

UNIVERZITA PARDUBICE

FAKULTA ELEKTROTECHNIKY A INFORMATIKY

BAKALÁŘSKÁ PRÁCE

2017

Ondřej Němec

Univerzita Pardubice

Fakulta elektrotechniky a informatiky

Měření rychlosti CW radarem

Ondřej Němec

Bakalářská práce

2017

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: 2016/2017

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Ondřej Němec**
Osobní číslo: **I14015**
Studijní program: **B2612 Elektrotechnika a informatika**
Studijní obor: **Komunikační a mikroprocesorová technika**
Název tématu: **Měření rychlosti CW radarem**
Zadávající katedra: **Katedra elektrotechniky**

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

Cílem práce je navrhnout řetězec zpracování mezifrekvenčního signálu z Dopplerovského CW radaru. Rozsah zpracovávaných radiálních rychlostí bude uvažován 1 až 50 m/s. Požadované rozlišení v rychlosti je 0,1 m/s. Řetězec zpracování signálu bude složen z antialiasingového filtru, AD převodníku, 32 bit MCU, paměti pro uložení vzorků a sériového rozhraní pro odeslání dat do PC. Navzorkovaný signál bude zpracován rychlou Fourierovou transformací, dále bude proveden odhad rychlosti cíle. Předpokládá se přítomnost jednoho pohyblivého cíle v dosahu senzoru s vysílacím výkonem až 15 dBm v pásmu 24 GHz.

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování bakalářské práce: **tištěná**

Seznam odborné literatury:

- [1] BEZOUŠEK, Pavel a Pavel ŠEDIVÝ. Radarová technika. Vyd. 1. Praha: Vydavatelství ČVUT, 2004, vi, 217 s. ISBN 80-010-3036-9.
- [2] WOLFF, Christian. Radar tutorial [online]. 1998 [cit. 2014-10-11]. Dostupné z: <http://www.radartutorial.eu/>
- [3] MAHAFFZA, Bassem R a Atef Z ELSHERBENI. MATLAB simulations for radar systems design. Boca Raton, FL: CRC Press/Chapman, c2004, 682 p. cm. ISBN 15-848-8392-8.

Vedoucí bakalářské práce: **Ing. Michal Řezníček**

Datum zadání bakalářské práce: **31. října 2016**

Termín odevzdání bakalářské práce: **12. května 2017**



Ing. Zdeněk Němec, Ph.D.
děkan



L.S.



Ing. Jan Pidanič, Ph.D.
vedoucí katedry

V Pardubicích dne 15. listopadu 2016

Prohlášení autora

Prohlašuji, že jsem tuto práci 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.

V Pardubicích dne 12. 5. 2017

Ondřej Němec

PODĚKOVÁNÍ

Na tomto místě bych chtěl poděkovat Ing. Michalovi Řezníčkovi za velice cenné rady, připomínky a čas, který mi věnoval. Dále bych chtěl poděkovat své rodině a přítelkyni za to, že mi vždy byli oporou a vždy mě podporovali.

ANOTACE

Bakalářská práce se zabývá návrhem řetězce zpracování mezifrekvenčního signálu z Dopplerovského radarového senzoru. Navržený řetězec se skládá z antialiasingového filtru a mikroprocesoru. Mikroprocesor číslicově zpracovává navzorkovaný mezifrekvenční signál pomocí naprogramovaného algoritmu DIT pro výpočet Diskrétní Fourierovy transformace a také provádí základní odhad rychlosti.

KLÍČOVÁ SLOVA

CW radar, antialiasingový filtr, rychlá Fourierova transformace, rychlost, Dopplerův jev

TITLE

Velocity measurement by CW radar

ANNOTATION

This thesis deals with design of intermediate frequency signal processing chain from Doppler radar sensor. Designed chain consists of an anti-aliasing filter and a microcontroller. The microprocessor digitally processes the sampled intermediate frequency signal using the programmed DIT algorithm for calculating the Discrete Fourier transform, and also performs a basic velocity estimation.

KEYWORDS

CW radar, anti-aliasing filter, fast Fourier transform, velocity, Doppler's effect

OBSAH

Úvod.....	15
1 Základní poznatky.....	16
1.1 Radar	16
1.1.1 Princip funkce radaru a radarová rovnice	16
1.1.2 Impulzní radar	17
1.1.3 CW radar.....	18
1.2 Dopplerův jev	19
1.2.1 Princip měření rychlosti CW radarem	21
1.3 Reprezentace signálu ve frekvenční oblasti	24
1.3.1 Fourierova transformace	24
1.3.2 Diskrétní Fourierova transformace	25
1.3.3 Rychlá Fourierova transformace.....	25
1.3.4 Rychlá Fourierova transformace s redukcí v čase	26
1.4 Číslicové zpracování signálu.....	29
1.4.1 Analogový signál	29
1.4.2 Číslicový signál.....	29
1.4.3 Řetězec číslicového zpracování signálu	29
1.4.4 A/D převod	30
1.4.5 Aliasing	33
1.4.6 Antialiasingový filtr	36
1.4.7 Frakční čísla.....	36
1.4.8 IQ data.....	37
2 Návrh řetězce zpracování IF signálu	40
2.1 Blokové schéma systému pro zpracování IF signálu	40
2.2 Radarové senzory využité k měření rychlosti	40
2.2.1 CDM324	40

2.2.2	K_MC3	42
2.3	Návrh antialiasingových filtrů.....	44
2.3.1	Určení zisku antialiasingového filtru.....	44
2.3.2	Určení mezních frekvencí antialiasingového filtru.....	46
2.3.3	Určení ostatních parametrů antialiasingového filtru.....	46
2.3.4	Rovnice popisující aktivní dolní propust' typu Sallen-Key	47
2.3.5	Zapojení dolní propusti v topologii Sallen-Key s virtuální nulou	50
2.3.6	Sestavení antialiasingového filtru pro K_MC3 ze stupňů Sallen-Key	52
2.3.7	Sestavení antialiasingového filtru pro CDM324 ze stupňů Sallen-Key	53
2.3.8	Ověření funkce navržených filtrů	54
2.4	Mikroprocesor a nastavení periférií	55
2.4.1	Výběr mikroprocesoru	55
2.4.2	Konfigurace A/D převodníků	56
2.4.3	Konfigurace DMA	58
2.4.4	Sériové rozhraní.....	59
2.5	Výpočet Fourierovy transformace.....	59
2.5.1	Rozbor algoritmu DIT FFT.....	59
2.5.2	Algoritmus DIT v matlabu.....	62
2.5.3	Algoritmus DIT v jazyce C.....	64
2.5.4	Knihovna CMSIS DSP	66
2.5.5	Porovnání jednotlivých knihoven FFT	66
2.6	Odhad rychlosti	69
2.7	Výsledné schéma, DPS a program pro mikroprocesor	71
2.7.1	Schéma a DPS.....	71
2.7.2	Program pro mikroprocesor	73
2.8	Ověření funkce celého řetězce zpracování IF signálu.....	73
2.8.1	Metodika měření	73

2.8.2	Rozbor naměřených dat pomocí senzoru K_MC3.....	74
2.8.3	Rozbor naměřených dat pomocí senzoru CDM324.....	77
2.8.4	Vzájemné porovnání obou senzorů.....	79
	Závěr	81
	Použitá literatura	83
	Příloha A – seznam příloh na CD	85

Seznam ilustrací

Obrázek 1.1 – Princip monostatického aktivního radiolokátoru [1].....	16
Obrázek 1.2 – Princip impulzního radaru [1].	17
Obrázek 1.3 – Homodynní přijímač [2].....	18
Obrázek 1.4 – Ilustrace vzinku Dopplerova jevu [2].....	20
Obrázek 1.5 – Umístění radarového senzoru na pohybujícím se objektu [4, 5].....	22
Obrázek 1.6 – Orientovaný graf motýlek	28
Obrázek 1.7 – Blokové schéma řetězce pro zpracování číslicového signálu.	30
Obrázek 1.8 – Analogový signál.....	30
Obrázek 1.9 – Časový průběh vzorkovací funkce [9].	31
Obrázek 1.10 – Signál spojitý v amplitudě a nespojitý v čase.	31
Obrázek 1.11 – Číslicový signál.	32
Obrázek 1.12 – Spektrum vzorkovací funkce [9].....	34
Obrázek 1.13 – Smyšlené spektrum vzorkovaného signálu [9].....	35
Obrázek 1.14 – Spektrum ovzorkovaného signálu bez aliasingu [9].	35
Obrázek 1.15 – Spektrum ovzorkovaného signálu s aliasingem [9].....	36
Obrázek 1.16 – Komplexní číslo zakreslené v komplexní rovině.	37
Obrázek 1.17 – Modifikovaný homodynní přijímač pro složky I a Q [2].....	38
Obrázek 2.1 – Zamýšlené blokové schéma řetězce zpracování IF signálu.	40
Obrázek 2.2 – Radarový senzor CDM324.....	41
Obrázek 2.3 – Blokové schéma senzoru CDM324 [13].	41
Obrázek 2.4 – Radarový senzor K_MC3	42
Obrázek 2.5 – Blokové schéma senzoru K_MC3 [14].	43
Obrázek 2.6 – Schéma zapojení pro zjištění zisku AAF.	45
Obrázek 2.7 – Aktivní filtr v topologii Sallen-Key s obecnými admitancemi [16, 17, 18].....	48
Obrázek 2.8 – Aktivní dolní propust' v topologii Sallen-Key [16, 17].....	49
Obrázek 2.9 – Aktivní dolní propust' v topologii Sallen-Key se zakresleným napájením [16, 17].....	51
Obrázek 2.10 – Aktivní dolní propust' v topologii Sallen-Key s virtuální nulou [17].....	51
Obrázek 2.11 – Konfigurace A/D převodníku.	58
Obrázek 2.12 – Rozložení vzorků v paměti MCU.....	59
Obrázek 2.13 – Algoritmus DIT zakreslený pomocí orientovaných grafů [8].....	60

Obrázek 2.14 – Algoritmus DIT zakreslený pomocí orientovaných grafů s bitově inverzním pořadí [8].....	62
Obrázek 2.15 – Kód v matlabu pro výpočet fázovacích členů.	62
Obrázek 2.16 – Skript v matlabu pro předpočítání tabulek pro bitově inverzní pořadí.	63
Obrázek 2.17 – Skript v matlabu pro výpočet DFT pomocí algoritmu FFT.	64
Obrázek 2.18 – Závislost času výpočtu FFT jednotlivých algoritmů na počtu vzorků	68
Obrázek 2.19 – Závislost obsazení programové paměti MCU jednotlivých algoritmů na počtu vzorků	69
Obrázek 2.20 – Spektrum IF signálu získaného pomocí senzoru K_MC3.....	70
Obrázek 2.21 – Spektrum IF signálu s vykresleným prahem detekce.....	71
Obrázek 2.22 – Osazená DPS	72
Obrázek 2.23 – Grafické zobrazení jednoho odběhu dat.....	75
Obrázek 2.24 – Spektrum IF signálu získaného pomocí senzoru K_MC3.....	76
Obrázek 2.25 – Spektrum IF signálu s vykresleným prahem detekce a chybně provedeným odhadem rychlosti.....	77
Obrázek 2.26 – Grafické zobzrazení jednoho odběhu dat.....	78
Obrázek 2.27 – Spektrum IF signálu s vykresleným prahem detekce.....	79

Seznam tabulek

Tabulka 1 – Vybrané parametry senzoru CDM324 [13].	42
Tabulka 2 – Vybrané parametry senzoru K_MC3 [14].	44
Tabulka 3 – Naměřená data algoritmů FFT ve formátu Q1.15.	67
Tabulka 4 – Naměřená data algoritmů FFT ve formátu Q1.31.	67

Seznam zkratk a značek

AAF	Aktivní antialiasingový filtr
AC	Střídavé napětí/proud
ADC	A/D převodník
CFAR	Konstantní úroveň falešného poplachu
CW	Kontinuální signál
DC	Stejnoseměrné napětí/proud
DFT	Diskrétní Fourierova transformace
DIT	Redukce v čase
EIRP	Ekvivalentně izotropicky vyzářený výkon
FFT	Rychlá Fourierova transformace
FMCW	Frekvenčně modulovaný kontinuální signál
FT	Fourierova transformace
IF	Mezifrekvenční
LCD	LCD displej
LNA	Nízkošumový zesilovač
MCU	Mikrokontrolér
RCS	Efektivní odrazivá plocha
SPORT	Sériový port

Úvod

Tato práce se zabývá návrhem řetězce zpracování mezifrekvenčního signálu z Dopplerovského CW radaru. Především je řešen návrh antialiasingového filtru, převod mezifrekvenčního signálu na signál číslicový a následné zpracování číslicového signálu pomocí rychlé Fourierovy transformace.

Práce je rozdělena na dvě hlavní části, teoretickou a praktickou. V teoretické části jsou zmíněny všechny teoretické poznatky potřebné ke správnému návrhu řetězce zpracování mezifrekvenčního signálu. Praktická část pak pojednává o samotném návrhu řetězce zpracování mezifrekvenčního signálu.

První kapitola teoretické části vysvětluje základní princip CW radaru s jeho blokovým schématem a také uvádí rovnice popisující výstupní mezifrekvenční signál CW radarového senzoru. V kapitole druhé je popsán Dopplerův jev pro elektromagnetické vlnění a také je zde uveden jeden z principů měření rychlosti objektu pomocí CW radaru. Třetí teoretická kapitola se zabývá reprezentací signálu ve frekvenční oblasti, kde jsou uvedeny základní definiční vztahy pro Fourierovu transformaci a Diskrétní Fourierovu transformaci. Dále jsou v této kapitole odvozeny vztahy popisující algoritmus DIT pro výpočet DFT. Čtvrtá teoretická kapitola se zabývá převodem analogového signálu na signál digitální.

V první kapitole praktické části práce je proveden odhad podoby celého řetězce zpracování mezifrekvenčního signálu. Kapitola druhá obsahuje popis použitých radarových senzorů. V třetí praktické kapitole je pak proveden rozbor parametrů mezifrekvenčního signálu. Výsledky rozboru jsou pak použity ke stanovení parametrů aktivního antialiasingového filtru. Ve stejné kapitole je pak proveden kompletní návrh antialiasingového filtru podle stanovených parametrů. Čtvrtá kapitola je pak věnována výběru vhodného mikroprocesoru ke zpracování navzorkovaného mezifrekvenčního signálu a nastavení jeho integrovaných periférií, především A/D převodníků. Následující pátá kapitola je věnována naprogramování algoritmu DIT v jazyce C a ověření jeho funkce. V této kapitole je také provedeno porovnání naprogramovaného algoritmu s jinou knihovnou pro výpočet FFT. Šestá kapitola praktické části je věnována provedení odhadu rychlosti objektu z naměřených dat. Předposlední kapitola pak popisuje celkové schéma a DPS navrženého řetězce zpracování mezifrekvenčního signálu a také je zde popsán program pro výslednou DPS. V poslední kapitole je pak ověřena funkce navrženého zařízení s jednotlivými radarovými senzory a také je zde provedeno zhodnocení dosažených parametrů celého řetězce zpracování IF signálu.

1 Základní poznatky

V této kapitole je pojednáno o teoretických poznatcích potřebných k návrhu řetězce zpracování mezifrekvenčního signálu z Dopplerovského CW radaru.

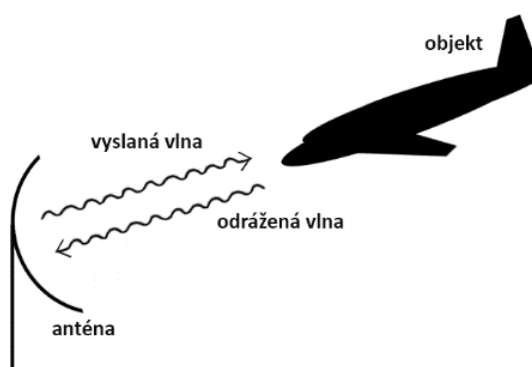
1.1 Radar

Radar je zařízení, jehož název pochází z anglického výrazu „radio detection and ranging“ což lze volně přeložit jako „detekce předmětů a měření jejich vzdálenosti pomocí rádiových vln“. Radar se tedy používá k detekci předmětů, měření jejich vzdálenosti a polohy vzhledem k radaru [1].

Dříve se radary používaly hlavně k ostraze vzdušného prostoru před nepřátelskými letadly, tedy v armádě. Dnes jsou radary mnohem více rozšířená zařízení, která se mimo jiné používají v letecké a lodní dopravě, k navádění raket a v poslední době dochází k jejich využití i v automobilové dopravě, kde jsou například používány jako součást adaptivních tempomatů nebo antikolizních systémů.

1.1.1 Princip funkce radaru a radarová rovnice

Princip monostatického impulzního aktivního radiolokátoru ukazuje obrázek 1.1. Radar vysílá elektromagnetickou vlnu do svého okolí. Pokud se v dosahu radiolokátoru objeví objekt, vyslaná vlna se od tohoto objektu odrazí a radar ji zachytí pomocí přijímací antény [1].



Obrázek 1.1 – Princip monostatického aktivního radiolokátoru [1].

Radarová rovnice (1.1) popisuje poměr mezi vyslaným výkonem a výkonem přijatého odrazu

$$P_p = P_v G_v G_p \frac{\lambda^2 \sigma}{(4\pi)^3 R^4} \frac{1}{L_{at}^2} \quad (1.1)$$

, kde

P_p ...přijatý výkon,

P_v ...vyslaný výkon,

G_v ...zisk vysílací antény,

G_p ...zisk přijímací antény,

λ ...vlnová délka,

σ ...efektivní odrazivá plocha objektu (RCS – „Radar cross section“)

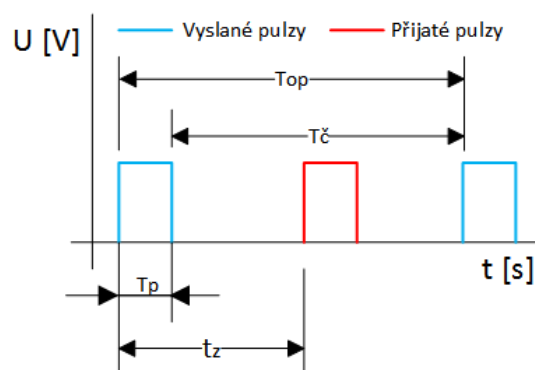
R ...šikmá vzdálenost,

L_{at} ...ztráty v atmosféře.

Přijatý výkon závisí na vysílacím výkonu, zisku přijímací antény, zisku vysílací antény a vlnové délce vysílaného signálu. Dominantním členem je však člen $\frac{1}{R^4}$, který říká, že přijatý výkon klesá se čtvrtou mocninou vzdálenosti od radaru. To je zapříčiněno tím, že se signál nejdříve šíří od radaru k objektu, od kterého se odráží, a pak se stejnou cestou šíří zpět k radaru. Přijatý výkon je také přímo úměrný efektivní odrazivé ploše σ neboli RCS (radar cross section). Rozměrem této veličiny je m^2 , nejedná se však o prostou plochu určenou z geometrických rozměrů objektu, který danou vlnu odráží. RCS závisí jak na geometrických rozměrech tělesa, tak i na úhlu dopadu vlny na těleso, nebo na materiálu, ze kterého je toto těleso vyrobeno. Obecně není určení toho parametru jednoduché [1].

1.1.2 Impulzní radar

Funkci impulzního radaru ilustruje obrázek 1.2. Tento radar pracuje tak, že se vždy na počátku cyklu přepne do vysílacího režimu, po určitou dobu T_p vysílá a následně se po určitém časovém intervalu přepne do přijímacího režimu, ve kterém po určitou dobu T_{ζ} očekává odražené impulzy. Tento děj se neustále opakuje s periodou T_{OP} [1].



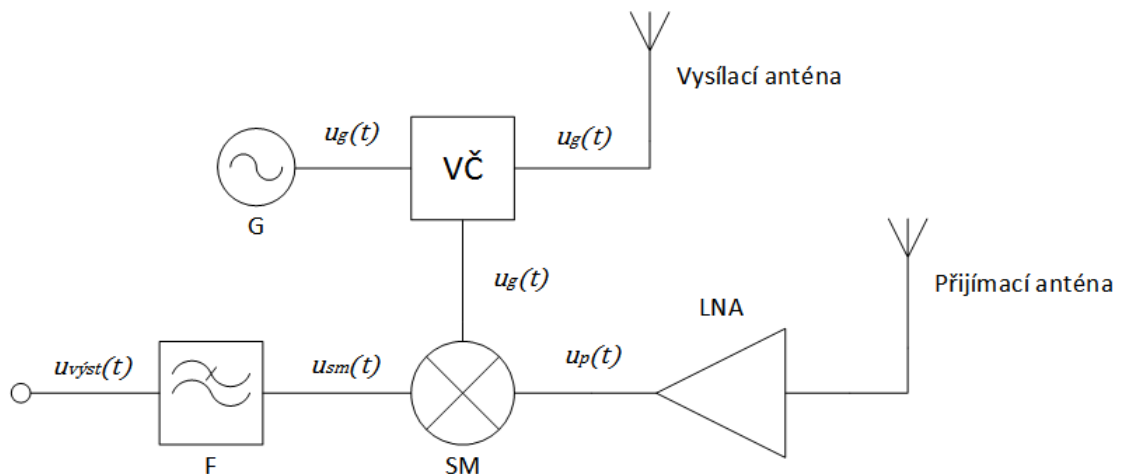
Obrázek 1.2 – Princip impulzního radaru [1].

1.1.3 CW radar

Radar se spojitým signálem neboli „Continuous Wave radar“ neustále vysílá elektromagnetickou energii do svého okolí a na rozdíl od radaru impulzního zároveň přijímá odrazy od objektů v dosahu radaru, které následně zpracovává.

CW radary se využívají hlavně tam, kde není třeba vysoký vysílací výkon. Nevýhodou tohoto druhu radarů je, že s využitím jednoho přijímače a vysílače nelze měřit vzdálenost cíle od radaru. Aby bylo možné určit vzdálenost objektu od radaru, je třeba využít například frekvenční modulace vysílaného signálu. Takto upravený CW radar se nazývá FMCW radarem neboli „frequency modulated continuous wave radar“. CW radar s jedním vysílačem a přijímačem, který nevyužívá žádné modulace vysílaného signálu lze využít pouze k měření vzájemné radiální rychlosti přijímače a cíle, na základě Dopplerova jevu a za určitých podmínek k měření změn vzdálenosti [2].

Jednou z možností realizace přijímače CW radaru je přímé použití zeslabeného signálu z generátoru vysílače ke směšování v přijímací straně. Takovému přijímači se říká homodynní, nebo přijímač s přímým směšováním. U tohoto druhu přijímače je signál směšován do základního pásma. Blokové schéma homodynního přijímače ukazuje obrázek 1.3 [2].



Obrázek 1.3 – Homodynní přijímač [2].

Signál z generátoru G označený jako $u_G(t)$, rovnice (1.2), je přiveden přes vazební člen VČ na vysílací anténu, kterou je v podobě elektromagnetické energie vyzářen do okolí radaru. Signál, který se odrazí od cíle je zachycen přijímací anténou a zesílen nízkošumovým zesilovačem

označeným LNA, toto označení pochází z anglického „low noise amplifier“. Přijatý a zesílený signál označen nyní jako $u_p(t)$, rovnice (1.3), je přiveden na směšovač SM spolu se zeslabeným signálem $u_G(t)$, který je přiveden z druhé odbočky vazebního členu VČ. Pomocí rovnic (1.2) a (1.3) popisující vstupní signály směšovače spolu s rovnicí (1.4), která popisuje operaci prováděnou se signály ve směšovači SM, lze odvodit vztah pro výstupní signál směšovače $u_{SM}(t)$, rovnice (1.5). Tuto rovnici lze pomocí goniometrických vzorců upravit do tvaru podle rovnice (1.6).

$$u_G(t) = U_{GMAX} \cos(2\pi f_G t) \quad (1.2)$$

$$u_p(t) = U_{PMAX} \cos(2\pi f_P t) \quad (1.3)$$

$$u_{SM}(t) = u_p(t)u_G(t) \quad (1.4)$$

$$u_{SM}(t) = \frac{U_{GMAX}U_{PMAX}}{1} \cos(2\pi f_G t) \cos(2\pi f_P t) \quad (1.5)$$

$$u_{SM}(t) = \frac{U_{GMAX}U_{PMAX}}{2} \cos[2\pi(f_G - f_P)t] + \frac{U_{GMAX}U_{PMAX}}{2} \cos[2\pi(f_G + f_P)t] \quad (1.6)$$

Signál $u_{SM}(t)$ směšovače je dále přiveden na filtr F typu dolní propust', který odstraní součtovou složku tohoto signálu a vznikne výstupní signál $u_{výst}$ popsany rovnicí (1.7).

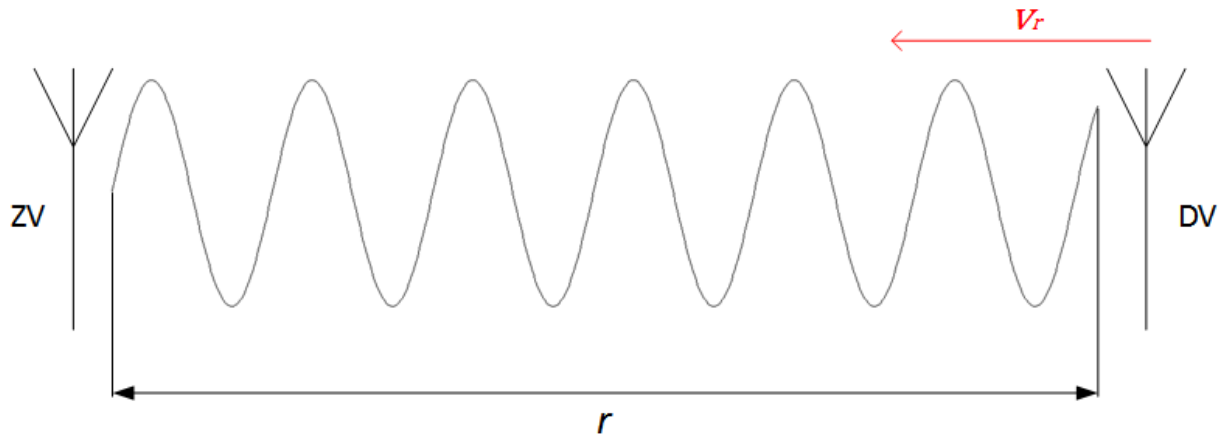
$$u_{výst}(t) = \frac{U_{GMAX}U_{PMAX}}{2} \cos[2\pi(f_G - f_P)t] \quad (1.7)$$

Kde U_{GMAX} je amplituda okamžitého napětí $u_G(t)$ generátoru G, U_{PMAX} je amplituda přijatého napětí $u_p(t)$, f_G je frekvence napětí generátoru, f_P je frekvence přijatého signálu a $u_{SM}(t)$ je okamžitá hodnota napětí na výstupu směšovače SM.

1.2 Dopplerův jev

Dopplerův jev byl objeven rakouským fyzikem Johanem Christianem Dopplerem v roce 1842 a roku 1845 jej v Holandsku experimentálně ověřil Buys Ballot. Tento jev se projevuje u všech druhů vlnění, například u zvuku, elektromagnetických vln i u viditelného světla. Příčinou tohoto jevu je vzájemná nenulová radiální rychlost zdroje vlnění a detektoru tohoto vlnění. Příkladem důsledku tohoto jevu může být například situace, kdy automobil vysílá do svého

okolí zvuk s frekvencí 1000 Hz. Pokud se detektor vůči tomuto automobilu pohybuje, bude detekovat jinou hodnotu frekvence zvuku, například 1096 Hz pro radiální rychlost 120 km/h. V radarové technice se tohoto jevu využívá například k měření radiální rychlosti cíle [3].



Obrázek 1.4 – Ilustrace vzinku Dopplerova jevu [2].

Obrázek 1.4 znázorňuje situaci, kde je zdroj elektromagnetického vlnění ZV v klidu a detektor vlnění DV je vůči tomuto zdroji v pohybu s radiální rychlostí v_R . Zdroj elektromagnetického vlnění vysílá do svého okolí vlnění s vlnovou délkou λ_V a tomu odpovídající frekvencí f_V , tyto dvě veličiny jsou spolu svázány rovnicí (1.8), kde c je rychlost světla, tedy i rychlost šíření elektromagnetického vlnění ve vakuu.

$$\lambda_V = \frac{c}{f_V} \quad (1.8)$$

Vlnění vyslané zdrojem ZV se dostává k detektoru vlnění DV po uražení vzdálenosti r . Jedné uražené vlnové délce vysílaného signálu λ_V pak odpovídá úhel 2π . Z této znalosti lze napsat pomocí přímé úměry rovnici (1.9), protože uražené vzdálenosti r pak odpovídá úhel φ [2].

$$r = \lambda_V \frac{\varphi}{2\pi} \quad (1.9)$$

Detektor se přibližuje ke zdroji s radiální rychlostí, pro kterou platí rovnice (1.10).

$$v_R = \frac{dr}{dt} \quad (1.10)$$

Pro frekvenci obecně platí vztah (1.11).

$$f = \frac{1}{2\pi} \frac{d\varphi}{dt} \quad (1.11)$$

Derivací rovnice (1.9) podle času vznikne rovnice (1.12).

$$\frac{dr}{dt} = \frac{\lambda_V}{2\pi} \frac{d\varphi}{dt} \quad (1.12)$$

Dosazením rovnic (1.8), (1.10) a (1.11) do rovnice (1.12) a mírnou úpravou vznikne rovnice (1.13), kde f_D je Dopplerova frekvence [2].

$$f_D = \frac{f_V v_R}{c} \quad (1.13)$$

Pro frekvenci f_{DV} detekovanou detektorem vlnění DV platí vztah (1.14), který říká, že frekvence f_{DV} je dána součtem vyslané frekvence f_V a frekvence Dopplerovy f_D [2, 3].

$$f_{DV} = f_V + f_D \quad (1.14)$$

Dosazením rovnice (1.13) do rovnice (1.14) a mírnou úpravou vznikne rovnice (1.15), což je výsledný vztah pro frekvenci detekovanou detektorem DV [2].

$$f_{DV} = f_V \left(1 + \frac{v_R}{c}\right) \quad (1.15)$$

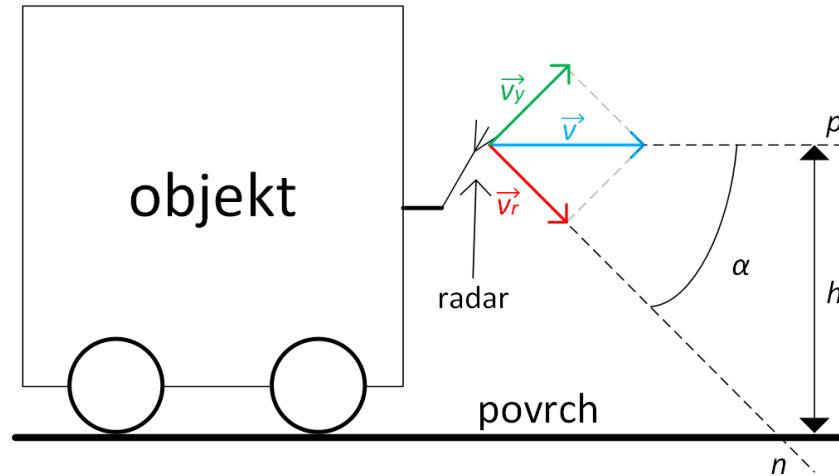
Podle rovnice (1.15) lze říci, že pokud se zdroj a detektor vlnění vůči sobě pohybují zápornou radiální rychlostí v_R , tedy se od sebe vzdalují, bude frekvence f_{DV} detekovaná detektorem vlnění menší než frekvence f_V . Naopak bude-li tato rychlost kladná a detektor se zdrojem vlnění se budou vůči sobě přibližovat, bude frekvence f_{DV} větší než frekvence f_V . Pro případ nulové radiální rychlosti bude frekvence f_{DV} rovna frekvenci f_V , k žádnému posuvu frekvence tedy nedojde [2].

1.2.1 Princip měření rychlosti CW radarem

Jak již bylo uvedeno v kapitole (1.1.3), CW radarem lze pomocí Dopplerova jevu měřit rychlost objektu. Jednou z možností, jak lze toto měření provést, je umístění senzoru na pohyblivý objekt, který je namířen na stacionární cíl.

Tuto situaci ilustruje obrázek 1.5, kde je radar umístěn na objekt, který se vůči stacionárnímu povrchu pohybuje s rychlostí \vec{v} . Radarový senzor je na tomto objektu upevněn ve výšce h nad povrchem. Osa n hlavního laloku antény radaru je od vodorovné přímky p odkloněna o úhel

α směrem k povrchu. Radar pak ve směru osy n vysílá elektromagnetickou energii, která se od povrchu částečně odráží a v důsledku Dopplerova jevu vznikne v přijímaném signálu frekvenční posuv, ze kterého lze určit rychlost objektu, na kterém je radar umístěn [4, 5].



Obrázek 1.5 – Umístění radarového senzoru na pohybujícím se objektu [4, 5]

CW radar měří radiální rychlost \vec{v}_R a proto je nutné rychlost \vec{v} rozložit do dvou vzájemně kolmých složek \vec{v}_R a \vec{v}_Y . Rychlost \vec{v}_R je rovnoběžná s osou hlavního laloku antény n . Pro přepočítání mezi velikostmi rychlostí \vec{v} a \vec{v}_R lze psát vztah (1.16) [4, 5].

$$|v_R| = |v| \cos(\alpha) \quad (1.16)$$

Pro vyjádření frekvence přijímaného signálu f_P je nutné uvažovat Dopplerův posun dvakrát, protože nejdříve je signál s frekvencí f_V vyslán z pohybujícího se zdroje s radiální rychlostí \vec{v}_R vůči povrchu, tento signál se následně částečně odráží od tohoto povrchu a je detekován přijímací anténou radaru, která se také pohybuje s rychlostí \vec{v}_R . Pro frekvenci signálu dopadajícího na povrch lze psát rovnici (1.17) [2].

$$f_{POVRCH} = f_V \left(1 + \frac{v_R}{c}\right) \quad (1.17)$$

Signál o frekvenci f_{POVRCH} se následně odrazí a je přijímán přijímací anténou radaru. Pro frekvenci přijímaného signálu lze psát rovnici (1.18).

$$f_P = f_{POVRCH} \left(1 + \frac{v_R}{c}\right) \quad (1.18)$$

Dosazením rovnice (1.17) do (1.18) lze získat rovnici (1.19).

$$f_P = f_V \left(1 + \frac{v_R}{c}\right) \left(1 + \frac{v_R}{c}\right) \quad (1.19)$$

Roznásobením členů v rovnici (1.19), lze dojít k rovnici (1.20)

$$f_P = f_V \left(1 + \frac{2v_R}{c} + \frac{v_R^2}{c^2}\right) \quad (1.20)$$

Člen $\frac{v_R^2}{c^2}$ je možné v této rovnici zanedbat vzhledem k jeho malé velikosti vůči členu $\frac{2v_R}{c}$. Touto úpravou vznikne výsledná rovnice (1.21) pro frekvenci přijímaného signálu f_P [2].

$$f_P = f_V \left(1 + \frac{2v_R}{c}\right) \quad (1.21)$$

Signál přijatý přijímací anténou a zpracovávaný homodynním přijímačem bude po přijetí nejdříve zesílen pomocí LNA. Pro tento signál platí rovnice (1.22), kde $U_{P_{MAX}}$ je amplituda přijatého signálu po zesílení pomocí LNA.

$$u_p(t) = U_{P_{MAX}} \cos \left[2\pi f_V \left(1 + \frac{2v_R}{c}\right) t \right] \quad (1.22)$$

Signál $u_p(t)$ je následně směšován se zeslabeným vyslaným signálem, který má tvar podle rovnice (1.23) [4].

$$u_{v_{REP}}(t) = U_{V_{MAX_{REP}}} \cos(2\pi f_v t) \quad (1.23)$$

S využitím rovnice (1.6) lze pro signál na výstupu směšovače psát rovnici (1.24).

$$u_{SM}(t) = \frac{U_{V_{MAX_{REP}}} U_{P_{MAX}}}{2} \cos \left[2\pi (f_v - f_V \left(1 + \frac{2v_R}{c}\right)) t \right] + \frac{U_{V_{MAX_{REP}}} U_{P_{MAX}}}{2} \cos \left[2\pi (f_v + f_V \left(1 + \frac{2v_R}{c}\right)) t \right] \quad (1.24)$$

Úpravou dostáváme rovnici (1.25).

$$u_{SM}(t) = \frac{U_{V_{MAX_{REP}}} U_{P_{MAX}}}{2} \cos \left(2\pi \frac{2v_R f_V}{c} t \right) + \frac{U_{V_{MAX_{REP}}} U_{P_{MAX}}}{2} \cos \left[2\pi (f_v + f_V \left(1 + \frac{2v_R}{c}\right)) t \right] \quad (1.25)$$

Výstupní filtr homodynního přijímače potlačí součtovou složku signálu $u_{SM}(t)$ a na jeho výstupu je signál u_{IF} v tvaru podle rovnice (1.26) [4, 5].

$$u_{IF}(t) = \frac{U_{VMAXREP} U_{PMAX}}{2} \cos\left(2\pi \frac{2v_R f_V}{c} t\right) \quad (1.26)$$

Signál $u_{IF}(t)$ je nízkofrekvenční signál obsahující pouze frekvenci úměrnou velikosti radiální rychlosti v_R objektu pohybujícím se po povrchu. Je třeba však podotknout, že při odvození této rovnice bylo zanedbáno značné množství faktů, se kterými je nutné při reálném návrhu tohoto systému počítat. Přijímací i vysílací anténa totiž není úzce směrová. Díky tomu vysílací anténa ozařuje mnohem větší úsek povrchu, nikoliv jen velice úzký v ose n . Přijatý signál pak bude složen z více frekvenčních složek, které budou pro daný bod úměrné příslušné radiální rychlosti, plynoucí z proměnného úhlu α . Dále byl zanedbán šum generovaný okolím i vlastními obvody přijímače [2, 4, 5].

1.3 Reprezentace signálu ve frekvenční oblasti

Signály jsou většinou popisovány v časové oblasti, lze je však popsat i v oblasti frekvenční (spektrální). Pro časově nezávislé systémy (LTI – Linear Time Invariant) platí, že popis signálu v časové a frekvenční oblasti je ekvivalentní, obsahuje tedy veškeré informace o signálu. k převodu signálu z časové do frekvenční oblasti slouží řada nástrojů. U periodických signálů je tento převod realizován pomocí Fourierovy řady. U signálů neperiodických pak Fourierovou transformací. Pro signály diskrétní je tento převod realizován pomocí diskrétní Fourierovy transformace [6].

1.3.1 Fourierova transformace

Spektrum periodických signálů lze určit pomocí Fourierovy řady. Toto spektrum je čárové s amplitudami na celých násobcích základní frekvence, základní frekvenci pak odpovídá základní perioda. Pokud se začne tato perioda zvětšovat, budou se čáry ve spektru periodického signálu přibližovat k sobě, obálka spektra zůstane stejná. Pokud se základní frekvence bude zvětšovat až k nekonečnu, budou spektrální čáry nekonečně blízko u sebe. Spektrum se stane spojitým. Zvětšování základní periody signálu je hlavní myšlenka při odvození vztahů pro Fourierovu transformaci [6].

Vztah (1.27) je vztahem pro tzv. přímou nebo dopřednou Fourierovu transformaci. Zpětnou Fourierovu transformaci pak určuje vztah (1.28)

$$S(\omega) = \int_{-\infty}^{\infty} s(t) e^{-j\omega t} dt \quad (1.27)$$

$$s(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} S(\omega) e^{j\omega t} d\omega \quad (1.28)$$

Aby bylo možné určit spektrum $S(\omega)$ signálu $s(t)$, je nutné, aby tento signál vyhovoval Dirichletovým podmínkám. Musí mít konečný počet nespojitostí, konečný počet extrémů a musí být absolutně integrovatelný [7].

1.3.2 Diskrétní Fourierova transformace

Fourierova transformace a Fourierova řada jsou nástroji, kterými lze převést signál do spektrální oblasti a zpět. Pouze však v případě, že lze tento signál matematicky popsat a lze určit potřebné primitivní funkce dané Fourierovou transformací nebo řadou. V poslední době se signály zpracovávají většinou číslicově, což znamená, že není znám matematický popis signálu, ale jen jeho vzorky a pozice těchto vzorků v čase. Pro tyto případy lze spektrum signálu určit numericky pomocí diskrétní Fourierovy transformace (DFT). Jako přímá diskrétní Fourierova transformace je označován vztah (1.29) [8].

$$S(k) = \sum_{n=0}^{N-1} s(n) e^{-\frac{j2\pi}{N}kn} ; \quad k = 0, 1, \dots, N-1 \quad (1.29)$$

Inverzní diskrétní Fourierovu transformaci vyjadřuje vztah (1.30).

$$s(n) = \frac{1}{N} \sum_{k=0}^{N-1} S(k) e^{\frac{j2\pi}{N}kn} ; \quad n = 0, 1, \dots, N-1 \quad (1.30)$$

V obou předchozích vztazích $S(k)$ představuje k -tý člen spektra, $s(n)$ představuje n -tý člen vstupní posloupnosti vzorků, k představuje index ve spektru, n představuje index v posloupnosti vzorků signálu, N představuje počet vzorků posloupnosti signálu, nebo počet členů spektra.

1.3.3 Rychlá Fourierova transformace

Rychlá Fourierova transformace, zkráceně FFT (Fast Fourier Transform), je soubor algoritmů, které zrychlují výpočet DFT. DFT je velice výpočetně náročná, protože pro každé k je nutné provést N komplexních násobení a N součtů. Celkově je tedy třeba provést N^2 komplexních

násobení a stejný počet součtů. Doba výpočtu DFT bez aplikace algoritmů FFT je neúnosně dlouhá a nelze ji ani v dnešní době dobře použít. Jedním z prvních a nezákladnějších algoritmů FFT je algoritmus z roku 1965, který znovuobjevili Cooley a Turkey. Základní myšlenkou tohoto algoritmu je rozdělení velkého problému na několik menších, lépe řešitelných. V dnešní době je známo podstatně více algoritmů FFT, které se dají kombinovat pro dosažení co nejrychlejších výpočtů. Tyto algoritmy používané na moderní výpočetní technice umožňují zjišťování spektra signálu prakticky v reálném čase [8, 10].

1.3.4 Rychlá Fourierova transformace s redukcí v čase

Tento algoritmus se v anglické literatuře označuje jako „Decimation In Time“, zkráceně DIT. Jak již bylo uvedeno, tento algoritmus využívá rozdělení velkého problému na více menších, snáze řešitelných. Pokud je počet vzorků N roven mocnině čísla 2, lze vždy rozdělit posloupnost $S(k)$, vztah (1.31), na posloupnost sudých a lichých vzorků, což ukazuje vztah (1.33). Toto je možné v důsledku linearity DFT. Význam členu W_N ukazuje rovnice (1.32), což je komplexní exponenciála s periodou N [8, 10].

$$S(k) = \sum_{n=0}^{N-1} s(n) W_N^{nk} \quad (1.31)$$

$$W_N = e^{\frac{j2\pi}{N}} \quad (1.32)$$

$$S(k) = \sum_{n=0}^{\frac{N}{2}-1} s(2n) W_N^{2nk} + \sum_{n=0}^{\frac{N}{2}-1} s(2n+1) W_N^{(2n+1)k} \quad (1.33)$$

Tyto dvě menší posloupnosti obsahují pouze $\frac{N}{2}$ vzorků. Následně lze členy W_N^{2nk} a $W_N^{(2n+1)k}$ přepsat do tvaru (1.34) a (1.35) [8].

$$W_N^{2nk} = e^{\frac{j2\pi}{N}2nk} = e^{\frac{j2\pi}{\frac{N}{2}}nk} = W_{\frac{N}{2}}^{nk} \quad (1.34)$$

$$W_N^{(2n+1)k} = e^{\frac{j2\pi}{N}(2n+1)k} = e^{\frac{j2\pi}{\frac{N}{2}}nk} e^{\frac{j2\pi}{N}k} = W_{\frac{N}{2}}^{nk} W_N^k \quad (1.35)$$

Dosazením vztahů (1.34) a (1.35) do (1.33) vznikne vztah (1.36). Tento vztah již reprezentuje způsob, jakým lze rozložit výpočet DFT N vzorků na dvě menší DFT o $\frac{N}{2}$ vzorcích, vztahy (1.37) a (1.38) [8].

$$S(k) = \sum_{n=0}^{\frac{N}{2}-1} s(2n) W_N^{nk} + W_N^k \sum_{n=0}^{\frac{N}{2}-1} s(2n+1) W_N^{nk} \quad (1.36)$$

$$S_1(k) = \sum_{n=0}^{\frac{N}{2}-1} s(2n) W_N^{nk} \quad (1.37)$$

$$S_2(k) = \sum_{n=0}^{\frac{N}{2}-1} s(2n+1) W_N^{nk} \quad (1.38)$$

Vztah (1.36) lze nyní psát ve tvaru (1.39). Je třeba podotknout, že posloupnosti $S_1(k)$ a $S_2(k)$ jsou periodické s periodou $\frac{N}{2}$ v důsledku svých komplexních exponenciál a nevadí tedy, že k nabývá hodnot od 0 do $N-1$ [8].

$$S(k) = S_1(k) + W_N^k S_2(k) \quad (1.39)$$

Nicméně vztah (1.39) lze přepsat do dvou rovnic (1.40) a (1.41).

$$S(k) = S_1(k) + W_N^k S_2(k), \quad 0 \leq k \leq \frac{N}{2} - 1 \quad (1.40)$$

$$S(k) = S_1\left(k - \frac{N}{2}\right) - W_N^{k-\frac{N}{2}} S_2\left(k - \frac{N}{2}\right), \quad \frac{N}{2} - 1 \leq k \leq N \quad (1.41)$$

Původem záporného znaménka mezi členy v rovnici (1.41), je člen W_N^k , který v intervalu $\langle \frac{N}{2}; N-1 \rangle$ nabývá hodnot < 0 . Důkazem je například rovnice (1.42), kde $k = \frac{N}{2} + x$ [8].

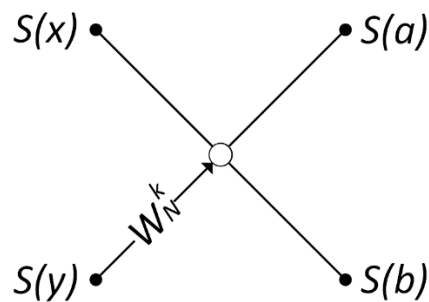
$$W_N^{\left(\frac{N}{2}+x\right)} = e^{\frac{j2\pi}{N}\left(\frac{N}{2}+x\right)} = e^{j\pi} e^{\frac{j2\pi}{N}x} = -W_N^x \quad (1.42)$$

Tento postup lze opakovat až do doby, kdy je počet vzorků v jedné posloupnosti roven 2. DFT dvou vzorků pak lze určit následovně [8].

$$S(0) = s(0) + W_2^0 s(1) = s(0) + s(1) \quad (1.43)$$

$$S(1) = s(0) + W_2^1 s(1) = s(0) - s(1) \quad (1.44)$$

Výpočet DFT pomocí algoritmu DIT tedy probíhá tak, že DFT N vzorků je rozložena na dílčí sudé a liché posloupnosti o dvou vzorcích, jejichž počet je roven $\frac{N}{2}$. Z těchto posloupností je následně určena DFT pomocí vztahů (1.43) a (1.44) o stejném počtu vzorků. Vzniklé posloupnosti dílčích sudých a lichých DFT jsou opakovaně zpracovávány vztahy (1.40) a (1.41), ze kterých vznikne nová posloupnost s dvakrát více členy než dvě předchozí, z nichž tato nová dílčí posloupnost vznikla. Tento postup je prováděn až do doby, dokud nově vzniklá posloupnost neobsahuje právě N členů, protože pak je tato posloupnost DFT určena z posloupnosti vstupních vzorků. Tento postup bude ilustrován a detailně rozebrán v dalším textu [8].



Obrázek 1.6 – Orientovaný graf motýlek

Algoritmus DIT lze rozkreslit do orientovaných grafů. Základním orientovaným grafem tohoto algoritmu je tzv. „motýlek“, obrázek 1.6. Tento orientovaný graf znázorňuje vztahy (1.40) a (1.41) pro jednu hodnotu k , kde vstupem jsou dva vzorky $S(x), S(y)$. Vzorek $S(y)$ je roznásoben členem W_N^k a následně je proveden součet a rozdíl těchto členů. Součet je následně uložen jako vzorek $S(a)$ a rozdíl jako vzorek $S(b)$ [8].

Postup určení DFT podle tohoto algoritmu vyžaduje pouze $N \log_2 N$ komplexních násobení, což je velkou úsporou vůči DFT určené přímo z definičního vztahu, která vyžaduje N^2 komplexních násobení. Některá z těchto násobení jsou však triviální a není třeba je uvažovat, protože je lze nahradit prostým součtem [8].

Algoritmus DIT je účinný a prostý. Výpočet DFT však lze dále zrychlit, protože existuje mnoho dalších algoritmů, které řeší výpočet DFT. Různé algoritmy pak lze kombinovat pro dosažení co nejrychlejšího výpočtu DFT s danou specifikací.

1.4 Číslicové zpracování signálu

Obor číslicového zpracování signálu je velice důležitý, a to obzvlášť v poslední době, kdy dochází ke zvyšování výkonu výpočetní techniky. Signály jsou v dnešní době zpracovávány téměř výhradně číslicově, protože to dovoluje provádět operace, které v analogové oblasti provázet nelze nebo je to velice obtížné. Číslicové obvody jsou velice variabilní oproti obvodům analogovým, protože pokud je třeba úprava nějaké funkce systému, není nutné pozměnit obvodové řešení systému, ale může stačit pouze pozměnit program, který provádí operace nad číslicovým signálem.

1.4.1 Analogový signál

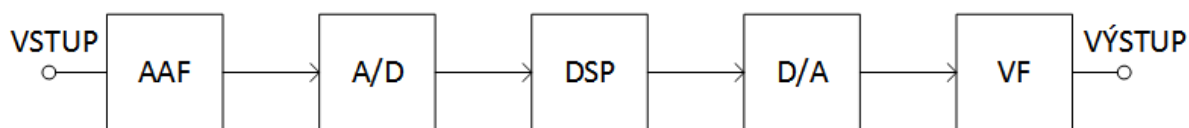
Analogový signál je takový signál, který je spojitý v čase i amplitudě. Takovýto signál má nekonečně mnoho hodnot v amplitudě i v čase, kde k určité hodnotě času existuje i příslušná hodnota amplitudy signálu. Většinu jevů v přírodě je reprezentována právě analogovým signálem [9].

1.4.2 Číslicový signál

Číslicový signál je signál, který již není spojitý v amplitudě ani v čase. Analogový signál není možné zpracovávat pomocí počítače, a je nutné jej převést na signál číslicový. To se děje tak, že se z analogového signálu odeberou vzorky v určitých časech, amplituda signálu se zaokrouhlí a přiřadí se jí binární číslo. Číslicový signál je pak tedy tvořen posloupností binárních čísel, které už mohou nabývat pouze určitých hodnot a existují pouze v určitých časech a neobsahuje veškerou informaci o původním analogovém signálu [9].

1.4.3 Řetězec číslicového zpracování signálu

Převod analogového signálu na signál číslicový je realizován pomocí řetězce číslicového zpracování signálu, jehož blokové schéma ukazuje obrázek 1.7.

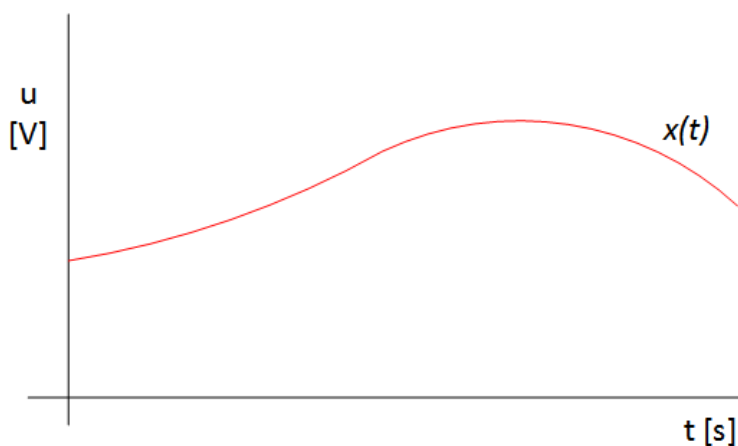


Obrázek 1.7 – Blokové schéma řetězce pro zpracování číslicového signálu.

Vstupní analogový signál je zde přiveden na antialiasingový filtr označen AAF, který je zde proto, aby zamezil aliasingu. Tento filtr je typu dolní propust'. Vyfiltrovaný signál je dále přiveden na A/D převodník. V tomto bloku probíhá odběr vzorků z analogového vyfiltrovaného signálu, jejich následnému zaokrouhlení a přiřazení čísla, například binárního. Blok digitálního zpracování signálu, označený jako DSP (Digital Signal Processing), provádí požadované operace s již digitalizovaným signálem. V případě potřeby je signál zpět převeden na analogový pomocí D/A převodníku a výstupního filtru VF.

1.4.4 A/D převod

Převod analogového signálu na signál digitální v reálných obvodech zabezpečuje A/D převodník, jehož konstrukce mohou být různé a každá má své výhody a nevýhody. Zde bude ukázán teoretický princip převodu analogového signálu na signál digitální, protože je důležitý pro objasnění vzniku aliasingu.



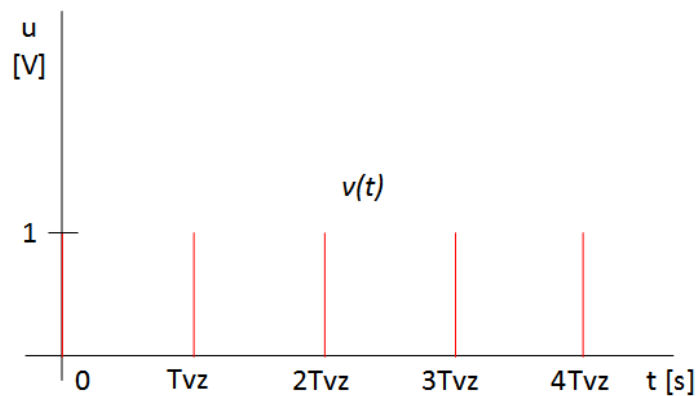
Obrázek 1.8 – Analogový signál.

Obrázek 1.8 ukazuje průběh analogového signálu označeného jako $x(t)$. Tento signál má být převeden na digitální signál, proto je nejdříve nutné odebrat z tohoto signálu vzorky. Vhodným nástrojem k získání vzorků je vzorkovací funkce, rovnice (1.46), která se skládá z časově posunutých jednotkových impulzů a využívá jejich vzorkovací vlastnosti. Její časový průběh

ukazuje obrázek 1.9. Jednotlivé jednotkové impulzy jsou zde od sebe rozmístěny vždy o T_{vz} , což je vzorkovací perioda, již odpovídá také vzorkovací frekvence f_{vz} , rovnice (1.45) [9].

$$f_{vz} = \frac{1}{T_{vz}} \quad (1.45)$$

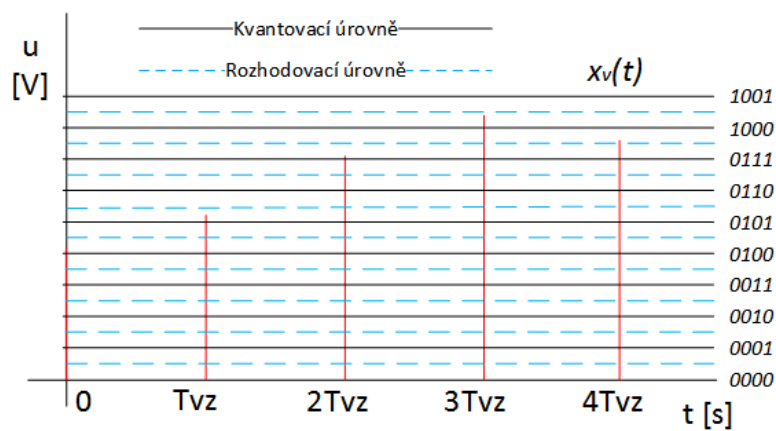
$$v(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT_{vz}) \quad (1.46)$$



Obrázek 1.9 – Časový průběh vzorkovací funkce [9].

K odebrání vzorků ze signálu dojde roznásobením analogového signálu a vzorkovací funkce, rovnice (1.47). Takto vzniklý signál $x_v(t)$ ukazuje obrázek 1.10 [9].

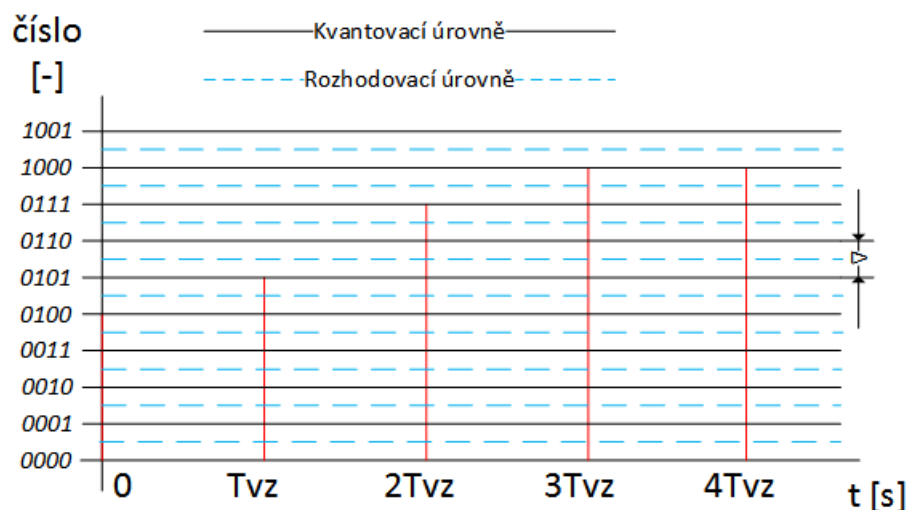
$$x_v(t) = x(t)v(t) = \sum_{n=-\infty}^{\infty} x(nT_{vz}) \delta(t - nT_{vz}) \quad (1.47)$$



Obrázek 1.10 – Signál spojitý v amplitudě a nespojitý v čase.

Signál podle obrázku 1.10 se již objevuje pouze v určitých časech daných vzorkovací periodou, je tedy diskrétní v čase. Jeho napětí však stále nabývá nekonečně mnoha hodnot, což je pro číslicový systém nepřijatelné. Z tohoto důvodu je nutné určit kvantizační úrovně, což jsou dané hodnoty amplitudy analogového signálu, jímž je přiřazeno určité číslo. Okamžité napětí analogového signálu se však může nacházet mezi těmito úrovněmi a je nutné rozhodnout, ke které kvantizační úrovni bude vzorek zaokrouhlen. Toto rozhodování se provádí na základě velikosti okamžitého napětí analogového signálu vůči rozhodovacím úrovním. Výsledný číslicový signál ukazuje obrázek 1.11. Celý proces přiřazení čísla okamžitému napětí analogového signálu je nazýván jako kvantifikace.

Počet kvantizačních úrovní je dán počtem bitů A/D převodníku. Pokud je počet bitů A/D převodníku označen jako n , pak celkový počet kvantizačních úrovní je 2^n . Kvantizační krok, často značen jako Δ , je rozdíl mezi dvěma sousedními rozhodovacími úrovněmi. Takto je také vyznačen na obrázku 1.11. Čím vyšší bude počet bitů převodníku, tím více úrovní okamžitého napětí lze rozeznat a chyba vzniklá převodem bude menší, protože je třeba méně zaokrouhlovat. Samotné pozice kvantizačních a rozhodovacích úrovní jsou pak dány společně referenčním napětím A/D převodníku a počtem kvantizačních úrovní.



Obrázek 1.11 – Číslicový signál.

Reálné A/D převodníky nepracují na principu pronásobování vstupního signálu vzorkovací funkcí, to je pouze cesta, kterou lze tento proces teoreticky popsat. Ve skutečném převodníku je pak obvod, který odebírá vzorky ze vstupního analogového signálu a tyto vzorky si pamatuje po celou dobu převodu například na nějaké kapacitě. Za tento obvod je následně zařazen vlastní

převodník napětí na číslo, který zajistí samotnou kvantizaci a jehož výstupem je právě binární číslo odpovídající danému okamžitému napětí na vstupu vzorkovacího obvodu.

1.4.5 Aliasing

Aliasing je jev, který vzniká při nedodržení tzv. Shannonova vzorkovacího teorému, který značí rovnice (1.48). Tento teorém říká, že pokud se ve vzorkovaném analogovém signálu vyskytuje nejvyšší frekvence f_{max} , pak je nutné tento signál vzorkovat vzorkovací frekvencí nejméně dvakrát větší [8, 9].

$$f_{vz} > 2f_{max} \quad (1.48)$$

Pakliže není tento teorém dodržen, dochází ke ztrátě informace o analogovém signálu přímo ve fázi vzorkování a takovýto signál nelze následně správně rekonstruovat. Následně bude ukázána příčina [9].

V kapitole 1.4.4 byl ukázán princip vzorkování pomocí vzorkovací funkce. Vzorkovací funkce využívá vzorkovací vlastnosti jednotkového impulzu. Průběh vzorkovací funkce ukazuje obrázek 1.9. k objasnění vzniku aliasingu je nutné určit spektrum této funkce [9].

Spektrum periodického signálu nelze určit užitím Fourierovy transformace přímo. Pakliže je spektrum $S(\omega)$ dáno jednotkovými impulzy na kmitočtech $k\omega_0$ s amplitudou $2\pi A_k$, rovnice (1.49), lze k němu použitím inverzní Fourierovy transformace určit odpovídající časový průběh $s(t)$ podle vztahu (1.50) [8, 9].

$$S(\omega) = \sum_{k=-\infty}^{\infty} 2\pi A_k \delta(\omega - k\omega_0) \quad (1.49)$$

$$s(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \sum_{k=-\infty}^{\infty} 2\pi A_k \delta(\omega - k\omega_0) e^{j\omega t} \quad (1.50)$$

Záměnou integrace a sumace společně s vyčíslením integrálu, užitím vzorkovací vlastnosti jednotkového impulzu vznikne vztah (1.51) [8, 9].

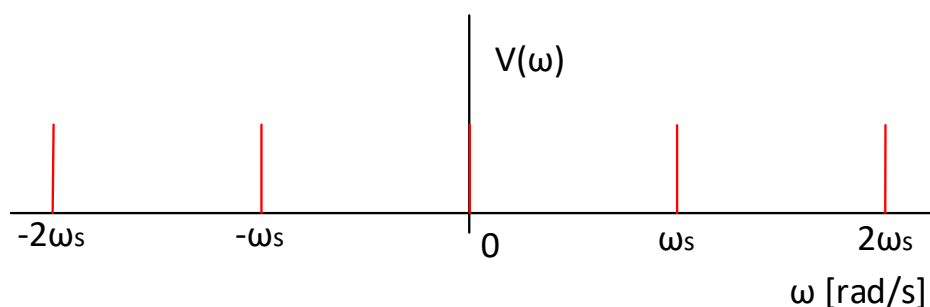
$$s(t) = \sum_{k=-\infty}^{\infty} A_k e^{jk\omega_0 t} \quad (1.51)$$

Vztah (1.51) je definičním pro komplexní Fourierovu řadu, kde A_k je k -tá komplexní amplituda určená příslušným integrálem. Spektrum periodického signálu lze tedy reprezentovat Fourierovou transformací, kde Fourierovým obrazem je sled jednotkových impulzů na frekvencích $k\omega_0$ a amplitudou $2\pi A_k$, kde A_k je k -tá komplexní amplituda určená příslušným integrálem z oblasti Fourierovy řady. Podobným způsobem lze určit Fourierův obraz vzorkovací funkce určené vztahem (1.46). Nejdříve je nutné určit A_k , rovnice (1.52) [8, 9].

$$A_k = \frac{1}{T_{vz}} \int_{-\infty}^{\infty} \delta(t) e^{-jk\omega_{vz}t} dt = \frac{1}{T} \quad (1.52)$$

Dosažením do vztahu (1.49) lze získat Fourierův obraz vzorkovací funkce, jenž ukazuje rovnice (1.53). Grafickým znázorněním tohoto vztahu je spektrum na obrázku 1.12. Toto spektrum je nekonečné [8, 9].

$$S_v(\omega) = \frac{2\pi}{T_{vz}} \sum_{k=-\infty}^{\infty} \delta\left(\omega - k \frac{2\pi}{T_{vz}}\right) \quad (1.53)$$

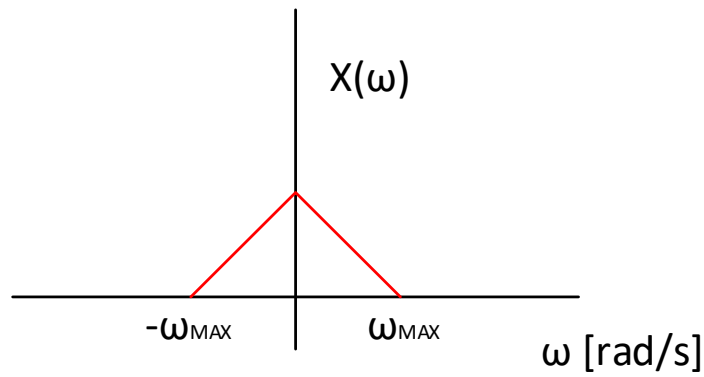


Obrázek 1.12 – Spektrum vzorkovací funkce [9].

Spektrum ovzorkovaného signálu popsaného rovnicí (1.47), lze následně určit pomocí věty o součinu signálů, rovnice (1.54), která říká, že Fourierův obraz součinu dvou signálů lze určit pomocí konvoluce jejich spekter dělené konstantou 2π [8, 9].

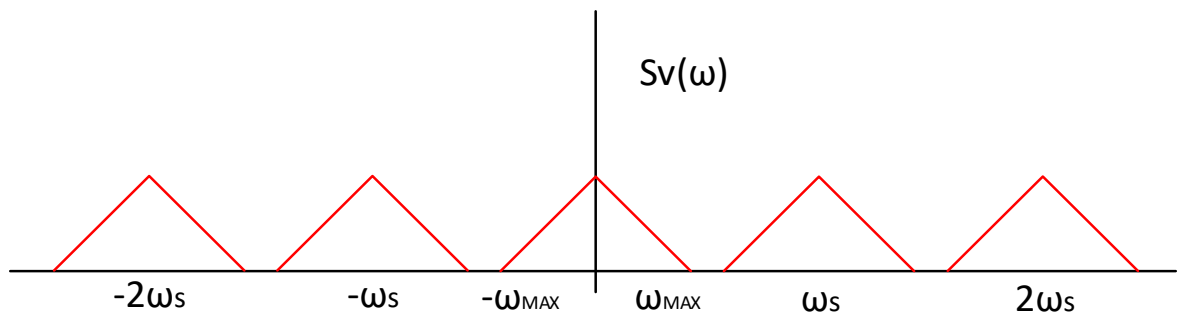
$$F[x(t) v(t)] = \frac{1}{2\pi} [X(\omega) * V(\omega)] \quad (1.54)$$

Na obrázku 1.13, je ukázáno smyšlené spektrum vzorkovaného signálu, označeného jako $X(\omega)$.



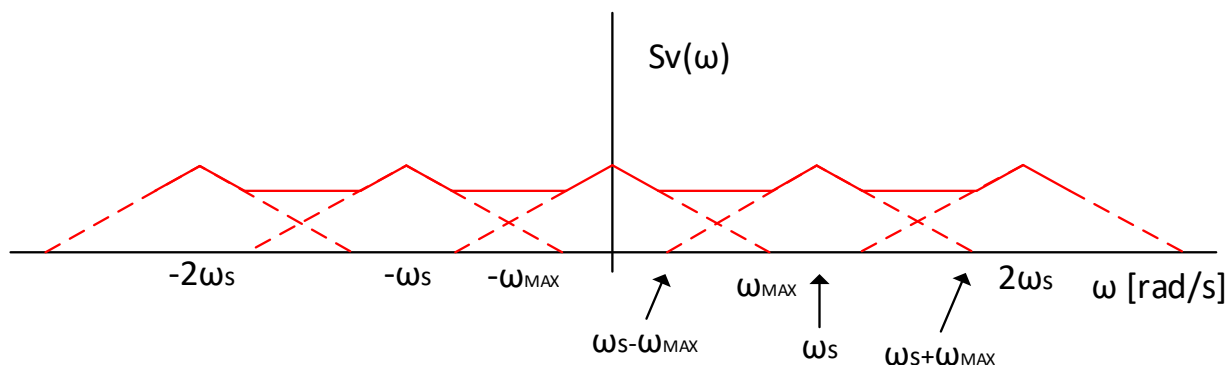
Obrázek 1.13 – Smyšlené spektrum vzorkovaného signálu [9].

Spektrum ovzorkovaného signálu určeného pomocí rovnice (1.54) ukazuje obrázek 1.14. Zde je vidět, že spektrum ovzorkovaného signálu obsahuje kopie základního spektra $X(\omega)$ vzorkovaného signálu na celistvých násobcích vzorkovací frekvence a že toto spektrum je nekonečné, což má příčinu v samotném principu vzorkování vzorkovací funkcí. V tomto konkrétním případě se také neprojevuje aliasing, protože byl dodržen Shannonův vzorkovací teorém [9].



Obrázek 1.14 – Spektrum ovzorkovaného signálu bez aliasingu [9].

Následující obrázek 1.15 však ukazuje případ, kdy Shannonův vzorkovací teorém dodržen nebyl a dochází zde k překrývání kopií spekter, což zapříčiňuje již zmíněnou ztrátu informace už při vzorkování vstupního analogového signálu. Aliasing je tedy důsledkem překrývání kopií základního spektra analogového signálu a tomuto jevu nelze zamezit jinak než dodržemím právě zmíněného Shannonova teorému, který vychází z předchozích poznatků [9].



Obrázek 1.15 – Spektrum ovzorkovaného signálu s aliasingem [9].

1.4.6 Antialiasingový filtr

Podle Shannonova vzorkovacího teorému je nutné signál vzorkovat nejméně dvakrát vyšší frekvencí, než je maximální frekvence obsažená v signálu. Pakliže není toto splněno vzniká aliasing a informace o analogovém signálu jsou degenerované. V signálech se však často vyskytují frekvenční složky, které již nenesou informace nutné ke správnému zpracování signálu a lze je zanedbat. Pokud by byly tyto složky ponechány, zbytečně by docházelo k navyšování vzorkovací frekvence. Pokud by byla vzorkovací frekvence ponechána na nižší hodnotě než kterou určuje Shannonův teorém, docházelo by k aliasingu. Antialiasingový filtr má za úkol potlačit nepotřebné vyšší frekvenční složky a tím zamezit aliasingu, aby mohla být zvolena nižší vzorkovací frekvence vůči frekvenčním složkám, které nesou důležité informace.

Antialiasingový filtr bývá typu dolní propust' a jeho strmost a tvar frekvenční amplitudové charakteristiky je nutné určit podle druhu signálu a účelu jeho zpracování. Vhodné jsou aktivní filtry, které dovolují signál zesílit. Mezní frekvence filtru by měla být zvolena tak, aby filtr propustil požadované frekvence a nechtěné frekvence potlačil dostatečně rychle vůči vzorkovací frekvenci. Vzorkovací frekvence by měla být zvolena tak, aby splnila Shannonův teorém i pro frekvence, které nejsou ještě spolehlivě potlačeny.

1.4.7 Frakční čísla

Při zpracování signálu je většinou nutné počítat s desetinnými čísly. Tato čísla bývají obvykle reprezentována ve formátu s pohyblivou řádovou čárkou. Většina mikroprocesorů však neobsahuje jednotku, která s těmito čísly umí pracovat. Pokud je požadován výpočet v tomto formátu, je nutné operace s těmito čísly, hlavně násobení a dělení, provádět složitě v rámci možností mikroprocesoru. Řešení tohoto problému přinášejí alespoň částečně frakční čísla, anglicky „fractional numbers“, označované písmenem Q [12].

Každé desetinné číslo lze zapsat jako zlomek, přičemž ten se skládá ze jmenovatele a čitatele. Pokud nezapisujeme toto číslo do zlomku, ale předpokládá se, že je jmenovatel pro všechny čísla stejný a známý, pak je číselník zapsán jako celé číslo. Jednotku pro výpočet násobení a dělení v celých číslech už mívají mikroprocesory implementovanou a dochází tak ke zrychlení výpočtů na úkor přesnosti výpočtu. Je však třeba zavést korekce výsledku plynoucí z násobení a dělení zlomkových čísel. Rozlišení formátu Q je totiž pevně dané počtem bitů za desetinnou čárkou [12].

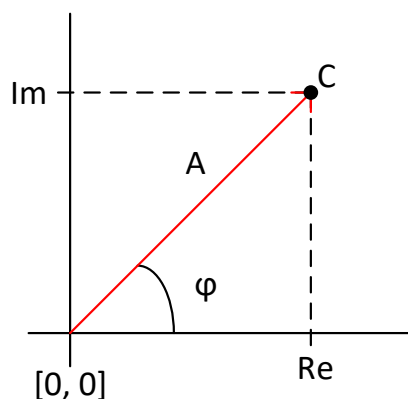
Frakční čísla jsou označována jako $Qm.n$, kde m určuje počet čísel před desetinnou čárkou a n pak počet čísel za desetinnou čárkou. Pro zápis čísla v tomto formátu je nutné vyhradit místo o $m + n$ bitech. Pokud bude tento formát uložen do datového typu signed integer, pak lze určit rozsah čísel, které lze do tohoto formátu uložit podle rovnice (1.55). Rozlišení r tohoto formátu pak lze určit pomocí rovnice (1.56) [12].

$$\langle -2^{m-1}, 2^{m-1} - 2^{-n} \rangle \quad (1.55)$$

$$r = 2^{-n} \quad (1.56)$$

1.4.8 IQ data

Komplexní číslo lze reprezentovat ve více tvarech. Obrázek 1.16 ukazuje komplexní číslo C zakreslené v komplexní rovině pomocí modulu A a fáze φ .



Obrázek 1.16 – Komplexní číslo zakreslené v komplexní rovině.

Reálnou složku Re lze určit z amplitudy a fáze pomocí rovnice (1.57). Složku imaginární Im pak pomocí rovnice (1.58) [2].

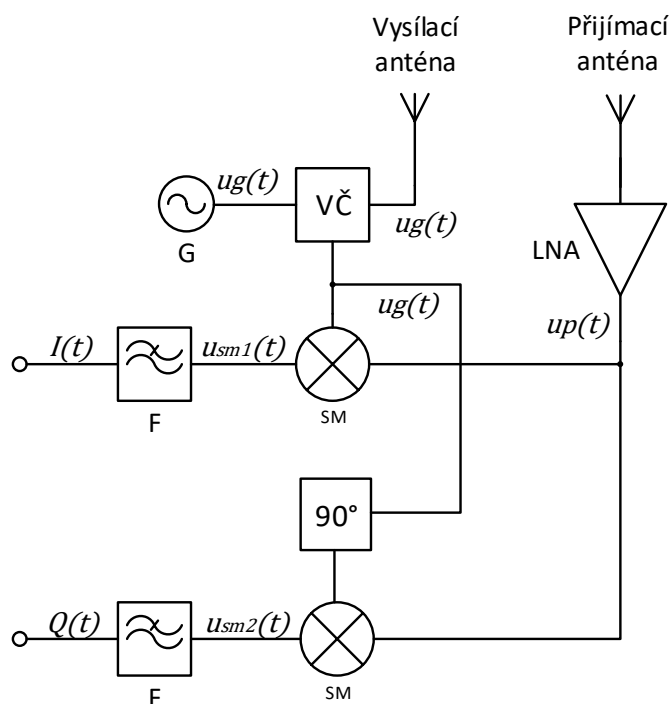
$$Re = A \cos(\varphi) \quad (1.57)$$

$$I_m = A \sin(\varphi) \quad (1.58)$$

V kapitole 1.1.3 o CW radarech bylo ukázáno základní schéma homodynního přijímače, který zpracovává pouze reálnou složku signálu označovanou jako I. Tato složce se v anglickém jazyce nazývá „in-phase“, protože je jednou z výstupních složek směšovače, který směšuje přijatý signál a vysílaný signál, který není fázově posunut. Při takovémto zpracování byla zanedbána imaginární složka signálu. Pokud ji ovšem chceme získat, je nutné upravit původní schéma homodynního přijímače podle obrázku 1.17. Zde byla přidána jedna směšovací větev, pomocí které se získává imaginární složka signálu, označována jako Q, anglicky „quadrature-phase“. Vysílaný signál je fázově posunut o $+90^\circ$ a následně je směšován spolu s přijatým signálem. Podobně jako v kapitole 1.1.3 o CW radarech lze odvodit vztah (1.59) [2].

$$Q(t) = \frac{U_{GMAX} U_{PMAX}}{2} \cos[2\pi(f_G - f_P)t - 90^\circ] = \quad (1.59)$$

V tomto vztahu figuruje fázový posuv -90° , který z funkce cosinus vytvoří funkci sinus. Výsledkem je potom vztah (1.60), který určuje komplexní složku signálu $Q(t)$.



Obrázek 1.17 – Modifikovaný homodynní přijímač pro složky I a Q [2].

$$Q(t) = \frac{U_{GMAX} U_{PMAX}}{2} \sin[2\pi(f_G - f_P)t] \quad (1.60)$$

Reálnou složku signálu pak určuje vztah (1.61).

$$I(t) = \frac{U_{GMAX}U_{PMAX}}{2} \cos[2\pi(f_G - f_P)t] \quad (1.61)$$

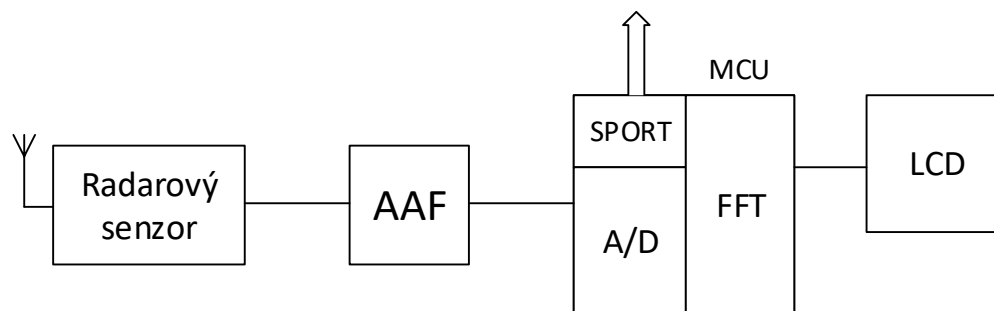
Příkladem použití je právě CW radar, který měří radiální rychlost. Pokud známe IQ složky přijímaného signálu, pak lze určit nejen rychlost objektu ale dokonce i zda se objekt přibližuje nebo vzdaluje, protože s pomocí IQ dat lze rozpoznat kladnou a zápornou frekvenci. Další využití nachází IQ reprezentace dat v moderních více stavových modulacích, kde lze pomocí různě velkých složek I a Q měnit amplitudu a fázi výsledného součtového signálu [2].

2 Návrh řetězce zpracování IF signálu

Tato část práce se zabývá návrhem řetězce zpracování mezifrekvenčního signálu z Dopplerovského CW radaru, realizací tohoto návrhu a ověření dosažených parametrů zpracování signálu.

2.1 Blokové schéma systému pro zpracování IF signálu

Obrázek 2.1 ukazuje blokové schéma celého řetězce zpracování IF signálu. Radarový senzor je připojen na aktivní antialiasingový filtr AAF. Jeho výstup je připojen na A/D převodník, který je integrován v mikroprocesoru MCU. Tento mikroprocesor má také integrováno sériové rozhraní SPORT pro přenos dat do PC. k mikroprocesoru je připojen LCD displej k usnadnění odladění programové části systému zpracování mezifrekvenčního signálu.



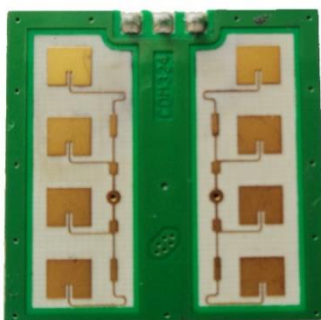
Obrázek 2.1 – Zamýšlené blokové schéma řetězce zpracování IF signálu.

2.2 Radarové senzory využité k měření rychlosti

Radarový senzor umožňující Dopplerovské měření rychlosti, zajišťující vysílání a příjem elektromagnetického signálu spolu s jeho základním zpracováním musel být vybrán podle zadaných parametrů. Těmito parametry byly vysílací výkon 15 dBm a frekvenční pásmo 24 GHz, které bývá označováno jako pásmo „K“. Senzorů splňujících obě specifikace je na trhu dostatečné množství. Jsou k dispozici levné senzory s cenou do stovek korun a senzory, jejichž cena je podstatně vyšší, odpovídá však kvalitě zpracování daného senzoru. Kvalita zpracování senzoru je klíčová pro dosažení dobrých parametrů celého řetězce měření rychlosti.

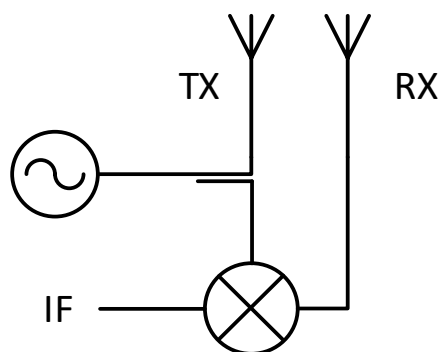
2.2.1 CDM324

Jedním z cenově velice dostupných senzorů je i senzor CDM324. Tento senzor je zobrazen na obrázku 2.2.



Obrázek 2.2 – Radarový senzor CDM324.

Nalézt informace o vnitřní konfiguraci tohoto senzoru, není příliš jednoduché. Lze však zjistit, že se tento senzor velice podobá senzorům IPM165 a AP96. Jejich parametry jsou téměř totožné. Vnitřní schéma tohoto senzoru pak ukazuje obrázek 2.3.



Obrázek 2.3 – Blokové schéma senzoru CDM324 [13].

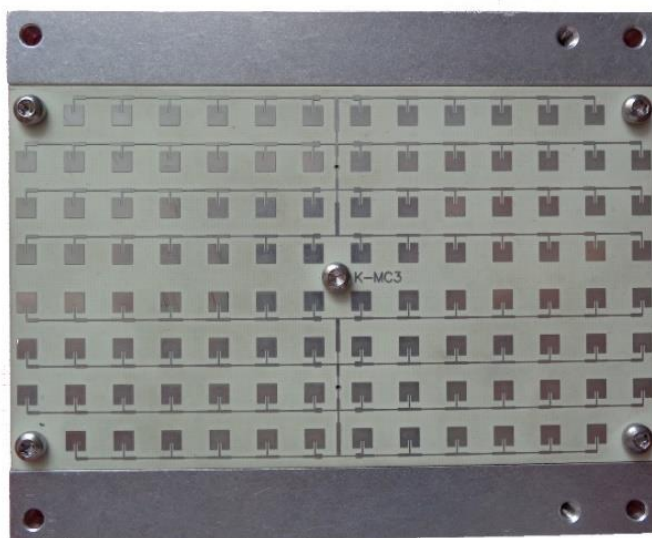
Blokové schéma podle obrázku 2.3 prakticky odpovídá ochuzenému homodynnímu přijímači. Skládá se z oddělené vysílací a přijímací antény, které jsou integrovány na DPS senzoru. Vysílací anténa je připojena na oscilátor, který generuje vysílaný signál s výstupním ekvivalentně izotropicky vyzářeným výkonem typicky 16 dBm. Přijímaný signál zde není zesílen zesilovačem LNA, ale je připojen přímo do směšovače spolu se zeslabeným vysílaným signálem. Podle dostupných informací zde není přítomen výstupní zesilovač. Výstupem z tohoto senzoru je pouze složka I. Šířka svazku antény tohoto senzoru je v azimutu 80° a v elevaci pak 35°. Zisk přijímací antény je udáván 9.5 dBi. Všechny parametry důležité pro další návrh jsou pro přehlednost uvedeny v následující tabulce 1 [13].

Tabulka 1 – Vybrané parametry senzoru CDM324 [13].

Název parametru	Označení	Hodnota
Napájecí napětí	U_{cc}	5 V
Vysílací frekvence	f_T	Typ 24,150 GHz
EIRP	$EIRP$	16 dBm
Šířka svazku v elevaci	φ	35°
Šířka svazku v azimutu	ϑ	80°
Zisk antény	G	9,5 dBi
Zisk LNA	A_{u-LNA}	1
Zisk výstupního zesilovače	A_{u-VZ}	1

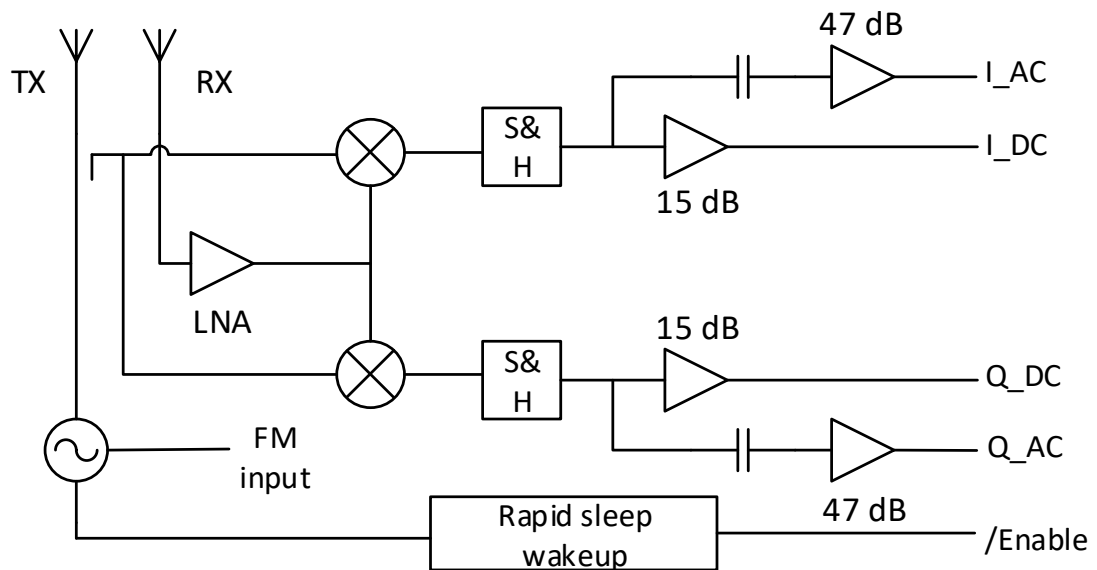
2.2.2 K_MC3

Senzor K_MC3 je radarový senzor od švýcarské firmy RFbeam microwave. Jeho fotografie je na obrázku 2.4.



Obrázek 2.4 – Radarový senzor K_MC3

Senzor K_MC3 využívá oddělené přijímací a vysílací antény a topologii homodynního přijímače. Celkové blokové schéma tohoto senzoru ukazuje obrázek 2.5.



Obrázek 2.5 – Blokové schéma senzoru K_MC3 [14].

Oscilátor generující vysílaný signál je připojený na vysílací anténu. Tento oscilátor lze frekvenčně rozmítat, a proto lze tento senzor využít k FMCW aplikacím. Signál z přijímací antény je zesílen pomocí zesilovače LNA se ziskem 16 dB a následně přiveden na dvojici směšovačů. První směšovač směšuje přijatý signál s vysílaným zeslabeným signálem a na jeho výstupu je tedy signál I. Druhý směšovač směšuje přijatý signál s vysílaným zeslabeným signálem fázově posunutým o 90° a na jeho výstupu je tedy složka Q. Oba výstupní signály směšovačů jsou vyvedeny na dvojici zesilovačů. Jeden pár zesilovačů je stejnosměrně vázaný se ziskem 15 dB a ořezovou frekvencí 500 kHz. Druhý pár je vázaný střídavě a jeho zisk je 47 dB, mezní frekvence jsou 40 Hz a 15 kHz. Ekvivalentní izotropicky vyzářený výkon senzoru K_MC3 je typicky 19 dBm, frekvence vysílaného signálu je pak typicky 24,150 GHz. Šířka svazku antény tohoto senzoru v azimutu je 7° a v elevaci pak 25° . Zisk antény je udáván jako 21 dBi. Všechny důležité parametry pro návrh jsou pro přehlednost uvedeny v následující tabulce 2 [14].

Tabulka 2 – Vybrané parametry senzoru K_MC3 [14].

Název parametru	Označení	Hodnota
Napájecí napětí	U_{cc}	5 V
Vysílací frekvence	f_T	Typ 24,150 GHz
EIRP	$EIRP$	19 dBm
Šířka svazku v elevaci	φ	25°
Šířka svazku v azimutu	ϑ	7°
Zisk antény	G	21 dBi
Zisk LNA	A_{u-LNA}	16 dB
Zisk výstupního zesilovače	A_{u-VZ}	x_DC 15 dB x_AC 47 dB

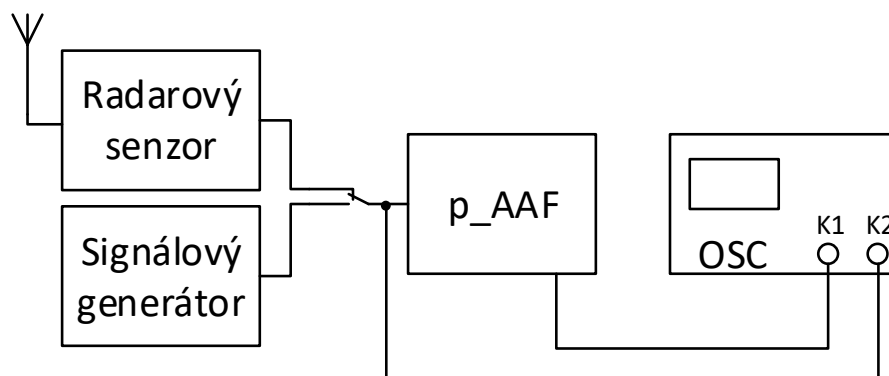
2.3 Návrh antialiasingových filtrů

Tato kapitola se zabývá návrhem antialiasingových filtrů. k jejich návrhu je nejdříve nutné určit všechny potřebné parametry ze znalosti zpracovávaného signálu.

2.3.1 Určení zisku antialiasingového filtru

Ze znalosti použitých radarových senzorů lze očekávat, že IF signál na výstupu obou senzorů nebude mít amplitudu signálu dostatečnou pro správné určení rychlosti objektu. Je tedy nutné signál zesílit na požadovanou úroveň. Při použití aktivního antialiasingového filtru, (dále jen AAF), lze zajistit zesílení a filtraci signálu současně. Zisk zesilovače by měl být určen tak, aby maximální amplituda IF signálu vyvolala na výstupu AAF maximální amplitudu signálu, kterou dokáže A/D převodník zpracovat. Pokud by byl tento zisk moc veliký, docházelo by ke zkreslení signálu a určení rychlosti by nebylo přesné.

Ze znalosti amplitudy přijímaného signálu radarovým senzorem na výstupu přijímací antény a celkového napěťového přenosu senzoru lze určit amplitudu IF signálu. Celkový napěťový přenos senzoru lze snadno určit. Stačí sečíst zisky zesilovačů v přijímací části a připočíst ztrátu směřováním. k určení amplitudy přijímaného signálu se tedy nabízí radarová rovnice, protože většina potřebných parametrů je známa. Senzor má být umístěn na pohyblivém objektu a určovat jeho rychlost. Povrch, na který bude namířen je asfaltový, případně kamenný a jeho efektivní odrazivá plocha však není známa. Její určení by bylo pracné a odhad by mohl být nepřesný.



Obrázek 2.6 – Schéma zapojení pro zjištění zisku AAF.

Zisk AAF byl proto určen experimentálně. Obrázek 2.6 ukazuje principiální schéma zapojení obvodu pro zjištění potřebného zisku AAF. Radarový senzor byl připevněn na automobil a byl umístěn tak, jako by byl v reálné aplikaci. To znamená ve stejné výšce a nakloněn k povrchu pod požadovaným úhlem. Na výstup radarového senzoru byl připojen zkonstruovaný provizorní AAF označen jako p_AAF, jehož zisk bylo možné měnit. Na výstup p_AAF byl připojen osciloskop, na jehož obrazovce byla pozorována amplituda výstupního signálu. Následně se automobil začal pohybovat a na výstupu senzoru byl pak pozorován IF signál. Zisk AAF byl zvyšován až do doby, dokud nebylo dosaženo požadované úrovně výstupního IF signálu. Při jeho dosažení byl zisk zesilovače zafixován a na jeho vstup byl následně připojen generátor sinusového signálu se známou amplitudou. Na výstupu zafixovaného AAF byl stále připojen osciloskop, z jehož obrazovky se odečetla amplituda výstupního signálu a zisk zesilovače byl pak určen pomocí rovnice (2.1), kde A_u je napěťový zisk zesilovače, U_2 je amplituda výstupního signálu aktivního AAF a U_1 je amplituda vstupního signálu z generátoru.

$$A_u = \frac{U_2}{U_1} \quad (2.1)$$

Pro senzor CDM byl potřebný zisk AAF určen jako $A_{u_{CDM}} = 53 \text{ dB}$. Senzor K_MC3 má dva páry výstupů. Jeden z nich je vázán stejnosměrně s malým ziskem výstupního zesilovače a druhý střídavě s velkým ziskem výstupního zesilovače. Pokud by byl přijímaný signál příliš velký, na výstupech x_AC by docházelo ke zkreslení signálu, které by nešlo ovlivnit, což by znemožnilo určení rychlosti objektu. Proto byly jako hlavní používané výstupy vybrány x_DC, které mají v cestě zesilovače s nižším ziskem. Pro výstupy x_DC byl tedy určen potřebný napěťový zisk $A_{u_{MC3-DC}} = 25 \text{ dB}$. Oba zisky však nejsou maximální možné. k dosažení plné amplitudy výstupního signálu filtrů byla ponechána rezerva vůči jiným druhům povrchu, než na kterých byl senzor zkoušen a také vůči objektům odrážející elektromagnetické vlnění

ve větší míře. Protože jsou zisky pro oba senzory signifikantně rozdílné bude pro každý senzor navržen zvláštní filtr.

2.3.2 Určení mezních frekvencí antialiasingového filtru

Pro správný návrh AAF je třeba znát maximální a minimální frekvence obsažené v signálu na výstupu radarového senzoru. Ze zadaných radiálních rychlostí (1 m/s – 50 m/s), které mají být zpracovávány, lze zjistit odpovídající frekvence IF signálu na výstupu radarového senzoru. Přepisem rovnice (1.21) a úpravou lze získat rovnici (2.2) pro Dopplerovu frekvenci f_D .

$$f_P = f_V \left(1 + \frac{2v_R}{c}\right) \Rightarrow f_D = \frac{2v_R f_V}{c} \quad (2.2)$$

Dosažením maximální radiální rychlosti 50 m/s lze zjistit maximální Dopplerovu frekvenci $f_{D_{MAX}}$ IF signálu, rovnice (2.3).

$$f_{D_{MAX}} = \frac{2v_{R_{MAX}} f_V}{c} = \frac{2 \times 50 \times 24,125 \times 10^9}{3 \times 10^8} = 8,0417 \text{ kHz} \quad (2.3)$$

A dosažením minimální radiální rychlosti 1 m/s lze zjistit $f_{D_{MIN}}$, tedy minimální Dopplerovu frekvenci IF signálu, rovnice (2.4).

$$f_{D_{MIN}} = \frac{2v_{R_{MIN}} f_V}{c} = \frac{2 \times 1 \times 24,125 \times 10^9}{3 \times 10^8} = 160,83 \text{ Hz} \quad (2.4)$$

Parametr c je rychlost světla a f_V je frekvence vysílaného signálu.

AAF budou vázány na radarový senzor střídavě, protože při směšování, které bylo popsáno v kapitole 1.2.1 může vznikat napěťová stejnosměrná složka, jejíž velikost by mohla s vysokým ziskem AAF znemožnit určení rychlosti objektu. Dolní mezní frekvence bude tedy volena $f_{c_{min}} = 10 \text{ Hz}$ a horní mezní frekvence pak s rezervou $f_{c_{max}} = 8,5 \text{ kHz}$. Horní mezní frekvence by neměla být moc vysoká, protože pak by bylo nutné volit zbytečně vysokou vzorkovací frekvenci, což by mohlo učinit problémy s rozlišením Fourierovy transformace. Je však ještě nutné podotknout, že horní a dolní mezní frekvence AAF jsou stejné pro oba radarové senzory.

2.3.3 Určení ostatních parametrů antialiasingového filtru

V předchozích dvou kapitolách 2.3.1 a 2.3.2 byly určeny dva důležité parametry, podle kterých by měl být navrhnout AAF. k tomu, aby bylo možné filtr navrhnout, je třeba zvolit další parametry.

Jedním z těchto parametrů je druh a topologie filtru. Je zřejmé, že s ohledem na velikost amplitudy IF signálu a potřebu zisku je třeba volit filtr aktivní, který dovolí signál zesílit a zároveň filtrovat. Aktivní filtry s operačními zesilovači jsou vhodné pro tento účel. V literatuře lze nalézt velké množství topologií filtrů s operačními zesilovači. Každá z topologií má své výhody a nevýhody a různě náročnou konstrukci. Ze všech možných topologií aktivních filtrů byla vybrána topologie Sallen-Key, protože jde o jednu z konstrukčně méně náročných topologií [16, 17].

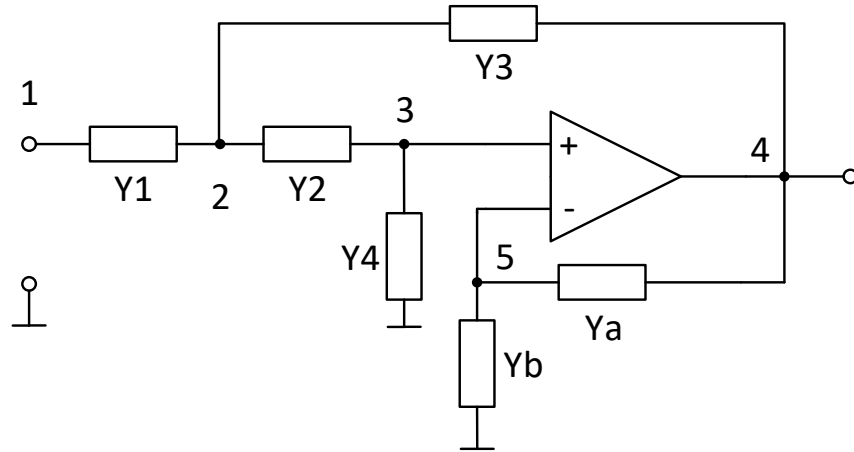
Dále bylo třeba zvolit aproximaci frekvenční charakteristiky aktivního filtru. Každá z aproximací má své vlastnosti, které poskytují určité výhody na úkor ostatních parametrů. Pro tuto aplikaci však byla zvolena Butterworthova aproximace, která se vyznačuje maximálně plochou amplitudovou frekvenční charakteristikou v propustném pásmu. Přechod z propustného do nepropustného pásma je zde pomalejší ve srovnání s Čebyševovou aproximací. Odezva na jednotkový skok vykazuje záškuby [16, 17, 18].

Výběr parametrů aktivního filtru byl završen volbou jeho řádu. Řád filtru je parametr, který určuje strmost klesání frekvenční charakteristiky v nepropustném pásmu. Filtr prvního řádu má strmost 20 dB/dekádu. Filtr n-tého řádu má pak strmost n-krát větší než filtr řádu prvního. Řád filtru pro zpracování IF signálu byl zvolen minimálně dva, což zajistí dostatečně rychlý pokles charakteristiky v nepropustném pásmu, aby nemusela být volena zbytečně velká vzorkovací frekvence A/D převodníku s ohledem na frekvence, které ještě nejsou dostatečně potlačeny. Hrozil by pak aliasing nebo problémy s rozlišením Fourierovy transformace.

2.3.4 Rovnice popisující aktivní dolní propust' typu Sallen-Key

K určení hodnot součástek potřebných k dosažení požadovaných parametrů je třeba určit rovnice, kterými se tyto hodnoty určí. Jedním ze způsobů, jak rovnice pro výpočet hodnot součástek odvodit, ukazuje následující postup. Zde je nutné podotknout, že algoritmus použitý k získání rovnic popisujících dolní propust' v topologii Sallen-Key je podrobně popsán například v [7].

Nejprve je třeba určit přenosovou funkci topologie SK pro obecné admitance nebo impedance. Schéma zapojení topologie SK s ideálním operačním zesilovačem a obecnými admitancemi ukazuje obrázek 2.7. Obvod podle obrázku 2.7 obsahuje také 5 uzlů označených příslušnými čísly. k určení přenosu je třeba znát napětí v jednotlivých uzlech obvodu. Jednotlivá napětí budou určena v následujících odstavcích pomocí algoritmické metody uzlových napětí spolu s metodou redukce počtu proměnných ideálním operačním zesilovačem [7, 16, 17, 18].



Obrázek 2.7 – Aktivní filtr v topologii Sallen-Key s obecnými admitancemi [16, 17, 18].

Nejdříve je tedy nutné sestavit zkrácenou admitanční matici \mathbf{Y} , kterou ukazují rovnice (2.5).

$$\mathbf{Y} = \begin{bmatrix} Y_1 & -Y_1 & 0 & 0 & 0 \\ -Y_1 & Y_1 + Y_2 + Y_3 & -Y_2 & -Y_3 & 0 \\ 0 & -Y_2 & Y_2 + Y_4 & 0 & 0 \\ 0 & -Y_3 & 0 & Y_3 + Y_A & -Y_A \\ 0 & 0 & 0 & -Y_A & Y_B + Y_A \end{bmatrix} \quad (2.5)$$

V důsledku použití ideálního operačního zesilovače lze matici podle rovnice (2.5) upravit následujícím způsobem. Sloupec 5 se přičte ke 3. sloupci, kde se pak sloupec číslo 5 vynechá, vynechá se také řádek číslo 4. Takto vznikne redukovaná matice \mathbf{Y}_R , rovnice (2.6).

$$\mathbf{Y}_R = \begin{bmatrix} Y_1 & -Y_1 & 0 & 0 \\ -Y_1 & Y_1 + Y_2 + Y_3 & -Y_2 & -Y_3 \\ 0 & -Y_2 & Y_2 + Y_4 & 0 \\ 0 & 0 & Y_B + Y_A & -Y_A \end{bmatrix} \quad (2.6)$$

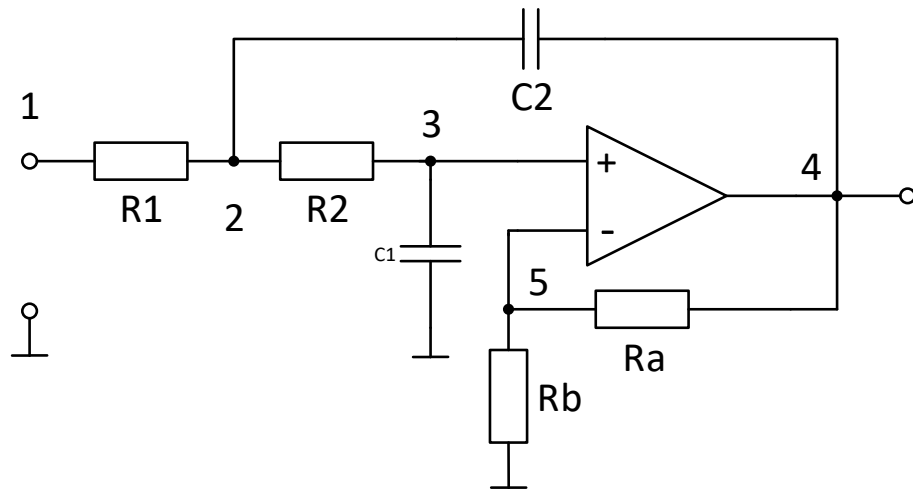
Z matice \mathbf{Y}_R lze určit přenos podle rovnice (2.7), kde $\Delta_{1:4}$ značí determinant matice \mathbf{Y}_R s vynechaným prvním řádkem a čtvrtým sloupcem. Druhý člen $\Delta_{1:1}$ má obdobný význam.

$$A_u = \frac{\Delta_{1:4} \cdot (-1)^{1+4}}{\Delta_{1:1} \cdot (-1)^{1+1}} = -\frac{\Delta_{1:4}}{\Delta_{1:1}} \quad (2.7)$$

Určením potřebných determinantů a úpravami, které jsou zdlouhavé lze určit přenos topologie SK, rovnice (2.8).

$$A_u = \frac{\frac{Y_1 Y_2 (Y_B + Y_A)}{Y_A}}{Y_1 Y_2 + Y_1 Y_4 + Y_4 Y_2 + Y_3 Y_4 - \frac{Y_3 Y_2 Y_b}{Y_A}} \quad (2.8)$$

Nyní již lze zvolit konkrétní součástky místo obecných admitancí (impedancí). Tím vznikne schéma podle obrázku 2.8, které již představuje konkrétní zapojení filtru typu dolní propust' zkonstruovaného v topologii Sallen-Key.



Obrázek 2.8 – Aktivní dolní propust' v topologii Sallen-Key [16, 17].

Dosazením konkrétních admitancí (impedancí) součástek místo admitancí obecných do rovnice (2.8), vznikne rovnice (2.9), která bude následně upravena.

$$A_u = \frac{\frac{G_1 G_2 (G_B + G_A)}{G_A}}{G_1 G_2 + G_1 p C_1 + p C_1 G_2 + p C_2 p C_1 - \frac{p C_2 G_2 G_B}{G_A}} \quad (2.9)$$

Přerovněním členů v rovnici (2.9) a vytknutím členu $C_1 C_2$ ve jmenovateli vznikne rovnice (2.10).

$$A_u = \frac{\frac{G_1 G_2}{C_2 C_1} \cdot \frac{G_B + G_A}{G_A}}{p^2 + p \left(\frac{G_1}{C_2} + \frac{G_2}{C_2} - \frac{G_2 G_B}{G_A C_1} \right) + \frac{G_1 G_2}{C_2 C_1}} \quad (2.10)$$

Náhradou vodivostí pomocí odporů v rovnici (2.10) podle vztahu $G_x = \frac{1}{R_x}$ vznikne rovnice (2.11).

$$A_u = \frac{\frac{1}{C_2 C_1 R_1 R_2} \cdot \frac{\frac{1}{R_B} + \frac{1}{R_A}}{\frac{1}{R_A}}}{p^2 + p \left(\frac{1}{C_2 R_1} + \frac{1}{C_2 R_2} - \frac{R_A}{R_B R_2 C_1} \right) + \frac{1}{C_2 C_1 R_1 R_2}} \quad (2.11)$$

Rovnice (2.11) napěťového přenosu A_u dolní propusti topologie SK bude následně porovnávána se známou rovnicí (2.12) pro přenos dolní propusti druhého řádu, kde A_0 má význam napěťového zesílení v propustném pásmu, Q činitele jakosti filtru a ω_0 je mezní úhlová frekvence dolní propusti [18].

$$A_{u_{DP}} = \frac{A_0 \omega_0^2}{p^2 + p \frac{\omega_0}{Q} + \omega_0^2} \quad (2.12)$$

Položením rovníčka mezi rovnice (2.11) a (2.12) lze určit soustavu tří rovnic (2.13), (2.14) a (2.15).

$$\omega_0^2 = \frac{1}{C_2 C_1 R_1 R_2} \quad (2.13)$$

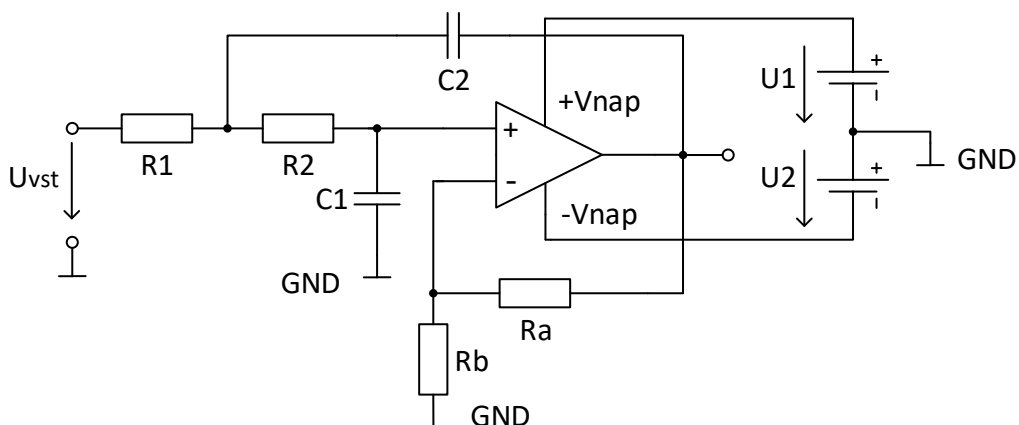
$$\frac{\omega_0}{Q} = \frac{1}{C_2 R_1} + \frac{1}{C_2 R_2} - \frac{R_A}{R_B R_2 C_1} \quad (2.14)$$

$$A_0 = \frac{\frac{1}{R_B} + \frac{1}{R_A}}{\frac{1}{R_A}} = 1 + \frac{R_A}{R_B} \quad (2.15)$$

Tato soustava obsahuje tři rovnice a 6 neznámých proměnných. Je tedy třeba volit tři parametry. Pro další výpočty byly vždy voleny hodnoty kapacit $C_1 = C_2 = C$ a hodnota odporu R_B . Zvolením těchto parametrů a dosazením známých hodnot mezní frekvence a činitele jakosti filtru s požadovaným zesílením lze určit hodnoty ostatních tří součástí.

2.3.5 Zapojení dolní propusti v topologii Sallen-Key s virtuální nulou

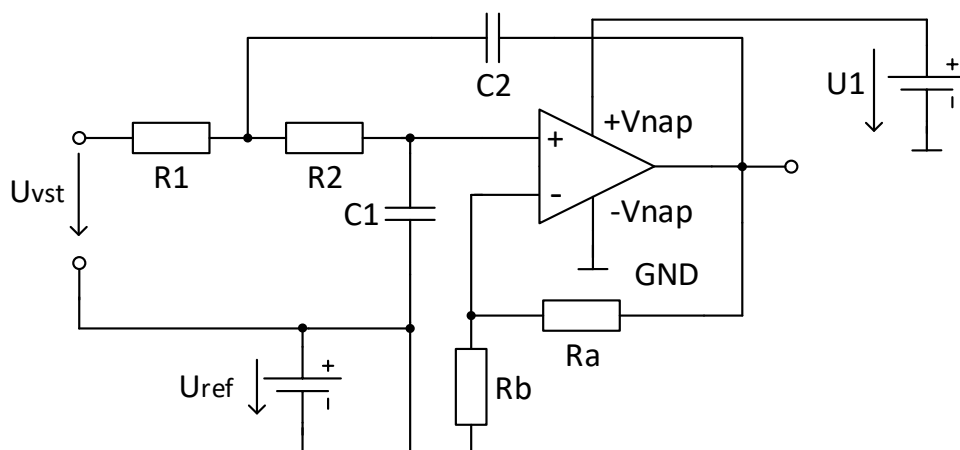
Užití operačních zesilovačů vyžaduje tvorbu symetrického napájecího napětí, což by vyžadovalo vytvoření složitějších napájecích obvodů. Vytvořením virtuální nuly a následnou úpravou základního obvodu se však lze vyhnout užití symetrického napájecího napětí OZ.



Obrázek 2.9 – Aktivní dolní propust' v topologii Sallen-Key se zakresleným napájením [16, 17].

Pakliže má být použito nesymetrické napájení OZ a má být zachována funkčnost celého obvodu, je třeba provést několik změn v zapojení podle obrázku 2.9, který ukazuje obvod dolní propusti se symetrickým napájením. Vynecháním zdroje napětí U_2 a připojením záporného napájecího vstupu OZ na nulový potenciál GND, je provedena první úprava. V druhém kroku je třeba všechny signály vztáhnout k jinému potenciálu než k nulovému (GND), čehož se dosáhne použitím zdroje napětí U_{ref} . Třetí a poslední úpravou je vztažení vstupního signálu k referenčnímu napětí U_{ref} [17].

Obrázek 2.10 ukazuje obvod dolní propusti po provedení změn. Výše zmíněnými úpravami byla v obvodu vytvořena tzv. umělá zem nebo-li virtuální nula – napětí U_{ref} . Vytvoření zdroje U_{ref} má svá pravidla. Zdroj napětí U_{ref} musí mít napětí v rozmezí mezi napětími U_1 a GND a musí být také dostatečně tvrdý. Reálně lze zdroj U_{ref} vytvořit například odporovým děličem odděleným od ostatních obvodů pomocí sledovače napětí s operačním zesilovačem [17].



Obrázek 2.10 – Aktivní dolní propust' v topologii Sallen-Key s virtuální nulou [17].

2.3.6 Sestavení antialiasingového filtru pro K_MC3 ze stupňů Sallen-Key

Senzor K_MC3 má dva výstupní signály I a Q. Ke každému ze signálů byl vytvořen stejný filtr, který byl navržen podle parametrů zvolených v předchozích kapitolách. Filtr měl mít zisk 25 dB, mezní frekvence 10 Hz a 8,5 kHz. Aproximace amplitudové frekvenční charakteristiky byla zvolena Butterworthova, řád filtru měl být minimálně dva. Celkové zapojení filtru k senzoru K_MC3, který se bude na výsledné DPS vyskytovat celkem dvakrát pro každý z výstupů I a Q, lze nalézt v přílohách, konkrétně v příloze 1 uložené na CD. Toto schéma je úplné včetně hodnot součástek.

Díky tomu, že byla zvolena Butterworthova aproximace amplitudové charakteristiky filtru, nebylo již třeba modifikovat žádné parametry filtru. Požadovanému 2. řádu filtru vyhovuje jedna sekce dolní propusti, a proto byl nejdříve sestrojen prototyp navrženého filtru s požadovaným ziskem a mezními frekvencemi. Při jeho sestrojení a oživení však vyšlo najevo, že s použitím obyčejného operačního zesilovače dochází k rozkmitání celého zapojení filtru. Proto byly zvoleny jiné hodnoty odporů a kapacit, které na tento jev neměly vliv načež k odstranění kmitů bylo třeba užít jiného kvalitnějšího operačního zesilovače nebo snížení zisku zapojení. Protože kvalitní operační zesilovače jsou drahé, bylo rozhodnuto, že filtr bude sestrojen ze dvou sekcí dolních propustí v topologii SK, kde bude použit levnější operační zesilovač.

Další zkušební filtr byl tedy sestaven ze dvou dolních propustí a jednoho invertujícího zesilovače. První sekce filtru je navržena na nižší zisk než sekce druhá. Celkově však musí zisk dávat v součtu požadovaných 25 dB. Mezní frekvence stejně tak jako jakosti jednotlivých sekcí jsou navrženy tak, aby bylo dosaženo požadovaných mezních frekvencí, 10 Hz a 8,5 kHz, a aby byl tvar frekvenční charakteristiky co nejlepší. Existují metody, jak jednotlivé sekce skládat dohromady, které však nebyly v tomto případě využity. Filtr byl navrhován tak, že se vždy odhadly parametry jednotlivých sekcí a určily se odpovídající hodnoty součástek. Následně bylo zapojení simulováno v simulačním programu, kde se kontroloval tvar frekvenčních charakteristik, dodržení zadaných mezních frekvencí a zisku. Třetí sekcí filtru je prostý invertující zesilovač, jenž není frekvenčně závislý, umožňující zvýšit či snížit výsledný zisk filtru podle aktuálních požadavků díky trimru ve zpětné vazbě. Důvodem použití této sekce je fakt, že měnit zisk dolní propusti není jednoduše možné, protože by docházelo ke změně frekvenčních charakteristik, což vyplývá z rovnic popisující dolní propust' Sallen-Key.

Protože by bylo zbytečné vytvářet na DPS symetrické napájení operačních zesilovačů, bylo rozhodnuto, že bude použito jednoduché napájení s tím, že se celý obvod upraví k použití virtuální nuly. Použití virtuální nuly obnášelo vytvořit zdroj referenčního napětí, který byl vytvořen pomocí odporového děliče a napěťového sledovače s operačním zesilovačem. Další úpravy obvodu si vyžádaly delší bádání nad způsobem, jak vázat jednotlivé sekce filtru. Nejlepší řešení sejevilo v použití odporového děliče spolu s vazebním kondenzátorem. Toto řešení bylo tedy použito. Je třeba ještě poznamenat, že bylo třeba zvolit operační zesilovače schopné pracovat s nesymetrickým napájecím napětím a také bylo nutné zvolit operační zesilovače, které dovolují potřebný rozkmit výstupního napětí.

Spodní mezní frekvenci filtru určuje kombinace vazebního kondenzátoru a vstupního odporového děliče. Odporový dělič se z pohledu střídavých signálů jeví jako paralelní kombinace jednotlivých odporů, která spolu s vazebním kondenzátorem tvoří horní propust. Vstupní část každé sekce byla navržena s ohledem na spodní mezní frekvenci, jenž byla určena jako 10 Hz. Všechny reálně zjištěné parametry těchto filtrů jsou uvedeny v kapitole 2.3.8.

2.3.7 Sestavení antialiasingového filtru pro CDM324 ze stupňů Sallen-Key

Filtr pro senzor CDM324 byl navrhován ve stejném smyslu jako filtr pro senzor K_MC3. Požadovaný zisk tohoto filtru je ovšem docela vysoký, a to 53 dB. Mezní frekvence jsou pro oba filtry totožné, stejně tak jako ostatní parametry. Celkové schéma filtru obsahující i hodnoty určených součástek lze nalézt v přílohách, konkrétně v příloze 1 uložené na CD.

Filtr pro senzor CDM324 bylo obtížné sestavit díky vysokému celkovému zisku zapojení. První zamýšlenou variantou bylo použití dvou sekcí dolních propustí. Po určení hodnot součástek bylo toto zapojení simulováno se zdárnými výsledky. Po sestavení a oživení filtru na DPS však vyšlo najevo, že výsledný filtr není stabilní, což simulace neodhalila. Byly zkoušeny různé konfigurace jednotlivých sekcí společně s hodnotami součástek, avšak neúspěšně. Jedinou další možností bylo použití velmi kvalitního a drahého operačního zesilovače, což bylo zamítnuto. Nabízelo se také snížení zisku jednotlivých sekcí dolních propustí a zařazení třetí sekce například prostého invertujícího zesilovače. Toto řešení však bylo zamítnuto, protože hrozilo riziko zesílení širokého pásma šumu.

Výsledný filtr byl tedy nakonec složen ze čtyř sekcí dolních propustí, což znamená, že výsledný filtr je filtrem osmého řádu a zisky jednotlivých sekcí jsou rozloženy vzestupně, kde první sekce má nejnižší zisk. Takovéto řešení filtru nebylo třeba, není však na škodu. Jednotlivé sekce filtru byly vždy navrhovány tak, jako v případě filtrů pro senzor K_MC3. Po určení parametrů

jednotlivých sekcí a všech hodnot součástek bylo přikročeno k simulaci celého zapojení a k následnému ověření frekvenčních charakteristik. Po ověření frekvenčních charakteristik pomocí simulačního programu byl filtr sestaven a jeho parametry byly ověřovány reálným měřením. Při těchto měřeních se však vyskytly další potíže, které spočívaly ve špatném nastavení horní mezní frekvence filtru. Reálně zjištěná horní mezní frekvence filtru se totiž diametrálně lišila vůči horní mezní frekvenci určené simulací daného zapojení. Po několika neúspěšných iteracích návrhu filtru bylo přikročeno k experimentálnímu sestavení filtru přímo z reálných součástek a vynechání kroku se simulací. Protože byly všechny hodnoty součástek určeny experimentálně, nebylo dosaženo požadované ořezové frekvence 8,5 kHz, ale frekvence vyšší, což bylo třeba vzít v potaz při volbě vzorkovací frekvence A/D převodníků. Reálně naměřené parametry filtru pro CDM324 budou uvedeny v následující kapitole 2.3.8.

Stejně jako v případě filtru pro K_MC3 bylo rozhodnuto, že i zde bude využito nesymetrické napájení, což obnášelo provést stejné úpravy jako u filtru pro K_MC3.

2.3.8 Ověření funkce navržených filtrů

Po vypočtení hodnot jednotlivých součástek filtru bylo vždy nutné ověřit výsledné vlastnosti filtru. Použitím simulačního programu byla fáze ověřování vlastností filtru ulehčena. Po zakreslení schématu do simulačního programu byly na každém filtru prováděny dva druhy analýz. Jednou ze dvou prováděných analýz byla analýza AC, která slouží ke zjištění kmitočtových vlastností daných zapojení. Ke zjištění časových průběhů signálů na výstupu filtrů bylo nutné využít tzv. tranzientní analýzy. Tranzientní analýza dovolila zjistit, zda je daná konfigurace filtru stabilní a například zda nedochází ke zkreslení výstupních signálů filtrů. Výsledky simulací je třeba brát s nadhledem, protože nemusí odpovídat realitě. Výsledné sestavení filtrů totiž není dokonalé a zapojení obsahuje značné množství parazitních vlastností, které nelze modely popsat přesně. V průběhu realizace se právě několikrát stalo, že filtr ověřený simulací nebyl stabilní a kmital nebo docházelo k posunu mezních frekvencí filtru. Po ověření zapojení a dané konfigurace filtrů pomocí simulačního programu, byly filtry sestaveny na DPS, kde byly zjišťovány jejich reálné frekvenční amplitudové charakteristiky. Všechny naměřené charakteristiky, včetně všech simulačních souborů a jejich výstupů, lze nalézt v přílohách, a to konkrétně v příloze 2 uložené na CD.

U zapojení filtru pro K_MC3 byly pomocí simulace určeny spodní ořezová frekvence 6,7 Hz a horní ořezová frekvence 8,7 kHz. Zisk je možné rozmítat v rozmezí 23 dB až 37 dB pomocí příslušného trimru ve zpětné vazbě invertujícího zesilovače, který je označen ve schématu.

Nasimulovaná frekvenční amplitudová charakteristika těchto filtrů vykazuje jisté anomálie při vyšších frekvencích, kde dochází ke zdvihu přenosu, což je zapříčiněno vlastnostmi operačního zesilovače a samotnou konstrukcí filtru. Zapojení Sallen-Key se totiž touto anomálií vyznačuje.

Pro senzor K_MC3 byly vytvořeny dva filtry, jeden pro složku IF signálu I a druhý pro složku IF signálu Q. Obě naměřené frekvenční charakteristiky jsou téměř totožné, a proto jsou zjištěné parametry také totožné. U obou filtrů byla určena spodní mezní frekvence jako 6 Hz, horní mezní frekvence 8,6 kHz a zisk cca 25 dB.

Při návrhu filtru k senzoru CDM324 docházelo k obrovským problémům s návrhem filtru. Po překonání nestability filtrů se objevovaly problémy s nastavením horní mezní frekvence filtru. Zapojení bylo simulováno přesně tak jako zapojení filtru k senzoru K_MC3. Jenže simulace vždy vycházely podle požadavků, načež po sestavení filtru a zjištění jeho parametrů vyšlo najevo, že výsledná mezní frekvence je nižší než frekvence požadovaná. Po několika iteracích návrhu, kde se výsledky simulace a dosažené reálné parametry velice lišily, byly součástky filtru zjištěny experimentálně, kde výsledná ořezová frekvence je poněkud vyšší, než by bylo třeba. V příloze 2 na CD je sice možné nalézt simulační soubory pro filtr CDM324, které však nejsou směrodatné, protože hodnoty součástek použitých na DPS jsou rozdílné. Výsledné hodnoty součástek použitých na DPS lze nalézt v příloze 1 na CD se schématem filtru nebo v příloze 6 na CD se schématem DPS. Výsledná naměřená horní mezní frekvence je 10,6 kHz, což je poněkud vyšší hodnota, než byla zamýšlena, ale vhodnou volbou vzorkovací frekvence a také vysokou strmostí výsledného filtru 160 dB/dekádu došlo k dodržení Shannonova vzorkovacího teorému. Dále byla naměřena spodní mezní frekvence 7 Hz a celkový zisk zapojení filtru 53 dB.

2.4 Mikroprocesor a nastavení periférií

V kapitole 2.1 bylo zmíněno blokové schéma systému pro zpracování IF signálu. Jeho další podstatnou částí je mikroprocesor, který by měl mít integrovány všechny potřebné periférie ke zpracování IF signálu zesíleného a vyfiltrovaného aktivním antialiasingovým filtrem.

2.4.1 Výběr mikroprocesoru

V zadání této práce je řečeno, že řetězec zpracování IF signálu má být složen z 32 bitového mikroprocesoru, A/D převodníku, sériového rozhraní a paměti pro uložení vzorků IF signálu. V dnešní době mívají mikroprocesory integrovány všechny výše zmíněné periférie.

Výpočet Fourierovy transformace je velice náročnou záležitostí jak na výpočetní výkon mikroprocesoru, tak na datovou paměť. Z těchto důvodů bylo vybíráno z mikroprocesorů obsahujících jádro ARM Cortex-M3. Mikroprocesory založené na tomto jádře totiž dosahují dobrých poměrů cena/výkon. Mikroprocesorů obsahující jádro ARM-Cortex-M3 je na trhu nepřeborné množství od velkého množství výrobců. Jednotlivé mikroprocesory se pak odlišují množstvím a vlastnostmi integrovaných periférií a také maximálními dosažitelnými takty. Na doporučení školitele byl vybrán mikroprocesor od firmy Cypress z rodiny PSoC-5LP. Důvodem tohoto rozhodnutí je fakt, že flexibilita a jednoduchost s jakou lze programovat mikroprocesory PSoC 5LP je bezkonkurenční. V porovnání s ostatními mikroprocesory ostatních firem je třeba vždy vynaložit vyšší míru úsilí k dosažení stejných výsledků. Flexibilita a jednoduchost programování mikroprocesorů PSoC 5LP tkví v promyšleném a velmi dobře zpracovaném vývojovém prostředí (IDE) s velmi dobře provedeným programovacím rozhraním (API). Nevýhodou použití API v mikroprocesorové technice jsou větší nároky na paměť a nižší rychlost běhu programu.

Z rodiny PSoC 5LP byl vybrán mikroprocesor s označením CY8C5868AXI-LP032, který obsahuje všechny požadované periferie s dostatečným paměťovým prostorem pro výpočet FFT. Maximální takt jádra může být až 67 MHz při napájení 5 V. Dále je daný mikroprocesor vybaven 256 kB flash paměti a 64 kB paměti SRAM. k dispozici jsou 3 A/D převodníky (2xSAR, 1xSigmaDelta) a integrované sériové rozhraní USB/UART. k přenosu dat je možné využít přímý přístup do paměti (DMA) s 24 kanály [15].

2.4.2 Konfigurace A/D převodníků

Mikroprocesor CY8C5868AXI-LP032 má integrovány celkem tři A/D převodníky. Dva z nich jsou A/D převodníky s postupnou aproximací (SAR), třetí pak převodník typu sigma-delta. Převodníky SAR jsou vhodné pro rychlá měření, nedosahují však rozlišení jako převodníky typu sigma delta, které jsou výrazně pomalejší. Senzor K_MC3 má dva výstupy I a Q a vyžaduje tedy vzorkovat dva kanály současně, proto byly použity právě převodníky SAR [15].

K uvedení převodníků do provozu bylo nutné nastavit jejich parametry. Převodníky měly být připojeny na výstupy AAF přímo. Protože bylo již dříve rozhodnuto, že všechny filtry budou pracovat v režimu asymetrického napájení, nebylo třeba využívat převodníků v diferenciálním režimu. Důvodem je fakt, že výstupní napětí filtrů jsou vždy stejné polarity. Výstupy filtrů totiž nejsou stejnosměrně odděleny.

Referenční napětí A/D převodníků bylo nutné volit s ohledem na výstupní signál AAF. Maximální výstupní napětí AAF je 5 V, proto bylo referenční napětí převodníků voleno také 5 V. Vstupem referenčního napájecího napětí byl volen napájecí pin VDDA.

Rozlišení A/D převodníků bylo možné volit buď 8, 10 nebo 12 bitů. Jelikož měl být signál zpracováván v 16 nebo 32 bitovém celočíselném formátu, bylo zvoleno rozlišení 12 bitů.

Jistým problémem byla volba vzorkovací frekvence A/D převodníku. Tato frekvence musela být volena s ohledem na maximální frekvence IF signálu a také s ohledem na rozlišení Fourierovy transformace. Rozlišení Fourierovy transformace totiž určuje, s jakou přesností lze určit rychlost objektu. Požadované rozlišení v rychlosti bylo zadáno jako 0,1 m/s. Tomuto rychlostnímu rozlišení odpovídá určitá rozlišovací frekvence, kterou lze jako v případě maximálních ořezových frekvencí určit dosazením do rovnice (2.2). Dosazením hodnoty 0,1 m/s za radiální rychlost do rovnice (2.2) lze určit potřebné frekvenční rozlišení f_{roz} [4, 5].

$$f_{roz} = \frac{2v_r f_V}{c} = \frac{2 \cdot 0,1 \cdot 24,125 \cdot 10^9}{3 \cdot 10^8} = 16,08 \text{ Hz} \quad (2.16)$$

Pro rozlišení FFT platí rovnice (2.16), kde f_{vz} značí vzorkovací frekvenci A/D převodníku a N_{vz} značí počet vzorků IF signálu ze kterého je určováno FFT [4, 5].

$$f_{roz} = \frac{f_{vz}}{N_{vz}} \quad (2.17)$$

Výpočet FFT se vždy provádí z počtu vzorků mocniny dvou. Pokud by byl počet vzorků zvolen jako $N_{vz} = 1024$, pak by odpovídající vzorkovací frekvence musela být určena rovnicí (2.18).

$$f_{vz} = f_{roz} \cdot N_{vz} = 16,08 \cdot 1024 = 16,465 \text{ kHz} \quad (2.18)$$

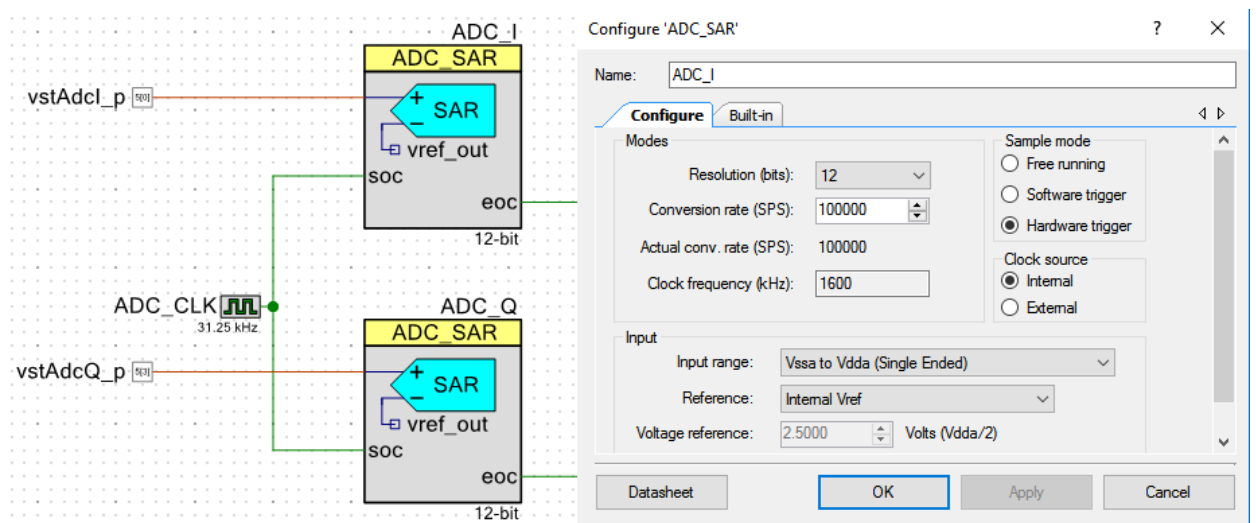
Hodnota vzorkovací frekvence určená rovnicí (2.18) je však nedostačující. Maximální frekvence IF signálu je totiž 8,5 kHz, kde je třeba ještě nechat rezervu. Vzorkovací teorém říká, že vzorkovací frekvence musí být minimálně dvakrát větší než maximální frekvence obsažená ve vzorkovaném signálu. V tomto případě by nebyl Shannonův teorém dodržen a docházelo by k aliasingu. Pakliže by byl počet vzorků volen jako $N_{vz} = 2048$, byla by odpovídající vzorkovací frekvence určena rovnicí (2.19).

$$f_{vz} = f_{roz} \cdot N_{vz} = 32,931 \text{ kHz} \quad (2.19)$$

Hodnota vzorkovací frekvence určená rovnicí (2.19) je tedy maximální možná. Zvolená vzorkovací frekvence pak tedy nesmí být vyšší, jinak by nedošlo k dodržení zadání práce. Na základě těchto poznatků byla zvolena vzorkovací frekvence $f_{vz} = 31,25 \text{ kHz}$. Bylo možné volit i menší hodnotu, ale takto je zahrnuta dostatečná rezerva vůči signálům, které nejsou ještě dostatečně potlačeny. Toto platí i s ohledem na fakt, že horní ořezová frekvence filtru pro senzor CDM324 vyšla při měření reálných frekvenčních charakteristik 10,6 kHz.

Problémem bylo, že u převodníků SAR nešlo takto nízkou vzorkovací frekvencí nastavit jednoduše. Dále musely být oba převodníky spouštěny současně, což nelze provést softwarově. Oba problémy byly vyřešeny tím, že převodníky byly ponechány na vyšší taktovací frekvenci, zde 1,6 MHz, a bylo použito hardwarové spouštění. Následně byly vytvořeny hodiny s požadovanou vzorkovací frekvencí, jejichž výstup byl připojen na vstupy hardwarového spouštění obou převodníků SAR. Vytvořené hodiny již bylo možné ovládat softwarově.

Celkovou konfiguraci A/D převodníků spolu se zakresleným blokovým schématem v prostředí PSoC Creator 4.0 ukazuje obrázek 2.11.



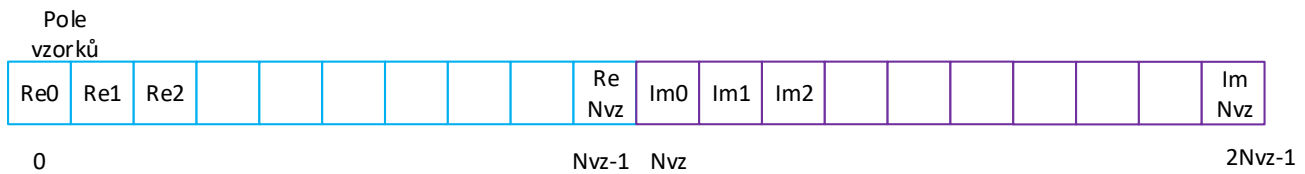
Obrázek 2.11 – Konfigurace A/D převodníku.

2.4.3 Konfigurace DMA

Výpočet FFT je náročný na výpočetní výkon, a proto byl využit přímý přístup do paměti, který umožňuje přesouvat data z A/D převodníků do paměti SRAM bez zásahu jádra mikroprocesoru.

Obrázek 2.12 ukazuje rozdělení vzorků v paměti mikroprocesoru. FFT má být vypočítáváno z N_{vz} vzorků. Pro senzor K_MC3 tedy každý ze dvou převodníků odebere ze signálu N_{vz} vzorků, které je třeba uložit do paměti. První polovina pole vzorků je vyhrazena vzorkům z A/D převodníku I, tedy reálné složce IF signálu. Druhá polovina je pak vyhrazena pro vzorky

z převodníku Q, tedy imaginární složce IF signálu. Pro senzor CDM324 je využita pouze první polovina pole vzorků, kde druhá je po každém výpočtu nulována.



Obrázek 2.12 – Rozložení vzorků v paměti MCU.

Pole vzorků má délku $2N_{vz}$, kde na indexech od 0 do $N_{vz} - 1$ jsou uložena reálná data a na indexech N_{vz} do $2N_{vz} - 1$ data imaginární.

Vybraný mikroprocesor rodiny PSoC 5LP má k dispozici několik kanálů DMA. Protože bylo zamýšleno, že všechna měření budou prováděna ve dvou formátech frakčních čísel Q1.15 a Q1.31, bylo třeba hned několik různých nastavení DMA. Jednotlivá nastavení DMA byla vytvořena pomocí nástroje integrovaného ve vývojovém prostředí PSoC Creator. Nastavení jednotlivých DMA se tedy liší jak pro použitý formát frakčních čísel, tak i pro použitý senzor. Všechna jednotlivá nastavení jsou rozebrána v komentářích ve zdrojových souborech programu k DPS, a proto zde nebudou rozebírána [15].

2.4.4 Sériové rozhraní

V zadání této práce bylo požadováno propojení systému zpracování IF signálu s PC. Mikroprocesory řady PSoC 5LP mají integrováno rozhraní USB/UART, kde stačí pouze použít tuto komponentu a není třeba nic víc nastavovat. Z důvodu jednoduchosti bylo využito právě této komponenty k sériovému spojení PC a celého systému zpracování IF signálu. Skript pro příjem dat v matlabu lze nalézt v přílohách, konkrétně v příloze 8 uložené na CD [15].

2.5 Výpočet Fourierovy transformace

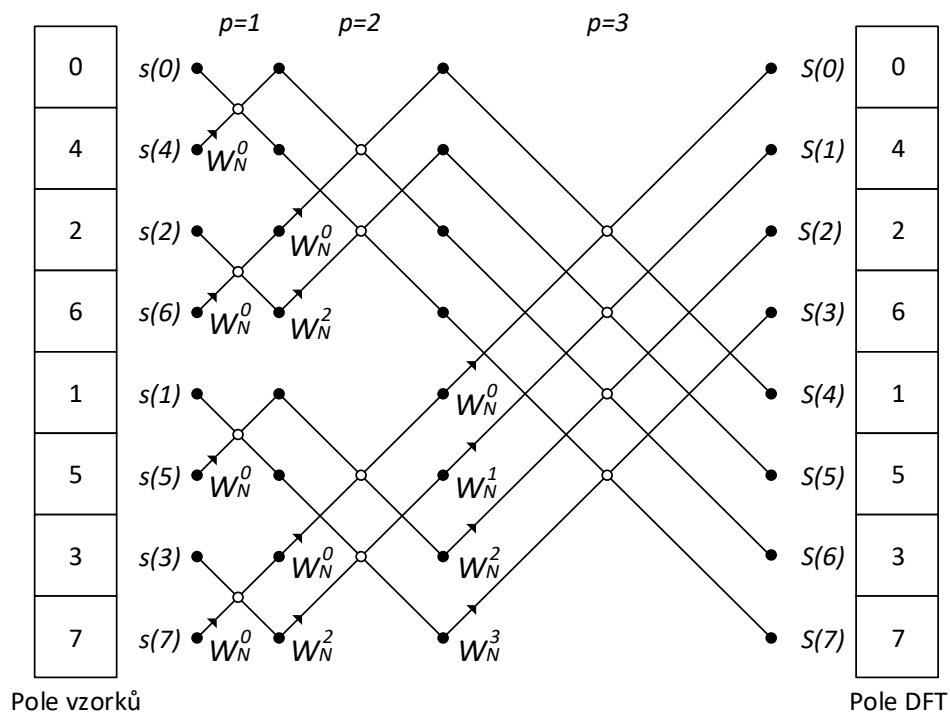
K určení Fourierovy transformace z nasbíraných vzorků signálu byly použity dvě rozdílné knihovny pro výpočet FFT. Jednou z knihoven je profesionálně vytvořená CMSIS DSP. CMSIS DSP obsahuje nepřeberné množství funkcí používaných ke zpracování signálů. Druhá použitá knihovna byla vytvořena autorem této práce na základě teoretických poznatků týkajících se FFT z kapitoly 1.3.4.

2.5.1 Rozbor algoritmu DIT FFT

V kapitole 1.3.4 byl odvozen postup, jak lze DFT o N_{vz} vzorcích rozložit na větší počet menších problémů, které lze snáze řešit. Rozborem příkladu určení DFT menšího počtu vzorků určeného

pomocí algoritmu DIT lze zjistit zákonitosti pro určení DFT o obecném počtu N_{vz} vzorků. Je samozřejmostí, že počet vzorků N_{vz} musí být vždy mocninou dvou, jinak by použití algoritmu DIT nebylo možné.

Příklad určení DFT z osmi vzorků vstupního signálu, pomocí algoritmu redukce v čase, zakresleného pomocí orientovaných grafů, ukazuje obrázek 2.13. Je třeba podotknout, že algoritmus je prováděn s použitím jediného datového pole. Na počátku jsou v tomto poli uloženy vzorky signálu, ze kterých se určuje DFT. Po dokončení algoritmu jsou v poli vzorků uloženy členy posloupnosti DFT. V praxi to znamená, že do motýlku vstupují dva vzorky z jistých paměťových míst a po provedení daných operací jsou výsledky uloženy na tatáž místa. Je tedy patrné, že výpočet probíhá ve třech úrovních. Na první úrovni je určeno DFT ze dvou vstupních vzorků vzdálených od sebe o čtyři místa, jejich počet je roven čtyřem. Fázovací člen W_N^k by zde měl mít tvar $W_{N/4}^k$, což se ovšem nechá psát jako W_N^{4k} . Na první úrovni může k nabývat pouze hodnoty nula. Na další druhé úrovni je třeba určit dvě DFT ze vzorků vzdálených od sebe o dvě místa. Fázovací člen zde má tvar W_N^{2k} a k může nabývat hodnot 0 a 1. Na úrovni poslední se již určuje pouze jedna DFT ze vzorků vzdálených od sebe o jedno místo, fázovací člen je tvaru W_N^k a k může nabývat hodnot 0,1,2,3. Výstupní posloupnost má pak bitově inverzní pořadí vzorků [8, 10, 11].



Obrázek 2.13 – Algoritmus DIT zakreslený pomocí orientovaných grafů [8].

Obecně tak lze shledat zákonitosti, které je možné použít k přepsání algoritmu například do programovacího jazyka C. Počet úrovní výpočtu m lze určit pomocí rovnice (2.20). Pokud je aktuální úroveň výpočtu označena jako p , pak je vzdálenost indexů vzorků Δ v datovém poli, vstupujících do jednotlivých DFT na této úrovni, určena pomocí rovnice (2.21). Počet DFT určovaných na jednotlivých úrovních, označený jako l , lze určit pomocí rovnice (2.22). Jednotlivé DFT na každé úrovni jsou vždy určeny pomocí i motýlků, rovnice (2.23) [8, 10, 11].

$$m = \log_2 N_{vz} \quad (2.20)$$

$$\Delta = \frac{N_{vz}}{2^p} \wedge p \in \langle 1, m \rangle \quad (2.21)$$

$$l = \frac{N_{vz}}{2^p} \quad (2.22)$$

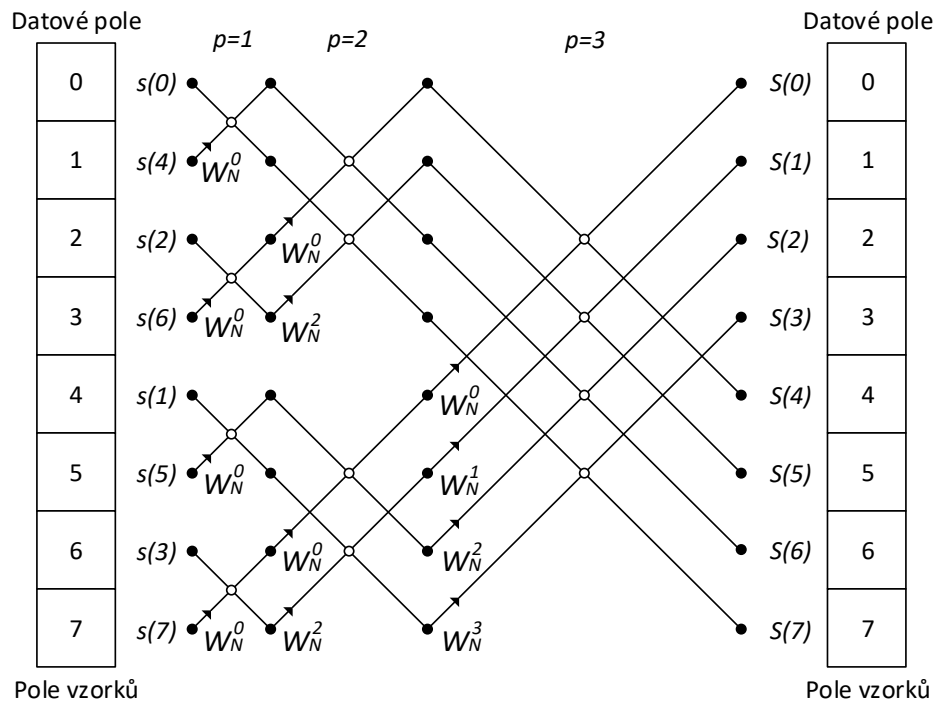
$$i = 2^{p-1} \quad (2.23)$$

Z obrázku 2.13 lze také jasně vidět, že výsledná posloupnost má bitově inverzní pořadí členů výsledné posloupnosti DFT. Datové pole obsahuje například na indexu 4 člen $S(1)$, a na indexu 6 člen $S(3)$. Bitově inverzní pořadí znamená, že pokud je vzato číslo 4 a přepsáno do dvojkové soustavy, což je 100_b , pak odpovídající index v bitově inverzním pořadí je číslo 001_b , tedy číslo 1 v dekadické soustavě. Bitově inverzní číslo lze tedy zjistit tak, že se dané číslo vezme, přepíše do binárního tvaru a zapíše s opačným pořadím cifer, kde po přepsání zpět do dekadického tvaru vznikne bitově inverzní číslo [8].

Aby byla výstupní posloupnost DFT ve správném pořadí, stačí vždy před provedením FFT zaměnit vzorek z indexu n se vzorkem na indexu bitově inverzním k indexu n . Následným provedením FFT na zaměněných vzorcích signálu se docílí správného pořadí výstupní posloupnosti DFT, což dokazuje obrázek 2.14. Zde již lze vidět, že například na indexu 1 v datovém poli leží vzorek signálu $s(4)$ a výstupní posloupnost DFT je již ve správném formátu [8].

Vzhledem k tomu, že jsou vzorky v odlišném pořadí, je třeba změnit rovnici (2.21) určující vzdálenost vzorků vstupujících do motýlka na rovnici (2.24) [8].

$$\Delta = 2^{p-1} \wedge p \in \langle 1, m \rangle \quad (2.24)$$



Obrázek 2.14 – Algoritmus DIT zakreslený pomocí orientovaných grafů s bitově inverzním pořadím [8].

2.5.2 Algoritmus DIT v matlabu

Z faktů vypočítaných v předchozích kapitolách 1.3.4 a 2.5.1 bylo možné naprogramovat algoritmus DIT v jazyce C. Jazyk C však nativně nepodporuje operace s komplexními čísly, a proto byl tento algoritmus naprogramován nejdříve v matlabu, kde byl vyzkoušen v několika verzích. Po odladění v matlabu byl algoritmus přepsán do jazyka C.

Z obrázku 2.14 je zřejmé, že při určení každé dílčí DFT je třeba násobit polovinu vstupních vzorků dané DFT členem $W_N^n = e^{-\frac{j2\pi}{N}n}$, kde n je určeno pomocí ostatních faktorů. Člen W_N^n je komplexním číslem a jeho výpočet v cyklu spolu s výpočtem DFT zdržuje celé určení DFT.

```

37     % vypocet twiddle faktoru
38 -   tabulkaExponentu = []; % twiddle faktory predpocitane
39
40 -   for i = 0:1:Nvz/2-1
41 -       tabulkaExponentu(i+1) = exp(-1j*2*pi/Nvz*i);
42 -   end

```

Obrázek 2.15 – Kód v matlabu pro výpočet fázovacích členů.

Jistým řešením tohoto problému je předpočítání členů W_N^n . Pozorováním lze zjistit, že potřebné indexy n k předpočítání členu W_N^n leží v mezích $0 - \frac{Nvz}{2} - 1$. Všechny ostatní členy jsou číselně totožné, jen s opačným znaménkem. Úryvek kódu v matlabu zajišťující výpočet těchto členů

ukazuje obrázek 2.15. Příslušné členy W_N^n jsou uloženy na indexech $n + 1$ v poli tabulkaExponentu.

```

45 % Tabulka bitově inverzního pořadí
46 - delkaIndexu = log2(Nvz); % počet bitů nutný k zapsání každého indexu
47 - revT = 0:Nvz-1; % naplnění indexů v normálním pořadí
48
49 - for c = 0:Nvz-1
50 -     pom = revT(c+1); % aby nebyla hodnota prepisována
51 -     revT(c+1) = 0; % anulace
52 -     index = delkaIndexu - 1; % ukazuje na bit odpovídající bitově inverzní cifry
53 -     for i = 0:delkaIndexu-1; % cyklus přes všechny bity
54 -         % v tomto cyklu je realizován klasický převod z desítkové do bin.
55 -         % soustavy
56 -         zbytek = mod(pom,2);
57 -         pom = pom / 2;
58 -         pom = floor(pom);
59
60 -         % zpětný převod otočené cifry do dekadické soustavy
61 -         revT(c+1) = revT(c+1) + zbytek*2^index;
62 -         index = index - 1;
63 -     end
64
65 -     % ochrana proti duplicitnímu přehození vzorku
66 -     if revT(c+1) <= c+1
67 -         revT(c+1) = c;
68 -     end
69 - end

```

Obrázek 2.16 – Skript v matlabu pro předpočítání tabulek pro bitově inverzní pořadí.

Další, co je třeba zajistit je přeházení vzorků signálu v duchu bitově inverzního pořadí. Podobným způsobem jako v přechodím případě byla předpočítána tabulka, kde index členu a hodnota členu na tomto indexu uložená ukazují na dva vzorky, které mají být zaměněny. Příkladem je index 1, na kterém je uložena hodnota 4. Znamená to tedy, že vzorky 1 a 4 mají být zaměněny. Odpovídající kód v matlabu ukazuje obrázek 2.16.

Po následném přeházení vzorků podle této tabulky již bylo možné přikročit k určení DFT pomocí algoritmu DIT. Celkově je třeba tří vnořených cyklů. Jeden vnější cyklus přepíná fázi výpočtu p . První vnořený cyklus přepíná aktuálně určenou DFT, jejichž celkový počet je l . Druhý vnořený cyklus určuje aktuálního motýlka, jejichž počet je k . V těle cyklu jsou vždy určeny nejdříve adresy vzorků vstupujících do aktuálního motýlka spolu s adresou příslušného exponenciálního členu. Následně jsou pomocí těchto adres načteny všechny potřebné členy z příslušných polí. A na konec je proveden motýlek, kde výsledky jsou uloženy zase na příslušné adresy.

Jak je ovšem vidět, je zde využito nativní podpory operací komplexních čísel v matlabu. Jazyk C tuto podporu ovšem nenabízí a je tedy třeba všechny operace s komplexními čísly přepsat na násobení a sčítání reálných čísel. V kapitole 2.4.3 o DMA bylo uvedeno, že signálová data jsou řazena za sebou tak, že nejdříve jdou všechna reálná a pak ta imaginární. Tabulku exponentů je

tedy nutné rozdělit na reálné a imaginární složky a seřadit je do podobného formátu jako pole vzorků. Všechny potřebné skripty do matlabu lze nalézt v přílohách, konkrétně v příloze 3 uložené na CD.

```

80
81  %% vlastní prepis algoritmu DIT
82
83  vzdalenostVz = 1;  % oznacena delta
84  nexp = Nvz/2;    % modifikator exponentu
85
86  while vzdalenostVz < Nvz  % prepina fazi p
87      for prepinaDFT = 0:2*vzdalenostVz:Nvz-1  % prepina jednotlivych DFT, celkem 1
88          for motylek = 0:1:vzdalenostVz-1  % prepina motylka
89              adr1 = prepinaDFT+motylek+1;
90              adr2 = adr1+vzdalenostVz;
91              pom1 = signal(adr1);
92              pom2 = signal(adr2) * tabulkaExponentu(nexp*motylek+1);
93
94              signal(adr1) = pom1 + pom2; %exp(-1j*2*pi/Nvz*nexp*cara);
95              signal(adr2) = pom1 - pom2; %exp(-1j*2*pi/Nvz*nexp*cara);
96          end
97      end
98      vzdalenostVz = vzdalenostVz * 2;  % vzdalenost se s kazdou fazi 2x zvetsi
99      nexp = nexp/2; % ale nasobek exponentu 2x zmensi
100  end

```

Obrázek 2.17 – Skript v matlabu pro výpočet DFT pomocí algoritmu FFT.

2.5.3 Algoritmus DIT v jazyce C

Po rozepsání operací s komplexními čísly již bylo možné jednoduše algoritmus naprogramovaný v matlabu přepsat do jazyka C. Protože jsou algoritmy naprogramovány totožně, není nutné je zde dále rozebírat. V přílohách lze nalézt soubory knihovny, kde je algoritmus naprogramován, konkrétně v příloze 4 na CD. Protože je algoritmus téměř totožný s tím v matlabu, je třeba mít předpočítány tabulky exponentů a tabulky pro přehazování vzorků v duchu bitově inverzního pořadí. V příloze 4 na CD lze také nalézt skripty, které generují tabulky podle zadaných kritérií ve formátech Q1.7, Q1.15 a Q1.31. Potřebné tabulky je nutné uložit do paměti mikroprocesoru a při volání funkce pro výpočet FFT je nutné tyto tabulky předat jako parametr.

Příloha 4 na CD obsahuje také soubory, ve kterých jsou naprogramovány operace s frakčními čísly. Důvod bude sdělen v následujících odstavcích. Číslo x zapsané ve formátu $Qm.n$ lze vyjádřit rovnicí (2.25). N_1 je čitatelem uloženým v paměti a číslo 2^n , tedy jmenovatel, je znám z použitého formátu Q. Tento vztah lze také použít k převodu čísla x do formátu $Qm.n$ pouhým vyjádřením členu N_1 [12].

$$x = \frac{N_1}{2^n} \quad (2.25)$$

Součet čísel ve formátu Q lze zapsat pomocí rovnice (2.26). Kde $N = N_1 + N_2$ je výsledné číslo uložené v paměti. Součet lze tedy provést velice jednoduše, pokud ovšem není brán ohled na přetečení rozsahu [12].

$$\frac{N}{2^n} = \frac{N_1 + N_2}{2^n} \quad (2.26)$$

S násobením dvou čísel je situace podstatně jiná. Rovnice (2.27) popisuje výsledek násobení dvou frakčních čísel uloženého bez korekce do paměti. Jak je tedy vidět, výsledné číslo $\frac{N}{2^n}$, které by bylo v paměti uloženo jako $N = N_1 \cdot N_2$ s předpokladem jmenovatele 2^n , by nebylo určeno správně. Rovnítko mezi levou a pravou stranou rovnice (2.27) totiž neplatí [12].

$$\frac{N}{2^n} \neq \frac{N_1}{2^n} \cdot \frac{N_2}{2^n} \quad (2.27)$$

Ke správnému určení výsledku je třeba levou stranu rovnice (2.27) roznásobit členem $\frac{1}{2^n}$, kde vznikne rovnice (2.28). Výsledné číslo uložené v paměti je tedy $\frac{N}{2^n} = \frac{N_1 \cdot N_2}{2^n}$, kde lze již předpokládat jmenovatel zlomku 2^n [12].

$$\frac{N}{2^n} = \frac{N_1}{2^n} \cdot \frac{N_2}{2^n} \quad (2.28)$$

Bez použití těchto zásad při násobení frakčních čísel by nebylo možné určit FFT v jakémkoli formátu Q. Což je i důvodem, proč je zde tento fakt zmiňován.

S použitím frakčních čísel ve funkcích pro výpočet DFT je také třeba zavést protiopatření proti přetečení rozsahu daného Q formátu. Rozsah čísel, které lze do daného formátu uložit je určen intervalem $\{-2^{m-1}, 2^{m-1} - 2^{-n}\}$. Aby se přetékání rozsahu zabránilo, je při každém načtení vzorků signálu toto číslo vyděleno dvěma. V jazyce C je dělení dvěma provedeno jako rotace o jedno místo doprava. Je nutné říci, že opatření proti přetečení rozsahu není nejlepší volbou, protože díky němu dochází k určitým nepřesnostem určení amplitudy signálu. Proporce vypočtené FFT jsou ovšem zachovány, což nevadí pro aplikaci měření rychlosti pomocí CW radaru. Lepším řešením by mohla být kontrola přetečení v každém motýlku, což by ovšem zvýšilo čas potřebný k výpočtu FFT, což je i důvod proč nebyl tento způsob použit [12].

2.5.4 Knihovna CMSIS DSP

Knihovna CMSIS DSP je profesionálně vytvořená knihovna, která obsahuje velké množství funkcí pro číslicové zpracování signálu. Proto zde nechybí ani funkce pro výpočet FFT. Tyto funkce však nevyužívají k výpočtu FFT pouze jednoho algoritmu, ale využívají kombinaci mnoha algoritmů. Autor této práce se také pokoušel využít k výpočtu FFT více algoritmů, nebylo však dosaženo požadovaných výsledků, a proto zde nejsou zmiňovány. Důvodem je fakt, že je třeba hlubších znalostí programování a daného mikroprocesoru, které autor nemá. Všechny důležité informace o této knihovně můžou být nalezeny ve velmi kvalitně zpracované dokumentaci této knihovny, viz [19].

Knihovna CMSIS DSP zde byla použita k porovnání vlastností knihovny pro výpočet FFT, kterou naprogramoval autor této práce.

2.5.5 Porovnání jednotlivých knihoven FFT

Porovnání knihovny naprogramované autorem založené na algoritmu DIT je nutné k zhodnocení dosažených vlastností této knihovny. Jako knihovna k porovnání byla vybrána právě knihovna CMSIS DSP.

Aby bylo možné knihovny porovnat, bylo třeba zajistit stejné výpočetní podmínky. Z tohoto důvodu byly vytvořeny dva separátní projekty v prostředí PSoC creator, kde byl každý z obou projektů nastaven stejně. Aby bylo možné odečítat výsledky časového měření na mikroprocesoru, byl použit časovač a LCD displej. Časovač měl nastaven časový krok 10 μ s. k jednoduchému převodu dat k zobrazení na LCD displeji pak byla ještě inkludována knihovna „newlib-nano“ s podporou formátování float. Takt mikroprocesoru byl nastaven na 60 MHz. Následně byla do každého projektu inkludována daná knihovna spolu s testovacími daty. Data měla pro každou knihovnu stejnou velikost, pouze rozdílné formátování, protože obě knihovny očekávají vstupní data v různém formátu. Oba projekty spolu se skripty pro generování testovacích dat lze nalézt v příloze 5 uložené na CD. Měření byla prováděna pro formáty frakčních čísel Q1.15 a Q1.31. Při měření byly sledovány vždy tři veličiny, vždy v závislosti na počtu vzorků vstupního signálu. První sledovanou veličinou byl čas nutný k výpočtu FFT, druhou pak obsazení programové paměti a poslední veličinou obsazení paměti SRAM, tedy paměti dat.

Tabulka 3 – Naměřená data algoritmů FFT ve formátu Q1.15.

Formát Q1.15, takt jádra 60 MHz						
	Algoritmus DIT			CMSIS DSP		
Počet vzorků [-]	Čas výpočtu [ms]	Obsazení flash [kB]	Obsazení SRAM [kB]	Čas výpočtu [ms]	Obsazení flash [kB]	Obsazení SRAM [kB]
64	0,43	14,43	3,23	0,16	17,63	3,23
128	1,00	14,93	3,48	0,42	18,18	3,48
256	2,29	15,93	3,98	0,81	19,30	3,98
512	5,29	17,93	4,98	2,02	21,52	4,98
1024	11,96	21,93	6,98	4,01	26,02	6,98
2048	26,34	29,93	10,98	9,59	34,96	10,98
4096	57,42	45,93	18,98	19,16	52,96	18,98

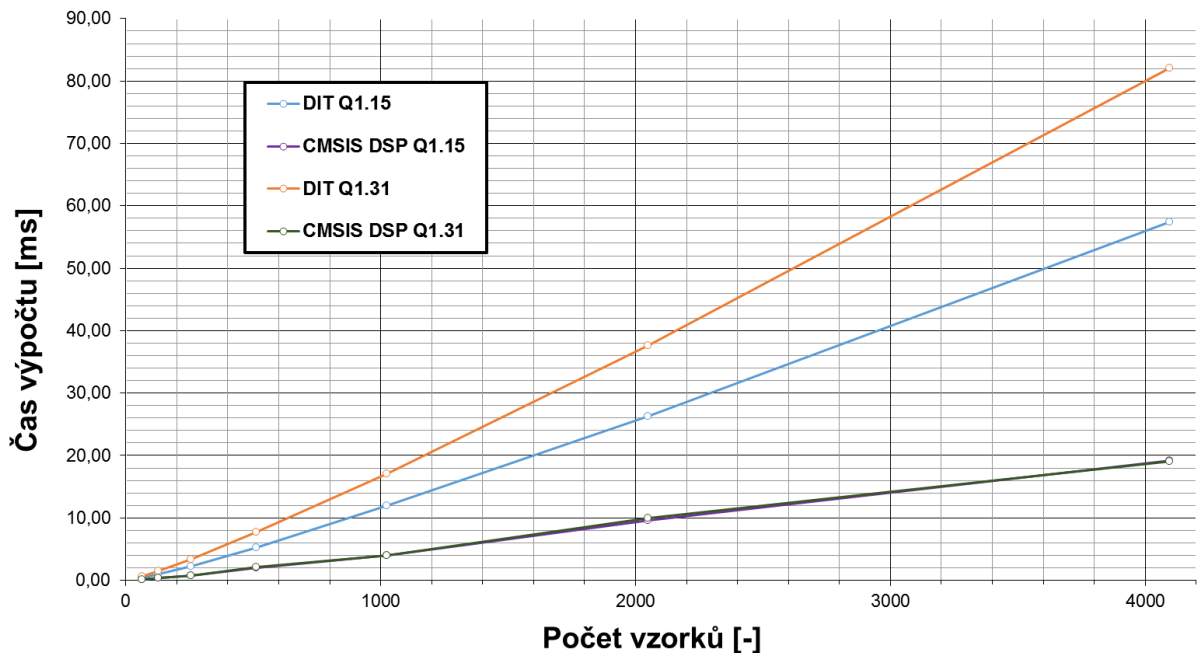
Tabulka 3 ukazuje naměřené hodnoty obou knihoven ve formátu Q1.15. Tabulka 4 pak ukazuje naměřené hodnoty pro formát Q1.31.

Tabulka 4 – Naměřená data algoritmů FFT ve formátu Q1.31.

Formát Q1.31, takt jádra 60 MHz						
	Algoritmus DIT			CMSIS DSP		
Počet vzorků [-]	Čas výpočtu [ms]	Obsazení flash [kB]	Obsazení SRAM [kB]	Čas výpočtu [ms]	Obsazení flash [kB]	Obsazení SRAM [kB]
64	0,63	14,78	3,48	0,15	17,48	3,48
128	1,47	15,66	3,98	0,42	18,46	3,98
256	3,36	17,41	4,98	0,78	20,46	4,98
512	7,67	20,91	6,98	2,08	24,43	6,98
1024	17,07	27,91	10,98	3,95	32,43	10,98
2048	37,56	41,91	18,98	9,97	48,10	18,98
4096	82,13	69,91	34,98	19,04	80,37	34,98

Vynesenou grafickou závislost výpočetního času jednotlivých algoritmů a formátů ukazuje obrázek 2.18. Zde je jasně vidět, že výpočet FFT pomocí knihovny CMSIS DSP je značně rychlejší. Rychlost výpočtu FFT pomocí knihovny CMSIS DSP, je ve formátech Q1.15 a Q1.31 téměř totožná. Algoritmus DIT naprogramovaný autorem této práce je pro formát Q1.31 pomalejší než pro formát Q1.15. Důvod, proč je knihovna naprogramovaná autorem pomalejší, spočívá hlavně v dokonalosti programování. Knihovna CMSIS DSP je programována s dokonalými znalostmi jednotlivých algoritmů výpočtu FFT a také dokonalou znalostí daných

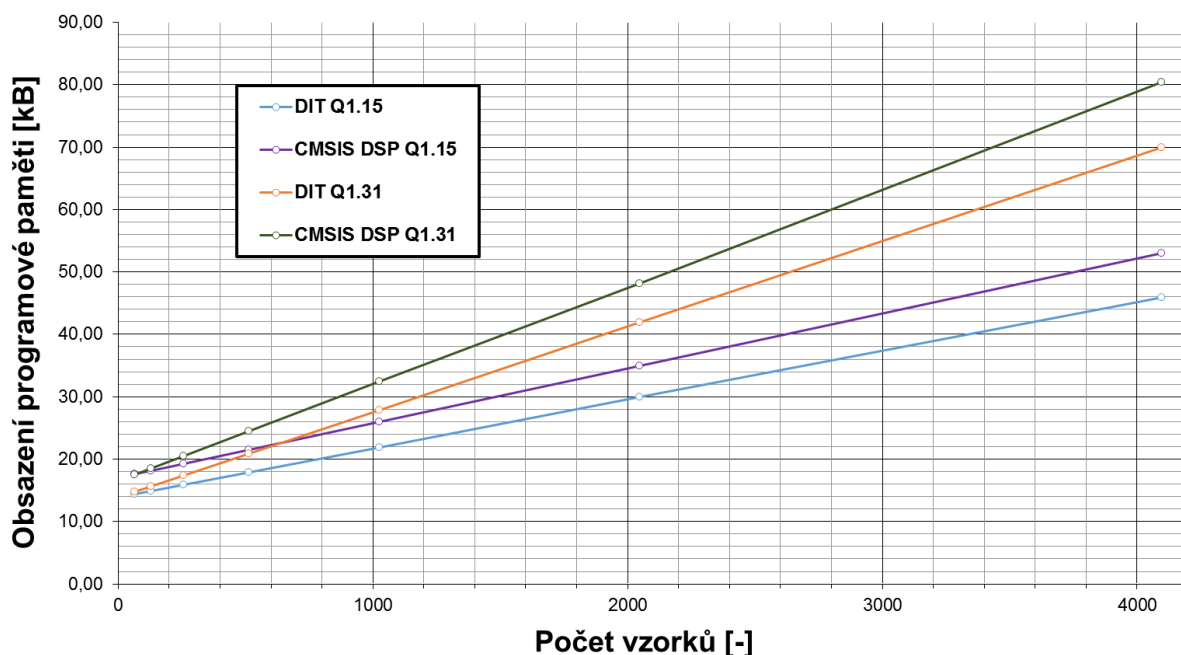
mikroprocesorů. Knihovna CMSIS DSP také kombinuje více algoritmů k maximálnímu snížení výpočetních časů. Když se autor této práce pokoušel o kombinaci více algoritmů, bylo dosaženo ještě horších výsledků než s jednoduchými algoritmy, což bylo zapříčiněno právě nedokonalou znalostí daného mikroprocesoru a nedostatečnou znalostí programování [19].



Obrázek 2.18 – Závislost času výpočtu FFT jednotlivých algoritmů na počtu vzorků

Grafickou závislost obsazení programové paměti na počtu vzorků vstupního signálu ukazuje obrázek 2.19. Z této závislosti lze zjistit, že knihovna naprogramovaná autorem je úspornější než knihovna CMSIS DSP. Programovou paměť zabírají hlavně předpočítané tabulky pro bitově inverzní pořadí a tzv. twiddle faktory, což jsou exponenty, kterými je vždy násobena polovina vzorků vstupujících do DFT v každé fázi výpočtu. Vzniklý rozdíl v obsazení programové paměti lze pravděpodobně odůvodnit tím, že autor předpočítával exponenciální členy pouze pro hodnoty $n \in \langle 0, \frac{N_{vz}}{2} - 1 \rangle$. Hodnoty v intervalu $n \in \langle \frac{N_{vz}}{2}, N_{vz} - 1 \rangle$ jsou stejně velké jen s opačnými znaménky, což bylo uvažováno už při realizaci samotných motýlků. V knihovně CMSIS DSP jsou nejspíše předpočítány všechny exponenciální členy [19].

Nabízelo se ještě vynést grafickou závislost obsazení paměti SRAM, tedy paměti dat. Tato závislost nebyla vynášena, protože obě knihovny mají tuto závislost pro stejné formáty Q totožnou. Což je zapříčiněno tím, že jsou v paměti dat uloženy vzorky signálu, které mají totožnou velikost a také tím, že oba projekty pro každou knihovnu byly nastaveny stejně.



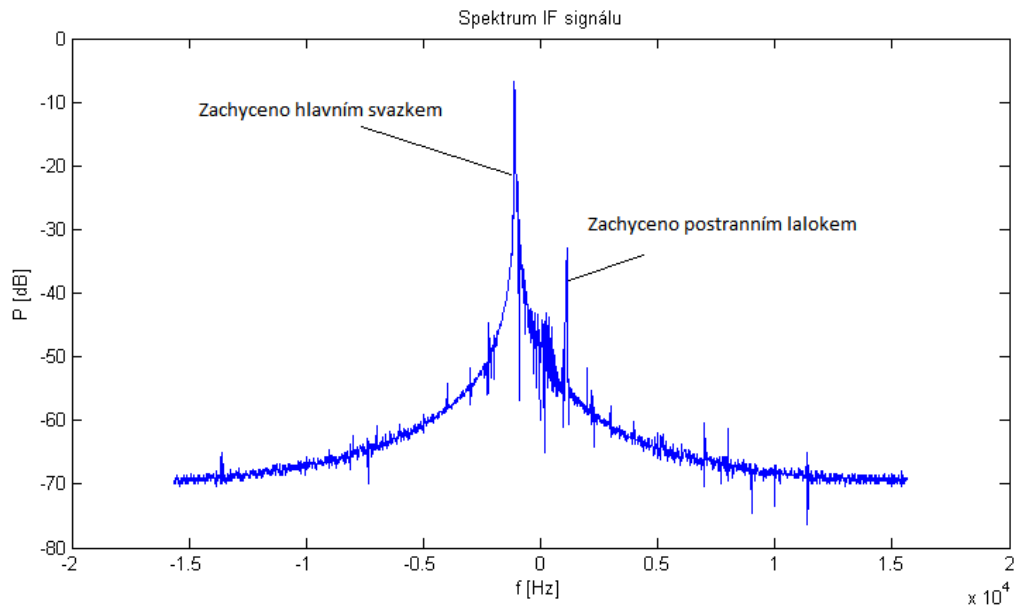
Obrázek 2.19 – Závislost obsazení programové paměti MCU jednotlivých algoritmů na počtu vzorků

2.6 Odhad rychlosti

Po převodu analogového signálu z výstupu AAF na signál digitální a určení jeho Fourierovy transformace je třeba odhadnout radiální rychlost objektu, na kterém je umístěn radarový senzor.

V kapitole 1.2.1 byl ukázán základní princip měření rychlosti objektu pomocí CW radarového senzoru. V téže kapitole byl také popsán vztah pro skutečnou rychlost cíle určenou pomocí rovnice (1.16) a vztah (1.26), který popisuje IF signál na výstupu radarového senzoru. Frekvence IF signálu na výstupu radarového senzoru je úměrná právě vzájemné radiální rychlosti radarového senzoru a povrchu, po kterém se tento objekt pohybuje.

Vztah (1.26) však platí pouze za předpokladu použití velice úzce směrové přijímací a vysílací antény radarového senzoru a přítomnosti pouze jediného pohybujícího se objektu v dosahu radarového senzoru. Nemá-li použitý radarový senzor velice úzce směrovou vysílací a přijímací anténu, dojde k modulaci IF signálu šířkou svazku vysílací a přijímací antény. Výsledný IF signál pak bude složen z více frekvenčních složek, což se projeví rozšířením frekvenčního spektra IF signálu. Lze říci, že čím menší bude šířka svazku vysílací a přijímací antény radarového senzoru, tím užší bude výsledné spektrum [4].



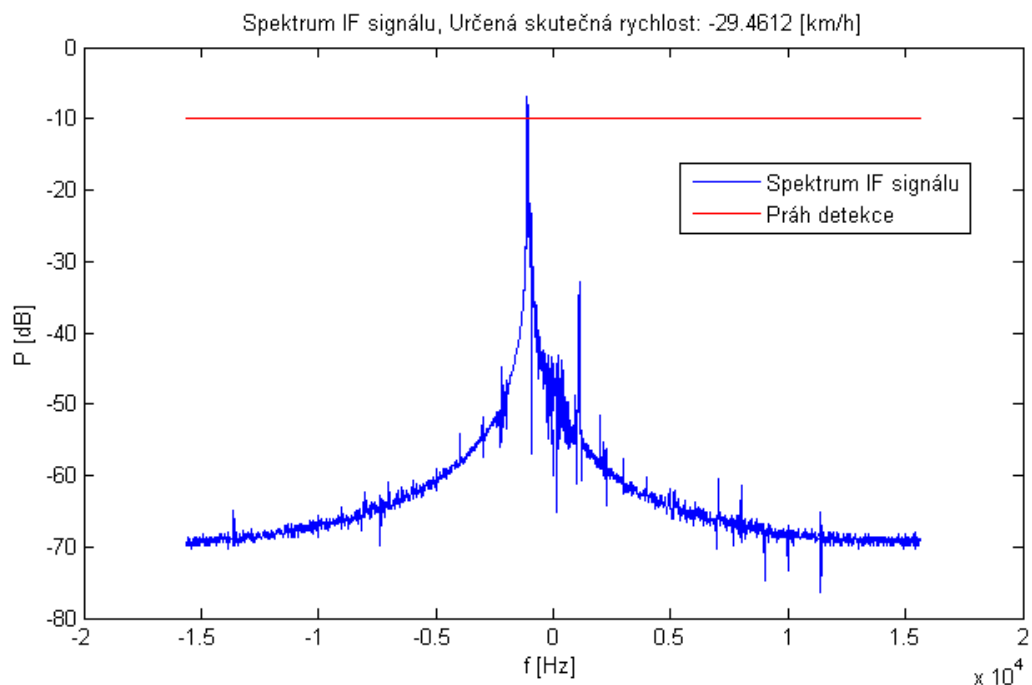
Obrázek 2.20 – Spektrum IF signálu získaného pomocí senzoru K_MC3

Obrázek 2.20 pak ukazuje spektrum IF signálu získaného pomocí senzoru K_MC3, který byl umístěn na pohybuícím se objektu a namířen na povrch, po kterém se tento objekt pohyboval. Ve spektru podle obrázku 2.20 lze nalézt dominantní špičku, která odpovídá odrazu od povrchu modulovaného šířkou svazku antény a zarovnanou na frekvenci, která odpovídá vzájemné radiální rychlosti radarového senzoru a povrchu. Menší dominantní špička odpovídá nejspíše odrazu zachyceného postranním lalokem přijímací antény. Spektrum obsahuje také šum.

Obecně může být spektrum IF signálu jakékoliv a pro správný odhad rychlosti cíle je nejprve nutné jej odhalit. k odhalení cíle lze užít například algoritmy CFAR. Konkrétně algoritmus CA - CFAR byl zkoušen pro detekci cíle v této práci. Zkoušením na reálných datech bylo zjištěno, že funguje velice dobře. Navzdory tomuto faktu bylo rozhodnuto, že zde nebude využit, protože by bylo třeba provádět více komplexnější analýzu jeho výstupů, což nebylo třeba [1, 4, 5].

Analýzou naměřených dat totiž bylo zjištěno, že spektrum IF signálu se téměř vždy skládá z hlavní špičky způsobené odrazem od povrchu a menší špičky způsobené pravděpodobně odrazem zachyceným postranním lalokem přijímací antény. Ze znalosti těchto faktů bylo usouzeno, že pro správnou detekci cíle a určení radiální rychlosti objektu postačuje zjistit maximální hodnotu spektra IF signálu a následně určit práh detekce, který je menší oproti maximu o 3 dB. V případě, že není ve spektru přítomen požadovaný odraz od cíle a nachází se zde pouze šum, dojde k nesprávnému odhadu rychlosti, protože se algoritmus zacílí na nejvyšší špičku šumu. Proto bylo zavedeno protiopatření ve formě porovnání určeného prahu detekce se

střední hodnotou spektra. Pokud je určený práh detekce alespoň o 20 dB vyšší, než je střední hodnota spektra, pak je práh určen správně. V opačném případě leží všechny amplitudy frekvenčních složek pod prahem detekce, nyní určeným jako střední hodnota spektra zvětšená o 20 dB. Následně jsou nalezeny všechny frekvenční složky s amplitudou větší než práh detekce. Z nalezených frekvenčních složek je následně určena střední frekvence, která je pomocí vztahu (1.21) přepočtena na danou radiální rychlost. Skutečná rychlost objektu je pak určena pomocí vztahu (1.16). Obrázek 2.21 ukazuje spektrum IF signálu s vykresleným prahem detekce, určeným právě popsáním postupem. Je třeba podotknout, že se předpokládá existence pouze jednoho cíle v dosahu radarového senzoru, jinak by daný algoritmus nemohl fungovat. To je však v souladu se zadáním této práce.



Obrázek 2.21 – Spektrum IF signálu s vykresleným prahem detekce

2.7 Výsledné schéma, DPS a program pro mikroprocesor

2.7.1 Schéma a DPS

K zakreslení schématu a k vytvoření desky plošných spojů byl použit program EAGLE. Schéma společně s deskou plošných spojů lze nalézt v příloze 6 na CD.

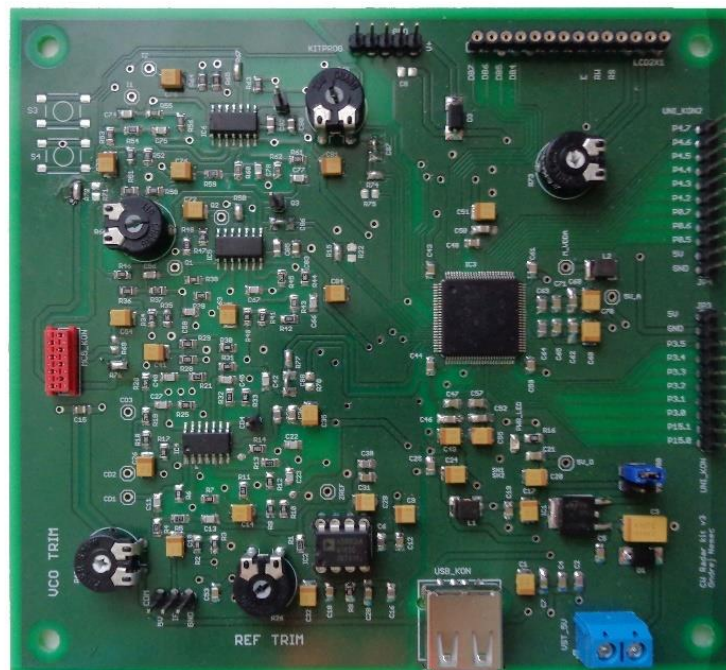
Výsledná DPS obsahuje celkem tři antialiasingové filtry. Dva z nich jsou určeny k senzoru K_MC3, který má dva výstupy, I a Q. Každý z těchto dvou filtrů obsahuje trimr k nastavení zisku. Třetí filtr je určen pro senzor CDM324. Všechny filtry pracují s nesymetrickým napájecím napětím, a proto k nim byl vytvořen referenční zdroj napětí, který je pro všechny

filtry společný. Jeho výstupní napětí lze regulovat trimrem. Senzor K_MC3 nabízí dva páry výstupů I a Q. Aby bylo možné volit mezi nimi i po vytvoření DPS, byly oba dva připojeny na vstup příslušného filtru. Propojení je realizováno pomocí nulových odporů. Výstupy všech filtrů jsou vyvedeny na zvláštní vývody mikroprocesoru. Kdyby bylo třeba, je zde vytvořena sada propojovacích míst, aby bylo možné využít převodník v diferenciálním režimu. Je třeba podotknout, že senzor K_MC3 má vstup určený k frekvenčnímu rozmítání. Aby měl vysílaný signál do prostoru požadovanou frekvenci, bylo třeba přivést na tento vstup napětí 2 V. k tomuto účelu je na DPS umístěn příslušně označený potenciometr.

DPS je možno napájet dvěma způsoby. Pro přivedení obecného napájecího napětí v rozmezí od 7 do 12 V je zde stabilizátor, který zajišťuje stabilizaci napětí na požadovaných 5 V. Druhou možností napájení desky je pomocí 5 V z USB portu. USB port zde realizuje propojení DPS s počítačem pomocí integrovaného rozhraní USB/UART přímo v mikroprocesoru.

Programování mikroprocesoru je realizováno pomocí programátoru KITPROG pomocí rozhraní SWD.

K zobrazování informací je na DPS vyvedena sběrnice pro LCD displej. Pro případ dalšího použití DPS nebo vzniku nečekaných událostí je zde také vyvedena řada pinů mikroprocesoru na výstupní hřebeny.



Obrázek 2.22 – Osazená DPS

Deska plošných spojů je oboustranná, kde všechny součástky jsou rozmístěny jen z jedné strany. Na DPS jsou odděleny analogové a digitální země. Obrázek 2.22 pak ukazuje již osazenou DPS.

2.7.2 Program pro mikroprocesor

Celý projekt vývojového prostředí PSoC Creator je spolu se zdrojovými soubory programu přiložen v příloze 7 na CD. Program je napsaný v jazyce C a je dostatečně popsán pomocí komentářů přímo ve zdrojových souborech.

V programu je realizována všechna obsluha potřebných periférií a určována FFT spolu s odhadem rychlosti. FFT je určována pomocí algoritmu DIT naprogramovaného autorem. Odhad rychlosti cíle je realizován způsobem, o němž bylo pojednáno v kapitole 2.6. Všechny informace jsou pak zobrazovány na LCD displeji.

Protože bylo nutné v průběhu testování měnit formáty frakčních čísel a také senzory, byl program vytvořen univerzálně. Pomocí maker lze jednoduše měnit parametry nastavení jednotlivých periférií, tedy DMA spolu s A/D převodníky, a také parametry výpočtů. Před změnou senzoru, či změnou formátu frakčních čísel je nutné DPS přeprogramovat nově vytvořeným programem. Toto řešení není nejrychlejší možné, ale je přehledné a velice univerzální.

Komunikace s PC je realizována pomocí integrovaného sériového rozhraní USB/UART přímo v mikroprocesoru. Do PC jsou odesílána všechna data, tedy vzorky signálu v nezměněné formě a všechny vzorky vypočtené FFT spolu s odhadem rychlosti. Příjem dat je v PC realizován pomocí matlabu, kde potřebný skript lze nalézt v příloze 8 na CD.

Pokud není komunikace s PC požadována, je možné makrem přenastavit program pro mikroprocesor do režimu smyčky, kdy je cyklicky určována FFT a prováděn odhad rychlosti.

2.8 Ověření funkce celého řetězce zpracování IF signálu

V této kapitole je popsána metodika ověření funkce celého řetězce zpracování IF signálu a je zde proveden rozbor naměřených dat.

2.8.1 Metodika měření

Pokud měl být ověřen celý řetězec číslicového zpracování signálu, bylo nutné umístit radarový senzor na pohyblivý objekt a namířit jej na povrch, po kterém se tento objekt pohyboval. Z praktických důvodů byl radarový senzor umístěn na automobil, konkrétně na tažné zařízení. Pro oba radarové senzory byla vytvořena samostatná polystyrenová schránka. Senzory pak byly

do této schránky ukotveny tak, aby nebyla zastíněna přijímací a vysílací anténa a nedocházelo tak k přidavným útlumům vyslané a přijímané elektromagnetické vlny. Celá schránka pak byla připevněna na tažné zařízení automobilu a nakloněna směrem k povrchu. Po připevnění na tažné zařízení byl určen úhel α pod kterým byl senzor natočen k povrchu od vodorovného směru, viz. kapitola 1.2.1. Úhel α je nutné znát, aby bylo možné z určené radiální rychlosti zjistit skutečnou rychlost objektu.

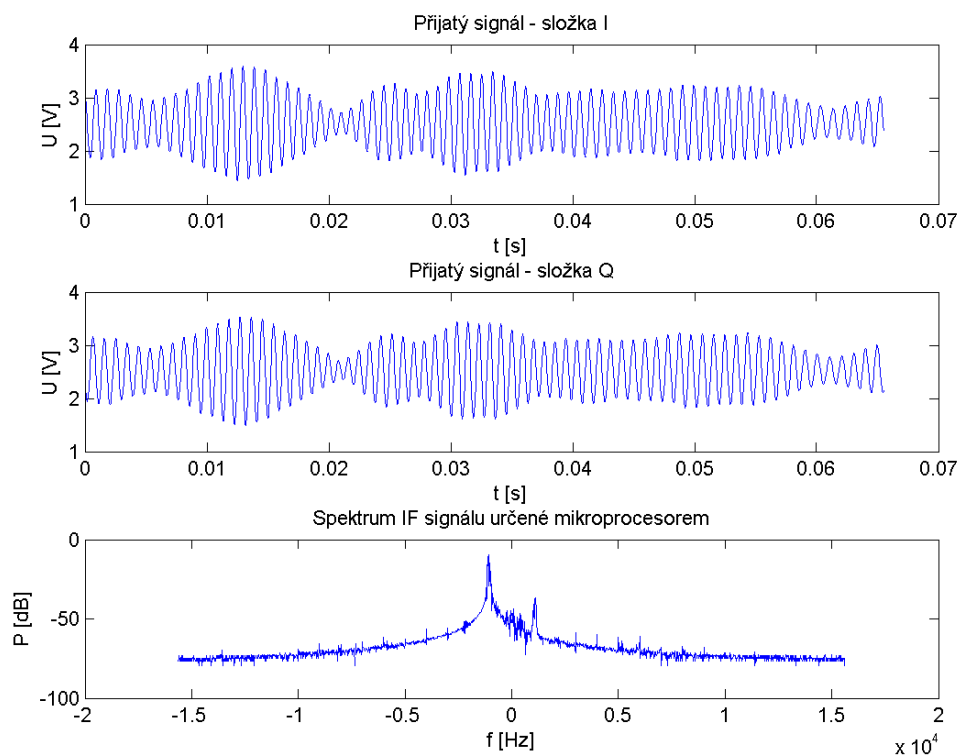
Radarový senzor byl následně připojen do DPS. Při použití radarového senzoru K_MC3 bylo nutné před započítím měření doladit zisk AAF pro I a Q výstupy, tak aby při jízdě nedocházelo k omezení rozkmitu výstupního signálu těchto filtrů. Bylo také třeba zajistit, aby oba zisky byly totožné, čehož bylo dosaženo v laboratoři připojením známého signálu na vstup AAF a následným laděním.

Automobil se následně začal pohybovat a s tím také započal sběr dat. Aby bylo možné ověřit správnost měření, bylo třeba na začátku každé sady měření vytvořit úsek s konstantní a známou rychlostí automobilu, aby bylo možné ověřit správnost určení úhlu α . Po získání kalibračního úseku začal volný úsek, kdy se automobil začal pohybovat nekonstantní rychlostí. Z DPS byly do PC vždy odesílány kompletní sady dat, všechny vzorky signálu, posloupnost FT určené na mikroprocesoru a odhad skutečné rychlosti proveden mikroprocesorem. S každým radarovým senzorem bylo nasbíráno několik sad odběhů.

2.8.2 Rozbor naměřených dat pomocí senzoru K_MC3

Z mikroprocesoru, který číslicově zpracovává IF signál, byly vždy do počítače odeslány kompletní data obsahující samotné vzorky signálu, vzorky posloupnosti určené Fourierovy transformace a také informace o odhadu radiální rychlosti, data byla sbírána ve formátu Q1.31. Radarový senzor K_MC3 byl vždy umístěn na tažném zařízení automobilu s úhlem natočení od vodorovného směru $\alpha = 32^\circ$.

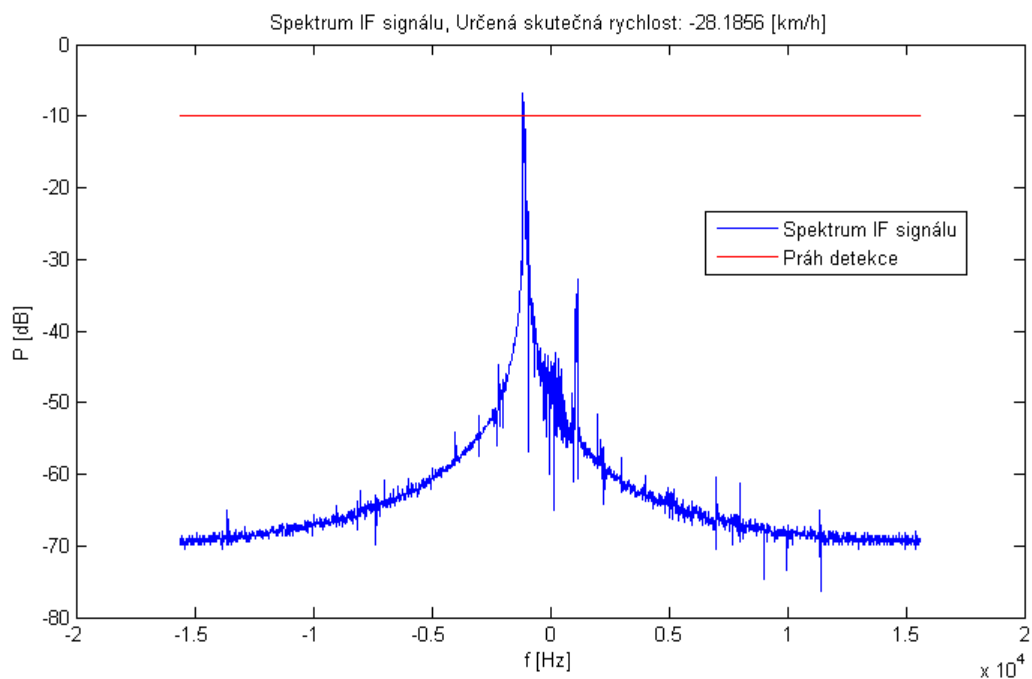
Data naměřená pomocí senzoru K_MC3 lze nalézt v příloze 8 na CD. Tato příloha obsahuje několik sad odběhů spolu s jedním kalibračním odběhem, kde byla držena téměř konstantní rychlost automobilu 30 km/h. Všechny sady dat jsou uloženy do matlabovského formátu, proto lze v příloze také nalézt skript pro jednoduché vykreslení těchto dat. Obrázek 2.23 ukazuje vykreslené průběhy z jednoho odběhu kalibrační sady dat.



Obrázek 2.23 – Grafické zobrazení jednoho odběhu dat.

Na obrázku 2.23 lze nalézt vykreslené jednotlivé složky I a Q IF signálu, ze kterých je určováno spektrum IF signálu na třetím grafu. Téměř všechna spektra určená z naměřených dat měla takovýto průběh. Zde lze totiž nalézt jednu dominantní špičku, která odpovídá právě hlavnímu odrazu od povrchu zarovnaného na Dopplerově frekvenci úměrné radiální rychlosti radarového senzoru a povrchu. Druhá menší špička má nejspíše význam odrazu zachyceného postranním lalokem přijímací antény. Zmíněnou špičku lze totiž nalézt v každém spektru, její pozice se mění s radiální rychlostí. Hodnota šumu je také v každém spektru téměř totožná.

Radarový senzor K_MC3 má šířku svazku vysílací a přijímací antény 7° na 25° , díky čemuž je výsledná hlavní špička ve spektru velice úzká. A právě díky tomuto faktu lze poměrně snadno určit radiální rychlost, která je následně přepočtena na skutečnou rychlost pomocí vztahu (1.16).

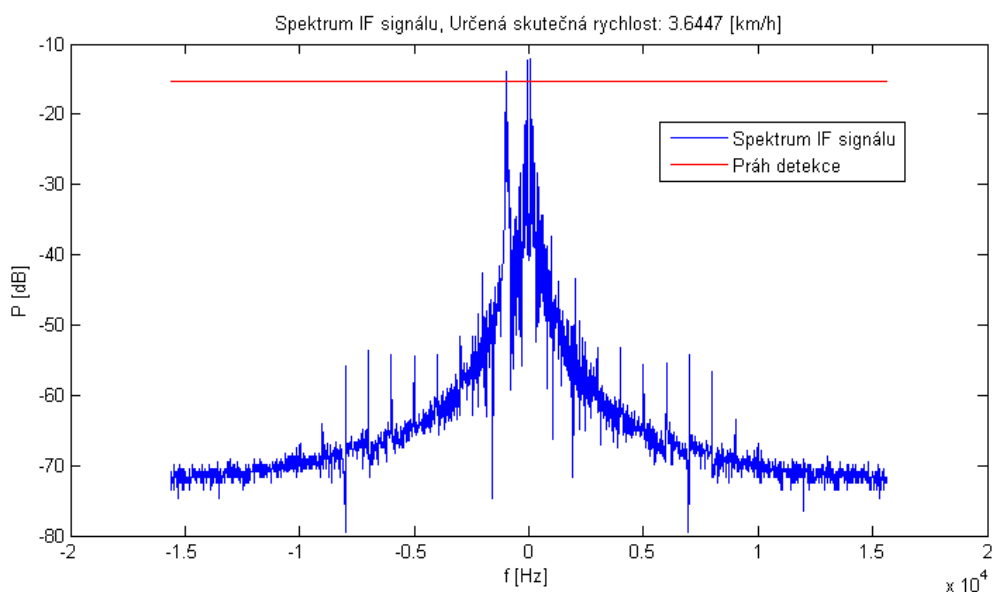


Obrázek 2.24 – Spektrum IF signálu získaného pomocí senzoru K_MC3.

Obrázek 2.24 ukazuje spektrum IF signálu vybraného z kalibrační sady odběhů. Je zde také zakreslen práh detekce cíle. V kapitole 2.6 bylo řečeno, že práh pro detekci cíle je určován jako pokles o 3 dB proti maximální hodnotě nalezené v daném spektru. Z frekvenčních složek, jejichž amplituda je vyšší než určený práh detekce cíle, je určena nejprve průměrná Dopplerova frekvence. Z průměrné Dopplerovy frekvence je určena vzájemná radiální rychlost radarového senzoru a povrchu, po kterém se pohybuje automobil. Z této radiální rychlosti je pak určena skutečná rychlost automobilu, která je vždy vypočtena v kilometrech za hodinu. Obrázek 2.24 byl vygenerován pomocí skriptu, který je také přiložen v příloze 8 na CD.

Určená skutečná rychlost byla vždy o trochu nižší než rychlost, která byla odečtena z tachometru automobilu v době odběru dat. Vše by však mělo být v pořádku, protože se lze domnívat, že tachometry automobilu ukazují hodnotu okamžité rychlosti vyšší, než skutečně je. Ke skutečné kalibraci celého zařízení by bylo nutné uspořádat náročnější experiment, což nebylo provedeno s ohledem na potřebné vybavení. Pokud by byly zjištěny odchylky od skutečné rychlosti, lze shledat příčinu například v chybně určeném úhlu α nebo například odchylkou vysílací frekvence senzoru od hodnoty použité pro výpočet skutečné rychlosti zapříčiněnou například teplotními výkyvy.

Při ověřování funkce celého řetězce zpracování IF signálu se však objevilo několik situací, kdy docházelo k chybnému určení skutečné rychlosti automobilu.

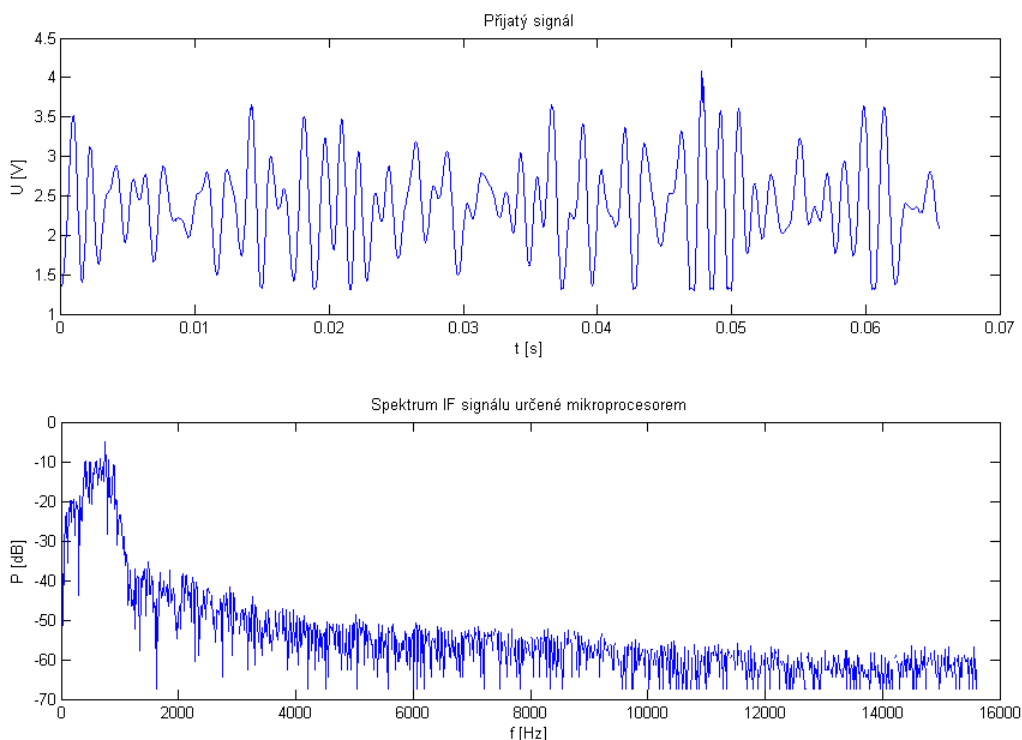


Obrázek 2.25 – Spektrum IF signálu s vykresleným prahem detekce a chybně provedeným odhadem rychlosti.

Obrázek 2.25 ukazuje spektrum IF signálu určeného z odběhu sejmutého za špatných povětrnostních podmínek, tedy deště a mokré vozovky. Ve spektru IF signálu lze na první pohled spatřit zvýšený obsah šumu, jehož špičkové hodnoty převyšují špičku zapříčiněnou hlavním odrazem od povrchu, po kterém se pohyboval automobil. Tvzení, že se jedná o šum lze odůvodnit tím, že je šum shodně rozložen okolo nulové frekvence. Díky použitému algoritmu určení radiální rychlosti pak dochází k chybnému určení této rychlosti, protože algoritmus se zacítil na maximum šumových hodnot, ze kterých se určil práh pro detekci. Pokud by byly použity složitější algoritmy pro odhalení cíle a následně provedeno komplexní zpracování výsledků, bylo by možné i přes hodnoty šumu určit rychlost automobilu. Protože by pravděpodobně byla indikováno větší množství cílů, kde by ke každému z nich byla přiřazena příslušná radiální rychlost.

2.8.3 Rozbor naměřených dat pomocí senzoru CDM324

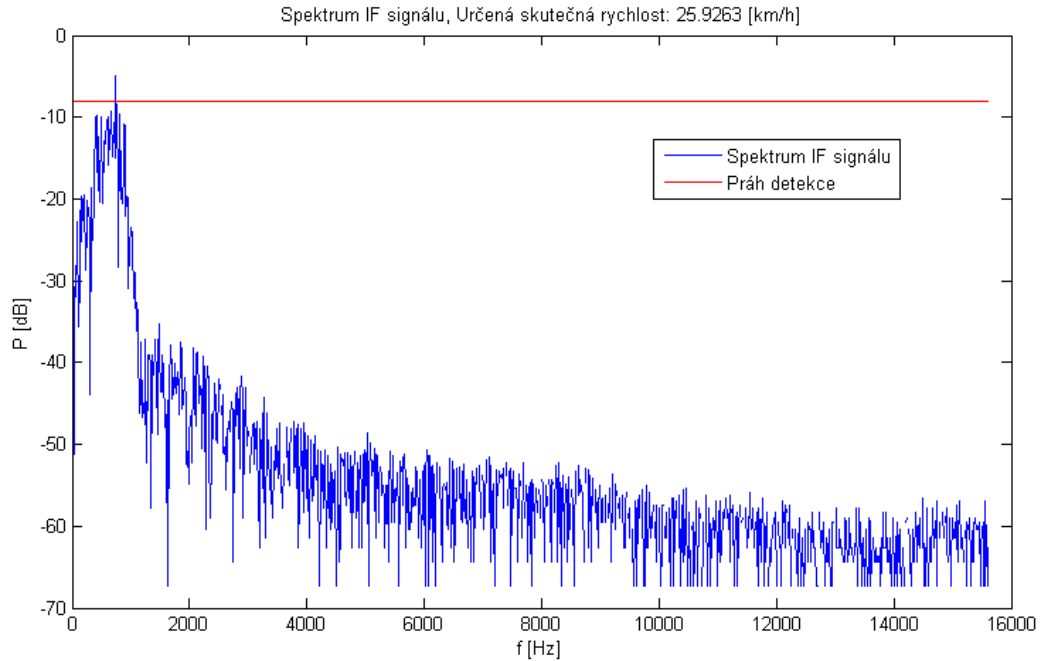
Podobně jako u senzoru K_MC3 byly i pro senzor CDM324 posílány do počítače kompletní data obsahující IF signál a FFT určenou z tohoto signálu. Senzor CDM324 má pouze jeden výstup, a to složku I, v důsledku čehož není možné určit kladnou a zápornou frekvenci. Senzor CDM324 byl také umístěn na tažném zařízení automobilu s úhlem natočení od vodorovného směru $\alpha = 60^\circ$.



Obrázek 2.26 – Grafické zobrazení jednoho odběhu dat.

Obrázek 2.26 ukazuje typické průběhy IF signálu a jednostranného spektra tohoto IF signálu. Pouhým pohledem na průběh samotného IF signálu je patrné, že je složen z více frekvenčních složek, což je potvrzeno průběhem spektra. Ve spektru IF signálu je patrná hlavní špička, která je dosti široká a odpovídá hlavnímu odrazu od povrchu zarovnanému na Dopplerovu frekvenci odpovídající dané vzájemné radiální rychlosti a povrchu. Spektrum tohoto signálu také obsahuje mnohem více šumu, jehož úroveň je dosti vysoká. Data naměřená pomocí senzoru CDM324 pak lze nalézt v příloze 8 na CD.

Spektrum IF signálu s určeným prahem detekce ukazuje obrázek 2.27. Stejně jako v případě senzoru K_MC3 je i zde určena střední hodnota ze všech frekvenčních složek, které mají amplitudu vyšší než práh detekce. Z této frekvence se následně určí radiální rychlost a z ní pak skutečná rychlost jedoucího automobilu. Stejně jako v případě senzoru K_MC3 byla i zde porovnávána rychlost určená mikroprocesorem s rychlostí zobrazenou tachometrem automobilu. V případě senzoru CDM324 však byly zjištěny větší výkyvy než v případě senzoru K_MC3.



Obrázek 2.27 – Spektrum IF signálu s vykresleným prahem detekce

2.8.4 Vzájemné porovnání obou senzorů

Oba použité radarové senzory CDM324 a K_MC3 mají odlišné vlastnosti, a proto jsou data získaná pomocí jednotlivých senzorů rozdílná.

Hlavní rozdíly v obou senzorech lze sledovat například ve směrovosti přijímacích a vysílacích antén obou senzorů. Šířka svazku vysílací a přijímací antény senzoru CDM324 je 35° v elevaci a 80° v azimutu. Šířka svazku přijímací a vysílací antény senzoru K_MC3 je 7° v elevaci a 25° v azimutu. Senzor K_MC3 pak navíc oproti senzoru CDM324 obsahuje zesilovače integrované přímo na DPS senzoru.

Porovnáním typických průběhů spekter IF signálů získaných jednotlivými senzory, zobrazených na obrázcích 2.24 a 2.27 lze sledovat podstatné rozdíly. V obou spektrech lze sledovat špičky odpovídající hlavnímu odrazu od povrchu, tyto špičky jsou ale různě široké. To je zapříčiněno právě vlastnostmi přijímacích a vysílacích antén jednotlivých senzorů. S vyšší šířkou svazku jednotlivých přijímacích a vysílacích antén totiž dochází k větší modulaci přijímaného signálu. Což je zapříčiněno tím, že vysílací anténa radarového senzoru ozáří na povrchu značně velkou plochu, od které se odráží větší množství energie, která je pak přijímána přijímací anténou s různými zisky, podle směru odkud přichází. Toto je také důvod, proč je hlavní špička ve spektru IF signálu získaného senzorem CDM324 podstatně širší než ve spektru IF signálu získaného senzorem K_MC3. Tento fakt tak do jisté míry znesnadňuje odhad

rychlosti objektu, na kterém je radarový senzor upevněn. Což je také důvod, proč bylo dosaženo nižší přesnosti odhadu rychlosti právě se senzorem CDM324. Je třeba také podotknout, že určitý vliv na obě spektra IF signálů má jev nazývaný prosakování spektra. Dochází zde totiž k modulaci spektra IF signálu spektrem obdélníkového průběhu ve tvaru $\frac{\sin(x)}{x}$. Tento jev je možné do určité míry potlačit použitím vhodné váhovací funkce, též nazývané oknové funkce. Oknovací funkce však nebylo využito.

Další rozdíly mezi typickými průběhy spekter signálu podle obrázků 2.24 a 2.27 lze sledovat v obsahu šumu. Spektrum IF signálu získaného senzorem CDM324 jej obsahuje podstatně více s vyšší úrovní. Příčinou je nejspíše kombinace směrovosti vysílací a přijímací antény a absence nízko šumových zesilovačů přímo na DPS senzoru CDM324. Sensory byly totiž s AAF propojeny pomocí propojovacího kabelu dlouhého několik desítek centimetrů, a tak zde mohlo docházet k indukci šumu. Díky vysokému zisku AAF byl pravděpodobně tento šum zesílen spolu s IF signálem a šumem přidaným samotnými zesilovacími obvody. Senzor K_MC3 obsahuje dvojici zesilovačů integrovaných přímo na DPS senzoru, což je z hlediska šumových vlastností lepší variantou, protože následný AAF nemusel mít tak velký zisk.

Celkově lze tedy říci, že senzor K_MC3 je podstatně lepší pro daný účel měření rychlosti. Celkově je lépe zpracovaný, umožňuje detekovat kladné a záporné rychlosti, má integrovány také lepší plošné antény a obsahuje zesilovače integrované přímo na DPS. Díky těmto faktům lze s tímto senzorem měřit rychlost podstatně přesněji než se senzorem CDM324. Je však nutné podotknout, že tomu odpovídá i jeho cena. Senzor CDM324 pak zaostává ve všech ohledech za senzorem K_MC3. Senzor CDM 324 totiž neobsahuje žádné zesilovací obvody, neumožňuje detekovat kladné a záporné rychlosti a jeho antény nejsou příliš směrové. Cena tohoto senzoru je však velice nízká, a tak se tento senzor hodí spíše například k detekci pohybu osob, díky širšímu svazku antén dovoluje pozorovat zároveň větší část prostoru. Senzor CDM324 lze také využít k měření rychlosti, ale jen v určitých mezích.

Závěr

Cílem této bakalářské práce bylo navrhnout řetězec zpracování mezifrekvenčního signálu z Dopplerovského radarového senzoru pro rozsah radiálních rychlostí 1 m/s až 50 m/s, s rozlišením 0,1 m/s. Navrhovaný řetězec měl být složen z antialiasingového filtru a mikroprocesoru, jenž měl provést základní zpracování navzorkovaného IF signálu pomocí FFT a také odhad rychlosti objektu.

K získání IF signálu byly použity dva radarové senzory, které poskytovaly IF signály s různými amplitudami. Protože byly amplitudy obou signálů velice rozdílné, byly navrženy dva aktivní antialiasingové filtry. Při návrhu obou filtrů bylo třeba řešit několik problémů s nestabilitou celkového zapojení filtrů a u filtru pro senzor CDM324 bylo nutné vyřešit ještě problém s nastavením horní mezní frekvence. Problémy s nestabilitou filtrů byly úspěšně vyřešeny a naměřené parametry dvojice filtrů pro senzor K_MC3 téměř odpovídají parametrům požadovaným. U filtru pro senzor CDM324 byla naměřena vyšší horní mezní frekvence. Výsledná vyšší mezní frekvence filtru však nemá vliv na další zpracování signálu, protože byla volena dostatečně velká vzorkovací frekvence A/D převodníků.

Při návrhu antialiasingových filtrů byl brán v potaz fakt, že radarový senzor bude namířen pouze na asfaltový povrch, který odráží elektromagnetické vlnění v malé míře. Pokud by byl radarový senzor namířen na povrch s vyšší odrazivostí, mohlo by dojít ke zkreslení IF signálu díky vysokému zisku antialiasingových filtrů a následnému chybnému odhadu rychlosti. Vzniklý problém by bylo možné vyřešit pomocí antialiasingového filtru s proměnným ziskem, kde by byl zisk řízen podle aktuálních podmínek.

Zesílený a vyfiltrovaný IF signál je pak převáděn na signál číslicový, jehož spektrum je určováno pomocí algoritmu DIT naprogramovaného autorem práce. Naprogramovaný algoritmus byl porovnán s profesionálně vytvořenou knihovnou CMSIS DSP z hlediska času výpočtu a také z hlediska obsazení programové a datové paměti. Bylo zjištěno, že výpočet spektra pomocí algoritmu DIT naprogramovaného autorem je vždy pomalejší vůči knihovně CMSIS DSP. Příčinou je vysoká optimalizace kódu knihovny CMSIS DSP plynoucí z dokonalé znalosti daného mikroprocesoru a jeho instrukční sady. Algoritmus DIT je však úspornější z hlediska obsazení programové paměti.

Z vypočteného spektra IF signálu je prováděn odhad rychlosti objektu, na kterém je umístěn radarový senzor. k odhadu rychlosti byl použit jednoduchý algoritmus, který předpokládá existenci pouze jednoho cíle v dosahu senzoru. Pakliže je v dosahu více objektů s různou

radiální rychlostí, není odhad rychlosti proveden správně. Pokud je radarový senzor umístěn na objektu a namířen na povrch, po kterém se tento objekt pohybuje, pak je požadavek jednoho cíle v dosahu senzoru splněn téměř vždy, což bylo ověřeno analýzou naměřených dat.

Výsledně sestavený řetězec zpracování IF signálu byl ověřen pomocí několika experimentů se senzory CDM324 a K_MC3. Vzájemným porovnáním naměřených dat získaných oběma senzory pak lze dojít k závěru, že senzor CDM324 není vhodný k měření rychlosti. Spektrum IF signálu, získaného pomocí senzoru CDM324, je totiž velice ovlivněno šířkou svazku vysílací a přijímací antény senzoru, což znesnadňuje za daných podmínek odhad rychlosti. Senzor K_MC3 je naopak velice vhodný pro měření rychlosti, protože spektrum IF signálu získaného pomocí tohoto senzoru vykazuje mnohem menší modulaci šířkou svazku vysílací a přijímací antény, což umožnilo provést přesnější odhady rychlosti objektu. Odhady rychlosti byly vždy porovnávány s rychlostí automobilu zobrazenou na tachometru v tomto automobilu. Tachometr automobilu vykazuje vždy jistou toleranci, a tak nelze říci, jak přesně byl daný odhad proveden. Ke zjištění dosažené přesnosti by bylo třeba provést náročnější experiment, což nebylo možné v dané době zařídit. Správnějších odhadů však bylo dosahováno vždy se senzorem K_MC3, protože odhad rychlosti provedený pomocí tohoto senzoru ležel vždy do 1,5 km/h od rychlosti určené tachometrem automobilu. U senzoru CDM324 byly vždy zaznamenány podstatnější rozdíly, někdy i v řádu desítek km/h.

Použitá literatura

- [1] BEZOUŠEK, Pavel a Pavel ŠEDIVÝ. *Radarová technika*. Praha: Vydavatelství ČVUT, 2004. ISBN 978-800-1030-363.
- [2] *Radartutorial* [online]. Neubrandenburg: Christian Wolff, 1998 [cit. 2017-05-08]. Dostupné z: <http://www.radartutorial.eu/>
- [3] HALLIDAY, David, Robert RESNICK a Jearl WALKER. *Fyzika: vysokoškolská učebnice obecné fyziky*. Brno: VUTIUM, 2000. Překlady vysokoškolských učebnic. ISBN 80-214-1869-9.
- [4] ŘEZNÍČEK, Michal a Pavel BEZOUŠEK. *Commercial CW Doppler Radar Design and Application*. Pardubice, 2017.
- [5] ŘEZNÍČEK, Michal. *Zpracování signálu z Dopplerovského CW senzoru*. Pardubice, 2016.
- [6] NĚMEC, Zdeněk. *Výukové materiály předmětu ISTE*. Pardubice, 2016. Dostupné také z: www.stag.upce.cz
- [7] BRTNÍK, Bohumil. *Lineární elektrické obvody*. Pardubice, 2016. Dostupné také z: www.stag.upce.cz
- [8] VEJRAŽKA, František. *Signály a soustavy*. Vyd. 4. Praha: České vysoké učení technické, 1996. ISBN 80-010-0450-3.
- [9] UHLÍŘ, Jan a Pavel SOVKA. *Číslicové zpracování signálů*. Vyd. 2. přeprac. Praha: Vydavatelství ČVUT, 2002. ISBN 80-010-2613-2.
- [10] OPPENHEIM, Alan V., Ronald W. SCHAFER a John R. BUCK. *Discrete-time signal processing*. 2nd ed. Upper Saddle River, N.J.: Prentice Hall, 1999. ISBN 01-375-4920-2.
- [11] BRIGHAM, E. Oran. *The fast Fourier transform*. Englewood Cliffs, N.J.: Prentice-Hall, 1974. ISBN 01-330-7496-X.

- [12] Q (number format). In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2017-05-08]. Dostupné z: [https://en.wikipedia.org/wiki/Q_\(number_format\)](https://en.wikipedia.org/wiki/Q_(number_format))
- [13] *Data Sheet IPM-165*. 8.5. Donnersdorf: InnoSenT, 2014. Dostupné také z: http://www.innosent.de/fileadmin/media/dokumente/DATASHEETS_2016/Datenblatt_IPM-165_V8.5.pdf
- [14] *K-MC3 RADAR TRANSCEIVER: Datasheet*. 2.0. St.Gallen: RFbeam, 2011. Dostupné také z: https://www.rfbeam.ch/files/products/17/downloads/Datasheet_K-MC3.pdf
- [15] *PSoC® 5LP: CY8C58LP Family Datasheet*. L. San Jose: Cypress, 2017. 001-84932. Dostupné také z: <http://www.cypress.com/file/45906/download>
- [16] JUNG, Walter G. *Op Amp applications handbook*. Burlington, MA: Newnes, 2006. ISBN 07-506-7844-5.
- [17] MANCINI, Ron. a Bruce CARTER. *Op amps for everyone*. 3rd ed. Boston: Newnes, 2009. ISBN 978-185-6175-050.
- [18] BRTNÍK, Bohumil. *Analogové soustavy*. Praha: BEN - technická literatura, 2013. ISBN 978-80-7300-490-3.
- [19] *CMSIS: Cortex Microcontroller Software Interface Standard* [online]. ARM, 2017 [cit. 2017-05-08]. Dostupné z: <http://www.keil.com/pack/doc/CMSIS/General/html/index.html>

Příloha A – seznam příloh na CD

- Příloha_1: Obsahuje samostatná schémata filtrů pro CDM324 a K_MC3 ve formátu pdf.
- Příloha_2: Obsahuje simulační schémata pro filtry senzorů CDM324 a K_MC3 zakreslené v programu LTspice. Výstupem simulačních souborů jsou charakteristiky „CDM_SIMUL“ a „MC3_SIMUL“. Tato příloha také obsahuje naměřené charakteristiky filtrů označené jako „CDM324_ZMERENO“ a „MC3_ZMERENO“ ve formátu pdf.
- Příloha_3: Obsahuje skript do matlabu naprogramovaného algoritmu DIT s názvem „algoritmusDIT“. Dále jsou zde přiloženy skripty pro generování testovacích dat do jazyka C s názvem „genTestDat“ a „genTestDat_CMSIS_DSP“. Také je zde přiložen skript pro generování tabulek do jazyka C s názvem „genTabulek_DIT_C“.
- Příloha_4: Obsahuje soubory s naprogramovaným algoritmem DIT pro výpočet FT v jazyce C. Jsou zde také vloženy skripty pro generování testovacích dat a potřebných tabulek z přílohy číslo 3.
- Příloha_5: Obsahuje projekty pro měření časové a paměťové náročnosti výpočtu FT pomocí algoritmu DIT a knihovny CMSIS_DSP.
- Příloha_6: Tato příloha obsahuje schéma výsledné DPS a návrh výsledné DPS. Oba soubory jsou vytvořené programem EAGLE.
- Příloha_7: Obsahuje projekt se zdrojovými soubory programu pro výslednou DPS, vytvořený ve vývojovém prostředí PSoC Creator 4.0, zabalený v archivu zip.
- Příloha_8: Obsahuje data uložená v matlabu naměřená se senzory CDM324 a K_MC3 rozdělená do zvláštních složek. V každé složce lze nalézt skripty do matlabu určené k vykreslení celé sady dat, nebo jen spektra s prahem detekce. Je zde také přiložen skript pro příjem dat z DPS do matlabu s názvem „prijem_z_DPS“.