



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

VLÁČEK ŘÍZENÝ POČÍTAČEM

MODEL TRAIN CONTROLLED BY COMPUTER

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

ZDENĚK KOLBA

VEDOUCÍ PRÁCE

SUPERVISOR

prof. Dr. Ing. PAVEL ZEMČÍK, dr. h. c.

BRNO 2023

Zadání bakalářské práce



148591

Ústav: Ústav počítačové grafiky a multimédií (UPGM)
Student: **Kolba Zdeněk**
Program: Informační technologie
Specializace: Informační technologie
Název: **Vláček řízený počítačem**
Kategorie: Vestavěné systémy
Akademický rok: 2022/23

Zadání:

1. Prostudujte způsob řízení modelu vláčku počítačem, případně "embedded počítačem", a to jak studiem literatury, tak i existujících realizací takového řízení.
2. Navrhněte vhodný způsob řízení vláčku a také způsob jeho ovládání uživatelem, přitom zvažte použití například mobilního zařízení jako prostředku uživatelského rozhraní, monitorování vláčku kamerou apod.
3. Analyzujte a popište možnosti a dosažitelné vlastnosti navrženého řešení.
4. Navržené řešení implementujte a demonstруйте například na jednoduchém malém kolejišti.
5. Diskutujte dosažené výsledky a možnosti pokračování práce.

Literatura:

Dle pokyů vedoucího

Při obhajobě semestrální části projektu je požadováno:

Body 1-3 zadání

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

Vedoucí práce: **Zemčík Pavel, prof. Dr. Ing., dr. h. c.**

Vedoucí ústavu: Černocký Jan, prof. Dr. Ing.

Datum zadání: 1.11.2022

Termín pro odevzdání: 10.5.2023

Datum schválení: 5.5.2023

Abstrakt

Práce na téma „Vláček řízený počítačem“ se zabývá ověřením základního kontextu modelové železnice, pojmů a existujících standardů nebo zvyklostí jako základu pro následné zhodnocení možnosti návrhu řízení vláčku počítačem. Záměrem bylo navrhnout základní řešení, které by se dalo následně rozšiřovat. Byly preferovány otevřené komponenty jako základní mikroprocesor a volně dostupné protokoly nebo standardy. Výsledkem práce je rozšiřitelná aplikace pro ovládání modelu železnice s podporou jednoduché automatizace.

Abstract

Bachelor thesis „Computer controlled train model“ is focused on basic evaluation of terms and existing standards as basis for train control system design. Idea was to design basis solution ready for next extension. Open components, such as microprocessor unit, protocols or standards were preferred. The result of the thesis is an extensible application for controlling the railway model with support for simple automation.

Klíčová slova

Řízení modelové železnice, DCC, OpenCV

Keywords

Model railway control, DCC, OpenCV

Citace

KOLBA, Zdeněk. *Vláček řízený počítačem*. Brno, 2023. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce prof. Dr. Ing. Pavel Zemčík, dr. h. c.

Vláček řízený počítačem

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením prof. Dr. Ing. Pavla Zemčíka, dr. h. c. Uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....
Zdeněk Kolba
30. července 2023

Poděkování

Děkuji tímto prof. Dr. Ing. Pavlu Zemčikovi, dr. h. c. za cenné připomínky a rady, které mi poskytl při vypracování závěrečné práce.

Obsah

1	Úvod	3
2	Problematika modelové železnice	4
2.1	Způsoby ovládání	4
2.2	Vývoj kolejí pro modelovou železnici	7
2.3	Rozchody kolejí a užívaná měřítka modelů	9
3	Shrnutí současného stavu	10
3.1	Komerční řešení	10
3.2	Open source řešení	12
4	Návrh řešení	17
4.1	Analýza současných řešení	17
4.2	Mechanická a elektrická konstrukce – komponenty	17
4.3	Software – použité komponenty a knihovny	18
4.4	Návrh grafického uživatelského rozhraní	21
5	Realizace	22
5.1	Hardware	22
5.2	Grafické uživatelské rozhraní	26
5.3	Sériová komunikace	28
5.4	Zpracování videa	30
5.5	Podpůrné funkce	33
6	Závěr	38
	Literatura	39
A	Obsah přiloženého média	41

Seznam obrázků

2.1	Analogový ovladač značky Hammant & Morgan schopen řízení až 5 vlaků.	5
2.2	Určení směru jízdy podle polaroty kolejí.	5
2.3	Určení směru jízdy podle polaroty trakčního vedení.	5
2.4	Logo DCC.	6
2.5	Schéma DCC.	7
2.6	Profilové koleje s „dřevěnými“pražci	8
2.7	Profilové koleje s podložím.	8
2.8	Parametry profilových kolejí dle normy NEM 311.	8
2.9	Velikosti modelů.	9
3.1	Ovladač Model train Central Station 3 plus.	10
3.2	Grafické uživatelské rozhraní aplikace Rocrail.	11
3.3	Grafické uživatelské rozhraní aplikace Railroad Automation.	12
3.4	Deska Arduino UNO WiFi REV2.	14
3.5	Schéma zapojení vývodů Arduino UNO WiFi REV2.	15
3.6	Deska Arduino Motor Shield Rev3.	16
4.1	Příklad struktury přenášeného paketu.	19
4.2	ArUco značky.	20
5.1	Kolejiště	22
5.2	Napojení výhybky	23
5.3	Napájení kolejiště	23
5.4	Svorkovnice	24
5.5	Výhybkový dekodér	24
5.6	Kamera C920	25
5.7	Manuální ovládání	27
5.8	Automatické ovládání	27
5.9	Umístění ArUco značky	31
5.10	Umístění ArUco značky	31
5.11	Hranice úseků v zobrazeném videu	32
5.12	Nastavení kamery - Zesilovač zpracování videa	33
5.13	Nastavení kamery - Ovládání fotoaparátu	34
5.14	Výpis provedených kroků automatizace	37

Kapitola 1

Úvod

Modelová železnice je inspirována zájmem realizovat zmenšenou podobu prvků skutečné železnice jako jsou lokomotivy, vlaky, trať a její prvky včetně blízkého okolí železnice. Podle preference se může jednat o zmenšeninu skutečné podoby části trati včetně budov a terénu, nebo náhodnou instalaci podle autorovy fantazie. Mým záměrem bylo v této práci realizovat jednoduché kolejiště se základními prvky umožňujícími řízení vláčku počítačem.

Práce se zabývá zmapováním základního kontextu modelové železnice, pojmů a existujících standardů nebo zvyklostí jako základu pro následné zhodnocení možnosti návrhu řízení vláčku počítačem. Vzhledem k tomu že mým záměrem bylo navrhnout základní řešení které by se dalo následně rozšiřovat, preferoval jsem otevřené komponenty jako základní mikroprocesor a volně dostupné protokoly nebo standardy.

Chtěl jsem si ověřit možnost praktického použití softwarových komponent pro ovládání mechanických prvků a použití zpracování obrazu jako zpětné vazby do řídicího systému. Práci bych rád v budoucnu využil jako základ pro implementaci pokročilejších metod zpracování obrazu a algoritmů řízení.

V první části jsem zmapoval existující řešení v různé úrovni náročnosti na realizaci, ale na druhou stranu s omezenými možnostmi následného rozšíření. Druhá část se zabývá návrhem řešení řízení vláčku a jeho implementací. Obsahuje část mechanického návrhu a realizace kolejiště, návrh řídicí elektroniky jak výkonové, tak digitální a poslední částí SW aplikace pro ovládání a základní sledování modelu vláčku na modelové železnici.

Kapitola 2

Problematika modelové železnice

V této kapitole jsou shrnuty způsoby ovládání modelových železnic, jejich jednotlivé součásti, jako jsou koleje a modely vlaků, a související normy. Jsou uvedeny zejména postupy bezprostředně související s bakalářskou prací. Kapitola nemá encyklopedický charakter, protože to rozsah bakalářské práce neumožňuje.

2.1 Způsoby ovládání

Napájení modelové železnice

Při zavedení elektrifikace modelové železnice se výrobcům nejvíce osvědčilo použití dvou vzájemně izolovaných kolejnic. Historicky se však objevila i jiná řešení daného problému. Například firma Märklin zavedla trojkolejný systém, kde vnější kolejnice byly vzájemně elektricky propojené a tvořily jeden pól. Prostřední kolejnice představovala druhý pól. Firma Trix zkombinovala oba přístupy, kde všechny tři kolejnice byly vzájemně izolovány a umožňovaly tak provoz dvou vlaků na jednom úseku trati při použití analogového ovládání. Navíc bylo možné pro všechny přístupy rozšířit napájení o trolejové vedení.

Důležitým prvkem designu kolejí bylo i propojení jednotlivých dílů, specificky jednotlivých kolejnic. Pevné a přesné spojení je důležité jak pro průjezd vlaku a zabránění vykolejení, tak i pro spolehlivé vedení proudu kolejnicemi.[17, 19]

Analogové ovládání a řízení

Existují dva hlavní způsoby analogového ovládání – polohové a impulsní. Polohové řízení nelze využít pro automatizaci, jelikož je zcela závislé na vstupu uživatele pomocí páček a jiných typů mechanických spínačů (2.1). Impulsní ovládání může využívat řídicí jednotku, které je zaslán uživatelem impuls a ta následně nastaví potřebné výhybky a sepne napájení na určenou kolej podle naplánované trasy.

Dále lze impulsní řízení dělit na manuální, poloautomatické a plně automatické, kde manuální impulsní řízení nemá žádnou elektronickou zpětnou vazbu z kolejiště a je zcela ovládané uživatelem. Úkony jsou tedy zahajovány i ukončovány impulsem z tlačítka nebo jiného ovládacího prvku. Poloautomatické sekvenční ovládání už má částečnou zpětnou vazbu, kterou poskytují dotekové kontakty na kolejích. Činnosti stále zahajuje uživatel, ale je možné je ukončit impulsem z dotekových kontaktů.

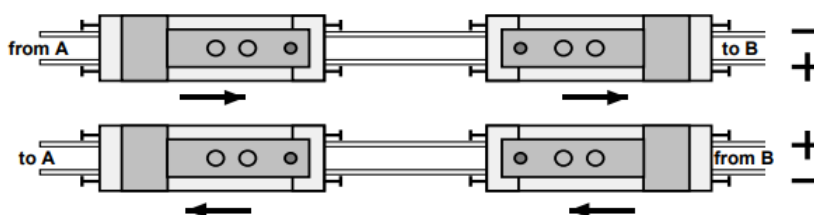
Automatické impulsní řízení už má plnou zpětnou vazbu v podobě programu na řídicí jednotce. Na tu uživatel zasílá impulsy pro zahájení vykonávání programu. Ten se následně řídí zcela na základě zpětné vazby z dotykových kontaktů na kolejích.[17, 19]



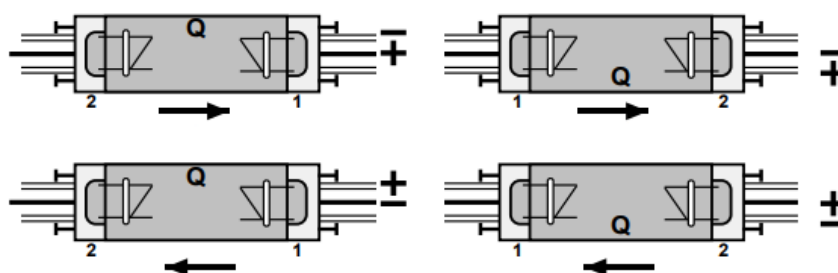
Obrázek 2.1: Analogový ovladač značky Hammant & Morgan schopný řízení až 5 vlaků. ¹

Norma NEM 631 a 640

Modelářská norma NEM 631 definuje způsob zapojení analogového kolejiště. Definuje orientaci vozidla na základě pozice komínu, označení „V“ nebo „1“ u přední kabiny. Směr provozu lze určit podle bodů trasy. Dále definuje směr provozu na trati podle polaritě kolejí jak pro dvoukolejný systém (2.2), tak i pro systém s trakčním vedením (2.3). [9]



Obrázek 2.2: Určení směru jízdy podle polaritě kolejí. ²



Obrázek 2.3: Určení směru jízdy podle polaritě trakčního vedení. ³

¹Převzato z <https://dccwiki.com/File:HMCotroller.jpg>

²Převzato z https://www.morop.org/images/NEM_register/NEM_E/nem631_en_1985.pdf

³Převzato z https://www.morop.org/images/NEM_register/NEM_E/nem631_en_1985.pdf

Norma NEM 640 definuje efektivní napájecí napětí jako 16 V střídavého napětí o frekvenci 60 Hertz. Směr jízdy vozidla je určen pouze směrovým spínačem, který vysílá pulz 24 V po dobu mezi 0,1 a 3 sekundami. Motory vozidel musí být nezávislé na polaritě napájecího napětí. [10]

Digitální ovládání

Tento způsob ovládání může působit náročněji na ovládání, ale přináší mnoho nových možností vůči analogovému ovládání. Systém digitálního ovládání se v základu skládá z několika prvků, mezi které patří napájecí zdroj, řídicí centrála, ovladač a lokomotivní dekodér. Dále může být systém rozšířen o digitálně řízené výhybky, dekodéry další prvků na trati či pomocné zesilovače. Základní výhodou digitálního ovládání je možnost ovládání celé trati a všech vozidel na ní najednou.[17, 19]

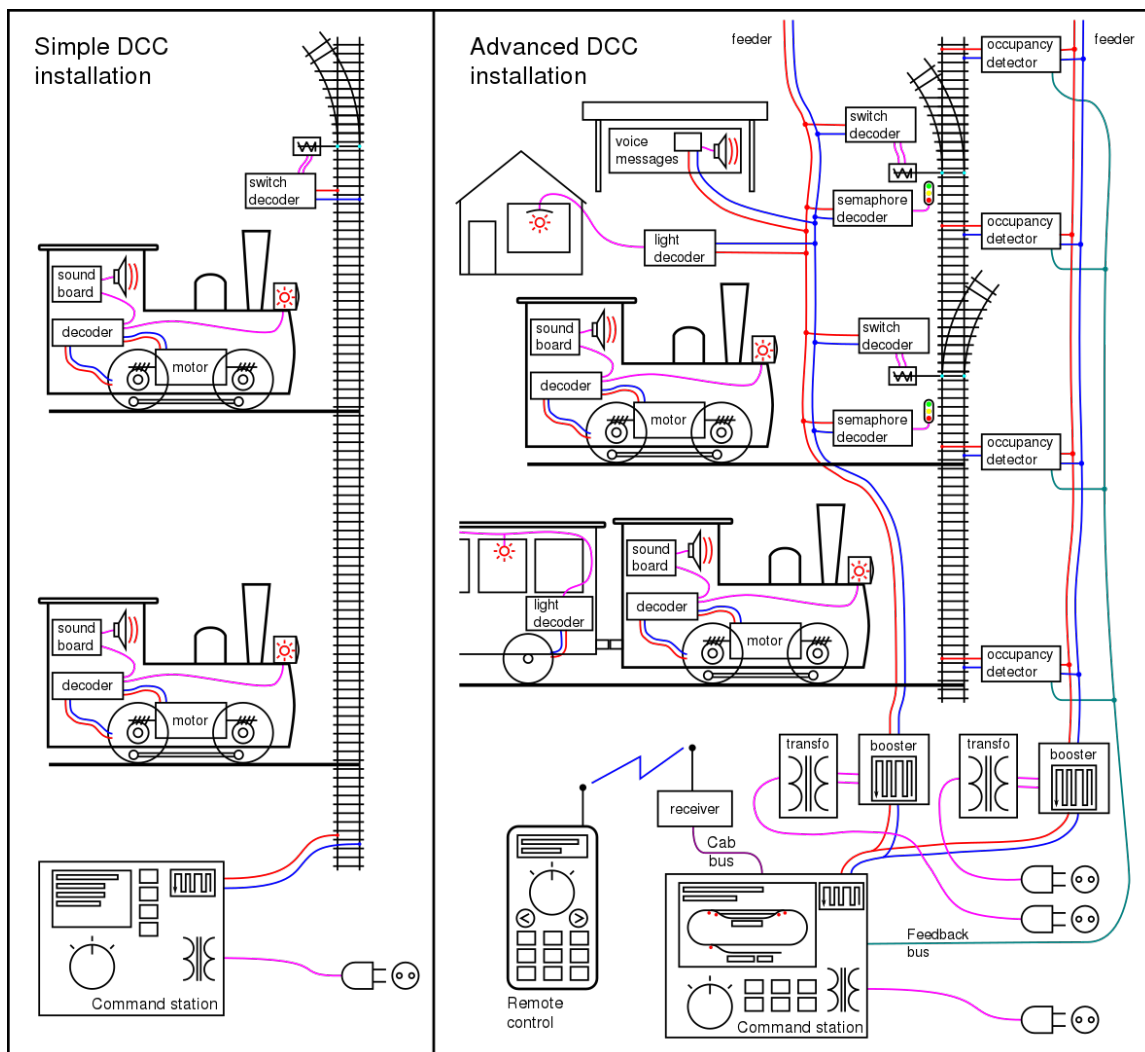
Standard Digital Command Control (DCC)

Standard DCC (2.4) je již velmi rozšířený v železničním modelářství. Díky zasílání paketů s informacemi je možné ovládat vlaky a jejich příslušenství. Mezi klíčové vlastnosti patří adresování, kdy každý aktivní prvek na trati má vlastní unikátní číselnou adresu, a tak je možné ovládat více vlaků najednou. Zároveň standard přináší obousměrnou komunikaci umožňující zpětnou vazbu od modelů. Tím, že informace o požadované rychlosti a směru jsou předávány pakety, může být napájecí napětí stabilní. Dále rozšiřuje možnosti o funkce, které kromě světel, které stejně jako rychlost nemusí být závislé na napájecím napětí, mohou nyní zahrnovat i zvukové módy. Standardizace zajišťuje kompatibilitu mezi výrobci a zároveň umožňuje digitalizaci analogových modelů, které jsou tomu přizpůsobeny. S použitím zesilovačů je možné takový digitální model škálovat do větších rozměrů (2.5). Standard spravuje organizace NMRA. [18]



Obrázek 2.4: Logo DCC. ⁴

⁴Převzato z <https://www.nmra.org/dcc-working-group>



Obrázek 2.5: Schéma DCC.⁵

2.2 Vývoj kolejí pro modelovou železnici

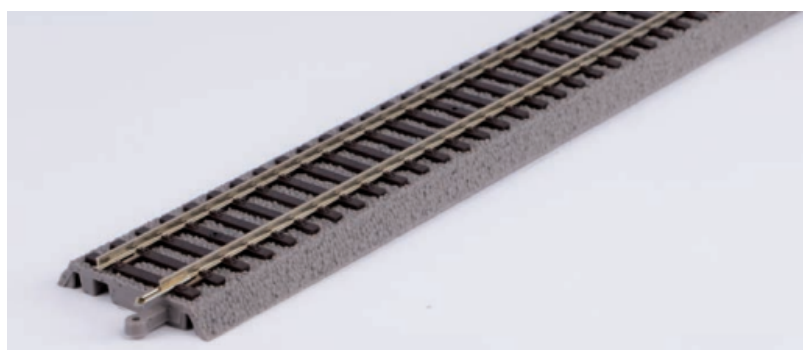
Úplně první modely vůbec nevyžadovaly k pohybu koleje, to ovšem zcela nenaplněovalo podstatu modelu. Kvůli omezeným technologiím výroby se první kolejnice vyráběly ze dřeva, to bylo později nahrazeno hlavně plechem, který byl ohýbán nebo lisován do tvaru obráceného „U“ nebo do tvaru, který odpovídal skutečným kolejnicím, avšak měl kulatou hlavu.

V dnešní době jsou již častější plné kolejnice lisované z drátu nebo obráběné do specifických rozměrů a tvarů profilů (2.8). Nejčastěji lze pořídit kolejnice s pražci, které vypadají jako dřevěné (2.6) nebo betonové, nebo dokonce i s plastovým podložím (2.7). [17, 19]

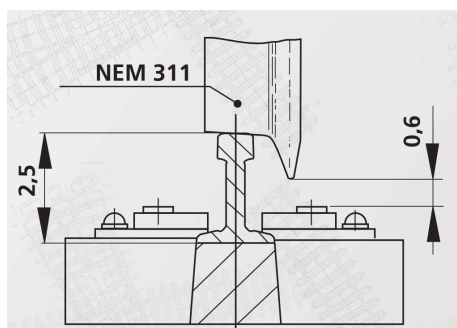
⁵Převzato z https://en.wikipedia.org/wiki/Digital_Command_Control



Obrázek 2.6: Profilové koleje s „dřevěnými“ pražci ⁶



Obrázek 2.7: Profilové koleje s podložím. ⁷



Obrázek 2.8: Parametry profilových kolejí dle normy NEM 311. ⁸

⁶Převzato z https://www.piko.de/DE/images/Prospekte-PDF/99556__A-Gleis_Proспект_2019.pdf

⁷Převzato z https://www.piko.de/DE/images/Prospekte-PDF/99556__A-Gleis_Proспект_2019.pdf

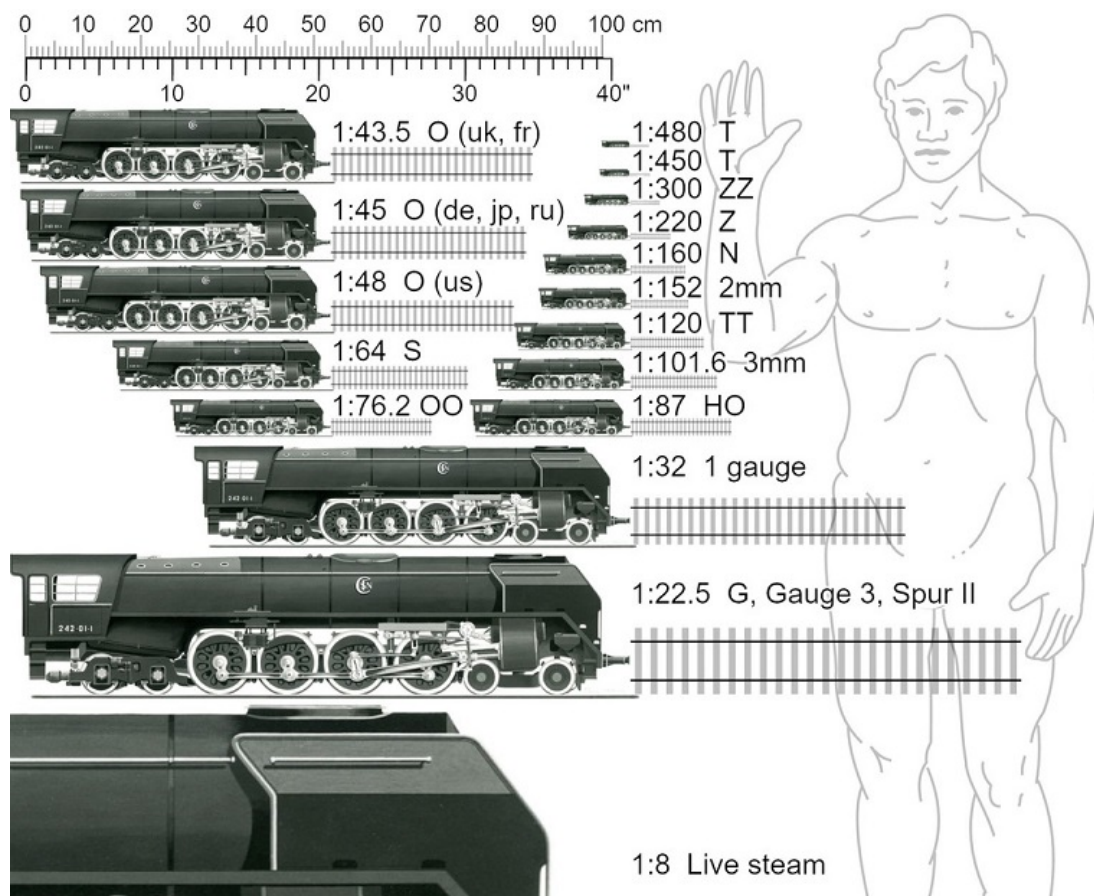
⁸Převzato z https://www.piko.de/DE/images/Prospekte-PDF/99556__A-Gleis_Proспект_2019.pdf

2.3 Rozchody kolejí a užívaná měřítka modelů

V rámci železničního modelářství existuje mnoho měřítek modelů a příslušných rozchodů kolejí. Rozhodnutí, který rozměr vybrat, často záleží na požadované rozloze modelu železnice a dostupném prostoru v místnosti, kde bude model umístěn. Dalším aspektem je množství detailů na modelech vlaků, náročnost manipulace s modely menších měřítek a tvorba okolní scenérie modelu železnice – čím menší, tím náročnější. Porovnání měřítek je možné vidět na obrázku 2.9.

Mezi prvními se používala velikost „I“ s rozchodem kolejí 45 milimetrů a měřítkem 1:32. Takový rozměr umožňoval využití skutečného parního pohonu. Dalším výrobcí oblíbeným rozměrem byla velikost „0“ s rozchodem 32 milimetrů a měřítkem 1:45 nebo 1:43,5. Dřívější měřítka byla často odvozena z obrysů a skříní železničních vozidel. Takto rozměrné modely umožňovaly zpracování do nejmenších detailů.

V současnosti patří mezi nejužívanější velikosti TT a H0. Velikost TT s rozchodem 12 milimetrů a měřítkem 1:120 je často volbou pokročilejších modelářů, jelikož práce s těmito modely vyžaduje značnou zručnost. Výhodou ovšem je, že zabírá méně prostoru, takže je vhodná například do bytu. Rozměr H0 je nejdostupnější na současném trhu s rozchodem 16,5 milimetru a měřítkem 1:87, kdy tato velikost stále poskytuje dostatek prostoru pro detaily, ale práce na modelu a okolní krajině je vhodná i pro začátečníky. [17, 19]



Obrázek 2.9: Velikosti modelů. ⁹

⁹Převzato z <https://midwestmodelrr.com/blog/model-train-scales-what-they-mean/>

Kapitola 3

Shrnutí současného stavu

Pro digitální ovládání modelové železnice díky rozšířenosti DCC protokolu existuje celá řada komerčně dostupných systémů řízení. Vzhledem k definici standardů a protokolu je lze i vzájemně kombinovat.

3.1 Komerční řešení

Již dostupná komerční řešení jsou často nevhodnější pro pokročilé modeláře, jelikož jsou připravené pro ovládání rozměrných modelů a jejich pořizovací cena je často vysoká.

Märklin & Cie GmbH - Model train Central Station 3 plus

Jedná se o více protokolový ovladač s vysokým rozlišením, moderní barevnou dotykovou obrazovkou, dvěma ovladači lokomotivy a také vestavěným centrálním panelem kolejového diagramu. Na lokomotivách lze ovládat až 32 funkcí a z příslušenství lze ovládat až 320 solenoidů Motorola nebo 2 048 DCC. Má vestavěné 2 čtečky lokomotivních karet (pro uložení údajů o lokomotivě na kartu lokomotivy nebo pro rychlé vyvolání lokomotivy zasunutím karty lokomotivy), slot na SD kartu pro rozšířené úložiště, vestavěný reproduktor pro přehrávání typických zvuků modelové železnice. Je zabudován výkonný zesilovač, který zajišťuje napájení pro vlaky a příslušenství (3.1). [11]



Obrázek 3.1: Ovladač Model train Central Station 3 plus. ¹

¹Převzato z

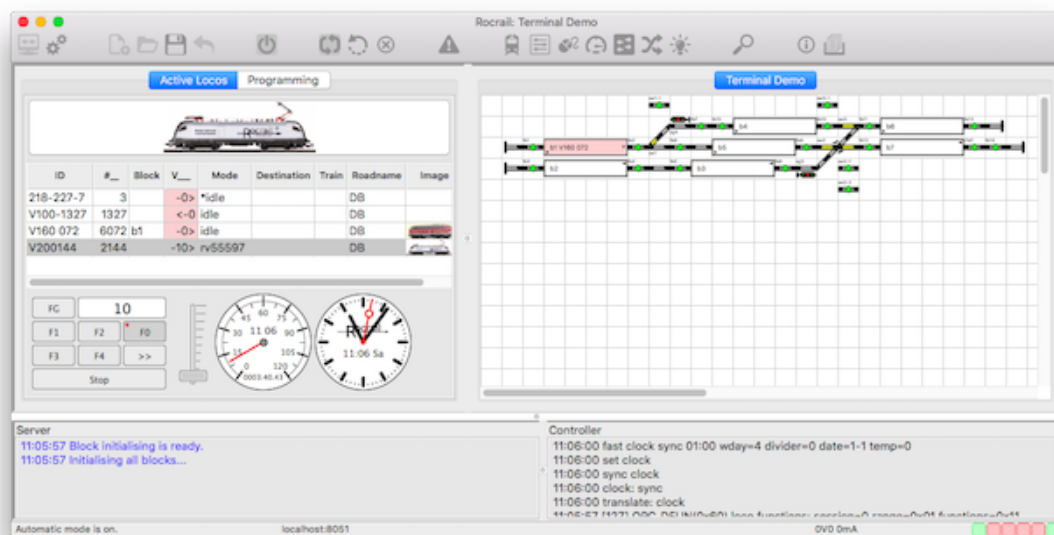
https://streaming.maerklin.de/public-media/anleitungen/CS3_Manual_EN_final-lo.pdf

Přehled vlastností:

- Protokoly mfx, mfx+, DCC, Motorola
- Barevný dotykový displej s vysokým rozlišením
- Vestavěná centrální řídicí deska schématu kolejí
- Pouzdro s centrálním tlačítkem stop a 2 otočnými knoflíky
- Až 32 říditelných funkcí lokomotivy
- 2 vestavěné čtečky lokomotivních karet
- Vestavěný reproduktor
- Vestavěný slot pro SD kartu
- Vestavěné, výkonné zesilovače
- Maximální výstup 5,0 A
- Řízení trasy (včetně řízení kyvadlové dopravy) [11]

Rocrail

Rocrail je volně dostupný software pro ovládání modelové železnice a jejího příslušenství. Vlaky mohou být řízeny buď ručně, nebo zcela řízeny softwarem v automatickém režimu. Smíšený provoz, tzn. je možné ovládat některé vlaky ručně a jiné vlaky v plně automatickém režimu. Je dostupný pro operační systémy Linux (Intel & ARM64), Apple macOS (Apple Silicon & Intel), Windows (Intel & ARM64) a PiOS (ARMHF & ARM64) (3.2). [16]

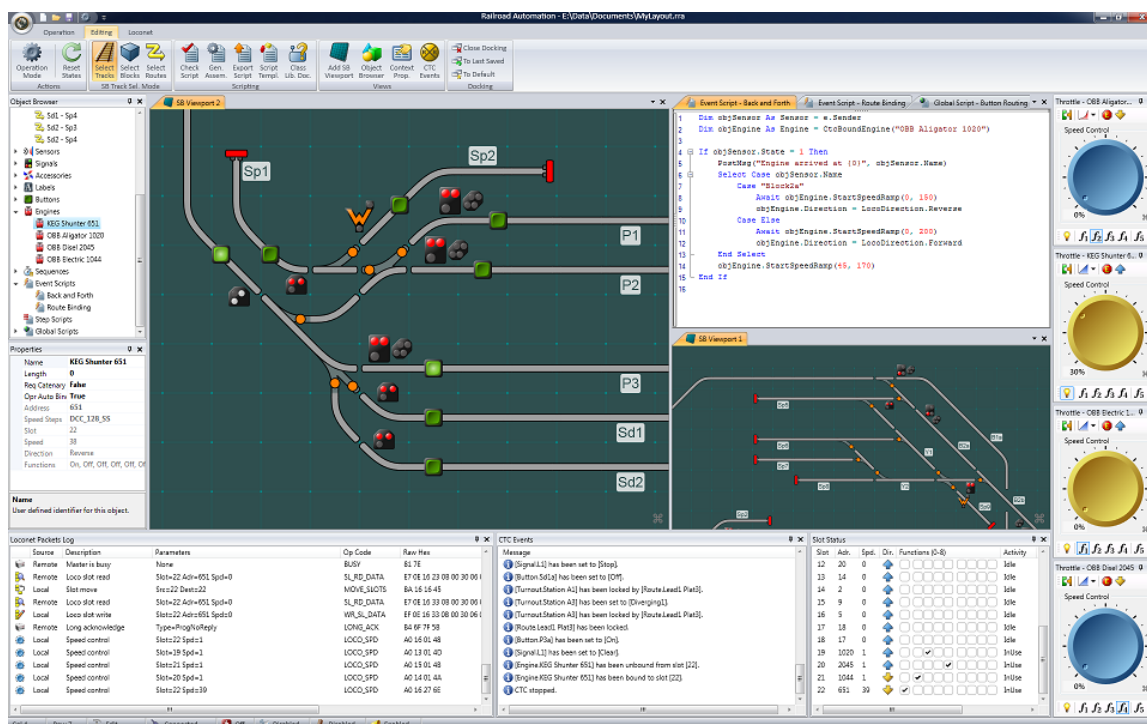


Obrázek 3.2: Grafické uživatelské rozhraní aplikace Rocrail. ²

²Převzato z <https://wiki.rocrail.net/doku.php?id=english#rocview>

Railroad Automation

Příkazový a řídicí software pro DCC modelové železnice s podporou síťové sběrnice Loconet. Běží nad platformou Microsoft .NET na operačních systémech Windows 7 a novějších. Umožňuje použití různých objektů, jako jsou koleje, výhybky, bloky, trasy, senzory, signály, příslušenství, štítky a tlačítka. Obsahuje také vektorový rozvaděč s více výřezy a podporuje vázání událostí a skriptování. Navíc umožňuje synchronizované softwarové/hardwarové ovládání více škrtkících klapek a dálkové ovládání škrtkící klapky přes HTTP pro moderní mobilní zařízení. Software má také ovládání rozpoznáváním řeči a režim simulace pro odpojené relace. Mezi další funkce patří nezpracované sekvenční nahrávání Loconet, inspektor slotů v reálném čase s čištěním, inspektor paketového provozu Loconet, plnohodnotný programátor LocoIO a programátor dekodéru DCC (3.3). [15]



Obrázek 3.3: Grafické uživatelské rozhraní aplikace Railroad Automation. ³

3.2 Open source řešení

EX-CommandStation

EX-CommandStation je DCC řídicí centrála, která je přímočará a robustní a lze ji sestavit pomocí běžně dostupných desek Arduino. Splňuje podstatnou část standardů NMRA Digital Command Control.

³Převzato z <https://www.perecli.com/rrauto/>

To umožňuje následující:

- Ovládání mnoha lokomotiv a jejich funkcí současně
- Regulační dekodéry příslušenství/funkcí
- Provádění programování na hlavní dráze
- Využití programovací koleje pro účely programování

Dále je uveden výběr z možného hardwaru. [8]

Arduino UNO REV3

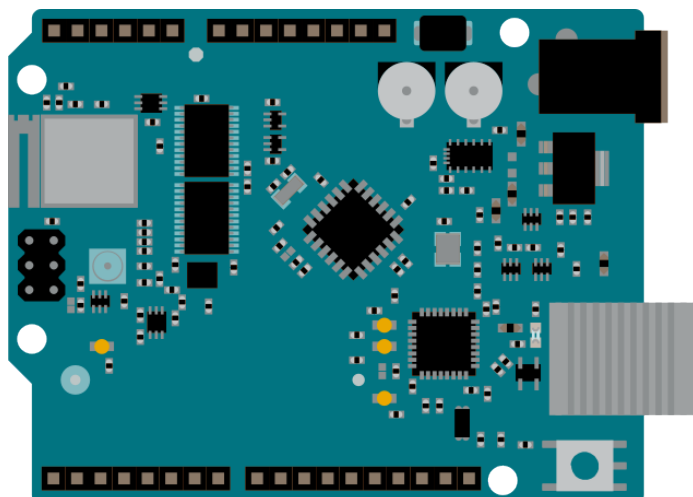
Deska mikrokontroléru založená na ATmega328P známá jako Arduino Uno má širokou škálu funkcí. Poskytuje 14 digitálních vstupních/výstupních pinů (z toho 6 s možností výstupu PWM), 6 analogových vstupů, keramický rezonátor 16 MHz (CSTCE16M0V53-R0), připojení USB, napájecí konektor, záhlaví ICSP a resetovací tlačítko. [2]

Arduino UNO WiFi REV2

Arduino deska UNO WiFi REV2 (3.4) funkčně vychází z desky Arduino UNO REV3. Oproti klasickému UNO ale přináší řadu vylepšení. Deska UNO WiFi REV2 je postavená na novém mikrokontroléru ATmega4809, má zabudovanou inerciální měřicí jednotku IMU a WiFi komunikaci zpracovává kryptovací akcelerátor ECC608. WiFi modul ESP32 u-blox NINA-W13 je systém na čipu SoC s integrovaným TCP/IP protokolem, který poskytuje připojení k WiFi síti nebo může fungovat jako WiFi přístupový bod. WiFi modul má zabudovanou WiFi anténu a komunikuje bezdrátově až na vzdálenost 400 m.

WiFi modul vyniká podporou zabezpečeného připojení, která ho předurčují pro IoT aplikace, kde je bezpečnost na prvním místě. Podpora AT příkazů je samozřejmostí. Arduino UNO WiFi REV2 má 14 digitálních vstupů/výstupů - 5 z nich můžete využít jako PWM výstupy (3.5). UNO WiFi REV2 má podobně jako klasické Arduino UNO také 6 analogových vstupů, USB konektor, napájecí a ICSP konektor a RESET tlačítko.

Jednoduché programování UNO WiFi REV2 zkrátka obsahuje vše potřebné pro řešení Arduino projektu s WiFi. Stačí desku připojit k počítači pomocí USB kabelu nebo napájet adaptérem a lze začít programovat pomocí Arduino IDE online nebo offline (tabulka 3.1). [3]



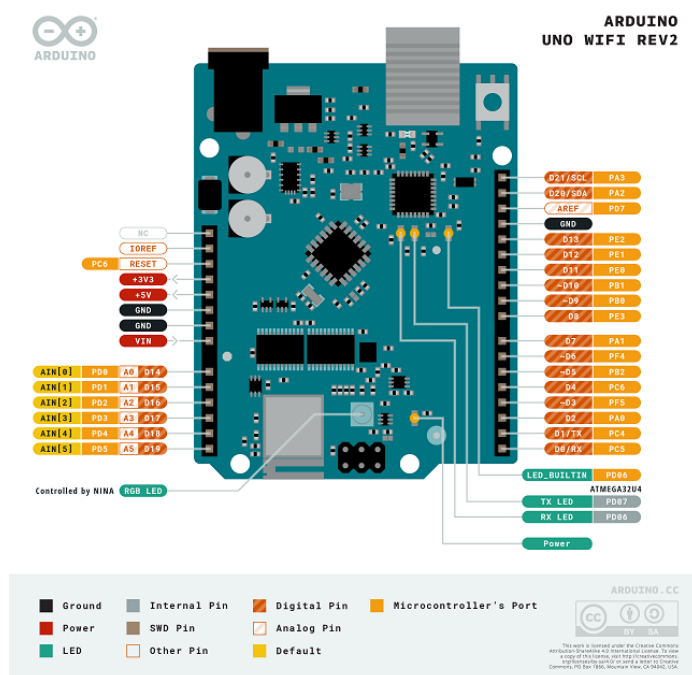
Obrázek 3.4: Deska Arduino UNO WiFi REV2. ⁴

Tabulka 3.1: Parametry Arduino UNO WiFi REV2⁵

Mikrokontrolér	ATmega4809
Provozní napětí	5V
Vstupní napětí (doporučené)	6 - 20V
Digitální I/O piny	14 — 5 poskytuje PWM výstup
PWM digitální I/O piny	5
Analogové vstupní piny	6
DC proud na I/O pin	20 mA
DC proud pro 3.3V pin	50 mA
Flash paměť	48 KB (ATmega4809)
SRAM	6,144 B (ATmega4809)
EEPROM	256 B (ATmega4809)
Clock Speed	16 MHz
Rádiový modul	u-blox NINA-W102
Bezpečnostní prvek	ATECC608A
Vnitřní měřicí jednotka	LSM6DS3TR
LED_BUILTIN	25
Délka	68.6 mm
Šířka	53.4 mm
Váha	25 g

⁴Převzato z <https://docs.arduino.cc/hardware/uno-wifi-rev2>

⁵Převzato z <https://docs.arduino.cc/hardware/uno-wifi-rev2>



Obrázek 3.5: Schéma zapojení vývodů Arduino UNO WiFi REV2. ⁶

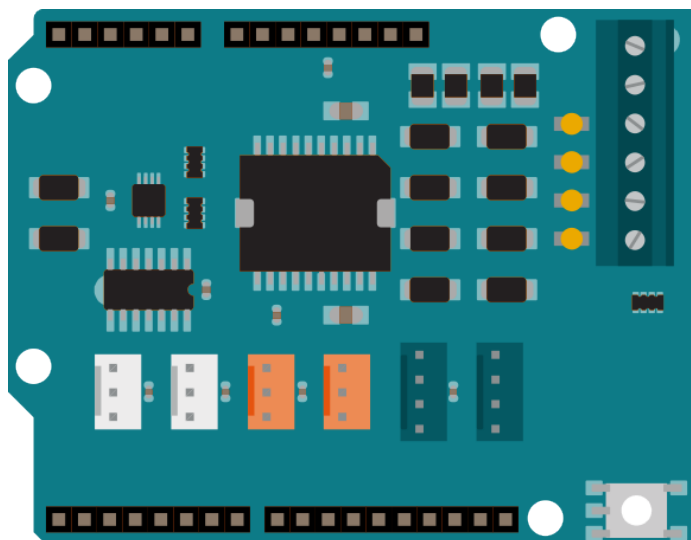
Elegoo Mega 2560 R3

Elegoo Mega 2560 R3 je deska mikrokontroléru, která využívá ATmega2560 a je speciálně určena pro složité projekty. Tato deska je dobře informovanou volbou pro činnosti, jako je 3D tisk a robotika, protože poskytuje 54 digitálních I/O pinů (z nichž 15 může fungovat jako PWM výstupy) a 16 analogových vstupů. [5]

Arduino Motor Shield Rev3

Arduino motor Shield Rev3 (3.6) je nejnovější produkt určený k pohonu indukčních zátěží, jako jsou relé, elektromagnety, stejnosměrné a krokové motory. Umožňuje řídit 2 stejnosměrné motory s deskou Arduino a nezávisle ovládat rychlost a směr každého z nich. Můžete také měřit proudovou absorpci jednotlivých motorů, mimo jiné. Je připraven k rozhraní SHIELD, což znamená, že lze rychle vytvářet projekty přímým zapojením modulů do desky (tabulka 3.2). [1]

⁶Převzato z https://content.arduino.cc/assets/Pinout-UNOwifirev2_latest.pdf



Obrázek 3.6: Deska Arduino Motor Shield Rev3. ⁷

Tabulka 3.2: Parametry Arduino Motor Shield Rev3⁸

Provozní napětí	5V až 12V
Ovladač motorů	L298P, řídí 2 DC motory nebo 1 krokový motor
Maximální proud	2A na kanál 4A maximálně (s přídatným zdrojem napájení)
Snímání proudu	1.65V/A
Funkce Free running stop and brake	

Deek-Robot Motor Shield

Funkčně řečeno je tento modul prakticky totožný s modulem Arduino, přičemž další výhodou je, že je často levnější. Dokáže zpracovat až 2A na kanál, ale může reálně poskytnout 1,3 nebo 1,5 A s instalací chladiče a chladicího ventilátoru. [6]

EX-Rail

EXtended Railroad Automation Instruction je jazyk, který umožňuje snadné vytváření sekvencí příkazových sekvencí, které se mají automaticky provádět nad rozvržením vašeho modelu železnice. Tyto sekvence jsou definovány pomocí jednoduchého souboru příkazového skriptu a nahrány do Command Station pro konfiguraci. Po spuštění se EX-Rail spustí automaticky na EX-CommandStation nebo může být aktivován ručně nebo v reakci na specifikované události. [7]

⁷Převzato z <https://docs.arduino.cc/hardware/motor-shield-rev3>

⁸Převzato z <https://docs.arduino.cc/hardware/motor-shield-rev3>

Kapitola 4

Návrh řešení

Při návrhu řešení je třeba hledět nejen na nejvhodnější technické a funkční parametry možných komponent, ale také jejich pořizovací cenu a výslednou technickou náročnost pro uživatele. To je důležité, jelikož cílem práce je řešení vhodně pro začátečníky a technicky méně zdatné uživatele modelové železnice.

4.1 Analýza současných řešení

Komerční řešení poskytují mnoho základních i pokročilých funkcí a jsou kompatibilní s širokou škálou konfigurací modelů železnic. Ale protože cílem této práce bylo vytvořit jednoduchou aplikaci pro nové či technicky méně zdatné uživatele, kteří s železničním modelářstvím teprve začínají, tak komerční řešení nebyla vhodnou volbou. Také by bylo velmi složité s komerčním softwarem propojit použité funkce počítačového vidění. Posledním negativním bodem je i vysoká pořizovací cena potřebného hardwaru.

Open source řešení EX-CommandStation spolu s EX-Rail se díky širší podpoře a pořizovací ceně kompatibilního HW jevílo jako lepší volba k realizaci. Propojení implementovaných funkcionalit by bylo mnohem jednodušší než u komerčních řešení a nabídlo by jednodušší rozšiřitelnost do budoucna. Avšak kvůli zpracování obrazu za účelem automatizace by bylo lepší využít pouze samotnou EX-CommandStation bez rozšíření Ex-Rail.

Byť jsou podporované možnosti HW velmi podobné, tak by dle mého názoru desky Arduino poskytly vyšší kvalitu, zaručenou kompatibilitu s Arduino IDE a za cenu mírně vyšší pořizovací ceny rychlejší doručení při pořizování, než desky jiných výrobců.

Návrh řešení je rozdělen do tří tematických oblastí, mechanická a elektrická konstrukce, software pro řízení modelu lokomotivy a software pro sledování modelu vlaku v kolejišti.

4.2 Mechanická a elektrická konstrukce – komponenty

Cílem je sestavit jednoduché kolejiště, model lokomotivy a elektroniky potřebné pro následné ovládání modelu.

Modelové kolejiště — zvoleny jsou profilové koleje s podložím z důvodu pevného mechanického a spolehlivého elektrického propojení jednotlivých kolejnic.

Lokomotiva — vzhledem k záměru použít ovládání lokomotivy DCC protokolem je zvolena lokomotiva „Taurus Railjet ČD“ s vestavěným DCC dekodérem.

Mikroprocesorová řídicí jednotka — pro řízení kolejiště je zvolena univerzální vývojová mikroprocesorová deska na bázi ARDUINO UNO, konkrétně typ ARDUINO UNO WiFi REV2. [3]

Výkonové analogové rozhraní — mezi mikroprocesorovou řídicí jednotku a kolejiště je potřeba zařadit výkonové analogové rozhraní. Pro uvedenou aplikaci postačí „Arduino Motor Shield Rev3“. [1]

Kamera — USB kamera s HD rozlišením bude umístěna nad kolejištěm tak, aby záběr mohl být použit ke sledování vlaku na kolejišti.

PC — pro běh jednotlivých aplikací bude použito PC případně Notebook. Propojení s mikroprocesorovou jednotkou a kamerou bude realizováno prostřednictvím USB rozhraní.

4.3 Software – použité komponenty a knihovny

Bude zvoleno digitální ovládání modelové železnice. Vzhledem k záměru připravit řešení, které bude možno v budoucnu upravovat a rozšiřovat budou preferovány otevřené standardy. Pro digitální ovládání je nejrozšířenější protokol DCC. Bude implementován do mikroprocesorové jednotky, která ovládá kolejiště.

Uživateli bude dostupná aplikace běžící na PC a komunikující s mikroprocesorovou jednotkou. SW umožní základní ovládání směru a rychlosti jízdy vlaku. Sledování polohy vlaku bude řešeno analýzou obrazu z kamery nad kolejištěm. Bude použito polně dostupných řešení pro počítačové vidění.

Sledování polohy vlaku bude řešeno analýzou obrazu z kamery nad kolejištěm. Bude použito polně dostupných řešení pro počítačové vidění. V rámci této práce bude řešen nástroj pro základní sledování polohy jako základu pro další rozšíření v budoucnu.

Vývojové prostředí

Vývoj aplikace je řešen v jazyce Python na prostředí MS Windows. Jako vývojové prostředí je použit PyCharm. V řešení jsou použity knihovny uvedené v tabulce 4.1.

Tabulka 4.1: Verze knihoven

Python	python-3.11.3-amd64
PyCharm	pycharm-community-2023.1
OpenCV on Wheels	opencv-contrib-python 4.5.5.62
PySerial	pyserial 3.5
NumPy	numpy 1.24.2
IMUtils	imutils 0.5.4
Pillow	pillow 9.5.0

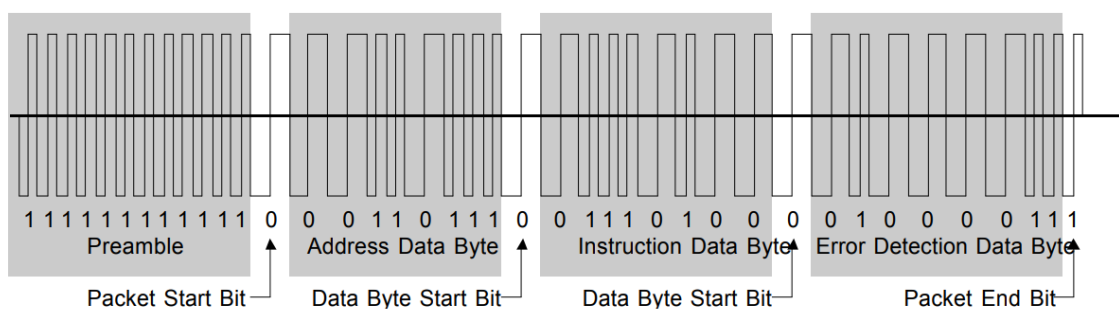
Komunikační Protokol - Digital Command Control (DCC)

Jak již bylo dříve zmíněno protokol DCC je definován „Digital Command Control Working group of the National Model Railroad Association (NMRA)“. Popis jednotlivých vrstev protokolu je dostupný na stránkách NRMA¹.

¹www.nmra.org

DCC pakety jsou způsob, jak přenášet informace o řízení vlaků a příslušenství pomocí digitálního signálu v DCC systému. Tyto pakety obsahují informace o rychlosti, směru, světle, zvukových efektech a dalších funkcích, které jsou potřebné pro řízení vlaku.

Paket se skládá z preamble, která obsahuje řadu bitů o hodnotě „1“. Řada musí být o délce 10, 12 nebo 14 v závislosti na použití. Následuje počáteční bit, který odděluje preamble a adresový byte, jeho hodnota je „0“. Adresový byte je první datový byte paketu, kde první přenesený bit je ten nejvýznamnější. Další počáteční bit odděluje adresový byte od datového bytu. Datový byte obsahuje informace o adrese, instrukci nebo chybě. Stejně jako u adresového bytu je první přenesený bit ten nejvýznamnější. Celý paket je zakončen ukončovacím bitem o hodnotě „1“. Příklad je možné vidět na obrázku 4.1. [12]



Obrázek 4.1: Příklad struktury přenášeného paketu. ²

Každý DCC balíček obsahuje několik informačních polí. První pole obsahuje informaci o adrese vlaku, tzn. identifikátor, který vlak identifikuje na trati. V DCC systému je k dispozici celkem 9999 adres pro vlaky, a každý balíček musí obsahovat adresu, pro kterou je určen.

Další pole obsahuje informaci o rychlosti vlaku. Každý vlak v DCC systému má 127 úrovní rychlosti, od 0 (zastavení) do 126 (maximální rychlost). Rychlost se může měnit pomocí řídicích prvků na řídicím panelu nebo pomocí počítačového software.

Další pole obsahuje informaci o směru jízdy vlaku. Toto pole určuje, zda bude vlak jezdit vpřed nebo vzad. Další pole obsahuje informace o funkcích vlaku, např. zapnutí a vypnutí světel, píšťaly, zvukových efektů apod. Každá funkce je přiřazena k určitému číslu, a vlak může mít až 28 funkcí.

Každý DCC balíček obsahuje také kontrolní součet, který se používá pro ověření správnosti přenosu dat. Kontrolní součet zabraňuje chybám v přenosu dat a zajišťuje spolehlivost DCC systému.

DCC pakety jsou vysílány po kolejích a jsou přijímány dekodérem vlaku, který tyto informace interpretuje a řídí vlak a jeho funkce. Díky DCC paketům je možné řídit více vlaků na trati současně a snadno programovat a konfigurovat funkce vlaků.

Celkově lze říci, že DCC pakety jsou klíčovou součástí DCC systému a umožňují snadné a přesné řízení modelové železnice. [12]

²Převzato z <https://www.nmra.org/sites/default/files/s-92-2004-07.pdf>

Knihovna DCC++

DCC++ je knihovna pro Arduino, která umožňuje vytvořit DCC řídicí systém pro modelové železnice. Tato knihovna byla vyvinuta pro Arduino UNO a Arduino MEGA a umožňuje ovládat vlaky, výhybky, semaforey a další funkce modelové železnice pomocí digitálních signálů.

DCC++ knihovna je velmi flexibilní a umožňuje přizpůsobit řídicí systém potřebám a požadavkům modeláře. Knihovna podporuje mnoho různých typů výhybek a semaforů, umožňuje vytvářet různé scénáře a programovat přepínání funkcí na výhybkách. DCC++ také umožňuje ovládat modelovou železnici pomocí mobilních zařízení s pomocí aplikace.

DCC++ knihovna byla vyvinuta jako open source projekt a je k dispozici zdarma ke stažení a použití na GitHubu. Knihovna je dobře zdokumentovaná a obsahuje návody a příklady pro použití. [4]

Knihovna OpenCV

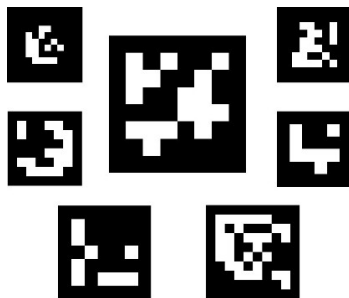
Knihovna původně vyvinuta společností Intel, je nyní open source a obsahuje mnoho nástrojů pro zpracování obrazu a počítačové vidění. Je k dispozici pro řadu programovacích jazyků jako je C++, Python či Java a je podporována na platformách Windows, Linux, OS X, Android a iOS.

Pro jazyk Python je k dispozici API, které spojuje vlastnosti programovacího jazyka a nástrojů knihovny OpenCV. Základem je knihovna Numpy, která dále se svými operacemi a datovými strukturami umožňuje optimální použití dalších knihoven jazyka Python jako jsou například Matplotlib a SciPy. [14]

ArUco Marker

Modul Aruco knihovny OpenCV je specializovaný modul pro detekci a rozpoznávání ArUco markerů (4.2) v obraze. ArUco markery jsou čtvercové značky, které obsahují unikátní identifikátor a jsou často používány v počítačovém vidění pro kalibraci kamer, sledování objektů a vizuální navigaci. Modul Aruco v OpenCV umožňuje snadno detekovat a rozpoznávat tyto markery v reálném čase pomocí kamery.

Modul Aruco v OpenCV obsahuje funkce pro detekci ArUco markerů, dekodování identifikátorů a odhadování pozice a orientace markerů. Modul také umožňuje vytváření vlastních markerů a kalibraci kamery pro vylepšení přesnosti detekce.



Obrázek 4.2: ArUco značky. ³

³Převzato z https://docs.opencv.org/4.x/d5/dae/tutorial_aruco_detection.html

Modul Aruco v OpenCV je užitečným nástrojem pro aplikace v oblasti rozšířené reality, počítačového vidění, robotiky a mnoha dalších oblastí. Díky jednoduchému použití a vynikající výkonnosti je modul Aruco oblíbeným nástrojem pro detekci a rozpoznávání ArUco markerů.

Značky Aruco jsou typem 2D čárového kódu, který se běžně používá v aplikacích počítačového vidění pro úkoly, jako je kalibrace fotoaparátu nebo kamery, sledování objektů a rozšířená realita. V knihovně OpenCV proces vytváření Aruco značky zahrnuje několik kroků:

1. Definice velikost značky a slovníku – velikost značky Aruco je definována zadáním počtu bitů podél každé hrany. Slovník je předdefinovaná sada kódů značek a používá se ke generování jedinečných značek. OpenCV poskytuje několik slovníků, například `DICT_4X4_50`, `DICT_5X5_100` a `DICT_6X6_250`.
2. Generování kódu značky – Kód značky je jedinečný binární vzor, který se používá k identifikaci značky. Kód je generován pomocí algoritmu Hammingova kódu, který zajišťuje, že každý kód má minimální vzdálenost 2, což znamená, že se libovolné dva kódy liší alespoň o dva bity. Tato vlastnost umožňuje robustní detekci a rozpoznávání značek.
3. Generování obrázku značky – Jakmile je kód značky vygenerován, použije se k vytvoření obrázku značky. Obrázek je černobílá reprezentace kódu značky, kde černá představuje 0 a bílá představuje 1. Obrázek je generován pomocí šablony, která definuje tvar značky a polohu kódu v rámci značky.
4. Přidání ohraničení a identifikátoru – Obrázek značky Aruco má vnější ohraničení, které pomáhá při detekci značky v obrázku. Také je přidán identifikátor markeru, což je jedinečné číslo pro identifikaci markeru.
5. Tisk nebo uložení značky – posledním krokem je tisk nebo uložení obrázku značky Aruco. Značku lze tisknout na papír nebo jiné materiály, jako jsou samolepky nebo plasty, pro použití ve fyzickém prostředí. Alternativně lze značku uložit jako soubor obrázku pro použití v digitálním prostředí. [13]

4.4 Návrh grafického uživatelského rozhraní

V této části je třeba nahlížet na to, kdo bude cílový uživatel aplikace. Jelikož bude aplikace určena pro začátečníky a technicky méně zdatné uživatele, je třeba dobře vyvážit jednoduché ovládání aplikace spolu s obsažením veškerých nabízených funkcionalit. Okno grafického uživatelského rozhraní by nemělo pro přehlednost vyžadovat zobrazení na celou obrazovku.

Nejvhodnější proto bude GUI rozdělit na pomyslné dvě poloviny, kde levá polovina bude převážně obsahovat aktivní ovládací prvky a pravá strana pasivní informativní prvky.

Dále je třeba rozdělit část určenou pro ovládání na část pro manuální ovládání a část pro automatické řízení. Tyto dvě části by nemělo být možné užívat současně.

Co se týče informativní části, tak bude obsahovat video a stav sektorů. Oba prvky by měly být dostatečně velké, tak aby se v nich uživatel mohl snadno orientovat a nebylo obtížné z nich vyčíst informace. Priorita by přitom ale měla padnout na zobrazení videa.

Kapitola 5

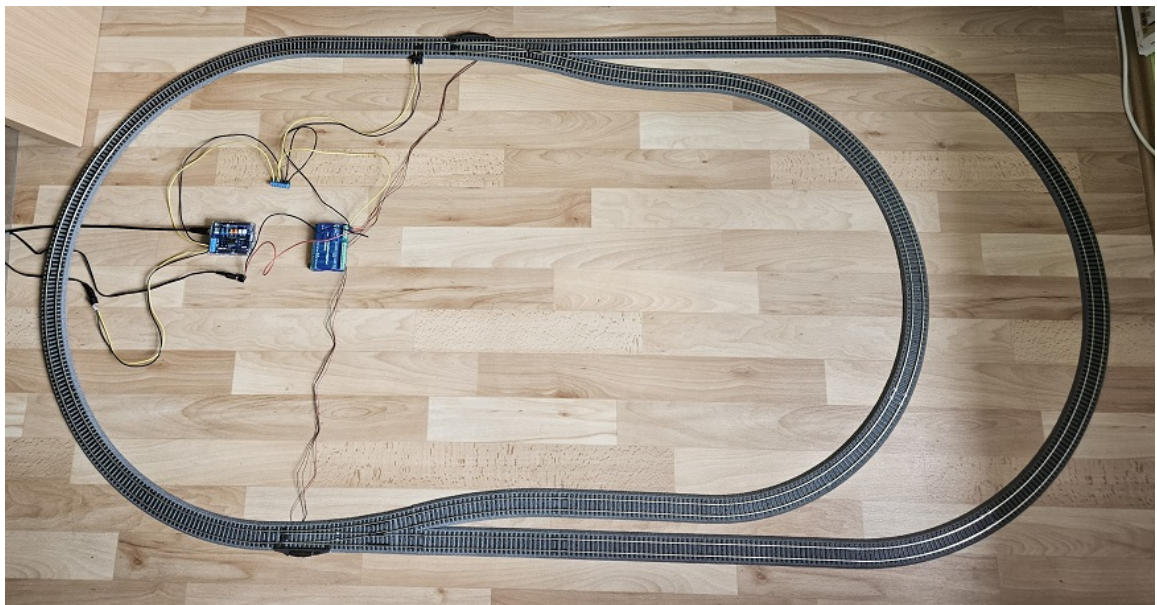
Realizace

Největší část realizace tvoří uživatelská aplikace obsahující jak grafické uživatelské rozhraní, tak funkční logiku. I když je hardware řešen jednodušeji a je složen z komerčně dostupných částí, nebyly by bez něj vývoj a testování softwarové části možné.

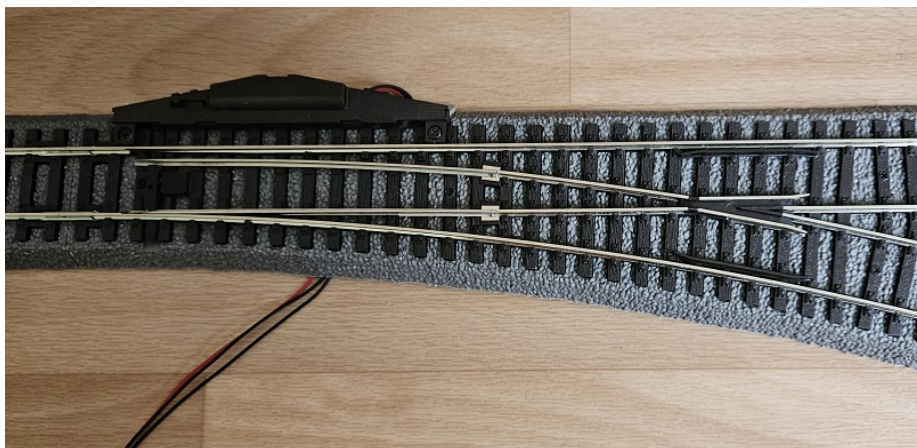
5.1 Hardware

Kolejiště

V rámci realizace jsou použity dvě kompatibilní sady A a C kolejí A-Gleis značky PIKO (5.1). Jedná se o profilové koleje s podloží a plán kolejiště je mírně upraven oproti doporučení od výrobce kvůli lepšímu rozdělení dvoukolejného oblouku do samostatných úseků. Dvě přítomné výhybky jsou osazeny elektromagnetickými přestavníky značky PIKO, které mají 3 vývody pro ovládání (5.2).



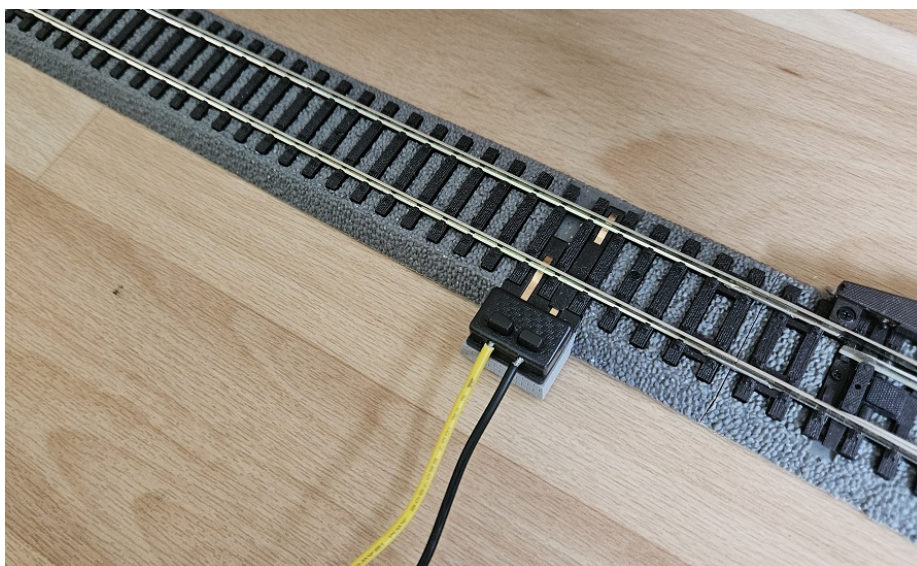
Obrázek 5.1: Kolejiště



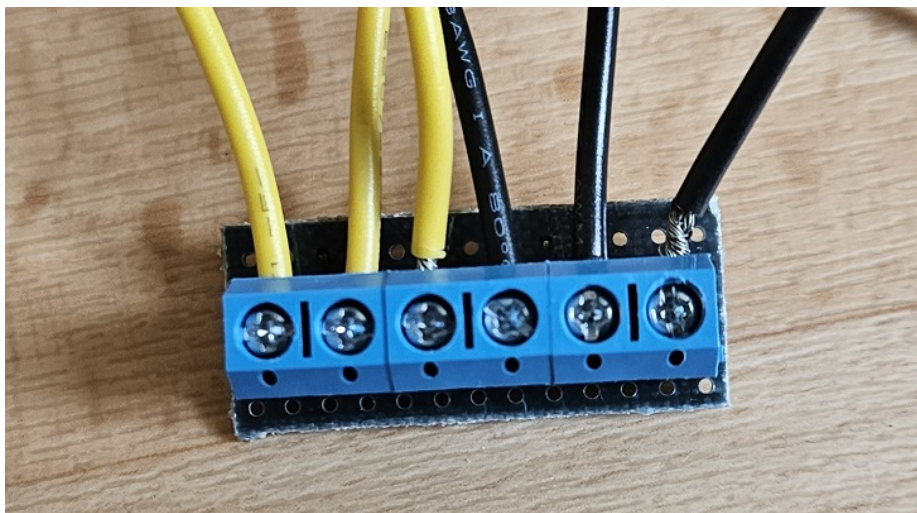
Obrázek 5.2: Napojení výhybky

Řídicí centrála

Za tímto účelem je využita deska Arduino Uno WiFi Rev2 společně s modulem Arduino Motor Shield. Samotná mikroprocesorová jednotka je napájena z počítače přes USB kabel, který zároveň slouží k sériové komunikaci. Je na ní nainstalovaný EX-CommandStation software, který se spustí ihned po připojení napájení. Motor Shield má svůj vlastní napájecí zdroj 12V DC a dvoupólový výstup pro napájení modelu trati a zaslání DCC informací. Tento výstup vede to rozdělovací svorkovnice (5.4), kde se dělí mezi napájecí konektor trati (5.3) a dekodér výhybek (5.5).



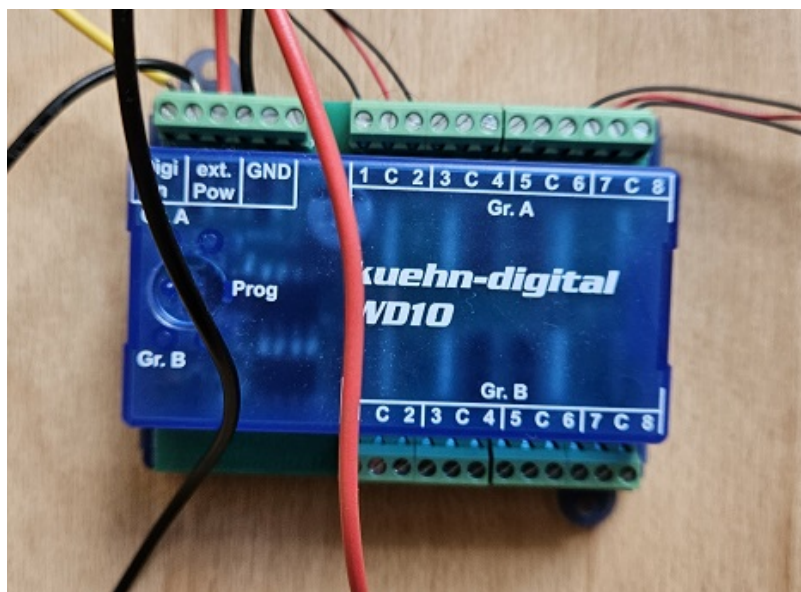
Obrázek 5.3: Napájení kolejiště



Obrázek 5.4: Svorkovnice

Dekodér výhybek

Univerzální dekodér příslušenství Kuehn-digital WD10 (5.5) je připojen přes rozdělovací svorkovnici k napájecímu a komunikačnímu výstup řídicí jednotky, ale využívá jej pouze k příjmu DCC informací. K napájení je využit externí adaptér 14V AC. Vývody z výhybek jsou připojeny na první a čtvrtou skupinu výstupů skupiny A. Toto zapojení definuje lineární adresy výhybek jako 1 a 4.



Obrázek 5.5: Výhybkový dekodér

Modely vlaků

Za účelem implementace automatického provozu více vlaků najednou jsou použity dva modely vlaků. První je lokomotiva Taurus s dvěma vagóny, která je již z výroby vybavená DCC dekodérem a zvukovým modulem. Druhý je model diesellové lokomotivy BR 218, který je dodatečně osazen DCC dekodérem a neobsahuje zvukový modul. Tato lokomotiva nemá zapojené žádné vagóny.

Oba vlaky jsou na střechách osazeny značkami Aruco. Každá lokomotiva i vagón mají 3 značky, na každém konci jednu a uprostřed jednu otočenou od 45 stupňů za účelem lepšího rozpoznání v obloucích.

Kamera

Pro snímání obrazu je použita kamera v rozlišení Full HD 1080p. Je zvolen produkt společnosti Logitech C920 HD PRO WEBCAM (5.6). Parametry kamery:

- Maximální rozlišení 1080 p / 30 fps – 720p / 30 fps
- Camera mega pixel - 3
- Typ zaostření - Funkce autofocus
- Typ objektivu - skleněný
- Diagonální zorné pole (dFoV) - 78°



Obrázek 5.6: Kamera C920 ¹

¹Převzato z

<https://support.logi.com/hc/cs/articles/360023307294-C920-Technical-Specifications>

5.2 Grafické uživatelské rozhraní

Grafické uživatelské rozhraní je implementováno za použití knihoven Tkinter a Pillow. Až na výjimky několika dočasných oken pro nastavení kamery či výběr souboru pro automatizaci provozu, je celé grafické uživatelské rozhraní dostupné v jednom hlavním okně. V rámci tohoto okna se dále dělí na několik funkcionálně odlišných oddílů a pododdílů. Hlavní dva oddíly jsou ovládací panel a monitorovací panel. Nad těmito oddíly je také horní lišta pro důležité ovládací prvky celého systému.

Ovládací panel

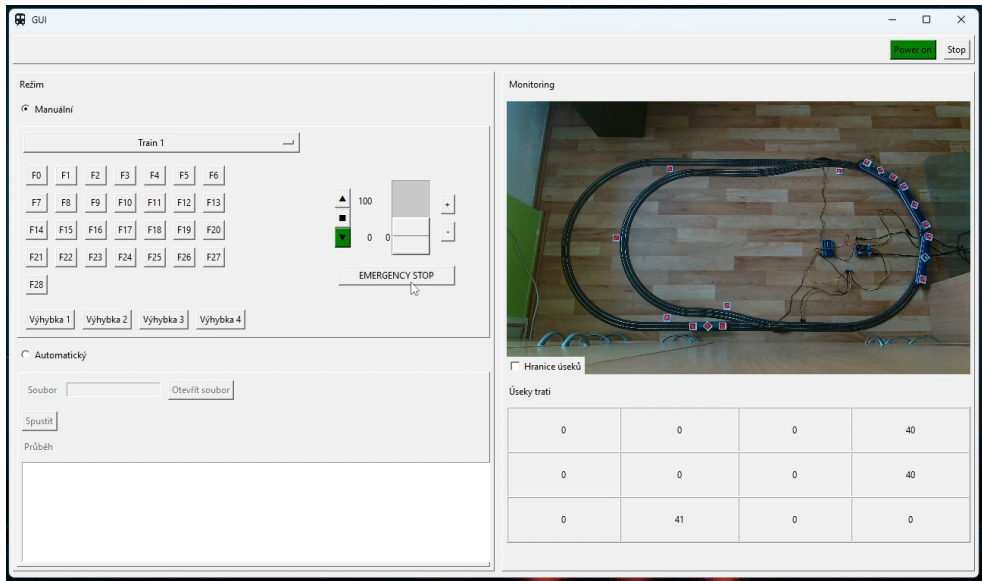
Tento panel zaujímá levou polovinu okna grafického uživatelského rozhraní. Slouží hlavně pro ovládání modelů vlaků a výhybek na trati. Obsahuje dva na sobě nezávislé pododdíly, jeden pro manuální ovládání a druhý pro automatické ovládání. Mezi panely lze přepínat pomocí přepínačů, které umožňují vždy využívat pouze jeden ze dvou ovládacích režimů.

Manuální ovládací panel

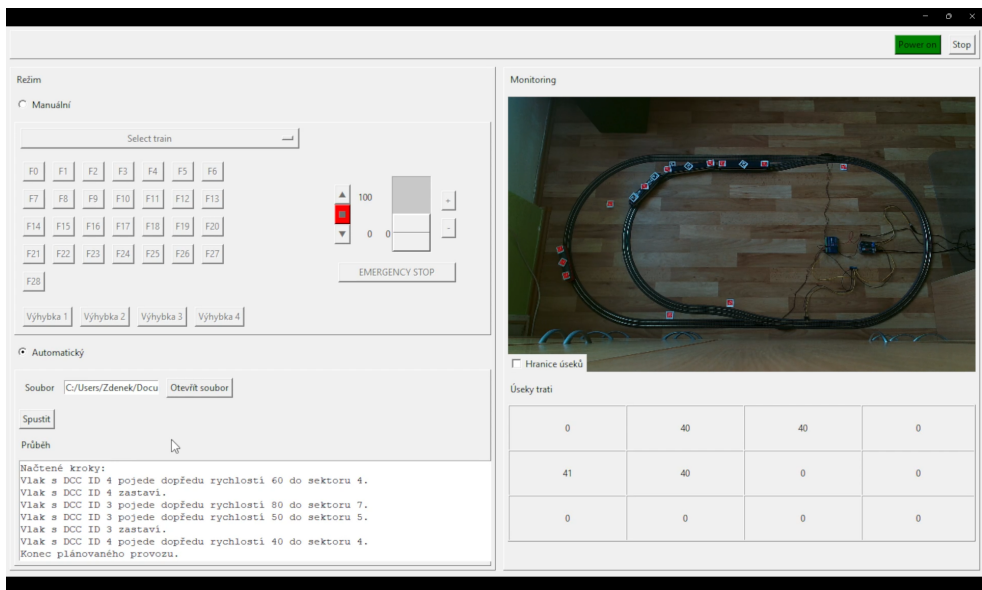
Ovládací prvky v tomto panelu umožňují ovládání vybraného modelu vlaku a všech výhybek. Rozevírací výběr možností obsahuje seznam všech definovaných vlaků. Po spuštění není vybrán žádný vlak. Nastavené ovládací parametry si jednotlivé lokomotivy zachovávají i po výběrů jiného vlaku, avšak nastavení v rámci ovládacích prvků se nezachová. Hlavními manuálními ovládacími prvky je posuvník a tlačítka rychlosti neboli regulátor tahu, 3 tlačítka voliče směru jízdy a tlačítka nouzové brzdy. Nastavit rychlost lze dvěma způsoby – posuvníkem, který umožňuje rychlejší a plynulejší nastavení větších změn v rychlosti modelu vlaku, a tlačítka plus a mínus, které mění rychlost po jednotkách rychlosti pro přesnější nastavení konkrétní rychlosti. Rozsah rychlosti je od 0 do 100, tudíž má posuvník 100 možných poloh. Při nastavení rychlosti na 0 dochází k postupnému zastavení modelu vlaku. Vlastnosti tohoto postupného zastavení určuje samotný dekodér modelu vlaku. Tlačítka voliče směru nastavují jízdu vpřed, vzad a zastaveno. Je možné vybrat pouze jednu z těchto možností. Během nastavení voliče na zastaveno nemá změna rychlosti žádný účinek. Zároveň tato volba zastaví model vlaku, pokud je v pohybu. Tlačítka nouzové brzdy model vlaku zastaví okamžitě. Délka brzděné dráhy nouzového zastavení je závislá na vlastnostech elektrických motorů modelu lokomotivy. Dalšími prvky jsou přepínací tlačítka funkcí modelu lokomotivy a přepínací tlačítka výhybek modelu trati. Tlačítek funkcí je 29 a jsou označeny F0 až F28. Tlačítka pro ovládání výhybek jsou 4 (5.7).

Automatický ovládací panel

Hlavní funkcí tohoto panelu je spuštění automatického provozu na modelu železnice. Tlačítka Otevřít soubor spustí dialogové okno pro výběr vstupního souboru CSV s jednotlivými kroky. Po načtení vstupního souboru je jeho absolutní cesta zobrazena v textovém řádku vedle tlačítka a do textového bloku ve spodní polovině tohoto panelu jsou vypsané jednotlivé kroky a jejich celkový počet. Tlačítkem Start se spustí provádění jednotlivých kroků automatizace a aktuálně prováděné kroky jsou vypisovány do textového bloku (5.8).



Obrázek 5.7: Manuální ovládání



Obrázek 5.8: Automatické ovládání

Monitorovací panel

Panel zaujímající pravou polovinu grafického uživatelského rozhraní má převážně účel sdělovat uživateli informace. Hlavními prvky je zobrazení kamery a vizualizace úseků modelu trati.

Zobrazení kamery

Nachází se v horní polovině monitorovacího panelu a zobrazuje zpracované video z kamery. Video má rozlišení 640 pixelů na šířku a 360 pixelů na výšku, obsahuje zvýraznění ArUco značek a pokud je uživatelem povolena, tak i mřížku úseků. Lišta pod videem slouží pro dva účely. Zobrazují se zde informace o stavu kamery při inicializaci a nastavování a také je zde tlačítko pro dokončení nastavení kamery.

Vizualizace úseků

Ve spodní polovině monitorovacího panelu je tabulka, jejíž počet řádků a sloupců odpovídá počtu řádků a sloupců mřížky úseků na modelu železnice. Pro lepší čitelnost dat mají buňky tabulky stejnou výšku i šířku, které však nemusí odpovídat poměru velikostí úseků ve skutečnosti. Jednotlivé buňky obsahují identifikátor vlaku nebo 0 pokud není žádný vlak v úseku. Pokud je detekována možná kolize, tak se pozadí buňky, která tuto kolizi detekovala, změní na červenou barvu.

Horní lišta

Zahrnuje dvě tlačítka pro důležité systémové funkce. Tlačítko napájení slouží k zapínání a vypínání napájení modelu železnice z modulu Arduino Motor Shield. Tlačítko má zelené pozadí při zapnutém napájení a červené pozadí při vypnutém napájení. Druhé tlačítko Stop zastaví veškerou aktivitu na modelové železnici, kdy všem definovaným lokomotivám zašle příkaz k nouzovému zastavení a po krátké prodlevě vypne napájení.

5.3 Sériová komunikace

Obsahuje funkce pro výměnu dat s mikroprocesorovou jednotkou prostřednictvím virtuálního sériového portu. Veškerá funkcionalita je definována třídou SerialConnection s využitím balíčku pySerial a modulu queue. Při inicializaci objektu této třídy je pomocí pokusů o připojení k celému rozsahu virtuálních sériových portů COM sestaven seznam těch, které jsou k dispozici. Tento seznam je později použit k vytvoření nabídky COM portů uživateli. Instance disponuje proměnnými pro předávání čísla vybraného COM portu, instance otevřené sériové komunikace, identifikátoru modelu vlaku, hodnoty rychlosti a nastavení směru. Pro předávání hodnot stavu napájení modelu železnice a stavu funkcí modelu lokomotivy jsou použity fronty. Tento způsob zjednodušuje a zabezpečuje předávání dat mezi vlákny procesů. Při zavolání funkce připojení dochází k navázání sériové komunikace na vybraném portu s přenosovou rychlostí 115 200 Baud. Po vytvoření spojení dochází ke čtení úvodní zprávy z mikroprocesorové jednotky. Poté je sériové spojení připraveno k běžné komunikaci. Komunikace probíhá pomocí paketů definovaných knihovnou DCC++. Využity jsou 4 typy paketů, a to pakety pro spínání výstupů napájení z Arduino Motorshield modulu, T pakety pro ovládání rychlosti a směru modelu lokomotivy, F pakety pro spínání funkcí modelů lokomotiv a A pakety pro ovládání výhybek na modelu trati. Pro vytváření jednotlivých

paketů jsou definovány funkce, které výsledný paket ukládají do fronty k odeslání. Obecné formátování paketů odděluje jednotlivé položky mezerou a celý paket ohraničuje ostrými závorkami.

Paket napájení

Obsahuje dvě informace pro mikroprocesorovou jednotku. Na první pozici je bit reprezentující stav výstupu napájení, 0 pro stav vypnuto a 1 pro stav zapnuto. Na druhé pozici je identifikátor výstupu, přičemž jsou k dispozici dva výstupy napájení, jeden pro hlavní trať, identifikovaný řetězcem MAIN a druhý pro programovací kolej s identifikátorem PROG. V rámci realizace je využit pouze výstup pro hlavní trať, proto paket má pouze dvě podoby - <0 MAIN> a <1 MAIN>. K dispozici je také identifikátor JOIN, který spojí hlavní trať a programovací kolej v jednu hlavní trať, ten ovšem není pro tuto realizaci použit z důvodu absence využití programovací koleje.

T paket

Položkami T paketu je identifikátor typu paketu „t“, registr, identifikátor lokomotivy, hodnota rychlosti a hodnota směru jízdy. Registr je pozůstatek starších systémů a je třeba jej zachovat pro zachování jejich kompatibility. Hodnotou může být libovolná jednotková číslice a pro realizaci byla vybrána číslice 1. Identifikátor lokomotivy může být v krátkém tvaru adresy od 0 do 127 a v dlouhém tvaru adresy od 128 do 10293. Standardně mají všechny modely lokomotiv adresu 3, v tomto případě byla tato adresa zachována pro model lokomotivy Taurus a adresa druhého modelu lokomotivy byla přeprogramována na hodnotu 4. Rychlost může nabývat hodnot od 0 do 126 se speciální výjimkou hodnoty -1, která nouzově zastaví model vlaku a resetuje rychlost na hodnotu 0. Hodnota směru je reprezentována hodnotou 1 pro jízdu vpřed a hodnotou 0 pro jízdu vzad. Při úspěšném přijetí a provedení příkazu odešle mikroprocesorová jednotka zpět paket obsahující T jako identifikátor odpovědi na T paket a obdržené hodnoty vyjma identifikátoru lokomotivy.

F paket

Skládá se z identifikátoru typu paketu „f“, adresy lokomotivy a dvou bytů. Funkce se dělí do 5 skupin, z toho první tři skupiny, tj. funkce F0 – F12 využívají pouze první byte a druhý byte není vůbec zahrnut do paketu. Pro poslední dvě skupiny má první byte pevnou hodnotu 222, respektive 223 a pro nastavení stavů funkcí je určen právě druhý byte. Každá skupina má vzorec pro výpočet hodnoty bytu na základě stavu funkcí.

Tabulka 5.1: Vzorce pro výpočet hodnoty bytu skupiny funkcí

F0 – F4	$128 + F1 \times 1 + F2 \times 2 + F3 \times 4 + F4 \times 8 + F0 \times 16$
F5 – F8	$176 + F5 \times 1 + F6 \times 2 + F7 \times 4 + F8 \times 8$
F9 – F12	$160 + F9 \times 1 + F10 \times 2 + F11 \times 4 + F12 \times 8$
F13 – F20	byte 1 = 222 byte 2 = $F13 \times 1 + F14 \times 2 + F15 \times 4 + F16 \times 8 + F17 \times 16 + F18 \times 32 + F19 \times 64 + F20 \times 128$
F21 – F28	byte 1 = 223 byte 2 = $F21 \times 1 + F22 \times 2 + F23 \times 4 + F24 \times 8 + F25 \times 16 + F26 \times 32 + F27 \times 64 + F28 \times 128$

A paket

Paket je identifikován znakem „a“ a obsahuje lineární adresu stacionárního příslušenství, v rámci této realizace se jedná pouze o výhybky. Použitý univerzální dekodér příslušenství má k dispozici adresy od 1 do 4 pro skupinu A výstupů dekodéru a adresy od 5 do 8 pro skupinu B. Posledním prvkem paketu je požadovaný stav příslušenství, kde 0 reprezentuje výhybku ve směru rovně a 1 ve směru pro odbočení.

Odesílání paketů

Funkce vyjme vždy pouze jeden paket z fronty. Paket je následně zakódován do binární podoby kódováním UTF-8 a odeslán. Čtení odpovědi od mikroprocesorové jednotky probíhá cyklicky po jednotlivých řádcích.

5.4 Zpracování videa

Probíhá za pomoci balíčku OpenCV a modulu aruco ze stejného balíčku. Video je snímáno kamerou umístěnou nad modelem železnice.

Inicializace a nastavení kamery

Nástroje potřebné pro zpracování obrazu jsou definovány třídou Camera. Po spuštění aplikace dochází k inicializaci této třídy. Součástí inicializace je vytvoření instance třídy VideoCapture, jejíž účelem je zachytávání snímku z kamery. Použitými parametry je identifikátor kamery a preferované backend API pro zachytávání snímků. Pokud je instance úspěšně propojena s kamerou, dochází k nastavení dalších vlastností jako je rozlišení snímků a čas expozice. Rozlišení je nastaveno na maximální možné s použitou kamerou, to je 1920 pixelů na šířku a 1080 pixelů na výšku. Doba expozice je nastavena na hodnotu -8, která reprezentuje 5 milisekund. Nižší doba expozice zvyšuje spolehlivost rozpoznání značek ArUco v pohybu.

Součástí inicializace je také vytvoření instance třídy Marker, která poskytuje nástroje spojené s detekcí ArUco značek. Parametrem inicializace je slovník ArUco značek. Tyto slovníky jsou definovány v rámci modulu aruco a pro tuto realizaci byl vybrán slovník DICT_4X4_50.

Nástroje ArUco

Třída Marker definuje nástroje pro detekci značek ArUco v obraze. Při inicializaci je nastaven slovník značek a parametry detektoru. Funkce pro generování jednotlivých značek na základě vstupního parametru s identifikátorem značky volá funkci drawMarker a předává slovník značek a identifikátor značky jako vstupní parametry.



Obrázek 5.9: Umístění ArUco značky



Obrázek 5.10: Umístění ArUco značky

Pro detekci značek je vstupní snímek převeden na stupně šedé a předán jako vstupní parametr funkci detectMarkers spolu se slovníkem značek a parametry detektoru. Návratovými hodnotami jsou pole obsahující detekované rohy značek, detekované identifikátory značek a odmítnuté body, které nebyly rozpoznány jako značky ArUco.

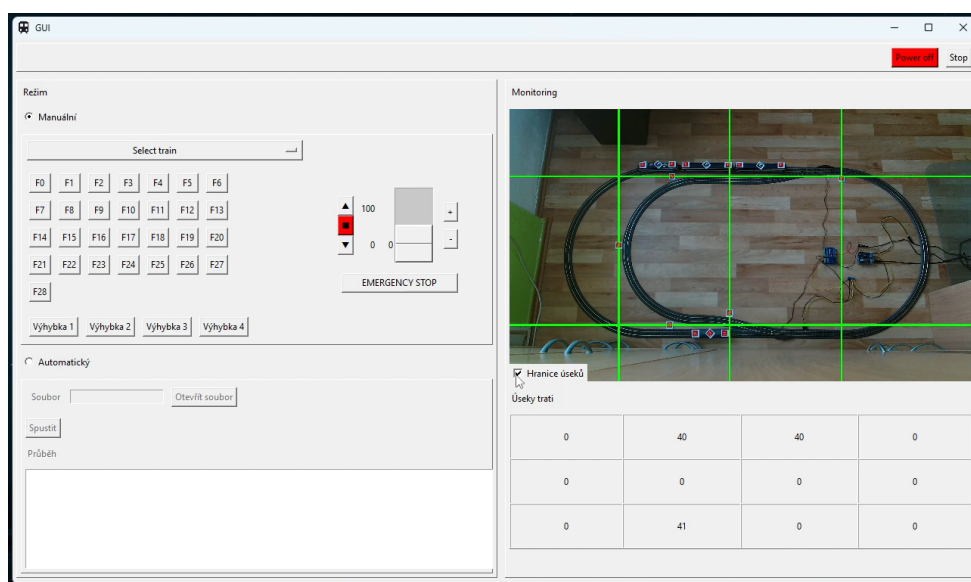
Modul aruco sice disponuje metodou pro vykreslení hranic značek, ale ta pro toto řešení není ideální. Jelikož je třeba takto graficky zvýraznit aruco značky ještě před tím, než je snímek zmenšen na rozlišení pro zobrazení, tak by docházelo ke ztrátám zvýrazněných hranic značek. Z tohoto důvodu je implementována vlastní logika, která vykresluje částečně průhledné červené čtyřúhelníky v místě značek.

Získání snímku

Pro získání snímku z kamery je definována metoda `get_frame`. V rámci této metody je z instance `VideoCapture` získán snímek metodou `read`. Získaný snímek je předán metodě pro detekci ArUco značek, detekované rohy značek jsou předány metodě pro jejich grafické zvýraznění ve snímku. Posledním krokem přípravy snímku pro zobrazení je zmenšení jeho velikosti na velikost, kterou vyžaduje uživatelské rozhraní, tj. 640 pixelů na šířku a 360 pixelů na výšku.

Získání matice úseků

Metoda zavolá metodu `get_frame` pro získání jednoho snímku. Po ověření, zda byl detekován nenulový počet značek, jsou vypočteny souřadnice středu pro detekované značky, jejichž identifikátory spadají do rozsahu značek pro dělení železnice na úseky. Tento rozsah je od 4 do 8, takže celkem existuje 5 značek určujících hranice úseků. Pokud nebyly detekovány všechny značky z rozsahu, souřadnice středu již detekovaných značek jsou uloženy a metodu je třeba zavolat znovu. Jakmile jsou všechny značky detekovány, dochází k vytvoření matice úseků, kde každá položka reprezentující úsek obsahuje souřadnice levé, pravé, horní a dolní hranice úseku. Zároveň jsou souřadnice těchto hranic uloženy pro případné vykreslení úseků do zobrazovaného videa (5.11).



Obrázek 5.11: Hranice úseků v zobrazeném videu

Lokalizace vlaků

Pomocí metody `locate` dochází k výpočtu souřadnic středů detekovaných značek s identifikátory z definovaného rozsahu identifikátorů vlaků, v tomto případě se jedná o identifikátory 40 a 41. Pokud nebyly detekovány žádné značky z rozsahu vlaků, výpočet se neprovádí. Výsledné souřadnice jsou porovnány se souřadnicemi hranic úseků z matice úseků a identifikátor značky vlaku je přiřazen příslušnému úseku v matici obsazenosti úseků. Tato matice obsazenosti úseků je při každém provedení této metody inicializována znovu a po dokončení metody předána jako návratová hodnota.

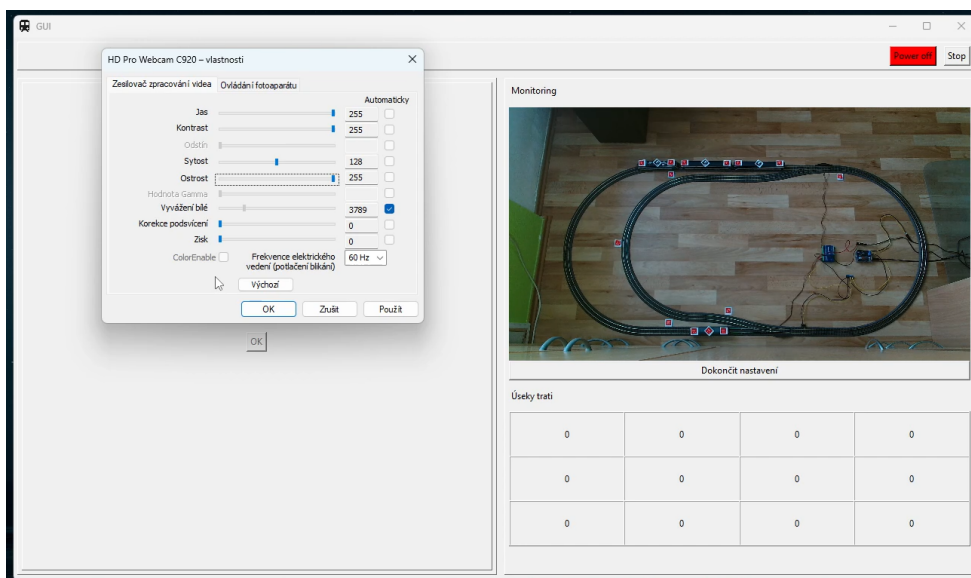
5.5 Podpůrné funkce

Inicializace a nastavení kamery

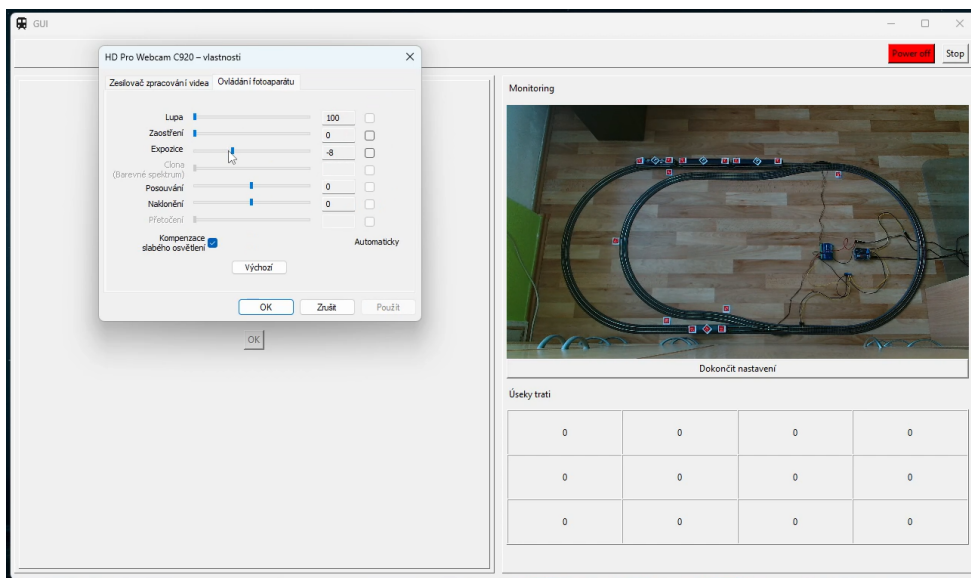
Mezi použité metody patří `camera_startup`, `popup_camera_settings`, `camera_settings`, `end_camera_startup`.

Nastaví text informačního řádku pod displejem na „Probíhá inicializace kamery...“, poté vytvoří a inicializuje instanci třídy `Camera`. Tento proces může trvat několik sekund. Jakmile je inicializace dokončena, změní text v informačním řádku na „Probíhá nastavení kamery...“ a otevře nové okno s možností volby nastavení kamery uživatelem. Uživatel má k dispozici volbu „ano“, při které je okno zavřeno a otevře se nové okno s nastavením kamery (5.12, 5.13) a informační řádek pod displejem je nahrazen tlačítkem „Dokončit nastavení“, kterým uživatel potvrdí konec nastavení kamery. Druhá možnost je „ne“ a pouze zavře aktuální okno. Nezávisle na volbě uživatele je nezávisle v samostatném vlákně zavolána metoda `monitoring_handle`.

Poslední metoda ve startovací sekvenci monitorovací části je `end_camera_startup`, která volá metodu `monitoring_handle` pro získání matice úseků a následně zpřístupní prvky pro nastavení COM sériového portu ovládací části.



Obrázek 5.12: Nastavení kamery - Zesilovač zpracování videa



Obrázek 5.13: Nastavení kamery - Ovládání fotoaparátu

serial_connection_startup

Zavoláno potvrzením volby COM portu uživatelem. Nastaví text informačního řádku nastavení sériové komunikace na „Vytváření sériového spojení...“ a pokusí se o navázání sériového spojení. O výsledku pokusu informuje uživatele v informačním řádku a při úspěšném navázání spojení po krátké prodlevě uzavírá panel nastavení sériové komunikace.

monitoring_handle

Pokouší se získat snímek z fronty a pokud je fronta prázdná, tak je výjimka odchycena a v novém vlákně zavolána metoda video. Metoda monitoring_handle je ukončena a po 10 milisekundách zavolána znovu. Jakmile je snímek získán, je zobrazen uživateli v oblasti displeje.

Pokud stále probíhá startovací sekvence monitorovací části, je metoda ukončena a za 10 milisekund opět zavolána.

Z fronty matic obsazenosti úseků je získána matice. Pokud je fronta prázdná, je v novém vlákně zavolána metoda update_sectors a metoda monitoring_handle je ukončena zavolána znovu za 10 milisekund.

Dále je ověřeno, zda je vytvořeno sériové spojení. V případě že ne, tak je metoda monitoring_handle ukončena a zavolána znovu za 10 milisekund.

Posledním krokem této metody je aktualizace zobrazení obsazenosti úseků na základě získané matice a také detekce kolizí z této matice. Kolize jsou detekovány, pokud jsou obsazeny dva sousední úseky různými vlaky a zároveň tyto úseky sdílejí stejnou trať. Strany, se kterými úsek sdílí trať jsou definovány v matici sectors_path.

Po úspěšném ukončení je metoda zavolána po 10 milisekundách znovu.

model_stop

Metoda volaná při detekci kolize nebo při stisknutí tlačítka „Stop“ v horní liště. Při automatickém režimu oznamuje uživateli „STOP: Kolizní kurz detekován.“. Pro všechny definované vlaky je volána metoda `emergency_stop` a následně je vypnuto napájení hlavní trati.

collision_detected

Volá metodu `model_stop` a nastaví barvu pozadí úseku, ve kterém byla kolize detekována, na červenou.

update_sectors

Volá metodu `locate` instance třídy `Camera` a pokud je vrácena matice obsazenosti úseků, tak ji vkládá do fronty pro zobrazení a pro automatické řízení.

change_state

Slouží k zpřístupnění vybraného ovládacího panelu a zablokování nevybraného ovládacího panelu.

video

Zavolá metodu `get_frame` instance třídy `Camera` a ověří dostupnost snímku. Konvertuje barvený prostor z BGR, který využívá OpenCV, na RGBA a následně převede snímek z formátu pole na formát podporovaný balíčkem Tkinter. Upravený snímek je vložen do fronty pro zobrazení.

increase_throttle, lower_throttle

Metody volané tlačítkem „+“, respektive „-“, které získají aktuální hodnotu posuvníku plynu a pokud změna není mimo maximální nebo minimální limit, je zvýšena, respektive snížena o 1.

emergency_stop

Nastaví posuvník plynu na hodnotu 0 zavolá metodu `manual_control_handle` s parametrem akce `EMERGENCY_BRAKE`.

direction_forward, direction_backward, direction_stop

Metody volané tlačítky voliče směru. Upravují vizuální zvýraznění jednotlivých tlačítek, nastavují proměnnou směru, `direction_stop` navíc nastavuje posuvník plynu na hodnotu 0. Nakonec je volána metoda `manual_control_handle` s parametrem akce `DIRECTION`.

set_power

Přepíná stavovou proměnnou napájení a nastavuje vzhled tlačítka napájení. Volá metodu `manual_control_handle` s parametrem akce `POWER`.

function_toggle

Přepíná mezi stisknutým a nestisknutým stavem tlačítek funkcí. Nastavuje matici stavů funkcí a volá metodu `manual_handle_control` s parametrem akce `FUNCTIONS` a indexem funkce.

turnout_toggle

Přepíná mezi stisknutým a nestisknutým stavem tlačítek výhybek. Nastavuje stav výhybky v poli stavů výhybek a volá metodu `manual_handle_control` s parametrem akce `TURNOUTS`, indexem výhybky a stavem výhybky.

select_train

Nastavuje index vlaku podle výběru uživatele a volá metodu `manual_handle_control` s parametrem akce `TRAIN` a identifikátorem vlaku.

manual_control_handle

Metoda na základě vybrané akce upravuje potřebné proměnné. Následně volá metodu pro získání paketu v závislosti na vybrané akci z instance třídy `SerialCommunication` a metodu `send_command` z instance stejné třídy pro odeslání paketu. Obě metody jsou zavolány v samostatných vláknech.

automatic_control_handle

Naplní frontu načtenými příkazy a vyjme první příkaz. V samostatném vlákně zavolá metodu `execute_command` a předá parametry z příkazu. Dále v samostatném vlákně zavolá metodu `progress_output` pro výpis prováděných kroků (5.14).

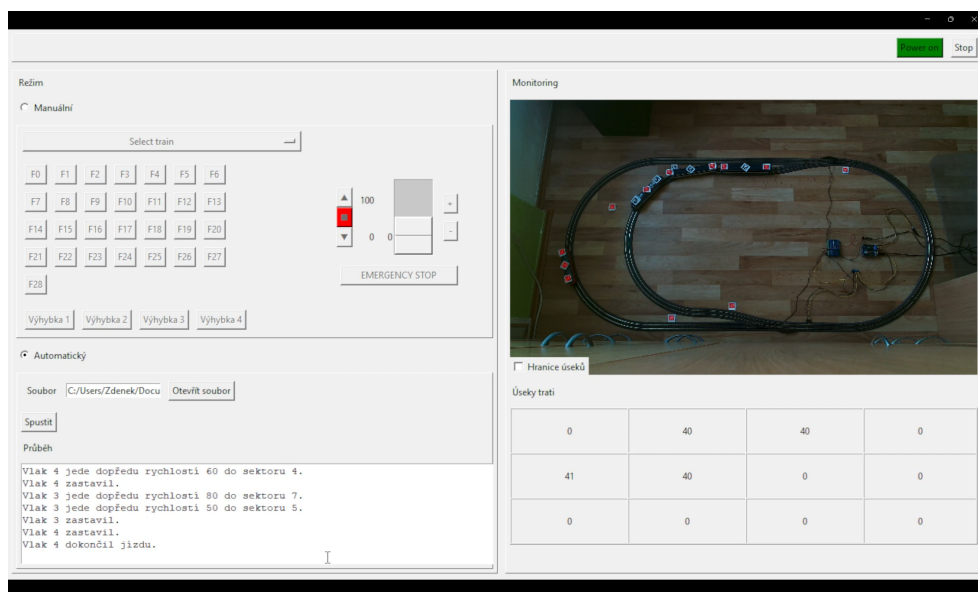
execute_command

Pokud je aktivní událost `event_stop`, tak je metoda ukončena. Je proveden pokus o získání matice obsazenosti úseků. Při neúspěchu čeká metoda 10 milisekund a spouští se znovu v novém vlákně. Pokud není zapnuté napájení, jsou provedeny všechny kroky potřebné jeho zapnutí.

Všechny parametry jsou porovnány s nastavenými hodnotami a při nerovnosti nastaveny jako nové hodnoty. Pokud dojde ke změně, je nastaven příznak. Pokud se vlak nachází v cílovém úseku, tak nastává několik možností. Vlak zůstane stát pokud nemá další cílový úsek, jinak pokud není další příkaz ve frontě, tak je automatizace ukončena. Poslední možností je vyčkání po dobu 10 milisekund a zavolání metody znovu s parametry z dalšího příkazu. Pokud došlo ke změně hodnot, je vytvořen paket a odeslán.

load_command_file

Volán tlačítkem „Otevřít soubor“. Pokud není seznam příkazů prázdný, je vyprázdněn. Otevře dialogové okno pro výběr cesty CSV souboru s příkazy. Ověří, zda byla cesta vybrána nebo byl výběr zrušen. Otevře soubor pro čtení a zpracovává jednotlivé řádky. Ukládá pole hodnot příkazů do pole příkazů a vypisuje do textového okna informace o načtených příkazech. Nakonec vypíše jejich celkový počet.



Obrázek 5.14: Výpis provedených kroků automatizace

Kapitola 6

Závěr

Cílem práce bylo zjistit stávající možnosti a navrhnout řešení ovládání modelu vláčku počítačem. Nejprve jsem se seznámil s problematikou modelové železnice. Důležitým zjištěním je existence standardů, které se používají a jsou definovány v podobě National Model Railroad Association – NRMA.ORG, dále pro digitální řešení definice DCC protokolu a jeho široké uplatnění. I když jsem identifikoval celou řadu komerčně dostupných řešení od jednoduchých komponent až po komplexní modulární systémy bylo mým záměrem navrhnout vlastní variantu postavenou na otevřených standardech s možností další modifikace v budoucnu – maximálně se vyhnout „black box“ komponentám. Celé řešení jsem postavil na digitálním protokolu DCC který poskytuje největší flexibilitu a možnosti řízení.

Při implementaci jsem musel vyřešit komplikace s rozpoznáváním obrazu, které použito pro monitorování polohy vlaků. Jedná se o vhodné nastavení kamery pro spolehlivé rozpoznávání ArUco markerů v pohybu a za různých světelných podmínek. Podařilo se mi nalézt vhodnou kombinaci parametrů pro vyhovující provoz. Další identifikovanou komplikací je provoz při umělém osvětlení za použití „blikajícího“ světelného zdroje jako například dnes používané LED žárovky nebo stolní lampy. Pro účely práce bylo vyřešeno kombinací různých světelných zdrojů.

Výsledkem práce je sestava kolejiště, mikroprocesorové ovládání kolejiště s možností implementace dalších funkcionalit a analogový výkonový interface ve funkční podobě. Po překonání výše uvedených komplikací je monitorování polohy vlaku kamerou a OpenCV knihovnou rovněž funkční a spolehlivé. Vše je propojeno do řídicí aplikace, která umožňuje jak ruční ovládání vlaků na kolejišti, tak jednoduchou formu automatizace a monitorování polohy vlaků.

V budoucnu se nabízí celá řada rozšíření. První možností je optimalizovat identifikaci polohy vlaků z obrazu pomocí pokročilejších algoritmů bez použití ArUco markerů. Dále je možné zdokonalit algoritmy pro automatizaci. Například plynulý rozjezd a zastavení, řízení rychlosti vlaku v závislosti na místě trati (např. zatáčka, výhybka apod.) nebo pokročilejší zamezení kolize. Mikro procesová řídicí jednotka ARDUINO s WiFi modulem nabízí možnost dalšího rozšíření například napojení tabletů nebo chytrých telefonů.

Literatura

- [1] ARDUINO.CC. *Arduino Motor Shield Rev3* [online]. [cit. 2023-05-08]. Dostupné z: <https://docs.arduino.cc/hardware/motor-shield-rev3>.
- [2] ARDUINO.CC. *ARDUINO UNO REV3* [online]. [cit. 2023-05-08]. Dostupné z: <https://docs.arduino.cc/hardware/uno-rev3>.
- [3] ARDUINO.CC. *ARDUINO UNO WiFi REV2* [online]. [cit. 2023-05-08]. Dostupné z: <https://docs.arduino.cc/hardware/uno-wifi-rev2>.
- [4] AUTOR_NEUVEDEN. *DCC++ Components* [online]. [cit. 2023-05-08]. Dostupné z: https://dccwiki.com/DCC_Plus_Plus.
- [5] ELEGOO_INC. *ELEGOO MEGA 2560 R3 Board With USB Cable Compatible With Arduino IDE* [online]. [cit. 2023-05-08]. Dostupné z: <https://www.elegoo.com/products/elegoo-mega-2560-r3-board>.
- [6] FRED DECKER, M. K. *Deek-Robot Motor Shield* [online]. [cit. 2023-05-08]. Dostupné z: <https://dcc-ex.com/reference/hardware/motorboards/deek-robot-motor-shield.html>.
- [7] FRED DECKER, M. K. *EX-RAIL (Automation/Animation) Overview* [online]. [cit. 2023-05-08]. Dostupné z: <https://dcc-ex.com/ex-rail/overview.html>.
- [8] FRED DECKER, M. K. *What is EX-CommandStation?* [online]. [cit. 2023-05-08]. Dostupné z: <https://dcc-ex.com/ex-commandstation/index.html>.
- [9] MOROP. *NEM 631* [online]. [cit. 2023-05-08]. Dostupné z: https://www.morop.org/images/NEM_register/NEM_E/nem631_en_1985.pdf.
- [10] MOROP. *NEM 640* [online]. [cit. 2023-05-08]. Dostupné z: https://www.morop.org/images/NEM_register/NEM_E/nem640_en_1988.pdf.
- [11] MÄRKLIN&CIE_GMBH. *Central Station 3 plus* [online]. [cit. 2023-05-08]. Dostupné z: https://streaming.maerklin.de/public-media/anleitungen/CS3_Manual_EN_final-lo.pdf.
- [12] NMRA.ORG. *Communications Standards For Digital Command Control; A. General Packet Format* [online]. [cit. 2023-05-08]. Dostupné z: <https://www.nmra.org/sites/default/files/s-92-2004-07.pdf>.
- [13] OPENCV. *Detection of ArUco Markers* [online]. [cit. 2023-05-08]. Dostupné z: https://docs.opencv.org/4.x/d5/dae/tutorial_aruco_detection.html.

- [14] OPENCV_TEAM. *Introduction to OpenCV-Python Tutorials* [online]. [cit. 2023-05-08]. Dostupné z: https://docs.opencv.org/3.4/d0/de3/tutorial_py_intro.html.
- [15] PERECLI. *What is Railroad Automation?* [online]. [cit. 2023-05-08]. Dostupné z: <https://www.perecli.com/rrauto/>.
- [16] ROCRAIL.NET. *Innovative Model Railroad Control System* [online]. [cit. 2023-05-08]. Dostupné z: <https://wiki.rocrail.net/doku.php>.
- [17] STÁREK, Z. *Modelová železnice: Od historie modelů po digitální ovládání kolejiště*. 1. vyd. Brno: CPress, 2013. ISBN 978-80-251-2199-3.
- [18] WIKIPEDIA. *Digital Command Control* [online]. [cit. 2023-05-08]. Dostupné z: https://en.wikipedia.org/wiki/Digital_Command_Control.
- [19] ZBYNĚK STÁREK, V. V. *Železniční modelářství pro každého*. 2. vyd. Praha: Computer Press, 2003. ISBN 80-251-0168-1.

Příloha A

Obsah přiloženého média

- Bc_práce_PDF - Tato práce ve formátu PDF
- Bc_práce_Zdrojový_tvar - Tato práce ve zdrojovém tvaru
- Dokumentace - Demonstrační video použití aplikace, Instalační soubory použitého vývojového prostředí
- Zdrojové_texty - Zdrojové texty aplikace