



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**

FACULTY OF INFORMATION TECHNOLOGY

**ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ**

DEPARTMENT OF COMPUTER SYSTEMS

**LOKALIZACE OBJEKTŮ V BEZDRÁTOVÉ  
KOMUNIKAČNÍ SÍTI NA BÁZI WIFI**

LOCATION ESTIMATION OF OBJECTS IN WIFI COMMUNICATION NETWORK

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Bc. JAN KOHOUTEK**

**VEDOUcí PRÁCE**

SUPERVISOR

**Ing. VÁCLAV ŠIMEK**

BRNO 2020

## Zadání diplomové práce



Student: **Kohoutek Jan, Bc.**  
Program: Informační technologie    Obor: Počítačové a vestavěné systémy  
Název: **Lokalizace objektů v bezdrátové komunikační síti na bázi WiFi**  
**Location Estimation of Objects in WiFi Communication Network**  
Kategorie: Vestavěné systémy

### Zadání:

1. Zpracujte přehledovou studii základních principů lokalizace zařízení v bezdrátové síti WiFi.
2. Zvolte vhodnou techniku a navrhnete detailní koncepci systému pro určování polohy objektů v interiéru v komunikační síti na bázi WiFi.
3. Vyberte vhodné komponenty a v návrhovém systému připravte schéma zapojení lokalizačních modulů.
4. Realizujte desky plošných spojů lokalizačních modulů v potřebném množství s ohledem na zvolenou techniku lokalizace.
5. Implementujte potřebný komunikační firmware modulů a vytvořte jednoduché komunikační rozhraní pro správu a monitorování činnosti lokalizačního systému.
6. Proveďte důkladné otestování navrženého systému v reálných podmínkách.
7. Zhodnoťte dosažené výsledky a diskutujte možnosti dalšího rozšíření.

### Literatura:

- Dle pokynů vedoucího.

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Šimek Václav, Ing.**  
Vedoucí ústavu: Sekanina Lukáš, prof. Ing., Ph.D.  
Datum zadání: 1. listopadu 2019  
Datum odevzdání: 31. července 2020  
Datum schválení: 25. října 2019

## Abstrakt

Tato práce si klade za cíl návrh a implementaci systému pro lokalizaci zařízení v bezdrátové síti Wi-Fi, v prostorách interiéru malé, či střední firmy. Zabývá se používanými technologiemi a poskytuje vlastní řešení sensorové sítě a centrálního zpracovatelského uzlu. V závěru uvádí dosažené výsledky a zhodnocuje splnění zadaných požadavků.

## Abstract

The purpose of the thesis is to design and physically implement a system for location of devices in a Wi-Fi based networks deployed in the interior of small and medium-sized enterprises. First of all, the attention is given to the survey of commonly available technologies suitable for determination of the object position. Then, based on these findings, architecture of a location system comprising an array of sensor nodes and central processing unit is proposed. In conclusion, the results are presented and the fulfillment of the requirements is evaluated.

## Klíčová slova

Wi-Fi, lokalizace, bezdrátová síť, interiér

## Keywords

Wi-Fi, localization, wireless network, interior

## Citace

KOHOUTEK, Jan. *Lokalizace objektů v bezdrátové komunikační síti na bázi WiFi*. Brno, 2020. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Václav Šimek

# Lokalizace objektů v bezdrátové komunikační síti na bázi WiFi

## Prohlášení

Prohlašuji, že jsem diplomovou práci vypracoval samostatně pod vedením pana Ing. Václava Šimka a že jsem uvedl všechny literární prameny a publikace, ze kterých jsem čerpal.

.....

Jan Kohoutek

29. července 2020

## Poděkování

Rád bych využil této části a poděkoval svému lektorovi, panu Ing. Václavu Šimkovi, za jeho přínosné postřehy, rady, spolupráci a trpělivost. Dále bych rád také poděkoval panu Ing. Milanu Doubkovi, který byl ochotný vypsát firemní zadání a poskytnout zdroje i prostory pro zdárné dokončení práce.

# Obsah

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Úvod</b>  | <b>3</b>  |
| <b>2</b> | <b>Teorie lokalizace subjektů</b>                                | <b>5</b>  |
| 2.1      | Obecné principy lokalizace . . . . .                             | 6         |
| 2.1.1    | Trilaterace . . . . .  | 6         |
| 2.1.2    | Triangulace . . . . .  | 7         |
| 2.1.3    | Fingerprinting . . . . .   | 7         |
| 2.2      | Techniky lokalizace využívané v bezdrátových sítích . . . . .    | 7         |
| 2.2.1    | RSS – síla signálu . . . . .                                     | 8         |
| 2.2.2    | TOA – čas doručení . . . . .                                     | 9         |
| 2.2.3    | AOA – úhel dopadu signálu . . . . .                              | 9         |
| 2.3      | Vylepšené techniky se zaměřením na Wi-Fi síť . . . . .           | 10        |
| 2.3.1    | OFDM . . . . .   | 10        |
| 2.3.2    | CSI . . . . .  | 11        |
| 2.3.3    | Existující lokalizační CSI systémy a jejich vlastnosti . . . . . | 12        |
| <b>3</b> | <b>Návrh lokalizačního systému</b>                               | <b>17</b> |
| 3.1      | Specifikace požadavků . . . . .                                  | 17        |
| 3.2      | Návrh architektury systému . . . . .                             | 18        |
| <b>4</b> | <b>Realizace systému</b>   | <b>21</b> |
| 4.1      | Aspekty nasazení a použití lokalizačního systému . . . . .       | 21        |
| 4.2      | Detaily součástí systému . . . . .                               | 22        |
| 4.2.1    | Centrální serverová instance . . . . .                           | 23        |
| 4.2.2    | Bezdrátový senzor v roli přístupového bodu . . . . .             | 29        |
| 4.2.3    | Aktivní přispění lokalizovaného zařízení . . . . .               | 34        |
| <b>5</b> | <b>Testování</b>   | <b>37</b> |
| 5.1      | Testovací prostředí . . . . .                                    | 37        |
| 5.2      | Naměřené výsledky . . . . .                                      | 40        |
| 5.2.1    | Pasivní metoda lokalizace . . . . .                              | 40        |
| 5.2.2    | Aktivní metoda lokalizace . . . . .                              | 41        |
| 5.2.3    | Další testované efekty . . . . .                                 | 42        |
| 5.3      | Zhodnocení výsledků . . . . .                                    | 44        |
| <b>6</b> | <b>Závěr</b>   | <b>47</b> |
|          | <b>Literatura</b>  | <b>48</b> |

|          |   |           |
|----------|---|-----------|
| <b>A</b> | <b>Obsah přiloženého paměťového média</b>     | <b>51</b> |
| <b>B</b> | <b>Manuál pro spuštění systému</b>            | <b>53</b> |
|          | B.1 Server modul . . . . .                    | 53        |
|          | B.2 Senzor modul . . . . .                    | 54        |
| <b>C</b> | <b>Příklad API příkazů</b>                    | <b>55</b> |
| <b>D</b> | <b>Ukázka webové aplikace</b>                 | <b>58</b> |
|          | D.1 Lokace zařízení v čase . . . . .          | 58        |
|          | D.2 Seznam lokací . . . . .                   | 59        |
|          | D.3 Nastavení systému . . . . .               | 59        |
| <b>E</b> | <b>Lokalizační modul</b>                      | <b>60</b> |
|          | E.1 Foto vytvořeného prototypu . . . . .      | 60        |
|          | E.2 Modely desky tištěného spoje . . . . .    | 62        |
|          | E.3 Tabulka použitých součástek . . . . .     | 64        |
|          | E.4 Schéma obvodu přístupového bodu . . . . . | 66        |

# Kapitola 1

## Úvod

Lokalizace zařízení a předmětů uvnitř budov je již dlouhou dobu v hledáčku výzkumných skupin po celém světě. Geografickou navigaci lidstvo úspěšně využívá již řadu let, pomocí satelitních systémů jako je GPS, GLONASS, nebo připravovaný Galileo. Ve venkovním otevřeném prostředí jsou tyto systémy běžnou součástí našich životů. Bohužel v prostředí interiérů budov selhávají a jejich použitelnost je výrazně omezena. Hledá se proto spolehlivý zástupce využitelný v hustých urbanizovaných oblastech, velkých interiérových komplexech a halách i v menších, například kancelářských prostorách.

Žhavým kandidátem pro tuto pozici se stávají lokální bezdrátové sítě, díky jejich hromadnému nasazování a rozšiřování. Prim v této oblasti hraje IEEE institut, se svými standardy pro sítě jako Bluetooth, ZigBee, nebo UWB. Nicméně do popředí se dostává zejména jejich technologie Wi-Fi. Její masivní využití hlavně v hustě obydlených oblastech a zavádění v interiérech všeho typu z ní dělá slibnou možnost doplnění geografických satelitních systémů tam, kde jejich spolehlivost selhává.

Lokalizace uvnitř budov nachází velice široké uplatnění. Například složky záchranného sboru mohou snáze a rychleji zjistit pozici zásahu. Umožní lepší orientaci na letištích, v nádražních halách, nebo obchodních centrech. Může sloužit pro navigaci robotů ve výrobních prostorech, nebo sledování pacientů v nemocničních zařízeních.

Cíl této práce je ovšem ve skromnějším měřítku. Úkolem je navrhnout systém, který umožní lokalizovat Wi-Fi zařízení v interiérech kancelářských prostor v malé, nebo střední firmě. Hlavními požadavky na systém jsou nízká cena, využití pouze Wi-Fi (bez dalších senzorů, nebo sítí) a pokud možno, tak “pasivní” sledování. Tedy tak, aby na cílovém lokalizovaném zařízení nemusel být instalován a spuštěn žádný software navíc. Posledním požadavkem je potom snadná škálovatelnost, aby při změně prostor, šlo snadno lokalizační síť rozšířit. Má tedy vzniknout systém s možností sledování ve které místnosti, či její části, se nachází nebo nacházelo hledané zařízení.

Práce nejprve uvádí teoretický souhrn dnes využívaných lokalizačních prostředků v terestriálních sítích. V kapitole 2 jsou shrnuty principy lokalizace obecně, následované popisem využívaných technik v sítích Wi-Fi. Dále kapitola tyto techniky rozvádí v části věnované metodám, jež vylepšují lokalizaci pomocí dostupných prostředků z novějších standardů Wi-Fi.

Kapitola 3 potom popisuje návrh systému. Jeho architekturu vycházející ze sensorových modulů, tvořících bezdrátovou lokalizační síť a centrální logickou část, která zodpovídá za zpracování dat, ukládání a jejich vizualizaci, nebo konfiguraci sensorové sítě.

Následující kapitola 4 uvádí realizaci systému. Jsou rozebrány jednotlivé části architektury a to z pohledu hardware i implementace software. Také je zhodnocen systém jako celek a uvažovaný způsob použití, respektive nasazení do provozu.

V poslední části 5 jsou shrnuty testy a jejich výsledky v reálném prostředí menší vývojářské firmy. Součástí je i úvaha nad naměřenými hodnotami a nastíněny některé možnosti vylepšení, nebo směřování projektu.

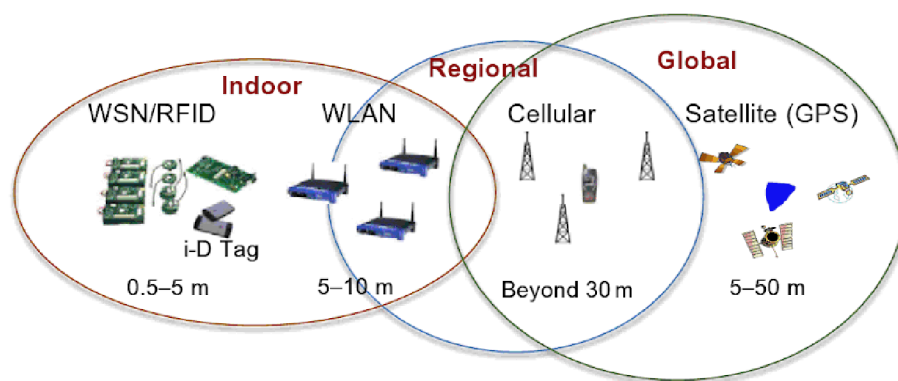


## Kapitola 2

# Teorie lokalizace subjektů

Nedílnou součástí našich životů dneška, je určení polohy. Ať už pro potřeby navigace dle trasy k danému cíli, pro orientaci v prostoru, nebo pro nalezení vzdáleného zařízení. Lokalizace se obecně dělí na dva druhy, a to **vzdálenou lokalizaci**, kdy centrální jednotka určuje polohu hledaného subjektu (vyhledání mobilního telefonu v buňkové síti GSM) a **sebelokalizaci**. V tomto případě sám lokalizovaný objekt určuje svoji polohu (GPS). Lokalizace se provádí v bezdrátových rádiových sítích (v drátových postrádá smyslu) a ty lze dle [4] rozdělit na dvě majoritní skupiny. Podrobnější dělení pak znázorňuje obrázek 2.1.

- **Globální** – spoléhající na satelity obíhající v přesně daných a známých drahách kolem planety. Jsou od počátku vytvářeny právě pro účel lokalizace a fungují na principu sebelokalizace. Zástupci v době psaní tohoto textu jsou především americké GPS, ruský GLONASS, čínský BeiDou a několik dalších. Evropská unie také připravuje svůj systém nazvaný Galileo.
- **Terestriální** – pozemské bezdrátové sítě mají tu nevýhodu, že většinou byly vyvíjeny pro jiný než navigační účel. I když vznikají výjimky, díky již zavedeným technologiím (Wi-Fi, Bluetooth), které jsou velmi rozšířené, vzniká potřeba doplnit funkcionalitu těchto sítí právě o lokalizační možnosti.



Obrázek 2.1: Rozdělení lokalizačních systémů. Převzato ze [4]

Podle zadání má náš vyvíjený systém pracovat v bezdrátové síti Wi-Fi, což je obchodní označení standardů rodiny *IEEE 802.11*, které spravuje společnost *Wi-Fi Alliance* [18]. V kontextu Wi-Fi se o lokalizaci a navigaci hovoří už řadu let. Funkcionalita samotné

Wi-Fi prozatím není lokalizaci příliš nakloněna a ve standardech se s ní nepočítalo. Až nově vyvíjený standard *IEEE 802.11az* má přinést přesnější určování pozice i úspornější provoz pro podporu senzorových lokalizačních sítí. Nicméně dokončení se očekává až za několik let, jak uvádí [15]. Pro naše potřeby tedy musí vystačit využívané techniky lokalizace uvedené v dalších částech práce.

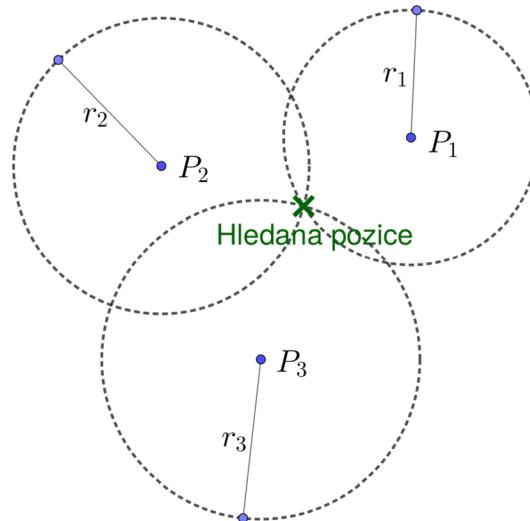
## 2.1 Obecné principy lokalizace

Pro lokalizaci jednoho subjektu v lokální oblasti lze využít praktiky geometrie, využívané například v geodézii. Nalezení bodu v prostoru vyžaduje několik pevných bodů, od nichž můžeme určit vzdálenost hledaného bodu, nebo relativní úhel (směr) ke hledanému bodu. Lze také průzkumem lokalizační oblasti sestavit množinu možných pozic a z ní poté vybírat nejbližší vhodnou.

Všechny následující postupy jsou ovšem zatíženy chybami. Pokud v převodu z reality do matematického (geometrického) modelu vzniknou nepřesnosti, nebude odpovídat konfigurace pevných bodů a výsledky měření budou nepřesné. Další problém potom vzniká měřícími chybami jednotlivých technik zjišťování vzdálenosti, či úhlu. Vliv tohoto problému lze minimalizovat pomocí vylepšení, z nichž některé jsou uvedeny v další části práce (kap. 2.3). [16] [4]

### 2.1.1 Trilaterace

První z principů k určení pozice využívá zmíněnou vzdálenost od kotev (pevných bodů). Pro přesné určení pozice ve 2D prostoru jsou nutné minimálně tři takové pevné body. Proto název trilaterace. Techniky, jak v bezdrátových sítích získat vzdálenost, jsou uvedeny v sekci 2.2. Principem trilaterace je pomocí vzdálenosti od kotvy určit poloměr kružnice, na které hledané zařízení leží. Pokud takto vzniknou kružnice z každého ze tří pevných bodů, průsečík těchto kružnic je právě hledaná pozice. Znázorněno je to na obrázku 2.2, kde  $P_x$  jsou souřadnice pevných bodů a  $r_x$  poloměry kružnic, odpovídající vzdálenosti kotvy a hledaného zařízení.

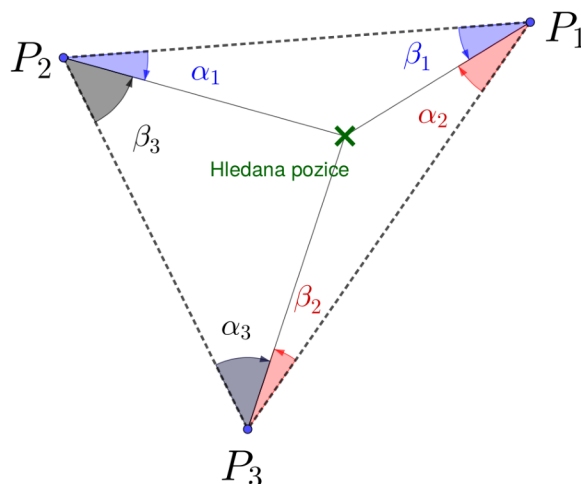


Obrázek 2.2: Princip trilaterace

Existují poměrně jednoduché a rychlé algoritmy, jak spočítat průsečík tří kružnic. Problém ale tkví v přesném určení vzdálenosti a toho, jak bude popsáno v následujících kapitolách, není snadné v bezdrátových sítích založených na Wi-Fi dosáhnout.

### 2.1.2 Triangulace

Jak název napovídá, druhý princip využívá metody měření úhlů. Díky tomu, že známe pozice kotev, můžeme určit směr ke hledané pozici. Mezi každými dvěma pevnými body se vytvoří základna trojúhelníku a od ní se dají určit úhly ke hledanému zařízení. Princip tohoto postupu dobře znázorňuje obrázek 2.3.



Obrázek 2.3: Příklad triangulace.  $P_x$  jsou pevné body,  $\alpha_x$  a  $\beta_x$  potom úhly ke hledané pozici.

### 2.1.3 Fingerprinting

Název tohoto principu by se dal do českého jazyka přeložit jako „snímání a porovnávání otisků“ a podle toho také pracuje. Je založený na vytvoření co nejširší databáze měření, pokud možno v každém bodě zájmového prostředí. A to ještě před samotným provozem lokalizace, jak uvádí [3]. Tato databáze je potom použita pro vyhledávání nejbližšího kandidáta ve vztahu s naměřenými hodnotami. Tak je rozhodnuto o přibližné pozici hledaného zařízení.

Postupy a algoritmy použité při vytváření databáze i vyhodnocování lokace se mění napříč různými řešeními. Nicméně trpí stejnými nedostatky. Zejména jde o bezpodmínečnou nutnost vytvoření databáze, nebo složité (popřípadě žádné) možnosti přizpůsobení se dynamickým změnám v lokalizačních prostorech. Nicméně výsledky mohou být uspokojivé.

## 2.2 Techniky lokalizace využívané v bezdrátových sítích

Pro dosažení určení pozice podle výše zmíněných geometrických postupů a praktik, je potřeba určit vzdálenosti pevných bodů, resp. úhly. Následující techniky jsou často využívány nejen v sítích Wi-Fi, ale i v řadě dalších systémů, včetně satelitních. [16]

### 2.2.1 RSS – síla signálu

RSS, někdy také RSSI (Received Signal Strength Indication) je jednou ze základních praktik měření vzdálenosti využívané v bezdrátových sítích, nejen ve Wi-Fi. Princip využívá faktu, že vysílaný signál logaritmicky slábne s vysílací vzdáleností. Lze tedy poměrně snadno vypočítat vzdálenost, neboť vysílací výkon je znám. Výhodou je, že měření i vyhodnocení může provádět pouze přístupový bod. Zařízení, jehož vzdálenost určujeme nepotřebuje nic, než být v dosahu pokrytí sítě. Na druhou stranu je tato praktika poměrně silně ovlivňována negativními vlivy, jako jsou překážky, nebo vícecestné šíření rádiového signálu [11]. V praxi je potom tato technika využívána s různými vylepšeními. Některé z nich jsou uvedeny dále v sekci 2.3.

#### Určení vzdálenosti pomocí RSSI

Protože je RSSI naměřená hodnota v logaritmických jednotkách, je nutné ji přepočítat pro získání vzdálenosti. K tomu lze použít ztrátový model signálu, převzatý z [6] a [23]. Jde o standardní, poměrně populární a také velice jednoduchý model. Lze jej vyjádřit vztahem 2.1.

$$\overline{PL}(dB) = \overline{PL}(d_0) - 10 * \eta * \log_{10} \left( \frac{d_i}{d_0} \right) \quad (2.1)$$

kde mají jednotlivé symboly následující význam:

- $\overline{PL}(dB)$  – je naměřená hodnota síly signálu uvedená v logaritmických jednotkách (dBm)
- $\overline{PL}(d_0)$  – uvádí referenční sílu signálu v pro referenční vzdálenost  $d_0$ . Tato hodnota může být naměřena při provozu, nebo stanovena předem.
- $d_i$  – je hledanou vzdáleností dvou bodů
- $\eta$  – představuje faktor útlumu signálu, jehož hodnota se mění v závislosti na složitosti prostředí měření. Nabývá hodnot od 2 (nejméně rušné, exteriér) po 6 (velice rušné, členitý interiér s překážkami)

Jak je patrné, uvedený model je skutečně jednoduchý. Tomu ovšem odpovídá také nevalná přesnost výsledků při jeho použití. Proto se v literatuře objevují různé varianty jeho vylepšení. V drtivé většině jde ale o doplnění konstant, které dodávají více znalostí o prostředí, jež ale ne vždy jsou známé.

#### Příklady RSSI systémů

Pro doplnění je dále uvedeno několik příkladů navržených řešení, využívajících právě RSSI techniku.

Jeden z raných<sup>1</sup> systémů využívající Wi-Fi a metodu fingerprintingu se nazývá **RA-DAR** [1]. Databáze pozic je nejdříve naplněna off-line daty, ke kterým jsou vypočteny odchylky a mediány síly signálu ke každé z pevných přijímajících stanic, těm data zasílá lokalizovaný klient broadcastovými UDP pakety. Poté pomocí půdorysu budovy, v níž je lokalizace prováděna, nastaví systém potřebné parametry pro co nejpřesnější ztrátový model

---

<sup>1</sup>Zveřejněn v roce 2000

(počet stěn mezi zařízeními, jejich materiál apod.). Různými přístupy také řeší statické cíle a pohybuující se uživatele. Přesnost nebyla nijak přesvědčivá (cirka 2.5 m), ale i tak mohl systém splňovat potřeby některých uživatelů.

Další z uvedených příkladů systémů využívajících RSSI techniky je **HORUS** [22]. Oproti RADARu jde o novější přístup k problematice lokalizace, nicméně pracuje na podobném principu. Stejně jako předchozí uvedený, i HORUS spoléhá na vytvoření off-line databáze měření, na které pokročilými pravděpodobnostními modely řeší lokalizaci. To a znatelné vylepšení ztrátového modelu energie rádiového signálu vedlo k rapidnímu zlepšení přesnosti až na cirka půl metru. Systém je ale postaven jako „self-localize“, tedy zařízení musí samo ověřovat a dopočítávat svoji pozici ve sledovaném prostoru.

Za zmínku ještě stojí i systém nazvaný Redpin [2]. Jedná se také o jeden z lokalizačních systémů, který využívá RSSI k výpočtům, nicméně tento projekt míří trošku dál. Využívá k lokalizaci nejen Wi-Fi sítě, ale nabízí možnost podpořit data i o informace ze sítí GSM, nebo Bluetooth. Trénovací množinu pak tvoří sami uživatelé pohybuující se ve sledované oblasti s mobilní aplikací. Za jejich přispění může být vytvořena rozsáhlá globální síť pro veřejné (i neveřejné) budovy a napomoci tak k lepší orientaci dalším uživatelům. I k těmto cílům mohou systémy s pomocí RSSI vzhlížet.

### 2.2.2 TOA – čas doručení

Pro určení vzdálenosti dvou zařízení, je možné k výpočtu použít TOA (Time Of Arrival), někdy také nazývaný TOF (Time Of Flight). Což je změřený čas zpoždění od vyslání signálu oproti jeho doručení. K výpočtu vzdálenosti pak stačí dosadit rychlost šíření signálu. Tato technika je ale poměrně obtížná, neboť vyžaduje zcela přesnou synchronizaci jak vysílače, tak přijímače. Což v našem případě pasivního sledování není možné zajistit. Problémem jsou také malé vzdálenosti, protože rádiový signál Wi-Fi (2.4 GHz, 5 GHz) putuje velice rychle. Pro správné změření zpoždění by bylo zapotřebí vybavit jak přístupové body, tak cílová zařízení velice přesnými hodinami s vysokým rozlišením.

Pro eliminaci zmíněných problémů se dá využít některých variant dané techniky. **DTOA** (Differential Time Of Arrival) je velice podobné technice TOA, jen s tím rozdílem, že pro výpočet vzdálenosti se nevyužívá zpoždění signálu, nýbrž rozdíl v čase doručení na různé přístupové body. Tímto způsobem je úspěšně vyřešen problém s nutností synchronizace zařízení s přístupovým bodem, ovšem stále vyžaduje přesnou synchronizaci mezi pevnými stanicemi a hodiny s vysokým rozlišením.

Další variantou potom je **RTT** (Round Trip Time), kdy je měřen časový úsek od vyslání signálu z pevné stanice, až do přijetí odpovědi na něj od lokalizovaného zařízení. Synchronizace mezi stanicemi potom nemusí být zcela přesná, ale měření času na krátkých vzdálenostech jaké jsou v interiéru vyžaduje kvalitní hodiny [11]. Aby tedy byl dodržen požadavek na nízkou cenu systému, není tato metoda příliš využitelná.

Techniky TOA však i přes jeho nedostatky využívá například geografický lokalizační systém GPS [8], kde vzdálenost stanic (zde satelitů) a lokalizovaných zařízení dosahuje desítek tisíc kilometrů.

### 2.2.3 AOA – úhel dopadu signálu

Pro využití principu triangulace, uvedené v sekci 2.1.2, máme k dispozici techniku AOA (Angle Of Arrival). Její princip spočívá v určení rozdílu úhlu dopadu signálu od lokalizovaného zařízení a úhlu referenčního. Z tohoto rozdílu je potom možné dopočítat a určit směr ke hledanému zařízení. Je nutné využít antény citlivé na směr signálu, což vyžaduje insta-

laci speciálního hardware. Další nevýhodou je silná závislost na vzdálenosti. I malá chyba v měření úhlu při větší vzdálenosti hledaného zařízení, může způsobit fatální nepřesnost. I z tohoto důvodu je tato technika velmi citlivá na vícecestné šíření signálu, resp. na určení přímé cesty. Odražený signál totiž zpravidla dopadá pod jiným úhlem, než signál původní [11].

## 2.3 Vylepšené techniky se zaměřením na Wi-Fi síť

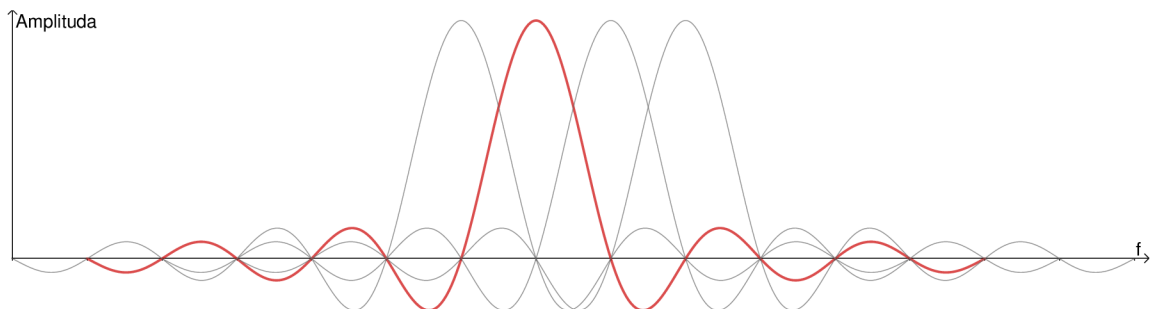
Zmíněné techniky v sekci 2.2 samy o sobě nejsou v interiérové lokalizaci příliš přesné. Jsou citlivé na rušení, dynamický pohyb v prostorech lokalizace, vícecestné šíření rádiového signálu atp. Je tedy přirozené, že pro praktické použití je vhodné využít jejich kombinaci. Pro potřeby našeho projektu navíc nelze využívat speciálního vybavení, jako jsou anténová pole, nebo směrové antény. Zároveň není možné spoléhat se na jakýkoli nástroj na straně hledaného zařízení. Je omezeno pouze na příjem a vysílání Wi-Fi signálu.

V následující části budou představeny vyvinuté nástroje a postupy, které zmíněné problémy částečně odstraňují. Před tím je ale potřeba objasnit některé další pojmy a funkce, které k tomu využívají. Závěr sekce je pak doplněn o zajímavé příklady opírající se právě o doplňující prvky, jako je speciální HW, nebo extra vybavení na lokalizovaném mobilním zařízení.

### 2.3.1 OFDM

První z pojmů je OFDM (Orthogonal Frequency Division Multiplexing), což je modulační schéma využívající rozkladu provozu na několik subnosných frekvenčních kanálů, vysílaných paralelně. Pokud ještě každý takový kanál je zakódovaný bezpečnostním kódem, hovoříme o COFDM (Coded OFDM) [12] [13].

Díky ortogonalitě, mohou být jednotlivé subnosné kanály poměrně blízko u sebe (v kontextu frekvenčního spektra), protože překrytí postranních pásem nevadí a maximum jedné subnosné frekvence je přesně v místě, kde ostatní mají hodnotu rovnu nule. Je tak s výhodou šetřeno přenosové pásmo. Dobře viditelné je to na následujícím obrázku (2.4).



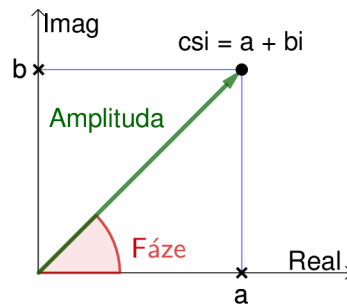
Obrázek 2.4: Rozdělení pásma na subnosné kanály. Funkce subnosné frekvence je  $S = \frac{\sin x}{x}$ . Převzatý obrázek z [13].

Samotná myšlenka rozložení frekvenčního pásma na subnosné kanály už existuje desítky let, a dokonce se využívala v praxi. Často se totiž stává, že je zarušena pouze úzká část spektra, což zde ochromí pouze zlomek provozu. Není tedy znehodnoceno celé pásmo jako u jiných modulací. U Wi-Fi se OFDM poprvé objevuje ve standardu *IEEE 802.11a* (pro Evropu *HiperLAN2*) ve frekvenčním pásmu 5 GHz. Díky rozdělení na více subnosných

kanálů a paralelnímu přenosu se zvýšila i rychlost. Proto se také od standardu *IEEE 802.11g* využívá OFDM modulace i v pásmu 2.4 GHz. K plnému využívání OFDM přispěl také fakt, že modulaci i demodulaci lze poměrně efektivně implementovat algoritmy založenými na rychlé Fourierově transformaci (FFT) [13].

### 2.3.2 CSI

Díky tomu, že dnes takřka všechny Wi-Fi zařízení využívají OFDM modulaci (kap. 2.3.1), dají se i v běžném provozu využít pro přesnější lokalizaci dodatečné informace o chodu vysílače. Jedná se o CSI (Channel State Information), což je informace o každé subnosné frekvenci a jejím provozu. CSI obsahuje pro každou subnosnou frekvenci údaje, vyjádřené jedním komplexním číslem ve tvaru  $csi = a + bi$ . Příklad a využitelné hodnoty jsou uvedeny na obrázku 2.5. [21]



Obrázek 2.5: Znázornění uložené informace CSI. Převzato z [21].

Wi-Fi standard, kterému musí zařízení odpovídat, aby bylo možné přistupovat k CSI je *IEEE 802.11n (a novější)* [16]. Ten totiž přináší jako první technologii označovanou **MIMO** (Multiple Input–Multiple Output), což je využití vícero antén pro paralelní přenos. CSI potom nese informaci pro každou subnosnou, násobeno počtem antén přijímacích i vysílacích. Počet komplexních čísel se dá tedy vyjádřit následujícím vztahem (2.2), kde  $M$  vyjadřuje počet vysílacích antén,  $N$  počet antén přijímacích a  $sub$  počet subnosných frekvencí na kanál.

$$Suma_{CSI} = M * N * sub \quad (2.2)$$

Poměrně běžné jsou dnes přístupové body se třemi anténami, zatímco v běžných zařízeních, které potřebujeme lokalizovat bývá anténa jedna, zřídka kdy dvě. Je to způsobeno převážně snahou ušetřit příkon, neboť každé Wi-Fi rádio navíc je poměrně silnou zátěží pro mobilní zdroje. Co se počtu subnosných frekvencí týká, podle [16] je standardní šířka kanálu 20 MHz a je rozdělen na 56 subnosných. Standard ale umožňuje šířku kanálu zdvojnásobit a tím umožňuje zvýšit tento počet na 114. Výsledkem tedy může být až

$$684 = 3 * 2 * 114 \quad (2.3)$$

komplexních hodnot. Pokud uvažujeme na každou složku čísla např. 10 B (liší se výrobce od výrobce), pak je provoz pouze pro CSI 6 kB na jedno jediné zařízení. Je tedy vhodné zvážit, zda se nevyplatí zpracovávat tyto informace a transformovat je na poziční souřadnice přímo na každém z pevných přístupových bodů. Problém také je, že ne každý výrobce obvodů umožňuje k této informaci přístup. [21] [7]

### 2.3.3 Existující lokalizační CSI systémy a jejich vlastnosti

#### FILA

První z nich je systém, nebo postup nazvaný FILA (Fine-grained Indoor Localization) [19]. Vychází z faktu, že lokalizace ve Wi-Fi sítích pouze pomocí RSSI techniky není dostatečně přesná. Ve vnitřních prostorách je silně zatížena odrazy signálu, které způsobují vícecestné šíření a překážkami, které znemožňují přesný výpočet vzdálenosti ze ztráty síly signálu (kap. 2.2.1). Překážky mohou být i dynamického rázu, jako otevření/zavření dveří, nebo pohybující se lidé. Výzkumná skupina složená ze členů institutu *IEEE* proto hledala náhradu v podobě nějaké spolehlivější informace, než je amplituda signálu, jako je tomu u RSSI. Odpovědí jim byla CSI (kap. 2.3.2). Nese totiž nejen amplitudu signálu, ale navíc i fáze subnosných frekvencí.

Systém FILA potom funguje v několika krocích.

1. Prvním krokem je shromažďování kanálové informace (CSI) ze subnosných frekvencí, které jsou rozděleny do třiceti skupin přes šířku pásma.
2. Další je potom zpracování této informace, které probíhá dvojitým způsobem. Je potřeba zmírnit dopady chyb na určení vzdálenosti, které mohou být zapříčiněny ztrátou signálu (path loss), vícecestným šířením (multipath propagation), nebo ztrátou síly signálu při průchodu překážkami (shadowing).

**Zmírnění vícecestného šíření v časové doméně** využívá faktu, že odražené signály mají jiné zpoždění než signál LOS (Line Of Site). Ten se dá rozeznat tak, že má větší amplitudu než signály odražené. Takže lze použít časové “okno” pro odfiltrování odražených signálů.

**Kompenzace slábnutí signálu ve frekvenční doméně** je další možností, jak zmírnit negativní dopady. Díky vysoké pravděpodobnosti, že slábnoucí signál ovlivní jen některé subnosné, může být provedena kompenzace menšího slábnutí pomocí vážených průměrů a různé subnosné jsou transformovány do jedné, označené jako  $CSI_{eff}$  použité v následujících krocích.

3. Následuje kalibrace podle modelu šíření používaného pro RSSI. Jeho základní verze nebyla pro interiérové lokace dostačující, a tak je variabilní model a jeho parametry nejprve nastaven pomocí trénovacího algoritmu. A to pro každý přístupový bod zvlášť.
4. Pro určení pozice lokalizovaného zařízení využívá systém FILA metody fingerprintingu<sup>2</sup> (porovnávání kalibračních hodnot). Musí tedy nejdříve být vytvořena podrobná mapa, ve které jsou zaneseny CSI údaje. Tyto informace jsou ukládány také po výše zmíněném zpracování, a to pro každý přístupový bod a pozici.
5. Nakonec každý přístupový bod nezávisle pomocí např. Pearsonovi korelace a pravděpodobnosti určí nejbližší možnou pozici z dříve naučené množiny.

Systém byl testován ve třech různých prostředích. V laboratoři, v přednáškové hale a v prostoru chodby. Cílem bylo ukázat přínos oproti metodám, které používají pouze RSSI techniky pro určení pozice (například dříve zmíněný HORUS, popsáný v sekci 2.2.1). FILA založená na CSI informaci se ukázala jako výrazně stabilnější a rychlejší technika, než porovnávané. Medián přesnosti vyšel na 0,65 m. Ve složitějších podmínkách koridoru, byl

<sup>2</sup>Podle [20] používala původní varianta pro určení pozice výpočet vzdálenosti a trilateraci.



naměřen medián přesnosti 1,07 m. V obou uvedených případech jde ale o zhruba 25% zisk, oproti metodě Horus. Na CSI založená metoda FILA dosáhla výrazně lepších výsledků, než metody bez této informace.

## Orion

Jako příklad systému, který vylepšuje AoA techniku, uvedme Orion<sup>3</sup>.

Princip spočívá ve shromažďování CSI jak na straně přístupového bodu, tak na straně hledaného zařízení, a to pouze ze standardních tří antén. Běžně je pro přesné stanovení úhlu signálu potřeba rotační anténa, nebo anténové pole (v interiéru 6–8 antén) [9]. Díky CSI ale není potřeba speciální hardware. Orion systém ze získaných informací o subnosných vytvoří matici pro algoritmus MUSIC [14] a pomocí něj vypočte odhad nejen AoA, ale i AoD (Angle of Departure), tedy úhel odletu. Nakonec pomocí filtrů poměrně přesně stanoví úhly odchozího i příchozího provozu.

Systém jako takový není navržen přímo pro lokalizaci. Jde spíše o ukázkou, jak lze CSI využít i pro AoA techniku v interiérových prostorách, a to poměrně snadno na dnes běžných, relativně levných, tří anténových Wi-Fi kartách.

## Chronos

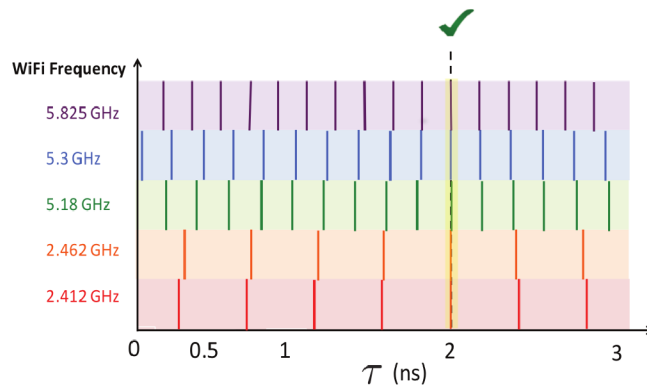
Kde Orion řeší přesnější techniku měření úhlu dopadu bezdrátového signálu, tam systém nazvaný Chronos [17] opět pomocí CSI umožňuje radikálně zpřesnit měření času přenosu signálu. Jak bylo uvedeno v sekci věnující se technice ToA (kapitola 2.2.2), měření času musí být velice přesné, což na běžně dostupném HW nelze zajistit. Systém Chronos tak pro měření využívá fázového posunu (získaného pomocí CSI), který se cyklicky mění s časem „letu“ signálu. Kvůli cyklickému průběhu změny fáze, jsou po matematickém přepočtu získány násobky určitého času. Nicméně který z těchto násobků je ten hledaný, není možné určit.

Zde přichází na řadu hlavní myšlenka algoritmu a to, že se násobek času mění na základě vysílané frekvence. Byl tedy implementován protokol, který řídí vysílač a zajišťuje, že synchronně i na straně přijímače, bude docházet k pravidelnému přepnutí provozní frekvence. A to jak v pásmu 2.4 GHz, tak i v 5 GHz oblastech. Po několika frekvenčních „hopech“ tak vzniká soustava rovnic, kde hledaný měřený časový násobek je správným řešením této soustavy. Bez hlubšího matematického popisu je uveden obrázek 2.6, jako ukázkou tohoto přístupu.

V praktickém světě samozřejmě vznikají nepříjemné efekty, se kterými je potřeba se vypořádat. Jedním z nich je i zpoždění reakce při příjmu paketu. Wi-Fi zařízení zkrátka nějakou dobu trvá, než zareaguje na příchozí paket. Musí načíst a reagovat na prvních několik vzorků, a to se neobejde bez zpoždění. S řešením tohoto problému opět pomáhá CSI, které zaznamenává příjem na každé subnosné frekvenci a pomocí interpolace mezi těmito záznamy je možné zpoždění eliminovat. S vícecestným šířením signálu se systém vypořádává pomocí upravené inverzní neuniformní diskretní Fourierovy transformace a matematicky také řeší drobný fázový posun, způsobený přepínáním frekvence vysílání, nebo vyrovnává malé frekvenční rozdíly mezi subnosnými kanály.

Konečné vzdálenosti a pozice lokalizovaných subjektů řeší Chronos tak, že čas letu podělí rychlostí šíření rádiového signálu ve vzduchu, tedy rychlostí světla. Tím dostane výsledné vzdálenosti zařízení od přístupového bodu. Kotva, která musí být vybavena více anténami

<sup>3</sup>Systém Orion dostupný na adrese: <https://github.com/naoufal51/Orion>.



Obrázek 2.6: Ukázka frekvenčních pásem a řešení času přenosu. Převzato z [17].

má známou pozici, ale také jsou známy vzdálenosti a umístění jednotlivých jejích antén. Vzniká tak další soustava rovnic, v nichž vystupují vzdálenosti od antén a pomocí toho je možné vypočítat pozici. Geometrie poskytuje prostředky pro vyřazení nereálných umístění a optimalizací je určena co nejpřesnější lokace zařízení.

Při experimentech dosahoval Chronos poměrně dobrých výsledků. Naměřený medián chyby byl 65 cm při LOS (Line Of Site) a 98 cm při NLOS. I když se výsledky mohou zdát nepřesvědčivé, tak je potřeba přihlídnout k faktu, že hledání pozice je za pomoci pouze jediného přístupového bodu. Což je také jedna z největších výhod uvedeného systému. Nevýhody ovšem plynou zejména z nutnosti vysílání CSI paketů a také kvůli častému střídání vysílacích frekvencí, což je pro běžný provoz internetu v zařízení v podstatě nepoužitelné. Problém může být také omezení provozu ostatních klientů, připojených k přístupovému bodu. Tedy systém Chronos dobře slouží k lokalizačním úlohám, ale k ničemu jinému. S ohledem na stanovené požadavky je tedy pro nás tento systém opět nevyužitelný.

## SpotFi

Kombinaci výše uvedeného dobře využívá další systém nazvaný SpotFi. Jeho premisou je levné a snadné nasazení systému do provozu, tedy využití existující bezdrátové infrastruktury (je-li to možné) a přitom se přesností vyrovnat systémům využívajícím speciální hardware a nástroje. I zde jsou využity dnes již standardní přístupové body pouze se třemi anténami a možnost zasílat a přijímat CSI obohacené pakety.

Lokalizace pomocí SpotFi je dosaženo ve třech krocích.

1. První na řadě je zjištění AoA a ToF údajů z různých cest šíření signálu. Při tom je využit algoritmus MUSIC [14]. Aby se dosáhlo co nejlepších výsledků, transformuje SpotFi fázové posunutí mezi subnosnými na ToF. Umožní tak rozšířit sensorové pole přijímače z  $N$  antén, na  $n * subcarriers$ . To ve spolupráci s MUSIC umožní dobře a přesně rozlišit jednotlivé cesty signálu.
2. Dalším krokem je pak určit, která z daných cest je přímá (Direct Path). Obecně se dá říct, že DP by měla být ta, která dorazí za nejkratší čas, tedy s nejnižším ToF. Nicméně může nastat případ, že přímá cesta neexistuje. SpotFi se s tímto problémem vypořádává tak, že nasbírá deset po sobě jdoucích paketů v každém přístupovém bodě. Z pozorování totiž vyplývá, že hodnoty (AoA a ToF) přímé cesty se oproti odraženým příliš nemění. Pokud se tedy vynesou hodnoty ze všech cest a paketů do

grafu, vytvoří shluky. Ten ze shluků, který má nejnižší rozptyl a zároveň nejnižší ToF, je označen jako kandidát na DP. Kandidátní cestě se přiřadí váha, která vyjadřuje pravděpodobnost, že tato cesta je hledaná přímá. To, aby se předešlo zásadnímu ovlivnění výsledku v případě, že je cesta zvolena chybně, pokud DP neexistuje.

3. Nakonec zbývá jen zařízení lokalizovat. Až zde začínají jednotlivé přístupové body spolupracovat. Resp. serverová instance zpracovávající údaje z kotev. Protože nelze zcela přesně synchronizovat čas na různých přístupových bodech, jsou k lokalizaci využity pouze údaje AoA a přidá se RSSI. Po složení výsledků ze všech kotev, s přihlédnutím na jejich pravděpodobnosti z minulého bodu, je určena oblast možné pozice. Využitím různých optimalizačních technik je potom tato oblast minimalizována na jednu přesnou pozici.

Co se naměřených výsledků týká, vede si SpotFi skutečně dobře i proti systémům, jež vyžadují extra úpravy a hardware. Medián chyby v prostředí běžné kanceláře byl naměřen 40 cm, což se minimálně vyrovná systémům s rotačními anténami. Jiné systémy takové přesnosti zdaleka nedosahují. Testy proběhly i v prostorách, kde nebyl dostatek přístupových bodů, nebo přímých cest signálu. I v těchto ztížených podmínkách je SpotFi s mediánem chyby (přesnosti) 1,6 m přijatelné řešení lokalizace s ohledem na to, že nijak zásadně nebrání běžnému síťovému provozu.

SpotFi tedy ukazuje, že s možnostmi CSI a vhodnými matematickými postupy, lze dosáhnout se standardním vybavením dneška přesností, kterými se mohly chlubit jen specializované, drahé a nepohodlně nasazované systémy. Z pohledu našeho zadání ale zůstává největší nevýhodou nutnost klienta vysílat CSI pakety, což není standardní provoz. Výrobci dnešních karet neumožňují přístup k CSI běžně a je tedy nutné upravovat firmware, což může vést jednak k negativnímu dopadu na provoz zařízení, ale také to na, zejména mobilních zařízeních, nemusí být vůbec možné. Pro popis zmíněného systému bylo využito zdroje [9].

Podobných postupů a metod, využívajících CSI pro lokalizaci přirozeně vznikla a vzniká celá řada. Vyšší přesnosti ale dosahují zejména díky použití dodatečných senzorů, jako jsou kamery, akcelerometry, ultrazvuk, kompas apod. Vzhledem k uvedeným požadavkům na námi navržený systém tedy není možné takových vylepšení využít. Pro úplnost ale uvádím několik zástupců s hrubým popisem jejich funkčnosti.

### Systémy využívající dodatečné prvky

**Ubicars** je první ze zmíněných systémů využívajících více prvků pro lokalizaci, mimo samotné Wi-Fi [10]. Pracuje na principu radaru se syntetickou aperturou (angl.: Synthetic aperture radar – SAR). Sledované zařízení musí být vybaveno dvěma anténami a na vyzádání jej uživatel otočí kolem vodorovné osy. Ve spolupráci s vysíláním informací o kanálech subnosných frekvencí, je v podstatě nahrazeno kruhové pole antén. Je tak možné získat polohu relativní k přístupovým bodům v okolí. Samozřejmě hraje roli spousta negativních vlivů, jako některé, co již byly zmíněné dříve, ale také například nedokonalé otočení z důvodu lidské chyby. Je tedy potřeba dostatečná porce matematického zpracování dat pro získání pozice.

Pro zpřesnění určené lokace, nebo pro využití takzvaného „geotaggingu“, což je obecný název pro proces přidávání geografických dat k různým objektům, může být na dané lokaci provedeno několik snímků kamerou. Toto může být využito například pro lepší určení pozice

zajímavých objektů v muzeu, oddílu v knihovně, nebo produktu ve skladu. A to včetně svíslé osy, tedy ve 3D prostoru.

S výsledným naměřeným mediánem chyby 39 cm se jistě jedná o zajímavé řešení. Nicméně nevýhody jsou zde také. Zdaleka ne všechna mobilní zařízení jsou vybavena dvěma anténami, spíše naopak. Dále je potřeba dostatečný výkon pro zpracování obrazových dat apod.

Systém nazvaný **Phaser** [5] na druhou stranu spoléhá na speciální, i když dle slov vývojářů levný, hardware. Ten umožňuje propojení více Wi-Fi modulů do většího anténového pole, mezi nimiž je zajištěna synchronizace. Je tak dosaženo kvalitního získávání úhlů dopadu signálu za pomoci měření rozdílů fázových posunů na jednotlivých anténách. Systém se také snaží potlačit vliv vícecestného šíření. Při čerpání informací o fázi se opět spoléhá na CSI a jeho výstupem jsou pouze úhly. Pro stanovení pozice je tedy zapotřebí výpočetní nástavba. I když tedy využívá pouze Wi-Fi, je nutné pro zavedení v praxi použít extra HW, což může být nepraktické.

Podobných systémů přirozeně vzniklo a vzniká velmi mnoho. Málo z nich se však hodí pro širší využití v praxi, a to je způsobeno zejména problémovým čerpáním CSI na dnešních standardních Wi-Fi modulech. Zůstávají tedy spíše výzkumnými nástroji.

## Kapitola 3

# Návrh lokalizačního systému

Následující kapitola se věnuje návrhu lokalizačního systému. V první části je rozbor zadání a specifikace požadavků, následována vlastním návrhem systému, jeho architektury a jednotlivých částí.

### 3.1 Specifikace požadavků

Zadání pro reálný projekt by mohlo vypadat následovně: „Firma s nevelkými kancelářskými prostorami požaduje vytvoření systému, který by jí umožnil sledovat pozice mobilních zařízení, popřípadě pracovníků jež tato zařízení obsluhují.“ V několika následujících bodech jsou shrnuty hlavní požadavky na vyvíjený systém vyplývající z diskuse se zadavatelskou firmou, tedy potencionálním zákazníkem a uživatelem navrhovaného systému.

- Systém pracuje pouze na bázi Wi-Fi. Žádné doplňující sítě ani senzory nejsou přípustné. Systém má za úkol sledovat různorodá zařízení, která jsou s jistotou vybavena jen Wi-Fi. (Např. notebooky nebývají vybaveny gyroskopy, nebo akcelerometry.)
- Co nejnižší cena systému. Poměrně standardní požadavek, který dovoluje využití levných a dostupných nástrojů a hardware. Instalace nákladných specializovaných prvků pro lokalizaci by tedy byla proti tomuto požadavku.
- Snadná škálovatelnost systému. Pokud se změni prostory kanceláří (přesune se nábytek, nebo stěna), měl by systém na tuto změnu reagovat bez nutných větších zásahů. Zároveň pokud dojde k radikálnější změně v půdorysu (rozšíření prostor), mělo by jít snadno upravit i systém, například přidáním kotev. Tento požadavek znemožňuje využití metod fingerprintingu, popsaném v sekci 2.3.3. Obnovovat naučenou databázi pozic v provozu není žádoucí.
- Rozlišení pozic na úrovni místností. Přesnost dosažitelná pouze pomocí Wi-Fi by tomuto požadavku měla vyhovět. Zároveň není nijak fatální, když nebude pozice určena správně. Je tedy nutné stanovit správný kompromis mezi výhodami a nevýhodami použitelných postupů.
- Pasivní lokalizace. Sledované zařízení by nemělo být jakkoli upravováno a přizpůsobováno pro potřeby lokalizace. Pozice zařízení by měla být zaznamenávána pouze po připojení do sítě, bez dalších nutných akcí.

## 3.2 Návrh architektury systému

Mnou navržený systém tedy musí splňovat uvedené požadavky z minulé sekce. Protože není přípustné využít speciální vybavení v podobě anténových polí, nebo antén citlivých na směr, nelze využít techniku AOA popsanou v sekci 2.2.3. Také drahé a přesné hodiny, které by umožnily použít metody TOA (TOF) ze sekce 2.2.2 nejsou použitelné, neboť by došlo k razantnímu zvýšení ceny vysílače/přijímače signálu. Z nabízených technik v kapitole 2.2 tedy zůstává jediná využitelná, a to měření RSSI (kap. 2.2.1).

Při využití této techniky se přímo nabízí zapojit do měření také CSI zmiňované v sekci 2.3.2. Bohužel však z výzkumu vyplynulo, že ani tato cesta nepovede k cíli. Totiž ne každý výrobce Wi-Fi obvodů umožňuje přístup k CSI a jen některé čipy a síťové karty (NIC) jsou (resp. byly) schopny tuto informaci dodat. Nové výrobní řady již toto neumožňují vůbec.

Příklady systémů, jež CSI využívají staví na teorii, že je možné využít nástroj **Atheros CSI Tool**<sup>1</sup>. Jak se ale ukázalo, firma Atheros spojila v roce 2011<sup>2</sup> síly se společností Qualcomm a od té doby není možné čerpat CSI informace z jimi vyráběných NIC čipů. Nástroj již tedy není s novými kartami kompatibilní a starší modely již nejsou dostupné na trhu. Vytvořený systém se tak musí bez tohoto cenného doplňku, jakým CSI bez pochyby je, bohužel obejít.

Určování pozice podle samotného měření síly signálu je značně nepřesné. Existující systémy v praxi (sekce 2.2.1) vylepšují výsledky pomocí předem naměřené databáze hodnot a jejich následnému porovnávání. Z požadavků ovšem vyplývá, že tato metoda pro naše řešení není vhodná, neboť systém má být schopen přizpůsobit se i podstatnějším změnám půdorysu pokrytého prostoru a „dotváření“ databáze měření není žádoucí.

Pro splnění požadavku na přesnost, tedy rozlišení alespoň na úrovni místností by však mělo postačovat i samotné měření síly signálu. A to za předpokladu, že by byla každá oddělená lokace vybavena měřicím senzorem. Je-li tedy k dispozici dostatek měřicích jednotek, které jsou jednoduché na zapojení a zprovoznění, jejich počet se může snadno měnit bez dalších větších zásahů a jsou-li dostatečně levné, měl by systém z nich vytvořený vyhovovat předloženým požadavkům. S ohledem na výše uvedené, jsem navrhl architekturu, skládající se z několika částí. Náznový obrázek 3.1 naznačuje jednotlivé vrstvy, které jsou dále v textu samostatně popsány.

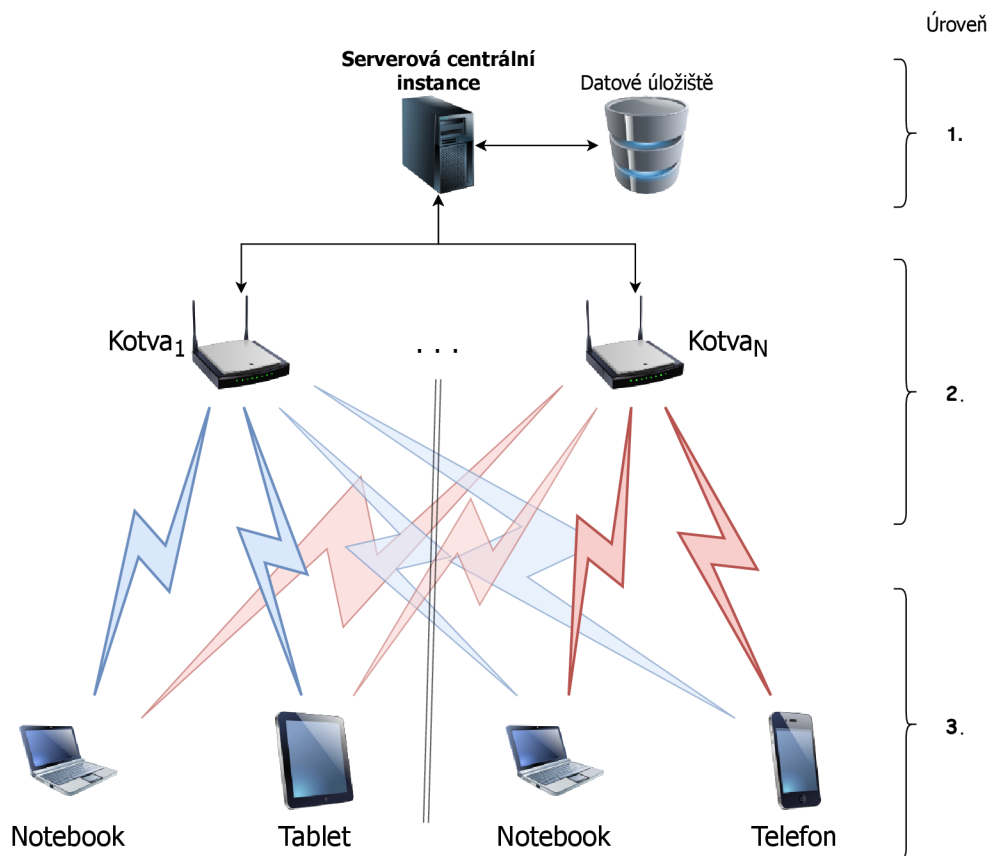
- **Úroveň 1:** Návrh vychází z předpokladu, že většina kancelářských interiérů je již vybavena rozvodem internetové sítě. Proto systém tvoří několik měřicích jednotek a centrální stanice propojeny standardní ethernetovou sítí. I v případě že síťová infrastruktura není v potencionálních cílových prostorách vytvořena, její zavedení spolu s nasazením systému je rozhodně výhodou.

Nejvyšší úroveň architektury tedy obsazuje právě serverová instance, která zodpovídá za sběr, ukládání a prezentaci dat. Až na této úrovni dochází ke spojení dat z přístupových bodů, aby byla dodržena škálovatelnost. Instance pro snadnější nasazení může mít podobu aplikace, kterou lze nahrát v podstatě na jakýkoli počítač v síti. Zde také dochází k potřebným výpočtům pozic a umožněna musí být i konfigurace jednotlivých měřicích senzorů.

Výpočetní aplikace musí být schopna zpracovávat větší množství dat. Ze zadání a požadavků ovšem vyplývá, že není nutné provádět mnoho složitých výpočtů a ani

<sup>1</sup>CSI Tool: <https://wands.sg/research/wifi/AtherosCSI/>

<sup>2</sup><https://archive.is/20120907021250/http://www.qca.qualcomm.com/corporate/content.php?nav1=119&news=321>



Obrázek 3.1: Tří úrovněová architektura navrženého systému. (1. centrální aplikace, 2. přístupové body v roli senzorů, 3. lokalizovaná zařízení)

není kritická rychlost jejich řešení. Je tedy možné zvolit programovací jazyk, který není určený přímo pro matematické úlohy. Nutné je ovšem zajistit snadnou práci s velkým množstvím dat, a tak je na místě sáhnout po některém z databázových systémů.

Prezentaci dat je vhodné řešit ve dvou verzích. Jednak je potřeba navrhnout robustní rozhraní, které umožní napojení na různorodé další (například docházkové) systémy firmy, nebo jiné uživatelské aplikace. Pro snadné použití je ale také možné využít přímou grafickou prezentaci dat pro uživatele jakožto člověka. Rozhraní by mělo být multiplatformní a co nejvíce dostupné. Proto se jako ideální kandidát jeví implementace pomocí web serveru.

- **Úroveň 2:** Další úrovní je potom sada na míru vytvořených senzorů, v podobě Wi-Fi přístupových bodů. Tyto senzory jsou navrženy tak, aby byly velmi jednoduché a tedy i levné. Jejich použití je také velmi snadné. V podstatě je stačí připojit do sítě, zapnout napájení a už by měly samy začít pracovat. Žádné zdlouhavé, nebo složité nastavování není žádoucí a ani potřeba.

V provozu musí zajišťovat komunikaci s centrální stanicí, a to bez rušení lokalizační bezdrátové sítě, tedy nejlépe drátovým spojením. Přístupové body musí být schopny shromažďovat data o zařízeních ve vysílané bezdrátové síti Wi-Fi a tato data dopravovat na centrálu. Výpočty spojené s lokalizací jsou prováděny až na serverové instanci,

což umožní vypustit kooperaci mezi senzory a dovolí snadnou škálovatelnost a další úpravy v topologii zapojení systému.

- **Úroveň 3:** Poslední a nejnižší úroveň v navrženém systému zastávají samotná lokalizovaná zařízení. Obecně není potřeba žádných speciálních modifikací. V zásadě tedy pro zahájení lokalizace není potřeba se zařízením manipulovat jinak, než pouze aktivovat Wi-Fi a připojit jej k lokalizační síti. Systém ovšem umožňuje zařízením zasílat i vlastní měření rozšířené o dodatečné informace, které mohou napomoci k přesnější lokalizaci.

Navíc díky tomu že je použito standardní Wi-Fi a propojení přístupových bodů zajišťuje ethernetová síť, není problém při lokalizaci přenášet také internetový provoz, což je jistě výhodou při použití systému.

Je tedy nutné navrhnout a vyrobit moduly umožňující plnit roli Wi-Fi přístupových bodů. Na jejich operačním systému pak sestavit firmware, který bude zajišťovat sběr dat, nutných pro lokalizaci připojených zařízení, zároveň ale bude schopen přenášet i běžný síťový provoz. Další částí pak bude implementace síťové aplikace, která umožní vhodným rozhraním přeposílat jednak data na centrální stanici, ale také zpětně nastavovat parametry senzorů. Uložená data bude potřeba zpracovat v další části implementované aplikace a konečně, umožnit jejich prezentaci. Posledním doplňujícím dílkem pak bude prozkoumání možností tvorby nástroje, který by zapojil do lokalizace i samotná zařízení.

Testování výsledného systému bude provedeno v interiéru kanceláří zadavatele. Protože systém nemá za úkol přesně lokalizovat cíl v prostoru, ale „pouze“ určit místnost výskytu, výsledkem může být procentuální vyjádření úspěšných pokusů. Také bude vhodné sledovat vliv dynamické změny prostředí, jako je pohyb osob, nebo otevírání/zavírání dveří a jak moc negativní dopad budou na výsledky tyto změny mít. Protože lokalizační síť využívá bezdrátové médium, měla by být zhodnocena změna úspěšnosti, při testech v klidném prostředí a za plného provozu s vysokým počtem zařízení, tedy při vyšší míře rušení.



## Kapitola 4

# Realizace systému

Kapitola uvádí detaily tvorby systému. V úvodní části popisuje výtvar jako celek a zamýšlený způsob použití v praxi. V následujícím textu je pak rozebrán systém po jednotlivých úrovních uvedených v sekci 3.2 a to jak z pohledu hardwaru, tak i implementace softwarové části.

### 4.1 Aspekty nasazení a použití lokalizačního systému

Navržený a vytvořený systém skládající se z několika Wi-Fi přístupových bodů a centrální instance, slouží k pokrytí interiéru malé, či střední firmy bezdrátovým internetovým signálem. Navíc ale oproti existujícím řešením poskytuje možnost lokalizovat do této sítě připojená zařízení, a to bez nutnosti jakýchkoli úprav lokalizovaných subjektů. Systém je cenově dostupný, škálovatelný a snadno zprovoznitelný v již zavedené infrastruktuře. Přibližně takto by se dal shrnout popis vytvářeného produktu.

Pokud se klient, resp. firma rozhodne nasadit tento systém ve svých prostorách, je potřeba učinit několik nezbytných kroků k úspěšnému zprovoznění.

1. **Plán pokrytí sledovaných prostor.** Kolik bude potřeba přístupových bodů pro pokrytí požadavků lze určit následujícím způsobem.

Každá místnost (popřípadě lokace, jež se skládá z více menších místností) by měla být nejlépe vybavena vlastním senzorem. Pokud je místnost větší než 20 m<sup>2</sup>, je lepší rozmístit kotev více. Lokace, které mají být rozlišovány, se nesmějí překrývat a zamýšlený prostor pokrytý jedním senzorem musí být spojitý. Přičemž pojem „místnost“ neoznačuje nutně prostor ohraničený zdmi. V jedné větší místnosti může být více nezávislých lokalizačních (přístupových) bodů.

Jinými slovy, každou část interiéru, kterou je žádoucí z lokalizačního hlediska odlišit, je nutné vybavit zvlášť senzorem. Problém může nastat při snaze rozdělit prostor na velmi malé jednotlivé části. Například menší než 2 m<sup>2</sup>. V tak hustém provozu by mohlo dojít k přílišnému zarušení bezdrátového média.

2. **Příprava síťové infrastruktury.** Prototypová verze senzoru se pro přenos naměřených dat na centrální uzel spoléhá výhradně na ethernetové kabelové spojení. Je tedy potřeba počítat s rozvodem internetového spojení do všech lokalizovaných prostor. Nastavení DHCP, FireWall pravidel a dalších síťových parametrů musí být zajištěno na rozvodných uzlech. Sensory samotné neumožňují tyto vlastnosti ovlivňovat.

3. **Instalace centrálního uzlu.** Ke zprovoznění serveru systému je zapotřebí pouze stroj připojený v síti, který je, pokud možno stále spuštěn, a jemu dostupná databázová instance. Stroj může být firemním serverem, jedním z používaných počítačů k pracovní činnosti, nebo extra vyhrazený pro účely lokalizace. Aplikace není náročná na výpočetní zdroje, jediným požadavkem je tak dostatečně prostorné úložiště pro databázi.
4. **Rozestavení senzorů.** Přístupové body v lokalizačních prostorech je vhodné umisťovat s ohledem na obecné doporučení pro dobré pokrytí Wi-Fi sítí. Upřednostněny jsou tedy volnější prostranství a správná orientace antén vzhledem k požadovanému pokrytí.
5. **Postupné spuštění systému.** Pokud je zprovozněn centrální uzel, je možné provést potřebná nastavení, jako je název bezdrátové sítě, nebo použitá frekvence (kanál). Poté lze postupně zprovožňovat jednotlivé kotvy. Každá se po uvedení do provozu nejprve ohlásí serveru, který ji uloží a umožní její identifikaci. Postupné uvádění do provozu pomáhá s identifikací jednotlivých senzorů, které se ohlašují pouze svojí MAC adresou.
6. **Používání, ovládání a rozšiřování.** Pokud jsou již senzory v provozu, probíhá odesílání, ukládání a zpracovávání dat, je možné čerpat a zobrazovat lokalizační informace. Dostupné jsou v databázi, kde MAC adresou identifikovaná zařízení, jsou přiřazena k lokaci jejich výskytu každých několik sekund (implicitně 5). Tato data je umožněno získávat přes dvě rozhraní. Čistě programové API, anebo graficky pomocí web serveru, který je součástí centrální stanice.

Pokud je potřeba lokalizační síť rozšířit, stačí jen připojit do sítě další senzor. Ten se opět po spuštění ohlásí na centrální instanci a je možné jej identifikovat, nebo provést nastavení. Analogicky je tomu i s odebráním nepotřebných senzorů.

V první verzi systému je zavedena i možnost nasazení speciální aplikace na samotná lokalizovaná zařízení. To může zajistit lepší výsledky měření (aktivní lokalizace). Každé sledované zařízení ale může být jiné, proto je vývoj takové zpřesňující aplikace ponechán na uživateli a implementovaný skript slouží pouze jako příklad. Navrženo je tedy pouze komunikační rozhraní pro zasílání dat aktivní lokalizace na server (kap. 4.2.3).

Uvedený postup není nikterak složitý, ale některé body mohou být problematické. Zejména nutnost kabelového rozvodu sítě. Více detailů a nastínění možných řešení obsahuje další část textu, jež se věnuje detailnímu popisu funkcionality jednotlivých úrovní systému.

## 4.2 Detaily součástí systému

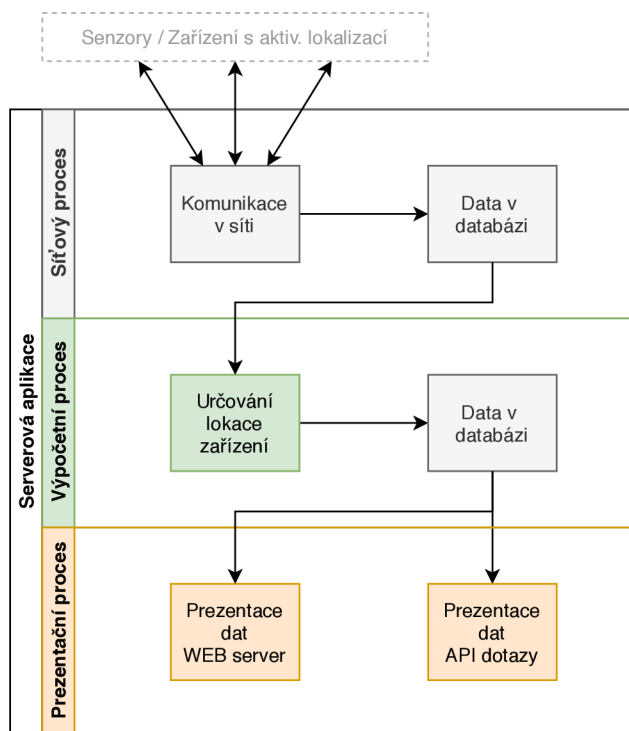
V následujících odstavcích jsou podrobněji popsány jednotlivé úrovně systému, podle rozdělení ze sekce 3.2. Jednotlivé dílčí části jsou rozebrány z pohledu hardware i software vybavy. Zároveň jsou odhaleny největší problémy a pokud je to možné i navržena jejich řešení.

### 4.2.1 Centrální serverová instance

Hardwarové nároky pro instalaci serverové strany systému nejsou nijak vysoké. Je sice možné očekávat zpracovávání velkého množství dat, nicméně jednotlivé výpočty nejsou složité. Důležité je skutečně pouze poskytnout dostatečné úložiště pro databázová data. Databáze ale může být umístěna kdekoli v síťovém dosahu. Nemusí být na stejném stroji jako centrální serverová instance, nebo dokonce může být dostupná i přes internet. Musí být ovšem dodrženo, že centrální uzel (program) jako takový je připojen v lokální síti spolu se senzory. Samotné moduly odesílají broadcast požadavek pro zapsání se na server, ten jim tedy musí být schopen naslouchat a odpovědět.

Jako programovací jazyk pro centrální uzel byl zvolen **python**<sup>1</sup>. To jednak z důvodu jeho poměrně dobře a snadno řešitelné multiplatformnosti, ale i z mého osobního zájmu si tento skriptovací jazyk vyzkoušet. Díky použití pythonu je tedy instalace velice jednoduchá a to v podstatě na kterémkoli operačním systému, jež tento jazyk podporuje. Vyvíjen a testován byl na GNU/Linuxové distribuci Ubuntu s Debian jádrem. Nicméně program nevyužívá žádných speciálních systémových nástrojů, nebo nastavení. Je tedy s minimálním úsilím možné přizpůsobit jej a zprovoznit na jiných operačních systémech.

Serverová aplikace jako celek je složena ze tří spolupracujících částí. Každá z nich je implementována a spuštěna jako paralelní podproces. Je tak zajištěna síťová komunikace se senzory pro získávání dat, z nich jsou dále vypočítávány a určovány přibližné lokace zařízení a nakonec jsou výsledky prezentovány uživateli. Podprocesy běží v režimu „daemon“, čímž je mimo jiné i zajištěno, že po ukončení nadřazeného procesu budou také ukončeny. V následujících řádcích bude objasněno chování těchto jednotlivých částí, přičemž znázornění jejich spolupráce je uvedeno na obrázku 4.1.



Obrázek 4.1: Znázornění spolupracujících podprocesů spouštěných centrální aplikací.

<sup>1</sup><https://www.python.org/>

Všechny části programu a skripty využívají vytvořený modul `config_local`. Ten slouží zejména pro načítání a ukládání nastavení, ale poskytuje také možnost záznamu do logovacího souboru. Parametry systému jsou pak uloženy v JSON<sup>2</sup> formátu a pro serverovou instanci se skládají zejména z přístupových údajů k databázi, používaných čísel portů, nebo cesty k souboru pro ukládání výstupu (log).

## Síťový vysílač a přijímač

Po spuštění serverové instance je jako první aktivován síťový modul. Ten startuje vlákna zajišťující chod NTP (Network Time Protocol)<sup>3</sup> serveru, který na vyžádání předkládá systémový čas stroje, na němž instance běží. Jde o způsob, jak synchronizovat senzory na stejný čas. I v případě že by nebyl zcela přesně stanovený, důležité je že všechny senzory se jím řídí a měřicí cykly jsou na nich spouštěny ve stejném okamžiku. Zdrojové kódy NTP serveru byly z velké většiny převzaty<sup>4</sup>.

Modul se následně připojuje k MySQL/MariaDB databázi a poté spouští paralelně běžící procesy. Za využití python modulu `multiprocessing` je nejdříve vytvořen a spuštěn podproces `broadcast_catch`. Jak název napovídá, jde o proces, jež naslouchá na jedinečném portu (číslo podle nastavení) a přijímá UDP broadcastové datagramy. Pokud zachytí takový provoz, ověří jeho formátování a pokud je správné, jde o hlásící se senzor v síti, nebo připojené zařízení využívající aktivní lokalizaci. Senzor musí nahlásit jedinečnou MAC adresu, podle níž bude dále identifikován a také jemu přidělenou IP(v4) adresu. IP adresy v lokální síti musí nutně přiřazovat DHCP server nakonfigurovaný a jedoucí na některém ze síťových prvků. Samotné senzory totiž nemohou plnit tuto roli, a tak půjde s největší pravděpodobností o směrovač, který v kancelářích zodpovídá za rozvod sítě. A to až dříve existující, nebo dodaný při nasazování našeho systému.

MAC a IP adresami ohlášený přístupový bod je uložen do DB (databáze) a pomocí IP adresy s ním lze komunikovat přímo. Pokud je provoz vyvolán zařízením pokoušejícím se o aktivní lokalizaci, záznam ukládán není a jde v podstatě jen o zjištění dostupnosti serveru a jeho IP adresy.

Další spuštěný a paralelně běžící podproces má název `listening`. Jde o hlavní proces pro sbírání odeslaných dat z jednotlivých senzorů. Na dalším jedinečném portu (`serverPort`) naslouchá a očekává příjem dat. Pracuje neblokujícím způsobem a za pomoci python modulu `select` vytváří fronty, které slouží jak pro ukládání přijatých zpráv, tak pro shromažďování a odesílání odpovědí. Všechny zprávy přijímané i vysílané serverovou instancí jsou ve formátu JSON.

Průběh komunikace se senzory v závislosti na čase je znázorněn na diagramu 4.2. Tučnými šipkami jsou znázorněny broadcast hlášení při inicializaci senzoru v síti. Jakmile je senzor zařazen do systému, začíná s měřením a odesíláním lokalizačních dat. Pro senzor číslo 2 je potom uveden případ, kdy dojde k přerušování přenosu.

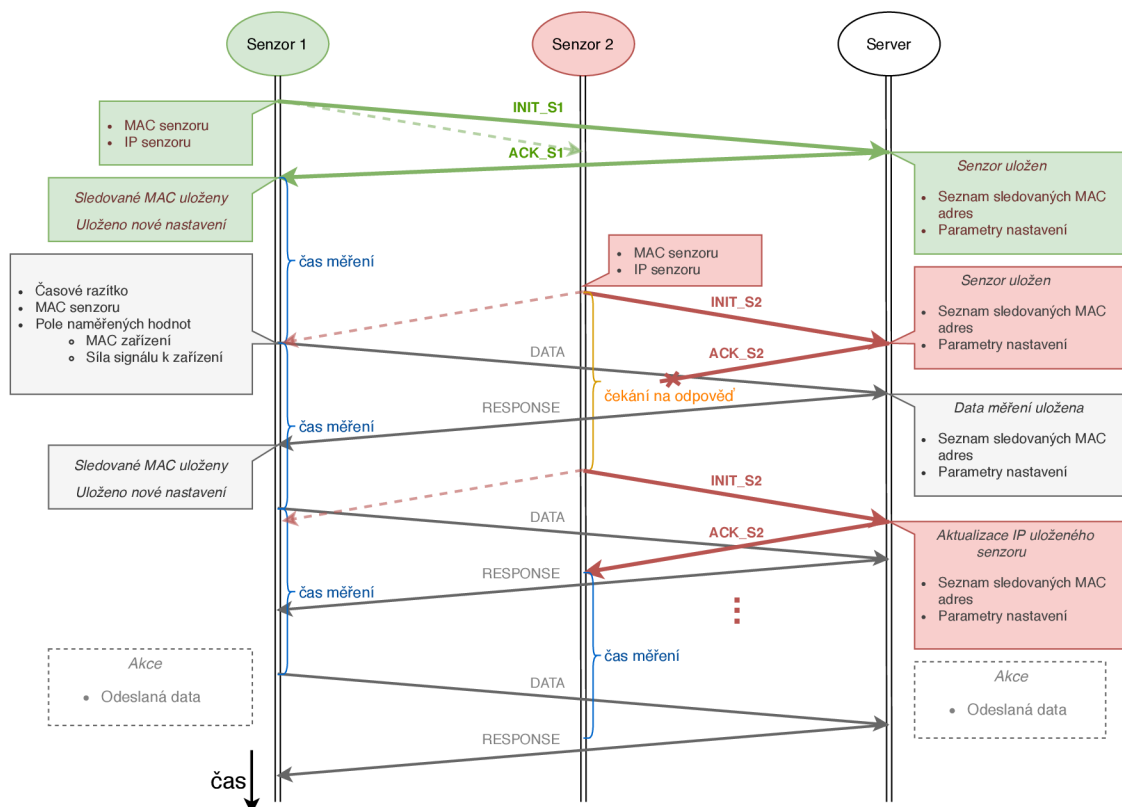
Komunikace se zařízeními provádějící aktivní lokalizaci je velmi podobná. Broadcast požadavek zjistí dostupnost a IP adresu serveru a pak už stačí pouze zasílat naměřená data. Více detailů o rozdílech pasivní a aktivní lokalizace je uvedeno v následujícím textu.

---

<sup>2</sup>JSON: <https://www.json.org/>

<sup>3</sup><http://www.ntp.org/>

<sup>4</sup>z <https://github.com/limifly/ntpserver/blob/master/ntpserver.py>



Obrázek 4.2: Diagram komunikace senzorů a serverové instance v čase. Tučné čáry označují broadcast požadavek/odpověď. Obdélníky obsahují kurzívou uvedenou akci, která je provedena s přijatými daty a poté bodový seznam dat odesílaných jako součást odpovědi.

## Určení správné lokace

Jakmile je funkční správné ukládání dat, přichází na řadu další podproces, který zajistí výpočet a určení pozice z nich. Nese název `count_positions` a opět musí pracovat dvojím způsobem.

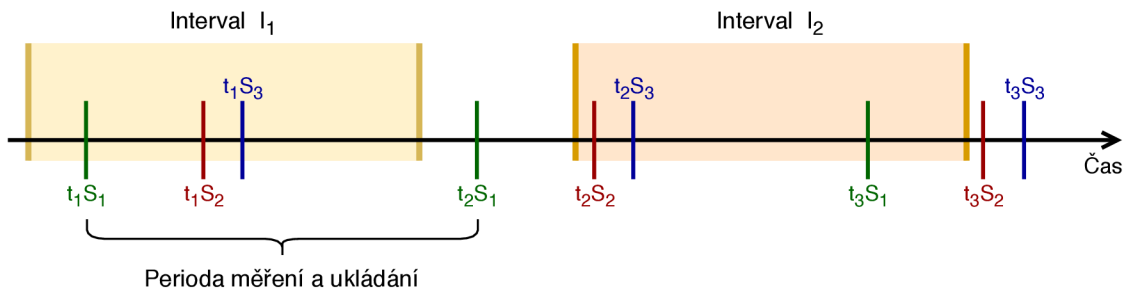
Systém vyhodnocuje lokaci v časových okamžicích daných nastaveným cyklem měření. Nicméně načítání dat z databáze je nutné provádět po časových intervalech. I když se měřicí cyklus na senzorech spouští ve stejný okamžik, pro každý z nich může monitorování provozu a zaslání dat trvat různě dlouhou dobu, tedy data jimi poskytnutá mohou mít odlišná časová razítka. Využil jsem ale faktu, že tento cyklus je na všech senzorech nastaven totožný. Je tedy možné z DB čerpat data ve stejně dlouhém intervalu jako je dlouhý cyklus a tím zajistit, že pro každý načtený interval získáme data ode všech pracujících senzorů.

Programová smyčka, jež zajišťuje načítání a vyhodnocování dat postupuje od těch nejnovějších. Zároveň nejsou načítány údaje, které už jednou byly použity. Proto není možné vzít jednoduše poslední údaj a od něj určit načítaný interval. Algoritmus výběru dat pracuje v následujících zjednodušeně popsaných krocích.

1. V kroku jedna je získáno časové razítka posledního pasivního záznamu dat.

2. V následujícím kroku je pak nalezen první zatím nepoužitý záznam, který je ale starší než razítko z kroku předešlého plus perioda cyklu ukládání dat od senzoru. Tím je získán horní časový okraj hledaného intervalu.
3. V posledním kroku už stačí jen stanovit dolní okraj, což je horní, mínus opět perioda jednoho celého měřicího cyklu. Následně jsou načtena všechna nepoužitá data v tomto časovém rozmezí.

Protože data poskytnutá senzory jsou uložena v různých časech, může dojít k efektu, kdy načítaný interval nepokrývá záznamy ze stejného cyklu měření všech senzorů. Na obrázku 4.3 je tento efekt uveden jako „Interval  $I_2$ “. Vzniknuvší nepřesnost však není až tak zásadní problém, protože jakmile se pozice lokalizovaného zařízení ustálí, budou data z dalších cyklů totožná. Hlavním úkolem systému je lokalizovat statické pozice zařízení a proto toto zakolísání přesnosti není nijak fatální.



Obrázek 4.3: Ukázka časového intervalu při načítání dat z DB. Ideální stav je použití intervalu  $I_1$ , ale ani posunutí jako při  $I_2$  není závažným problémem.

Po načtení dat přichází na řadu jejich vyhodnocení. **Pasivní** data obsahují pouze sílu signálu, naměřenu ve vztahu k lokalizovaným zařízením. Rozhodnutí, která lokace je správná je tak učiněno pouze na základě toho, který senzor zjistil největší sílu signálu k danému cíli. Z toho je pak možné určit a do databáze uložit, že v daném čase (nebo zde v časovém intervalu) bylo zařízení umístěno v té lokaci, ke které je senzor přiřazen. Je zřejmě cítit, že metoda to není nikterak přesná a je silně zatížena chybami. Může ji ovlivnit nejen nepřesné měření, ale také nevhodné umístění senzorů ve sledovaných prostorách.

Mírné vylepšení tohoto postupu přináší **aktivní** způsob lokalizace, jehož data obsahují krom síly signálu i jeho kvalitu. Kvalita je vyjádřena v procentech a jde o údaj poměru šumu oproti síle signálu. Může tedy nastat situace, že se zařízení nachází blízko přístupového bodu, naměří tedy vysokou sílu signálu, ale kvalita je kvůli překážkám nízká. V takovém případě může být za „nejbližší“ kotvu vybrána ta, která nemusí naměřit nejvyšší sílu, ale signál od ní je kvalitnější. Na základě řady experimentů byla stanovena následující rovnice, jejíž výsledek je určení „míry“ s jakou dojde k přiřazení zařízení k senzoru (resp. lokaci).

$$value = signal\_strength + (100 - quality) \quad (4.1)$$

Výsledná hodnota je tedy síla signálu a přičtený „doplňek kvality“. Senzor určující výslednou lokaci je pak ten, jehož vypočtená hodnota je minimální.

Samozřejmě je i tento způsob zatížen nepřesnostmi. Mimo již zmíněné nedostatky je také problémem, že tato metoda se spoléhá na oddělení jednotlivých rozlišovaných lokací překážkou. Ať už jde o stěnu, nábytek, nebo cokoli dalšího. Pokud by ve větší volné místnosti bylo potřeba oddělit dvě lokace, může dojít k chybám v závislosti na umístění jednotlivých

přístupových bodů. Nicméně výzkum odhalil, že převažují případy, kdy jsou jednotlivé lokace logicky rozděleny fyzickými překážkami, a tak i kvalita signálu hraje významnou roli.

Poslední možností, jak by bylo možné provádět lokalizaci z uložených dat, byla v době návrhu uvažována **trilaterace** (popsána v sekci 2.1.1). A to z dat naměřených jak ze senzorů, tak od zařízení. Systém by tak dostal do vínku možnost určit podoblast v dané lokalitě. Zároveň by bylo možné rozdělit oblast pokrytou jedním senzorem, na několik menších, bez nutnosti přidávání dalších přístupových bodů. Pro tento účel mělo v původním návrhu sloužit čerpání informace CSI, popsané v teoretické části 2.3.2. Mohlo by tak být dosaženo velice slušné přesnosti s mediánem chyby menším než jeden metr. Bohužel se ukázalo, že CSI není na drtivé většině Wi-Fi modulů vůbec (nebo jen velice těžko) dostupné, což je uvedeno v návrhu systému (kapitola 3). Bylo by také možné pro trilateraci použít pouze standardně naměřenou RSSI, ale při testech vyšlo najevo, že určení vzdálenosti dosahuje chyb i přes 5 metrů. Což je skutečně naprosto mimo přijatelné hodnoty, neboť se pohybujeme v interiérech malých a středních firem.

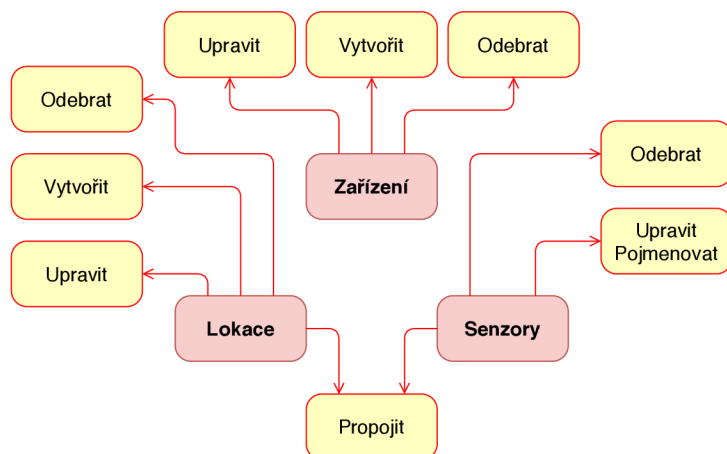
Daná technika by také vyžadovala další vrstvu nastavení a dat. Nejen že by bylo nutné mít změřené přesně pozice jednotlivých senzorů, ale také by musely být nějakým způsobem uvedeny rozměry a tvary okrajů jednotlivých lokalit. Šlo by například s výhodou využít některou z prostorových postrelačních databází apod., nicméně z důvodu nepoužitelné přesnosti byla tato možnost ze systému vypuštěna.

## Prezentační část

Posledním modulem užívaným pro obsluhu systému je prezentační program. Implementována jsou dvě paralelně jedoucí řešení. Jedním z nich je API (Application programming interface), tedy pouze programové rozhraní. Slouží především pro komunikaci s dalšími systémy nebo programy. Komunikace probíhá ve formě JSON požadavků a odpovědí a příklad takové komunikace je uveden v příloze C. Druhé řešení naopak cílí spíše na lidské uživatele systému. Jedná se o web server, s přístupem k centrální databázi a plní v podstatě roli grafického rozhraní. Je spuštěn také jako samostatné vlákno při startu serverové centrální instance a příklady jsou uvedeny v příloze D.

Obě části jsou velmi jednoduché a nabízí četné možnosti pro rozšíření, nebo přizpůsobení na míru cílovému klientu. Hlavní a nejpodstatnější informace, které musí obě rozhraní poskytovat bude zřejmě výskyt jistého zařízení (identifikovaného dle MAC adresy) v některé z lokací v daném čase. Jde vlastně o cílová data celého systému. Pokud by mělo dojít v uložených datech k rozporu v určení správné lokace při pasivní, nebo aktivní lokalizaci, dává systém přednost té aktivní. Podle provedených výzkumů a testování jde totiž skutečně o přesnější metodu a očekává se, že i uživatel bude těmto datům dávat přednost a to už proto, že musel vynaložit úsilí pro jejich získávání.

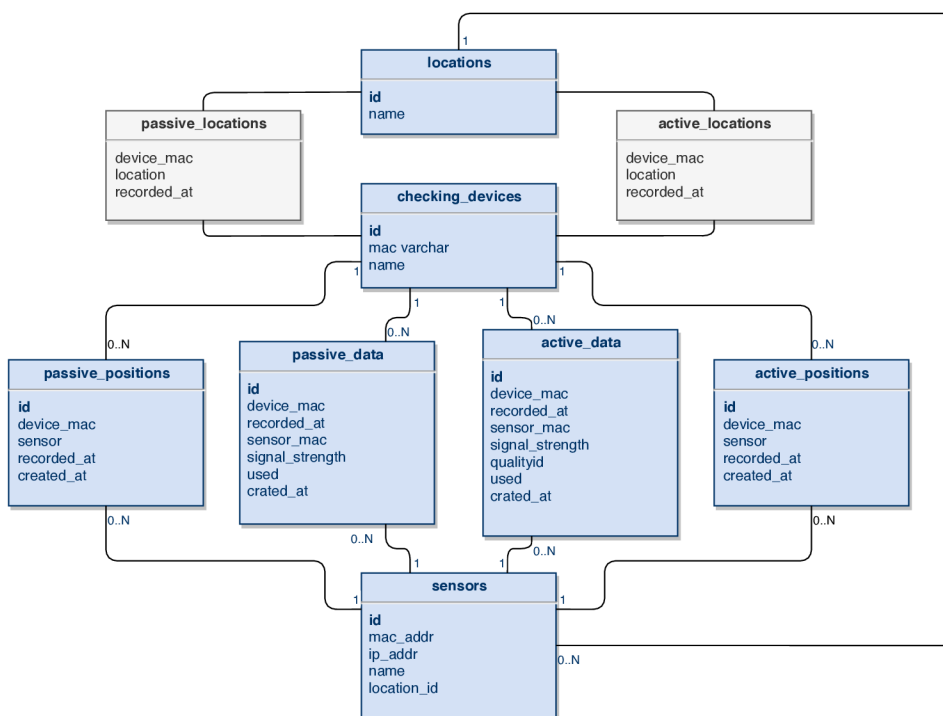
Co se týká provádění změn v datech DB, tak rozhraní ve vytvořené prototypové verzi poskytují akce uvedené na diagramu 4.4. Většina z akcí je plně popsána už svým názvem. Dovolím si tedy objasnit funkcionalitu jen těch méně zřejmých z nich. První takovou akcí je „Upravit/Pojmenovat“ při senzoru. Senzor, tedy přístupový bod, nemůže být do systému přidán uživatelem jako nový záznam v databázi. Nový senzor se do systému přidává sám po zapojení a spuštění. Identifikován je ale pouze MAC adresou, tedy je záhodno přiřadit mu i jednoznačné jméno. Dále jde pak o akci „Propojit“ mezi lokacemi a senzory. Ty jsou vzájemně ve vztahu takovém, že jedné lokaci může být přiřazeno 0 až  $N$  senzorů a toto přiřazení musí určit uživatel. Jako příklad uvedu situaci, kdy mám ve svém interiéru



Obrázek 4.4: Diagram akcí pro úpravu dat, jež poskytují implementovaná rozhraní.

rozvedeny 3 senzory. Dva z nich jsou umístěny ve velké hlavní hale pro pracovníky a jeden potom v menší zasedací místnosti. Musím tedy vytvořit obě tyto lokace v databázi a k nim přiřadit v nich umístěné senzory, abych dosáhl rozlišení těchto dvou prostor.

## Architektura databáze



Obrázek 4.5: Diagram popisující architekturu databáze systému.

Pro doplnění uvádím ještě vytvořenou databázovou architekturu. Na diagramu 4.5 je názorně vidět rozdělení dat získaných pasivním, respektive aktivním měřením. Šedě vyznačené entity `passive_locations` a `active_locations` jsou databázové pohledy. Jde vlastně

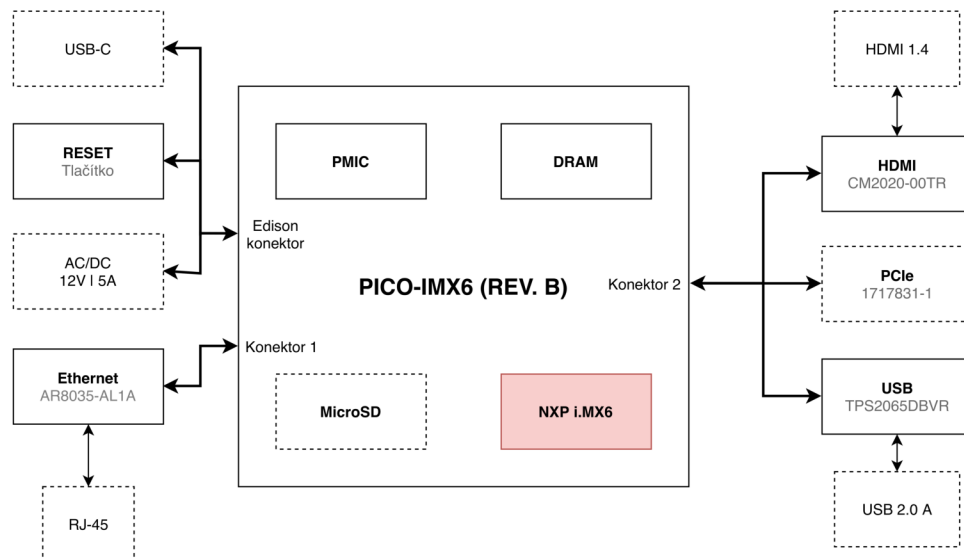


o výsledná data systému, která uživatele zřejmě zajímají nejvíce. Shrnují, ve které lokaci se každé zařízení nacházelo v daném čase. Tabulky se sufixem `_data` uchovávají surová naměřená data včetně kvality a síly bezdrátového signálu. Naproti tomu tabulky končící na `_positions` jsou výsledkem lokalizačního algoritmu, který vybírá jeden výsledný senzor z těch, jež poskytly data v průběhu jednoho měřicího cyklu.

#### 4.2.2 Bezdrátový senzor v roli přístupového bodu

Vytvořený Wi-Fi modul byl sestavován s ohledem na cenu, složitost obvodu a samozřejmě na splnění požadavků ohledně funkcionality. Uvedený blokový diagram (obr. 4.6) názorně zobrazuje jednotlivé části obvodu, které jsou dále podrobněji popsány. Snímek reálného prototypu, s vyznačením těchto částí je uveden v příloze E.3. Jsou zhodnoceny i nejcitlivější nedostatky hardware a pokud to lze, navrženo řešení do příštích verzí. Následuje popis softwaru pro senzor, kde jsou rozvedeny detaily funkčnosti.

#### Hardware senzoru



Obrázek 4.6: Diagram vytvořené první verze senzoru. (Čárkované jsou externí konektory.)

- **CPU Modul** – jak je z diagramu patrné, hlavní a největší částí je modul nazvaný PICO-IMX6<sup>5</sup>, který obsahuje procesor, RAM paměť i konektor paměťové karty. Modul vyrábí firma TechNexion ve spolupráci s NXP, kteří dodávají procesory. Je postaven kolem CPU i.MX6<sup>6</sup>, vybaveným jádrem Arm<sup>®</sup> Cortex-A9. Protože odpadla možnost CSI zpracování, není ani takto silné jádro potřeba a mohla by se slabším HW být snížena cena. RAM paměť s kapacitou 1 GB poskytuje více než dostatečné prostředky pro běh operačního systému. Modul byl zvolen především z důvodu zjednodušení obvodu prototypu. Při sériové výrobě by ale velice nepříznivě ovlivnil cenu

<sup>5</sup><https://www.technexion.com/products/pico-compute-modules/detail/PICO-IMX6>

<sup>6</sup><https://www.nxp.com/products/processors-and-microcontrollers/arm-based-processors-and-mcus/i.mx-applications-processors/i.mx-6-processors/i.mx-6solo-processors-single-core-multimedia-3d-graphics-arm-cortex-a9-core:i.MX6S>

celého senzoru a je tak na místě uvažovat o pevném umístění procesoru na tištěný spoj. Operační systém potom běží z připojené MicroSD karty, která také slouží jako nevolatilní úložiště. Modul je ke zbytku senzoru připojen třemi 70-pin Hirose konektory, z nichž jeden hlavní odpovídá standardu pro Edison konektor<sup>7</sup> od firmy Intel<sup>®</sup>. Další dva jsou doplňkové pro různé externí signály.

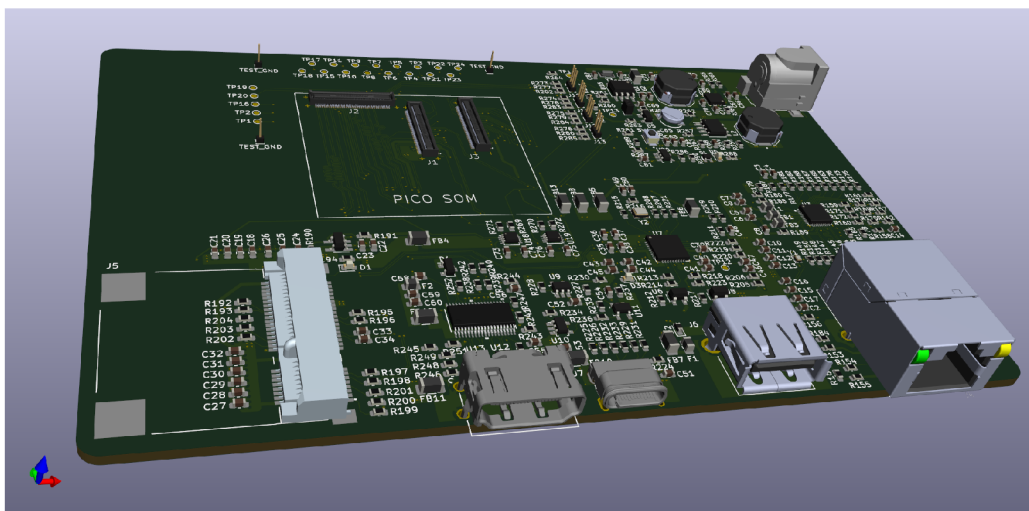
- **Mini PCIe** – připojený konektor přímo k procesoru. Nejsou využity všechny nabízené prvky standardu, jako jsou SIM karta, nebo USB. Procesor dokáže pracovat i s novějšími kartami Wi-Fi, je tedy možné využít dnes dostupné moduly na trhu. Důvod proč bezdrátový čip není přímo součástí obvodu je jednak zjednodušení a pak také cena, i když pokud bude potřeba pro použití dokoupit i Wi-Fi kartu, nemusí být poslední tvrzení pravdou. Díky připojenému pouze konektoru, je ale poskytnuta modulárnost senzoru a je tak možné vyměnit kartu za novější typ, bez nutnosti pořízení nového lokalizačního modulu. Což ovšem z komerčního hlediska pro potencionálního výrobce systému není příliš výhodné. Protože je také rozhraní dostupné pouze jedno, je senzorová síť omezena na jednu frekvenční hladinu. Což už je znatelný nedostatek.
- **Napájení** – není kritickou oblastí návrhu. Sensory budou většinou umístěny ve vnitřních prostorách s dostupnými zdroji energie. Wi-Fi není příliš úsporné, a tak jedinou reálnou volbou se zdá být napájení přímo z veřejného elektrického obvodu. Baterie, nebo jiné mobilní zdroje napájení by byly nepraktické. Napájecí konektor je standardního typu 5.5/2.1, tedy běžný adaptérový s voltáží 12 V. Zde je veliký prostor pro rozšíření, neboť napájecích technik ze sítě lze využít hned několik.
- **RESET tlačítko** – jako standardní a v podstatě jediný ovládací prvek bude připraveno tlačítko s funkcí *RESET*.
- **Ethernet** – pro komunikaci s centrálou a přenos internetového provozu, je modul osazen Ethernet radičem a konektorem. Je tak možné jej připojit do lokální sítě kabelem.
- **HDMI** – konektor HDMI pro možnost získání grafického výstupu slouží u prototypového kusu zejména pro ladící účely. Ve finální komerční podobě senzoru by mohl být vypuštěn.
- **USB-C** – konektor typu C připojený přímo k procesorovému modulu je v režimu OTG (On-The-Go). Je tedy schopný zastávat nejen roli hostujícího USB zařízení, ale také se dokáže přizpůsobit a sloužit jako podřízená strana komunikace. Slouží zejména k přístupu k datům za běhu, ale šlo by tohoto rozhraní využít i pro napájení. Na prototypu ovšem toto není implementováno.
- **USB-A** – spíše jako doplněk je na obvod přidán ještě USB konektor typu A v host režimu. Může sloužit jako dodatečná odpojitelná paměť, nebo doplňující ovládací periferie.

Model desky tištěného spoje i schémata byla vytvořena v programu KiCAD<sup>8</sup>, který byl zvolen díky tomu, že má volně šiřitelné zdrojové kódy a poměrně solidní základnu uživatelů, tedy i návodů, skriptů a rozšiřujících modulů. Deska samotná je potom složena

<sup>7</sup>[https://www.intel.com/content/dam/support/us/en/documents/edison/sb/edison\\_pb\\_331179002.pdf](https://www.intel.com/content/dam/support/us/en/documents/edison/sb/edison_pb_331179002.pdf)

<sup>8</sup><https://kicad-pcb.org/>

ze čtyř vrstev. Horní a spodní slouží pro vedení datových signálů, z vnitřních vrstev je pak na jedné „rozlitý“ zemnicí polygon a na druhé jsou rozvedeny napájecí polygony různých napěťových větví. Všechny součástky jsou umístěny pouze na horní straně desky, aby bylo co nejvíce zjednodušeno osazování. Součástí je i celá řada testovacích plošek pro účely ladění. Oba tyto prvky ovšem negativně ovlivňují rozměry výsledného výrobku. Komerční prodávaná verze by tedy musela být notně upravena před uvedením do výroby. 3D model tištěného spoje vygenerovaný programem KiCAD je pro představu k vidění na obrázku 4.7



Obrázek 4.7: Vygenerovaný 3D model programem KiCAD.

Realizovaný prototyp je před použitím nutné vybavit operačním systémem. K tomu slouží čtečka paměťových Micro SD karet umístěná na procesorovém modulu. Tento způsob je výhodný zejména při vývoji a testování, neboť lze pohodlně provádět změny nastavení OS i mimo vytvořený obvod. Po vložení paměťové karty a připojení CPU modulu na desku může být senzor spuštěn. Červeně zbarvená dioda indikuje připojení napájení, zelená potom start bootovací sekvence. Před spuštěním operačního systému by také měla být připojena PCI Wi-Fi karta, protože implicitně je její připojení rozpoznáno pouze při startu. Pokud je připojen také ethernetový kabel, není potřeba již nic dalšího. Operační systém je nastaven tak, aby po zahájení provozu přemostoval síťový provoz z kabelu do Wi-Fi, nastavil senzor do režimu přístupového bodu, přidal monitorovací rozhraní a spustil obslužnou aplikaci. Do několika sekund po zapnutí by se tedy měl senzor sám ohlásit na server a začít s pasivní lokalizací.

Představená první verze senzoru zřejmě neoslíní. Kvůli zjednodušení výroby byl použit silně a zbytečně naddimenzovaný procesorový modul, který zásadně ovlivňuje cenu. Také některé prvky a rozhraní jsou zařazeny do výbavy speciálně pro vývojové účely (například USB, HDMI, testovací výstupy atp.) Řada nedostatků senzoru byla již uvedena dříve v textu, jako je nutnost kabelového připojení k síti, nebo omezení na pouze jednu frekvenční hladinu Wi-Fi standardu.

S ohledem na nedostatky si dovoluji uvést několik návrhů změn další případné verze modulu, došlo-li by na jeho realizaci.

- Rozhodně by bylo vhodné přidat CPU přímo jako součást modulu. Nemusí být nikterak výkonný. Je potřeba pouze provozovat a přemostovat internetový provoz, plus poskytovat možnost spuštění vlastního nenáročného programu. Například při použití

operačního systému OpenWRT<sup>9</sup>, nebo dokonce na míru sestaveného linuxového jádra (např. pomocí Yocto Project<sup>10</sup>) je možné osadit senzor i procesorem nižší, a tedy levnější třídy, než je tomu u prototypu. Také RAM paměť by měla dostačovat i např. velikosti 128 MB (doporučené minimum pro OpenWRT).

- Ani využívání Micro SD karty jako úložiště by v produkční verzi nebylo vhodné. Doporučil bych jej nahradit vestavěnou pamětí (například typu eMMC) s již předchystaným operačním systémem.
- Napájení by jistě mohlo nabídnout lepší variabilitu. Umožnit například přivádět proud přes USB, nebo pomocí ethernetového připojení (PoE, Power over Ethernet) by mohlo pomoci v elektronickou síť nepokrytých oblastech. Nicméně stále platí nevýhody neúsporného Wi-Fi, tedy mobilní napájení v daném standardu není příliš praktické.
- MiniPCIe konektor na stávající verzi sice nestojí mnoho a dovoluje jistou modularitu a svobodu. Nicméně nevýhodou mohou být odlišné vysílací výkony karet různých výrobců (horší přesnost lokalizace) a nutnost dokupovat takové prvky. Věřím že by se vyplatilo zakomponovat Wi-Fi modul přímo na desku senzoru a to tak, aby mohly být pokryty obě frekvenční hladiny (2.4 GHz i 5 GHz). Nicméně konektor by mohl zůstat, jako možné bezdrátové připojení do lokální sítě, tedy jako náhrada za kabel. Toto ale může nepříznivě ovlivňovat zarušení síťového média, proto doporučuji ponechat tuto možnost pouze jako variantu. Také volba M.2 konektoru by při jeho stoupající popularitě měla být uvážena.
- Ve finálním sériově vyráběném modulu by grafický výstup postrádal smyslu. Měl by ale být nahrazen například sériovou linkou, pro zachování možnosti provádět diagnostiku, či upravovat nastavení.

## Softwarová výbava senzoru

Díky naddimenzovanému procesorovému modulu uvedenému v předchozí části textu, není problém provozovat na senzoru operační systém. Pro řešení prototypu byl zvolen GNU/Linux Ubuntu s jádrem Debian, který byl ale později nahrazen Linuxem kompilovaným pomocí Yocto frameworku. Výrobce neposkytuje v předgenerovaných distribucích podporu PCI modulů a tak musela být vytvořena speciální verze jádra. Operační systém tohoto typu dokáže pohodlně pracovat i na pouhých 200 MB RAM paměti, a to i proto že prováděné operace měření a zasílání dat nejsou nijak výrazně náročné.

Aplikace řídící lokalizaci je potom, stejně jako centrální instance, napsaná v jazyce python. Zde je s výhodou využita nejen multiplatformnost, ale také poměrně pohodlná práce s řetězci a voláním systémových nástrojů.

Před samotným popisem funkcí řídicích skriptů je vhodné uvést proces inicializace senzoru. Ve finálním produktu by samozřejmě pro tuto fázi nebylo místo. Prodávané moduly by byly předchystané tak, aby je šlo ihned po zakoupení použít v systému. Pro pochopení funkcionality se ale hodí průvodní nastavení uvést.

Po instalaci potřebných nástrojů, jako je interpret jazyka python3 s potřebnými moduly a další využívané programy (*hostapd*, *arpalert* atd.), je možné popsat konfiguraci v následujících několika bodech.

---

<sup>9</sup><https://openwrt.org/>

<sup>10</sup><https://www.yoctoproject.org/>

1. První je úprava síťových rozhraní. Je přidáno přemostění s názvem `br0`, které zajišťuje že síťový provoz z kabelového spojení, bude dostupný i ve vysílané bezdrátové síti. A to včetně internetu.
2. Dále je pak přidáno bezdrátové rozhraní `mon0`, jež slouží pro monitorování a zaznamenávání provozu na síti pro účely měření.
3. Program **hostapd**<sup>11</sup> je využit k tomu, aby mohl senzor pracovat jako „přístupový bod“. Zařízení se tak mohou k němu a tím i k bezdrátové lokalizační síti (skrže `br0`) připojovat. Wi-Fi čipset je tak nastaven do režimu `access-point`.
4. Pro pasivní monitorování provozu sítě je využit systémový program **iw**<sup>12</sup>. Ten se snaží na Wi-Fi NIC modulu zprovoznit druhý režim s názvem `monitor`. Bohužel jen málokterý čipset dokáže provozovat tyto dva režimy zároveň, nelze tak použít libovolnou Wi-Fi kartu na trhu.
5. Posledním krokem inicializace je pak konfigurace programu **arpalert**<sup>13</sup>, jež sleduje provoz na síti a ukládá do souboru mapování MAC adres a jim přiřazených IP. Taková informace je využita k vytváření umělého provozu (aplikací **fping**<sup>14</sup>), který je možné sledovat a měřit.

Jakmile je inicializace úspěšně dokončena, je senzor připraven na běžný chod. Po startu systému je automaticky spuštěna samotná python aplikace, jejíž funkcionality je popsána v následujícím textu. Podobně jako u centrální instance i zde je využit modul `config_local` pro načítání a ukládání nastavení. Zejména se jedná o čísla portů, název lokalizační sítě, nebo délku měřicího cyklu. Pomocí tohoto modulu (a tedy do JSON souboru) jsou také za běhu ukládány MAC adresy zařízení, jež chce uživatel v síti lokalizovat.

Prvním krokem aplikace je přihlášení senzoru do systému. To zajišťuje smyčka, která se snaží odeslat UDP broadcast zprávu a kontaktovat tak server. Pokud nedostane odpověď, vyčká 10 sekund a pokus opakuje. To umožní spustit senzory dříve než serverovou aplikaci. Jakmile server v síti přijme tuto broadcast zprávu, zašle odpověď, která obsahuje mimo jiné i seznam MAC adres zařízení, jež má systém vyhledávat. Tento seznam vytváří uživatel obsluhou centrálního uzlu.

Po uložení nastavení pak program provede synchronizaci systémového času podle serverové instance. Pomocí protokolu NTP získá čas od serveru a nastavením systémové aplikace `timesyncd` zajistí, že bude synchronizace prováděna vždy maximálně po deseti minutách provozu. Všechny běžící připojené senzory se tak řídí stejně nastavenými hodinami.

Následně senzor přechází do hlavní programové smyčky. Ta je spuštěna jako proces nazvaný jednoduše `scanning`. Před samotným měřením ovšem hlavní proces čeká na okamžik spuštění měřicího cyklu. Čekání je implementováno pomocí python modulu **pause**<sup>15</sup>, kterému je předán absolutní čas, do kterého má být program uspán. Ten je určen následujícím výpočtem (4.2).

$$next = timestamp + (cycle\_length - (timestamp \bmod cycle\_length)) \quad (4.2)$$

Funkce implementující tento výpočet přičte k aktuálnímu času v sekundách (`timestamp`) zbývající délku cyklu (`cycle_length`) do dalšího okamžiku spuštění, jež určuje zbytek po

<sup>11</sup><http://w1.fi/hostapd/>

<sup>12</sup><https://wireless.wiki.kernel.org/en/users/documentation/iw>

<sup>13</sup><https://www.arpalert.org/arpalert.html>

<sup>14</sup><https://fping.org/>

<sup>15</sup><https://github.com/jgillick/python-pause>

dělení času délkou cyklu. Tím je určena absolutní časová hodnota, kdy má dojít k dalšímu spuštění měřicího cyklu. Protože mají všechny senzory synchronizovaný systémový čas, dochází ke spuštění ve stejný okamžik. Minimální doba cyklu je stanovena na 5 sekund, protože měření trvá 3 s a nějaký čas je potřebný pro sestavení a odeslání zprávy. Jakmile nastane určený čas, proces `scanning` započne sběr údajů o provozu pomocí dvou následujících paralelně spuštěných podprocesů.

- Jeden z běžících načítá seznam k sobě (jakožto přístupovému bodu) připojených zařízení a jejich MAC adres. Dle mapování z programu `arpalert` se potom každé zařízení pokusí několikrát za sebou oslovit zprávou ICMP protokolu, tedy „pingem“. Bezdrátová zařízení omezují provoz Wi-Fi sítě na minimum z důvodu šetření baterie. Je tak nutné vyvolávat umělou komunikaci, která pak může být monitorována. Cílem je, aby tato komunikace byla zachycena i na ostatních senzorech a tím mohlo dojít k porovnání z více zdrojů.
- Druhý dílčí proces tedy zajišťuje monitorování provozu na síti. Na to je použit python modul `ScaPy`<sup>16</sup>, který po 3 sekundy zaznamenává všechny médiiem přenášené pakety. Je jich samozřejmě obrovské množství (zejména pokud jsou zařízení aktivní na internetu), takže je po ukončení skenování filtruje. Vybírá jen ty, které mají MAC adresu odesílatele uvedenu v seznamu lokalizovaných zařízení a obsahují hlavičku `RadioTap`<sup>17</sup>. To jsou metadata přidávaná k bezdrátovému provozu, z nichž k účelu lokalizace využívají senzory zejména sílu signálu. Nejde ale o data přenášená jakožto součást provozu. Jde spíše o doplnění až na zařízení příjemce, tedy přístupového bodu. Problémem tohoto přístupu je, že není součástí *IEEE 802.11* standardu. A není tedy zaručena 100% správnost, nebo dostupnost těchto údajů. Jde ale o jedinou možnost, jak pasivní (tedy bez přičinění vysílače) měření provádět. I proto je v systému umožněna aktivní lokalizace, která zajišťuje plnou spolehlivost (i když ne přesnost).

Zjednodušeně lze tedy říct, že v měřícím cyklu každý senzor vyvolá komunikaci k sobě připojeným zařízením, ale zároveň sleduje provoz na síti od všech zařízení v promiskuitním módu Wi-Fi karty. Získané pakety pak program rozdělí podle jednotlivých MAC adres lokalizovaných zařízení a z jejich zjištěných sil signálu spočítá průměr.

Po dokončení operace skenování a měření je sestavena zpráva obsahující získaná data. Ta je odeslána na server ve formátu JSON. Příklad komunikace senzor–server je uveden na obrázku 4.2. Součástí odpovědi ze serveru je opět seznam MAC adres hledaných zařízení a tak může být před další iterací toto nastavení aktualizováno.

Z popisu vyvstávají jistě některé nedostatky. Jedním takovým může být nedostupnost, nebo nesprávnost `RadioTap` hlavičkových údajů. Nenalezl jsem ale jiný možný postup pasivního skenování a je tedy bohužel nutné se na tuto techniku zkrátka spolehnout. Hodí se zde také uvést fakt, že aby bylo možné sledovat síťový provoz, musí být všechny přístupové body nastaveny na stejný frekvenční kanál. To může vést ke vzájemnému rušení. Testy ale neodhalily, že by docházelo k výraznějšímu omezování provozu. Pokud tedy bude rozmístěno „rozumné“ množství senzorů, nemusel by tento problém nastat.

### 4.2.3 Aktivní přispění lokalizovaného zařízení

Pro zlepšení výsledků lokalizace, poskytuje systém možnost pokusit se o aktivní způsob získávání dat. Jde o metodu, kdy samo lokalizované zařízení aktivně vyhledává přístupové

<sup>16</sup><https://scapy.net/>

<sup>17</sup><https://www.radiotap.org/>

body v okolí a zjištěné údaje zaslá na server. Přidaná hodnota oproti pasivnímu způsobu je zejména v možnosti měřit navíc kvalitu signálu. Ta vyjadřuje poměr síly signálu k lokálnímu šumu a může pomoci zpřesnit výsledné určení lokace. Je také vyloučena nespolehlivost RadioTap hlaviček, protože skenování okolních bodů a jejich vysílacích sil je standardizovanou funkcí Wi-Fi.

Ke zprovoznění stačí jednoduchých pár kroků, které jsou snadno implementovatelné. Vytvořena byla i zkušební aplikace, pro systém GNU/Linux, a její funkcionalitu zde uvádím jako příklad.

1. Jako první je potřeba zjistit, jakou má přiřazenu IP adresu server, kam hodláme výsledky posílat. Samozřejmě ji lze zadat ručně, ale je také možné využít odeslání broadcast datagramu, podobně jako je tomu u počátečního zapsání senzorů. Protože na lokální síti odpoví na tento požadavek pouze serverová instance, můžeme z odpovědi IP adresu získat.
2. Jakmile máme k dispozici adresu, může být spuštěna hlavní smyčka.
  - (a) Provedeme sken přístupových bodů v dosahu. Na systémech Linux je to možné příkazem `iwlist`, ale operace je to standardní a všechna zařízení provozující Wi-Fi musí být tohoto skenování schopna.
  - (b) Zjištěné informace je potřeba filtrovat. Chceme jen přístupové body, které jsou součástí lokalizační sítě (mají správné SSID) a u těchto jde o informace nesoucí MAC adresu senzoru, sílu signálu a jeho kvalitu.
  - (c) Data je nutné formátovat a přidat informace o čase záznamu a MAC adresu lokalizovaného zařízení.
  - (d) Přichystanou zprávu odešleme na server. Požadovaný formát zprávy je uveden ve výpisu 4.1.

```
1 {
2   "command": "data-device",
3   "mac": "XX:XX:XX:XX:XX:XX", # Mac adresa zarizeni
4   "time": "2020-07-06 13:25:03", # Cas zaznamu
5   "data": [
6     {
7       "mac": "XX:XX:XX:XX:XX:XX" # MAC adresa pristupoveho bodu
8       "signal": "56" # sila signalu (dBm)
9       "quality": "85" # kvalita signalu (%)
10    },
11    ...
12   {}
13 ]
14 }
```

Výpis 4.1: Příklad zprávy pro centrální stanici při aktivní lokalizaci. Opět ve formátu JSON.

Iterace je možné (a vhodné) provádět s časovým odstupem, například 5 sekund. Takto zajistíme, že naše zařízení samo napomáhá k dosažení vyšší přesnosti. Je zjevné, že častý sken okolí a odesílání dat, nebudou příliš šetřit baterii. I při pasivním způsobu lokalizace ale musí zařízení velice často vykazovat provoz na bezdrátové síti, tedy ve výsledku by

k ušetření nemuselo dojít, pokud by aktivní způsob nebyl využit. Také je možné získávat a měřit data o okolních stanicích, aniž by bylo zařízení připojené do lokalizační sítě. Nicméně je potřeba v síti odesílat data určená pro centrálu, tedy připojení je doporučeno.

Navržená aplikace nevyužívá v podstatě žádné speciální funkce. Jde o standardní Wi-Fi sken okolí, formátování zpráv a odeslání do sítě. Na nejpopulárnější systémy je samozřejmě možné takovou aplikaci předem vyvinout. Ať už se jedná o mobilní Android, či iOS, nebo i počítačové systémy jako Windows, Mac OS apod. Funkcionalita je však natolik jednoduchá, že zkušený uživatel zvládne aplikaci vytvořit i na nestandardním systému, jež používá zrovna on.



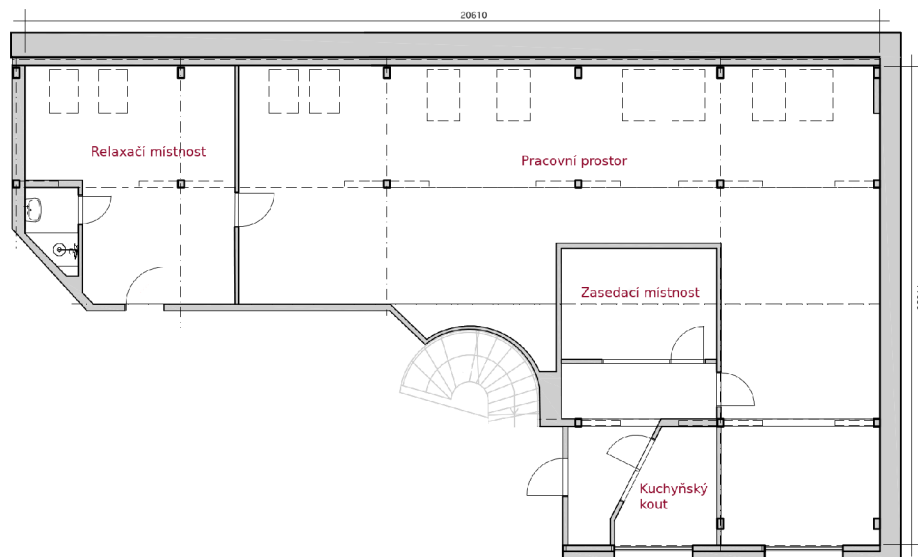
## Kapitola 5

# Testování

Navržený a vytvořený prototyp systému byl testován v kancelářských prostorách zadavatelské firmy v centru Brna. Testu se účastnily 4 osoby, 5 senzorů, 16 lokalizovaných zařízení a čistého času systém běžel bezmála 8 hodin. Vyhodnoceno bylo 57 933 záznamů z více než milionu zachycených paketů, jež byly zaznamenávány v pětisekundovém cyklu. Kapitola v úvodu nastiňuje podobu testovacího prostředí. Pokračuje uvedením testovacích metod a naměřených hodnot, které jsou v závěrečné sekci vyhodnoceny.

### 5.1 Testovací prostředí

Půdorys kanceláře je uveden na obrázku 5.1 a komplex se skládá z velkého otevřeného prostoru pro pracovníky, kuchyňky, zasedací místnosti a odpočívárny. Rozdělení na tři hlavní lokace, které měly být rozlišeny při lokalizování zařízení je potom znázorněno na obrázku 5.2. Bylo tedy nutné zvolit vhodný počet a rozmístění senzorů.



Obrázek 5.1: Půdorys prostor pro testování.

Protože je cena prototypu senzoru příliš vysoká, byl vyroben pouze jediný exemplář. Proto pro účely testování musely být senzory nahrazeny jiným hardware, což ovšem nezpů-

sobilo problémy, protože chod senzoru (jednoduchá python aplikace na Linux OS) dokážou simulovat i běžně dostupné počítače. Důležité bylo pouze dodržet osazení Wi-Fi kartou a „vyvedení“ antén. Příklad takového zástupce je zachycen na snímku E.5. Stroje a bezdrátové Wi-Fi čipsety jsou uvedeny v následující tabulce, přičemž uvedené identifikační znaky slouží pro určení rozdělení přístupových bodů v prostoru na obrázku 5.2. Roli serverové instance převzal notebook Lenovo Y700 se systémem GNU/Linux Ubuntu. Lokální databázi pak poskytl nástroj **XAMPP**<sup>1</sup>.

Tabulka 5.1: Zařízení plnící roli senzorů

| ID | Zařízení             | Wi-Fi čipset             |
|----|----------------------|--------------------------|
| A  | ASUS M50V            | Qualcomm Atheros QCA9880 |
| B  | Lenovo M720Q         | Qualcomm Atheros QCA9880 |
| C  | HP ProDesk 400 G2    | Broadcom BCM43224        |
| D  | ASUS K50IP           | Qualcomm Atheros QCA9880 |
| E  | Intel NUC NUC5i3RYHS | Intel AC 7265            |

Jak je z tabulky patrné, vyrobený prototyp senzoru nebyl při testování použit. Problémy při jeho zprovoznění způsobily že při prováděných testech bohužel nebyl funkční. Návrh i výroba byly časově náročné. Výrobu desky zajistila firma Gatema<sup>2</sup>, nicméně osazení tištěného spoje proběhlo na fakultě pod taktovkou vedoucího práce. I když byl výrobní proces dokončen včas, spuštění a nastavení operačního systému si vyžádalo daleko větší péči než jsem předpokládal. Výrobce procesorového modulu (tedy TechNexion) na svých stránkách nabízí ke stažení předkompilované verze linuxové distribuce Ubuntu. Bohužel žádná z nabízených verzí nemá v nastavení jádra povolenu podporu PCI Wi-Fi modulů a proto jediná cesta byla kompilace vlastní verze kernelu<sup>3</sup>. Samotná kompilace je zejména časově velmi náročná a musela být provedena několikrát po sobě, kvůli iteračnímu přizpůsobování konfigurace. Jakmile byl OS funkční, nastal další problém se zprovozněním nástrojů, které jsou na populárních distribucích běžně k dostání. Některé nástroje tak musely být manuálně kompilovány a konfigurovány (knihovna libcap, nebo arpalert program). Nakonec musela být i samotná obslužná aplikace přizpůsobena speciálnímu OS, kvůli starší verzi python interpretu. Výsledný výrobek je ale funkční a může jako senzor v síti pracovat. Jen se jej nepodařilo zprovoznit před prováděnými testy.

Firma také poskytla sadu zařízení do role lokalizovaných subjektů. Tabulka 5.2 obsahuje jejich názvy a číselnou identifikaci pro přiřazení rozmístění na obrázku 5.2. Zařízení byla po kanceláři umisťována takovým způsobem, aby se jednak nacházela na rozhraních lokací, ale také v místech kde by je ukládali uživatelé v běžném reálném provozu firmy.

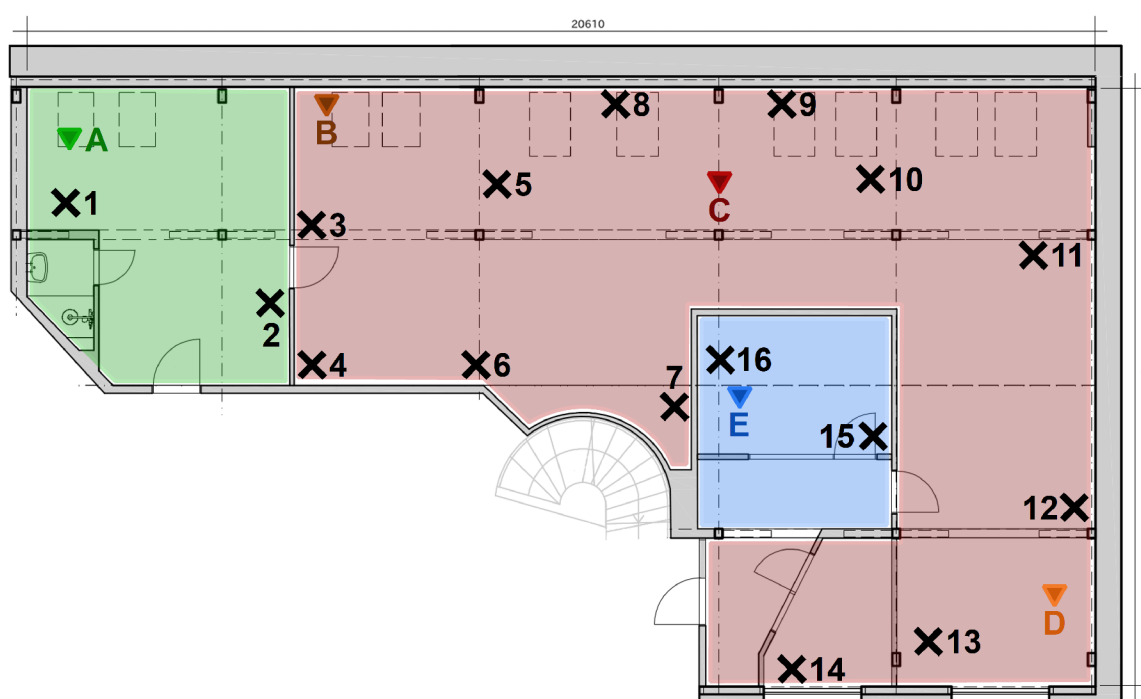
<sup>1</sup>XAMPP: <https://www.apachefriends.org/index.html>

<sup>2</sup><https://www.gatema.cz/>

<sup>3</sup><https://github.com/TechNexion/tn-imx-yocto-manifest>

Tabulka 5.2: Lokalizovaná zařízení

| ID | Zařízení            | ID | Zařízení                       |
|----|---------------------|----|--------------------------------|
| 1  | Samsung Galaxy S10  | 9  | Samsung Galaxy S4              |
| 2  | Motorola Nexus 6    | 10 | Samsung Galaxy S5              |
| 3  | LG L Bello D331     | 11 | Samsung Galaxy S2              |
| 4  | Apple iPhone 5      | 12 | Apple iPad 5 (tablet)          |
| 5  | Huawei Nexus 6P     | 13 | Samsung Galaxy Tab S2 (tablet) |
| 6  | Apple iPhone 6 Plus | 14 | Sony VAIO (notebook)           |
| 7  | LG Nexus 5          | 15 | Lenovo A1000                   |
| 8  | Xiaomi Redmi 3      | 16 | Lenovo TAB P10 (tablet)        |



Obrázek 5.2: Rozdělení lokací, senzorů a zařízení v prostoru. Písmena označují senzory, číselné hodnoty pak lokalizovaná zařízení.

S výše uvedeným a připraveným testovacím prostředím byl zahájen provoz systému. Proběhlo několik kol měření a zaznamenávání, z nichž bylo žádoucí vysledovat zejména úspěšnost systému při určení lokace, v níž se zařízení nachází. Sledovány byly hodnoty naměřené pasivním způsobem, tedy bez jakýchkoli změn na zařízeních, ale i aktivní cestou (kdy zařízení aktivně odesílá do systému data). Měření probíhalo za klidového stavu, kdy zařízení držela statické pozice, byly zavřené dveře a pohyb věcí a lidí v testovacím prostoru byl minimální, ale i v dynamičtějším prostředí, kde byla přesouvána nejen samotná zařízení, ale také drobný nábytek (židle, krabice, šanony atp.), osoby se pohybovaly v prostoru a byly otevírány a zavírány dveře. Bylo testováno, jaký vliv bude mít přesunutí všech sledovaných zařízení k jedinému ze senzorů, nebo jak lokalizace ovlivňuje spotřebu energie na mobilních zařízeních.

## 5.2 Naměřené výsledky

Protože probíhalo několik kol měření, kdy každé bylo zaměřeno na jinou část systému a přineslo jiné výsledky, je i tato kapitola rozdělena na jednotlivé části.

### 5.2.1 Pasivní metoda lokalizace

První kolo měření bylo určeno pouze pro pasivní lokalizaci za statického stavu (zavřené dveře a minimální pohyb). Původní výsledky se zdály být velice nepřesné. Úspěšnost pouze asi 67 % se zdála být až příliš nereálná. Ukázalo se, že senzory prováděly komunikaci s připojenými zařízeními (pomocí ICMP protokolu) pouze třikrát za sebou, což je tak málo provozu, že je jen velmi těžce zachytitelný. Proto došlo po asi hodině testování k úpravě algoritmu.

Nově byly požadavky vysílány padesátkrát za sebou s rozestupem 50 ms. Tím je dosaženo rovnoměrného rozložení komunikace po téměř celé 3 sekundy kdy se všechny senzory pokouší tento provoz zachytit. Původní implementace šetřila s ICMP dotazy, protože panovala obava ze zbytečného zahlcení, což by mohlo vést k omezování chodu internetu v lokalizační síti. Experimenty ale ukázaly (přehrávání videa, hudby, nebo přenos souborů) že nedochází k žádnému výraznému omezení a je tedy možné zvýšit přesnost zatížením komunikačního média bez vedlejších efektů.

Další měření tedy probíhala s upraveným algoritmem umělé komunikace. Při rozdělení na pouhé tři lokace z obrázku 5.2, byla zjištěna úspěšnost určení té správné v 95,1 % záznamů. Výsledek je to poměrně dobrý, ale jde o podmínky takřka ideální. Je nejspíše zřejmé, že nějaký pohyb objektů a osob po kanceláři v reálném provozu prostě bude. Proto bylo další kolo testů zaměřeno na dynamické změny prostředí. V prostorách procházely osoby, byly zavírány i otevírány dveře, přesunoval se drobný nábytek a další objekty. Také lokalizovaná zařízení měnila své umístění, pro navození reálných podmínek.

V takto dynamickém prostředí se přesnost systému přirozeně snížila. A to o více než 10 procent oproti statickému testování. Výsledná hodnota úspěšnosti je tedy 86,25 %. Protože se ale firma zabývá vývojem software, což je poměrně statická práce, dá se očekávat že změny prostředí nebudou natolik intenzivní ani tak časté. Pokud tedy stanovíme, že dynamický provoz, jaký byl navozen při testování se odehrává asi čtvrtinu reálného času (kde čtvrtina je jen smyšlená hodnota založená na zkušenostech z praxe ve firmě, nejde o přesně změřené číslo), můžeme celkovou výslednou přesnost vyjádřit pomocí váženého průměru následovně.

$$\frac{75 * 95,1 + 25 * 86,25}{75 + 25} = 92,89 \text{ [%]} \quad (5.1)$$

Výsledek je to stále slušný ovšem další faktor, který může tuto hodnotu negativně ovlivnit je jednak rozložení senzorů ale i rozdělení prostoru na jednotlivé lokace. V testovacím příkladu (obr. 5.2) jsou lokace odděleny fyzickou překážkou, tedy stěnou. Průchod bezdrátového signálu příčkou (i když jen ze sádkartonu) zapříčiní jeho výrazné zeslábnutí. Není pak velký problém určit, ke kterému ze senzorů je signál nejsilnější. Chybné vyhodnocení je v takovém případě většinou způsobeno nezachycením provozu než jeho nepřesnou hodnotou.

Pokud ale navodíme scénář, kdy je požadováno rozlišovat při lokalizaci i oblasti, kterým fyzické ohraničení chybí, je úspěšnost lokalizace drasticky snížena. Jako příklad mohu uvést rozdělení na lokace podle obrázku 5.3. Velký otevřený pracovní prostor je zde rozdělen a každý senzor má přiřazenu vlastní lokaci.



Obrázek 5.3: Příklad rozdělení otevřeného prostoru na jednotlivé lokace.

Chybné vyhodnocení lokace se děje nepřekvapivě zejména na hraničních oblastech (například zařízení 10 a 11, nebo 5 a 8). Zde už nejde o problém zachytávání provozu, ale skutečně se stává, že je několik vzorků měření určeno do jedné lokace a několik do druhé. A poměrně nezdárka je chybná téměř polovina hodnot. Vyjádřeno čísly, úspěšnost při rozložení uvedeném na obrázku 5.3 ve statickém prostředí dosáhla 80,24 % a v dynamickém dokonce pouze 70,3 %. Je tedy jasné, že rozdělení prostor, které nejsou fyzicky odděleny bude problém. Vylepšit hodnoty je možné například vhodnějším umístěním senzorů a nastavením jejich antén. Teoreticky by mělo pomoci nastavit senzory tak, aby v dané přiřazené lokaci byla umístěná zařízení vždy v přímé viditelnosti k danému senzoru, a to pouze k tomuto jednomu. Ne vždy je ale možné toto zajistit. Navíc takový postup by vyžadoval určitou míru testování a nastavování před samotným provozem což vzhledem k požadavkům není žádoucí. Nicméně je jasné, že úvodní nastavení systému, rozložení senzorů a rozdělení prostoru na jednotlivé lokace bude mít poměrně zásadní vliv na přesnost výsledků.

## 5.2.2 Aktivní metoda lokalizace

Pro aktivní lokalizaci je zapotřebí účast samotného lokalizovaného zařízení. Potřebný provoz a kroky jsou popsány v kapitole 4.2.3. Aktivnímu způsobu lokalizace nebyla věnována taková pozornost jako tomu pasivnímu, který je hlavním cílem systému. I proto k testování byla použita pouze dvě zařízení. Jednak notebook SONY Vaio s operačním systémem Fedora a pak také jednodeskový počítač Raspberry Pi 2 s OS Raspbian. Na obou byl spuštěn python program, který měřil a odesílal data opět v pětisekundových intervalech. Rozložení senzorů bylo totožné jako při pasivním měření. Zařízení pak byla přesouvána na různé pozice, přičemž na každé z nich byla ponechána určitý čas. Ve výsledcích je tedy zahrnuta

jak statická pozice, tak i dynamický pohyb zařízení. Jednotlivá místa pro měření byla volena velmi podobná, jako u pasivního způsobu.

Dle očekávání je přesnost v tomto režimu lepší nežli u pasivního měření. Úspěšnost určení správné lokace byla 96,38 %. Jak je patrné, rozdíl (oproti 92,89 %) není nikterak velký. Navíc i při aktivním způsobu lokalizace trpí systém stejnými problémy jako při pasivním. Pokud dojde k rozdělení otevřeného prostoru na jednotlivé odlišované lokace, opět přesnost upadá a naměřené hodnoty úspěšnosti 84,57 % naznačují podobnou míru zhoršení. Aktivní způsob lokalizace tedy může mírně pomoci, ale není samospasný. Je otázkou na uživatele aby rozhodl, zda se vyplatí investovat do úprav na zařízeních v poměru k možnému zisku. I zde platí že rozdělení senzorů a lokací má na výsledky zásadní vliv.

### 5.2.3 Další testované efekty

Jednou z obav při provozu systému byla i nadměrná spotřeba energie mobilních zařízení. Protože Wi-Fi už ze standardu není příliš nakloněno nízké spotřebě, je na mobilních zařízeních zpravidla uspáváno, jak často to jen jde. Náš systém ale vytváří umělou komunikaci se zařízením a tím jej nutí být neustále aktivní na bezdrátové síti.

Měření spotřeby proběhlo za pomoci mobilního telefonu Lenovo A1000 s operačním systémem Android. Telefon byl vždy nabit do plné kapacity a po odpojení 3 hodiny zamčený se zhasnutou obrazovkou. První referenční vzorek byl naměřen při standardním Wi-Fi provozu, tedy nebyla prováděna lokalizace. Spotřebovaná energie rádiem po uplynutí měřené doby byla 3 % (z celkové kapacity baterie). V dalším testu byla spuštěna i lokalizace a to s měřícím cyklem dlouhým 5 sekund. Při takto častém vyvolávání komunikace se spotřeba po třech hodinách vyšplhala na 6 % kapacity. Jde tedy o značný nárůst, jež by skutečně v určitých podmínkách mohl způsobit problém. Třetí test pak proběhl pro ověření, zda by prodloužení intervalu vyvolávání komunikace vedlo k ušetření. Bylo tedy nastaveno opakování po 20 sekundách a proveden poslední test. Výsledná spotřeba Wi-Fi byla bezmála 4 %, tedy je možné tímto způsobem ušetřit. Nicméně záleží na koncovém uživateli, aby našel správný kompromis mezi spotřebou a četností vzorků lokalizace. Pokud se například lokalizace týká vývojových zařízení, která jsou stále umístěna v kanceláři, nebude zřejmě zvýšená spotřeba překážkou.

Jedním z testovaných faktorů byla také odolnost systému. V tomto testu bylo všech 16 zařízení přemístěno do dosahu jediného senzoru. I v takto nevyváženém případě byla naměřená úspěšnost určení správné lokace přes 90 %. To je důkazem že systém je i poměrně robustní.

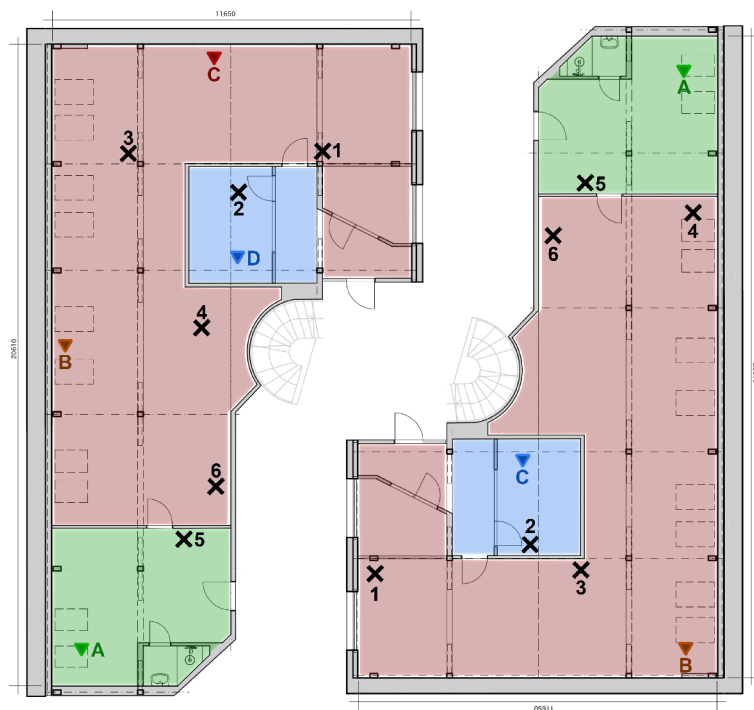
Negativně působící efekt by také mohl být rozdíl, mezi přesnostmi při použití Wi-Fi na notebooku oproti mobilnímu telefonu. Teorie vychází z faktu, že notebooky typicky poskytují vyšší vysílací výkony a mají také lépe umístěné antény díky rozměrům. Na bezmála patnácti stech vzorcích naměřených explicitně pro tento účel se ale ukázalo, že obavy nejsou na místě. Rozdíl v přesnosti nedosahoval ani jednoho procenta a hodnoty ostatních testovacích kol potvrzují, že je jen velmi malý, až zanedbatelný rozdíl v úspěšnosti určení správné lokace mezi notebookem a mobilním telefonem.

Při záznamu komunikace na senzoru je načtena řada paketů se silami signálu, z nichž je na server odeslán pouze průměr. Tato hodnota je pak využita při určení „nejbližšího“ senzoru k zařízení. Pro doplnění jsem se pokusil funkci průměru nahradit funkcí modus, nebo medián. Také jsem do pokusu zařadil metodu, která nejdříve odstranila z řady 3 nejvyšší a nejnižší hodnoty (pro odstínění extrémů) a až poté došlo ke složení dat. Je pravda, že se projevíly změny ve výsledcích, nicméně jich bylo jen zanedbatelné množství. Přesnost

byla ovlivněna jen skutečně neznatelně. Je to způsobeno zejména faktem, že zaznamenané pakety mají stejné nebo velice podobné hodnoty.

Pro doplnění je vhodné také uvést pokusy, kvůli jejichž výsledkům byla zavržena metoda trilaterace (v kapitole 4). Pro úspěšné provedení lokalizace pomocí trilaterace je nutné nejdříve znát co nejpřesnější vzdálenost kotvy od lokalizovaného subjektu. Pokud se budeme držet jednoduchého ztrátového modelu uvedeného vzorcem 2.1, dá se po nastavení správného faktoru útlumu dosáhnout poměrně dobré přesnosti, ale bohužel jen asi do dvou metrů. Ve vzdálenosti 3 m od kotvy už byl medián chyby lokalizace téměř 2 m a ve vzdálenosti 4 m dokonce devítimetrová odchylka. Dal by se samozřejmě vylepšit ztrátový model, aby lépe odpovídal realitě. Problém by ale znovu nastal v případě, že by byl jeden ze senzorů za překážkou (např. zdí) a ostatní by se nacházely v přímé viditelnosti. Tím by opět docházelo k nepřesnostem, a to poměrně zásadním (2,5 m za příčkou je vypočítáno jako 7 m vzdálenost). Aby bylo měření a vyhodnocování dostatečně přesné, musela by každá oddělená lokace obsahovat minimálně tři senzory, anebo by bylo nutné přednastavit systém tak aby dokázal poznat, že je daný senzor za překážkou a upravit tak výslednou vzdálenost. Ani jedno z řešení ale nevyhovuje požadavkům na systém, a proto bylo od tohoto způsobu lokalizace upuštěno.

Další zcela zásadní faktor je kromě rozdělení a nastavení, také počet senzorů. Provedl jsem dvě speciální sady testů, pro zjištění míry negativního dopadu na přesnost při nízkém počtu senzorů. Při těchto testech byl zúžen také počet zařízení, umístěných jen do kritických oblastí.



Obrázek 5.4: Rozložení senzorů a zařízení při jejich sníženém počtu. Levá část obsahuje 4 senzory, pravá pouze 3.

První kolo proběhlo se čtyřmi senzory v prostoru a šesti zařízeními. Jak je vidět na přehledu 5.4 vlevo, musely být upraveny pozice senzorů C a D. Výsledná procentuální úspěšnost správně určených lokací nabyla hodnoty 86,7 %, přičemž bylo testováno jak statické, tak i

dynamické prostředí. (Byl dodržen poměr 75–25 uvedený v kapitole 5.2.1.) Snížení úspěšnosti je očekávatelné. Zajímavý je ovšem fakt, že při rozdělení otevřeného prostoru poklesla účinnost pouze o 3,1 %. Tedy na hodnotu 83,6 %, což je srovnatelná přesnost s aktivním způsobem lokalizace. Zdá se tedy, že v tomto konkrétním případě by snížení počtu senzorů mohlo zmírnit dopady rozdělení lokací bez fyzických překážek. Na druhou stranu jsme ztratili možnost oddělit třetí takovou lokaci, neboť jeden senzor dokáže rozlišovat pouze jednu oblast.

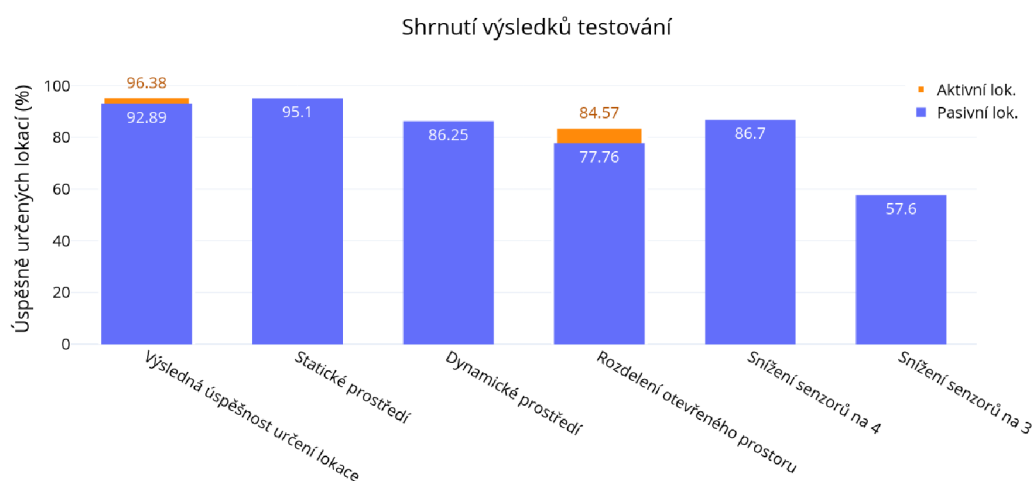
Pro úplnost byl proveden i test pouze se třemi senzory. Úspěšnost určení lokace při rozložení podle pravé části obrázku 5.4 dosahovala pouhých 57,6 % ve statickém prostředí. Taková hodnota je již zcela mimo úroveň akceptovatelných výsledků, proto v testu nebylo pokračováno. Například zařízení označena čísly 4 a 6 byla chybně lokalizována v každém ze získaných vzorků. V kombinaci s rozměry kanceláře (uvedenými na půdorysech) se dá ale dedukovat, že pro dosažení přesnosti, resp. míry úspěšnosti alespoň 90 % by měl jeden senzor pokrývat plochu otevřeného prostoru maximálně 25–30 m<sup>2</sup>.

### 5.3 Zhodnocení výsledků

Pro přehledné shrnutí výsledků byly zařazeny získané hodnoty do grafu (obr. 5.5). Jak je patrné, aktivní způsob lokalizace nepřináší nijak extrémní vylepšení přesnosti. V některých situacích ale může být spolehlivější, jelikož netrpí problémy s promiskuitním monitorováním sítě, které může snadno selhat.

Také je patrné, že negativní vliv dynamických změn prostředí a pohyb lokalizovaných subjektů není až tak zásadní. To se bohužel nedá říci o snaze rozdělit otevřený prostor několika senzory na rozlišované lokace. Zde má pasivní metoda velké rezervy a rozhodně by tato oblast zasloužila další výzkum s cílem zpřesnit výsledky.

Z grafu lze také vyčíst, že snižování hustoty pokrytí prostoru odebráním některých senzorů značně negativně ovlivňuje výsledky. Od určité úrovně už jsou měření nepoužitelná a to i v případě, že jsou rozlišované lokace odděleny fyzickými překážkami. Z výsledků byla odvozena maximální plocha otevřeného prostoru, kterou by měl pokrývat jeden senzor (25–30 m<sup>2</sup>).



Obrázek 5.5: Graf pro shrnutí naměřených výsledků.



Protože systém neurčuje přesnou pozici hledaného subjektu v prostoru, je nemožné stanovit průměrnou odchylku, nebo medián chyby. Je tedy obtížné výsledky porovnávat se systémy uvedenými v teoretické části práce (kap. 2.2.1). Navíc se mi nepodařilo nalézt podobný existující systém, který by se zaměřoval zejména na pasivní metodu lokalizace ve Wi-Fi síti s rozlišením na místnosti. Bohužel tedy nedokážu dost objektivně posoudit, jak si mnou navržený a vytvořený systém „vede mezi konkurencí“.

Ze získaných výsledků lze však vyhodnotit splnění požadavků stanovených v úvodu práce (kapitola 3.1).

- Systém jako celek pracuje skutečně pouze na technologii Wi-Fi. Lokalizovaná zařízení není nutné vybavovat žádnou nástavbou.
- Systém je i poměrně snadno škálovatelný. Přidání/odebrání senzoru je velice snadné a ani jejich úprava, nebo přesunutí nijak nenaruší chod systému. Není třeba provádět žádná složitá nastavování, nebo vytvářet databáze vzorků před spuštěním provozu.
- Rozlišení na úrovni místností je také dodrženo, i když při snaze rozdělovat otevřené prostory je přesnost silně omezena. Pokud ale bude dodrženo oddělení lokací fyzickou překážkou, jsou výsledky akceptovatelné.
- Systém umožňuje pasivní lokalizaci. Není vyžadován žádný dodatečný software, ani speciální postupy pro zařazení mezi lokalizované subjekty.
- Jediným nedodrženým požadavkem je tedy cena zavedení systému. Serverová instance je jednoduchý program, nasaditelný takřka kdekoli v síti, ale problémem jsou senzory. Protože jde o výpočetní zařízení s operačním systémem, je jejich cena poměrně vysoká. Vytvořený prototyp dokonce stál vyšší jednotky tisíc, a i když by došlo k úpravám a sériové výrobě, může toto být vážný problém při přechodu ve funkční produkt. Je i na místě uvažovat o dokupování existujících řešení (např.: Router boardy) a jejich přizpůsobení, minimálně v počáteční fázi zavádění systému na trh.

Získané a naměřené výsledky byly prezentovány zástupcům zadavatelské firmy. Dle jejich vyjádření by nemuselo být nasazení daného systému v kancelářích nereálné. Sice je dle očekávání nepříjemné, že lokalizace neurčuje skutečnou pozici zařízení, ale i rozlišení na místnosti je využitelné. Pokud by došlo k mírným úpravám (například by systém musel být schopen „rozšířit“ stávající Wi-Fi síť, místo jejího nahrazení), dalo by se výsledků využít pro sledování pohybu pracovníků, resp. jejich zařízení, v pracovní době, nebo propojení s docházkovým systémem tak, aby předvyplňoval příchody a odchody zaměstnanců.

Bylo by samozřejmě vhodné provést i další drobné úpravy. Zejména uživatelská rozhraní by si zasloužila pozornost. Web server by měl nabízet širší možnosti nastavení, jako například zabezpečení lokalizační sítě, ale i dat. Také zobrazení výsledků by mohlo být obohaceno o grafické znázornění a ve stávající verzi chybí i možnost detailnější filtrace. Ukládání veškerého provozu je vhodné pro testování, ale ve výsledném produktu by bylo potřeba zajistit odstranění nepotřebných údajů po jejich použití, aby se dodržovala minimální velikost databázových dat. Načítání záznamů pro určení správné lokace by také potřebovalo úpravy. Za předpokladu že jsou senzory i server synchronizovány, neměl by být problém ukládat data s identifikátorem cyklu měření namísto přesného času záznamu (který může být různý od různých senzorů). Načítat pak data podle identifikátoru by bylo snazší i přesnější. Ve stávající verzi totiž může docházet k chybě, při které nejsou načteny a použity všechny záznamy z jednoho měřícího cyklu.

Samozřejmě nesplněný požadavek ohledně nízké ceny senzoru by musel být také vyřešen. Možnost nakoupit existující obvody, zejména jako bazarové zboží, by mohlo snížit cenu natolik, aby se prototypová verze dala reálně nasadit. Silnými stránkami jsou podle vyjádření firmy snadné zavedení a používání systému, nebo možnost skutečně pasivní lokalizace, bez nutnosti jakýchkoli dodatečných úprav zařízení. Také provozování internetu při spuštěné lokalizaci je jistě velkou výhodou.

Pokud by prototyp systému dokázal splňovat požadavky a ukázal by se zájem trhu, dalo by se dle slov zástupce firmy uvažovat dokonce i o nabídnutí systému jako produktu. Nabídka by zřejmě byla postavena velmi obecně, kolem pasivní lokalizace v síti Wi-Fi a až pro potenciální zájemce by se systém upravoval dle zpětné vazby a požadavků.

## Kapitola 6

# Závěr

Práce si kladla za cíl navrhnout, vytvořit a otestovat systém pro určování polohy objektů v interiéru, na bázi bezdrátové sítě Wi-Fi. Byla tedy sestavena přehledová studie základních principů lokalizace a uvedeno několik příkladů vyvíjených a používaných systémů.

Návrh architektury a funkcionality se řídil požadavky, jež byly nastaveny ve spolupráci se zadavatelskou firmou, tedy potenciálním uživatelem systému. Bylo také navrženo schéma zapojení a vyroben prototyp lokalizačního senzoru, který pomocí vytvořeného obslužného programu zajišťuje možnost pasivní lokalizace.

Realizovaný prototyp systému byl následně otestován v kancelářských prostorách zadavatelské firmy v centru Brna. Výstupem práce je tedy funkční systém, který je snadno zaveditelný a škálovatelný, pracuje jen a pouze na bázi Wi-Fi a umožňuje pasivní lokalizaci určující místnost, nebo podobnou lokaci v níž se hledaný subjekt nachází. A to s poměrně vysokou mírou úspěšnosti okolo devadesáti procent.

Vysoká cena navrženého senzoru je tak jedinou překážkou, která brání v nasazení prototypu do provozu. Řešením by mohla být další iterace návrhu, anebo použití některých existujících výrobků s následným přizpůsobením. Pokud by také proběhla sada nezbytných úprav, jako možnost rozšíření stávající zavedené bezdrátové sítě, nebo umožnění nastavení zabezpečení, byla by i zadavatelská firma ochotna systém nasadit a využívat. Zejména jako pomůcku pro automatizování docházky, nebo možnost sledovat pohyb zaměstnanců po pracovišti.

Pozornost by si také zasloužily další části systému. Webové rozhraní by mohlo být obohaceno o možnost filtrování a třídění dat, nebo jejich grafickou prezentaci. Další experimenty by jistě pomohly odhalit nedostatky a vylepšit aktivní způsob lokalizace, který by měl poskytovat stabilnější a spolehlivější výsledky oproti pasivnímu. I vylepšený způsob ukládání a načítání záznamů by mohl pomoci zlepšit přesnost systému. Serverová centrální instance by měla zajišťovat nejen vyhodnocování dat, ale i zpětné odstraňování již nepotřebných záznamů, pro omezení velikosti databáze. Podobných úprav a vylepšení by se jistě našla celá řada. A to zejména při pokusu o dlouhodobý provoz systému.

Další praktická rozšíření mohou následovat. Například přidání možnosti zavedení ve vícepatrových prostorách (tedy třetí rozměr), nebo vývoj aplikací na hlavní užívané platformy pro aktivní způsob lokalizace.

# Literatura

- [1] BAHL, P. a PADMANABHAN, V. N. RADAR: an in-building RF-based user location and tracking system. In: *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies ( Cat. No.00CH37064)*. March 2000, sv. 2, s. 775–784 vol.2. DOI: 10.1109/INFCOM.2000.832252. ISSN 0743-166X.
- [2] BOLLIGER, P. Redpin - Adaptive, Zero-configuration Indoor Localization Through User Collaboration. In: *Proceedings of the First ACM International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments*. New York, NY, USA: ACM, 2008, s. 55–60. DOI: 10.1145/1410012.1410025. ISBN 978-1-60558-189-7. Dostupné z: <http://doi.acm.org/10.1145/1410012.1410025>.
- [3] CHAN, S. a SOHN, G. INDOOR LOCALIZATION USING WI-FI BASED FINGERPRINTING AND TRILATERATION TECHNIQUES FOR LBS APPLICATIONS. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. Červen 2012, XXXVIII-4/C26, s. 1–5. DOI: 10.5194/isprsarchives-XXXVIII-4-C26-1-2012.
- [4] DARDARI, D., LUISE, M. a FALLETTI, E. *Satellite and Terrestrial Radio Positioning Techniques: A Signal Processing Perspective*. Elsevier Science, 2012. ISBN 978-0-123-82085-3.
- [5] GJENGSET, J., XIONG, J., MCPHILLIPS, G. a JAMIESON, K. Phaser: Enabling Phased Array Signal Processing on Commodity WiFi Access Points. In: *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking*. New York, NY, USA: ACM, 2014, s. 153–164. MobiCom '14. DOI: 10.1145/2639108.2639139. ISBN 978-1-4503-2783-1. Dostupné z: <http://doi.acm.org/10.1145/2639108.2639139>.
- [6] GUPTA, D. a JOSHI, S. IN-BUILDING RADIO PROPAGATION AT 900 MHZ IN MULTI STORIED BUILDING. *International Journal of Distributed and Parallel systems*. Listopad 2011, sv. 2. DOI: 10.5121/ijdps.2011.2617.
- [7] HALPERIN, D., HU, W., SHETH, A. a WETHERALL, D. 802.11 with multiple antennas for dummies. *ACM SIGCOMM Computer Communication Review*. leden 2010, sv. 1, č. 40, s. 19–25. ISSN 0146-4833. [Online; navštíveno 2019-01-11]. Dostupné z: <http://www.sigcomm.org/node/2821>.
- [8] KAPLAN, E. D. a HEGARTY, C. J. *Understanding GPS Principles and Applications*. Artech House, Inc., 2006. ISBN 1-58053-894-0.

- [9] KOTARU, M., JOSHI, K., BHARADIA, D. a KATTI, S. SpotFi: Decimeter Level Localization Using WiFi. In: *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*. New York, NY, USA: ACM, 2015, s. 269–282. SIGCOMM '15. DOI: 10.1145/2785956.2787487. ISBN 978-1-4503-3542-3. Dostupné z: <http://doi.acm.org/10.1145/2785956.2787487>.
- [10] KUMAR, S., GIL, S., KATABI, D. a RUS, D. Accurate Indoor Localization with Zero Start-up Cost. In: *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking*. New York, NY, USA: ACM, 2014, s. 483–494. MobiCom '14. DOI: 10.1145/2639108.2639142. ISBN 978-1-4503-2783-1. Dostupné z: <http://doi.acm.org/10.1145/2639108.2639142>.
- [11] MALIK, A. *RTLS For Dummies*. Wiley Publishing, Inc., 2009. ISBN 978-0-470-39868-5.
- [12] PAHLAVAN, K. a LEVESQUE., A. H. *Wireless Information Networks*. John Wiley and Sons, Inc., 2005. ISBN 978-0-471-72542-8.
- [13] SCHILLER, J. H. *Mobile Communications*. Addison-Wesley, 2003. ISBN 0-321-12381-6.
- [14] SCHMIDT, R. Multiple emitter location and signal parameter estimation. *IEEE Transactions on Antennas and Propagation*. March 1986, sv. 34, č. 3, s. 276–280. ISSN 0018-926X.
- [15] SEGEV, J. *IEEE P802.11 – Task Group AZ - MEETINGS UPDATE* [Institute of Electrical and Electronics Engineers]. [Online; navštíveno 10.01.2019]. Dostupné z: [http://www.ieee802.org/11/Reports/tgaz\\_update.htm](http://www.ieee802.org/11/Reports/tgaz_update.htm).
- [16] SZABÓ, P. *Lokalizace počítačových zařízení ve Wifi sítích*. Brno, CZ, 2015. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Dostupné z: <https://www.fit.vut.cz/study/thesis/17942/>.
- [17] VASISHT, D., KUMAR, S. a KATABI, D. Decimeter-level Localization with a Single WiFi Access Point. In: *Proceedings of the 13th Usenix Conference on Networked Systems Design and Implementation*. Berkeley, CA, USA: USENIX Association, 2016, s. 165–178. NSDI '16. ISBN 978-1-931971-29-4. Dostupné z: <http://dl.acm.org/citation.cfm?id=2930611.2930623>.
- [18] WI FI\_ALLIANCE. *Wi-Fi Generations* [Wi-Fi Alliance]. červenec 2013. [Online; navštíveno 10.01.2019]. Dostupné z: <https://www.wi-fi.org/discover-wi-fi>.
- [19] WU, K., XIAO, J., YI, Y., CHEN, D., LUO, X. et al. CSI-Based Indoor Localization. *IEEE Transactions on Parallel and Distributed Systems*. leden 2013, sv. 7, č. 24, s. 1300–1309. ISSN 1045-9219. [Online; navštíveno 2019-01-11]. Dostupné z: <http://ieeexplore.ieee.org/document/6244790/>.
- [20] WU, K., XIAO, J., YI, Y., GAO, M. a NI, L. M. FILA: Fine-grained indoor localization. In: *Proceedings IEEE INFOCOM*. Orlando, FL, USA: IEEE, 2012, s. 2210–2218. ISBN 978-1-4673-0773-4. [Online; navštíveno 2019-01-11]. Dostupné z: <http://ieeexplore.ieee.org/document/6195606/>.

- [21] XIE, Y., LI, Z. a LI, M. Precise Power Delay Profiling with Commodity WiFi. In: *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking - MobiCom '15*. New York, NY, USA: ACM Press, 2015, s. 53–64. ISBN 978-1-4503-3619-2. [Online; navštíveno 2019-01-11]. Dostupné z: <http://doi.acm.org/10.1145/2789168.2790124>.
- [22] YOUSSEF, M. a AGRAWALA, A. The Horus WLAN Location Determination System. In: *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services*. New York, NY, USA: ACM, 2005, s. 205–218. MobiSys '05. DOI: 10.1145/1067170.1067193. ISBN 1-931971-31-5. Dostupné z: <http://doi.acm.org/10.1145/1067170.1067193>.
- [23] ZHU, X. a FENG, Y. RSSI-based Algorithm for Indoor Localization. *Communications and Network*. Leden 2013, sv. 05, s. 37–42. DOI: 10.4236/cn.2013.52B007.

## Příloha A

# Obsah příloženého paměťového média

Obsah příloženého CD/DVD je rozdělen na čtyři hlavní části.

- `Technicka_zprava` nepřekvapivě obsahuje zdrojové kódy pro vygenerování této zprávy. Ta je vytvořena pomocí L<sup>A</sup>T<sub>E</sub>X šablony poskytnuté na stránkách <https://www.fit.vut.cz/study/theses/master-theses/.cs>.
- `Navrh_senzoru` obsahuje jednak zdrojové soubory pro program KiCAD (v. 5.1.6), ale také BOM soubor se seznamem jednotlivých součástí. Součástí jsou i nastavení nutná pro kompilaci kernelu pomocí Yocto frameworku.
- `Zdrojove_kody` obsahuje zdrojové kódy obslužné aplikace systému. V kořenové složce jsou hlavní skripty pro spuštění serverové instance, resp. senzoru. Je přiložen archiv s upravenými verzemi pro zapůjčenou desku SABRE od NXP, ale i pro vytvořený prototyp senzoru. Je zde také přidán SQL soubor s úvodní inicializací databáze. Následuje popis jednotlivých podsložek.
  - `device` obsahuje návrh aplikace pro aktivní způsob lokalizace
  - `install` složka obsahuje skripty a konfigurační soubory pro nastavení prostředí senzoru, na běžném počítači a operačním systému Debian
  - `senzor` a `server` ukrývají zdrojové kódy skriptů obslužné aplikace
  - A obsahem `web_app` je implementace web serveru pro obsluhu systému
- A konečně adresář `Vysledky_testu` obsahuje sadu SQL exportovaných souborů, kdy každý jeden zastupuje jinou vlnu testovacího provozu. Součástí je i ODS soubor se shrnutými výsledky pro lepší přehlednost.

```
xkohou03/
├── Technicka_zprava
│   ├── zadani.pdf
│   ├── xkohou03-Lokalizace-ve-WiFi-30-prilohy-appendices.tex
│   ├── xkohou03-Lokalizace-ve-WiFi-20-literatura-bibliography.bib
│   ├── xkohou03-Lokalizace-ve-WiFi-01-kapitoly-chapters.tex
│   ├── xkohou03-Lokalizace-ve-WiFi.tex
│   ├── template-fig/
│   ├── obrazky-figures/
│   ├── Makefile
│   ├── fitthesis.cls
│   └── bib-styles/
├── Navrh_senzoru/
│   ├── KiCAD/
│   ├── PICO-IMX6-Yocto/
│   └── BOM-DIP-WiFi_Loc_BaseB.ods
├── Zdrojove_kody/
│   ├── server/
│   ├── sensor/
│   ├── install/
│   ├── device/
│   ├── dactyl_pos_init.sql
│   ├── locator-sensor.py
│   └── locator-server.py
└── Vysledky_testu/
    └── VysledkyTestu.ods
```



## Příloha B

# Manuál pro spuštění systému

Systém se skládá z centrální serverové instance a až N senzorů. Je tedy nutné pro ověření základní funkcionality mít tyto dvě entity v provozu. Systém byl vyvíjen a testován na operačním systému Ubuntu (Debian) s doinstalováním následujících nástrojů.

```
apt-get install python3 python3-pip
pip3 install netifaces scapy
```

### B.1 Server modul

Serverová část se skládá ze sady python skriptů a přiložený kód je přizpůsoben verzi 3.6. Je tedy nutné disponovat interpretem dané verze. Je možné že bude před spuštěním vyžadována instalace některých doplňkových modulů.

Protože server potřebuje ke svému provozu také MySQL/MariaDB databázi, je nutný první krok nastavení připojení v souboru `server/server-config.json`. Implicitně je při spuštění použito nastavení z pole `net_mysql`. Přednastavena je databáze poskytnutá zadavatelskou firmou. Pro použití lokálního databázového serveru je možné použít nastavení v poli `local_mysql`, které je pak načteno při použití přepínače `--local`.

Po nastavení by mělo být možné spustit centrální instanci z kořenové složky pomocí následujícího příkazu s připojením k databázi v poli `net_mysql`. (Práva roota jsou nutná kvůli NTP serveru.)

```
sudo python3.6 locator-server.py
```

Pro použití lokální databáze z pole `local_mysql` vypadá příkaz následovně.

```
sudo python3.6 locator-server.py --local
```

Jakmile je server spuštěn, vypisuje logovací výstupy do souboru `dlocal-server.log`. Je zde možné sledovat jaké jsou spuštěny subprocessy, nebo jaká data server přijímá a odesílá.

Jedním z podprocesů je také web server sloužící jako grafické rozhraní. Přistoupit na něj je možné v prohlížeči na portu 8080.

```
http://localhost:8080/index.html
```

## B.2 Senzor modul

Reálný senzor využívá speciální Wi-Fi čipset, který dokáže pracovat ve dvou typech provozu zároveň (Monitor a Master/Access Point). Toto ne každá bezdrátová karta umožňuje a proto je zprovoznění senzoru poněkud složitější. Složka `install/` obsahuje instalační skripty a konfigurační soubory pro přizpůsobení počítače s Linux OS do role senzoru.

Nicméně pro potřeby testování komunikace a přenosu dat v systému je obslužný software senzoru vybaven možností fungovat v takzvaném „fake“ režimu. V tomto režimu nejsou nutná žádná speciální nastavení, nebo hardware a na server budou odesílána podvržená data. Stejně jako centrála je i SW senzoru napsán v jazyce python, a tak je příkaz spuštění podobný (opět z kořenové složky projektu). V tomto režimu je možné spustit senzor na stejném stroji, kde běží i server. Pro spuštění ve standardním režimu, tedy včetně vytvoření přístupového bodu a monitorování bezdrátového provozu, stačí spustit senzor bez přepínače.

```
sudo python3.6 locator-sensor.py --fake
```

Je umožněn i provoz senzoru (zejména pro testovací účely) takový, že odesílá na server i hodnoty každého zaznamenaného paketu. Takového režimu se dá dosáhnout přepínačem `--packets`. Oba zmíněné přepínače lze kombinovat.

```
sudo python3.6 locator-sensor.py --packets
```

Stejně jako server, i senzor loguje svůj výstup do souboru `dlocal-sensor.log`, jež je umístěn v kořenové složce.

**POZOR!** Pokud dojde ke změně nastavení parametrů sítě (například přes webové rozhraní), běžící software senzoru tuto změnu zaznamená a pokusí se restartovat stroj na kterém je spuštěn (příkaz `reboot`)! Mohlo by tak dojít k nežádoucím ztrátám neuložených dat.

## Příloha C

# Příklad API příkazů

Součástí systému je programové rozhraní pro přizpůsobení na míru uživateli. Uvádím zde příklady takového způsobu ovládání.

```
1 # Prikaz upravi vlastnosti bezdratove lokalizacni site
2 # Po ulozeni a rozeslani budou vsechny senzory
3 # restartovany pro aplikaci noveho nastaveni
4 #
5 # V tomto pripade ale chybi uvest jeste sitove porty
6 # proto je odpoved negativni
7 {
8     "command": "change-settings",
9     "network":{
10         "ssid": "DactylLocal",
11         "channel": 8
12     },
13     "others":{
14         "sleepCycle": 5
15     }
16 }
17 # -> Odpoved:
18 {
19     "command": "response",
20     "state": "nok",
21     "data": "Neni zadana spravna struktura nastaveni"
22 }
```

Výpis C.1: Příkaz pro uložení nového nastavení sítě.

```

1 # Aby bylo zarizeni lokalizovano, musi znat system jeho MAC adresu
2 # Senzory filtruji provoz na pridane MAC adresy a ukladaji jen ty sledovane
3 # Pro lepsi identifikaci zarizeni jej muzeme pojmenovat
4 {
5     "command": "add-device",
6     "name": "Honza pracovni telefon",
7     "mac": "54:27:58:XX:XX:XX"
8 }
9 # -> Odpoved:
10 {
11     "command": "response",
12     "state": "ok",
13     "data": "Zarizeni pridano!"
14 }

```

Výpis C.2: Příklad příkazu pro přidání pasivně lokalizovaného zařízení.

```

1 # Senzor musi byt prirazeno lokaci, aby byla jeho data zobrazovana
2 # Lokace muze mit prirazeno vice senzorů
3 {
4     "command": "add-sensor-to-location",
5     "sensor-name": "V relax mistnosti",
6     "location-name": "Relax mistnost"
7 }
8 # -> Odpoved:
9 {
10     "command": "response",
11     "state": "ok",
12     "data": "Nove propojeni ulozeno"
13 }

```

Výpis C.3: Příkaz pro přisouzení senzoru lokaci.

```

1 # Vypis lokalizacnich dat. Jde o hlavni vysledky systemu
2 # Zarizeni je identifikovano MAC adresou
3 # Jsou pridane parametry pro moznost ‘strankovani’
4 {
5   "command": "get-device-location",
6   "mac": "54:27:58:XX:XX:XX",
7   "limit": "20",
8   "page": "3"
9 }
10 # -> Odpoved:
11 {
12   "command": "response",
13   "state": "ok",
14   "data": [
15     {
16       "location": "Relax mistnost",
17       "recorded_at": "2020-07-01 12:02:00"
18     },
19     {
20       "location": "Relax mistnost",
21       "recorded_at": "2020-07-01 12:06:00"
22     },
23     {
24       "location": "Zasedacka A",
25       "recorded_at": "2020-07-01 12:08:00"
26     }
27   ]
28 }

```

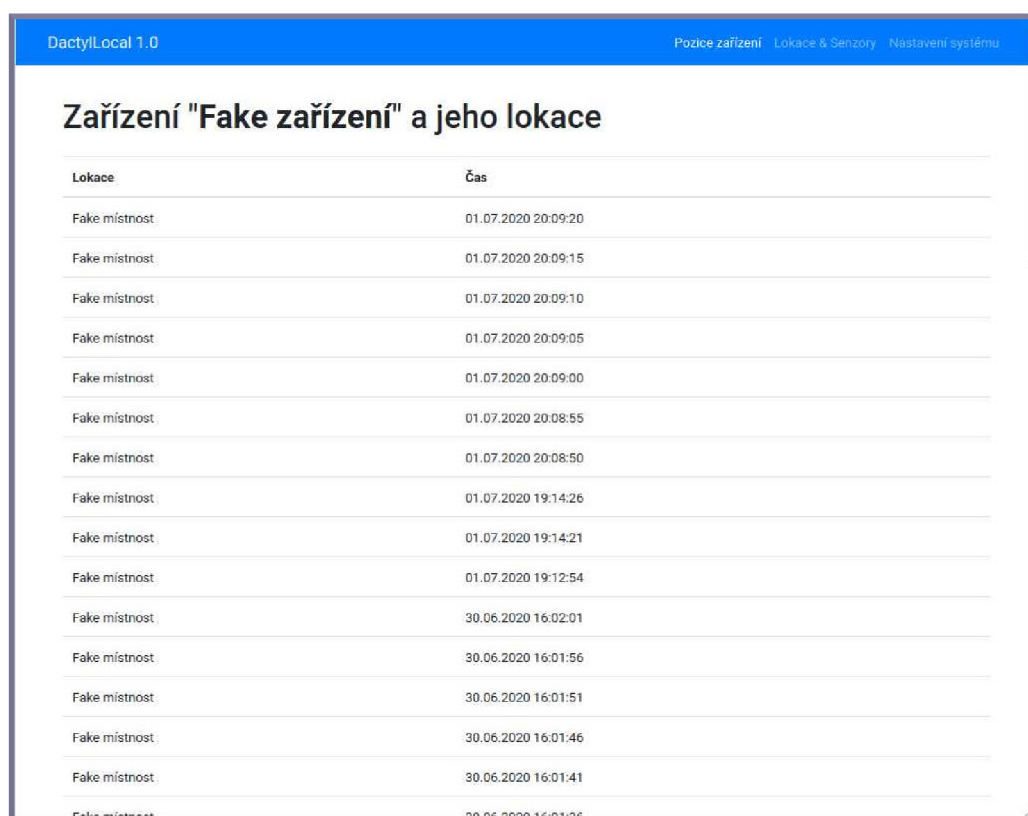
Výpis C.4: Příklad získání lokací, ve kterých se zařízení nacházelo.

# Příloha D

## Ukázka webové aplikace

Jako grafické rozhraní pro ovládání uživatelem je součástí systému webový server. Ten nabízí základní přehledy, i akce pro nutná nastavení systémových částí.

### D.1 Lokace zařízení v čase

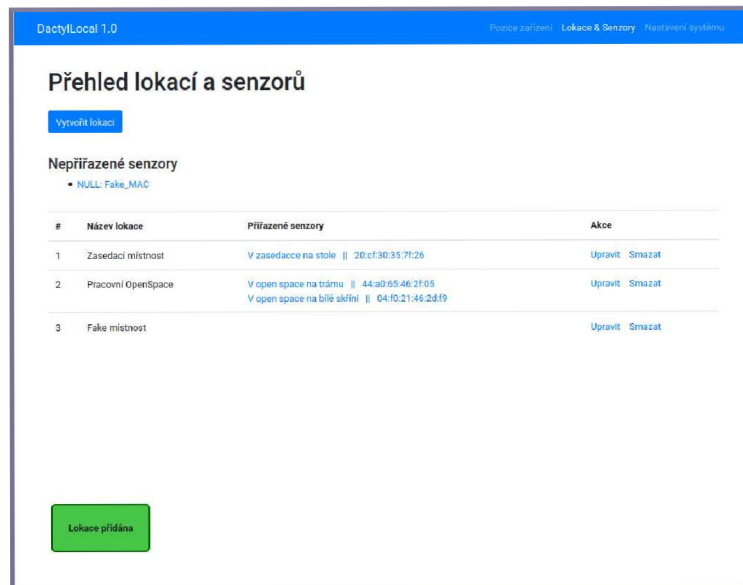


The screenshot shows a web application interface for 'DactylLocal 1.0'. The top navigation bar includes 'Pozice zařízení', 'Lokace & Senzory', and 'Nastavení systému'. The main content area is titled 'Zařízení "Fake zařízení" a jeho lokace' and contains a table with two columns: 'Lokace' and 'Čas'. The table lists 16 entries, each with the location 'Fake místnost' and a timestamp. The timestamps are arranged in two groups: the first group shows times from 20:09:20 down to 19:12:54 on 01.07.2020, and the second group shows times from 16:02:01 down to 16:01:26 on 30.06.2020.

| Lokace        | Čas                 |
|---------------|---------------------|
| Fake místnost | 01.07.2020 20:09:20 |
| Fake místnost | 01.07.2020 20:09:15 |
| Fake místnost | 01.07.2020 20:09:10 |
| Fake místnost | 01.07.2020 20:09:05 |
| Fake místnost | 01.07.2020 20:09:00 |
| Fake místnost | 01.07.2020 20:08:55 |
| Fake místnost | 01.07.2020 20:08:50 |
| Fake místnost | 01.07.2020 19:14:26 |
| Fake místnost | 01.07.2020 19:14:21 |
| Fake místnost | 01.07.2020 19:12:54 |
| Fake místnost | 30.06.2020 16:02:01 |
| Fake místnost | 30.06.2020 16:01:56 |
| Fake místnost | 30.06.2020 16:01:51 |
| Fake místnost | 30.06.2020 16:01:46 |
| Fake místnost | 30.06.2020 16:01:41 |
| Fake místnost | 30.06.2020 16:01:36 |

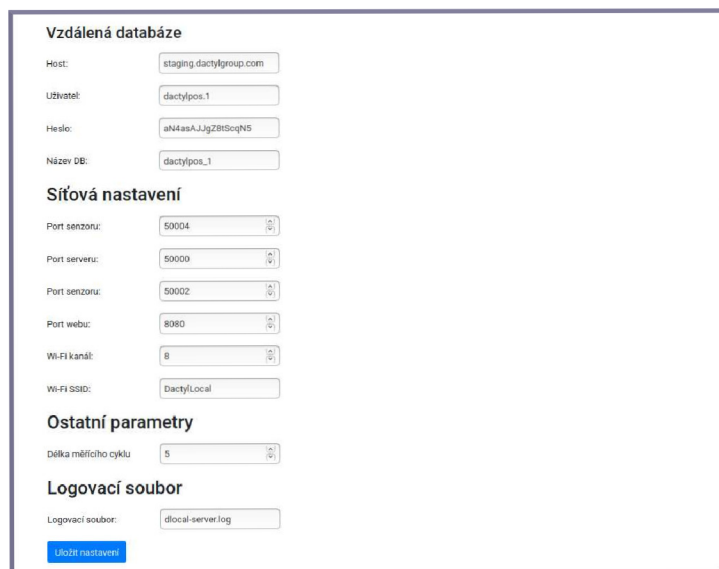
Obrázek D.1: Obrazovka zobrazující, ve kterých lokacích se nacházelo zařízení v minulosti.

## D.2 Seznam lokací



Obrázek D.2: Obrazovka se seznamem lokací pro jejich přehled, ale i úpravu.

## D.3 Nastavení systému



Obrázek D.3: Obrazovka nastavení síťových parametrů systému.

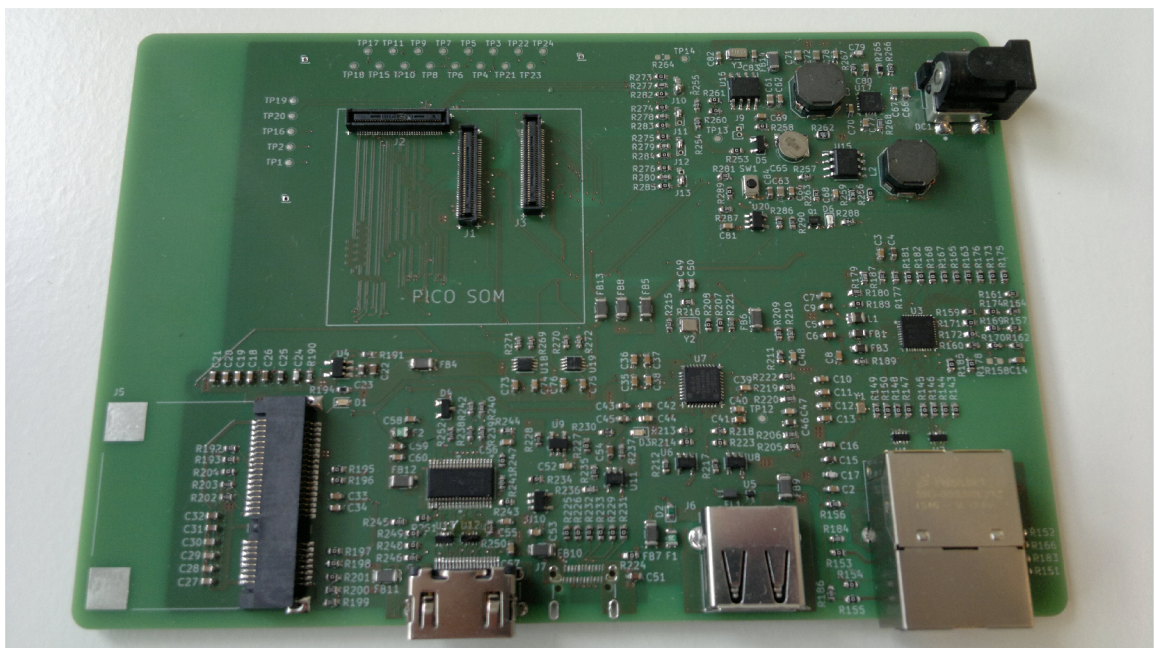
## Příloha E

# Lokalizační modul

Senzor je složen z několika dílčích částí. V této kapitole je uvedeno schéma zapojení, navržený model desky tištěného spoje a nechybí ani snímek vytvořeného prototypu.

### E.1 Foto vytvořeného prototypu

S ohledem na cenu, byl vytvořen jediný prototyp senzoru.

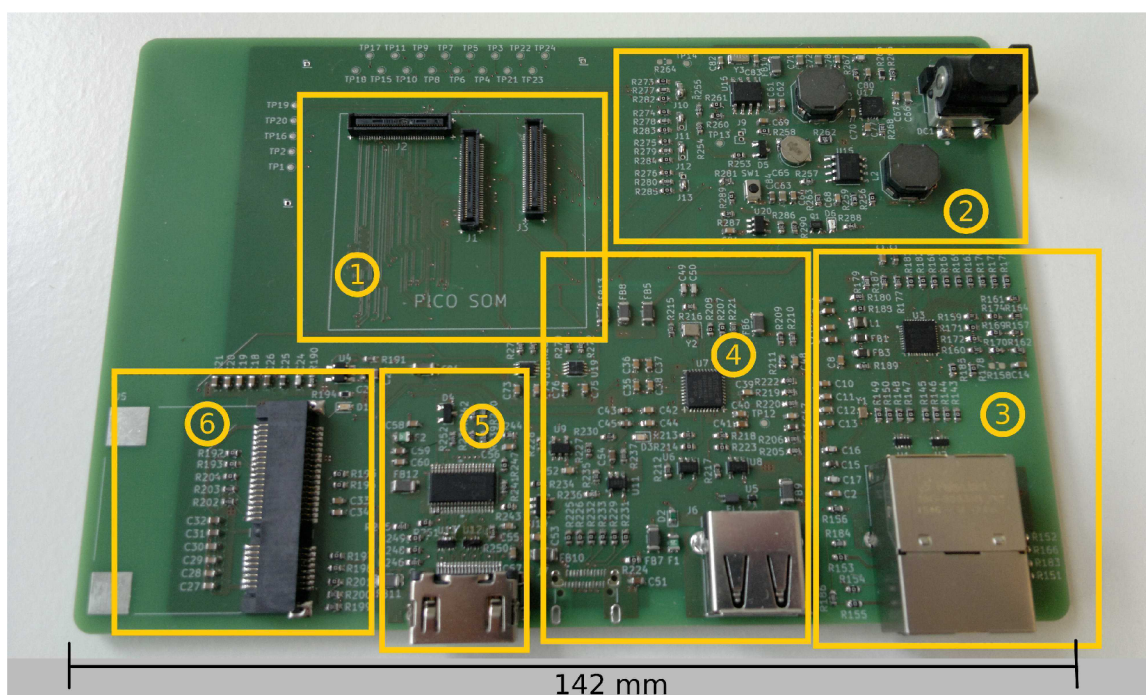


Obrázek E.1: Samotná deska tištěného spoje.



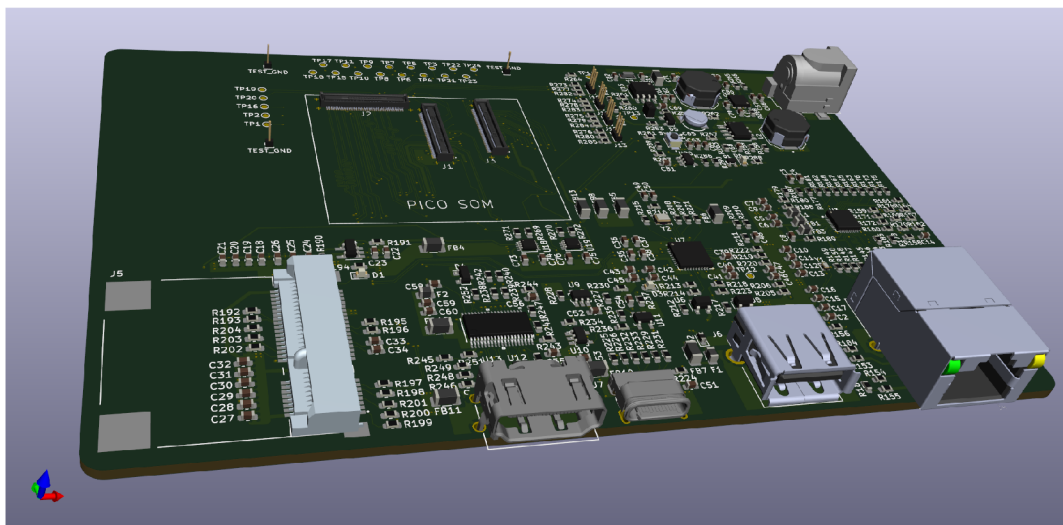


Obrázek E.2: Prototyp senzoru osazený CPU modulem i Wi-Fi.



Obrázek E.3: Prototyp senzoru s vyznačenými hlavními částmi. 1. HIROSE CPU, 2. Napájení, 3. Ethernet, 4. USB, 5. HDMI, 6. Mini PCIe

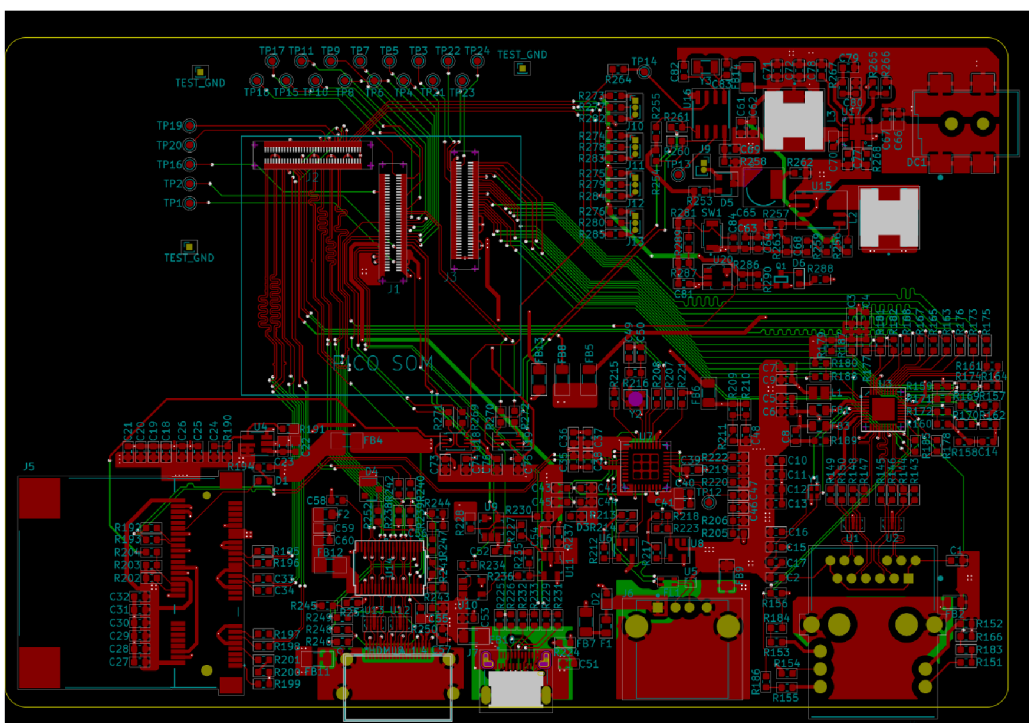
## E.2 Modely desky tištěného spoje



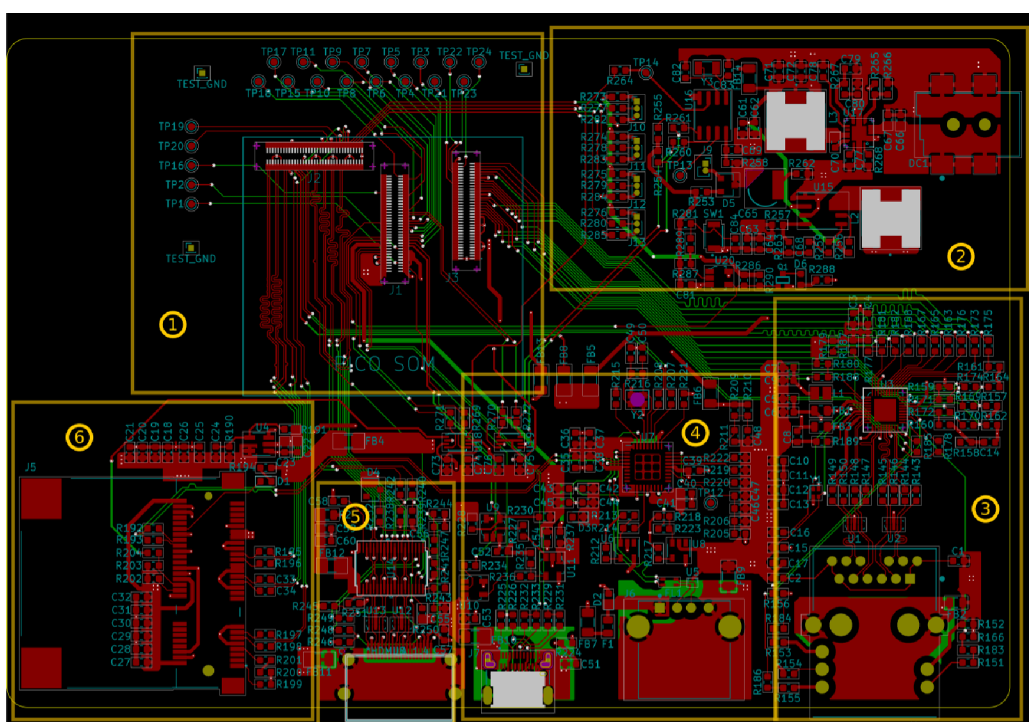
Obrázek E.4: Vygenerovaný 3D model programem KiCAD.



Obrázek E.5: Příklad alternativního senzoru, tedy zástupného zařízení. Zde notebook ASUS K50IP.



Obrázek E.6: Tištěný spoj senzoru. Součástky jsou orientovány na horní vrstvu, pro snadné strojové osazení. Vytvořeno v programu KiCAD.



Obrázek E.7: Tištěný spoj senzoru s vyznačenými hlavními částmi. 1. HIROSE CPU, 2. Napájení, 3. Ethernet, 4. USB, 5. HDMI, 6. Mini PCIe

## E.3 Tabulka použitých součástek

Tabulka E.1: Seznam použitých součástek

| Značka       | Počet | Hodnota         | Pouzdro   | Dodavatel |
|--------------|-------|-----------------|-----------|-----------|
| C1, C47, C68 | 3     | 1000 pF         | 0603      | Mouser    |
| C4–C45       | 7     | 1 $\mu$ F       | 0603      | Mouser    |
| C6–C54       | 8     | 10 $\mu$ F      | 0603      | Mouser    |
| C16, C17     | 2     | 22 pF           | 0603      | Mouser    |
| C21, C41     | 2     | 4.7 $\mu$ F     | 0603      | Mouser    |
| C24          | 1     | 100 pF          | 0603      | Mouser    |
| C49, C50     | 2     | 33 pF           | 0603      | Mouser    |
| C61–C72      | 5     | 22 $\mu$ F      | 0603      | DigiKey   |
| C65          | 1     | 80 mF           | 2 SMD     | DigiKey   |
| C78          | 1     | 0.01 $\mu$ F    | 0603      | Mouser    |
| C79          | 1     | 180 pF          | 0603      | Mouser    |
| C80          | 1     | 6800 pF         | 0603      | Mouser    |
| C2–C84       | 49    | 0.1 $\mu$ F     | 0603      | Mouser    |
| D1, D3       | 2     | 571 nm (G)      | 0603      | Mouser    |
| D2           | 1     | 150 V           | 0603      | Mouser    |
| D4, D5       | 2     | Schottky        | SOT-23    | Mouser    |
| D6           | 1     | 631 nm (R)      | 0603      | Mouser    |
| DC1          | 1     | DC-IN           | 4-SMD     | Mouser    |
| F1, F2       | 2     | 6 V, 1 A (Fuse) | 0805      | Mouser    |
| FB1, FB3     | 2     | 150 $\Omega$    | 0603      | Mouser    |
| FB2          | 1     | 220 $\Omega$    | 0603      | Mouser    |
| FB4–FB14     | 12    | 31 $\Omega$     | 0603      | Mouser    |
| FL1          | 1     | 90 $\Omega$     | 4 SMD     | Mouser    |
| J1, J2, J3   | 3     | DF40HC          | 70 SMD    | Mouser    |
| J4           | 1     | Ethernet        | Thr. Hole | Mouser    |
| J5           | 1     | MINI PCIE-52    | 52 SMD    | Mouser    |
| J6           | 1     | USB_A           | Thr. Hole | Mouser    |
| J7           | 1     | USB_C           | Thr. Hole | Mouser    |
| J8           | 1     | HDMI_A_1.4      | Thr. Hole | Mouser    |
| L1, L2, L3   | 3     | 4.7 $\mu$ H     | 2 SMD     | Mouser    |
| Q1           | 1     | BSS138W         | SOT-323   | Mouser    |
| R179         | 1     | 750 $\Omega$    | 0603      | Mouser    |
| R180–R268    | 9     | 100 k $\Omega$  | 0603      | Mouser    |
| R183–R280    | 7     | 150 $\Omega$    | 0603      | Mouser    |
| R189         | 1     | 2.37 k $\Omega$ | 0603      | Mouser    |
| R190, R262   | 2     | 1 k $\Omega$    | 0603      | Mouser    |
| R191         | 1     | 4.02 k $\Omega$ | 0603      | DigiKey   |
| R194         | 1     | 330 $\Omega$    | 0603      | Mouser    |
| R202–R230    | 4     | 4.7 k $\Omega$  | 0603      | Mouser    |
| R215         | 1     | 12 k $\Omega$   | 0603      | Mouser    |
| R216         | 1     | 1 M $\Omega$    | 0603      | Mouser    |
| R236         | 1     | 100 $\Omega$    | 0603      | Mouser    |
| R237         | 1     | 22 $\Omega$     | 0603      | Mouser    |

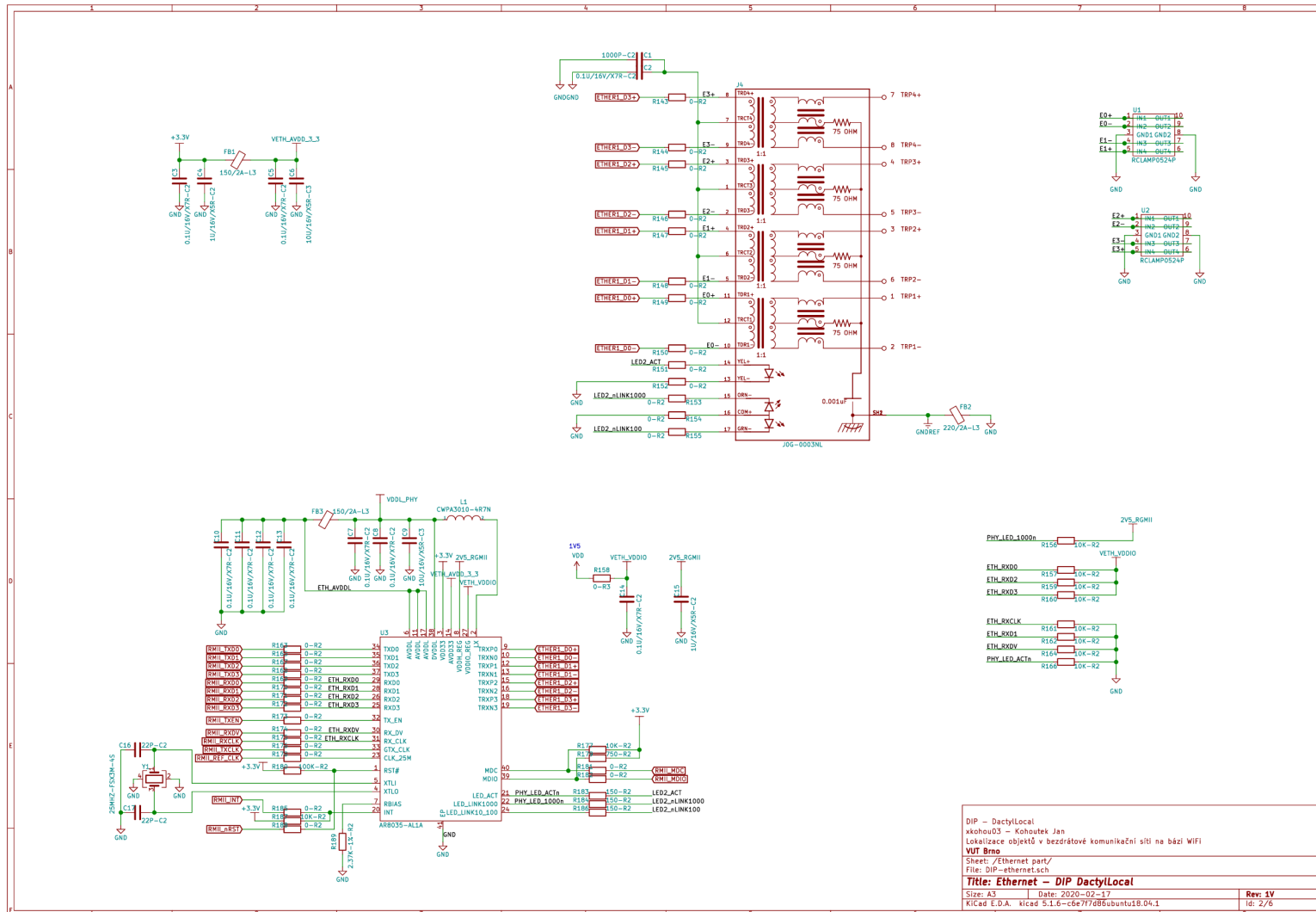
Tabulka E.1 – Pokračování z minulé stránky

| Značka        | Počet | Hodnota         | Pouzdro   | Dodavatel |
|---------------|-------|-----------------|-----------|-----------|
| R238–R255     | 6     | 2.2 kΩ          | 0603      | Mouser    |
| R240–R272     | 6     | 10 Ω            | 0603      | Mouser    |
| R241, R250    | 2     | 47 kΩ           | 0603      | Mouser    |
| R244          | 1     | 15 kΩ           | 0603      | Mouser    |
| R252          | 1     | 51 kΩ           | 0603      | Mouser    |
| R257          | 1     | 31.6 kΩ         | 0603      | Mouser    |
| R259          | 1     | 20 kΩ           | 0603      | Mouser    |
| R265          | 1     | 176 kΩ          | 0603      | Mouser    |
| R266          | 1     | 24 kΩ           | 0603      | Mouser    |
| R267          | 1     | 4,3 kΩ          | 0603      | Mouser    |
| R288          | 1     | 576 Ω           | 0603      | Mouser    |
| R156–R290     | 24    | 10 kΩ           | 0603      | Mouser    |
| R143–R224     | 72    | 0 Ω             | 0603      | Mouser    |
| SW1           | 1     | SW-MPTCFGP2-V   | 2 SMD     | Mouser    |
| U3            | 1     | AR8035-AL1B     | 40-QFN EP | Arrow     |
| U4            | 1     | AME8850AEEVADJZ | SOT-25    | Mouser    |
| U5            | 1     | TPD2EUSB30      | SOT-3     | Mouser    |
| U6, U8        | 2     | TPS2065DBVR     | SOT-3-5   | Mouser    |
| U7            | 1     | USB2512B_Bi     | 36-VFQFN  | Mouser    |
| U9, U10, U20  | 3     | SN74LVC1G07DBVR | SOT-23-5  | Mouser    |
| U11, U18, U19 | 3     | TXS0102DCUR     | SOT-23-5  | Mouser    |
| U14           | 1     | CM2020-00TR     | 38-TFSOP  | Mouser    |
| U15           | 1     | AOZ1021AI       | 8-SOIC    | DigiKey   |
| U16           | 1     | RTC             | 8-SOIC    | Mouser    |
| U17           | 1     | RT7298BHGQW     | 8-SOIC    | DigiKey   |
| U1–U13        | 4     | RCLAMP0524P     | 10-UFDNFN | Mouser    |
| Y1            | 1     | 25 MHz          | 4-SMD     | DigiKey   |
| Y2            | 1     | 24 MHz          | 4-SMD     | DigiKey   |
| Y3            | 1     | 32,768 kHz      | 2-SMD     | Mouser    |

## E.4 Schéma obvodu přístupového bodu

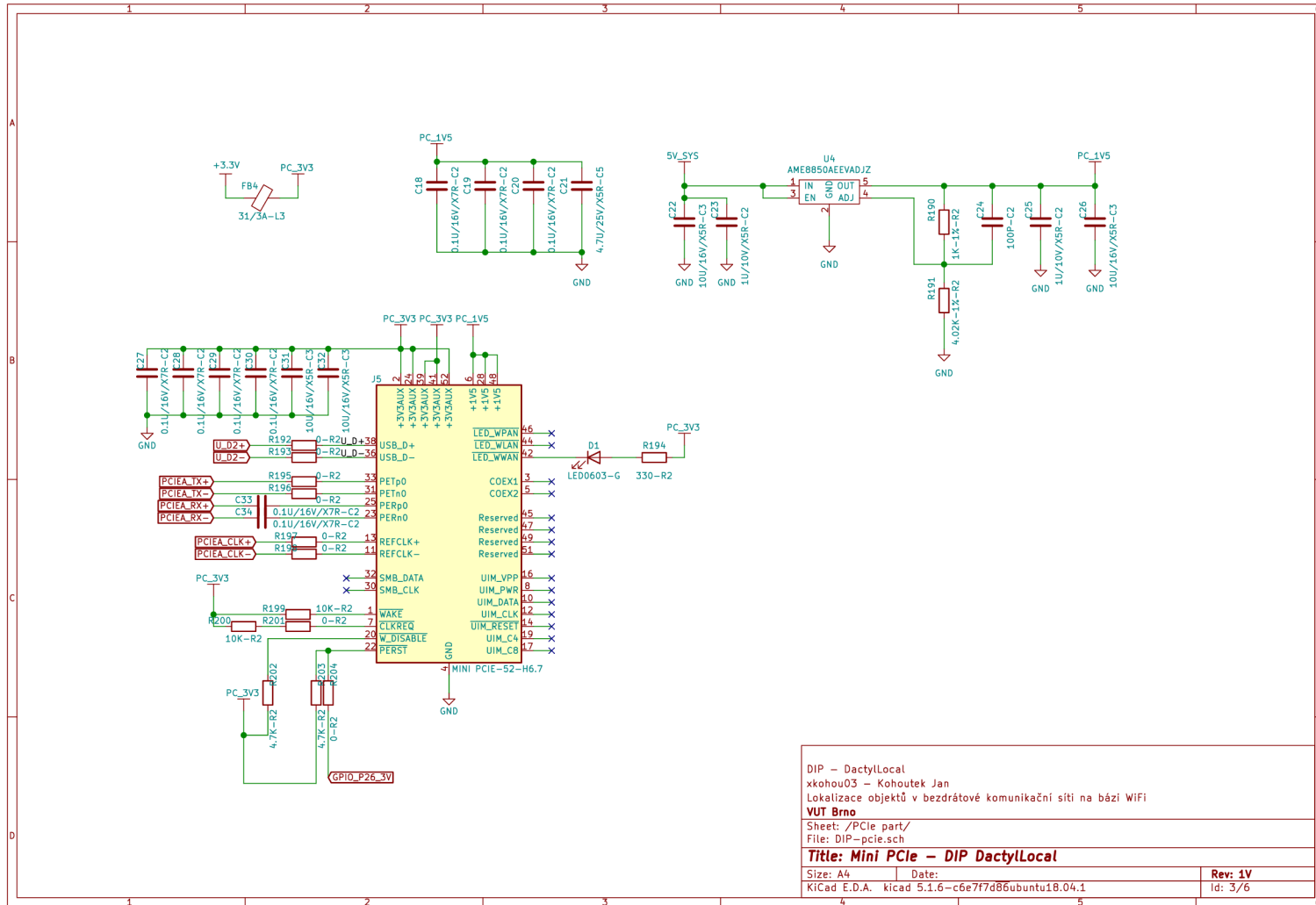
Následující stránky obsahují ukázky schémat zapojení jednotlivých částí obvodu senzoru. Schéma bylo vypracováno v programu KiCAD.



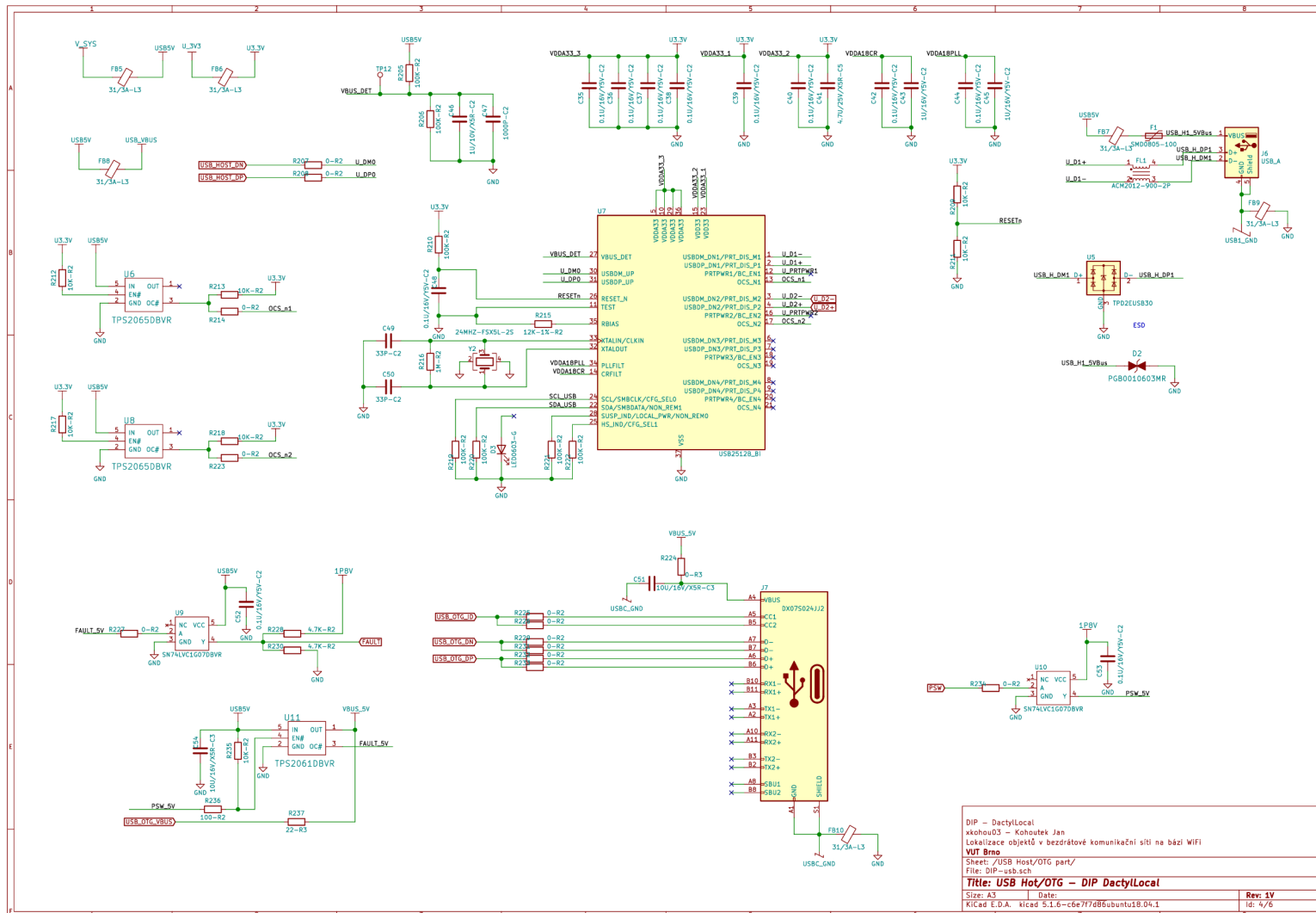


Obrázek E.9: Schéma kabelového síťového připojení. RJ45 konektor a řadič AR8035-AL1A

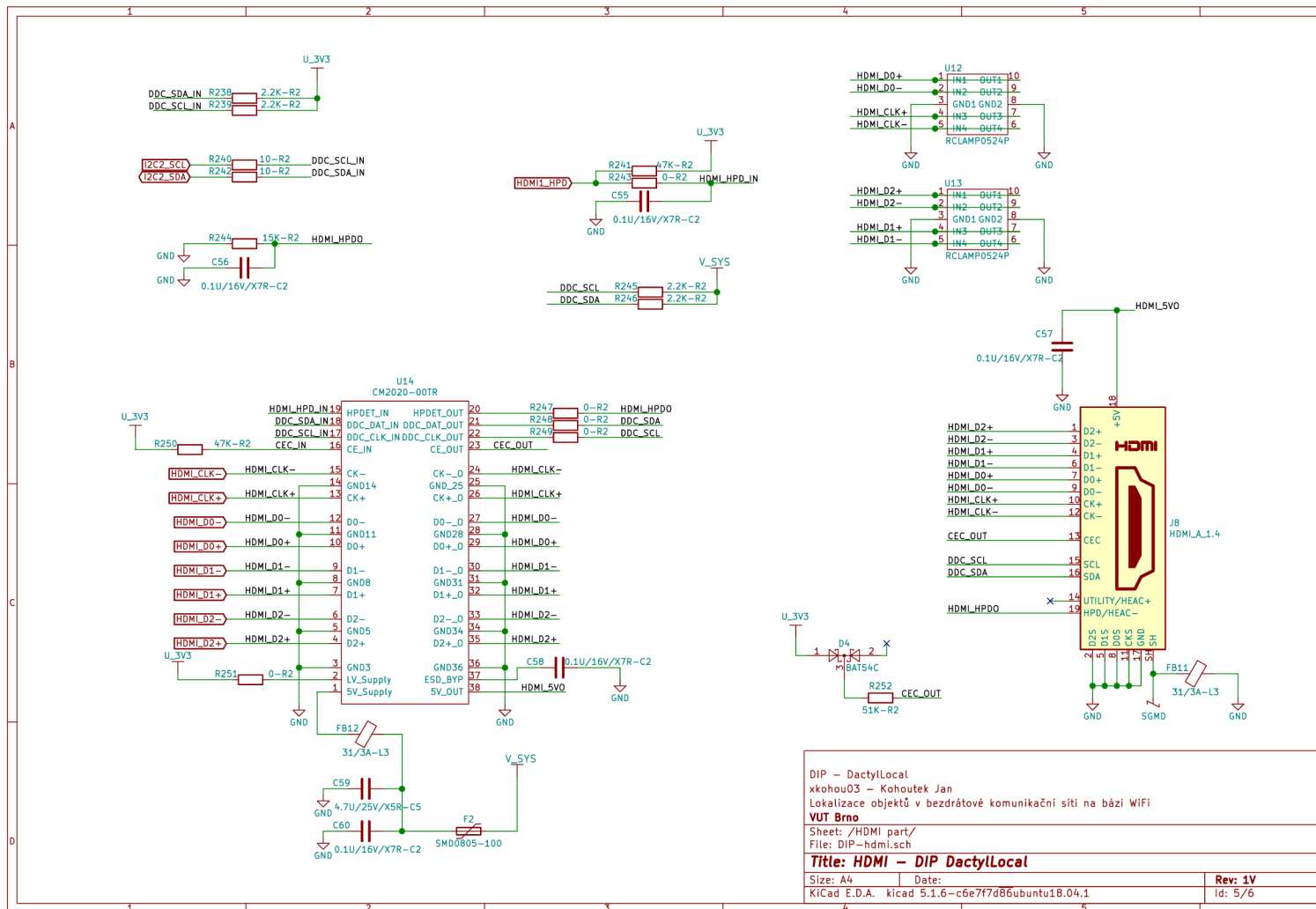




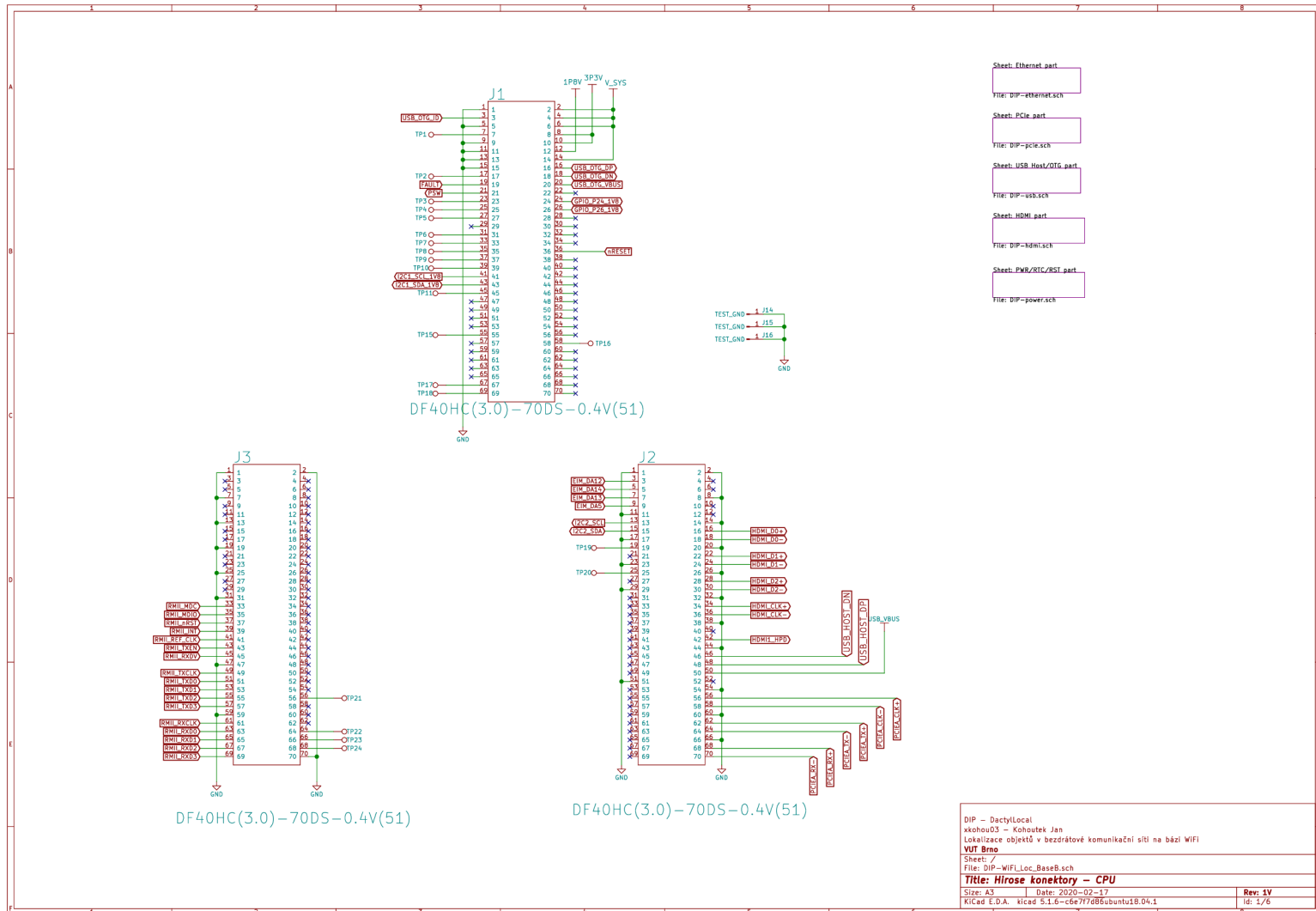
Obrázek E.10: Schéma zapojení PCIe konektoru pro připojení bezdrátového Wi-Fi modulu.



Obrázek E.11: Schéma zapojení konektorů USB. USB-C v režimu OTG a USB typu A.



Obrázek E.12: Schéma zapojení HDMI konektoru pro grafický výstup.



Obrázek E.13: Procesor, včetně paměti, je připojen trojicí Hirose konektorů.