

Obsah adresáře:

**ChmelarP\_StabilizacePolohy\_PR\_2011.pdf** – vlastní text práce

složka **Software**:

- **Ovládání\_pomocí\_PC** – obsahuje software pro ovládání quadcopteru a nastavování regulátoru v PC včetně zdrojového kódu
- **Převod\_čísla\_na\_znaky** – obsahuje jednoduchý program pro převod čísla na pole znaků se zdrojovým kódem
- **StabilizacePolohy** – vlastní program MCU se zdrojovým kódem
- **BluetoothApplication** – aplikace dálkového ovládání pro mobilní telefon včetně zdrojového kódu

složka **Matlab**:

- **Filtrace dat** – zdrojové kódy pro filtraci
- **Nastavení\_konstant** – určení konstant regulátoru ze zaznamenaného průběhu

složka **Návrh** – obsahuje podklady pro výrobu desky plošného spoje