



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

VÝUKOVÝ PŘÍPRAVEK PRO AUTONOMNÍ ŘÍZENÍ VOZIDEL

TUTORIAL FOR AUTONOMOUS DRIVING

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. Vojtěch Stražil

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Peter Honec, Ph.D.

BRNO 2023

Diplomová práce

magisterský navazující studijní program **Kybernetika, automatizace a měření**

Ústav automatizace a měřicí techniky

Student: Bc. Vojtěch Stražil

ID: 211178

Ročník: 2

Akademický rok: 2022/23

NÁZEV TÉMATU:

Výukový přípravek pro autonomní řízení vozidel

POKyny PRO VYPRACOVÁNÍ:

Cílem práce je vytvořit a zprovoznit výukový přípravek pro demonstraci autonomního řízení, který má sloužit primárně jako ukázka využití kamerových systémů ve vozidlech pro žáky druhého stupně základních škol a studenty středních škol. V rámci diplomové práce student provede rešerši stávajících kamerových, radarových a jiných snímačů a jejich využití v asistenčních systémech automobilů, realizuje průzkum trhu fyzických i výpočetních platforem, které by byly vhodné pro realizaci výukového přípravku. Dále student navrhne a zprovozní výukový přípravek a vytvoří tři příklady použití vytvořeného přípravku. Student v rámci práce také celý přípravek otestuje a předvede jeho funkčnost včetně vytvořených algoritmů.

1. Proveďte rešerši asistenčních systémů užívaných ve vozidlech
2. Proveďte rešerši snímačů užívaných v autonomním řízení
3. Proveďte průzkum trhu fyzických a výpočetních platforem použitelných pro výukový přípravek
4. Navrhněte fyzickou a výpočetní platformu pro výukový přípravek
5. Navrhněte potřebné algoritmy a vytvořte tři příklady použití výukového přípravku pro autonomní řízení (jízda po čáře, jízda mezi čarami, zastavení před objektem)
6. Přípravek otestujte a demonstруйте jeho funkčnost

DOPORUČENÁ LITERATURA:

HLAVAC V., SONKA M., BOYLE R.: Image Processing, Analysis, and Machine Vision, ISBN 978-0495082521

Termín zadání: 6.2.2023

Termín odevzdání: 17.5.2023

Vedoucí práce: Ing. Peter Honec, Ph.D.

doc. Ing. Petr Fiedler, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové 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

Tato práce je věnována návrhu výukové pomůcky pro autonomní řízení určené primárně pro žáky druhého stupně základních škol a studenty škol středních. Cílem této práce bylo vytvoření robotu, který je schopen samostatně se pohybovat po čáře či v jízdním pruhu, jenž je tvořen dvěma čarami, na základě dat získaných z kamerového snímače. Práce se zabývá návrhem mechanické části robotu, řídicí elektroniky i softwarového vybavení. Mechanický základ robotu je tvořen RC autem, které bylo upraveno díly vytištěnými na 3D tiskárně. Řídicí elektroniku tvoří jednodeskový počítač Raspberry Pi verze 4, který je doplněn o Adept Robot HAT. Na toto rozšíření je následně připojena většina potřebných periférií. Obslužný software byl napsán v jazyce Python a v rámci práce bylo vytvořeno několik příkladů použití. Tyto ukázky obsahují algoritmy pro rozpoznávání čar i PSD regulátor.

KLÍČOVÁ SLOVA

Robot, vestavěný systém, Raspberry Pi, počítačové vidění, kamera, zpracovávání obrazu, PSD regulace

ABSTRACT

This thesis is devoted to the design of a teaching aid for autonomous driving intended primarily for second grade primary school students and secondary school students. The aim of this work was to create a robot that is able to move autonomously along a line or in a lane that is formed by two lines, based on data obtained from a camera sensor. The thesis deals with the design of the mechanical part of the robot, the control electronics and the software. The mechanical basis of the robot is formed by an RC car that has been modified with parts printed on a 3D printer. The control electronics is a Raspberry Pi version 4 single board computer, which is complemented by an Adept Robot HAT. Most of the necessary peripherals are then connected to this extension. The operating software was written in Python and several use cases were created as part of the work. These examples include line recognition algorithms as well as a PSD controller.

KEYWORDS

Robot, embedded system, Raspberry Pi, computer vision, camera, image processing, PSD regulation

STRAŠIL, Vojtěch. *Výukový přípravek pro autonomní řízení vozidel*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky, 2023, 92 s. Diplomová práce. Vedoucí práce: Ing. Peter Honc, Ph.D.

Prohlášení autora o původnosti díla

Jméno a příjmení autora: Bc. Vojtěch Stražil
VUT ID autora: 211178
Typ práce: Diplomová práce
Akademický rok: 2022/23
Téma závěrečné práce: Výukový přípravek pro autonomní řízení vozidel

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 diplomové práce panu Ing.Peteru Honcovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Obsah

Úvod	13
1 Asistenční systémy ve vozidlech	14
1.1 Pasivní asistenční systémy	14
1.2 Aktivní asistenční systémy	16
2 Možnosti snímání dopravní situace ve vozidlech	17
2.1 Radar	17
2.2 Ultrazvukový snímač	18
2.3 Optický snímač	19
2.3.1 Lidar	20
2.3.2 ToF kamera	20
2.3.3 Kamera	22
3 Počítačové vidění užívané ve vozidlech	25
3.1 Předzpracování obrazu	25
3.1.1 Jasová transformace	25
3.1.2 Geometrické transformace	26
3.1.3 Lokální předzpracování	27
3.2 Segmentace	28
3.3 Konvoluční neuronová síť - CNN (Convolutional neural network)	29
3.3.1 Konvoluční vrstva	29
3.3.2 Pooling vrstva	30
3.3.3 Fully-Connected vrstva	31
3.4 Rekurentní neuronová síť - RNN (Recurrent neural networks)	31
4 Robotické platformy	33
4.1 Typy podvozků	33
4.1.1 Diferenciální podvozek	33
4.1.2 Tříkolový podvozek s řízeným předním kolem	34
4.1.3 Synchronní podvozek	34
4.1.4 Ackermanův podvozek	35
4.2 Přehled podvozků na trhu	35
4.2.1 Stavebnice s diferenciálním řízením	35
4.2.2 Stavebnice s Ackermanovým řízením	36

5	Výpočetní platformy	39
5.1	Obecné architektury řízení	39
5.2	Přehled výpočetních platforem na trhu	39
5.3	Přehled rozšiřujících perifériích na trhu	41
6	Návrh a realizace výukového prostředku	44
6.1	Hardware	44
6.1.1	Mechanická část	44
6.1.2	Elektronická část	47
6.2	Software	52
6.2.1	Ovládání periférií	53
6.2.2	Detekce čar	56
6.2.3	Regulátor	61
6.3	Aplikace	65
6.3.1	Hlavní část programu	65
6.3.2	Testování funkce	66
	Závěr	71
	Literatura	73
	Seznam příloh	76
	A Výkresová dokumentace tištěných dílů	77
	B Obsah CD	92

Seznam obrázků

1.1	Ukázka asistenčních systémů ve vozidle	14
2.1	Blokové schéma FMCW radaru	17
2.2	Možný průběh vstupních a výstupních signálů radaru	18
2.3	Princip ToF snímače	19
2.4	Princip ToF kamery	21
2.5	Kamerové snímače	22
2.6	Blokové schéma CCD snímače	23
2.7	Blokové schéma CMOS snímače	24
3.1	Příklad funkcí pro jasovou transformaci	26
3.2	Znázornění konvoluce	30
3.3	Příklad operace max-pooling	31
3.4	Rekurentní neuronová síť	32
4.1	Diferenciální podvozek	33
4.2	Tříkolový podvozek	34
4.3	Synchronní podvozek	34
4.4	Ackermanův podvozek	35
4.5	Diferenciálně řízené podvozky	37
4.6	Ackermanovy podvozky	38
5.1	Řídicí jednotka F4 5A AIO FC	40
5.2	Výpočetní platformy	42
5.3	Rozšiřující moduly	43
6.1	Model horního krytu podvozku	45
6.2	Podvozek vozidla	46
6.3	Adaptér motoru	46
6.4	Navržené držáky snímačů	47
6.5	Napěťový dělič	50
6.6	Robot osazený snímači	52
6.7	Blokový diagram zapojení	53
6.8	Účinnost filtru	57
6.9	Standardizace čáry	58
6.10	Zpracování obrazu	60
6.11	Zpracování obrazu	62
6.12	Vývojový diagram ON-OFF regulátoru	63
6.13	Měření akčního zásahu	65
6.14	Vývojový diagram	67
6.15	Testovací dráhy	68
6.16	Prodloužení držáku kamery	69

6.17 Vytvořený výukový robot	72
--	----

Seznam tabulek

2.1	Shrnutí jednotlivých typů lidarů	21
6.1	Ovládaní H-můstku	54
6.2	Testování jízdy v jízdním pruhu	69
6.3	Testování jízdy po čáře	70

Seznam výpisů

6.1 Implementace PSD regulátoru	64
---	----

Úvod

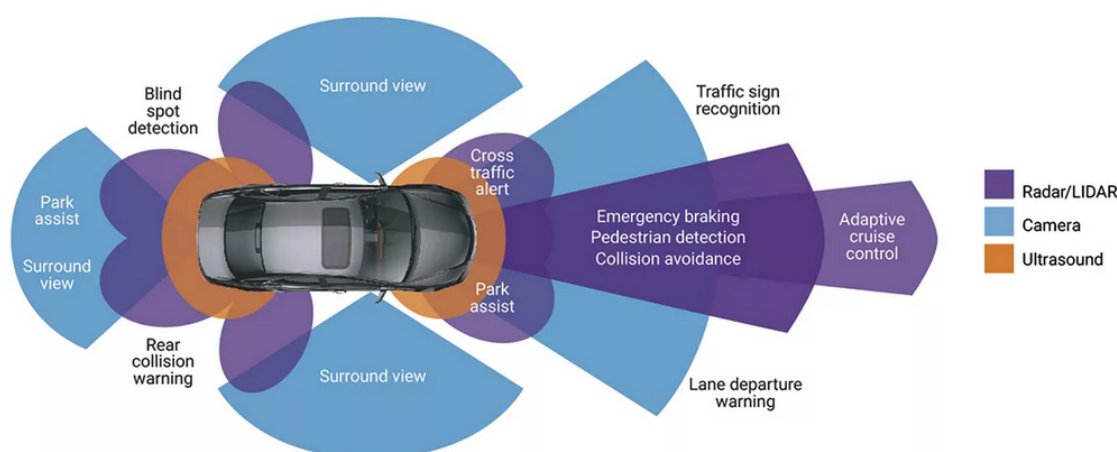
V současné době disponují vozidla mnoha asistenčními systémy, které mají za úkol především zvýšit bezpečnost jízdy a zlepšit komfort posádky. Některé automobilky již představují koncepty plně autonomních vozidel, které jsou schopny dovést posádku z bodu A do bodu B bez jakéhokoliv externího zásahu do řízení. Tímto způsobem má být omezena možnost selhání lidského faktoru, který je největším zdrojem dopravních nehod, čímž by se měla výrazně zvýšit bezpečnost dopravy.

Jestliže má být autonomní vozidlo opravdu bezpečné, je nutné, aby jeho řídicí systémy fungovaly za všech okolností. Většina řídicích algoritmů vyžaduje pro svou správnou funkci zpětnou vazbu, která je vytvářena nejčasněji pomocí různých snímačů. V automobilech je používáno nespočet senzorů, které jsou určeny k měření příslušných fyzikálních veličin. Jako příklady takových snímačů lze uvést např. ultrazvukové snímače pro měření vzdálenosti vozidla od objektů či Hallovy sondy pro měření otáček hřídelů. V současnosti jsou hojně používány i kamerové snímače, jelikož dokáží poskytnout velmi komplexní informace o prostoru, ve kterém se vozidlo pohybuje. Díky nárůstu výkonu výpočetních jednotek lze nyní do automobilů implementovat algoritmy počítačového vidění a hlubokých neuronových sítí, které jsou schopny rozeznávat jednotlivé jízdní pruhy, značky atd. Často je v asistenčních systémech využívána fúze dat, která kombinuje výstupní data z různých snímačů, čímž jsou omezeny některé nežádoucí vlivy způsobené např. špatnými povětrnostními, či světelnými podmínkami. Vytvoření takového asistenčního systému je tak velmi komplexní záležitost.

Jelikož je vývoj asistentů řízení nebo autonomních vozidel velmi perspektivní oblast, která je však opomíjena ve výukových programech středních či odborných škol, klade si tato práce za cíl vytvořit robot, který by sloužil jako prostředek pro výuku řídicích algoritmů spojených s počítačovým viděním určený především pro žáky středních škol nebo vyšších ročníků základních škol. Téma této práce vychází z potřeb odborného robotického kroužku na Gymnáziu Zábřeh, kde autor této práce působí jako lektor. V úvodu této práce bude vytvořen přehled některých asistenčních systémů užívaných v současných automobilech. V následující kapitole budou rozebrány jednotlivé snímače a principy jejich funkce. Jelikož se má práce zaměřit na řízení na základě počítačového vidění, budou v práci zahrnuty některé algoritmy počítačového vidění používané ve vozidlech. V další kapitole budou popsány výsledky průzkumů trhu s robotickými a výpočetními platformami. Poslední kapitolu této práce bude tvořit samotná realizace potřebného hardwaru i softwaru a výsledky testování vytvořeného přípravku.

1 Asistenční systémy ve vozidlech

Veškeré automobily a jiná vozidla, se kterými se lze v silničním provozu setkat, disponují alespoň základními asistenčními systémy. Některé asistenční systémy se dokonce stávají povinnou výbavou automobilů uváděných na trh, např. od roku 2022 nelze v EU homologovat osobní vozy bez systému nouzového brzdění nebo bez systému udržování vozidla v jízdním pruhu. V dnešní době jsou tyto systémy provedeny v nejrůznějších podobách a s odlišnou mírou asistence. Některé asistenty pouze upozorňují řidiče na vzniklé dopravní situace pomocí akustické, vizuální či haptické výstrahy. Pokročilejší systémy jsou schopny zasáhnout do řízení, a zabránit tak nebezpečné situaci. Všechny tyto asistenty spojují dva společné cíle. Prvním z nich je usnadnění řízení vozidla, čímž poskytují větší komfort. Druhým cílem je zvýšení bezpečnosti silničního provozu. Tyto asistenční systémy jsou souhrnně označovány zkratkou ADAS (Advanced Driver Assistance Systems) a jsou zpravidla děleny do dvou skupin, a to na pasivní asistenční systémy a aktivní asistenční systémy.



Obr. 1.1: Ukázka asistenčních systémů ve vozidle. Dostupné z [1]

1.1 Pasivní asistenční systémy

Pasivními asistenčními systémy se rozumí systémy, které se aktivně nepodílí na řízení vozidla a slouží pouze jako upozornění pro řidiče vozidla pomocí akustické, vizuální nebo haptické odezvy. Mezi tyto asistenty patří např:

Varování před opuštěním jízdního pruhu - LDW (Lane Departure Warning)

Tento asistenční systém upozorní řidiče na vybočení z jízdního pruhu. Například automobily značky VOLVO upozorní řidiče akustickým signálem a také vibracemi

volantu, jejichž intenzita se zvětšuje společně s časem stráveným v jiném jízdním pruhu. Tato výstraha u této konkrétní značky není aktivována, pokud řidič aktivuje znamení o změně jízdního pruhu, sešlápne brzdový pedál, rychle sešlápne plynový pedál, v případě rychlého pohybu volantu a pokud rychlost vozidla nepřekročila hodnotu 65 km/h nebo nejsou detekovány vodící čáry.

Varování před čelním nárazem - FCW (Forward Collision Warning)

Asistenční systém, který má řidiče vozidla upozornit na nebezpečí čelní srážky s jiným vozidlem, osobou, zvířetem nebo jiným objektem. U nemalého počtu automobilek se již tento systém řadí mezi aktivní asistenční systémy, neboť v případě nebezpečí nehody zasáhne do řízení tím, že vozidlo autonomně zabrzdí, případně se objektu na vozovce bude snažit vyhnout. Stále ale existují systémy, které pouze upozorní řidiče akustickou, případně vizuální odezvou. U tohoto asistenta se lze setkat i s odlišnými názvy, jako je např.: IBA (Intelligent Brake Assist), CAS (Collision Avoidance System), Front Assistant atd.

Hlídní mrtvého úhlu - Blind Spot Detection

Tento asistenční systém má za úkol upozornit řidiče na jiné vozidlo, které se nachází za automobilem, v případě, kdy chce řidič změnit směr jízdy. Pokud se totiž vozidlo za řidičem nachází v mrtvém úhlu, nelze toto vozidlo z řidičova pohledu vidět ve zpětných zrcátkách, což může vést v krajním případě ke kolizi automobilů. Většinou je tato funkce implementována v blízkosti zpětných zrcátek a systém varuje řidiče pomocí vizuálního upozornění. Některé automobilové značky např. Mercedes-Benz tento systém povyšují na úroveň aktivního asistenčního systému tím, že pokud se nachází vozidlo v mrtvém úhlu, není možné zahájit proces přejíždění do jiného pruhu.

Parkovací asistence - Parking Assistance

Asistenční systém, který má za úkol upozornit řidiče na objekty, jež jsou v blízkosti nárazníků jeho vozidla. Tento asistent je určen primárně pro případ, kdy dochází k manévrování s vozidlem při nízké rychlosti. Snímače této asistenční jednotky tedy nemají příliš velký dosah a jsou určeny zejména pro snímání bezprostředního okolí vozidla. Často tento systém bývá doplněn zadní kamerou, nebo celou soustavou kamer, která umožňuje pomocí transformací obrazu zobrazit řidiči celé okolí jeho automobilu.

1.2 Aktivní asistenční systémy

Aktivní asistenční systémy aktivně zasahují do řízení vozidla tak, aby v případě, že by došlo k nebezpečné dopravní situaci a k ohrožení posádky automobilu, tomuto nebezpečí úplně zabránily nebo alespoň zmírnily jeho dopad na posádku a okolní vozidla. Mezi tyto aktivní systémy lze zařadit např.:

Automatické nouzové brždění - AEB (Automatic Emergency Braking)

Tento asistenční systém má zabraňovat kolizím s ostatními objekty na vozovce. Snímače proto sledují dopravní situaci před i za vozidlem. Algoritmy neustále vyhodnocují potřebnou brzdnou dráhu k zastavení vozidla. Pokud řidič vzdálenost nutnou k zastavení vozidla nedodrží, je zpravidla nejdříve upozorněn akustickým nebo vizuálním signálem. Jestliže na toto upozornění nezareaguje, systém automaticky začne vůz zpomalovat. V krajním případě může tento asistenční systém vozidlo úplně zastavit, a zabránit tak srážce, případně razantně zmírnit její dopad.

Asistent nouzového zásahu do řízení - ESA (Emergency Steering Assist)

Asistent nouzového zásahu do řízení velmi úzce spolupracuje s asistencí AEB. Má za úkol zamezit kolizi tím, že zasáhne do řízení vozidla a pomůže řidiči v řízení vozidla tak, aby se vyhnulo překážce, před kterou už nelze zastavit. Vozidla některých automobilových značek se po tomto manévru vrátí do původního jízdního pruhu a pokračují ve své jízdě.

Adaptivní tempomat - ACC (Adaptive Cruise Control)

Podstata tempomatu vychází z potřeby udržování konstantní rychlosti vozidla během dlouhé jízdy, např. na dálnici, tak, aby se zvýšil komfort řidiče, který tak nemusí ovládat plynový pedál. Adaptivní tempomat dokáže kromě udržování nastavené rychlosti upravit rychlost vozidla podle dopravní situace před vozidlem.

Aktivní parkovací asistent - Automatic Parking Assist

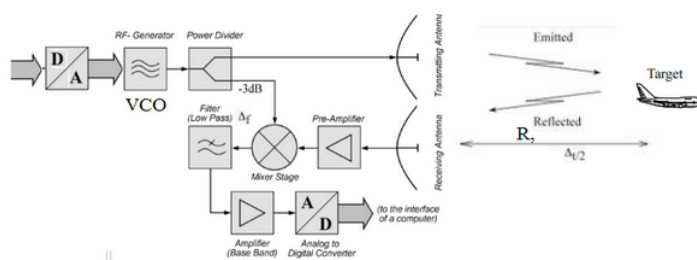
Tento parkovací asistent je u výrobců označován mnoha názvy. Značky jako jsou Mercedes-Benz, Ford nebo Cadillac tento systém označují jako Automatic Parking Assist. U jiných např. Volkswagen, Jaguar, Audi se tato asistence nazývá Park Assist. Tento systém má za úkol samočinně zaparkovat vůz na příslušné parkovací místo. Úroveň pomoci se u různých značek liší. Některé systémy zaparkují zcela bez přičinění řidiče, jiné zase vyžadují řidičovu pomoc v úkonech, jako jsou zařazení příslušného převodového stupně, ovládání pedálu akcelerace a decelerace.

2 Možnosti snímání dopravní situace ve vozidlech

Aby bylo možné realizovat ve vozidlech jakýkoliv asistenční systém, je potřeba mít dokonalý přehled o okolí, v němž se vozidlo pohybuje. K tomuto účelu se používají různé typy snímačů. Dnešní vozidla jsou vybavena celou řadou snímačů jako jsou radary, lidary, ultrazvukové snímače a kamery.

2.1 Radar

V současné době se v automobilové průmyslu nejčastěji využívají tzv. FMCW (Frequency Modulated Continuous Wave) radary. Tento typ radarů má oproti jiným typům mnoho zásadních výhod. Nabízí dostatečně přesné měření vzdálenosti. Nevyžaduje tak vysoký výpočetní výkon jako ostatní typy radarů. Kromě měření vzdálenosti od objektu dokáže měřit díky Dopplerově jevu i jeho rychlost. Měření je navíc velmi rychlé a spolehlivě funguje i za špatného počasí.



Obr. 2.1: Blokové schéma FMCW radaru [2]

Tento typ radarů vysílá frekvenčně modulovanou spojitou radiovou vlnu, jejíž frekvence roste společně s časem. Tento signál je vyslán vysílací anténou. Odezva toto signálu (echa) je přijímána pomocí přijímací antény. Vyslané a přijaté signály jsou navzájem porovnány a frekvenční rozdíl mezi těmito dvěma signály Δf je přímo úměrný vzdálenosti R radaru od objektu. [2] Pro výpočet vzdálenosti platí tento vztah:

$$R = \frac{c \cdot \Delta f}{Kf \cdot 2} [2] \quad (2.1)$$

R ...Vzdálenost radaru od objektu [m]

c ...Rychlost světla [$\frac{m}{s}$]

Δf ...Rozdíl mezi vyslanou a přijatou frekvencí [Hz]

Kf ...Sklon změny frekvence nebo posunu frekvence za jednotku času [$\frac{Hz}{s}$]

Díky Dopplerově jevu lze určit, zda se objekt od radaru přibližuje, či oddaluje. Frekvence přijatého signálu se totiž zvyšuje, pokud se objekt k radaru přibližuje. Když se objekt od radaru vzdálí, přijatá frekvence se naopak sníží. [2] Pro výpočet rychlosti objektu lze využít tohoto vzorce:

$$f_D = 2 \cdot f_t \cdot \frac{v_r}{c} [2] \quad (2.2)$$

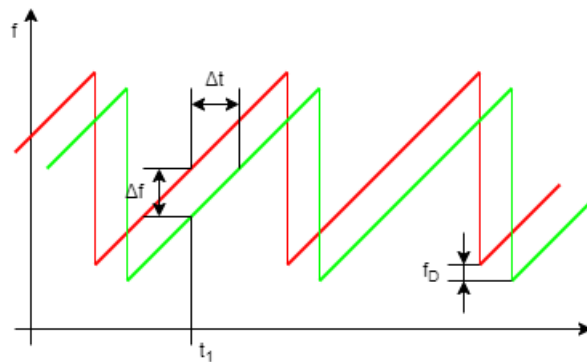
f_D ...Dopplerova frekvence [Hz]

f_t ...Frekvence přijímaného signálu[Hz]

v_r ...Rychlost objektu[$\frac{m}{s}$]

c ...Rychlost světla[$\frac{m}{s}$]

Na obrázku č. 2.2 je znázorněn možný průběh odeslaných (červený) a přijatých (zelený) signálů FMCW radarů. Lze si všimnout posunutí obou signálů ve vertikální ose, z čehož lze vypočítat vzdálenost objektu od radarového systému. Viditelné je také horizontální posunutí, ze kterého lze vypočítat rychlost objektu.

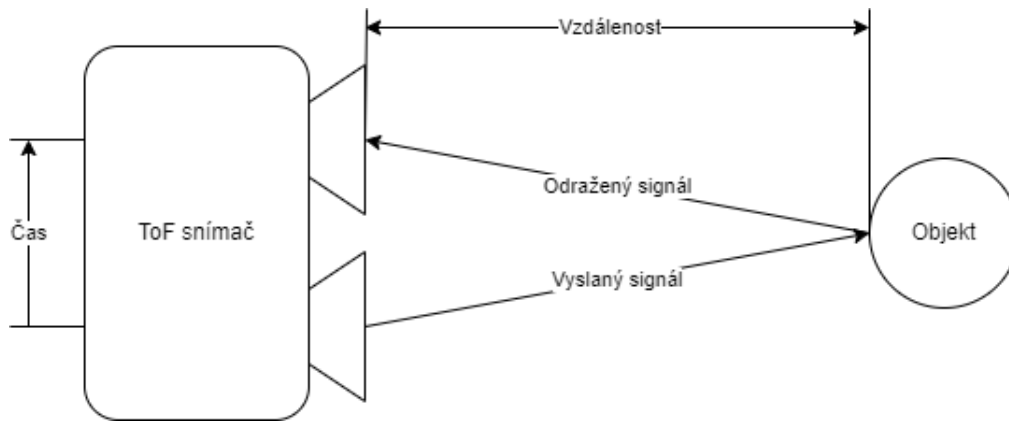


Obr. 2.2: Možný průběh vstupních a výstupních signálů radaru [2]

2.2 Ultrazvukový snímač

Ultrazvukové snímače se v automobilovém průmyslu používají již dlouhou dobu a i v dnešní době jsou stále hojně užívané. Důvodem, proč tomu tak je, může být například jejich příznivá cena, odolnost vůči rušení a nepříznivým povětrnostním vlivům. Tyto snímače se v automobilovém průmyslu nejčastěji využívají pro parkovací asistenty, jelikož spolehlivě měří vzdálenost přibližně do 5 m. Existují ovšem i ve verzích, které jsou schopny měřit vzdálenost do 20 m. Jelikož se řadí ke snímačům s pomalejší odezvou, jsou nejčastěji používány pro měření vzdálenosti v případě, že vozidlo nepřesáhne rychlost přibližně 10 km/h.

Tyto snímače fungují na principu vysílání a následném snímání akustických impulzů, které jsou nejčastěji generovány pomocí piezoelektrických emitorů. Tyto pulzy



Obr. 2.3: Princip ToF snímače

jsou vysílány ve frekvencích vyšších než 20 KHz . Zpravidla od 23 KHz do 40 KHz . Člověk tyto vlny tedy není schopen slyšet. Následně je měřena doba, jež uplynula od vyslání pulzu k jeho detekci. Tento typ měření je v anglické literatuře označován jako Time of Flight (doba letu), zkráceně ToF. Podle následujícího vzorce lze vypočítat vzdálenost objektu od měřicího systému:

$$d = \frac{c \cdot t}{2} \quad (2.3)$$

c ...rychlost šíření zvuku ve vzduchu [$\frac{m}{s}$]

t ...čas, který uplynul od vyslání vlny po její detekci[s]

Tyto snímače jsou vyráběny v několika provedení. Existují tzv. dvouhlavé systémy, které využívají jeden měnič pro vyslání a druhý pro příjem signálu. Další snímače mají pouze jeden měnič, který vyše příslušný signál, poté se přepne do režimu přijímače a odražený pulz sám detekuje. Tento typ snímačů se nejčastěji využívá v automobilovém průmyslu. Dalším typem snímačů je tzv. ultrazvuková zápora, kde jsou proti sobě namířeny přijímače a vysílače a je detekován stav, kdy se nějaký objekt vyskytne mezi měniči.

2.3 Optický snímač

V automobilovém průmyslu jsou hojně využívané optické snímače hned z několika důvodů. Oproti jiným technologiím jde o velmi rychlé snímače. Některé typy optických snímačů dokáží snímat objekty daleko před vozidlem. Jiné zase mají velmi velké zorné pole a jsou schopny zabrat prakticky celé okolí vozidla. Jejich výhodou je také možnost použití společně s algoritmy strojového učení k rozpoznávání dopravních značek a jejich schopnost komplexně zachytit scénu před snímačem.

2.3.1 Lidar

Slovo lidar pochází z anglického light detection and ranging. Jedná se o techniku, jež je známa už od minulého století. Prakticky se jedná o metodu Time of Flight, jež je popsána v podkapitole 2.2, s tím rozdílem, že vzdálenost není měřena pouze od jednoho bodu, nýbrž od několika stovek bodů. V literatuře se pro tuto skupinu bodů ujal pojem mračno bodů. V průmyslu, a především v robotice, se užívá několik typů lidarů s rozdílnými vlastnostmi.[3]

Systémy založené na mechanických skenerech často využívají soustavy zrcadel, nebo hranolů, které jsou připojeny k mechanickému pohonu společně s detektory. Zrcadla mohou být u tohoto typu skeneru rotační nebo oscilační. Pomocí tohoto senzoru lze pokrýt celé okolí vozidla, jelikož zorný úhel tohoto snímače se může rovnat až 360° . Mohou mít také velký dosah. Nevýhoda těchto systémů je ale především jejich vysoká hmotnost a větší rozměry. Dále jsou tyto snímače náročnější na spotřebu elektrické energie. Pro automobilový průmysl také nejsou příliš vhodné z důvodu údržby mechanických částí a jejich náchylnosti na otřesy a vibrace.[3]

Mikroelektromechanické systémy, jenž jsou založeny na MEMS technologii, umožňují programovatelné ovládaní, a jsou tedy schopny změřit vzdálenost od konkrétního bodu ve scéně. Tento typ snímačů velmi často nahrazuje mechanické skenery, jelikož jsou více robustní a nenáročné na údržbu. Dokážou také pracovat na frekvenci o několika kHz. Jejich nevýhodou je nutnost omezení výkonu laseru působící na zrcadla, což je problém především při měření na velké vzdálenosti.[3]

Optická fázová soustava (OPA - optical phased array) je typ polovodičového zařízení, které dokáže řídit optické vlnění. Jeho princip činnosti je prakticky totožný s principem fázových polí, kde je směr paprsku řízen pomocí ladění fázového vztahu mezi soustavami vysílacích antén. Jelikož tento typ snímače neobsahuje žádné pohybující se mechanické části, je velmi robustní a dokáže velmi rychle a přesně řídit paprsek. Tyto systémy jsou také velmi rychlé a mohou pracovat na frekvenci až několika stovek kHz. Nicméně tyto snímače se stále nachází ve fázi vývoje.[3]

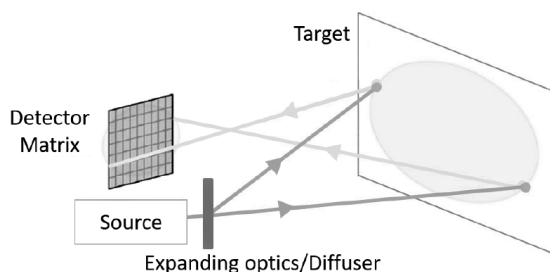
2.3.2 ToF kamera

Jelikož v automobilovém průmyslu nejsou populární pohyblivé systémy kvůli náročnosti jejich údržby a náchylnosti na vibrace, byly vytvořeny systémy, které jsou schopny měřit vzdálenost od objektů podobně jako lidary, ale neobsahují žádné pohyblivé části. Takovými měřicími systémy jsou tzv. ToF kamery. U tohoto typu kamer osvětlují optické prvky celou scénu a matice detektorů přijímá signály z jednotlivých podsekcí. Tímto způsobem lze získávat data o vzdálenosti soustavy od objektu paralelně. Jedná se tak o velmi rychlý způsob měření, který lze použít

Tab. 2.1: Shrnutí jednotlivých typů lidarů [3]

	Mechanické skenery	MEMS skenery	Optická fázová soustava
Princip činnosti	Rotující zrcadla/ hranoly	MEMS microzrcadla	Fázová pole antén
Výhody	360° zorný úhel v jedné rovině	Malé rozměry	Žádné pohyblivé části
Nevýhody	Pohyblivé části, objemné	Výkon laseru na zrcadlech	Velký dosah pouze v lab. podmínkách

k měření vzdálenosti do několika metrů. Scéna může být osvětlována pulzně, nebo kontinuálně.[3]



Obr. 2.4: Princip ToF kamery [3]

V případě pulzně osvětlované scény měřicí systém produkuje v krátkých impulzech světlo, jež osvítlí celou zájmovou oblast scény. Odražené světlo je poté přijímáno detektory. Každý tento detektor vyhodnocuje vzdálenost od objektu pomocí principu doby letu. Jelikož každý detektor v poli reaguje samostatně na odražený impulz, je prostorové rozlišení silně závislé na rozlišení kamery. Tyto senzory se začínají v automobilovém průmyslu využívat pro měření vzdáleností do několika metrů. Nevýhodou tohoto typu ToF kamer je, že se lze v silničním provozu setkat se situacemi, které dokážou celou matici detektorů oslepit, jelikož dojde k nasycení detektorů různými vnějšími vlivy, jako jsou např.: přímé sluneční záření, světlo mety protijedoucích vozidel, interference s dalšími senzory nebo speciální nátěr na dopravních značkách. Přesto se jedná o velmi robustní metodu, jelikož je snímání jednotlivých snímků velmi rychlé.[3]

Soustavy, které osvětlují scénu kontinuálně, pracují na principu měření AMCW (Amplitude Modulated Continuous Wave - amplitudově modulované spojité vlny). Zařízení tedy moduluje intenzitu zdroje záření a poté vyhodnocuje fázový rozdíl mezi zdrojem záření a detekovaným signálem na každém detektoru matice. Tyto

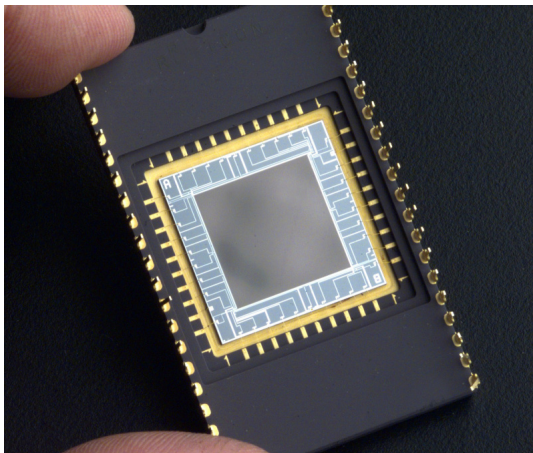
snímače mají obvykle dosah přibližně 10 m , ovšem nejsou příliš vhodné pro použití ve venkovních prostorech především kvůli potřebě zvýšeného počtu digitalizačních úrovní potřebných ke spolehlivému měření fáze.[3]

2.3.3 Kamera

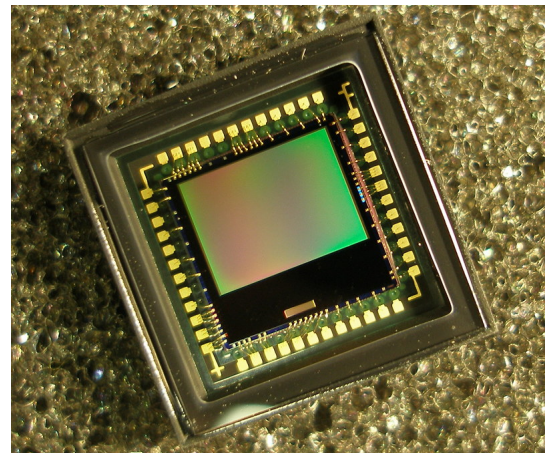
Kamery jsou v dnešní době v automobilovém průmyslu hojně využívaným snímačem. Může za to pravděpodobně relativně nízká cena, komplexnost dat snímané scény a velký dosah snímače. Jsou často využívány v asistenčních systémech jako je např.: hlídání jízdního pruhu (LDW), adaptivní tempomat (ACC), rozpoznávání dopravních značek či jiných objektů před vozidlem.

Hlavní funkcí těchto sensorů je převod světelné informace na elektrický signál. První kamery využívaly principu fotoemise využívající fotoelektrický jev. Jedná se o fyzikální jev, při kterém se volné elektrony uvolňují z obalu atomu kvůli energii, kterou nesou fotony dopadající na materiál. Emise uvolněných elektronů je nazývána jako fotoemise. Tento efekt se nejvíc projevuje u kovových materiálů. [4]

V aplikacích, které souvisí s analýzou obrazu, se tohoto jevu nejvíce využívalo ve vakuových trubicových kamerách. Kvůli rozvoji polovodičů se dnes používají senzory, jež jsou založeny na fotovoltaiickém principu. Tento princip se zakládá na faktu, že elektron opustí valenční pás a přejde do vodivostního pásu vlivem působení energie fotonu. Excitovaný elektron je zdrojem elektrického napětí, které se projevuje jako elektrický proud. Množství energie, kterou obsahují přicházející fotony, je přímo úměrné elektrickému proudu. Snímače založené na fotovoltaiickém principu jsou dále děleny do dvou podskupin: CCD a CMOS. [4]



(a) CCD senzor [5]

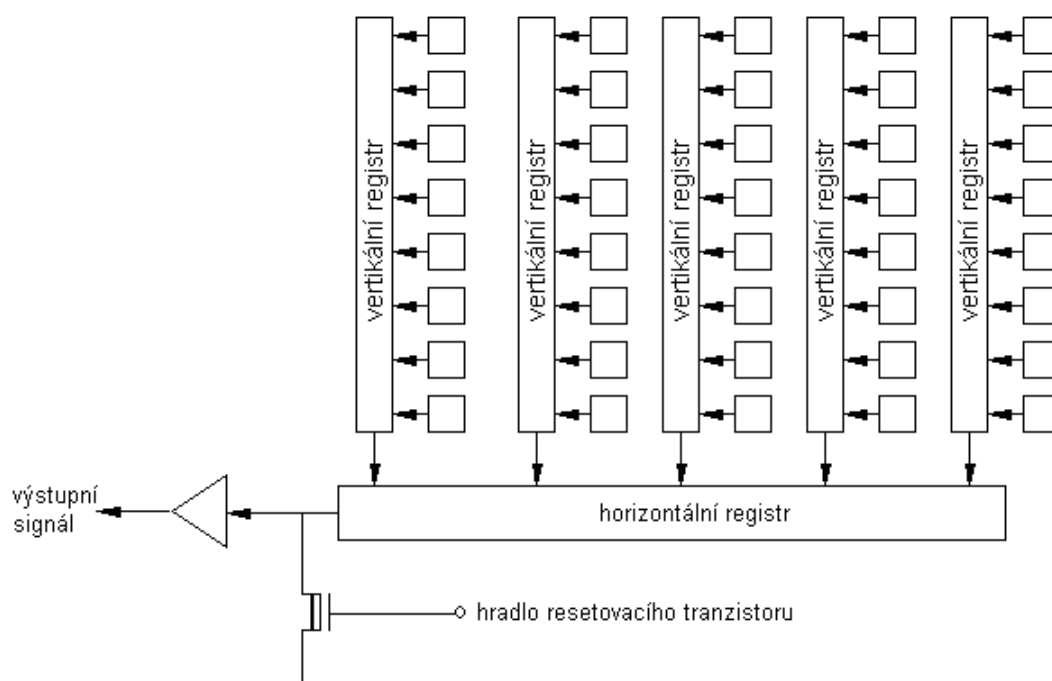


(b) CMOS senzor [6]

Obr. 2.5: Kamerové snímače

CCD

U tohoto typu snímače se náboj každého pixelu přenáší přes jeden výstupní uzel. Tam je převeden na napětí, uložen do vyrovnávací paměti a odeslán jako analogový signál. K zachycení světla může sloužit celá plocha pixelu. Snímače tohoto typu obsahují Schottkyho diody a tranzistory. Dopadající fotony uvolňují elektrony z krystalové mřížky a vytvářejí díry. Tímto způsobem vzniká elektrický náboj, jenž je přímo úměrný intenzitě dopadajícího světla a doby dopadů fotonů. Snímací prvky jsou umístěny do matice, ve které se náboje přenáší z vertikálního posuvného registru do horizontálního registru. Tímto způsobem vzniká videosignál. Tento typ snímačů vykazuje několik problémů. Prvním z nich je vzájemné ovlivňování nábojů v sousedních pixelech. Další problém je ten, že nelze adresovat jednotlivé pixely, jelikož čtení lze provádět pouze přes posuvné registry. Posledním problémem je, že dobře chlazený CCD čip je schopný zvládnout přibližně jen čtyři řády intenzity (jasu), kdežto lidské oko vnímá až devět řádů.[4]

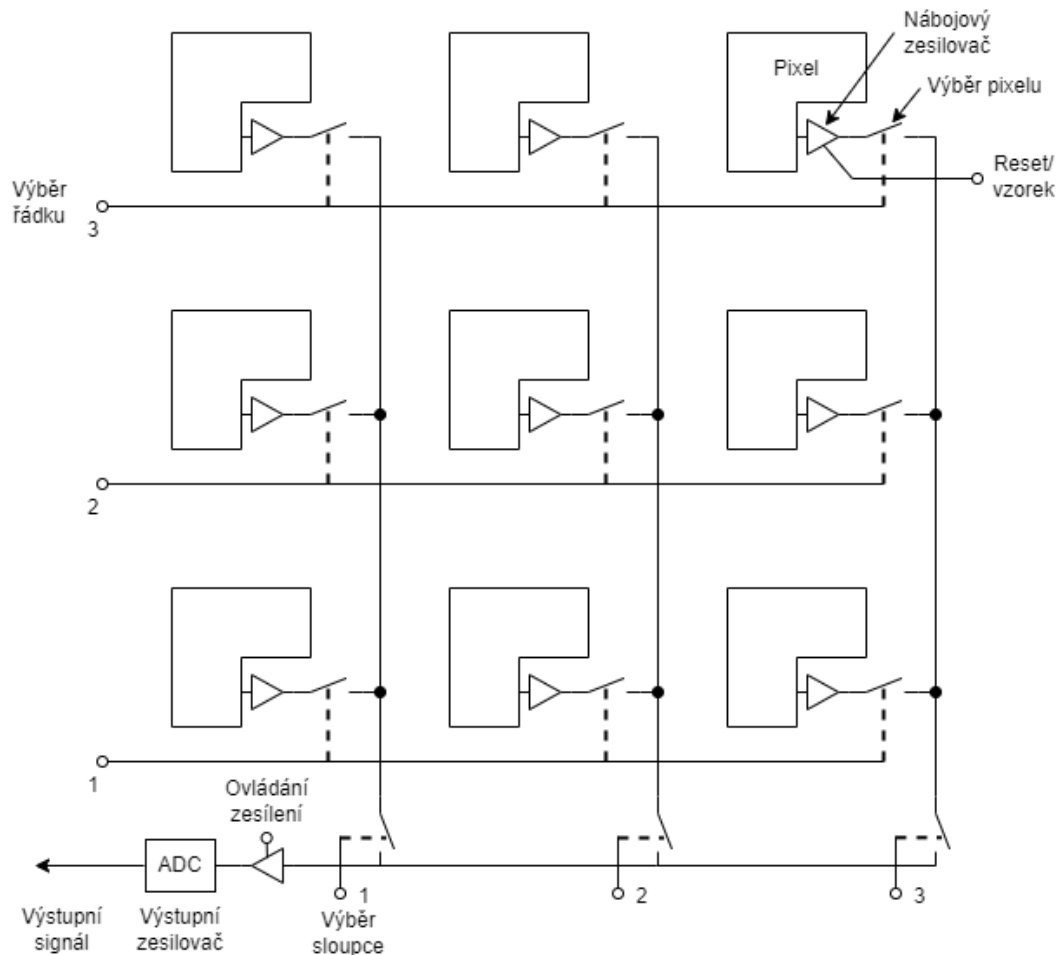


Obr. 2.6: Blokové schéma CCD snímače [7]

CMOS

CMOS snímače se rozšířily kvůli rozvoji polovodičového průmyslu společně s procesory a paměťmi, jelikož jsou všechna tato zařízení vyráběna stejnou technologií. Díky tomu jsou CMOS kamery relativně levné na výrobu. U tohoto typu snímače

je převod náboje na napětí realizován v každém pixelu samostatně. Jednotlivé pixely snímače často obsahují zesilovače, obvody pro korekci šumu a digitalizaci, což zvyšuje náročnost konstrukce a v důsledku toho se zmenšuje i plocha zachycující fotony. Výhodou těchto snímačů je větší rozsah intenzit než u CCD, vysoká rychlost čtení a náhodný přístup k jednotlivým pixelům. Tyto snímače ale mají větší úroveň šumu.[4]



Obr. 2.7: Blokové schéma CMOS snímače [8]

3 Počítačové vidění užívané ve vozidlech

Počítačové vidění je pro autonomní řízení velmi důležitým prvkem, jelikož poskytuje velmi komplexní informace o prostředí, ve kterém se vozidlo pohybuje. Pomocí počítačového vidění lze rozeznat jednotlivé objekty na vozovce, např. ostatní vozidla, chodce, dopravní značení nebo jednotlivé jízdní pruhy.

Jelikož optické snímače poskytují pouze informace o úrovních jasu ve snímané scéně, je nutné tyto informace dále zpracovat. Pro zpracování obrazových dat se v současné době používají metody hlubokého strojového učení, které mají za úkol v této oblasti zjistit, jaké objekty se na nasnímané scéně nacházejí. Protože jsou takto zpracovaná data využívána v systémech jako jsou např. systémy nouzového brždění (AEB), asistence hlídání jízdního pruhu (LDW) či parkovací asistence, je nutné, aby rozpoznávání probíhalo v reálném čase za použití minima výpočetního výkonu. Z důvodu velké energetické náročnosti vozidla není možné osadit výkonnou výpočetní jednotkou. V současnosti jsou stále vyvíjeny nové algoritmy, které budou klást menší nároky na výpočetní výkon, přitom si však zachovají potřebnou míru přesnosti.

3.1 Předzpracování obrazu

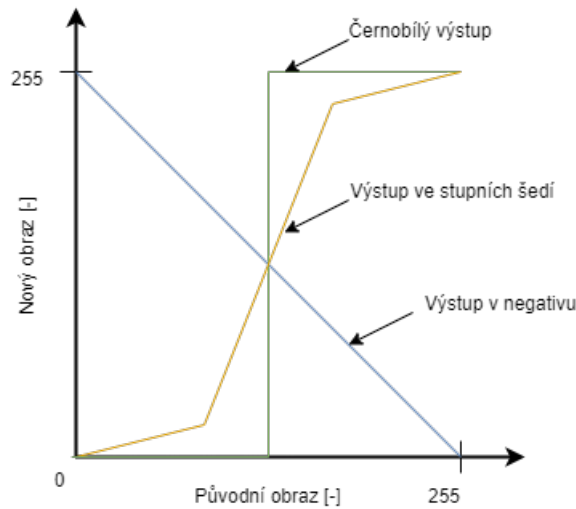
Snímaný obraz obsahuje velké množství informací, které by bylo velmi obtížné zpracovat současně, proto je předzpracování obrazu nedílnou součástí počítačového vidění. Jeho cílem je totiž potlačení nežádoucích informací a zvýraznění podstatných rysů, které se v obraze nachází. Předzpracováním však nelze získat nové informace. Nežádoucí data jsou potlačena celou řadou metod, mezi které patří například jasové či geometrické transformace. Vybrané postupy budou blíže popsány v následujících podkapitolách.

3.1.1 Jasová transformace

Jasové transformace se obvykle dělí do dvou tříd. První z nich má za úkol individuálně korigovat jas pixelu tak, aby byl maximálně konzistentní v celé ploše obrazu. Snímač totiž obvykle ve všech svých částech nemá stejnou citlivost. Problémy s nehomogenní jasovou citlivostí může způsobovat i nerovnoměrné osvětlení scény. [4]

Do druhé třídy jasové transformace lze zařadit operaci zvanou prahování. Tato metoda spočívá ve změně jasových úrovní původního obrazu na nové s použitím podobných funkcí, jako jsou uvedeny na obrázku č. 3.1. V závislosti na tvaru funkce lze dosáhnout černobílého obrazu, obrazu v úrovních šedi či negativu. Touto metodou

lze docílit zvýraznění požadovaných prvků v obrazu a potlačení těch nežádoucích. [4]



Obr. 3.1: Příklad funkcí pro jasovou transformaci

3.1.2 Geometrické transformace

Geometrické transformace jsou v počítačovém vidění užívány zejména kvůli odstranění nežádoucího geometrického zkreslení, ke kterému často dochází při pořizování snímku. K této operaci je používána vektorová funkce, která původní body mapuje na základě pozic x , y na nové pozice x_{nove} , y_{nove} . Některé transformační funkce jsou předem známe. Mezi tyto funkce lze zařadit např. rotaci, translaci či změnu měřítka. Ostatní je třeba odvodit na základě původních a transformovaných obrazů. Tento úkon je rozdělen do dvou fází. [4]

V první fázi dochází k transformaci souřadnic pixelů pomocí tzv. Jacobiho determinantu. Ten udává informace o tom, jak se souřadnicový systém mění. Pokud neexistuje inverzní tvar transformace, lze ji prohlásit za singulární a hodnota determinantu je rovna 0. Jestliže je plocha obrazu neměnná, determinant odpovídá hodnotě 1. [4]

V druhé fázi transformace dochází k interpolaci jasu, jelikož výstupem první fáze je matice reálných čísel, která neodpovídá obrazovému rastru. Mezi běžně metody užívané k interpolaci jasu se řadí lineární, bikubická či metoda nejbližšího souseda. [4]

3.1.3 Lokální předzpracování

Úkolem lokálního předzpracování je nejen potlačení či úplné odstranění šumu ze snímku, ale i nalezení hran. Efektu eliminace šumu je často docíleno pomocí lehkého rozmazání obrazu. Při použití některé z metod lokálního předzpracování je nutné dbát především na to, aby touto operací nezakly potřebné informace, které se v obraze nachází.

Eliminace šumu

První často využívaná metoda lokálního předzpracování je vyhlazování pomocí Gaussova nebo Butterwortova filtru. Prakticky se jedná o speciální typ konvoluce, jež bude popsána v kapitole 3.3. Velikost konvoluční masky přímo ovlivňuje míru redukce šumu a rozmazání snímků.

Některé z metod se snaží eliminovat míru rozmazání tím, že průměrují pouze ty pixely, které splňují nějaké kritérium. Jedno z kritérií může být např. určení intervalu, ve kterém se pravděpodobně nachází jasová úroveň šumu. Zbytek pixelů není k průměrování použito. Jako další kritérium lze použít např. velikost změny jasu v určitém intervalu. [4]

Průměrování pomocí rotující masky se řadí, na rozdíl od předešlých metod, k metodám nelineárním. Následkem toho není rozmazání hran příliš markantní. Tato metoda vytváří kolem každého pixelu devět masek, ze kterých je použita ta s nejmenším rozptylem. [4]

Hranové detektory

Detektory hran jsou velmi důležitou součástí lokálního předzpracování. Hranu lze chápat jako vlastnost, jež je přiřazená ke každému pixelu a je vypočítávána z obrazové funkce v okolí daného bodu. Jedná se o vektorovou proměnnou, která nese informace o směru a velikosti. Detekce hran je méně náchylná na proměnlivé osvětlení a změnu úhlu pohledu. Tyto metody jsou používány především pro spolehlivé zobecnění obrazových dat, což je výhodné zejména pro jejich další zpracování pomocí umělých neuronových sítí. Při použití hranových detektorů ale může nastat problém se špatnou interpretací scény, protože se z obrazu vytratí důležité informace. K detekci hran slouží tzv. operátory, které lze chápat jako konvoluční masky. [4]

Mezi velmi populární operátory patří například Sobelův operátor, který je často používán jako detektor horizontálních a vertikálních hran. Matice h_1 slouží pro detekci horizontálních hran. Operátor, který je znázorněn pomocí matice h_3 , se používá jako detektor vertikálních hran. [4]

$$\mathbf{h}_1 = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}, \mathbf{h}_2 = \begin{bmatrix} 0 & 1 & 2 \\ -1 & 0 & 1 \\ -2 & -1 & 0 \end{bmatrix}, \mathbf{h}_3 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (3.1)$$

Často používanou metodou k nalezení hran je i Cannyho detektor. Ten se zaměřuje primárně na tři kritéria. První kritérium dbá na správnost výsledků. Při použití tohoto detektoru by nemělo docházet k nacházení falešných hran. Další kritérium souvisí se vzdáleností mezi detekovanou a skutečnou hranou. Ta by měla být minimální. Poslední podmínkou je zaručení, že hrana původního obrazu bude detekována pouze jednou. [4]

3.2 Segmentace

Hlavním úkolem segmentace je rozdělení obrazu na části, které mají silnou korelaci s objekty z reálného světa. Každá tato část je reprezentována uzavřenou hranicí. Tyto hranice jsou zpravidla vytvářeny pomocí tří metod. [4]

První metoda se nazývá prahování. Tato metoda je relativně rychlá, jelikož se řadí mezi ty jednodušší. Pomocí konstantního prahu je obraz rozdělován na objekty a pozadí. Hodnoty prahu jsou většinou určovány automaticky a lze je aplikovat na celý obraz, nebo pouze na určitou část. V reálném prostředí je velmi těžké dosáhnout uspokojivých výsledků za použití pouze jednoho prahu, proto se většinou využívá tzv. lokálního prahování. Hodnota prahu se obvykle určuje pomocí histogramu. [4]

Segmentace založená na detekci hran využívá pro svou práci hranové detektory. Ty jsou společně s dalšími metodami schopny od pozadí separovat objekt, který se nachází v obraze. Jako příklad jedné z metod lze uvést např. Hougovu transformaci, kterou lze použít na objekty známého tvaru. U hledaných objektů je ale nutné znát jejich analytické rovnice. Tato metoda je velmi odolná proti šumu a velmi úspěšná při rozpoznávání zakrytých oblastí. [4]

Poslední metoda rozděluje obraz pomocí regionů. Jedním z postupů určování segmentů v obraze je slučování regionů. Nejprve je v obraze vytvořeno velké množství segmentů, které jsou na základě určitých kritérií slučovány do sebe. Opačný postup je aplikován v další metodě, kde je na začátku zpracování obrazu vytvořen jeden segment, který je následně dělen na menší celky tak, aby tyto oblasti splňovali dané podmínky. Složitější algoritmy mohou kombinovat segmentaci vytvořenou pomocí regionů společně s použitím hranových detektorů. [4]

3.3 Konvoluční neuronová síť - CNN (Convolutional neural network)

Konvoluční neuronová síť je mimořádně efektivní ve zpracovávání obrazových a zvukových informací, proto je často využívána pro rozpoznávání objektů v nasnímané scéně. Jedná se o dopřednou neuronovou síť, která obsahuje několik typů vrstev, proto patří do skupiny sítí hlubokého strojového učení.

Tato síť funguje na principu učení s učitelem. Síti jsou tedy předkládány vzory, podle kterých síť nastavuje hodnoty vah v jednotlivých neuronech, které se nachází v jednotlivých vrstvách. Vzory jsou předkládány do doby, kdy je dosaženo uspokojivých výsledků. Předkládané vzory jsou v zahraniční literatuře označovány jako dataset.

Jednotlivé vrstvy v počátku zpracování obrazového materiálu extrahují jednoduché rysy jako jsou např. barvy a hrany. Jak obrazové informace prostupují skrytými vrstvami této sítě, začínají být rozpoznávány větší obrazové celky. Výstupem je pak rozpoznáný objekt.

3.3.1 Konvoluční vrstva

Tato vrstva tvoří základ konvoluční neuronové sítě, jelikož extrahuje jednotlivé příznaky ze vstupního obrazu pomocí konvoluce. Vrstva se skládá z několika komponent. Tvoří ji vstupní vrstva, filtr (konvoluční jádro) a výstupní matice zvaná jako mapa příznaků, jejichž výsledná velikost závisí na velikosti vstupní matice a velikosti filtru. Hloubka této vrstvy se pak odvíjí od počtu použitých konvolučních jader. Filtr obvykle má rozměr 3×3 pixelů, může mít ale libovolnou velikost. Tato vrstva je postupně posouvána přes vstupní matici, kdy je v každém kroku spočítán jeden prvek výsledné mapy příznaků. Velikost posunu filtru mezi kroky udává parametr *stride*. Dalším důležitým parametrem je *padding*. Tento parametr udává, o kolik řádků a sloupců nul byla vstupní matice rozšířena, aby nedocházelo ke zmenšování vstupní matice, a bylo tak možné vytvářet hluboké konvoluční neuronové sítě.

Pokud se velikost kroku posunu filtru rovná jedné a není použit *padding* pro výpočet mapy příznaků, platí tento vztah:

$$y(m, n) = \sum_{i=0}^{A-1} \sum_{j=0}^{B-1} f(i, j) \cdot x(m - i, n - j) \quad (3.2)$$

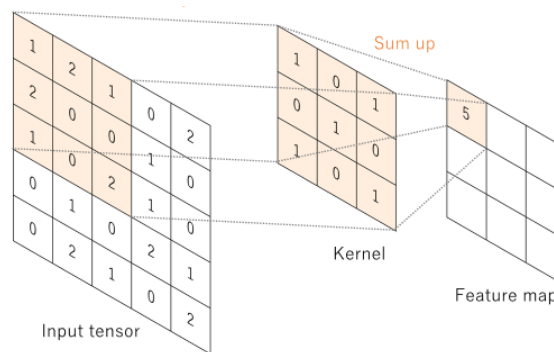
y ...mapa příznaků

m, n ...souřadnice aktuálně počítaného prvku mapy příznaků

A, B ...rozměry filtru

f ...filtr

x ...vstupní matice



Obr. 3.2: Znázornění konvoluce[10]

Na výslednou mapu příznaků je aplikována nelineární aktivační funkce ReLU, jež do systému vnáší nelinearitu.

3.3.2 Pooling vrstva

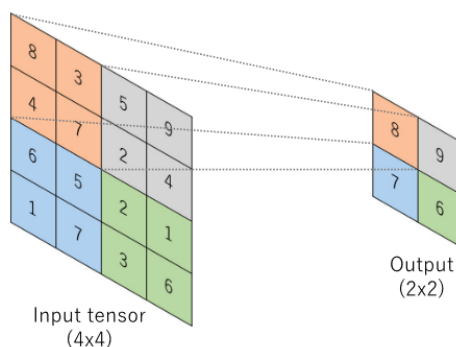
Tato vrstva snižuje počet parametrů na vstupu, což zvyšuje efektivitu konvoluční neuronové sítě a omezuje riziko nadprůměrného přizpůsobení. Aplikace této vrstvy na vstupní vrstvu probíhá podobně jako v případě konvoluční vrstvy. Na rozdíl od konvoluční vrstvy ale neobsahuje žádné váhy. Lze se ovšem setkat s parametrem *stride* stejně jako u konvolučních vrstev. Nejčastěji používaná velikost jádra je 2×2 . Pro výpočet výstupní vrstvy se nejčastěji používají dva přístupy, které budou popsány v následujících odstavcích:

Max-pooling vrstva

Jedná se o nejčastěji používaný typ pooling vrstvy. Ze vstupní vrstvy je vždy vybrán pixel s největší hodnotou. Tato hodnota je zapsána do výstupní vrstvy a poté dochází k posunu jádra. Na obrázku č. 3.3 je znázorněn příklad operace max-pooling.

Average-pooling vrstva

Tento typ vrstvy se obvykle používá pouze jednou před použitím fully-connected vrstvy, jelikož snižuje počet parametrů, které se lze naučit a umožňuje přijímat vstupy o proměnlivé velikosti. Tato vrstva funguje na jednoduchém principu průměrování vybraných pixelů.



Obr. 3.3: Příklad operace max-pooling[10]

3.3.3 Fully-Connected vrstva

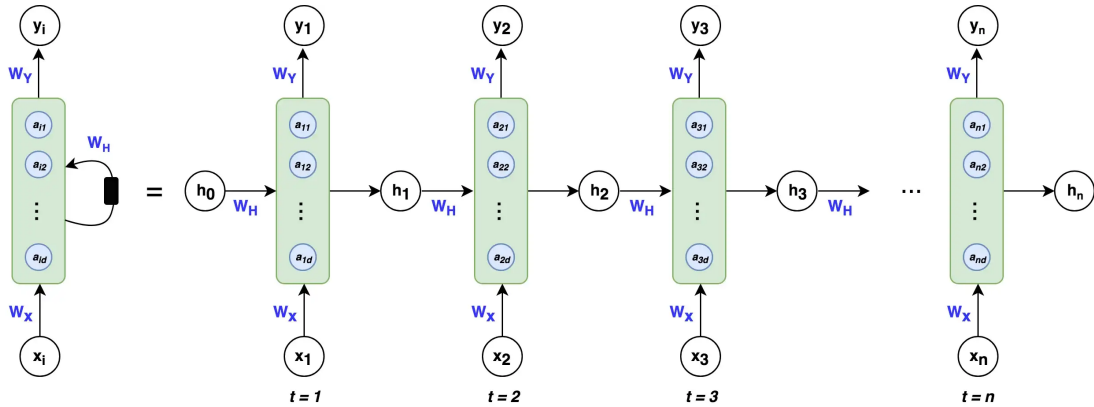
Tato vrstva se nachází na konci celého procesu, kdy je mapa příznaků zploštělá pouze do jednorozměrného pole. Jak již název napovídá, tato vrstva propojuje vstupní data s výstupními daty pomocí nastavených vah, čímž zajišťuje klasifikaci objektů na základě rysů, které byly extrahovány v předchozích vrstvách. Tato vrstva obvykle nevyužívá aktivační funkci ReLU (Rectified Linear Units), nýbrž je používána funkce softmax. Tato aktivační funkce vhodně klasifikuje vstupy s pravděpodobností od 0 do 1.

3.4 Rekurentní neuronová síť - RNN (Recurrent neural networks)

Konvoluční neuronová síť dokáže velmi dobře klasifikovat objekty v obrazu. V případě videa tak rozpoznává objekty samostatně v jednotlivých snímcích. Nedokáže ale vyvodit jakoukoliv spojitost mezi snímky. Jelikož rekurentní neuronová síť zpracovává data cyklicky, je určena pro interpretaci časových a sekvenčních dat. Díky tomu lze predikovat některé sekvence.

Tento typ neuronových sítí je velmi často používán pro rozpoznávání jazyka nebo psaného textu. V automobilovém průmyslu je často používán pro sledování pohybujícího se objektu, jako jsou například jedoucí vozidla nebo chodci. Lze jimi

predikovat i pravděpodobnou dráhu objektu. Toto je potřebné zejména u asistence nouzového brzdění (AEB) nebo asistence nouzového zásahu do řízení (ESA). Jelikož tato neuronová síť dokáže lépe zachytit kontext scény, v některých případech může být zařazení tohoto typu sítě mnohem efektivnější v rozpoznávání jednotlivých objektů, které se nachází v okolí vozidla.



Obr. 3.4: Rekurentní neuronová síť[13]

Na obrázku č. 3.4 znak x reprezentuje vstupy, y představuje predikované výstupy a h uchovává informace o předešlých vstupech. Základní rekurentní neuronová síť vrací v každém časovém kroku jeden výstup:

$$y_t = f(W_{ht}h_t + b_y) \quad (3.3)$$

y_t ...výstup

W_{ht} ...matice vah

h_t ...výstup skryté vrstvy

b_y ...bias

Parametr h je spočítán pomocí následující rovnice:

$$h_t = f(W_{xh}x_t + W_{hh}x_{t-1} + b_h) \quad (3.4)$$

W_{xh} , W_{hh} ...matice vah

Základní architektura rekurentní neuronové sítě ale není schopna zpracovat delší sekvence dat kvůli prudce rostoucímu, nebo naopak mizejícímu gradientu. Chyby vypočítané při zpětném šíření se totiž v každém kroku navzájem vynásobí. Z tohoto důvodu byly vytvořeny komplexnější algoritmy, například LSTM (Long short-term memory network) nebo GRU (Gated recurrent unit), které tento problém potlačují.

4 Robotické platformy

Tato práce se má zabývat vytvořením funkčního modelu vozidla určeného pro autonomní jízdu, proto je důležité vybrat správný typ podvozku. V současnosti je v robotice používáno několik druhů podvozků. Každý typ se hodí do rozdílného prostředí, liší se také schopností prostupovat terénem. Při výběru vhodného podvozku je nutné zamyslet se nad náročností jeho řízení. V následující kapitole budou nejprve popsány jednotlivé typy podvozků, jejich výhody a nevýhody.

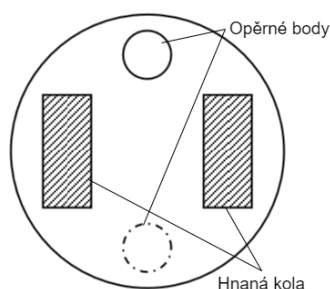
4.1 Typy podvozků

V této podkapitole budou popsány základní typy podvozků, které by mohly být vhodné k účelům této práce.

4.1.1 Diferenciální podvozek

Diferenciální podvozek je tvořen dvojicí kol s nezávislými pohony. Tato kola jsou obvykle umístěna na dvou protilehlých stranách robota. Aby byl podvozek stabilní, je často doplněn o nehnané kolo či nějaký kluzný bod, který tvoří podpůrnou konstrukci.

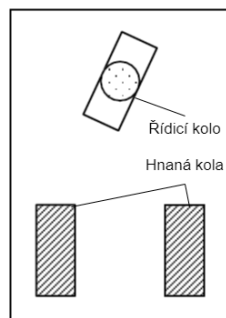
Tento typ podvozku je v robotice velmi rozšířen, jelikož díky tomuto podvozku se dokáže robot otočit na místě. Navíc řízení tohoto podvozku patří k nejjednodušším, jelikož změnu směru lze vyvolat pouhou změnou úhlové rychlosti nebo změnou směru otáčení jednoho z kol. Problémem tohoto pohonu je především dosažení jízdy v přímém směru, jelikož hnací kola jsou nezávislá. Motory totiž nejsou identické a přidružené mechanické části (např. převodovky) mají téměř vždy odlišné tření. Popsaný problém způsobuje jev, kdy při nastavení stejné rychlosti u motorů je jeden z motorů pomalejší, což neblaze ovlivňuje směr robota. Často se tedy využívá softwarové korekce parametrů, kdy se zadaná rychlost násobí vhodnou konstantou. Díky tomu je kompenzováno např. odlišné tření obou motorů.[15]



Obr. 4.1: Diferenciální podvozek[15]

4.1.2 Tříkolový podvozek s řízeným předním kolem

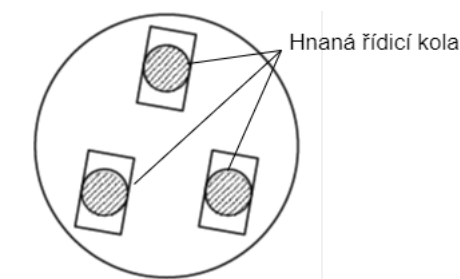
Součástí tohoto typu podvozku je nehnané přední kolo, pomocí kterého lze měnit orientaci vozidla, a dvojce hnaných kol. Těmito koly lze uvést vozidlo do pohybu. Tento typ podvozku zpravidla obsahuje diferenciál, aby se předešlo mechanickému poškození hnacích kol. Výhodou tohoto podvozku je především lepší průchodnost terénem, nicméně ve stejné konfiguraci, jako je uvedena na obrázku č. 4.2, není robot příliš stabilní. [15]



Obr. 4.2: Tříkolový podvozek[15]

4.1.3 Synchronní podvozek

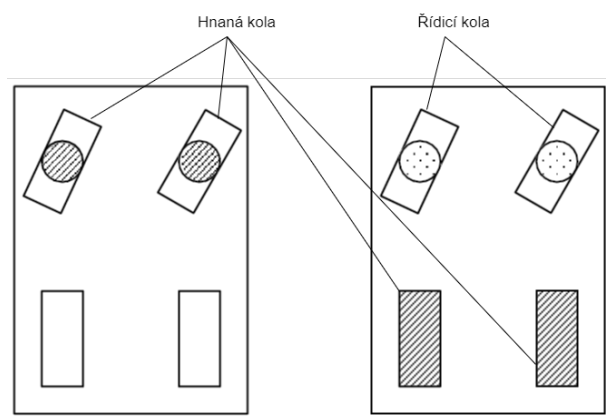
Synchronní podvozek obsahuje dvojici motorů. Jeden z motorů zajišťuje translační pohyb. Druhý z motorů zajišťuje změnu směru. U tohoto typu podvozku jsou všechna kola spřažena, což vyžaduje poměrně náročnou konstrukci. Manévrovatelnost a průchodnost terénem je u této konstrukce podvozku velice dobrá. Nevýhodou může být menší stabilita, tu lze ale kompenzovat přidáním dalšího kola. Pokud podvozek obsahuje více než tři kola, je nutné vytvořit nějaký tlumicí systém, který zajistí, že se všechna kola budou dotýkat podložky, což opět výrazně komplikuje konstrukci podvozku. [15]



Obr. 4.3: Synchronní podvozek[15]

4.1.4 Ackermanův podvozek

Tento typ podvozku většinou obsahuje dva páry kol. Přední kola ovlivňují směr jízdy. Zadní kola obsahují diferenciál a zajišťují translační pohyb. Nejedná se ovšem o jedinou možnost. Často používanou variantou je, že k pohonu slouží přední kola. Existuje i možnost, kdy jsou poháněna všechna čtyři kola, což zlepšuje průchodnost terénem. Tento typ podvozku je často užíván v automobilech a i v jiných vozidlech, protože je tento podvozek mimořádně stabilní ve vyšších rychlostech a jeho řízení je pro člověka velmi přirozené. Průchodnost i náročnějším terénem je také velmi dobrá. Nevýhodou takového podvozku je ale složitá konstrukce a horší manévrovatelnost, jelikož se s tímto podvozkem není možné otočit na místě. Podvozek je také náročnější autonomně řídit, jelikož malá chyba v natočení kol může způsobit velké chyby odometrie. [15]



Obr. 4.4: Ackermanův podvozek[15]

4.2 Přehled podvozků na trhu

Na trhu v současnosti existuje nepřehledné množství různých typů stavebnic a platforem pro vytvoření autonomního vozidla. Jelikož celý trh již není možné zmapovat, budou v této práci vybrány pouze platformy z českých internetových obchodů, zahraničního internetového obchodu Botland, který má zastoupení v ČR, a jedno z největších internetových tržišť na světě s názvem Aliexpress.

4.2.1 Stavebnice s diferenciálním řízením

Platformy s tímto typem řízení jsou na trhu nejvíce rozšířeny. Může za to pravděpodobně fakt, že lze tyto podvozky snadno řídit a jejich konstrukce není příliš složitá.

Na internetové stránce laskakit.cz lze zakoupit stavebnici podvozku obsahující platformu podvozku z plexiskla, otočné kolo sloužící jako opěrný bod, držák na baterie typu AA a dvojice motorů s převodovkami, na které lze umístit kola o průměru 66 mm , jež jsou také obsažena v balení. Motory jsou určeny pro napětí $3 - 6\text{ V}$ a proud $100 - 120\text{ mA}$. Rychlost otáčení hřídelů po zpřevodování je rovna $100 - 240\text{ ot/min}$ v závislosti na napětí. Cena této stavebnice je rovna 268 Kč (březen 2023). Stavebnice nabízí také velké množství montážních otvorů pro výpočetní platformy, snímače, akumulátory a jiné periferie, které jsou potřebné pro provoz robota. Tato stavebnice ve složeném stavu je vyobrazena na 4.5a.

Stejný internetový obchod nabízí také jinou variantu této stavebnice, která obsahuje čtyři motory, tudíž by autonomní vozidlo mělo mít čtyři hnaná kola. V balení této stavebnice se nachází podobné součásti, jako obsahuje dvoukolová varianta, jen je uzpůsobeno pro čtyřkolovou konfiguraci. V balení tedy nenajdeme otočné kolo. U této varianty je ale možnost zvolit si barvu plexiskla. Cena této stavebnice činí 438 Kč (březen 2023). Platforma je uvedena na obrázku č. 4.5b

Poslední stavebnicí z tohoto internetového obchodu, která by byla vhodná jako fyzická platforma pro výukový přípravek, je produkt s označením Makeblock MBot klon. V balení lze nalézt dva motory s převodovkami, které mají stejné parametry jako u předešlých produktů. Dvě kola o průměru 66 mm a jednu kovovou otočnou kouli sloužící jako opěrný bod, jelikož se jedná také o diferenciálně řízenou platformu. Produkt se výrazně liší od ostatních zmíněných stavebnic především materiálem, ze kterého je platforma zhotovena. V tomto případě byl na vytvoření podvozku použit hliník. Cena tohoto produktu je 428 Kč (březen 2023). Produkt se nachází na obrázku č. 4.5c.

Zahraniční internetový obchod s názvem Botland má ve své nabídce stavebnici, která opět obsahuje dvojici motorů s převodovkami, dvojici kol o průměru 65 mm , kovovou otočnou kouli a držák baterií, do kterého lze vložit pět článků typu AA. Motory mají stejné parametry jako motory dodávané v ostatních stavebnicích. Konstrukce je vyrobena z plastu. Výhodou tohoto podvozku je především velký počet montážních otvorů. Cena stavebnice činí 429 Kč (březen 2023). Vyobrazenou platformu lze nalézt na obrázku č. 4.5d.

4.2.2 Stavebnice s Ackermanovým řízením

Podvozky s tímto typem řízení jsou často používány v RC (Radio Controlled) modelech, jelikož tento typ podvozku je používán u většiny dopravních prostředků pohybujících se na vozovce. Řízení modelu tak simuluje řízení skutečného vozidla se všemi jeho aspekty. Konstrukce takových podvozků je složitější na výrobu a zpravidla obsahují vyšší počet dílů. Tomu odpovídá i cena jednotlivých stavebnic. Trh se



(a) Stavebnice podvozku - 2 kola [16]



(b) Stavebnice podvozku - 4 kola [17]



(c) Makeblock MBot podvozek klon [18]



(d) Magician Chassis v2 [20]

Obr. 4.5: Diferenciálně řízené podvozky

stavebnicemi Ackermanových podvozků není tak rozsáhlý, jako je tomu u platform s diferenciálním řízením.

Stavebnice s názvem JetRacer, kterou má ve své nabídce internetový obchod Botland, obsahuje kromě samotné platformy i část řídicí elektroniky a ostatní příslušenství, jako je např. kamera, jež je potřebná k vytvoření přípravku pro autonomní řízení. Od tohoto faktu se odvíjí i cena. Celou stavebnici lze pořídit za *6 881 Kč* (březen 2023). Balení obsahuje hliníkovou platformu, na které jsou umístěny dva DC (direct current) motory s převodovkami. Tyto motory jsou určeny pro napětí *12 V* a jejich rychlost otáčení je *740 ot/min*. O natáčení předních kol se stará servopohon s označením MG996R. Toto servo vyžaduje napětí *4,8 V* a jeho točivý moment je *9 Kg · cm*. Součástí balení jsou také kromě zmíněných komponent čtyři kola, spojovací materiál, nářadí potřebné k postavení stavebnice, řídicí deska, jež obsahuje WiFi adaptér, a kamerový modul, který má následující parametry: Rozlišení *8 MPx*, zorné pole *160°*. V balení se ovšem nenachází řídicí počítač Nvidia Jetson Nano, pro který byla řídicí deska navržena. Ten je nutný dokoupit samostatně. Produkt je uveden na obrázku č. 4.6a.

Stavebnice 4WD Ackerman Steering RC Car lze zakoupit na internetovém tržišti s názvem Aliexpress. Cena tohoto produktu je 1794 Kč (březen 2023) a nezahrnuje náklady na dopravu. Jedná se o platformu s pohonem všech čtyř kol, jež má Ackermannův způsob řízení. Pohon obstarávají čtyři motory s převodovkami s označením GB37-520, které dosahují při napětí 12 V přibližně 330 ot/min . Na tyto motory jsou také připojeny rotační enkodéry pro měření otáček, což je potřebné zejména pro odometrii. Servo s označením MG996r slouží k natočení kol pro změnu směru jízdy. Toto servo dosahuje točivého momentu $9,8\text{ Kg} \cdot \text{cm}$ při napětí $4,8\text{ V}$. Platforma je vyrobena z plastu. Sestavená stavebnice lze vidět na obrázku č. 4.6b.

Podvozek s názvem 4WD Smart Steering Robot Car lze zakoupit za 1604 Kč (březen 2023) také na tržišti Aliexpress. Náklady na dopravu nejsou do ceny započítány a jsou přibližně o 200 Kč vyšší než u předešlé stavebnice. Balení obsahuje platformu vyrobenou z kovu doplněnou o potřebné držáky a ostatní komponenty, jež jsou vyrobeny z plastu nebo plexiskla. Uvnitř lze nalézt i potřebný spojovací materiál a čtyři kola. Prodejce uvádí, že podvozek má hnaná všechna 4 kola, nicméně pohon obstarává pouze jeden DC motor, který není blíže specifikován. Natáčení kol je realizováno pomocí serva, jehož parametry prodejce také neuvádí. Obrázek sestavené platformy lze nalézt na obrázku č. 4.6c.



(a) JetRacer [21]



(b) 4WD Ackerman Steering RC Car [22]



(c) 4WD Smart Steering Robot Car [23]

Obr. 4.6: Ackermanovy podvozky

5 Výpočetní platformy

V současnosti existuje několik přístupů řízení vozidel a jiných dopravních prostředků. Tyto způsoby jsou využívány v celé řadě odlišných aplikací. V této kapitole bude uveden stručný přehled základních přístupů a několik komerčně dostupných výpočetních platforem, které se jeví jako vhodné pro použití v této práci.

5.1 Obecné architektury řízení

Řízení robota lze řešit několika způsoby. Jeden z elementárních přístupů, jak robotickou platformu řídit, je použití manuálního ovládání, kde operátor řídí veškeré akce systému. Řídicí jednotka nezpracovává data a neobsahuje žádné algoritmy pro řízení vozidla.

Mezi vyšší způsoby řízení lze zařadit tzv. autopilot. Tento přístup již vyžaduje senzorické vybavení, jelikož řídicí jednotka již zasahuje operátorovi do řízení, a napomáhá mu tak v řízení. Jako příklad tohoto způsobu řízení lze uvést adaptivní tempomat, případně i jiné asistenční systémy uvedené v kapitole 1.2. Vozidlo totiž dokáže díky senzorickému vybavení upravit svoji rychlost tak, aby nedošlo ke kolizi s ostatními objekty, které se nachází v jeho blízkosti. Směr pohybu vozidla ale stále ovlivňuje řidič vozidla.

Autonomní řízení patří mezi nejnáročnější přístupy. Systém totiž musí obsahovat značné množství senzorů, aby měl dokonalý přehled o situaci v jeho okolí, a dostatečně výkonnou výpočetní jednotku, jelikož je nutné použití pokročilých algoritmů řízení, počítačového vidění atp. Systém je v některých případech schopen např. predikovat trajektorii sebe sama pomocí umělých neuronových sítí.

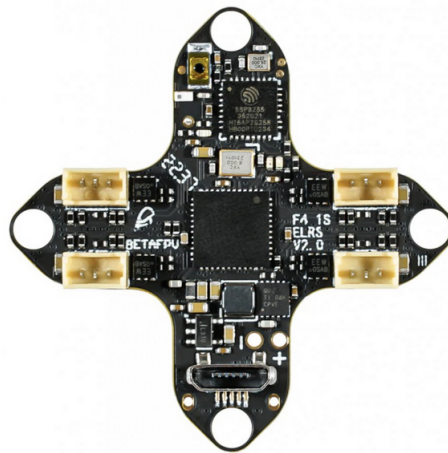
5.2 Přehled výpočetních platforem na trhu

V současnosti se na trhu nachází nepřehledné množství výpočetních platforem a řídicí elektroniky pro řízení RC modelů nebo robotických platforem. V této kapitole bude blíže popsáno několik z nich, přičemž největší důraz bude kladen na výpočetní platformy pro nejvyšší úroveň řízení.

Manuální řízení je totiž tak triviální, že se často využívají pouze jednoduché obvody, které například generují PWM (Pulse Width Modulation) signál pro řízení pohonu na základě zpracování povelů operátora. Lze také využít přijímač radiového signálu, jenž je využíván u RC modelů.

Poloautonomní způsob řízení vyžaduje již sofistikovanější elektroniku. Jako zástupce těchto řídicích jednotek byla vybrána řídicí jednotka s označením F4 5A AIO FC, která je určena primárně pro řízení FPV (first person view) dronů. Tato

jednotka obsahuje mikrokontrolér STM32F411CEU6, který je schopen pracovat až na frekvenci 100 MHz . Disponuje také FLASH pamětí o velikosti 512 KB , SRAM s kapacitou 128 KB , 8 časovači a několika 12-bitovými AD (analogově digitální) převodníky. Tento mikrokontrolér podporuje také komunikaci skrz SPI, I2C i USART. Řídicí jednotka obsahuje blíže nespecifikované ovládání motorů s maximálním kontinuálním proudovým zatížením 5 A . RC přijímač s označením ExpressLRS 2.4G, který je součástí řídicí jednotky, slouží pro radiovou komunikaci s vysílačem běžící na standardní frekvenci $2,4\text{ GHz}$. Řídicí jednotka disponuje také šestiosým gyroskopickým snímačem BMI270, jenž je schopen komunikovat s mikrokontrolérem pomocí rozhraní SPI či I2C. Cena tohoto produktu činí na stránkách internetového obchodu s názvem FYFT 1349 Kč (březen 2023).



Obr. 5.1: Řídicí jednotka F4 5A AIO FC[26]

Plně autonomní řízení je velmi náročné na výpočetní výkon hardwaru. Za prvního zástupce těchto výkonnějších výpočetních platform lze považovat například mikrokontroler s názvem ESP32. Toto zařízení je velmi dobře dostupné napříč všemi internetovými obchody samostatně nebo s různými variantami periférií. Tento mikrokontrolér může obsahovat 32-bitový procesor, který má jedno či dvě jádra s maximálním taktům 240 MHz . Dále disponuje ROM pamětí o velikosti 448 KB , SRAM pamětí s kapacitou 520 KB , 12-bitovým převodníkem s 18 kanály a komunikačním rozhraním SPI, I2C, UART, WiFi i Bluetooth. Jelikož je pro autonomní řízení výhodné použití kamery pro snímání okolí, byl pro tento průzkum trhu vybrán modul obsahující kamerový modul s označením OV2640. Snímač kamery poskytuje snímky o rozlišení až $1600 \times 1200\text{ MPx}$. Celý tento set nabízí internetový obchod laskakit.cz za částku 358 Kč (březen 2023).

Podstatně výkonnější výpočetní jednotkou je například jednodeskový počítač Raspberry Pi. Jeho nejnovější verze nese označení 4 a na stránkách oficiálního čes-

kého distributora lze pořídit za *1650 Kč* (březen 2023) ve verzi se *4 GB* paměti RAM. Kromě této varianty výrobce nabízí i verze s *1, 2 a 8 GB* RAM. Dále počítač obsahuje 64-bitovým procesor se čtyřmi jádry, která mají takt *1,8 GHz*. Na desce tištěných spojů se nachází dva porty USB 2.0, dva porty USB 3.0, jeden port USB-C, jenž slouží pro napájení, 2 microHDMI porty, konektor pro MIPI DSC display, konektor pro MIPI CSI kameru, audio jack a konektor RJ-45 pro připojení až gigabitového Ethernetu. Počítač také podporuje komunikační rozhraní WiFi, Bluetooth, SPI, I2C i UART. Velikost paměti je určena velikostí MikroSD karty, která s deskou standardně není dodávána.

Alternativou k počítači Raspberry Pi je například také jednodeskový počítač s označením Banana Pi Bpi, jehož nejnovější verze nese označení M6. Tato varianta se ovšem na českých internetových obchodech neprodává, proto bude v této práci popsána starší verze s označením M5. Tento počítač disponuje 64-bitovým procesorem se čtyřmi jádry, které běží na taktu *2 GHz*. Paměť RAM má kapacitu *4 GB*. Velikost interní FLASH paměti činí *16 GB* a paměť je možné rozšířit i pamětovou kartu typu MicroSD ve velikosti až *256 GB*. Na desce se nachází jeden HDMI port, audiojack, čtyři porty USB 3.0, a konektor pro Ethernet. Kromě Ethernetu deska nabízí i tyto komunikační rozhraní: UART, I2C, SPI. Pokud by chtěl uživatel využít rozhraní WiFi, či Bluetooth, lze zakoupit rozšiřující kartu, jež se zapojí do GPIO sběrnice, případně je možné využít USB rozšiřující karty. Cena tohoto jednodeskového počítače je na českém internetovém obchodě Conrad *2559 Kč* (březen 2023).

Posledním zástupcem jednodeskových počítačů, který bude v této práci zmíněn, je počítač od firmy NVIDIA s označením Jetson Nano Developer Kit. Toto zařízení obsahuje 64-bitový čtyřjádrový procesor s taktom *1,43 GHz*, který spolupracuje s pamětí RAM o velikosti *4 GB*. Na desce se nachází slot pro SD kartu, která souží jako uložště pro operační systém, micro USB port, čtyři USB 3.0 porty, HDMI konektor, DisplayPort, DC jack pro napájení, dva konektory pro MIPI CSI kameru a RJ-45 pro připojení Ethernetu. Komunikace s ostatními perifériemi lze zajistit pomocí SPI, I2C, I2S a UART. Na internetovém obchodu RPishop.cz lze tento vývojový kit pořídit za *4799 Kč* (březen 2023). Vyšší cena tohoto produktu se odvíjí především od jeho velmi výkonné grafické karty s 128 jádry, která je určena primárně pro zpracovávání algoritmů hlubokého učení a počítačového vidění.

5.3 Přehled rozšiřujících perifériích na trhu

Výuková platforma bude vyžadovat řízení motorů a sběr dat ze snímačů. Motory ale nelze připojit přímo k počítači skrz GPIO směrnice, jelikož maximální proudové zatížení výstupů například u Raspberry Pi činí pouhých *16 mA*, proto je potřeba doplnit výpočetní elektroniku dalším příslušenstvím. Důraz musí být kladen především



(a) Ai-Thinker ESP32-CAM [19]



(b) Raspberry Pi 4 [28]



(c) Banana Pi Bpi - M5 [27]



(d) Jetson Nano Developer Kit [29]

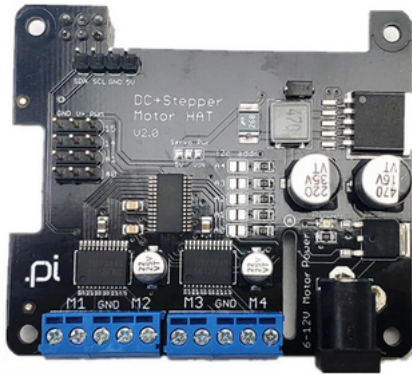
Obr. 5.2: Výpočetní platformy

na výběr vhodného H-můstku, kterým budou ovládány motory vozidla.

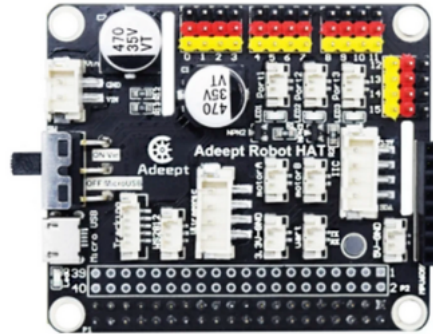
Na trhu se vyskytuje nepřehledné množství modulů. Mezi zástupce této skupiny produktů se řadí i rozšiřující deska s názvem DC Stepper Motor HAT. Modul obsahuje dva H-můstky s označením TB6612FNG, jejichž průměrné proudové zatížení může dosahovat až $1,2 A$. Na tuto desku je možné připojit až čtyři DC motory. Dále modul obsahuje konektory pro připojení až čtyř servopohonů, připojení periférií pomocí směrnice I2C, ochranou proti přepólování a regulátorem napětí, jehož výstupní napětí činí $5 V$. Na internetovém tržišti Aliexpress je tento rozšiřující modul dostupný za $613 Kč$ (březen 2023).

Za dalšího zástupce této skupiny lze označit modul s názvem Adept Robot HAT. Na této desce se nachází H-můstek s označením L289P, jehož průměrné proudové zatížení může činit až $2 A$. Tímto modulem lze řídit pouze dva DC motory. Rozšiřující deska dokáže ovládat až šestnáct servomotorů pomocí ovladače PCA9685, který komunikuje s nadřazenou řídicí jednotkou pomocí I2C sběrnice. Na modulu lze nalézt i regulátor napětí, přepínač, konektor microUSB pro napájení. Deska má dále vyvedené konektory pro modul MPC6050 (akcelerometr a gyroskop), směrnici I2C,

směrnici UART, ultrazvukový senzor, senzor pro detekci čáry, programovatelný LED pásek nesoucí označení WB2812 a konektory s vyvedeným napětím $3,3\text{ V}$ a 5 V . Cena tohoto modulu na tržišti Aliexpress je 918 Kč (březen 2023).



(a) DC Stepper Motor HAT [24]



(b) Adept Robot HAT [25]

Obr. 5.3: Rozšiřující moduly

6 Návrh a realizace výukového prostředku

Tato práce se zabývá vytvořením fyzického modelu vozidla, který má sloužit k demonstraci algoritmů užívaných se v autonomních vozidlech a také má být nápomocen ve výuce těchto algoritmů. Aby mohla tato výuková pomůcka vzniknout, je třeba zvolit vhodný hardware a vytvořit k němu odpovídající software. V následujících podkapitolách budou popsány oba tyto kroky.

6.1 Hardware

Kapitola o fyzických částech je rozdělena do dvou částí. První část se primárně bude zabývat mechanickou částí modelu vozidla. V druhé části bude popsána především výpočetní jednotka a ostatní periferie, které jsou potřebné pro algoritmy autonomního řízení a bezpečnost provozu modelu.

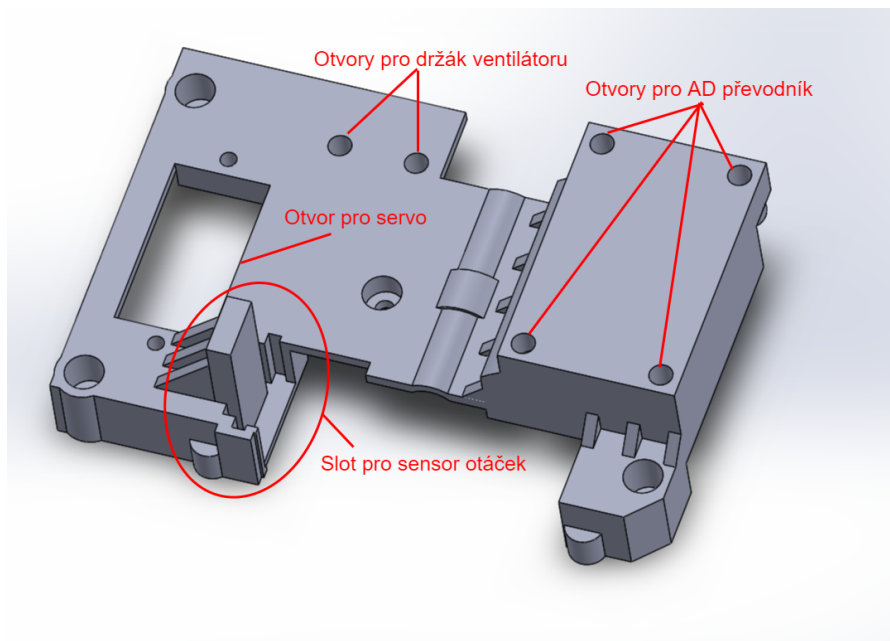
6.1.1 Mechanická část

Vytvořený robot má v maximální možné míře odpovídat reálnému vozidlu, byla zvolena platforma s Ackermanovým způsobem řízení. Tento typ podvozku byl blíže popsán v kapitole 4.1.4.

Při průzkumu trhu bylo vybráno hned několik vhodných stavebnic, nicméně žádný z návrhů, jenž jsou uvedeny v kapitole 4.2.2, nakonec nebyl použit především kvůli vysoké ceně. Náklady na stavbu této pomůcky totiž nemají být příliš vysoké. Z toho důvodu byl pořízen RC model vozidla s označením X-Knight RTR 4WD za 1090 Kč. Tento model disponoval dvěma DC motory a řídicí elektronikou, jež přijímala radiový signál o frekvenci 2,4 GHz generovaný vysílačem. Jeden z DC motorů byl určen pro pohyb vozidla. Druhý motor zajišťoval natočení kol, které bylo možno vychýlit pouze do tří stavů - doprava, doleva, rovně. Jelikož mechanická část řízení úhlu natočení kol společně s motorem neumožňovala plynulou regulaci úhlu, bylo nutné tuto část upravit. Autonomní robot bude totiž s velkou pravděpodobností využívat PSD regulátor, který vyžaduje přesné řízení úhlu natočení kol.

DC motor byl vyměněn za servomotor s označením MG90S. Toto servo obsahuje kovové převody a dokáže se pohybovat v rozsahu $0-180^\circ$. Jeho točivý moment při napětí 4,8 V činí 1,8 Kg · cm. Úhel natočení lze ovládat pomocí PWM signálu. Jelikož DC motor a servo nebylo možné zaměnit, bylo nutné odstranit několik plastových částí, které držely původní motor, a navrhnout nový způsob připevnění serva. Konečným řešením je přišroubování serva na horní kryt podvozku, který byl navržen v programu SolidWorks a vytištěn na 3D tiskárně z materiálu PETG. Vytvořený

díl je možné připevnit ke zbytku konstrukce pomocí původních šroubů. 3D model tohoto krytu je uveden na obrázku č. 6.1.

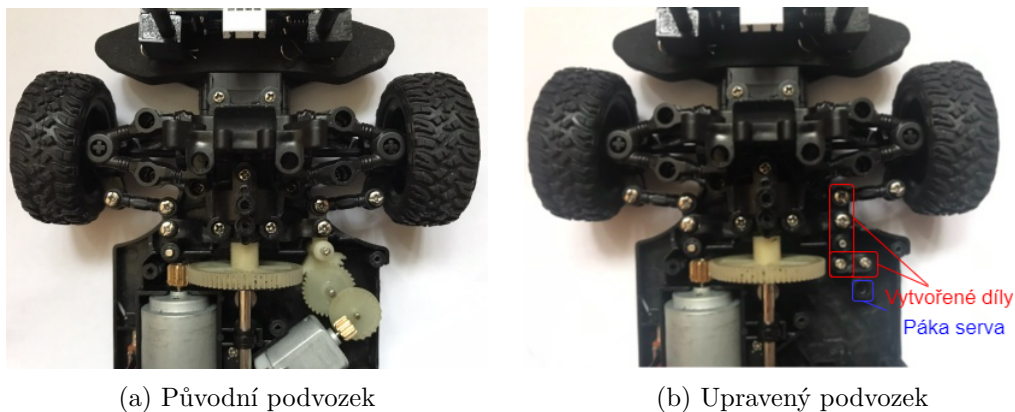


Obr. 6.1: Model horního krytu podvozku

U původního řešení bylo natáčení kol zajištěno pomocí soustavy táhel a ozubených kol. Tento způsob řešení nebyl pro použití serva vhodný. Z tohoto důvodu byly navrženy dva díly, jež umožnily propojení pohybu serva se stávající soustavou natočení předních kol. Na obrázku číslo 6.2a lze vidět původní řešení. Současné řešení je uvedeno na obrázku 6.2b. Vytištěné díly jsou na obrázku zvýrazněny pomocí červených rámečků. Modrým rámečkem je označena páka serva. Samotné servo na obrázcích není zobrazeno, jelikož překrývá svým objemem část upraveného mechanismu.

Na vytvořeném krytu podvozku se nachází ventilátor, jehož držák je také vytištěn na 3D tiskárně z materiálu PETG. Kvůli snazšímu tisku na 3D tiskárně byl tento díl navržen samostatně a je k hornímu krytu přišroubován pomocí dvojice šroubů. Na dílu je připevněn také senzor otáček, jenž snímá počet otáček hlavní hřídele, a je zde i prostor pro upevnění AD převodníku. Pro hlavní řídicí elektroniku byla vytvořena platforma, která lze osadit na místo původního krytu na elektroniku.

Zakoupené RC auto bylo vybaveno DC motorem, jehož nominální otáčky při napětí 6 V byly přibližně 16000 ot/min . Motor společně s převody, které vozidlo obsahovalo, umožnil robotu pohybovat se rychlostí až $5,6\text{ m/s}$, což je příliš vysoká hodnota. Pokud byly otáčky tohoto motoru sníženy pomocí regulace pod určitou mez, motor neměl dostatečný točivý moment, aby vozidlo rozpohyboval. Jelikož

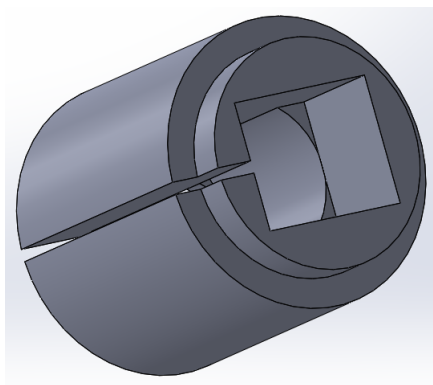


Obr. 6.2: Podvozek vozidla

byla ozubená kola nalisována na hřídele, nebylo možné změnit jejich poměr, navíc by v konstrukci pro větší ozubené kolo nebyl prostor.

Původní motor musel být vyměněn za jiný. Pro tuto práci byl vybrán DC motor, jehož součástí je i převodovka s předním soukolím. Výstupní hřídel tohoto motoru se při napětí 6 V otáčí rychlostí 1000 ot/min . Maximální rychlost vozidla je po záměně motoru $0,35\text{ m/s}$.

Jelikož nový motor měl jiné rozměry než původní motor, bylo potřeba navrhnout nový způsob uchycení. Tento problém byl vyřešen vytvořením adaptéru, který byl vytištěn na 3D tiskárně z materiálu PETG. Po zasunutí nového motoru do adaptéru jsou jeho rozměry totožné s původním motorem, proto ho bylo možné osadit na původní místo. Na obrázku č. 6.3 je tato redukce znázorněna.

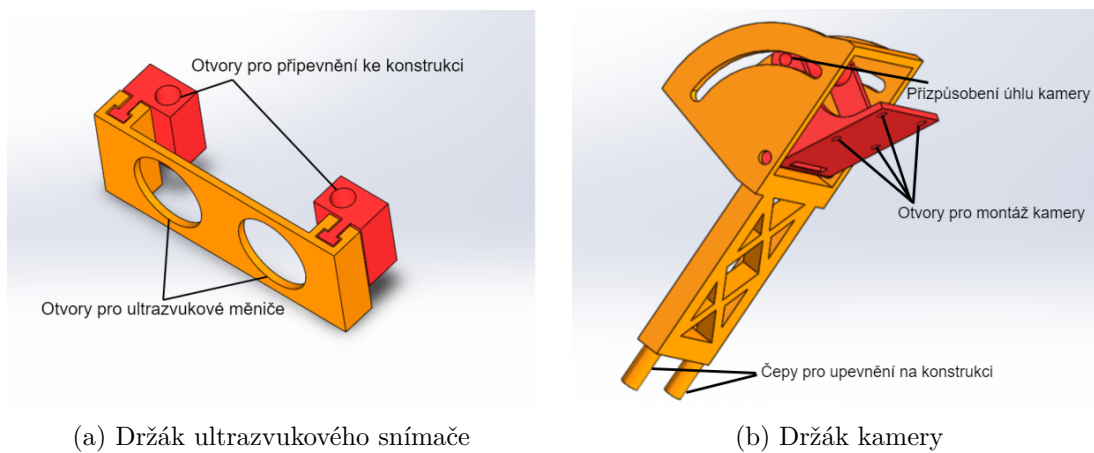


Obr. 6.3: Adaptér motoru

V programu SolidWorks byly také vytvořeny držáky pro snímače. Ultrazvukový snímač je připevněn pomocí pomoci tří dílů. Model je uveden na obrázku 6.4a. Jednotlivé díly jsou odlišeny barvami. Tento držák lze upevnit na osy původní konstrukce, které držely plastovou karoserii RC modelu.

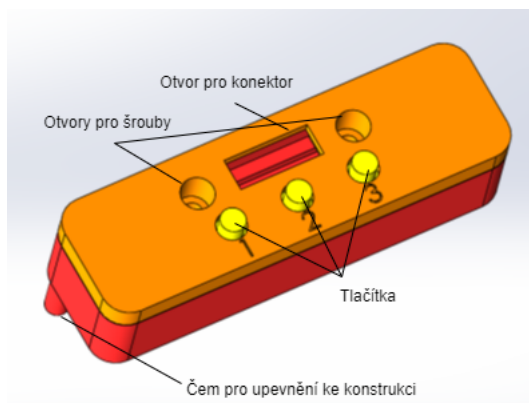
Další komponenty původně sloužící k upevnění karoserie byly použity pro držák kamery, jenž je uveden na obrázku 6.4b. Tento držák byl navržen tak, aby bylo možné upravit sklon kamery podle potřeby pouhým povolením dvojice šroubů.

Poslední navrženou částí je kryt pro tlačítka, které umožňují uživateli robot ovládat. Tato část je k robotu připevněna pomocí dvou čepů, které jsou zasunuty do otvorů v podvozku.



(a) Držák ultrazvukového snímače

(b) Držák kamery



(c) Držák tlačítek

Obr. 6.4: Navržené držáky snímačů

6.1.2 Elektronická část

Elektronická část je nedílnou součástí výukového prostředku pro autonomní řízení. Aby byly vybrány vhodné komponenty pro řízení RC modelu, je nezbytné vytvořit si seznam požadavků, které by měla elektronika splňovat. Touto problematikou se zabývá první podkapitola této části práce. V druhé části jsou popsány vybrané komponenty.

Požadavky

Stanovení požadavků je velmi důležité, jelikož jsou podle nich vybírány komponenty, které budou model řídit. V následujícím seznamu je uvedeno, jaké funkce by měla řídicí elektronika splňovat, aby bylo možné vytvořit plně autonomní vozidlo.

Seznam požadavků:

- Jednodeskový PC s dostatečným výkonem pro algoritmy počítačového vidění
- Kamera s dostatečným rozlišením určená ke snímání situace před vozidlem
- Ovládání rychlosti vozidla i řízení směru jízdy
- Měření vzdálenosti objektů před vozidlem
- Akumulátorový provoz

Řídicí komponenty

Nejdůležitějším komponentem řídicí elektroniky je bezesporu řídicí jednodeskový počítač. Pro tuto práci byl vybrán počítač nesoucí označení Raspberry Pi 4 Model B v konfiguraci se 4 GB RAM. Podrobnější specifikace tohoto zařízení jsou uvedeny v kapitole 5.2. Toto zařízení bylo vybráno především kvůli dostatečnému výkonu, přijatelné ceně a široké uživatelské základně.

Jelikož vybraný mikropočítač není přímo kompatibilní s připojením některých komponent, bylo nutné tuto výpočetní jednotku doplnit rozšiřující jednotkou. Pro tuto práci byla vybrána rozšiřující deska od společnosti Adept s označením Robot HAT. Bližší informace k této desce jsou uvedeny v kapitole 5.3. Tato rozšiřující deska obsahuje všechny požadované funkce, mezi které patří například připojení DC motoru, serva a senzorického vybavení. Navíc je tuto desku možné rozšířit několika dalšími zařízeními v případě potřeby doplnění funkce, kterou v současnosti robot nedisponuje.

Napájení zařízení

Rozšiřující deskou lze napájet kromě přímo připojených komponent i Raspberry Pi, jelikož v sobě obsahuje step-down měnič GS2678 pro regulaci napětí ze vstupního napětí na 5 V, přičemž vstupní napětí může dosahovat až 25 V. Takto vysoké vstupní napětí ale nelze použít, neboť originální DC motor byl původně napájen z akumulátoru, jehož nominální napětí činilo 6 V. Při použití vyššího napětí by tedy pravděpodobně došlo k přebuzení motoru, které by vedlo ke zvýšenému zahřívání motoru a snížení jeho životnosti. V krajním případě by mohlo dojít až ke zničení pohonné jednotky. Z toho důvodu byly vybírány akumulátory s nominálním napětím do 10 V, jelikož nepatrné přebuzení sníží životnost motoru minimálně.

Na výběr je v současnosti hned z několika typů akumulátorů. Původní RC auto bylo napájeno pomocí NiMH akumulátoru s nominálním napětím 6 V a kapacitou

přibližně 700 mAh . Tento akumulátor tedy pravděpodobně obsahuje 5 sériově zapojených článků. Výhodou tohoto zdroje energie je především jeho bezpečnost při používání, jelikož při proražení obalu prakticky neexistuje riziko požáru či výbuchu. V současnosti jsou často využívány akumulátory typu Li-ion či Li-Po. Tyto akumulátory jsou výhodné zejména kvůli schopnosti ukládat velké množství energie. Jako další výhodu lze zmínit i schopnost dodat velké vybíjecí proudy. V případě Li-Po akumulátorů mezi výhody patří i nízká hmotnost, nevýhodou těchto článků je ale vyšší cena. Dalším mínusem lithiových akumulátorů je náchylnost na podbití nebo zvýšené nebezpečí při jejich používání, jelikož zde existuje riziko vzplanutí. Další nevýhodou je nutnost použití složitější elektroniky pro nabíjení článků. Nominální napětí jednoho článku bývá rovno $3,6 \text{ V}$.

Akumulátor dodaný společně s RC modelem pro tuto práci nebyl vhodný. Proudová zatížitelnost v angličtině označovaná jako discharge rate je u NiMH akumulátorů přibližně 5 C . Maximální proud, jež je akumulátor schopný dodat, lze vypočítat podle následujícího vzorce:

$$I = Cr \cdot C \quad (6.1)$$

I ...výstupní proud [A]

Cr ...proudová zatížitelnost [C]

b_y ...kapacita [Ah]

Maximální vybíjecí proud dodaného akumulátoru je přibližně $3,5 \text{ A}$. Výrobce Raspberry Pi doporučuje zdroj elektrické energie, který je schopný dodat minimálně 3 A . K tomuto číslu je nutné připočítat i odběr motorů a ostatních periférií. Je patrné, že dodaný akumulátor není schopen dodat potřebný výkon. Články mají navíc i nízkou kapacitu. Z těchto důvodů byl vytvořen akumulátor ze dvou článků typu Li-ion, které byly zapojeny sériově. Výsledný zdroj energie má nominální napětí rovno $7,2 \text{ V}$ a jeho kapacita činí přibližně 2200 mAh . Jelikož proudová zatížitelnost Li-ion akumulátorů se pohybuje okolo 10 C , je tento akumulátor podle rovnice 6.1 schopen dodat až 22 A . Vytvořený akumulátor má podobné rozměry jako původní zdroj energie, lze ho tedy umístit do stejného prostoru. Nevýhodou lithiových článků je jejich citlivost na podbití, proto je nutné napětí na jednotlivých člancích měřit.

Vybraný mikropočítač bohužel neobsahuje žádné AD převodníky, které by byly schopné měřit napětí na bateriích, proto byl k Raspberry Pi připojen 8-bitový AD převodník s označením PCF8591. Tento AD převodník umožňuje až čtyřkanálové měření a s výpočetní jednotkou komunikuje přes sběrnici I2C. Kromě toho AD převodník disponuje také DA převodníkem, který v této práci ovšem nebude využit. Referenční napětí v katalogovém listu značeno jako V_{ref} je rovno 5 V . Podle následujícího vzorce lze vypočítat minimální kvantovací krok:

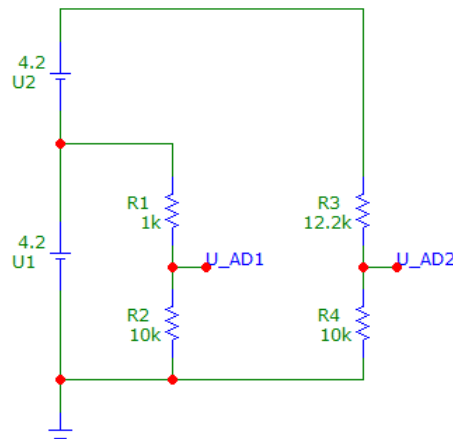
$$q = \frac{V_{REF}}{2^n} \quad (6.2)$$

q ...kvantovací krok [V]

V_{REF} ...referenční napětí [V]

n ...počet bitů [-]

Po dosažení hodnot do rovnice činí velikost kvantovacího kroku $19,53 \text{ mV}$. Takové rozlišení lze považovat pro měření stavu nabití akumulátoru za dostatečné. Aby bylo možné připojit monitorovací vývody na vstupní piny AD převodníků, bylo potřeba snížit vstupní napětí. Napětí na akumulátoru může dosahovat až $8,4 \text{ V}$, avšak na vstup lze připojit pouze napětí o hodnotě $5,5 \text{ V}$. Proto byl na modul AD převodníku byl vytvořen jednoduchý elektrický obvod na univerzálním tištěném spoji, který obsahuje dvojici napěťových děličů, jež snižují napětí z jednotlivých článků akumulátoru na požadovanou hodnotu. Schéma zapojení je uvedeno na obrázku 6.5.



Obr. 6.5: Napěťový dělič

Nejprve byly stanoveny hodnoty odporů $R1$ a $R3$, které omezují hodnotu proudu, který protéká obvodem. Jelikož je známo vstupní napětí, které činí maximálně $4,2 \text{ V}$ a $8,4 \text{ V}$, a žádané výstupní napětí, jež nemá překročit hodnotu $5,5 \text{ V}$, lze podle následujícího vzorce vypočítat hodnoty odporů $R2$ a $R4$:

$$R_2 = R_1 \frac{U_{AD1}}{U_1 - U_{AD1}} \quad (6.3)$$

R_2, R_1 ...hodnoty odporů [Ω]

V_1 ...vstupní napětí [V]

V_{AD1} ...výstupní napětí [V]

Výsledné hodnoty rezistorů jsou uvedeny na obrázku č. 6.5. Hodnoty výstupního napětí děličů nesoucí označení V_{AD1} a V_{AD2} jsou rovna přibližně $3,8 \text{ V}$. Hodnoty

odporu jednotlivých rezistorů byly vybírány i s ohledem na depozitář autora práce, proto není využit plný rozsah AD převodníku.

Senzorické vybavení

K tomu, aby se vozidlo mohlo pohybovat samostatně, je potřeba několik snímačů. Prvním z těchto snímačů je kamera s označením Raspberry Pi Camera 3 pořízená za *695 Kč* (březen 2023). K počítači se připojuje přes konektor CSI. Tato kamera disponuje CMOS snímačem nesoucí označení Sony IMX708, který má rozlišení *11,9 Mpx*. Modul má pevnou ohniskovou vzdálenost rovnu *4,74 mm*. Horizontální zorný úhel kamery činí *66°*. Kamera umožňuje také automatické ostření.

Dalším snímačem, kterým je robot osazen, je ultrazvukový snímač pro měření vzdálenosti s označením HC-SR04. Tento dvouhlavový snímač je umístěn v přední části vozidla tak, aby byl schopen měřit vzdálenost před vozidlem. Princip měření vzdálenosti pomocí ultrazvuku byl zvolen především kvůli rozdílnému způsobu získávání dat, než který vyžívají optické snímače. V řídicích algoritmech tak může být vytvořena fúze dat z obou snímačů, což může vést k přesnějším výsledkům. Snímač má navíc velmi dobrý dosah, jelikož dokáže měřit až do vzdálenosti *4 m*. Standardně tento senzor využívá dva datové signály s označením Trig a Echo. Novější verze podporuje i ostatní komunikační sběrnice jako jsou I2C, UART či 1Wire. Robot ale stále využívá starší způsob komunikace pomocí signálů Trig a Echo.

Jelikož je potřeba monitorovat otáčky hnacího motoru, ať už například pro měření ujeté vzdálenosti, či počítání aktuální rychlosti vozidla, byl robot vybaven optickou závorou, která na robotu funguje jako inkrementální enkodér. Tento snímač je za pomoci stínítka, které bylo vymodelováno, vytištěno 3D tiskárnou a nalepeno na ozubené kolo hlavního hřídele, schopno poskytnout až 25 pulzů na otáčku. Při použití kol o průměru *4,9 cm*, kterými byl RC model vybaven, takový počet pulzů odpovídá rozlišení *6,1 mm*. Tuto hodnotu lze pro tuto aplikaci považovat za dostatečnou. V případě potřeby je možno při použití jemnějšího stínítka generovat větší počet pulzů na otáčku, což by vedlo k vyššímu rozlišení snímače.

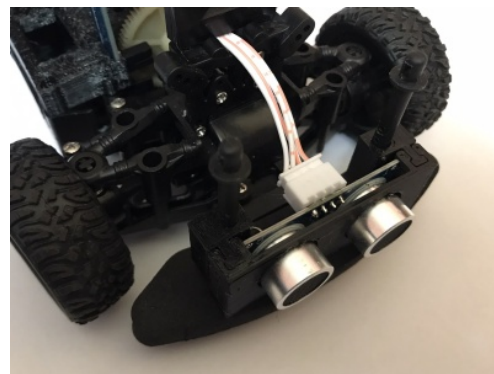
Mezi snímače lze také zařadit trojici tlačítek, které slouží k interakci robotu s uživatelem. Zapojení těchto snímačů je velmi jednoduché. Jeden kontakt tlačítek je připojek k signálu GND. Druhý kontakt byl připojen přímo do Raspberry PI, jelikož počítač obsahuje vnitřní pull-up rezistory.

Zapojení

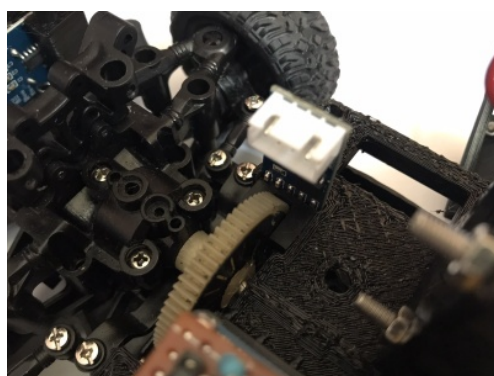
Na obrázku č. 6.7 lze vidět blokový diagram zapojení jednotlivých součástí k robotu. K výpočetní jednotce Raspberry Pi je připojen uvedený HAT, ke kterému



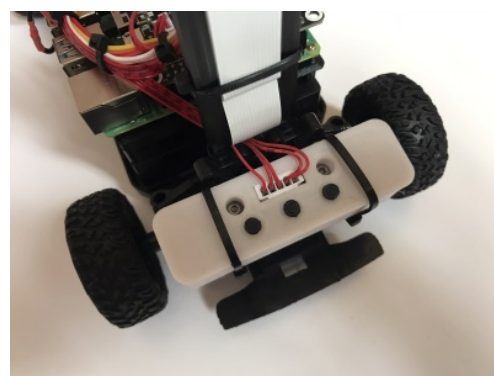
(a) Raspberry Pi Camera 3



(b) HC-SR04



(c) Snímač otáček



(d) Trojice tlačítek

Obr. 6.6: Robot osazený snímači

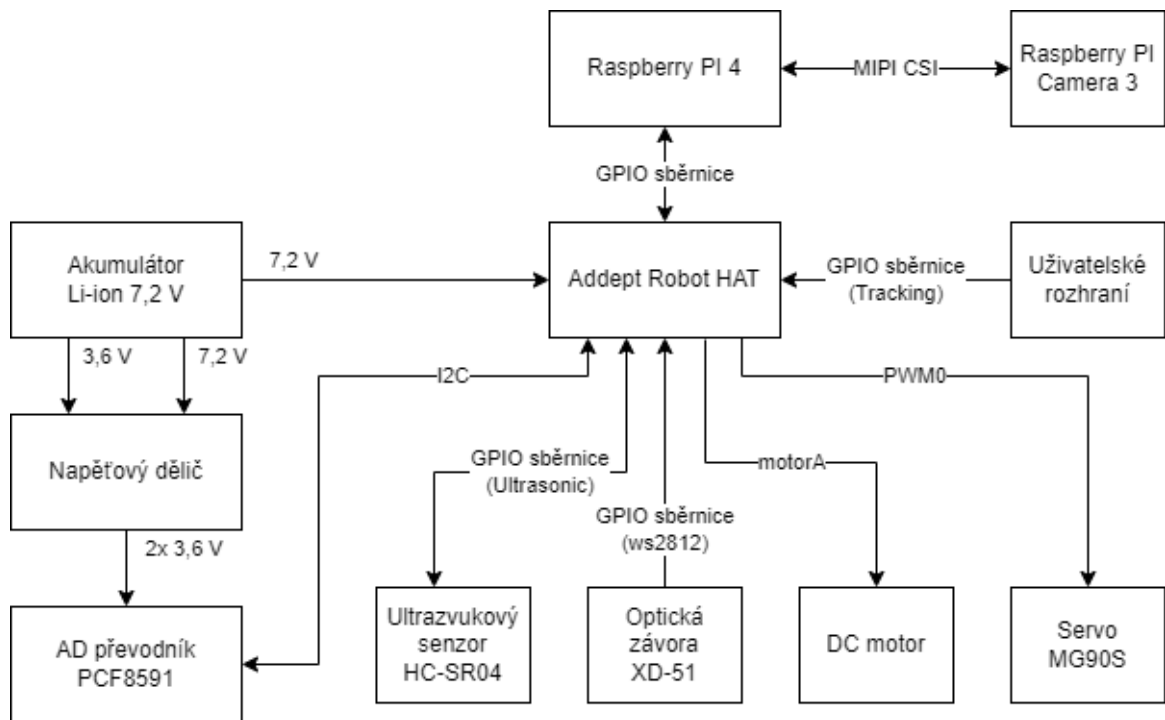
jsou připojeny prakticky všechny ostatní periferie. Rozšíření je detailněji popsáno v kapitole 6.1.2.

Rozšiřující HAT je napájen z akumulátoru napětím U_{bat} . Napětí U_1 , U_2 z jednotlivých článků je připojeno k napěťovému děliči, jehož výstupem jsou dvě standardizovaná napětí U_{AD1} a U_{AD2} .

Jelikož jsou některé periferie připojeny k portům, které byly původně určeny pro jiné účely, popisky konektorů nesou jiné označení. Proto jsou na obrázku uvedeny názvy jednotlivých portů. Tyto porty obvykle nabízí vodiče pro napájení a vodiče připojené k GPIO sběrnici.

6.2 Software

Po dokončení hardwarové části robota je nutné navrhnout vhodný software, který bude sloužit k ovládání všech připojených periférií tak, aby bylo možné na základě zpětné vazby ze snímačů vytvořit algoritmy, jež budou schopny autonomně řídit vozidlo. Jelikož je tato pomůcka vytvářena primárně pro účely výuky na střední



Obr. 6.7: Blokový diagram zapojení

škole, kde se žáci učí programovat v programovacím jazyku Python, budou obslužné knihovny i hlavní algoritmus také vytvořen v tomto jazyce. Kód bude rozdělen do několika částí, aby byla zachována maximální přehlednost kódu. V následujících podkapitolách budou tyto části jednotlivě popsány.

6.2.1 Ovládání periférií

Vydavatelé rozšiřující desky Adept Robot HAT dodávají pro obsluhu periférií, kterými toto rozšíření disponuje, i software. Programy je možné stáhnout na internetových stránkách výrobce. Bohužel, dodané soubory slouží spíše jako demonstrace kódu, jelikož by se v rozsáhlejšímu programu nedaly použít. Několik portů na rozšíření je také využito k jinému účelu, než výrobce zamýšlel, proto bylo vytvořeno několik knihoven pro obsluhu všech periférií včetně kamery.

Řízení motorů

Pro ovládání H-můstku, který je určen pro řízení DC motorů, byla v souboru `MotorDriverUtilities.py` vytvořena třída s názvem `Motor`. Funkce H-můstku je závislá celkově na šesti signálech, přičemž tři signály ovládají motor A a tři signály ovládají motor B. V následující tabulce jsou uvedeny příklady funkcí H-můstku v reakci na danou kombinaci řídicích signálů pro motor A:

Enable A	Input 1	Input 2	Funkce
0	-	-	Motor je vypnutý
1	0	0	Motor je zastaven
1	1	0	Motor se otáčí proti směru hodinových ručiček
1	0	1	Motor se otáčí po směru hodinových ručiček
1	1	1	Motor je brzděn

Tab. 6.1: Ovládaní H-můstku

Pinem s označením *Enable A* nebo *Enable B* lze řídit otáčky motoru za pomoci PWM signálu. Z tohoto důvodu jsou v konstruktoru této funkce nastavovány nejen piny, kterými počítač komunikuje s H-můstkem, ale i frekvence PWM signálu, jež je ve výchozím stavu nastavena na *100 Hz*. Třída obsahuje následující funkce:

- `changeFrequency` - zavoláním této funkce lze změnit frekvence PWM signálu.
- `stop` - touto funkcí jsou zastaveny oba motory a je zastaveno generování PWM signálu.
- `set` - tato funkce slouží pro ovládaní motorů.
- `__speedControl` - tato privátní funkce slouží k přepočtu procent na hodnotu střídy PWM signálu.

Řízení serva

Pro řízení serv byla vytvořena třída s názvem `Servo`, která se nachází v souboru `ServoDriverUtilities.py`. Jelikož se o řízení serv stará integrovaný obvod s označením PCA9685, jenž komunikuje s Raspberry Pi pomocí směrnice I2C, je v této třídě využívána knihovna s označením `Adafruit_PCA9685`. Tato třída obsahuje kromě konstruktoru, ve kterém je inicializována zmíněná knihovna a počáteční nastavení proměnných, také dvě funkce:

- `__setValue` - tato privátní funkce slouží k nastavení požadované střídy na daném kanálu.
- `set` - touto funkcí lze nastavit úhel natočení serva v určeném rozsahu.

Interakce s uživatelem

Soubor s názvem `HidUtilities.py` obsahuje dvě třídy. První třída `Buttons` zajišťuje ovládaní robotu pomocí tlačítek. V konstruktoru dochází k inicializaci proměnných, nastavení pinů Raspberry Pi na vstupy a nastavení přerušení. V této třídě se dále nachází dvě funkce:

- `__setFlag` - tato funkce je volána přerušením, které nastane, když bude na některém signálu tlačítek detekována náběžná hrana

- `getValue` - pokud je zavolána tato funkce, vrátí hodnotu 1 v případě, že bylo stisknuto příslušné tlačítko. V opačném případě bude hodnota rovna 0

Další třída v této knihovně nese název `Leds`. Tato část programu byla vytvořena pro obsluhu LED (Light Emitting Diode), které se nachází na rozšiřující desce. V konstruktoru této třídy dochází k inicializaci pinů na počítači. Pro samotné ovládání diod byly vytvořeny dvě funkce:

- `setOn` - zapnutí příslušné LED.
- `setOff` - vypnutí příslušné LED.

Obsluha senzorického vybavení

Soubor s názvem `SensorUtilities.py` obsahuje knihovnu pro obsluhu senzorického vybavení robotu. Tento soubor obsahuje tři třídy pro jednotlivé senzory.

První třída s názvem `UltrasonicSensor` slouží k obsluze ultrazvukového snímače. V konstruktoru této třídy jsou inicializovány proměnné, nastaveny piny Raspberry Pi tak, aby plnily požadovanou funkci a jsou zde nastaveny přerušení. Tato třída obsahuje tři funkce:

- `__getT` - po zavolání této funkce je vyslán signál *trig* o délce $20 \mu s$ a je uložen aktuální čas.
- `__getDeltaT` - tato funkce je volána přerušením, které způsobí odezva na signál *echo*. Funkce uloží do proměnné čas mezi signály *trig* a *echo*.
- `distance` - tato funkce vrací změřenou vzdálenost v *cm*.

Další třída slouží pro obsluhu snímače otáček hlavní hřídele, kterou tvoří optický optočlen fungující jako optická závora. Konstruktor této funkce opět slouží k inicializaci proměnných, pinů i přerušení. Součástí této třídy jsou další čtyři funkce:

- `__counter` - tato funkce je volána přerušením, které reaguje na náběžnou hranu signálu z optočlenu. Při zavolání této funkce se k vytvořenému čítači pulzů přičte hodnota 1.
- `getPulseCounter` - tato funkce vrací aktuální hodnotu čítače.
- `clearPulseCounter` - funkce pro nulování čítače.
- `distance` - tato funkce vrací vzdálenost, kterou robot ujel.

Poslední třídou, která se nachází v tomto souboru, je `ADC`. Tato třída slouží k obsluze AD převodníku. Jelikož tento modul komunikuje s Raspberry přes I2C sběrnici, v konstruktoru jsou nastaveny všechny potřebné parametry potřebné k navázání spojení. Obsahem této funkce jsou tři funkce:

- `getAdcValue` - touto funkcí lze vyčíst data ze zadaného vstupu AD převodníku.
- `batteryState` - tato funkce vrací hodnotu napětí ve V, jež bylo změřeno na jednotlivých člancích akumulátoru.

- `batteryWarning` - pokud klesne napětí na jednom ze článků pod zadanou mez, tato funkce vrátí hodnotu 1, v ostatních případech 0.

Obsluha kamery

Pro obsluhu kamery byl vytvořen soubor s názvem `CameraUtilities.py`. Ten soubor obsahuje třídu s názvem `Camera`. Kód pro ovládání kamery využívá knihovny, které se nazývají `picamera2`, `libcamera`, `cv2`. Konstruktor nastaví podobu výstupu senzoru. Při inicializaci kamery je možno nastavit její rozlišení. U kamery je využito funkce automatického ostření. Kromě konstruktoru obsahuje třída další dvě funkce:

- `getImage` - tato funkce vrací snímání obraz.
- `stop` - touto funkcí lze zastavit činnost kamery.

Uživatelské funkce

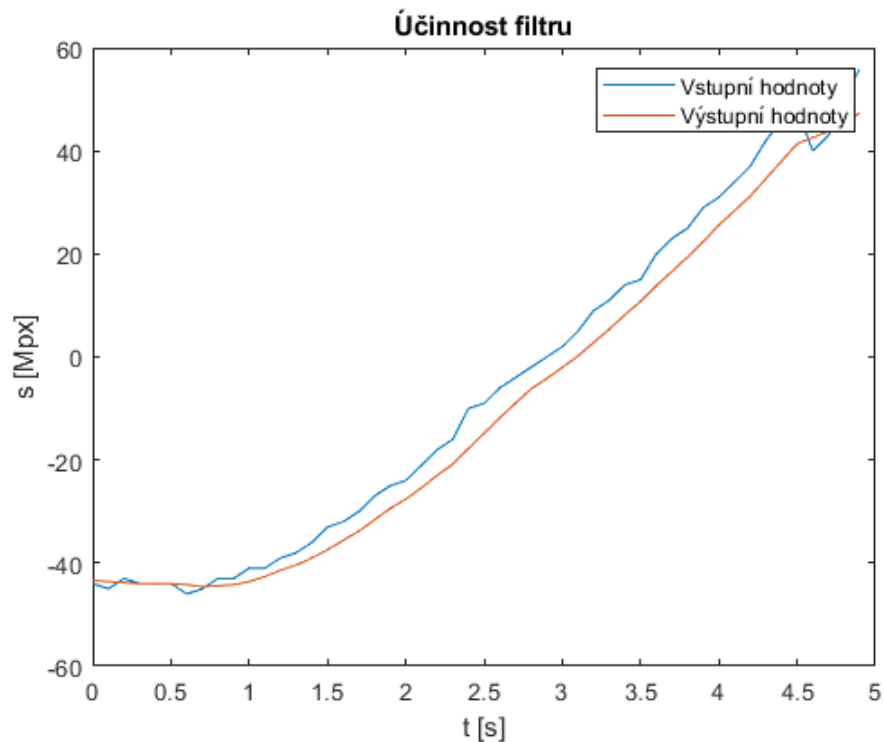
V souboru `UserUtilities.py` se nachází funkce, které nekorespondují s ostatními podprogramy či třídami. Lze zde najít funkce:

- `filter` - tato funkce průměruje pole hodnot.
- `aTan` - funkce, která slouží pro výpočet úhlu, jež svírá přepona s přilehlou odvěsnou.
- `exchange` - po zavolání této funkce dojde k prohození hodnot atributů.
- `average` - touto funkcí lze vypočítat aritmetický průměr dvou hodnot.

Za zmínku stojí funkce `filter`, která slouží k průměrování hodnot přijatých ze snímačů, aby byl omezen šum. Ve funkci je vytvořena cyklická fronta, ve které je přepisována nejstarší hodnota novou. Po začlenění nového prvku do fronty je celé pole sečteno a vyděleno počtem prvků v poli, čímž vznikne aritmetický průměr hodnot, které se ve frontě nachází. Tento výsledek je výstupem této funkce. Na obrázku 6.8 jsou znázorněny vstupní i výstupní hodnoty filtru. Tyto hodnoty představují míru translace robotu od čáry. Jelikož tento filtr způsobuje dopravní zpoždění, kruhovou frontu tvoří pouze pět hodnot. Při použití většího pole hodnot by bylo dosaženo přesnějších výsledků za cenu většího dopravního zpoždění.

6.2.2 Detekce čar

Hlavním úkolem této práce je vytvořit model vozidla, který bude schopen pomocí kamery identifikovat čáru (jízdní pruh) a následovat ji, proto je spolehlivá detekce čar v tomto úkolu velmi podstatnou částí. V této podkapitole budou popsány vytvořené algoritmy a principy zpracování obrazu, které tuto požadovanou funkci umožní.



Obr. 6.8: Účinnost filtru

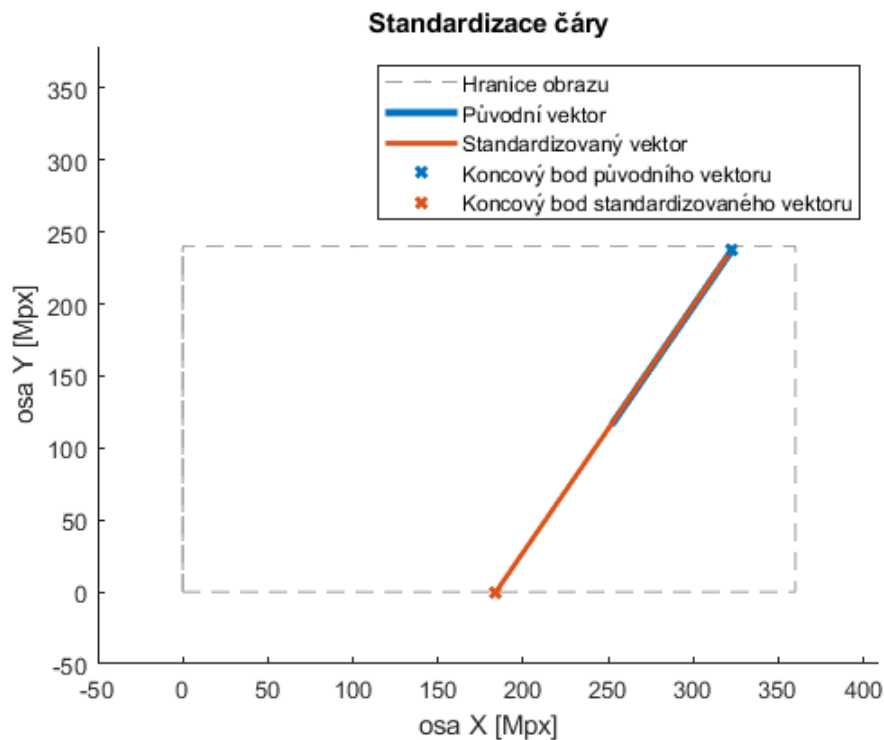
Popis podpůrné knihovny

Aby bylo možné vytvořit algoritmus pro sledování čar, bylo nutné naprogramovat několik podpůrných funkcí, které se nachází v souboru `LineDetectionUtilities.py`. V tomto souboru jsou využívány tři knihovny s názvy `cv2`, `numpy`, `math`, které jsou využity v následujících funkcích:

- `inicializeTrackbarsHSV` - inicializace posuvníků pro ladění odstínu, sytosti barev a jasu.
- `inicializeTrackbarsWarp` - inicializace posuvníků pro ladění geometrické transformace obrazu.
- `valTrackbarsWarp` - funkce vrací body pro geometrickou transformaci, které byly nastaveny pomocí posuvníků.
- `valTrackbarsHSV` - funkce vrací hodnoty odstínu, sytosti barev a jasu nastavené pomocí posuvníků.
- `drawPoints` - funkce pro vykreslení nastavených bodů pro geometrickou transformaci obrazu.
- `warpImg` - tato funkce provede geometrickou transformaci obrazu podle nastavených kritérií.
- `thresholding` - touto funkcí lze provést prahování obrazu.
- `edgeDetSobel` - hranová detekce pomocí Sobelova jádra.

- `edgeDetCanny` - zavoláním funkce lze detekovat hrany Cannyho metodou.
- `rgbToGray` - převod RGB obrátek na obraz ve stupních šedi.
- `standardiseLine` - tato funkce slouží k dopočítání vektoru přímky tak, aby se koncové body dané přímky vždy nacházely na hranici obrazu.

Funkce `standardiseLine` je velice podstatnou součástí řešení tohoto úkolu. Knihovna `opencv` nabízí detekci čar pomocí funkce s názvem `HoughLinesP`. Tato funkce vrací vektory jednotlivých čar, jež se nachází v obraze. Vektory ale mají velmi rozdílné vlastnosti, proto byla vytvořena funkce, která tyto vektory do určité míry standardizuje. Po použití této funkce se počátky i konce vektorů vždy nacházejí na hranicích obrazu. Toho lze využít pro vytvoření vhodné vstupní veličiny pro regulátor. Na obrázku č. 6.9 je znázorněn původní vektor čáry a vektor po použití této funkce.



Obr. 6.9: Standardizace čáry

Detekce jedné čáry

Funkce pro detekci čar jsou obsaženy v souboru s názvem `LineDetection.py`. Pro detekci jedné čáry byla vytvořena funkce `getLineAngleTranslation`. Vstupem této funkce je obraz z kamery. Výstup tvoří tři proměnné:

- `getLine` - pokud je čára detekována, hodnota proměnné je 1, jinak 0.

- **angle** - hodnota této proměnné obsahuje informaci o úhlu, který aktuálně svírá čára s rovinou kamery.
- **translation** - hodnota udává míru odchýlení čáry od středu obrazu.

Jelikož by funkce `HoughLinesP` nebyla schopna detekovat linie přímo z pořízeného snímku, dochází v této funkci k předzpracování obrazu následujícím způsobem.

Vstupní obraz je rozmazán funkcí `blur`, která průměruje hodnoty pomocí masky o velikosti 5×5 . Rozmazáním obrazu je redukován šum, jenž se ve snímku nachází. Na rozmazaný obraz je aplikována geometrická transformace pro úpravu perspektivy. Takto transformovaný obraz je upraven pomocí metody prahování, aby došlo k segmentaci linií od pozadí. V posledním kroku jsou detekovány hrany využitím Cannyho detektoru.

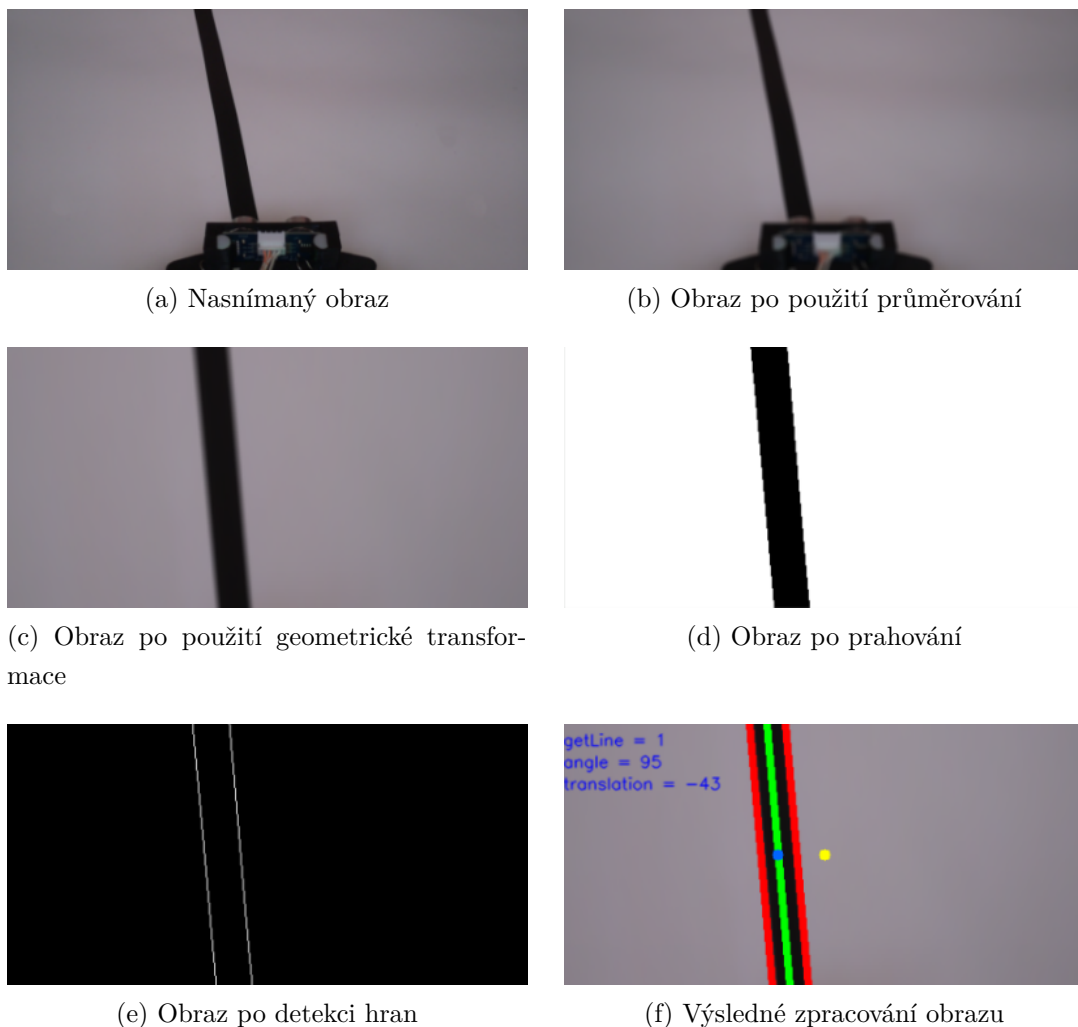
Po předzpracování je na výsledný obraz aplikována funkce `HoughLinesP`, která vrací pole vektorů detekovaných čar. Jelikož byla provedena detekce hran, každá fyzická čára obsahuje dvě hrany, což znamená, že zmíněná funkce má vrátit minimálně dva vektory. Pokud tomu tak není, nastaví se proměnná `getLine` na hodnotu 0. Jestliže jsou nalezeny minimálně dvě linie, jsou standardizovány pomocí funkce `standardiseLine` a následně rozděleny na pravou a levou. Pokud je rozdíl souřadnic X levé a pravé linie příliš velký, znamená to, že byl detekován okraj testovacího plátna místo čáry, proto je proměnná `getLine` nastavena na 0. Pokud se nevyskytne žádný z popsaných problémů, jsou počáteční a koncové body vektorů zprůměrovány, čímž vznikne linie, která reprezentuje střed čáry. Od této linie je vypočítán úhel, jenž tato středová linie svírá s osou X , a je určena míra odchýlení středu čáry od středu obrazu, čímž jsou vypočítány hodnoty, které jsou určeny pro zpracování regulátorem.

Na obrázku číslo 6.11g je uveden zpracovaný obraz. Červené přímky značí okraje vodící linie. Zelenou barvou je označen střed čáry. Modrý bod symbolizuje hodnotu souřadnice X v polovině výšky obrazu. Střed obrazu je označen žlutou barvou. Pro výpočet odchylky robotu od středu se využívá odečtení souřadnice X středového bodu od souřadnice X modrého bodu.

Detekce dvou čar

Funkce pro detekci dvou čar se nachází v třídě s názvem `LineDetectionUtilities.py`. Tato funkce byla označena jako `getLinesAngleTranslation`.

Úvodní část této funkce je velmi podobná odpovídající pasáží kódu ve funkci `getLineAngleTranslation`, poněvadž je potřeba nasnímaný obraz podrobit předzpracování. Prvním krokem předzpracování je rozmazání snímku, aby došlo k potlačení šumu. Poté je obraz geometricky transformován. Po tomto úkonu dochází k segmentaci vodících linií pomocí funkce `thresholding`. V takto upraveném obrazu jsou



Obr. 6.10: Zpracování obrazu

detekovány hrany pomocí Cannyho detektoru. Čáry jsou nalezeny funkcí z knihovny `openCV`, jež nese název `HoughLinesP`, která využívá tzv. Houghovu transformaci.

Původním řešením této funkce bylo nalezení okrajů obou čar a standardizace detekovaných čar pomocí vytvořené funkce `standardiseLine`. Poté byly nalezeny středy obou čar, pomocí kterých byla určena střední linie mezi dvěma čarami. Pomocí této linie měl být určen úhel, který svírají vodící linie s rovinou zadní nápravy, i translace středu jízdního pruhu od virtuálního bodu, jenž byl umístěn do středu obrazu. Ukázalo se ale, že tento postup není vhodný, jelikož při velmi prudkých zatáčkách docházelo ke ztrátě jedné z vodících čar. Algoritmus tak nebyl funkční a musel být použit jiný přístup.

Funkční řešení je realizováno následujícím způsobem. V prvním kroku je provedena detekce okrajové hrany jedné z vodících linií. Poté je nalezena vnitřní hrana této vodící čáry. Po nalezení okrajů vodící čáry je určen střed této linie, ke kterému

je přičtena, nebo odečtena konstanta, jež odpovídá polovině vzdálenosti mezi dvěma vodicími čarami. Vzdálenost vodicích čar u testovací dráhy je přibližně 20 cm , což odpovídá cca 300 px . Přičtením konstanty je docíleno toho, že se v ideálním případě nachází střed jízdního pruhu ve středu obrazu. Od vytvořené středové linie je opět počítán úhel, který svírá s rovinou zadní nápravy, a míra posunutí od virtuálního bodu ve středu obrazovky. Při volání této funkce lze zvolit, jaká z vodicích čar má být detekována, což může být použito v nějakém stavovém chování při samotné jízdě.

V sérii obrázků č. 6.11 je viditelný postup zpracovávání snímku a výsledky těchto úkonů. Oranžovými čarami je označena vždy levá hrana čáry, červenou pak pravá hrana vodicí linie. Zelenou barvou je označena výsledná přímka, u které je její střed vyznačen modrým bodem. Od toho bodu se počítá míra translace od středu obrazu. Střed obrazu je značen žlutým bodem.

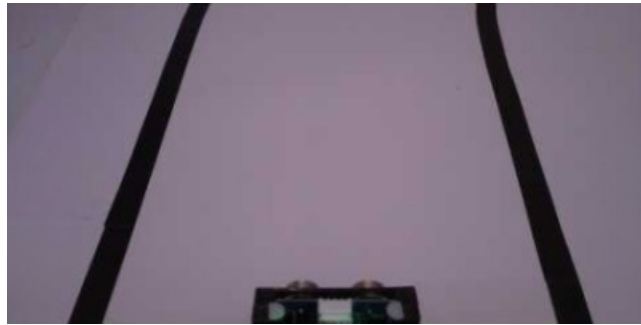
6.2.3 Regulátor

Soustava, která je reprezentována robotem, obsahuje mnoho vstupů i výstupů, a řadí se tak mezi tzv. MIMO systémy. U Ackermannova podvozku je zapotřebí regulovat rychlost pohybu vozidla i úhel natočení kol, což lze považovat za výstupní veličiny systému. Vstupní data jsou výstupy z jednotlivých snímačů, především se jedná o úhel, který svírá rovina snímače s vodicí linií a odchýlení středu vozidla od středu čáry. Jedná se tak o velice komplexní úlohu.

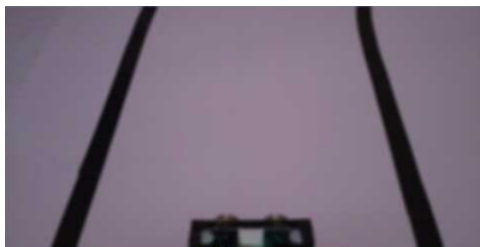
Jelikož má být model určen pro žáky střední školy, bylo potřeba regulaci systému v maximální možné míře ulehčit. V ideálním případě by měla soustava po zjednodušení tvořit systém s označením SISO. Rychlost jízdy vozidla byla nastavena na konstantní hodnotu, tímto krokem zanikla potřeba regulovat dvě výstupní veličiny. Omezení vstupních veličin bylo docíleno následujícím způsobem. Před vozidlem byl stanoven virtuální bod, ke kterému je měřena míra jeho odchýlení od vodicí linie. Tímto úkonem jsou v určité míře sloučeny dvě vstupní hodnoty. První vstupní hodnota je představována úhlem, který svírá rovina kamerového snímače s čarou. Druhá vstupní veličina je míra translace vozidla vůči čáře.

ON-OFF regulátor

Pro otestování funkce celého systému byl vytvořen základní třístavový ON-OFF regulátor. Způsob regulace je znázorněn na obr. č. 6.12. Proměnná *Translace* reprezentuje míru odchýlení virtuálního bodu od středu obrazu. Pomocí φ je značen úhel natočení kol, který je vztažen k zadní nápravě robotu. Úhly 63° , 117° odpovídají maximální možné míře natočení kol.



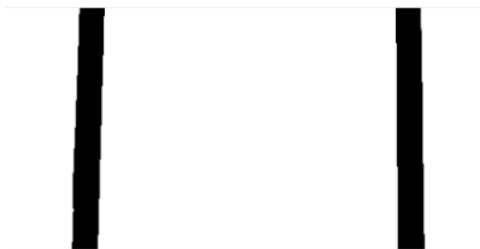
(a) Nasnímaný obraz



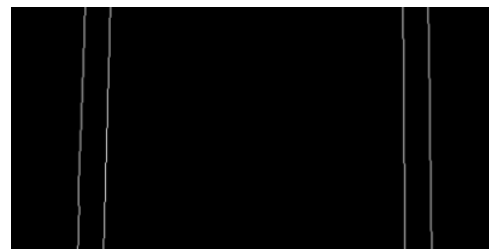
(b) Obraz po použití průměrování



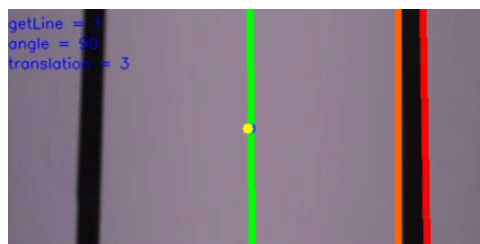
(c) Obraz po použití geometrické transformace



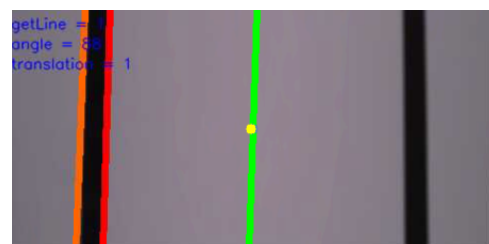
(d) Obraz po prahování



(e) Obraz po detekci hran



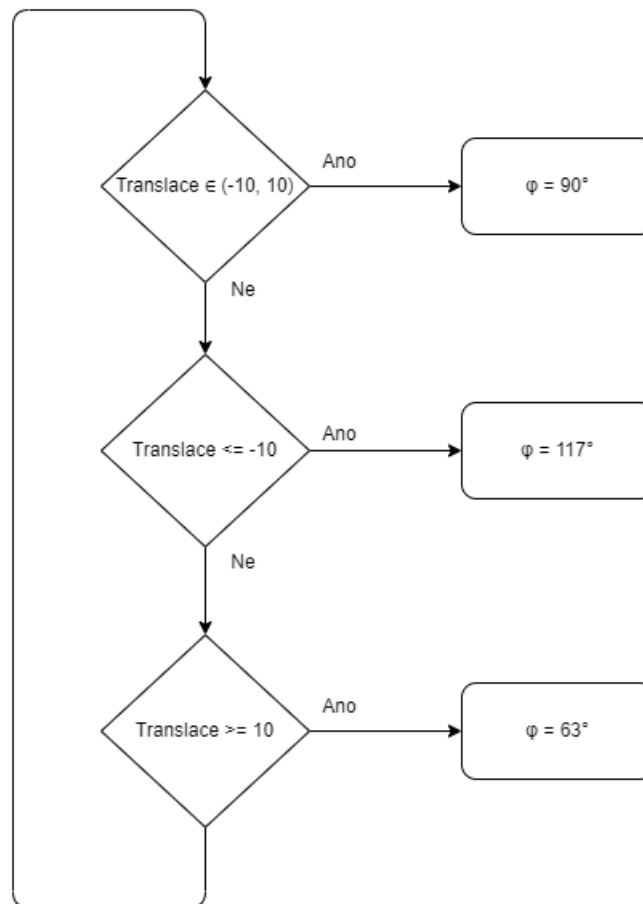
(f) Výsledné zpracování obrazu při využití pravé vodící čáry



(g) Výsledné zpracování obrazu při využití levé vodící čáry

Obr. 6.11: Zpracování obrazu

Při použití tohoto regulátoru byl robot schopen následovat vodící linii. Pohyb ale nebyl plynulý a často docházelo ke ztrátě čáry. Z tohoto důvodu bylo nutné navrhnout jiný typ regulátoru.



Obr. 6.12: Vývojový diagram ON-OFF regulátoru

PSD regulátor

Pro plynulejší regulaci úhlu natočení byl vybrán lineární regulátor s označením PSD, především kvůli nenáročné implementaci. Soustava obsahuje mnoho nelinearit, které tomuto typu regulátoru příliš neprospívají, nicméně pro účely této práce by výsledná regulace měla být dostatečná.

Algoritmus regulátoru je v práci implementován podobným způsobem, jako v následujícím pseudokódu:

Výpis 6.1: Implementace PSD regulátoru

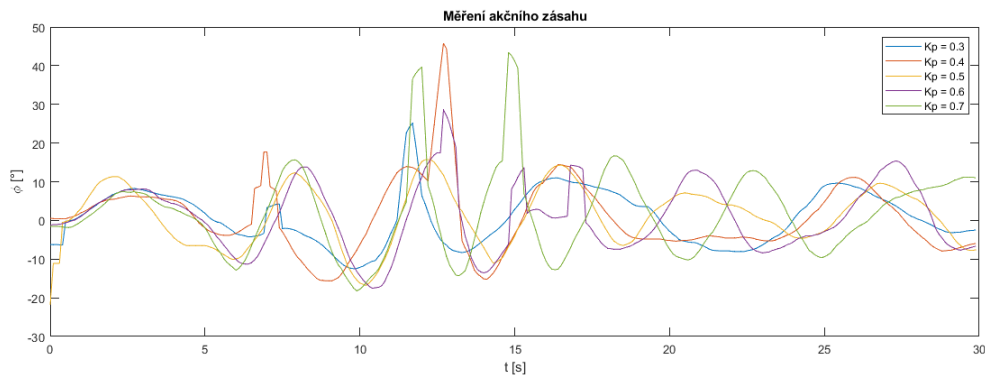
```
1 while True:
2     error = offset - translation
3     derivate = error - lastError
4
5     reg = (Kp * error) + (Ki * integral) + (Kd * derivate)
6     result = 90 - reg
7
8     servo.set(0, result)
9
10    integral += error
11    lastError = error
```

- K_p , K_i , K_d - konstanty jednotlivých složek.
- `error` - regulační odchylka.
- `lastError` - pomocná složka pro integraci regulační odchylky.
- `derivate` - derivace regulační odchylky.
- `integral` - integrace regulační odchylky.
- `reg` - akční zásah.
- `result` - přepočtená hodnota akčního zásahu.

Ackermannův podvozek svým charakterem obsahuje astatismus, použití integrační složky tedy není nezbytné. Z tohoto důvodu byl v této práci implementován pouze PD regulátor. Hodnoty konstant byly zjištěny metodou Ziegler-Nichols, jelikož v případě této metody není nutné identifikovat systém. Pro výpočet odpovídajících konstant je nutné zjistit pouze hodnoty kritického zesílení a kritické periody. Hodnoty konstant jednotlivých složek PD regulátoru byly zjištěny následujícím způsobem.

V první fázi návrhu byl zaznamenáván akční zásah regulátoru frekvencí 10 Hz při jízdě robotu po rovné čáře. Celkem proběhlo pět měření s různým nastavením proporcionální složky, kterou reprezentuje konstanta K_p . Inicializační hodnoty byly zjištěny experimentálně.

Na obrázku č. 6.13 jsou zobrazeny naměřené hodnoty. Cílem tohoto měření bylo zjištění kritického zesílení systému a kritické periody kmitů. Hledaná regulační odezva by měla mít charakter stabilního oscilátoru, amplituda kmitů by tedy měla být konstantní. Tomuto popisu se nejvíce blíží hodnota $0,5$. Tuto cifru lze považovat za



Obr. 6.13: Měření akčního zásahu

hodnotu kritického zesílení. Z grafu byla odečtena kritická perioda kmitů, která činí $4,3$ s. Poté byly podle následujících vzorců vypočítány jednotlivé konstanty:

$$Kp = 0,4 \cdot Kc = 0,4 \cdot 0,5 = 0,2 \quad (6.4)$$

Kp ...konstanta proporcionální složky regulátoru

Kc ...hodnota kritického zesílení

$$Kd = 0,125 \cdot Tc = 0,125 \cdot 4,3 = 0,54 \quad (6.5)$$

Kd ...konstanta derivační složky regulátoru

Tc ...hodnota kritické periody

6.3 Aplikace

V této práci mají být vytvořeny příklady, ve kterých budou využity všechny periférie a dříve zmíněné části kódu. Tyto příklady mají sloužit demonstraci algoritmů pro autonomní pohyb vozidla a nastínit funkci asistence hlídání jízdních pruhů opravdového vozidla.

6.3.1 Hlavní část programu

V předešlém textu této práce jsou uvedeny jednotlivé části programu, které je nutné vhodným způsobem zkombinovat tak, aby robot plnil požadovanou funkci. K tomuto účelu byly v programové části s názvem `main` vytvořeny tři úlohy, které jsou volány periodicky.

První úloha má za úkol kontrolovat stav akumulátoru, který vizualizuje pomocí tří LED na Adept Robot HAT. Pokud všechny LED svítí, akumulátor je plně nabitý.

Jestliže nesvítí žádná z diod, akumulátor je vybitý. Jelikož není potřeba kontrolovat stav akumulátoru příliš často, je tato úloha volána s periodou 10 s .

V další části programu jsou vyčítány hodnoty z funkcí pro detekci čáry a ultrazvukového snímače. Jelikož je na hodnoty translace virtuálního bodu od vodící linie a vzdálenosti robotu od okolních objektů použita funkce filtru s názvem `filter`, vykonává se tato úloha s periodou 1 ms .

Poslední úloha obsahuje stavový automat, který řídí celý rozhodovací algoritmus. Tato část programu je volána periodicky po 10 ms . Na obrázku č. 6.14 je uveden vývojový diagram. Tento automat je téměř totožný pro obě úlohy (jízda po čáře i mezi čarami). Algoritmy se liší pouze voláním jiných funkcí pro rozpoznávání čar. Programy obsahují celkově čtyři stavy:

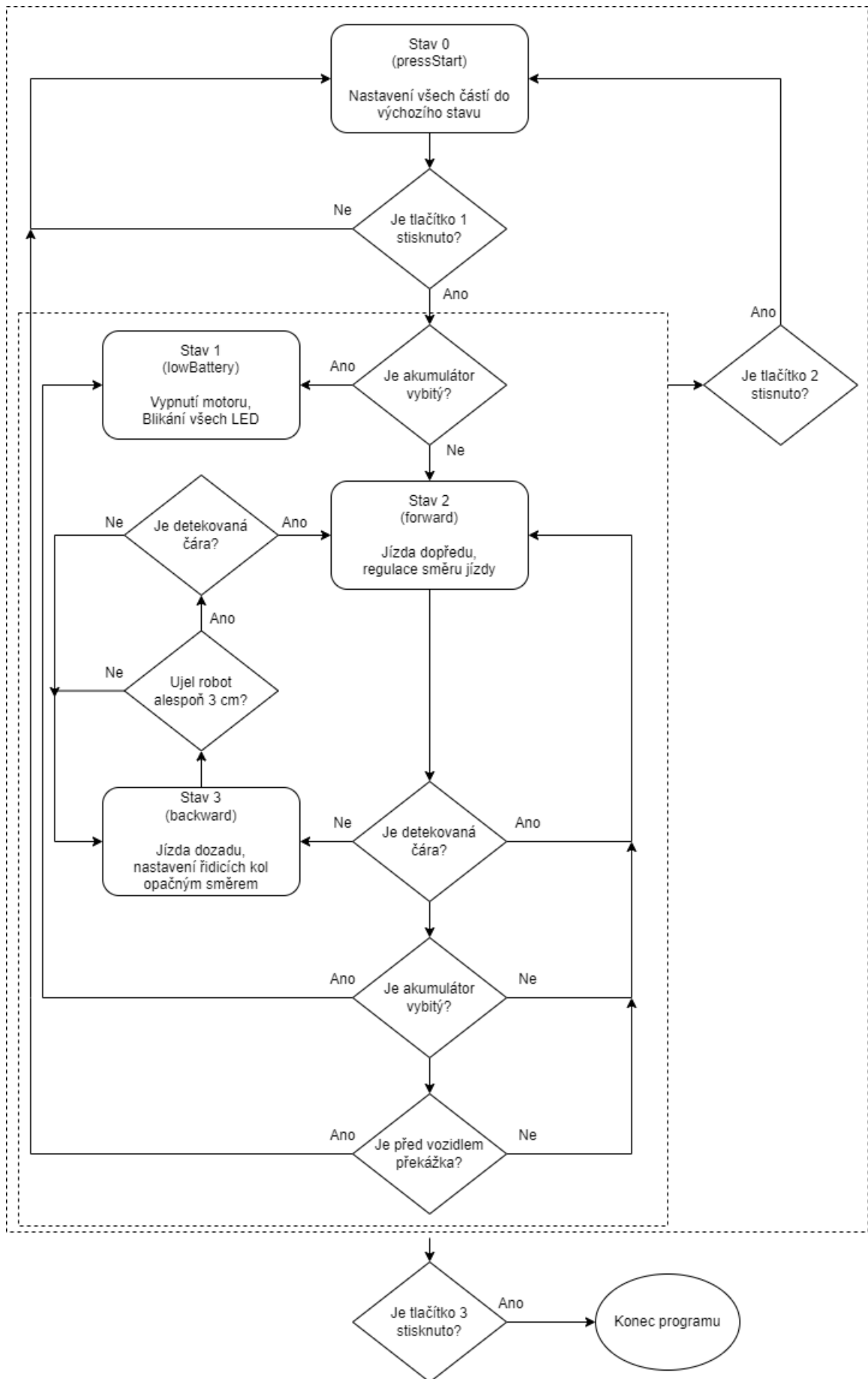
- Stav 0 - tento stav slouží primárně pro nastavení všech periférií a proměnných do výchozího stavu. Při spuštění programu se stavový automat nachází v tomto stavu.
- Stav 1 - tento stav nastane v případě, že je baterie vybitá. V tomto stavu robot ukončí veškerý pohyb a upozorní na stav baterií blikáním všech tří LED.
- Stav 2 - v tomto stavu se robot pohybuje směrem dopředu. Natáčení kol je regulováno pomocí vytvořeného PSD regulátoru.
- Stav 3 - pokud vodící linie není detekována, robot natočí kola opačným směrem, než ve kterém byly natočeny v předešlém stavu, a pohybuje se směrem dozadu, dokud neujede minimálně vzdálenost 3 cm .

Aby bylo možné robota ovládat bez nutnosti použití vzdálené plochy či připojení periférií přímo k Raspberry Pi, byly pro ovládání využity tlačítka vytvořeného HID modulu. Jednotlivá tlačítka v programu fungují následujícím způsobem:

- Tlačítko 1 - pomocí toho tlačítka lze spustit jízdu robotu po čáře, nebo mezi čarami.
- Tlačítko 2 - po stisku tohoto tlačítka se program z jakéhokoliv jiného stavu přepne do inicializačního stavu 0.
- Tlačítko 3 - stiskem tohoto tlačítka je program ukončen.

6.3.2 Testování funkce

Cílem testování funkce robotu byla především optimalizace všech konstant tak, aby byl robot schopen jízdy po vytyčené dráze s minimálním odchylením se od čáry. Robot měl být naprogramován tak, aby plnil dvě hlavní úlohy. První z nich je jízda po čáře. Druhým úkolem je jízda mezi dvěma čarami. Aby bylo možné funkčnost robota otestovat, bylo potřeba vytvořit vhodné testovací dráhy.

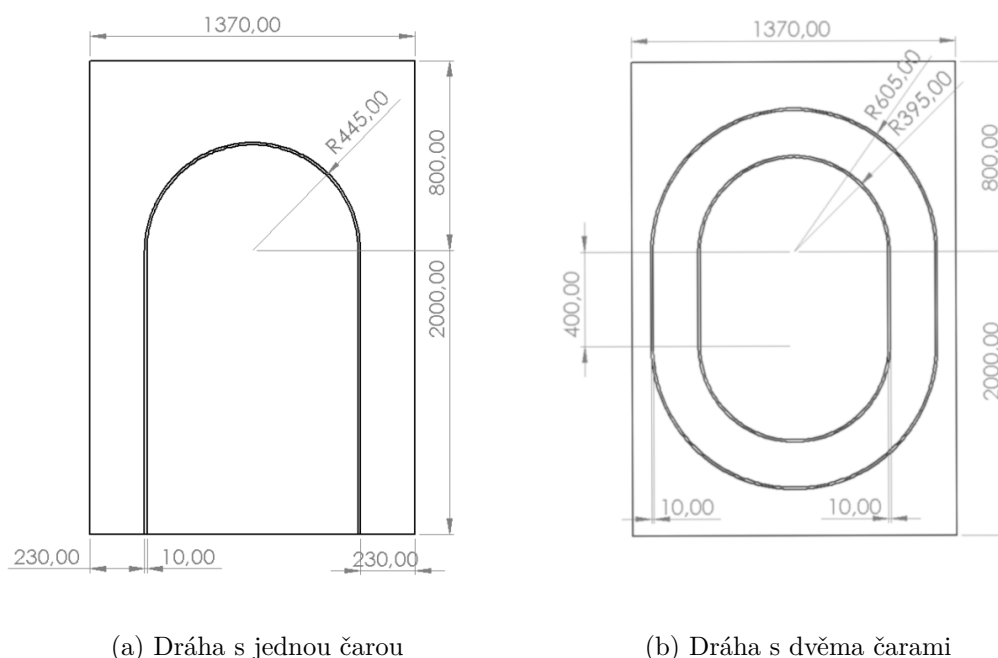


Obr. 6.14: Vývojový diagram
67

Testovací dráhy

Robot má simulovat jízdu po reálné vozovce. Tu obvykle tvoří tmavý podklad. Jelikož ale nebyl žádný vhodný podklad nalezen, byly testovací dráhy vytvořeny na bílém papíru o rozměrech 200×137 cm. Záměna barvy podkladu nehraje v této aplikaci velkou roli, jelikož jsou jednotlivé linie rozpoznávány pomocí hran, které jsou detekovány v obou případech. Vodicí čáry byly vytvořeny pomocí černé izolační lepicí pásky o šířce 10 mm.

Minimální poloměr otáčení modelu při maximálním natočení předních kol je přibližně 40 cm, proto se na vytvořených drahách vyskytují pouze zatáčky, jejichž poloměr je minimálně 45 cm. Na obrázcích 6.15 jsou uvedeny podoby dvou testovacích drah.

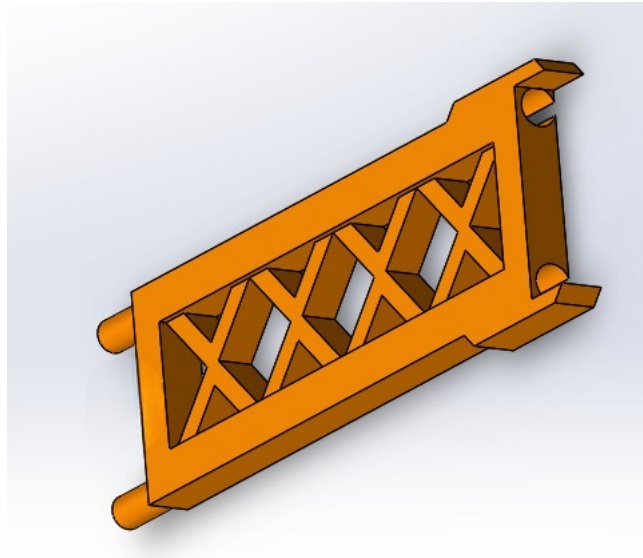


Obr. 6.15: Testovací dráhy

Testování jízdy mezi dvěma čarami

V úvodní fázi byla testována jízda robotu po jedné čáře spolu s třístavovým ON-OFF regulátorem, který je blíže popsán v kapitole 6.2.3. Tato konfigurace nefungovala spolehlivě, nicméně robot dokázal následovat vytyčenou dráhu. Při testování jízdy mezi čarami ale robot nebyl schopen jízdy, neboť zorný úhel použité kamery nebyl dostatečný. Jelikož ohniskovou vzdálenost u tohoto kamerového modulu není možné změnit, jediné řešení bylo umístění kamery do větší vzdálenosti od snímaného

povrchu. K tomuto účelu byl v programu SolidWorks vymodelován rozšiřující díl, který umožňuje snadné prodloužení držáku kamery o $88,4 \text{ mm}$. Díl je uveden na obrázku č. 6.16. Pro zvětšení vzdálenosti kamerového snímače od jízdního povrchu byla celá konstrukce posunuta až nad zadní nápravu.



Obr. 6.16: Prodloužení držáku kamery

V této konfiguraci byl robot schopen držet se vytyčené trasy. Laděním konstant regulátoru se ale nepodařilo docílit toho, aby se robot pohyboval pouze ve vytyčeném jízdním pruhu, neboť na testovací dráze v zatáčce o poloměru 45 cm docházelo ke ztrátě vodicích linií. Pokud by byla zvětšena vzdálenost mezi kamerovým snímačem a plochou, pravděpodobně by k tomuto problému nedocházelo. Zvětšením zorného úhlu by ale způsobovalo snímání okraje testovací dráhy, což bylo nežádoucí.

Při testování bylo provedeno pět měření, které mělo za účel zjistit minimální dobu potřebnou k projetí vytyčené trasy. V tabulce č. 6.2, jsou uvedeny výsledky jednotlivých jízd. Nejlepší čas, který robot zaznamenal, činil $37,63 \text{ s}$.

Číslo jízdy [-]	Čas [s]
1	38,75
2	38,65
3	41,13
4	37,63
5	40,28

Tab. 6.2: Testování jízdy v jízdním pruhu

Testování jízdy po jedné čáře

Při testování jízdy po jedné čáře nenastaly žádné komplikace, jelikož v tomto případě není nutné snímat tak rozsáhlou plochu podložky. Ke ztrátě vodící linie docházelo zřídka. Navíc je okraj testovací plochy dostatečně vzdálen od čáry, čímž nedocházelo k záměně okraje plochy za testovací čáru.

Měření doby, kterou robot potřebuje na projetí dráhy, proběhlo i v tomto případě pětkrát. Výsledky jednotlivých jízd jsou uvedeny v tabulce č. 6.3. Robot projel nejrychleji trasu za *25,98 s*.

Číslo jízdy [-]	Čas [s]
1	27,40
2	27,32
3	25,98
4	26,02
5	27,61

Tab. 6.3: Testování jízdy po čáře

Závěr

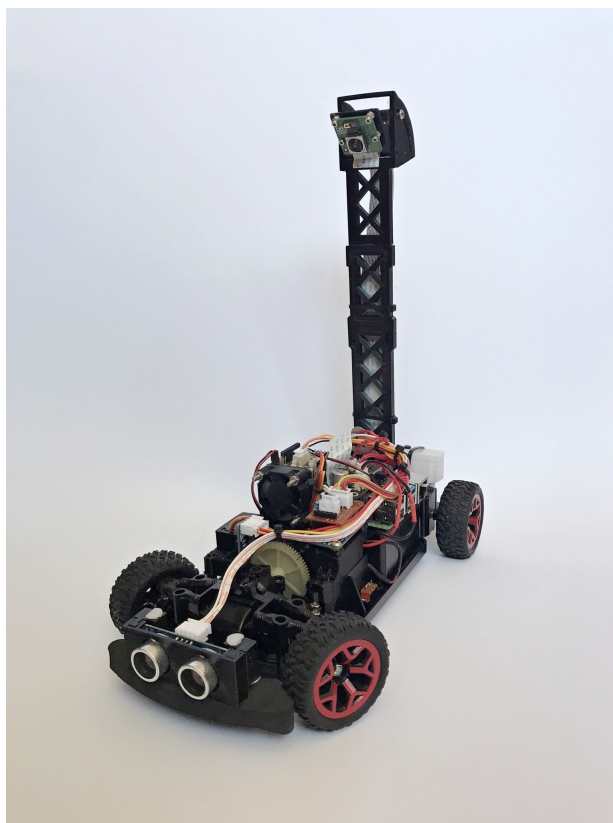
V současnosti téměř každé vozidlo disponuje asistenčními systémy, které mají za úkol primárně zabránit dopravním nehodám a zvýšit komfort řidiče. Některé automobilové společnosti dokonce usilují o vytvoření plně autonomního vozidla, které by bylo schopno pohybovat se v silničním provozu zcela samostatně bez pomoci řidiče. Cílem této práce bylo vytvořit výukový robotický model, na kterém si žáci středních nebo základních škol budou moci vyzkoušet naprogramovat elementární asistenční systémy, čímž by si měli rozšířit své znalosti v oblasti robotiky, počítačového vidění, embedded systémů, řízení a regulace.

V první kapitole této práce jsou uvedeny některé hojně používané asistenční systémy. V následující části je popsán princip funkce jednotlivých snímačů, které jsou v automobilovém průmyslu používány. Poté jsou v práci vysvětleny základní algoritmy počítačového vidění užívaného ve vozidlech a principy jejich funkce. V diplomové práci se také objevují výsledky průzkumu trhu, který byl zaměřen na dostupné robotické platformy a jejich řídicí jednotky. V poslední kapitole této práce je uvedena samotná realizace přípravku a její testování.

V rámci práce byl vytvořen robot, jenž byl postaven na platformě RC auta s označením X-Knight RTR 4WD. Jelikož RC auto neodpovídalo daným konstrukčním požadavkům bylo vytvořeno několik dílů, pomocí kterých se tyto požadavky podařilo splnit. Díly byly vymodelovány v programu SolidWorks a následně vytištěny z materiálu PETG na 3D tiskárně. Tento zásah umožnil plynule regulovat rychlost vozidla i míru natočení kol.

Aby byl robot schopen samostatné jízdy, musel být vybaven řídicí elektronikou a různými snímači pro zajištění žádané míry autonomie. Jako řídicí jednotka byl vybrán jednodeskový počítač Raspberry Pi 4, který poskytuje dostatečný výkon i při využití pokročilých algoritmů pro zpracování obrazu. Toto zařízení bylo doplněno deskou s názvem AdeepT Robot HAT, která umožnila připojení ostatních periférií. Robot disponuje snímači pro měření otáček hlavního hřídele, ultrazvukovým senzorem pro měření vzdálenosti objektu od vozidla, kamerovým snímačem s označením Raspberry Pi camera v3 a čtyřkanálovým 8-bitovým AD převodníkem pro měření napětí na jednotlivých člancích Li-ion akumulátoru, jenž napájí celé zařízení. Vozidlo obsahuje také tři signalizačními LED a HID (Human Interface Devices) modul, který má tři programovatelná tlačítka. Výsledný výukový přípravek je uveden na obrázku č. 6.17

Jako součást práce byly vytvořeny knihovny pro ovládání jednotlivých periférií, které byly napsány v programovacím jazyku Python. V tomto jazyku byly také napsány algoritmy pro detekci jedné i více čar, jež jsou snímány pomocí kamerového modulu.



Obr. 6.17: Vytvořený výukový robot

V rámci diplomové práce bylo úkolem vytvořit příklady použití výukového robotu. Jako první příklad byl vytvořen program, pomocí kterého robot dokáže sledovat vodící linii a zastavit v případě, že se před ním objeví jakákoliv překážka. Druhý algoritmus funguje podobným způsobem s tím rozdílem, že je uzpůsoben pro jízdu mezi čarami. K těmto úkonům byl pro ukázkou vytvořen PSD regulátor, jehož konstanty byly zjištěny pomocí metody Ziegler-Nichols. Při vytváření obou programů byl kladen důraz především na použití všech periférií, kterými robot disponuje. Oba dva algoritmy byly testovány na testovacích drahách, na kterých plnily svou předpokládanou funkci. Testování funkčnosti modelu ukázalo, že se podařilo dosáhnout vytyčených cílů.

Literatura

- [1] SYNOPSISYS: *What is ADAS?* [online]. www.synopsys.com, [cit. 15. 10. 2022]. Dostupné z URL: <<https://www.synopsys.com/automotive/what-is-adas.html>>.
- [2] Editorial Team: *What is a FMCW Radar* [online]. everythingRF, www.everythingrf.com, květen 2022, [cit. 31. 10. 2022]. Dostupné z URL: <<https://www.everythingrf.com/community/what-is-a-fmcw-radar/>>.
- [3] ROYO, S.; BALLESTA-GARCIA, M.: *An Overview of Lidar Imaging Systems for Autonomous Vehicles* [online]. Applied Sciences, vol.9, no. 19, 2019 [cit. 15. 11. 2022]. Dostupné z URL: <<https://www.mdpi.com/2076-3417/9/19/4093>>
- [4] HLAVAČ, V.; SONKA, M.; BOYLE, R.: *Image Processing, Analysis, and Machine Vision* [online]. 3rd ed. Toronto: Thomson, 2008 [cit. 16. 11. 2022]. ISBN isbn978-0495082521. Dostupné z URL: <<https://kgut.ac.ir/useruploads/1550563201478ety.pdf>>
- [5] *Charge-coupled device*. Wikipedia [online]. [cit. 22. 11. 2022]. Dostupné z URL: <https://en.wikipedia.org/wiki/Charge-coupled_device>
- [6] *Active-pixel sensor*. Wikipedia [online]. [cit. 22. 11. 2022]. Dostupné z URL: <https://en.wikipedia.org/wiki/Active-pixel_sensor>
- [7] HORÁK, K.; KALOVÁ, I.: *Řízení expoziční doby CCD kamery* [online]. Vysoké učení technické v Brně, 2006 [cit. 18. 11. 2022]. Dostupné z URL: <<http://www.elektrorevue.cz/clanky/06019/index.html>>
- [8] *Understanding Camera Sensors for Machine Vision Applications* Imaging Electronics 101 [online]. [cit. 18. 11. 2022]. Dostupné z URL: <[https://www.edmundoptics.com/knowledge-center/application-notes/imaging/understanding-camera-sensors-for-machine-vision-applications/](https://www.edmundooptics.com/knowledge-center/application-notes/imaging/understanding-camera-sensors-for-machine-vision-applications/)>
- [9] HORÁK, K.; KALOVÁ, I.: *Optické metody měření 3D objektů* [online]. Vysoké učení technické v Brně, 2005 [cit. 21. 12. 2022]. Dostupné z URL: <<http://www.elektrorevue.cz/clanky/05023/index.html>>
- [10] YAMASHITA, R.; NISHIO, M.; KINH GIAN DO, R.; TOGASHI, K.: *Convolutional neural networks: an overview and application in radiology* [online]. 2018 [cit. 28. 12. 2022]. Dostupné z URL: <<https://insightsimaging.springeropen.com/articles/10.1007/s13244-018-0639-9>>

- [11] MORAVOVÁ, A.: *Text recognition in images using recurrent neural networks* [online]. České vysoké učení technické v Praze, 2016 [cit. 29.12.2022]. Dostupné z URL: <https://dspace.cvut.cz/bitstream/handle/10467/67363/F3-BP-2016-Moravova-Alena-moravova_bakalarska_prace_2016.pdf?sequence=1&isAllowed=y>
- [12] KOSTADINOV, S.: *How Recurrent Neural Networks work* [online]. 2017 [cit. 29.12.2022]. Dostupné z URL: <<https://towardsdatascience.com/learn-how-recurrent-neural-networks-work-84e975feaaf7>>
- [13] KHUONG, B.: *The Basics of Recurrent Neural Networks (RNNs)* [online]. 2019 [cit. 29.12.2022]. Dostupné z URL: <<https://pub.towardsai.net/whirlwind-tour-of-rnns-a11effb7808f>>
- [14] GIRSHICK, R.; DONAHUE, J.; DARRELL, T.; MALIK, J.: *Rich feature hierarchies for accurate object detection and semantic segmentation* [online]. UC Berkeley, 2014 [cit. 30.12.2022]. Dostupné z URL: <<https://arxiv.org/pdf/1311.2524.pdf>>
- [15] GORIS, K.: *Autonomous Mobile Robot Mechanical Design* [online]. Vrije Universiteit Brussel, 2005 [cit. 15.2.2023]. Dostupné z URL: <http://mech.vub.ac.be/multibody/final_works/ThesisKristofGoris.pdf>
- [16] Láskakit: *Stavebnice podvozku - 2 kola* [online]. [cit. 13.3.2023]. Dostupné z URL: <<https://www.laskakit.cz/arduino-podvozek-2wd/>>
- [17] Láskakit: *Stavebnice podvozku - 4 kola* [online]. [cit. 13.3.2023]. Dostupné z URL: <<https://www.laskakit.cz/arduino-podvozek-4wd/>>
- [18] Láskakit: *Makeblock MBot podvozek klon* [online]. [cit. 13.3.2023]. Dostupné z URL: <<https://www.laskakit.cz/makeblock-mbot-podvozek-klon/>>
- [19] Láskakit: *Ai-Thinker ESP32-CAM* [online]. [cit. 13.3.2023]. Dostupné z URL: <<https://www.laskakit.cz/ai-thinker-esp32-cam-2-4ghz-wifi-bluetooth-modul/>>
- [20] Botland: *Magician Chassis v2* [online]. [cit. 13.3.2023]. Dostupné z URL: <<https://t.ly/vFfC>>
- [21] Botland: *JetRacer* [online]. [cit. 13.3.2023]. Dostupné z URL: <<https://t.ly/g2Q->>

- [22] Aliexpress: *4WD Ackerman Steering RC Car* [online]. [cit. 16. 3. 2023]. Dostupné z URL: <<https://www.aliexpress.com/i/1005001314610613.html>>
- [23] Aliexpress: *4WD Smart Steering Robot Car* [online]. [cit. 16. 3. 2023]. Dostupné z URL: <https://www.aliexpress.com/item/4001072024865.html?spm=a2g0o.detail.0.0.32631d4bByufFA&gps-id=pcDetailBottomMoreThisSeller&scm=1007.13339.291025.0&scm_id=1007.13339.291025.0&scm-url=1007.13339.291025.0&pvid=c3b42d38-a294-49dd-93df-81e36af5a105&_t=gps-id:pcDetailBottomMoreThisSeller,scm-url:1007.13339.291025.0,pvid:c3b42d38-a294-49dd-93df-81e36af5a105,tpp_buckets:668%232846%238116%232002&pdp_ext_f=%7B%22sku_id%22%3A%2210000014059562980%22%2C%22sceneId%22%3A%223339%22%7D&pdp_npi=3%40dis%21CZK%211646.7%211564.35%21%21%21%21%402101c84f16759351588603478e797b%2110000014059562980%21rec%21CZ%21885237905>
- [24] Aliexpress: *DC Stepper Motor HAT* [online]. [cit. 16. 3. 2023]. Dostupné z URL: <https://www.aliexpress.com/item/32969135692.html?spm=a2g0o.productlist.main.5.2ffbFJ4rFJ4rR0&algo_pvid=4d5b82f3-b0cc-40b9-bb8f-ca831323c820&algo_exp_id=4d5b82f3-b0cc-40b9-bb8f-ca831323c820-2&pdp_ext_f=%7B%22sku_id%22%3A%2266589653495%22%7D&pdp_npi=3%40dis%21CZK%21633.91%21633.91%21%21%21%21%40211be72e16770634835165800d06be%2166589653495%21sea%21CZ%21885237905&curPageLogUid=ZtwVAuZng7J1>
- [25] Aliexpress: *Adept Robot HAT* [online]. [cit. 16. 3. 2023]. Dostupné z URL: <https://www.aliexpress.com/item/1005004688639655.html?spm=a2g0o.order_list.order_list_main.5.21ef1802E1c5hd>
- [26] FYFT: *F4 5A AIO FC* [online]. [cit. 3. 4. 2023]. Dostupné z URL: <<https://www.fyft.cz/f4-5a-aio-fc-betafpv-1s-elrs-24g/>>
- [27] Conrad: *Banana Pi Bpi - M5* [online]. [cit. 4. 4. 2023]. Dostupné z URL: <https://t.ly/_FG9>
- [28] RPishop: *Raspberry Pi 4* [online]. [cit. 4. 4. 2023]. Dostupné z URL: <<https://rpishop.cz/raspberry-pi-4/1598-raspberry-pi-4-model-b-4gb-ram-765756931182.html>>
- [29] RPishop: *Jetson Nano Developer Kit* [online]. [cit. 4. 4. 2023]. Dostupné z URL: <<https://rpishop.cz/jetson-nano/1636-nvidia-jetson-nano-developer-kit-verze-b01.html>>

Seznam příloh

A	Výkresová dokumentace tištěných dělů	77
B	Obsah CD	92

A Výkresová dokumentace tištěných dílů

V této příloze se nachází veškerá výkresová dokumentace k vytvořeným dílům určených pro 3D tisk.

4 3 2 1

F

F

E

E

D

D

C

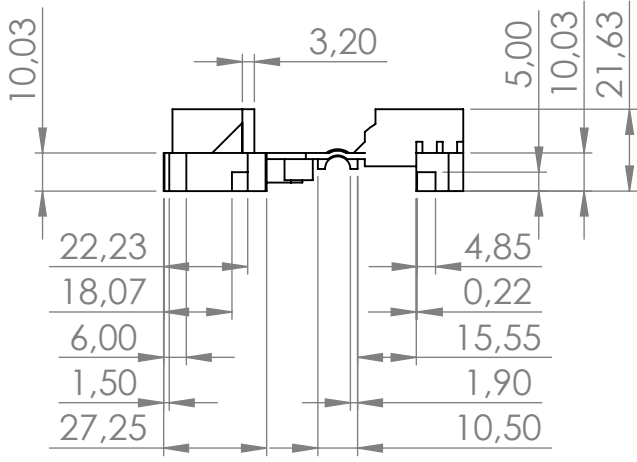
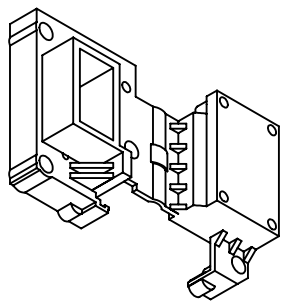
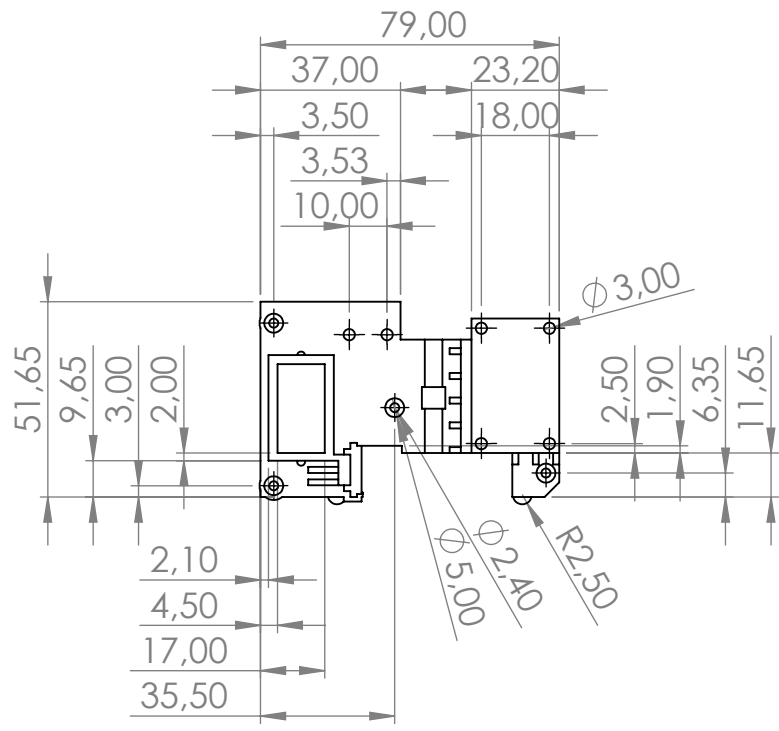
C

B

B

A

A



POKUD NENÍ UVEDENO JINAK:
 JEDNOTKY JSOU V MILIMETRECH
 DRSNOST:
 TOLERANCE:
 LINEÁRNÍ:
 ÚHLOVÁ:

OPRACOVÁNÍ:

NEUPRAVOVAT MĚŘÍTKO VÝKRESU

ZMĚNA

	JMÉNO	PODPIS	DATUM
NAVRHL	Vojtěch Stražil		
PŘEZKOUSĚL			
SCHVÁLIL			
VÝROBA			
Z. JAKOSTI			

NÁZEV:
Kryt motoru

Č. VÝKRESU
kryt_podvozku

MATERIÁL:
PETG

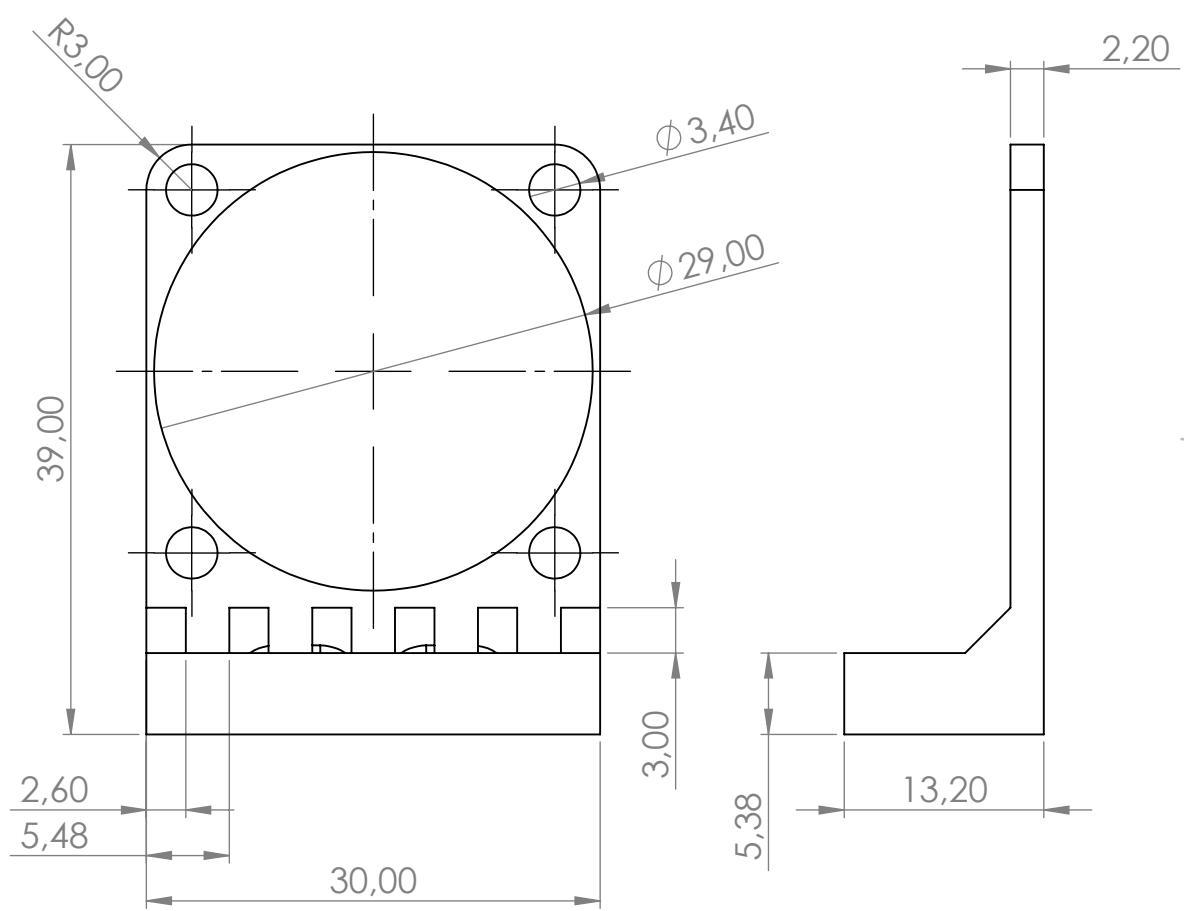
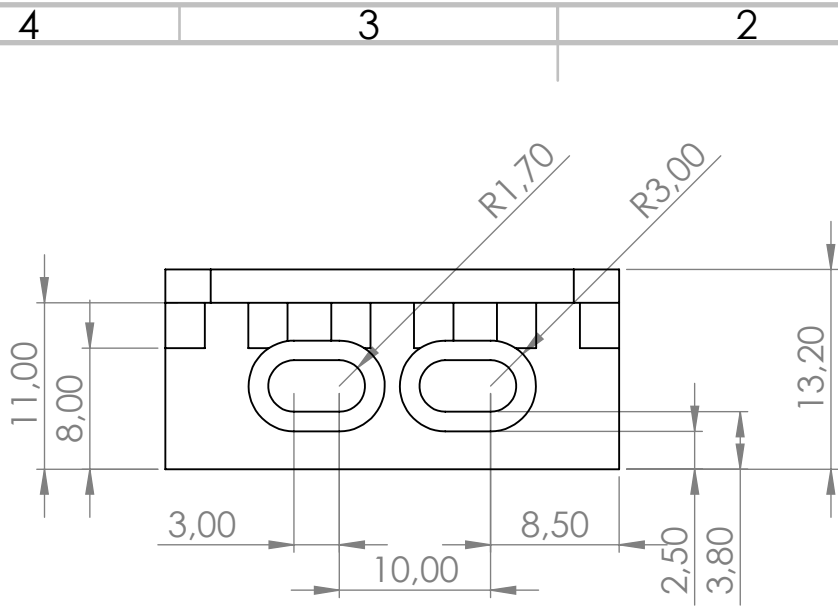
HMOTNOST:

Č. VÝKRESU
A4

MĚŘÍTKO: 1:2

LIST 1 Z 1 LISTŮ

4 3 2 1



POKUD NENÍ UVEDENO JINAK:
 JEDNOTKY JSOU V MILIMETRECH
 DRSNOST:
 TOLERANCE:
 LINEÁRNÍ:
 ÚHLOVÁ:

OPRACOVÁNÍ:

NEUPRAVOVAT MĚŘÍTKO VÝKRESU ZMĚNA

	JMÉNO	PODPIS	DATUM
NAVRHL	Vojtěch Stražil		
PŘEZKOUSHL			
SCHVÁLIL			
VÝROBA			
Z. JAKOSTI			

NÁZEV:
Držák ventilátoru

Č. VÝKRESU
drzak_ventilatoru

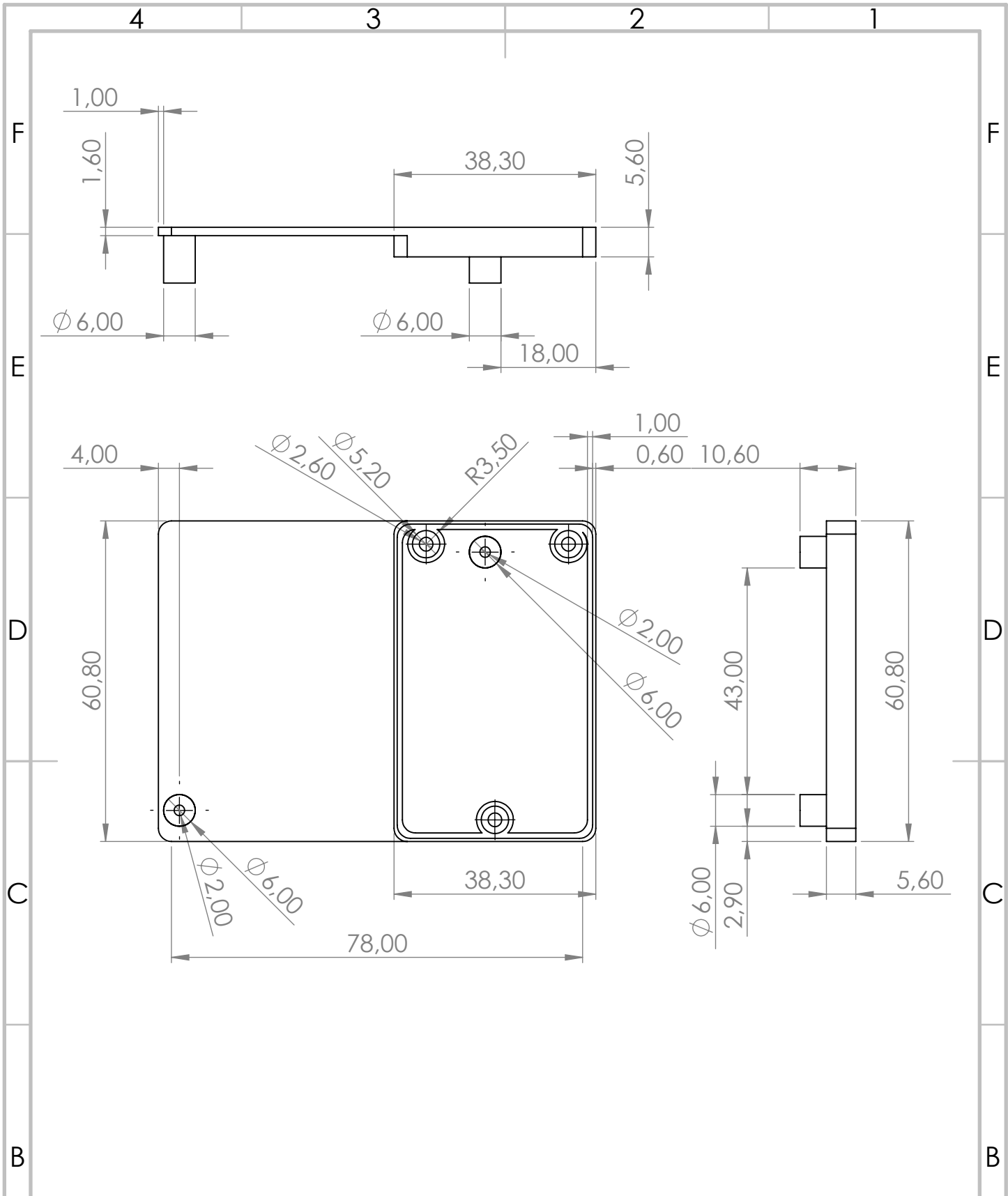
MĚŘÍTKO:2:1

LIST 1 Z 1 LISTŮ

MATERIÁL:
PETG

A4

HMOTNOST:



POKUD NENÍ UVEDENO JINAK:
 JEDNOTKY JSOU V MILIMETRECH
 DRSNOST:
 TOLERANCE:
 LINEÁRNÍ:
 ÚHLOVÁ:

OPRACOVÁNÍ:

NEUPRAVOVAT MĚŘÍTKO VÝKRESU

ZMĚNA

	JMÉNO	PODPIS	DATUM
NAVRHL	Vojtěch Stražil		
PŘEZKOUSĚL			
SCHVÁLIL			
VÝROBA			
Z. JAKOSTI			

NÁZEV:
Základna

Č. VÝKRESU
zakladna

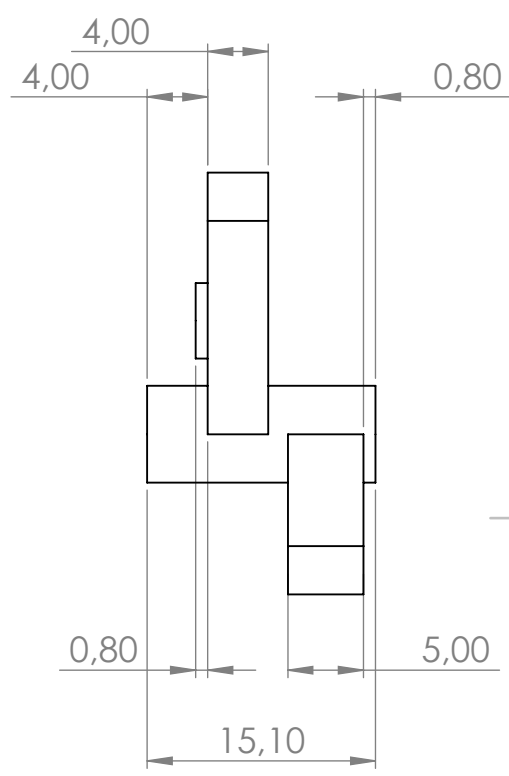
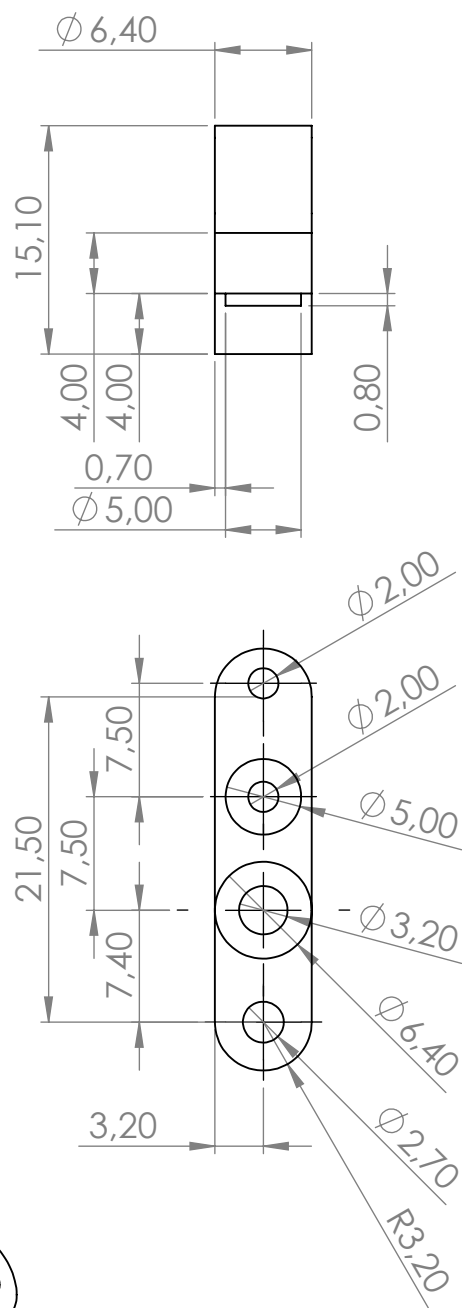
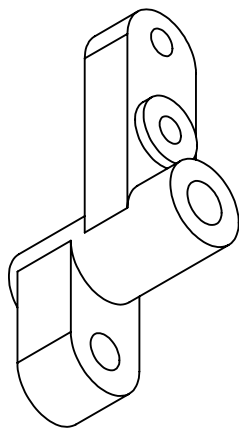
MĚŘÍTKO: 1:1

LIST 1 Z 1 LISTŮ

MATERIÁL:
PETG

A4

HMOTNOST:



POKUD NENÍ UVEDENO JINAK:
 JEDNOTKY JSOU V MILIMETRECH
 DRSNOST:
 TOLERANCE:
 LINEÁRNÍ:
 ÚHLOVÁ:

OPRACOVÁNÍ:

NEUPRAVOVAT MĚŘÍTKO VÝKRESU

ZMĚNA

	JMÉNO	PODPIS	DATUM
NAVRHL	Vojtěch Stražil		
PŘEZKOUSĚL			
SCHVÁLIL			
VÝROBA			
Z. JAKOSTI			

NÁZEV:

Táhlo řízení

MATERIÁL:

PETG

Č. VÝKRESU

tahlo_rizeni

A4

HMOTNOST:

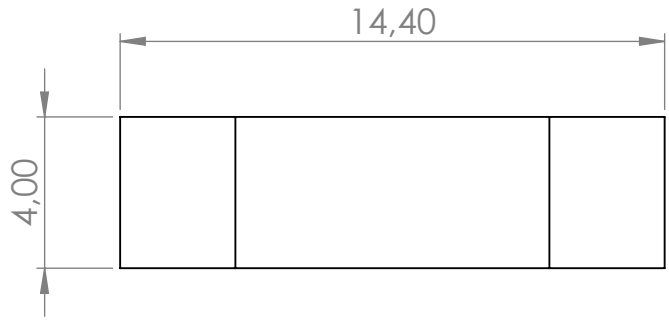
MĚŘÍTKO:2:1

LIST 1 Z 1 LISTŮ

4 3 2 1

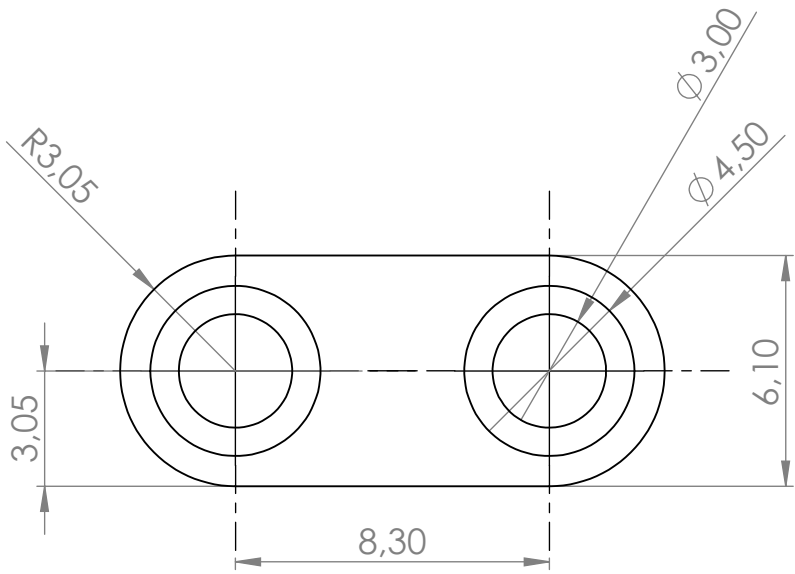
F

F



E

E

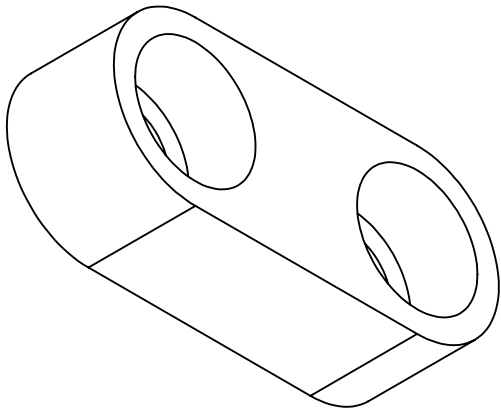


D

D

C

C



B

B

POKUD NENÍ UVEDENO JINAK:
 JEDNOTKY JSOU V MILIMETRECH
 DRSNOST:
 TOLERANCE:
 LINEÁRNÍ:
 ÚHLOVÁ:

OPRACOVÁNÍ:

NEUPRAVOVAT MĚŘÍTKO VÝKRESU

ZMĚNA

	JMÉNO	PODPIS	DATUM
NAVRHL	Vojtěch Stražil		
PŘEZKOUŠEL			
SCHVÁLIL			
VÝROBA			
Z. JAKOSTI			

NÁZEV:

Táhlo řízení spojka

Č. VÝKRESU

tahlo_rizeni_s

A4

MATERIÁL:
PETG

HMOTNOST:

MĚŘÍTKO:5:1

LIST 1 Z 1 LISTŮ

4 3 2 1

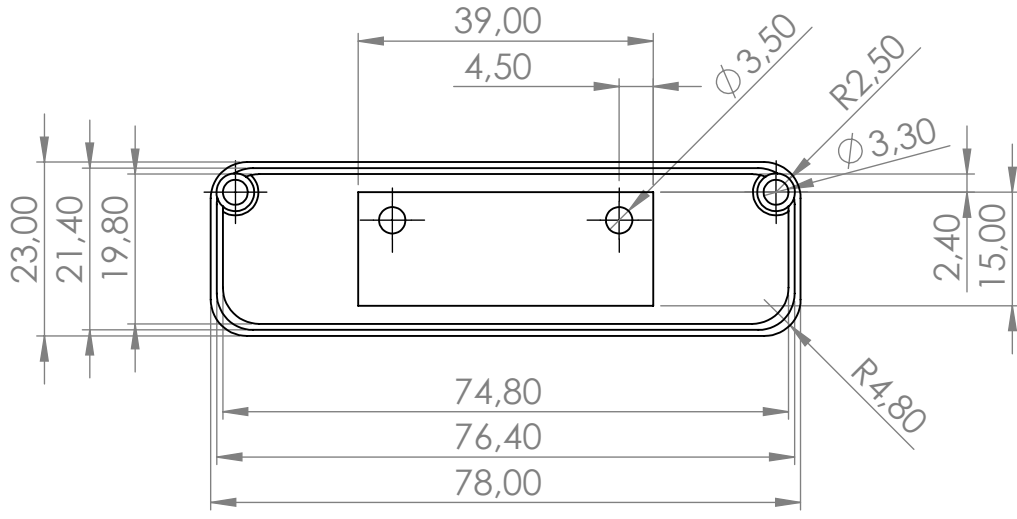
A

A

4 3 2 1

F

F

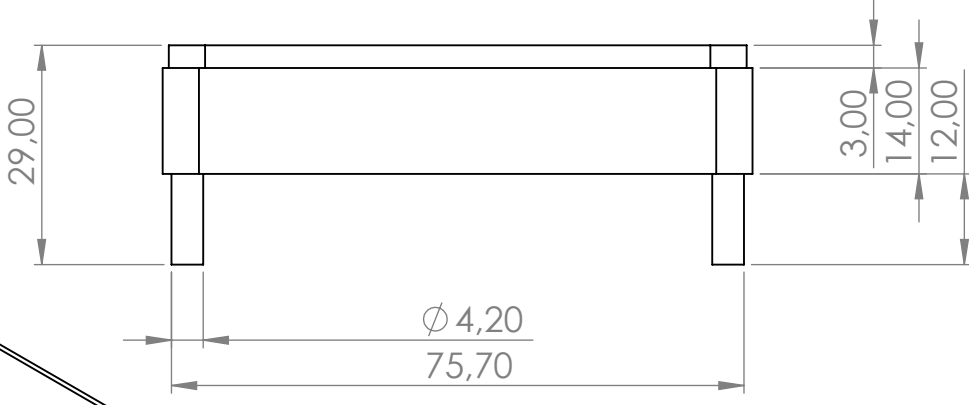


E

E

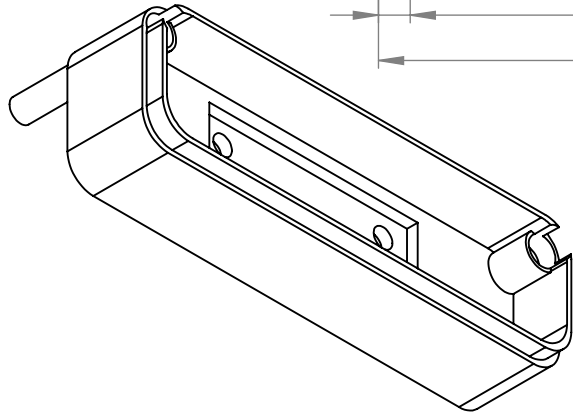
D

D



C

C



B

B

POKUD NENÍ UVEDENO JINAK:
 JEDNOTKY JSOU V MILIMETRECH
 DRSNOST:
 TOLERANCE:
 LINEÁRNÍ:
 ÚHLOVÁ:

OPRACOVÁNÍ:

NEUPRAVOVAT MĚŘÍTKO VÝKRESU

ZMĚNA

	JMÉNO	PODPIS	DATUM
NAVRHL	Vojtěch Stražil		
PŘEZKOUSĚL			
SCHVÁLIL			
VÝROBA			
Z. JAKOSTI			

NÁZEV:
Spodní část

Č. VÝKRESU
HID_spodek

MĚŘÍTKO: 1:1

LIST 1 Z 1 LISTŮ

MATERIÁL:
PETG

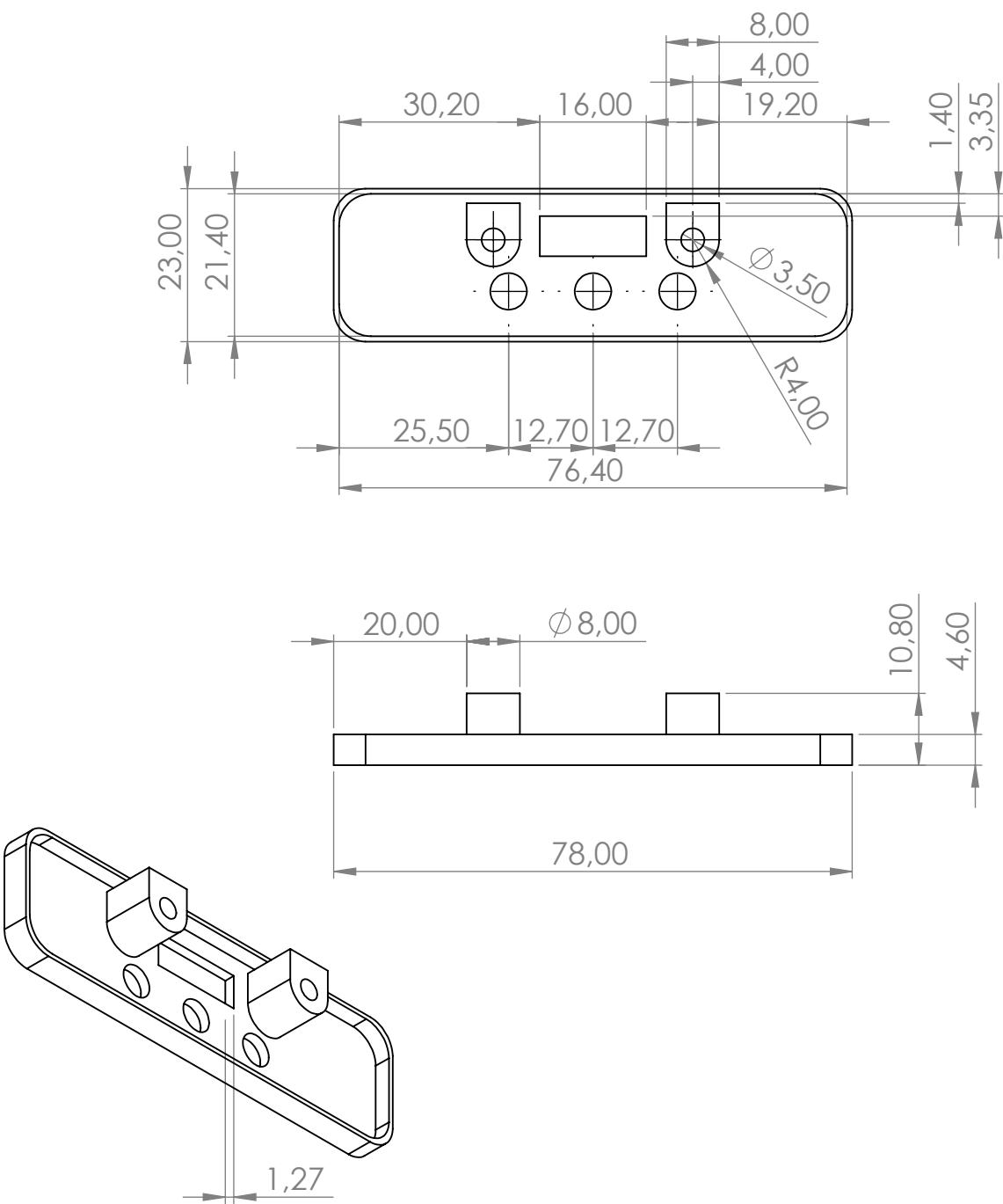
A4

HMOTNOST:

4 3 2 1

A

A



POKUD NENÍ UVEDENO JINAK:
 JEDNOTKY JSOU V MILIMETRECH
 DRSNOST:
 TOLERANCE:
 LINEÁRNÍ:
 ÚHLOVÁ:

OPRACOVÁNÍ:

NEUPRAVOVAT MĚŘÍTKO VÝKRESU

ZMĚNA

	JMÉNO	PODPIS	DATUM
NAVRHL	Vojtěch Stražil		
PŘEZKOUSEL			
SCHVÁLIL			
VÝROBA			
Z. JAKOSTI			

NÁZEV:

Vrch

MATERIÁL:

PETG

Č. VÝKRESU

HID_vrch

A4

HMOTNOST:

MĚŘÍTKO: 1:1

LIST 1 Z 1 LISTŮ

4 3 2 1

F

F

E

E

D

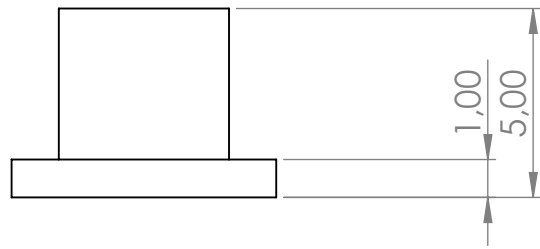
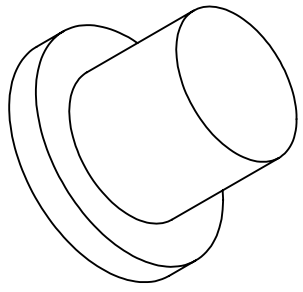
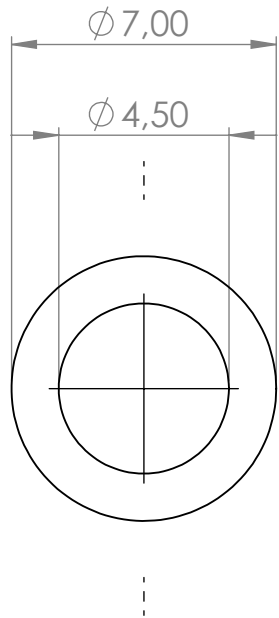
D

C

C

B

B



POKUD NENÍ UVEDENO JINAK:
 JEDNOTKY JSOU V MILIMETRECH
 DRSNOST:
 TOLERANCE:
 LINEÁRNÍ:
 ÚHLOVÁ:

OPRACOVÁNÍ:

NEUPRAVOVAT MĚŘÍTKO VÝKRESU

ZMĚNA

	JMÉNO	PODPIS	DATUM
NAVRHL	Vojtěch Stražil		
PŘEZKOUSĚL			
SCHVÁLIL			
VÝROBA			
Z. JAKOSTI			

NÁZEV: **Tlačítko**

Č. VÝKRESU: **HID_tlacitko**

MĚŘÍTKO: 5:1

LIST 1 Z 1 LISTŮ

MATERIÁL: **PETG**

A4

HMOTNOST:

4 3 2 1

A

A

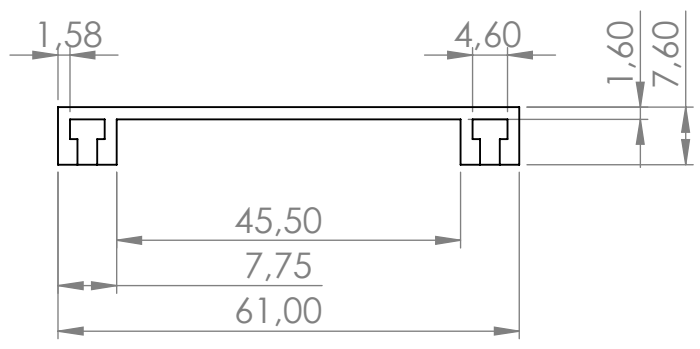
4 3 2 1

F

F

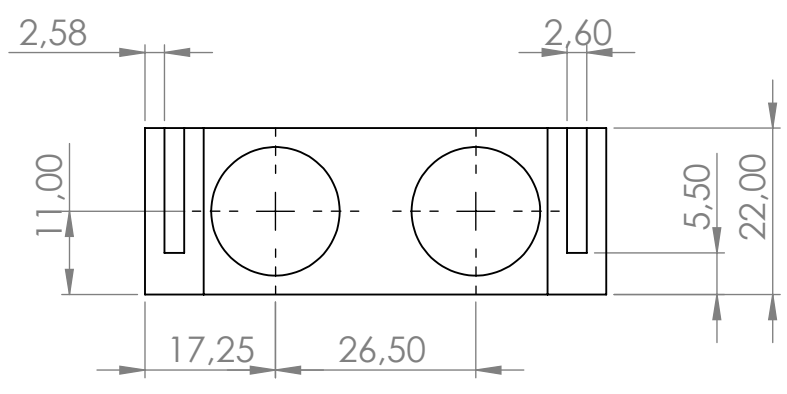
E

E



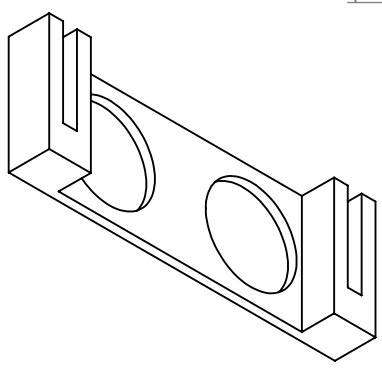
D

D



C

C



B

B

POKUD NENÍ UVEDENO JINAK:
 JEDNOTKY JSOU V MILIMETRECH
 DRSNOST:
 TOLERANCE:
 LINEÁRNÍ:
 ÚHLOVÁ:

OPRACOVÁNÍ:

NEUPRAVOVAT MĚŘÍTKO VÝKRESU

ZMĚNA

	JMÉNO	PODPIS	DATUM
NAVRHL	Vojtěch Stražil		
PŘEZKOUSĚL			
SCHVÁLIL			
VÝROBA			
Z. JAKOSTI			

NÁZEV:

Ultrazvuk

MATERIÁL:

PETG

Č. VÝKRESU

drzak_ultrazvuk

A4

HMOTNOST:

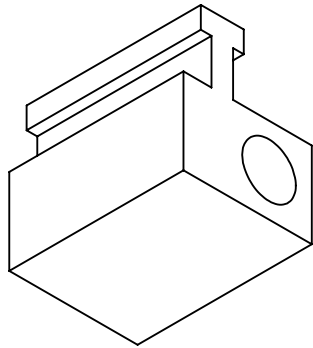
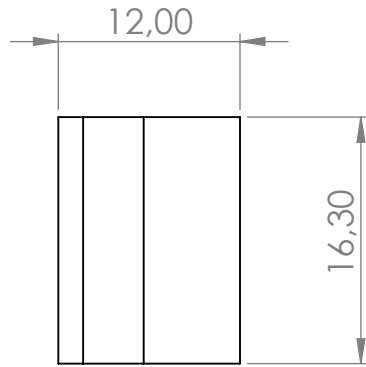
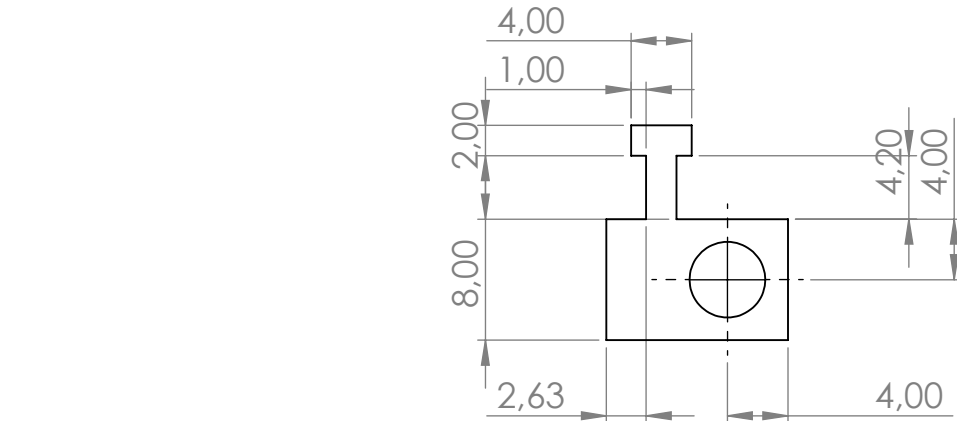
MĚŘÍTKO:1:1

LIST 1 Z 1 LISTŮ

4 3 2 1

A

A



POKUD NENÍ UVEDENO JINAK:
 JEDNOTKY JSOU V MILIMETRECH
 DRSNOST:
 TOLERANCE:
 LINEÁRNÍ:
 ÚHLOVÁ:

OPRACOVÁNÍ:

NEUPRAVOVAT MĚŘÍTKO VÝKRESU

ZMĚNA

	JMÉNO	PODPIS	DATUM
NAVRHL	Vojtěch Stražil		
PŘEZKOUŠEL			
SCHVÁLIL			
VÝROBA			
Z. JAKOSTI			

NÁZEV:

Jezdec

MATERIÁL:

PETG

Č. VÝKRESU

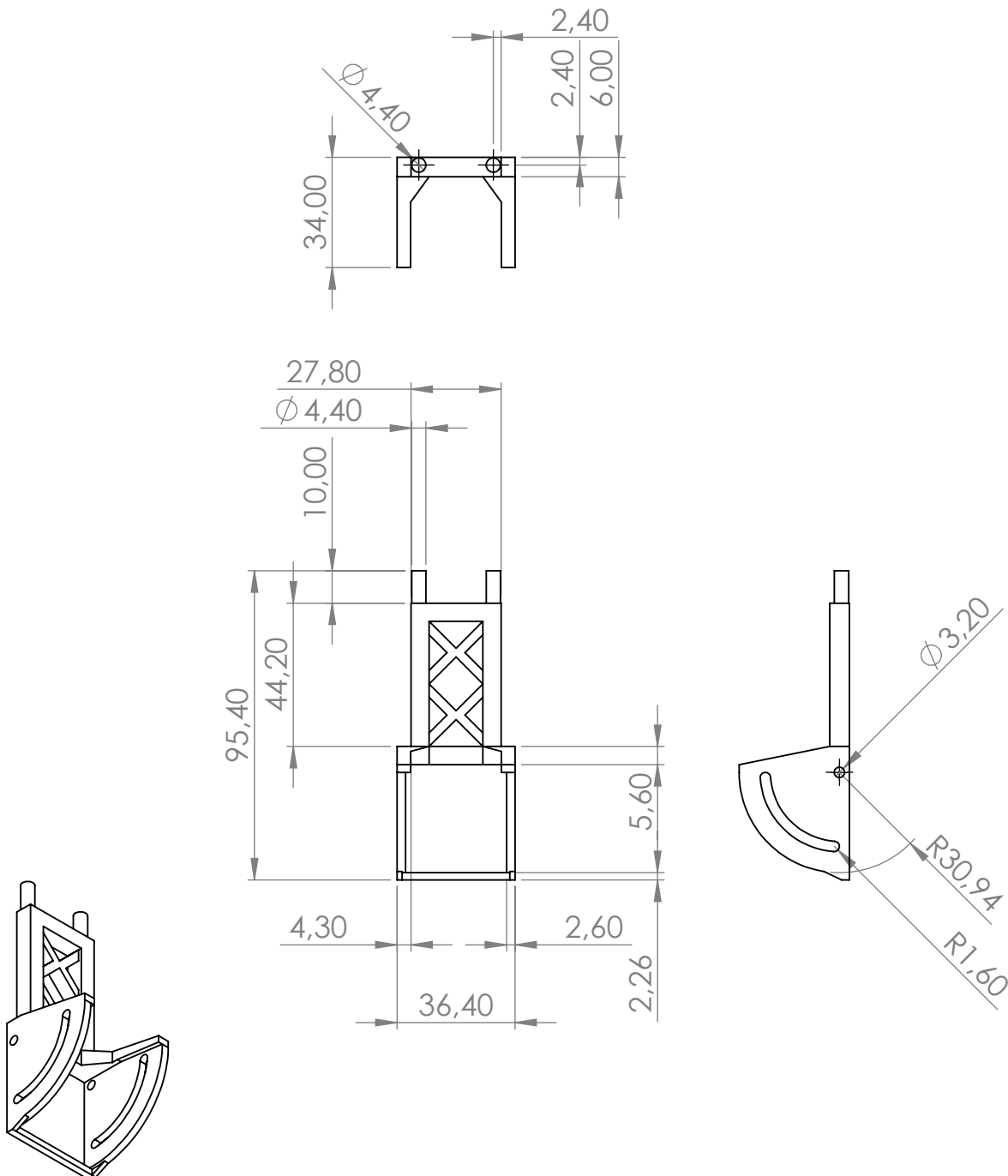
ultrazvuk_jezdec

A4

HMOTNOST:

MĚŘÍTKO:2:1

LIST 1 Z 1 LISTŮ



POKUD NENÍ UVEDENO JINAK:
 JEDNOTKY JSOU V MILIMETRECH
 DRSNOST:
 TOLERANCE:
 LINEÁRNÍ:
 ÚHLOVÁ:

OPRACOVÁNÍ:

NEUPRAVOVAT MĚŘÍTKO VÝKRESU

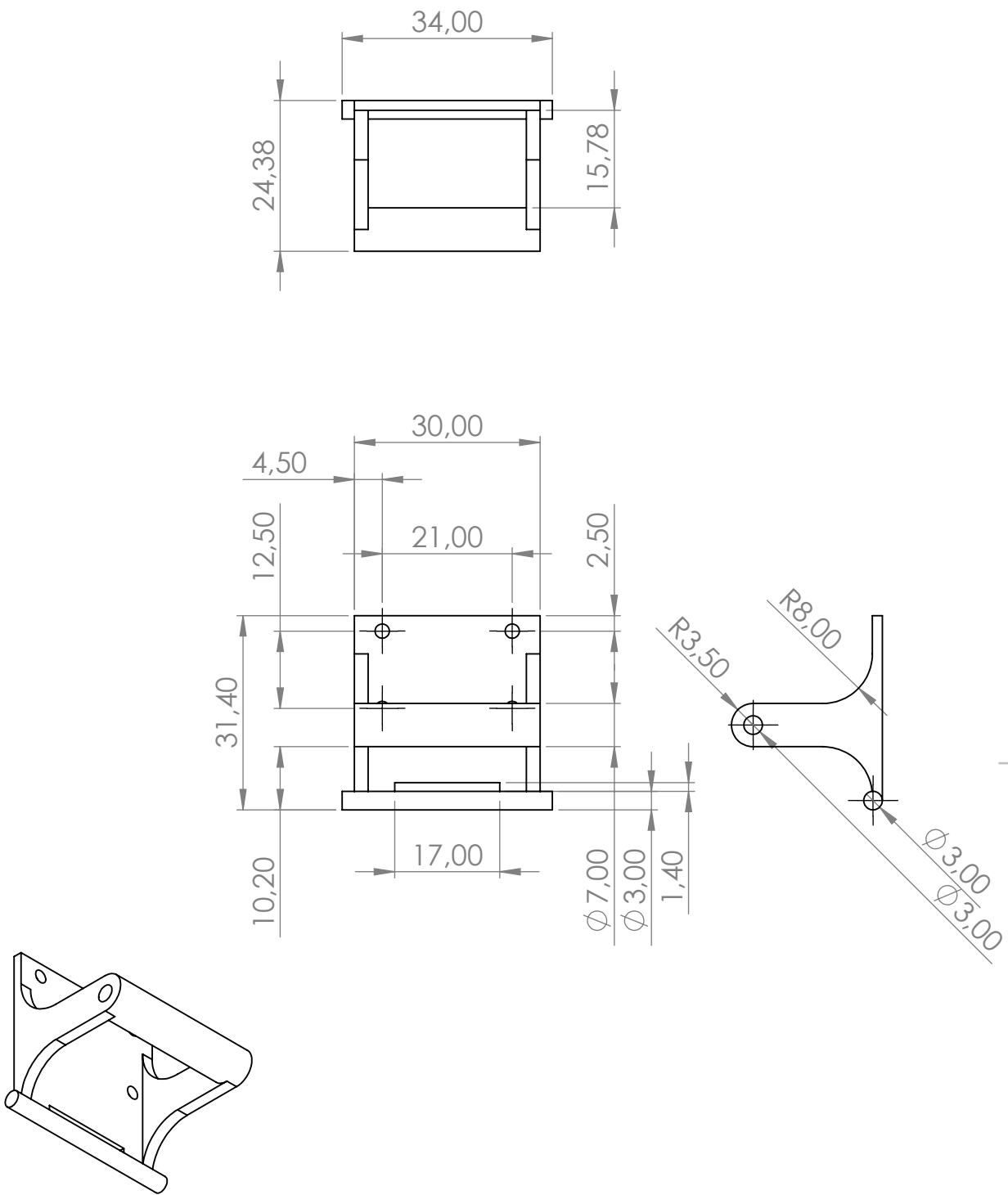
ZMĚNA

	JMÉNO	PODPIS	DATUM
NAVRHL	Vojtěch Stražil		
PŘEZKOUŠEL			
SCHVÁLIL			
VÝROBA			
Z. JAKOSTI			

NÁZEV:	Držák kamery	
Č. VÝKRESU	drzak_kamery	A4
MĚŘÍTKO:	1:2	LIST 1 Z 1 LISTŮ

MATERIÁL:
PETG

HMOTNOST:



POKUD NENÍ UVEDENO JINAK:
 JEDNOTKY JSOU V MILIMETRECH
 DRSNOST:
 TOLERANCE:
 LINEÁRNÍ:
 ÚHLOVÁ:

OPRACOVÁNÍ:

NEUPRAVOVAT MĚŘÍTKO VÝKRESU

ZMĚNA

	JMÉNO	PODPIS	DATUM
NAVRHL	Vojtěch Stražil		
PŘEZKOUSEL			
SCHVÁLIL			
VÝROBA			
Z. JAKOSTI			

NÁZEV:

Držák kamery úchyt

Č. VÝKRESU

drzak_kamery_u

A4

MATERIÁL:

PETG

HMOTNOST:

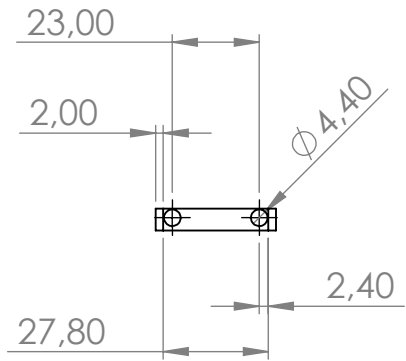
MĚŘÍTKO: 1:1

LIST 1 Z 1 LISTŮ

4 3 2 1

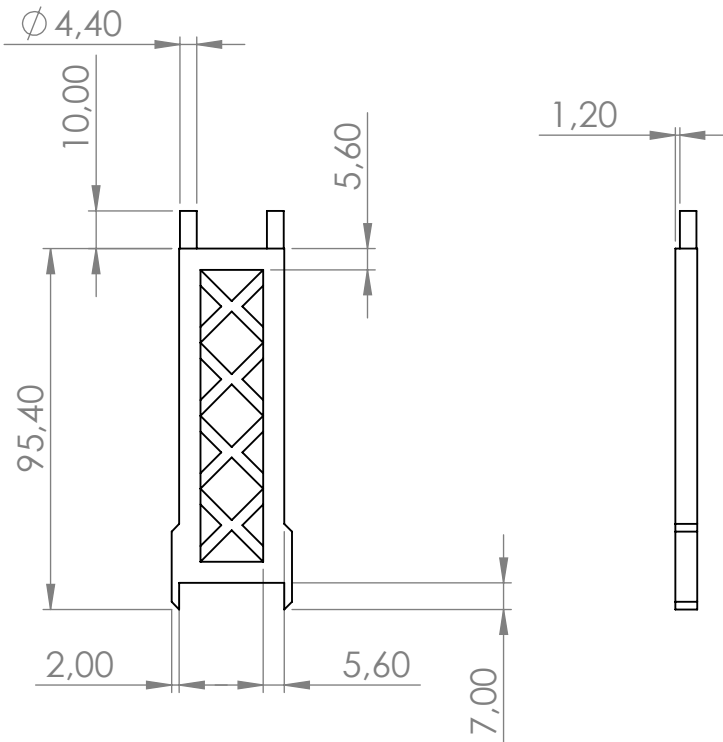
F

F



E

E

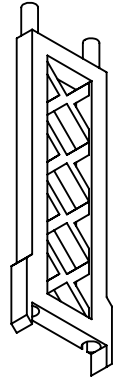


D

D

C

C



B

B

POKUD NENÍ UVEDENO JINAK:
 JEDNOTKY JSOU V MILIMETRECH
 DRSNOST:
 TOLERANCE:
 LINEÁRNÍ:
 ÚHLOVÁ:

OPRACOVÁNÍ:

NEUPRAVOVAT MĚŘÍTKO VÝKRESU

ZMĚNA

A

A

	JMÉNO	PODPIS	DATUM
NAVRHL	Vojtěch Stražil		
PŘEZKOUŠEL			
SCHVÁLIL			
VÝROBA			
Z. JAKOSTI			

NÁZEV:
**Držák kamery
 prodloužení**

Č. VÝKRESU
drzak_kamery_p

MĚŘÍTKO: 1:2

LIST 1 Z 1 LISTŮ

MATERIÁL:
PETG

A4

4 3 2 1

4 3 2 1

F

F

E

E

D

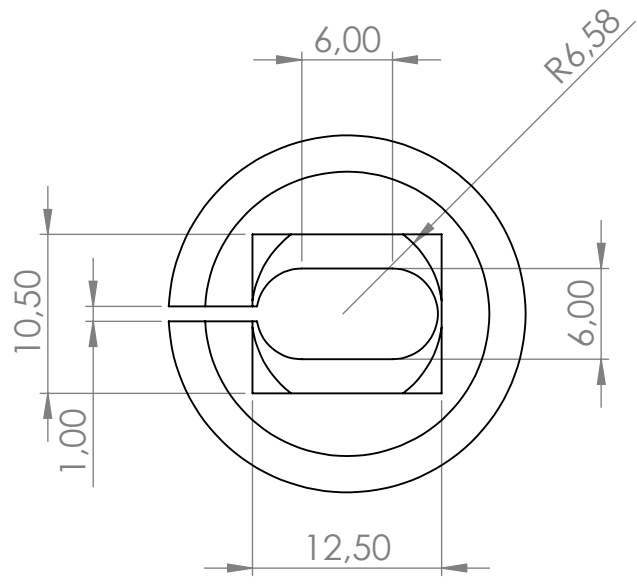
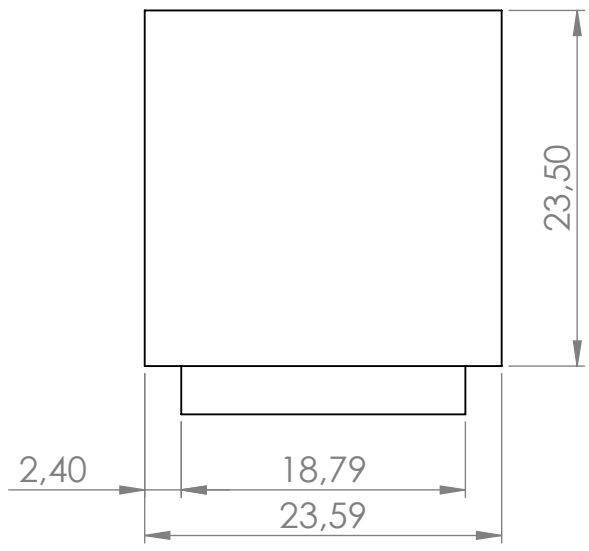
D

C

C

B

B



POKUD NENÍ UVEDENO JINAK:
 JEDNOTKY JSOU V MILIMETRECH
 DRSNOST:
 TOLERANCE:
 LINEÁRNÍ:
 ÚHLOVÁ:

OPRACOVÁNÍ:

NEUPRAVOVAT MĚŘÍTKO VÝKRESU

ZMĚNA

	JMÉNO	PODPIS	DATUM
NAVRHL	Vojtěch Stražil		
PŘEZKOUŠEL			
SCHVÁLIL			
VÝROBA			
Z. JAKOSTI			

NÁZEV:

Adaptér motoru

MATERIÁL:

PETG

Č. VÝKRESU

adapter_motoru

A4

HMOTNOST:

MĚŘÍTKO:2:1

LIST 1 Z 1 LISTŮ

4 3 2 1

A

A

B Obsah CD

Přiložené CD obsahuje elektronickou podobu diplomové práce ve formátu PDF. Na CD se dále nachází zdrojová složka textu této práce s příponou ZIP, která byla vyextrahována z webové aplikace Overleaf, a všechny ostatní soubory potřebné k zreplicování vytvořeného výukového robotu. Seznam příloh je uveden zde:

- Podklady k 3D dílům
 - Modely ve formátu SLDPRT
 - Modely ve formátu STL
 - Výkresy k modelům
- Zdrojové kódy vytvořených programů
 - Jízda po jedné čáře
 - Jízda mezi čarami
- Videozáznamy jízdy
 - Jízda po jedné čáře
 - Jízda po jedné čáře a zastavení na překážku
 - Jízda mezi čarami
 - Jízda mezi čarami a zastavení na překážku
- Zdrojový kód textu
- Elektronická podoba práce ve formátu PDF