

SW a HW systému pro řízení posuvné brány

Michal Paluřík

Bakalářská práce
2024



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav informatiky a umělé inteligence

Akademický rok: 2023/2024

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Michal Paluřík**
Osobní číslo: **A21109**
Studijní program: **B0613A140020 Softwarové inženýrství**
Forma studia: **Prezenční**
Téma práce: **SW a HW systému pro řízení posuvné brány**
Téma práce anglicky: **SW and HW System for Sliding Gate Control**

Zásady pro vypracování

1. Prostudujte dostupné pohony a řídicí jednotky posuvných bran včetně open source projektů.
2. Navrhněte architekturu řízení posuvné brány pro vzdálené ovládání a monitoring brány z počítače nebo mobilního telefonu.
3. Vyberte vhodné senzory pro detekci koncových bodů, překážek a přetížení motorů a otestujte jejich funkci.
4. Realizujte HW zapojení řídicího systému posuvné brány nebo jejího modelu.
5. Implementujte a otestujte ovládací SW.

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. HUGHES, Austin. Electric motors and drives: fundamentals, types, and applications. 3rd ed. Boston: Elsevier/Newnes, c2006. ISBN 0750647183.
2. ROUBÍČEK, Ota. Elektrické motory a pohony: příručka techniky, volby a užití vybraných druhů. Praha: BEN – technická literatura, 2004. ISBN 80-7300-092-X.
3. NOVÁK, Petr. Mobilní roboty: pohony, senzory, řízení. Praha: BEN – technická literatura, 2005. ISBN 80-7300-141-1.
4. ROUBAL, Jiří a Petr HUŠEK. Regulační technika v příkladech. Praha: BEN – technická literatura, 2011. ISBN 978-80-7300-260-2.
5. PINKER, Jiří. Mikroprocesory a mikropočítače. Praha: BEN – technická literatura, 2004. ISBN 80-7300-110-1.

Vedoucí bakalářské práce:

Ing. Tomáš Dulík, Ph.D.
Ústav informatiky a umělé inteligence

Datum zadání bakalářské práce: **5. listopadu 2023**

Termín odevzdání bakalářské práce: **13. května 2024**



doc. Ing. Jiří Vojtěšek, Ph.D. v.r.
děkan

prof. Mgr. Roman Jašek, Ph.D., DBA v.r.
ředitel ústavu

Ve Zlíně dne 5. ledna 2024

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne

Michal Paluřík, v.r.
podpis studenta

ABSTRAKT

Tato bakalářská práce se zabývá návrhem a realizací ovládní posuvné brány s využitím platformy ESP32. Teoretická část práce představuje přehled různých typů bran, jejich pohonů a řídicích jednotek. Dále práce podrobně zkoumá a porovnává různé platformy pro řízení a představuje existující projekty s podobnou tematikou. Praktická část práce je zaměřena na návrh a realizaci systému pro vzdálené ovládní a monitoring brány pomocí mobilního telefonu nebo počítače. Byly vybrány a testovány senzory pro detekci koncových bodů, překážek a zatížení motoru. Realizace zahrnovala vývoj hardwarového zapojení a programování ovládacího softwaru.

Klíčová slova: open-source, posuvná brána, vzdálené ovládní, ESP32, Arduino, Raspberry Pi

ABSTRACT

This bachelor's thesis deals with designing and implementing sliding gate control using the ESP32 platform. The theoretical part of the work presents an overview of different types of gates, as well as their drives and control units. Furthermore, the work examines and compares various management platforms in detail and presents existing projects with similar themes. The practical part of the work is focused on designing and implementing a system for remote control and monitoring of the gate using a mobile phone or computer. Sensors were selected and tested to detect endpoints, obstacles, and motor loads. The implementation included the development of hardware connection and programming the control software.

Keywords: open-source, sliding gate, remote control, ESP32, Arduino, Raspberry Pi

Rád bych poděkoval panu Ing. Tomáši Dulíkovi, Ph.D. za cenné rady, trpělivost a odborné vedení, které bylo klíčové pro úspěšné dokončení této práce.

Prohlašuji, že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

ÚVOD	9
I TEORETICKÁ ČÁST	10
1 TYPY BRAN	11
1.1 KŘÍDLOVÁ BRÁNA	11
1.1.1 Jednokřídlá brána	11
1.1.2 Dvoukřídlá brána.....	12
1.1.3 Skládací brána	13
1.2 POSUVNÁ BRÁNA.....	14
1.2.1 Posuvná brána samonosná.....	14
1.2.2 Posuvná brána pojezdová.....	14
1.2.3 Posuvná brána teleskopická	15
2 ŘÍDÍCÍ JEDNOTKA	16
3 POHONY	17
3.1 POHONY POSUVNÝCH BRAN	17
3.2 POHONY KŘÍDLOVÝCH BRAN.....	19
4 SNÍMACÍ A BEZPEČNOSTNÍ PRVKY BRAN	21
4.1 MECHANICKÝ KONCOVÝ SPÍNAČ	21
4.2 MAGNETICKÝ KONCOVÝ SPÍNAČ	22
4.3 INDUKČNÍ SNÍMAČE	22
4.4 ULTRAZVUKOVÉ SNÍMAČE	23
4.5 FOTOBUNKY	23
4.6 VÝSTRAŽNÝ MAJÁK.....	24
5 PLATFORMY PRO ŘÍZENÍ	25
5.1 MIKROKONTROLER	25
5.2 ARDUINO.....	25
5.2.1 Arduino UNO.....	26
5.2.2 Arduino Micro.....	26
5.2.3 Arduino Due.....	27
5.3 ESPRESSIF.....	27
5.3.1 ESP32.....	28
5.3.2 ESP8266.....	28
5.4 RASPBERRY PI.....	29
5.4.1 Raspberry Pi 4B	29
5.4.2 Raspberry Pi Pico	30
5.5 STMICRO	30
5.5.1 STM32L432KC.....	30
5.5.2 P-NUCLEO-WB55	31
5.5.3 NUCLEO-F103RB.....	31
5.6 POROVNÁNÍ PLATFORM	32
5.6.1 ATmega 8-bit a ARM Cortex	32
5.6.2 ESP32 a ESP 8266	32
5.6.3 ESP32 vs Raspberry Pi Pico	32

5.6.4	ESP32 vs P-NUCLEO-WB55	33
5.7	PROGRAMOVÁNÍ ESP32	33
5.7.1	Sériová komunikace	35
6	ALTIUM DESIGNER.....	36
7	AUTODESK FUSION 360	37
8	WEBSERVER	38
8.1	ARDUINO A ESP32 JAKO WEBSERVER	38
9	MODULY A SENZORY PRO MIKROKONTROLERY.....	39
9.1	ULTRAZVUKOVÝ SENZOR	39
9.2	INFRAČERVENÝ SENZOR	40
9.3	HALLŮV SENZOR	40
9.4	SENZOR NAPĚTÍ	41
9.5	PROUDOVÝ SENZOR.....	41
9.6	LASEROVÝ SNÍMAČ VZDÁLENOSTI.....	42
9.7	RELÉOVÝ MODUL	43
9.8	MODUL ROTAČNÍHO ENKODÉRU	43
9.9	H-MŮSTEK.....	44
10	OPEN SOURCE PROJEKTY	45
10.1	SMART GATE.....	45
10.2	RPI PARKING DOORS	45
10.3	GARAGEPI.....	46
10.4	RASPBERRY PI GARAGE DOOR OPENER	47
10.5	ARDUINO GATE OPENER	47
10.6	WIFI GARAGE DOOR CONTROLLER	47
10.7	GARAGE DOOR CONTROLLER	48
10.8	AUTOMATIC GATE OPENER	49
II	PRAKTICKÁ ČÁST	50
11	NÁVRH ARCHITEKTURY A REALIZACE ŘÍZENÍ BRÁNY	51
11.1	SCHÉMA ZAPOJENÍ MODELU POSUVNÉ BRÁNY	51
11.2	VYBRANÉ KOMPONENTY	53
11.2.1	ESP32 DevKitC	53
11.2.2	Servo motor FS90R.....	54
11.2.3	Snímač proudu	55
11.2.4	Laser KY-008.....	55
11.2.5	Modul laserového snímače.....	56
11.2.6	Koncové spínače	57
11.2.7	Ostatní součástky	57
11.2.8	Napájení modelu	58
11.3	REALIZACE HW ZAPOJENÍ ŘÍDICÍHO SYSTÉMU A CELÉHO MODELU BRÁNY.....	58
11.3.1	3D model.....	59
11.4	WEBOVÁ STRÁNKA	60
12	OVLÁDACÍ SW MODELU BRÁNY.....	62

12.1	POPIS ZDROJOVÉHO KÓDU	62
12.1.1	Import knihoven	62
12.1.2	Nastavení připojení	63
12.1.3	Definice pinů	63
12.1.4	Nastavení pozic motoru a blikání LED	64
12.1.5	Webserver	64
12.1.6	Monitorování a průběh proudu.....	64
12.1.7	Nastavení počtu pokusů u přihlášení.....	64
12.1.8	Uživatelské rozhraní.....	65
12.1.9	Funkce pro blikání LED diody.....	65
12.1.10	Nastavení součástek	66
12.1.11	Definice chování webserveru	66
12.1.12	Definice chování webserveru pro hlavní stránku.....	66
12.1.13	Definice chování webserveru při změně stavu koncového spínače	67
12.1.14	Zavírání brány a kontrola přerušení laseru.....	68
12.1.15	Kontrola přetížení motoru	70
	ZÁVĚR	71
	SEZNAM POUŽITÉ LITERATURY	72
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	78
	SEZNAM OBRÁZKŮ	81
	SEZNAM PŘÍLOH.....	84

ÚVOD

V dnešní době, kdy technologie neustále postupuje, se automatizace stává klíčovým prvkem moderního života, a posuvné brány nejsou výjimkou. Tyto systémy nejenže zvyšují bezpečnost a pohodlí, ale také přispívají k efektivitě a estetice rezidenčních i průmyslových prostor. Historie ovládání posuvných bran sahá do doby, kdy byla potřeba jednodušeji a bezpečněji řídit přístup do objektů.

V teoretické části rozebírám různé typy bran, jejich pohony a snímací prvky, které umožňují jejich efektivní fungování. Analyzuji také různé platformy pro řízení a poskytnu srovnání jejich výkonu a funkčnosti. Probírám zde také existující projekty, které využívají jinou platformu pro řízení brány.

Praktická část práce popisuje návrh, konstrukci a testování prototypu řídicího systému. Zabývá se výběrem a implementací senzorů pro detekci koncových bodů a překážek, stejně jako sledováním stavu motoru, aby bylo zajištěno bezpečné a efektivní ovládání brány. Vývoj hardwarového zapojení a ovládacího softwaru je klíčovým výstupem této práce, který demonstruje možnosti moderních řídicích technologií v praxi.

Tento projekt vznikl z původního zadání pro jednu firmu, která požadovala návrh systému pro zavírání brány. Když však došlo ke změně priorit u zaměstnavatele, rozhodl jsme se místo reálné implementace zaměřit na vytvoření teoretického modelu.

I. TEORETICKÁ ČÁST

1 TYPY BRAN

V dnešní době máme široké množství bran, které můžeme použít pro vjezd vozidel do firemních objektů nebo plochu před domem. Mezi nejrozšířenější typy bran se řadí křídlové brány a posuvné brány. Tyto brány lze otevírat manuálně nebo také automaticky přes pohon u brány. [1]

1.1 Křídlová brána

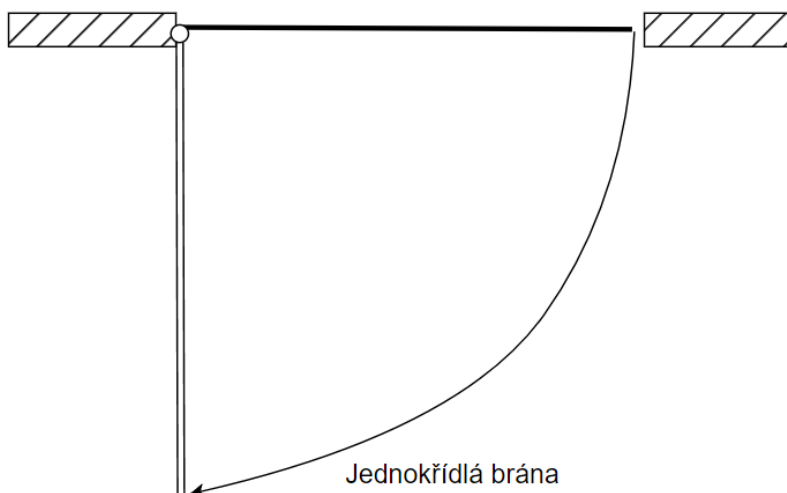
Tento typ brány můžeme nejčastěji vidět u vjezdů velkých výrobních hangárů nebo hal. Křídlové brány jsou široce využívané v mnoha aplikacích, od soukromých nemovitostí po průmyslové areály. Tyto brány jsou složeny z jednoho nebo více křídel, která se otvírají směrem ven nebo dovnitř a otáčejí se kolem vertikální osy na pantech umístěných na robustních sloupech. Důležitým aspektem je dostatečný prostor pro pohyb křídel, aby nebyl jejich pohyb nijak omezen. Křídlové brány nabízí robustní a trvanlivé řešení pro vstupy, kde není možné nebo vhodné použít variantu posuvné brány. Tyto brány můžeme rozdělit do jedno, dvou či čtyř křídel. V dnešní době se tyto brány dají otevírat manuálně i automaticky přes dálkový ovladač či kartou. [1], [2]

1.1.1 Jednokřídlá brána

U těchto bran se setkáme s jedním křídlem, které se otevírá většinou do prostoru pozemku. Otevírání opačným směrem, a to do ulice je již problém, jelikož zákony České republiky říkají, že se jedná o překážku na pozemní komunikaci. U těchto bran se setkáme s maximální délkou tří metrů. Delší rozměry u brány poté bránu dělají nestabilní a panty, na kterých je brána nesena jsou více namáhané a snižuje se jejich životnost.



Obrázek 1 Jednokřídlá brána [3]



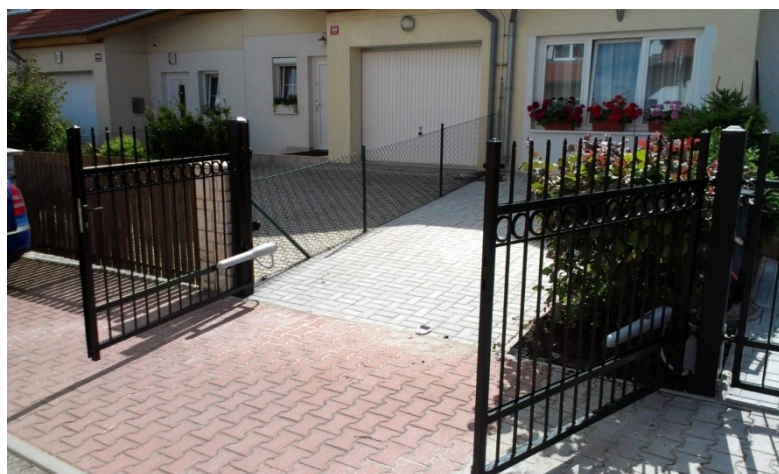
Obrázek 2 Otevírací rádius jednokřídlé brány

Mezi jednu z výhod jednokřídlých bran je ta, že křídlo se otevírá pouze k jedné straně. Další je její jednodušší montáž a následné seřizování.

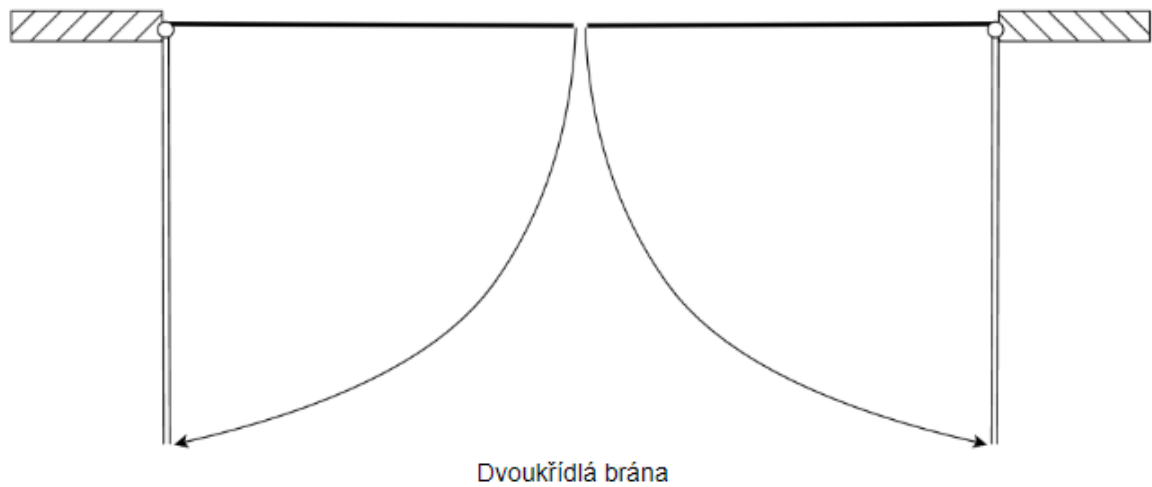
K nevýhodám patří samozřejmě nevyužitý prostor, kde se brána pohybuje, jelikož se jedná o jedno velké křídlo. [2]

1.1.2 Dvoukřídlá brána

Tato brána se příliš neliší oproti branám jednokřídlým. Při instalaci v užším vjezdu umožňuje použití kratších křídel, čímž se minimalizuje jejich zasahování do prostoru pozemku. Můžeme se zde setkat s obvyklou šířkou do 3,5 metru. Při použití speciálních pantů se dostaneme až do celkové šířky 6 metrů v součtu pro obě křídla. U této brány lze také udělat asymetrická křídla, kdy jedno bude kratší než to druhé. [2]



Obrázek 3 Dvoukřídlá brána [2]



Obrázek 4 Otevírací rádius dvoukřídle brány

1.1.3 Skládací brána

Skládací křídlové brány představují ideální řešení pro úsporu prostoru při vjezdech do dvorů nebo jiných ohraničených prostor. Tyto brány využívají systém pákových převodů, které umožňují skládání křídel podobně jako paraván. Tím se výrazně snižuje prostor, který by jinak byl potřebný pro jedno nerozložené křídlo. Tento typ brány je vhodný pro místa, kde je omezený prostor po stranách vjezdu. [4]



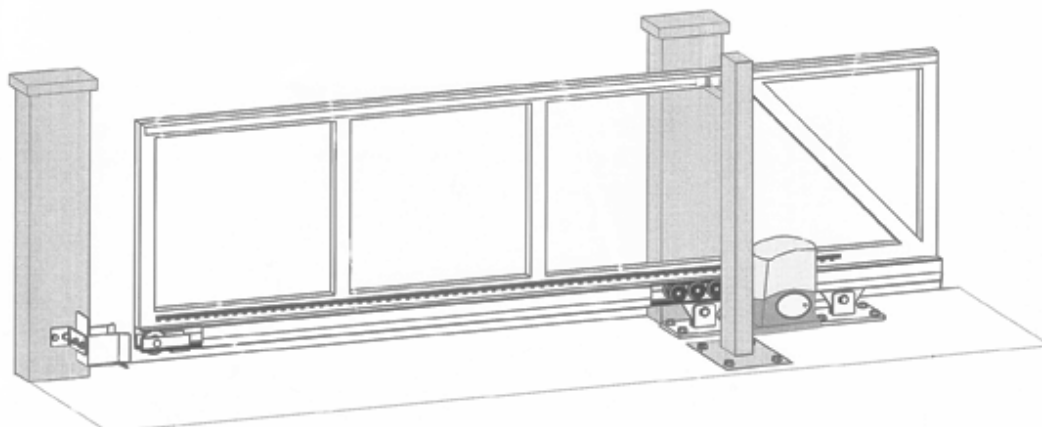
Obrázek 5 Křídlová skládací brána [5]

1.2 Posuvná brána

Posuvné brány získaly na oblíbenosti, jelikož se otevírají posunem do strany, nejčastěji podél plotu. Díky tomu nezabírají tolik prostoru například uvnitř dvora jako tomu je u křídlových a lze parkovat hned za bránou. Narozdíl od křídlových bran je u této potřeba mít dostatečný prostor k otevření do boku. Posuvné brány se rozdělují na samonosné a pojezdové. [6]

1.2.1 Posuvná brána samonosná

U posuvných bran samonosných je mechanismus založen na vozících a samonosném profilu C. Tento profil se pohybuje po dvou vozících pevně upevněných v betonovém bloku. U této brány nemá vliv na funkčnost ani malá vrstva sněhu, jelikož brána jezdí přibližně 8 cm nad zemí. Jedna z nevýhod konstrukce samonosné posuvné brány je, že musí být minimálně o jednu třetinu delší, než je její potřebná šířka průjezdu. [6], [7]

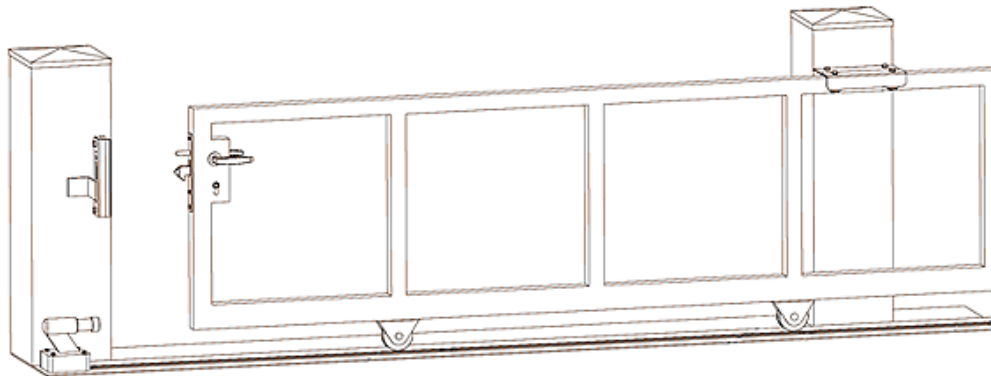


Obrázek 6 Posuvná brána samonosná [8]

1.2.2 Posuvná brána pojezdová

Základním prvkem u posuvné brány pojezdové je vodící kolejnice nebo úhelník, který je zabetonovaný v zemi. Po této kolejnici se poté brána pohybuje po pojezdových kolečcích. Z cenového hlediska je pojezdová brána levnější než samonosná, nicméně v zimních obdobích vyžaduje vyšší nároky na údržbu kvůli pojezdové kolejnici. U této brány nemusíme

počítat s šířkou navíc jako u samonosné brány, jelikož pojezdové brány jezdí po již zmíněné kolejnici zabudované v zemi. [6], [7]



Obrázek 7 Posuvná brána pojezdová [9]

1.2.3 Posuvná brána teleskopická

Teleskopické posuvné brány představují efektivní řešení pro místa s omezeným prostorem vedle vstupu. Tento typ brány je složen z více segmentů, které se při otevírání postupně skládají za sebe jako díly teleskopu. Tyto segmenty jsou obvykle vybaveny pojezdovými kolečky a vedou se pomocí vodících lišt, což zajišťuje hladký a stabilní pohyb brány. Výhodou teleskopických brán je jejich schopnost zaujmout mnohem menší prostor než klasické posuvné brány. Velikost brány se odvíjí od počtu segmentů a jejich šířky. [10]

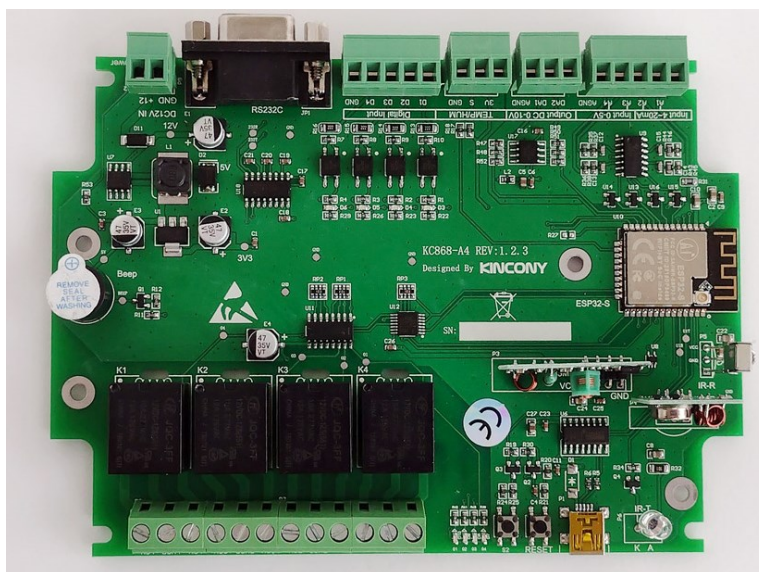


Obrázek 8 Posuvná brána teleskopická [10]

2 ŘÍDÍCÍ JEDNOTKA

Řídicí jednotky bran jsou klíčovým prvkem pro zajištění plynulého a bezpečného provozu automatizovaných bránových systémů. Na trhu existuje široké spektrum těchto jednotek, které se liší rozsahem funkcí, od základního otevírání a zavírání až po pokročilé bezpečnostní a komunikační možnosti. V průmyslových aplikacích se obvykle používají jednotky s důrazem na robustnost a bezpečnostní prvky, jako jsou automatické zastavení při detekci překážky nebo systémy pro automatické otevření při detekci vozidla. Pokročilejší jednotky mohou nabízet vzdálené ovládání přes mobilní aplikace, integraci s domácími automatizačními systémy, nebo dokonce využití biometrických dat pro přístup. Bezpečnostní funkce zahrnují různé senzory, jako fotobuňky nebo optické závory, které umožňují detekci překážek a předejdou tak možným úrazům nebo poškození majetku. Díky těmto technologiím je zajištěn vyšší stupeň bezpečnosti a uživatelského komfortu. [11]

Pro cenově efektivní řešení automatizačních projektů představuje řídicí deska KC868-A4 ideální volbu. Tato deska je vybavena výkonným mikrokontrolerem ESP32, který umožňuje použití standardního softwaru a senzorů, běžně nasazovaných v průmyslových aplikacích. KC868-A4 lze efektivně využít pro konstrukci reálné posuvné brány, a to za výrazně nižší náklady, pouhých 500 Kč. Díky této technologii je možné sestavit plně funkční řídicí jednotku za zlomek ceny oproti tradičním systémům. Flexibilita a cenová dostupnost této desky ji činí výbornou volbou pro širokou škálu aplikací, od robustních průmyslových řešení až po soukromé projekty, což přináší vysokou míru přizpůsobitelnosti a snadnou integraci do existujících systémů. [12]



Obrázek 9 Řídicí jednotka KC868-A4 [12]

3 POHONY

U výběru pohonu k bráně je klíčové zohlednit typ brány, jelikož motor vhodný pro křídlovou bránu nebudeme moct použít u posuvné a naopak. Dalším důležitým faktorem je i celková váha brány, protože motor musí mít odpovídající výkon, aby se mohla brána plynule pohybovat. Když bychom totiž vybrali motor s nedostatečným výkonem, mohlo by to vést k pomalejšímu otevírání, ale hlavně k přehřívání motoru. Toto přehřátí by mohlo způsobit nenávratné poškození motoru. Dále je požadovaná rychlost otevírání brány, což má přímý vliv na výkon motoru. S rostoucím výkonem se zvyšuje i jeho samotná cena, tudíž je nezbytné vybrat motor s dostatečným výkonem, aby byl schopen uvést bránu do pohybu. [13]

3.1 Pohony posuvných bran

Pohon posuvných bran je klíčovým prvkem pro jejich efektivní a hladký chod. Typicky je tento pohon instalován buď do betonového základu nebo na sloupek na jedné straně brány. Základním mechanismem je ozubené kolo, které zapadá do hřebene upevněného podél spodní hrany brány. Některé modely pohonů mohou být vybaveny různými typy koncových spínačů, které slouží k automatickému zastavení brány v požadované pozici. Zatímco některé pohony používají mechanické koncové spínače umístěné nad ozubeným kolem, jiné modely preferují magnetické koncové spínače pro vyšší přesnost a odolnost proti opotřebení.

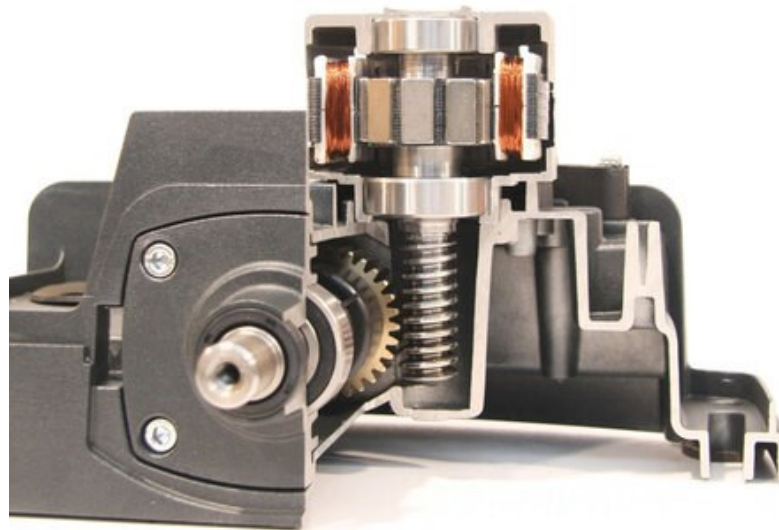


Obrázek 10 Pohon posuvných vrat [14]

Jestliže se brána s pohonem nachází na odlehlém místě bez přívodu elektrické energie, existují i pohony, do kterých lze zapojit záložní baterii nebo solární panel. Baterie se vkládají

vedle řídicí desky. Většinou mají tyto baterie napětí 12 nebo 24 V. V závislosti na váze brány zaručuje tato baterie 6 cyklů otevření a zavření brány po sobě. Solární panely baterii dobíjejí během dne a zajistí tak chod brány i v noci. [13], [15]

Uvnitř pohonů nalezneme šnekovou převodovku, která je poháněna jedním ze třech typů pohonů. Prvním z nich je střídavý motor, který pracuje na 230 V. Jeho hlavní výhodou je, že poskytuje vcelku vysoký výkon, což je užitečné v případech, které vyžadují silný pohon. Nevýhodou ale je, že tyto motory jsou obvykle větší a těžší. Druhým je stejnosměrný motor běžící na 24 V. Je menší a lehčí oproti prvnímu typu motoru. Pracuje se zde s menším napětím 24 V, což je považováno za bezpečnější pro obsluhu. Nevýhodou je, že oproti střídavým motorům poskytuje nižší výkon. Třetím motorem je bezkartáčový motor. Tyto motory, jelikož jsou bez kartáčů, mají nízké ztráty a tím se zvyšuje jejich účinnost. Nesmí chybět taky menší údržba, delší životnost a nízký hluk díky absenci kartáčů. Nevýhodou oproti výše zmíněným motorům je vyšší cena a složitější elektronika pro řízení. [16], [17]



Obrázek 11 Šneková převodovka [18]



Obrázek 12 Typy motorů [18]

3.2 Pohony křídlových bran

Pohon křídlových bran představuje zásadní prvek pro jejich efektivní a spolehlivé fungování. Na trhu je dostupné několik typů pohonů: pákový, přímočarý a podzemní. Přímočarý pohon se obvykle montuje přímo na bránu a je viditelný. Tento typ je uchycen na jedné straně ke sloupku brány a na druhé ke křídlu. Jeho klíčovým mechanickým prvkem je kuličkový šroub, po kterém se posouvá matice poháněná elektrickým motorem, typicky o napětí 24 V. [19]



Obrázek 13 Přímočarý pohon [19]

Pákové pohony bran jsou populárním a efektivním řešením pro automatizaci křídlových bran. Tyto systémy jsou charakteristické svým ramenem. Instalují se na pilíře z domovní strany a díky svým kloubovým ramenům efektivně ovládají pohyb brány, umožňující její otvírání i v omezeném prostoru. [19]



Obrázek 14 Pákový pohon [19]

Posledním typem je pohon podzemní. Tyto pohony jsou, jak název napovídá, instalovány přímo pod povrchem země u nosného sloupku brány. Tato umístění zajišťují, že mechanické části pohonu nejsou viditelné, což přispívá k celkově čistšímu a nenápadnějšímu vzhledu brány. Pracovní princip podzemního pohonu je podobný jako u pákových pohonů, s tím rozdílem, že všechny pohyblivé a funkční části jsou skryty pod zemí. [19]



Obrázek 15 Podzemní pohon [19]

4 SNÍMACÍ A BEZPEČNOSTNÍ PRVKY BRAN

Brány jsou vybaveny různými ochrannými prvky a spínači, které zajišťují bezpečnost a plynulý provoz. K indikaci, zda je brána v otevřené poloze nebo zavřené, se používají koncové snímače. K detekci překážek jsou zde poté fotobuňky nebo ultrazvukový senzor. [20]

4.1 Mechanický koncový spínač

Funkce těchto koncových spínačů funguje na změně polohy mechanických kontaktů. Při dojezdu do žádané polohy se tento kontakt sepne a dá řídicí jednotce signál, že je brána v poloze otevřeno nebo zavřeno. Spínač je skládá z kontaktů a mechanickým dílem, který tyto kontakty přepíná. Jednou z nevýhod těchto koncových spínačů je, že v zimním období mohou zamrznat. Dalším problémem je mechanické opotřebení, tudíž je výrobcem dán maximální počet cyklů u daného spínače. [20]



Obrázek 16 Mechanický koncový spínač [21]

4.2 Magnetický koncový spínač

Základem spínače je jazýčkový kontakt, který reaguje na přiblížení magnetu a aktivuje spínač. Tyto spínače jsou obzvláště užitečné v podmínkách, kde jsou mechanické snímače náchylné k selhání kvůli mrazu nebo znečištění, jelikož nevyžadují přímý kontakt. Je důležité si uvědomit, že jejich funkčnost může být ovlivněna přítomností externích magnetických polí. [22], [36]



Obrázek 17 Magnetický koncový spínač [23]

4.3 Indukční snímače

Princip těchto snímačů je, že vytváří ve svém okolí magnetické pole. Pokud se do tohoto pole dostane kovový předmět, impedance cívky se změní. Snímač obsahuje elektroniku, která tuto změnu vyhodnotí a převede na daný výstup. Výrobci nabízejí dva typy snímačů. Prvním je stíněný a druhý nestíněný. Rozdílem je, že stíněný snímač má plastovou „hlavičku“ zapuštěnou, zatímco u nestíněného vyčnívá. Výhodou snímače je jeho odolnost proti vnějším vlivům. Dále je zde, že není potřeba na snímaném zařízení nemusíme magnet, jak tomu je bylo u magnetických spínačů. [20], [36]



Obrázek 18 Indukční snímač [24]

4.4 Ultrazvukové snímače

Tyto snímače mají vždy vysílač a přijímač ultrazvukového signálu. Generuje se signál o frekvenci vyšší než 20 kHz a ten se poté odráží od měřeného objektu zpět k přijímači. Snímač měří čas mezi vyslaným a přijatým, podle které se poté spočítá dráha, kterou signál urazil. Tyto čidla se používají nejčastěji k detekci překážek. Mezi výhody patří bezkontaktní měření nebo možnost detekce širokého spektra materiálů. Do nevýhod můžeme zařadit problém s detekcí úzkých či malých objektu. [20], [36]



Obrázek 19 Ultrazvukový senzor [25]

4.5 Fotobuňky

Skládají se také s přijímače a vysílače. Vysílač vysílá infračervený paprsek, přijímač jej přijme. Když paprsek detekuje vozidlo, osobu nebo jinou překážku, přijímač zaznamená tuto změnu a informuje řídicí jednotku. Řídicí jednotka poté zareaguje podle předem definovaného nastavení. Ve většině případů dojde k zastavení a po chvíli k otevření brány. Mezi typy fotobuněk patří reléové, s odrazkou, bateriové nebo s datovou sběrnici. [26], [27]



Obrázek 20 Fotobuňka s odrazkou [26]

4.6 Výstražný Maják

Jedná se o zařízení, které slouží k signalizaci, zda je brána v pohybu. To je důležité ke zvýšení bezpečnosti a prevenci nehod, jelikož varuje osoby, které jsou v blízkosti brány, že se brány otevírá nebo zavírá. Mezi tyto majáky patří rotační majáky s halogenovou žárovkou nebo led majáky. Rotační pracuje na principu otáčejícího se zrcadla okolo světla. Samozřejmě halogenové žárovky jsou dobře viditelné za všech světelných podmínek, ale jejich nevýhodou je vysoký odběr. Poté přicházejí led majáky, které mohou mít vyšší pořizovací cenu, ale odebírají značně menší proud a mají větší životnost. [28], [29]



Obrázek 21 Výstražný maják na bráně [28]

5 PLATFORMY PRO ŘÍZENÍ

V dnešní době máme mnoho mikrokontrolerů, které můžeme využívat k různým účelům. Na trhu můžeme nalézt platformy jako jsou Arduino, ESP, STMico, Raspberry Pi, Pine64 nebo Banana Pi.

5.1 Mikrokontroler

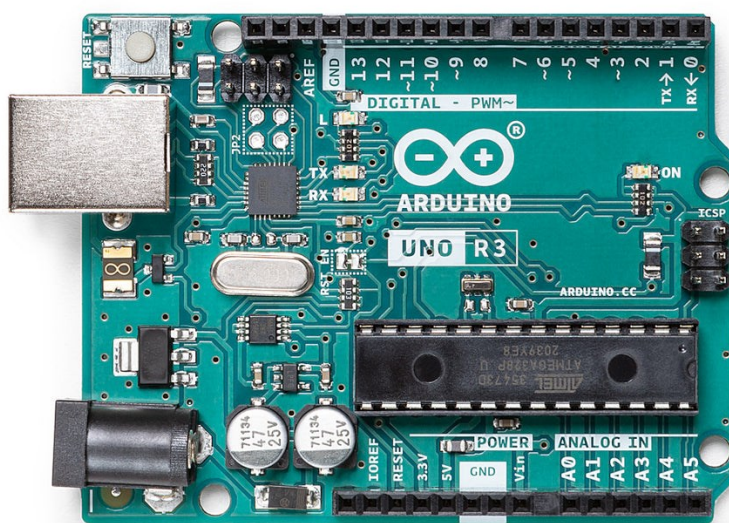
Mikrokontroler je malý integrovaný obvod vyrobený pro specifické operace v embedded systémech. Typický mikrokontroler obsahuje procesor, paměť, která je ve většině případech typu Flash nebo EPROM, poté paměť RAM a vstupní nebo výstupní periferie. U mikrokontroleru se mohou vyskytovat i řadiče přerušení, čítače, časovače, watchdog nebo A/D převodník. [30]

5.2 Arduino

Arduino, založené v Itálii v roce 2005, se rychle stalo revoluční platformou v oblasti elektroniky, která umožňuje jednoduchý vstup do světa programování a hardware pro studenty i profesionály. Cílem zakladatelů z Interaction Design Institute bylo poskytnout cenově dostupné a snadno použitelné vývojové sady, které byly alternativou k dražším řešením. Arduino je známé svou open source povahou, což umožnilo uživatelům po celém světě sdílet návody, kódy a schémata a tím podporovat komunitní rozvoj a inovace. Platforma během let rozšířila své portfolio produktů o různé verze desek, některé ve spolupráci s velkými technologickými společnostmi jako Intel. Na trhu se objevují také mnohé klony, které, i přes podobné technické specifikace, často postrádají oficiální podporu a dokumentaci od Arduina. Arduino desky jsou osazeny mikrokontrolery, převážně od společnosti Atmel, a nabízejí široké možnosti připojení senzorů a snímačů pro realizaci rozličných projektů, od automatizace domácnosti po robotické systémy. Díky přístupnosti a bohaté komunitní podpoře je Arduino ideální volbou pro ty, kteří se chtějí naučit základy elektroniky a programování. [31], [32]

5.2.1 Arduino UNO

Arduino UNO je postaveno na procesoru ATmega328P, který umožňuje řízení a interakci s různými elektronickými komponenty. Deska je vybavena USB portem typu B pro snadné připojení k počítači, 14 digitálními piny, z toho šest podporuje PWM výstupy, a šesti analogovými vstupy, což umožňuje širokou škálu připojení pro senzory a další moduly. Napájení desky je možné buď přes USB, nebo externím adaptérem, což zvyšuje její flexibilitu v různých aplikacích a projektech. [33]

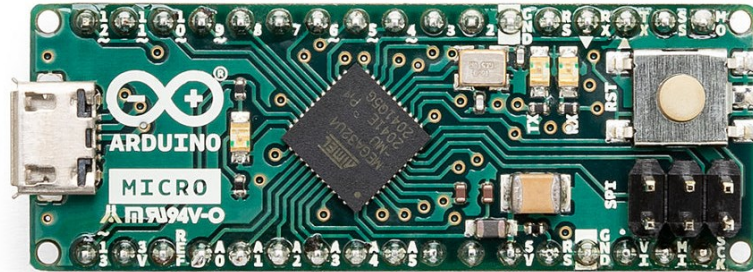


Obrázek 22 Arduino UNO rev3 [33]

5.2.2 Arduino Micro

Arduino Micro je kompaktní deska založená na mikrokontroleru ATmega32U4. Nabízí 20 digitálních vstupů/výstupů, z nichž 7 může generovat PWM signály, a dále 12 analogových vstupů. Díky integrovanému micro USB portu může deska fungovat jako zařízení USB, například jako klávesnice nebo myš, což usnadňuje vývoj interaktivních projektů. Tato

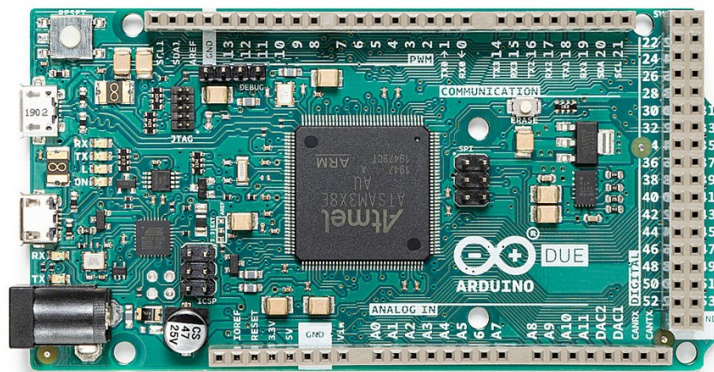
univerzálnost a přímá komunikace s počítačem činí Arduino Micro ideální volbou pro návrhy, kde je potřeba vstupní simulace zařízení. [34]



Obrázek 23 Arduino Micro [34]

5.2.3 Arduino Due

Arduino Due je výkonná deska vybavená 32bitovým ARM Cortex-M3 procesorem Atmel SAM3X8E, který pracuje na frekvenci 84 MHz. Tato deska disponuje 54 digitálními I/O piny a 12 analogovými vstupy, což umožňuje rozmanité použití. Má dva micro USB porty, jeden pro programování a druhý pro připojení externích zařízení jako myši nebo klávesnice. Důležitou charakteristikou je provoz na 3,3 V, což vyžaduje opatrnost při připojování komponent kvůli riziku poškození. [35]



Obrázek 24 Arduino Due [35]

5.3 Espressif

Espressif Systems byla založena v roce 2008 v Šanghaji. Díky příznivé ceně se jim podařilo roku 2014 prorazit na trh s ESP8266. O čtyři roky později uvedli na trh svůj vlajkový čip ESP32, který nabízel více funkcí společně s Bluetooth připojením. [37], [38]

5.3.1 ESP32

Jedná se o desku postavenou na mikrokontroleru ESP32 od společnosti Espressif. Může být vybavena jednojádrovým nebo dvoujádrovým čipem Xtensa LX6, který běží na frekvenci až 240 MHz. Deska je vybavena 16 MB Flash pamětí a micro USB konektorem. Dále disponuje 38 GPIO piny, z nichž 34 je konfigurovatelných jako vstupně výstupní a 18 analogových vstupů. Některé piny slouží pro speciální funkce jako je SPI, UART nebo I2C.

Nechybí nám tu ani konektivita v podobě Wi-Fi a Bluetooth. U Wi-Fi je poté maximální rychlost 150 Mbit/s na standartu 802.11n při 40MHz.

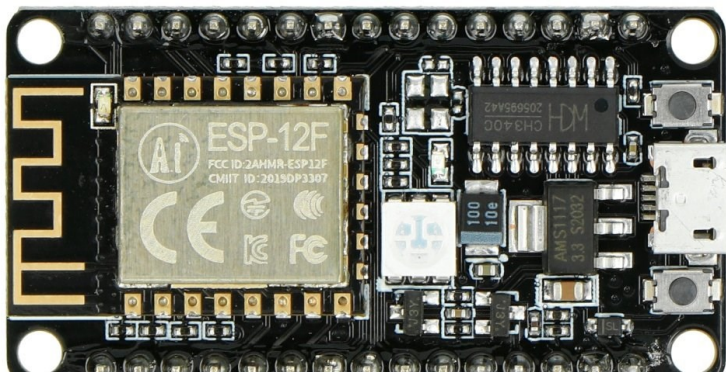
Deska také podporuje protokol ESP-NOW, který byl vyvinut firmou Espressif. Tento protokol je navržen pro efektivní komunikaci mezi zařízeními s nízkou latencí. [37], [39]



Obrázek 25 ESP32 DevKitC

5.3.2 ESP8266

Na desce nalezneme 17 GPIO pinů, ze kterých 11 je konfigurovatelných jako vstupně výstupní a 1 analogový vstup. Samotná deska je poháněna 32bitovým procesorem Tensilica Xtensa LX106, který dosahuje frekvence až 160 MHz. Některé piny mají také speciální funkce jako je UART, PWM Také nám tu nechybí konektivita Wi-Fi. [37], [40]



Obrázek 26 ESP8266 [41]

5.4 Raspberry Pi

Příběh začal v roce 2006, když se Eben Upton pustil do vytvoření jednodeskového počítače. Inspiroval se od počítače BBC Micro a o šest let později bylo vytvořeno první Raspberry Pi. Hlavním cílem bylo ukázat a pomoci mladým lidem objevit počítače za nízkou cenu. [42]

5.4.1 Raspberry Pi 4B

Tento mikropočítač je vybaven procesorem Broadcom BCM2711. Jedná se o čtyř jádrový Cortex-A72 procesor běžící na frekvenci 1,8GHz. Desku můžeme koupit ve variantě s 1, 2, 4 nebo 8 GB SDRAM. Deska disponuje i širokou škálou portů, jako jsou dva USB 3.0, dva USB 2.0, dva mikro HDMI porty podporující obraz až do 4K, USB-C port, 4pólový audio konektor, ethernet port podporující Power over Ethernet a slot na micro SD kartu. Nesmíme zapomenout na Wi-Fi a Bluetooth 5.0. Na desce také nalezneme 40pinový GPIO konektor. [43]



Obrázek 27 Raspberry Pi 4B

5.4.2 Raspberry Pi Pico

Tato deska je postavena na mikrokontroleru RP2040. Základem je procesor ARM Cortex-M0+, který běží na frekvenci 133MHz. Dále je deska vybavena 256KB RAM, 2 MB Flash, 26 GPIO piny, z nichž tři mohou fungovat jako analogové vstupy. Na desce také nalezneme micro USB port pro připojení k počítači. [44]



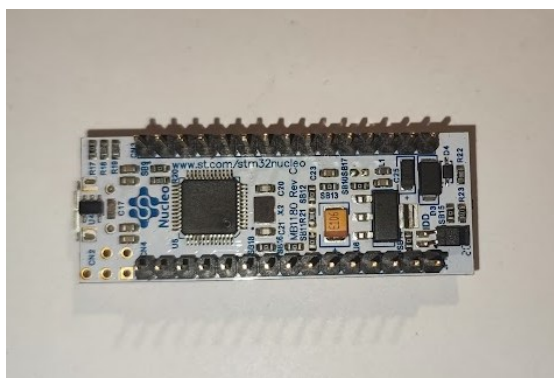
Obrázek 28 Raspberry Pi Pico [44]

5.5 STMicro

Společnost STMicroelectronics se zrodila roku 1987 spojením společností Thomson Semiconductors a SGS Mictoelettronica. Mezi lety 1990 až 2010 se strategicky přesunula na vývoj čipů pro mobilní telefony a bezdrátové technologie. S tím také přišel vývoj a expanze celé společnosti během deseti let od roku 2000. Během této doby se začali zaměřovat i na mikrokontrolery nebo senzory. [45]

5.5.1 STM32L432KC

Deska je postavena na procesoru ARM Cortex-M4, běžící na frekvenci 80MHz. Obsahuje také jednotku s plovoucí desetinnou čárkou a jednotku ochrany paměti, které zvyšují bezpečnost aplikací. Deska také poskytuje přímý přístup k online zdrojům Arm Mbed Enabled. Nalezneme zde také kompatibilitu s mnoha hardwarovými doplňky, včetně připojení Arduino Uno Revision 3 a rozšiřujících kolíků ST Morpho. [46], [47]



5.5.2 P-NUCLEO-WB55

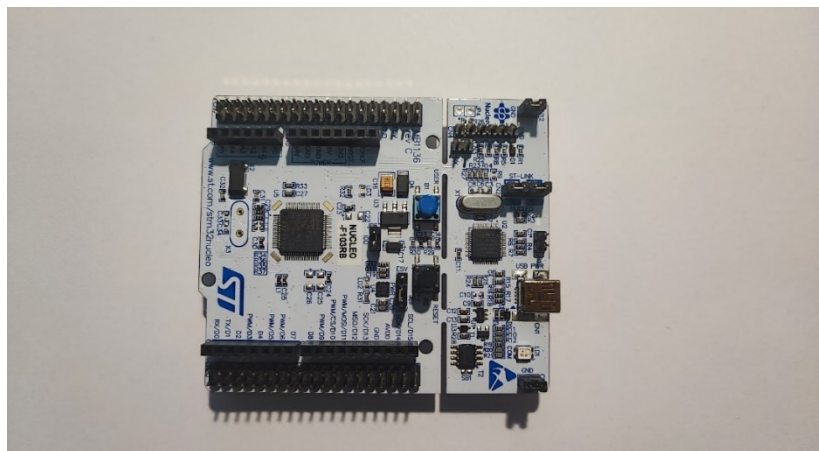
Deska je vybavena 32bitovým procesorem ARM Cortex-M4. Nalezneme zde také 2,4 GHz RF vysílač, který podporuje Bluetooth 5.0 a IEEE 802.15.4-2011 PHY a MAC, což muS umožňuje komunikovat s širokou škálou bezdrátových zařízení. Obsahuje také adaptivní akcelerátor v reálném čase, vestavěnou zásuvku pro baterii CR2032 a vestavěný debugger ST-LINK. [48]



Obrázek 30 NUCLEO-WB55RG [48]

5.5.3 NUCLEO-F103RB

Deska NUCLEO-F103RB je vybavena 32bitovým ARM Cortex-M3 CPU, který dosahuje rychlosti až 72 MHz. Obsahuje 128 KB Flash paměti a 20 KB SRAM, což umožňuje efektivní správu aplikací. S 64 GPIO piny deska podporuje rozmanité periférie a komponenty. Navíc je kompatibilní s Arduino Uno V3. Nabízí také dvě uživatelská tlačítka pro resetování a interakci, a integrované nástroje pro ladění a programování ST-LINK. [49]



Obrázek 31 Nucleo-F103RB

5.6 Porovnání platforem

5.6.1 ATmega 8-bit a ARM Cortex

Řada ATmega328, je 8bitový mikrokontroler známý svou spolehlivostí a jednoduchostí, díky čemuž je vhodný pro různé elektronické projekty. Na druhou stranu procesory ARM Cortex, jako jsou ty, které se nacházejí v deskách ze série STM32L4xx Nucleo, nabízejí pokročilejší 32bitovou architekturu s vyššími výkonnostními možnostmi, větší kapacitou paměti a rychlejším taktem.

Řada ATmega, včetně ATmega328, je cenově výhodná a efektivní pro jednodušší aplikace. Naproti tomu procesory ARM Cortex poskytují vysokorychlostní provoz, více paměti a širší škálu komunikačních protokolů, díky čemuž jsou ideální pro aplikace vyžadující zvýšený výkon a škálovatelnost. Volba mezi architekturami ATmega a ARM Cortex v projektech Arduino nakonec závisí na konkrétních požadavcích projektu, jako jsou potřeby výkonu, spotřeba energie a omezení nákladů. [50]

5.6.2 ESP32 a ESP 8266

Při srovnání mikrokontroleru ESP32 a ESP8266, se nám objeví několik rozdílů. ESP32 je výkonnější varianta s dvoujádrovým procesorem s taktovací frekvencí až 240 MHz, zatímco ESP8266 používá jednojádrový procesor běžící až na 160 MHz. Pokud jde o GPIO, ESP32 nabízí větší flexibilitu s 34 GPIO piny ve srovnání s 16 piny ESP8266. Kromě toho ESP32 podporuje analogová měření na 18 kanálech, zatímco u ESP8266 nalezneme pouze jeden 10bitový ADC pin. ESP32 obsahuje podporu Bluetooth, která u ESP8266 chybí, což z něj činí univerzálnější volbu pro projekty vyžadující připojení Bluetooth. I když je ESP8266 šetrnější k rozpočtu, vyšší cena ESP32 je odůvodněna jeho vynikajícím výkonem a dalšími funkcemi, díky čemuž je preferovanou volbou pro mnoho projektů. Oba mikrokontrolery lze naprogramovat pomocí Arduino IDE nebo MicroPython, což nabízí flexibilitu ve vývoje- vých prostředích. [51]

5.6.3 ESP32 vs Raspberry Pi Pico

Když srovnáme mikrokontrolery ESP32 s Raspberry Pi Pico je zřejmé, že vyhovují odlišným preferencím a projektovým požadavkům díky jejich kontrastnímu výkonu a funkcím. Jeden významný rozdíl spočívá ve výkonu a paměti. ESP32 se může pochlubit mírně lepším výkonem s taktováním CPU na 240 MHz ve srovnání s Raspberry Pi Pico, spolu s

rychlejšími instrukčními rychlostmi a nižšími požadavky na napětí pro provoz. Naopak Raspberry Pi Pico obsahuje šest nezávislých RAM bank, které usnadňují efektivní zpracování paralelních úloh. Raspberry Pi Pico navíc integruje kartu Micro-SD pro programování a spouštění operačního systému, což je funkce, která v ESP32 chybí. [52]

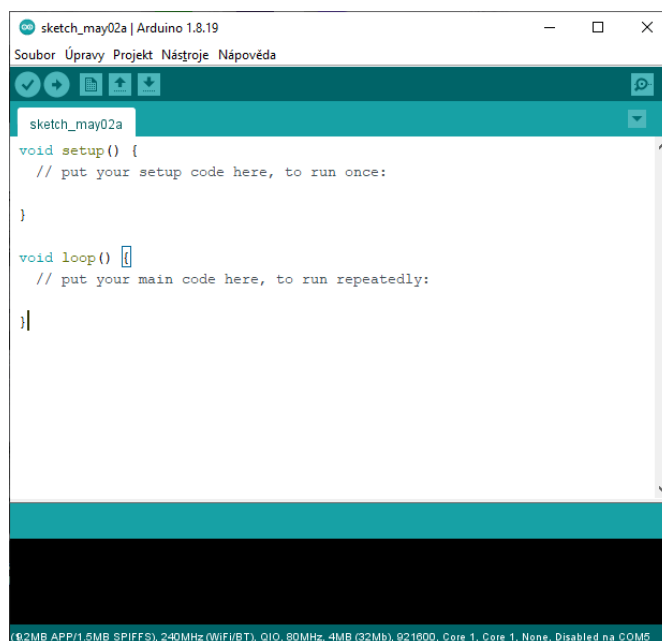
5.6.4 ESP32 vs P-NUCLEO-WB55

ESP32 je velmi dobře známo pro své možnosti Wi-Fi a kompatibilitu s Arduino, zatímco P-NUCLEO-WB55, konkrétně mikrokontroler STM32WB55, nabízí specializovanější přístup s funkcemi jako Bluetooth 5.2 a komunikační schopnosti IEEE 802.15.4. Mikrokontroler STM32WB55, který je součástí vývojového balíčku P-NUCLEO-WB55, představuje sadu funkcí přizpůsobených konkrétním aplikacím vyžadujícím tyto funkce. Mezi tyto funkce můžeme zařadit hardwarové šifrování, jedinečné generování ID nebo nízkou spotřebu energie. Kromě toho se mikrokontroler STM32WB55RG, který je součástí startovací sady NUCLEO-WB55RG, může pochlubit působivými specifikacemi, jako je 1 MB Flash, 256 KB SRAM a kombinace jader ARM Cortex-M4 a Cortex-M0+. [48], [53]

5.7 Programování ESP32

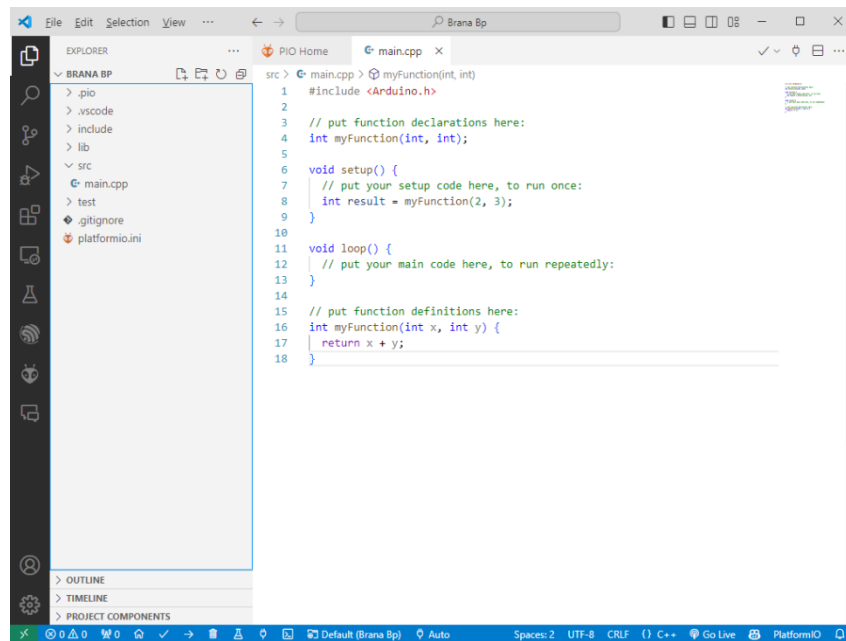
Pro programování mikrokontroleru ESP32 je možné využít několik vývojových prostředí mezi které patří Arduino IDE a Visual Studio Code. Arduino IDE, které je dostupné pro stažení na stránkách Arduino.cc, je tradiční volbou pro práci s mikrokontrolery z rodiny Arduino a je napsáno v jazyce Java. Visual Studio Code je zase více univerzální editor kódu, který podporuje rozšíření pro programování ESP32, jako je PlatformIO nebo rozšíření pro Arduino.

Po spuštění nového projektu v Arduino IDE jsou k dispozici dvě základní funkce: void setup() a void loop(). Funkce void setup() se spouští jednou při startu programu nebo po stisknutí tlačítka reset a slouží k inicializaci nastavení, zatímco void loop() obsahuje kód, který se neustále opakuje a řídí běh programu. V horní části Arduino IDE jsou umístěna tlačítka pro ověření kódu a jeho nahrání do Arduina, zatímco dolní část poskytuje nástroje jako sériový monitor pro sledování dat nebo diagnostiku. Arduino IDE také poskytuje sériový monitor, který se aktivuje pomocí ikony lupy. Tento nástroj je klíčový pro diagnostiku a monitorování dat přenášovaných mezi ESP32 a počítačem. [54]



Obrázek 32 Arduino IDE

V případě vývoje ve Visual Studio Code s PlatformIO, vývojáři oceňují integraci s mnoha dalšími nástroji a knihovnami, což umožňuje efektivnější správu složitějších projektů. Visual Studio Code také poskytuje výhody jako pokročilé debugovací možnosti a přímou integraci s verzovacími systémy, jako je Git, což usnadňuje spolupráci na větších projektech. [54]



Obrázek 33 Visual Studio Code

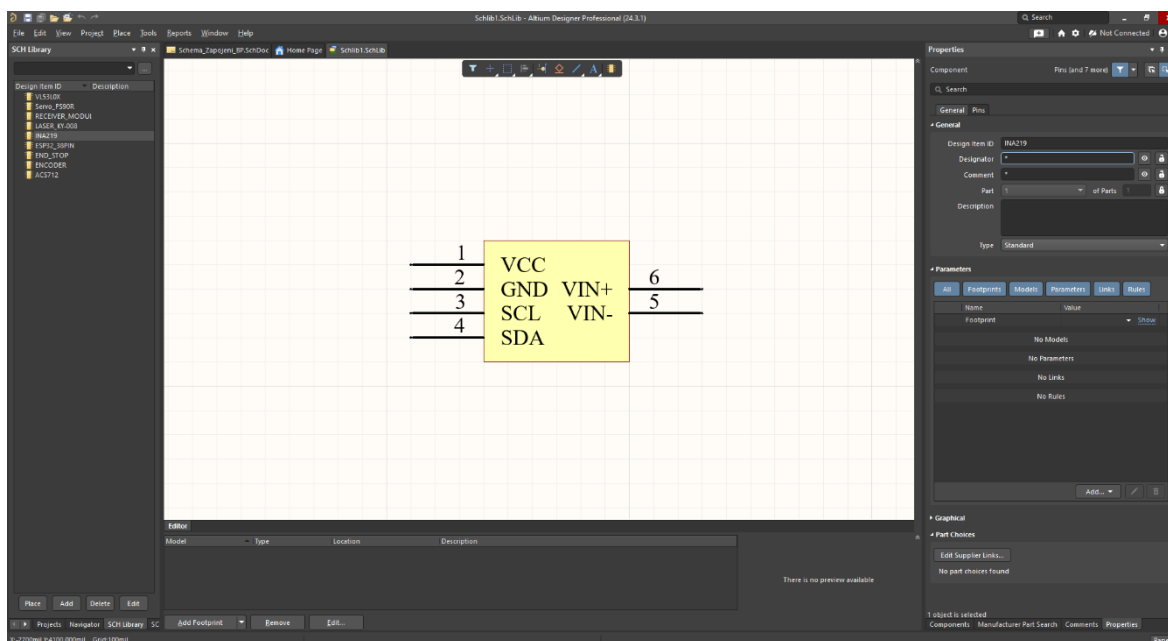
5.7.1 Sériová komunikace

Sériová komunikace je nezbytnou součástí pro interakci mezi mikrokontrolerem a počítačem, umožňující nejen posílání příkazů k řízení mikrokontroleru, ale i zpětné odesílání dat z mikrokontroleru do počítače. Tato komunikace je klíčová zejména při ladění aplikací, kdy může vývojář sledovat chování programu v reálném čase přes sériový monitor. V případě vývojového prostředí Arduino IDE nebo Visual Studio Code s PlatformIO pro ESP32, je sériová komunikace základem pro diagnostiku a efektivní odstranění problémů. Pokud dojde k problémům, jako například zaseknutí programu, programátor může prostřednictvím sériového monitoru vypisovat a sledovat kroky prováděného programu, což mu umožní identifikovat místo, kde může být chyba. Inicializace sériové komunikace v Arduino IDE se provádí pomocí příkazu `Serial.begin(115200)`, kde hodnota 115200 specifikuje rychlost komunikace. Po inicializaci je možné pomocí příkazu `Serial.println("Váš text");` posílat z mikrokontroleru textové zprávy do počítače, což se může hodit pro zobrazení stavů nebo hodnot získaných z různých senzorů. [54]

6 ALTIUM DESIGNER

Altium Designer je software pro návrh elektronických obvodů a tištěných spojů, který poskytuje komplexní nástroje pro 2D a 3D modelování desek, tvorbu schémat a simulaci elektronických obvodů. Program je známý svou uživatelsky přívětivou rozhraním a integrovaným přístupem k návrhu, což umožňuje inženýrům rychle a efektivně vyvíjet složité elektronické komponenty a systémy.

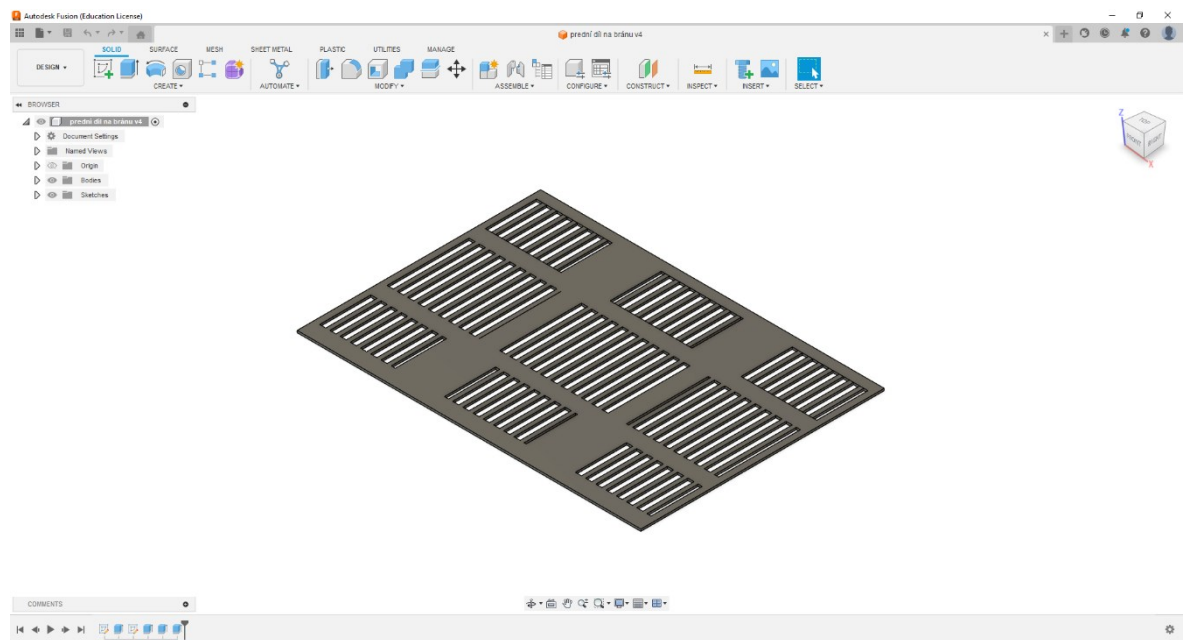
Altium Designer podporuje řadu funkcí, včetně vytváření knihoven komponent, správu dat projektu, auto-routing pro automatické vedení spojů, a mnoho dalších nástrojů, které zjednodušují proces návrhu. Tento software je oblíben mezi profesionálními designéry elektroniky pro jeho schopnost integrovat všechny aspekty návrhu PCB v jednom nástroji, což zlepšuje efektivitu a zkracuje dobu potřebnou pro vývoj produktu. [55]



Obrázek 34 Prostředí Altium Designer

7 AUTODESK FUSION 360

Fusion 360 je cloudově založený 3D CAD, CAM a CAE software, který je vyvíjen společností Autodesk. Je to komplexní nástroj pro návrh produktů, který integruje různé aspekty inženýrství, včetně 3D modelování, počítačem podporované výroby, počítačem podporovaného inženýrství, a generativního designu. Software je vhodný pro různé typy projektů od průmyslového designu přes mechanické inženýrství až po výrobu. Fusion 360 umožňuje uživatelům rychle a efektivně vytvářet 3D modely produktů, provádět simulace pro testování funkčnosti a výkonu produktů, a využívat pokročilé CAM nástroje pro přípravu CNC obrábění. Jednou z klíčových vlastností Fusion 360 je jeho schopnost kolaborace a sdílení projektů v reálném čase, což usnadňuje týmovou práci a iterace designu, a to vše prostřednictvím cloudové platformy, která zajišťuje přístup k projektům odkudkoliv a kdykoliv. [55]



Obrázek 35 Prostředí Fusion 360

8 WEBSERVER

Webový server je počítačový systém, který ukládá, zpracovává a poskytuje webový obsah, jako jsou webové stránky, aplikace, obrázky a videa prostřednictvím internetu. Hraje klíčovou roli v modelu klient-server, kde klient (obvykle webový prohlížeč) požaduje obsah od webového serveru pomocí protokolu HTTP. Webový server poté odpovídá zasláním požadovaného obsahu, který je následně zobrazen v prohlížeči klienta. Webové servery nabízejí řadu funkcí podporujících jejich základní funkčnost, včetně protokolování souborů, autentizace a serverového webového skriptování. Mohou být také použity jako virtuální servery pro provozování více aplikací, webových stránek, dat a dalších služeb. [57]

8.1 Arduino a ESP32 jako webserver

Desky Arduino lze vybavit Ethernetovým štítem, čímž se promění na jednoduché webové servery. To umožňuje uživatelům přistupovat k webovému serveru Arduina pomocí webového prohlížeče na jakémkoli počítači připojeném k téže síti jako Arduino. Arduino může číst stav spínače, ovládat hardware z webové stránky a číst hodnoty ze senzorů, a to vše prostřednictvím jednoduchého webového rozhraní. [58]

ESP32 lze také použít jako webový server. Může fungovat ve třech režimech: režim stanice, režim přístupového bodu a oba režimy současně. V režimu stanice se ESP32 připojuje k existující Wi-Fi síti a nastavuje webový server, který poskytuje webové stránky všem připojeným zařízením na této síti. V režimu přístupového bodu ESP32 vytváří vlastní Wi-Fi síť a funguje jako hub, který poskytuje webové stránky všem připojeným zařízením. [59]

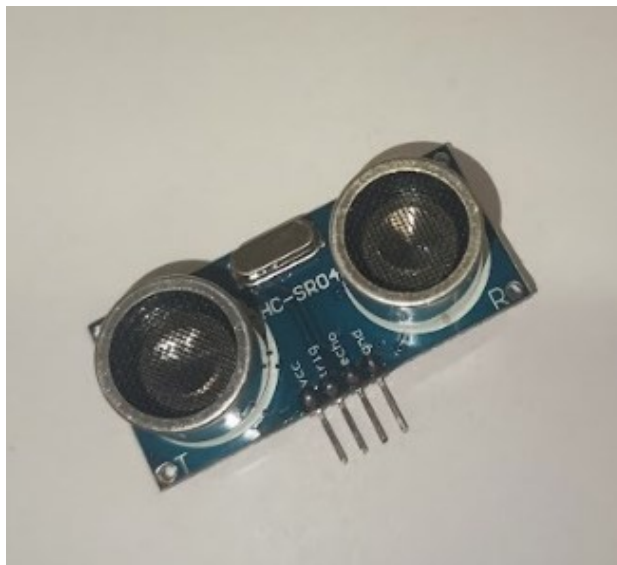
9 MODULY A SENZORY PRO MIKROKONTROLERY

V této kapitole se podíváme a rozebereme snímače a čidla, které by mohly být využity při konstrukci modelu posuvné brány. Nalezneme zde čidla pro detekci překážek, modul pro řízení stejnosměrných motorů, relé moduly a moduly pro měření napětí a proudu.

9.1 Ultrazvukový senzor

Ultrazvukové senzory používají k měření vzdálenosti zvukové vlny. Často je můžeme vidět v systémech vyhýbaní se překážkám, v robotice nebo třeba jako parkovací senzory u aut. Mezi populární senzory patří HC-SR04, HY-SRF05 nebo JSN-SR04T.

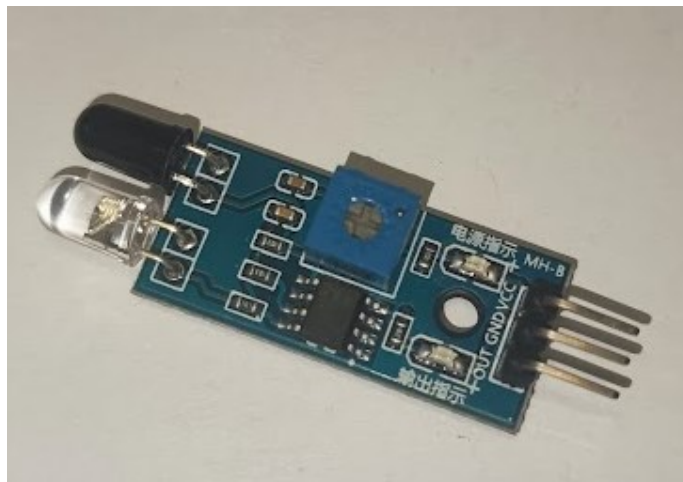
Senzor HC-SR04 má typický rozsah od 2 do 400 cm. Měření mimo tento rozsah je nespolehlivé. Teplota, vlhkost nebo proudění vzduchu ovlivní měření. Samozřejmě tady hraje i povrch, od kterého se vlny odrážejí. Ideální je, aby byl povrch rovný a kolmý k senzoru. Šikmé či jinak zdeformované povrchy mohou zvuk absorbovat nebo jej odrážet mimo senzor. [60]



Obrázek 36 Ultrazvukový senzor HC-SR04

9.2 Infračervený senzor

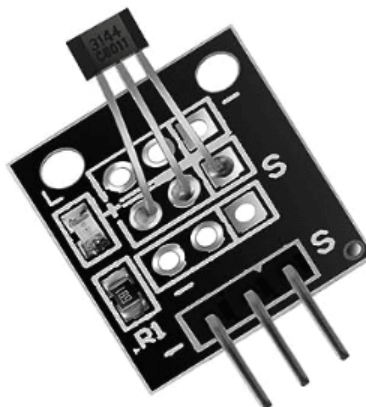
Jádrem tohoto senzoru je fototranzistor nebo fotodioda, která je citlivá na infračervené světlo. Citlivost je typicky v rozsahu 700 nm až 1 mm vlnové délky. Hlavním principem tohoto senzoru je, že detekuje a měří světlo, které se odráží od objektu. Senzor, který máme na obrázku níže může měřit vzdálenost od 2 do 30 cm. Spínací vzdálenost se může měnit s okolní teplotou nebo také podle odrazivosti snímaného objektu. [60]



Obrázek 37 Infračervené čidlo

9.3 Hallův senzor

Tyto senzory s Hallovým efektem, jako jsou magnetometry, mohou detekovat magnetické pole. Můžeme je najít uplatněné v úlohách jako je detekce rychlosti nebo snímání plochy. Připevněním například malého magnetu na rotující předmět lze Hallův senzor využít k měření otáček za minutu. Nicméně tyto senzory mohou být ovlivňovány vnějším elektromagnetickým rušením, tudíž je dobře zajistit vhodné umístění nebo stínění. [60]



Obrázek 38 Hallův senzor [60]

9.4 Senzor napětí

Senzory napětí stejnosměrného napětí fungují jako děliče napětí, které snižují vyšší napětí na úroveň bezpečnou pro vstup mikrokontroleru. Jejich užitečnost se rozšiřuje u aplikací, kde je potřeba například sledování stavu baterie a úrovně nabití. Pro měření vyšších napětí je vhodnější zvolit senzor s galvanickou nebo optickou izolací, aby byl mikrokontroler ochráněn před potenciálním poškozením. Možnou nevýhodou tohoto senzoru je, že může být ovlivněn elektromagnetickým šumem z blízkých vodičů. [60]

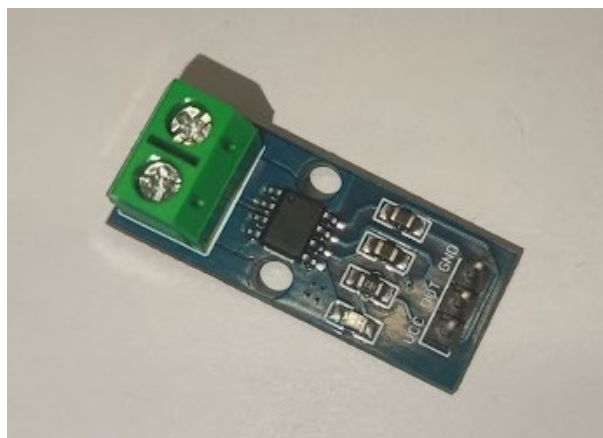


Obrázek 39 Senzor napětí

9.5 Proudový senzor

Hallův jev lze využít i pro měření toku elektrického proudu. Tento přístup nabízí výhodu zajištění elektrické izolace mezi měřeným obvodem a samotným senzorem. Proudové senzory nacházejí široké uplatnění při zabezpečení elektronických zařízení proti nadproudu a ke sledování úrovně spotřebované energie. Jako u napěťových senzoru, i zde mohou externí elektromagnetické ručení ovlivnit naměřené hodnoty. U senzoru ACS712 si lze vybrat z variant pro měření 5 A, 20 A nebo 30 A. Jiným proudový senzorem je INA219. Tento senzor

měří proud procházející přes nízko odporový bočníkový rezistor a vypočítává hodnotu proudu na základě úbytku napětí přes tento rezistor. [60]



Obrázek 40 Proudový senzor ACS712

9.6 Laserový snímač vzdálenosti

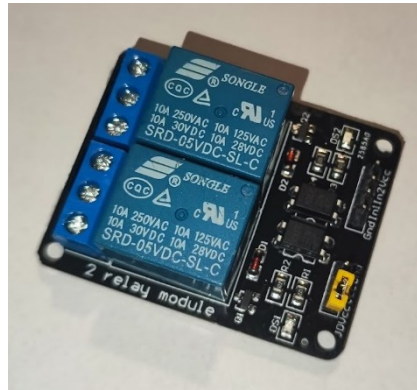
Stejně jako ultrazvukové senzory, laserový snímač vzdálenosti měří vzdálenost. Místo zvuku se v tomto případě využívá světla. Na rozdíl od ultrazvukových nabízejí lepší přesnost a rychlost měření. Jedním takovým senzorem je senzor VL53L0X od společnosti HiLetgo. Jedná se kompaktní a široce používaný čip pro měření času letu. Měří dobu trvání laserového pulzu, který se odrazí od cíle. I když laserové snímače nabízejí ve srovnání s ultrazvukovými mnohem lepší přesnost a odolnost, jsou citlivé na faktory okolního prostředí. Mezi tyto faktory patří třeba intenzita slunečního záření, prach nebo zařízení vyřazující infračervené záření. U objektů, které jsou tmavší barvy může dojít k absorpci vyzařovaného světla, což poté snižuje efektivní dosah snímače. [60]



Obrázek 41 Laserový snímače vzdálenosti VL53L0K

9.7 Reléový modul

Tyto moduly poskytují schopnost regulovat vysokonapěťová zařízení prostřednictvím nízkonapěťového digitálního výstupního pinu na mikrokontroleru. Moduly jsou zvláště užitečné pro ovládání lamp nebo jiných zařízení ve spojení s Arduinem. Tyto relé moduly jsou ve variantě jednoho kanálu, dvou nebo až čtyřech kanálů. [61]



Obrázek 42 Relé modul

9.8 Modul rotačního enkodéru

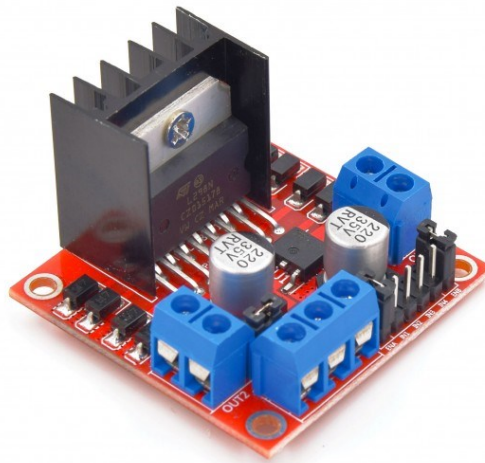
Rotační enkodéry poskytují polohový data ze snímače a umožňují přesné ovládání krokových a servo motorů. Toto je činí důležitou součástí v aplikacích vyžadujících přesné řízení pohybu. [61]



Obrázek 43 Rotační enkodér [61]

9.9 H-můstek

H-můstek je široce používán při regulaci rychlosti motoru a směru otáčení. Kromě řízení motoru nachází uplatnění při napájení různých zařízení, jako jsou pole LED, relé nebo solenoidy. Motorový ovladač typu L298N je vybaven robustním chladičem, dokáže efektivně ovládat motory v rozsahu od 5 do 25 V s maximálním proudem 2 A. Tento model je vybaven i vestavěným 5 V regulátorem schopným dodávat až 1 A. Je však důležité, pokud zařízení potřebuje více než 12 V, je vhodné tento 5 V regulovaný zdroj nepoužívat. [62]



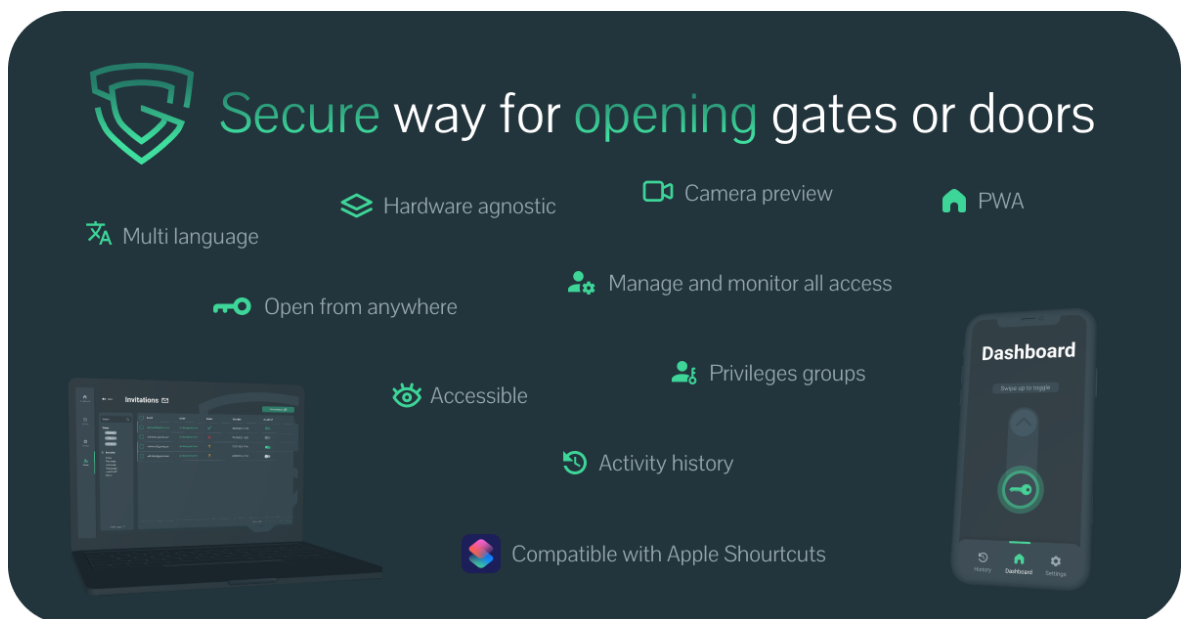
Obrázek 44 H-můstek [62]

10 OPEN SOURCE PROJEKTY

V této kapitole si rozebereme již vytvořené projekty pro brány či garážová vrata. Většina těchto projektů se nachází na platformě GitHub.

10.1 Smart Gate

Projekt Smart Gate je open-source systém navržený pro správu přístupu k fyzickým vstupním branám nebo dveřím a nabízí chytré ovládání stávajících elektrických zámků. Tento systém vytvořili Jakub Józwiak a Mateusz Nestorowicz. Pro implementaci Smart Gate si uživatelé musí připravit Raspberry Pi a začlenit 3 V reléový modul mezi Raspberry Pi a bránu pro vzdálené otevírání. Systém zjednodušuje nasazení tím, že poskytuje předdefinované šablony pro nastavení serveru, databáze a webové aplikace. Uživatelé mohou transformovat své tradiční brány nebo dveře na chytré systémy podle pokynů k nastavení, které zahrnují nasazení klientských, serverových a databázových instancí pomocí poskytnutých šablon. [63]

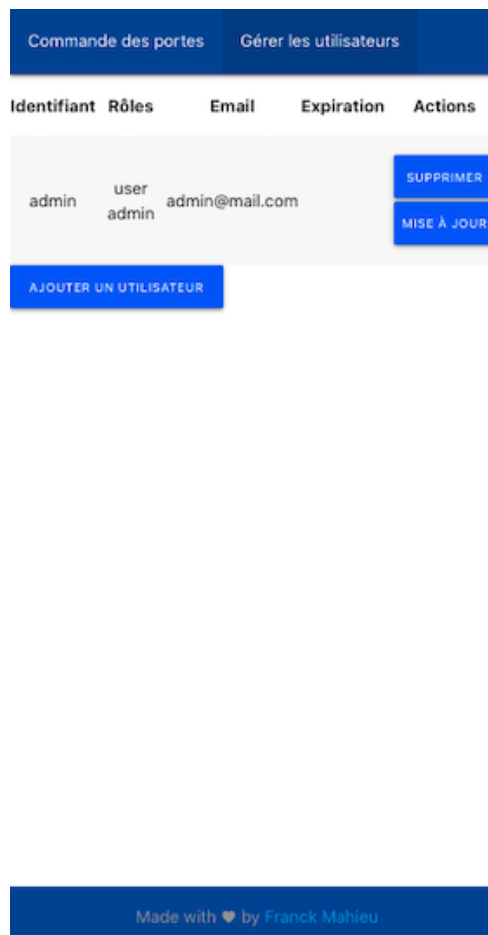


Obrázek 45 Smart Gate [63]

10.2 RPI Parking Doors

Jedná se o systém na bázi Raspberry Pi určený k ovládání vrat garáže pomocí webové aplikace. Systém využívá relé, které jsou připojena k Raspberry Pi a slouží k simulaci stisknutí vysílače brány a otevření dveří. V aplikaci nalezneme i stránku pro správu uživatelů, která umožňuje přidávat, mazat nebo upravovat hesla. Pro ověření uživatele se vygeneruje

jedinečný token a vloží se do adresy URL, což zajistí nepřerušovaný přístup bez opakovaného přihlašování. Dále v aplikaci nalezneme samotné tlačítko na otevření dveří. To po stisknutí aktivuje relé na 10 vteřin, které simuluje stisk tlačítka na ovladači a brána se otevře. Zatímco projekt upřednostňuje uživatelskou přívětivost před robustními bezpečnostními opatřeními, budoucí iterace plánují implementaci protokolu OAuth pro vylepšení bezpečnosti. [64]



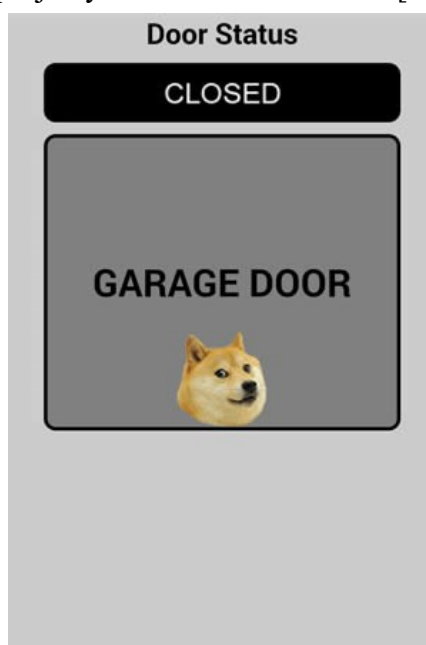
Obrázek 46 RPI-parking doors mobilní aplikace [64]

10.3 GaragePi

Tento projekt stejně jako již probrané funguje také na Raspberry Pi a ovládání pomocí relé. Systém je navržen tak, aby umožňoval uživatelům ovládat garážová vrata z pohodlí jejich gauče nebo na dálku prostřednictvím vhodného připojení VPN. Tento projekt je založen na aplikaci Flask napsané v Pythonu. Obsahuje JSON/jQuery pro udržování aktuálního stavu a Bootstrap pro uživatelské rozhraní front-endu. Je zde také externí bod GET API, který umožňuje jiným aplikacím otevřít nebo zavřít dveře bez ověření. Projekt je přepracované řešení od autora Chrise Driscolla, jehož projekt projdeme. [65]

10.4 Raspberry Pi Garage Door Opener

Projekt od autora Chrice Driscolla postavený na Raspberry Pi B+. Projekt započal roku 2014 a jeho poslední aktualizace byla v roce 2015. Projekt využívá Raspberry Pi, WebIOPi a otvírač garážových vrat. Raspberry Pi je připojeno k otvírač garážových vrat pomocí pinů GPIO, které WebIOPi ovládá. To umožňuje uživateli ovládat otvírač garážových vrat pomocí internetového připojení z libovolného zařízení. Pro nastavení projektu musí uživatel postupovat podle pokynů v příspěvku na blogu. To zahrnuje instalaci WebIOPi a zajištění, aby fungovaly v domovském adresáři na Raspberry Pi. Autor také poskytuje odkaz na soubor garage.html, který by měl být spuštěn z Raspberry Pi. Projekt je cenným příkladem toho, jak lze Raspberry Pi použít pro projekty domácí automatizace. [66]



Obrázek 47 Raspberry Pi Garage Door Opener [66]

10.5 Arduino gate opener

Zde již místo Raspberry Pi využíváme k otevírání brány Arduino. Projekt zahrnuje vytvoření webové stránky s tlačítkem, které po stisknutí odešle zprávu na server WebSocket. Server poté odešle pokyny servu na Arduino, které se otáčí, aby aktivovalo kliknutí na otvírač brány, což umožňuje vzdálené otevření brány. [67]

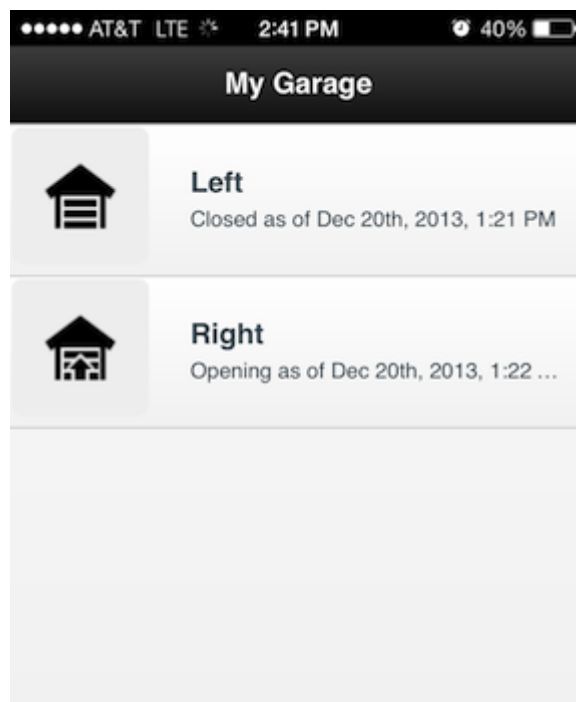
10.6 Wifi Garage Door Controller

Projekt je navržen tak, aby umožňoval aktivaci standardního otvírače garážových vrat pomocí chytrého telefonu a Google Assistant. Využívá mikrokontroler NodeMCU 1.0,

tranzistor 2n2222a, odpor 560 Ohm a další součástky. K řízení se zde používá aplikace Blynk. Tuto aplikaci si uživatel stáhne do telefonu, zaregistruje a poté si vytvoří nový projekt s NodeMCU vybraným jako hardware. Uživatel si poté nastaví widget tlačítka v aplikaci Blynk a přiřadí jej k fyzickému digitálnímu pinu. Následně si připojí desku k počítači, ve vhodném vývojovém prostředí doinstaluje potřebné knihovny. Následně si stáhne kód z repositáře GitHubu, který si překopíruje do vývojového prostředí a poté nahraje do mikrokontroleru. U tohoto projektu je i návod, kde je popsáno, jak přidat Google asistenta, aby podle hlasového příkazu aktivoval otevření brány. [68]

10.7 Garage Door Controller

Tento projekt je postaven na Raspberry Pi, ke kterému jsou připojeny magnetické kontakty a reléový modul. Magnetické kontakty slouží k indikaci, zda je brána otevřená nebo zavřená. Reléový modul je použit ve smyslu napodobení stisknutí tlačítka k otevření nebo zavření dveří. Software podporuje funkce, jako je sledování stavu garážových vrat, dálkové ovládání garážových vrat, časová razítka poslední změny stavu u každého dveří, protokolování veškeré aktivity garážových. Instalace je zde taktéž popsána a zahrnuje naklonování GitHub repositáře, instalaci potřebných knihoven, úpravu konfiguračního souboru, případného vytvoření SSL certifikátu a připojení spínačů a relé modulů. [69]



Obrázek 48 Garage Door Controller [69]

10.8 Automatic Gate opener

Součástí projektu je mikrokontroler Arduino Uno, který řídí ovladač motoru LM293D. Dále zde máme infračervený senzor, který detekuje předměty nebo překážky, propojovací kabeláž a zdroj. Funkce je zde velice jednoduchá. Když infračervený senzor detekuje předmět, Arduino spustí servomotor k otevření brány. Brána se po dvou vteřinách automaticky zavře, tento čas lze upravit v kódu. [70]

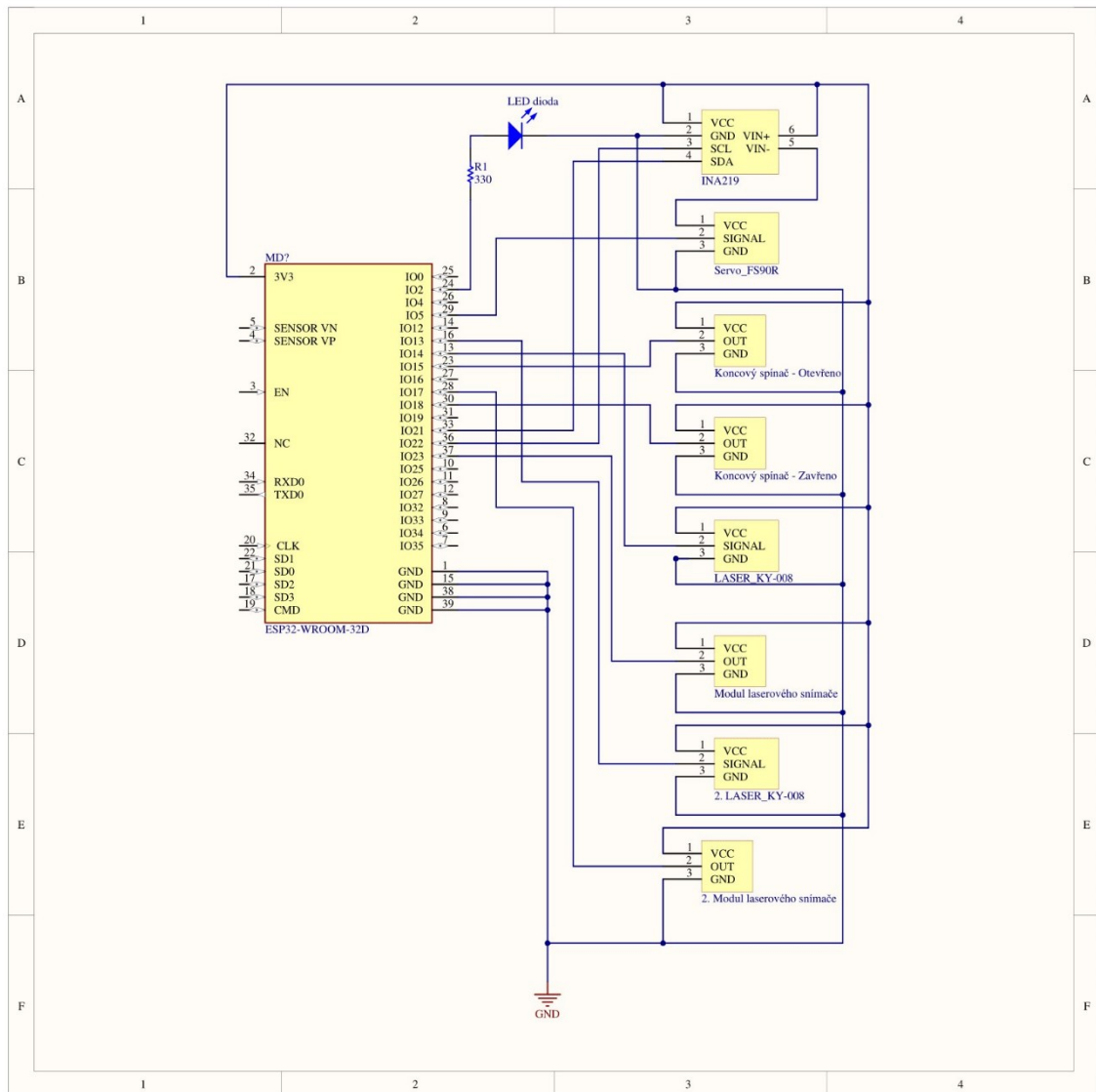
II. PRAKTICKÁ ČÁST

11 NÁVRH ARCHITEKTURY A REALIZACE ŘÍZENÍ BRÁNY

V této kapitole si rozebereme vybrané komponenty a samotnou realizaci modelu posuvné brány.

11.1 Schéma zapojení modelu posuvné brány

Pro vytvoření schématu zapojení byl použit software Altium Designer. Tento software se používá nejen na vytváření schémat, ale i na vytváření návrhu PCB. [55]



Obrázek 49 Schéma zapojení

Stručný popis návrhu modelu ovládání posuvné brány: k řízení posuvné brány byla použita platforma ESP32, na které běží webserver s tlačítky a informacemi o bráně.

Po zapnutí systému si uživatel ve webovém prohlížeči zadá adresu ESP32, která ho přesměruje na webovou stránku. Uživatel je vyzván, aby zadal přihlašovací údaje. Pokud zadá špatné údaje bude znovu vyzván k zadání údajů a pokud se udělá chybu pětkrát po sobě, systém zareaguje a na minutu zablokuje přihlášení. Jakmile zadá správné údaje, bude přesměrován na hlavní stránku pro ovládání brány. Na hlavní stránce uživatel uvidí dvě tlačítka, jedno na otevření brány, druhé na její zavření. Dále v pravém horním rohu uvidí tlačítko na odhlášení a vedle něj tlačítko na změnu motivu mezi tmavým a světlým motivem. Pod tlačítky ovládání brány nalezne základní informace o koncových spínačích, stavu brány a odběru motoru.

Když uživatel klikne na tlačítko „Otevřít bránu“, tak pokud je brána v zavřeném, tak poté se začne otevírat. Během otevírání začne blikat signalizační LED dioda, která bliká 500ms intervalem a na stránce se v kolonce „stav brány“ zobrazí nápis „Otevírání“. Jakmile brána dorazí na „koncový spínač – otevřeno“, motor se zastaví, LED dioda zhasne a na stránce se u „stavu brány“ zobrazí „Otevřeno“. Když uživatel opětovně stiskne tlačítko pro otevírání, nic se nestane, jelikož brána je již v otevřeném stavu.

Jakmile uživatel stiskne tlačítko „Zavřít bránu“, aktivuje se laser pro detekci překážek a poté se začne brána zavírat a na stránce se u „stavu brány“ změní nápis na „Zavírání“. Pokud je ale brána již v zavřeném stavu, nic se nestane. Když se brána zavírá tak se taktéž rozbliká LED dioda s intervalem 500ms. Pokud přerušíme paprsek laseru, který dopadá na senzor po dobu větší jak 200ms, brána se zastaví, u „stavu brány“ se zobrazí „Detekována překážka – otevírání“ a po dalších 200ms se brána začne otevírat. Poté se brána otevírá, dokud nenajede na koncový spínač pro otevřený stav. Dále se nám zobrazí na stránce u popisku „Otevřený koncový spínač“ zobrazí „Aktivován“.

Při otevírání brány i zavírání se kontroluje velikost odebíraného proudu u servo motoru. Ochranný mechanismus funguje tak, že pokud průměrný proud, který se skládá z deseti hodnot, dostane nad hranici 130 mA po dobu větší jak 400ms, tak motor se vypne a u popisku „stav brány“ se zobrazí „Přetížení motoru – zastaveno“.

Když se uživatel odhlásí, bude přesměrován na stránku, která mu sdělí, že se úspěšně odhlásil a pokud klikne na hypertextový odkaz v tomto textu, bude přesměrován na opětovné přihlášení.

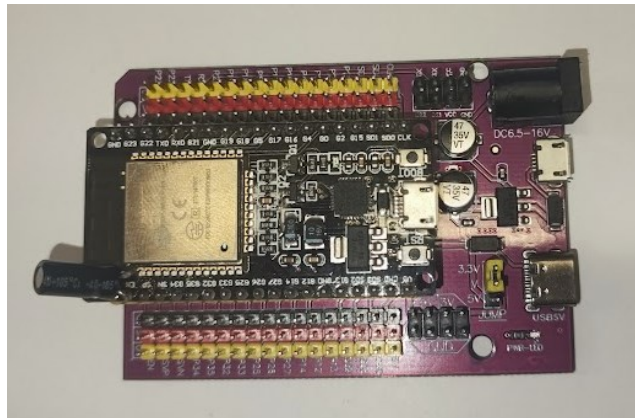
11.2 Vybrané komponenty

V této kapitole si rozebereme použité komponenty pro model posuvné brány, některé z nich již byly popsány v kapitole 9 „Moduly a senzory pro mikrokontrolery“. U součástek se nachází i fotografie a popis.

11.2.1 ESP32 DevKitC

Při výběru platformy pro realizaci projektu jsem se rozhodl pro desku ESP32 DevKitC. Jedná se o výkonnější desku oproti populárním Arduino deskám. Na rozdíl od Arduina, které má nižší taktovací frekvenci a méně RAM, ESP32 nabízí rychlejší zpracování dat a větší flexibilitu díky integrovanému Bluetooth a Wi-Fi. Ve srovnání s Raspberry Pi 4, které je sice výkonnější, ale má vyšší spotřebu energie a vyžaduje složitější nastavení pro IoT aplikace, ESP32 DevKitC poskytuje jednodušší implementaci s nižší spotřebou energie, což je klíčové pro trvalé nasazení v terénu. Deska disponuje 36 digitálními vstupy/výstupy a podporou pro I2C komunikaci, což usnadňuje rozšíření o další senzory. Jedním z těchto senzorů je proudový senzor INA219, který také použijeme při realizaci. Na první pohled je zřejmé, že se nejedná jen o desku ESP32, ale je zde pár součástek navíc. První součástkou je připájený 10uF kondenzátor mezi piny EN a GND. Tato modifikace byla přidána, jelikož deska nechtěla automaticky přejít do nahrávacího režimu při nahrávání nového kódu. Bez tohoto kondenzátoru se při nahrávání nového kódu muselo držet tlačítko BOOT, které je na desce hned vedle napájecího USB portu. Druhou součástkou je rozšiřující deska, která usnadní zapojení ostatních komponent. Na rozšiřující desce jsou umístěny tři napájecí konektory, zemnicí lišta a napájecí lišta, která umožňuje volbu mezi napětím 3.3 V nebo 5 V. Kromě toho jsou na desce přítomny signální piny, které odpovídají každému pinu na modulu ESP32, což umožňuje efektivní distribuci signálů a napájení pro připojené komponenty a senzory. Jedním z takových komponent může být třeba servo motor, který si rozebereme dále.

Máme zde i signální led diodu a speciální piny na 5 V, 3.3 V a GND linku, poté i SDA, SCL, TX a RX. [71]



Obrázek 50 ESP32 DevKitC s rozšiřující deskou

11.2.2 Servo motor FS90R

Jako pohon pro modelovou bránu se použilo servo FS90R. Jedná se výstupní součástku. Většinou se tyto servo motory používají u RC modelů na otáčení kol, kde se otáčejí v rozsahu 0° až 180° . Nicméně v tomto případě servo FS90R má kontinuální chod a může se tedy točit na obě strany bez pevně daného rozsahu. Napájecí napětí servo motoru je v rozsahu od 4,8 V do 6 V. U servo motoru najdeme i kabel, který má tři vývody. Jedním je napájení, dalším je uzemnění a poslední je signální vodič, který bude zapojen na pinu 5. [72]



Obrázek 51 Servo motor FS90R

11.2.3 Snímač proudu

Jelikož bylo potřeba přesnější měření proudu s vyšší citlivostí, byl pro projekt použit proudový senzor INA219 místo ACS712. Senzor ACS712, ačkoliv běžně používaný pro měření proudu v širším rozsahu, má nižší citlivost, což jej činí nevhodným u použití s vybraným servo motorem, kde je potřeba rozlišit malé změny proudu, jako je odchylka 60 mA. Naopak INA219 je vysoce přesný proudový senzor. Umožňuje měřit stejnosměrný proud až do výše 3,2 ampérů, a to jak v kladném, tak záporném směru. Měřicí metoda tohoto modulu je založena na principu detekce úbytku napětí na bočníkovém rezistoru s odporovou hodnotou 0,1 ohmu. Tento způsob měření umožňuje přesně určit malé změny proudu, které ACS712 nedokáže rozlišit. Senzor také komunikuje pomocí rozhraní I2C, takže bude stačit připojit senzor k napájení, SCL a SDA. [73]

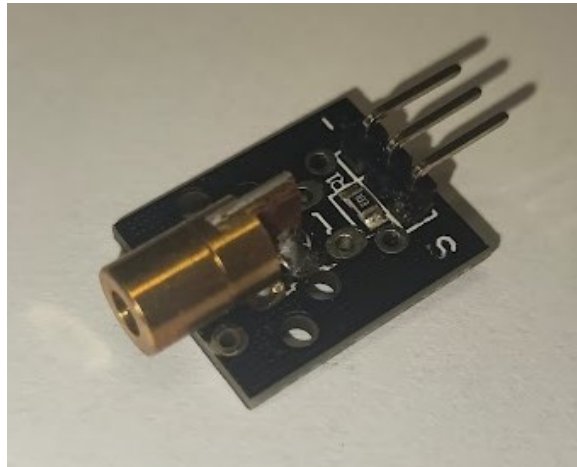


Obrázek 52 Proudový senzor INA219

11.2.4 Laser KY-008

Modul KY-008 se skládá z malého červeného laserového diodu, který emituje světlo o vlnové délce přibližně 650nm. Jeho hlavní předností je schopnost generovat úzký a vysoce zaměřený laserový paprsek. Na opačné straně tento paprsek bude snímat senzor a pokud se paprsek jakýmkoliv způsobem přeruší na dobu 200ms, brána zareaguje na detekci překážky. Napájecí napětí modulu KY-008 je typicky mezi 3,3 V a 5 V, což jej činí kompatibilním s většinou mikrokontrolerů, včetně Arduino a Raspberry Pi. Modul je vybaven třemi vývody:

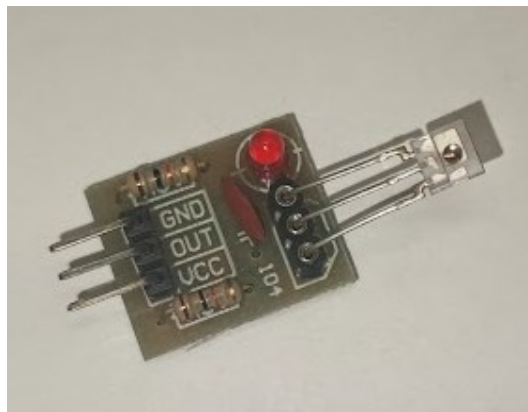
napájecím, zemním a signálním. Signální pin se používá k aktivaci laseru prostřednictvím digitálního výstupu mikrokontroleru, který připojíme na pin 14. [74]



Obrázek 53 Laser KY-008

11.2.5 Modul laserového snímače

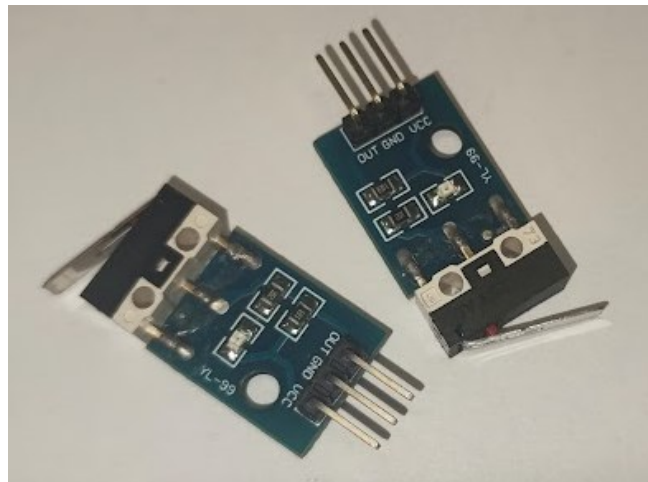
Napájení modulu je typicky v rozmezí 3,3 V až 5 V. Snímač je citlivý na červené světlo a je nejcitlivější při vlnové délce kolem 760nm. Snímač poskytuje digitální výstup, kdy logická 1 signalizuje detekci laseru. Senzor obsahuje tři vývody: napájení, uzemnění a výstupní pin, který bude připojen na pin 23 na ESP32. [75]



Obrázek 54 Modul laserového snímače

11.2.6 Koncové spínače

Pro indikaci otevření nebo zavření brány byly zvoleny mechanické koncové spínače kvůli jejich jednoduchosti. Na modulu koncového spínače nalezneme i diodu, která se rozsvítí, jakmile se spínač sepne. Koncový spínač pro signalizaci stavu otevřené brány bude připojen na pin 15 a spínač pro signalizaci stavu zavřené brány na pinu 18. [76]



Obrázek 55 Mechanické koncové spínače

11.2.7 Ostatní součástky

U modelu brány jsem taktéž použil LED diodu pro vizuální signalizaci, kdy se brána otevírá a zavírá. LED dioda byla zapojena tak, že její anoda byla připojena k výstupnímu pinu s označením 2 na mikrokontroleru přes rezistor o hodnotě 330 ohmů. Použití tohoto rezistoru bylo nezbytné pro omezení proudu procházejícího LED diodou, aby nedošlo k jejímu poškození způsobeném příliš vysokým proudem. Katoda LED diody byla následně připojena na uzemnění. Propojovací kabely byly použity k fyzickému spojení jednotlivých komponent.



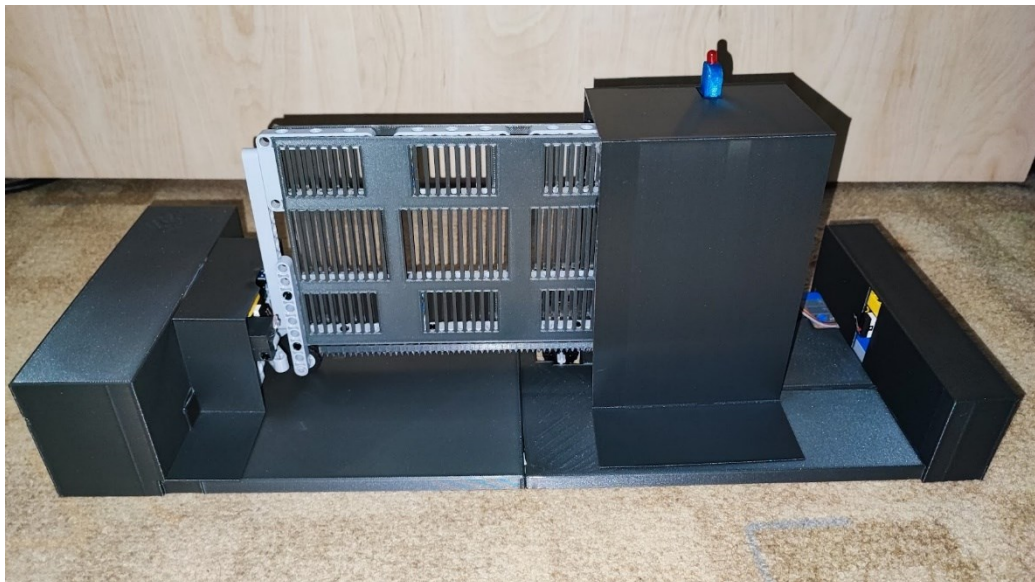
Obrázek 56 LED dioda, rezistor a propojovací kabely

11.2.8 Napájení modelu

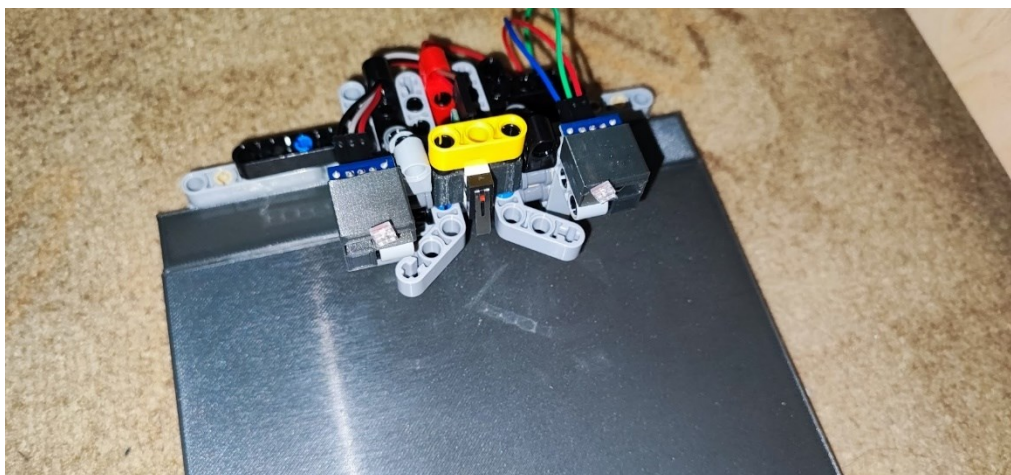
K napájení modelu jsem využil powerbanku. Ta byla připojena s rozšiřující deskou ESP32 pomocí micro USB kabelu. Rozšiřující deska má tři různé konektory, kterými lze napájet: micro USB, USB-C a DC konektor, přes který jde desku napájet až 16 V zdrojem.

11.3 Realizace HW zapojení řídicího systému a celého modelu brány

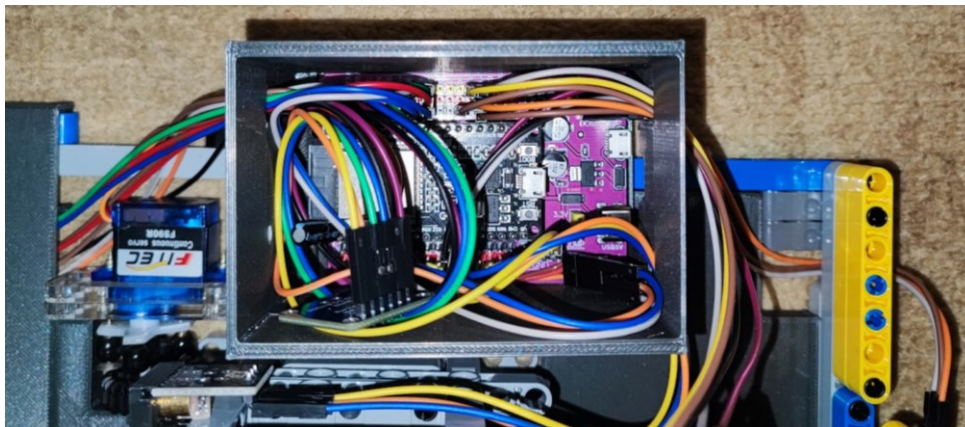
Samotná kostra posuvné brány byla vytvořena ze stavebnice Lego Technic. Krycí díly byly vymodelovány v programu Fusion 360 a poté vytisknuty na 3D tiskárně. Dílky na přichycení senzorů, laseru a koncových spínačů byly taktéž vytvořeny v programu Fusion 360 a následně byly vytisknuty na 3D tiskárně.



Obrázek 57 Přední strana modelu



Obrázek 58 Uchycení čidel pro laser



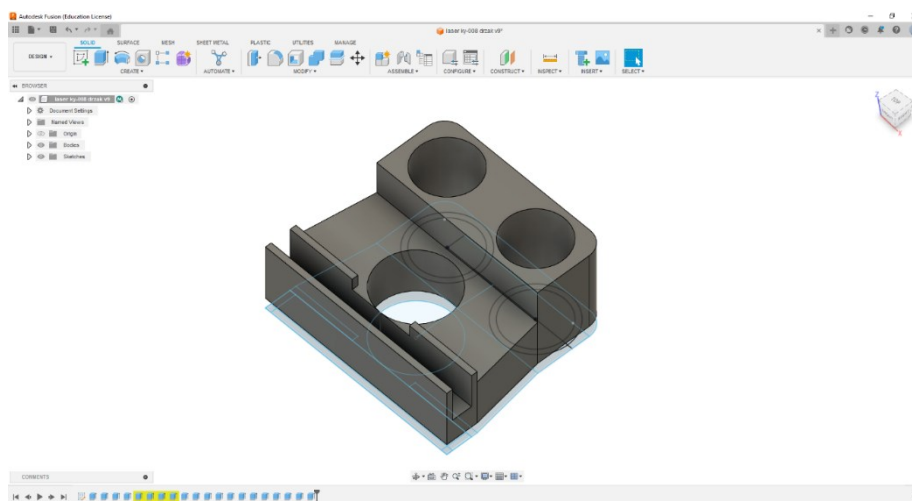
Obrázek 59 Krabička pro ES32 a kabeláž

Uvnitř krabičky nalezneme samotné ESP32 vložené do rozšiřující desky, do které jsou zapojené všechny komponenty. V krabičce jsou i otvory na připojení USB kabelu pro napájení.

11.3.1 3D model

Jak již bylo zmíněno, tak dílky pro přichycení laserů, čidel a koncových byly vytvořeny v programu Autodesk Fusion 360. [56]

Pro modelování jsem si změřil potřebné součástky jako jsou otvory a velikost lego kostky a samotné čidlo. Tyto naměřené hodnoty jsem poté přenesl do 2D modelu. Poté z tohoto 2D modelu jsem pomocí nástroje „Extrude“ potřebné části vytáhl a tím nám vzniknul 3D model.

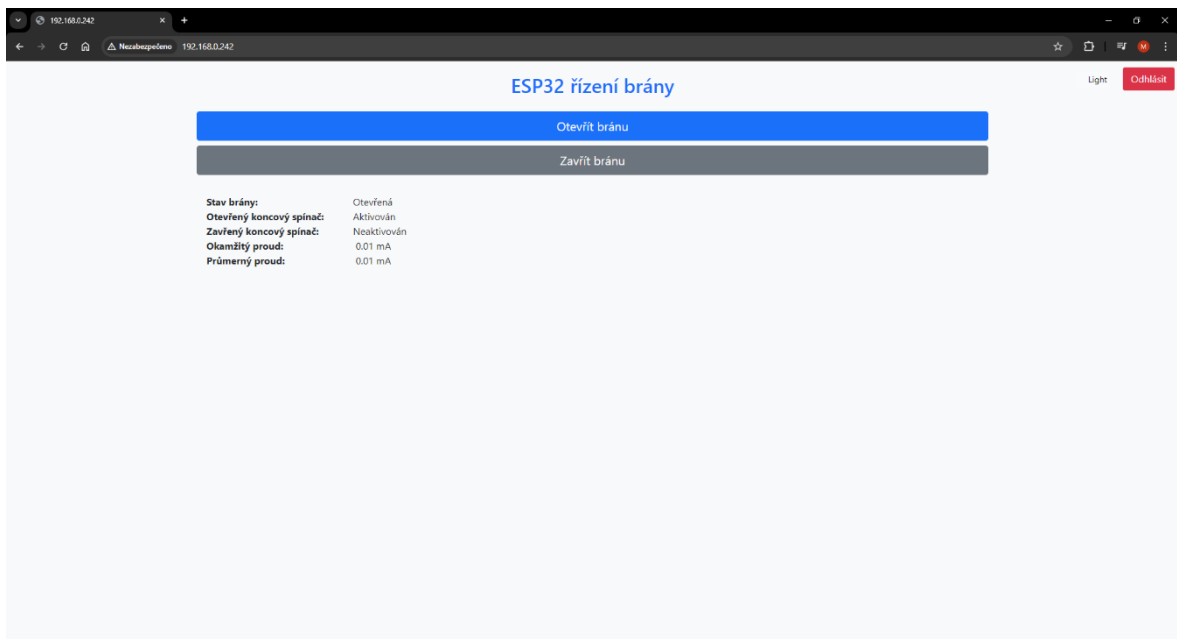


Obrázek 60 Fusion 360 model pro přichycení laseru

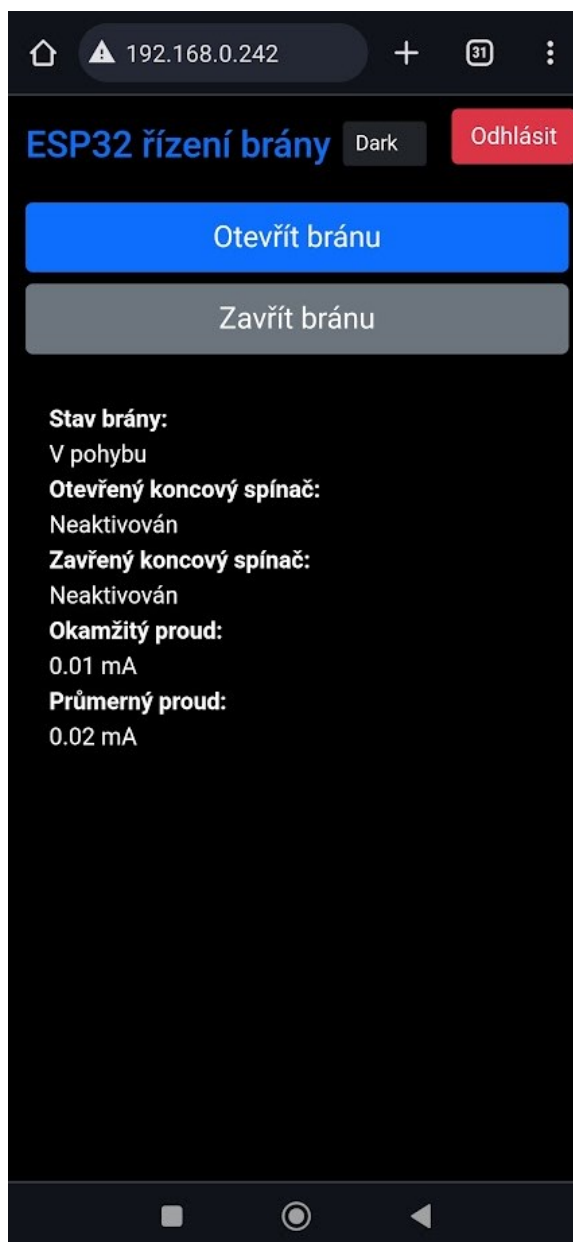
Podobným způsobem jsem postupoval i při modelování zbytku součástek použitých v modelu. U 3D tisku jsem si musel dát pozor na přesnost tiskárny, jelikož u pár modelů byly otvory pro spojovací lego dílky o kousek menší než v načrtnutém modelu. Tuto chybu jsem opravil malým zvětšením díry v náčrtu.

11.4 Webová stránka

V horní části stránky je zobrazen název "ESP32 řízení brány", což naznačuje účel stránky. Hlavička také obsahuje tlačítka pro otevření a zavření brány, což jsou primární akce, které mohou uživatelé vykonávat. V pravém horním rohu je přepínač režimu "Light" pro změnu tématu stránky na světlý nebo tmavý režim a tlačítko "Odhlásit" pro odhlášení uživatele ze systému. Pod hlavními ovládacími tlačítky pak nalezneme aktuální informace o stavu brány jako je: stav brány, stav otevřených koncových spínačů, průměrný a okamžitý proud.



Obrázek 61 Webová stránka – zobrazení na počítači



Obrázek 62 webová stránka – zobrazení na telefonu

Zde můžeme vidět zobrazení webové stránky pro telefon. Na stránce se využívá responzivního designu s použitím Bootstrap komponent.

12 OVLÁDACÍ SW MODELU BRÁNY

V této kapitole si projdeme jednotlivé části kódu, které si popíšeme.

12.1 Popis zdrojového kódu

Než se spustí hlavní konfigurační smyčka `void setup`, je nezbytné importovat potřebné knihovny pro ovládání posuvné brány. To zahrnuje knihovny pro ovládání servo motoru, senzorů a Wi-Fi komunikaci. Je nutné nastavit konkrétní GPIO piny pro ovládání LED diod, servo motoru a senzorů, jako je laserový modul a jeho senzor.

Dále ve smyčce `void setup` konfigurujeme výstupní piny pro LED diodu, která indikují pohyb brány a servo motor, který fyzicky pohybuje branou. Nastavuje se také laserový modul pro detekci překážek v dráze brány. Dále se inicializuje Wi-Fi připojení a mDNS služby pro snadný přístup k webovému rozhraní brány.

Hlavní smyčka `void loop` zahrnuje logiku pro ovládání pohybu brány. Kontroluje se stav koncových spínačů, aby se zjistilo, zda je brána plně otevřená nebo zavřená. Při detekci překážky laserovým modulem se brána zastaví a pokusí se vrátit do bezpečné polohy. Systém také monitoruje přetížení motoru pomocí proudu měřeného senzorem INA219 a reaguje na možné problémy.

`AsyncWebServer` spuštěný na portu 80 poskytuje uživatelské rozhraní, kde lze ovládat otevření a zavření brány, zobrazovat její stav a sledovat proudové hodnoty. Uživatelé se mohou autentizovat pomocí uživatelského jména a hesla, a při opakovaném neúspěšném zadání hesla dojde k dočasnému zablokování přihlášení.

Webové rozhraní využívá moderní CSS a JavaScript pro zlepšení uživatelské zkušenosti, včetně responzivního designu a tmavého režimu, který lze přepínat uživatelem. JavaScriptové funkce zajišťují dynamické aktualizace stavu brány a senzorů bez nutnosti obnovování stránky.

12.1.1 Import knihoven

Začínáme importem esenciálních knihoven pro ovládání hardwaru jako jsou `ESP32Servo` pro ovládání servo motorů, `ArduinoJson` pro práci s JSON objekty, `Adafruit_INA219` pro měření elektrického proudu, a sítové knihovny pro Wi-Fi a webový server.

```
#include <ESP32Servo.h>
#include <ArduinoJson.h>
#include <Adafruit_INA219.h>
#include <WiFi.h>
#include <AsyncTCP.h>
#include <ESPAsyncWebServer.h>
#include <ESPmDNS.h>
```

Obrázek 63 Import knihoven

12.1.2 Nastavení připojení

Definujeme konstanty pro SSID a heslo Wi-Fi, které jsou základem pro připojení k síti. Poté si nastavíme i přihlašovací údaje pro přihlášení na webserveru.

```
// Nastavení přístupových údajů k WiFi
const char *ssid = "BP_brana";
const char *password = "BranaESP32*";

// Přístupové údaje pro HTTP server
const char *http_username = "brana";
const char *http_password = "espbrana";
```

Obrázek 64 Nastavení přístupových údajů

12.1.3 Definice pinů

V této části kódu jsme si definovali piny pro LED diodu, servo, koncové spínače, lasery a senzory laseru.

```
// Definice pinů pro různé součástky
const int ledOpenPin = 2;           // pin pro LED
const int servoPin = 5;             // pin pro ovládání servomotoru
const int openEndSwitchPin = 15;    // pin pro koncový spínač otevření
const int closeEndSwitchPin = 18;   // pin pro koncový spínač zavření
const int laserModulePin = 14;      // pin pro laserový modul
const int laserSensorPin = 23;      // pin pro laserový senzor
const int laserModulePin2 = 13;     // pin pro laserový modul 2
const int laserSensorPin2 = 17;     // pin pro laserový senzor 2
```

Obrázek 65 Definice pinů

12.1.4 Nastavení pozic motoru a blikání LED

Zde definujeme pozice pro ovládání brány servomotorem a nastavujeme intervaly pro blikání LED, v počátečním stavu nastavujeme stav LED diody na zhasnuto.

```
// Pozice servomotoru
const int servoOpenPos = 75;      // Pozice pro otevření brány
const int servoClosedPos = 105;   // Pozice pro zavření brány
const int servoStopPos = 90;     // Pozice pro zastavení serva

// Interval blikání (v milisekundách)
unsigned long previousMillis = 0; // Ukládá poslední čas aktualizace LED
const long blinkInterval = 500;  // Interval blikání LED
bool ledState = LOW;
```

Obrázek 66 Nastavení pozic servomotoru a blikání LED

12.1.5 Webserver

Na portu 80 je inicializován asynchronní web server, což umožňuje dálkové ovládání nebo monitorování systému přes síť.

```
AsynchWebServer server(80);
```

Obrázek 67 Asynchronní webserver

12.1.6 Monitorování a průběh proudu

Implementujeme zde sledování a průměrování proudu s cílem detekovat možné problémy s motorem, jako je přetížení. Při překročení definovaných limitů dojde k automatickému zastavení motoru.

```
// Proměnné pro průměrování proudu
const int numReadings = 10;
float readings[numReadings]; // Pole pro ukládání hodnot proudu
int readIndex = 0;           // Index aktuálního čtení
float total = 0;             // Běžný celkový součet
float average = 0;          // Průměrná hodnota proudu
float current_mA = 0;       // Aktuální hodnota proudu v mA

const unsigned long overcurrentTimeThreshold = 400; // Doba trvání přetížení v milisekundách
unsigned long overcurrentStartTime = 0;           // Čas zahájení přetížení
bool isOvercurrent = false;                       // Příznak stavu přetížení
bool MotorOver = false;
```

Obrázek 68 Monitorování proudu

12.1.7 Nastavení počtu pokusů u přihlášení

Pokud uživatel zadá, v tomto případě pětkrát špatné jméno nebo heslo, bude na 60 s zablokován a nebude se moci během této doby přihlásit.

```
// Počet pokusů pro vstup správného hesla
const int maxAttempts = 5;
int attemptCounter = 0;
unsigned long lockoutStartTime;
bool isLockedOut = false;
const unsigned long lockoutDuration = 60000; // Doba uzamčení na 1 minutu
```

Obrázek 69 Počet pokusů přihlášení

12.1.8 Uživatelské rozhraní

Stránka umožňuje uživatelům otevření a zavření brány, zobrazuje aktuální stav a měření proudu. Dále jsou zde definovány i styly pro celkový vzhled stránky a skripty pro aktualizace stavu brány, koncových spínačů a hodnoty proudu.

```
.btn-dark-mode:hover {
  background-color: #343a40; /* Tmavě šedá barva při najetí myši */
}
</style>
</head>
<body>
  <div class="container">
    <div class="header">
      <h2 class="text-center text-primary mb-4">ESP32 řízení brány</h2>
      <div class="toggle-and-logout">
        <input type="checkbox" data-toggle="toggle" data-on="Dark" data-off="Light" data-onstyle="dark" data-offstyle="light" data-size="small" id="dark-mode-toggle">
        <button class="btn btn-danger logout-button" onclick="logoutButton()">Odhlásit</button>
      </div>
    </div>
    <div class="d-grid gap-2">
      <button class="btn btn-lg btn-primary" onclick="openGate()">Otevřít bránu</button>
      <button class="btn btn-lg btn-secondary" onclick="closeGate()">Zavřít bránu</button>
    </div>
  </div>
```

Obrázek 70 Nastavení stylů a tlačítek

12.1.9 Funkce pro blikání LED diody

Jakmile uživatel klikne na tlačítko „Otevřít bránu“ nebo „Zavřít bránu“, brána se uvede do pohybu a během tohoto pohybu začne blikat i LED dioda v intervalu 500 ms, který jsme si definovali na začátku. Nepoužívá se zde funkce zpoždění, nýbrž funkce, která získá aktuální čas v milisekundách. Ten se pak porovnává, zda uplynul dostatečný čas od posledního bliknutí. Pokud je rozdíl větší nebo roven, pak nastal čas pro bliknutí LED diody.

```
// Funkce pro blikání LED
void blinkLED(int pin)
{
  unsigned long currentMillis = millis();
  if (currentMillis - previousMillis >= blinkInterval)
  {
    previousMillis = currentMillis; // uloží poslední čas blikání LED
    ledState = !ledState;          // změní stav LED
    digitalWrite(pin, ledState);   // nastaví LED podle stavu proměnné
  }
}
```

Obrázek 71 Funkce pro blikání

12.1.10 Nastavení součástek

Na začátku funkce `void setup()` inicializujeme součástky, nastavujeme senzory jako výstup, inicializujeme senzor proudu s nastavením kalibrace, spouštíme připojení k Wi-Fi a inicializujeme mDNS.

```
void setup()
{
    // Spustí sériový port pro ladění
    Serial.begin(115200);

    // Inicializace servomotoru
    gateServo.attach(servoPin);

    // Nastavení pinu pro laserový modul jako výstup
    pinMode(laserModulePin, OUTPUT);
    pinMode(laserModulePin2, OUTPUT);

    //Vypne laserový modul na začátku
    digitalWrite(laserModulePin, LOW);
    digitalWrite(laserModulePin2, LOW);

    pinMode(ledOpenPin, OUTPUT);

    // Připojení k Wi-Fi
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED)
    {
        delay(1000);
        Serial.println("Connecting to WiFi..");
    }

    // Vytiskne lokální IP adresu ESP
    Serial.println(WiFi.localIP());
}
```

Obrázek 72 Inicializace součástek

12.1.11 Definice chování webserveru

V kódu je definováno několik funkcí pro různé účely, které jsou implementovány jako obslužné metody pro specifické HTTP GET požadavky na webovém serveru. Každá z těchto funkcí slouží k poskytování různých informací nebo ke správě různých aspektů brány. Pro příklad jsme se zaměřili na konkrétní funkce, z nichž jedna poskytuje informace o stavu koncových spínačů a druhá pro chování hlavní stránky.

12.1.12 Definice chování webserveru pro hlavní stránku

Tato část kódu definuje chování webového serveru pro hlavní cestu `"/`, která je přístupná pomocí HTTP GET požadavku. Nejdříve se ověřuje, zda je systém uzamčen kvůli opakovanému zadání nesprávného hesla. Pokud ano a od posledního neúspěšného pokusu neuplynula definovaná doba, požadavek je zamítnut s chybovým kódem 403 a zobrazí se zpráva informující uživatele o uzamčení. Pokud systém není uzamčen, pokračuje se v autentizaci

uživatele pomocí zadaného uživatelského jména a hesla. Pokud autentizace selže, inkrementuje se počítadlo pokusů o přihlášení. Po překročení maximálního počtu pokusů o přihlášení

```
server.on("/", HTTP_GET, [(AsyncWebServerRequest *request){
if(isLockedOut && (millis() - lockoutStartTime) < lockoutDuration) {
// Pokud je systém uzamčený, odešle chybovou zprávu, aby uživatel věděl, že je uzamčený
request->send(403, "text/plain", "Zadali jste 5x špatné heslo. Zkuste to znovu za chvíli.");
} else {
isLockedOut = false; // Resetovali jsme lockout
if(!request->authenticate(http_username, http_password)) {
attemptCounter++;
if(attemptCounter >= maxAttempts) {
// Příliš mnoho špatných pokusů, uzamknutí systému
isLockedOut = true;
lockoutStartTime = millis();
attemptCounter = 0; // resetuje počítadlo pokusů
}
return request->requestAuthentication();
} else {
attemptCounter = 0; // vynulujte počítadlo pokusů po úspěšném přihlášení
// Pokračuje s odesláním stránky
request->send_P(200, "text/html", index_html, processor);
}
} });
```

Obrázek 73 Chování webserveru pro hlavní stránku

je systém uzamčen na určitou dobu, aby se zabránilo dalším pokusům o přihlášení. Tato metoda lze považovat za lehkou ochranu proti útokům brute force. Po úspěšné autentizaci se počítadlo pokusů vynuluje a server odesílá klientovi hlavní HTML stránku, která je zpracována pomocí funkce processor k dynamickému generování obsahu stránky na základě aktuálního stavu systému.

12.1.13 Definice chování webserveru při změně stavu koncového spínače

Zde máme kód, který zajišťuje odpověď na HTTP GET požadavek na cestě "/gatestatus" na webovém serveru a poskytuje aktuální stav brány ve formátu JSON. Nejprve je inicializován JSON dokument, který bude použit k uložení stavových informací o bráně. Stav brány je určen na základě vstupů z koncových spínačů a interních proměnných systému. Kód zjišťuje, zda jsou koncové spínače aktivní a zda jsou detekovány specifické situace, jako je blokáce laserem nebo přetížení motoru. Na základě toho je do JSON dokumentu zapsán příslušný stav. Po nastavení stavu brány je JSON dokument serializován do řetězce a tento řetězec je poslán jako odpověď na požadavek klienta s HTTP status kódem 200, čímž se informuje klient o aktuálním stavu brány.

```
server.on("/gatestatus", HTTP_GET, [(AsyncWebServerRequest *request) {
    StaticJsonDocument<64> jsonDoc; // Vytvoření JSON dokumentu pro stav brány
    // Nastavení stavu brány v JSON objektu
    if (digitalRead(openEndSwitchPin)==LOW && digitalRead(closeEndSwitchPin)==HIGH) {
        jsonDoc["gateStatus"] = "Otevřená";
    } else if (digitalRead(closeEndSwitchPin)==LOW && digitalRead(openEndSwitchPin)==HIGH) {
        jsonDoc["gateStatus"] = "Zavřená";
    } else if (laserBlocked == true) {
        jsonDoc["gateStatus"] = "Detekována překážka - otevírání";
    } else if (MotorOver == true) {
        jsonDoc["gateStatus"] = "Přetížení motoru - Zastaveno";
    } else if (opening == true) {
        jsonDoc["gateStatus"] = "Otevírání";
    } else if (closing == true) {
        jsonDoc["gateStatus"] = "Zavírání";
    } else {
        jsonDoc["gateStatus"] = "V pohybu";
    }
    String response;
    serializeJson(jsonDoc, response); // Serializace JSON objektu do řetězce
    request->send(200, "application/json", response); // Odeslání JSON odpovědi
});
```

Obrázek 74 Chování webserveru při změna stavu koncových spínačů

12.1.14 Zavírání brány a kontrola přerušení laseru

Tato část kódu popisuje proces zavírání brány s použitím servomotoru a bezpečnostních funkcí zahrnujících laserové senzory. Jakmile uživatel klikne na stránce na tlačítko „Zavřít bránu“, proměnná „closing“ se nastaví na „true“. Blikající LED indikuje, že brána se právě zavírá. Laserové moduly jsou zapnuty pro detekci překážek v průběhu zavírání. Pokud je koncový spínač pro zavření aktivován, servomotor je nastaven do pozice zastavení, což signalizuje, že brána je zcela zavřená. Laserové moduly jsou vypnuty a proces zavírání končí. Během zavírání se průběžně kontroluje, zda není laserový paprsek přerušen překážkou. Pokud ano, zaznamená se čas zjištění přerušení a zkontroluje se, zda přerušení trvá déle než 200 ms. Jestliže ano, proces zavírání je přerušen a brána se začne opět otevírat, aby se zabránilo potenciálnímu nebezpečí nebo poškození. Po detekci blokace je brána znovu otevřena a čeká se, dokud se úplně neotevře. Po dokončení opětovného otevření se zastaví všechny procesy, laserové moduly jsou vypnuty a stav „laserBlocked“ je resetován na „false“.

```

if (closing)
{
    blinkLED(ledOpenPin); // Blikání LED při zavírání
    digitalWrite(laserModulePin, HIGH); // Zapnutí laseru
    digitalWrite(laserModulePin2, HIGH); // Zapnutí laseru 2
    gateServo.write(servoClosedPos); // Nastavení serva do pozice pro zavření

    if (digitalRead(closeEndSwitchPin) == LOW)
    {
        gateServo.write(servoStopPos); // Zastavení serva
        digitalWrite(laserModulePin, LOW); // Vypnutí laseru
        digitalWrite(laserModulePin2, LOW); // Vypnutí laseru 2
        closing = false; // Ukončení procesu zavírání
        digitalWrite(ledOpenPin, LOW); // Vypnutí LED
    }
    else
    {
        // Kontrola přerušení laseru
        if (digitalRead(laserSensorPin) == LOW || digitalRead(laserSensorPin2) == LOW)
        {
            gateServo.write(servoStopPos); // Zastavení serva
            unsigned long startTime = millis(); // Zaznamenání času začátku přerušení

            // Kontrola, zda je laser stále přerušen
            while (digitalRead(laserSensorPin) == LOW || digitalRead(laserSensorPin2) == LOW)
            {
                if (millis() - startTime >= 200)
                {
                    laserBlocked = true; // Označení, že laser je blokován
                    break; // Přerušení smyčky
                }
            }

            // Reakce na přerušení laseru
            if (laserBlocked)
            {
                delay(200); // Krátká pauza před znovuotevřením brány
                gateServo.write(servoOpenPos); // znovuotevření brány
                blinkLED(ledOpenPin); // Blikání LED
            }
        }
    }

    // Čekání na úplné otevření brány v případě přerušení laseru
    if (laserBlocked)
    {
        while (digitalRead(openEndSwitchPin) != LOW)
        {
            // Čekání na otevření brány
        }
        closing = false;
        gateServo.write(servoStopPos); // Zastavení serva
        laserBlocked = false;
        digitalWrite(laserModulePin, LOW); // Vypnutí laseru
        digitalWrite(laserModulePin2, LOW); // Vypnutí laseru 2
        digitalWrite(ledOpenPin, LOW); // Vypnutí LED
    }
}
}

```

Obrázek 75 Zavírání brány a kontrola přerušení laseru

12.1.15 Kontrola přetížení motoru

Kód neustále kontroluje, zda průměrná hodnota proudu překročila nastavený limit 130 mA. Pokud je přetížení zaznamenáno a trvá déle než 400 ms, je nastavená proměnná „MotorOver“ na „true“ a podle této proměnné se nám pak ve stavu brány zobrazí, že byl motor přetížen a brána je zastavena. Dále zde máme vypnutí motoru a LED diody, nastavení proměnných pro otevření a zavření na hodnotu „false“ a poté i vypnutí laseru, kdyby se tato ochrana aktivovala při zavírání brány.

```
if (average > 130)
{
    if (!isOvercurrent)
    {
        overcurrentStartTime = millis(); // Začátek sledování přetížení
        isOvercurrent = true; // Nastavení příznaku přetížení
    }
    else if (millis() - overcurrentStartTime > overcurrentTimeThreshold)
    {
        MotorOver = true; // Označení trvalého přetížení
        gateServo.write(servoStopPos); // Zastavení serva
        digitalWrite(ledOpenPin, LOW); // Vypnutí LED
        opening = false; // Zastavení jakýchkoli pokusů o otevření
        closing = false; // Zastavení jakýchkoli pokusů o zavření
        digitalWrite(laserModulePin, LOW); // Vypnutí laseru
        digitalWrite(laserModulePin2, LOW); // Vypnutí laseru 2
    }
}
else
{
    isOvercurrent = false; // Reset příznaku přetížení, pokud není detekováno
}
```

Obrázek 76 kontrola přetížení servo motoru

ZÁVĚR

Tato bakalářská práce se zabývala návrhem a realizací systému pro vzdálené ovládání a monitoring posuvné brány s využitím mikrokontroleru ESP32. Hlavním cílem bylo vytvořit funkční a efektivní systém, který by integroval moderní technologické prvky a byl by přístupný pro široké spektrum uživatelů díky open-source platformám.

V teoretické části byly detailně prozkoumány a popsány různé typy bran a pohonů, stejně jako funkce řídicích jednotek. Tato část také zahrnuje komplexní srovnání různých řídicích platforem, popis existujících projektů s podobnou tematikou a diskusi o různých čidlech vhodných pro mikrokontrolery. Dále je představen přehled programů použitých při realizaci projektu.

Praktická část práce se zaměřila na návrh a realizaci systému pro řízení posuvné brány s využitím mikrokontroleru ESP32, který umožňuje vzdálené ovládání a monitoring brány. Klíčovým prvkem systému bylo vytvoření webového serveru běžícího na ESP32, což umožnilo snadné ovládání brány prostřednictvím webového rozhraní dostupného na mobilních zařízeních a počítačích.

Pro detekci koncových bodů, překážek a zatížení motoru byly vybrány a testovány specifické senzory. Důležitou součástí systému bylo také zpracování a integrace signálů z těchto senzorů, které zajišťují bezpečné a efektivní fungování brány.

Vývoj ovládacího softwaru zahrnoval programování webového serveru, který poskytuje intuitivní uživatelské rozhraní pro kontrolu a monitoring brány. Toto rozhraní umožňuje uživatelům pohodlně monitorovat stav brány a ovládat ji v reálném čase.

Tato práce demonstruje, jak lze s využitím mikrokontroleru ESP32 a integrovaného webového serveru efektivně řešit potřeby automatizace v oblasti posuvných bran.

SEZNAM POUŽITÉ LITERATURY

- [1] Druhy vjezdových bran. Online. 2024. Dostupné z: <https://www.dops.cz/novinky-s-foto/branky-a-brany-vse-dulezite-co-o-nich-potrebujete-vedet>. [cit. 2024-03-15].
- [2] Jednokřídle a dvoukřídle brány. Online. 2024. Dostupné z: <https://www.brany-vrata-ploty.cz/vjezdove-brany/kridlove>. [cit. 2024-03-15].
- [3] Jednokřídla brána. Online. 2024. Dostupné z: <https://www.peritusgroup.cz/menu/jednokridle>. [cit. 2024-03-15].
- [4] Skládací brána. Online. 2024. Dostupné z: <https://www.plotyodoty.cz/produkty/brany-a-branky/brana-skladaci>. [cit. 2024-03-16].
- [5] Skládací brána. Online. 2024. Dostupné z: <https://www.ho-pa.cz/reference/garazova-vrata-a-brany-kovovyroba-3/brany-104/kridlova-skladaci-brana-nepomuk-204>. [cit. 2024-03-16].
- [6] Typy posuvných bran. Online. 2024. Dostupné z: <https://www.brany-vrata-ploty.cz/posuvne-brany>. [cit. 2024-03-17].
- [7] Posuvné brány. Online. 2024. Dostupné z: <https://www.esako.cz/posuvne-brany--1>. [cit. 2024-03-19].
- [8] Posuvná brána samonosná. Online. 2024. Dostupné z: <https://www.svarmont-kovo.cz/cs/cena-samosnosne-posuvne-brany.html>. [cit. 2024-03-20].
- [9] Posuvná brána pojezdová. Online. 2024. Dostupné z: <https://www.svarmont-kovo.cz/cs/cena-pojezdove-posuvne-brany.html>. [cit. 2024-03-20].
- [10] Teleskopické brány. Online. 2024. Dostupné z: <https://www.herzen.cz/brany-a-ploty/teleskopicke-brany/>. [cit. 2024-03-21].
- [11] Řídící jednotka. Online. 2024. Dostupné z: <https://www.flbrany.cz/blog/jak-spravne-namontovat-pohon-na-posuvnou-branu-a-lze-to-zvladnout-svepomoci>. [cit. 2024-03-24].
- [12] KC868-A4 HARDWARE DESIGN DETAILS. Online. 2022. Dostupné z: <https://www.kincony.com/kc868-a4-hardware-design-details.html>. [cit. 2024-05-02].
- [13] Pohony bran. Online. 2024. Dostupné z: <https://www.gateautomation-supplies.co.uk/gate-automation-supplies-blog/sliding-gate-motor>. [cit. 2024-03-24].

- [14] Pohon posuvné brány. Online. 2024. Dostupné z: <https://www.goletto.cz/naradi/motor-pohonu-pro-posuvna-vrata#&gid=1&pid=1>. [cit. 2024-03-25].
- [15] Pohon posuvné brány s baterií. Online. 2024. Dostupné z: https://www.azpohony.cz/ln432km-pohon-moovo-pro-posuvnou-branu-do-hmotnosti-300kg/p368?v=10368&utm_source=google&utm_medium=gmc&gad_source=1. [cit. 2024-03-30].
- [16] Motor 230V a 24V. Online. 2024. Dostupné z: <https://umbrashading.co.uk/24v-vs-230v-roller-blinds-which-motor-is-right-for-you/>. [cit. 2024-03-30].
- [17] Bezkartáčové motory. Online. 2024. Dostupné z: <https://www.parvalux.com/advantages-of-brushless-motors/>. [cit. 2024-03-30].
- [18] Typy motorů. Online. 2024. Dostupné z: <https://www.epohony.cz/pohony-bran-a-vrat/pohony-posuvnych-bran/prumyslove-pohony/kit-bh30-810-r-roger-brushless-pohony-posuvnych-bran-do-800-1000kg~p652>. [cit. 2024-04-5].
- [19] Pohony pákové, přímočaré a podzemní. Online. 2024. Dostupné z: <https://www.ipohony.cz/mohlo-by-vas-zajimat/spravny-vyber-pohonu-pro-kridlovou-branu/>. [cit. 2024-04-05].
- [20] Koncové spínače. Online. 2024. Dostupné z: <https://navody.dratek.cz/technikuv-blog/o-koncovem-spinaci.html>. [cit. 2024-04-05].
- [21] Mechanický koncový spínač. Online. 2024. Dostupné z: <https://www.ipohony.cz/nice-splsm00100-mechanicky-koncovy-spinac-pohonu-brany-nice-a-mhouse/>. [cit. 2024-04-06].
- [22] Magnetický snímače. Online. 2024. Dostupné z: <https://automatizace.hw.cz/bezpecnost-stroju-komponenty/magneticke-bezpecnostni-spinace-pro-dvere-a-kryty.html>. [cit. 2024-04-06].
- [23] Magnetický koncový spínač. Online. 2024. Dostupné z: <https://branyposuvne.sk/na-posuvnu-branu/bezpecnostny-spinac>. [cit. 2024-04-06].
- [24] Indukční snímač. Online. 2024. Dostupné z: <https://www.cncshop.cz/ips-12-indukcni-snimac-m12x1-kabel-2m/pro1500.html>. [cit. 2024-04-06].
- [25] Ultrazvukový snímač. Online. 2024. Dostupné z: https://www.pepperl-fuchs.com/czech_republic/cs/24176.htm. [cit. 2024-04-06].
- [26] Fotobuňky - jsou důležité? Online. 2024. Dostupné z: <https://www.ipohony.cz/mohlo-by-vas-zajimat/fotobunky-jsou-dulezite/>. [cit. 2024-04-06].

- [27] Fotobuňka. Online. 2024. Dostupné z: <https://www.azpohony.cz/slovník-pojmu-fotobunka/t1442>. [cit. 2024-04-06].
- [28] Maják - proč ho montovat? Online. 2024. Dostupné z: <https://www.i-pohony.cz/mohlo-by-vas-zajimat/majak-proc-ho-montovat/>. [cit. 2024-04-06].
- [29] Průvodce výběrem vhodného majáku jako výstražného světla. Online. 2024. Dostupné z: <https://www.autoprofi.cz/blog/pruvodce-vyberem-vhodneho-majaku-jako-vystrazneho-svetla-na-auto>. [cit. 2024-04-06].
- [30] Co je to mikrokontrolér? Online. 2024. Dostupné z: <https://blog.zonepi.cz/maker-uno-ve-vyuce-1-5-co-je-to-mikrokontroler/>. [cit. 2024-03-30].
- [31] ARDUINO, ČILI MIKROKONTROLÉRY PRO KAŽDÉHO. Online. 2024. Dostupné z: <https://www.tme.eu/cz/news/library-articles/page/42541/arduino-cili-mikrokontrolery-pro-kazdeho/>. [cit. 2024-03-30].
- [32] CO JE TO ARDUINO? Online. 2024. Dostupné z: <https://bastlirna.hwkitchen.cz/co-je-to-arduino/>. [cit. 2024-03-30].
- [33] Arduino Uno Rev3. Online. 2024. Dostupné z: <https://store.arduino.cc/products/arduino-uno-rev3>. [cit. 2024-03-30].
- [34] Arduino Micro. Online. 2024. Dostupné z: <https://store.arduino.cc/products/arduino-micro>. [cit. 2024-03-29].
- [35] Arduino Due. Online. 2024. Dostupné z: <https://store-usa.arduino.cc/products/arduino-due>. [cit. 2024-03-29].
- [36] MARTINEK, Radislav. Senzory v průmyslové praxi. Praha: BEN - technická literatura, 2004. ISBN 8073001144.
- [37] CAMERON, Neil. Electronics projects with the ESP8266 and ESP32 : building web pages, applications, and Wifi enabled devices. Berkeley: CA : Apress L. P., 2021. ISBN 9781484263365.
- [38] Espressif Systems. Online. 2024. Dostupné z: <https://www.espressif.com/en/company/about-us/ceo-letter>. [cit. 2024-03-30].
- [39] ESP32. Online. 2024. Dostupné z: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf. [cit. 2024-03-29].
- [40] ESP8266. Online. 2024. Dostupné z: https://www.espressif.com/sites/default/files/documentation/esp8266-technical_reference_en.pdf. [cit. 2024-03-29].

- [41] Modul NodeMCU WiFi ESP8266 - 4 MB. Online. 2024. Dostupné z: <https://bo-tland.cz/stazene-produkty/15136-modul-nodemcu-wifi-esp8266-4-mb-waveshare-16608-5904422343002.html>. [cit. 2024-04-05].
- [42] The Story of the Raspberry Pi. Online. 2024. Dostupné z: <https://raspberrypi.com/raspberry-pi-history/>. [cit. 2024-04-05].
- [43] Raspberry Pi 4 Model B. Online. 2024. Dostupné z: <https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-product-brief.pdf>. [cit. 2024-04-07].
- [44] Raspberry Pi Pico. Online. 2024. Dostupné z: <https://datasheets.raspberrypi.com/pico/pico-product-brief.pdf>. [cit. 2024-04-07].
- [45] STMicroelectronics. Online. 2024. Dostupné z: <https://investors.st.com/investor-resources/faqs>. [cit. 2024-04-14].
- [46] NUCLEO-L432KC. Online. 2024. Dostupné z: <https://os.mbed.com/platforms/ST-Nucleo-L432KC/>. [cit. 2024-04-14].
- [47] STMicroelectronics NUCLEO-L432KC STM32 Nucleo-32 Development Board. Online. 2024. Dostupné z: <https://cz.mouser.com/new/stmicroelectronics/stm-nucleo-l432kc-board/>. [cit. 2024-04-14].
- [48] NUCLEO-WB55RG. Online. 2024. Dostupné z: <https://os.mbed.com/platforms/ST-Nucleo-WB55RG/>. [cit. 2024-05-02].
- [49] ST Nucleo F103RB. Online. 2024. Dostupné z: https://www.st.com/resource/en/user_manual/um1724-stm32-nucleo64-boards-mb1136-stmicroelectronics.pdf. [cit. 2024-04-14].
- [50] ARM vs. AVR Microcontrollers: Understanding ARM and AVR Microcontroller Differences. Online. 2024. Dostupné z: <https://www.programming-boss.com/2023/09/arm-vs-avr-microcontrollers.html?m=1#gsc.tab=0>. [cit. 2024-04-15].
- [51] ESP32 vs ESP8266. Online. 2024. Dostupné z: <https://makeradvisor.com/esp32-vs-esp8266/>. [cit. 2024-04-15].
- [52] Raspberry Pi Pico vs ESP32. Online. 2024. Dostupné z: <https://linuxhint.com/raspberry-pi-pico-vs-esp32/>. [cit. 2024-04-15].
- [53] ESP32 for IoT: A Complete Guide. Online. 2024. Dostupné z: <https://www.na-bto.com/guide-to-iot-esp-32/>. [cit. 2024-04-15].

- [54] VS Code and PlatformIO IDE for ESP32 and ESP8266. Online. 2024. Dostupné z: <https://randomnerdtutorials.com/vs-code-platformio-ide-esp32-esp8266-arduino/>. [cit. 2024-04-25].
- [55] Altium Designer. Online. 2024. Dostupné z: <https://www.altium.com/altium-designer>. [cit. 2024-04-25].
- [56] Autodesk Fusion 360. Online. 2024. Dostupné z: <https://www.fusion360.cz/>. [cit. 2024-04-25].
- [57] Web Server. Online. 2024. Dostupné z: <https://www.wix.com/encyclopedia/definition/web-server>. [cit. 2024-04-25].
- [58] How to Use an Arduino as a Web Server. Online. 2024. Dostupné z: <https://www.allaboutcircuits.com/projects/using-an-arduino-as-a-web-server/>. [cit. 2024-04-24].
- [59] ESP32 Web Server. Online. 2024. Dostupné z: <https://lastminuteengineers.com/creating-esp32-web-server-arduino-ide/>. [cit. 2024-04-24].
- [60] Sensors for Arduino and other MCUs. Online. 2024. Dostupné z: <https://www.markerguides.com/sensors-for-arduino-overview/>. [cit. 2024-04-23].
- [61] Arduino Sensors and Modules. Online. 2024. Dostupné z: <https://www.thegeekpub.com/wiki/list-of-arduino-sensors-and-modules/>. [cit. 2024-04-23].
- [62] DUAL H-BRIDGE MOTOR DRIVER L298N. Online. 2024. Dostupné z: <https://www.smart-prototyping.com/L298N-Dual-H-bridge-Motor-Driver-Board>. [cit. 2024-04-24].
- [63] Smart Gate. Online. 2023. Dostupné z: <https://smart-gate-docs.vercel.app/docs/intro>. [cit. 2024-04-20].
- [64] Rpi-parking-doors. Online. 2021. Dostupné z: <https://github.com/franck-mahieu/rpi-parking-doors>. [cit. 2024-04-20].
- [65] GaragePi. Online. 2016. Dostupné z: <https://github.com/Responseless/GaragePi>. [cit. 2024-04-20].
- [66] Guide to a Raspberry Pi Garage Door Opener. Online. 2014. Dostupné z: <https://www.driscocity.com/idiots-guide-to-a-raspberry-pi-garage-door-opener/>. [cit. 2024-04-20].
- [67] Arduino-gate-opener. Online. 2021. Dostupné z: <https://github.com/kpabellan/arduino-gate-opener>. [cit. 2024-04-20].

- [68] Wifi-Garage-Door-Controller. Online. 2024. Dostupné z: <https://github.com/hector6298/Wifi-Garage-Door-Controller>. [cit. 2024-04-20].
- [69] Garage-door-controller. Online. 2024. Dostupné z: <https://github.com/andre-wshilliday/garage-door-controller>. [cit. 2024-04-20].
- [70] Automatic Gate opener using arduino and IR Sensor. Online. 2024. Dostupné z: <https://www.hackster.io/hrbhadrappa/automatic-gate-opener-using-arduino-and-ir-sensor-2906e6>. [cit. 2024-04-20].
- [71] Vývojeová deska ESP32-DevKitC 38pin. Online. 2024. Dostupné z: <https://dratek.cz/arduino/51547-esp32-devkitc-development-board-38pin.html>. [cit. 2024-05-02].
- [72] Mini 360stupňové servo FS90R s nepřetržitou rotací. Online. 2024. Dostupné z: <https://rpishop.cz/servomotory/1292-kitronik-mini-360stupnove-servo-fs90r-s-nepretrzitou-rotaci.html>. [cit. 2024-05-02].
- [73] Proudový snímač INA219. Online. 2024. Dostupné z: <https://www.hadex.cz/m439d-proudovy-snimac-ina219/>. [cit. 2024-05-02].
- [74] Laser 5V/650nm s optikou - modul KY-008. Online. 2024. Dostupné z: <https://www.hadex.cz/m498-laser-5v650nm-s-optikou----modul-ky-008---cerveny/>. [cit. 2024-05-02].
- [75] Modul laserového snímače. Online. 2024. Dostupné z: <https://www.laskakit.cz/arduino-modul-laseroveho-snimace/>. [cit. 2024-05-02].
- [76] Koncový doraz - spínač 2A 125VAC. Online. 2024. Dostupné z: <https://www.laskakit.cz/koncovy-doraz-spinac--2a-125vac/>. [cit. 2024-05-02].

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

cm	Centimetr
V	Volt
kHz	Kilohertz
EPROM	Erasable Programmable Read-Only Memory
RAM	Radnom Access Memory
A/D	Analog to digital
PWM	Pulse Width Modulation
USB	Universal Serial Bus
MHz	Megahertz
ARM	Advanced RISC Machine
PCIe	Peripheral Component Interconnect
MB	Megabyte
SPI	Seriál Peripheral Interface
UART	Universal Asynchronous Receiver-Transmitter
I2C	Inter-Integrated Circuit
Wi-Fi	Wireless Fidelity
Mbit/s	Megabits per second
GPIO	General Purpose Input/Output
GHz	Gigahertz
SDRAM	Synchronous Dynamic Random-Access Memory
HDMI	High-Definition Multimedia Interface
RF	Radio Frequency
IEEE	Institute of Electrical and Electronics Engineers
PHY	Physical Layer

MAC	Media Access Control
KB	Kilobyte
SRAM	Static Random-Access Memory
ADC	Analog to Digital Converter
IDE	Integrated Development Environment
CPU	Central Processing Unit
2D	Two-dimensional
3D	Three-dimensional
CAD	Computer-Aided Design
CAM	Computer-Aided Manufacturing
CAE	Computer-Aided Engineering
CNC	Computer Numeric Control
HTTP	Hypertext Transfer Protocol
nm	Nanometr
mm	Milimetr
A	Ampér
LED	Light Emitting Diode
OAuth	Open Authorization
URL	Uniform Resource Locator
JSON	JavaScript Object Notation
API	Application Programming Interface
VPN	Virtual Private Network
ms	Milisekunda
mA	Miliampér
IoT	Internet of Things
EN	Enable

GND	Ground
SDA	Serial Data
SCL	Serial Clock
TX	Transmitter
RX	Receiver
RC	Remote Control
DC	Direct Current
mDNS	Multicast Domain Name System

SEZNAM OBRÁZKŮ

Obrázek 1 Jednokřídlá brána [3].....	11
Obrázek 2 Otevírací rádius jednokřídlé brány	12
Obrázek 3 Dvoukřídlá brána [2]	12
Obrázek 4 Otevírací rádius dvoukřídlé brány	13
Obrázek 5 Křídlová skládací brána [5]	13
Obrázek 6 Posuvná brána samonosná [8]	14
Obrázek 7 Posuvná brána pojezdová [9]	15
Obrázek 8 Posuvná brána teleskopická [10].....	15
Obrázek 9 Řídící jednotka KC868-A4 [12].....	16
Obrázek 10 Pohon posuvných vrat [14]	17
Obrázek 11 Šneková převodovka [18].....	18
Obrázek 12 Typy motorů [18]	19
Obrázek 13 Přímočarý pohon [19].....	19
Obrázek 14 Pákový pohon [19]	20
Obrázek 15 Podzemní pohon [19]	20
Obrázek 16 Mechanický koncový spínač [21]	21
Obrázek 17 Magnetický koncový spínač [23]	22
Obrázek 18 Indukční snímač [24].....	22
Obrázek 19 Ultrazvukový senzor [25].....	23
Obrázek 20 Fotobuňka s odrazkou [26].....	24
Obrázek 21 Výstražný maják na bráně [28]	24
Obrázek 22 Arduino UNO rev3 [33]	26
Obrázek 23 Arduino Micro [34]	27
Obrázek 24 Arduino Due [35]	27
Obrázek 25 ESP32 DevKitC.....	28
Obrázek 26 ESP8266 [41]	28
Obrázek 27 Raspberry Pi 4B	29
Obrázek 28 Raspberry Pi Pico [44]	30
Obrázek 29 STM32L432KC.....	30
Obrázek 30 NUCLEO-WB55RG [48].....	31
Obrázek 31 Nucleo-F103RB	31
Obrázek 32 Arduino IDE.....	34

Obrázek 33 Visual Studio Code.....	35
Obrázek 34 Prostředí Altium Designer.....	36
Obrázek 35 Prostředí Fusion 360.....	37
Obrázek 36 Ultrazvukový senzor HC-SR04.....	39
Obrázek 37 Infračervené čidlo.....	40
Obrázek 38 Hallův senzor [60].....	40
Obrázek 39 Senzor napětí.....	41
Obrázek 40 Proudový senzor ACS712.....	42
Obrázek 41 Laserový snímače vzdálenosti VL53L0K.....	42
Obrázek 42 Relé modul.....	43
Obrázek 43 Rotační enkodér [61].....	43
Obrázek 44 H-můstek [62].....	44
Obrázek 45 Smart Gate [63].....	45
Obrázek 46 RPI-parking doors mobilní aplikace [64].....	46
Obrázek 47 Raspberry Pi Garage Door Opener [66].....	47
Obrázek 48 Garage Door Controller [69].....	48
Obrázek 56 Schéma zapojení.....	51
Obrázek 49 ESP32 DevKitC s rozšiřující deskou.....	54
Obrázek 50 Servo motor FS90R.....	54
Obrázek 51 Proudový senzor INA219.....	55
Obrázek 52 Laser KY-008.....	56
Obrázek 53 Modul laserového snímače.....	56
Obrázek 54 Mechanické koncové spínače.....	57
Obrázek 55 LED dioda, rezistor a propojovací kabely.....	57
Obrázek 57 Přední strana modelu.....	58
Obrázek 58 Uchycení čidel pro laser.....	58
Obrázek 59 Krabice pro ES32 a kabeláž.....	59
Obrázek 60 Fusion 360 model pro přichycení laseru.....	59
Obrázek 61 Webová stránka – zobrazení na počítači.....	60
Obrázek 62 webová stránka – zobrazení na telefonu.....	61
Obrázek 63 Import knihoven.....	63
Obrázek 64 Nastavení přístupových údajů.....	63
Obrázek 65 Definice pinů.....	63

Obrázek 66 Nastavení pozic servomotoru a blikání LED	64
Obrázek 67 Asynchronní webserver	64
Obrázek 68 Monitorování proudu.....	64
Obrázek 69 Počet pokusů přihlášení.....	65
Obrázek 70 Nastavení stylů a tlačítek.....	65
Obrázek 71 Funkce pro blikání.....	65
Obrázek 72 Inicializace součástek	66
Obrázek 73 Chování webserveru pro hlavní stránku.....	67
Obrázek 74 Chování webserveru při změna stavu koncových spínačů.....	68
Obrázek 75 Zavírání brány a kontrola přerušeni laseru.....	69
Obrázek 76 kontrola přetížení servo motoru	70

SEZNAM PŘÍLOH

PŘÍLOHA P I – CD

PŘÍLOHA P I: CD

Zdrojový kód pro ESP32

- ESP32_rizeni_brany.ino

3D modely

- Drzak_laseru.m3f
- Drzak_senzoru_laseru_levy.3mf
- Drzak_senzoru_laseru_pravy.3mf
- Drzak_koncovy_spinac_open.3mf
- Drzak_koncovy_spinac_closed.3mf
- Drzak_LED.3mf
- Krabicka_ESP32.3mf
- Krytk_krabicky.3mf
- Predni_dil_brany.3mf
- Zadni_dil_brany.3mf
- Podlaha_pod_branu_1.3mf
- Podlaha_pod_branu_2.3mf
- Podlaha_pod_branu_3.3mf
- Podlaha_pod_branu_4.3mf
- Kryt_cidel_laseru_1.3mf
- Kryt_cidel_laseru_2.3mf
- Spojovaci_dil_krytu.3mf
- Kryt_koncoveho_spinace.3mf
- Kryt_spodni_casti_modelu.3mf
- Kryt_vyvysene_casti_modelu.3mf
- Ozubena_lista.3mf