

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA STAVEBNÍ

OBOR GEODÉZIE A KARTOGRAFIE



## DIPLOMOVÁ PRÁCE

Vývoj nízkonákladového GNSS zařízení

Development of low-cost GNSS device

Autor: Bc. Štěpán Hodík  
Vedoucí práce: Ing. Zdeněk Vyskočil, Ph.D.  
Praha, 2019



## ZADÁNÍ DIPLOMOVÉ PRÁCE

### I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: <u>Hodík</u>	Jméno: <u>Štěpán</u>	Osobní číslo: <u>439259</u>
Zadávací katedra: <u>K155 – Katedra geomatiky</u>		
Studijní program: <u>Geodézie a kartografie</u>		
Studijní obor: <u>Geodézie a kartografie</u>		

### II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce: <u>Vývoj nízkonákladového GNSS zařízení</u>	
Název diplomové práce anglicky: <u>Development of low-cost GNSS device</u>	
Pokyny pro vypracování: Studium technické dokumentace GNSS UBLOX Neo Návrh struktury zařízení GNSS přijímače Sestavení komponentů Testování	
Seznam doporučené literatury:	
Jméno vedoucího diplomové práce: <u>Ing. Zdeněk Vyskočil, Ph.D.</u>	
Datum zadání diplomové práce: <u>22. 2. 2019</u>	Termín odevzdání diplomové práce: <u>19.5.2019</u> <i>Údaj uveďte v souladu s datem v časovém plánu příslušného ak. roku</i>
_____	_____
Podpis vedoucího práce	Podpis vedoucího katedry

### III. PŘEVZETÍ ZADÁNÍ

*Beru na vědomí, že jsem povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je nutné uvést v diplomové práci a při citování postupovat v souladu s metodickou příručkou ČVUT „Jak psát vysokoškolské závěrečné práce“ a metodickým pokynem ČVUT „O dodržování etických principů při přípravě vysokoškolských závěrečných prací“.*

_____	_____
Datum převzetí zadání	Podpis studenta(ky)

## **ABSTRAKT**

Tato diplomová práce se zabývá vývojem zařízení, které umožňuje určení své polohy pomocí signálu vysílaných globálními navigačními systémy. Zařízení bylo složeno z mikrokontroléru Arduino Due, GNSS desky s čipem od společnosti u-Blox a bezdrátového bluetooth modulu HC-05.

K tomuto zařízení byla ve vývojovém frameworku Apache Cordova vytvořena obslužná aplikace pro platformu android, ve které jsou spravována data zasílaná vytvořeným GNSS přijímačem. Tato aplikace také umožňuje funkcionalitu NTRIP Klienta, díky které je možný příjem korekčních dat ze sítě permanentních stanic a následné měření v režimu RTK. V rámci práce jsou popsány komunikační protokoly využívané v oblasti GNSS a dále jsou zde stručně vysvětleny metody měření a princip satelitní navigace.

## **KLÍČOVÁ SLOVA**

GNSS, RTK, NTRIP, u-blox NEO-M8P, Apache Cordova, Android

## **ABSTRACT**

This master thesis deals with the development of the device, which enables to determine his position by using of the radio signal transmitted by global navigation systems. The device consists of Arduino Due microcontroller, GNSS board with chip from u-blox and wireless bluetooth module HC-05.

An android application was created for this device in the Apache Cordova development framework, which manages the data sent by the generated GNSS receiver. This application also enables NTRIP Client functionality, which allow the reception of correction data from the network of permanent stations and subsequent measurement in the RTK mode. The thesis describes communication protocols used in the GNSS area and also explains the measurement methods and the principle of satellite navigation.

## **KEYWORDS**

GNSS, RTK, NTRIP, u-blox NEO-M8P, Apache Cordova, Android

## **Prohlášení**

Já, Štěpán Hodík, prohlašuji, že jsem autor této diplomové práce, kterou jsem napsal pod vedením Ing. Zdeňka Vyskočila, Ph.D., za použití zdrojů uvedených v seznamu.

Souhlasím s dalším využitím mé práce i programu na ČVUT Fakultě stavební, v souladu se zákonem č.121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) v platném znění.

V Praze dne .....

.....

Bc. Štěpán Hodík



## **Poděkování**

*Na tomto místě bych rád poděkoval své rodině, která mě nejenom při psaní této práce, ale po celou dobu mého studia velmi podporovala. Hlavní poděkování patří mému vedoucímu Ing. Zdeňku Vyskočilovi, Ph.D., jenž svým odborným vedením, vstřícným přístupem a podnětnými návrhy přispěl k dokončení této práce největším dílem. V neposlední řadě bych chtěl poděkovat své přítelkyni za podporu a trpělivost, kterou projevila při psaní této práce.*

*Bc. Štěpán Hodík*

# Obsah

Úvod . . . . .	8
<b>I Satelitní navigace</b>	<b>9</b>
1 Globální navigační satelitní systémy (GNSS) . . . . .	9
1.1 Struktura GNSS systémů . . . . .	10
2 Princip určení polohy . . . . .	11
2.1 Kódové měření . . . . .	11
2.2 Fázové měření . . . . .	13
3 Signály GPS družic . . . . .	13
4 Faktory a chyby ovlivňující přesnost určení polohy . . . . .	15
4.1 Umělé ovlivňování kvality signálu . . . . .	15
4.2 Chyby související s družicemi . . . . .	15
4.3 Chyby související s přijímačem . . . . .	17
4.4 Chyby způsobené průchodem signálu prostředím . . . . .	17
4.5 Chyby související s umístěním přijímačem . . . . .	19
5 Metody měření polohy pomocí GNSS . . . . .	19
5.1 Statická metoda . . . . .	20
5.2 Rychlá statická metoda . . . . .	20
5.3 Metoda Stop and Go . . . . .	21
5.4 RTK (Real Time Kinematic) . . . . .	21
5.5 DGNSS . . . . .	21
6 Komunikační protokoly . . . . .	22
6.1 NMEA . . . . .	22
6.1.1 RMC . . . . .	23
6.1.2 GSA . . . . .	24
6.1.3 GGA . . . . .	24
6.1.4 GSV . . . . .	25
6.1.5 GST . . . . .	26
6.2 RTCM . . . . .	27
6.3 NTRIP protokol . . . . .	27
6.3.1 Komunikace NTRIP Klienta s NTRIP Casterem . . . . .	28
6.3.2 Zdrojová tabulka . . . . .	30
6.4 RINEX . . . . .	31
<b>II Vývoj aplikace</b>	<b>32</b>
7 Přístupy vývoje mobilních aplikací . . . . .	32
8 Apache Cordova . . . . .	34
8.1 Instalace nezbytných programů . . . . .	35
8.2 Založení projektu . . . . .	36
8.3 Přidání platformy a pluginů . . . . .	38
8.4 Spuštění aplikace na zařízení . . . . .	38
8.5 Programování aplikace . . . . .	38
8.6 Ladění aplikace . . . . .	39
9 Aplikace GNSS Kontrolér . . . . .	40
9.1 Ukládání informací . . . . .	40

9.2	Struktura plochy aplikace . . . . .	42
9.3	Okno - Domovská obrazovka . . . . .	43
9.4	Okno - Nastavení . . . . .	44
9.5	Okno - Měření . . . . .	45
9.6	Okno - Vytyčení . . . . .	45
9.7	Okno - Skyplot . . . . .	46
<b>III</b>	<b>Hardware a software</b>	<b>48</b>
10	Části GNNS přijímače . . . . .	48
11	Použitá zařízení a princip jejich propojení . . . . .	49
11.1	Arduino Due . . . . .	49
11.1.1	UART . . . . .	49
11.2	HC-05 . . . . .	51
11.2.1	Zapojení HC-05 . . . . .	52
11.2.2	Konfigurace modulu . . . . .	52
11.3	GNSS moduly . . . . .	53
11.3.1	Ublox NEO-M8T . . . . .	54
11.3.2	SparkFun GPS-RTK Board - NEO-M8P-2 . . . . .	55
11.3.3	Schéma zapojení . . . . .	56
11.4	GNSS anténa . . . . .	56
11.5	Zdroj napájení . . . . .	57
11.6	Kryt přijímače . . . . .	57
12	U-blox . . . . .	58
12.1	U-center . . . . .	58
12.1.1	Představení programu . . . . .	58
12.1.2	Konfigurace parametrů modulu . . . . .	59
13	Postup testování přijímače . . . . .	61
13.1	Porovnání měřených souřadnic . . . . .	62
13.2	Zhodnocení testování . . . . .	63
14	Cenová kalkulace projektu . . . . .	65
	Závěr . . . . .	66
	Reference . . . . .	69
	Seznam obrázků . . . . .	70
	Seznam tabulek . . . . .	71
	Seznam použitých zkratk . . . . .	72
	Seznam použitých programů . . . . .	73
	Seznam příloh . . . . .	73
	Elektronická příloha A . . . . .	74



## Úvod

Technologie globálních navigačních satelitních systémů (dále jen GNSS) umožňují přesné určování polohy a času téměř na jakémkoliv místě na Zemi. Pro využívání těchto technologií je však nezbytné použít GNSS přijímač, který na základě radiových signálů vysílaných z jednotlivých družic a v nich zanesených informací dokáže určit svou polohu. Přijímačů existuje celá řada, od těch méně přesných umístěných v mobilních telefonech a další nositelné elektronice až po specializované geodetické přístroje, které mohou dosahovat relativní přesnosti určení polohy v řádu několika milimetrů.

Aplikace podporující technologie GNSS jsou neustále na vzestupu a využívají se v oblastech dopravy, telekomunikací, zemědělství, civilní bezpečnosti, geodézie a mnoha dalších. Trh s těmito produkty a službami se neustále zvětšuje a očekává se, že v roce 2020 budou v provozu 3 miliardy přijímačů družicové navigace [44].

V rámci této práce je následně popsán vývoj takového zařízení, přičemž text je rozdělen do 3 částí. První z nich je teoretická a jsou v ní vyjmenovány hlavní zástupci satelitních systémů, je zde stručně popsán princip satelitní navigace, hlavní faktory ovlivňující přesnost určování polohy a také výčet geodetických metod měření těmito technologiemi. Dále je zde uvedena struktura komunikačních protokolů, které jsou v oblasti GNSS nejvíce využívané - NMEA, RTCM, NTRIP a RINEX.

V druhé části textu je uveden výčet možných způsobů vývoje mobilních aplikací, včetně shrnutí jejich výhod a omezení. Větší část tohoto oddílu pak popisuje vývoj hybridních aplikací ve vývojovém frameworku Apache Cordova. Jsou zde zmíněny všechny potřebné programy pro tvorbu aplikací v tomto prostředí a dále také popis struktury aplikace, která je demonstrována na ukázkovém projektu. Nakonec je zde představena vytvořená aplikace GNSS Kontrolér, jejímž hlavním úkolem je sběr a vizualizace GNSS dat z přijímače.

Poslední část je pak věnována celkovému popisu jednotlivých komponentů, které jsou nezbytné pro sestavení GNSS přijímače. Především popisu GNSS desky od společnosti SparkFun, a také možnostem a způsobu její konfigurace v programu u-center. V úplném závěru je uveden postup testování celého systému (přijímač + aplikace GNSS Kontrolér) a zhodnocení dosažených výsledků.

Toto téma jsem si vybral především proto, že mě zajímá tvorba projektů založených na mikrokontrolérech Arduino, s kterými jsem se poprvé setkal při psaní bakalářské práce. Dalším důvodem byla možnost tvorby mobilní aplikace, s kterou jsem doposud neměl žádnou zkušenost a tato práce mi poskytla motivaci se v tomto ohledu něčemu přiučit.



## Část I

# Satelitní navigace

Technologie satelitní navigace nás obklopuje jak v soukromé, tak komerční sféře. Pouze těžko bychom dnes hledali člověka, co je aktivně nepoužívá, či o nich dokonce nikdy neslyšel. Lidé už k orientaci a vyhledání trasy téměř nepoužívají papírové mapy a raději využijí podpory satelitní navigace ve svém chytrém mobilního telefonu, hodinkách, outdoorové navigaci a dalších zařízeních, které jim během chvíle zobrazí jejich polohu a nejkratší cestu k cíli i vzhledem k aktuálnímu stavu dopravy.

V oblasti geodézie se pak setkáváme s profesionálními přijímači, které dokáží za splnění určitých podmínek získat polohu bodu s přesností několika milimetrů. Je však důležité, nepřístupovat k těmto zařízením a technologiím pouze jako k tzv. *černým skříňkám*, které jako zázrakem poskytují informace o poloze přijímače. Proto je úvodní část tohoto textu věnována právě popisu satelitních systémů a principu jejich fungování.

## 1 Globální navigační satelitní systémy (GNSS)

GNSS je souhrnný název pro technologie umožňující pomocí příjmu satelitních signálů určovat svou pozici, rychlost, kurz a také přesný čas kdekoliv na Zemi. Moderní přijímače dokáží zpracovávat vysílané signály z více satelitních systémů, což vede ke zvýšení přesnosti a dostupnosti dané služby.

Současné GNSS systémy :

- **GPS-NAVSTAR** -  
je globální navigační systém provozovaný Ministerstvem obrany Spojených států amerických. Jeho plného operačního stavu bylo dosaženo 17. července 1995. V současné době se jeho kosmický segment skládá z 31 provozuschopných družic obíhajících po téměř kruhových orbitálních drahách ve výšce přibližně 20 200 km nad povrchem Země [19].
- **GLONASS** -  
tento satelitní systém spravuje ruské ministerstvo obrany. Plné operační způsobilosti dosáhl až koncem roku 2012. V současné době je na orbitálních drahách celkem 26 družic tohoto systému, přičemž 24 satelitů je provozních, 1 záložní a 1 satelit je v testovacím režimu. Hlavní rozdíl mezi systémy GPS a GLONASS je přístup k rozlišení signálu vysílaného jednotlivými satelity. Satelity systému GPS vysílají informace na stejných frekvencích a k jejich rozlišení se používají tzv. PRN<sup>1</sup> kódy, díky kterým je přijímač schopen rozpoznat signály jednotlivých družic. Oproti tomu systém GLONASS používá frekvenční rozlišení signálů, kdy každá družice vysílá své přidělené frekvenci [18].
- **Galileo** -  
globální navigační systém, který je provozován Evropskou unií ve spolupráci s Evropskou kosmickou agenturou (ESA). Na rozdíl od systémů GPS a GLONASS se však nejedná o armádní systém, nýbrž je primárně navržen jako projekt řízený a

---

<sup>1</sup>Pseudorandom Noise Code, pseudonáhodný fázový šum (kód).



spravovaný civilní správou. Systém je stále ve vývoji a při plném provozu by se měl kosmický segment skládat z 30 družic (27 operační + 3 záložní) obíhajících ve výšce 23 222 km po třech orbitálních rovinách se sklonem 56°. V současné době je operabilních 21 družic [17].

- **Beidou** -

někdy označovaný jako Compass, je navigační systém vyvíjený Čínskou lidovou republikou. Tento systém byl spouštěn ve dvou fázích, v první fázi bylo dosaženo pouze regionálního pokrytí a v rámci druhé fáze dojde k rozšíření služby na globální úroveň. Plného operačního stavu by se mělo dosáhnout v roce 2020 a jeho kosmický segment bude složen celkem z 35 družic, z nich se 27 bude pohybovat na střední orbitální dráze (21 528km), 5 na geostacionární dráze a 3 na geosynchronní dráze (35 787km) [4].

Všechny GNSS systémy fungují na stejných základních principech, které budou popsány v následujících kapitolách. V kapitole o satelitních signálech se zaměříme pouze na systém GPS a popis signálů ostatní satelitních systémů zde uveden nebude, jelikož jejich zmínka není pro tuto diplomovou práci nezbytná.

### 1.1 Struktura GNSS systémů

Základní struktura dosavadních globálních navigačních systémů je obdobná a skládá se z těchto 3 částí :

- **Kosmický segment** - je tvořen konstelací umělých družic Země, které obíhají po svých oběžných drahách a vysílají radiový signál směrem k Zemi. Kosmické segmenty jednotlivých navigačních satelitních systémů se liší následujícími parametry :
  - počtem orbitálních rovin,
  - počtem družic a jejich rozmístěním,
  - výškou a sklonem orbitální dráhy,
  - orientací drah vůči Zemi,
  - typem a vybavením satelitů.
- **Řídící segment** - je složen z pozemních stanic, které zajišťují provoz celého systému a podle své funkce se dále dělí na řídicí stanice, monitorovací stanice a na stanice, jichž úkolem je komunikace s družicemi. Tento segment při řízení systému zajišťuje převážně tyto činnosti :
  - monitorování signálu družic, z jejichž pomocí se určují nové efemeridy<sup>2</sup> a také výpočet atmosferických modelů,
  - synchronizaci palubního satelitního času,
  - předávání těchto informací družicím,
  - korigování drah jednotlivých satelitů,
  - údržbu družic, popřípadě rozhodnutí o jejich stažení z oběžné dráhy a vypuštění nové družice.
- **Uživatelský segment** - se skládá ze zařízení, která dokáží zachytit signály vysílané družicemi (tzv. *GNSS přijímače*) a díky nim získat informace o poloze a času. Může se jednat o mobilní telefony, turistické navigace až po speciální geodetické přístroje.

---

<sup>2</sup>Efemeridy satelitů obsahují informace potřebné pro definování polohy družice v daném čase.



## 2 Princip určení polohy

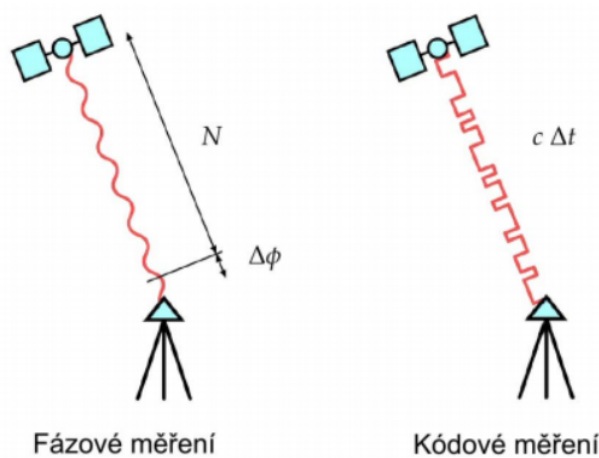
Principem satelitní navigace je tzv. „pasivní“<sup>3</sup> dálkoměrný systém, kdy satelity vysílají radiové signály, které kromě jiného obsahují zakódované informace o jejich poloze a čase vyslání. GNSS přijímače dokáží tyto signály přijmout a demodulovat. V závislosti na tom, jakým způsobem z přijatých radiových signálů určujeme vzdálenost mezi družicí a přijímačem, rozlišujeme dva základní typy měření a to **kódové** a **fázové** (viz obr. 2.1).

### Kódové měření :

- + Nízká cena přístrojového vybavení.<sup>4</sup>
- + Rychlé určení polohy pouze s jedním přijímačem.
- Nižší přesnost určení polohy (řádově metry).

### Fázové měření :

- + Daleko vyšší přesnost oproti kódovému měření (řádově centimetry až milimetry).
- Vyšší pořizovací cena přístrojového vybavení, nehledě na nutnost současného měření dvou přijímačů.
- Náročnější postupy určení polohy.
- Tato metoda je velice náchylná na výpadek příjmu satelitních signálů.



Obrázek 2.1: Princip měřené vzdálenosti - zdroj [13]

### 2.1 Kódové měření

Určení polohy přijímače s využitím kódového měření je založeno na znalosti rychlosti šíření radiového signálu a určování transitního času neboli doby, za kterou doputuje signál z družice do našeho přijímače. Součástí vysílaného signálu jsou také informace o poloze satelitu v době vyslání. Za určitých podmínek lze měřenou vzdálenost popsat následující rovnicí:

$$D_{sp} = \Delta t \cdot c \quad (2.1)$$

$$D_{sp} = \sqrt{(x_s - x_p)^2 + (y_s - y_p)^2 + (z_s - z_p)^2} \quad (2.2)$$

<sup>3</sup>GNSS přijímače signály z družic přijímají, ale nevysílají žádné informace zpět.

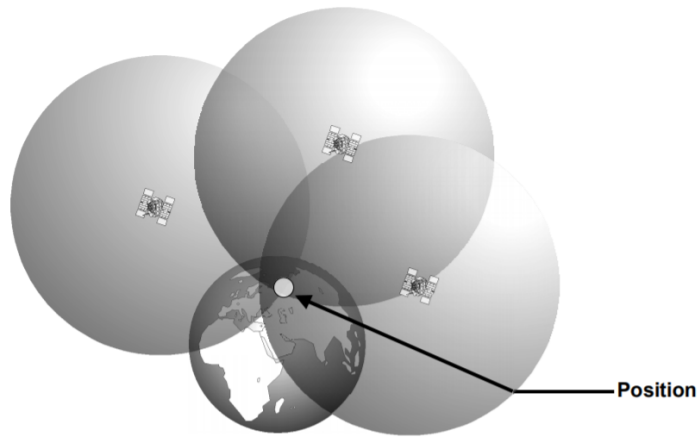
<sup>4</sup>Princip kódového měření je využit ve všech mobilních zařízeních.



kde :

$\Delta t$	- doba přenosu signálu
$c$	- rychlost světla
$D_{sp}$	- je měřená vzdálenost satelit - přijímač
$(x_s, y_s, z_s)$	- jsou známé souřadnice satelitu
$(x_p, y_p, z_p)$	- jsou určované souřadnice přijímače

Pokud známe souřadnice a vzdálenosti, alespoň ke třem satelitům, určení polohy přijímače přejde na problém vyřešení tří kvadratických rovnic o třech neznámých. Tento princip prostorového protínání z délek je graficky znázorněn na obrázku 2.2. Pozici přijímače vzhledem k satelitu popisuje množina bodů, která tvoří kulovou plochu, jejíž poloměr odpovídá měřené vzdálenosti a střed pozici satelitu. Průnikem tří takovýchto kulových ploch získáme souřadnice dvou bodů, přičemž jeden z nich leží mimo Zemi [29].



**Obrázek 2.2:** Princip určení polohy protínáním z délek - zdroj [43]

Tento postup je však pouze teoretický a lze ho uplatnit pouze za určitých předpokladů. Jedním z nich je velmi přesná časová synchronizace hodin přijímače s atomovými hodinami umístěnými na palubě satelitů, ku příkladu rozdíl synchronizace o velikosti  $1\mu s$  způsobí chybu v určení délky 300 m. Tuto synchronizaci však není možné jednoduše zajistit, představme si, že byl přijímač vypnutý a právě jsme ho zapnuli. Je tedy nezbytné do rovnice zavést ještě další neznámou a to **chybu hodin přijímače**. Z tohoto důvodu je pro určení polohy nutné přijímat signál nejméně ze čtyř satelitů a tuto opravu početně zjistit. Rovnice (2.2) pak přejde na následující tvar :

$$D_{sp} = \sqrt{(x_s^i - x_p)^2 + (y_s - y_p)^2 + (z_s - z_p)^2} + c \cdot \Delta \quad (2.3)$$

kde :

$\Delta$  - je korekce časové základny satelitů a přijímače

Dalším předpokladem je znalosti rychlost šíření radiového signálu, která vychází z rychlosti šíření světla vakuem, signál je však při průchodu atmosférou Země (zejména ionosférou a troposférou) zpomalován, což dále vede ke snížení přesnosti určované vzdálenosti. Je patrné, že měřená vzdálenost je zatížena mnoha chybami, proto bývá také označována jako *pseudovzdálenost* (pseudorange).





## 2.2 Fázové měření

Druhým způsobem určení vzdálenosti mezi satelitem a přijímačem je sledování fáze vlny nosného signálu, odtud označení fázové měření. Určovaná vzdálenost je poté definována součtem celých vlnových délek a zlomku fáze nosné vlny, jak vyplývá z rovnice (2.4).

$$D_{sp} = \Delta\lambda + N \cdot \lambda \quad (2.4)$$

kde :

- $\lambda$  - je vlnová délka nosného signálu
- $\Delta\lambda$  - je fázový posun
- $N$  - je počet celých vln

Přijímače jsou schopny velice přesně (v řádu několika milimetrů) změřit velikost fázového posunu, problém však spočívá v určení celého počtu vlnových délek. Tato nejednoznačnost v určení celkového počtu cyklů je také označován jako tzv. *počáteční fázová ambiguita* a je nutné ji zahrnout do výpočtů jako další neznámou pro každý satelit zvlášť. Pro prvotní aproximaci možného počtu  $N$  se využívají pseudovzdálenosti určené pomocí kódového měření, které však poskytnou pouze neceločíselný výsledek. Dále je tato hodnota složitými matematickými postupy zpřesňována, dokud není nalezena statisticky nejpravděpodobnější celočíselná hodnota počtu vlnových délek. Po této počáteční inicializaci je nezbytné, aby nedošlo k přerušení signálu, jelikož by přijímač nemohl zaznamenávat počet prošlých cyklů nosné vlny a došlo by k tzv. *fázovému skoku*, po kterém je nutné řešení ambiguit opakovat znovu.

Matematických metod, používaných k určení počáteční fázové ambiguita, existuje celá řada. Jejich popis a vysvětlení je uvedeno například zde [21].

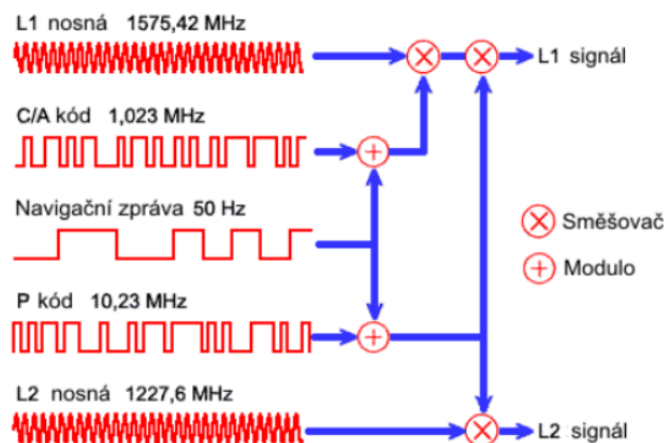
## 3 Signály GPS družic

Signály všech GPS družic tvoří kombinace nosné vlny, dálkoměrných kódů a navigační zprávy. Základní frekvence nosného signálu  $f_0 = 10,23$  MHz vychází z frekvence atomových hodin, které jsou umístěny na palubě satelitů a zajišťují dlouhodobou stabilitu signálu. Nosné vlny jsou pak vytvořeny jako násobky této základní frekvence, konkrétně :

- L1 - 1575,42 MHz
- L2 - 1227,60 MHz
- L5 - 1176,45 MHz

Nosné vlny L1 a L2 jsou v systému GPS používány již od jeho spuštění, zato L5 je nová civilní nosná vlna, která byla spuštěna v rámci modernizace GPS a vysílají ji pouze družice **bloku IIF** a novější.

Na tyto nosné vlny jsou pak modulovány tzv. PRN (*PseudoRandom Noise*) kódy, které jsou pro každou družici unikátní a přijímač díky tomu dokáže od sebe rozlišit signály jednotlivých družic. Tyto kódy následně umožňují měření vzdáleností a také obsahují informace o jednotlivých satelitech (viz obr. 3.1)[21][29].



Obrázek 3.1: Princip modulace GPS signálu - zdroj [29]

### C/A kód:

Tento pseudonáhodný kód je modulován pouze na nosné vlně L1 a není žádným způsobem šifrován, jak je patrné z jeho označení **C/A** (Clear/Access - volný přístup). Přijímače dokáží tento kód samy generovat a pomocí korelace přijatého a generovaného kódu určit dobu transitního času. C/A kód je složen z definované posloupnosti 1023 bitů a je vysílán na frekvenci 1,023 MHz, to znamená, že je opakován každou tisícinu sekundy. Přijímače jsou schopné určit místo maximální korelace s přesností přibližně 1%, výsledná přesnost určení vzdálenosti pomocí C/A kódu tedy odpovídá 3m, bereme-li v potaz pouze vliv určení transitního času. Z tohoto důvodu se tento kód využívá pouze pro časovou synchronizaci a navigaci, u které není potřebná vyšší přesnost.

### P kód:

Tento kód je označován jako přesný nebo také chráněný (Precision, Protected) a mohou ho používat pouze autorizovaní uživatelé. Oproti kódu C/A je modulován na obou nosných vlnách L1 i L2, což umožňuje eliminaci systematických jevů ionosférické a troposférické refrakce a tím dosáhnout daleko přesnějšího určení polohy. Délka P kódu odpovídá posloupnosti  $2,3547 \cdot 10^{14}$  bitů a při frekvenci vysílání 10,23 MHz dojde k opakování kódu jednou za přibližně 266,4 dne. Kód je následně rozdělen na sedmidenní sekvence, přičemž každé družici je přidělena jedna z nich a k její výměně dochází každý týden, v noci ze soboty na neděli GPS času. Tento systém kódování signálu umožňuje současné vysílání až 38 družic.

### Navigační zpráva:

Navigační zpráva je dalším typem kódu, který je vysílán družicemi na obou nosných frekvencích L1 a L2. V navigační zprávě jsou zaneseny tyto informace :

- **Efemeridy družice** - jsou parametry oběžné dráhy družice, díky kterým můžeme určit polohu družice v námi zvoleném čase.
- **Časovou značku vyslání zprávy**
- **Korekce družicového času**



- **Koeficienty ionosférického modelu** - popisují aktuální stav ionosféry. Přijímače pak tento model využívají k potlačení vlivu ionosférické refrakce.
- **Almanach** - obsahuje méně přesné informace o poloze (efemeridy) a stavu všech družic v systému. Znalost těchto parametrů urychluje vyhledání viditelných družic a následné určení polohy přijímače.
- **Stav družice** - podává uživateli informaci o „zdravotním“ stavu, tedy zda je možné přijatá data dané družice využít k výpočtu polohy.

### 4 Faktory a chyby ovlivňující přesnost určení polohy

Přesnost určení polohy přijímače závisí na spoustě faktorů. Signál, který se šíří mezi družicí a přijímačem, je podroben působením mnoha systematických jevů, které lze podle [38] obecně rozdělit na:

- umělé ovlivňování kvality signálu,
- chyby související s družicemi,
- chyby související s přijímačem,
- chyby způsobené průchodem signálu prostředím,
- chyby související s umístěním přijímače.

Informace uvedené v následujících podkapitolách byly čerpány z těchto zdrojů [2][22][38].

#### 4.1 Umělé ovlivňování kvality signálu

Navigační systém GPS byl primárně vytvářen pro vojenské účely armády USA. Aby tohoto systému nemohli využívat nepřátelé, byl vytvořen tzv. výběrový přístup (*Selective Available* - zkratka SA). Informace o údajích palubních hodin družic a efemeridy v navigační zprávě byly uměle degradovány podle utajovaného algoritmu a neautorizovaní uživatelé mohli určit svou polohu s přesností v řádu stovek metrů. Režim SA byl ukončen v květnu roku 2000 na pokyn amerického prezidenta Billa Clintona a v současnosti se již nevyužívá.

#### 4.2 Chyby související s družicemi

##### Chyba družicových hodin:

Přestože jsou v satelitech umístěny velice přesné atomové hodiny, dochází i u nich k nepatrnému zpoždování, které podle [13] může během každých 3 hodin dosáhnout až jedné biliontiny vteřiny. Tento časový nesoulad pak způsobí v měřené délce chybu odpovídající 30 cm. Hodiny jsou proto pozorovány monitorovacími stanicemi a v hlavním řídicím centru pak dochází k výpočtu korekčních parametrů odpovídajících časovým odchýlkám atomových hodin jednotlivých satelitů. Korekce satelitních hodin jsou následně vysílány satelity v navigační zprávě.

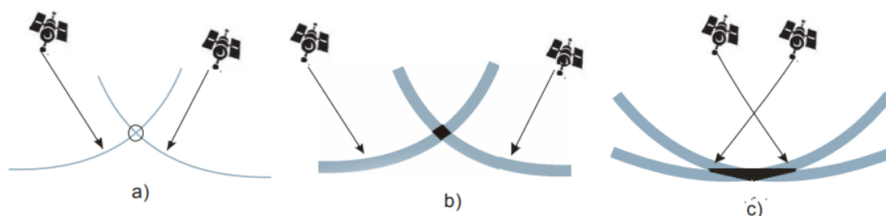


### Chyba polohy družice:

Poloha družice popsaná v navigační zprávě pomocí efemerid vychází z předpovědi pohybu družice, určené v monitorovacích stanicích rozmístěných po celém světě. Chyba této předpovědi může dosahovat velikosti okolo 2 m. Tato chyba ovlivňuje především autonomní určení polohy než výsledky diferenčních měření (viz kapitola 5). Satelity jsou neustále monitorovány a po určitém časovém odstupu je k dispozici zpřesněná verze jejich efemerid.

### Snížení přesnosti vlivem geometrie družic

Jak již bylo popsáno výše, absolutní poloha přijímače je určována pomocí protínání z délek. Přesnost určení polohy je závislé nejenom na počtu přijímaných satelitních signálů (resp. určených vzdáleností a jejich přesnosti), ale také na vzájemném geometrickém rozložení družic GNSS vůči přijímači, což je graficky znázorněno na obrázku 4.1. Tento vliv je v literatuře označován zkratkou GDOP (z angl. *Geometric Dilution of Precision*).



**Obrázek 4.1:** Vliv geometrie na přesnost polohy - zdroj [2]

Na obrázku 4.1a je naznačen případ, kdy měřená vzdálenost mezi přijímačem a satelity není zatížena žádnou chybou. V takovém případě je poloha přijímače reprezentována jediným bodem a to průnikem dvou kružnic. Obrázky 4.1b a 4.1c pak představují reálný případ, kdy je měřená vzdálenost určena s jistou přesností, poloha přijímače tak není reprezentována jedním bodem, ale prostorovým objektem, uvnitř kterého s jistou pravděpodobností leží námi určovaná poloha přijímače. Na obrázku 4.1c lze vidět, že se díky změně konstelace satelitů tento prostor zdatelně zvětšil [2].

DOP je bezrozměrné číslo, které nám poskytuje informaci o kvalitě řešení výpočtu v daném okamžiku. Čím více se jeho hodnota zvyšuje, tím horších výsledků dosahujeme. DOP lze proto využít jako indikátor kvality výpočtu polohy a jeho hodnoty lze určit z prvků kovarianční matice, kterou získáme při vyrovnání kódového měření metodou nejmenších čtverců. Podle toho, kterou veličinu posuzujeme, lze parametr DOP rozdělit na dílčí složky, které vyjadřují kvalitu určení daného typu informace:

- GDOP - parametr přesnosti určení polohy a korekce hodin přijímače
- PDOP - parametr přesnosti prostorového určení polohy
- HDOP - parametr horizontální přesnosti
- VDOP - parametr vertikální přesnosti
- TDOP - parametr přesnosti určení korekce hodin přijímače



### 4.3 Chyby související s přijímačem

#### Chyba hodin přijímače:

Jednou z největších chyb je již zmiňovaná nesynchronizovanost hodin přijímače a satelitů. Tato chyba se následně vylučuje početně, kdy je označena za další neznámou ve výpočtu. Kvůli tomuto kroku se pak zvýší minimální počet satelitů, které potřebujeme pro výpočet polohy a času na čtyři.

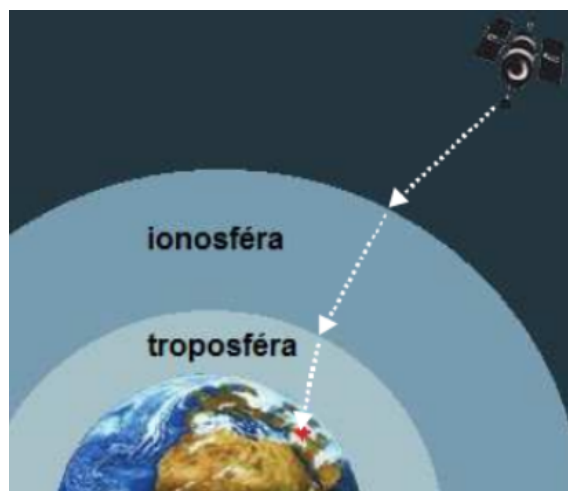
#### Variace fázového centra:

Fázové centrum GNSS přijímače definuje zdánlivý bod, ke kterému jsou vztaženy geometrické vzdálenosti mezi anténou a družicí. Tento zdánlivý bod není pevný, ale je závislý na frekvenci přijímaných signálů a také na azimutu a elevačním úhlu, ze kterého signál přichází. Pohybem družic vůči přijímači tedy dochází k změnám úhlů dopadu signálu a tedy k variacím (pohybu) tohoto fázového centra přijímače.

Velikost tohoto jevu dosahuje v horizontální složce několika *mm* a ve vertikální složce až *cm*, proto se projeví až při velmi přesných měřeních. K jejímu odstranění je potřeba do výpočtů zavést ofsety fázového centra, které se určují kalibrací antény.

### 4.4 Chyby způsobené průchodem signálu prostředím

Jak bylo popisováno v kapitole o kódovém měření, je výpočet pseudovzdáleností založen na předpokladu, že se GNSS signál šíří rychlostí světla. To však platí pouze v nejvyšších vrstvách atmosféry, kde jsou vlastnosti prostředí podobné vakuu. V nižších vrstvách atmosféry však vlivem koncentrace různých plynů, vlhkosti, teplotě a mnoha dalších faktorů dochází ke zbrždění rychlosti šíření. Také trajektorie signálu mezi přijímačem a satelitem není přímá, ale díky různému indexu lomu určitých vrstev atmosféry dochází k jejímu ohybu a zalomení (viz obr. 4.2). Stejně jako u elektronických dálkoměrů je tento jev označován jako tzv. *refrakce*. Největší vliv na GNSS signál mají vrstvy atmosféry označované jako **troposféra** a **ionosféra**.



Obrázek 4.2: Trajektorie signálu - zdroj [2]



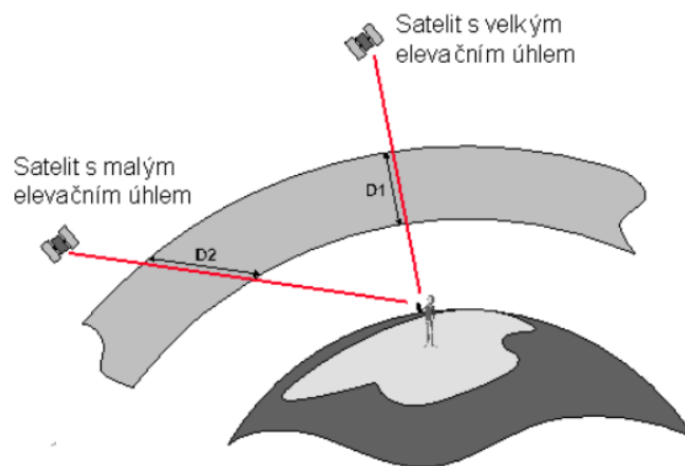
##### Ionosférická refrakce:

Ionosféra je část zemské atmosféry, která se nachází přibližně ve výšce 70-1000 km nad povrchem Země. V této vrstvě dochází vlivem sluneční energie a kosmického záření k ionizaci molekul plynů ( $H_2$ ,  $H_e$ ,  $O_2$  a  $N_2$ ), tedy uvolňování elektronů z jejich molekul. Ionosférické zpoždění signálu je závislé na množství volných elektronů v trajektorii signálu, které se mění v závislosti na denní době, průběhu ročního období a také na sluneční aktivitě.

Prostředí ionosféry působí také odlišně na signály o různých frekvencích, kdy signály s nižší frekvencí jsou ovlivňovány více. Tohoto jevu využívají dvou frekvenční přijímače, které dokáží vhodnou lineární kombinací přijatých signálů téměř eliminovat vliv ionosférické refrakce. Jednofrekvenční přijímače využívají k opravě této refrakce tzv. *ionosférický model*, jehož koeficienty jsou předávány přijímači v navigační zprávě.

##### Troposférická refrakce:

Troposféra je nejnižší vrstvou atmosféry, jejíž výška nad povrchem je proměnlivá. V okolí zemských pólů se pohybuje okolo 9 km, zato na rovníku dosahuje výšky až 16 km. Na rozdíl od ionosféry, prostředí troposféry není disperzní pro frekvence až do 15 GHz. Z tohoto vyplývá, že všechny satelitní signály jsou touto vrstvou atmosféry ovlivněny stejně, bez ohledu na jejich frekvenci. Proto tuto chybu nelze eliminovat měřením na dvou frekvencích jako v případě ionosférické refrakce.



**Obrázek 4.3:** Závislost délky průchodu atmosférou na elevačním úhlu - zdroj [22]

Velikost troposférické refrakce je závislá na době šíření signálu touto vrstvou atmosféry, tedy na elevačním úhlu satelitu a nadmořské výšce přijímače (viz obr. 4.3). Čím je tato doba kratší, tím je signál ovlivněn méně. Proto se v přijímačích nastavuje tzv. *elevační maska*, která definuje minimální hodnotu elevačního úhlu<sup>5</sup> a satelity s nižší elevací jsou z výpočtu polohy vyřazeny.

<sup>5</sup>Elevace je úhel, pod kterým se nachází družice vůči horizontální rovině přijímače.



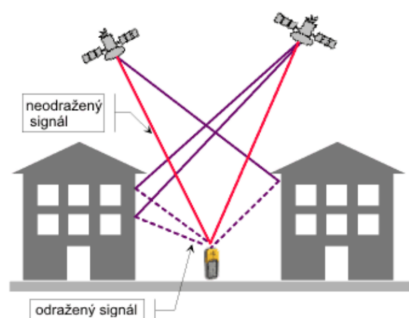
Vliv troposférické refrakce se následně potlačuje využitím matematických refrakčních modelů, vhodnou volbou elevační masky a také využitím diferenciálního měření, kdy pro nepřilíh vzdálené přijímače je ovlivnění satelitních signálů atmosférou silně závislé.

#### 4.5 Chyby související s umístěním přijímačem

##### Multipath:

Je označení pro vícecestné šíření signálů, při kterém dochází k odrazům radiových vln od reflexních objektů v okolí přijímače nebo od země (viz obr 4.4). K tomuto jevu nejčastěji dochází v zastavěném či vegetací zarostlém území. Přijímač následně zpracovává jak přímý, tak signál odražený od blízkých ploch. Trasa odraženého signálu je logicky delší a tím dochází k chybám ve zpracování měření.

Tento efekt lze částečně eliminovat výběrem vhodné antény, která tyto odražené signály dokáže identifikovat<sup>6</sup> a vyloučit ze zpracování. Další možností je vhodná volba stanoviště měření nebo delší doba observace.



Obrázek 4.4: Multipath - zdroj [43]

##### Útlum signálu:

Průchod signálu prostředím způsobuje jeho útlum, nachází-li se v cestě jeho šíření nějaká překážka, může dojít k odrazu (viz **Multipath**), průchodu nebo úplnému pohlcení signálu v závislosti na typu a tloušťce překážky. Při cestě signálu k přijímači může dojít k takovému útlumu, že se svým výkonem dostane pod úroveň citlivosti přijímače a ten jej nezaznamená.

## 5 Metody měření polohy pomocí GNSS

Základní rozdělení metod GNSS měření je na **absolutní** (autonomní) a **relativní** (diferenciální). Přičemž absolutním určováním polohy rozumíme přímé určení souřadnic v reálném čase v daném souřadnicovém systému, konkrétně u GPS jsou v geocentrickém systému WGS-84. K měření postačí pouze jeden přijímač, který určuje svou polohu pomocí výpočtu z určených pseudovzdáleností. Nevýhodou této metody je nízká přesnost, která se pohybuje kolem několika metrů v závislosti na počtu družic, jejich geometrickému rozložení a mnoha dalších faktorech. Proto se tato metoda uplatňuje především k navigaci

<sup>6</sup>K identifikaci odraženého signálu se využívá principu polarizace nosného signálu, kdy přímočarý signál je polarizovaný pravotočivě, zatímco odražený signál levotočivě [13].



nebo mapování pro GIS.

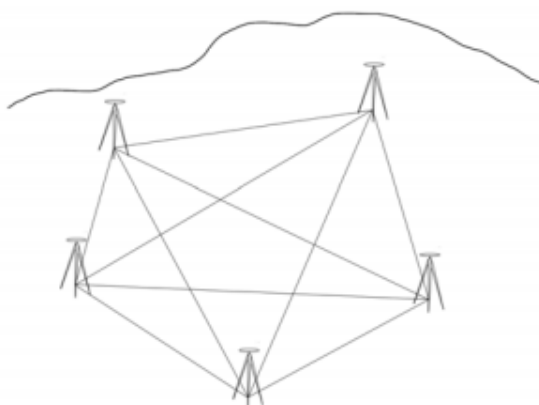
Pro geodetické účely se používá relativní metoda, kdy současně měří minimálně dva přijímače. Výsledkem je rozdíl poloh těchto přijímačů, tedy vektor spojnice bodů. Pro určení souřadnic je nutné, aby jeden z přijímačů (označovaný jako základnová stanice - „Base“) byl umístěn na bodě o známých geocentrických souřadnicích a druhý přijímač (pohyblivá stanice - „Rover“) na bodě, jehož souřadnice určujeme. Díky tomu, že známe přesné souřadnice základnové stanice, můžeme pro daný okamžik vypočítat chybu určení jeho polohy, která je také označovaná jako *korekce*. Poté je možné o tyto korekce opravit polohu nově změřených bodů, jelikož se vychází z předpokladu, že chyby, které nastávají u měření nepřímých vzdálených přijímačů, jsou silně závislé. Korekce lze zavádět přímo v reálném čase, kdy jsou přenášeny pomocí internetu nebo radiomodemů. Další možností je zpracování dat z referenční a pohyblivé stanice až po ukončení měření tzv. *postprocesingem*.

Následující podkapitoly popisují relativní metody měření a vychází z těchto zdrojů [13][29][38].

### 5.1 Statická metoda

Tato metoda zajišťuje nejlepší výsledky určení relativní polohy na dlouhé vzdálenosti, při níž lze dosáhnout přesnosti 3-5mm, kdy je nutné současné kontinuální měření dvou či více přijímačů. Vysoká přesnost je však vyvážena dobou měření, která může být pár hodin až dnů a následným složitým post-procesním zpracováním. Tuto metodu lze využít k budování geodetických polohových základů či sledování posunů a přetvoření.

Pro dosažení vysoké přesnosti je nezbytné měření plánovat a nalézt správný časový interval pro observaci z důvodu zajištění dostatečného počtu viditelných družic a jejich správné konfigurace (viz DOP). Dále je důležitá volba základny kvůli omezení počtu překážek.



Obrázek 5.1: Statická metoda měření - zdroj [13]

### 5.2 Rychlá statická metoda

Postup měření vychází ze statické metody, ale observace na měřených bodech je mnohem kratší (minuty až desítky minut). Zkrácení délky observace se projevuje ve snížení

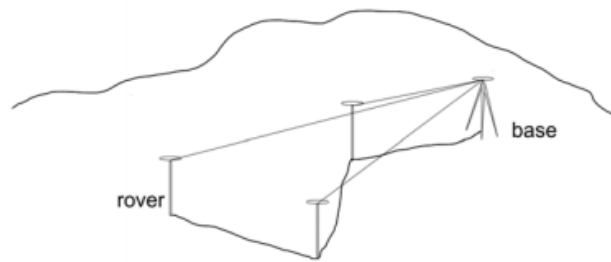




přesnosti určení polohy ( $5-10 \text{ mm} + 1 \text{ ppm}^7$ ) a také omezení vzdálenosti měřeného bodu od referenční stanice, které lze uskutečnit v okruhu 15 km. Tato metoda je nejčastěji využívána pro zhušťování základních i podrobných bodových polí.

### 5.3 Metoda Stop and Go

Tato metoda je podobná rychlé statické metodě, přijímač však nepřestává měřit ani při přechodu mezi body. Na prvním bodě je nutné setrvat do té doby dokud není spolehlivě vyřešena počáteční fázová ambiguita a pokud nedojde k výpadku signálu, lze na dalších podrobných bodech měřit jen velice krátkou dobu. Jestliže dojde při měření k výpadku signálu, měření přejde v rychlou statickou metodu.



Obrázek 5.2: Metoda Stop and Go - zdroj [13]

### 5.4 RTK (Real Time Kinematic)

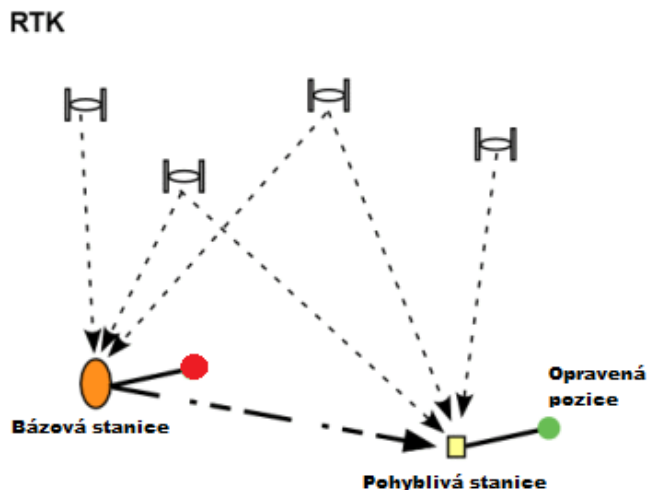
Tato metoda je v současné době jednoznačně nejpoužívanější a je zároveň nejmladší. Jak vyplývá z názvu metody („*Real time*“), poskytuje souřadnice přijímače s dostatečnou přesností (řádově centimetry) v reálném čase a díky tomu lze tuto metodu využít k vytyčování polohy podrobných bodů. K jejímu používání je nutný dostatečně výkonný hardware, který umožní rychlý výpočet ambiguit a dále zařízení pro přenos měřených dat (popřípadě korekcí) pomocí internetu nebo radiomodemu. K měření jsou tedy nutné dva přijímače umožňující fázové měření, je však možné místo jednoho přijímače využít permanentní stanici, která za poplatek poskytuje měřené informace.

Základnová stanice je umístěna na bodě o známých souřadnicích a přijímá data od viditelných družic. Pomocí známých souřadnic a souřadnic určených měřením se vypočtou korekce, které jsou pak předávány dalším přijímačům v okolí základnové stanice (viz obr. 5.3).

### 5.5 DGNSS

Tato metoda je svým principem totožná s metodou RTK, kdy se pomocí korekcí zasílaných referenční stanicí zpřesňují výsledky měření. Rozdíl metod spočívá v tom, že u DGNSS se zpracovávají pouze kódová měření a výsledná přesnost určení polohy tak dosahuje řádově několika decimetrů. Tyto korekce mohou být také přenášeny prostřednictvím geostacionárních družic, na území Evropy například systémem EGNOS.

<sup>7</sup>Parts per million (zkráceně **ppm**) označuje výraz pro jednu miliontinu celku. V tomto konkrétním případě je přesnost polohy určena konstantní složkou plus 1 mm za každý kilometr vzdálenosti mezi referenční stanicí a měřeným bodem.



Obrázek 5.3: Metoda RTK - zdroj [35] upraveno

## 6 Komunikační protokoly

K tomu abychom rozuměli datům poskytovaným elektronickými zařízeními, je nutné znát jejich komunikační protokol, který definuje formu a pravidla komunikace pro přenášení dat a instrukcí mezi připojenými zařízeními. Protokoly pro určité typy zařízení se standardizují proto, aby si každý výrobce nemusel definovat svůj vlastní protokol, což by výrazně ztížilo propojitelnost s ostatními přístroji a programy a tím by snížil prodej jejich výrobků. V oblasti GNSS se nejčastěji setkáme s protokoly NMEA, RTCM, NTRIP a také datovým formátem RINEX.

### 6.1 NMEA

NMEA je komunikační protokol vyvíjený a udržovaný americkým *Národním sdružením pro loďní elektroniku* (National Marine Electronics Association). Tento protokol byl vyvinut kvůli sjednocení komunikace mezi elektronickými zařízeními používanými pro loďní dopravu, avšak v tomto textu se budeme nadále zabývat pouze zprávami, které se využívají v oblasti GNSS. Přijímače posílají na výstupní port v pravidelném intervalu textová data s pevně danou strukturou (označované jako věty) skládající se z ASCII znaků, přičemž každá z nich obsahuje specifické informace získané přijímačem. NMEA zpráv je celá řada a každý přijímač dokáže vysílat jen určitou část z nich. Existují však věty, které jsou podporovány všemi zařízeními. Tento formát má však nevýhodu v tom, že data obsažená ve zprávách nelze využít k postprocesnímu zpracování.

Základní struktura NMEA zprávy:

- Každá zpráva začíná znakem „\$“ a končí <CR><LF>, tedy ukončením řádku a posunem na další.

\$ttsss,d1,d2,,...\*6D<CR><LF>

- Věta může obsahovat maximálně 80 znaků plus počáteční a koncové znaky, dohromady tedy 83 bajtů.
- První dva znaky „tt“ označují identifikátor mluvčího (např. GP - Global positioning system, GL - GLONASS nebo GN - kombinace GPS a GLONASS atd.).



- Znaky „sss“ specifikují daný typ věty (nejběžnější jsou popsány níže).
- Po specifikaci daného typu zprávy následují datové položky, přičemž jako oddělovač je použita čárka. Pokud některá data nejsou k dispozici zůstane datové pole prázdné, ale čárky oddělující pole zůstávají.
- Před ukončením zprávy je umístěn kontrolní součet, který začíná znakem „\*“ a za ním následují dvě hexadecimální číslice představující logickou operaci XOR (exclusive OR) ze všech znaků mezi „\$“ a „\*“.

Zde je uveden výpis jedné sekvence zpráv z modulu U-Blox NEO-M8T.

```
$GNRMC,113029.00,A,5025.44415,N,01411.37865,E,0.042,,030319,,A*60
$GNGGA,113029.00,5025.44415,N,01411.37865,E,1,12,0.91,216.1,M,44.1,M,,*4C
$GNGSA,A,3,24,15,13,19,10,30,20,28,17,,1.50,0.91,1.19*1D
$GNGSA,A,3,85,66,67,65,83,74,76,75,84,,1.50,0.91,1.19*11
$GPGSV,4,1,13,05,01,195,27,07,,24,10,09,332,33,12,02,219,22*42
$GPGSV,4,2,13,13,65,156,38,15,67,262,47,17,32,114,35,18,01,029,32*72
$GPGSV,4,3,13,19,18,141,45,20,20,307,46,24,36,278,48,28,46,057,36*73
$GPGSV,4,4,13,30,15,088,23*4D
$GLGSV,3,1,10,65,41,199,38,66,51,302,49,67,08,337,36,72,04,174,27*6F
$GLGSV,3,2,10,74,21,058,25,75,73,013,43,76,48,266,44,83,08,000,22*60
$GLGSV,3,3,10,84,23,054,22,85,12,109,26*6A
$GNGST,113029.00,26,,1.3,3.2,3.2*44
```

Každá sekvence zpráv v sobě obsahuje spoustu dat, ale na první pohled nemusí být zcela zřejmé, co která položka znamená. Pro správnou interpretaci informací je tedy nejdříve nutné popsat datovou strukturu zpráv. Informace o jednotlivých větech byly čerpány z těchto zdrojů [42][25].

### 6.1.1 RMC

Zpráva RMC (z angl. Recommended minimum specific GPS/Transit data) obsahuje doporučené minimum informací pro navigaci v prostoru a informace o čase.

Struktura zprávy:

```

      1      2      3      4      5      6  7  8      9 10 11 12
      |      |      |      |      |      |  |  |      |  |  |
$GNRMC,113029.00,A,5025.44415,N,01411.37865,E,0.042,,030319,,A*60
```

pozice:	formát	příklad	popis
1)	hhmmss.ss	11:30:29.00	Aktuální čas v systému UTC
2)	c	A	Status přijímače (A = OK , V = varování)
3)	ddmm.mmmm	50°25.44415"	Zeměpisná délka
4)	c	N	Identifikátor polohy N/S (sever/jih)
5)	ddmm.mmmm	14°11.37865"	Zeměpisná šířka
6)	c	E	Identifikátor polohy E/W (východ/západ)
7)	d.d	0.042	Vodorovná rychlost v uzlech
8)	d.d	Null	Kurz pohybu ve stupních
9)	ddmmyy	3.3.2019	Aktuální datum



10)	d.d	Null	Magnetická deklinace ve stupních
11)	I	A	Indikátor módu: <ul style="list-style-type: none"> <li>• A - autonomní mód</li> <li>• D - diferenční mód</li> <li>• E - odhadovaný mód</li> <li>• N - data nejsou platná</li> </ul>
12)	*xx	*60	Kontrolní součet

Tabulka 6.1: Popis částí NMEA zprávy RMC

### 6.1.2 GSA

GSA je jedna ze základních zpráv, která uvádí identifikační čísla satelitů, jež se používají k výpočtu polohy přijímače a parametry přesnosti DOP<sup>8</sup>.

Struktura zprávy:

```

                                11 13
      1 2 3  4  5  6  7  8  9 10 |12|14 15   16   17 18
      | | |  |  |  |  |  |  |  | | | |  |   |   |
$GNGSA,A,3,24,15,13,19,10,30,20,28,17,, , ,1.50,0.91,1.19*1D

```

pozice:	formát	příklad	popis
1)	c	A	Přepínání mezi n-rozměrnými módy (A=automatické,M=manuální)
2)	d	3	Typ fixování: <ul style="list-style-type: none"> <li>• 1 - není k dispozici</li> <li>• 2 - 2D fixování polohy</li> <li>• 3 - 3D fixování polohy</li> </ul>
3)-14)	d	24,15..	ID satelitů
15)	d.d	1.50	PDOP - parametr polohové přesnosti
16)	d.d	0.91	HDOP - parametr horizontální přesnosti
17)	d.d	1.19	VDOP - parametr vertikální přesnosti
18)	*xx	*1D	Kontrolní součet

Tabulka 6.2: Popis částí NMEA zprávy GSA

### 6.1.3 GGA

Další rozšířená věta má zkratku GGA - Global Positioning System Fix Data, která v sobě nese informace poloze přijímače a její kvalitě, popřípadě zda jsou použita korekční data z referenční stanice, její identifikační číslo a stáří dat.

<sup>8</sup>DOP (z angl. *dilution of precission*) je bezrozměrná veličina, která popisuje vliv geometrie GNSS satelitů a přijímače v dané epoše na přesnost určení polohy. Vyšší hodnota DOP signalizuje horší geometrické rozestavení a tím i klesající přesnost určení polohy.



Struktura zprávy:

13 15

```

1      2      3      4      5 6 7  8      9      10 11 12 | 14 |
|      |      |      |      | | |  |      |      | | | |
$GNGGA,113029.00,5025.44415,N,01411.37865,E,1,12,0.91,216.1,M,44.1,M,,*4C

```

pozice:	formát	příklad	popis
1)	hhmmss.ss	11:30:29.00	Aktuální čas v systému UTC
2)	ddmm.mmmm	50°25.44415"	Zeměpisná délka
3)	c	N	Identifikátor polohy N/S (sever/jih)
4)	ddmm.mmmm	14°11.37865"	Zeměpisná šířka
5)	c	E	Identifikátor polohy E/W (východ/západ)
6)	d	1	Indikátor GPS kvality: <ul style="list-style-type: none"> <li>• 0 - neplatné</li> <li>• 1 - GPS fix</li> <li>• 2 - DGPS fix</li> <li>• 3 - PPS - fix</li> <li>• 4 - RTK - fix</li> <li>• 5 - RTK - float</li> <li>• 6 - autonomní</li> <li>• 7 - manuální vstup</li> <li>• 8 - simulační mód</li> </ul>
7)	d	12	Počet viditelných satelitů
8)	d.d	0.91	HDOP - parametr horizontální přesnosti
9)	d.d	216.1	Výška přijímače nad geoidem <sup>9</sup>
10)	c	M	Jednotka pro předchozí údaj - metry
11)	d.d	44.1	Rozdíl mezi elipsoidem a geoidem
12)	c	M	Jednotka pro předchozí údaj
13)	d	Null	Stáří posledních DGNSS dat ve vteřinách → Null pokud se nepoužívají
14)	d	Null	Identifikační číslo referenční stanice
15)	*xx	*60	Kontrolní součet

**Tabulka 6.3:** Popis částí NMEA zprávy GGA

#### 6.1.4 GSV

Informace o pozici a síle signálů z viditelných satelitů nám poskytne GSV zpráva (z angl. GNSS Satellites in View). Každá NMEA zpráva má omezenou délku (83 bajtů), což v tomto případě vystačí pouze pro zápis informací o čtyřech satelitech. Informace proto bývají rozdělené do několika dílčích zpráv.

<sup>9</sup>Geoid si lze představit jako střední hladinu moří, která prochází i pod povrchem Země. Každý bod této nepravidelné plochy má stejnou hodnotu tíhového potenciálu.



Struktura zprávy:

```

1 2 3 4 5 6 7 8 ..... 12 ..... 16 ..... 20
| | | | | | | | ..... | ..... | ..... |
$GPGSV,3,1,11,16,07,300,14,21,41,189,26,23,05,330,15,25,41,126,16*7C

```

pozice:	formát	příklad	popis
1)	d	3	Celkový počet dílčích zpráv
2)	d	1	Aktuální číslo zprávy
3)	d	11	Počet viditelných družic
4)	d	16	Identifikační číslo satelitu
5)	d	7°	Elevační úhel satelitu nad horizontem
6)	d	300°	Azimut k satelitu
7)	d	14dB	SNR <sup>10</sup> – 0-99dB
8)-11)			Informace o druhém satelitu
12)-16)			Informace o třetím satelitu
16)-19)			Informace o čtvrtém satelitu
20)	*xx	*7C	Kontrolní součet

**Tabulka 6.4:** Popis částí NMEA zprávy GSV

### 6.1.5 GST

V této poslední NMEA zprávě, která zde bude zmíněna, jsou uvedeny statistické parametry popisující přesnost určení polohy přijímače.

Struktura zprávy:

```

1      2 3 4 5 6 7 8 9
|      | | | | | | | |
$GNGST,113029.00,26, , , ,0.8,0.6,1.5*44

```

pozice:	formát	příklad	popis
1)	hhmmss.ss	11:30:29.00	Aktuální čas v systému UTC
2)	d	26	RMS <sup>11</sup> - hodnota sm. odchylek pseudovzdáleností
3)	d	Null	Velikost hlavní poloosy elipsy chyb [m]
4)	d	Null	Velikost vedlejší poloosy elipsy chyb [m]
5)	d	Null	Azimut hlavní poloosy [°]
6)	d	0.8 m	$\sigma$ zeměpisné šířky [m]
7)	d	0.6 m	$\sigma$ zeměpisné délky [m]
8)	d	1.5 m	$\sigma$ nadmořské výšky [m]
9)	*xx	*44	Kontrolní součet

**Tabulka 6.5:** Popis částí NMEA zprávy GST

<sup>10</sup>SNR ( z angl. Signal Noise Ratio) vyjadřuje kvalitu navigačních signálů dané družice vyjádřenou hodnotou odstupe užitelného signálu od šumu v rozmezí 0-99 dB. Obecně lze tedy říct, že čím vyšší hodnota, tím je kvalita signálu lepší.

<sup>11</sup>RMS vychází z angl. *Root Mean Square* a jedná se v podstatě o kvadratický průměr určité hodnoty



## 6.2 RTCM

Protokol slouží k přenosu diferenčních korekcí z referenční stanice k jednotlivým uživatelům. Standard vyvinula americká organizace **R**adio **T**echnical **C**ommision for **M**aritime **S**ervices. Korekce jsou zaváděny z důvodu potlačení systematických chyb měření, která jsou při dodržení určitých podmínek pro referenční stanici a přijímač uživatele podobná [33]. Mezi tyto chyby patří :

- Ionosférická refrakce
- Troposférická refrakce
- Nepřesné hodnoty efemerid družic
- Nepřesnost v synchronizaci satelitních hodin
- Umělé zavedení chyby do nosného signálu

Data přenášená pomocí tohoto protokolu jsou v binární podobě a nelze je zobrazit pomocí terminálového programu, ve kterém se zobrazí pouze jako zmeř ASCII znaků. Například takto:

```
mRMP@PJfeUtNsmMFM{nVtIOTDbA^xGh~kDH`_FdW_yqLRryrDuh
cB\@}N`ozbSD@0^}nrGqkeTlpLLrYpDqAsrLRrQN{zW|uW@H`z]~aG
xWYt@I`_FxW_qqLRryrDCikA\@Cj]DE]|E@w_mlroMNjkK0smMFM{
WDwW@HVEbA^xGhLJQH`_F`W_aNsmMFM[WVLA\@S]amz@i1IuP
qx~IZhTCpLLrYpdP@k0smMFM[kVDHwVGbA^P{WWuNt_SW_yMs
mMnqdrhcC\@sE^ZfC@}vJmNGAHJVhTCqLRryrdviStW@H_GbA^
P{wxu[k
```

Tato data jsou složena z jednotlivých zpráv, přičemž každá z nich má v sobě zaneseny specifické informace. K jejich odlišení slouží identifikátor zprávy, který je pro každou z nich jedinečný. Popis jednotlivých verzí a zpráv tohoto formátu je možné za poplatek stáhnout na oficiálních stránkách [33].

## 6.3 NTRIP protokol

V této kapitole je čerpáno ze zdrojů [26][5].

NTRIP je komunikační protokol určený pro přenos dat prostřednictvím internetu, který byl vyvinut na počátku 21. století v německé BKG (Německá národní agentura pro kartografii a geodézii). Hlavním úkolem tohoto protokolu bylo distribuovat korekční data ve formátu RTCM a podle toho byl také pojmenován **Networked Transport of RTCM via Internet Protocol**. Umožňuje streamování<sup>12</sup> přes libovolnou mobilní síť (GSM, GPRS, EDGE nebo UMTS), jelikož je založen na protokolu TCP/IP.

Systém NTRIP je rozdělen do čtyř základních objektů (viz obr. 6.1) :

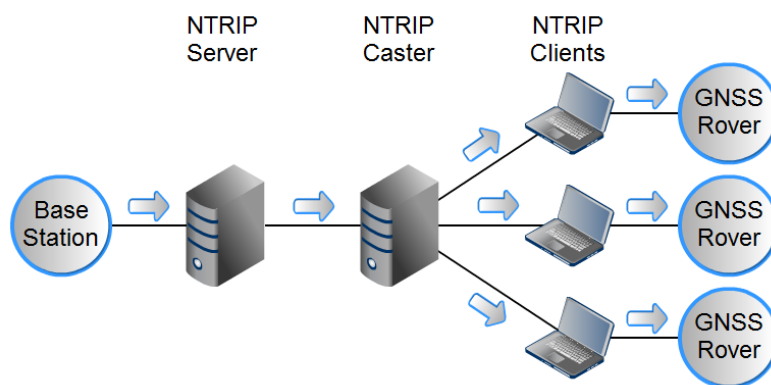
- **NTRIP Source** - je zdroj GNSS dat, konkrétně tedy přijímač umístěný na pozici o známých souřadnicích. Výpočetním postupem jsou pak z rozdílů známých souřadnic a souřadnic určených měřením získány korekce pro danou lokalitu. Zdroj korekcí může být také poskytnut z tzv. virtuální stanice, která se umístí poblíž lokace NTRIP

<sup>12</sup>Streaming je datová technologie umožňující kontinuální přenášení dat pomocí internetu v reálném čase.



Clienta a hodnoty korekcí jsou určeny síťovým řešením. Korekce jsou generovány převážně ve formátu RTCM a jsou následně přenášeny do NTRIP Serveru. Také může nastat situace, kdy je NTRIP Source zároveň NTRIP Serverem.

- **NTRIP Server** - je server, který je spojen s NTRIP Source a generovaná data dále přeposílá do NTRIP Casteru. NTRIP Server se připojí ke NTRIP Casteru, ve kterém je pro každý NTRIP Source administrátorem přidělen tzv. *mountpoint*, což je zástupné jméno pro NTRIP Source.
- **NTRIP Caster** - je opět server, který na požadavek od klientů přeposílá data z NTRIP Serverů. NTRIP Caster dále zajišťuje správu uživatelských účtů a hesel, přičemž data z mountpointů s omezeným přístupem poskytuje pouze po úspěšné autorizaci.
- **NTRIP Client** - je aplikace, která zašle pomocí TCP/IP požadavek NTRIP Casteru, který po autorizaci a ověření správnosti dotazu začne odesílat korekční data. Pokud klient neví, jaké mountpointy jsou dostupné, pošle požadavek o zaslání source-table, což je zdrojová tabulka obsahující informace o dostupných mountpointech. Struktura source-table je podrobněji popsána níže.



Obrázek 6.1: Schéma NTRIP sítě - zdroj [27]

Dále si ukážeme komunikaci mezi NTRIP Clientem a NTRIP Casterem, konkrétně na státě provozované plošné síti referenčních stanic CZEPOS. Požádáme o zaslání zdrojové tabulky dostupných mountpointů a následné poskytnutí korekcí z námi definovaného mountpointu.

### 6.3.1 Komunikace NTRIP Klienta s NTRIP Casterem

Jak již bylo zmíněno, komunikace mezi NTRIP Clientem a NTRIP Casterem probíhá pomocí TCP/IP spojení. Nejdříve se tedy klient připojí na adresu a port odpovídajícího NTRIP Casteru, a poté co je spojení navázáno, zašle požadavek GET, který má následující strukturu:

```
GET [mountpoint] HTTP/1.0 <CR><LF>
Host: [hostname] <CR><LF>
User-Agent: NTRIP [produkt] <CR><LF>
Authorization: [metoda] [credentials] <CR><LF>
Connection: close <CR><LF>
<CR><LF>
```





s parametry :

- **Mountpoint** - je zástupný název NTRIP Source, z kterého požadujeme data.
- **Hostname** - je adresa serveru, ke kterému jsme připojeni.
- **Produkt** - předává informace o NTRIP klientovi.
- **Metoda** - definuje způsob identifikace uživatele.
- **Credentials** - obsahuje řetězec znaků „jmeno:heslo“, který je zašifrován pomocí algoritmu base64.

Pokud neznáme název dostupných mountpointů, odešleme tuto zprávu bez parametru [mountpoint]. Odpovědí na tento dotaz bude zasláná zdrojová tabulka, kterou by server odeslal i v případě, že bychom se dotazovali na neexistující mountpoint. Popis její struktury je v následující kapitole.

Dále mohou nastat tyto scénáře :

1. Pokud se uživatel správně autorizuje a požaduje data z existujícího mountpointu, NTRIP Caster odpoví :

```
ICY 200 OK
Server: GNSS Spider 7.0.0.7245/1.0
Date: pá, 19 4 2019 21:50:59 GMT Standard Time
```

a následně začne posílat korekční data.

2. Pokud uživatel zadá neplatné přihlašovací údaje, NTRIP Caster odpoví:

```
HTTP/1.0 401 Unauthorized
ICY 200 OK
Server: GNSS Spider 7.0.0.7245/1.0
Date: pá, 19 4 2019 21:44:34 GMT Standard Time
```

a ukončí spojení s klientem.

3. Dále může nastat situace, kdy požadujeme korekční data z virtuální referenční stanice, které je nutné zaslat naši polohu ve formě NMEA zprávy GGA (viz kap. 6.1.3), aby mohla být ref. stanice vygenerována v našem okolí. Výsledný požadavek bude mít následující tvar.

```
GET /iMAX3C HTTP/1.0
Host: czeaposr.cuzk.cz
User-Agent: NTRIP test
Authorization: Basic Z3JhdHVsdWppUHJlbG96aWxKc2lWZXR1
```

```
$GNGGA,113029.00,5025.44415,N,01411.37865,E,1,12,0.91,216.1,M,44.1,M,,*4C
```

Pokud bychom GGA správu neodeslali, server by chvíli vyčkával a následně spojení ukončil.

Detailní popis komunikace je uveden zde [5].



### 6.3.2 Zdrojová tabulka

Tato tabulka obsahuje informace o NTRIP Source, které vysílají data skrze připojený NTRIP Caster. Tyto informace jsou klíčové pro správné připojení a komunikaci s NTRIP Castrem. Informují nás, zda je pro připojení nutné se autorizovat či zaslat naši polohu atd.

Poté co zašleme požadavek bez definování mountpointu nebo s uvedením nesprávného jména, zašle NTRIP Caster zpět zdrojovou tabulku v následujícím formátu:

```
SOURCETABLE 200 OK
Server: GNSS Spider 7.0.0.7245/1.0
Date: pá, 19 4 2019 22:52:06 GMT Standard Time
Content-Type: text/plain
Content-Length: 4082

STR;iMAX3C;VRS.iMAX-CZEPOS;RTCM 3;;;2;GPS;CZEPOS;CZ;49.50;15.50;1;1;CZEP . . . .
STR;RTK3;RTK.NEAREST.SITE-CZEPOS;RTCM 3.1;;;2;GPS & GLONASS;CZEPOS . . . .
. . . . .
. . . . .
. . . . .
ENDSOURCETABLE
```

Odpověď obsahuje informace o jednotlivých mountpointech, kdy pro každý mountpoint jsou uvedeny informace na jednom řádku, ve kterém jsou odděleny středníkem. Popis informací uvedených v jednotlivých polích je v tabulce 6.6.

pozice:	formát	příklad	popis
1)	3 znaky	STR	Typ záznamu
2)	max. 100 znaků	CPRG3	Název mountpointu
3)	neomezeně znaků	Praha	Název NTRIP Source (zdroje)
4)	neomezeně znaků	RTMC3	Formát dat
5)	neomezeně znaků		Dodatečné informace o formátu dat
6)	číslo	2	<u>Informace o nosné vlně:</u> <ul style="list-style-type: none"> <li>• 0 : žádné informace</li> <li>• 1 : nosná vlna L1</li> <li>• 2 : nosná vlna L1+L2</li> </ul>
7)	neomezeně znaků	GPS	Název používaného GNSS (GPS, GLONASS)
8)	neomezeně znaků	CZEPOS	Název sítě referenčních stanic
9)	3 znaky	CZE	ISO zkratka státu ve kterém leží stanice
10)	desetinné číslo	50.24	Zeměpisná délka mountpointu
11)	desetinné číslo	15.76	Zeměpisná šířka mountpointu
12)	číslo	0	<u>Typ mountpointu:</u> <ul style="list-style-type: none"> <li>• 0 : reálný mountpoint</li> <li>• 1 : virtuální mountpoint</li> </ul>
13)	číslo	0	<u>Původ dat:</u> <ul style="list-style-type: none"> <li>• 0 : data z jedné stanice</li> <li>• 1 : síťové řešení korekčních dat</li> </ul>
14)	neomezeně znaků		Generátor dat
15)	neomezeně znaků		Algoritmus pro kompresy dat



16)	1 znak	B	<u>Způsob autorizace klientů:</u> <ul style="list-style-type: none"><li>• N : none (žádná)</li><li>• B : basic (základní)</li><li>• D : digest (zajišťuje vyšší bezpečnost)</li></ul>
17)	1 znak	Y	<u>Zpoplatnění dat z mountpointu:</u> <ul style="list-style-type: none"><li>• Y : zpoplatněno</li><li>• N : nezpoplatněno</li></ul>
18)	číslo	9600	Rychlost datového toku (bps)
19)	neomezeně znaků		Dodatečné informace

**Tabulka 6.6:** Popis zdrojové tabulky zaslané z NTRIP Casteru CZEPOS

## 6.4 RINEX

Receiver Independent Exchange neboli ve zkratce RINEX, je uznávaný standard zápisu dat, který slouží pro zpracování observačních dat z pozorování GNSS, většinou v post-procesním režimu zpracování. Byl vyvinut univerzitním institutem astronomie v Bernu pro snadnou výměnu dat získaných v rámci evropské kampaně EUREF 89 ze 60 GPS přijímačů od 4 výrobců. Soubor navigační zprávy RINEX v.1 byl totožný s dřívějším NGS výměnným formátem [31].

Formát RINEX byl sestaven ze sedmi souborů :

- Soubor s observačními daty
- Soubor s navigační zprávou
- Soubor s meteorologickými daty
- Soubor s navigační zprávou GLONASS
- Soubor s navigační zprávou GEO
- Soubor s daty satelitů a hodin přijímačů
- Vysílací datový soubor SBAS (Space-based augmentation system)

Zpracování GNSS dat vychází z pozorovaných hodnot pro měření nosné fáze na jedné nebo dvou nosných vlnách měřených pseudovzdálenosti, času měření a Dopplerovského posunu, které jsou získávány přijímači a v neupravené (nezpracované) podobě jsou uloženy do těchto souborů. V současnosti, po optimalizaci dat nového formátu RINEX v.3.XX, se používají převážně první tři uvedené soubory.

Data v těchto souborech jsou zapisována pomocí ASCII znaků, přičemž na každém řádku jich může být maximálně 80. Formátování souboru je striktní, což znamená, že záleží na pozici v rámci řádky. Díky těmto vlastnostem jsou data dobře čitelná a je patrná jejich struktura. Každý soubor obsahuje hlavičku, kde je uveden název stanice, typy měřených dat, výška antény, přibližná poloha a jiné popisné informace, které se po celou dobu observace nemění. Za hlavičkou jsou dále obsažena měřená data, jejichž délka registrace není omezena. Soubor observace a meteorologických dat může obsahovat záznam z jednoho nebo více míst a systémů. To se však vzhledem k hospodárnosti dat nedoporučuje, protože by data pořízená ve stejný okamžik byla uložena několikrát. Soubor navigační zprávy umožňuje záznam z více satelitních systémů [32].



## Část II

# Vývoj aplikace

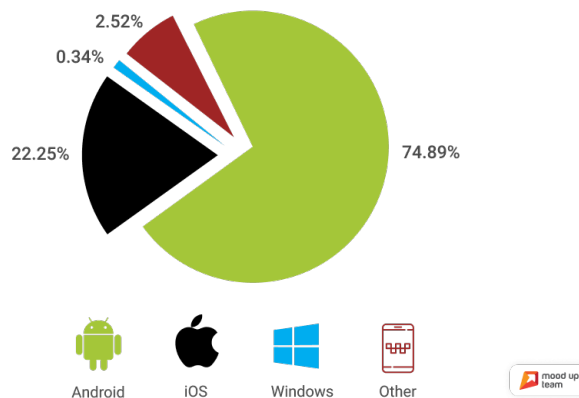
V této části bude popsáno vývojové prostředí Apache Cordova, které bylo použito pro vytvoření aplikace GNSS Kontrolér, postup jeho instalace a základní konfigurační příkazy a ovládání, které je nezbytné pro správné sestavení aplikace. Pro názornost si zde vytvoříme ukázkový projekt, pomocí něhož popíšeme adresářovou strukturu projektu, způsob vývoje a ladění projektu.

Dále zde bude představena samotná aplikace, především pak ukázka struktury jednotlivých funkčních oken a jejich funkcionalit.

## 7 Přístupy vývoje mobilních aplikací

Nejdříve si uvedeme základní přístupy vývoje mobilních aplikací, porovnáme jejich klady, zápory a omezení při vývoji určitých typů aplikací. Před samotným vývojem je proto nutné si rozmyslet několik důležitých podmínek, které následně ovlivňují přístup vývoje. Jedním z nejdůležitějších rozhodnutí je výběr platformy, pro kterou budeme aplikaci vytvářet a tedy jakému okruhu uživatelů bude poskytnuta. V dnešní době (rok 2019) na trhu převládají tři hlavní platformy, a to Android (Google), iOS (Apple Inc.) a malý podíl má i Windows Phone (Microsoft). Rozdělení trhu je shrnuto na obrázku 7.1.

Mobile Operating System Market - Dec 2018



**Obrázek 7.1:** Rozdělení mobilním trhu 2018 - zdroj [www.staging.moodup.team](http://www.staging.moodup.team)

Z obrázku je patrné, že většinový podíl mobilního trhu náleží firmě Google, konkrétně jejímu operačnímu systému Android. Pokud tedy požadujeme, aby byla naše aplikace dostupná pro co nejvíce uživatelů, pak je Android správnou volbou. Android aplikace jsou primárně dostupné přes online distribuční službu Google Play, kde jsou k dispozici tisíce aplikací. Android aplikace je však možné distribuovat i volně na internetových stránkách, kdy je pro instalaci takto dodané aplikace nutné v telefonu povolit instalaci aplikací z neznámých zdrojů<sup>13</sup>. Tento krok je však bezpečnostním rizikem, jelikož si můžete do telefonu zanechat škodlivý kód, a proto je důležité u takto instalovaných aplikací důsledně sledovat, jaká oprávnění daná aplikace vyžaduje a pokud nám nějaká

<sup>13</sup>Toto povolení se nejčastěji nachází v Nastavení/Zabezpečení



oprávnění připadají zvláštní, bude nejlepší aplikaci okamžitě vymazat. Kdybychom si například instalovali aplikaci s funkcí budíku a požadovala po nás oprávnění pro využití našich kontaktů a vnitřního uložště, je nasnadě se domnívat, že se jedná o malware<sup>14</sup>.

Apple, který své aplikace poskytuje pomocí svého obchodu App Store, si naopak více cení soukromí a bezpečnosti, a proto nepovoluje instalaci aplikací odjinud nežli ze svého obchodu. Před distribucí aplikace na App Store dále dochází k její kontrole. Uživatelé iOS se také vyznačují tím, že jsou ochotní za aplikace platit daleko více. Proto mnoho vývojářů, i přes menší počet potenciálních uživatelů, vyvíjí aplikace právě pro tuto platformu.

Základní dělení přístupů vývoje mobilních aplikací je následující [20][45]:

### Nativní aplikace :

Vývoj nativních mobilních aplikací je v dnešní době stále nejrozšířenějším přístupem a je i přímo doporučován poskytovateli mobilních platform. Pokud aplikaci označíme jako nativní, myslíme tím to, že je navržena a naprogramována pouze pro jednu mobilní platformu. Pokud tedy chceme aplikaci poskytovat pro iOS i Android, vyvíjí se pro každou platformu zvlášť, přestože je funkcionality totožná. Tento způsob vývoje pro více platform naráz je tedy značně zdoluhavý a finančně náročný. Výhodou tohoto přístupu je optimalizace kódu pro konkrétní platformu a díky tomu dosažení maximálního výkonu aplikace. Pokud má tedy naše aplikace zobrazovat 3D grafiku, upravovat audio/video soubory nebo zpracovávat jiné podobně náročné úkony, nezbude nám jiná možnost, než využít princip nativního vývoje.

Každá mobilní platforma využívá pro vývoj nativních aplikací svůj vlastní programovací jazyk:

- Android - Java / Kotlin
- iOS - ObjectiveC / Swift
- Windows Phone - C++ / C#

### Webové aplikace :

Webové aplikace, někdy také označované jako mobilní weby, jsou pouze dobře optimalizované webové stránky a nejedná se tedy o plnohodnotnou aplikaci. Tento přístup se nejčastěji využívá v případě, kdy aplikace pouze kopíruje funkcionality již běžícího webu. Nevýhodou je, že pro využití aplikace musí být uživatel připojen k internetu a také chybějící napojení na API telefonu, tedy možnost využití vnitřních funkcí našeho zařízení. Vývoj tímto přístupem je však velice rychlý a přímočarý.

### Hybridní aplikace :

Jak už vyplývá z názvu hybridní aplikace, je tedy určitou kombinací dvou výše popsaných přístupů programování. Základem každé hybridní aplikace je tzv. „WebView“, což je instance webového prohlížeče. Zjednodušeně řečeno, hybridní aplikace fungují tak, že se okno aplikace vyplní instancí WebView, ve které pak běží optimalizovaný obsah

<sup>14</sup>Malwarem označujeme jakýkoliv typ škodlivého softwaru, který se snaží infikovat počítač nebo mobilní zařízení.



webové aplikace. Hybridní aplikace jsou tedy založeny na jazycích určených pro tvorbu webu a to HTML, JavaScript (JS) a CSS. Pro využití nativních prvků telefonu se pak využívá instalace pluginu, které tyto funkce zpřístupňují a lze je využít voláním přímo z JS.

Výhodou tohoto přístupu je rychlý vývoj aplikací pro více mobilních platform naráz, jelikož většina napsaného kódu v HTML, JS a CSS se mezi platformami sdílí. Dalším bonusem je to, že se většina programovací práce dá testovat ve webovém prohlížeči a není tedy nutné aplikaci po každé změně znovu sestavovat a instalovat na fyzické nebo virtuální zařízení, díky tomu se vývoj enormně urychlí.

Po průzkumu metod vývoje mobilních aplikací a získání informací o jejich slabých a silných stránkách, jsem se přiklonil k vývoji hybridní aplikace, a to zejména z důvodu větší zkušenosti s „webovými“ programovacími jazyky než s těmi nativními. Pro vytvoření aplikace byl následně zvolen program Apache Cordova, jehož popis je v následujících kapitolách.

## 8 Apache Cordova

Apache Cordova (AC) je open-source vývojový frame-work pro vývoj mobilních aplikací založený na výše zmíněných principech hybridního vývoje, což nám tedy umožní používat standardní webové technologie. Framework je poskytován zdarma v rámci licence Apache Licence firmou Adobe. Tento vývojový nástroj si od doby svého vzniku (2008) prošel několika změnami názvů, přičemž ten původní, PhoneGap, dnes firma Adobe stále používá pro jiný framework, který je nadstavbou AC a doplňuje jeho funkcionalitu pomocí dalších produktů společnosti Adobe [15].



**Obrázek 8.1:** Ikona Apache Cordova - zdroj [7]

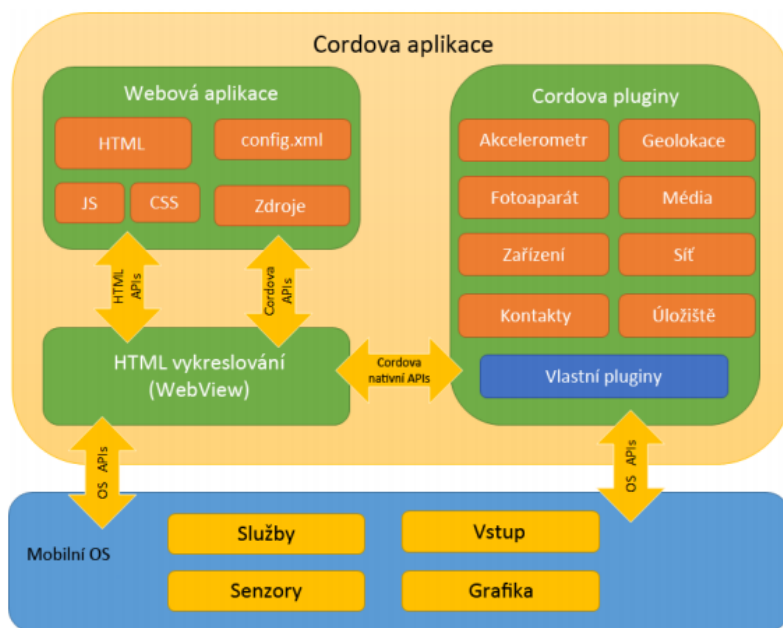
Vývoj aplikací v AC je v různé míře dostupný pro platformy Android, iOS, BlackBerry, Windows Phone, OS X a další (kompletní seznam je uveden zde [7]).

Schéma architektury aplikace vytvořené v AC je zobrazeno na obrázku **8.2**. Samotná aplikace je implementována jako webová stránka, jejíž výchozí soubor je defaultně nastaven jako `index.html`, který se následně odkazuje na soubory JS, CSS a další zdroje, jež jsou nezbytné pro správné zobrazení a ovládání aplikace. Dalším důležitým prvkem je konfigurační soubor `config.xml` obsahující zásadní informace o naší aplikaci, jako její název, popis, seznam nainstalovaných pluginů, ale také například reakci aplikace na změnu orientace přístroje.



Pluginy jsou nedílnou součástí systému AC, jedná se o kód napsaný v nativním jazyce pro každou platformu, který nám poskytuje vazby na standardní nativní API našeho zařízení. Díky tomu může naše aplikace přistupovat k uložišti přístroje, seznamu kontaktů nebo získat informace o poloze telefonu a mnoho dalších funkcí. Seznam všech standardních pluginů, označovaných také jako „Core plugins“, je na oficiálních stránkách AC [15]. Dále existuje možnost využití pluginů třetích stran, které zpřístupňují méně obvyklé funkce zařízení a lze je získat například z knihoven jako GitHub, NPM a dalších.

Podrobný popis jednotlivých souborů a adresářové struktury bude uveden v založeném ukázkovém projektu v kapitole 8.2.



Obrázek 8.2: Schéma architektury aplikace - zdroj [15]

### 8.1 Instalace nezbytných programů

AC neobsahuje žádné desktopové prostředí, ve kterém by se vývoj prováděl a jeho obsluha je realizována pomocí příkazů v terminálu, proto je vhodné si nainstalovat některý z textových editorů kódu pro vývoj webových stránek, který zobrazuje syntaxi použitého programovacího jazyka, automaticky doplňuje rozepsaný text a dokonce zvýrazňuje chyby kódu. Pro vývoj aplikací v AC je dále nutné si pro každou platformu zvlášť nainstalovat její SDK<sup>15</sup> [3]. V tomto textu se budeme nadále zabývat pouze platformou Android, proto implementace SDK pro jiné platformy zde není uvedena.

Seznam nutných programů je uveden zde, včetně odkazů na jejich dokumentaci a instalační soubory.

- [Node.js](#)<sup>16</sup>

Jedná se o softwarový systém umožňující spouštět JavaScript kód mimo prostředí prohlížeče, společně se s tímto systémem nainstaluje i systém NPM, což je správce JavaScriptových balíčků a jedním z nich je námi požadovaný balíček `cordova`.

<sup>15</sup>Software Development Kit (SDK) je sada vývojových nástrojů umožňující vytváření aplikace pro jednotlivé platformy.

<sup>16</sup><https://nodejs.org/en/>



- [Git](#)<sup>17</sup>  
Některé instalační soubory jsou uloženy v repozitářích, proto je nutné mít v počítači nainstalovaný verzovací systém Git.
- [Android Studio](#)<sup>18</sup>  
Je program pro tvorbu nativních aplikací vyvíjený společností Google. Součástí instalačního balíčku je samotné Android Studio IDE, Android SDK Tools, kompilátor Android a také základní emulátory s plnohodnotným systémem Android. Díky využití emulátorů nemusíme kvůli vývoji vlastnit fyzické zařízení, ale můžeme aplikaci spustit na virtuálním zařízení přímo v počítači.
- [JDK - Java Development Kit](#)<sup>19</sup>  
Nativní aplikace pro Android jsou psány v programovacím jazyce Java, proto je nutné kromě samotného Android Studia ještě nainstalovat JDK, což je produkt od společnosti Oracle, který obsahuje soubor základních nástrojů pro platformu Java.
- [Visual Studio Code](#)<sup>20</sup>  
Jak již bylo zmíněno výše, k tvorbě kódu je nutný některý z textových editorů, kterých existuje opravdu velká spousta a je pouze na uživateli, pro který se rozhodne. VS Code však obsahuje všechnu potřebnou funkcionalitu k psaní kódu, včetně implementované terminálu. Tento program také umožňuje instalaci dalších rozšíření, která například správně formátují náš napsaný kód a mnoho dalších funkcionalit příměňujících práci.

## 8.2 Založení projektu

Poté co máme nainstalované všechny předešlé programy můžeme konečně přistoupit k instalaci samotné Cordovy, ta je součástí balíčkovacího systému NPM. Instalaci je nutné provést v příkazovém řádku<sup>21</sup> vepsáním následujícího příkazu:

```
C:\> npm install -g cordova
```

Parametr `-g` označuje globální instalaci programu AC, tedy že ho lze využít kdekoliv v počítači a nejen v aktuální složce. Pokud si chceme být jistí, že máme program nainstalovaný, spustíme příkaz :

```
C:\>cordova -version
9.0.0
```

Odpovědí by měla být aktuální verze programu, pokud se tak nestane, instalace neproběhla v pořádku. Po úspěšné instalaci můžeme vytvořit nový projekt. Nejdříve se přesuneme do složky, ve které chceme soubory projektu ukládat a následně spustíme příkaz `create` s následujícími parametry :

```
C:\Users\Steve\Desktop>cordova create dp cz.test.dp DPTest
```

<sup>17</sup><https://git-scm.com/>

<sup>18</sup><https://developer.android.com/studio/index.html>

<sup>19</sup><https://www.oracle.com/technetwork/java/javase/downloads/index.html>

<sup>20</sup><https://code.visualstudio.com/>

<sup>21</sup>Příkazový řádek se spouští pomocí klávesové zkratky **Win + R**, čímž se spustí nabídka pro spuštění, do které se vypíše název programu `cmd`.





Prvním parametrem „dp“ je název hlavního adresáře projektu, do kterého se vkládají všechny další soubory. Druhý a třetí parametr je nepovinný a označuje identifikátor a jméno aplikace.

Tímto byl založen projekt, jehož základní adresářová struktura (viz obr. 8.3) obsahuje následující prvky[7]:

- **hooks** [Složka] -  
Cordova Hooks jsou speciální skripty, které mohou být přidány vývojářem některého z rozšíření nebo systémem pro sestavení aplikace za účelem přizpůsobení Cordova příkazů. Pokud bychom například používali nějaký nástroj pro kontrolu formátování kódu, mohli bychom vytvořit tzv. „before\_build hook“, který by tento nástroj před sestavením aplikace spustil.
- **platforms** [složka] -  
Tato složka obsahuje podadresáře pojmenované podle jednotlivých platforem, pro které je aplikace určena. V každém podadresáři je vygenerována spustitelná nativní aplikace.
- **plugins** [složka] -  
Obsahuje zdrojové kódy všech nainstalovaných zásuvných modulů pro každou platformu.
- **www** [složka] -  
Tato složka je z pohledu programování nejzajímavější, obsahuje veškeré soubory webové aplikace, které následně vyplní instanci „WebView“. Je zde umístěn soubor `index.html`, který se ve výchozím nastavení zobrazí při spuštění aplikace.
- **config.xml** -  
Soubor `config.xml` je globální konfigurační soubor, který definuje mnoho aspektů a chování Cordova aplikací. V tomto souboru se definují použitá rozšíření, ale také třeba povolení přístupu aplikacím k hardwarovým prostředkům mobilních zařízení jako je například přístup k uložišti.

Název	Přípona	Velikost	Datum
[..]	<DIR>		17.04.2019 19:40
[hooks]	<DIR>		17.04.2019 19:40
[platforms]	<DIR>		17.04.2019 19:40
[plugins]	<DIR>		17.04.2019 19:40
[www]	<DIR>		17.04.2019 19:40
package	json	408	17.04.2019 19:40
config	xml	973	17.04.2019 19:40

Název	Přípona	Velikost	Datum
[..]	<DIR>		17.04.2019 19:40
[css]	<DIR>		17.04.2019 19:40
[img]	<DIR>		17.04.2019 19:40
[js]	<DIR>		17.04.2019 19:40
index	html	2 481	26.10.1985 10:15

Obrázek 8.3: Adresářová struktura projektu

Po založení projektu jsou složky `platforms` a `plugins` prázdné. Je zapotřebí definovat cílovou platformu aplikace a názvy potřebných pluginů. Nejprve se však musíme přemístit do nově vytvořeného podadresáře, ve kterém je umístěný náš projekt.

```
C:\Users\Steve\Desktop>cd dp
```



### 8.3 Přidání platformy a pluginů

Definice konkrétní platformy mobilního zařízení se pak provede následujícím způsobem:

```
C:\Users\Steve\Desktop\dp>cordova platform add android
```

K úspěšnému přidání je nutné mít nainstalované vývojové nástroje pro tvorbu projektu dané platformy. Přidávání nativních funkcí se provádí příkazem `plugin add` s parametrem označujícím název pluginu, popřípadě cestou k umístění zdrojových kódů, které mohou být uloženy na našem počítači nebo na vzdáleném Git repositáři.

```
C:\Users\Steve\Desktop\dp>cordova plugin add cordova-plugin-camera
```

### 8.4 Spuštění aplikace na zařízení

Jak již bylo řečeno výše, aplikaci můžeme testovat buď na emulátoru dané platformy, nebo fyzickém zařízení připojeném k počítači. Emulátor poskytuje téměř všechny funkce jako fyzické zařízení a nabízí nám možnost konfigurace, například velikosti displeje, paměti RAM a podobně. Jeho nevýhoda spočívá v tom, že je pomalý i na relativně výkonném počítači. Pro spuštění aplikace na fyzickém zařízení je nutné zapnout tzv. Vývojářský režim<sup>22</sup>, který nám v telefonu zpřístupní další možnosti nastavení telefonu. V této nabídce je pak nutné povolit „Ladění USB“, to nám umožní kopírování dat mezi počítačem a zařízením a následnou instalaci aplikace. Sestavení aplikace a její spuštění se provede takto :

```
C:\Users\Steve\Desktop\dp>cordova build
C:\Users\Steve\Desktop\dp>cordova run
```

### 8.5 Programování aplikace

Programování aplikace probíhá ve složce `www`, kam se nahrávají veškeré soubory, které se následně sdílí mezi jednotlivými platformami. Níže je umístěn ukázkový Javascriptový kód, kterým by měla začínat každá aplikace. Jedná se o konstruktor aplikace, který vytvoří event (událost) `deviceready`. Tato událost čeká na načtení všech zdrojů aplikace a až teprve poté spustí callback funkci, v tomto případě funkci `onDeviceReady`. Tento postup zamezí tomu, abychom se nedotazovali na nějaký nativní kód, který není doposud načtený, což by způsobilo pád aplikace.

```
1 var app = {
2   // Konstruktor aplikace
3   initialize: function() {
4     // Vytvoreni eventu cekajiciho na nacteni vsech zdroju
5     document.addEventListener("deviceready", app.onDeviceReady)
6   },
7   // Funkce, ktera se spusti pote co je zarizeni pripraveno
8   onDeviceReady: function() {
9     console.log("Spusteni probehlo v poradku!")
10  }
11 }
12 app.initialize()
```

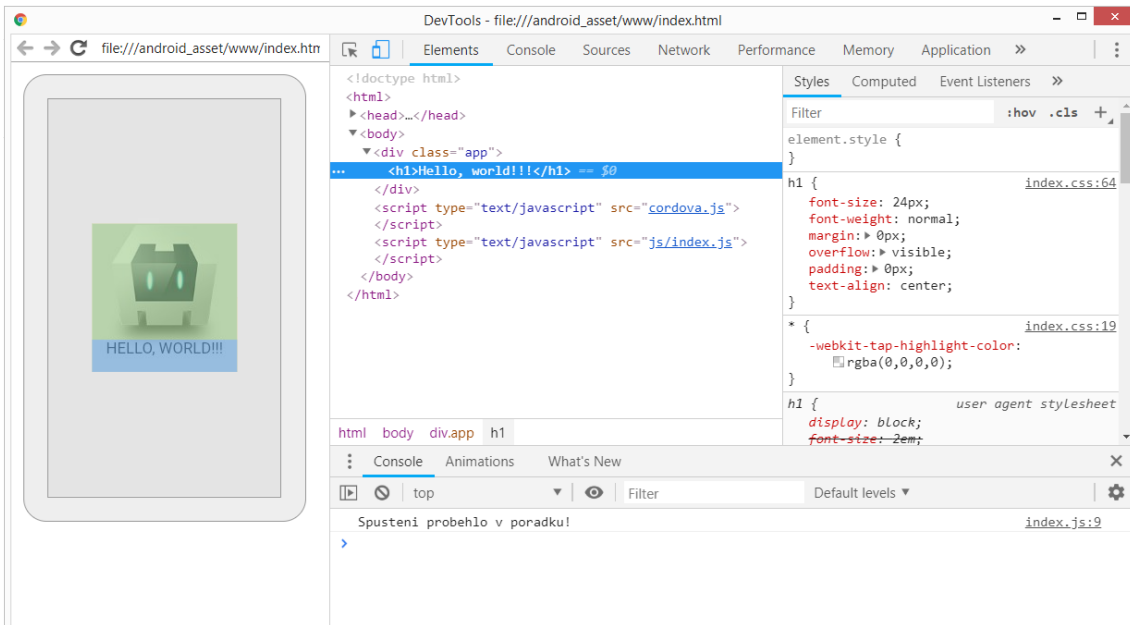
**Zdrojový kód 1:** Ukázka z `index.js`

<sup>22</sup>Vývojářské možnosti se u Android zařízení povolí v nabídce `Nastavení` → `0 zařízení` a tam 7x klepneme na číslo sestavení zařízení.



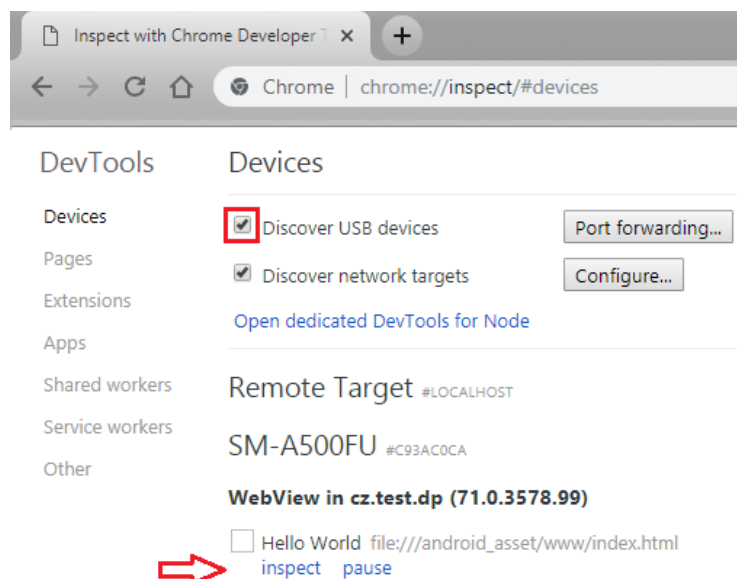
## 8.6 Ladění aplikace

Při vývoji jakkoliv rozsáhlé aplikace dochází k překlepům, chybám při výpočtech a dalším problémům, které je nutné před vypuštěním aplikace nalézt a odstranit. K usnadnění této práce lze využít inspekční nástroj v prohlížeči Google Chrome (viz obr. 8.4), který umožňuje zobrazení ladících zpráv v konzolovém terminálu, sledování hodnot proměnných nebo dokonce úpravu stylů CSS stylů a HTML souborů za běhu aplikace.



Obrázek 8.4: Vývojový nástroj Google Chrome

Spuštění tohoto nástroje se provádí vložením odkazu `chrome://inspect/#devices` do navigačního panelu prohlížeče (viz obr. 8.5). Poté se otevře karta s nástrojem pro ladění webu, zde je nutné zkontrolovat potvrzení volby „Discover USB devices“ a pokud máme zařízení připojené k počítači a spuštěnou aplikaci, zobrazí se nám mezi vzdálenými zařízeními [14].



Obrázek 8.5: Přístup k inspekčnímu nástroji



## 9 Aplikace GNSS Kontrolér



Obrázek 9.1: Ikona aplikace GNSS Kontrolér

V této kapitole je popsána struktura a jednotlivé funkce vytvořené aplikace, která byla pojmenována jako **GNSS Kontrolér** (dále jen GK). Jejím hlavním úkolem je zobrazení, vizualizace a ukládání dat, které vysílá GNSS přijímač. Spojení přijímače a aplikace (respektive mobilního telefonu) probíhá prostřednictvím bezdrátové komunikace Bluetooth (dále jen BT).

Pro zpřístupnění této a dalších nativních funkcí telefonu bylo nezbytné nainstalovat pluginy, přičemž soupis těch nejdůležitějších je uveden zde:

- **cordova-plugin-bluetooth-serial**
- **cordova-plugin-file**
- **cordova-plugin-network-information**
- **cordova-plugin-local-notification**
- **cordova-plugin-simple-toast**
- **cordova-plugin-vibration**

Formát zpráv vysílaných přijímačem vychází z komunikačního standardu NMEA (viz kapitola 6.1) a formátu UBX. Aplikace jednotlivé NMEA zprávy dekóduje a vybírá z nich potřebné informace, které jsou pak vhodně zobrazeny v jednotlivých funkčních oknech. Informace z UBX zpráv nejsou v aplikaci nijak zpracovávány, ale jsou ukládány do vytvořeného *\*.ubx* souboru, pokud uživatel spustí funkci ukládání surových dat. Z dat uložených v tomto souboru lze následně vytvořit RINEXový soubor, díky kterému můžeme zpracovat měření až v postprocesním režimu.

Další důležitou částí aplikace je funkcionality tzv. NTRIP Klienta, která uživateli poskytuje možnost přihlášení k poskytovateli korekcí skrze internetové připojení a následně tyto korekce zasílat do GNSS přijímače. Tato funkce má zcela zásadní vliv na přesnost určení polohy, jelikož můžeme využívat metod diferenčního měření DGNSS (modul NEO-M8T) a RTK (modul NEO-M8P). Popis použitých GNSS modulů je uveden v kapitole 11.3.

### 9.1 Ukládání informací

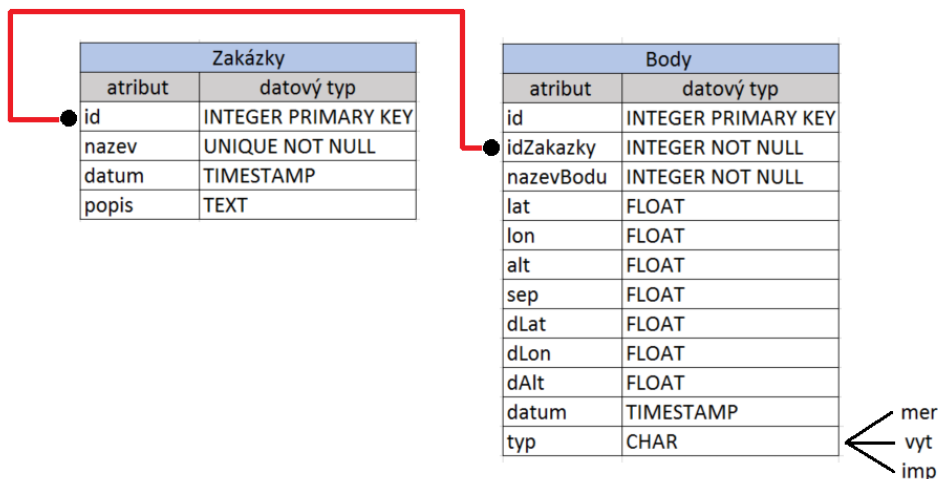
Informace poskytované přijímačem lze v aplikaci přímo ukládat do lokální webové SQL<sup>23</sup> databáze, jejíž struktura je velmi jednoduchá a skládá se pouze ze 2 tabulek. Jejich

---

<sup>23</sup>SQL (Structured Query Language) je standardizovaný jazyk pro ukládání, manipulaci a získávání dat z databáze.



propojení a struktura je zobrazena na obrázku 9.2. Uživatel musí v aplikaci vytvořit zakázku, jejímž jediným povinným parametrem je její název. Ten musí být zvolen tak, aby byl v rámci celé databáze jedinečný. Propojení tabulek je uskutečněno pomocí identifikátoru zakázky `id` (znázorněno červeným propojením). Do tabulky `body` jsou ukládány informace o poloze jednotlivých měření, vytyčených popřípadě importovaných bodů a to v souřadném systému WGS-84.



Obrázek 9.2: Struktura SQL databáze

Popis zaznamenaných parametrů v tabulce `body`:

- `lat` – zeměpisná šířka
- `lon` – zeměpisná délka
- `alt` – výška přijímače vůči střední hladině moře (geoidu<sup>24</sup>)
- `sep` – výškový rozdíl mezi geoidem a elipsoidem systému WGS-84
- `datum` – čas uložení bodu do databáze
- `typ` – typ uloženého bodu

Následující atributy mají jiný význam v závislosti na typu bodu :

- `dLat`      **mer** : maximální odchylka od průměru ze zaznamenaných dat
- `dLon`    → **vyt** : rozdíl mezi vytyčovanou a aktuální pozicí
- `dAlt`      **imp** : u importovaných bodů jsou tyto položky prázdné

Dále je možné ukládat „surová“ (z angl. raw) data, která obsahují informace o poloze družic a měřená data (pseudovzdálenosti, fázová měření atd.) potřebná pro vytvoření RINEX souborů. V adresáři pojmenovaném podle aktuální složky se vytvoří soubor<sup>25</sup>, do kterého jsou tato data ukládána. Zápis dat není kontinuální, ale dochází k němu přibližně každých deset vteřin. Tento postup byl zvolen proto, aby při pádu aplikace nedošlo ke ztrátě celého měření.

Data uložená v souboru „ubx“ je možné zobrazit v aplikaci u-Center, kde lze přehrát jednotlivé záznamy v časové posloupnosti nahrávání. Další možností je využití programu `CONVBIN.exe` z knihovny RTKlib [34], pomocí kterého lze tento soubor převést do formátu RINEX.

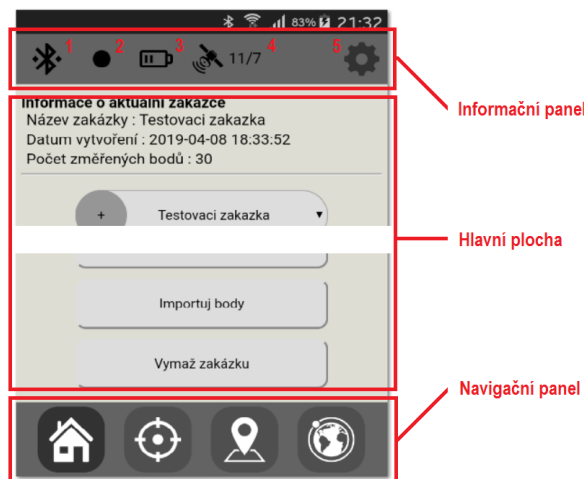
<sup>24</sup>Parametry modelu geoidu lze definovat v programu u-center, defaultně je nastaven model EGM96.

<sup>25</sup>Pojmenování souboru nezadávat uživatel, ale vychází z aktuálního data, času a jména bodu. Výsledný název má pak tuto strukturu - 190508\_220315\_4001.ubx



## 9.2 Struktura plochy aplikace

Vizuální vzhled aplikace byl navržen tak, aby byl co nejintuitivnější a uživatel i při prvním spuštění neměl problém s jejím ovládáním. Struktura obrazovky je rozdělena do 3 dílčích částí (viz obr.9.3) :



Obrázek 9.3: Struktura okna aplikace

### • Informační panel -

Zobrazuje nejdůležitější informace o připojeném GNSS přijímači. Uživatelé jsou zde předávány informace o stavu připojení s přijímačem, počtu satelitů či kvalitě určování polohy, ať se nachází v jakémkoliv okně aplikace.

1. Stav BT připojení - informuje uživatele pomocí grafických ikon, zda je přijímač k telefonu připojený, odpojený či dochází k vyhledávání dostupných zařízení.
2. Indikátor kvality - podle barvy tohoto objektu je uživatel informován o kvalitě určení pozice přijímače a typu využívaných korekcí. Inicializace nebo ztráta FIX řešení je také signalizována zvukovým znamením.
  - - Kódové měření
  - - Měření s využitím korekcí pro kódová měření (DGNSS)
  - - RTK (float), určení počtu vlnových délek jako neceločíselnou hodnotu
  - - RTK (fix), vyřešení počáteční fázové ambiguity s celočíselným výsledkem
3. Zbývající kapacita baterie přijímače.
4. Počet družic, jejichž signály jsou využity k výpočtu polohy. První číslo představuje počet GPS družic a druhé počet družic systému GLONASS .
5. Tlačítko pro přechod do nastavení aplikace.

### • Navigační panel -

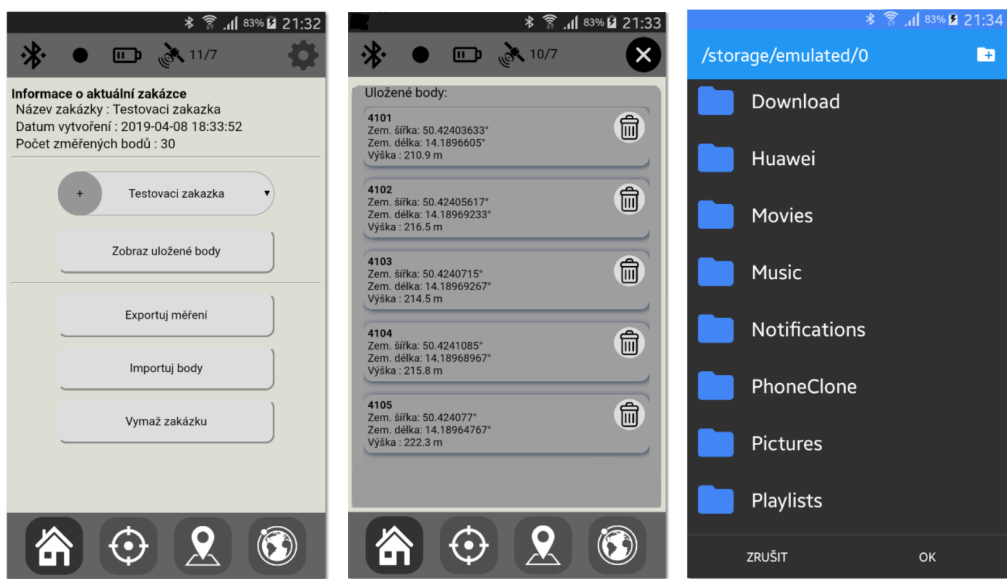
Tento panel obsahuje navigační tlačítka, díky kterým může uživatel přecházet mezi jednotlivými funkčními okny aplikace, jako domovskou obrazovku, měření bodů, vytyčením bodů a zobrazení viditelných družic. Podrobný popis jednotlivých oken aplikace a jejich funkcí je uveden níže.

### • Hlavní plocha -

V této části se zobrazují informace v závislosti na zvoleném funkčním okně.



### 9.3 Okno - Domovská obrazovka



Obrázek 9.4: Domovská obrazovka

Po spuštění aplikace se uživateli zobrazí domovská obrazovka (viz obr. 9.4 vlevo). V horní části hlavní plochy jsou vypsané informace o aktuálně zvolené zakázce - název, datum vytvoření a počet změřených bodů. V dalším oddíle hlavní plochy můžeme vytvořit novou zakázku kliknutím na šedé tlačítko + a v nově zobrazeném okně zvolit název zakázky, datum založení, popřípadě její popis.

Po úspěšném vytvoření se zakázka nastaví jako výchozí a nově měřené/importované body budou ukládány do ní. Seznam všech bodů, včetně výpisu jejich souřadnic, lze vyvolat tlačítkem **Zobraz uložené body** (viz obr. 9.4 uprostřed). U každého bodu tohoto výpisu se nachází tlačítko pro vymazání bodu z databáze.

Další funkce domovské obrazovky :

- **Exportuj měření-**

Po zvolení této funkce je vytvořen nový adresář s názvem aktuální zakázky ve složce GNNS\_Kontroler na lokálním úložišti mobilního telefonu. V tomto adresáři se pak vytvoří soubor, který obsahuje informace o aktuální zakázce a výpis uložených informací o všech jejích bodech.

- **Importuj body-**

Souřadnice podrobných bodů je možné naimportovat stiskem tohoto tlačítka. Uživateli se zobrazí adresářový průzkumník (viz obr. 9.4 vpravo), v němž si zvolí textový soubor se seznamem bodů, který musí mít následující strukturu :

```
<cislo_bodu>,<zem_sirka>,<zem_delka>,<vyska><CR><LF>
```

např. : 4001,50.42404339,14.18559139,217.333

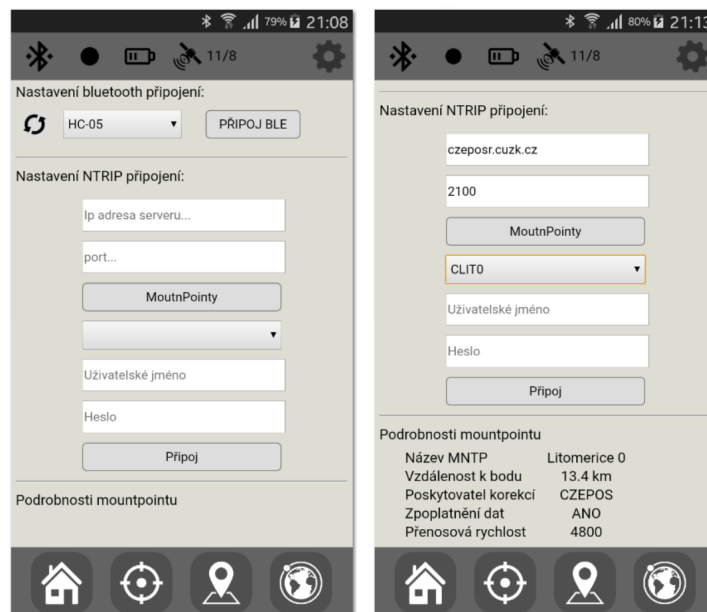
Díky tomu je možné do aplikace nahrát podrobné body, které lze následně vytyčit.

- **Vymaž zakázku-**

Stiskem tohoto tlačítka se nejprve zobrazí varovné hlášení, zda chce uživatel opravdu vymazat zakázku a všechny uložené informace z lokální SQL databáze. Po potvrzení této volby dojde k odstranění zakázky.



## 9.4 Okno - Nastavení



Obrázek 9.5: Aplikace - okno Nastavení

Při prvním spuštění aplikace je nutné definovat BT zařízení, ke kterému se má aplikace připojit. Zařízení je nejprve nutné s telefonem spárovat, to se provádí v nastavení telefonu v záložce Bluetooth. Spárovaná zařízení se následně zobrazí v roletové nabídce v horní části hlavního okna. Volbu konkrétního zařízení potvrdíme stiskem tlačítka **Připoj BT**. O úspěchu či selhání pokusu o připojení je uživatel informován vyskakovací zprávou, popřípadě změnou ikony v informačním panelu. Při dalším spuštění aplikace již dochází k automatickému připojení podle předchozích nastavených parametrů.

Další část tohoto okna obsahuje vstupy pro nastavení NTRIP připojení. Uživatel musí nejprve vyplnit adresu a port zvoleného poskytovatele korekcí a poté zaslat tlačítkem **MountPointy** žádost o poskytnutí Source table (zdrojové tabulky). Zkratky všech dostupných moutpointů se umístí do roletové nabídky. Podrobné informace o zvoleném zdroji korekcí se zobrazí v dolní části okna (viz obr. 9.5 vpravo).

Vzdálenost zdroje korekcí od našeho přijímače má zásadní vliv na výslednou přesnost určení polohy (viz kap. 5), z tohoto důvodu je tato vzdálenost součástí podrobného výpisu. Uživatel si pak díky tomu může vybrat vhodnou nejbližší ref. stanici, popřípadě zvolit službu virtuální stanice. Aplikace tuto volbu kontroluje a v případě, že se jedná o virtuální stanici, zašle spolu s požadavkem o poskytování korekcí také aktuální polohu v podobě poslední GGA zprávy.

K získávání korekcí je samozřejmě nutné internetové připojení. Pokud není přístup internetu k dispozici, je uživatel informován varovnou hláškou, že tuto funkci nemůže využívat.





## 9.5 Okno - Měření



Obrázek 9.6: Aplikace - okno Měření

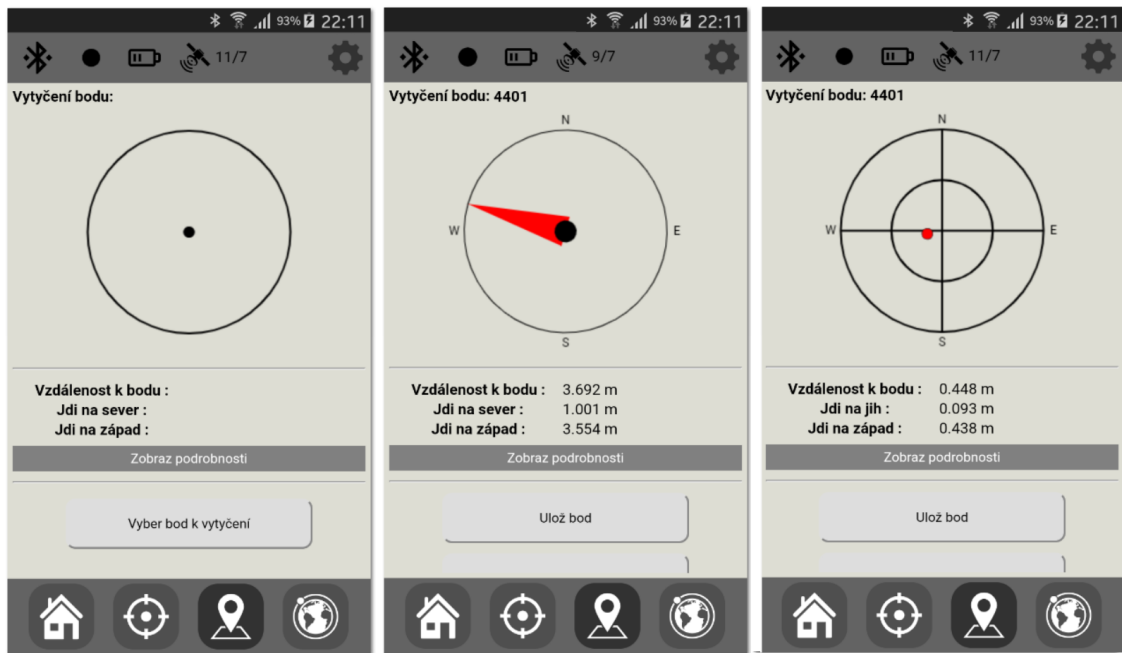
V této části aplikace se zobrazují informace o poloze, které posílá GNSS přijímač a to v podobě zeměpisné šířky, délky a nadmořské výšky v souřadném systému WGS-84. Informace jsou aktualizovány po jedné vteřině. Kromě toho je u každé z těchto hodnot v závorce vypsána jejich směrodatná odchylka a dále také parametr přesnosti PDOP. Všechny tyto informace poskytuje přijímač v podobě NMEA zpráv.

Dále jsou zde umístěna dvě vstupní pole pro zadání informací o výšce antény a názvu měřeného bodu, bez kterých není možné polohu bodu do databáze uložit. Měření se spouští stiskem tlačítka **MĚR**, poté je každou vteřinu ukládána poloha přijímače do pomocného objektu. Po ukončení měření se výsledná poloha bodu určí jako aritmetický průměr ze všech epoch měření a následně se uloží do lokální databáze. Uživatel také může zadat délku měření bodu a po uplynutí této doby se změní barva tlačítka pro uložení nebo v případě, že je aplikace při měření minimalizována, obdrží uživatel notifikační zprávu o dosažení nastaveného času měření.

Ve spodní části hlavní plochy se nachází tlačítko **Loguj Raw data**, při jeho stisku se spustí měření, stejně jako v předchozím případě, ale současně se začnou všechna vysílaná data nahrávat do souboru (viz kap. 9.1). Při měření polohy bodu nebo zápisu surových dat je možné toto funkční okno opustit, aniž by došlo k přerušování zápisu. Uživatel tedy může aplikaci minimalizovat nebo přecházet do ostatních funkčních oken aplikace.

## 9.6 Okno - Vytyčení

Tato část aplikace, jak vyplývá z jejího názvu, poskytuje uživateli možnost vytyčení polohy podrobných bodů, buďto těch dříve změřených, nebo importovaných z textového



Obrázek 9.7: Aplikace - okno Vytyčení

souboru. Uživatel musí nejprve vybrat bod k vytyčení ze seznamu a poté je již k němu navigován pomocí vzdáleností vztažených k světovým stranám.

Pokud se nacházíme dál než 3 metry od vytyčovaného bodu, navigační okno zobrazí směr k bodu v podobě směrovky (viz obr. 9.7 uprostřed), která zobrazuje azimut spojnice aktuální polohy přijímače a vytyčovaného bodu.

Po přiblížení k bodu se grafické okno změní na vytyčovací kříž, který usnadní nalezení vytyčované pozice. Střed kříže zastupuje polohu přijímače a červený kroužek polohu vytyčovaného bodu (viz obr. 9.7 vpravo). Po rozkliknutí nabídky **Zobraz podrobnosti** jsou uživateli zobrazeny informace o převýšení<sup>26</sup> mezi měřeným a vytyčovaným bodem a také hodnoty aktuální horizontální a vertikální přesnosti určení polohy přijímače. Finální vytyčenou pozici bodu lze do databáze uložit stiskem tlačítka **Ulož bod**.

## 9.7 Okno - Skyplot

V posledním okně aplikace je zobrazen graf aktuálně viditelných satelitů, který se obecně označuje jako tzv. *Skyplot*. Poloha jednotlivých satelitů je získávána z NMEA zpráv GSV, kde je uveden jejich azimut a elevace. Satelity GPS jsou znázorněny čtvercem a GLONASS satelity kolečky, dále zde dochází k barevnému rozlišení. Signál ze satelitů označených zelenou barvou se využívá k výpočtu polohy, zato červeně označených nikoliv.

<sup>26</sup> Výšku antény přijímače nelze v tomto okně konfigurovat a používá se poslední uložená hodnota při měření bodu. K její změně je tedy nutné přejít do okna měření.



Obrázek 9.8: Aplikace - okno Skyplot

Důvodem k tomu, že některé satelity nejsou k určení polohy využity, může být velký útlum signálu, jejich zdravotní stav nebo například pozice satelitu pod nastavenou hodnotou elevační masky. Přetáhnutím grafu směrem vzhůru zobrazíme podrobné informace o jednotlivých satelitech v textové podobě (viz obr. 9.8 vpravo).



## Část III

# Hardware a software

V poslední části této diplomové práce budou představena všechna nezbytná zařízení pro sestavení GNSS přijímače. Budou také popsány jejich parametry, způsob zapojení a následně sepsány programy využívané pro jejich konfiguraci.

Dále zde bude popsán způsob testování vytvořeného zařízení, jeho dosažené výsledky a také návrhy pro možná budoucí vylepšení celého projektu. A protože se jedná o „nízkonákladový“ přijímač, jak deklaruje název, je v úplném závěru diplomové práce uveden seznam použitých částí (součástí) spolu s jejich cenovou kalkulací.

## 10 Části GNNS přijímače

Každé geodetické GNSS zařízení se skládá z těchto částí :

- **GNSS modul/čip**

Ať tuto část pojmenujeme jakkoliv, je jasné, že se jedná o klíčovou součást celého zařízení. Čip řídí celý systém přijímače, kdy z přijímaných digitalizovaných signálů získává potřebné informace pro vyřešení navigační úlohy - určení měřených pseudovzdáleností, fázové měření, efemeridy družic apod. Výstupem jsou pak informace o poloze v definovaném souřadném systému včetně parametrů přesnosti a dalších informací (např. času), které jsou dále předávány uživateli prostřednictvím kontrolního zařízení či ukládány na paměťový disk. V literatuře je tato část přijímače dále rozdělena na radiofrekvenční jednotku a mikroprocesor [6].

- **Anténa**

Hlavním účelem antény je přeměna elektromagnetických signálů vysílaných družicemi na elektrické impulsy, které mohou být následně zpracovány. Typ a konstrukce antény má zcela zásadní vliv na přesnost měření, zejména pak velikost variace fázového centra, dále pak filtrace a stínění odražených signálů (viz kap. 4). Přijímaný signál je velmi slabý, proto jsou antény ještě doplňovány předzesilovačem.

- **Zdroj napětí**

Napětí v elektrickém obvodu bývá převážně zajištěno dobíjecími bateriemi, nejčastěji typu Li-Ion (Lithium-iontové).

- **Kontrolní zařízení**

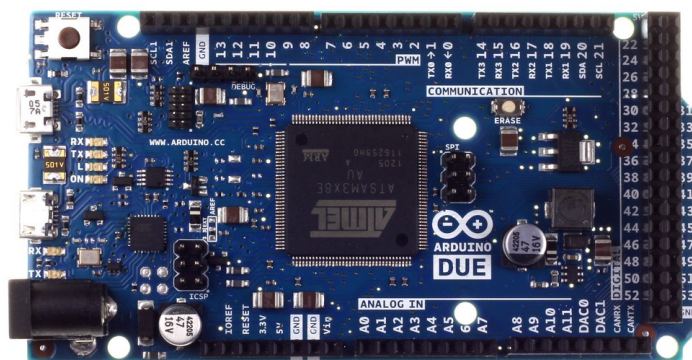
Většina geodetických zařízení pro GNSS měření obsahuje kromě samotného přijímače ještě kontrolér, který zajišťuje komunikaci a vizualizaci měřených dat. Uživatel má díky tomu možnost zadávat další popisné informace (číslo měřeného bodu, výšku antény apod.) či konfigurovat nastavení přijímače. Funkci kontroléru pro GNSS zařízení vyvíjené v rámci této diplomové práce zastoupí mobilní aplikace, která byla popsána v předchozí části tohoto textu.



## 11 Použitá zařízení a princip jejich propojení

### 11.1 Arduino Due

Středobodem celého projektu je vývojová deska Arduino Due (viz obr. 11.1), pomocí které se propojí ostatní části GNSS přijímače. Tato vývojová deska je založena na výkonném procesoru Atmel SAM3EX8E ARM Cortex-M3 s 32-bitovou architekturou, což umožňuje zapojení této desky do výpočetně náročných projektů. Rozdíl oproti dalším Arduino mikrokontrolérům je také v tom, že běží na 3,3V místo častěji používaných 5V [10]. Rozložení jednotlivých pinů má zcela totožné s deskou Arduino MEGA, která byla v projektu původně zamýšlena, hlavně z finančních důvodů.



Obrázek 11.1: Vývojová deska Arduino Due

Přestože v tomto projektu není nutný takto výkonný mikrokontrolér, byl zvolen především kvůli tomu, že obsahuje 4 UART sériové linky a také druhý mikro-USB port podporující OTG. Cena originálu této desky se v ČR pohybuje kolem 1100 Kč, proto byl pořízen její klon z čínského e-shopu za 13\$ (v přepočtu přibližně 300 Kč).

Tento cenový rozdíl je až zarážející avšak použití klonu této desky, se neobešlo bez drobných obtíží. U Arduino desek obecně platí, že při připojení napájení se automaticky spustí kód nahraný v desce. U této konkrétní desky však občas docházelo k tomu, že kód spuštěn nebyl a pro jeho správný start bylo nutné stisknout tlačítko **RESET**. Průzkum internetu poukázal na to, že se nejedná o ojedinělý případ a tento problém řešilo už více uživatelů. Popis příčiny problému, včetně postupu jeho odstranění pomocí připájení 10K odporu, je uveden v následujícím článku [16].

Krom této drobné závady pracovala deska bez problému a byla použita k vytvoření komunikačního uzlu sériové komunikace UART jednotlivých zařízení (modulů).

#### 11.1.1 UART

Zkratka UART by mohla být volně přeložena jako *univerzální asynchronní přijímač/vysílač*. Jedná se tedy o hardwarový princip komunikace mezi jednotlivými zařízeními. Asynchronní je označován kvůli tomu, že vysílač i přijímač mají vlastní generátor hodinového signálu<sup>27</sup>. Výhoda této UART sériové komunikace je v tom, že k jejímu uskutečnění jsou potřebné pouze 3 dráty, s kterými je nutné propojit zem, vysílačí

<sup>27</sup>Existuje také synchronní způsob komunikace, který se označuje jako USART.



pin (TX-*Transmit*) prvního zařízení s přijímacím pinem (RX-*Read*) druhého zařízení a naopak. Jelikož se jedná o asynchronní komunikaci, je nutné definovat počáteční a koncové bity, podle nichž zařízení pozná, kdy má začít přijímat data a kdy dojde k jejich ukončení. Dalším parametrem je komunikační (přenosová) rychlost, která musí být na obou zařízeních nastavena stejně, jinak by nebyla schopna zaslaná data správně přijmout [39].

V níže uvedeném zdrojovém kódu, je ukázáno vytvoření několika sériových spojení mezi Arduino deskou - BT modulem, GNSS deskou a také komunikační port skrze druhé nativní micro-USB, díky kterému mohou být data kabelem přenášena do počítače či další řídicí desky.

```
1 | byte prichozByte = 0;
2 |
3 | void setup() {
4 |     Serial1.begin(19200);    // Bluetooth
5 |     Serial2.begin(19200);    // GNSS
6 |     SerialUSB.begin(19200); // Native USB
7 | }
8 |
9 | void loop() {
10 |    // Pokud jsou dostupna data z telefonu, poslou se do GNSS
11 |    // prijimace.
12 |    if(Serial1.available()>0){
13 |        Serial2.write(Serial1.read());
14 |    }
15 |
16 |    // Pokud jsou dostupna data zaslana skrze USB, poslou se do
17 |    // GNSS prijimace. Diky tomu je mozne zasilat korekce skrze
18 |    // USB, pripadne propojeni s programem U-center.
19 |    if(SerialUSB.available()>0){
20 |        Serial2.write(SerialUSB.read());
21 |    }
22 |
23 |    // Pokud jsou dostupna data z GNSS prijimace, poslou se do
24 |    // telefonu a zaroven do zarizeni, ktere je pripojeno USB
25 |    // kabelem.
26 |    if(Serial2.available()>0){
27 |        prichozByte = Serial2.read();
28 |        Serial1.write(prichozByte);
29 |        SerialUSB.write(prichozByte);
30 |    }
31 | }
```

**Zdrojový kód 2:** *arduinoUART.ino*

Program pro Arduino mikrokontroléry je možné psát v jazycích C nebo C++, ale nejrozšířenějším způsobem je však využití knihovny C++ *wiring*. Tato knihovna obsahuje všechny potřebné nástroje pro tvorbu kódů a díky její komplexnosti je někdy označována jako samostatný programovací jazyk [11].

Program se obecně dělí na několik částí, na úplném začátku se inicializují proměnné, které se používají v celém programu. Dále je zde funkce `setup()`. Ta se spouští pouze jednou při startu programu, popřípadě při resetování desky. Do této funkce se tedy vkládá

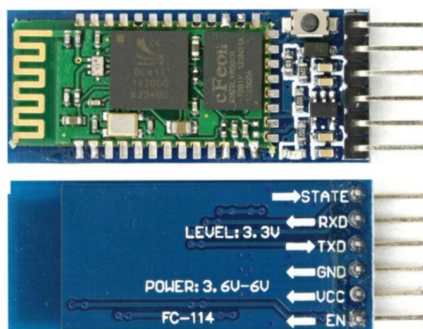


kód, který by měl proběhnout pouze jednou - inicializace proměnných, volání knihovnic funkcí, popřípadě nastavení režimu pinu (vstupní/výstupní). V našem případě zde dochází k spuštění několika linek sériové komunikace zápisem příkazu `SerialX.begin(Y)`. Kde hodnota **X** definuje piny sériové komunikace a **Y** komunikační rychlost. Poslední nutnou částí každého programu je funkce `loop()` (v překladu smyčka). Do této funkce se umísťuje kód, který je spouštěn stále dokola a zajišťuje ovládání našeho programu. V této smyčce sledujeme jednotlivé sériové linky a pokud po nich přišla nějaká data, jsou přeposlány dále (viz komentáře kódu).

Takto zapsaný program je nutné přeložit (zkompilovat) do čipu srozumitelných instrukcí a ty následně nahrát do Arduino desky. K tomuto procesu lze využít standardní program **Arduino IDE** (*Integrated Development Enviromet*), což je open-source program vyvíjený společností Arduino. Dále existuje program **Fritzing**, který kromě možnosti zápisu a nahrání kódu také umožňuje schématický návrh zapojení jednotlivých elektronických součástí a následný export desky plošného spoje (DPS). Návrh DPS byl vytvořen právě v tomto programu a je součástí elektronických příloh.

## 11.2 HC-05

Komunikace mezi GNSS přijímačem a aplikací je zajištěna pomocí bluetooth modulu HC-05 (viz obr. 11.2). Jedná se o velice rozšířený modul, který vytváří bezdrátovou sériovou linku mezi zařízeními, v tomto případě mezi deskou mikrokontroléru Arduino Due a mobilním telefonem. Modul podporuje dva režimy komunikace **MASTER** a **SLAVE**. Režim **MASTER** umožňuje vyhledávání dostupných zařízení, automatické připojení k zařízení v režimu **SLAVE**, a také připojení k více zařízení najednou. Parametry modulu HC-05 jsou uvedeny v tabulce 11.1.



**Obrázek 11.2:** Bluetooth modul HC-05 - zdroj [36]

Aplikační logika čipu	3,3V
Vstupní napětí modulu	3,6 - 6V
Odběr proudu	30-40 mA
Způsob komunikace	UART (RX,TX)
Komunikační rychlosti	4800 - 1382400 bps <sup>28</sup>
Maximální dosah	přibližně 10 m
Konfigurace	pomocí AT příkazů
Specifikace BT	v.2v0+EDR

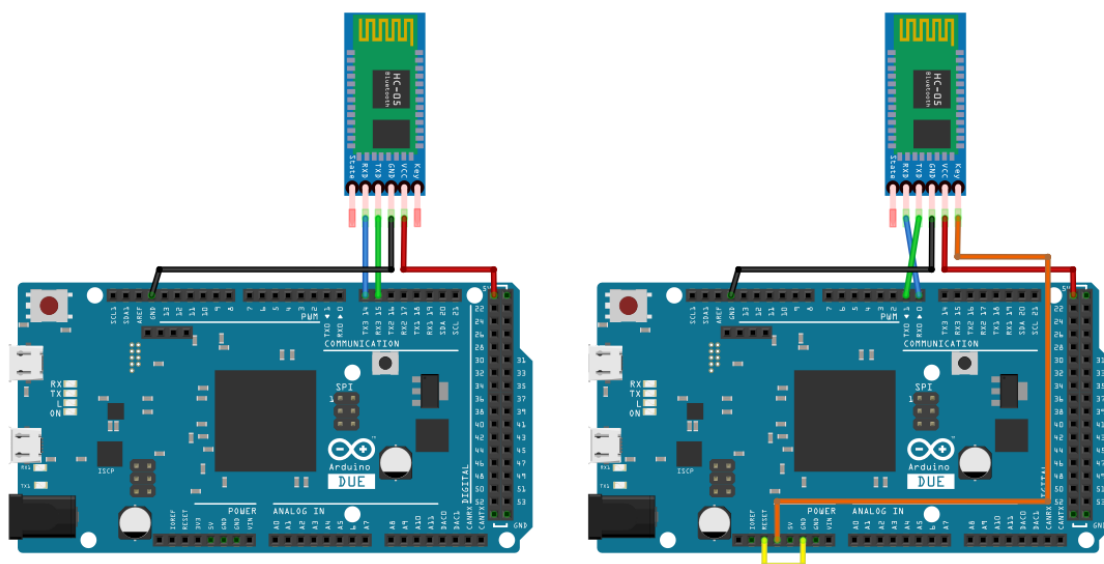
**Tabulka 11.1:** Technické specifikace modulu HC-05 - zdroj [36]



### 11.2.1 Zapojení HC-05

Připojení tohoto modulu k Arduino desce je velmi jednoduché a postačí nám k tomu pouze čtyři vodiče. Zapojení usnadňuje fakt, že logické obvody desky Arduino Due pracují s napětím 3,3V stejně jako u čipu HC-05. Pokud bychom využili jiné mikrokontroléry s 5V logikou, museli bychom napěťové úrovně převádět (více viz [36]).

Schéma zapojení je znázorněno na obrázku 11.3 vlevo. K modulu je nutné přivést zdroj napětí 5V a připojit zem, dále se vzájemně propojí porty sériové komunikace RX (*Read*) a TX (*Transmit*), které je samozřejmě nutné překřížit. Tedy vysílací pin BT desky je propojen s přijímacím pinem Arduino desky a naopak. Jak již bylo zmíněno Arduino Due má celkem 4 dvojice hardwarových sériových portů<sup>29</sup> a je pouze na nás, které z nich použijeme. Pouze musíme podle příslušného zapojení upravit řídicí kód.



Obrázek 11.3: Schéma zapojení modulu HC-05 : vysílací režim (vlevo) a konfigurační režim (vpravo)

### 11.2.2 Konfigurace modulu

Pro nastavení modulu musíme nejdříve spustit konfigurační režim, ve kterém lze nastavit chování a parametry modulu (jméno, heslo, roli atd.) pomocí tzv. AT<sup>30</sup> příkazů. Režim zapneme pokud při spuštění přivedeme na pin **EN** logickou 1 (tzn. zapojíme 3,3V). O tom, že je modul přepnut z vysílacího do konfiguračního režimu, nás informuje červená indikační dioda, která začne blikat s nižší frekvencí než obvykle.

Pro zaslání konfiguračních příkazů do modulu potřebuje použít USB/TTL převodník, díky kterému můžeme připojit BT modul k počítači a zasílat potřebné příkazy. V případě, že tento převodník nemáme, můžeme jeho funkci jednoduše nahradit vhodným propo-

<sup>28</sup>Jedná se o jednotku přenosové rychlosti (z angl. *bit per second*), tedy kolik bitů lze přenést za jednu sekundu.

<sup>29</sup>Pokud by nám pro nějaký projekt nestačily 4 hardwarové sériové linky, je možné vytvořit další softwarové komunikační porty s využitím knihovny *SoftwareSerial.h*, které však s sebou přináší určitá omezení.

<sup>30</sup>**AT** (z angl. *attention*) je řádkový příkaz, se kterým se nejčastěji setkáme při konfiguraci modemů.





jením BT modulu s naší Arduino deskou, jak je znázorněno na obrázku **11.3 vpravo**. Změnou oproti minulému zapojení je připojení 3,3V k pinu **EN**, využití portů sériové komunikace RX0 a TX0 bez jejich překřížení a nakonec uzemnění pinu **RESET** Arduino desky. Toto zapojení způsobí, že nahraný kód v desce nebude spuštěn, a pokud pošleme nějaké příkazy po sériové lince arduinu, budou rovnou přeposlány BT modulu. Nyní již můžeme spustit sériový terminál, ve kterém je nutné nastavit přenosovou rychlost 38400 bps (defaultní rychlost pro konfigurační režim) a zaslání znaků <CR><LF> (odřádkování) po každém zasláném příkazu.

Zde je soupis základních AT příkazů, jejich úplný výčet je uveden v dokumentaci[12]. Zeleně jsou uvedeny zasláné příkazy a modře přijaté odpovědi.

- Testování správného spuštění konfiguračního režimu a nastavení terminálu.  
AT  
OK
- Dotaz na pojmenování zařízení a jeho následná změna.  
AT+NAME  
+NAME:HC-05  
OK  
AT+NAME=GNSS\_Kontroler  
+NAME:GNSS\_Kontroler  
OK
- Dotaz o zaslání hesla, které je nutné pro spárování zařízení.  
AT+PSWD  
+PSWD:1234  
OK
- Požadavek o zaslání konfiguračních parametrů sériové komunikace,  
AT+UART  
+UART:9600,0,0  
OK  
kdy jednotlivá čísla v odpovědi značí přenosovou rychlost (bps), počet stop bitů a nastavení parity. Změna tohoto nastavení se provede následovně.  
AT+UART=19200,0,0  
OK
- Posledním uvedeným příkazem se dotazujeme na MAC<sup>31</sup> adresu BT modulu.  
AT+ADDR  
+ADDR:98d3:32:311c8d  
OK

### 11.3 GNSS moduly

V rámci této práce byly využity dva GNSS moduly, přičemž oba obsahují čipy od společnosti u-blox (viz kap. 12). Jedná se o moduly Neo-M8T a SparkFun GPS-RTK Board (dále jen SF deska/modul) s čipem NEO-M8P-2. S prvním uvedeným

<sup>31</sup>MAC adresa (z angl. *Media Access Control*) je identifikační označení zařízení, které by mělo být pro každé zařízení zcela jedinečné.



modulem bylo testováno vzájemné propojení a komunikace s Arduino deskou a také následné vyčítání informací ze zasílaných zpráv ve vytvořené aplikaci pro platformu Android.

Druhý zmíněný modul je zajímavý hlavně tím, že podporuje příjem a zpracování korekcí RTCM 3.x a díky tomu s ním lze provádět měření v režimu RTK. Jedná se o poměrně nový produkt společnosti ublox, s velkým potenciálem v oblasti velmi přesných GNSS aplikací, a proto je v textu věnována větší pozornost právě jemu.

Informace o použitých modulech byly přebrány z oficiální dokumentace společnosti U-blox [23][24] a z webových stránek společnosti SparkFun [37].

### 11.3.1 Ublox NEO-M8T

Jedná se o 72 kanálový jednofrekvenční přijímač, který podporuje souběžný příjem signálů z několika GNSS systémů - GPS, Galileo, GLONASS a BeiDou. Možné kombinace nastavení příjmu z jednotlivých systémů jsou znázorněny na obrázku 11.4, přičemž černá tečka představuje povolení daného systému. Modul také umožňuje příjem z podpůrných systému SBAS a QZSS, ale pouze za předpokladu, že je k navigaci povolen systém GPS.

GPS	Galileo	GLONASS	BeiDou
•	•	-	-
•	•	•	-
•	•	-	•
•	-	•	-
•	-	-	•
-	•	•	-
-	•	-	•
-	-	•	•

Obrázek 11.4: Možné kombinace GNSS systémů - [24]

Označení „T“ (*Timing*) představuje zaměření modulu převážně na synchronizaci a měření času s rozlišením zlomků mikrosekund. Výrobce udávaná přesnost určení polohy v autonomním režimu<sup>32</sup> je 2,5m. Modul také poskytuje informace o surových datech, měřených na civilní nosné vlně L1 - pseudovzdálenosti, fázová měření, dopplerovská měření a SNR.



Obrázek 11.5: GNSS modul - NEO-M8T

<sup>32</sup>Autonomním režimem je myšleno řešení polohy pomocí kódového měření bez použití korekčních dat z podpůrných systémů či permanentních stanic.

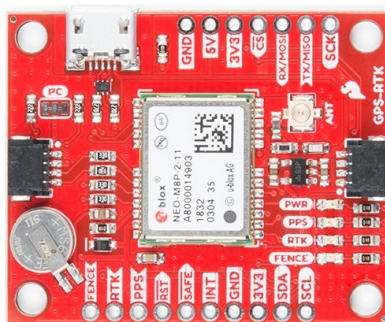


### 11.3.2 SparkFun GPS-RTK Board - NEO-M8P-2

Čip NEO-M8P je jedním z nejnovějších produktů společnosti u-blox, který kombinuje vysoký výkon produktové řady M8 s využíváním technologií RTK. Díky tomu je možné s tímto čipem dosahovat přesnosti řádově několika centimetrů. Jelikož se jedná o čip stejné produktové řady jako je výše zmiňovaný, má co se týče počtu kanálů, typu přijímaných frekvencí a výčtu možných satelitních systémů stejné vlastnosti.

Produkty M8P jsou dále členěny do dvou dalších podtypů :

- **NEO-M8P-0** – který zastupuje případ pohyblivé stanice ( „rover“ - viz kap. 5) a umožňuje příjem RTCM 3.x korekčních dat.
- **NEO-M8P-2** – tento čip umožňuje nejen funkci pohyblivé, ale také báze stanice, tedy generování korekčních dat RTCM 3.x, které je možné dále distribuovat.



**Obrázek 11.6:** SparkFun GPS-RTK Board NEO-M8P-2 – zdroj [37]

Pro využití čipu M8P-2 ve vyvíjeném zařízení byla zvolena deska SparkFun GPS-RTK Board (viz obr.11.6), která umožňuje propojení s dalšími zařízeními pomocí čtyř komunikačních linek - USB, UART, I2C a SPI.

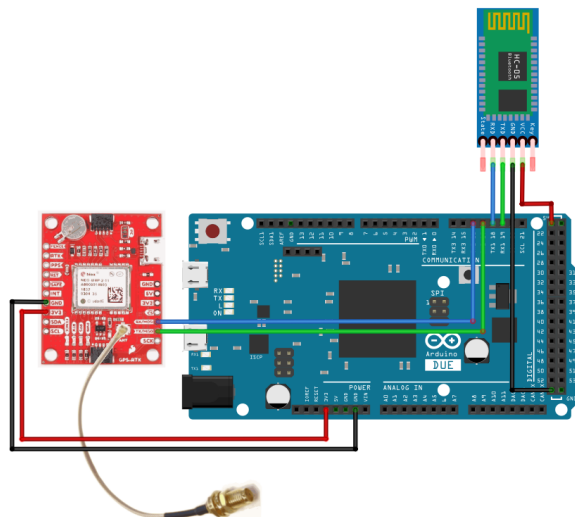
Pro připojení antény je na desce umístěn konektor typu U.FL. Deska také obsahuje záložní baterii, díky které se udržují informace o konfiguraci modulu či parametry drah satelitů i po odpojení napájení modulu. Výrobce udává, že baterie by měla tato data udržet až po dobu dvou týdnů. Na desce jsou také umístěny čtyři indikační led diody.

- **POWER** -  
Kontrolka napájení indikuje správné zapojení a přívod 3,3V k čipu.
- **PPS** -  
Dioda po úspěšné inicializaci pozice a času začne blikat s frekvencí 1Hz. (PPS - z angl. *Pulse Per Second*).
- **RTK** -  
Tato dioda po zapnutí trvale svítí, jakmile spustíme příjem RTCM dat začne blikat a pokud přejde status řešení polohy na RTK-fix, dioda zhasne.
- **FENCE** -  
Tato dioda může být konfigurována v programu u-center a využívá se k aplikacím monitorování vymezeného prostoru.



### 11.3.3 Schéma zapojení

Jak již bylo zmíněno, SF modul umožňuje sériovou komunikaci, pro kterou má vyvedené piny RX a TX. Díky tomu je zapojení k Arduino desce zcela totožně jako u BT modulu. Schéma zapojení obou dvou zařízení je znázorněno na obrázku 11.7, na kterém je také zobrazena anténní redukce z typu konektoru U.FL na SMA. Pro ucelenost celého projektu byla následně vyhotovena deska plošného spoje, která nahradila propojení jednotlivých zařízení pomocí drátů. Návrh DPS byl vytvořen v programu **Fritzing** spolu s těmito grafickými schématy zapojení.



Obrázek 11.7: Schéma zapojení modulů k Arduino desce

### 11.4 GNSS anténa

Pro příjem satelitních signálů byla zvolena levná magnetická anténa od společnosti u-blox s označením **AN-MS-0-005** (viz obr. 11.8) (dále označovaná jako u-blox anténa). Jedná se o výkonnou anténu o rozměrech 48x40x13 mm s integrovaným nízkošumovým zesilovačem, kterou lze provozovat při napětí 2,7-5,5V [9].



Obrázek 11.8: Anténa- u-blox AN-MS-0-005 - zdroj [8]

Pro porovnání výsledků u-blox antény byla dále využita profesionální geodetická anténa společnosti Leica typ AX1202 GG (viz obr. 13.3). Zhodnocení dosažených výsledků měření s využitím těchto antén je uvedeno na konci textu v kapitole 13, která popisuje postup testování přijímače.



### 11.5 Zdroj napájení

Každé elektronické zařízení potřebuje mít ke své funkci zajištěný zdroj napájení a žádnou výjimkou není ani tento GNSS přijímač. Napájení je přivedeno do Arduino desky, která ho pak rozvádí do ostatních připojených zařízení. V tomto případě byla zvolena nejjednodušší varianta napájení Arduino desky pomocí připojeného micro-USB kabelu, jehož druhý konec je zapojen do **power banky**<sup>33</sup>(viz obr. 11.9).

Tento konkrétní záložní zdroj byl zvolen kvůli velké kapacitě (12500mAh) a dvěma výstupním USB portům, díky nimž lze nabíjet nejen GNSS přijímač, ale také mobilní telefon, na kterém bude spuštěna obslužná aplikace. Dalším bonusem je existence malého displeje, na němž se zobrazuje zbývající kapacita zdroje.

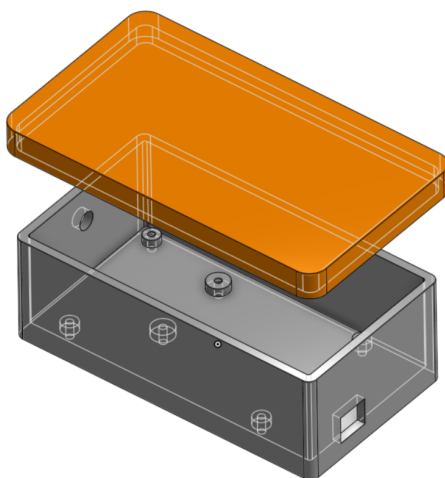


Obrázek 11.9: Zdroj napětí - power banka ADATA [30]

### 11.6 Kryt přijímače

Poslední částí GNSS přijímače je jeho kryt, který slouží jako ochrana před vnějšími vlivy, především před prachem a vlhkostí. Kryt byl vymodelován v bezplatném webovém programu **OnShape** (viz obr. 11.10) a je složen ze dvou částí. Ve spodním dílu jsou vytvořeny distanční sloupky s otvory, které slouží k upevnění Arduino desky v požadované pozici pomocí několika šroubků. Dále se zde nachází otvory pro vývod anténního SMA konektoru a vstup mikro-USB kabelu. Vrchní díl je pouze krycí.

Vytvořené modely byly následně vytištěny na autorově 3D tiskárně **Creality Ender 3**. Soubory jednotlivých částí jsou součástí elektronických příloh ve formátu *\*.stl*.



Obrázek 11.10: Model krytu GNSS přijímače - program OnShape [28]

<sup>33</sup>Power banka je záložní zdroj elektrické energie, který se skládá z jednoho či více akumulátorů a stejnosměrného měniče. Využívá se především k napájení přenosné elektroniky, jako jsou mobilní telefony, tablety apod. [46]



## 12 U-blox

Společnost u-blox, jejíž produkty byly v rámci této práce použity, je předním světovým dodavatelem zařízení pro určování polohy a jejich výrobky se využívají především v automobilovém průmyslu, zemědělství, dopravě a v neposlední řadě také k prototypování. Společnost byla založena roku 1997 ve švýcarském městě Thalwil, dnes má však pobočky po celém světě. U-blox nabízí rozsáhlé portfólio produktů, které zaštiťují celou oblast GNSS aplikací, včetně svého softwarového řešení pro zpracování měřených dat v podobě programu U-center [40].

### 12.1 U-center

U-center je software produkovaný společností u-blox a je volně k dispozici na jejich webových stránkách<sup>34</sup>, v době psaní této práce byla k dispozici verze 19.03. Tento program uživatelům umožňuje [41] :

- připojení přijímače pomocí sériového portu a následnou komunikaci využívající protokolů UBX<sup>35</sup> nebo standardu NMEA a následné zobrazení dat v binární či textové podobě,
- konfiguraci nastavení GNSS přijímače, jako například typ vypisovaných zpráv, komunikační rychlost, specifikaci povolených satelitních systémů a mnoho dalších parametrů,
- upgrade firmwaru přijímače,
- možnost zaznamenání měřených dat a jejich opětovné přehrávání,
- strukturální a grafickou vizualizaci dat v reálném čase,
- připojení k NTRIP Castru a využití korekčních dat,
- nebo také využití služby Google Earth pro zobrazení měřené polohy na mapovém podkladu.

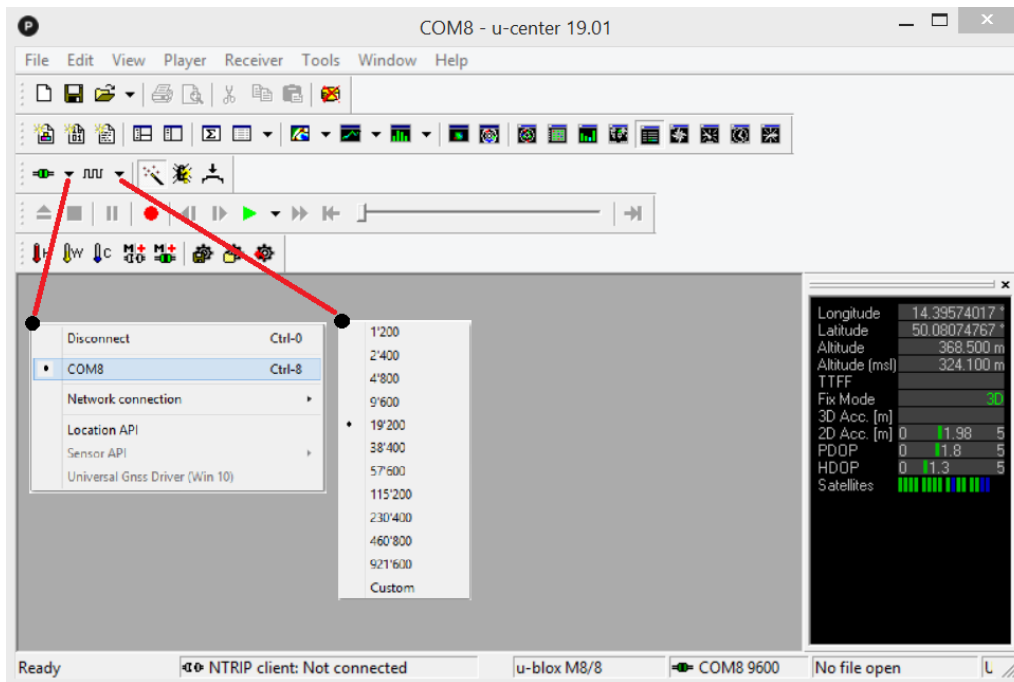
#### 12.1.1 Představení programu

K připojení zařízení je nutné po spuštění programu vybrat komunikační port a přenosovou rychlost (viz obr. 12.1). Pokud neznáme komunikační rychlost daného modulu (běžně 9600 bps), musíme zkusit jednotlivé možnosti, dokud nedojde k úspěšnému připojení.

Na deklarované sériové lince jsou přijímána data od připojeného modulu, která můžeme sledovat v jednotlivých konzolích (viz obr. 12.2) a oknech či vizuálně zobrazit v grafech. Součástí programu je také databáze, do které jsou nahrávány všechny zasílané informace, s kterými lze provádět další statistické analýzy jednotlivých parametrů - výpočet průměru, minima, maxima a směrodatné odchylky. Tuto databázi lze také naplnit pomocí logovacího souboru *\*.ubx*, který je součástí výstupu z vytvořené aplikace GNSS Kontrolér.

<sup>34</sup><https://www.u-blox.com/en/product/u-center>

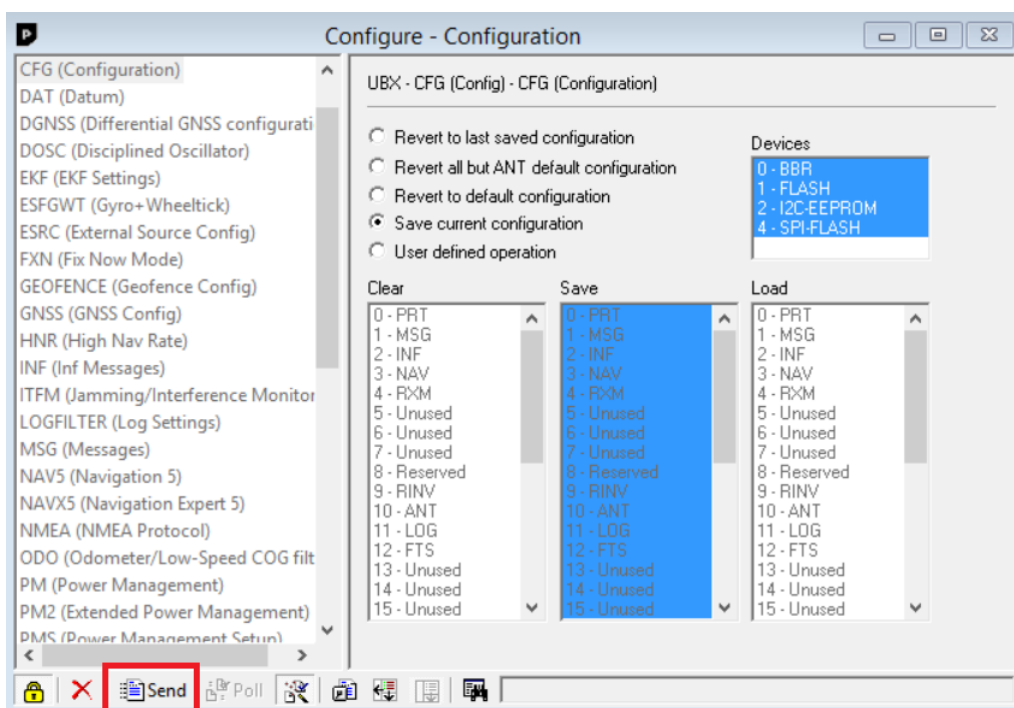
<sup>35</sup>UBX je vlastní komunikační protokol vyvinutý společností U-blox, který navazuje na standard NMEA.



Obrázek 12.1: U-Center - výpis zpráv v konzolích

### 12.1.2 Konfigurace parametrů modulu

Nejdůležitější funkcionalitou programu u-center je jistě možnost konfigurace připojených GNSS modulů založených na u-blox čipech. Nastavení je možné provádět v konfiguračním okně (viz obr. 12.3), popřípadě nahráním konfiguračního souboru, který obsahuje kompletní nastavení čipu. Soubor lze vytvořit v záložce `tools/GNSS Configuration`. nebo pokud tento soubor už máme, můžeme ho ve stejném okně nahrát<sup>36</sup>.



Obrázek 12.3: U-Center - uložení konfigurace modulu

<sup>36</sup>Konfigurační soubor pro čip Neo-M8P je součástí elektronických příloh



```
Binary Console
14:22:52 0000 24 47 4E 47 47 41 2C 31 34 32 32 35 32 2E 30 30 $GNGGA,142252.0,
0010 2C 35 30 32 35 2E 34 34 36 36 33 2C 4E 2C 30 31 ,5025.44663,N,
0020 34 31 31 2E 33 37 38 34 39 2C 45 2C 31 2C 31 32 411.37849,E,1,
0030 2C 30 2E 38 34 2C 32 33 32 2E 35 2C 4D 2C 34 34 ,0.84,232.5,M,
0040 2E 31 2C 4D 2C 2C 2A 34 44 0D 0A ,1,M,,*4D...

Text Console
14:22:52 $GNGGA,142252.00,5025.44663,N,01411.37849,E,1,12,0.84,232.5,M,44.1,M
14:22:52 $GNGSA,A,3,02,05,12,21,26,29,31,25,,,,,1.51,0.84,1.26*1D
14:22:52 $GNGSA,A,3,73,83,82,81,80,,,,,,1.51,0.84,1.26*14
14:22:52 $GPGSV,3,1,11,02,19,044,29,05,18,079,36,12,15,116,30,14,01,242,*7F
14:22:52 $GPGSV,3,2,11,16,01,296,29,21,30,190,35,23,05,336,25,25,51,120,41*76
14:22:52 $GPGSV,3,3,11,26,29,298,50,29,77,052,50,31,52,258,51*47
14:22:52 $GLGSV,3,1,10,65,10,359,31,66,07,055,29,73,51,315,52,74,07,325,22*65
14:22:52 $GLGSV,3,2,10,79,09,151,44,80,60,167,35,81,15,043,28,82,66,037,50*64
14:22:52 $GLGSV,3,3,10,83,57,228,34,84,02,224,19*60
14:22:52 $GNGST,142252.00,32,,,,,1,1,0.84,2.1*75

Packet Console
14:22:52 R -> NMEA GNGGA, Size 75, 'Global Positioning System Fix Data'
14:22:52 R -> NMEA GNGSA, Size 38, 'GNSS DOP and Active Satellites'
14:22:52 R -> NMEA GNGSA, Size 52, 'GNSS DOP and Active Satellites'
14:22:52 R -> NMEA GPGSV, Size 68, 'GNSS Satellites in view'
14:22:52 R -> NMEA GPGSV, Size 70, 'GNSS Satellites in view'
14:22:52 R -> NMEA GPGSV, Size 57, 'GNSS Satellites in view'
14:22:52 R -> NMEA GLGSV, Size 70, 'GNSS Satellites in view'
14:22:52 R -> NMEA GLGSV, Size 70, 'GNSS Satellites in view'
14:22:52 R -> NMEA GLGSV, Size 44, 'GNSS Satellites in view'
14:22:52 R -> NMEA GNGST, Size 40, 'GNSS Pseudorange Error Statistics'
```

Obrázek 12.2: U-Center - výpis zpráv v konzolích

Parametry nastavení jsou rozděleny do jednotlivých kategorií, jejichž seznam je uveden v levé části konfiguračního okna. Je zde například záložka GNSS (GNSS Config), kde lze definovat GNSS systémy, které budou využívány pro výpočet polohy. Dále pak nastavení jednotlivých komunikačních periférií (USB, UART, SPI a I2C) včetně volby vstupních i výstupních zpráv, v záložkách PRT (Ports) a MSG (Messages).

Změny nastavení v jednotlivých záložkách je nutné zaslat modulu pomocí tlačítka **Send** v dolní části okna. Takto zasláné změny mají platnost pouze do té doby, než dojde k odpojení napájení modulu. Pro trvalý zápis je nutné přejít do záložky CFG (Configuration) (viz obr. 12.3), ve které lze nastavení nahrát do paměti modulu BBR a FLASH.

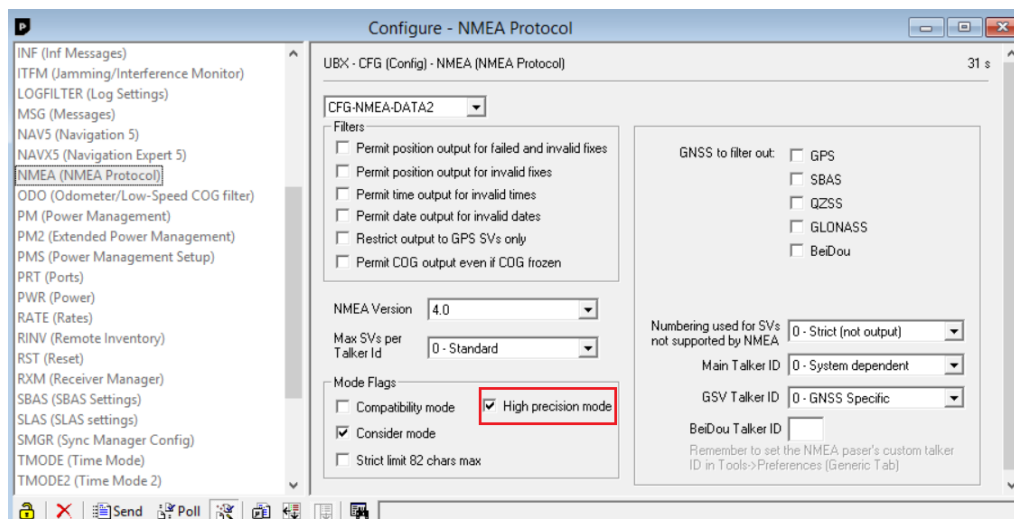
V rámci testování bylo zjištěno, že i když jsou modulu zasílána korekční data a jeho status určení polohy byl **RTK-Fix**, vypisované zprávy o nadmořské výšce byly uváděny pouze v řádu decimetrů, zároveň i zeměpisná šířka a délka byly ve výpisu jistým způsobem zaokrouhlovány. Problém spočíval v tom, že je v modulech primárně nastaven základní mód výpisu NMEA zpráv. Mód lze v konfiguračním okně změnit na výpis s vysokou přesností zatržením pole *High precision mode* v nabídce znázorněné na obrázku 12.4.

Vliv změny tohoto nastavení je demonstrován na následujícím výpisu GGA zpráv, přičemž první řádek představuje výpis v základním módu a na druhém je mód s vysokou přesností<sup>37</sup>.

```
$GNGGA,105256.00,5004.84284,N,01423.75749,E,1,12,1.15,328.9,M,44.4,M,,*4A
$GNGGA,105159.00,5004.8413887,N,01423.7686012,E,1,12,1.08,338.444,M,44.372,M,,*4B
```

<sup>37</sup>Odlišnost vypisovaných souřadnic je způsobena využitím kódového měření se stíženými podmínkami - anténa umístěná za oknem u stěny domu.





Obrázek 12.4: U-Center - změna módu výpisu NMEA zpráv

### 13 Postup testování přijímače

Pro zjištění přesnosti přijímače a funkčnosti vytvořené aplikace GNSS Kontrolér bylo nutné provést testovací měření. Test byl uskutečněn na střeše budovy B Fakulty stavební ČVUT v Praze, na které se nachází několik pilířů umožňujících nucenou centraci. Pro testování byly využity první tři pilíře, dále označované jako body 4001, 4002 a 4003. Souřadnice těchto bodů jsou známé, jelikož jsou proměřovány v rámci výuky geodetických přístrojů rychlou statickou metodou (délka observace 30 minut). Hodnoty souřadnic v systému WGS-84 jsou uvedeny v tabulce 13.

č. pilíře	$\varphi$ [°]	$\lambda$ [°]	el. výška [m]
4001	50.103612908	14.387472811	303.624
4002	50.103653786	14.387527650	303.631
4003	50.103694794	14.387581583	303.599

Tabulka 13.1: Seznam souřadnic pilířů

Hlavním důvodem pro výběr tohoto místa k testování jsou perfektní observační podmínky. Budova vyčnívá nad okolními objekty, což zajišťuje dobrý výhled na oblohu a minimalizuje množství odražených signálů (viz obr. 13.1). Dále se v blízkosti nachází permanentní stanice sítě CZEPOS, která je umístěna na přibližně 6km vzdálené budově Českého úřadu zeměměřického a katastrálního (ČÚZK) v Kobylisích.

Testování bylo měřeno metodou RTK s využitím korekcí z výše zmíněné referenční stanice. Na každém pilíři bylo observováno celkem třikrát po dobu 5 minut, při kterých byla veškerá data zapisována do souboru „\*.ubx“ v mobilní aplikaci. Tyto soubory jsou součástí elektronických příloh. Každý pilíř byl zaměřen jednou s anténou u-blox a dvakrát s využitím antény Leica, u které byla mezi jednotlivými observacemi zhruba hodinová prodleva, jež zaručovala nezávislé rozstavení družic oproti předchozí observaci. Při měření byla zaznamenávána poloha po jedné vteřině a každá observace tak obsahovala přibližně 300 záznamů.



Obrázek 13.1: Testování přijímače - měření souřadnic pilíře magnetickou anténou u-blox

### 13.1 Porovnání měřených souřadnic

Soubory s naměřenými daty byly následně nahrány do aplikace u-center, ze které byly exportovány souřadnice jednotlivých záznamů. Při exportu byl také sledován typ určení polohy a nezafixované hodnoty byly z dalších výpočtů vyloučeny, to však nastalo pouze u pár záznamů na bodě 4002 při měření s magnetickou anténou.

Výsledné zprůměrované hodnoty souřadnic jsou vypsány v tabulce 13.2, přičemž uvedené elipsoidické výšky jsou opraveny o výšku antény a v případě antény Leica o velikost vertikálního offsetu fázového centra<sup>38</sup>. V tabulce 13.3 je uvedeno porovnání s původními souřadnicemi a prostorová vzdálenost k původnímu bodu. Přičemž průměrná vzdálenost k původnímu bodu při použití magnetické antény je **21 mm** a u antény Leica **6 mm**.

č. pilíře	anténa	$\varphi$ [°]	$\lambda$ [°]	el. výška [m]
4001	u-blox	50.103612978	14.387472847	303.641
	Leica - I.	50.103612913	14.387472852	303.622
	Leica - II.	50.103612930	14.387472780	303.625
4002	u-blox	50.103653804	14.387527672	303.640
	Leica - I.	50.103653823	14.387527682	303.623
	Leica - II.	50.103653772	14.387527635	303.627
4003	u-blox	50.103694812	14.387581629	303.634
	Leica - I.	50.103694862	14.387581616	303.604
	Leica - II.	50.103694815	14.387581562	303.602

Tabulka 13.2: Výsledné zprůměrované souřadnice jednotlivých observací

<sup>38</sup>Kalibrační protokol této konkrétní antény nebyl bohužel k dispozici, proto byla použita přibližná hodnota 63,4 mm. Tato hodnota byla nalezena v diplomové práci [1], která se zabývala tvorbou softwaru k zařízení pro absolutní kalibraci GNSS antén a v jejímž rámci byla testována na stejném typu antény.



č. pilíře	anténa	$\Delta\varphi$ [m]	$\Delta\lambda$ [m]	$\Delta H$ [m]	vzd. [m]
4001	u-blox	-0.008	-0.004	-0.017	0.019
	Leica - I.	-0.001	-0.005	0.002	0.005
	Leica - II.	-0.002	0.003	-0.001	0.004
4002	u-blox	-0.002	-0.002	-0.009	0.009
	Leica - I.	-0.004	-0.004	0.008	0.011
	Leica - II.	0.002	0.002	0.004	0.005
4003	u-blox	-0.002	-0.005	-0.035	0.035
	Leica - I.	-0.007	-0.004	-0.005	0.009
	Leica - II.	-0.002	0.002	-0.003	0.004

**Tabulka 13.3:** Souřadnicové odchylky měřených bodů

Dále byly určeny průměrné hodnoty výběrových směrodatných odchylek jednotlivých souřadnic v závislosti na použité anténě, které jsou uvedeny v tabulce 13.4. Tyto hodnoty však nevyjadřují polohovou přesnost, ale pouze stabilitu výsledků v rámci observace a daného fixovaného řešení polohy. Výsledky obou antén byly v tomto ohledu zcela srovnatelné, z čehož lze usuzovat, že výškové odchylky při měření s anténou u-blox mohou být částečně systematického charakteru (nejspíše offset fázového centra antény). Velikost této systematické chyby nebylo možné určit, vzhledem k celkové přesnosti GNSS měření a malému rozsahu měřených dat.

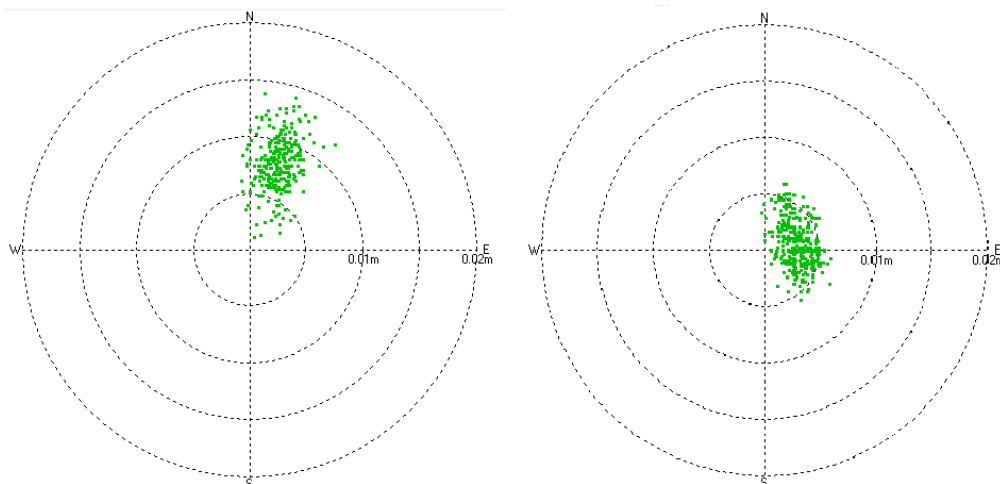
č. pilíře	anténa	$\sigma\varphi$ [°]	$\sigma\varphi$ [m]	$\sigma\lambda$ [°]	$\sigma\lambda$ [m]	$\sigma H$ [m]
4001	u-blox	0.0000000206	0.002	0.0000000185	0.002	0.004
	Leica	0.0000000182	0.002	0.0000000180	0.002	0.005
4002	u-blox	0.0000000391	0.004	0.0000000251	0.003	0.005
	Leica	0.0000000149	0.002	0.0000000238	0.003	0.004
4003	u-blox	0.0000000183	0.002	0.0000000279	0.003	0.005
	Leica	0.0000000210	0.002	0.0000000192	0.002	0.004

**Tabulka 13.4:** Výběrové směrodatné odchylky souřadnic v závislosti na anténě

Pro zajímavost byly v programu u-center vytvořeny deviační grafy (viz obr. 13.2) observace bodu 4001, na kterých jsou znázorněny všechny zaznamenané hodnoty vůči pozici pilíře. Z vypočtených výběrových směrodatných odchylek i zde na grafickém znázornění lze sledovat větší rozptyl při použití magnetické antény u-blox. To je však vzhledem k její pořizovací ceně a provedení zcela přirozené, přesto při porovnání horizontální přesnosti dosahuje vynikajících výsledků.

### 13.2 Zhodnocení testování

Z naměřených dat je patrné, že tyto levné magnetické antény mohou být za určitých podmínek použity pro měření polohy v řádech centimetrů, a že jejich výstup může být srovnatelný s několikanásobně dražšími profesionálními anténami. Jedním z nejdůležitějších faktorů je zabránění vstupu odražených signálů (převážně od země), jelikož tyto antény nejsou vybaveny žádným stíněním. Tento závěr potvrzuje článek [9], který pojednává o použití těchto levných antén pro aplikace s vysokou polohovou přesností. V článku se doporučuje umístění kovové stínící desky o dostatečné tloušťce na spodek antény, která následně tlumí účinky vícecestného šíření. Dalším rozdílem byla stabilita



**Obrázek 13.2:** Znázornění rozptylu bodů při observaci na pilíři 4001 - mag. anténa u-blox (vlevo) a anténa Leica AX1202 GG (vpravo)

fix řešení, kdy při užití magnetické antény docházelo k občasným výpadekům. Měření pak muselo být buďto opakováno, nebo byla tato nezafixovaná data vymazána až posléze. Celkově však výsledky dosahované touto levnou anténou předčily veškerá očekávání. Testování tohoto zařízení bude nadále pokračovat v rámci jiných projektů, v kterých se například počítá s osazením tohoto přijímače na tělo dronu.

Vytvořená aplikace fungovala v rámci testování zcela v pořádku, pouze v jednom případě došlo k výpadku BT spojení, které se následně vyřešilo restartováním přijímače a aplikace.



**Obrázek 13.3:** Testování přijímače - měření souřadnic pilíře geodetickou anténou Leica AX1202GG



## 14 Cenová kalkulace projektu

Výpis všech všech položek je uveden v tabulce 14. Většina komponentů byla objednána ze zahraničních e-shopů, ve kterých byla jejich cena i několikrát nižší než v tuzemských kamenných či online obchodech. Některé položky byly koupeny již dříve, a proto je u nich uvedena přibližná cena.

Položka	cena [Kč]
SparkFun RTK board	5720
power banka	450
anténa u-blox	400
Arduino Due + kabel	300
anténní redukce a kabely	200
DPS + spoj. materiál	200
HC-05	70
3D tisk	40
Součet	<b>7380</b>

**Tabulka 14.1:** Cenová kalkulace projektu



## Závěr

Cílem této diplomové práce bylo navrhnout strukturu zařízení pro příjem GNSS signálů a jeho následné sestavení z vybraných komponentů. Jako základ celého zařízení byl zvolen GNSS modul od společnosti SparFun, který obsahuje čip u-blox Neo-M8P. Byl vybrán především proto, že podporuje příjem korekčních dat RTCM 3.x, což umožňuje přesná měření s využitím technologie RTK. Tento modul společně s mikrokontrolérem Arduino Due a Bluetooth modulem HC-05 představují hlavní kostru celého GNSS přijímače.

V rámci této práce byla také vyvinuta aplikace pro mobilní platformu Android, která následně zaznamenává a zobrazuje data vysílaná vytvořeným GNSS přijímačem. Uživatel si může v aplikaci vytvořit vlastní zakázku, do které si následně ukládá jednotlivé měřené body, popřípadě vytvořit záznam surových měření do souboru ve formátu „\*.ubx“, který lze použít pro následné post-procesní zpracování. Aplikace dále umožňuje vytyčení uložených bodů a také zobrazení aktuálních viditelných družic, včetně dalších popisných informací o jejich pozici a síle signálu. Všechny v aplikaci zobrazované informace jsou přebírány ze zpráv formátu NMEA, jejímuž popisu byla věnována kapitola v úvodní části práce. Aplikace dále zajišťuje přístup ke korekcím poskytovaných ze sítě referenčních stanic (např. CZEPOS), což je klíčové pro využití RTK měření pouze s jedním přijímačem.

V závěru tohoto textu je pak popsán postup testování vytvořeného GNSS přijímače, jež bylo uskutečněno na střeše budovy B Fakulty stavební ČVUT v Praze, na které se nachází pilíře o známých souřadnicích v systému WGS-84. Pilíře byly přijímačem několikrát zaměřeny metodou RTK, přičemž délka jednotlivých observací byla 5 minut se záznamem polohy po jedné vteřině. V rámci testování byla srovnávána kvalita dosažených výsledků při použití levné magnetické antény AN-MS-0-005 od společnosti u-blox a profesionální geodetické antény Leica AX1202 GG. Výsledná průměrná prostorová vzdálenost měřených bodů od původních souřadnic pilířů byla **6mm** při použití antény Leica a **21 mm** s anténou u-blox. Z výsledků uvedených v kapitole 13 dále vyplývá, že je možné tuto levnou anténu používat pro RTK měření. Je však nutné zajistit odstínění odražených signálů, které zhoršují či úplně znemožňují zafixování polohy. Z časových důvodů nebyla uskutečněna žádná komplexnější testování, která však budou provedena v rámci dalších projektů.

Celkově bych cíle práce považoval za splněné. Výsledkem jejího snažení je funkční prototyp GNSS zařízení včetně kontrolní aplikace pro platformu Android, jejíž vytvoření zabralo většinu času. Z toho, že se aplikaci podařilo zprovoznit, jsem měl největší radost, protože to bylo poprvé co jsem vyvíjel takto komplexní projekt. Na jejím vývoji a úpravách bych rád pokračoval i v budoucnu, přičemž hlavním prvkem ke zlepšení by byla možnost zobrazení výstupní polohy přijímače, nejen v souřadném systému WGS-84 ale i dalších (např. S-JTSK). Zdrojový kód aplikace je dostupný na tomto GitHub repozitáři :

<https://github.com/GNNS-Kontroler/aplikace>





## Reference

- [1] GALATA, Marek. Tvorba softwarové podpory k zařízení pro absolutní kalibraci GNSS antén. Diplomová práce, Univerzita Hradec Králové, Brno, 2016. Dostupné z: [https://www.vutbr.cz/www\\_base/zav\\_prace\\_soubor\\_verejne.php?file\\_id=183317](https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=183317).
- [2] JEDLIČKA, Jakub. Analýza vlivů působících na signály GPS a jejich dopady na výslednou chybu vypočtené polohy. Diplomová práce, Západočeská univerzita v Plzni, Plzeň, 2012. Dostupné z: <https://dspace5.zcu.cz/handle/11025/2326>.
- [3] MAŠEK, Pavel. Vývoj multiplatformních mobilních aplikací. Diplomová práce, Vysoké učení technické v Brně, Hradec Králové, 2017. Dostupné z: <https://theses.cz/id/20k2p8/STAG89569.pdf>.
- [4] NOVATEL. *An Introduction to GNSS*. Calgary, Alberta, Canada: NovAtel Inc, 2015. ISBN 978-0-9813754-0-3.
- [5] TRUHLÁŘ, Luboš. Vývoj NTRIP Casteru. Diplomová práce, České vysoké učení technické v Praze, Praha, 2012. Dostupné z: <http://gama.fsv.cvut.cz/~cepek/proj/dp/2013/lubos-truhlar-dp-2013.pdf>.
- [6] ŠVÁBENSKÝ, Otakar; WEIGEL, Josef a FIXEL, Jan. *Základy GPS a jeho praktické aplikace*. : CERM, 1995.
- [7] WARGON, John M. *Apache Cordova 4 programming*. Indiana: Addison Wesley, 2015. ISBN 978-0-13-404819-2.
- [8] *U-blox : Product Selector* [online]. [cit. 2019-05-10]. Dostupné z: <https://www.u-blox.com/en/product-search>.
- [9] *U-blox : Achieving Centimeter Level Performance with Low Cost Antennas* [online]. [cit. 2019-05-10]. Dostupné z: [https://cdn.sparkfun.com/assets/0/c/0/1/c/AntennasForRTK\\_WhitePaper\\_\\_UBX-16010559\\_.pdf](https://cdn.sparkfun.com/assets/0/c/0/1/c/AntennasForRTK_WhitePaper__UBX-16010559_.pdf).
- [10] *Arduino.playground.cc: Arduino Due* [online]. [cit. 2019-05-10]. Dostupné z: <https://store.arduino.cc/due>.
- [11] *Zbyšek Voda a Tým HW Kitchen - Průvodce světem Arduina* [online]. [cit. 2019-05-10]. Dostupné z: <https://www.robotikabrno.cz/docs/arduino/Pr%C5%AFvodce-sv%C4%9Btem-Arduina-CZ.pdf>.
- [12] *HC-05 Bluetooth Module - User's manual* [online]. [cit. 2019-05-10]. Dostupné z: <https://www.gme.cz/data/attachments/dsh.772-148.1.pdf>.
- [13] ČÁBELKA, Miroslav. *Úvod do GPS* [online]. [cit. 2019-05-10]. Dostupné z: <https://www.natur.cuni.cz/geografie/geoinformatika-kartografie/ke-stazeni/vyuka/gps/skriptum-uvod-do-gps/>.
- [14] *Chrome DevTools* [online]. [cit. 2019-05-10]. Dostupné z: <https://developers.google.com/web/tools/chrome-devtools/>.
- [15] *Apache Cordova - documentation* [online]. [cit. 2019-05-10]. Dostupné z: <https://cordova.apache.org/docs/en/latest/>.



- [16] *Dimdim's Blog : Fixing startup issues with Arduino DUEs* [online]. [cit. 2019-05-10]. Dostupné z: <http://www.dimdim.gr/2016/03/fixing-startup-issues-with-arduino-dues/>.
- [17] *GALILEO - Evropský globální navigační družicový systém* [online]. [cit. 2019-05-10]. Dostupné z: <http://www.czechspaceportal.cz/3-sekce/gnss-systemy/galileo/>.
- [18] *Information and analysis center for positioning, navigation and timing - GLONASS status* [online]. [cit. 2019-05-10]. Dostupné z: <https://www.glonass-iac.ru/en/GLONASS/>.
- [19] *GPS - Space Segment* [online]. [cit. 2019-05-10]. Dostupné z: <https://www.gps.gov/systems/gps/space/>.
- [20] *Hybridní aplikace - kdy ano a kdy ne?* [online]. [cit. 2019-05-10]. Dostupné z: <https://pixelfield.cz/blog/hybridni-mobilni-aplikace-kdy-ano-a-kdy-ne/>.
- [21] *Vlastimil Kratochvíl - Globální navigační satelitní systémy (GNSS)* [online]. [cit. 2019-05-10]. Dostupné z: [https://www.fce.vutbr.cz/ged/opvk/inovace/blok\\_C/GE07/Texty\\_GNSS.pdf](https://www.fce.vutbr.cz/ged/opvk/inovace/blok_C/GE07/Texty_GNSS.pdf).
- [22] *Informační systém Masarykovy univerzity - Možnosti polohových satelitních systémů* [online]. [cit. 2019-05-10]. Dostupné z: [https://is.muni.cz/e1/1431/jaro2015/Z8109/um/54520744/Moznosti\\_polohovych\\_satelitnich\\_systemu.pdf](https://is.muni.cz/e1/1431/jaro2015/Z8109/um/54520744/Moznosti_polohovych_satelitnich_systemu.pdf).
- [23] *Ublox datasheet : u-blox M8 High Precision GNSS Modules* [online]. [cit. 2019-05-10]. Dostupné z: [https://www.u-blox.com/sites/default/files/NEO-M8P\\_DataSheet\\_%28UBX-15016656%29.pdf](https://www.u-blox.com/sites/default/files/NEO-M8P_DataSheet_%28UBX-15016656%29.pdf).
- [24] *Ublox datasheet : NEO/LEA-M8T - u-blox M8 concurrent GNSS timing modules* [online]. [cit. 2019-05-10]. Dostupné z: [https://www.u-blox.com/sites/default/files/NEO-LEA-M8T-FW3\\_DataSheet\\_%28UBX-15025193%29.pdf](https://www.u-blox.com/sites/default/files/NEO-LEA-M8T-FW3_DataSheet_%28UBX-15025193%29.pdf).
- [25] *Standard NMEA-0183 sentences description* [online]. [cit. 2019-05-10]. Dostupné z: [freenmea.net/docs](http://freenmea.net/docs).
- [26] *Networked Transport of RTCM via Internet Protocol (Ntrip)* [online]. [cit. 2019-05-10]. Dostupné z: [https://igs.bkg.bund.de/root\\_ftp/NTRIP/documentation/NtripDocumentation.pdf](https://igs.bkg.bund.de/root_ftp/NTRIP/documentation/NtripDocumentation.pdf).
- [27] *NTRIP Components Names* [online]. [cit. 2019-05-10]. Dostupné z: <http://lefebure.com/articles/ntrip-names/>.
- [28] *OnShape - webový modelovací program* [online]. [cit. 2019-05-10]. Dostupné z: <https://www.onshape.com/>.
- [29] *Pavel Tesař - Úvod do GNSS* [online]. [cit. 2019-05-10]. Dostupné z: <ftp://athena.fsv.cvut.cz/EMEG/uvodGNSS.pdf>.
- [30] *PhotoPoint: Powerbanka - ADATA* [online]. [cit. 2019-05-10]. Dostupné z: <https://www.photopoint.ee/en/energy/1141880-power-bank-adata-ap12500d-dgt-5v-cbk-12500mah-usb-20-black-color>.
- [31] *RINEX - The Receiver Independent Exchange Format* [online]. [cit. 2019-05-10]. Dostupné z: <ftp://igs.org/pub/data/format/rinex304.pdf>.





- [32] *GIS Resources - RINEX - The Receiver Independent Exchange Format* [online]. [cit. 2019-05-10]. Dostupné z: <http://www.gisresources.com/rinex-receiver-independent-exchange-format/>.
- [33] *RTCM* [online]. [cit. 2019-05-10]. Dostupné z: <http://www.rtcmm.org/>.
- [34] *RTKLIB: An Open Source Program Package for GNSS Positioning* [online]. [cit. 2019-05-10]. Dostupné z: <http://www.rtklib.com/>.
- [35] *WAAS, EGNOS, RTCM, RTK and a sea of acronyms!* [online]. [cit. 2019-05-10]. Dostupné z: <https://ruggedbits.wordpress.com/2007/10/19/waas-egnos-rtcm-rtk-and-a-sea-of-acronyms/>.
- [36] *Lukáš Kořínek : Bezdrátové moduly - Bluetooth HC-05* [online]. [cit. 2019-05-10]. Dostupné z: <http://www.sakul.cz/bluetooth-hc-05/n>.
- [37] *SparkFun GPS-RTK Board - NEO-M8P-2* [online]. [cit. 2019-05-10]. Dostupné z: <https://www.sparkfun.com/products/15005>.
- [38] *Otakar Švábenský a spol. - Metodika GPS měření a vyhodnocení* [online]. [cit. 2019-05-10]. Dostupné z: [http://fast.darmy.net/opory%20-%20IV%20nMgr/HE09\\_M01-Semin%C3%A1%C5%99%20GPS.pdf](http://fast.darmy.net/opory%20-%20IV%20nMgr/HE09_M01-Semin%C3%A1%C5%99%20GPS.pdf).
- [39] *Circuit Basics - Basics of UART communication* [online]. [cit. 2019-05-10]. Dostupné z: <http://www.circuitbasics.com/basics-uart-communication/>.
- [40] *Ublox - History* [online]. [cit. 2019-05-10]. Dostupné z: <https://www.u-blox.com/en/history-0>.
- [41] *u-center User Guide* [online]. [cit. 2019-05-10]. Dostupné z: [https://www.u-blox.com/sites/default/files/u-center\\_UserGuide\\_%28UBX-13005250%29.pdf](https://www.u-blox.com/sites/default/files/u-center_UserGuide_%28UBX-13005250%29.pdf).
- [42] *Ublox M8 : Receiver description - Including Protocol Specification* [online]. [cit. 2019-05-10]. Dostupné z: [https://www.u-blox.com/sites/default/files/products/documents/u-blox8-M8\\_ReceiverDescrProtSpec\\_%28UBX-13003221%29\\_Public.pdf](https://www.u-blox.com/sites/default/files/products/documents/u-blox8-M8_ReceiverDescrProtSpec_%28UBX-13003221%29_Public.pdf).
- [43] *GPS - Essentials of Satellite Navigation* [online]. [cit. 2019-05-10]. Dostupné z: [https://www.u-blox.com/sites/default/files/gps\\_compendiumgps-x-02007.pdf](https://www.u-blox.com/sites/default/files/gps_compendiumgps-x-02007.pdf).
- [44] *GNSS - Global Navigation Satellite System* [online]. [cit. 2019-05-10]. Dostupné z: <http://www.czechspaceportal.cz/3-sekce/gnss-systemy/>.
- [45] *Jak vyvíjet mobilní aplikace* [online]. [cit. 2019-05-10]. Dostupné z: <http://janvaclavik.cz/jak-vyvijet-mobilni-aplikace/>.
- [46] *Wikipedie: Powerbanka* [online]. [cit. 2019-05-10]. Dostupné z: <https://cs.wikipedia.org/wiki/Powerbanka>.



## Seznam obrázků

2.1	Princip měřené vzdálenosti - zdroj [13]	11
2.2	Princip určení polohy protínáním z délek - zdroj [43]	12
3.1	Princip modulace GPS signálu - zdroj [29]	14
4.1	Vliv geometrie na přesnost polohy - zdroj [2]	16
4.2	Trajektorie signálu - zdroj [2]	17
4.3	Závislost délky průchodu atmosférou na elevačním úhlu - zdroj [22]	18
4.4	Multipath - zdroj [43]	19
5.1	Statická metoda měření - zdroj [13]	20
5.2	Metoda Stop and Go - zdroj [13]	21
5.3	Metoda RTK - zdroj [35] upraveno	22
6.1	Schéma NTRIP sítě - zdroj [27]	28
7.1	Rozdělení mobilním trhu 2018 - zdroj <a href="http://www.staging.moodup.team">www.staging.moodup.team</a>	32
8.1	Ikona Apache Cordova - zdroj [7]	34
8.2	Schéma architektury aplikace - zdroj [15]	35
8.3	Adresářová struktura projektu	37
8.4	Vývojový nástroj Google Chrome	39
8.5	Přístup k inspekčnímu nástroji	39
9.1	Ikona aplikace GNSS Kontrolér	40
9.2	Struktura SQL databáze	41
9.3	Struktura okna aplikace	42
9.4	Domovská obrazovka	43
9.5	Aplikace - okno Nastavení	44
9.6	Aplikace - okno Měření	45
9.7	Aplikace - okno Vytyčení	46
9.8	Aplikace - okno Skyplot	47
11.1	Vývojová deska Arduino Due	49
11.2	Bluetooth modul HC-05 - zdroj [36]	51
11.3	Schéma zapojení modulu HC-05 : vysílací režim (vlevo) a konfigurační režim (vpravo)	52
11.4	Možné kombinace GNSS systémů - [24]	54
11.5	GNSS modul - NEO-M8T	54
11.6	SparkFun GPS-RTK Board NEO-M8P-2 – zdroj [37]	55
11.7	Schéma zapojení modulů k Arduino desce	56
11.8	Anténa- u-blox AN-MS-0-005 - zdroj [8]	56
11.9	Zdroj napětí - power banka ADATA [30]	57
11.10	Model krytu GNSS přijímače - program OnShape [28]	57
12.1	U-Center - výpis zpráv v konzolích	59
12.3	U-Center - uložení konfigurace modulu	59
12.2	U-Center - výpis zpráv v konzolích	60
12.4	U-Center - změna módu výpisu NMEA zpráv	61
13.1	Testování přijímače - měření souřadnic pilíře magnetickou anténou u-blox	62
13.2	Znázornění rozptylu bodů při observaci na pilíři 4001 - mag. anténa u-blox (vlevo) a anténa Leica AX1202 GG (vpravo)	64
13.3	Testování přijímače - měření souřadnic pilíře geodetickou anténou Leica AX1202GG	64

*\*Obrázky bez uvedeného zdroje byly vytvořeny autorem práce.*



## Seznam tabulek

6.1	Popis částí NMEA zprávy RMC . . . . .	24
6.2	Popis částí NMEA zprávy GSA . . . . .	24
6.3	Popis částí NMEA zprávy GGA . . . . .	25
6.4	Popis částí NMEA zprávy GSV . . . . .	26
6.5	Popis částí NMEA zprávy GST . . . . .	26
6.6	Popis zdrojové tabulky zaslané z NTRIP Casteru CZEPOS . . . . .	31
11.1	Technické specifikace modulu HC-05 - zdroj [36] . . . . .	51
13.1	Seznam souřadnic pilířů . . . . .	61
13.2	Výsledné zprůměrované souřadnice jednotlivých observací . . . . .	62
13.3	Souřadnicové odchylky měřených bodů . . . . .	63
13.4	Výběrové směrodatné odchylky souřadnic v závislosti na anténě . . . . .	63
14.1	Cenová kalkulace projektu . . . . .	65



## Seznam použitých zkratek

API	-	Aplikation Programming Interface (rozhraní pro programování aplikací)
BT	-	Bluetooth
COM	-	Hardwarové rozhraní seriového portu
CSS	-	Cascading Style Sheets
CZEPOS	-	Síť permanentních stanic GNSS České Republiky
DOP	-	Dilution of precision (parametr přesnosti polohy)
DPS	-	Deska plošného spoje (angl. zkratka PCB)
ESA	-	European Space Agency (Evropská kosmická agentura)
GK	-	Aplikace GNSS Kontrolér
GNSS	-	Global Navigation Satellite System (globální navigační satelitní systém)
HTML	-	Hypertext Markup Language
JS	-	JavaScript
NMEA	-	National Marine Electronics Association (formát zpráv vysílaných přijímačem)
NTRIP	-	Networked Transport of RTCM via Internet Protocol (formát pro přenos korekcí)
PRN	-	Pseudo-Random Noise (Pseudonáhodný kód)
RINEX	-	Receiver Independent Exchange (standard zápisu dat)
RTCM	-	Radio Technical Commission for Maritime Services (formát zápisu korekčních dat)
SDK	-	Software development kit (systémové vývojové nástroje)
SQL	-	Structured Query Language (jazyk pro práci s databázemi)
WGS-84	-	World Geodetic System 1984 (souřadnicový systém)



## Seznam použitých programů

- Arduino IDE
- Fritzing
- Matlab R2014a (Mathworks)
- OnShape
- Simplify3D
- Texmaker 4.3 (Pascal Brachet)
- U-center v 19.01
- VS-studio Code

## Seznam příloh

**A Elektronická příloha**

**74**



## A Elektronická příloha

CD obsahuje následující soubory :

- Text diplomové práce
- Aplikace GNSS Kontrolér
- Zdrojový kód pro desku Arduino
- Měřená testovací data
- Modely krytu přijímače
- Návrh desky plošného spoje
- Konfigurační soubor GNSS modulu