



České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická
Katedra měření

System pro akustickou detekci a lokalizaci založený na dvou mikrofonech

Matěj Trnka

Vedoucí: Ing. Jakub Svatoš, Ph.D.
Studijní program: Elektronika a komunikace
Květen 2023

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Trnka** Jméno: **Matěj** Osobní číslo: **499145**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávající katedra/ústav: **Katedra elektromagnetického pole**
Studijní program: **Elektronika a komunikace**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Systém pro akustickou detekci a lokalizaci založený na dvou mikrofonech

Název bakalářské práce anglicky:

System for Acoustic Detection and Localization Based on Two Microphones

Pokyny pro vypracování:

Nastudujte problematiku detekce akustických impulzních událostí a navrhnete systém, obsahující mikrokontrolér a dva MEMS mikrofony, umožňující detekci a přibližnou lokalizaci směru zdroje události. Ověřte funkčnost navrženého systému vhodnými experimenty. Součástí práce bude také grafické rozhraní zobrazující výsledek detekce, např. formou polárního grafu.

Seznam doporučené literatury:

- [1] Sensor Fusion, Sensitivity Analysis and Calibration in Shooter Localization Systems, Akman, C., at al., Sensors and Actuators A: Physical Volume 271, 2018, Pages 66-75.
- [2] Evaluation of Gunshot Detection Algorithms, Chacón-Rodríguez, A., at al., IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, vol. 58, No. 2, 2011

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Jakub Svatoš, Ph.D. katedra měření FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **02.02.2023**

Termín odevzdání bakalářské práce: **26.05.2023**

Platnost zadání bakalářské práce: **22.09.2024**

Ing. Jakub Svatoš, Ph.D.
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Mgr. Petr Páta, Ph.D.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

Datum převzetí zadání

Podpis studenta

Poděkování

Rád bych poděkoval svému vedoucímu Ing. Jakubu Svatošovi, Ph.D. za cenné rady, připomínky a vstřícný přístup během realizace této práce.

Prohlášení

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

V Praze, 26. května 2023

Abstrakt

Tato práce se zabývá návrhem demonstračního zařízení pro detekci a lokalizaci impulzních akustických událostí. Je zde zvolen přístup spočívající v použití dvou MEMS mikrofonů SPH0645LM4H a mikrokontroléru ESP32. V práci je uvedena problematika zpracování dat z mikrofonů, detekce akustické události, určení zpoždění signálů a výpočet odhadu úhlu, pod kterým akustická událost dorazila. Dále je zde popsán návrh finálního zařízení a způsob vizualizace naměřených hodnot v reálném čase pomocí webového rozhraní. Na závěr je uvedena také kapitola zabývající se ověřením funkčnosti.

Klíčová slova: akustická detekce, akustická lokalizace, TDoA, ESP32, SPH0645LM4H

Vedoucí: Ing. Jakub Svatoš, Ph.D.

Abstract

This work focuses on the development of a demonstration device for detecting and localizing acoustic impulse events. The chosen approach involves using two MEMS microphones SPH0645LM4H and the ESP32 microcontroller. The thesis addresses the issues related to data processing from the microphones, acoustic event detection, estimation of signal delays, and calculation of the estimated angle at which the acoustic event arrived. Furthermore, it presents the design of the final device, along with the real-time visualization of measured values through a web interface. Lastly, a chapter dedicated to verifying the device's functionality is included.

Keywords: acoustic detection, acoustic localization, TDoA, ESP32, SPH0645LM4H

Title translation: System for Acoustic Detection and Localization Based on Two Microphones

Obsah

Použité zkratky	1		
1 Úvod	3		
2 Cíle práce	4		
3 Teoretický rozbor	5		
3.1 Zvuk	5		
3.1.1 Rychlost zvuku	6		
3.1.2 Odraz zvuku na rozhraní	6		
3.2 Akustická událost	7		
3.2.1 Impulzní akustická událost	7		
3.2.2 Detekce impulzních akustických událostí	7		
3.3 Akustická lokalizace	8		
3.3.1 Time Difference of Arrival	9		
3.3.2 Nejednoznačnost určení zdroje signálu způsobená odrazy	9		
3.4 Signálový model systému	10		
3.5 Nalezení časového zpoždění mezi signály	11		
3.5.1 Vzájemná korelační funkce	11		
3.5.2 Generalized Cross-Correlation with Phase Transform	13		
3.5.3 Steered-Response Power Phase Transform	14		
3.6 Výpočet úhlu z časového zpoždění	15		
3.6.1 Odvození	15		
3.7 Akustický detekční systém	17		
3.7.1 MEMS mikrofon	18		
3.8 Mikrokontrolér ESP32	20		
3.8.1 Volba vývojového prostředí	21		
3.8.2 FreeRTOS	21		
3.9 Mikrofon SPH0645LM4H	21		
3.9.1 Sběrnice I ² S	22		
4 Praktická část	24		
4.1 Postup zpracování dat	24		
4.2 Volba vzorkovací frekvence	24		
4.3 Vyčítání mikrofonů SPH0645LM4H mikrokontrolérem ESP32	24		
4.3.1 Filtrace DC složky	26		
4.4 Implementace zpracování dat v ESP32	27		
4.5 Detekce impulzní akustické události	28		
4.5.1 Potlačení akustických událostí způsobených odrazy	29		
4.6 Nalezení zpoždění dvou signálů	29		
4.6.1 Vzájemná korelační funkce	29		
4.7 Výpočet úhlu z časového zpoždění	32		
4.7.1 Maximální dosažitelný úhel vzhledem k vzorkovací frekvenci a vzdálenosti mikrofonů	32		
4.7.2 Chyba odhadu úhlu způsobená vzorkováním	33		
4.7.3 Nejistota odhadu úhlu	34		
5 Webové rozhraní	36		
5.1 ESP32 webový server	36		
5.2 Komunikace mezi ESP32 a zařízením	36		
5.2.1 HTTP	36		
5.2.2 Web Sockets	37		
5.2.3 Server-Sent Events	37		
5.3 Klientská webová stránka	37		
5.3.1 Zmenšení velikosti webové stránky	38		
6 Návrh HW	40		
6.1 Zapojení	40		
6.1.1 Mikrokontrolér ESP32	40		
6.1.2 Mikrofony SPH0645LM4H	41		
6.1.3 Napájení	41		
6.2 Plošný spoj	42		
6.3 Praktické zhotovení	42		
7 Ověření funkčnosti	44		
7.1 Zhodnocení výsledků	45		
8 Závěr	46		
Seznam použité literatury	47		
A Zdrojový kód	51		
B Schéma zapojení	52		
C Plošný spoj	54		


Obrázky

2.1 Cíl práce.	4	4.3 Frekvenční charakteristika DC filtru pro $R = 0,995$	28
3.1 Blízké a vzdálené zvukové pole. Převzato z [3].	5	4.4 Vychýlený odhad vzájemné korelační funkce pro dva stejné, navzájem zpožděné signály s viditelným systematickým tlumením.	30
3.2 Zvuk v blízkém poli v bodě A a ve vzdáleném poli v bodě B , kde má kulová vlnoplocha natolik malé zakřivení, že ji lze aproximovat rovinou. Převzato z [3].	6	4.5 Nevychýlený odhad vzájemné korelační funkce pro dva stejné, navzájem zpožděné signály s viditelnou náhodnou chybou na okrajích.	31
3.3 Volné zvukové pole (žádné odrazy) a difúzní zvukové pole (pouze odrazy). Převzato z [3].	7	4.6 DoA – trigonometrické odvození.	33
3.4 Akustická lokalizace z roku 1927. Převzato z [7].	8	4.7 Závislost úhlu θ na vzorkovém posunu mezi signály n pro $f_s = 44,1$ kHz a rozteč $d = 0,2$ m.	34
3.5 Ukázka rozdílné délky dráhy akustického vlnění.	9	4.8 Maximální chyba způsobená vzorkováním pro $f_s = 44,1$ kHz a rozteč $d = 0,2$ m.	35
3.6 Vzájemná korelační funkce a interpolovaná parabola se zvýrazněnými body maxim. Převzato z [14].	12	5.1 Webové rozhraní.	37
3.7 DoA – exaktní odvození.	15	6.1 Doporučené zapojení mikrokontroléru ESP32-WROOM-32E. Převzato z [39].	40
3.8 Vlastnosti a významné body hyperboly. Převzato z [17].	16	6.2 Doporučené zapojení dvojice mikrofonů SPH0645LM4H-B. Převzato z [28].	41
3.9 Možné pozice zdroje akustického signálu pro různé hodnoty zpoždění τ a rozteč $d = 0,2$ m s vynecháním záporné části hyperboly.	18	6.3 Snímek 3D náhledu navrženého zařízení.	42
3.10 Blokové schéma systému.	19	6.4 Zhotovené zařízení.	43
3.11 Stavba kapacitního MEMS mikrofonu. Převzato z [20].	19	7.1 Testovací prototyp na veletrhu AMPER 2023.	44
3.12 Řez pouzdrem MEMS mikrofonu. Převzato z [22].	20		
3.13 Čip mikrofonu SPH0645LM4H. Převzato z [28].	21		
3.14 Závislost citlivosti mikrofonu SPH0645LM4H na frekvenci zvuku pro různé hodinové frekvence. Převzato z [28].	22		
3.15 Ovládací zařízení master v roli přijímače. Převzato z [29].	22		
3.16 Základní časování sběrnice I ² S. Převzato z [29].	22		
4.1 Blokový diagram vyhodnocení.	25		
4.2 Poloha nulového bodu a pólu v Z-rovině pro $R = 0,995$	27		



Použité zkratky

AoA	Angle of Arrival. 9
ASIC	Application-Specific Integrated Circuit. 20
CCF	Cross-Correlation Function. 11–13
CSS	Cascading Style Sheets. 36, 38
DMA	Discrete Memory Access. 20, 25, 27
DoA	Direction of Arrival. 9, 46
DPS	Deska Plošných Spojů. 19, 20, 42
FFT	Fast Fourier transform. 14, 31
GCC	Generalized Cross-Correlation. 13–15, 46
GCC-PHAT	Generalized Cross-Correlation with PHAse Transform. 11, 13, 15
HTML	Hypertext Markup Language. 36, 38
HTTP	Hypertext Transfer Protocol. 36, 37, 39
I²S	Inter-IC Sound. 18, 20–22, 24–26, 34, 46
IIR	Infinite Impulse Response. 26
JS	JavaScript. 36, 38
LMS	Least Mean Squares. 11
LSB	Least Significant Bit. 23
MEMS	Micro-Electro-Mechanical Systems. 18–21
MSB	Most Significant Bit. 21, 23
PDM	Pulse-Density Modulation. 18, 20
PHAT	PHAse Transform. 14, 15



Sass	Syntactically Awesome Style Sheets. 38
SMD	Surface Mounted Device. 21
SMP	Symmetric Multiprocessing. 21
SNR	Signal to Noise Ratio. 12-14
SPI	Serial Peripheral Interface. 36
SPIFFS	SPI Flash File Storage. 36 , 38 , 39
SRP	Steered Response Power. 15
SRP-PHAT	Steered-Response Power PHAse Transform. 11
SSE	Server-Sent Events. 36 , 37
TCP	Transmission Control Protocol. 36
TDE	Time Delay Estimation. 12 , 13
TDoA	Time Difference of Arrival. 9
UDP	User Datagram Protocol. 36
WS	Web Sockets. 36 , 37

Kapitola 1

Úvod

Akustická detekce a lokalizace představuje klíčovou disciplínu v oblasti zpracování zvuku s širokým spektrem aplikací. Její využití se v dnešní době rozšířilo do různých oblastí, jako je bezpečnost, robotika, telekonferenční systémy, virtuální realita nebo medicína. Tato technologie nabízí mnoho možností a přínos v každé z těchto oblastí.

V oblasti bezpečnosti může být akustická detekce a lokalizace klíčová pro různé úkoly, jako je detekce a lokalizace impulzních akustických událostí, kterými mohou být výstřely či exploze. Schopnost spolehlivě detekovat a přesně lokalizovat zdroj události je v tomto případě důležitá pro rychlou reakci a zajištění bezpečnosti [1]. Dalším příkladem je detekce a lokalizace bezpilotních letounů, kde je nezbytné rychle identifikovat a sledovat jejich polohu a pohyb [2].

Tato práce se zabývá návrhem jednoduchého systému akustické detekce a lokalizace impulzních akustických událostí (např. tlesknutí nebo úder předmětu) pro demonstrační účely. Tento systém bude založen na mikrokontroléru ESP32 a mikrofonech SPH0645LM4H.

Cíle práce budou popsány v kapitole 2. Kapitola 3 bude věnována teoretické části práce, budou v ní představeny metody detekce a lokalizace akustických událostí a blíže popsány požadavky na akustický detekční systém. Dále zde bude podrobně popsán mikrokontrolér ESP32 a mikrofony SPH0645LM4H.

V kapitole 4 bude prezentována praktická implementace závěrů z teoretické části a bude zde navrhnout a realizován systém pro akustickou detekci a lokalizaci impulzních akustických událostí. V kapitole 5 bude ukázán vývoj webového rozhraní, které slouží k zobrazení směru detekovaných akustických událostí. Kapitola 6 se zaměří na návrh finálního hardwaru systému.

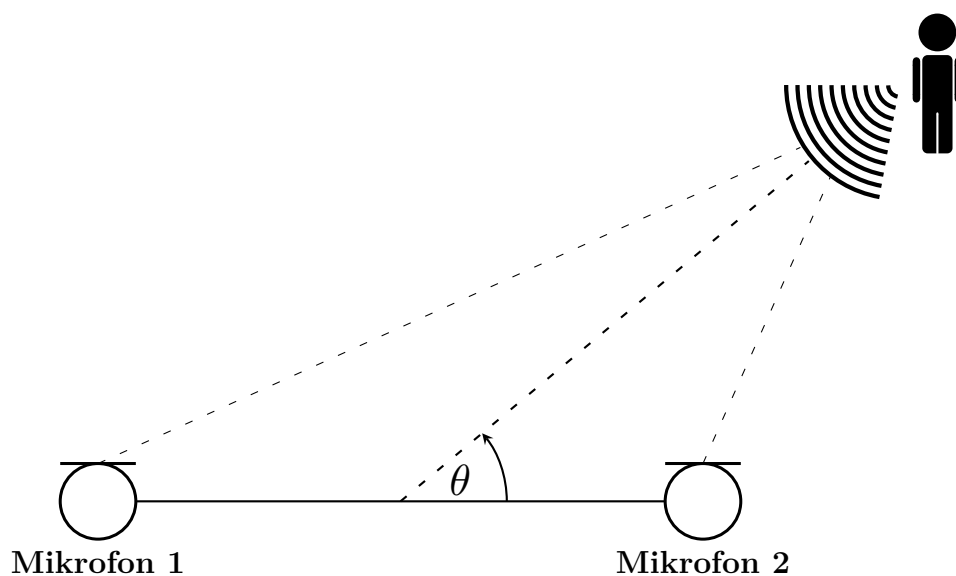
Na závěr v kapitole 7 bude provedeno ověření funkčnosti navrženého systému a zhodnocení dosažených výsledků. Bakalářská práce si klade za cíl přispět k lepšímu porozumění a praktickému využití akustické detekce a lokalizace impulzních akustických událostí.

Kapitola 2

Cíle práce

Cílem této práce je navrhnout jednoduché demonstrační zařízení, které bude schopno detekovat impulzní akustickou událost (např. tlesknutí), spočítat odhad úhlu, pod kterým tato událost k zařízení dorazila (viz obr. 2.1), a tento úhel poté zobrazit pomocí webového rozhraní. Zároveň je žádoucí, aby se jednalo o finančně nenákladné zařízení s přijatelnými rozměry a proudovou spotřebou.

K uskutečnění tohoto cíle bude využito mikrokontroléru a dvou MEMS mikrofonů. Mikrokontrolér bude provádět zpracování dat z mikrofonů a následně odesílat informaci o velikosti úhlu do webového rozhraní klientského zařízení, kterým může být např. telefon či PC.



Obrázek 2.1: Cíl práce.

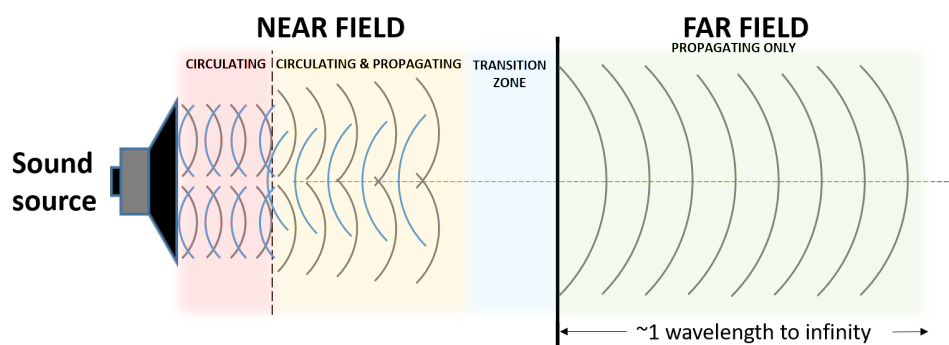
Kapitola 3

Teoretický rozbor

3.1 Zvuk

Zvuk je mechanické vlnění, šířící se v látkovém prostředí (pevném, kapalném nebo plynném). Jedná se o podélné (longitudinální) vlnění, kdy je amplituda kmitů rovnoběžná se směrem šíření vlnění. V případě pevných látek může být navíc příčné (transverzální), kdy je amplituda kmitů kolmá na směr šíření.

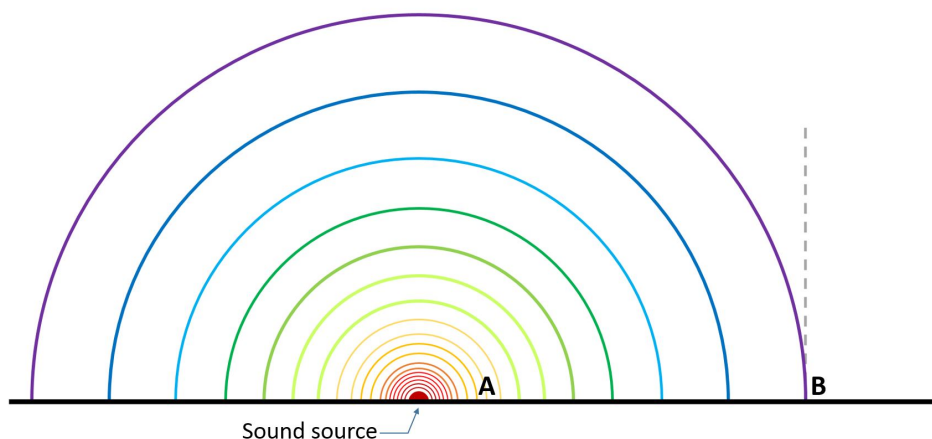
Přibližný interval frekvencí, které je schopen člověk vnímat, se pohybuje od 16 Hz do 20 kHz.



Obrázek 3.1: Blízké a vzdálené zvukové pole. Převzato z [3].

Důležitými pojmy jsou tzv. blízké a vzdálené zvukové pole. V blízkém poli, které je přibližně do vzdálenosti jedné vlnové délky, se zvuk chová komplexněji a kmitání částic není nutně ve směru šíření [3] (viz obr. 3.1). Oproti tomu ve vzdáleném poli, které nastává přibližně ve vzdálenosti větší než jedna vlnová délka, se v ideálním případě zvuk šíří kulovou vlnoplochou. Z těchto důvodů bude v této práci uvažován zdroj zvuku ve vzdáleném poli. V případě vzdáleného zvukového pole lze od určité vzdálenosti aproximovat kulovou vlnoplochu pomocí roviny (viz obr. 3.2).

Dalšími důležitými pojmy jsou volné zvukové pole a difúzní zvukové pole, jak je znázorněno na obrázku 3.3. Ve volném poli nedochází k žádným odrazům. Oproti tomu v difúzním poli se vyskytuje mnoho několikanásobných odrazů, které znemožňují posluchači určit přesný zdroj akustického signálu. V případě této práce je žádoucí volné zvukové pole, avšak není vždy možné dosáhnout



Obrázek 3.2: Zvuk v blízkém poli v bodě **A** a ve vzdáleném poli v bodě **B**, kde má kulová vlnoplocha natolik malé zakřivení, že ji lze aproximovat rovinou. Převzato z [3].

ideálních podmínek a musíme počítat s přítomností odrazů, které mohou negativně ovlivnit přesnost vyhodnocení.

■ 3.1.1 Rychlost zvuku

Rychlost zvuku udává, jakou rychlostí se zvukové vlny šíří v přenosovém médiu. Pro naše účely je důležitá rychlost zvuku ve vzduchu, která je závislá na atmosférických podmínkách, především pak teplotě vzduchu.

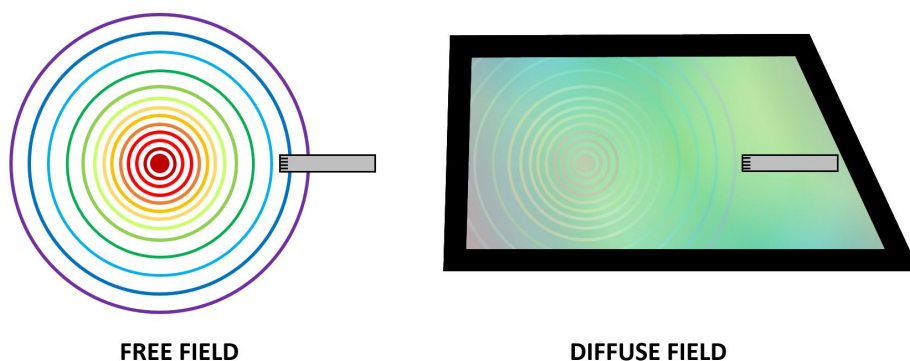
V dalších výpočtech budeme uvažovat rychlost zvuku 343 m/s, která je stanovena pro vzduch o teplotě 20 °C.

■ 3.1.2 Odraz zvuku na rozhraní

Odraz zvuku na rozhraní mezi dvěma prostředími je jev, který nastává, když zvukové vlny narazí na překážku nebo změnu akustických vlastností prostředí. Při odrazu dochází k částečnému nebo úplnému odražení zvukových vln zpět do původního prostředí.

Odraz zvuku na rozhraní může mít za následek zesílení vlnění a pokud se zvuk vrátí zpět do blízkosti zdroje zvuku, vzniká ozvěna. Tento jev může způsobit prodloužení doby dozvuku. Dalšími důsledky mohou být změna frekvenčního spektra nebo posun fáze, kdy může dojít k interferencím mezi původními a odraženými vlnami.

V aplikaci akustické lokalizace hrají odrazy zvuku důležitou roli, protože mohou negativně ovlivnit výsledek měření.



Obrázek 3.3: Volné zvukové pole (žádné odrazy) a difúzní zvukové pole (pouze odrazy). Převzato z [3].

3.2 Akustická událost

Akustickou událostí může být jakýkoliv akustický signál, který není součástí zvukového pozadí. Příkladem může být například řečový signál, který je charakterizován počátkem a koncem této události. Počátek a konec události ohraničují interval, v rámci kterého je událost odlišitelná od šumového (nebo jiného) zvukového pozadí. Tato práce je zaměřena především na impulzní akustické události, které se vyznačují vysokou hodnotou energie v krátkém časovém úseku.

3.2.1 Impulzní akustická událost

Impulzní akustická událost je krátký a intenzivní zvukový signál, který se vyznačuje vysokým špičkovým akustickým tlakem a rychlou změnou akustického tlaku v čase. Tento druh události je obvykle charakterizován krátkou dobou trvání, rychlým nástupem a rychlým poklesem intenzity zvuku.

Nejběžnějšími zdroji impulzních akustických událostí mohou být tlesknutí, bouchnutí dveřmi, úder předmětu o jiný předmět, exploze nebo výstřel z palné zbraně.

Délka a intenzita impulzní akustické události může být ovlivněna příčinou události, vzdáleností od zdroje, prostředím nebo frekvenčním složením akustické události. U běžného tlesknutí se délka trvání pohybuje v jednotkách milisekund a největší množství energie je v rozsahu frekvencí 1–10 kHz pro tlesknutí s plochými dlaněmi a 0,1–2 kHz pro tlesknutí s nastavenými (prohnutými) dlaněmi [4].

3.2.2 Detekce impulzních akustických událostí

Pro detekci impulzních akustických událostí lze využít různých algoritmů. V článku [1] je pro detekci výstřelů využit algoritmus založený na mediánovém

filtru. V článku [5] je využito výpočtu hladiny energie přes zvolené časové (vzorkové) okno a porovnávání pomocí adaptivního thresholdu¹. Dalšími metodami mohou být spektrální analýza signálu nebo metody založené na strojovém učení.

Jelikož se jedná o demonstrační zařízení, nejsou na detekci akustické události kladeny žádné zvláštní požadavky. Stačí tedy, aby bylo zařízení schopno detekovat impulzní akustické události, které se vyznačují vysokou špičkovou hodnotou v krátkém čase. Oproti tomu při praktickém využití této technologie může být požadavkem detekce a lokalizace výstřelů z palné zbraně na veřejných místech, kdy není žádoucí, aby byly detekovány ostatní impulzní akustické události [1], [6]. V části 4.5 bude probrána konkrétní volba algoritmu pro detekci impulzních akustických událostí.

3.3 Akustická lokalizace

Cílem akustické lokalizace je určit polohu zdroje nebo reflektoru zvuku pomocí akustických signálů. Může se jednat o aktivní lokalizaci, kdy je nejdříve vytvořen akustický signál a poté je analyzován vzniklý odraz (např. sonar), nebo pasivní lokalizaci, kdy jsou pouze analyzovány akustické události.



Obrázek 3.4: Akustická lokalizace z roku 1927. Převzato z [7].

S akustickou lokalizací se můžeme setkat každý den. Náš mozek, spolu se sluchovým ústrojím, je schopen tento nelehký úkol provádět bez většího úsilí. Díky dvěma ušima je možné provádět lokalizaci zdroje signálu a k tomu je mimo

¹Thresholding je operace, kdy se hodnota porovnává s danou prahovou hodnotou.

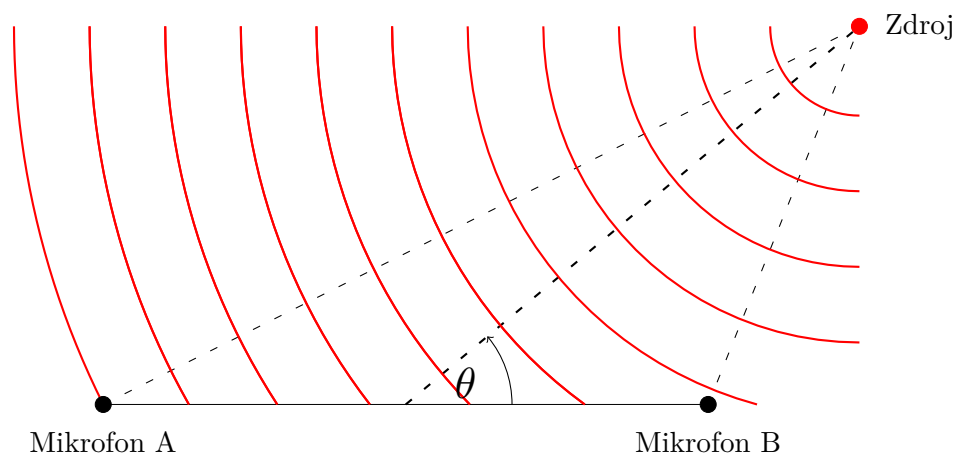
jiné využito i vyhodnocení časového zpoždění zvukového signálu mezi levým a pravým uchem, podobně jako bude uvedeno v této práci. Kromě pasivní akustické lokalizace lze nalézt u některých živočichů i lokalizaci aktivní, např. u netopýrů nebo delfínů.

Snahy o technické využití zvuku k lokalizaci objektů se objevují již na počátku 20. století. Ještě před vznikem radaru byla lokalizace nepřátelských letadel obtížným úkolem. Na obrázku 3.4 můžeme vidět akustickou lokalizaci z roku 1927, jejímž cílem bylo lokalizovat nepřátelská letadla pomocí dvou párů mikrofónů [7].

Při pasivní lokalizaci je nejčastěji využíváno mikrofónových polí, kdy je použita informace o zpoždění (fázovém posunu) mezi signály, zachycenými rozdílnými mikrofóny. Tato práce se bude zabývat především lokalizací založenou na dvojici mikrofónů, kdy je zařízení schopno určit pouze úhel ve vodorovné rovině, pod kterým akustická událost dorazila (azimut).

3.3.1 Time Difference of Arrival

Běžným způsobem určení směru zdroje zvuku, **Direction of Arrival (DoA)** nebo **Angle of Arrival (AoA)**, je tzv. **Time Difference of Arrival (TDoA)**, kdy se zjišťuje časové zpoždění mezi signály zachycenými dvojicí senzorů (mikrofónů). Zpoždění mezi signály vzniká v důsledku rozdílné dráhy, kterou akustické vlnění musí urazit k jednotlivým mikrofónům (viz obr. 3.5). Nalezení zpoždění lze docílit např. pomocí vzájemné korelační funkce, jak bude podrobněji popsáno v sekci 3.5.



Obrázek 3.5: Ukázka rozdílné délky dráhy akustického vlnění.

3.3.2 Nejednoznačnost určení zdroje signálu způsobená odrazy

Při detekci akustických událostí může být nelehkým úkolem vyfiltrování akustických událostí způsobených odrazy v dozvukových prostředích, např. v místnosti, které způsobují nejednoznačnost určení zdroje akustické události.

Tento úkol se ještě zkomplikuje v případě, že zdrojů akustických událostí je více a odrazy jsou několikanásobné.

V případě jednoho zdroje lze očekávat dvě situace, kdy v prvním případě dorazí do zařízení zvuk přímou cestou a následně dorazí zvuk, který se šířil odraženou cestou. Zvuk, který se šířil přímou cestou, musí do zařízení dorazit jako první, protože jeho dráha byla kratší a uvažujeme prostředí s konstantní rychlostí zvuku a bez překážek způsobujících zpoždění. V druhém případě se zvuk přímou cestou k zařízení vůbec nedostane, a není tedy ani možné odhadnout správně úhel dopadu, protože všechny detekované události přicházejí z jiného směru, než v kterém se nachází zdroj akustické události.

Nalezením správné akustické události v grafu vzájemné korelační funkce pomocí využití autokorelační funkce se zabývá článek [8]. Je zde představena metoda *Disambiguation of TDOA estimates in multi-path multi-source environments* (DATEMM), která využívá skutečnosti, že vrcholy vzájemné korelační funkce se objevují v rastru se známými vzdálenostmi. Pro více mikrofonů než dva pak článek přichází s dalšími způsoby, jak určit správné vrcholy vzájemné korelační funkce.

Praktickým potlačením akustických událostí způsobených odrazy (odstraněním nejednoznačností) se poté zaobírá část 4.5.1.

3.4 Signálový model systému

Uvažujme systém s dvěma mikrofony, $m = 1, 2$. Diskrétní signál zachycený mikrofonom m popíšeme pomocí

$$x_m[n] = s[n] * h_m[n] + n_m[n], \quad \text{pro } m = 1, 2, \quad (3.1)$$

kde $s[n]$ je signál od zdroje zvuku, $h_m[n]$ je impulzní odezva m . mikrofону a $n_m[n]$ je šum zachycený m . mikrofonom (uvažujeme, že šum $n_m[n]$ a signál $s[n]$ jsou navzájem nekorelované a šумы $n_m[n]$ jsou mezi sebou nekorelované). Vzhledem k tomu, že nás zajímá signál, který dorazil pouze přímou cestou a ne odrazem, nebudeme odrazy v signálovém modelu systému brát v potaz.

Budeme uvažovat, že mikrofony mají impulzní odezvu popsanou pouze funkcí $h_m[n] = \alpha_m \cdot \delta[n - k_m]$, kde $\alpha_m \in \mathbb{R}$ je konstanta, která mění amplitudu signálu, a $\delta[n - k_m]$ je jednotkový skok ve vzorku k_m , který vyjadřuje, že signál do každého z mikrofonů dorazí v jiném čase (neboli vzorku k_m). Uvažujeme tedy, že mikrofón během přeměny signálu mění pouze jeho amplitudu, a to pomocí škálovacího faktoru α_m . Tímto se rovnice (3.1) zjednoduší na

$$x_m[n] = \alpha_m \cdot s[n - k_m] + n_m[n], \quad \text{pro } m = 1, 2. \quad (3.2)$$

Abychom se zbavili části proměnných, budeme uvažovat, že signál v prvním mikrofónu bude mít stejnou velikost jako signál $s[n]$ (tedy $\alpha_1 = 1$) a že časový posun je nulový ($k_1 = 0$). Tímto dostaneme rovnice pro mikrofony ve tvaru

$$x_1[n] = s[n] + n_1[n], \quad (3.3)$$

$$x_2[n] = \alpha_2 \cdot s[n - k_2] + n_2[n]. \quad (3.4)$$

Po přejmenování proměnných dostaneme

$$x_1[n] = s[n] + n_1[n], \quad (3.5)$$

$$x_2[n] = \alpha \cdot s[n - D] + n_2[n], \quad (3.6)$$

kde D vyjadřuje vzorkový posun mezi signály a časový posun (časové zpoždění) mezi signály je poté dán jako $\tau = D \cdot \frac{1}{f_s}$, kde f_s je vzorkovací frekvence.

3.5 Nalezení časového zpoždění mezi signály

Typický úkol akustické lokalizace je nalézt pozice N akustických zdrojů pomocí M mikrofonů. Tato úloha se ještě komplikuje v případě, že začneme uvažovat odrazy zvuku v dozvukových prostředích (např. běžná místnost). Pro naše účely budeme uvažovat, že máme pouze jeden zdroj akustické události a dva mikrofony.

Způsobů, jak nalézt zpoždění mezi signály, existuje několik. Nejběžnější metodou pro nalezení zpoždění mezi signály je využití vzájemné korelační funkce. Často je tato metoda nahrazována tzv. **Generalized Cross-Correlation with PHase Transform (GCC-PHAT)**, navrženou pány Knapp a Carter [9] (viz 3.5.2). Další často používanou metodou, v případě většího počtu mikrofonů než dva, je tzv. **Steered-Response Power PHase Transform (SRP-PHAT)** (viz 3.5.3). V práci [10] je využito adaptivního filtru s metodou nejmenších středních čtverců (**LMS – Least Mean Squares**). V článku [11] je využito hledání maxim v histogramu fázových rozdílů mezi frekvenčními obrazy signálů z mikrofonů (fázový rozdíl je dán $\angle \left(\frac{X_1(k,l)}{X_2(k,l)} \right)$, kde k vyjadřuje frekvenční rozsah a l časové okno).

Vzhledem k tomu, že běžná vzájemná korelační funkce dosahuje v našem případě dobrých výsledků a je výpočetně méně náročná, byla zvolena jako metoda pro určení zpoždění.

3.5.1 Vzájemná korelační funkce

Nejběžnější metodou nalezení časového zpoždění mezi dvěma signály je využití vzájemné korelační funkce (**CCF – Cross-Correlation Function**), která bude podrobně rozebrána v sekci 4.6.1 a je definovaná jako

$$R_{x_1x_2}[k] = \mathbb{E} [x_1[n]x_2[n+k]], \quad (3.7)$$

kde operátor $\mathbb{E}[\cdot]$ vyjadřuje střední hodnotu a $x_1[n]$, $x_2[n]$ jsou zachycené signály.

Odhad časového zpoždění mezi signály je poté dán nalezením maxima vzájemné korelační funkce

$$\hat{D} = \underset{k}{\operatorname{argmax}} R_{x_1x_2}[k]. \quad (3.8)$$

Nevýhodou této metody je nízká spolehlivost v případě, že poměr signálu k šumu (SNR) je nízký [12]. Další nevýhodou vzájemné korelační funkce (a také jiných metod v časové doméně) je nízké rozlišení, způsobené vzorkováním (více v části 4.7.2). Zlepšení rozlišení lze dosáhnout použitím interpolace, jak bude uvedeno v následující části.

Zvýšení rozlišení pomocí kvadratické interpolace

Vzhledem k tomu, že se pohybujeme v diskrétním čase, je i odhad CCF diskrétní. V případě odhadu zpoždění mezi signály (TDE – Time Delay Estimation) nás zajímá jen daný symetrický interval okolo 0. Rozlišení na tomto intervalu je pak poměrně malé a je dáno velikostí vzorkovací frekvence.

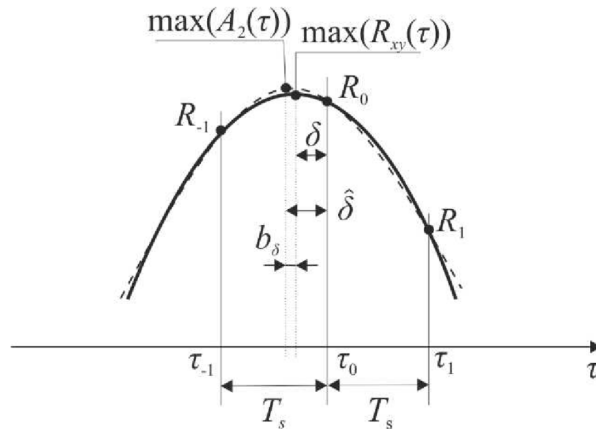
V článku [13] je ukázán způsob zlepšení rozlišení pomocí kvadratické interpolace. Předpokladem kvadratické interpolace je, že měřený signál je pásmově omezen ve frekvenčním intervalu $(-W, W)$ a díky tomu lze signál v časové doméně interpolovat pomocí funkce sinc ($\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x}$ pro $x \neq 0$ a $\text{sinc}(0) = 1$). V případě CCF pak za ideálních podmínek tento předpoklad platí a v bodě okolo zpoždění D je poté symetrický vrchol, který lze aproximovat parabolou [13].

Pro výpočet pozice maxima interpolační funkce vyjdeme z rovnice pro kvadratickou funkci

$$y = ax^2 + bx + c, \quad (3.9)$$

kde $a, b, c \in \mathbb{R}$ jsou konstanty. Maximum funkce (v případě konvexní funkce minimum) je poté ve stacionárním bodě, tedy v bodě s nulovou první derivací, a to odpovídá bodu

$$x = -\frac{b}{2a}. \quad (3.10)$$



Obrázek 3.6: Vzájemná korelační funkce a interpolovaná parabola se zvýrazněnými body maxim. Převzato z [14].

Abychom získali koeficienty a, b, c kvadratické funkce z rovnice (3.9), potřebujeme tři body. Pro tento účel vezmeme nalezené maximum CCF R_{\max}

v bodě n a dva sousední body, tedy v bodech $n - 1$ a $n + 1$. Bod maxima označíme jako R_0 , levý sousední bod jako R_{-1} a pravý jako R_1 . Vzhledem k tomu, že absolutní pozice není pro výpočet koeficientů důležitá, přemístíme celý interpolační problém do okolí bodu 0, tedy dostáváme body $(-1, R_{-1})$, $(0, R_0)$, $(1, R_1)$. Po dosazení těchto bodů do rovnice (3.9) dostáváme tři rovnice o třech neznámých. Provedeme-li řešení, dostaneme rovnice pro jednotlivé koeficienty

$$\begin{aligned} a &= \frac{R_{-1} - 2R_0 + R_1}{2}, \\ b &= \frac{R_1 - R_{-1}}{2}, \\ c &= R_0. \end{aligned} \quad (3.11)$$

Dosazením do rovnice (3.10) pro bod extrému kvadratické funkce a posunutím do bodu n dostaneme zpřesněnou pozici maxima vzájemné korelační funkce

$$n_{\text{QI}} = n + \frac{R_{-1} - R_1}{2(R_{-1} - 2R_0 + R_1)}. \quad (3.12)$$

Na obrázku 3.6 lze vidět reálnou křivku **CCF** s maximem označeným jako $\max(R_{xy}(\tau))$ a parabolu proloženou třemi body, s maximem označeným jako $\max(A_2(\tau))$. Je zde vyznačena odchylka δ nalezeného maxima diskrétní **CCF** od reálného maxima a také odchylka b_δ nalezeného maxima interpolační paraboly od reálného maxima.

Kromě kvadratické interpolace pomocí tří bodů lze využít i jiné, výpočetně náročnější metody. Dle [14] je pro určení zpoždění (**TDE**) výše uvedená metoda ve většině případů dostatečná. Metodu zpřesnění pomocí interpolace lze použít i v případě **GCC**, kde je maximum funkce více zvýrazněno.

3.5.2 Generalized Cross-Correlation with Phase Transform

Tato část se zabývá popisem generalizované vzájemné korelační funkce (**GCC**) pro jeden zdroj signálu, která zlepšuje poměr signálu k šumu (**SNR**) a byla navržena v článku [9] pány Knapp a Carter v roce 1976. Analýzou **GCC-PHAT** pro více zdrojů signálů se poté zabývá práce [15].

Uvažujme systém popsán rovnicemi

$$x_1(t) = s(t) + n_1(t), \quad (3.13)$$

$$x_2(t) = \alpha \cdot s(t - \tau) + n_2(t). \quad (3.14)$$

Ve frekvenční doméně je popsán rovnicemi

$$X_1(f) = S(f) + N_1(f), \quad (3.15)$$

$$X_2(f) = \alpha \cdot S(f) \cdot e^{-i2\pi f\tau} + N_2(f), \quad (3.16)$$

kde i symbolizuje komplexní jednotku definovanou pomocí $i^2 = -1$. Běžný způsob nalezení odhadu zpoždění signálů τ je pomocí vzájemné korelační

funkce. V případě, že je maximum vzájemné korelační funkce ostře ohraničeno, je spolehlivost vyšší [12]. Zvýraznění maxima vzájemné korelační funkce lze dosáhnout použitím váhovací funkce, která vyhladí vstupní signál.

Vzájemná korelační funkce signálů $x_1(t)$ a $x_2(t)$ je se vzájemnou výkonovou spektrální hustotou svázána známým vztahem skrze inverzní Fourierovu transformaci [9]

$$R_{x_1x_2}(t) = \int_{-\infty}^{\infty} G_{x_1x_2} e^{i2\pi ft} df. \quad (3.17)$$

Knapp a Carter navrhli metodu zvanou **Generalized Cross-Correlation (GCC)**, která používá váhovací funkci ke zlepšení poměru signálu k šumu (**SNR**) a je popsána vztahem

$$R_{x_1x_2}^{(g)}(t) = \int_{-\infty}^{\infty} \psi_g(f) G_{x_1x_2} e^{i2\pi ft} df, \quad (3.18)$$

kde $\psi_g(f)$ symbolizuje váhovací funkci. Často používanou váhovací funkcí je fázová transformace (**PHAT – PHase Transform**) [15], která využívá převrácenou hodnotu velikosti vzájemné výkonové spektrální hustoty, tedy $\psi_g(f) = \frac{1}{|G_{x_1x_2}|}$. V případě nekorelovaného šumu (tedy $G_{n_1n_2}(f) = 0$) je $|G_{x_1x_2}| = \alpha \cdot G_{s_1s_1}(f)$ a poté $\frac{G_{x_1x_2}}{|G_{x_1x_2}|} = e^{i2\pi f\tau}$ [9]. Díky tomu po provedení inverzní Fourierovy transformace dostaneme $R_{x_1x_2}^{\text{PHAT}} = \delta(t - \tau)$.

Pro výpočet vzájemné výkonové spektrální hustoty využijeme odhadu $\hat{G}_{x_1x_2} = X_1(f)X_2^*(f)$, kde * značí komplexní sdružení. Váhovaná vzájemná výkonová spektrální hustota má poté tvar

$$\hat{G}_{\text{PHAT}}(f) = \frac{\hat{G}_{x_1x_2}(f)}{|\hat{G}_{x_1x_2}(f)|} = \frac{X_1(f)X_2^*(f)}{|X_1(f)X_2^*(f)|}, \quad (3.19)$$

kde po provedení inverzní Fourierovy transformace dostaneme vzájemnou korelační funkci $\hat{R}_{x_1x_2}^{\text{PHAT}}(t)$. Odhad časového zpoždění mezi signály lze poté získat jako $\hat{\tau} = \arg\max_t \hat{R}_{x_1x_2}^{\text{PHAT}}(t)$ [12].

Tuto metodu lze použít také v diskrétním čase a to s použitím **FFT**, tedy efektivní implementace diskrétní Fourierovy transformace.

■ 3.5.3 Steered-Response Power Phase Transform

Ačkoliv se práce zabývá akustickou lokalizací pomocí páru mikrofonů, je vhodné uvést i metodu používanou při větším množství mikrofonů. Metoda používá tzv. *beamforming*, který lze definovat jako tzv. *filter-and-sum* proces, kdy signály z mikrofonů postavených do mikrofonového pole nejdříve vyfiltrujeme (v našem případě časově posuneme) a poté všechny sečteme do jednoho signálu [16]. V případě, že pozice zdroje signálu není známa, lze využít *beamforming* ke skenování zvoleného prostorového regionu úpravou časových zpoždění, tedy zkusíme určitou množinu možností časových zpoždění

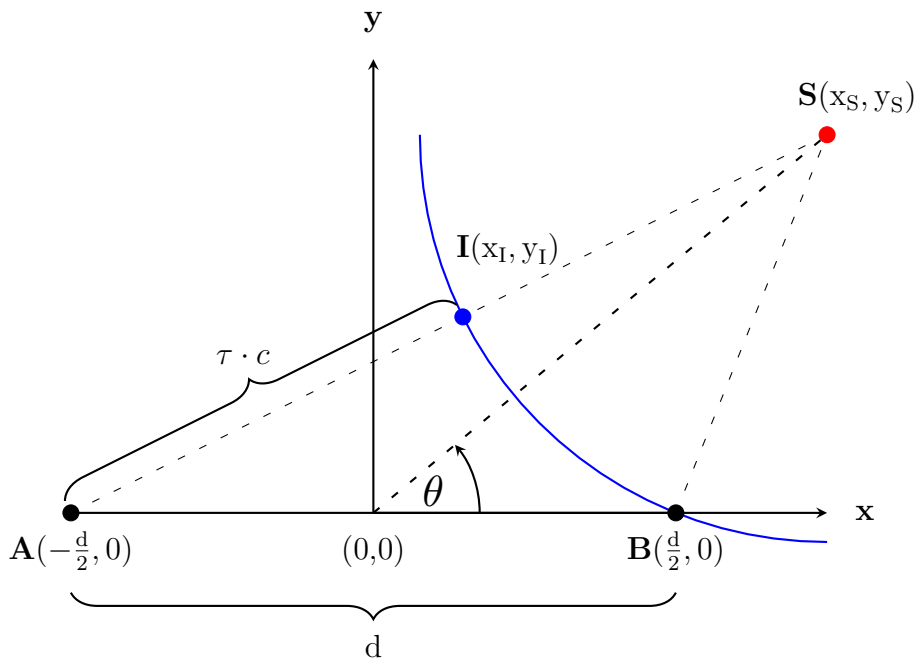
a hledáme možnost, kdy dojde k maximální hodnotě sečteného *filter-and-sum* signálu. Tato metoda se nazývá **Steered Response Power (SRP)**. **SRP** může mít několik vrcholů, ale v případě dobrých podmínek bude její maximum v bodě, kde časová zpoždění odpovídají reálným zpožděním [16]. **SRP** využívá výše uvedenou **GCC** a často s váhování pomocí fázové transformace **PHAT**. Pro 2 mikrofony je tato metoda ekvivalentní výše uvedené funkci **GCC**, případně **GCC-PHAT**.

3.6 Výpočet úhlu z časového zpoždění

Po nalezení zpoždění mezi signály je dalším krokem výpočet odhadu úhlu, pod kterým akustická událost do zařízení dorazila. V následující části bude odvozena rovnice pro výpočet úhlu dopadu.

3.6.1 Odvození

Odvození řešení provedeme dle obrázku 3.7. Máme zde zdroj zvukového signálu v bodě **S**, dva mikrofony v bodech **A** a **B**, které jsou symetricky rozloženy okolo počátku $(0,0)$. Dále je zde vyznačena část kulové vlnoplochy s poloměrem $r = \|\vec{S} - \vec{B}\|^{2,3}$, kde $\|\cdot\|$ značí normu a v případě této práce je ve 2D prostoru definována jako $\|(x, y)\| = \sqrt{x^2 + y^2}$. Bod **I** značí průsečík vyznačené vlnoplochy o poloměru r s úsečkou **AS**.



Obrázek 3.7: DoA – exaktní odvození.

²V této části budeme body značit jako vektory pomocí značení $\vec{\cdot}$.

³Vyjádření $\|\vec{X} - \vec{Y}\|$ značí vzdálenost mezi body (vektory) \vec{X} a \vec{Y} a lze snadno ukázat, že platí $\|\vec{X} - \vec{Y}\| = \|\vec{Y} - \vec{X}\|$.

Víme, že délka úsečky \mathbf{AI} je rovna trase, kterou musí zvuk urazit během časového zpoždění mezi signály τ a odpovídá $\tau \cdot c$, kde c je rychlost zvuku. Dráhový rozdíl označíme jako $\Delta = \tau \cdot c$. Z geometrie lze odvodit následující rovnici

$$\|A - S\| = \Delta + \|I - S\|, \quad (3.20)$$

kde po dosazení rovnosti $\|I - S\| = \|B - S\|$ a přeorganizování členů dostaneme rovnici

$$\|S - A\| - \|S - B\| = \Delta. \quad (3.21)$$

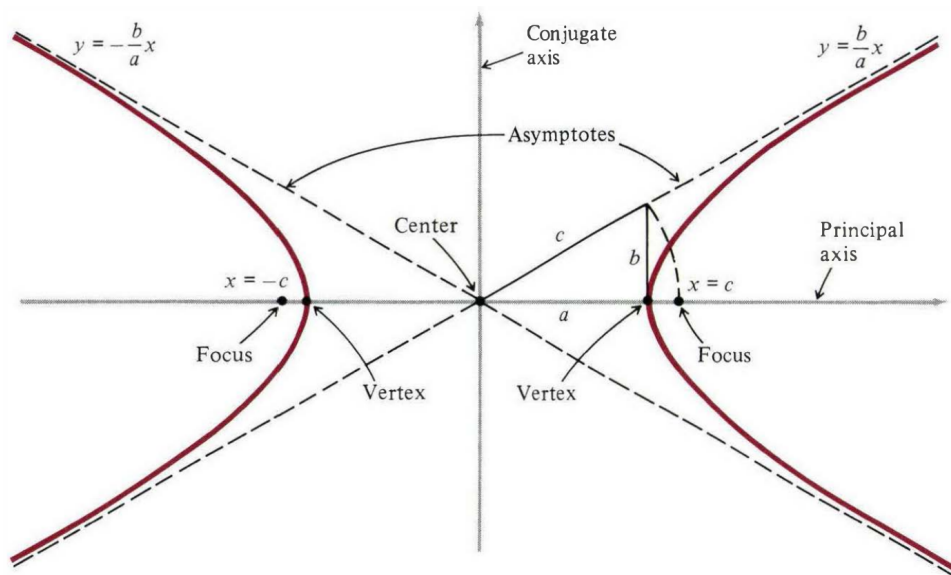
Po dosazení souřadnic do rovnice (3.21) a nahrazením metrik $\|\cdot\|$ za odmocniny získáme

$$\sqrt{\left(x_S - \left(-\frac{d}{2}\right)\right)^2 + (y_S - 0)^2} - \sqrt{\left(x_S - \frac{d}{2}\right)^2 + (y_S - 0)^2} = \Delta \quad (3.22)$$

a po upravení a odstranění indexů souřadnic x_S a y_S dostaneme

$$\sqrt{\left(x + \frac{d}{2}\right)^2 + y^2} - \sqrt{\left(x - \frac{d}{2}\right)^2 + y^2} = \Delta. \quad (3.23)$$

Rovnice (3.23) popisuje pravou (v případě záporného Δ levou) část hyperboly symetrické okolo osy y .



Obrázek 3.8: Vlastnosti a významné body hyperboly. Převzato z [17].

Vezmeme-li rovnici hyperboly zadanou implicitně dle [17]

$$\sqrt{(x + c)^2 + y^2} - \sqrt{(x - c)^2 + y^2} = \pm 2a, \quad (3.24)$$

můžeme ji převést na rovnici (3.23) nahrazením proměnných $c = \frac{d}{2}$ a $a = \frac{\Delta}{2}$ (\pm zanedbáme). Vlastnosti a významné body hyperboly jsou vyznačené na obrázku 3.8.

Úpravami lze rovnici (3.24) převést do tzv. kanonického tvaru

$$\frac{x^2}{a^2} - \frac{y^2}{b^2} = 1, \quad (3.25)$$

kde $b^2 = c^2 - a^2$ [17] (viz obr. 3.8). Po úpravě lze získat explicitní funkci hyperboly popsanou

$$y = \pm \sqrt{b^2 \cdot \left(\frac{x^2}{a^2} - 1 \right)}. \quad (3.26)$$

Dosazení našich proměnných do rovnice (3.26) dostaneme rovnici

$$y = \pm \sqrt{\frac{d^2 - \Delta^2}{4} \cdot \left(\frac{4x^2}{\Delta^2} - 1 \right)}. \quad (3.27)$$

Aby rovnice (3.27) platila, musí být $\frac{4x^2}{\Delta^2} \geq 1$. Abychom dostali správnou část hyperboly dle rovnice (3.23), budeme požadovat podmínku $x \geq \frac{\Delta}{2}$.

Asymptota hyperboly v kladné části je dána rovnicí

$$y = \frac{b}{a}x. \quad (3.28)$$

Úhel, který svírá asymptota s osou x , lze spočítat pomocí rovnice $\cos \theta = \frac{a}{c}$ (viz obr. 3.8). Dosazením proměnných a a b a úpravou získáme rovnici

$$\cos \theta = \frac{\Delta}{d} = \frac{\tau \cdot c}{d}. \quad (3.29)$$

Aplikací inverzní funkce k funkci cosinus dostáváme rovnici pro výpočet odhadu úhlu

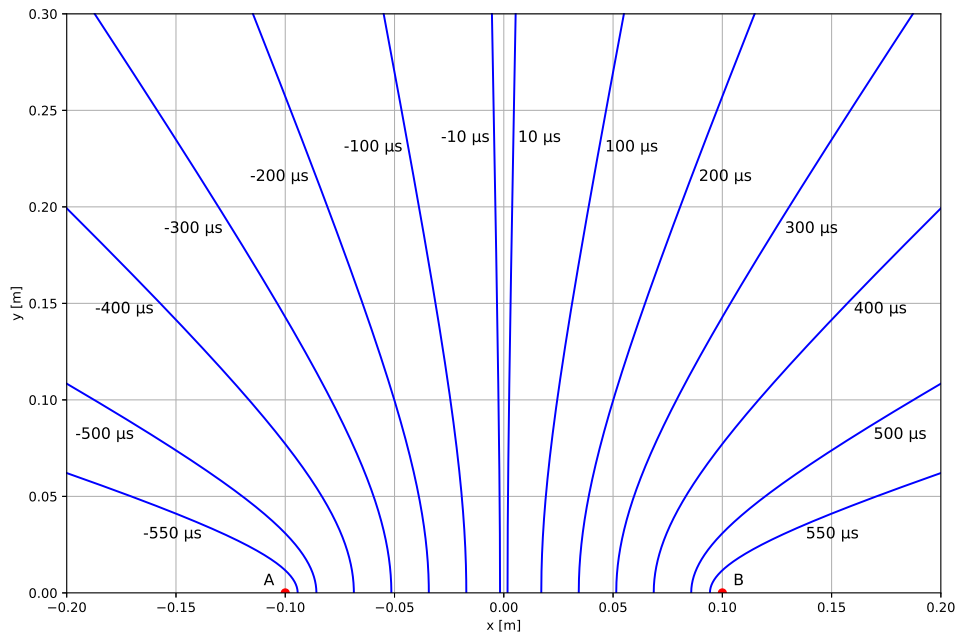
$$\theta = \arccos \left(\frac{\tau \cdot c}{d} \right). \quad (3.30)$$

Na obrázku 3.9 lze vidět vykreslené možné pozice zdroje signálu pro dané hodnoty zpoždění τ (vykresleno pouze v rozsahu $[-0,2 \text{ m}, 0,2 \text{ m}] \times [0 \text{ m}, 0,3 \text{ m}]$). V nezobrazené záporné části osy y je graf symetrický podle osy x , tedy pro daný čas může být zdroj signálu na jedné nebo druhé straně (uvažujeme-li všesměrové mikrofony), ale v našem případě nás zajímá pouze kladná část, která odpovídá pozici z přední strany zařízení (mikrofonů).

3.7 Akustický detekční systém

Z požadavků na akustický detekční systém (viz zadání práce a obr. 2.1) vyplývá potřeba mikrokontroléru s možností propojení s PC (případně jiným zařízením, např. typu telefon či tablet) a mikrofonů s všesměrovou charakteristikou (viz obr. 3.10).

Mezi nejběžnější možnosti, jak přenášet data z mikrokontroléru do PC, je drátová technologie USB nebo bezdrátové technologie Bluetooth a Wi-Fi. Nevýhodou technologie USB je náročné (v řadě případů téměř nemožné) propojení mezi mobilními zařízeními (např. telefon či tablet) a mikrokontrolérem.



Obrázek 3.9: Možné pozice zdroje akustického signálu pro různé hodnoty zpoždění τ a rozteč $d = 0,2\text{ m}$ s vynecháním záporné části hyperboly.

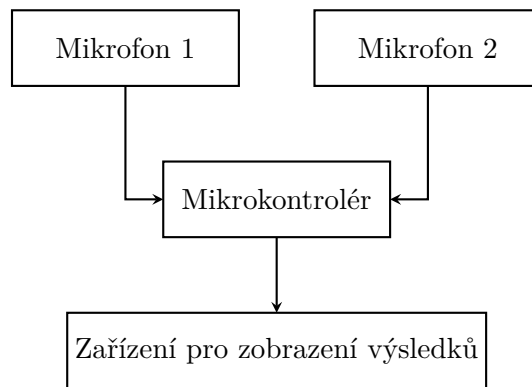
Další nevýhodou USB (a také i Bluetooth) je nutnost tvorby multiplatformní aplikace. Vzhledem k tomu, že nejsou kladeny žádné požadavky na spotřebu zařízení, byla zvolena technologie Wi-Fi, kde k tvorbě grafického rozhraní lze využít webových technologií, které fungují ve většině dnešních webových prohlížečů, a tedy i na většině operačních systémů.

Potřebujeme tedy mikrokontrolér, který bude disponovat konektivitou skrze technologii Wi-Fi, bude schopen vyčítat mikrofony s dostatečnou vzorkovací frekvencí a zároveň bude disponovat dostatečným výkonem pro zpracování audio dat a obsluhování webového serveru. Volba padla na známý mikrokontrolér ESP32 od firmy Espressif (více v části 3.8), který je i v době vysoké nedostupnosti polovodičových prvků stále k dispozici a za přijatelnou cenu. Tento mikrokontrolér je schopen vyčítat sběrnici **I²S (Inter-IC Sound)** a zpracovávat **PDM (Pulse-Density Modulation)** bez zatěžování hlavního procesoru. Na volbu mikrofonů, z pohledu způsobu komunikace, proto nejsou kladeny žádné zvláštní nároky.

Z důvodu snadné dostupnosti a přijatelné ceny byl zvolen **MEMS** mikrofon SPH0645LM4H od firmy Knowles, který splňuje námi kladené požadavky. Více bude tento mikrofon popsán v části 3.9.

3.7.1 MEMS mikrofon

První **MEMS** mikrofon byl uveden v roce 1983 [18]. Ještě před příchodem mikrofonů vyrobených technologií **MEMS** dominovaly trhu elektretové mikrofony, které nabízejí vysokou citlivost a široké frekvenční pásmo za nízkou cenu. Díky zlepšení výrobních technologií mohou být vyráběny **MEMS** mikrofony,



Obrázek 3.10: Blokové schéma systému.

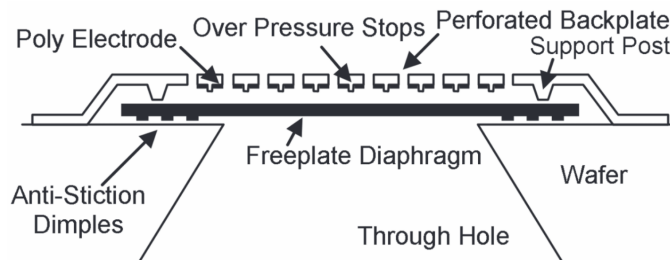
kteří oproti elektretovým nabízí lepší vlastnosti, spolehlivost, menší rozměry, nižší citlivost na mechanické otřesy, možnost osazení na **DPS** při vyšších teplotách a integrovatelnost s CMOS elektronikou na stejném čipu nebo ve stejném pouzdře [18].

■ Technologie MEMS

MEMS je zkratkou **Micro-Electro-Mechanical Systems**. Jedná se o technologii, kterou lze definovat jako mechanické nebo elektromechanické prvky o velmi malých rozměrech [19]. Výhodou této struktury je možnost integrovatelnosti, vyšší přesnosti a jednodušší sériové výroby a tím i nižší ceny.

■ Princip činnosti

MEMS mikrofony mohou využívat různých principů přeměny zvuku na elektrický signál, jako např. piezorezistivního principu nebo optické detekce. Více než 80 % **MEMS** mikrofonů však využívá kapacitního principu [18], proto v této sekci bude popsán pouze kapacitní **MEMS** mikrofon.

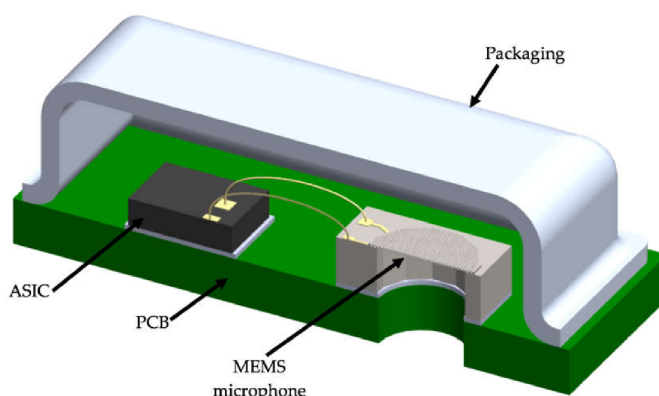


Obrázek 3.11: Stavba kapacitního MEMS mikrofonu. Převzato z [20].

Struktura kapacitního **MEMS** mikrofonu je zobrazena na obr. 3.11. Základem je membrána, která se prohýbá působením zvukových vln z jedné strany. Zvuk ve vzduchu představuje podélné vlnění, které se šíří v látkovém prostředí. Toto vlnění způsobuje zhušťování a zředování látkového prostředí, v důsledku čehož dochází k místním změnám tlaku. Poté, co vlna dorazí

na membránu **MEMS** mikrofonu, dojde v závislosti na tlaku k deformaci membrány jedním nebo druhým směrem.

Z druhé strany membrány je s mírným odstupem napevno umístěna děrovaná elektroda, která umožňuje vyrovnání tlaku na tlak okolí a zároveň spolu s membránou tvoří kondenzátor, u kterého při deformaci membrány dojde ke změně kapacity. Další částí je **Application-Specific Integrated Circuit (ASIC)**, který se stará o převod kapacity na elektrický signál a v případě digitálního výstupu také o převod na digitální signál ve vhodné formě [21]. Nejčastějšími formami digitálního výstupu jsou **PDM** a sběrnice **I²S**, která bude detailněji popsána v části 3.9.1.



Obrázek 3.12: Řez pouzdrem MEMS mikrofonu. Převzato z [22].

MEMS prvek s membránou a **ASIC** obvod jsou poté umístěny na **DPS** a zapouzdřeny pomocí kovového pouzdra. Zároveň je v cuprexitové desce vytvořen otvor pro průchod zvukových vln, jak lze vidět na obrázku 3.12.

3.8 Mikrokontrolér ESP32

Mikrokontrolér ESP32 od firmy Espressif je úspěšným nástupcem známé řady ESP8266. Oproti ESP8266 přináší velkou řadu vylepšení při zachování relativně nízké ceny. Vyrábí se v jednojádrové nebo dvoujádrové variantě s hodinovým taktem až 240 MHz, disponuje konektivitou Wi-Fi a Bluetooth, HW podporou různých sběrnic a dalšími funkcemi [23].

Část dnešních mikrokontrolérů obsahuje HW podporu **I²S** sběrnice. Zvolený mikrokontrolér ESP32 touto podporou disponuje, a to v režimech master/slave pro receiver/transmitter (přijímač/vysílač). Zároveň disponuje **DMA (Discrete Memory Access)** kontrolérem, který umožňuje zapisovat vyčtená data z **I²S** sběrnice přímo do operační paměti mikrokontroléru bez zatěžování CPU, přičemž v případě dokončení přenosu (zaplnění **DMA** bufferu) je vyvoláno přerušení pro CPU [24]. Díky tomu lze využít výpočetní výkon CPU k jiným operacím, jako např. ke zpracování zaznamenaných dat.

3.8.1 Volba vývojového prostředí

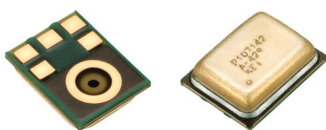
Z důvodu dobré podpory, rozšířenosti a rozsáhlého množství funkcí bylo zvoleno vývojové prostředí **PlatformIO** spolu s editorem **Visual Studio Code**. K dispozici jsou dva **frameworky**, kterými jsou **Arduino** a **ESP-IDF** [25]. ESP-IDF je oficiální vývojový framework od firmy Espressif, který je založen na jazyce C. Oproti tomu Arduino je založeno na jazyce Wiring, který vychází z jazyka C++, a na pozadí pro ESP32 používá framework ESP-IDF. Z důvodu výhod jazyka C++ a rozsáhlého množství knihoven byl zvolen vývojový framework Arduino, nicméně zároveň jsou využívány i funkce z ESP-IDF, např. pro nastavení sběrnice **I²S**.

3.8.2 FreeRTOS

FreeRTOS (Real-Time Operating System) je otevřený operační systém reálného času pro mikrokontroléry [26]. FreeRTOS je do frameworku ESP-IDF integrován pro všechny architektury používané mikrokontroléry ESP (Xtensa a RISC-V) a je upraven tak, aby podporoval **SMP** (**Symmetric Multiprocessing**⁴) na více jádrech (v případě ESP32 dvě jádra, oficiální FreeRTOS podporuje pouze jedno) [27]. Díky tomu jsou do frameworku ESP-IDF dodány další funkcionality, jako např. podpora úloh běžících na specifikovaném jádře pomocí funkce `xTaskCreatePinnedToCore`, které je využito pro zpracovávání dat z mikrofonů (více v části 4.4).

3.9 Mikrofon SPH0645LM4H

Z důvodu snadné dostupnosti byl zvolen mikrofon SPH0645LM4H technologie SiSonic^{TM5} od firmy Knowles, komunikující skrze sběrnici **I²S**, se zvukovým portem na spodní straně **SMD** čipu (viz obr. 3.13), s nízkou spotřebou, všesměrovou charakteristikou a podporou vzorkovací frekvence 16 kHz až 64 kHz [28]. Jak je uvedeno v [28, s. 6], formát dat je 24 bitů, dvojkový doplněk, **MSB** (**Most Significant Bit**) je první, rozlišení dat je 18 bitů a zbytek je doplněn nulami.

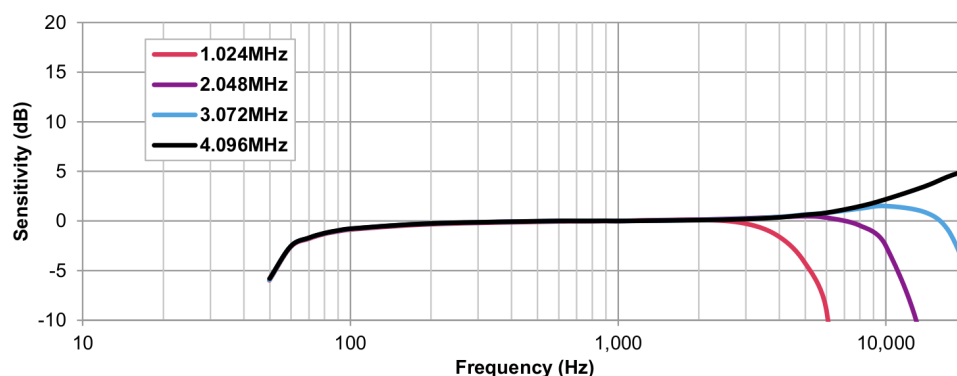


Obrázek 3.13: Čip mikrofonu SPH0645LM4H. Převzato z [28].

Konkrétní implementací komunikace s tímto mikrofonem a zpracováním vyčtených hodnot se zabývá část 4.3.

⁴Výpočetní architektura, kde dvě nebo více identických jader používá stejnou sdílenou paměť a je ovládáno jedním operačním systémem.

⁵SiSonicTM je první komerčně dostupný kondenzátorový **MEMS** mikrofon, který byl optimalizován pro nízké výrobní náklady, ale zároveň dobré vlastnosti a spolehlivost [20].



Obrázek 3.14: Závislost citlivosti mikrofonu SPH0645LM4H na frekvenci zvuku pro různé hodinové frekvence. Převzato z [28].

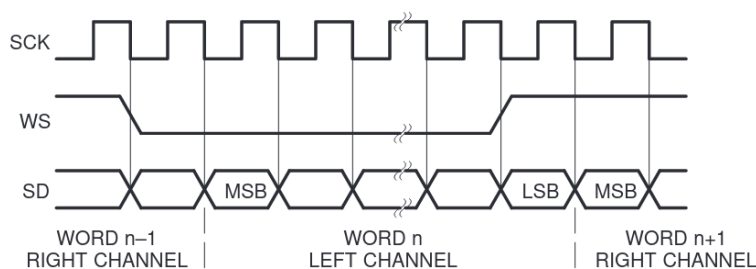
Na obrázku 3.14 je znázorněna závislost citlivosti mikrofonu SPH0645LM4H na frekvenci zvuku pro různé hodinové frekvence. V případě nastavení provedeném v této práci se jedná přibližně o křivku pro frekvenci 3,072 MHz.

3.9.1 Sběrnice I²S

Jak je uvedeno v datasheetu [29] od firmy Philips, sběrnice I²S (Inter-IC Sound) byla poprvé představena danou firmou v roce 1986 pro přenos audio dat mezi různými komponenty. Sběrnice se skládá ze tří vodičů – hodinového signálu SCK (Serial Clock), datového signálu SD (Serial Data) a volby slova WS (Word Select). Ovládací zařízení (tzv. master) generuje hodinový signál SCK a volbu slova WS. Propojení zařízení typu master v roli přijímače je zobrazeno na obrázku 3.15.



Obrázek 3.15: Ovládací zařízení master v roli přijímače. Převzato z [29].



Obrázek 3.16: Základní časování sběrnice I²S. Převzato z [29].

Sériová data jsou vysílána ve formě dvojkového doplňku s **MSB** prvním. Není potřeba, aby vysílač věděl, kolik bitů je přijímač schopen přijmout (a naopak), protože buďto jsou data za **LSB (Least Significant Bit)** ignorována nebo jsou doplněna nulami.

Výběr slova (WS) slouží k indikaci, který kanál je právě přenášen. Stav $WS = 0$ značí kanál 1 (levý) a stav $WS = 1$ značí kanál 2 (pravý). WS se může měnit během sestupné nebo náběžné hrany a mění se 1 periodu hodinového signálu před **MSB** bitem, díky čemuž může vysílač typu slave odvodit správné synchronní časování vysílaných dat a přijímač uložit a vymazat vstup pro příjem nových dat.

Časování na sběrnici I²S je zajištěno zařízením typu master, které vysílá hodinový signál skrze vodič SCK. Zařízení typu slave využije tento hodinový signál ke správné synchronizaci. Základní časování sběrnice je zobrazeno na obrázku 3.16. Vysílač synchronizuje vysílání dat buďto se sestupnou hranou hodinového signálu HIGH-to-LOW nebo s náběžnou hranou LOW-to-HIGH. Přijímač musí tato data vyčíst ve stavu, kdy jsou ustálena.

Kapitola 4

Praktická část

4.1 Postup zpracování dat

Na obrázku 4.1 je zobrazen blokový diagram postupu zpracování vyčtených hodnot. Ze všeho nejdříve je potřeba provést nastavení celého systému a poté zahájit proces sběru audio vzorků. U každého vzorku se testuje, zda byla detekována akustická událost (více v části 4.5). V případě, že byla akustická událost detekována, je započat sběr vzorků. Z těchto vzorků je poté spočítán odhad časového zpoždění mezi signály (část 4.6) a v případě úspěšného nalezení zpoždění (myšleno ve smyslu úspěšného nalezení maxima vzájemné korelační funkce) je proveden výpočet odhadu úhlu (část 4.7), pod kterým akustická událost dorazila. Výsledný odhad úhlu je poté zobrazen pomocí webového rozhraní (kapitola 5).

Zdrojový kód lze nalézt v příloze A.

4.2 Volba vzorkovací frekvence

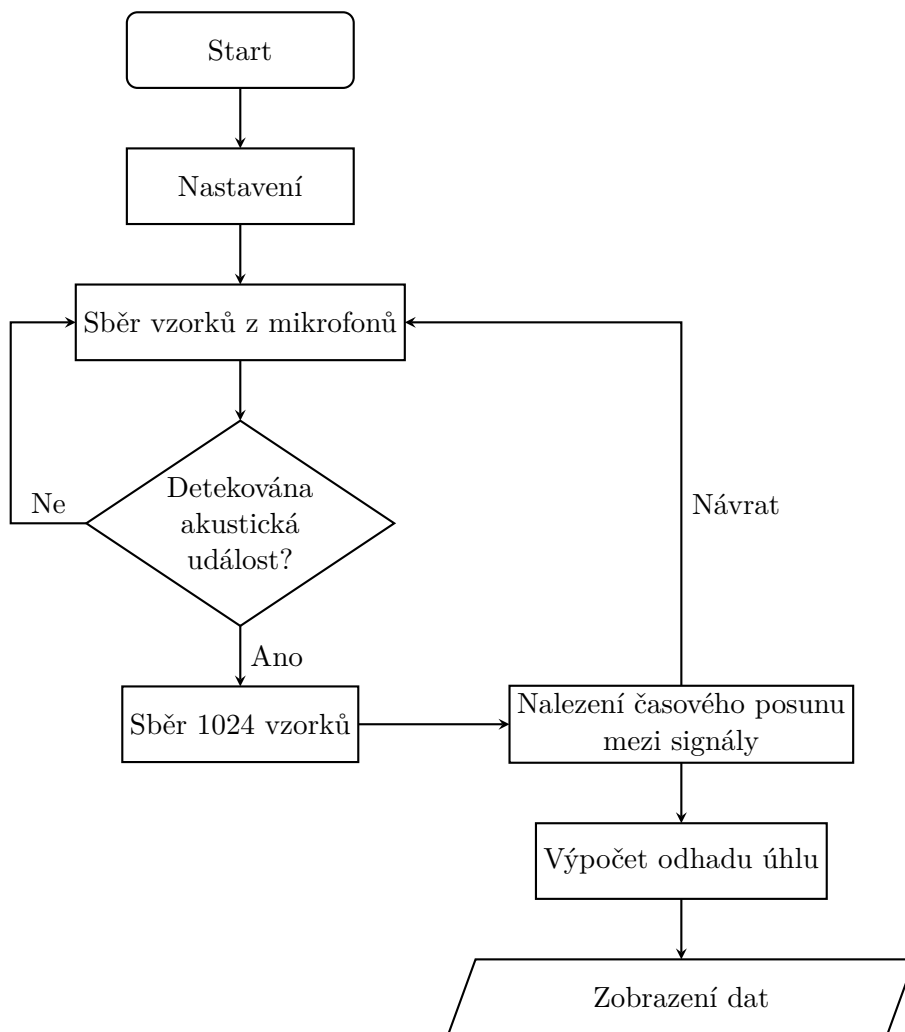
Jak je uvedeno v kapitole 3.9, podporovaná vzorkovací frekvence mikrofону SPH0645LM4H je od 16 kHz do 64 kHz. Platí, že čím vyšší vzorkovací frekvence, tím vyšší úhlové rozlišení (více v sekci 4.7.2), ale zároveň vyšší nároky na operační paměť a výpočetní výkon použitého mikrokontroléru.

Dle Nyquistova vzorkovacího teorému potřebujeme pro úspěšnou rekonstrukci signálu vzorkovací frekvenci alespoň dvojnásobnou, než je maximální frekvence obsažená ve vzorkovaném signálu. Pro frekvenční rozsah do 20 kHz tedy požadujeme $f_s > 40$ kHz.

Z těchto důvodů byla zvolena pro vzorkovací frekvenci standardní hodnota 44,1 kHz.

4.3 Vyčítání mikrofónů SPH0645LM4H mikrokontrolérem ESP32

Vyčítání páru mikrofónů SPH0645LM4H pomocí mikrokontroléru ESP32 probíhá skrze sběrnici I²S. Vzhledem k tomu, že zvolený mikrokontrolér



Obrázek 4.1: Blokový diagram vyhodnocení.

disponuje **DMA** kontrolérem a ve frameworku ESP-IDF je obsluha tohoto kontroléru implementována, je použití značně zjednodušeno.

Samotné nastavení bylo provedeno dle [30] a [31]. Velikost **I²S** slotu byla zvolena na 32 bitů, vzorkovací frekvence byla použita 44,1 kHz (viz část 4.2), délka **DMA** bufferu byla zvolena na 1024 vzorků (tedy 512 vzorků pro levý a 512 pro pravý kanál) a počet **DMA** bufferů byl zvolen 14. Díky vlastnostem sběrnice **I²S** je možné obsluhovat oba mikrofony pomocí jednoho **I²S** portu, a to nastavením jednoho mikrofonu na levý a druhého na pravý kanál.

Vytvořené nastavení je aplikováno pomocí funkce `i2s_driver_install`. Pro zvolený mikrofon SPH0645LM4H je potřeba provést úpravy časování pomocí kódu [31]

```
REG_SET_BIT(I2S_TIMING_REG(i2s_port), BIT(9));
REG_SET_BIT(I2S_CONF_REG(i2s_port), I2S_RX_MSB_SHIFT);
```

kde proměnná `i2s_port` udává číslo **I²S** portu a v našem případě je nastavena

na I2S_NUM_0, tedy port 0.

Velikost skutečných vzorků z mikrofonu SPH0645LM4H je 18 bitů. Z důvodu úspory operační paměti byla zvolena velikost datové proměnné, držící vzorky zvuku během sběru vzorků pro výpočet časového zpoždění, na 16 bitů. Po vyčtení dat skrze sběrnici **I²S** je tedy potřeba převést 32 bitové vzorky tak, aby je bylo možné uložit do 16 bitových proměnných. Jak je uvedeno v **3.9**, mikrofon SPH0645LM4H posílá 24 bitů dat, ale posledních 6 bitů jsou nuly. Bitovým posunem `sample » 14` provedeme odstranění nepotřebných 14 bitů (8 + 6). Poté provedeme logickou operaci AND s maskou `0xFFFF` a uložíme do proměnné typu `int16_t`. V případě, že došlo k přetečení (dojde ke změně znaménka, protože datový typ je typu `signed`), vrátíme buďto hodnotu `INT16_MIN` nebo `INT16_MAX`, čímž provedeme ořez (limitaci) 18 bitové proměnné do 16 bitové.

Po vyčtení dat z mikrofonů lze pozorovat nenulový DC offset (stejnosemná složka). Jak je uvedeno v datasheetu mikrofonu SPH0645LM4H [28, s. 2] „DC offset can be removed from the PCM samples if desired by implementing a DC-blocking or high-pass filter“, lze DC offset odstranit digitálním DC filtrem. Zároveň je v datasheetu uvedeno, že DC offset může mít maximální hodnotu 5 % z celého rozsahu. Jedná se o mikrofon s přesností dat 18 bitů, z čehož plyne, že maximální hodnota rozsahu je 2^{18} , a tedy DC offset může být až $\text{floor}(2^{18} \cdot \frac{5}{100}) = 13\,107$. Měřením bylo zjištěno, že DC offset je přibližně 10 000, tedy v uvedené toleranci. Popisem DC filtru se zabývá kapitola **4.3.1**.

4.3.1 Filtrace DC složky

Pro odstranění stejnosměrné složky ze signálu byl zvolen DC filtr typu **IIR** (**Infinite Impulse Response** – filtr s nekonečnou impulzní odezvou) dle [32], který je popsán diferenční rovnicí

$$y[n] = x[n] - x[n - 1] + R \cdot y[n - 1], \quad (4.1)$$

kde $x[n]$ je vstupní signál, $y[n]$ výstupní a R konstanta.

Provedením \mathcal{Z} -transformace dostaneme

$$Y(z) = X(z) - X(z) \cdot z^{-1} + R \cdot Y(z) \cdot z^{-1} \quad (4.2)$$

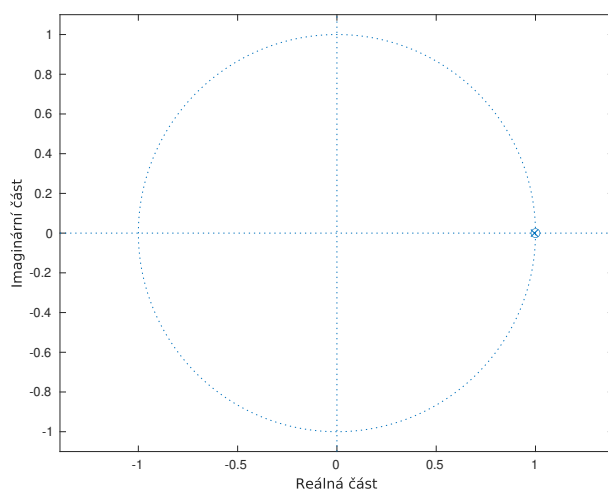
a po úpravě potom přenosovou funkci ve tvaru racionální lomené funkce

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 - z^{-1}}{1 - R \cdot z^{-1}}. \quad (4.3)$$

Vynásobíme-li přenosovou funkci z rovnice (4.3) výrazem $\frac{z}{z}$, dostaneme tvar

$$H(z) = \frac{z - 1}{z - R}, \quad (4.4)$$

z něhož lze vidět, že filtr má nulový bod (hodnota parametru z , která způsobí vynulování čitatele, a tedy i celé funkce) v $z = 1$ a pól (hodnota parametru z , kde dochází k vynulování jmenovatele – singularita) v bodě $z = R$.



Obrázek 4.2: Poloha nulového bodu a pólu v Z -rovině pro $R = 0,995$.

Využijeme-li Nyquistovy normalizované frekvence $\theta = \omega \cdot T_s = 2\pi \cdot \frac{f}{f_s}$ a za argument z dosadíme $z = e^{i\theta}$ (spektrum diskretních signálů je periodické), dostaneme

$$H(e^{i\theta}) = \frac{e^{i\theta} - 1}{e^{i\theta} - R}.$$

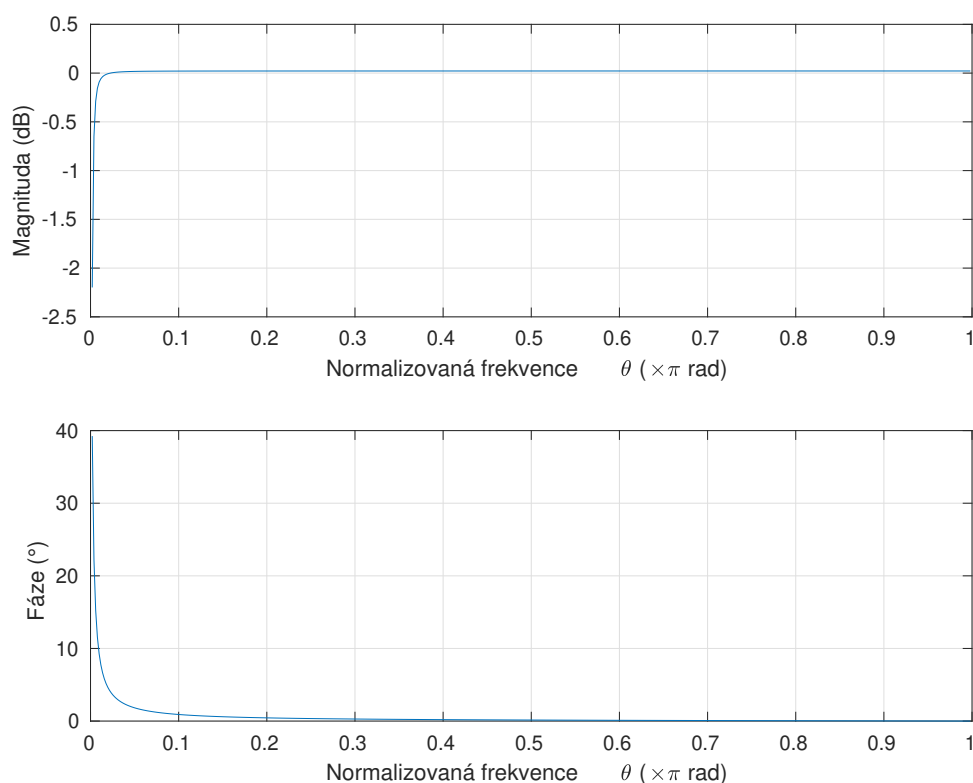
Poloha pólu určuje, jak moc bude daný filtr potlačovat DC složku (DC složka je v bodě $\theta = 0$) a naopak zkreslovat střídavou složku. Zároveň se musí pól nacházet uvnitř jednotkové kružnice, aby byla zaručena stabilita IIR filtru. V bodě $\theta = 0$ je $H(e^0)$ rovno 0 (v logaritmické stupnici se blíží k $-\infty$), a tedy DC složka je potlačena. K nejmenšímu zkreslení střídavé složky dochází v bodě $\theta = \pi$, což odpovídá polovině vzorkovacího kmitočtu, kdy $H(e^{i\pi}) \rightarrow 1$ pro $R \rightarrow 1$ (nula a pól se navzájem vyruší).

Jak je uvedeno v [32], R by mělo být v rozsahu $[0, 1)$ a pro vzorkovací frekvenci 44,1 kHz je vhodné $R = 0,995$. Poloha nulového bodu a pólu v Z -rovině je vykreslena na obr. 4.2. Frekvenční charakteristika je vykreslena na obr. 4.3, kde amplitudová charakteristika je tvaru $20 \cdot \log |H(e^{i\theta})|$ a fázová charakteristika tvaru $\angle H(e^{i\theta})$.

4.4 Implementace zpracování dat v ESP32

V části 4.3 bylo uvedeno použití mikrofону SPH0645LM4H s mikrokontrolérem ESP32. Pro realizaci funkčního zařízení je potřeba, aby mikrokontrolér kontinuálně vyhodnocoval data z mikrofónů a zároveň byl schopen obsluhovat webový server. Mikroprocesor ESP32 (námi zvolená verze ESP32-WROOM-32E) disponuje dvěma procesorovými jádry a tedy i dostatečným výkonem pro splnění těchto požadavků.

Aby bylo možné kontinuálně vyhodnocoval data z mikrofónů (konkrétně vyčítat data z DMA bufferů pomocí funkce `i2s_read`), je potřeba vytvořit separátní úlohu, v níž bude vyhodnocení prováděno. Pro vytvoření úlohy



Obrázek 4.3: Frekvenční charakteristika DC filtru pro $R = 0,995$.

běžící na daném jádře lze využít FreeRTOS funkce frameworku ESP-IDF `xTaskCreatePinnedToCore` pomocí následujícího kódu:

```
xTaskCreatePinnedToCore(
    dsp_func, // Function to implement the task
    "DSP",   // Name of the task
    10000,   // Stack size in words
    &mics,    // Task input parameter
    2,       // Priority of the task
    &dsp_task, // Task handle
    1       // Core ID
);
```

Ve funkci `dsp_func` poté běží nekonečná smyčka, ve které jsou vyčítána a zpracovávána data z mikrofónů.

4.5 Detekce impulzní akustické události

V části 3.2.2 byly uvedeny metody detekce impulzních akustických událostí. Vzhledem k tomu, že zařízení v této práci má za cíl detekovat všechny impulzní akustické události, nejsou na detekční algoritmus kladeny žádné speciální požadavky.

Pro detekci impulzních akustických událostí byl zvolen následující algoritmus, založený na rekurentním výpočtu střední hodnoty

$$\hat{\mu}[n] = \frac{p \cdot \hat{\mu}[n-1] + (1-p) \cdot |x[n]|}{2}, \quad (4.5)$$

kde $x[n]$ je aktuální vzorek, $\hat{\mu}[n-1]$ a $\hat{\mu}[n]$ jsou odhady předchozí a aktuální střední hodnoty a konstanta p popisuje, jak moc rychle lze střední hodnotu měnit. Experimentálně byla hodnota této konstanty stanovena na $p = 0,8$.

Pro každý vzorek audio signálu je pro oba audio kanály vypočtena zvlášť hodnota $\hat{\mu}_m[n]$ ($m \in \{1, 2\}$) a v případě, že oba kanály mají hodnotu $\hat{\mu}_m[n]$ větší než experimentálně nastavená prahová hodnota 500, je detekována impulzní akustická událost.

4.5.1 Potlačení akustických událostí způsobených odrazy

V části 3.3.2 byla uvedena metoda DATEMM pro odstranění nejednoznačností určení zdroje signálu vlivem odrazů v prostředí s dozvuky. Vzhledem k tomu, že se při experimentech ukázalo přidání běžné časové prodlevy jako efektivní a postačující, byla tato metoda zvolena i při praktické realizaci.

Při detekci impulzní akustické události je započat sběr vzorků pro výpočet vzájemné korelační funkce. Pokud je detekována nová impulzní akustická událost v časové prodlevě menší než 100 ms od předchozí detekce, je tato událost považována za falešnou a sběr vzorků s následným výpočtem neproběhne. Velikost časové prodlevy 100 ms byla zvolena odhadem a ověřena pomocí vhodných experimentů.

4.6 Nalezení zpoždění dvou signálů

V části 3.5 byly popsány metody nalezení časového zpoždění mezi signály z mikrofonů. Pro tuto práci byla zvolena metoda výpočtu pomocí vychýleného odhadu vzájemné korelační funkce, protože poskytuje postačující výsledky s nižší výpočetní náročností než ostatní metody. Dále byla implementována kvadratická interpolace, zmíněná v 3.5.1, avšak vzhledem k požadavkům této práce je vliv tohoto typu zpřesnění zanedbatelný.

V této části bude vzájemná korelační funkce více popsána, především pak její vychýlený a nevychýlený odhad a bude uvedena praktická implementace a volba velikosti okna pro výpočet odhadu vzájemné korelační funkce.

4.6.1 Vzájemná korelační funkce

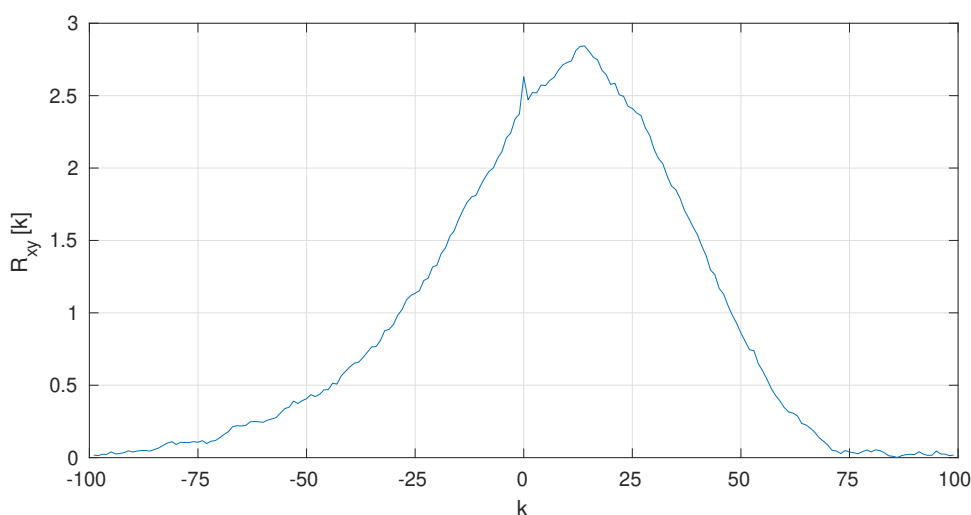
Vzájemná korelační funkce určuje podobnost dvou signálů. V případě, že porovnáváme dva podobné signály (případ této práce), tak lze zjistit zpoždění mezi těmito signály. Budou zde ukázány případy vychýleného a nevychýleného odhadu vzájemné korelační funkce pro reálné signály, avšak existují i jiné typy odhadů a realizaci lze provést také pro komplexní signály.

■ Vychýlený odhad vzájemné korelační funkce

Vychýlený odhad vzájemné korelační funkce pro reálné signály je definován vztahem [33]

$$R_{xy}[k] = \frac{1}{N} \cdot \sum_{n=0}^{N-1-|k|} x[n+k] \cdot y[n], \quad \text{pro } k = 0, \pm 1, \dots, \pm(N-1). \quad (4.6)$$

Vlastností vychýleného odhadu je, že s rostoucí velikostí argumentu k (v kladné i záporné části, tedy $|k|$) klesá interval, přes který počítáme součin funkcí, avšak součet normujeme konstantou $\frac{1}{N}$. To má za následek, že hodnota korelační funkce je systematicky tlumena (viz obr. 4.4). Díky této vlastnosti je tento typ odhadu vhodnější pro náhodné signály, kde maximum korelační funkce by mělo být v bodě, kde se signály přesně překrývají (např. pro stejný bílý šum dosažený za x i y to bude v bodě 0).



Obrázek 4.4: Vychýlený odhad vzájemné korelační funkce pro dva stejné, navzájem zpožděné signály s viditelným systematickým tlumením.

Při praktické implementaci je potřeba brát v potaz, že proměnné $x[n]$ a $y[n]$ budou většinou implementovány formou pole nebo jiných datových struktur, které často podporují pouze indexování v rozmezí $0, 1, \dots, N-1$, kde N je délka pole (datové struktury). Problémem však je, že pro záporné hodnoty argumentu k je pro některá n hodnota $n+k$ také záporná. Proto je potřeba vztah (4.6) upravit do následujícího tvaru

$$R_{xy}[k] = \begin{cases} \frac{1}{N} \cdot \sum_{n=0}^{N-1-|k|} x[n+k] \cdot y[n], & \text{pro } k \geq 0, \\ \frac{1}{N} \cdot \sum_{n=0}^{N-1-|k|} y[n-k] \cdot x[n], & \text{pro } k < 0, \end{cases} \quad (4.7)$$

který záporný posun signálu x vůči signálu y řeší kladným posunem signálu y vůči signálu x .

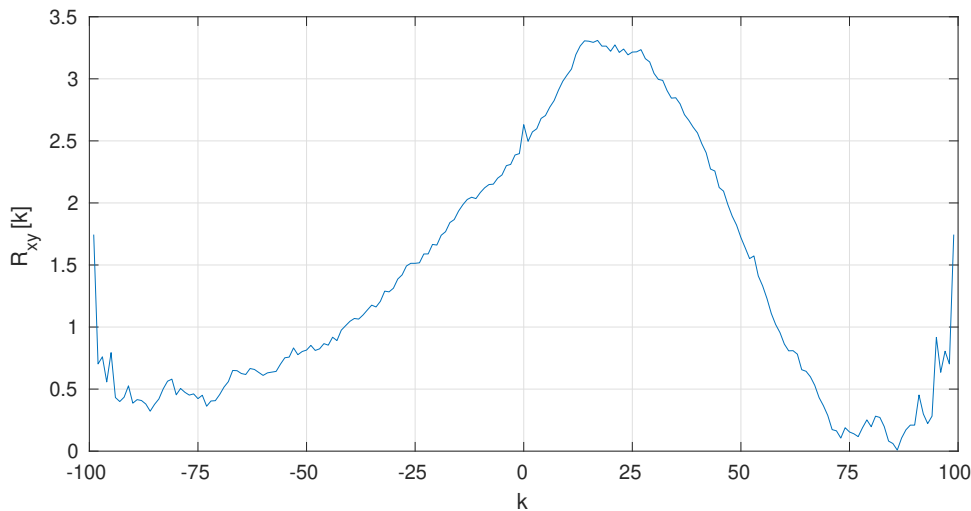
Analogicky lze postupovat v případě implementace výpočtu nevychýleného odhadu vzájemné korelační funkce.

■ Nevychýlený odhad vzájemné korelační funkce

Nevychýlený odhad vzájemné korelační funkce pro reálné signály je definován vztahem [33]

$$R_{xy}[k] = \frac{1}{N - |k|} \cdot \sum_{n=0}^{N-1-|k|} x[n+k] \cdot y[n], \quad \text{pro } k = 0, \pm 1, \dots, \pm(N-1). \quad (4.8)$$

Lze si všimnout, že vzorec je až na normalizaci $\frac{1}{N-|k|}$ téměř identický s vychýleným odhadem. Nevýhodou je, že pro hodnoty $|k| \rightarrow N$ může především u náhodných signálů dojít k náhodné chybě, která se může projevit nárůstem hodnoty vzájemné korelační funkce, jak lze vidět na obrázku 4.5.



Obrázek 4.5: Nevychýlený odhad vzájemné korelační funkce pro dva stejné, navzájem zpožděné signály s viditelnou náhodnou chybou na okrajích.

■ Implementace vzájemné korelační funkce

Vzájemnou korelační funkci lze počítat dvěma způsoby, a to přímou metodou, tedy z definice anebo skrze frekvenční spektrum pomocí FFT (rychlá Fourierova transformace). Výpočetní komplexita FFT je $\mathcal{O}(n \log n)$, oproti tomu výpočetní komplexita vzájemné korelační funkce je $\mathcal{O}(n^2)$. Vzhledem k tomu, že zpoždění signálů se může nacházet jen v rozsahu $[-N_{\max}, N_{\max}]$, není potřeba počítat celou vzájemnou korelační funkci. Pro funkci $R_{xy}[k]$ stačí tedy uvažovat argument $k \in \{x \in \mathbb{Z} \mid x \in [-N_{\max}, N_{\max}]\}$. Díky tomu lze dosáhnout lineární výpočetní komplexity $\mathcal{O}(n)$, a tedy i významné úspory operací.

Při praktické implementaci přímé metody je důležité vzít v úvahu omezenou kapacitu proměnné, do které ukládáme součet součinů signálů. V případě, že tato kapacita je překročena a dojde k přetečení, může dojít k znehodnocení výsledku. Přetečení lze předejít použitím datového typu s velkým rozsahem

(v našem případě je využito datového typu `int64_t`) anebo méně efektivní cestou, kterou je dělení součinu signálu před každým sčítáním.

■ Délka okna odhadu vzájemné korelační funkce

Vzhledem k tomu, že délka běžného tlesknutí je v jednotkách milisekund [4], byla délka okna odhadnuta na 25 ms. Počet vzorků v okně pro výpočet odhadu vzájemné korelační funkce je poté dán vztahem $N_{\text{window}} = T_{\text{window}} \cdot f_s$. V našem případě $N_{\text{window}} = 25 \text{ ms} \cdot 44,1 \text{ kHz} \doteq 1103$ a tato hodnota byla zaokrouhlena k nejbližší mocnině 2, tedy na hodnotu $N_{\text{window}} = 1024$.

■ 4.7 Výpočet úhlu z časového zpoždění

V části 3.6 bylo provedeno odvození úhlu θ skrze asymptotu hyperboly. V této části bude ukázáno trigonometrické odvození za předpokladu zanedbání zakřivení kulových vlnoploch.

Na obrázku 4.6 lze vidět zdroj akustického vlnění ve vzdáleném poli, označený jako **S**, u kterého zanedbáme zakřivení vlnoploch a budeme ho považovat za zdroj rovinných vlnoploch. Vlnoplochy jsou kolmé na přímkou procházející počátkem $(0,0)$ a bodem **S**, která je rovnoběžná s přímkou procházející body **A** a **I**. V čase t tyto vlnoplochy dorazí do mikrofonu **B** a v čase $t + \tau$ do mikrofonu **A**. Rozdílná dráha, kterou musí zvuk urazit za časové zpoždění τ , je dána vynásobením rychlostí zvuku c . Tato dráha zároveň tvoří přilehlou odvěsnu v pravoúhlém trojúhelníku (tvořeném body **A**, **I** a **B**) na obrázku 4.6 a vzdálenost mezi mikrofony d tvoří přeponu tohoto trojúhelníku. Využitím funkce cosinus dostáváme vztah

$$\cos \theta = \frac{\tau \cdot c}{d}, \quad (4.9)$$

z něhož po aplikaci inverzní funkce k funkci cosinus vzejde již známý vztah

$$\theta = \arccos \left(\frac{\tau \cdot c}{d} \right). \quad (4.10)$$

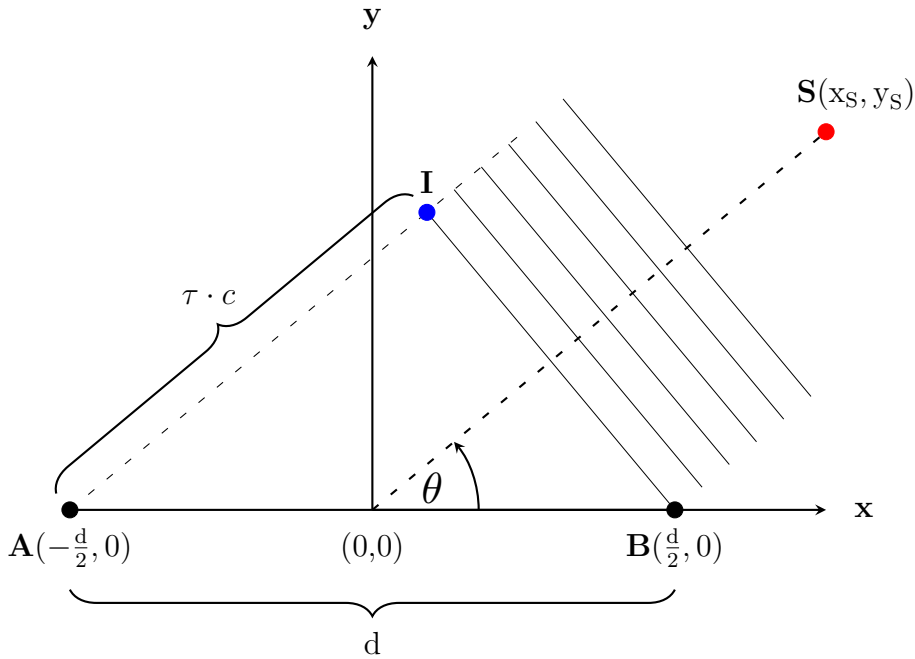
Z rovnice (4.10) s použitím vztahu $\tau = n \cdot T_s$ definujeme pro vzorkový posun n funkci

$$\theta[n] = \arccos \left(\frac{n \cdot T_s \cdot c}{d} \right). \quad (4.11)$$

■ 4.7.1 Maximální dosažitelný úhel vzhledem k vzorkovací frekvenci a vzdálenosti mikrofonů

Vzhledem k tomu, že signál je vzorkován s konečně velikou vzorkovací frekvencí, je maximální dosažitelný rozsah úhlů a maximální chyba určení úhlu dána vzorkovací frekvencí f_s a vzdáleností mezi mikrofony. Úpravou rovnice (4.9) do tvaru

$$\tau = \cos \theta \cdot \frac{d}{c} \quad (4.12)$$



Obrázek 4.6: DoA – trigonometrické odvození.

určíme, jaký čas odpovídá úhlu 0° a úhlu 180°

$$\tau_{0^\circ} = -\tau_{180^\circ} = \frac{d}{c}. \quad (4.13)$$

Číslo vzorku určíme pomocí vztahu $n = \text{floor}\left(\frac{\tau}{T_s}\right) = \text{floor}(\tau \cdot f_s)$, kde $n \in \mathbb{Z}$. Zaokrouhlujeme dolů, protože v opačném případě by po dosazení do rovnice (4.11) byl argument funkce arccos mimo interval $[-1, 1]$. Dosazením $\pm n$ do rovnice (4.11) dostaneme minimální a maximální dosažitelný úhel.

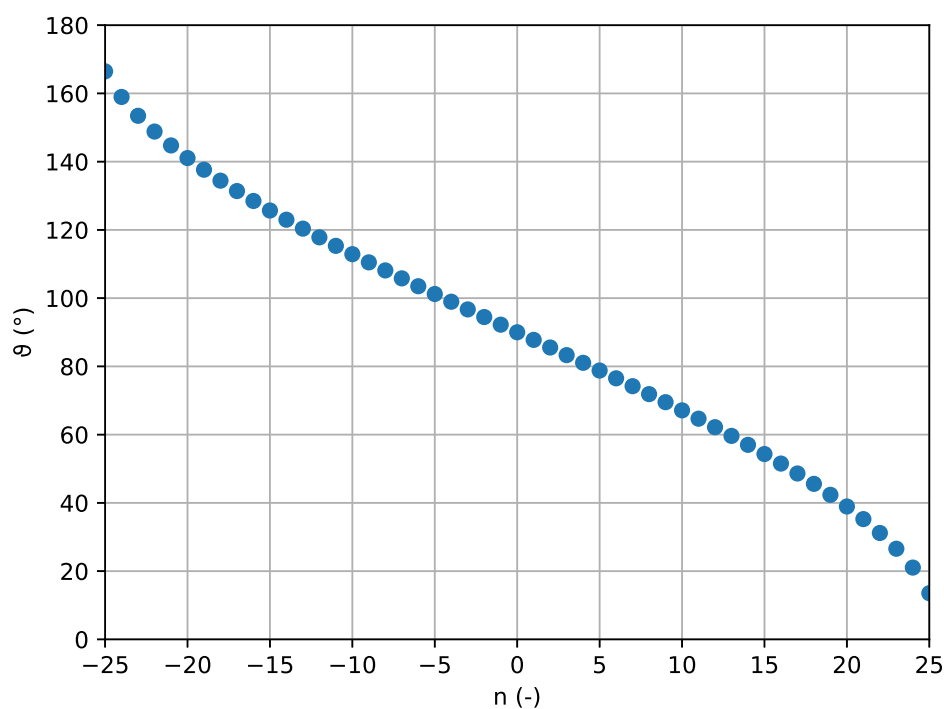
Pro vzorkovací frekvenci $f_s = 44,1 \text{ kHz}$ a rozteč mikrofonů $d = 0,2 \text{ m}$ je $n = \text{floor}(\tau_{0^\circ} \cdot f_s) = \text{floor}\left(\frac{d}{c} \cdot f_s\right) = \text{floor}\left(\frac{0,2}{343} \cdot 44\,100\right) = 25$. Dosazením $n = \pm 25$ do (4.11) dostaneme $\theta_{\min} \doteq 13,54^\circ$ a $\theta_{\max} \doteq 166,46^\circ$.

Vzhledem k tomu, že se v okrajových úhlech vyskytují i jiné nežádoucí jevy s významnějšími vlivy (např. tzv. *spatial aliasing*, kdy rozdílová dráha mezi signály je delší než $\frac{\lambda}{2}$ a nalezené časové zpoždění pak neodpovídá reálnému), není potřeba toto kritérium zohledňovat při volbě vzorkovací frekvence a rozteče mikrofonů.

4.7.2 Chyba odhadu úhlu způsobená vzorkováním

K odvození maximální chyby odhadu úhlu θ , způsobené vzorkováním, využijeme rovnice (4.11). Vzhledem k tomu, že funkce arccos není lineární, nebudou ani vypočtené úhly, odpovídající daným vzorkovým posunům, rozloženy lineárně (viz obr. 4.7).

Chybu Δ_{\max} , která udává maximální možnou odchylku od odhadu úhlu



Obrázek 4.7: Závislost úhlu θ na vzorkovém posunu mezi signály n pro $f_s = 44,1$ kHz a rozteč $d = 0,2$ m.

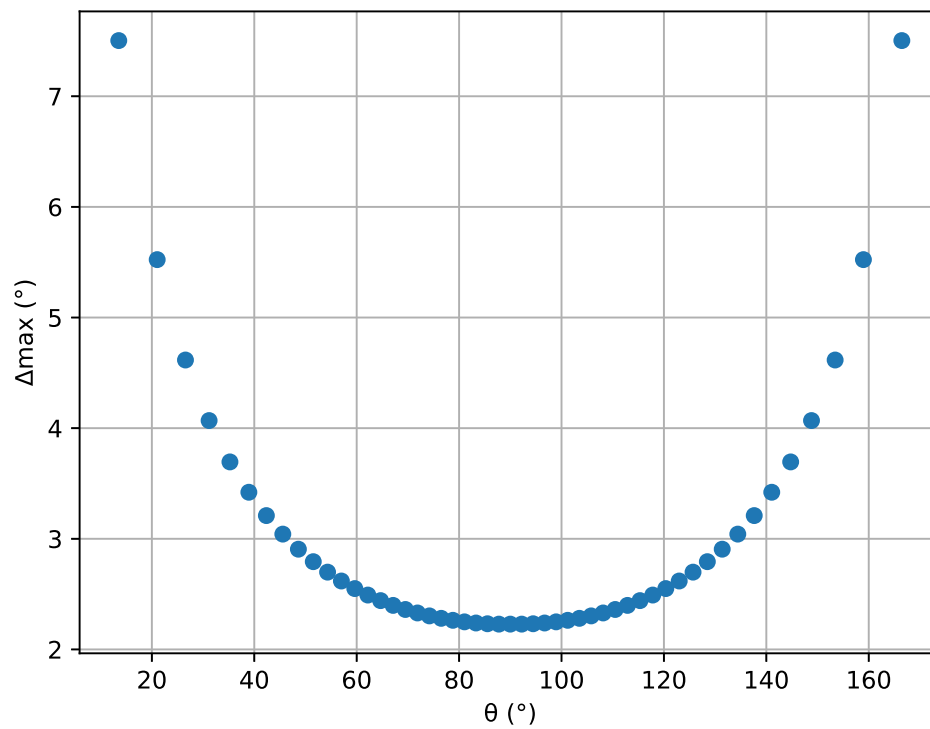
(způsobenou vzorkováním), definujeme jako

$$\Delta_{\max}[n] = \begin{cases} \theta[n] - \theta[n+1] & \text{pro } n \leq 0, \\ \theta[n-1] - \theta[n] & \text{jinak,} \end{cases} \quad (4.14)$$

kde $n \in \{N_{\min}, \dots, 0, \dots, N_{\max}\}$ a pro případ $f_s = 44,1$ kHz a $d = 0,2$ m je $N_{\min} = -25$ a $N_{\max} = 25$. Chyba definovaná dle rovnice (4.14) se vzorkovými posuny n , převedenými na úhel θ , je vykreslena na obrázku 4.8.

■ 4.7.3 Nejistota odhadu úhlu

Největší vliv na nejistotu odhadu úhlu má vzorkování omezenou vzorkovací frekvencí (viz 4.7.2). Dalšími vlivy může být rychlost zvuku odlišná od $c = 343$ m/s, nepřesná rozteč mikrofونů d nebo nesynchronnost kanálů vlivem vyčítání skrze sběrnici I²S (viz 3.9.1).



Obrázek 4.8: Maximální chyba způsobená vzorkováním pro $f_s = 44,1$ kHz a rozteč $d = 0,2$ m.

Kapitola 5

Webové rozhraní

Webové rozhraní umožňuje vizualizaci dat bez nutnosti instalace přídatných aplikací (kromě webového prohlížeče). Na realizaci takového rozhraní je potřeba webový server, kterým je v našem případě mikrokontrolér ESP32, a webová stránka, složená obvykle z minimálně jednoho **HTML** souboru a několika statických souborů, které dodávají funkcionalitu a vzhled.

5.1 ESP32 webový server

Pro zhotovení webového **HTTP** serveru byla využita volně dostupná knihovna **ESPAsyncWebServer** [34], která disponuje rozsáhlým množstvím funkcí a zároveň je asynchronní. Tato knihovna také umožňuje odesílání statických souborů z paměti **SPIFFS** (**SPI Flash File Storage**), která slouží pro uložení dat a souborů do **SPI** Flash paměti mikrokontroléru ESP32.

5.2 Komunikace mezi ESP32 a zařízením

Pro přenos statických souborů (**HTML**, **JS**, **CSS**) byl zvolen standardní protokol **HTTP**. Pro přenos informací o zdetekovaném úhlu byly implementovány obě technologie (**WS** a **SSE**), a protože není potřeba posílat data z klientského zařízení na server (mikrokontrolér), byla zvolena jednodušší technologie **SSE**.

Data odesílaná mikrokontrolérem ESP32 do webového prohlížeče jsou ve formátu $\theta; \Delta_{\max}$, kde θ je výsledek funkce $\theta[n]$ z rovnice (4.11) a Δ_{\max} je odhad chyby z rovnice (4.14).

5.2.1 HTTP

HTTP je zkratkou **Hypertext Transfer Protocol**. Jedná se o základní a nej-používanější protokol pro přenos hypertextových dokumentů, souborů a dat. V původních dvou verzích využívá protokolu **TCP** a nově, ve verzi 3, přichází s využitím protokolu **UDP**. Ve standardním provedení využívá portů 80 pro nešifrovanou a 443 pro šifrovanou komunikaci, avšak lze jej provozovat na jakémkoliv portu (po zvážení případné kolize s jinými službami).

5.2.2 Web Sockets

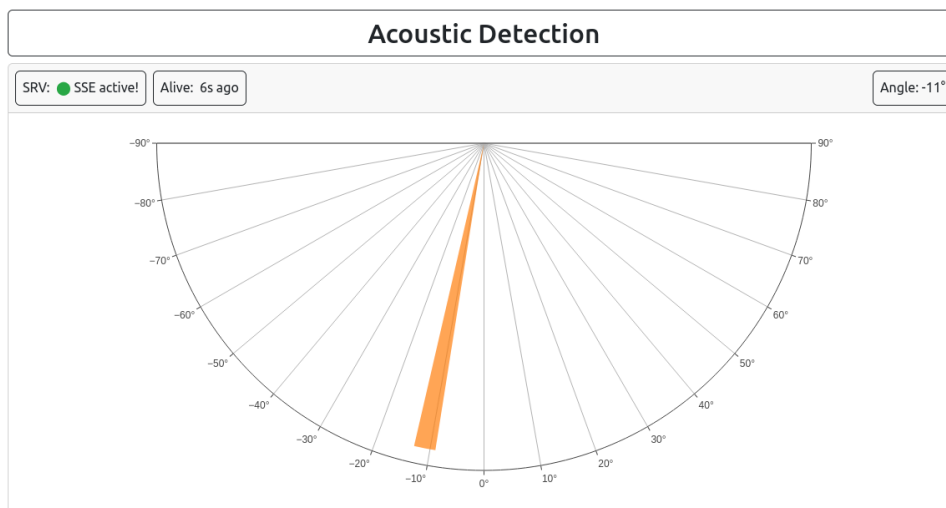
Web Sockets (WS) je komunikační protokol, který umožňuje obousměrnou komunikaci v reálném čase. Byl navržen tak, aby byl použitelný ve webových prohlížečích a na webových serverech. Oproti **HTTP** protokolu poskytuje komunikaci v reálném čase s nižší režii – není potřeba pro každá data vytvářet separátní request (požadavek), případně pro přijetí dat není potřeba periodicky provádět požadavky na server nebo využívat metod na pozdržení spojení (tzv. *long polling*). Pro navázání spojení nejdříve klient provede **HTTP** požadavek a poté server odešle odpověď na upgrade komunikace. Využívá standardních **HTTP** portů, zmíněných výše, tedy z hlediska firewallu by ve většině případů nemělo docházet k blokování.

5.2.3 Server-Sent Events

Server-Sent Events (SSE) je technologie, která umožňuje serveru posílat zprávy klientovi, tedy jednosměrnou komunikaci. Komunikace je založena na protokolu **HTTP**, kdy klient provede požadavek na server a ten poté ponechá komunikaci otevřenou a případně odešle klientovi zprávu, který ji přijme bez nutnosti uzavření komunikace. Oproti standardní **HTTP** komunikaci je rozdíl v tom, že se komunikace udržuje otevřená, dokud nedojde k nechtěnému uzavření nebo jedna ze stran ji neuzavře.

5.3 Klientská webová stránka

Webová stránka pro klientské zařízení je ukázána na obrázku 5.1.



Obrázek 5.1: Webové rozhraní.

Zobrazený graf byl vykreslen pomocí knihovny **Plotly.js** [35], která podporuje přes 40 typů nejrůznějších grafů. Pro naše použití disponuje možností vykreslení pouze poloviny polárního grafu a zároveň umožňuje aktualizovat

data v grafu bez nutnosti destrukce a znovuvytvoření celého **JS** objektu. Pozice zobrazeného paprsku je dána odhadem úhlu θ a šířka je dána dvojnásobkem odhadu chyby $2\Delta_{\max}$ (Δ_{\max} na každou stranu).

Vzhled stránky byl nastýlován pomocí stylovací šablony **Bootstrap** [36], která poskytuje rozsáhlé množství předpřipravených komponentů na tvorbu webových stránek. Knihovna je napsána pomocí skriptovacího jazyka **Sass**, který je nadstavbou stylovacího jazyka **CSS** a pro vytvoření výsledné šablony je do jazyka **CSS** také zkompileován.

Pro lepší přehlednost je úhel $\theta \in [0^\circ, 180^\circ]$ převeden na úhel vůči ose y pomocí $\beta = \theta - 90$, tedy $\beta \in [-90^\circ, 90^\circ]$.

■ 5.3.1 Zmenšení velikosti webové stránky

Vzhledem k tomu, že velikost statických souborů potřebných pro vykreslení webového rozhraní převyšuje kapacitu paměti **SPIFFS** mikrokontroléru ESP32, bylo potřeba provést úpravy na zmenšení velikosti. V následujících sekcích budou popsány metody, kterými byla zmenšena velikost jednotlivých statických souborů.

■ Zmenšení velikosti šablony Bootstrap

Velikost stylovací šablony (knihovny) Bootstrap lze zmenšit odstraněním nevyužitých komponentů. Toho lze dosáhnout použitím nástroje **PurgeCSS** [37], který zanalyzuje **HTML** soubor, využívající danou **CSS** šablonu, a vytvoří verzi šablony obsahující pouze komponenty použité v zadaném **HTML** souboru. Díky tomu lze z původních 220 kB dostat knihovnu o velikosti necelých 10 kB.

■ Zmenšení velikosti knihovny Plotly.js

Největším problémem, co se týče velikosti, je knihovna Plotly.js, kde zminifikovaná verze dosahuje velikosti přibližně 3,5 MB. Velikost knihovny lze snížit vytvořením vlastního balíku knihovny (tzv. *custom bundle*), který bude obsahovat pouze potřebné části knihovny Plotly.js. Toho lze dosáhnout použitím předem připraveného skriptu `custom-bundle` pro správce balíčků **npm** [38]. Po této operaci dostaneme soubor o velikosti přibližně 960 kB.

■ Minifikace JS souborů

Další možností, jak zmenšit velikost statických souborů, je minifikace **JS** (**JavaScript**) souborů. Minifikace spočívá v odstranění nepotřebných znaků (tedy především mezery, tabulátory, středníky nebo komentáře) a v případě pokročilé minifikace také zkrácení názvů proměnných a funkcí, díky čemuž je možné dosáhnout výrazného zmenšení velikosti. V našem případě není úspora velikosti příliš znatelná, ale u rozsáhlých knihoven (např. Plotly.js) se může jednat až o několikanásobné zmenšení.

■ Gzip komprese

Gzip (zkratka GNU zip) je program na kompresi dat. Jedná se o jeden z kompresních formátů, který lze použít v komunikačním protokolu **HTTP**. Ještě před nahráním souborů do paměti **SPIFFS** mikrokontroléru ESP32 je provedena gzip komprese pro zmenšení obsahu a webový server v mikrokontroléru ESP32 poté odesílá rovnou zkomprimované soubory. Díky této kompresi se z původní velikosti téměř 1 MB dostaneme na velikost necelých 340 kB pro celou webovou stránku.

Kapitola 6

Návrh HW

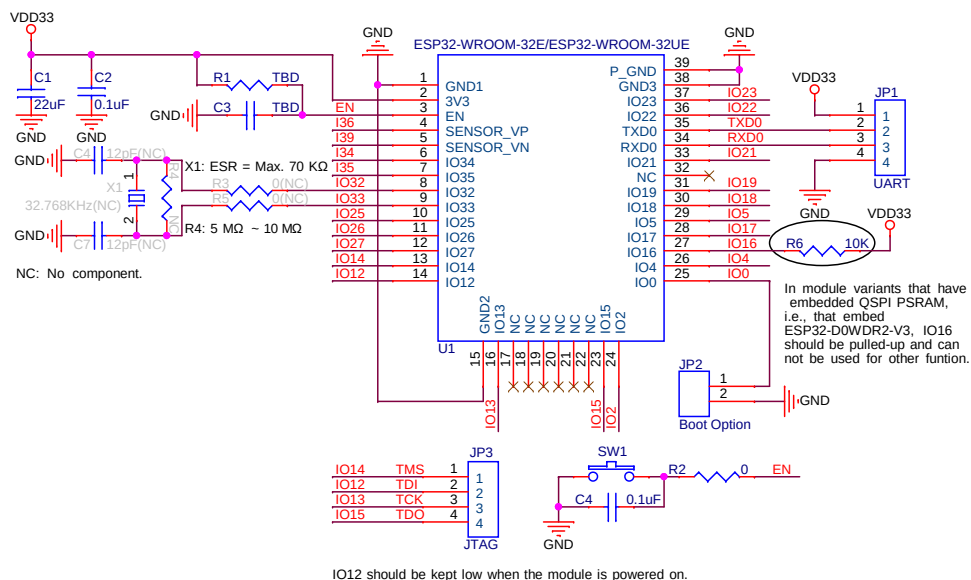
Tato kapitola se zabývá návrhem zapojení, plošného spoje a postupem výroby finálního zařízení.

6.1 Zapojení

V následujících sekcích budou probírána zapojení jednotlivých prvků, tedy mikrokontroléru ESP32, mikrofonů SPH0645LM4H a napájení. Celé schéma zapojení lze nalézt v příloze B.

6.1.1 Mikrokontrolér ESP32

Mikrokontrolér byl zapojen dle doporučeného zapojení na obr. 6.1 z data-sheetu [39]. Pro naše účely není potřeba připojovat část s krystalem X1.

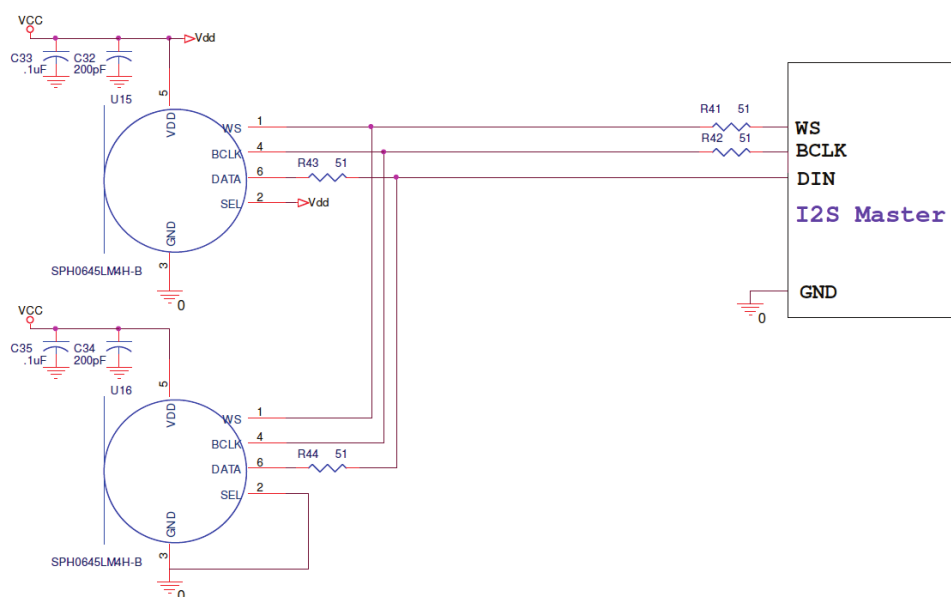


Obrázek 6.1: Doporučené zapojení mikrokontroléru ESP32-WROOM-32E. Převzato z [39].

Požadavek na napájení čipu ESP32 je napájecí napětí 3,0 V až 3,6 V (ideálně 3,3 V) a proud minimálně 0,5 A [39, s. 14].

6.1.2 Mikrofony SPH0645LM4H

Mikrofony SPH0645LM4H (konkrétně SPH0645LM4H-B) byly zapojeny dle doporučeného zapojení na obr. 6.2 z datasheetu [28]. Kondenzátory uvedené ve schématu 6.2 (C32 až C35) by měly být umístěny v blízkosti mikrofonů. Kondenzátory C33 a C35 (velikost 100 nF) slouží jako blokovací kondenzátory. Kondenzátory C32 a C34 (velikost 200 pF) slouží jako filtrovací a měly by být umístěny co nejbližší odpovídajícím mikrofonům. Rezistory R41 až R44 (velikost 51 Ω) slouží jako k zakončovací, aby nedocházelo k odrazům na vedení, které by mohly způsobit rušení.



Obrázek 6.2: Doporučené zapojení dvojice mikrofonů SPH0645LM4H-B. Převzato z [28].

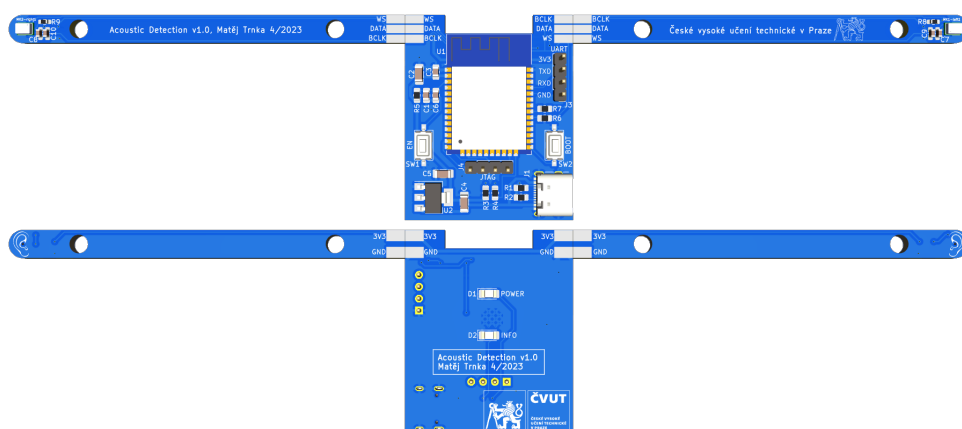
Napájecí napětí mikrofonu SPH0645LM4H-B je 1,62 V až 3,6 V a proudový odběr ve standardním režimu je 600 μ A [28, s. 2].

6.1.3 Napájení

Jako způsob konverze z napájecího napětí o velikosti 5 V na požadované napájecí napětí mikrokontroléru ESP32 a mikrofonů SPH0645LM4H o velikosti 3,3 V byl z důvodu snadné použitelnosti a nízké ceny využit integrovaný stabilizátor. ESP32 požaduje minimální proud 0,5 A a odběr mikrofonů je vůči tomu zanedbatelný. Z těchto důvodů byl stanoven požadavek na napětí 3,3 V a proud alespoň 1 A. Zvážením aktuální dostupnosti byl vybrán stabilizátor LDL1117S33R od firmy STMicroelectronics v pouzdře SOT-223.

6.2 Plošný spoj

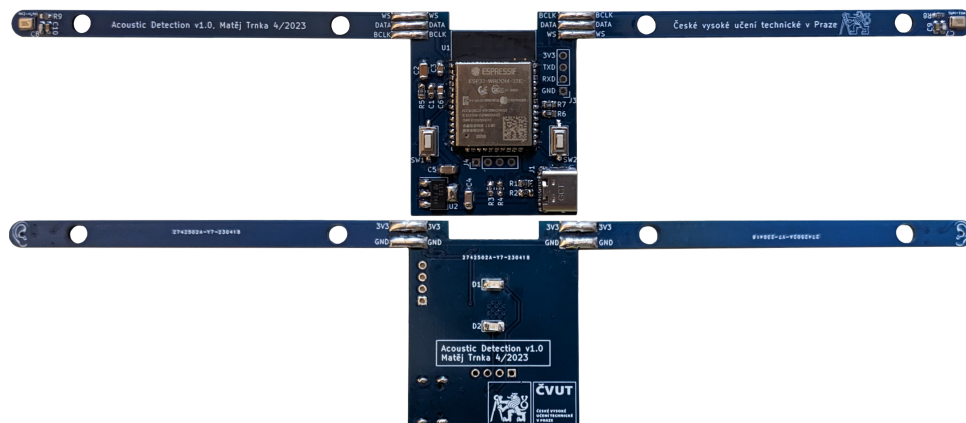
Z důvodu snazší výroby a následného osazování bylo zvoleno rozdělení plošného spoje na tři části, kde hlavní část obsahuje mikrokontrolér ESP32 spolu s napájením a dvě podlouhlá ramena obsahují na jednom ze svých konců mikrofon SPH0645LM4H spolu s potřebnými součástkami. Jako propojení těchto částí bylo zvoleno využití obdélníkových pájitelných kontaktů. Z jedné strany **DPS** jsou pro každé rameno a odpovídající část hlavní desky použity dvojice napájecích kontaktů, zatímco z druhé strany jsou pro každé rameno a odpovídající část hlavní desky použity trojice datových kontaktů. Při kompletaci poté stačí kontakty pouze proletovat a tím propojit odpovídající části zařízení. Navržený plošný spoj lze nalézt v příloze **C**. Snímek 3D náhledu navrženého plošného spoje je zobrazen na obrázku **6.3**.



Obrázek 6.3: Snímek 3D náhledu navrženého zařízení.

6.3 Praktické zhotovení

Pro osazení použitých součástek na vyrobený plošný spoj bylo využito metody letování pomocí pájecí pasty a nahřívané desky. Proletování propojovacích plošek bylo provedeno pomocí mikro páječky a cínové pájky v drátové podobě. Zhotovené zařízení lze vidět na obrázku **6.4**.

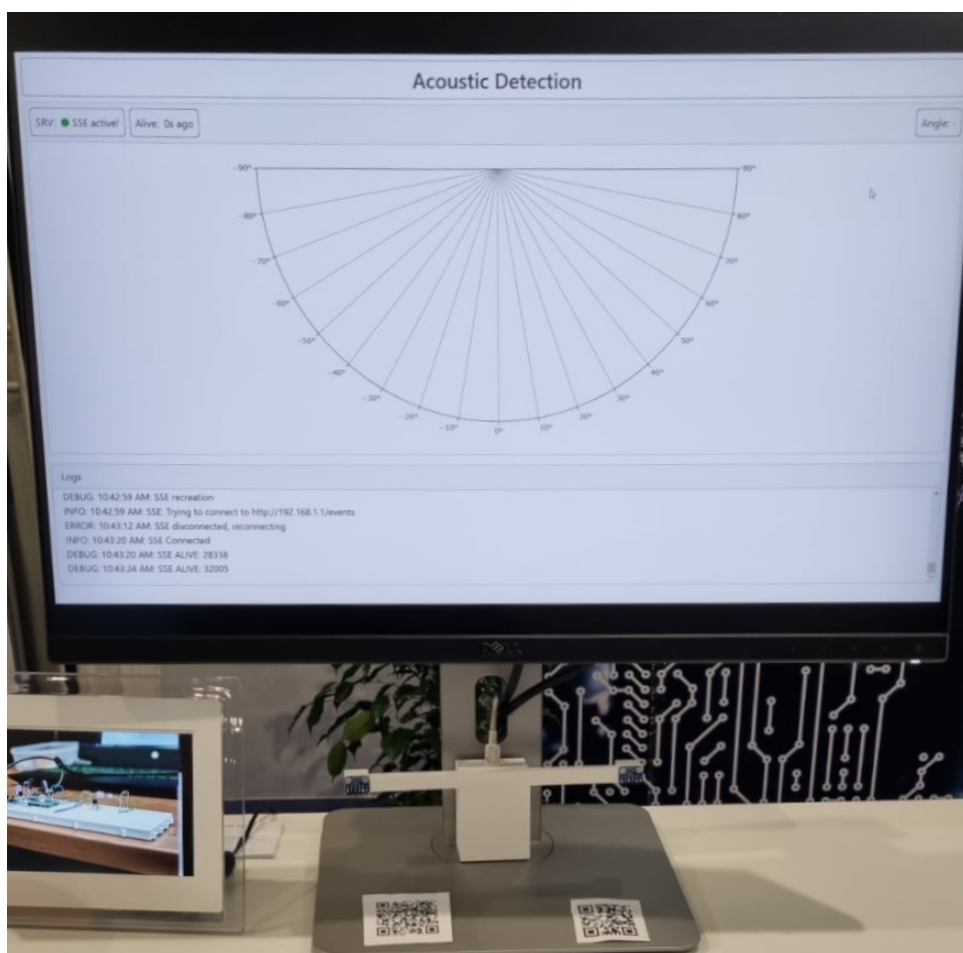


Obrázek 6.4: Zhotovené zařízení.

Kapitola 7

Ověření funkčnosti

Během ověření funkčnosti byla současně testována jak akustická detekce, tak i akustická lokalizace. Jako zdroje impulzní akustické události bylo využito především běžného tlesknutí. I přes nepříznivé podmínky (místnost a hluk) se zařízení ukázalo jako funkční a schopné adaptace na hlučné pozadí, kterým byla především puštěná hudba.



Obrázek 7.1: Testovací prototyp na veletrhu AMPER 2023.

Funkčnost testovacího prototypu byla úspěšně ověřena na veletrhu AMPER 2023. Foto z veletrhu je na obr. 7.1.

7.1 Zhodnocení výsledků

Během provedených testů se zařízení ukázalo jako funkční, a to i v případě, že bylo použito v běžné místnosti s hlukem na pozadí. Zvolené metody pro detekci a lokalizaci impulzních akustických událostí se ukázaly jako postačující a vyhovující požadavkům této práce.

Kapitola 8

Závěr

V předložené bakalářské práci byla ukázána teoretická i praktická část zhotovení demonstračního zařízení pro detekci a přibližnou lokalizaci směru impulzních akustických událostí. Byly zde vysvětleny teoretické předpoklady, především záležitosti týkající se zvuku a akustické lokalizace.

Na začátku byla uvedena problematika detekce impulzních akustických událostí a v praktické části byla rozebrána zvolená metoda založená na rekurentním výpočtu střední hodnoty. Vzhledem k požadavku detekce všech impulzních akustických událostí se tato metoda ukázala jako postačující.

Pro nalezení zpoždění mezi signály byly uvedeny metody založené na vzájemné korelační funkci a její upravené verzi GCC [9], adaptivním filtru [10] a výpočtu skrze spektrum [11]. Zvolená metoda pomocí vychýleného odhadu vzájemné korelační funkce se ukázala jako postačující a proto byla použita. Jako problém se ukázaly falešné akustické události způsobené odrazy v dozvukovém prostředí. Pro vyřešení tohoto problému byla uvedena metoda DATEMM [8] a vzhledem k nízkým požadavkům byla zvolena metoda založená na časové prodlevě. Tato metoda, i přes svou jednoduchost, poskytuje uspokojivé výsledky.

V části 3.6 bylo provedeno odvození rovnice DoA pomocí asymptoty hyperbolické funkce a v části 4.7 potom trigonometrické odvození při zanedbání zakřivení vlnoploch. Část 4.7.2 se zabývala chybou, jakou způsobuje vzorkování s omezenou vzorkovací frekvencí, a bylo ukázáno, že tato chyba není lineární a její velikost je závislá na úhlu θ .

Část 3.8 se zabývala popisem mikrokontroléru ESP32, část 3.9 poté popisem mikrofonu SPH0645LM4H a sběrnice I²S. V části 4.3 pak bylo popsáno propojení těchto komponentů a způsob zpracování vyčtených hodnot.

V kapitole 5 byla popsána tvorba webového rozhraní, zvolené technologie a knihovny. Kapitola 6 shrnula návrh plošného spoje a výrobu finálního zařízení.

Na závěr bylo uvedeno ověření funkčnosti, v rámci kterého se navržené a zhotovené zařízení ukázalo jako funkční a splňující zadání bakalářské práce. Možné zlepšení lze nalézt v implementaci více komplexních algoritmů pro nalezení časového zpoždění mezi signály, především pak rozlišení několika zdrojů akustických událostí a odrazů.

Seznam použité literatury

- [1] J. Svatos a J. Holub, „Impulse Acoustic Event Detection, Classification, and Localization System“, *IEEE Transactions on Instrumentation and Measurement*, roč. 72, s. 1–15, 2023. DOI: [10.1109/TIM.2023.3252631](https://doi.org/10.1109/TIM.2023.3252631).
- [2] X. Chang, C. Yang, J. Wu, X. Shi a Z. Shi, „A Surveillance System for Drone Localization and Tracking Using Acoustic Arrays“, in *2018 IEEE 10th Sensor Array and Multichannel Signal Processing Workshop (SAM)*, ISSN: 2151-870X, čvc. 2018, s. 573–577. DOI: [10.1109/SAM.2018.8448409](https://doi.org/10.1109/SAM.2018.8448409).
- [3] „Sound Fields: Free versus Diffuse Field, Near versus Far Field“. (29. čvc. 2020), URL: <https://community.sw.siemens.com/s/article/sound-fields-free-versus-diffuse-field-near-versus-far-field> (cit. 20.05.2023).
- [4] N. H. Fletcher, „SHOCK WAVES AND THE SOUND OF a HAND-CLAP — a SIMPLE MODEL“, *Acoustics Australia*,
- [5] J. Kotus, K. Lopatka a A. Czyzewski, „Detection and localization of selected acoustic events in acoustic field for smart surveillance applications“, *Multimedia Tools and Applications*, roč. 68, č. 1, s. 5–21, 1. led. 2014, ISSN: 1573-7721. DOI: [10.1007/s11042-012-1183-0](https://doi.org/10.1007/s11042-012-1183-0). URL: <https://doi.org/10.1007/s11042-012-1183-0> (cit. 15.05.2023).
- [6] J. Svatos a J. Holub, „Smart Acoustic Sensor“, in *2019 IEEE 5th International forum on Research and Technology for Society and Industry (RTSI)*, ISSN: 2687-6817, zář. 2019, s. 161–165. DOI: [10.1109/RTSI.2019.8895591](https://doi.org/10.1109/RTSI.2019.8895591).
- [7] S. Corps, *T3 aircraft sound locator*, 1. lis. 1927. URL: https://commons.wikimedia.org/wiki/File:T3_sound_locator.jpg (cit. 04.05.2023).
- [8] J. Scheuing a B. Yang, „Disambiguation of TDOA Estimates in Multi-Path Multi-Source Environments (DATEMM)“, in *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, ISSN: 2379-190X, sv. 4, květ. 2006, s. IV–IV. DOI: [10.1109/ICASSP.2006.1661099](https://doi.org/10.1109/ICASSP.2006.1661099).

- [9] C. Knapp a G. Carter, „The generalized correlation method for estimation of time delay“, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, roč. 24, č. 4, s. 320–327, srp. 1976, Conference Name: IEEE Transactions on Acoustics, Speech, and Signal Processing, ISSN: 0096-3518. DOI: [10.1109/TASSP.1976.1162830](https://doi.org/10.1109/TASSP.1976.1162830).
- [10] C. F. Scola a M. D. B. Ortega, „Direction of arrival estimation: A two microphones approach“, 2010. URL: <https://www.semanticscholar.org/paper/Direction-of-arrival-estimation-%3A-A-two-microphones-Scola-Ortega/4d6bcea5d6dac27b80887d94c464bfd997595a95#extracted> (cit. 14. 05. 2023).
- [11] G. Fabregat, J. A. Belloch, J. M. Badía a M. Cobos, „Design and Implementation of Acoustic Source Localization on a Low-Cost IoT Edge Platform“, *IEEE Transactions on Circuits and Systems II: Express Briefs*, roč. 67, č. 12, s. 3547–3551, pros. 2020, Conference Name: IEEE Transactions on Circuits and Systems II: Express Briefs, ISSN: 1558-3791. DOI: [10.1109/TCSII.2020.2986296](https://doi.org/10.1109/TCSII.2020.2986296). (cit. 14. 05. 2023).
- [12] „Generalized cross-correlation - MATLAB gccphat“. (c1994-2023), URL: <https://www.mathworks.com/help/phased/ref/gccphat.html> (cit. 10. 05. 2023).
- [13] G. Jacovitti a G. Scarano, „Discrete time techniques for time delay estimation“, *IEEE Transactions on Signal Processing*, roč. 41, č. 2, s. 525–533, ún. 1993, Conference Name: IEEE Transactions on Signal Processing, ISSN: 1941-0476. DOI: [10.1109/78.193195](https://doi.org/10.1109/78.193195).
- [14] O. Guschina, „Refining Time Delay Estimate of Complex Signal Using Polynomial Interpolation in Time Domain“, in *2021 Systems of Signals Generating and Processing in the Field of on Board Communications*, břez. 2021, s. 1–6. DOI: [10.1109/IEEECONF51389.2021.9416076](https://doi.org/10.1109/IEEECONF51389.2021.9416076).
- [15] B. Kwon, Y. Park a Y.-s. Park, „Analysis of the GCC-PHAT technique for multiple sources“, in *ICCAS 2010*, říj. 2010, s. 2070–2073. DOI: [10.1109/ICCAS.2010.5670137](https://doi.org/10.1109/ICCAS.2010.5670137).
- [16] J. H. DiBiase, „A high-accuracy, low-latency technique for talker localization in reverberant environments using microphone arrays“, květ. 2000. (cit. 14. 05. 2023).
- [17] G. F. Simmons, *Calculus with analytic geometry*, 2nd ed. New York: McGraw-Hill, 1996, 887 s., ISBN: 978-0-07-057642-1.
- [18] P. Malcovati a A. Baschiroto, „The evolution of integrated interfaces for MEMS microphones“, *Micromachines*, roč. 9, č. 7, s. 323, čvc. 2018, Number: 7 Publisher: Multidisciplinary Digital Publishing Institute, ISSN: 2072-666X. DOI: [10.3390/mi9070323](https://doi.org/10.3390/mi9070323). URL: <https://www.mdpi.com/2072-666X/9/7/323> (cit. 09. 05. 2023).
- [19] „What is MEMS Technology?“, URL: <https://www.mems-exchange.org/MEMS/what-is.html> (cit. 05. 12. 2022).

- [20] P. Loeppert a S. Lee, „SiSonic™ - THE FIRST COMMERCIALIZED MEMS MICROPHONE“, in *2006 Solid-State, Actuators, and Microsystems Workshop Technical Digest*, Hilton Head, South Carolina, USA: Transducer Research Foundation, Inc., 4. čvn. 2006, s. 27–30, ISBN: 978-0-9640024-6-3. DOI: [10.31438/trf.hh2006.7](https://doi.org/10.31438/trf.hh2006.7). URL: https://transducer-research-foundation.org/technical_digests/HiltonHead_2006/hh2006_0027.pdf (cit. 09.05.2023).
- [21] „What Is A MEMS Microphone? (Micro-Electro-Mechanical Systems)“. (2020), URL: <https://mynewmicrophone.com/what-is-a-mems-micro-electro-mechanical-systems-microphone/> (cit. 05.12.2022).
- [22] N. Peña García, L. Aguilera-Cortés, M. González-Palacios, J.-P. Raskin a A. Herrera-May, „Design and Modeling of a MEMS Dual-Backplate Capacitive Microphone with Spring-Supported Diaphragm for Mobile Device Applications“, *Sensors*, roč. 18, s. 3545, 19. říj. 2018. DOI: [10.3390/s18103545](https://doi.org/10.3390/s18103545).
- [23] „ESP32 Series of Modules“, URL: <https://www.espressif.com/en/products/modules/esp32> (cit. 26.01.2023).
- [24] C. Greening. „ESP32 I2S DMA Settings“. (20. břez. 2021), URL: <https://www.atomic14.com/2021/04/20/esp32-i2s-dma-buf-len-buf-count.html> (cit. 26.01.2023).
- [25] „Espressif 32 — PlatformIO v6.1 documentation“. (2022), URL: <https://docs.platformio.org/en/stable/platforms/espressif32.html#frameworks> (cit. 16.05.2023).
- [26] „FreeRTOS - market leading RTOS (real time operating system) for embedded systems with internet of things extensions“, FreeRTOS, URL: <https://www.freertos.org/index.html> (cit. 07.05.2023).
- [27] „FreeRTOS (Overview) - ESP32 - ESP-IDF Programming Guide latest documentation“, URL: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/system/freertos.html> (cit. 08.05.2023).
- [28] „SPH0645LM4H-B Rev C Datasheet“, URL: https://cdn-shop.adafruit.com/product-files/4455/4455_SPH0645LM4H-B.pdf (cit. 06.12.2022).
- [29] „I2S bus specification“. (ún. 1986), URL: <https://www.sparkfun.com/datasheets/BreakoutBoards/I2SBUS.pdf> (cit. 05.12.2022).
- [30] „Inter-IC Sound (I2S)“. (c2016 - 2022), URL: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/peripherals/i2s.html> (cit. 06.12.2022).
- [31] C. Greening. „ESP32 I2S Audio“. (c2023), URL: https://github.com/atomic14/esp32_audio (cit. 26.01.2023).
- [32] J. O. Smith, „DC Blocker“, in *Introduction to Digital Filters with Audio Applications*, 2007. URL: https://ccrma.stanford.edu/~jos/filters/DC_Blocker.html (cit. 06.12.2022).

- [33] „Cross-correlation - MATLAB xcorr“. (c1994-2023), URL: <https://www.mathworks.com/help/matlab/ref/xcorr.html> (cit. 24. 01. 2023).
- [34] M. N. Dev, *ESPAsyncWebServer*, original-date: 2015-12-19T16:53:57Z, 13. květ. 2023. URL: <https://github.com/me-no-dev/ESPAsyncWebServer> (cit. 14. 05. 2023).
- [35] „Plotly.js“, URL: <https://plotly.com/javascript/> (cit. 14. 05. 2023).
- [36] „Bootstrap“, URL: <https://getbootstrap.com/> (cit. 14. 05. 2023).
- [37] „PurgeCSS - Remove unused CSS | PurgeCSS“, URL: <https://purgecss.com/> (cit. 19. 05. 2023).
- [38] *Plotly.js - Custom bundle*, original-date: 2015-11-05T23:27:17Z, 26. dub. 2023. URL: https://github.com/plotly/plotly.js/blob/master/CUSTOM_BUNDLE.md (cit. 26. 04. 2023).
- [39] „ESP32WROOM32E Datasheet“, URL: https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e_esp32-wroom-32ue_datasheet_en.pdf (cit. 05. 04. 2023).



Příloha A

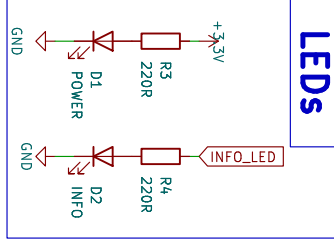
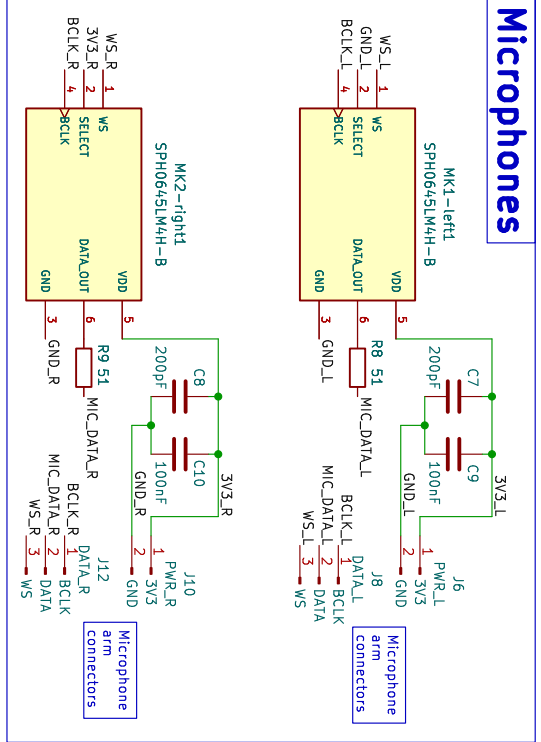
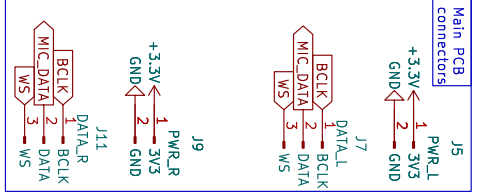
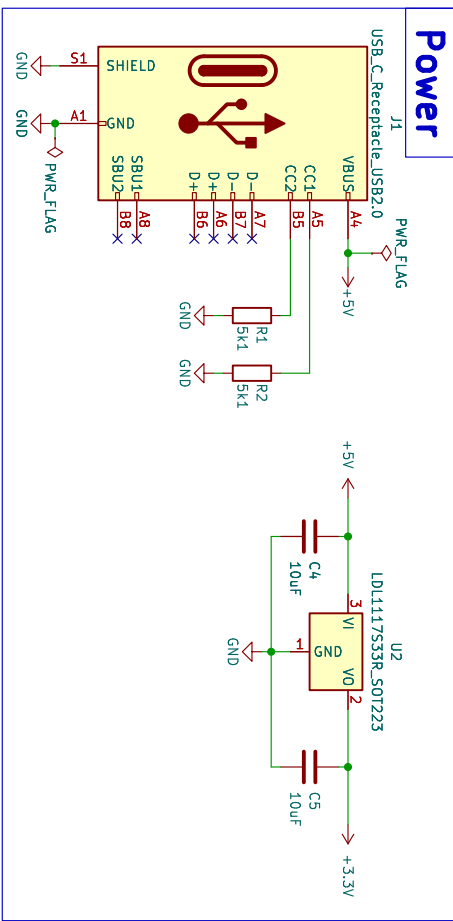
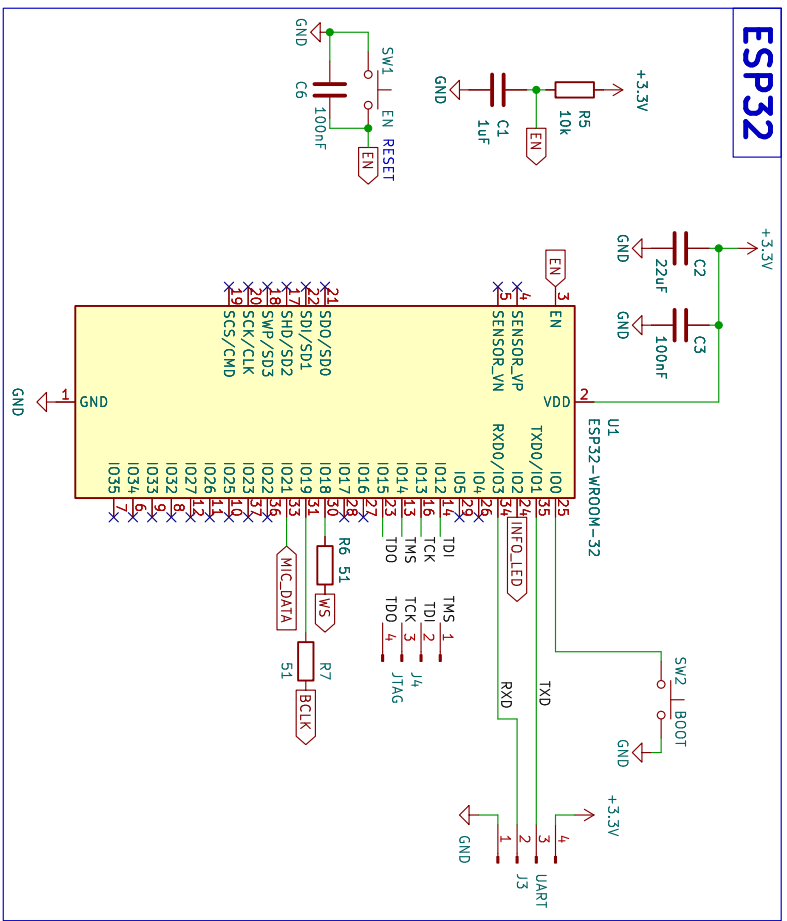
Zdrojový kód

Zdrojový kód lze nalézt v repozitáři na adrese <https://github.com/matrn/AcousticDetection>.



Příloha B

Schéma zapojení



Added for ERC

3V3_L PWR_FLAG
 GND_L PWR_FLAG
 3V3_R PWR_FLAG
 GND_R PWR_FLAG

Matěj Trnka, CTU FEL, Department of Measurement

Sheet: /
 File: AcousticDetection_HW.kicad_sch

Title: Acoustic Detection

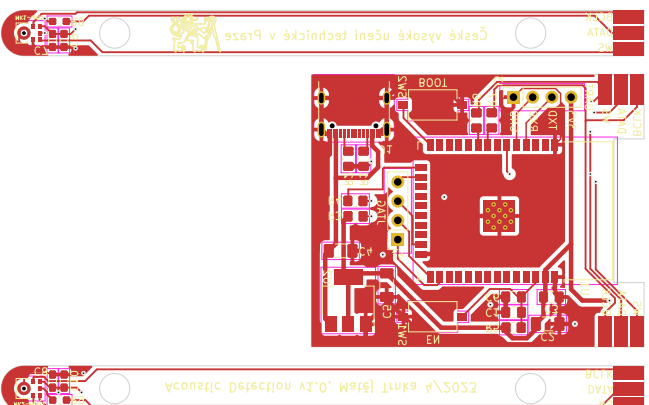
Size: A4 Date: 2023-04-10
 Kicad E.D.A. Kicad 7.0.2-6445011f42-172-ubuntu22.04.1

Rev: Id: 1/1



Příloha C

Plošný spoj



Matěj Trnka, CTU FEL, Department of Measurement

Sheet:

File: AcousticDetection_HW.kicad_pcb

Title: Acoustic Detection

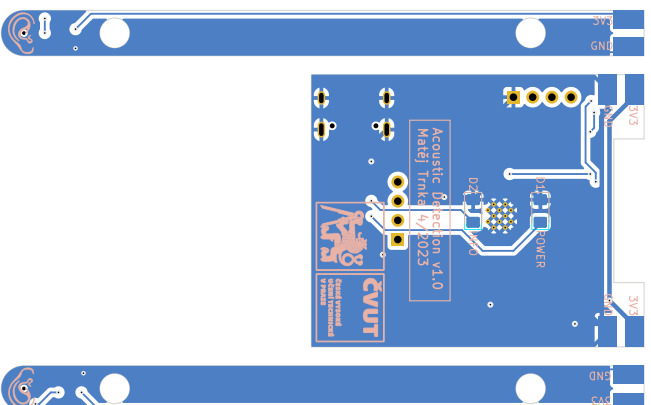
Size: A4

Date: 2023-04-10

KiCad E.D.A. KiCad 7.0.2-6a45011f42-172-ubuntu22.04.1

Rev:

Id: 1/1



Matěj Trnka, CTU FEL, Department of Measurement

Sheet:

File: AcousticDetection_HW.kicad_pcb

Title: Acoustic Detection

Size: A4

Date: 2023-04-10

KiCad E.D.A. KiCad 7.0.2-6445011f42-172-ubuntu22.04.1

Rev:
Id: 1/1