



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

TESTOVACÍ PRACOVNÍSTĚ PRO VZDÁLENÉ OMEZENÍ RYCHLOSTI VOZIDLA

TESTBED FOR REMOTE VEHICLE SPEED LIMITATION

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Jindřich Šíma

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Jan Dvořák, Ph.D.

BRNO 2023

Bakalářská práce

bakalářský studijní program **Informační bezpečnost**

Ústav telekomunikací

Student: Jindřich Šíma

ID: 231287

Ročník: 3

Akademický rok: 2022/23

NÁZEV TÉMATU:

Testovací pracoviště pro vzdálené omezení rychlosti vozidla

POKYNY PRO VYPRACOVÁNÍ:

Cílem bakalářské práce je návrh a realizace pracoviště, které bude demonstrovat nouzové zastavení pohybujícího se vozidla. Základním stavebním prvkem pracoviště bude OBU (On Board Unit) zařízení, které slouží pro komunikaci mezi vozidly nebo komunikaci vozidla se stanicí (tzv. V2X – vehicle to everything) pomocí ITS-G5 systému. Pomocí zpráv IVI (In-Vehicle Information) bude provedeno snížení otáček a rychlosti vozidla, případně dalších parametrů (rozblíknání směrovek atd.). Ovlivnění těchto parametrů bude prezentováno pomocí dodané přístrojové desky a pomocí výpisů zachycené komunikace. Navržené zařízení bude mít implementováno GPS zařízení pro lokalizaci polohy a spolu s přístrojovou deskou bude zpracováno do vzhledného boxu. Prostudujte vlastnosti a funkce dodané open-source OBU jednotky, tyto jednotky zprovozněte a vytvořte požadovanou komunikaci. Zaměřte se na typy a vlastnosti potřebných zpráv pro správnou funkci pracoviště. Prostudujte i možnosti zabezpečení komunikace mezi jednotkami a možnosti analýzy přenášeného signálu. Zaměřte se i na možnosti omezení rychlosti v reálných motorových vozidlech. Navržené pracoviště vhodně otestujte, získané výsledky zpracujte a vhodně komentujte.

DOPORUČENÁ LITERATURA:

- [1] POSPÍŠIL, J.; DVOŘÁK, J.; ŠEDA, P.; MALINA, L.; MARTINÁSEK, Z. Capturing the Vehicle to Everything Communication Using Software Defined Radio in ITS-G5. In 45nd International Conference on Telecommunications and Signal Processing (TSP 2022). IEEE, 2022.
- [2] MILANES, Vicente, Jorge VILLAGRA, Jorge GODOY, Javier SIMO, Joshué PEREZ a Enrique ONIEVA. An Intelligent V2I-Based Traffic Management System. IEEE Transactions on Intelligent Transportation Systems. 2012, 13(1), 49-58. DOI: 10.1109/TITS.2011.2178839.

Termín zadání:

Termín odevzdání: 17.8.2023

Vedoucí práce: Ing. Jan Dvořák, Ph.D.

doc. Ing. Jan Hajný, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Bakalářská práce se zabývá návrhem a implementací pracoviště pro vzdálené omezení rychlosti vozidla. V teoretické části práce je rozbor inteligentních dopravních systémů, palubních jednotek OBU (On-Board Unit), zpráv v komunikaci jednotek a knihovny Vanetza, která implementuje standardy pro komunikaci OBU jednotek.

Praktická část práce obsahuje návrh a realizaci pracoviště pro vzdálené omezení rychlosti vozidla, popis součástí pracoviště, postup jeho nastavení, zachycení a analýzu komunikace v pracovišti.

KLÍČOVÁ SLOVA

Arduino Nano, Blender, CAM, CAN, GPS, inteligentní dopravní systémy, IVIM, KiCad, MCP2515, mosquitto, MQTT, on-board unit, road side unit, Vanetza, vehicle to vehicle

ABSTRACT

Bachelor thesis focuses on planing and realisation of workplace for remote vehicle speed limitation. In the theoretical part there is an analysis of intelligent transportation systems, on-board units, messages in the communication of units and the Vanetza library, which implements standards for communication of units.

Practical part consists of planing and realisation of workplace for remote vehicle speed limitation, description of workplace components, the procedute for setting it up, capturing and analyzing communication in the workplace.

KEYWORDS

Arduino Nano, Blender, CAM, CAN, GPS, intelligent transportation systems, IVIM, KiCad, MCP2515, mosquitto, MQTT, on-board unit, road side unit, Vanetza, vehicle to vehicle

ŠÍMA, Jindřich. *Testovací pracoviště pro vzdálené omezení rychlosti vozidla*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2023, 57 s. Bakalářská práce. Vedoucí práce: Ing. Jan Dvořák, Ph.D.

Prohlášení autora o původnosti díla

Jméno a příjmení autora: Jindřich Šíma
VUT ID autora: 231287
Typ práce: Bakalářská práce
Akademický rok: 2022/23
Téma závěrečné práce: Testovací pracoviště pro vzdálené omezení rychlosti vozidla

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno

.....

podpis autora*

* Autor podepisuje pouze v tištěné verzi.

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce panu Ing. Janu Dvořákovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Obsah

Úvod	11
Cíle práce	12
1 Teoretická část studentské práce	13
2 Inteligentní dopravní systémy	14
2.1 Kooperativní inteligentní dopravní systémy	14
3 Global Positioning System	15
4 Vehicle-to-Everything	16
5 Roadside Unit	17
6 On-Board Unit	18
6.1 Komerční OBU	18
6.2 Linuxové OBU jednotky	20
7 Vanetza	22
7.1 Socktap	22
8 Zabezpečení	24
8.1 Infrastruktura pro správu a distribuci veřejných klíčů	25
8.2 Certify	25
9 Sada Standardů 802.11	26
9.1 802.11p	26
10 Typy zpráv	27
10.1 Cooperative Awareness Messages	27
10.2 Infrastructure to Vehicle Information Messages	28
10.3 Decentralized Environmental Notification Message	28
11 Controller Area Network	30
12 MQ Telemetry Transport	32
12.1 Knihovna Mosquitto	33

13 Výsledky studentské práce	34
13.1 Schéma zapojení základního pracoviště pro vzdálené ovlivnění rychlosti vozidla	34
13.1.1 Základní nastavení OBU jednotky	34
13.1.2 Základní komunikace mezi OBU	35
13.1.3 GPS modul	36
13.2 Pracoviště pro vzdálené ovlivnění otáček obsahující přístrojovou desku	38
13.2.1 Návrh plošného spoje	39
13.2.2 Arduino Nano	40
13.2.3 CAN Převodník MCP2515	42
13.2.4 Box pro přístrojovou desku a zpracování CAN zpráv	42
13.2.5 Realizace komunikace mezi jednotkami a ovlivnění otáček na přístrojové desce	44
13.3 Zachycená komunikace	47
Závěr	50
Literatura	51
Seznam symbolů a zkratk	54
A Obsah elektronické přílohy	57

Seznam obrázků

2.1	Architektura C-ITS systémů	14
3.1	GPS segmenty	15
4.1	Komunikace V2X, Převzato a upraveno z [3]	16
5.1	RSU od Cohda Wireless, Zdroj: [6]	17
6.1	Commsignia ITS-OB4, Zdroj: [9]	18
6.2	Cohda MK6C EVK, Zdroj: [10]	20
6.3	Unex OBU-201E, Zdroj: [11]	20
6.4	Linuxová OBU jednotka	21
8.1	Schéma digitálního podpisu	24
10.1	Zpráva CAM, Převzato a upraveno z [24]	27
10.2	Zpráva IVIM, Převzato a upraveno z [24]	28
10.3	Zpráva DENM, Převzato a upraveno z [24]	29
11.1	Schéma CAN bus v běžném automobilu	30
11.2	Rámec CAN zprávy	30
12.1	Příklad MQTT komunikace	32
13.1	Schéma zapojení základního pracoviště	34
13.2	Síťové rozhraní vysílající OBU jednotky	35
13.3	Výpis paketů do konzole	36
13.4	Výpis vybraných informací z paketů	36
13.5	GPS modul	37
13.6	GPS data v NMEA formátu	37
13.7	Výpis cgps aplikace	38
13.8	Zjištění názvu a adresy GPS přijímače	38
13.9	Návrh pracoviště na omezení otáček	39
13.10	Zapojení pracoviště na omezení otáček	39
13.11	Schéma plošného spoje v KiCadu	40
13.12	Návrh plošného spoje v KiCadu	41
13.13	Arduino Nano na plošném spoji	42
13.14	Zapojení CAN převodníku MCP2515	43
13.15	Návrh boxu v Blenderu	43
13.16	Vytisknutý box pro osazení přístrojové desky	44
13.17	Spuštění socktapu na vysílací OBU	45
13.18	Spuštění socktapu na přijímací OBU	45
13.19	Otáčky po omezení	46
13.20	Zobrazení odeslané mosquito zprávy	46
13.21	Komunikace před omezením otáček	48
13.22	Komunikace při omezení otáček	48

13.23	Zachycená komunikace pomocí strace	49
-------	--	----

Úvod

V dnešní době se na silnicích pohybuje stále více vozidel. To může vést k většímu počtu nehod, pomalejšímu provozu a zvýšení spotřeby paliv kvůli neustálému popojíždění. Inteligentní dopravní systémy by mohly s těmito problémy pomoci. Za využití kooperativních inteligentních dopravních systémů je možné pomoci například policejním složkám při stíhání vozidla tak, že by díky GPS a za použití palubní jednotky identifikovaly podezřelé vozidlo, následně jej plynule a bezpečně zastavily.

Tato práce se zabývá komunikací, kterou zajišťují palubní jednotky OBU (On-Board Unit), zprávami, které se v této komunikaci vyskytují, technologiemi a zařízeními, umožňující komunikaci a možnostmi jejich využití.

První část bakalářské práce se zabývá inteligentními dopravními systémy, jejich komunikací a komunikací s částmi vozidla. Obsahuje také popsání zpráv v komunikaci mezi vozidly i uvnitř nich.

Ve druhé části je zobrazeno schéma zapojení základního pracoviště, nastavení palubních jednotek a GPS modulu, zprovoznění základní komunikace a popis pracoviště pro ovlivnění otáček vozidla.

Cíle práce

Cílem bakalářské práce je návrh a realizace pracoviště, které bude demonstrovat nouzové zpomalení pohybujícího se vozidla. Základním stavebním prvkem pracoviště budou OBU (On-Board Unit) zařízení, které slouží pro komunikaci mezi vozidly nebo komunikaci vozidla se stanicí (tzv. V2X – Vehicle-to-everything) pomocí ITS-G5 systému. Pomocí zpráv IVI (In-Vehicle Information) bude provedeno snížení otáček a rychlosti vozidla, případně ovlivnění dalších funkcí. Ovlivnění těchto parametrů bude prezentováno pomocí dodané přístrojové desky. Navržené zařízení bude spolu s přístrojovou deskou zpracováno do vzhledného boxu, který bude obsahovat požadované periferie. Součástí práce je prostudování vlastností a funkcí dodané open-source OBU jednotky. Zaměření především na komunikaci mezi OBU jednotkami a typy potřebných zpráv pro správnou funkci pracoviště. Dalšími cíli jsou prostudování možností zabezpečení komunikace mezi jednotkami a možnosti analýzy přenášeného signálu. Cílem je, také vhodně otestovat pracoviště a získané výsledky zpracovat.

1 Teoretická část studentské práce

V dnešní době existují vozidla, která disponují připojením k síti Wi-Fi (Wireless Fidelity), Bluetooth, GPS (Global Positioning System), vestavěnými aplikacemi, připojením k internetu, komunikace s chytrým telefonem. Proto dnes některé automobily připomínají spíš pojízdné počítače, např. automobily značky Tesla disponují výpočetním výkonem až 362 TFLOPS (teraFLOPS – Floating-Point Operations Per Second), operační pamětí 16 GB, radarem, kamerami atd. Díky tomu poskytují automobily Tesla částečně autonomní řízení (přizpůsobení rychlosti okolnímu provozu, pomoc s řízením v označeném pruhu, automatické parkování). Blížíme se tedy do doby, kdy plně autonomní řízení nebude nic neobvyklého.

Tato práce popisuje systémy, které by měly pomoci snížit počet nehod a zvýšit plynulost provozu v každodenním životě. Tyto systémy jsou využitelné jak v lidmi řízených vozidlech, tak výhledově v autonomně řízených vozidlech. Aby tyto systémy fungovaly efektivně, tak musí vozidla předávat a přijímat informace od jiných zařízení. Komunikovat s ostatními zařízeními mohou buď pomocí sady standardů 802.11 (zařízení v okolí) nebo pomocí LTE (Long-Term Evolution) sítí.

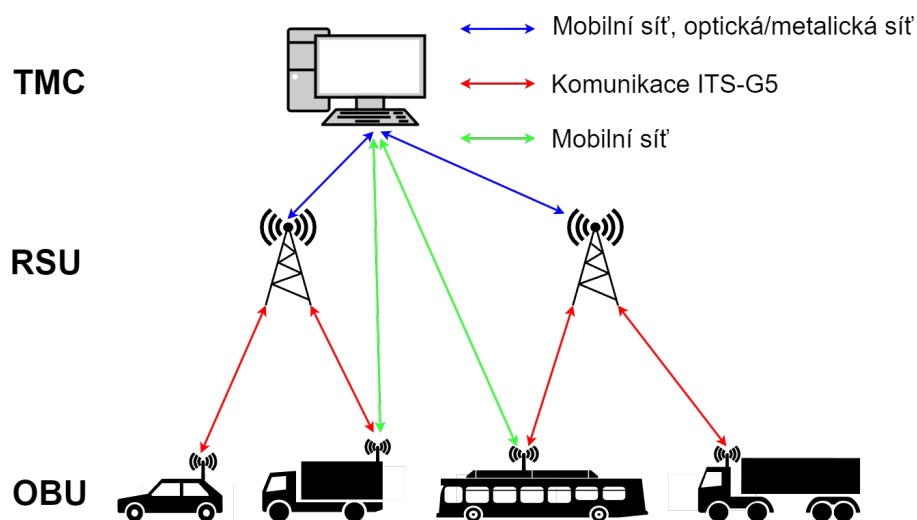
2 Inteligentní dopravní systémy

Inteligentní dopravní systémy ITS (Intelligent Transport Systems) jsou systémy, které za využití informačních a komunikačních technologií pracují na ochraně zdraví osob, snížení počtu dopravních nehod a zvýšení plynulosti dopravy.

2.1 Kooperativní inteligentní dopravní systémy

Kooperativní inteligentní dopravní systémy C-ITS (Cooperative-Intelligent Transport Systems) jsou založené na vzájemné výměně informací a slouží k zabezpečenému přenosu relevantních dat mezi vozidly a ostatními zařízeními. Kooperativní systémy mohou předcházet nebezpečným situacím, včasně varovat před nebezpečím a zajistit plynulejší provoz. C-ITS stanice využívají komunikace krátkého dosahu na bázi Wi-Fi a komunikaci na delší vzdálenost pomocí mobilních sítí [1]. Hlavní části C-ITS systémů jsou **OBU**, **RSU** a **TMC**.

- **TMC – Traffic Management Centre**, je centrální systém, který sbírá data z provozu, analyzuje je, vytváří C-ITS zprávy a posílá je vozidlům. Za centrální systém lze považovat např. Národní dopravní informační centrum.
- **OBU – On-Board Unit**, je jednotka ve vozidle, která vysílá informace infrastruktuře (RSU), nebo dalším OBU jednotkám v okolí. Často bývá propojena s Human-Machine-Interface, aby mohla přijaté informace sdělit řidiči.
- **RSU – Roadside Unit**, je zařízení u pozemních komunikací, které sbírá data od OBU projíždějících vozidel, vysílá C-ITS zprávy přijaté od centrálních systémů a vyměňuje si C-ITS zprávy s centrálním systémem.

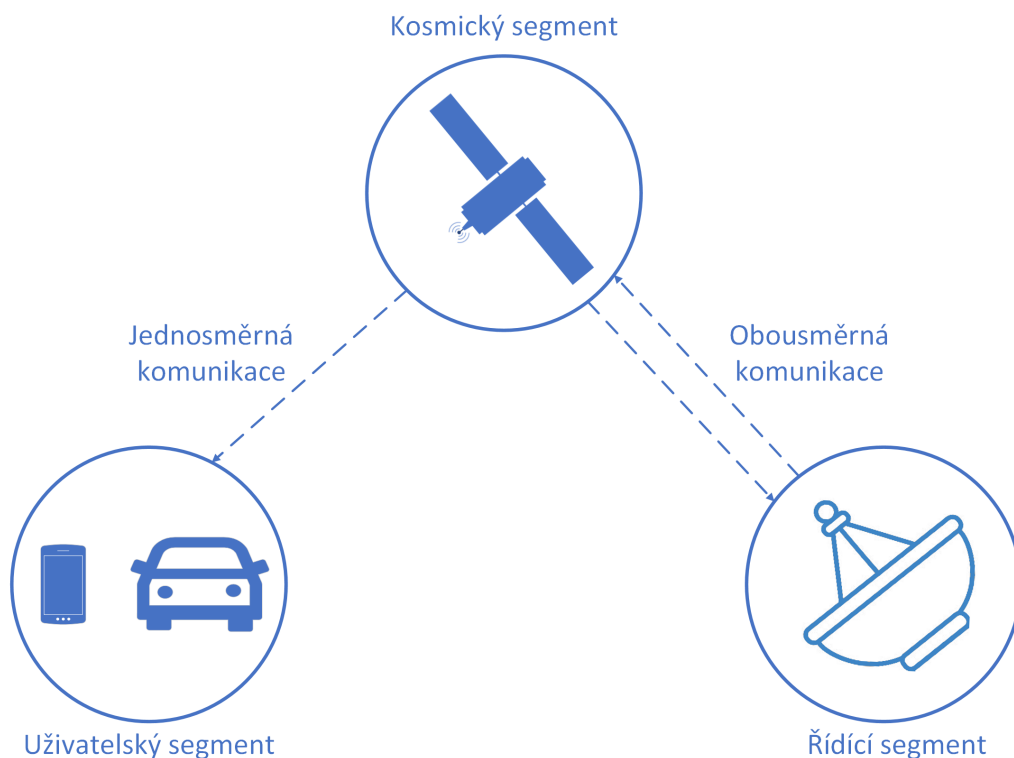


Obr. 2.1: Architektura C-ITS systémů

3 Global Positioning System

Global Positioning System (GPS), nebo-li globální polohový systém, je družicový systém vlastněný USA, který umožňuje určit přesnou polohu přijímače na povrchu Země. Skládá se ze tří segmentů: kosmický, řídicí a uživatelský, viz Obr. 3.1.

- **Kosmický segment** – je složen z více než 30 satelitů, které mají atomové hodiny (zdroj velmi přesného času důležitého pro určení polohy) a obíhají ve výšce 20 000 km nad povrchem Země.
- **Řídicí segment** – zahrnuje střediska pro kontrolu, řízení a údržbu kosmického segmentu. Střediska zpracovávají data od satelitů a vypočítávají korekce, které jsou zpětně odeslány do satelitů.
- **Uživatelský segment** – je tvořen přijímači, které získávají polohu a čas od satelitů a díky tomu vypočítávají vlastní polohu, nadmořskou výšku a rychlost. Získávají také aktuální datum a čas [2].

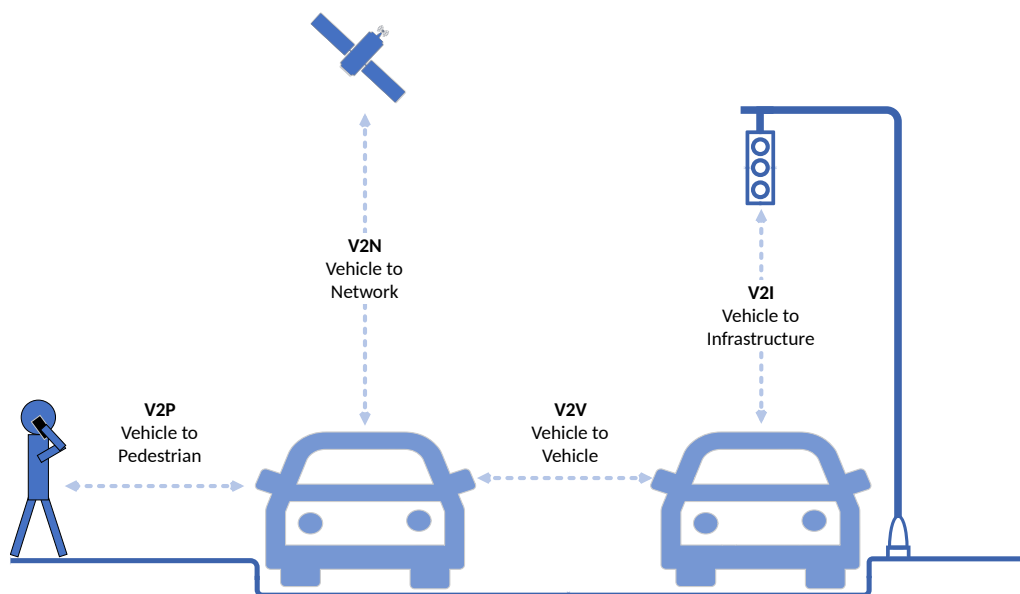


Obr. 3.1: GPS segmenty

4 Vehicle-to-Everything

Z názvu Vehicle-to-Everything (V2X) je zřejmé, že se jedná o proces, kdy jsou informace předávány z vozidla jinému objektu, který může ovlivnit vozidlo a naopak. Cílem je, aby vozidla komunikovala se všemi účastníky provozu v okolí (ostatní vozidla, chodci, infrastruktura). Lze to interpretovat i jako přiblížení k plně autonomnímu řízení. Příklady komunikace viz Obr. 4.1.

- **V2I – Vehicle-to-Infrastructure** je výměna informací mezi vozidlem a zařízeními podél vozovky nebo-li RSU. V2I se většinou používá na posílání informací o provozu a nouzových upozornění řidičům.
- **V2N – Vehicle-to-Network** je komunikace vozidla přistupujícího k síti pro cloudové služby.
- **V2V – Vehicle-to-Vehicle** je komunikace mezi vozidly navzájem. Na rozdíl od informací ze sensorů vozidla, může být zdroj informací V2V vzdálen několik set metrů daleko, za jinými vozidly nebo budovami.
- **V2P – Vehicle-to-Pedestrian** je komunikace mezi vozidlem a chodcem [3].



Obr. 4.1: Komunikace V2X, Převzato a upraveno z [3]

5 Roadside Unit

Roadside Unit (RSU), nebo-li jednotka na straně infrastruktury, je bezdrátové komunikační zařízení, které se nachází v blízkosti silnic a slouží jako prostředník v komunikaci mezi vozidly a infrastrukturou (další RSU a semaforey), viz Obr. 5.1.

RSU zpracovává data vyslaná vozidly (informace o rychlosti, poloze, směru jízdy a další). Tato data poté odesílá do centrálního střediska systému dopravy, kde mohou být použita pro řízení dopravy, semaforů, rychlosti vozidel a vylepšení bezpečnosti na silnici. Mohou také vysílat informace z centrálního střediska do vozidel (informace o dopravní situaci, nehodách, uzavírkách a další) [4][5].



Obr. 5.1: RSU od Cohda Wireless, Zdroj: [6]

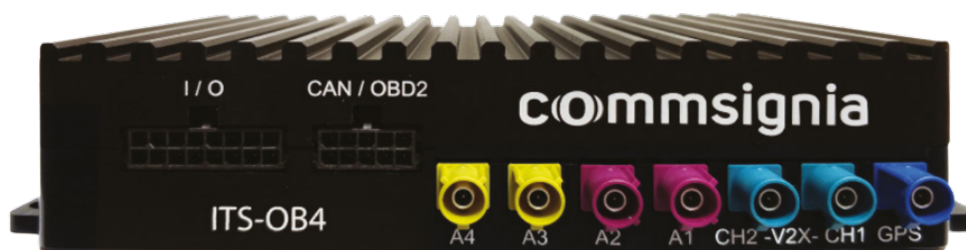
6 On-Board Unit

Palubní systém OBU je zařízení na nebo ve vozidle, které může odesílat a přijímat zprávy od jiných vozidel nebo infrastruktury. Například některé složky integrovaného záchranného systému OBU využívají pro přepnutí SSZ (Světelné Signalizační Zařízení, jinak také semafor) na zelenou ve směru jízdy a v ostatních směrech na červenou. To nejen snižuje riziko srážky v křižovatce, ale také umožňuje složkám dorazit na místo incidentu rychleji. Další možné využití je například pro Policii České republiky na zpomalení individuálního vozidla na dálku. Pro civilní vozidla je možné OBU využít na upozornění blížícího se vozidla IZS (Integrovaný Záchranný Systém) pro včasné uvolnění cesty nebo pro upozornění na blížící se vlak na železničním přejezdu se špatnou viditelností atd.

6.1 Komerční OBU

Palubní systémy mají mnoho využití. Mohou se používat pro placení mýtného na silnicích a dálnicích jako například v Belgii [7] a Maďarsku [8]. Mezi výrobce OBU jednotek patří Commsignia, Cohda Wireless, Unex a další. Jednotky od těchto tří výrobců jsou popsány dále v textu a jejich parametry porovnány v tabulce 6.1.

- **Commsignia ITS-OB4** – zástupce čtvrté generace palubních systémů od maďarské společnosti Commsignia, která mimo hardwaru nabízí i softwarové řešení komunikace V2X. Jednotka podporuje komunikační kanály DSRC (Dedicated short-range communications) / ETSI-G5 (European Telecommunication Standards Institute), C-V2X a LTE-V2X. Nabízí také rozhraní ETH (Ethernet), USB (Universal Serial Bus), CAN (Controller Area Network), OBD-II (On-Board Diagnostics), Wi-Fi a Bluetooth, viz Obr. 6.1. Součástí je i HSM (Hardware Security Module) modul a možnost ověřit a šifrovat komunikaci pomocí ECDSA (ECDSA – Elliptic Curve Digital Signature Algorithm) [9].



Obr. 6.1: Commsignia ITS-OB4, Zdroj: [9]

Tab. 6.1: Porovnání komerčních OBU

OBU	Commsignia ITS-OB4	Cohda MK6C EVK	Unex OBU-201E
CPU	800MHz Freescale/NXP i.MS 6	i.MX 8QXP	Autotalks CRATON V2X CPU, 3x 240MHz
RAM	2GB DDR3	-	32MB NOR, 128 MB DDR
Secure module (HSM)	SLI97	SXF1800 FIPS 140-2 level 3 compliant	Infineon SLE97
GNSS	Advanced GNSS	GPS/GLONASS/Galileo/Beidou	Telit SL869 (GPS, Glonass, Galileo, QZSS), 10 Hz
WiFi	Dual band a/b/g/n	802.11ac	802.11p
Ethernet	10/100/1000 Mbps	ano	1x 10/100 Mbps
Bluetooth	volitelně	Bluetooth 4.1 + BLE	-
CAN	ano	ano	2x 500kbps CAN 2.0b DE-9 (DB9)
Napájení	8 až 32V DC	8 až 24 V DC	6 až 32 V DC
OS	Linux/RTOS (V2X)	Linux 4.9.88	ThreadX RTOS

- **Cohda MK6C EVK** – malý modul od australské firmy Cohda Wireless. Je z nižší cenové kategorie, který podporuje rozhraní Wi-Fi, Bluetooth, ETH, CAN, GNSS (Global Navigation Satellite System) systémy GPS/GLONASS (Global Navigation Satellite System)/Galileo/Beidou a komunikaci C-V2X, viz Obr. 6.2. Vyhovuje bezpečnostní politice SXF1800 FIPS (Federal Information Processing Standards) 140–2 úrovně 3 [10].
- **Unex OBU-201E** – palubní jednotka od taiwanské firmy UNEX. Má tříjádrový procesor a operační systém ThreadX RTOS (Real-Time Operating System). Podporuje ETSI TC-ITS V2X protocol stacks, rozhraní CAN, ETH, RS232 (Recommended Standard 232) a GNSS systémy GPS a GLONASS, viz Obr. 6.3. Obsahuje i HSM modul a možnost zabezpečení pomocí ECDSA [11].



Obr. 6.2: Cohda MK6C EVK, Zdroj: [10]



Obr. 6.3: Unex OBU-201E, Zdroj: [11]

6.2 Linuxové OBU jednotky

V praktické části této práce se používají dvě Linuxové OBU jednotky, jedná se o kopie komerční OBU jednotky s operačním systémem založeným na operačním systému Linux. Pro účely této práce představují téměř plnohodnotnou náhradu komerčních OBU jednotek, která je plně dostačující. Linuxová OBU jednotka je cenově

dostupnější než komerční systémy a byla nakonfigurována na pracovišti ústavu telekomunikací pro testování nouzového zastavení vozidla a případně dalších funkcí.

Podporuje operační systém Linux Ubuntu a rozhraní ETH, RS-232 a USB, viz Obr. 6.4. OBU jednotka obsahuje čtyřjádrový procesor AMD GX-412TC SOC založen na architektuře X86-64. Disponuje také operační pamětí 3,8 GB. K uložení dat slouží SATA SSD disk s kapacitou 14 GB. Obsahuje také čtyři gigabitové ethernet porty od firmy Intel typu I211 a síťovou kartu Qualcomm Atheros AR928X.



Obr. 6.4: Linuxová OBU jednotka

7 Vanetza

Vanetza je open-source implementace sady protokolů ETSI C-ITS. Mezi jinými podporuje následující protokoly a funkce [12]:

- **GeoNetworking (GN)** – protokol síťové vrstvy, který poskytuje geografické adresování. Aplikace a zařízení navržené pro GeoNetworking využívají tyto funkce k šíření varovných nebo informačních zpráv do geograficky vymezených oblastí [13].
- **Basic Transport Protocol (BTP)** – poskytuje End-to-end transport informací bez připojení v ITS síti (hlavním účelem je multiplexování zpráv z různých zařízení na vrstvě ITS zařízení) [13].
- **Zabezpečení** – podporuje generování a zpracování bezpečnostních certifikátů.
- **Podpora zpráv typu ASN.1 (Abstract Syntax Notation One) např. CAM a DENM** – ASN.1 je formální zápis používaný k popisu dat přenášených telekomunikačními protokoly bez ohledu na implementovaný jazyk a fyzickou reprezentaci dat [14].
- **Decentralizované řízení přetížení DCC (Decentralized Congestion Control)** – povinná součást sady protokolů ITS-G5, která snižuje přetížení radiového kanálu, zamezuje rušení a zvyšuje dosah [15].

I když byla Vanetza navržena pouze pro provoz v kanálech ITS-G5 ve VANET (Vehicular Ad Hoc Network), může být kombinována s jinými komunikačními technologiemi.

7.1 Socktap

Socktap je aplikace demonstrující některé funkce knihovny Vanetza, ale nenahrazuje plnohodnotnou ITS-G5 stanici.

Dříve existovalo mnoho variant Socktapu jako samostatné spustitelné soubory. Dnes je Socktap samostatná aplikace, která se dá konfigurovat pomocí různých parametrů. Tato aplikace nahradila zastaralé spustitelné soubory.

Mnoho systémů ITS-G5 závisí na pozičních datech. Ideálně je k počítači připojen GPS přijímač, na kterém běží gpsd (služba monitorující jeden nebo více signálů GPS nebo AIS (Automatic Identification System) přijímačů) a Socktap. Když není dostupný GPS přijímač, tak je možné nastavit statickou polohu manuálně [16], to ale není z pohledu budoucího využití v terénu použitelné. Konfigurace aplikace se realizuje přes následující parametry:

- `--help` – nápověda pro všechny parametry
- `--interface=wlp5s0` – nastavení rozhraní, na kterém se budou provádět dané operace

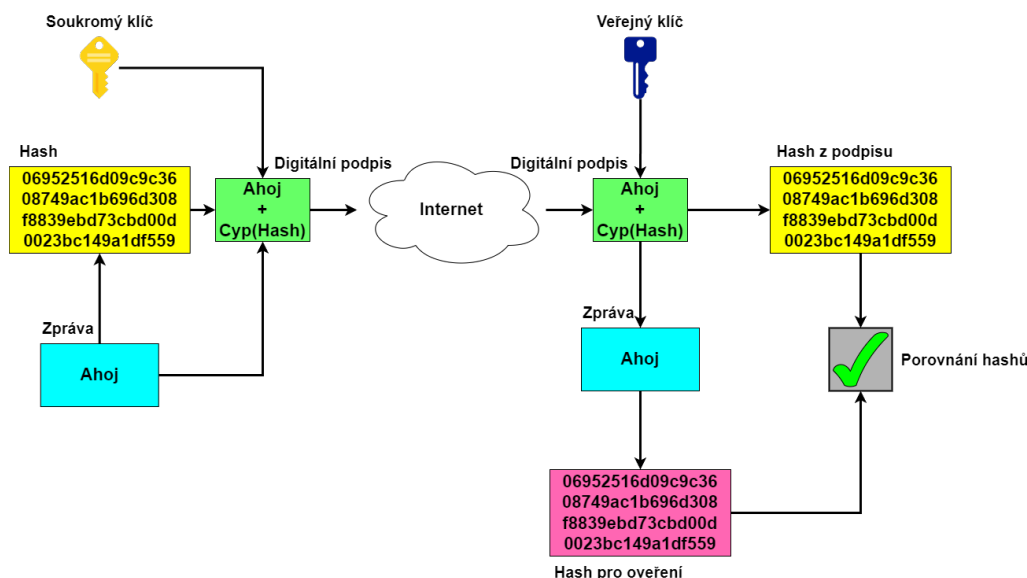
- `--security=certs/none` – vysílání podepsaných nebo nepodepsaných zpráv
- `--non-strict` – neověřené zprávy se také zpracují
- `--applications=ca/de/ivi/map/sr/ss` – pro jednotlivé služby
- `--tx-cams/tx-denms/tx-ivis` – pro vysílání testovacích zpráv
- `--print-rx-cam/print-rx-denm/print-rx-ivi` – výpis přijatých zpráv
- `--print-tx-cam/print-tx-denm/print-tx-ivi` – výpis vyslaných zpráv
- `--positioning=static` – fixní GPS pozice
- `--latitude=48.131...`, `--longitude=11.456...` – nastavení GPS pozice
- `--pin` – pin pro připojení k HSM modulu
- `--certificate=ticket.cert` – soubor s certifikátem
- `-certificate-key=at.key` – ID klíče v HSM
- `--certificate-chain`, `--trusted-certificate` – další soubory s certifikáty
- `--can-serial=/dev/ttyUSB0` – sériová linka, na kterou se mají poslat CAN zprávy

8 Zabezpečení

Komunikaci mezi OBU jednotkami je potřeba zabezpečit, aby se předešlo zneužití. Knihovna Vanetza, která je v této práci použita, umožňuje zabezpečení za použití infrastruktury pro správu a distribuci veřejných klíčů, ta vychází z principů asymetrické kryptografie.

V asymetrické kryptografii se data šifrují a dešifrují pomocí dvou propojených klíčů. Jeden klíč je soukromý a je potřeba jej držet v tajnosti, druhý je veřejně dostupný. Když chce někdo poslat zprávu držiteli soukromého klíče, tak zprávu zašifruje jeho veřejným klíčem (dešifrovat ho lze jen za využití soukromého klíče). Soukromý klíč nelze zjistit z veřejného klíče.

Díky asymetrické kryptografii lze také vytvářet digitální podpisy. Nejdříve se ze zprávy, kterou chceme podepsat vytvoří hash. Hash je sled symbolů a čísel pevně dané délky, který je výsledkem hashovací funkce. Hashovací funkce z libovolně dlouhého vstupu vytvoří hash předem dané délky. Z něj nelze zjistit, co bylo na vstupu, ale ze stejného vstupu bude vždy stejný hash. Hash se zašifruje soukromým klíčem podepisovatele a připojí se ke zprávě. Příjemce dešifruje hash pomocí veřejného klíče podepisovatele a vytvoří hash z přijaté zprávy. Poté oba porovná, jestli se shodují, viz Obr. 8.1. Pokud ano, může si být příjemce jistý autenticitou podepisovatele a integritou zprávy [17].



Obr. 8.1: Schéma digitálního podpisu

8.1 Infrastruktura pro správu a distribuci veřejných klíčů

Infrastruktura pro správu a distribuci veřejných klíčů (PKI) je systém pro vytváření, uchovávání a distribuci digitálních certifikátů, které ověřují, že daný veřejný klíč patří danému subjektu. PKI vytváří digitální certifikáty, které přiřazují veřejné klíče subjektům, bezpečně je uchovává a když je potřeba, tak je ruší [18]. PKI obsahuje tyto prvky:

- **Certifikační autorita** – vytváří, ukládá a podepisuje digitální certifikáty
- **Registrační autorita** – ověřuje totožnost subjektů, které požadují vytvoření nebo uložení digitálních certifikátů
- **Databáze Certifikátů** – bezpečné úložiště, kde jsou uloženy digitální certifikáty a informace o nich (především doba platnosti)
- **Certifikační politika** – udává požadavky PKI na její postupy, umožňuje lidem zvenčí určit důvěryhodnost PKI [19]

8.2 Certify

Certify je nástroj pro vytváření a zobrazení certifikátů, který je součástí knihovny Vanetza. Je možné ho použít pro vytvoření testovací infrastruktury pro správu a distribuci veřejných klíčů v V2X komunikaci. Aby byl nástroj dostupný, tak je potřeba v instalaci Vanetzi při sestavování pomocí `cmake` použít parametr `-D BUILD_CERTIFY=ON`.

Certify podporuje následující příkazy:

- **generate-key** – vytváření nových soukromých klíčů (`certify generate-key private.key`)
- **extract-public-key** – získání veřejného klíče ze soukromého klíče (`certify extract-public-key -private-key private.key public.pub`)
- **generate-root** – vytvoření root certifikátu (`certify generate-root -subject-key private.key root.cert`)
- **generate-aa** – vytvoření autentizační autority (`certify generate-aa -sign-key private.key -sign-cert root.cert -subject-key subject.key`)
- **generate-ticket** – vytvoření autorizačního ticketu (`certify generate-ticket -sign-key aa.key -sign-cert aa.cert -subject-key ticket.key ticket.cert`)
- **show-certificate** – zobrazení informací o certifikátu, např. kdo ho vydal, podepsal a dobu platnosti (`certify show-certificate root.cert`) [20]

9 Sada Standardů 802.11

IEEE (Institute of Electrical and Electronics Engineers, česky Institut pro elektrotechnické a elektronické inženýrství) 802.11 je sada standardů pro bezdrátovou komunikaci v lokálních sítích WLAN (Wireless Local Area Network, česky bezdrátová lokální síť). Je vyvíjena mezinárodní neziskovou organizací IEEE, která má nejvíce členů technické profese ve světě. Všechny specifikace používají technologii Ethernet a CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) pro sdílení cesty. CSMA/CA je protokol, který před začátkem vysílání poslouchá, jestli je přenosové medium volné, pokud ne, tak čeká než se medium uvolní. Označení 802.11x se používá pro novější doplňky k původnímu standardu IEEE 802.11. Tato sada obsahuje několik specifikací a příležitostně se přidávají další [21]. Pro potřeby této práce nás nejvíce zajímá doplněk 802.11p, proto další standardy nebudou podrobněji popsány.

9.1 802.11p

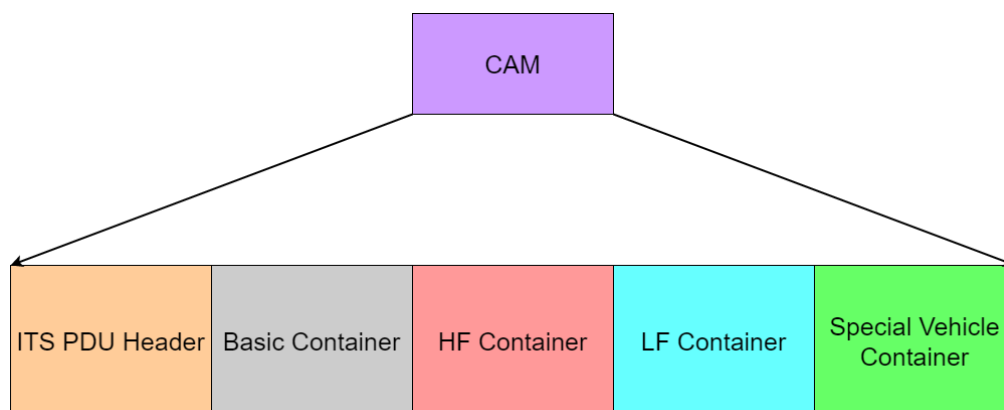
Standard 802.11p je doplněk, který přidává WAVE (Wireless Access in Vehicular Environments - prostředí poskytující efektivní a spolehlivou radiokomunikaci pro aplikace na zlepšení bezpečnosti a pohodlí systémů ITS) a obohacuje standard 802.11 o možnost podpory ITS aplikací. To zahrnuje výměnu dat mezi rychle jedoucími vozidly a ITS infrastrukturou. Komunikace probíhá ve frekvenčním pásmu 5,850–5,925 GHz. Vývoj standardu IEEE 802.11p byl také částečně přisuzován pokroku v DSRC, který umožňuje poskytovat komunikační služby mezi vozidly a silniční infrastrukturou (V2V a V2I). Primárním účelem je zaručit bezpečnost a lepší řízení silničního provozu [22].

10 Typy zpráv

V komunikaci s OBU jednotkami se může objevit více typů zpráv. Mezi nejčastější patří CAM (Cooperative Awareness Messages), IVIM (Infrastructure to Vehicle Information Messages) a DENM (Decentralized Environmental Notification Message). Zprávy CAM se většinou objevují v komunikaci mezi vozidly a obsahují informace o tom, kde se vysílající vozidlo nachází a v jakém jede směru. IVIM zprávy lze nalézt v komunikaci mezi vozidly a infrastrukturou. Informují o značkách a pracích na silnici. DENM zprávy upozorňují na zvláštní podmínky na silnici.

10.1 Cooperative Awareness Messages

Standard CAM je definován ETSI pro přenášení pozičních dat dalším vozidlům. Zprávy CAM jsou vysílány periodicky a obsahují základní informace jako poloha, zrychlení, rychlost, ID a směr. Jsou definovány čtyři profily v závislosti na typu ITS stanice: Vozidlo, Infrastruktura, Záchrané vozidlo a Veřejná doprava. CAM zprávy mohou obsahovat další informace v závislosti na profilu, například status sirény u záchraných vozidel [23]. Struktura CAM zprávy viz Obr. 10.1.

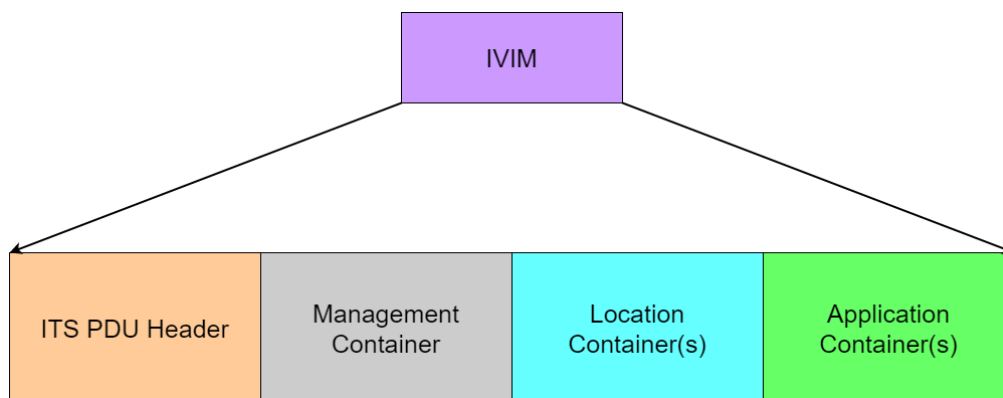


Obr. 10.1: Zpráva CAM, Převzato a upraveno z [24]

- **ITS PDU Header** – upřesňuje verzi protokolu, typ zprávy a ID stanice
- **Basic Container** – obsahuje typ stanice, pozici vozidla v době generace CAM zprávy
- **HF (High-Frequency) Container** – označuje dynamické nebo rychle se měnící informace o vozidle
- **LF (Low-Frequency) Container** – označuje statické nebo pomalu se měnící informace o vozidle
- **Special vehicle container** – obsahuje přídatné informace na základě typu vozidla specifikovaném v LF Container

10.2 Infrastructure to Vehicle Information Messages

Infrastructure to Vehicle Information Messages jsou zprávy, které poskytují informace o statických a dynamických dopravních značkách. Mohou také varovat před pracemi na silnici, upozornit na maximální povolenou rychlost nebo zdali se vozidlo nachází ve speciální zóně. IVIM komunikace probíhá mezi RSU nebo ITS stanicí podél silnice a OBU jednotkou ve vozidle. Mohou přenášet obrazové a textové informace [25]. Struktura IVIM zprávy viz Obr. 10.2.

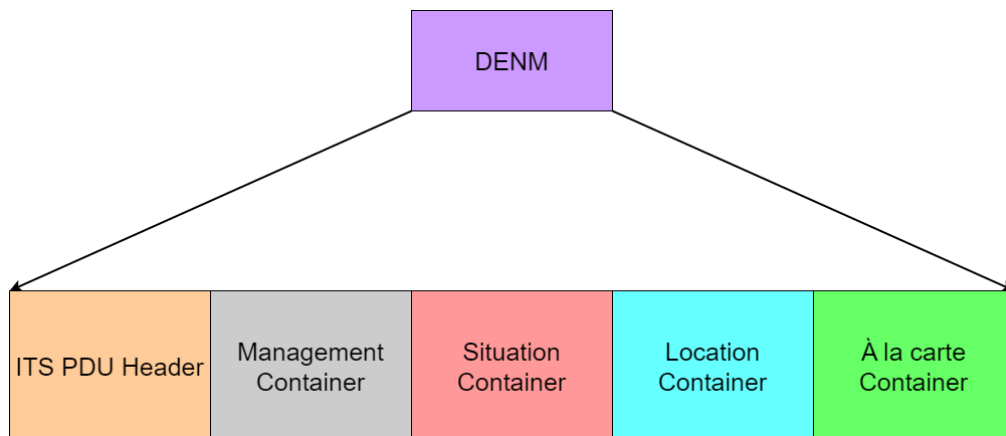


Obr. 10.2: Zpráva IVIM, Převzato a upraveno z [24]

- **ITS PDU Header** – upřesňuje verzi protokolu, typ zprávy a ID stanice
- **Management Container** – obsahuje obecné informace o IVIM zprávě které pomohou vozidlům rozhodnout, jestli chtějí zprávu přijmout
- **Location Container** – informuje o zónách kde je IVIM zpráva platná
- **Application Container** – obsahuje další užitečné informace

10.3 Decentralized Environmental Notification Message

Zprávy DENM se používají k informování ostatních účastníků provozu o zvláštních podmínkách a událostech na silnici, které mohou představovat nebezpečí pro řidiče. Zprávy jsou odeslány pouze při detekci takové události a poté jsou pravidelně aktualizovány pokud přetrvávají [26]. Struktura DENM zprávy viz Obr. 10.3.

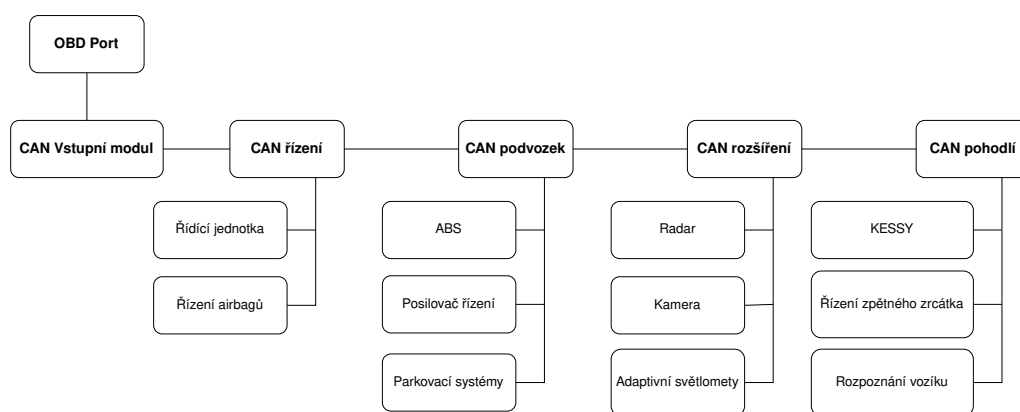


Obr. 10.3: Zpráva DENM, Převzato a upraveno z [24]

- **ITS PDU Header** – upřesňuje verzi protokolu, typ zpráv a ID stanice
- **Management Container** – obsahuje informace související s řízením DENM zprávy
- **Situation Container** – popisuje detekovanou událost
- **Location Container** – obsahuje pozici události
- **À la carte Container** – obsahuje další užitečné informace, které nejsou v ostatních kontejnerech

11 Controller Area Network

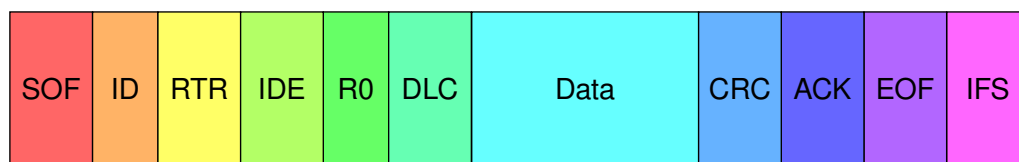
CAN bus je sériová sběrnice, která se využívá v automobilovém, leteckém i lodním průmyslu, ale také ve výtazích nebo zdravotnických přístrojích. Umožňuje spolehlivou komunikaci mezi různými částmi vozidla a oproti předchozím řešením nepotřebuje tolik vodičů, což snižuje náklady i hmotnost systému. CAN rámce jsou upřednostňovány podle ID, kdy důležitější data mají přednost před méně důležitými. Systém je vysoce odolný vůči elektrickému i elektromagnetickému rušení, což ho činí vhodným k aplikaci v odvětvích s nutností vysoké spolehlivosti (automobily 11.1, letadla, lékařství).



Obr. 11.1: Schéma CAN bus v běžném automobilu

Zařízení připojené do CAN bus jsou nazývány uzly. Každý uzel má procesor, CAN controller a transceiver (zařízení, které dokáže přijímat i vysílat). Všechny uzly mohou odesílat a přijímat data, ale ne ve stejnou chvíli. Uzly nemohou komunikovat přímo, místo toho posílají data do sítě, kde jsou dostupná každému uzlu, kterému byla určena. Rozdělení CAN zprávy viz Obr. 11.2. Části CAN zprávy jsou popsány v Tab. 11.1. Všechna data jsou posílána v jednom ze čtyř typů rámců [27]:

- **Datové rámce** – přenášejí data jiným uzlům
- **Vzdálené rámce** – žádají o data od jiných uzlů
- **Chybové rámce** – hlásí chyby
- **Rámce přetížení** – hlásí stav přetížení



Obr. 11.2: Rámec CAN zprávy

Tab. 11.1: Popis polí v CAN rámci

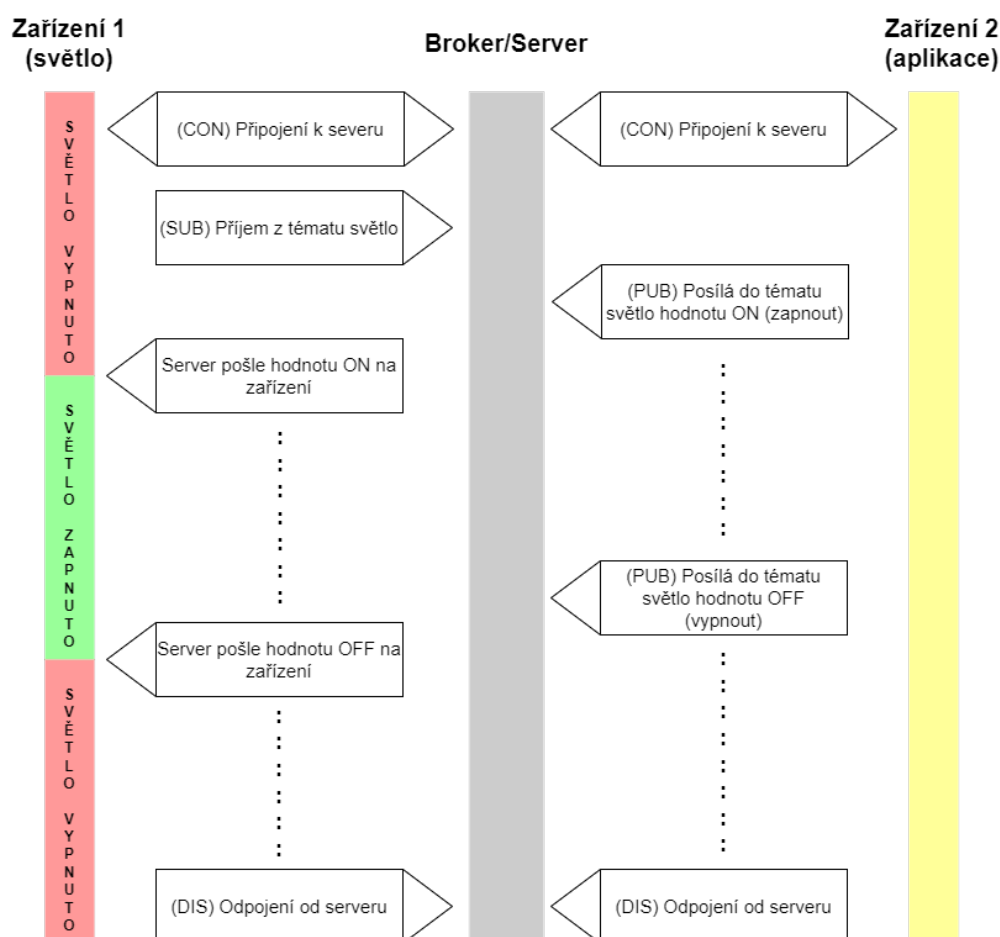
Název Pole	Délka [bit]	Popis
SOF	1	Start Of Frame, tento bit označuje začátek zprávy a synchronizuje uzly po období nečinnosti
ID	11	identifikátor, který udává důležitost zprávy, nižší číslo znamená vyšší prioritu
RTR	1	Remote Transmission Request, tento bit je dominantní, když informace vyžaduje jiný uzel
IDE	1	The Identifier Extension bit, označuje, zda se přenáší standardní nebo rozšířené ID
R0	1	rezervováno pro použití v budoucnosti
DLC	4	The Data Length Code, obsahuje počet přenášených bajtů
Data	0 - 64	přenášená data
CRC	16	Cyclic Redundancy Check, obsahuje kontrolní součet pro kontrolu chyb v přenosu
ACK	2	když uzel úspěšně přijme zprávu, potvrdí ji přepsáním na dominantní bit, pokud uzel najde chybu ve zprávě, umožní bitu zůstat recesivní a zprávu ignoruje
EOF	7	End Of Frame označuje konec každého CAN rámce
IFS	3+	Inter Frame Space, čas který controller potřebuje na posunutí zprávy do bufferu

12 MQ Telemetry Transport

Message Queuing Telemetry Transport (MQTT) je protokol pro jednoduchou komunikaci. Je navržen tak, aby byl použitelný i v sítích s malou šířkou pásma a v zařízeních s omezenou výpočetní kapacitou, proto se hojně využívá v Internet of Things aplikacích (chytré domácnosti, chytrá města).

Komunikace funguje na principu témat. Témata se oddělují lomítkem a většinou mají popisný název, např. `house1/floor2/room4/table/lamp` by označovalo lampu na stole ve čtvrtém pokoji ve druhém patře prvního domu.

- **Zařízení** – mohou přijímat (subscribe) informace z témat nebo posílat (publish) do tématu.
- **Broker/Server** – broker, nebo také server přijímá všechny zprávy a přeposílá je všem zařízením, která přijímají dané téma.



Obr. 12.1: Příklad MQTT komunikace

12.1 Knihovna Mosquitto

Mosquitto je open–source knihovna která implementuje protokol MQTT. Knihovna poskytuje následující funkce a vlastnosti:

- **Klientské a serverové funkce** – umožňují vytváření MQTT klientů pro odesílání a příjem zpráv a také serverů pro příjem a distribuci zpráv
- **Různé typy spojení** – podporované jsou TCP, TLS, WebSocket a další
- **Autentizace a autorizace** – podporuje možnosti autentizace klientů a autorizace přístupu ke konkrétním tématům na základě konfigurovaných pravidel
- **Publikování a odběr zpráv** – klienti mohou odebírat zprávy z určených témat a také publikovat zprávy na specifická témata
- **Podpora různých úrovní a správa QoS** – umožňuje nastavovat úroveň QoS pro odesílané a přijímané zprávy
- **Multiplatformní** – je k dispozici pro různé operační systémy, včetně Linuxu, Windows, macOS a dalších

Na Linuxových distribucích lze nainstalovat knihovnu pomocí:

```
sudo apt-get install mosquitto
```

Odebírat zprávy z určitého tématu lze pomocí:

```
mosquitto_sub -t '/test/prvniTest'
```

kde text v uvozovkách udává téma.

Publikovat zprávy do daného tématu lze pomocí:

```
mosquitto_pub -t '/test/prvniTest' -m 'První test'
```

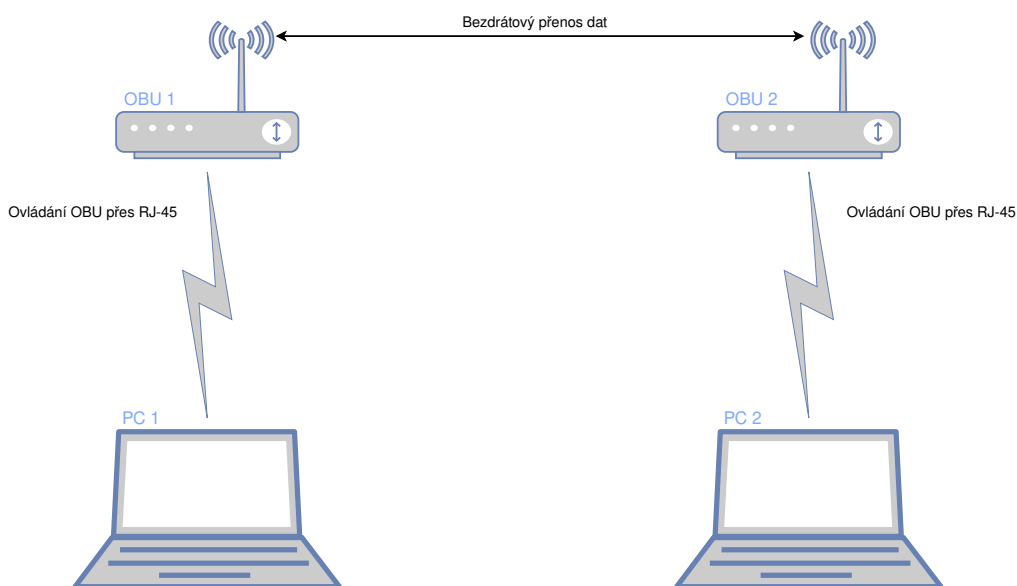
kde text v prvních uvozovkách udává téma a text ve druhých uvozovkách publikovanou zprávu.

13 Výsledky studentské práce

Cílem praktické části bylo vytvořit základní komunikaci mezi dvěma OBU jednotkami a otestovat správnost komunikace. Před samotným zapojením pracoviště bylo potřeba OBU jednotky nakonfigurovat do základního stavu, jak je uvedeno v kapitole 13.1.1. V rámci semestrální práce bylo pracováno s fixními hodnotami pro GPS, ale v rámci bakalářské práce už byl připojen GPS modul pro nalezení aktuální polohy.

13.1 Schéma zapojení základního pracoviště pro vzdálené ovlivnění rychlosti vozidla

Každá OBU jednotka je připojena do počítače pomocí síťového konektoru RJ-45, kde lze zobrazovat zachycené pakety, nastavovat parametry vysílání a příjmu paketů. Jednotky OBU spolu komunikují bezdrátově pomocí Wi-Fi 802.11p. Schéma zapojení viz Obr. 13.1.



Obr. 13.1: Schéma zapojení základního pracoviště

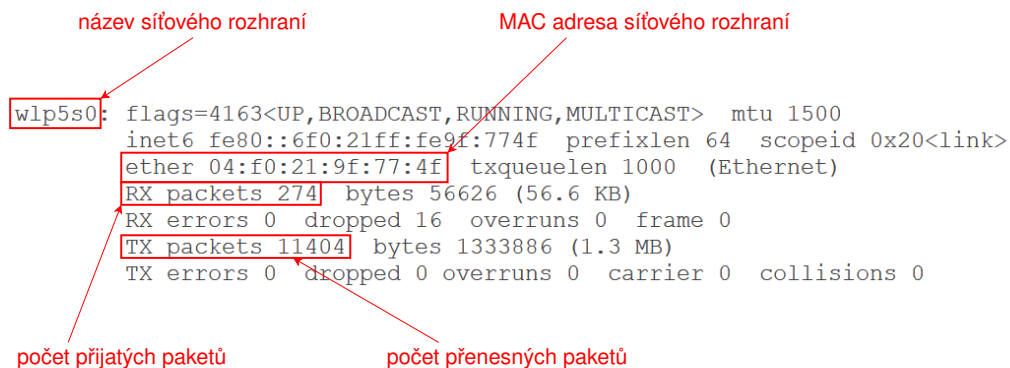
13.1.1 Základní nastavení OBU jednotky

Základní nastavení OBU jednotek proběhlo podle návodů v repozitáři eloryks-docs dostupném na gitlabu [28]. Pro nastavení podle návodů je potřeba přístup k internetu. Před počátečním zprovozněním komunikace bylo nutné poupravit nastavení dodaných jednotek stylem popsaným níže.

- **Partition resize** – Linuxová OBU jednotka má nativní datový oddíl nastavený na 8 GB, většina komerčních OBU jednotek má nastaveno 16 GB.
- **AR9280 mods** – potřeba nastavit SSH klíče v gitlabu a OBU jednotkách, nainstalovat M2crypto, wireless-regdb a crda z větve pro čipovou sadu AR9280.
- **Vanetza installation** – instalace Vanetza a Socktap, při sestavování pomocí cmake je potřeba přidat parametr `-DSOCKETAP_WITH_GPSD=ON`, aby bylo možné získávat informace o pozici z GPS přijímače.

13.1.2 Základní komunikace mezi OBU

Po základním nastavení je potřeba jednotky přepnout do OCB (Out of Base Context) módu, aby fungovala komunikace bez přihlášení do sítě. To lze provést spuštěním skriptu v souboru `osb.sh`, při spuštění se nastaví frekvence kanálu na 5900 MHz a šířka kanálu na 10 MHz. Díky příkazu `ifconfig` jsme zjistili název našeho bezdrátového rozhraní, viz Obr. 13.2. Pak můžeme, spuštěním následujícího příkazu na první jednotce, začít posílat pakety.



Obr. 13.2: Síťové rozhraní vysílající OBU jednotky

```
sudo socktap --interface=wlp5s0 --security=none --applications=ca
        --positioning=static --latitude=48.131231 --longitude=11.45645464
```

Tento příkaz začne vysílat CAM (Cooperative Awareness Messages) z rozhraní `wlp5s0` s informacemi o poloze a bez zabezpečení (přenos není šifrovaný). Na druhé jednotce zadáme příkaz na zachytávání z rozhraní `wlp5s0` (jediné bezdrátové rozhraní na jednotce).

```
sudo tcpdump -i wlp5s0 -v
```

Druhá OBU jednotka začne vypisovat obsah přijatých paketů z rozhraní `wlp5s0` do konzole, viz Obr. 13.3.

```

igor@c2x:~$ sudo tcpdump -i wlp5s0 -v
tcpdump: listening on wlp5s0, link-type EN10MB (Ethernet), capture size 262144 byte
09:48:04.655765 GeoNet [src:04:f0:21:9f:77:4f] (oui Unknown) v:1 NH:1-BTP-A HT:0-0-A
ny HopLim:128 Payload:8272 GN ADDR:00:2d:01:00:80:00:04:f0 [lat:-459223037] [lon:48131
2310]; BTP Dst:0 Src:0; ItsPduHeader v:7 t:209-unknown (209)
0x0000: 07d1 0000 0202 0000 0001 d019 005a 4aa4 .....ZJ.
0x0010: e6ce 43bd f8df ffff fc23 b774 3e00 0001 ..C.....#.t>...
0x0020: 2000 003f eled 0403 ffe3 fff4 00 ....?.....

```

Obr. 13.3: Výpis paketů do konzole

Změna výpisu

Příkaz z Obr. 13.4 vypisuje příchozí pakety z rozhraní wlp5s0, zobrazí podrobnosti paketu (parametr -V) a jen vybrané pole (parametr -T fields a mezi názvy polí -e).

```

igor@c2x:~$ sudo tshark -i wlp5s0 -v -T fields -e frame.number -e frame.time_delta -e eth.src -e eth.dst -e geonw.src_pos.long -e geonw
.src_pos.lat -e frame.len
Running as user "root" and group "root". This could be dangerous.
Capturing on 'wlp5s0'
1 0.000000000 04:f0:21:9f:77:4f ff:ff:ff:ff:ff:ff 114564546 481312310 99
2 1.000497572 04:f0:21:9f:77:4f ff:ff:ff:ff:ff:ff 114564546 481312310 99
3 1.000220518 04:f0:21:9f:77:4f ff:ff:ff:ff:ff:ff 114564546 481312310 99
4 1.000723947 04:f0:21:9f:77:4f ff:ff:ff:ff:ff:ff 114564546 481312310 99
5 1.000330075 04:f0:21:9f:77:4f ff:ff:ff:ff:ff:ff 114564546 481312310 99
6 1.000381009 04:f0:21:9f:77:4f ff:ff:ff:ff:ff:ff 114564546 481312310 99
7 1.000564715 04:f0:21:9f:77:4f ff:ff:ff:ff:ff:ff 114564546 481312310 99
8 1.000487616 04:f0:21:9f:77:4f ff:ff:ff:ff:ff:ff 114564546 481312310 99
9 1.000520534 04:f0:21:9f:77:4f ff:ff:ff:ff:ff:ff 114564546 481312310 99
29 packets captured

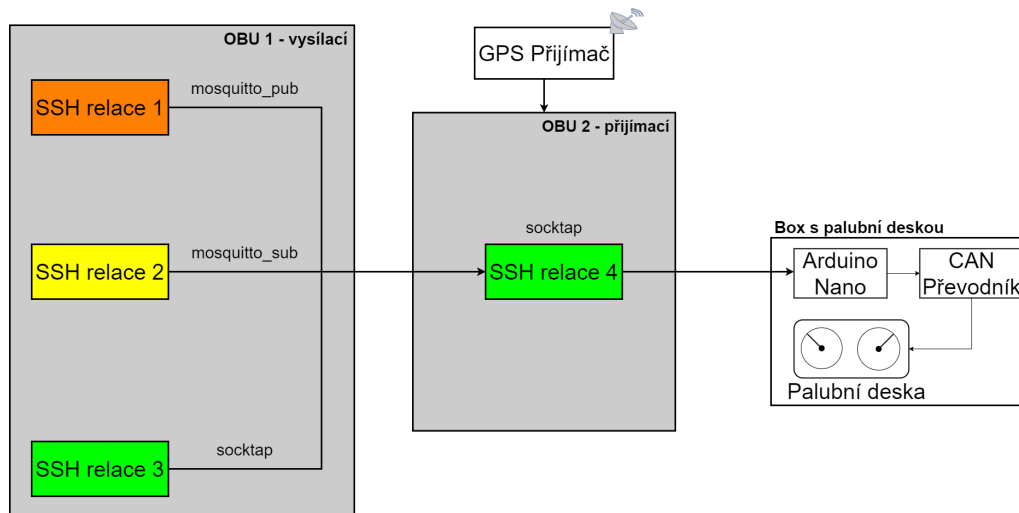
```

Obr. 13.4: Výpis vybraných informací z paketů

13.1.3 GPS modul

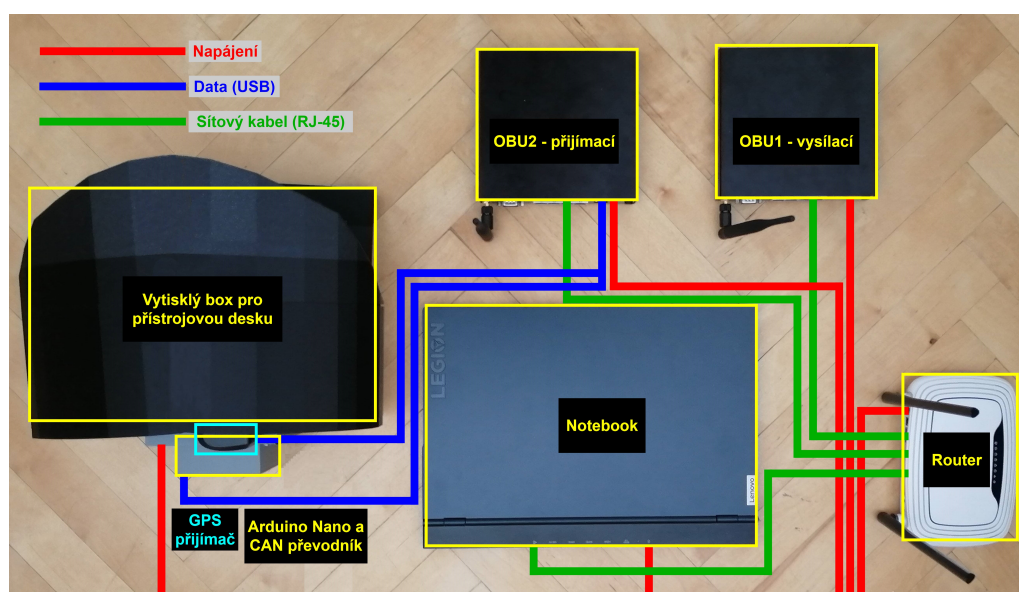
Jelikož Linuxové OBU jednotky použité v této práci nemají zabudovaný GPS přijímač, musí se připojit externí. Pro tuto práci byl použit GPS přijímač VK-162 G-Mouse, viz Obr. 13.5. GPS přijímač je použit, aby data posílaná z OBU měla aktuální polohu zařízení, místo pevně daných souřadnic. Tento přijímač posílá data v NMEA formátu. Tato data lze zobrazit buď přímo ze vstupu pomocí nastavení linky a čtení z ní, nebo pomocí speciální aplikace (cgps):

- `sudo stty -F /dev/ttyACM0 ispeed 4800` – `ttyACM0` je vstupní zařízení (GPS přijímač) a pomocí `ispeed` se nastaví rychlost vstupu a výstupu na 4800 Baudů.
- `cat </dev/ttyACM0` – čtení dat přímo z přijímače, viz Obr. 13.6.
- `sudo apt-get install gpsd-clients` – instalace aplikace `cgps`, která přehledně zobrazuje informace z přijímače.
- `cgps` – samotné spuštění aplikace, viz Obr. 13.7.



Obr. 13.9: Návrh pracoviště na omezení otáček

vodník mění zprávy od Arduina tak, aby byly vhodné pro CAN sběrnici. Palubní deska zobrazuje otáčky motoru, rychlost vozidla, stavové kontrolky a multifunkční displej, zapojení pracoviště viz Obr. 13.10.

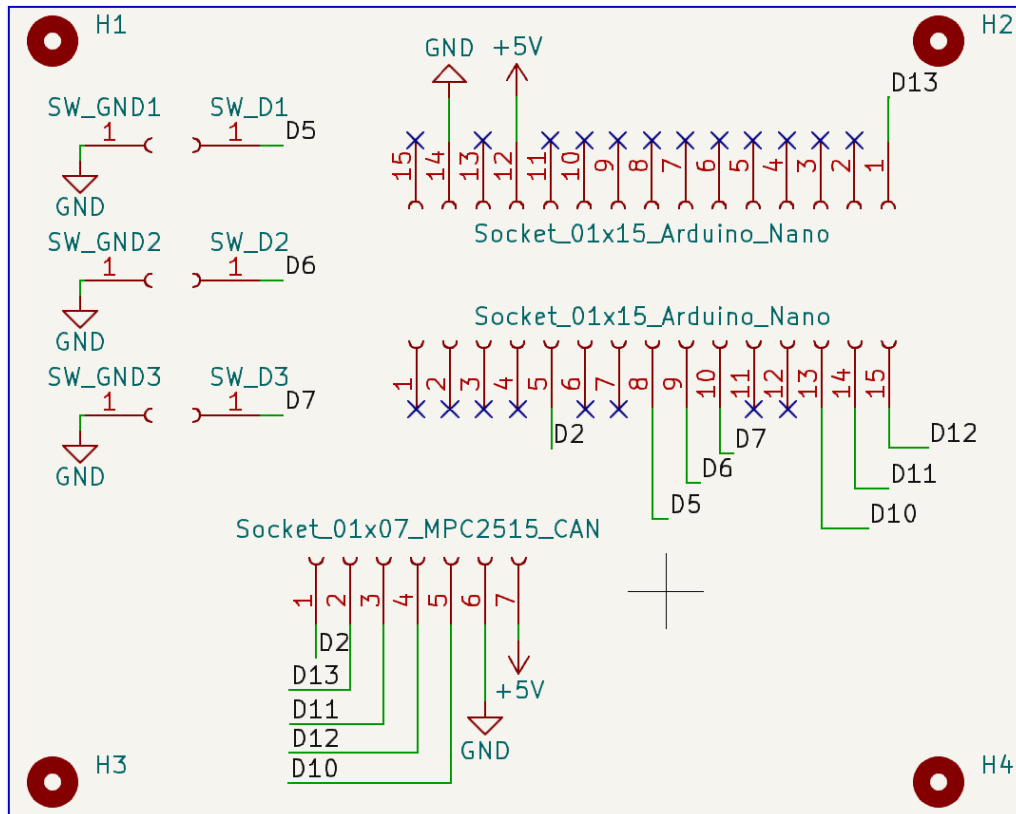


Obr. 13.10: Zapojení pracoviště na omezení otáček

13.2.1 Návrh plošného spoje

Plošný spoj nahradil původní nepájivé pole pro větší mechanickou odolnost, zamezení oxidace kontaktů a zmenšení rozměrů.

Návrh plošného spoje probíhal v softwaru KiCad, který je přehledný a obsahuje vše potřebné pro návrh desek plošných spojů. Nejdříve bylo potřeba vytvořit schématický návrh, který obsahuje všechny součástky a jejich propojení, viz Obr. 13.11. Poté bylo potřeba součástkám přiřadit správná pouzdra (konkrétní typ a vzhled součástky).



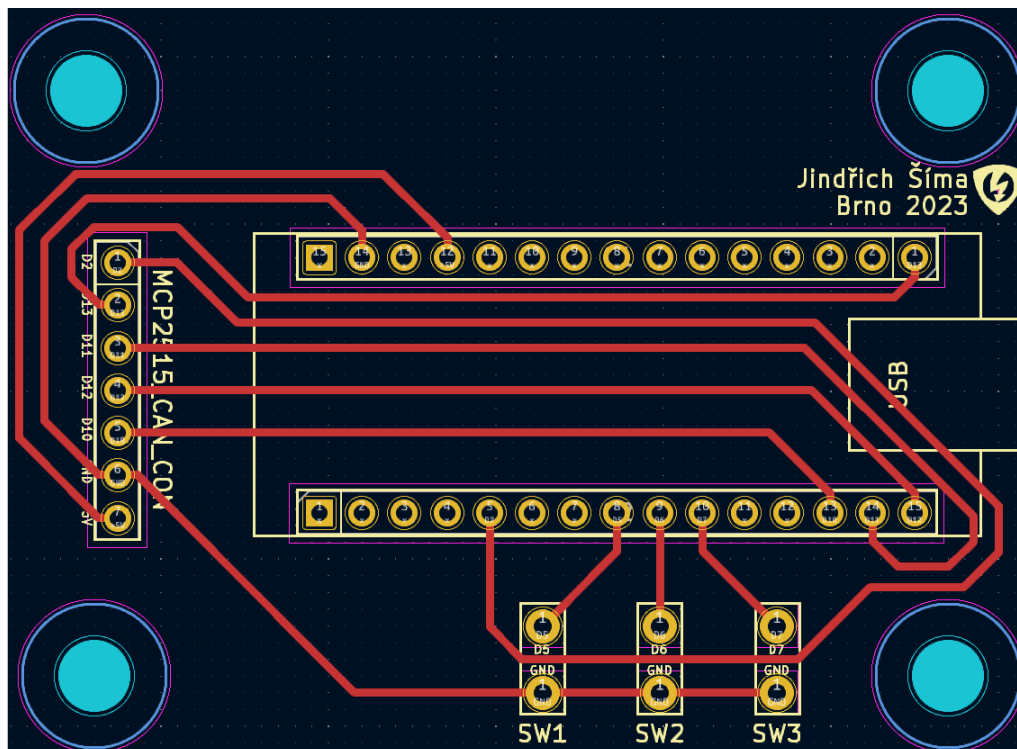
Obr. 13.11: Schéma plošného spoje v KiCadu

Dále bylo možné přejít k návrhu plošného spoje, kde se udává přesná poloha součástek na plošném spoji. Plošný spoj obsahuje zdířky pro lišty, kde bude umístěno Arduino Nano a CAN převodník. Je zde i šest zdířek pro vodiče které povedou ke třem tlačítkům. Má také čtyři otvory o průměru 4 mm pro upevnění v boxu, viz Obr. 13.12.

Samotná výroba plošného spoje proběhla pomocí externí firmy, které byly poslány všechny podklady a zpět přišla hotová deska. Na ni byly napájeny lišty pro Arduino Nano a CAN převodník, ale i vodiče vedoucí k tlačítkům.

13.2.2 Arduino Nano

V této práci se používá Arduino Nano, což jedna z mnoha Arduino desek, které nabízí plnohodnotné vývojové prostředí. Mezi jeho hlavní vlastnosti a funkce patří:

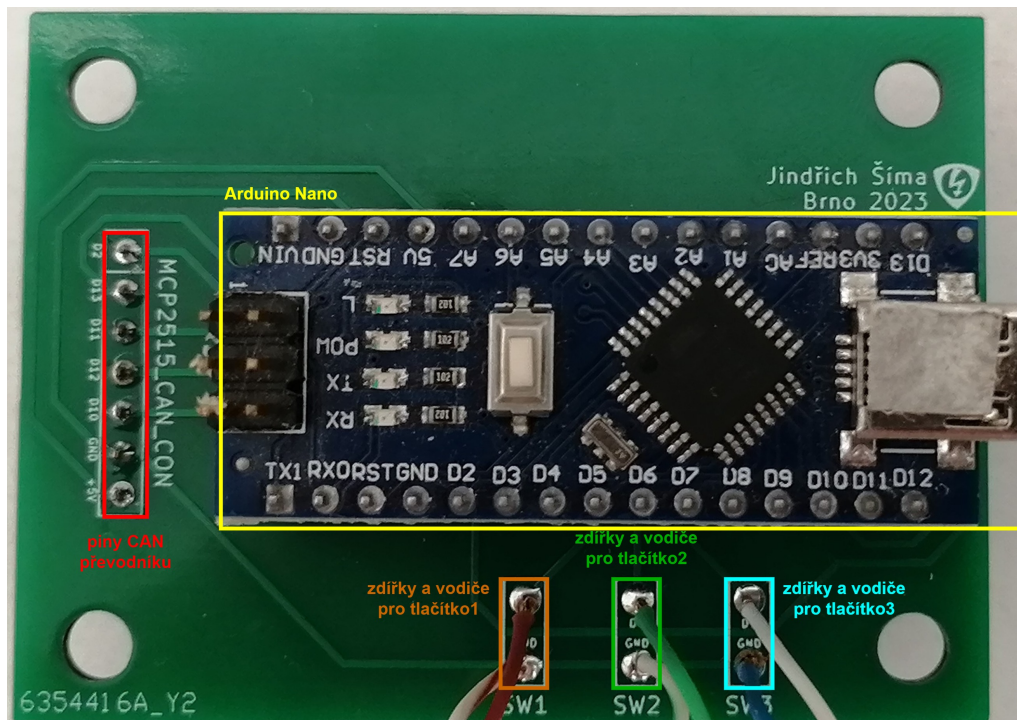


Obr. 13.12: Návrh plošného spoje v KiCadu

- **Velikost** – jak už přívlástek Nano napovídá, je tato deska s velikostí 18×45 mm menší než většina ostatních.
- **Mikrokontrolér** – je vybaveno mikrokontrolérem ATmega328 založený na architektuře AVR a s frekvencí 16 MHz. Mikrokontrolér má také 32 kB paměti Flash, 1 kB EEPROM a 2 kB SRAM.
- **Napájení** – může být napájeno různými způsoby: přes USB konektor nebo adaptérem se vstupním napětím v rozmezí 7-12 V, provozní napětí je 5 V.
- **Digitální vstupy/výstupy** – tato deska má 14 digitálních pinů, které umožňují připojení dalších senzorů a periférií pro komunikaci s jinými zařízeními.
- **Programování** – deska je programovatelná pomocí vývojového prostředí Arduino IDE. Pro programování je potřeba připojit Arduino Nano pomocí USB kabelu k počítači [29], dále nainstalovat knihovnu `autowp-mcp2515` (ta umožňuje CAN funkce). Knihovnu lze nainstalovat přímo v Arduino IDE v záložce `Tools` → `Manage libraries` a poté stačí vyhledat název knihovny a kliknout na `Install`. Je také dostupná na githubu [30].

Na desku byl nahrán kód z elektronické přílohy, který periodicky odesílá zprávy na CAN převodník a simuluje tím funkci auta (rychlost, otáčky motoru, čas, najetou vzdálenost a hodnoty pro kontrolky). Čeká také na data poslaná přes sériovou linku od OBU jednotky, která omezují otáčky.

Arduino Nano je v pracovišti umístěno na desce plošných spojů, kde je propojeno se třemi tlačítky a CAN převodníkem. Každé tlačítko je propojeno se zemí (GND pin Arduino) a s rozdílným digitálním pinem (D5-D7), viz Obr. 13.13.



Obr. 13.13: Arduino Nano na plošném spoji

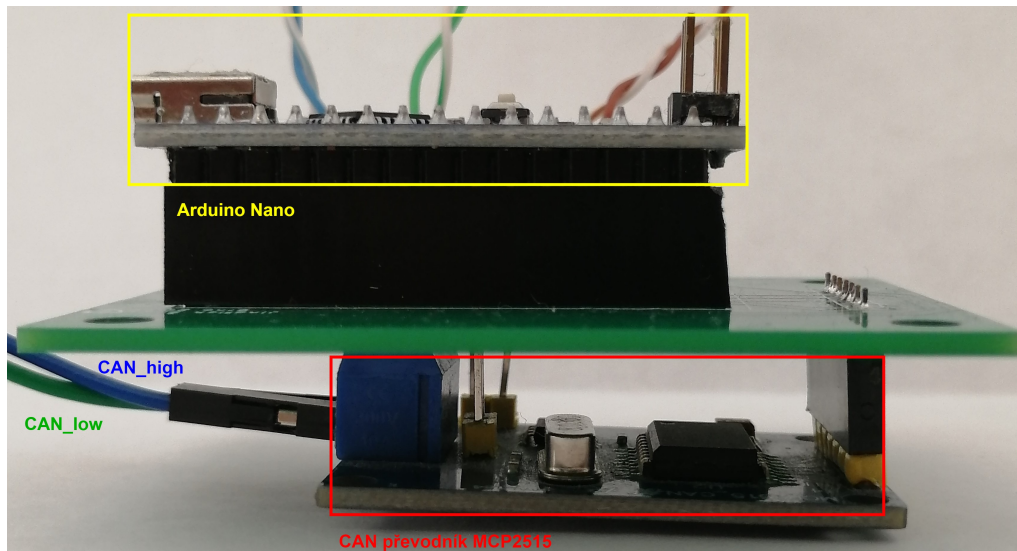
13.2.3 CAN Převodník MCP2515

Arduino Nano je spojeno s CAN převodníkem MCP2515 s budičem TJA1050. Ten podporuje CAN specifikaci 2.0B, může přijímat a vysílat standardní i rozšířené rámce a maximální rychlost komunikace je 1 Mbit/s. Napájecí napětí je 5 V, pracovní proud 5 mA, maximální proud 70 mA a rozměry tohoto prvku jsou 44 × 28 mm [31]. Zapojení CAN převodníku viz Obr. 13.14.

Převodník čeká na příkaz s daty od Arduino, po přijetí vytvoří CAN zprávu s daty, identifikací a délkou datového pole. Poté se nastaví další CAN parametry jako priorita a rychlost přenosu. CAN zpráva se odešle a stane se dostupnou pro všechna zařízení v CAN sběrnici. Pokud je vyžadováno potvrzení zprávy, tak převodník na toto potvrzení čeká.

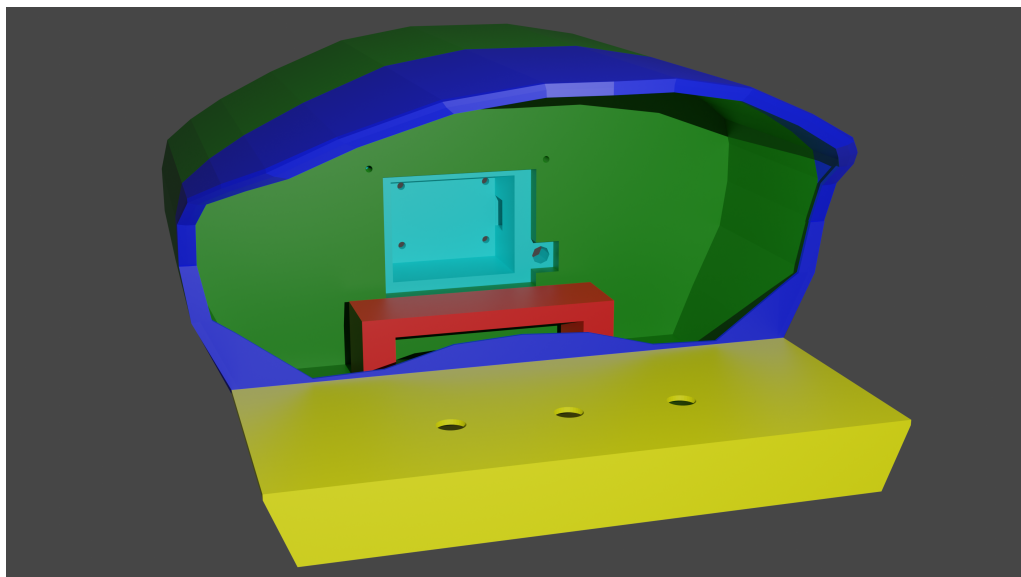
13.2.4 Box pro přístrojovou desku a zpracování CAN zpráv

Návrh boxu probíhal v open-source softwaru Blender. Nejdříve bylo potřeba změřit reálné součástky, aby mohly být převedeny do softwaru a kolem nich vytvořen box.



Obr. 13.14: Zapojení CAN převodníku MCP2515

Inspirace tvaru boxu pochází z reálného umístění v automobilu a hlavně z tvaru palubní desky, která je z použitých součástek největší, viz Obr. 13.15.



Obr. 13.15: Návrh boxu v Blenderu

V přední části boxu je místo na tři tlačítka, uprostřed je největší část pro palubní desku, úchytu pro přišroubování palubní desky a podpěra, aby palubní deska stála rovně. Zadní část je určena pro plošný spoj s Arduinem Nanem a CAN převodníkem. Tato část je připojena ke středové části pomocí šroubů, takže je odnímatelná pro možné úpravy a opravy.

Všechny části boxu byly vytisknuty na 3D tiskárně z materiálu PLA (Polylactic acid), což vzhledem k velikosti boxu představovalo několik úskalí. Celek se musel rozdělit na několik částí, aby se vešel na tiskovou plochu a zároveň neplýtval mnoho materiálu na podpěry. Po vytisknutí se části musely zbavit podpěr a obrousit, aby se mohly spojit pomocí lepidla nebo šroubů do kompletního boxu.

Box slouží pro spojení součástek do jednoho pevného krytu. Nachází se v něm deska plošného spoje s připojeným Arduinem Nanem a CAN převodníkem, tlačítka pro počáteční nastavení a palubní deska z vozidla, viz Obr. 13.16.



Obr. 13.16: Vytisknutý box pro osazení přístrojové desky

13.2.5 Realizace komunikace mezi jednotkami a ovlivnění otáček na přístrojové desce

Spuštění socktapu na vysílací OBU1 jednotce (zatím bude vysílat pouze CAM zprávy, `-p static` protože GPS přijímač má pouze přijímací OBU) probíhá tak, že socktap naslouchá na adrese 127.0.0.1 (v základu nastavený loopback), spouští se aplikace CAM (`-a ca`) pro simulaci v reálném provozu (auta vysílají CAM zprávy v daném intervalu), spouští se aplikace DENM (`-a de`), pro případné varování, spouští se také aplikace IVIM (`-a ivi`), ta však zatím nevysílá a čeká na MQTT zprávu v tématu `/vut/rcv/ivi` knihovny `mosquitto`, viz Obr. 13.17:

```
OBU1: sudo ./socktap_vysilaci -i wlp5s0 -a ivi -a ca -a de -p static
```

Socketap na přijímací OBU2 jednotce spouští CAM aplikaci (simulace v provozu), DENM aplikaci (umožňuje přijímat DENM zprávy), IVIM aplikaci (umožňuje přijímat IVIM zprávy), CAN aplikaci (bez ní by nebylo možné přeposílat data

```

igor@c2x:~$ sudo ./socktap_vysilaci -i wlp5s0 -a ivi -a ca -a de -p static
{INF} mqtt_interface.cpp:101 MqttInterface() -> client started
{INF} mqtt_interface.cpp:34 on_connect() -> Connected with code:Success
{INF} json_interface.cpp:116 createJsonStatus() -> {"active":"running","since":"
2023-08-15T16:02:12+00:00"}
{INF} mqtt_interface.cpp:49 PublishMsg() -> Sending to topic:/vut/snd/status siz
e:56 1
{INF} mqtt_interface.cpp:58 on_publish() -> Published message with id:1 cnt:0
{INF} time_trigger.cpp:14 TimeTrigger() -> Starting runtime at 2023-Aug-15 16:02
:12.582394
{INF} mqtt_interface.cpp:76 SubscribeTopic() -> /vut/rcv/ivi subscribe OK
{INF} mqtt_interface.cpp:76 SubscribeTopic() -> /vut/rcv/cam subscribe OK
{INF} mqtt_interface.cpp:76 SubscribeTopic() -> /vut/rcv/denm subscribe OK
{INF} main.cpp:273 main() -> Enable application 'ca'...
{INF} main.cpp:273 main() -> Enable application 'de'...
{INF} main.cpp:273 main() -> Enable application 'ivi'...
{INF} mqtt_interface.cpp:63 on_subscribe() -> Susbcribe OK id:2
{INF} mqtt_interface.cpp:63 on_subscribe() -> Susbcribe OK id:3
{INF} mqtt_interface.cpp:63 on_subscribe() -> Susbcribe OK id:4

```

Obr. 13.17: Spuštění socktapu na vysílací OBU

na sériovou linku), nastavuje se identifikační číslo stanice pro eliminaci chyb v komunikaci a přiřazuje se rozhraní, na které se budou data přeposílat (v našem případě /dev/ttyUSB0, složka /dev/ na Linuxu obsahuje speciální soubory, které reprezentují zařízení), viz Obr. 13.18:

```

OBU2: sudo ./socktap_can -i wlp5s0 -a ivi -a can -a ca -a de
--station-id=10 --can-type=/dev/ttyUSB0

```

```

igor@c2x:~$ sudo ./socktap_can -i wlp5s0 -a ivi -a can -a ca -a de --station-id=
10 --can-type=/dev/ttyUSB0
{INF} mqtt_interface.cpp:101 MqttInterface() -> client started
{INF} mqtt_interface.cpp:34 on_connect() -> Connected with code:Success
{INF} json_interface.cpp:116 createJsonStatus() -> {"active":"running","since":"
2023-08-15T16:09:14+00:00"}
{INF} mqtt_interface.cpp:49 PublishMsg() -> Sending to topic:/vut/snd/status siz
e:56 1
{INF} mqtt_interface.cpp:58 on_publish() -> Published message with id:1 cnt:0
{INF} time_trigger.cpp:14 TimeTrigger() -> Starting runtime at 2023-Aug-15 16:09
:14.628490
{INF} can_serial.cpp:18 init() -> serial init:/dev/ttyUSB0
{INF} mqtt_interface.cpp:76 SubscribeTopic() -> /vut/rcv/ivi subscribe OK
{ERR} io.cpp:22 OpenPin() -> IO error opening pin:500
{INF} mqtt_interface.cpp:76 SubscribeTopic() -> /vut/rcv/can subscribe OK
{INF} mqtt_interface.cpp:76 SubscribeTopic() -> /vut/rcv/cam subscribe OK
{INF} mqtt_interface.cpp:76 SubscribeTopic() -> /vut/rcv/denm subscribe OK
{INF} main.cpp:273 main() -> Enable application 'ca'...
{INF} mqtt_interface.cpp:63 on_subscribe() -> Susbcribe OK id:2
{INF} main.cpp:273 main() -> Enable application 'can'...
{INF} main.cpp:273 main() -> Enable application 'de'...
{INF} main.cpp:273 main() -> Enable application 'ivi'...
{INF} mqtt_interface.cpp:63 on_subscribe() -> Susbcribe OK id:3
{INF} mqtt_interface.cpp:63 on_subscribe() -> Susbcribe OK id:4

```

Obr. 13.18: Spuštění socktapu na přijímací OBU

MQTT zpráva do tématu /vut/rcv/ivi mosquitto knihovny, nastavuje parametry IVIM zpráv. IVIM zprávy se budou posílat periodicky, jejich stav bude nastaven na 0 (to znamená nová zpráva) a ID zóny bude 456. Socktap po přijetí začne vysílat

IVIM zprávy na omezení otáček na 500 otáček za minutu, které se předají za pomoci modulu s Arduinem až na palubní desku, viz Obr. 13.19:

```
OBU1: mosquitto_pub -t '/vut/rcv/ivi' -m '{"sending":"periodic",
      "iviStatus":0, "geographicContainer":{"zoneId":456},
      "textContainer":{"data":"500"}}'
```

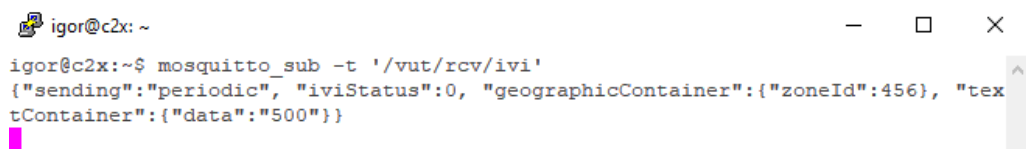


Obr. 13.19: Otáčky po omezení

Při omezování otáček, nastavování směrovek a dalších funkcí je potřeba znát identifikátory a potřebné hodnoty pro správnou funkci. V programu nahraném na Arduino Nano je definována funkce `CanSend` s parametry `short address`, `byte lng`, `byte a-h`, kde `address` znamená ID zprávy, `lng` je délka zprávy v bajtech a `a-h` jednotlivé bajty dat, které budou odeslány. Díky této funkci lze omezovat otáčky, nastavovat směrovky a kontrolky palubní desky, viz Tab. 13.1 (tučně proměnné v programu).

Pro zobrazení zpráv k omezení otáček, které se budou vysílat, viz Obr. 13.20:

```
OBU1: mosquitto_sub -t '/vut/rcv/ivi'
```



Obr. 13.20: Zobrazení odeslané mosquitto zprávy

Po přijetí IVIM zprávy s omezujícími otáčkami přijímací OBU odešle tuto hodnotu na Arduino, které dá příkaz CAN převodníku k vytvoření CAN zprávy a jejímu

Tab. 13.1: Identifikátory a data pro jednotlivé funkce palubní desky

Funkce	ID	Data v hexadecimálním formátu
Imobilizér	0x3D0	8, 0, 0x80, 0, 0, 0, 0, 0, 0
ESP	0xDA0	8, 0x01, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
Tempomat	0x289	8, 0x00, B00000001, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00
Dieselový motor	0x480	8, 0x00, engineControl , 0x00, 0x00, 0x00, tempDpfWarning , 0x00, 0x00
Rychlost motoru	0x320	8, 0x00, (speedL * 100), (speedH * 100), 0x00, 0x00, 0x00, 0x00, 0x00
Otáčky motoru	0x280	8, 0x49, 0x0E, rpmL , rpmH , 0x0E, 0x00, 0x1B, 0x0E
Rychlost vozidla	0x5A0	8, 0xFF, speedL , speedH , dMode , 0x00, lo8(distCtr), hi8(distCtr), 0x00
ABS	0x1A0	8, 0x18, speedL , speedH , 0x00, 0xfe, 0xfe, 0x00, 0xff
Airbag	0x050	8, 0x00, 0x80, tempSeatBelt , 0x00, 0x00, 0x00, 0x00, 0x00
Směrovky	0x531	8, 0, 0, value , 0, 0, 0, 0, 0
Imobilizér	0x3D0	8, 0, 0x80, 0, 0, 0, 0, 0, 0

odeslání pomocí dvou linek na palubní desku. Palubní deska nastaví hodnotu otáček na přijatou hodnotu a spustí blikání směrovek. Při pokusu o nastavení vyšších otáček, se opět změní na nastavenou hodnotu (v našem příkladu na 500), jelikož se IVIM zpráva s otáčkami odesílá periodicky.

13.3 Zachycená komunikace

Zachycená radiokomunikace v době před vysláním omezovací zprávy obsahuje pouze CAM zprávy z obou OBU jednotek, viz Obr. 13.21.

V zachycených zprávách jsou MAC adresy (zdrojová je rozhraní wlp5s0 na OBU jednotce, cílová je broadcast, takže je k dispozici pro všechny, kdo ji přijmou), verze protokolu, typ zprávy (v našem příkladu je 2, to znamená CAM zpráva s podrobnějšími informacemi o vozidle), ID OBU jednotky (1 v základu v socketu, 10 jsme nastavili na přijímací OBU), typ (5 znamená osobní vozidlo), poloha (GPS souřadnice a nadmořská výška), rychlost, směr jízdy (0 znamená dopředu), šířka, délka vozidla a další parametry.

Po vyslání omezovací zprávy jsou v radiové komunikaci zachyceny CAM zprávy jako v předchozím případě a navíc IVIM zprávy z vysílací OBU, viz Obr. 13.22.

No.	Time	Source	Destination	Protocol	Length	Info
10	8.002956	1.5.203.04:f0:21:9f:77:4f	Broadcast	CAM	99	CAM
11	8.125078	1.5.203.04:f0:21:9f:77:63	Broadcast	CAM	99	CAM
12	9.003378	1.5.203.04:f0:21:9f:77:4f	Broadcast	CAM	99	CAM


```

<
> Frame 11: 99 bytes on wire (792 bits), 99 bytes captured (792 bits)
> Ethernet II, Src: CompexPt_9f:77:63 (04:f0:21:9f:77:63), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> GeoNetworking
> BTP-B
> Intelligent Transport Systems
  > ItsPduHeader
    > protocolVersion: 2 Verze Protokolu
    > messageID: cam (2) Typ zpravy
    > stationID: 10 ID OBU jednotky
  > CoopAwareness
    > generationDeltaTime: 15985
    > camParameters
      > basicContainer
        > stationType: passengerCar (5) Typ vozidla
        > referencePosition Poloha vozidla
      > highFrequencyContainer: basicVehicleContainerHighFrequency (0)
        > basicVehicleContainerHighFrequency
          > heading
          > speed Rychlost vozidla
          > driveDirection: forward (0) Směr jizdy vozidla
          > vehicleLength Délka vozidla
          > vehicleWidth: unavailable (62) Šířka vozidla
          > longitudinalAcceleration
          > curvature
          > curvatureCalculationMode: yawRateUsed (0)
          > yawRate
  
```

Obr. 13.21: Komunikace před omezením otáček

No.	Time	Source	Destination	Protocol	Length	Info
73	37.013749	1.5.203.04:f0:21:9f:77:4f	Broadcast	CAM	99	CAM
74	37.014107	1.5.203.04:f0:21:9f:77:4f	Broadcast	IVIM	104	IVIM
75	37.340872	1.5.203.04:f0:21:9f:77:63	Broadcast	CAM	100	CAM


```

<
> Frame 74: 104 bytes on wire (832 bits), 104 bytes captured (832 bits)
> Ethernet II, Src: CompexPt_9f:77:4f (04:f0:21:9f:77:4f), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> GeoNetworking
> BTP-B
> Intelligent Transport Systems
  > ItsPduHeader
    > protocolVersion: 2 Verze protokolu
    > messageID: ivim (6)
    > stationID: 1 ID OBU jednotky
  > IviStructure
    > mandatory
      > serviceProviderId
      > iviIdentificationNumber: 5
      > iviStatus: new (0) Stav IVIM zpravy
    > optional: 2 items
      > Item 0
        > IviContainer: glc (0)
          > glc
            > referencePosition Poloha vozidla
            > parts: 1 item
              > Item 0
                > GlcPart
                  > zoneId: 456 ID zóny
      > Item 1
        > IviContainer: tc (3)
          > tc: 1 item
            > Item 0
              > TcPart
                > detectionZoneIds: 1 item
                > relevanceZoneIds: 1 item
                > data: 353030 Přenášená data
  
```

Obr. 13.22: Komunikace při omezení otáček

IVIM zprávy obsahují MAC adresy (zdrojová je podle rozhraní wlp5s0 na OBU, cílová je broadcast takže pro všechny, kdo ji přijmou), verzi protokolu (druhá verze používá formát JSON místo XML v první verzi, umožňuje rozšíření o další vlastnosti a atributy), ID OBU jednotky (v základu 1), stav zprávy (0 znamená nový), polohu (zeměpisná délka a šířka, nadmořská výška), ID zóny (ve zprávě mosquitto nastavena na 456), ale hlavně hodnotu na kterou chceme omezit otáčky vozidla (v tomto případě 500 -> 353030 v hexadecimálním formátu).

Pomocí příkazu `lsof` jsme získali informace o souboru `/dev/ttyUSB0` např. který proces ho používá (socktap s Process ID 1762 a File Descriptorem 13). K zachycení komunikace s Arduinem byl použit nástroj `strace` s PID získaný díky `lsof`, který slouží ke sledování systémových volání (odeslané a přijaté zprávy, informace o čekání socktapu). V komunikaci lze vidět, že se odesílá na soubor s FD 13 hodnota `":\1\364\r"`, která má velikost 4 bajty a v hexadecimálním formátu `3a 01 f4 0d`, kde 2 bajty `01 f4` definují hodnotu otáček, viz Obr. 13.23.

```

igor@c2x: ~
igor@c2x:~$ sudo lsof /dev/ttyUSB0 Informace o souboru
COMMAND PID USER  FD  TYPE DEVICE SIZE/OFF NODE NAME
socktap_c 1762 root  13u  CHR 188,0    0t0 465 /dev/ttyUSB0
igor@c2x:~$ sudo strace -p 1762 -e write=13 -e read=13 Sledování procesu
strace: Process 1762 attached
write(1, "{INF} router_context.cpp:59 indi"..., 93) = 93
timerfd_settime(8, 0, {it_interval={tv_sec=0, tv_nsec=0}, it_value={tv_sec=0, tv_nsec
write(1, "{INF} ivim_application.cpp:59 in"..., 102) = 102
write(1, "{INF} ivim_application.cpp:96 in"..., 76) = 76
write(1, "{INF} ivim_application.cpp:103 i"..., 59) = 59
timerfd_settime(8, 0, {it_interval={tv_sec=0, tv_nsec=0}, it_value={tv_sec=0, tv_nsec
recvmsg(10, {msg_namelen=128}, 0)          = -1 EAGAIN (Resource temporarily unavailable
epoll_wait(7, [], 128, 0)                  = 0
epoll_wait(7, [{EPOLLIN, {u32=2905975524, u64=94006855183076}}], 128, -1) = 1
timerfd_settime(8, 0, {it_interval={tv_sec=0, tv_nsec=0}, it_value={tv_sec=0, tv_nsec
timerfd_settime(8, 0, {it_interval={tv_sec=0, tv_nsec=0}, it_value={tv_sec=0, tv_nsec
timerfd_settime(8, 0, {it_interval={tv_sec=0, tv_nsec=0}, it_value={tv_sec=0, tv_nsec
epoll_wait(7, [], 128, 0)                  = 0
epoll_wait(7, [{EPOLLIN, {u32=2905975524, u64=94006855183076}}], 128, -1) = 1
timerfd_settime(8, 0, {it_interval={tv_sec=0, tv_nsec=0}, it_value={tv_sec=0, tv_nsec
write(13, "\1\364\r", 4)                   = 4
| 00000 3a 01 f4 0d
write(1, "{INF} can_application.cpp:105 on"..., 79) = 79
timerfd_settime(8, 0, {it_interval={tv_sec=0, tv_nsec=0}, it_value={tv_sec=0, tv_nsec
sendmsg(10, {msg_name=NULL, msg_namelen=0, msg_iov={iiov_base=NULL, iiov_len=0}, {io
\1\0\224\313\4\360!\237woc\222Pf\35XF\241\t\350\215\31"..., iiov_len=40}, {io
0\0\0\nPz\0Z_\245\363\316\246K\343\237\377\374"Dh\366@\244\257\300-~?"..., iiov_l
timerfd_settime(8, 0, {it_interval={tv_sec=0, tv_nsec=0}, it_value={tv_sec=0, tv_nsec
epoll_wait(7, [], 128, 0)                  = 0

```

Obr. 13.23: Zachycená komunikace pomocí strace

Po detekci dat na sériové lince se na Arduinu spustí metoda `serialEvent()`, která přijme data odeslaná OBU jednotkou a čeká na znak dvojtečky a poté čeká na 2 bajtovou hodnotu otáček (v tomto případě 500 v desítkové soustavě) a poslední bajt značí návrat vozíku. Pokud je hodnota přijatých otáček v předem daném rozsahu, tak se odešle příkaz na modul MCP2515. Ten zprávu vhodně upraví pro CAN sběrnici a odešle ji na přístrojovou desku.

Závěr

V rámci bakalářské práce byla vypracována teoretická část zaměřená na inteligentní dopravní systémy, palubní jednotky, zprávy které se mohou objevovat v jejich komunikaci a jejich propojení s vozidly.

Byla úspěšně zprovozněna základní komunikace mezi OBU jednotkami podle schématu zapojení základního pracoviště. Byl připojen a nastaven externí GPS přijímač, takže přijímací OBU jednotka vysílá v CAM a IVIM zprávách aktuální polohu. Bylo zapojeno, popsáno a zprovozněno pracoviště pro vzdálené ovlivnění otáček obsahující přístrojovou desku, které obsahuje dvě OBU jednotky, externí GPS přijímač, Arduino Nano, CAN převodník MCP2515 a palubní desku ze Škody Octavie II. Otáčky byly omezeny za použití knihovny mosquitto implementující MQTT protokol, IVIM zpráv a komunikace na sériové lince a sběrnici CAN.

Byl vypracován návrh desky plošného spoje pro Arduino Nano a převodník CAN. Deska byla vytvořena a poté na ni byly napájeny součástky. Dále byla v Blenderu vytvořena podoba vzhledného boxu pro součástky. Box byl vytisknut na 3D tiskárně po částech a poté spojen v jeden celek. Byla zachycena a analyzována komunikace mezi OBU jednotkami a OBU jednotkou a Arduinem Nano. V zachycené komunikaci byla vidět nezabezpečená poslaná data.

V budoucnosti by mohlo být pracoviště rozšířeno o více možností a funkcí. Také by se mohlo otestovat omezení otáček na reálném vozidle, praktický dosah a zpoždění signálů.

Literatura

- [1] O systémech C-ITS. *ITS knihovna* [online]. [cit. 10. 11. 2022]. Dostupné z URL: <<https://www.its-knihovna.cz/cz/knihovna/projekty/c-roads/systemy-c-its/o-systemech-c-its>>.
- [2] Co je GPS?. *gpsnavigace* [online]. [cit. 26. 03. 2023]. Dostupné z URL: <http://www.gpsnavigace.cz/prispevky/co_je_gps.htm>.
- [3] V2X: What is Vehicle to Everything?. *Thalesgroup* [online]. [cit. 05. 12. 2022]. Dostupné z URL: <<https://www.thalesgroup.com/en/markets/digital-identity-and-security/iot/industries/automotive/use-cases/v2x>>.
- [4] What is RSU (Road-Side Unit). *IGI Global* [online]. [cit. 26. 03. 2023]. Dostupné z URL: <<https://www.igi-global.com/dictionary/rsu-road-side-unit/34931>>.
- [5] The Future Begins with The Road Side Unit. *Medium* [online]. [cit. 26. 03. 2023]. Dostupné z URL: <<https://medium.com/predict/edge-computing-is-so-much-more-fun-ac2a8a23e696>>.
- [6] Roadside Unit. *Digitales Testfeld Stadtverkehr* [online]. [cit. 26. 03. 2023]. Dostupné z URL: <<https://testfeldstadtverkehr.berlin/en/roadside-unit/>>.
- [7] ON BOARD UNIT. *Viapass* [online] [cit. 05. 11. 2022]. Dostupné z URL: <<https://www.viapass.be/en/qa/on-board-unit/>>.
- [8] About the introduction of the system. *hu-go* [online] [cit. 05. 11. 2022]. Dostupné z URL: <<https://www.hu-go.hu/articles/article/about-the-introduction-of-the-system>>.
- [9] Product Brief: ITS-OB4. In: *Commsignia* [online]. 2020 [cit. 04. 12. 2022]. Dostupné z URL: <https://www.commsignia.com/wp-content/uploads/2020/11/Commsignia_ITS_OB4_ProductBrief_v0.9.5_22062020_web.pdf>.
- [10] MK6C EVK: Cohda Wireless C-V2X EVK. In: *Cohdawireless* [online]. [cit. 04. 12. 2022]. Dostupné z URL: <https://www.cohdawireless.com/wp-content/uploads/2022/10/CW_Product-Brief-sheet-MK6C-EVK-v2.pdf>.

- [11] V2X On-Board-Unit, Autotalks SDK for V2X protocol stacks porting: OBU-201 Specification. In: *Taiwantrade* [online]. [cit. 04. 12. 2022]. Dostupné z URL: <<https://info.taiwantrade.com/EP/resources/member/1329/productcatalog/740799f5-fbec-4e08-91f2-b204e0edda3d.pdf>>.
- [12] Vanetza. *Vanetza* [online]. [cit. 13. 11. 2022]. Dostupné z URL: <<https://www.vanetza.org>>.
- [13] Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking. *ETSI* [online]. [cit. 11. 12. 2022]. Dostupné z URL: <https://www.etsi.org/deliver/etsi_en/302600_302699/3026360601/01.02.01_60/en_3026360601v010201p.pdf>.
- [14] Introduction to ASN.1. *ITU* [online]. [cit. 11. 12. 2022]. Dostupné z URL: <<https://www.itu.int/en/ITU-T/asn1/Pages/introduction.aspx>>.
- [15] Intelligent Transport Systems (ITS); ITS-G5 Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band. *ETSI* [online]. [cit. 11. 12. 2022]. Dostupné z URL: <https://www.etsi.org/deliver/etsi_en/302600_302699/302663/01.03.01_30/en_302663v010301v.pdf>.
- [16] Socktap. *Vanetza* [online]. [cit. 15. 11. 2022]. Dostupné z URL: <<https://www.vanetza.org/tools/socktap/>>.
- [17] Certifikační infrastruktury veřejných klíčů, PKI. *Petr Hanáček, Jan Staudek* [online]. [cit. 24. 05. 2023]. Dostupné z URL: <https://www.fi.muni.cz/usr/staudek/vyuka/security/stud_lit/D01_C.pdf>.
- [18] Public Key Infrastructure, Building Trusted Applications and Web Services. *John R. Vacca* [online]. [cit. 24. 05. 2023]. Dostupné z URL: <<https://books.google.cz/books?id=3kS8XDALWWYC>>.
- [19] What is PKI? And how it secures just about everything online. *Josh Fruhlinger* [online]. [cit. 24. 05. 2023]. Dostupné z URL: <<https://www.csoonline.com/article/3400836/what-is-pki-and-how-it-secures-just-about-everything-online.html>>.
- [20] Certify. *Vanetza* [online]. [cit. 24. 05. 2023]. Dostupné z URL: <<https://www.vanetza.org/tools/certify/>>.
- [21] 802.11. *Techtarget* [online]. [cit. 07. 12. 2022]. Dostupné z URL: <<https://www.techtarget.com/searchmobilecomputing/definition/80211>>.

- [22] ARENA, Fabio, Giovanni PAU a Alessandro SEVERINO. A Review on IEEE 802.11p for Intelligent Transportation Systems. *MDPI* [online]. 2020, 9(2) [cit. 07. 12. 2022]. ISSN 2224-2708. Dostupné z URL: <<https://doi.org/10.3390/jsan9020022>>.
- [23] Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service. *ETSI* [online]. [cit. 01. 12. 2022]. Dostupné z URL: <https://www.etsi.org/deliver/etsi_en/302600_302699/30263702/01.03.02_60/en_30263702v010302p.pdf>.
- [24] Analysis of V2X Performance and Rollout Status with a Special Focus on Austria. *Andrea Ulbel* [online]. May 2021 [cit. 05. 12. 2022]. Dostupné z URL: <https://www.researchgate.net/publication/351443279_Analysis_of_V2X_Performance_and_Rollout_Status_with_a_Special_Focus_on_Austria#pf4a>.
- [25] Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Facilities layer protocols and communication requirements for infrastructure services. *ETSI* [online]. [cit. 01. 12. 2022]. Dostupné z URL: <https://www.etsi.org/deliver/etsi_ts/103300_103399/103301/01.02.01_60/ts_103301v010201p.pdf>.
- [26] Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service. *ETSI* [online]. [cit. 01. 12. 2022]. Dostupné z URL: <https://www.etsi.org/deliver/etsi_en/302600_302699/30263703/01.02.01_30/en_30263703v010201v.pdf>.
- [27] What Is CAN Bus. *dewesoft* [online]. [cit. 09. 12. 2022]. Dostupné z URL: <<https://dewesoft.com/daq/what-is-can-bus>>.
- [28] Eloryks-docs. *Github* [online]. [cit. 09. 12. 2022]. Dostupné z URL: <<https://gitlab.com/eloryks/eloryks-docs>>.
- [29] Arduino Nano. *Arduino* [online]. [cit. 23. 05. 2023]. Dostupné z URL: <<https://store.arduino.cc/products/arduino-nano>>.
- [30] arduino-mcp2515. *Github* [online]. [cit. 23. 05. 2023]. Dostupné z URL: <<https://github.com/autowp/arduino-mcp2515>>.
- [31] Převodník s MCP2515 a TJA1050. *Pájeničko* [online]. [cit. 23. 05. 2023]. Dostupné z URL: <<https://pajenicko.cz/prevodnik-s-mcp2515-tja1050-can-bus-na-spi>>.

Seznam symbolů a zkratek

ABS	Anti-lock Brake System
ASN.1	Abstract Syntax Notation One
AIS	Automatic Identification System
C-ITS	Cooperative-Intelligent Transport Systems – Kooperativní inteligentní dopravní systémy
CAM	Cooperative Awareness Messages
CAN	Controller Area Network
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
DCC	Decentralized Congestion Control
DSRC	Dedicated short-range communications
DENM	Decentralized Environmental Notification Message
ECDSA	Elliptic Curve Digital Signature Algorithm
EEPROM	Electrically Erasable Programmable Read-Only Memory
ESP	Electronic Stability Program – Elektronický stabilizační program
ETH	Ethernet
ETSI	European Telecommunications Standards Institute
FIPS	Federal Information Processing Standards
FD	File Descriptor
GB	Gigabajt
GLONASS	Global Navigation Satellite System – globální družicový polohový systém využívaný ruskou armádou
GNSS	Global Navigation Satellite Systems
GPS	Global Positioning System – globální družicový polohový systém vlastněný USA a provozovaný Vesmírnými silami Spojených států amerických

HF	High-Frequency
HSM	Hardware security module
HW	Hardware
ID	Identifier – Identifikátor
IDE	Integrated development environment
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
ITS	Intelligent Transport Systems – Inteligentní dopravní systémy
IVIM	Infrastructure to Vehicle Information Messages
IZS	Integrovaný Záchranný Systém
JSON	JavaScript Object Notation
LF	Low-Frequency
LTE	Long-term Evolution
MAC	Media Access Control
MQTT	MQ Telemetry Transport
NMEA	National Marine Electronics Association
OBD-II	On-Board Diagnostics
OBU	On Board Unit – Palubní jednotka
OCB	Out of Base Context
PDU	Protocol Data Unit
PID	Process IDentification
PKI	Public Key Infrastructure
PLA	Polylactic acid
QoS	Quality of Service
RS232	Recommended Standard 232

RSU	Roadside Unit
RTOS	Real-Time Operating System
SATA	Serial Advanced Technology Attachment
SRAM	Static Random Access Memory
SSD	Solid-state drive
SSH	Secure Shell
SSZ	Světelné Signalizační Zařízení
SW	Software
TCP	Transmission Control Protocol
TFLOPS	Tera Floating-Point Operations Per Second
TLS	Transport Layer Security
TMC	Traffic Management Centre
USB	Universal Serial Bus
VANET	Vehicular Ad Hoc Network
V2I	Vehicle-to-Infrastructure
V2N	Vehicle-to-Network
V2P	Vehicle-to-Pedestrian
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything
VANET	Vehicular Ad Hoc Network
WAVE	Wireless Access in Vehicular Environments
Wi-Fi	Wireless Fidelity
WLAN	Wireless Local Area Network – bezdrátová lokální síť
XML	Extensible Markup Language

A Obsah elektronické přílohy

Elektronická příloha xsimaj03-bp.zip obsahuje program pro Arduino Nano, který lze náhrát na zařízení pomocí Arduino IDE. Dále jsou součástí soubory zachycené komunikace před omezením otáček i v průběhu. Zachycenou komunikaci lze zobrazit například v programu Wireshark. Součástí je také soubor s návrhem boxu pro palubní desku.

```
/.....kořenový adresář přiloženého archivu
├── Arduino_Nano.....návrh desky plošného spoje
│   ├── Arduino_Nano_PCBv2-backups.....zálohy
│   ├── Gerber.....soubory pro výrobu desky
│   ├── Img.....vzor pro logo na desce
│   ├── Arduino_Nano_PCBv2.kicad_pcb.....návrh plošného spoje
│   └── Arduino_Nano_PCBv2.kicad_sch.....schéma plošného spoje
├── Zachycená komunikace.....soubory se zachycenou komunikací
│   ├── CAM a IVIM.pcap.....zachycená komunikace při omezování otáček
│   └── CAM.pcap.....zachycená komunikace v běžném provozu
├── Arduino_Nano.ino.....program pro Arduino
└── BoxV4.blend.....návrh boxu pro palubní desku
```