VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2018 Bc. Tomáš Trenz

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ
DEPARTMENT OF TELECOMMUNICATIONS

AUTOMATIZOVANÝ POSUN PRO POŘIZOVÁNÍ ČASOSBĚRNÝCH SNÍMKŮ
AUTOMATED DEVICE FOR SHIFT TIME-LAPSE CAPTURE

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE Bc. Tomáš Trenz
AUTHOR

VEDOUCÍ PRÁCE Ing. Ondřej Krajsa, Ph.D.
SUPERVISOR

BRNO 2018

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Diplomová práce

magisterský navazující studijní obor **Telekomunikační a informační technika**
Ústav telekomunikací

Student: Bc. Tomáš Trenz ***ID:*** 164623
Ročník: 2 ***Akademický rok:*** 2017/18

NÁZEV TÉMATU:

Automatizovaný posun pro pořizování časosběrných snímků

POKYNY PRO VYPRACOVÁNÍ:

Vytvořte zařízení pro automatický posun a spoušť digitálního fotoaparátu pro pořizování časosběrných snímků. Pro řízení systému vyberte vhodný mikrokontroler a krokové motory. Analyzujte možnost ovládání fotoaparátu z PC a přenos snímků v reálném čase.

DOPORUČENÁ LITERATURA:

[1] PEREA, Francis. Arduino Essentials. Olton Birmingham: Packt Publishing, 2015. ISBN 9781784398569.

[2] PURDUM, Jack. Beginning C for Arduino. Berkeley, CA: Apress, 2015. DOI: 10.1007/978-1-4842-0940-0. ISBN 1484209419

Termín zadání: 5.2.2018 ***Termín odevzdání:*** 21.5.2018

Vedoucí práce: Ing. Ondřej Krajsa, Ph.D.
Konzultant:

prof. Ing. Jiří Mišurec, CSc.
předseda oborové rady

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

ABSTRAKT

Tato diplomová práce se zabývá návrhem automatizovaného posunu pro pořizování časosběrných snímků a vytvoření funkčního vzorku. V rámci práce jsou rozebrány jednotlivé komponenty a samotný řídící program.

KLÍČOVÁ SLOVA

Automatizace, automatický posun, časosběr, mikrokontrolér, Atmel Studio, Arduino Due, Wemos D1

ABSTRACT

Automatic slider for time lapse photography and the creation of a functional sample is described in this diploma work. The individual components and the control program itself are analyzed.

KEYWORDS

Automation, automatic slider, time lapse photography, microcontroller, Atmel studio, Arduino Due, Wemos D1

TRENZ, Tomáš. *Automatizovaný posun pro pořizování časosběrných snímků*. Brno, 2017, 58 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedoucí práce: Ing. Ondřej Krajsa, Ph.D.

Vysázeno pomocí balíčku `thesis` verze 2.63; `http://latex.feec.vutbr.cz`

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Automatizovaný posun pro pořizování časosběrných snímků“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

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

Brno ....

podpis autora

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Ondřeji Krajsovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci. Také bych chtěl poděkovat za zapůjčeni vývojové desky.

Brno ....

podpis autora

Faculty of Electrical Engineering
and Communication
Brno University of Technology
Purkynova 118, CZ-61200 Brno
Czech Republic
http://www.six.feec.vutbr.cz

PODĚKOVÁNÍ

Výzkum popsaný v této diplomové práci byl realizován v laboratořích podpořených z projektu SIX; registrační číslo CZ.1.05/2.1.00/03.0072, operační program Výzkum a vývoj pro inovace.

Brno ....

podpis autora

MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY

EVROPSKÁ UNIE
EVROPSKÝ FOND PRO REGIONÁLNÍ ROZVOJ
INVESTICE DO VAŠÍ BUDOUCNOSTI

2007–13
OP Výzkum a vývoj pro inovace

OBSAH

Úvod **10**

1 Teoretická část **11**

1.1 Popis zařízení 11

1.2 Blokové schéma 12

1.2.1 Mikrokontrolér 12

1.2.2 Wi-Fi modul 13

1.2.3 Rozhraní 13

1.2.4 Uživatelské rozhraní 16

1.2.5 Spínač fotoaparátu 18

1.2.6 Krokový motor 18

1.2.7 Senzory 19

2 Výsledky diplomové práce **20**

2.1 Výběr komponentů 20

2.1.1 Mikrokontrolér 20

2.1.2 Wi-Fi modul 21

2.1.3 Krokový motor 22

2.1.4 Ovladač krokového motoru 22

2.1.5 Spínač fotoaparátu 22

2.1.6 Displej 23

2.1.7 Zdroje napětí 23

2.2 Deska plošných spojů 24

2.3 Mechanická konstrukce 25

2.4 Programové řešení 27

2.5 Program pro Arduino Due 28

2.5.1 Soubor definic 28

2.5.2 Kontrolér 28

2.5.3 Ovladače periferií 31

2.5.4 Ovládání I2C komunikace 34

2.6 Program pro Wemos D1 36

2.6.1 Soubor definic 36

2.6.2 Kontrolér 36

2.6.3 Ovládání Wi-Fi 37

2.6.4 Komunikace I2C 37

2.6.5 Řízení internetové stránky 38

2.6.6 Připojení na internetovou stránku s ovládáním 39

2.7 Přenos dat z fotoaparátu 39

3 Závěr **41**

Literatura **42**

Seznam symbolů, veličin a zkratek **44**

Seznam příloh **45**

A Schéma zapojení a osazení **46**

B Obsah přiloženého CD **57**

SEZNAM OBRÁZKŮ

1.1 Posloupnost funkcí automatického posunu. 11
1.2 Blokové schéma automatického posunu. 12
1.3 Sběrnice I^2C. 14
1.4 Sběrnice SPI. 15
1.5 Schéma pinu s pull-up rezistorem pro detekci sepnutí spínače. 16
2.1 Arduino Due 20
2.2 Wemos D1 21
2.3 Osazená deska plošných spojů. 24
2.4 Začátek dráhy. 26
2.5 Konec dráhy s vozíkem. 26
2.6 Sestavené zařízení pro automatický posun 27
2.7 Vývojový diagram běhu systému. 28
2.8 Vývojový diagram inicializace systému. 29
2.9 Vývojový diagram události hodin. 31
2.10 Internetová stránka. 39
A.1 Schéma zapojení modulů 1/2. 46
A.2 Schéma zapojení modulů 2/2. 47
A.3 Osazení horní strany. 48
A.4 Osazení spodní strany. 48
A.5 Plošný spoj spodní strana. 49

ÚVOD

Tato diplomová práce se věnuje návrhu automatického posunu se spouští, pro digitální fotoaparát. Jeho hlavní využití je při pořizování časosběrných snímků, u kterých je žádána změna pozice mezi jednotlivými snímky. V tomto případě se změnou pozice myslí posun po kolejnici. Tato změna polohy se hojně využívá pro podtržení prostorového vnímání u výsledných videozáběrů, složených z jednotlivých časosběrných snímků. Jedná se tedy o kombinaci technik fotografie a filmu.

V rámci práce bude pojednáváno o výběru mikrokontroléru, který řídí celé zařízení. Pro účely vzdáleného ovládání bude vybrán a popsán modul Wi-Fi. A také bude popsán samotný způsob vzdáleného ovládání. Dále bude pojednáno o periferiích použitých k funkci zařízení a obstarávajících komunikaci s uživatelem. Krokový motor, používaný pro pohyb posunu, bude vybrán na základě požadavků pro výkon, velikost a cenu. Popsány budou i další komponenty použité při stavbě zařízení.

Cílem práce je také vytvoření funkčního vzorku a ověření správnosti návrhu. Pro tento účel budou vytvořeny 3D modely hlavních dílů a zhotovení bude provedeno pomocí 3D tisku. Jednotlivé moduly budou propojeny pomocí desky plošných spojů. Ta bude navrhnuta s ohledem na požadované rozměry a vyrobena.

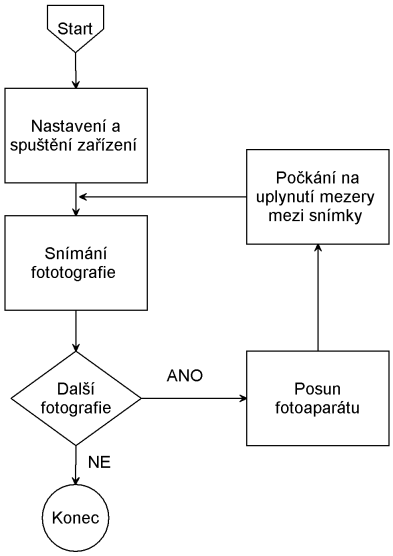
Vytvořený software pro dané MCU (Mikrokontrolér, jednočipový počítač) a Wi-Fi modul bude rozebrán po blocích.

1 TEORETICKÁ ČÁST

1.1 Popis zařízení

Zařízení pro automatický časosběr je takové zařízení, které je schopné v určitých okamžicích po určitou dobu spínat spoušť digitálního fotoaparátu a s jeho pomocí vytvářet sérii fotografií. Vše by mělo být možné nastavit uživatelem. Hlavními parametry je počet fotografií a délka pauzy mezi fotografiemi. Další nastavení, které může být vyžadováno uživately, je délka stisku spouště, a to pro případ snímání časem delším, než umožňuje časovač ve fotoaparátu.

U automatického posunu se jedná o zařízení, které má navíc plnit funkci pohybovou. Je zde tedy zapotřebí, aby obsahovalo pohonnou jednotku a vodící kolejnice, po kterých se bude moci fotoaparát pohybovat. V tomto případě k požadovaným funkcím přibude ještě možnost nastavit vzdálenost, jakou má fotoaparát urazit. Na diagramu 1.1 jsou znázorněny jednotlivé kroky běhu zařízení.



Obr. 1.1: Posloupnost funkcí automatického posunu.

Pro správnou činnost je tedy nezbytné sloučit několik samostatných zařízení do jednoho celku a jako celek celé zařízení ovládat. K tomuto účelu se hodí využít MCU, jakožto univerzální a programovatelný obvod, který je schopen zpracovat velké množství vstupních veličin a na základě vyhodnoceného stavu provést požadované úkony v téměř reálném čase.

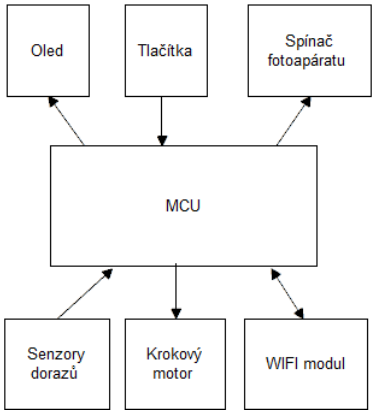
Další částí zařízení je jednotka, která umožní uživateli vzdálené ovládání zařízení. Pro tuto funkci může být využito například bezdrátového ovladače na bázi optického spojení. Lepší variantou je pro ovládání využít například internetové stránky. Tu je

možné zobrazit z téměř jakéhokoliv osobního telefonu, počítače a podobných zařízení. Není tedy nutné přidávat další jednoúčelové zařízení pro nastavení parametru, ale je využito zařízení, které uživatel již ve většině případů vlastní.

Výsledkem bude tedy zařízení, které obsluhuje fotoaparát, řídí pohonnou jednotku a umožňuje uživateli připojit se pomocí bezdrátového ovládání.

1.2 Blokové schéma

Automatický posun pro časosběr by měl obsahovat alespoň bloky pro spínání spouště fotoaparátu, posun konstrukce s fotoaparátem a také bloky uživatelského rozhraní. Lokální rozhraní se skládá ze zobrazovací jednotky a tlačítek pro nastavení požadovaných hodnot. Vzdáleným uživatelským rozhraním je modul Wi-Fi, který díky vhodnému programu umožňuje ovládání zařízení. Hlavním blokem je MCU, který zajišťuje řízení všech periferií. Tedy výsledné schéma celého zařízení může vypadat jako to na Obr. 1.2.



Obr. 1.2: Blokové schéma automatického posunu.

Pro komunikaci mezi MCU a periferiemi budou použity následující rozhraní. I^2C pro komunikaci s modulem Wi-Fi, GPIO pro tlačítka, motor a senzory a SPI pro komunikaci s OLED displejem.

1.2.1 Mikrokontrolér

Jedná se o integrovaný obvod obsahující kompletní mikropočítač. MCU obsahuje vše, co je potřeba pro danou aplikaci, a to především paměť pro uchování jak programu, tak na výsledky výpočtů. Dále také obsahuje různá rozhraní a zdroj hodinového impulzu. Ten bývá generován na základě vnitřního nebo vnějšího krystalu.

Základní architekturou, kterou se různé MCU liší, je buď von Neumannova architektura, nebo Harvardská architektura. U Harvardské architektury je paměť oddělena od programu. To má za následek složitější strukturu, ale bezpečnější běh programu, jelikož nemůže dojít k nechtěnému přepsání programu zápisem hodnot do paměti. Díky tomu, že obsahuje dvě sběrnice, může být jejich šířka různá a tím pádem můžeme pracovat s pamětmi majícími jinou bitovou délku než MCU. Dále také může číst instrukce a data v jeden okamžik. Von Neumannova architektura se vyznačuje jednou společnou pamětí pro program i data. Z toho důvodu je zde pouze jedna sběrnice a je možné dosáhnout větší integrace. Hodí se tedy pro aplikace, kde je požadován výkon, ale i co nejmenší možné rozměry. Společná paměť má nevýhodu v nutnosti zajistit, že paměť s uloženým programem nebude nikdy přepsána. Operace s pamětí je také omezena, není možné číst data a zároveň instrukce programu.

MCU se také dělí podle sady instrukcí, na procesory s redukovaným instrukčním souborem - Reduced Instruction Set Computing a na procesory s rozsáhlou sadou instrukcí - Complex Instruction Set Computing. RISC má omezenou sadu instrukcí pouze na základní funkce a při programování je nutné všechny operace vytvořit z těchto základních funkcí. U sady CISC je předností rozsáhlá sada instrukcí a díky ní je možné oproti RISC využít lépe programovou paměť, má ale větší požadavky na programový dekodér.

1.2.2 Wi-Fi modul

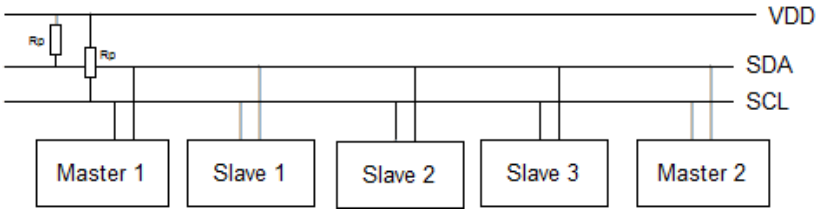
Jedná se o MCU obsahující, kromě běžných periferií, vysílač a přijímač schopný pracovat na dané frekvenci. Dále také má dostatečně vysoký výpočetní výkon a paměť, aby byl schopen vykonávat základní operace související s komunikací. Pro umožnění komunikace pomocí Wi-Fi specifikace, je zapotřebí vysílat na jednom z daných frekvenčních pásem. Dále pro kompatibilitu s běžně používanými zařízeními je nutné použít protokol TCP/IP. Protokol TCP/IP specifikuje způsoby komunikace a typy na jednotlivých vrstvách sítě.

1.2.3 Rozhraní

Rozhraním je myšlena část sloužící k propojení MCU s okolím. Pro konkrétní rozhraní jsou specifikovány podmínky chování společných částí různých zařízení, typ spojení a posloupnost úkonů, aby docházelo ke korektní komunikaci mezi těmito částmi zařízení.

Inter-Integrated Circuit I^{2}C

Sériová sběrnice pro připojení většího množství periferií a řídících zařízení. Využívá pouze 2 vodiče a to jeden pro hodiny a jeden pro data. Jedná se o nízkorychlostní komunikaci. Sběrnice s několika zařízeními je zobrazena na Obr. 1.3. Pro udržení kladné hodnoty v klidovém stavu jsou linky připojeny ke zdroji napájení pomocí pull-up rezistorů.



Obr. 1.3: Sběrnice I^{2}C.

Komunikace na této sběrnici je řešena pomocí 7 nebo 10 bitové adresy, pro kterou platí, že musí být na jedné sběrnici právě jedno zařízení s danou adresou. V opačném případě by nebylo možné kontaktovat dané jedno zařízení, ale skupinu. To by znamenalo problémy se zpětným potvrzováním komunikace. Aby nedocházelo k několikanásobnému přístupu ke sběrnici, dochází k detekci stavu sběrnice při vysílání a zařízení může vysílat pouze, pokud je sběrnice v klidovém stavu. Výměna dat probíhá vždy po 8 bitech.

Ke komunikaci dochází vždy z iniciativy řídícího zařízení, to začne vysílání start-bitem. Vyšle adresu zařízení, se kterým chce komunikovat. Pokud se taková adresa nachází na sběrnici, zařízení s onou adresou odpoví ACK bitem. Ten je reprezentován jako uzemnění datové linky v podřadném zařízení. V této části se totiž linka nachází v logické 1 a řídící zařízení detekuje jestli nastala změna, či nikoliv. Řídící zařízení detekuje toto potvrzení a pokračuje ve vysílání dat. V opačném případě pošle stop-bit a ukončí vysílaní. Zdali jsou vyžadována data na vrácení, či nikoliv, udává bit přidaný k adrese jako 8. bit. Na základě tohoto bitu tázané zařízení ví, jaká má být jeho reakce.

Toto rozhraní se hojně využívá pro nastavení periferií obsahující registry. Pro nastavení periferie řídící zařízení pošle start-bit následovaný adresou periferie a značkou, že bude prováděn zápis. Periferie odpoví ACK-bitem a následně řídící zařízení vyšle číslo registru a jeho hodnotu, popřípadě pokračuje pro následující registry. Komunikace je po každých 8 bitech potvrzena od periferie ACK-bitem a na konci je poslán stop-bit. Pokud má dojít ke čtení, pak po části s adresou následuje bit signalizující čtení. Komunikace je zase potvrzena ze strany periferie. Následuje číslo

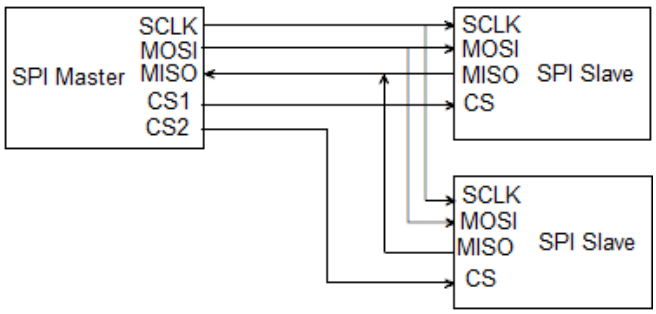
registru, které se má přečíst. Po potvrzení od periferie je proveden opakovaný start-bit a odeslána stejná adresa. Dále jsou, po dobu vyčítání registrů, generovány hodiny. Pro jeden registr je po 8 krocích hodin poslán stop-bit. Pokud chceme přečíst více registrů, pak je nutné generování dalších osmi kroků hodin.

Serial Peripheral Interface SPI

Jedná se o rozhraní používané pro komunikaci mezi jedním řídícím zařízením a jedním nebo více podřízenými zařízeními. Nejčastěji se využívá pro komunikaci mezi MCU a Flash pamětí, OLED displejem a dalšími podobnými zařízeními, u kterých je požadována vysoká přenosová rychlost.

Komunikace probíhá pomocí dvou datových vodičů a jednoho vodiče pro přenos synchronizačních hodin. Pro adresování se využívá vodiče, kterým řídicí zařízení určuje pro každého z připojených zařízení, jestli má právo číst a zapisovat na rozhraní. Na obrázku 1.4 je zobrazen případ pro jedno řídící zařízení a dvě podřízené zařízení.

Přenos je započat změnou úrovně na adresové lince, ta je také značena jako Slave Select. Poté dojde k začátku generování hodin. Jakmile jsou generovány hodiny mohou obě zařízení vysílat současně, a to každé na své výstupní lince, která vede na vstupní linku druhého zařízení. Délka dat záleží na konkrétním použití, ale obvykle se odvysílá blok po 8 bitech následovaný krátkou mezerou. K ukončení komunikace dojde, když řídící zařízení přestane vysílat hodiny a změní logickou úroveň na adresové lince. Logické úrovně a s tím i reakce na danou hranu se mohou lišit podle nastavení rozhraní. Jedná se o kombinaci klidové polarity hodinového signálu a reakci na první nebo druhou hranu.



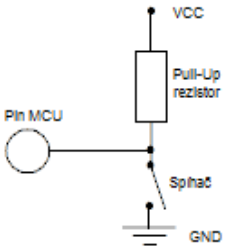
Obr. 1.4: Sběrnice SPI.

General-purpose input/output GPIO

Jedná se o piny, které nemají předem definovaný účel. Lze je nastavit na několik různých funkcí. U většiny mikrokontrolérů je téměř většina pinů dostupná jako GPIO.

Jejich další funkce využívají právě jejich možnosti nastavení určitého stavu. Tudíž by se dalo říct, že každé rozhraní částečně využívá GPIO vrstvu.

Toto rozhraní umožňuje nastavit jednotlivé piny na vstup a výstup, připojit k nim interně danou logickou hodnotu pomocí pull-up a pull-down rezistorů a číst aktuální úroveň napětí na daném pinu. Pull-up a down piny bývají realizovány pomocí odporů s velkou rezistencí, díky tomu při spojení pinu s opačnou logickou úrovní teče odporem malý proud a nedochází k vysoké spotřebě, avšak stále jsme schopni určit na jakou logickou hodnotu je pin připojen. Této konfigurace se využívá například pro realizaci tlačítek a spínačů, kde je, jak je vidět na Obr. 1.5, pin spojen pomocí Pull-up rezistoru na logickou 1 (napájení) a je tedy čtena hodnota rovna logické 1. Při stisku tlačítka je pin uzemněn a je na něm možno detekovat úroveň logické úrovně 0. Při stisku, díky velkému odporu, dochází k průchodu malého proudu a tím pádem nedochází k významnému zatěžovaní zdroje a samotného MCU.



Obr. 1.5: Schéma pinu s pull-up rezistorem pro detekci sepnutí spínače.

Hlavním využitím je právě čtení logické úrovně a možnost nastavit logickou úroveň na výstup. Je-li požadována hodnota pinu jiná, než logická a pin umožňuje použití pulzní šířkové modulace, pak je možné dosáhnout díky ní napětí mezi logickou úrovní 1 a 0. Funkce je založena na setrvačnosti napětí, díky kapacitě pinu a opakovaném připojování a odpojování logické 1 v pravidelných intervalech, ale s rozdílnou dobou trvání připojeného stavu.

K jednotlivým pinům je možno přiřadit přerušení, které je vyvoláno při změně stavu, odpovídajícímu detekčnímu stavu. Toto přerušení může být využito například pro senzory dorazu a podobné, kdy je potřeba okamžitě reagovat na určitý stav.

1.2.4 Uživatelské rozhraní

Uživatelské rozhraní se zpravidla dělí na dvě části, a to vstupní a výstupní. Vstupním jsou například tlačítka pro možnost nastavení parametrů a výstupním je displej, nebo světelná signalizace pomocí LED. Pokud je požadavek na zobrazení většího množství informací a měla by být možnost měnit obsah, je pro uživatele vhodnější

použít displej, na kterém se zobrazí informace. Další variantou může být zvuková odezva, ta ale není vždy vypovídající tak, jako obrazová.

Pro automatický posun je použito tlačítek a OLED displeje jako základní uživatelské rozhraní a dále také internetová stránka. Ta slouží pro bezdrátové ovládání zařízení a je vstupním i výstupním rozhraním.

Tlačítka

Jedná se o mechanické rozhraní mezi uživatelem a zařízením, užívané pro nastavení i samotné spouštění zařízení. Při stisku tlačítka se mění logická hodnota, která je detekována. Pro detekci stisku se využívá GPIO rozhraní. Podle důležitosti a možností MCU je možné k jednotlivým pinům přiřadit přerušení. Tím se například zajistí, že daný pin probudí zařízení z režimu snížené spotřeby. U mechanických tlačítek je při snímání potřeba provést ošetření záchvěvů. Ty se projevují jako několikanásobné stisknutí a mohou způsobovat nechtěné události. Záchvěvy se mohou ošetřit pomocí přidání kondenzátoru do obvodu, který nedovolí rychlé změny napětí, nebo také implementací správné logiky do obslužného programu. Obvykle se tento problém řeší programovou úpravou. Díky tomu je pak umožněno reagovat na různě dlouhé stisky tlačítka různým chováním.

Displej

Slouží pro zobrazení informací uživateli a má několik podob. Nejjednodušší podobou může být sedmi segmentový displej, který je vyroben ze sedmi LED uspořádaných tak, aby na něm bylo možné zobrazit čísla od 0 po 9. Pro vypsání textu není vhodný, obzvláště u písmen, které vyžadují šikmé čáry. Další variantou může být rozšířená verze a to 14 segmentů. Ta je již lépe uzpůsobena pro zobrazení písmen, nicméně ještě se nejedná o ideální zobrazení. Lepších výsledků je dosahováno na matici Led diod, avšak jejich řízení je náročné na počet řídících pinů. Z tohoto důvodu se tedy využívají v kombinaci s posuvnými registry, které nastaví své piny podle posloupnosti, která dojde na vstup. Díky tomu je možné pomocí 2 vodičů nastavit několikanásobné množství diod.

K nejvíce pokročilým metodám zobrazení s využitím na jednoduchých zařízeních patří OLED displej, konkrétně pasivní OLED. Ten je tvořen hustou sítí malých organických elektroluminiscenčních diod a je vhodný pro textové zobrazení. Jednotlivé diody jsou zapojeny v mřížové struktuře stejně jako matice diod, ale je možné dosáhnout větší integrace. Obsah se zobrazuje pomocí pulzů na jednotlivých řádcích. Tedy v jeden okamžik svítí pouze body na jednom řádku, ale díky setrvačnosti lidského oka a materiálu je obraz vnímán jako celek. K běžnému použití jsou displeje vyráběny v různých velikostech a s různou hustotou.

1.2.5 Spínač fotoaparátu

Pro vzdálené spínání fotoaparátů existují různé metody, nejjednodušší je kabelová spoušť. Ta pracuje na principu tlačítka spouště na kabelu připojeného do fotoaparátu. Jedná se o obvod se dvoustavovým spínačem, přičemž v neaktivním stavu jsou linky rozpojeny, při namáčknutí dojde k spojení linky ostření se zemí fotoaparátu a domáčknutí spojí se zemí linku spouště. Tato metoda se u výrobců liší pouze typem konektoru, který je použit. Zrcadlovky od značky Canon mají konektor JACK 2,5 stereo, Nikon na rozdíl od toho používá konektor USB mini.

Další možností je využít bezdrátové spouště v podobě infračerveného přenosu dat. Základem je přijímač umístěný v těle fotoaparátu a vysílač, který je obvykle v podobě dálkového ovladače. Když je přijímacím členem zachycena správná posloupnost dat, dojde k provedení dané operace, obvykle ostření a vyfotografování. Největší nevýhodou je, že dané sekvence dat nejsou kompatibilní mezi jednotlivými výrobci. Také není zajištěna bezpečnost proti zneužití ovladačem v držení jiné osoby.

Poslední možností jsou pokročilé metody ovládání a to pomocí protokolu PTP, přes rozhraní USB a nebo Wi-Fi. Toto ovládání je využito při ovládání z PC nebo ze smartphonu. Jedná se o protokol primárně určený pro přenos fotografií ze zařízení jako fotoaparáty. Je součástí USB standardu a díky tomu nejsou nutné další ovladače. Protokol umožňuje komunikaci pomocí specifických kódů, z nichž ty základní pro přenos informací jsou řešeny standardem a další si určují výrobci právě pro vzdálené ovládání fotoaparátu. Dále byla vyvinuta další vrstva PTP a to PTP/IP. Ta dovoluje komunikaci tímto protokolem ve všech sítích založených na IP struktuře. Díky tomu je možné využít Wi-Fi, která je již v novějších fotoaparátech použita, a řídit fotoaparát bezdrátově prostřednictvím internetové sítě, popřípadě stahovat fotografie ze zařízení.

1.2.6 Krokový motor

Krokový motor je založen na vícepólovém magnetu v rotoru a několika elektromagnetech ve statoru. Jeho vnitřní konstrukce umožňuje otáčení po předem definovaných krůčcích a je ovládán impulsy stejnosměrného napětí. Pro běh motoru je potřeba specifický řadič pro danou vnitřní stavbu motoru. Tento řadič generuje správné pořadí a periodu impulzů, které jsou potřeba pro chod motoru. Při nízkých rychlostech otáčení je pohyb nespojitý a dochází k znatelným skokům mezi jednotlivými kroky motoru. Jeho nevýhodou je zakmitávání při překročení mezní zátěže. Výhodou je naopak přesné ovládání počtem kroku a jejich rychlostí. Jejich hlavním využití je například pro pohyb tiskových hlav u 3D tisku, nebo jako pohon pro otáčení ploten u pevných disků, kde se využívá vlastností přesného řízení rychlosti.

1.2.7 Senzory

Senzor je zařízení, které sleduje určitou fyzikální veličinu nebo jev a převádí tento jev do elektrické podoby, aby bylo možné ji detekovat pomocí logických obvodů. Pomocí senzorů může být detekována úroveň určité veličiny například teploty, tlaku nebo jasu světla. Dále také přiblížení, dotknutí a vychýlení. Senzory jsou používány v dnešní době snad v každém zařízení, které má elektrickou řídící jednotku a vyžaduje kontrolu stavu. Primitivním senzorem muže být myšlena součástka, jako je spínač. Ten převádí mechanickou událost, stisknutí tlačítka, na elektrickou rozpojením, nebo spojením obvodu.

2 VÝSLEDKY DIPLOMOVÉ PRÁCE

2.1 Výběr komponentů

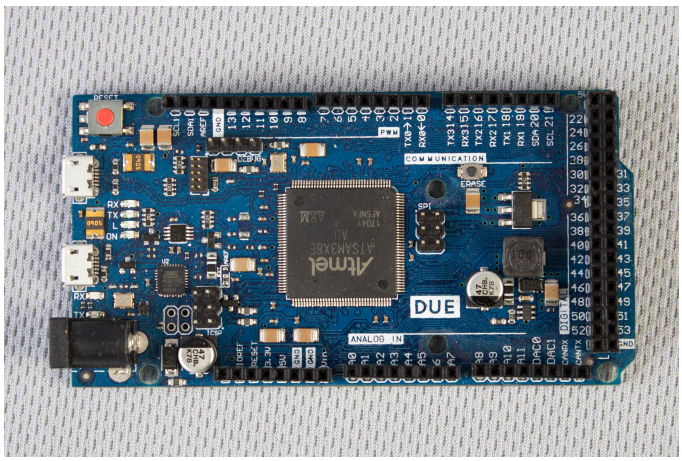
2.1.1 Mikrokontrolér

Mezi požadavky na MCU je, aby obsahoval rozhraní typu I^2C, SPI, dokázal řídit a číst úrovně napětí na jednotlivých pinech a obsahoval nativní podporu USB.

Pro vývoj byla vybrána byla platforma Arduino, z důvodu rozsáhlé podpory od komunity uživatelů a velkého množství open-source knihoven. Vývojová deska Arduino Due byla vybrána, jelikož MCU splňuje požadavky na podporované rozhraní. Mezi další zvažované vývojové desky patří Wemos Lolin. Jeho hlavní omezení je v počtu dostupných periferií, jejich možných kombinacích a absenci nativní podpory rozhraní USB. Další možnou vývojovou deskou je některá z verzí Nucleo od STMicroelectronics. Ty bývají osazené různými MCU, ale verze vývojových desek osazené typem MCU s nativní podporou USB, bývají obvykle větších rozměrů.

Arduino Due

Na tomto vývojovém kitu je osazen 32 bitový čip od společnosti Atmel, nesoucí označení SAM3X8E. Tento mikročip je založen na architektuře ARM konkrétně Cortex-M3. Vývojová deska umožňuje využít 54 pinů MCU a spoustu variant rozhraní pro komunukaci s periferiemi [1].



Obr. 2.1: Arduino Due

Hlavní předností je podpora USB rozhraní přímo mikrokontrolérem a požadované rozhraní. Výhodou je také možnost využít Atmel Studia pro vytvoření programu. To je díky pokročilému prostředí pro vývoj kódu přívětivější k uživateli, než prostředí

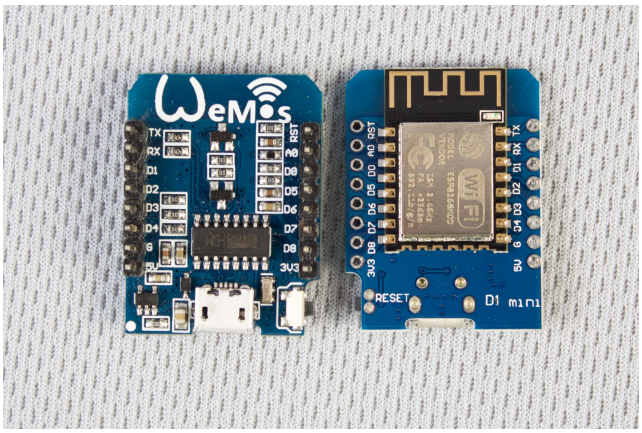
programu Arduino IDE. Díky konektorům v podobě dutinkové lišty, je možné vyžít celou vývojovou desku jako modul. Cena vývojové desky je přibližně 800 Kč [1].

2.1.2 Wi-Fi modul

Je základní blok pro komunikaci s uživatelem pomocí Wi-Fi. Musí mít možnost komunikovat s hlavním MCU a informace předávat ve vhodné formě uživateli pomocí Wi-Fi. Mezi oblíbené a hojně využívané Wi-Fi moduly patří čip ESP8266 od firmy Espressif. Jedná se o velmi malý a levný modul složený z řídícího MCU radiové části a paměti FLASH. Samotný čip již obsahuje anténu a potřebné periferie, ale různí výrobci přidávají své vlastní řešení napájení a převodníků pro naprogramování. Mezi nejvíce populární patří Wemos D1. Cena je sice o trochu vyšší než samotný ESP8266, ale na desce je osazen zdroj a převodník pro komunikaci s počítačem.

Wemos D1

Jedná se o vývojovou desku založenou na čipu ESP8266 a doplněnou o převodník mezi rozhraním USB a UART. Je zde také DC-DC měnič, který se stará o transformaci napájení z USB na požadované napětí 3,3 V. Díky tomu již není nutné řešit zdroj při vývoji a stačí modul připojit pomocí USB. Další výhoda této vývojové desky s ESP8266 je v tom, že jsou vyvedeny piny pro připojení na nepájivé pole. Díky tomu je možné využít i jiné rozhraní než UART, které je použito pro komunikaci mezi interním převodníkem USB-UART a ESP8266 modulem [2]. U nejlevnějších modulů bývá vyveden právě pouze UART. Pro tuto vývojovou desku existuje několik modulů, které odpovídají rozměrově, takže je zachována kompaktnost zařízení. Mezi nejzajímavější patří senzor teploty, OLED displej, nebo modul pro microSD kartu.



Obr. 2.2: Wemos D1

2.1.3 Krokový motor

Při výběru krokového motoru byl zvážen požadavek na výkon a na co nejmenší velikost.

Byl vybrán jeden z menších krokových motorů hojně využívaných v 3D tiskárnách. Motor má označení SX17-1003LQCEF a jeho výrobcem je Microcon. Podle technické specifikace na stránkách prodejce se jedná o motor s kroutícím momentem 0,3 Nm a přírubou NEMA 17 [3]. Jmenovitý proud je 1 A a odpor motoru je uváděn 4,2 Ω.

Pro přenos pohybu z motoru na posun je použito řemenu a řemenice, ty jsou dodávány jako standardní příslušenství pro daný typ motorů, a to na použití v 3D tisku. Jedná se o ozubenou verzi řemenu, která se používá pro posun tiskových hlav. Díky ozubení řemene není mechanizmus náchylný na prokluz po řemenici a také není potřeba speciálních napínáků, vyrovnávajících drobné změny v tahu.

2.1.4 Ovladač krokového motoru

Pro správnou funkci je nutné krokový motor řídit pomocí ovladače krokových motorů. Ten se stará o správné načasování a intenzitu pulzů pro jednotlivé cívky. Pro tento účel byl vybrán ovladač doporučený přímo pro daný motor ze stránek prodejce. Jedná se o obvod od výrobce Pololu a nese označení DRV8825 Stepper Motor Driver Carrier.

Je to ovladač, který dokáže zvýšit počet kroků řízením úrovní napětí. Maximální rozlišení je poté 1/32 původní délky kroku. Rozsah napětí, s kterým dokáže pracovat, je od 8,2 V až po 45 V při 1,5 A. Pokud je vyžadován větší proud, je možné dosáhnout až 2,2 A, ale je nutné zajistit účinné chlazení. Vstupním rozhraním je několik pinů, které akceptují systémy s napětím 3,3 V a 5 V [4].

2.1.5 Spínač fotoaparátu

Spínání fotoaparátu pro časosběr je možné udělat pomocí jednoduchého spínače, nebo pomocí propojení přes rozhraní USB.

V drátových spouštích pro digitální zrcadlovky bývá obvykle třístavový spínač. První stav je neaktivní, druhý vyvolá požadavek na ostření a třetí stav je snímání. Pro firmu Canon je možné použít konektor Jack 2,5 mm a pro značku Nikon je používaná koncovka mini USB.

Při připojení pomocí sběrnice USB a následného využití protokolu PTP, je sice větší možnost ovládat fotoaparát, ale příkazy nejsou totožné ani mezi samotnými modely jednoho výrobce, natož napříč mezi konkurenčními výrobci. Dále u některých modelů jsou omezeny tyto funkce jen na základní. Další nevýhodou je nesdílnost

výrobců. Jednotlivé kódy nebývají k dispozici, a je tedy nemožné, aby byla sestavena tabulka podporovaných zařízení bez jejich ověření. Z tohoto důvodu bylo pro ovládání fotoaparátu použito jednodušší metody, kde stačí mít správný konektor a je ovládaná pouze spoušť zařízení.

Jednodušší variantou je použít metodu drátové spouště a to pouze pro kanál spouště, jelikož časosběr se obvykle snímá s ručně nastaveným ostřením na určitou pozici. Pro sepnutí fotoaparátu je vhodné galvanicky oddělit elektrické obvody posunu a fotoaparátu, aby byla zajištěna ochrana, proti nechtěnému poškození elektroniky fotoaparátu. Pro tento účel se hodí využít optočlen. Jeho funkce je založena na LED a fotocitlivé polovodičové součástce. Při průchodu proudu diodou dochází k vyzařování světla, což otvírá tranzistor. Míra otevření tranzistoru závisí na intenzitě dopadajícího světla a to je závislé na velikosti proudu procházejícího diodou. Výsledkem je tedy obvod ovládaný pomocí napěťové úrovně a spíná jiný obvod, který nemusí být na stejném potenciálu.

Vybrán byl optočlen s označením EL817 firmy Everlight. Ten vyniká vysokým izolačním napětím, které dosahuje až 5000 V. Pro spínaní je vyžadováno napětí 1.2 V a proud je 20 mA [5]. Dioda optočlenu je zapojena do série s 100 Ω z důvodu omezení proudu a napětí na diodě.

2.1.6 Displej

Bylo zvažováno využití displeje určeného pro plánovaný modulu Wi-Fi, ale nakonec byl vybrán displej s větším rozlišením a rozhraním SPI. Jedná se o jednobarevný OLED displej s rozlišením 128 bodů na šířku a 64 bodů na výšku. Uhlopříčka je 0,96 palce. Tento OLED modul je osazen ovladačem SSD1306. [6].

Díky rozhraní SPI je dosaženo rychlejší komunikace než při použití I^2C a díky tomu je možné měnit zobrazený obsah rychleji. Pro statická data tato rychlost není rozhodující, ale v případě zobrazení animace je pro plynulé zobrazení nutné měnit data v dostatečné rychlosti.

2.1.7 Zdroje napětí

Pro napájení zařízení je plánováno použít Lithium-iontovou baterii, jejíž napětí se pohybuje v okolí 3,7 V. Ovladač krokového motoru má pracovní napětí v rozmezí 8.2 V až 45 V. Vývojová deska Arduino Due vyžaduje napájení pomocí 5 V, nebo při použití vnitřního měniče 7 V až 12 V. Je nutné zajistit, potřebné napájecí napětí, pomocí Step-up měniče .

Bylo zvažováno využití jednoho zdroje pro oba obvody, ale nakonec je využito dvou samostatných měničů napětí. Hlavním argumentem pro toto rozhodnutí bylo omezení modulů zdrojů obvykle na 1 až 2 A. Toto proudové omezení by mohlo být

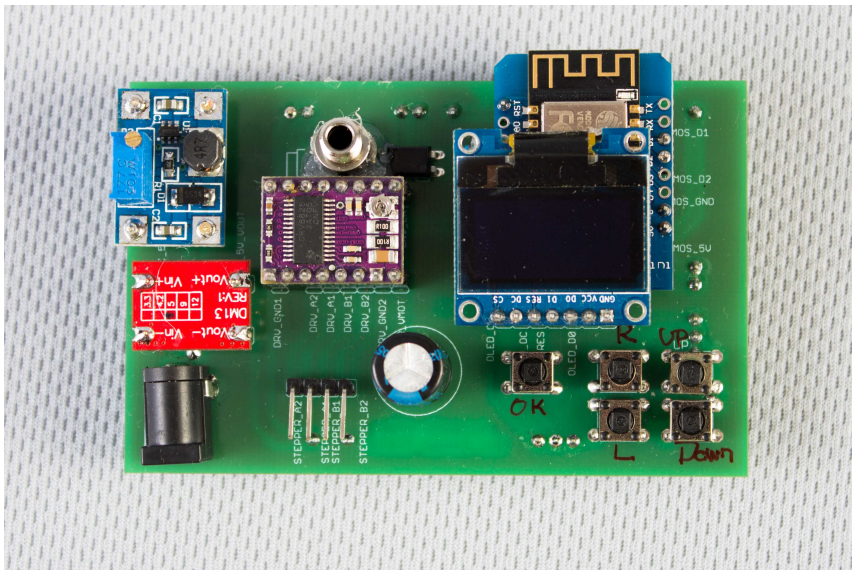
nedostatečné, jelikož při spínání krokového motoru by mohlo docházet k poklesu napětí a způsobovat neočekávané chování zdroje na vývojové desce, a v nejhorším případě i restartování zařízení.

Jako výkonový zdroj pro napájení krokových motorů byl využit modul Eclipsera SX1308. Jedná se o DC-DC Step-up měnič, který je určen pro změnu stejnosměrného napětí na napětí vyšší, než je vstupní. Jeho vstupní napětí je možné v rozsahu od 2 V do 24 V a výstupní napětí dosahuje maximálně 28 V s proudem 2 A [7].

Pro zdroj napájení obvodu vývojové desky byl vybrán také měnič stejnosměrného napětí tentokrát s pevným výstupním napětím 5V a proudem 2 A [8]. Díky připojení zdroje rovnou na 5 V okruh vývojové desky Arduina Due je možné z této větve napájet i Wi-Fi modul Wemos D1. Ten vyžaduje také napájecí napětí 5 V.

2.2 Deska plošných spojů

Pro vytvoření kompaktního zařízení s minimálním počtem samostatných částí byla navržena deska plošných spojů. Bylo využito programu Eagle verze 5.11.0 a následně vyrobena odbornou firmou Gatema a.s. se sídlem v Boskovicích. Ta se zabývá výrobou desek plošných spojů nejen pro sériovou výrobu, ale i pro jednotlivce. Firma nabízí POOL servis, který je určen pro kusové zakázky a tak je možné si nechat vyrobit jeden kus desky plošných spojů.



Obr. 2.3: Osazená deska plošných spojů.

Při návrhu desky bylo vybráno nejvhodnější rozložení jednotlivých modulů pro efektivní využití plochy. Rozměr byl zvolen obdobný, jako je velikost vývojové desky

Arduino Due, a to 10 cm šířka a 6 cm výška. Ze spodní strany desky jsou umístěny piny pro připojení na vývojovou desku Arduino Due. Ze strany vrchní jsou osazeny tlačítka pro ovládání, obvod spínače s optočlenem a konektor pro připojení baterie. Dále je zde kondenzátor pro vyhlazení napěťových špiček vznikajících spínáním krokového motoru a také dutinkové lišty pro připojení modulů zdrojů, krokového motoru, OLED displeje a Wi-Fi modulu.

Schéma zapojení je umístěno v příloze A.1. Ve schématu jsou pro větší přehlednost rozděleny jednotlivé propoje do bloků, které odpovídají názvům modulů. Osazení desky plošných spojů A.3 a desky plošného spoje A.5 se nachází v příloze sekci A. Zde je zobrazeno osazení ze spodní a vrchní strany samostatně. Deska plošných spojů je zobrazena pouze v jedné vrstvě tak, jak byla navrhnuta a vytvořena.

2.3 Mechanická konstrukce

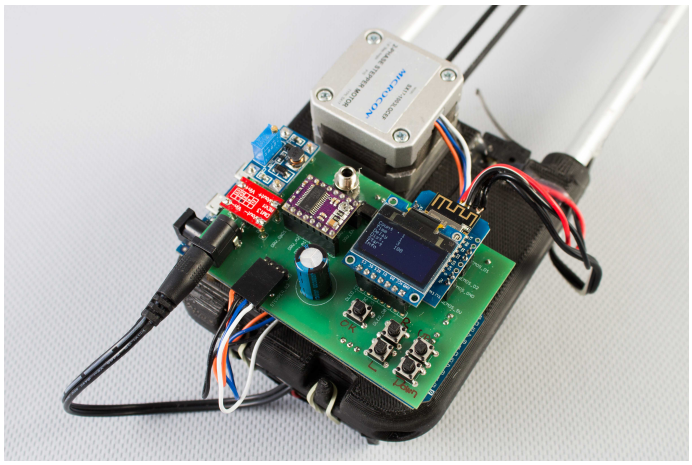
Základem posunu jsou dvě souběžné kolejnice, které jsou zakončeny z obou stran úchyty pro upevnění na stativu. Po kolejnici se pohybuje vozík, na kterém je umístěno kulové upevnění fotoaparátu. Díky tomu je možné nastavit úhel pohledu fotoaparátu. O posun se stará motor umístěný na jedné straně pojezdu a natažený řemen středem celého posunu ke kladce na protější straně.

Pro kolejnice byly použity dvě 80 cm tyče z hliníku, které mají v čele vyvrtané závity. Ty jsou pomocí šroubů připevněny na zakončení s držákem pro stativ. Délku tyčí je možné změnit, jen je nutné zajistit prodloužení nebo zkrácení řemene. V případě, že je požadováno větší vzdálenosti, je nutné zakoupit nový řemen. Ten se prodává v násobcích metru.

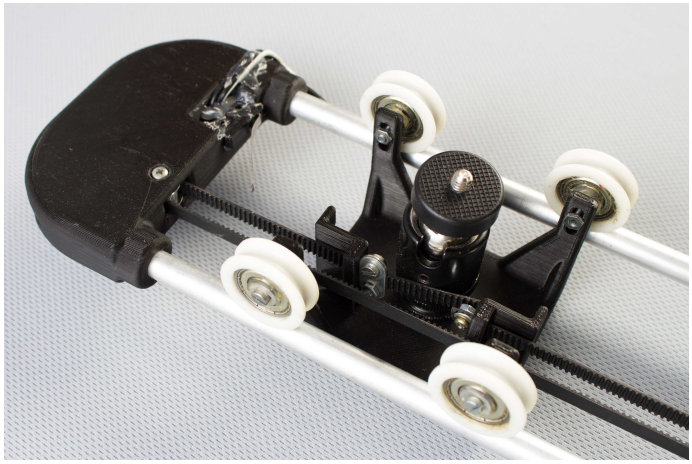
Konstrukce byla rozdělena na 3 části. Tyto části byly vymodelovány v programu SolidWorks verze 2014 a následně exportovány pro tisk na 3D tiskárně. 3D tisk byl proveden u W3D - 3D tisk.

Zakončení s úchyty pro motor a elektroniku je blok o rozměrech 11 cm šířka, 2 cm výška a 15 cm hloubka. Ten je uzpůsoben pro umístění motoru s řemenicí a je v něm připraven průchod pro řemen. Je zde nástavec na tyč použitou pro kolejnice s otvorem na zajišťovací šroub. Dále je zde upevnění na stativ pomocí válce, který se používá pro upevnění osvětlovací techniky u fotografie. Z bloku jsou také vyvedeny trny pro umístění ovládací elektroniky. Poslední úpravou je vykrojení pro umístění koncových spínačů a pro upevnění baterie pomocí gumičky. Model zakončení je v příloze s názvem Model začátku dráhy.

Zakončení druhého konce je menších rozměrů a to 11 cm šířka, 2 cm výška a 6 cm na hloubku. Zde je umístěn pouze úchyt pro stativ, vykrojení pro senzor dorazu a vykrojení pro umístění zpětné kladky pro řemen. Model zakončení je v příloze s názvem Model zakončení dráhy.



Obr. 2.4: Začátek dráhy.



Obr. 2.5: Konec dráhy s vozíkem.

Vozík, který slouží jako pohyblivá část, je navrhnut jako platforma o rozměrech 7 cm šířka, 0,5 cm výška a 8 cm na hloubka. Ze spodní strany je zde výřez pro upevnění kulového uchycení fotoaparátu. Z vrchní strany jsou umístěny konstrukce pro připevnění kol, plochy pro sepnutí dorazových čidel a otvorů pro připojení řemene. Model zakončení je v příloze Model vozíku pojezdu.

Pro složení zařízení je potřeba několik šroubů pro uchycení motoru, kolejnic do koncových modelů, utahováků na stativové válce a pro kladku na druhý konec pojezdu. Dále je doporučeno přilepit senzory dorazu, aby nedošlo k jejich vypadnutí. Posledním doporučovaným úkonem je přilepení dvoulinky k senzoru dorazu na spodní stranu jedné z kolejnic. Dokončené zařízení je na snímku 2.6.

Obr. 2.6: Sestavené zařízení pro automatický posun

2.4 Programové řešení

Program byl sestaven a kompilován v Atmel studiu 7.0 se zásuvným modulem Arduino 1.6/1.8 z řady rozšíření Visual Micro.

Při návrhu kódu byla snaha dodržet model systému založený na kontroléru a stavovém stroji. Díky rozdělení na jednotlivé stavy je možné centralizovaně řídit chování podle stavu v jakém se celé zařízení nachází.

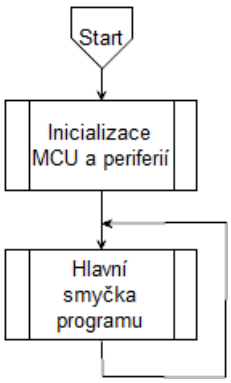
Základním blokem je kontrolér, který provádí požadované procesy a řídí všechny periferie. Ten je implementován hlavně do hodinového přerušení prováděného každou jednu milisekundu. Kontrolér k periferiím nepřistupuje přímo, ale využívá modelu Controller-View. To znamená, že vždy předává informace a dotazuje se na ně pouze kontrolér, nikoli samotný ovladač periferie. Díky tomu nedochází k problému s prioritou procesů a jejich případným cyklickým voláním.

Propojení s Wifi modulem je realizováno pomocí sběrnice I^2C. Původně bylo zamýšleno využít rozhraní SPI, ale knihovny pro toto rozhraní jsou stále ve fázi vývoje. Postrádají možnost fungovat jako podřízené zařízení, a proto bylo rozhraní nakonec změněno na I^2C. Knihovny pro I^2C jsou také ve vývoji, ale již Arduino Due podporuje režim podřízeného zařízení. Díky tomu je omezeno možné nastavení obou MCU a musela být vybrána struktura, kde Wi-Fi modul Wemos D1 je v pozici řídícího rozhraní a Arduino Due je podřízené zařízení. Nakonec se ukázalo, že tato struktura je výhodná i v případě neosazení Wi-Fi modulu, není nutné ověřovat jestli se mají data posílat po I^2C či nikoliv.

Pro komunikaci s Wemos D1 pomocí sběrnice I^2C byla do kódu přidána jedna samostatná obsluha přerušení, která odpovídá na požadavky od Wemos D1. Hodnoty a požadavky ukládá do bufferu. Ten je následně kontrolován kontrolérem a v případě nějakého požadavku jsou provedeny patřičné kroky.

2.5 Program pro Arduino Due

Hlavní kompilovatelný soubor se nazývá *due.ino*. Tento soubor obsahuje volání funkce, která provede počáteční nastavení, a také smyčky systému ve tvaru vyžadovaném kompilerem Arduino IDE. Vývojový diagram je zobrazen na obr. 2.7. Kód napsaný mimo soubor specifický pro kompilátor Arduino byl zvolen z možnosti budoucího využití pro jiné projekty. Pokud by se ke kompilaci pro stejný procesor mělo použít například Atmel studio, stačí vytvořit hlavní soubor a ten provázat se souborem *controller.h*. Inicializaci volat na začátku kódu místo ve funkci *setup* a v hlavní smyčce volat *Controller_loop*. Pokud by v kódu nebyly funkce specifické pouze pro prostředí Arduino, bylo by možné kód zkompilovat a pustit.



Obr. 2.7: Vývojový diagram běhu systému.

2.5.1 Soubor definic

Soubor *defines.h* obsahuje definice funkcí jednotlivých pinů, různé konstanty potřebné pro nastavení periferií a výchozí hodnoty pro rozsah menu. Dále také adresy pro rozhraní I^2C a definice zpráv pro komunikaci s Wi-Fi modulem. Jedná se o soubor, který je vložen do všech jednotlivých souborů, jelikož sdružuje nastavitelné definice. Výhodou této struktury je přehlednost v případě změny některé definic. Například, když je potřeba změnit délku dráhy, není nutné hledat, kde v kódu se nachází její přepočet, ale změní se konstanta v souboru definic.

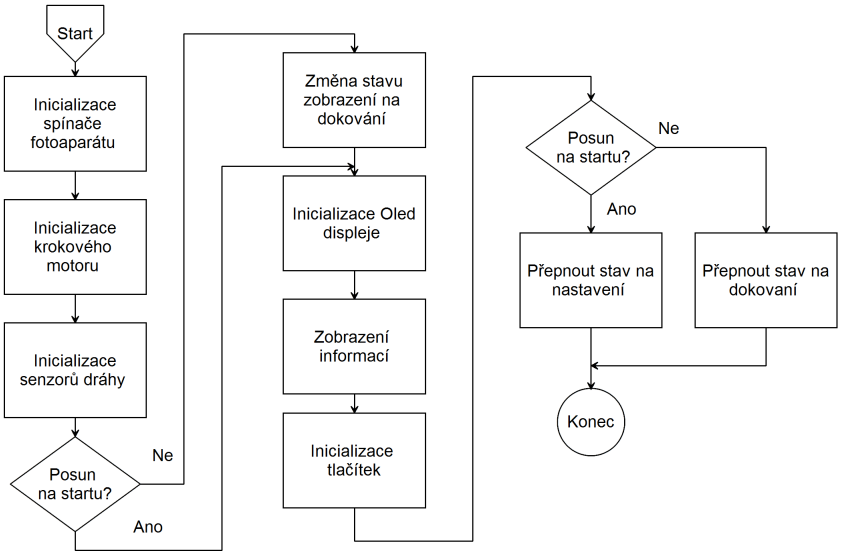
2.5.2 Kontrolér

Je zahrnut v souborech *controller.cpp* a *controller.h*. Zde jsou hlavní funkce celého systému. Patří mezi ně funkce pro úvodní nastavení periferií, smyčka systému, která momentálně není využita a také funkce události hodin. Smyčka se dá použít pro opakované volání funkcí, ale nemusí být časově přesná, jelikož vždy záleží jak dlouho

trval předchozí průchod smyčkou. Proto je při vytváření zařízení využito systémové přerušení prováděné každou 1 ms.

Inicializace

Funkce je pojmenována *initialSetup*. Jedná se o typ void bez parametru. Během inicializace dojde k nastavení jednotlivých periferií do výchozího stavu. Na konci je rozhodovací člen, který určí, do jakého stavu bude zařízení přepnuto, a to podle pozice pojezdu s fotoaparátem. Pokud je pojezd s fotoaparátem mimo svoji výchozí hodnotu, je po skončení inicializace spuštěn mód dokování. Ten spočívá v návratu pojezdu fotoaparátu na výchozí pozici. V případě, že již je pojezd na výchozí pozici, je spuštěn systém do stavu nečinného. Vývojový diagram je možno vidět na obr. 2.8.



Obr. 2.8: Vývojový diagram inicializace systému.

Pravidelná událost hodin

Funkce pojmenovaná *sysTicHookEvent* obstarává pravidelnou obsluhu celého zařízení. Tato funkce je volána systémovým přerušením hodin, a to každou 1 ms. Díky tomu, je vhodná pro využití na již zmiňovaný kontrolér a lze v průběhu funkce vytvářet celkem přesné časovače. Během této funkce nastane v závislosti na stavu, ve kterém se nachází, jak zpracovaní vstupů z klávesnice a zobrazení na displeji, tak obsluha motoru, fotoaparátu a senzorů. Možné stavy a jejich zpracování je na vývojovém diagramu Obr. 2.9 znázorněno pouze jako bloky. Jedná se o kostru události hodin. Jednotlivé zástupné bloky obvykle zastřešují několikanásobné průběhy

kontroly tlačítek, sledování stavu ovladače krokového motoru a další činnosti, které závisí na posloupnosti několika předchozích stavů.

Jednotlivé prováděné operace se dále dělí do funkcí pro ovládání menu pomocí tlačítek. Tato funkce mění hodnoty nastavení, provádí listování řádky a obsluhuje spouštění a ukončení jednotlivých stavů. Dále také předává příkazy pro změny zobrazených hodnot na OLED displeji. Pokud se jedná o změnu hodnoty na řádku, odešle do ovladače OLED displeje číslo řádku a novou hodnotu. Při změně vybraného řádku posílá pouze číslo nového řádku. Změna řádku slouží také jako potvrzení nové hodnoty a ta je předána do bufferu pro obsluhu I^2C komunikace. Tlačítko potvrzení přechází mezi stavy a také mění aktuální zobrazenou stránku. Pokud je použito na spuštění stavu snímání, je tato informace také předána do bufferu stavů pro obsluhu I^2C komunikace.

Další funkcí je obsluha řízení fotoaparátu, která provádí podle nastavených hodnot a aktuální hodnoty časovače určitou sekvenci úkonů pro správné provedení časování snímání. Má také na starosti posun fotoaparátu, předává požadavek na další krok do ovladače krokového motoru a ten odpovídá, jestli tento krok byl již proveden. Funkce pro obsluhu fotoaparátu je další stavový stroj, ve kterém se střídají 3 stavy.

První stav je výpočet všech potřebných hodnot jako je celková délka z nastavených procent. Také je vypočítán počet kroků, které jsou potřeba k posunutí pojezdu o délku jednoho posunu.

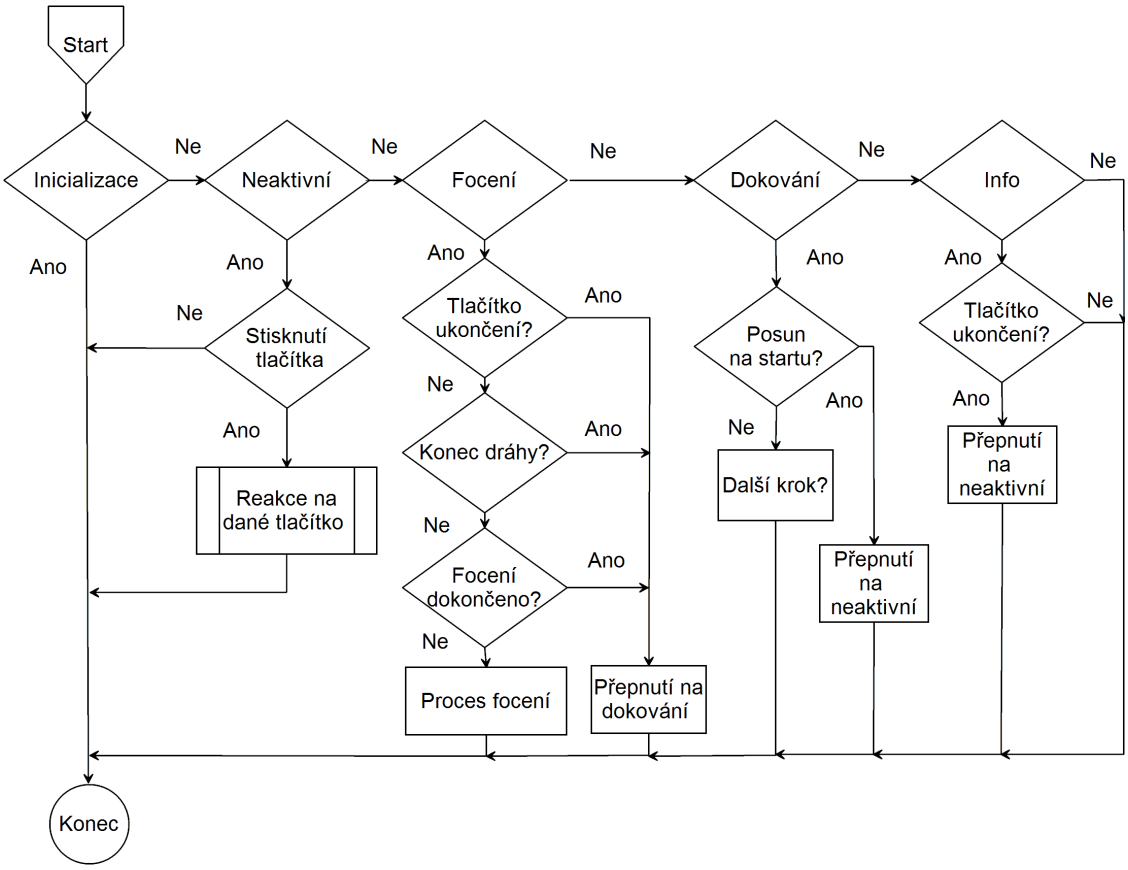
Následuje stav focení, ve kterém jsou již dva časovače, jeden je ovládán pouze z aktuálního stavu a druhý je sdílen se stavem posunu. Ve stavu focení se čeká na uplynutí doby pauzy mezi snímky a následně je sepnut spínač spouště. Dále je počítána doba snímání, poté je spoušť rozepnuta a přechází se na stav zodpovídající za pohyb posunu.

Ve stavu pohybu posunu je prvním průběhem nastaven a spuštěn ovladač motoru, a následně volán tak dlouho, dokud neodpoví, že urazil požadovanou vzdálenost. Během každé kontroly je přičítán časovač prodlevy mezi snímky, aby nedocházelo k neodpovídajícímu časovému rozestupu fotek oproti tomu, který byl uživatelem nastaven. Díky pomalejšímu pohybu je ale možné, že pro malé počty fotografií, jsou časové prodlevy značně zvětšené. Je to dáno tím, že uražená vzdálenost má přednost před časem mezi snímky. Zařízení bylo sestaveno primárně pro velké počty snímků, kde posuny trvají obvykle do 1 s a další 1 s byla brána jako čas potřebný pro ustálení fotoaparátu.

Po dokončení snímání poslední fotografie, nebo dojetí na konec dráhy, je změněn stav na dokování. To ukončí stav focení, restartuje časovače a čítače použité při snímání. Dokování je prováděno jako smyčka, která očekává při dokončení změnu stavu dorazového spínače na stav sepnuto. Je záměrně nastaven větší počet kroků

než maximální pro danou dráhu. Díky tomu je zaručeno dokončení posunutí vozíku až do výchozí pozice. Ukončení stavu je pomocí sepnutí koncového dorazu na začátku dráhy.

Událost dokování a následný posun zařízení je jedním z výchozích stavů zařízení po inicializaci. Pro zamezení nechtěným problémům s posunem a délkou pojezdu je vyžadováno, aby výchozí pozice byla vždy ta počáteční. Její indikací je právě stisknutí dorazového spínače na začátku dráhy.



Obr. 2.9: Vývojový diagram události hodin.

2.5.3 Ovladače periferií

Tlačítka

Skládá se z funkcí pro inicializaci pinů vývojové desky Arduino Due, ke kterým jsou připojeny tlačítka. Piny jsou nastaveny jako vstupní s pull-up rezistorem. Soubor s ovladačem je pojmenován *buttons_driver.cpp* a *buttons_driver.h*.

Pro nastavení pinů na požadovaný stav slouží funkce *pinMode* a následně lze hodnotu pinu vyčíst pomocí funkce *digitalRead*.

Příklad použití:

```
    //nastavení pinu jako vstup s pullup rezistorem
pinMode(PIN, INPUT_PULLUP);
    //uložení logické úrovně na pinu do proměné
hodnota = digitalRead(PIN);
```

Následně je možné použít funkci, jejíž návratová hodnota odpovídá logické úrovni napětí na daném pinu.

Hlavní funkcí je kontrola logické úrovně na požadovaných pinech. V klidovém stavu je na pinu úroveň napětí odpovídající logické 1. Pokud je tlačítko stisknuto, dojde k uzemnění a na pinech je nulové napětí.

OLED displej

Ovladač displeje je založen na komunikaci pomocí rozhraní SPI. Je využito knihovny *Adafruit_GFX.h* [9] a *Adafruit_SSD1306.h* [10]. Knihovna je volně dostupná k použití s BSD Licencí. Díky ní je možné knihovnu využívat svobodně za podmínek zachování původní verze beze změn. Soubory ovladače *oled_driver.cpp* a *oled_driver.h*.

Pro správnou inicializaci knihoven displeje je nutné zachovat následující posloupnost příkazu:

```
 //nastavení parametrů objektu display
Adafruit_SSD1306 display(DataPin, ResetPin, CsPin);

display.setRotation(2); //otočení obrazu o 180 stupňů
display.begin();        //Inicializace obrazu
display.clearDisplay(); //vymazáni dat
display.setTextColor(WHITE); //nastavení barvy
display.setTextSize(1); //nastavení velikosti textu
display.setCursor(X,Y); //souřadnice výchozího bodu
display.print("text");  //uloženní slova "text" do paměti
display.display();      //zobrazení dat z paměti
```

Ovladač displeje je rozdělen na několik samostatných funkcí. Jako základ bylo navrhnuto menu nastavení. V tomto menu je potřebné měnit hodnoty na určitých pozicích a také mít možnost tímto nastavením listovat. Jsou zde funkce pro zobrazení celé stránky najednou, nahrazení hodnoty na určitém místě a posun ukazatele řádku. Dále také funkce po sestavení obrazovky stavů dokování a informací o autorovi. Poslední je obrazovka stavu focení, kde je také aktualizováno číslo vyfoceného snímku.

Optický spínač

Soubory jsou pojmenovány *opto_switch.cpp* a *opto_switch.h*. Zde se nachází inicializace pinu, na který je napojen optočlen ve funkci spínače. Pin je nastaven jako výstupní. Ovladač obsahuje funkci, která podle parametru nastavuje hodnotu na pinu, a tím mění stav spínače. Jedná se pouze o přepínaní stavu z logické 0 na logickou úroveň 1 a zpět.

Senzory

Kromě inicializace pinů senzorů obsahuje i funkci pro získání aktuálního stavu senzorů a funkce pro vytvoření a zrušení přerušení. Vyčtení aktuálních hodnot vrací hodnotu 0 pokud není stisknut žádný senzor, pokud je aktivován senzor začátku je vrácena hodnota 1 a při stisknutí senzoru konce hodnota 2.

Přerušení na pinech senzorů je využíváno pro detekci změny stavu spínačů využitých jako senzory dojezdu. Přerušení proběhne v případě, že byla zaznamenána změna logické úrovně na pinu senzoru. Jedná se o odlehčení systému, díky kterému nedochází k pravidelnému čtení hodnoty pinů, ale pouze při změně. Ta je signalizována proměnou sloužící jako vlajka. Čtení těchto pinů by jinak probíhalo vždy, když je pojezd v pohybu. Ovladač je umístěn v souborech *sensors_driver.cpp* a *sensors_driver.h*.

Inicializace přerušení je vytvořena pomocí následující posloupnotsti:

```
typedef void (*GeneralFunction) (); //definice typu
GeneralFunction ukazatelFunkce; //vytvoření objektu

void funkce(void){
    //Definice funkce
}

ukazatelFunkce = funkce; //nastavení ukazatele funkce
    //vytvoří přerušení spouštěné změnou hodnoty na PINu
attachInterrupt(digitalPinToInterrupt(PIN),
                    ukazatelFunkce,CHANGE);
```

Krokový motor

Ovladač krokového motoru je vytvořen jako vrstva nad knihovnou *DRV8825.h* pro uzpůsobení jejich funkcí. Knihovna ovladače krokového motoru pochází od autora Laurentiu Badea a je k dispozici pod licencí MIT [11].

V nově vytvořené vrstvě je funkce pro inicializaci ovladače, ta nastaví rychlost otáčení a počet kroků. Dále také provede deaktivování krokového motoru, díky tomu není spotřebovávána energie stojícím motorem předtím, než je spuštěno snímání. Funkce provádění kroků je volána cyklicky, dokud návratovou hodnotou není hodnota menší než 0. Poté je motor deaktivován. Funkce pro zastavení zabrzdí motor a zajistí převod do režimu spánku, tím je dosaženo minimálního odběru ve stavu, kdy není motor využíván. Ovladač krokového motoru je nastaven na násobení počtu kroku šestnácti. Tím je dosaženo většího počtů kroků na dráhu, a tím pádem i přesnosti a maximální možný počet snímků je zvýšen.

Nastavení ovladače knihovny:

```
 //nastavení parametrů objektu motor a pinů
DRV8825 motor(PočetKroků, SměrPin, KrokPin, EnPin);
 //nastavení rychlosti otáčení a násobení kroků
motor.begin(RPM, MICROSTEPS);
motor.enable(); //aktivování ovladače
motor.startMove(kroky) //nastavení počtu kroků

//smyčka pro provedení kroků s kontrolou dokončení kroků
while(motor.nextAction() <= 0) {
      motor.disable();  //deaktivování ovladače
}
```

2.5.4 Ovládání I2C komunikace

V původním návrhu byla komunikace navržena pro SPI rozhraní. Jeho využití bylo výrazně omezeno, díky nedokončené podpoře módu reagovat jako podřízené zařízení u obou MCU. Po otestování různých nastavení podporovaných rychlostí a časování bylo pro komunikaci použito sběrnice I^2C.

Přenosová rychlost pro rozhraní I^2C byla nastavena na základní, a to 100 kbit/s. Zkoušeny byly i rychlosti vyšší, ale díky jednoduchosti komunikačního protokolu a malému objemu dat byla ponechána nízká rychlost. Tím je zajištěna stabilita komunikace, s kterou se mohou objevit problémy při vysoké rychlosti rozhraní. Ty vznikají například při nedodržení požadavků na přenosovou trasu. Komunikace je také více náchylná na rušení z okolního prostředí.

Hlavní kontrolér, který řídí chod MCU, nastavuje vlajky, když se změní některá z hodnot, která má být synchronizována s Wi-Fi modulem. Dále také hlavní kontrolér předává nová data do vyrovnávací paměti. I^2C kontrolér při příjmu požadavku na aktualizaci, zjistí jestli nedošlo k nějaké změně a předává data o aktuálním stavu zpátky nadřízenému zařízení, tedy Wi-Fi modulu. Pokud Wi-Fi modul předává nová

data, ta jsou uložena ve vstupní vyrovnávací paměti a kontrolér si data přebírá v pravidelných intervalech. O přítomnosti nových dat nebo příkazů se dozví z nastavených vlajek, sloužících pro signalizaci nových dat nebo příkazů.

Komunikační protokol

Základem je udržování komunikace. To znamená, že řídící zařízení se v pravidelných intervalech dotazuje a podřízené zařízení odpovídá, i když nenastala žádná událost. Díky tomu má řídící zařízení přehled jestli nedošlo ke selhání nebo odpojení podřízeného zařízení. Po I^2C sběrnici probíhá komunikace pomocí 8 bitů, aby se ušetřilo množství dat, jsou zprávy navrženy jako 8 bitové slovo.

Navázání spojení (0x01): probíhá jako první komunikace a z důvodu zrychlení je interval mezi posílání zpráv zkrácen na 300 ms, a to pro 3 opakované pokusy. Jedná se o zprávu, na kterou je možná pouze jedna odpověď, a to 0x10.

Kontrola stavu (0x02): jestliže bylo navázání stavu úspěšné, je nadále již posílán příkaz označující kontrolu stavu. Na ten je možné odpovědět právě 3 zprávami podle stavu podřízeného zařízení. Jestliže nedošlo ke změně, je poslán paket obsahující 0x20. Při změně hodnot je odpovězeno příkazem 0x30 a podřízené zařízení čeká na vyžádání nově nastavených hodnot. Poslední odpovědí je začátek focení 0x40. Po přijetí této zprávy je řídící zařízení, přepnuto do režimu focení.

Změna hodnot (0x03): je posílána pokud byly změněny data na straně řídícího zařízení. Podřízené zařízení by mělo odpovědět pomocí potvrzení přijímání, a to hodnotou 0x20. Následuje přenos dat směrem do podřízeného zařízení. Pokud ale již došlo ke změně hodnot na straně pořízeného zařízení, pak odpověď je 0xF0. Na to se musí zařízení nadřízené optat na aktuální stav. Tento stav může být vyvolán spuštěním režimu snímání a nebo změnou hodnot na HW tlačítkách, které mají přednost.

Inicializace snímání (0x04): pokud je podřízené zařízení v neaktivním stavu a nedošlo ke změně nastaveni, odpoví potvrzením pomocí 0x50 a následně zahájí režim snímání. Nastane-li ale nějaká změna, podřízené zařízení žádost odmítne pomocí 0xF0 a řídící zařízení se musí dotázat na stav.

Smyčka snímání: během této události je komunikace omezena na pouhé dva druhy zpráv. První se dotazuje na počet vyfoceních fotografií a druhá ruší režim snímání.

Získání pořadového čísla snímku (0x05): tato kontrola probíhá ve stejných intervalech jako kontrola stavu. Podřízené zařízení může na tento požadavek odpovědět pořadovým číslem, nebo zprávou o ukončení režimu snímání. Z důvodu možného počtu snímků vyššího než 255, je toto jediná zpráva, která se skládá ze 2 paketů sběrnice I^2C, a to 0xFFFF. Díky tomu je maximální počet fotografií omezen na

65634 snímku.

Zrušení režimu snímání (0x0F): po přijetí této zprávy dojde na podřízeném zařízeni k zrušení režimu snímání a je odpovězeno pomocí zprávy 0x60. Ta značí, že zpráva byla přijata.

Neočekávané chování: pokud je přijat jiný příkaz, než v daném stavu je očekáváno, vede toto ke zkrácení doby intervalu mezi zprávami a opakování požadavku.

Pokud není spojení navázáno, Wi-Fi modul přejde do chybového režimu a neumožňuje žádné operace, dokud není spojení navázáno. Pro Due má komunikace s Wi-Fi modulem pouze přidanou hodnotu, a pokud není přijat žádný požadavek, nemá to vliv na chod zařízení.

2.6 Program pro Wemos D1

Hlavní soubor pro Arduino IDE se nazývá *D1.ino*. Tento soubor obsahuje volání funkce, která provede počáteční nastavení a také obsluhy systému. Základní vývojové prostředí nepodporuje ESP8266 modul, proto je nutné doinstalovat knihovnu ESP8266 [12]. Díky ní je pak možné vytvářet program pro Wi-Fi stejným způsobem, jako tomu je pro Arduino platformu.

Program je rozdělen do 3 bloků. Ty jsou také jednotlivě inicializovány a obsluhovány samostatně. Rozdělení je vytvořeno pro lepší přehlednost. Jedná se o inicializaci a obsluhu Wi-Fi, webového rozhraní a také kontroléru.

2.6.1 Soubor definic

Soubor *Defines.h* obsahuje definice použitých pinů, definici I^2C adres a příkazů a různé konstanty potřebné pro časování událostí. Část obsahující I^2C je stejná pro Due.

2.6.2 Kontrolér

Je zahrnut v souborech *controller.cpp* a *controller.h*. Zde jsou hlavní funkce pro spojení komunikace mezi uživatelem a I^2C rozhraním, a s tím tedy Arduinem Due. Mezi hlavní funkce patří úvodní nastavení periferií a smyčka systému.

Během inicializace je provedeno nastavení informační diody obsažené na vývojové desce a zavolána inicializace I^2C rozhraní.

Funkce nazvaná *Controller_loop* slouží jako obsluha komunikace, která je volaná pravidelně. Ve funkci se jako první zkontroluje, jestli již uplynul stanovený počet cyklů, aby nedocházelo k zbytečně časté komunikaci mezi MCU. Pokud uplynul

požadovaný čas, je zkontrolováno v jakém stavu je kontrolér. Stavy se přepínají v závislosti na událostech.

Inicializace: je prvotní stav, při kterém se Wi-Fi modul snaží navázat spojení s Arduinem Due. Stav trvá 3 pokusy a podle úspěšnosti nastaví zobrazovanou stránku. Při neúspěchu přejde do stavu značícího problém s komunikací. Pokud uspěje se spojením, nastaví internetovou stránku na formulář, kterým se zařízení nastavuje a přejde do režimu aktivního.

Dotazování: pravidelně se dotazuje na stav Arduina Due a podle odpovědi přejde do příjmu dat, režimu snímání, nebo pokračuje dále v běhu. Pokud dojdou data jiná, než očekávána, zkrátí čas opětovného pokusu o dotaz.

Odesílání dat: pokud jsou nová data k dispozici, je odeslána informace a očekává se odpověď. Reakce na ni může být buď odeslání dat, dotázání znovu nebo přejití na režim dotazování.

Příjem dat: je vyvolán po přijetí informace, že jsou připravena nová data. Po přijetí dat je zavolána funkce, aktualizující data na internetové stránce.

Režim snímání: Dotazuje se pravidelně na pořadové číslo vyfoceného snímku. Pokud přijme informaci o konci snímání, přepne se na režim dotazování. V případě, že je z internetové stránky přijat požadavek na ukončení snímání, je odeslán tento požadavek a při potvrzení dojde k ukončení režimu snímání.

Spojení nenavázáno: pokud neproběhne navázání spojení, je pokračováno v pokusu spojení navázat, ale již s menší frekvencí. Opustit tento stav je možné pouze úspěšným navázáním spojení. V tomto stavu je uživatel informován o chybě pomocí internetové stránky.

2.6.3 Ovládání Wi-Fi

Ovladač Wi-Fi je založen na knihovně *ESP8266WiFi.h*. Funkce *Wifi_init* slouží k nastavení a vytvoření Wi-Fi přístupového bodu. O udržování chodu přístupového bodu se následně starají funkce definované Wi-Fi Stackem. Vytvořená Wi-Fi síť má SSID *HyperT* a její heslo je *thereisphoto*.

2.6.4 Komunikace I2C

Soubory ovladače sběrnice I^2C obsahují vrstvu vytvořenou nad knihovnou *Wire.h*. Jedná se o funkce, které se starají o správné pořadí příkazů. Díky tomu je zpřehledněn kód kontroléru.

Posílání příkazů a dat je složeno z funkce pro nastavení odesílání, tím je nastavena adresa příjemce. Další částí je odeslání dat a přenos je ukončen. Pro vyžádání příkazů a dat je využito funkce, která posílá na podřízené zařízení paket značící požadavek

na čtení. Po vyčtení dat je možné je získat ze vstupní vyrovnávací paměti. Tyto data jsou předávány jako návratová hodnota funkcí.

Příklad dotázaní na data o velikosti 5 bajtů:

```
uint8_t temp[5]; //vytvoření proměnné pro uložení dat
    //posílání žádosti na příjem 5 bajtů na danou adresu
Wire.requestFrom(ADDRESS,5);
    //provedení pro všechny přijaté bajty
while (Wire.available()) {
        //přečtení a uložení bajtů
    temp.data[i] = Wire.read();
    i++;
}
```

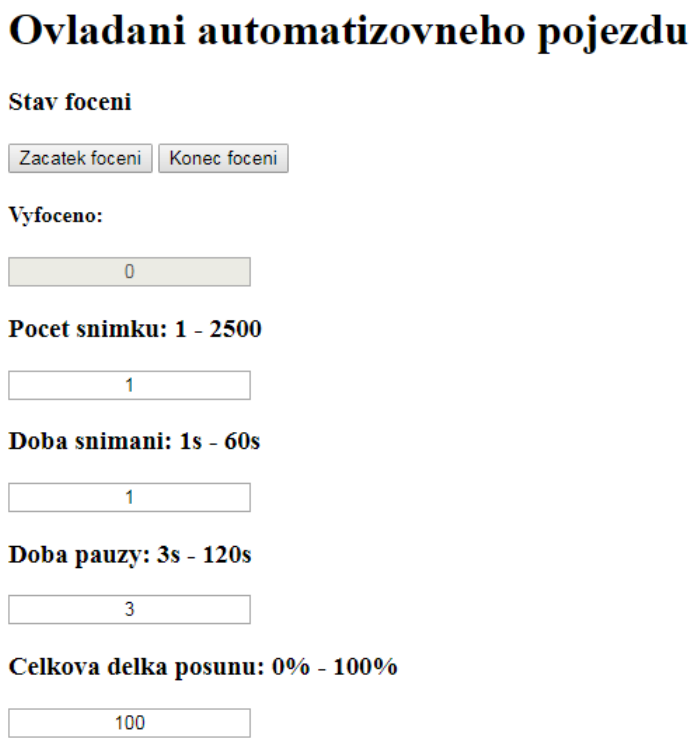
2.6.5 Řízení internetové stránky

Základem pro internetovou stránku je Web server *ESP8266WebServer.h* a Web Socket server *WebSocketsServer.h*. Další částí je překladač adres *ESP8266mDNS.h*, který funguje jako přesměrování požadavku z daného portu na konkrétní obslužné funkce. Knihovna pro WebSocket server je dostupná pod GNU licencí na [13].

Internetová stránka je do paměti uložena jako konstantní proměna. Jsou vytvořeny dvě varianty, první obsahuje hlášku, že nebylo možné se spojit se zařízením a ta druhá obsahuje formulář pro nastavení a zobrazení jednotlivých položek 2.10.

Základním jazykem pro internetovou stránku je HTML v kombinaci s funkcemi Websoketu. Struktura stránky je vytvořena jako tabulka se vstupními poli. Ty již mají přednastavenou hodnotu a v případě přijetí změny se aktualizují.

Po načtení stránky uživatelem, je vyvolána funkce, která vytvoří spojení mezi prohlížečem uživatele a Wi-Fi modulem. Jako první zpráva je odeslán aktuální stav zařízení a tím aktualizovány hodnoty na straně uživatele. Pokud uživatel zadá hodnotu nebo zapne režim snímání, dojde k přeložení dané funkce na paket Websocketu a ten je poslán na zařízení. Wemos po přijetí zprávy převede data z textu na hodnoty a ty zpracuje. Pokud příchozí hodnoty odpovídají hodnotám očekávaným, dojde k jejich předáním kontroléru. Jestliže se hodnoty vymykají rozsahu, dojde k jejich upravení na akceptovatelné maximální nebo minimální hodnoty. Server vždy posílá odpověď o výsledku všesměrovou zprávou pro případ, že by byly zároveň připojeny dvě a nebo více zařízení.



Obr. 2.10: Internetová stránka.

2.6.6 Připojení na internetovou stránku s ovládáním

Pro připojení na internetovou stránku je vyžadováno zařízení, které obsahuje Wi-Fi odpovídající standardu *IEEE 802.11 b/g/n*. Například je možné použít Smartphone, nebo notebook. Zařízení pro automatický časosběr poskytuje Wi-Fi síť, která má SSID *HyperT* a její heslo je *thereisphoto*. Po připojení na tento přístupový bod, je možné otevřít pomocí internetového prohlížeče, internetovou stránku s tabulkou pro nastavení zařízení.

Adresa pro zobrazení stránky je *192.168.4.1*. Stránka obsahuje tlačítka pro spuštění a zastavení snímání, dále také informaci o počtu nasnímaných fotografií a pole pro nastavení hodnot. Při zapsání nových hodnot do pole a kliknutí mimo pole, hodnota aktualizována i na samotném zařízení. Pokud jsou data změněna na zařízení, jsou hodnoty aktualizovány na internetové stránce.

2.7 Přenos dat z fotoaparátu

Poslední částí, která byla navrhnuta v zadání diplomové práce, je přenos dat do počítače v reálném čase. Původně mělo zařízení být připojeno k fotoaparátu pomocí

USB a ovládáno přes protokol MTP. Díky změně sběrnice pro komunikaci mezi MCU a Wi-Fi modulem, a to na I^2C rozhraní, byla snížena maximální přenosová rychlost. Výsledná rychlost již není dostatečná pro přenos fotografií. Maximální rychlost pro I^2C komunikaci je kolem 50000 bajtů za sekundu. Fotografie, snímané digitálním fotoaparátem, mají obvykle velikost kolem 6000000 bajtů, tedy by pro přenos jedné fotografie bylo potřeba přibližně 2 minuty. Během této doby je v nejrychlejším nastavení časosběrného zařízení nasnímáno 30 nových fotografií. Nebyla by tedy splněna podmínka reálného času.

Pro odstranění problémů s přenosovým kanálem by bylo nutné použít MCU, který má nativní podporu USB a zároveň integrovaný Wi-Fi modul. Další možností je propojení MCU a Wi-Fi modulu pomocí sběrnice SPI. Bylo by tedy potřebné změnit MCU a předělat strukturu programu.

Posledním řešením je použití karet, které kombinují úložné zařízení a Wi-Fi modul. Díky tomu je odstraněn problém s omezeným přenosovým kanálem a fotografie jsou po dokončení snímání k dispozici v lokální síti, kam je karta připojena jako síťové úložiště. Srovnání Wi-Fi karet od různých výrobců je k dispozici zde [14].

3 ZÁVĚR

Výsledkem diplomové práce je funkční vzorek automatického posunu pro časosběrné snímání fotografií. Na vozík zařízení je možné upevnit fotoaparát, pomocí kulové stativové hlavy a připojit ho k zařízení pomocí kabelové spouště. Následně má uživatel možnost nastavit počet snímků, které mají být nasnímané. Dále také je možné nastavit dobu snímání jednotlivých snímků a dobu čekaní mezi snímky. Poslední položkou na nastavení je délka dráhy v procentech. Po nastavení a spuštění zařízení automaticky posouvá fotoaparát a snímá fotografie.

Nastavení je možné provést pomocí tlačítek na samotném zařízení a zkontrolovat nastavené hodnoty na OLED displeji. Nastavit hodnoty je také možné na internetové stránce. Ta je dostupná po připojení na Wi-Fi přístupový bod s SSID *HyperT* a heslo je nastaveno na *thereisphoto*. Zařízení má adresu *192.168.4.1*, a na té je také dostupná internetová stránka s nastavením.

Pro sestavení funkčního vzorku byly vytvořeny modely tří hlavních částí a ty vytištěny pomocí 3D tisku. Dále byla navrhnuta a vyrobena deska plošných spojů.

Pro budoucí vývoj je zvažováno použití vývojového modulu *Wemos Lolin32*. U něho je možné využít technologie Bluetooth, pomocí které by zařízení mohlo být ovládáno z mobilní aplikace. Deska je také menších rozměrů a díky tomu by bylo vytvořeno kompaktnější zařízení.

LITERATURA

[1] Arduino *Due* [online]. Specifikace na stránkách výrobce [cit. 29. 11. 2017]. Dostupné z URL:
<https://store.arduino.cc/usa/arduino-due>.

[2] Wemos *D1* [online]. Specifikace na stránkách výrobce [cit. 25. 4. 2018]. Dostupné z URL:
<https://wiki.wemos.cc/products:d1:d1_mini>.

[3] Vše pro 3D tisk: *Krokový motor SX17-1003LQCEF* [online]. Specifikace krokového motoru [cit. 29. 11. 2017]. Dostupné z URL:
<www.vsepro3dtisk.cz/p/krokovy-motor-3d-tiskarna-sx17-1003lqcef>.

[4] Pololu: *Ovladač krokového motoru DRV8825* [online]. Specifikace ovladače krokového motoru [cit. 29. 11. 2017]. Dostupné z URL:
<https://www.pololu.com/product/2133>.

[5] Everlight: *Optočlen EL817 a103* [PDF]. Specifikace optočlenu [cit. 29. 11. 2017]. Dostupné z URL:
<https://everlightamericas.com/index.php?controller=attachment?id_attachment=3225>.

[6] Eclipsera: *Modul* OLED Displeje [online]. Specifikace modulu OLED displeje [cit. 25. 4. 2018]. Dostupné z URL:
<goo.gl/N6kgV7>.

[7] Eclipsera: *Step-UP měnič 2A nastavitelný SX1308* [online]. Specifikace modulu měniče s výrobním číslem 1458853583 [cit. 26. 4. 2018]. Dostupné z URL:
<https://arduino-shop.cz/docs/produkty/0/71/1458853583.pdf>.

[8] Eclipsera: *Step-UP měnič na 5 V a 2A* [PDF]. Specifikace na stránce prodejce [cit. 26. 4. 2018]. Dostupné z URL:
<goo.gl/k2E7JZ>.

[9] Adafruit: *Knihovna obsahující grafické jádro* [GitHub]. [cit. 3. 12. 2017]. Dostupné z URL:
<https://github.com/adafruit/Adafruit-GFX-Library>.

[10] Adafruit: *Knihovna obsahující ovladač pro* OLED dislej SSD1306 [GitHub]. [cit. 3. 12. 2017]. Dostupné z URL:
<https://github.com/adafruit/Adafruit_SSD1306>.

[11] Laurentiu Badea: *Knihovna obsahující ovladač pro driver krokového motoru DRV8825* [GitHub]. [cit. 3. 12. 2017]. Dostupné z URL: <https://github.com/laurb9/StepperDriver>.

[12] ESP8266 Community: *Knihovna obsahující ESP8266 jádro pro Arduino IDE* [GitHub]. [cit. 26. 4. 2018]. Dostupné z URL: <https://github.com/esp8266/Arduino>.

[13] Links2004: *Knihovna WebSockets server pro platformu Arduino* [GitHub]. [cit. 26. 4. 2018]. Dostupné z URL: <https://github.com/Links2004/arduinoWebSockets>.

[14] Dharmesh Goyal: *5 Best WiFi SD Cards For Cameras & Other Devices* [článek]. [cit. 4. 5. 2018]. Dostupné z URL: <https://technofizi.net/best-wifi-sd-cards>.

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

GPIO	General-purpose input/output, piny s předem nedefinovanou vlastností
I^2C	Inter-Integrated Circuit, sériová počítačová sběrnice
MCU	Mikrokontrolér, jednočipový počítač
OLED	Organic light-emitting diode, displej využívající organických elektroluminiscenčních diod
LED	light-emitting diode, světlo vyzařující dioda
PTP	Picture Transfer Protocol, protokol pro přenos multimediálních souborů
PWM	Pulse Width Modulation, pulzní šířková modulace
SPI	Serial Peripheral Interface, sériové periferní rozhraní
SSID	Service Set Identifier, Identifikátor Wi-Fi sítě
USB	Universal Serial Bus, univerzální sériová sběrnice
Wi-Fi	Wireless local area network, bezdrátovou komunikaci v počítačových sítích

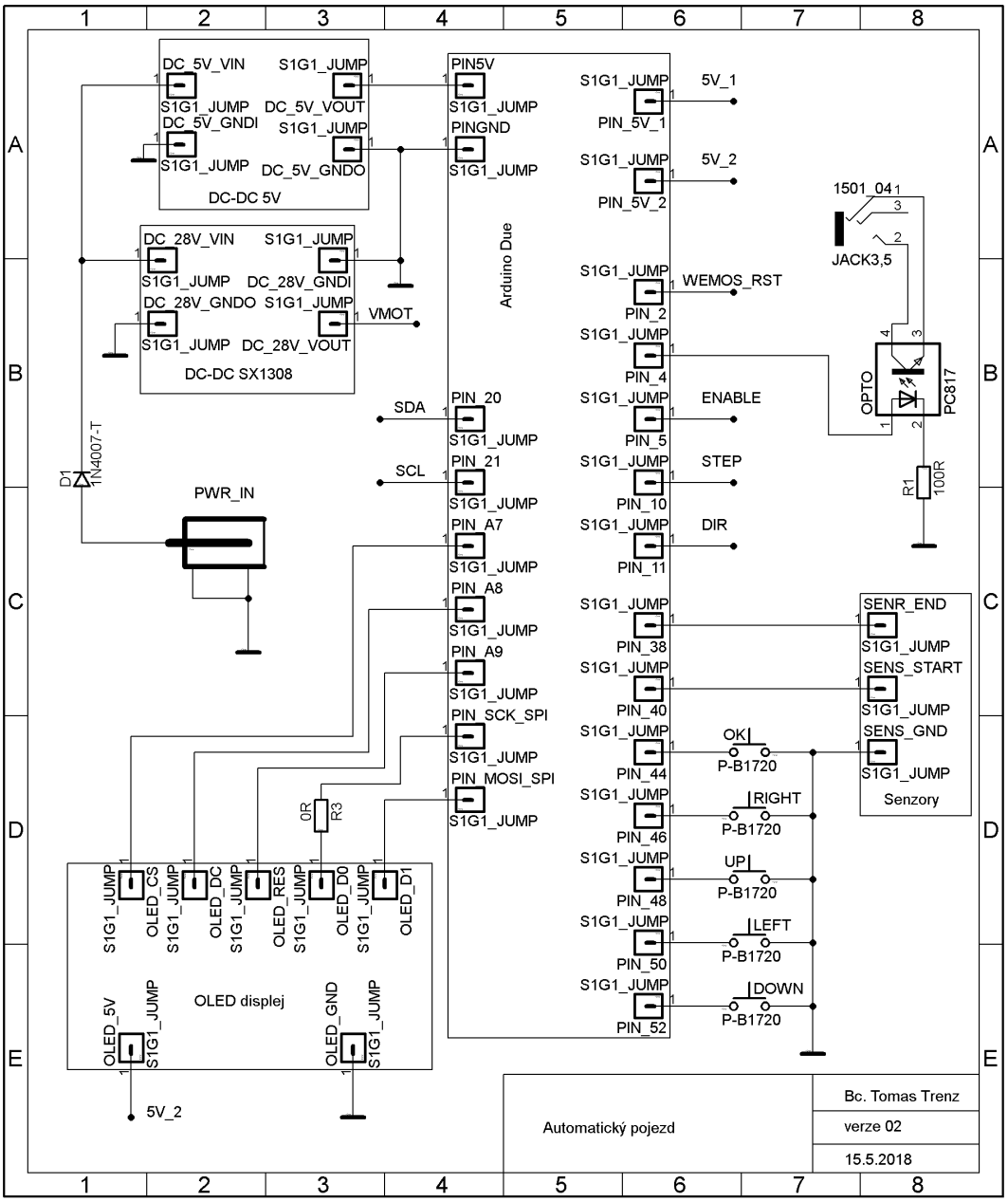
SEZNAM PŘÍLOH

A Schéma zapojení a osazení **46**

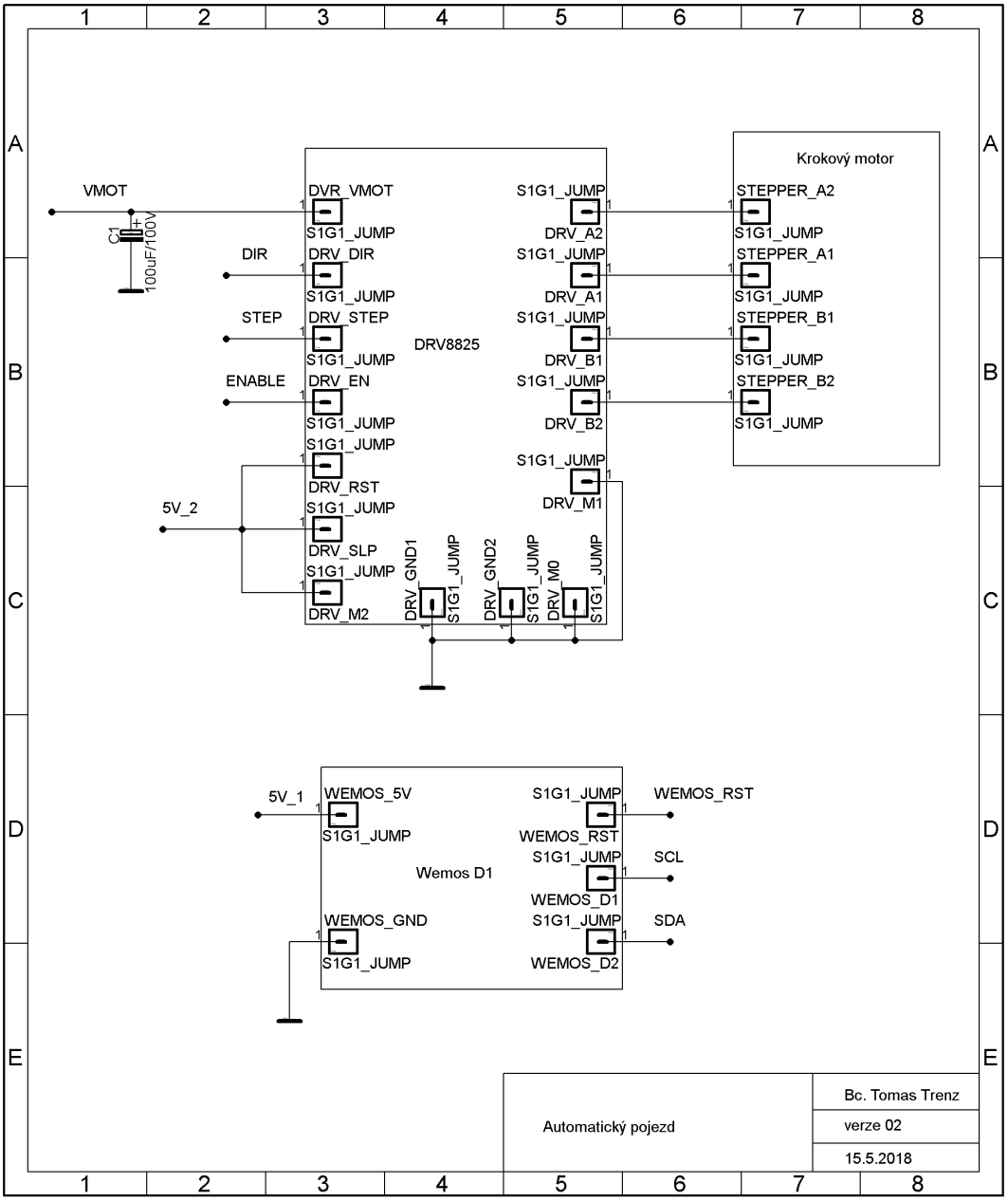
B Obsah přiloženého CD **57**

A SCHÉMA ZAPOJENÍ A OSAZENÍ

Schéma zapojení

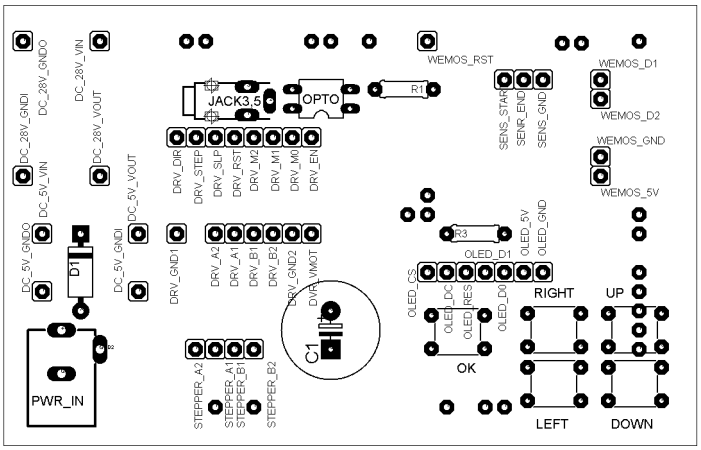


Obr. A.1: Schéma zapojení modulů 1/2.

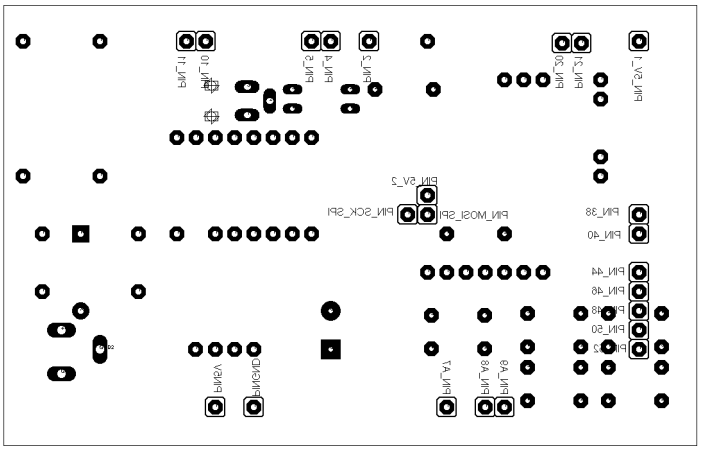


Obr. A.2: Schéma zapojení modulů 2/2.

Osazení

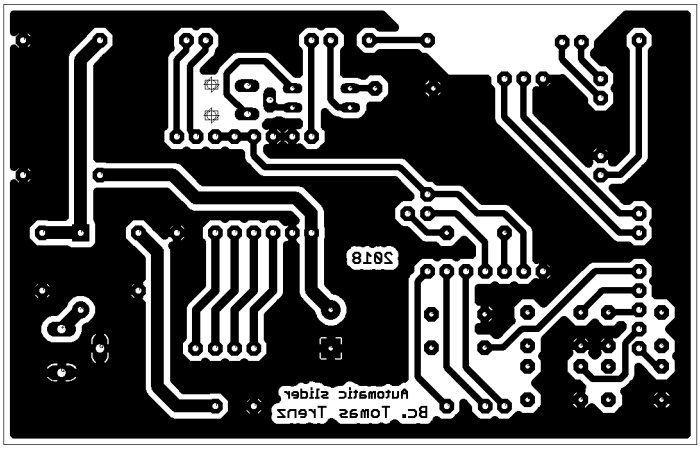


Obr. A.3: Osazení horní strany.

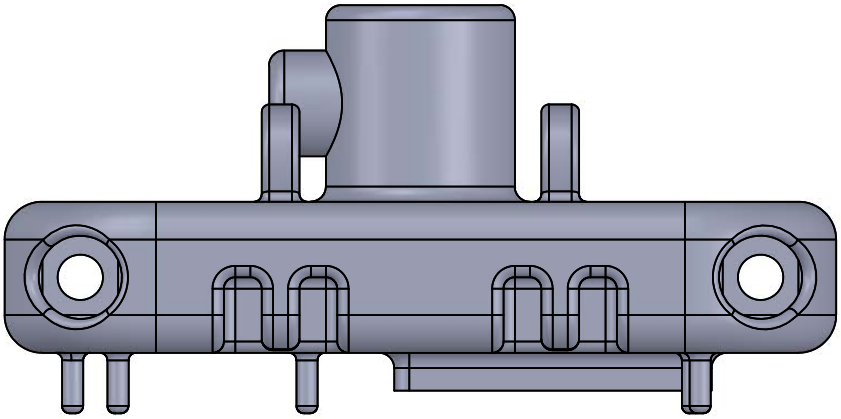


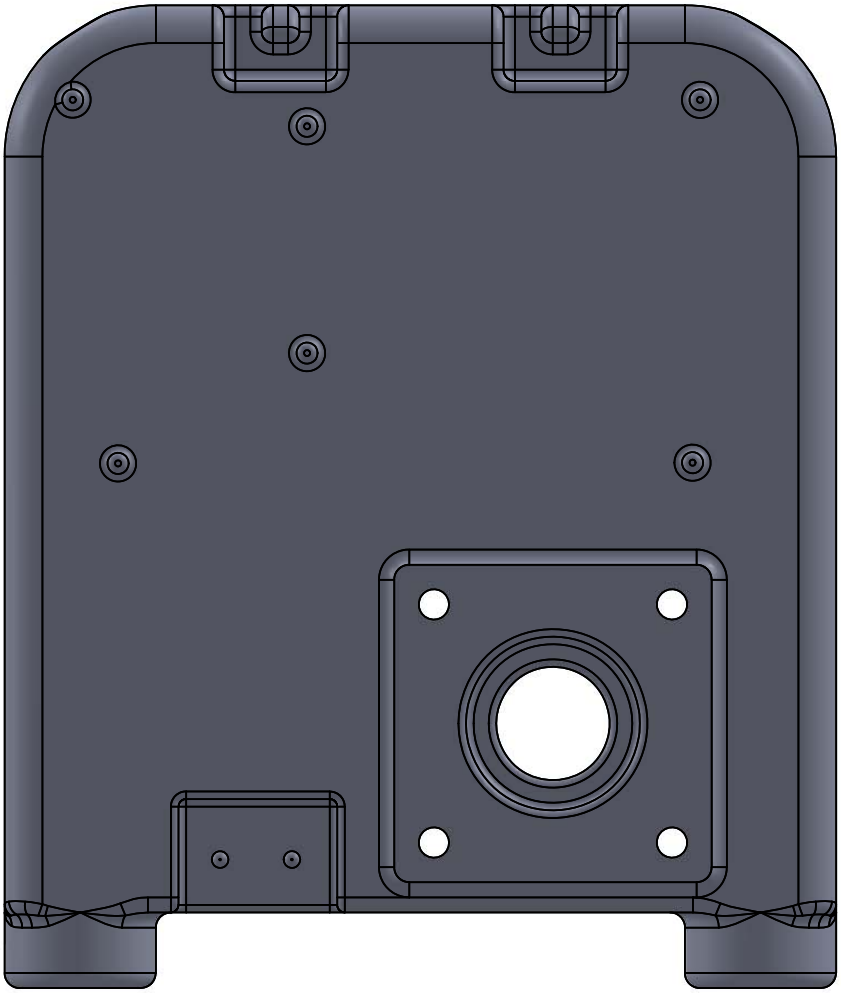
Obr. A.4: Osazení spodní strany.

Plošný spoj prototypové desky.

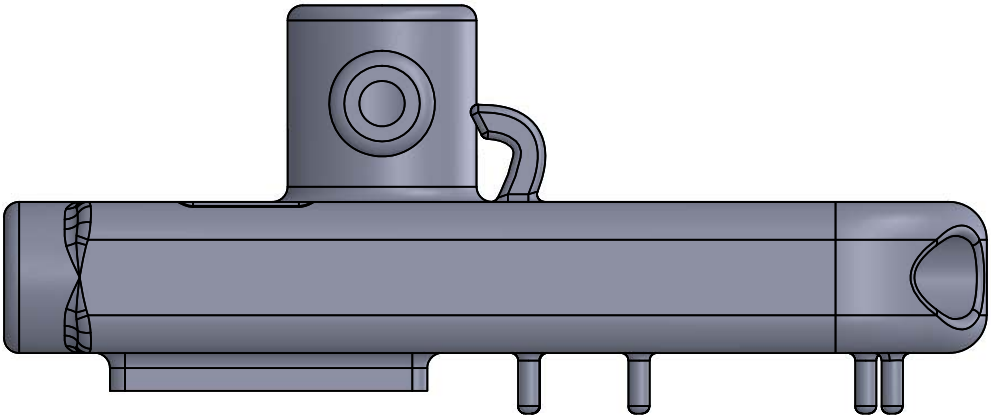


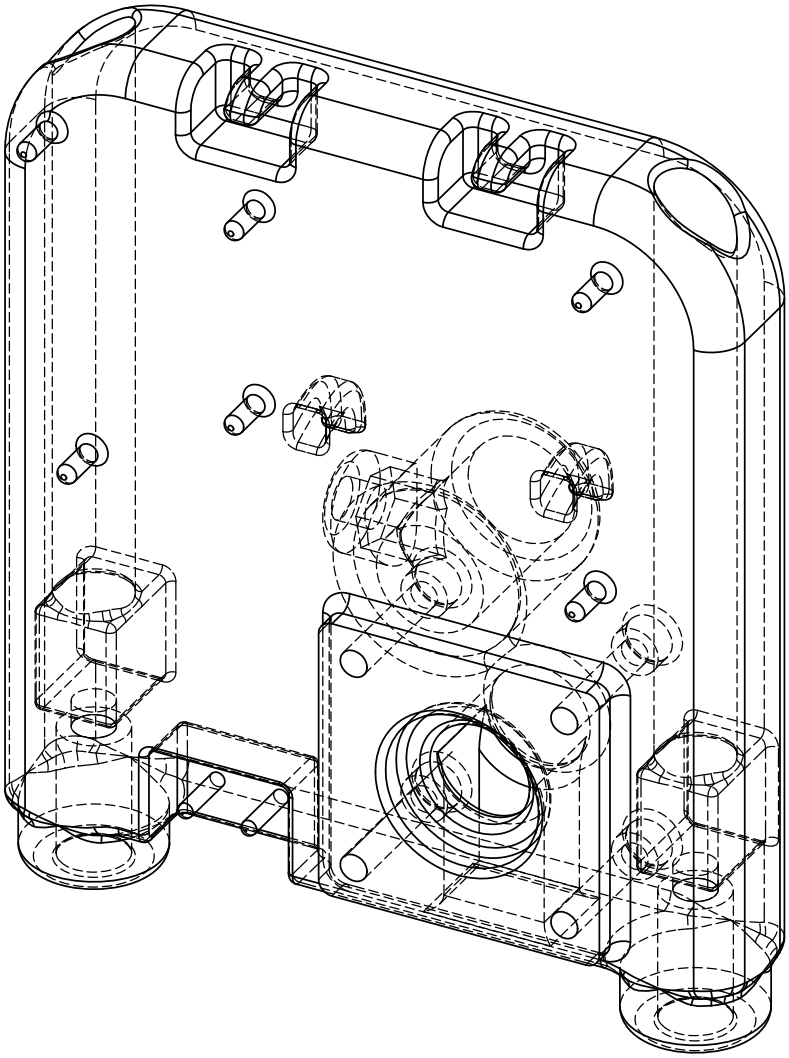
Obr. A.5: Plošný spoj spodní strana.



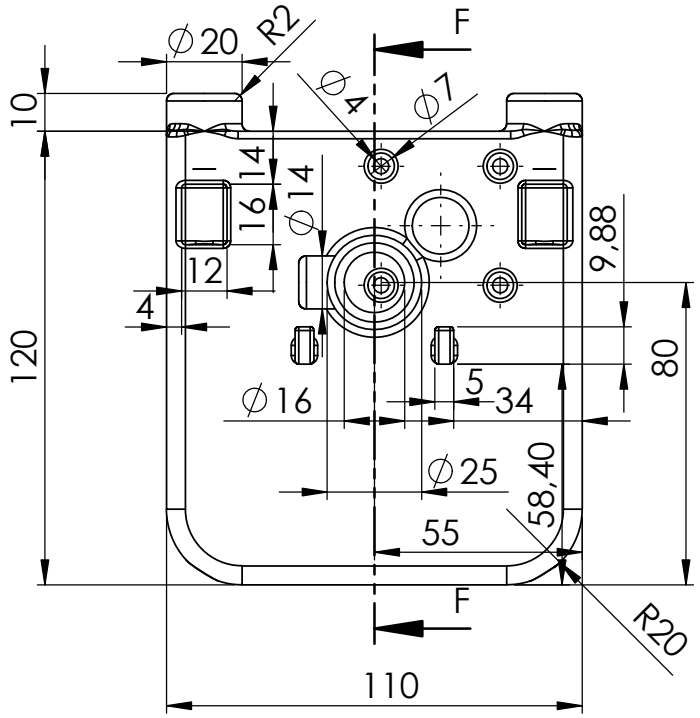


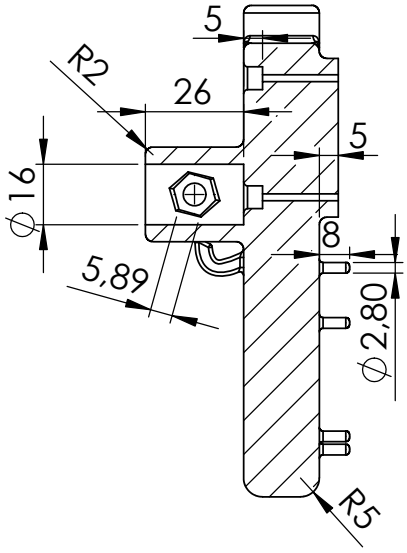
	NÁZEV	PODPIS	DATUM	MATERIÁL:	NÁZEV:	
NAVRHL	Bc. Tomáš Trenz		25.4.2018	ABS	Model začátku dráhy	A4
SCHVÁLIL						
VÝROBA					MĚŘÍTKO: 1:1	LIST 1 Z 2 LISTŮ



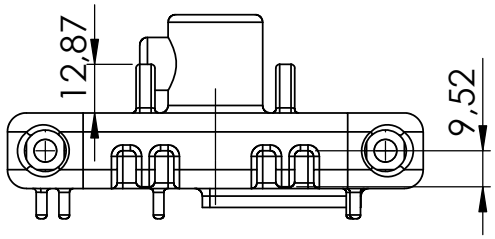


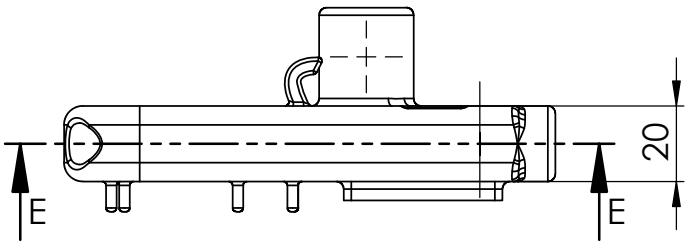
	NÁZEV	PODPIS	DATUM	MATERIÁL:	NÁZEV:	
NAVRHL	Bc. Tomáš Trenz		25.4.2018	ABS	Model začátku dráhy	A4
SCHVÁLIL						
VÝROBA					MĚŘÍTKO: 1:1	LIST 2 Z 2 LISTŮ

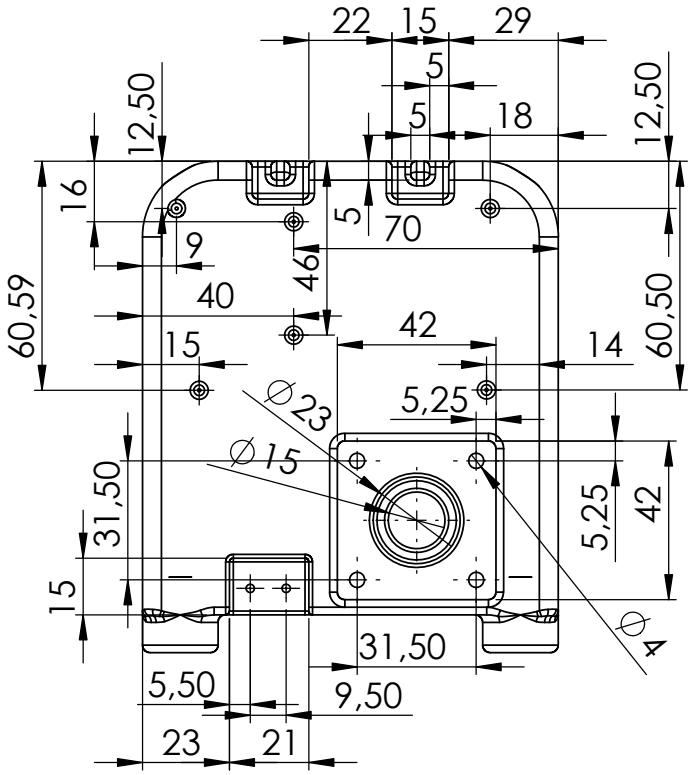


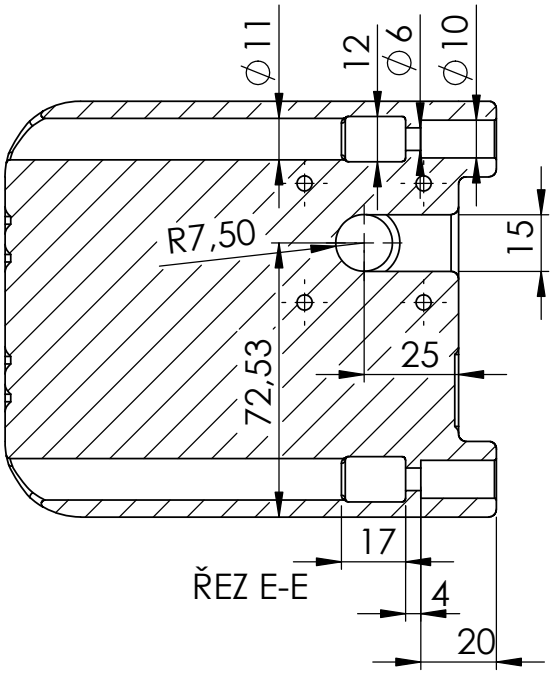


ŘEZ F-F



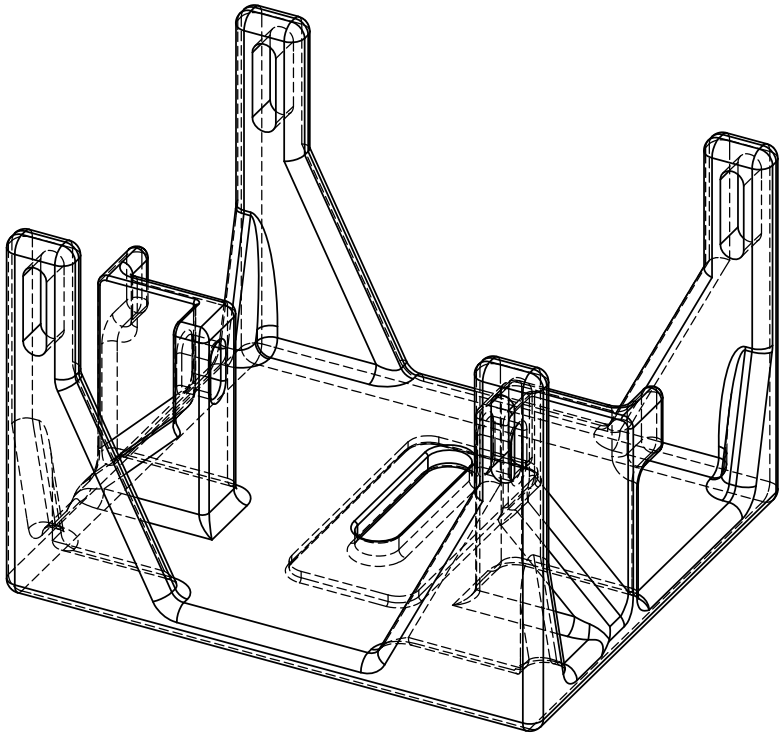


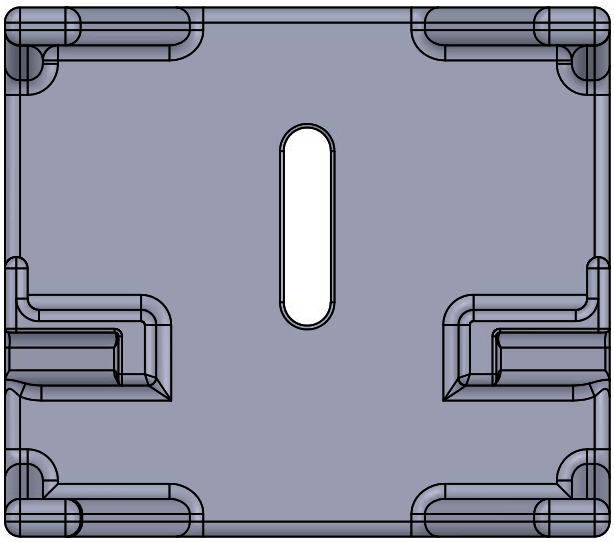


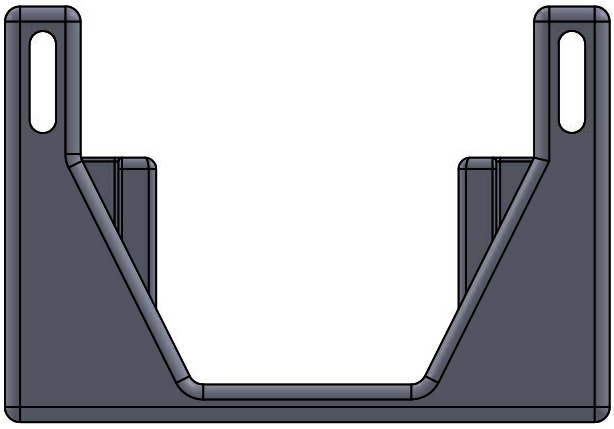


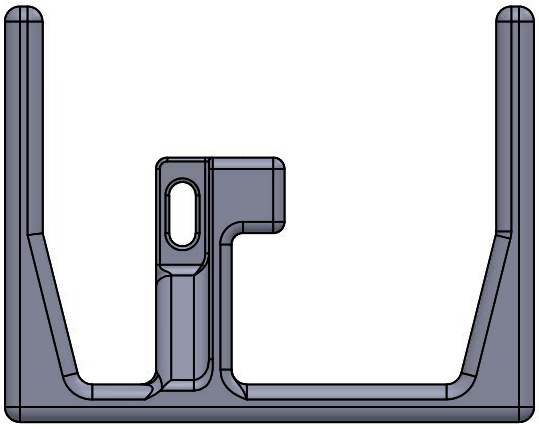
ŘEZ E-E

	NÁZEV	PODPIS	DATUM	MATERIÁL:	NÁZEV:	
NAVRHL	Bc. Tomáš Trenz		15.5.2018	ABS	Začátek dráhy	A4
SCHVÁLIL						
VÝROBA					MĚŘÍTKO: 1:2	LIST 1 Z 1 LISTŮ

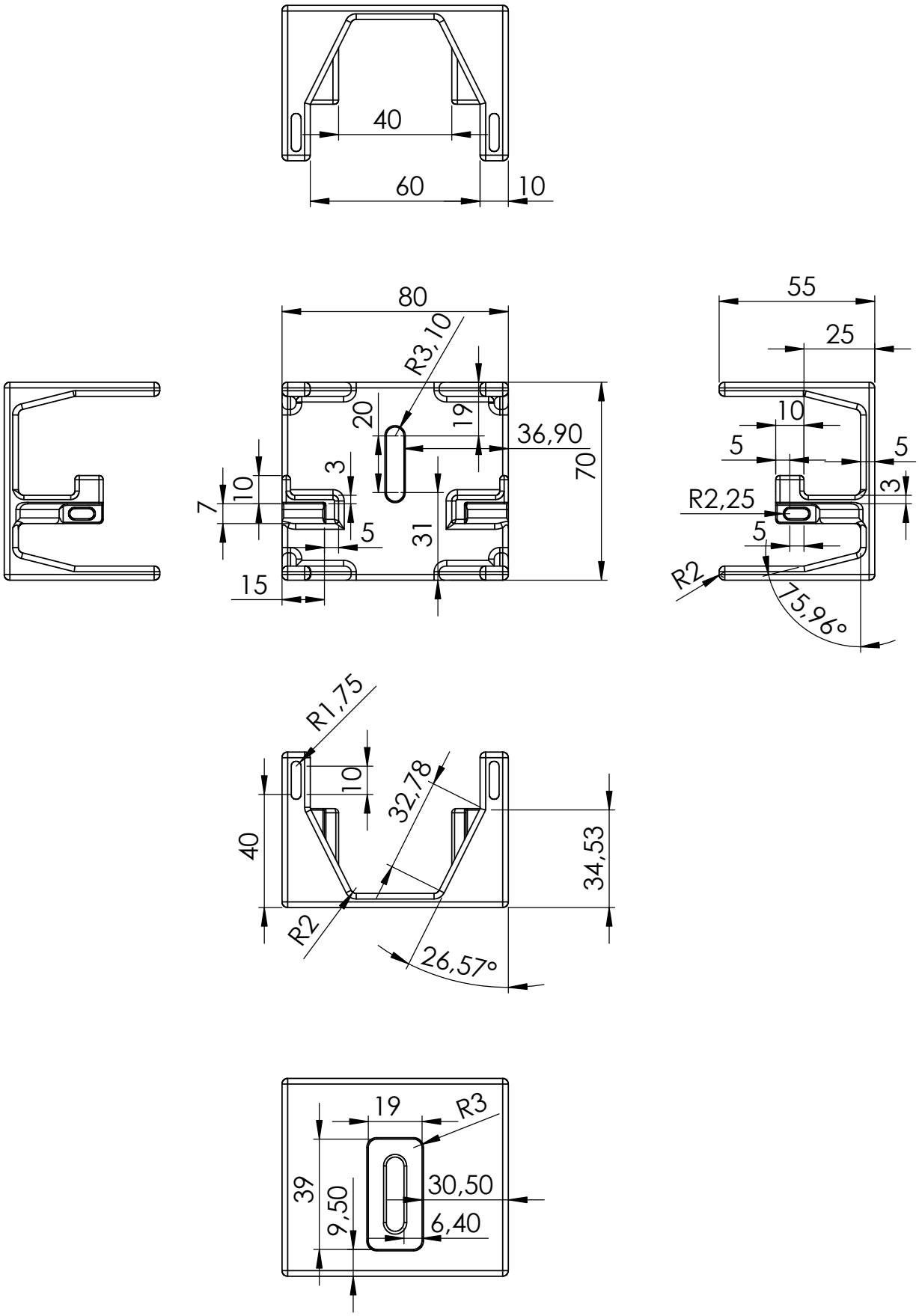




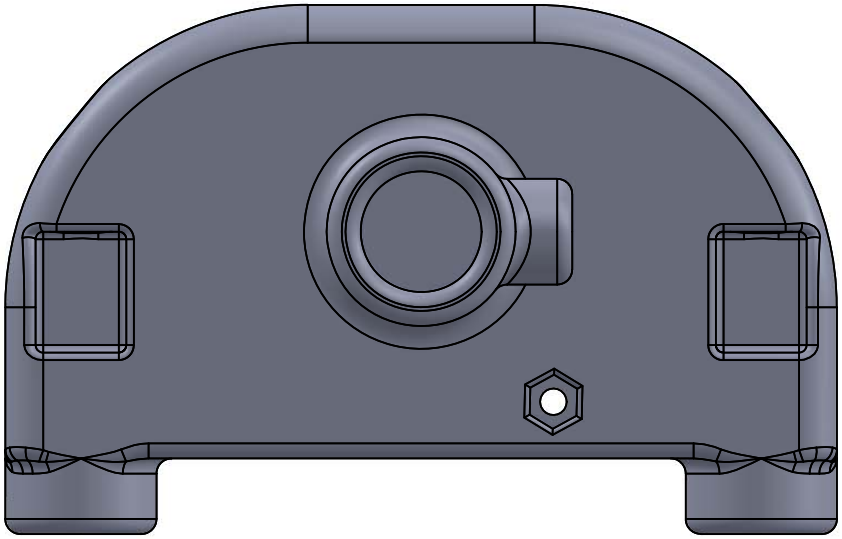


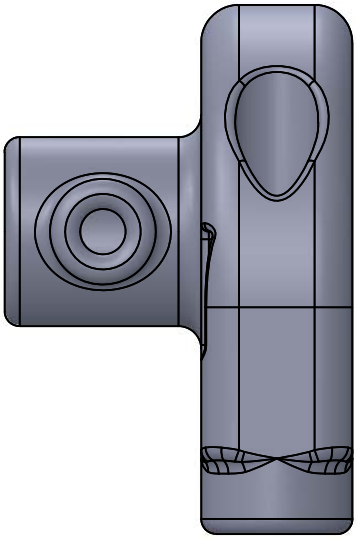


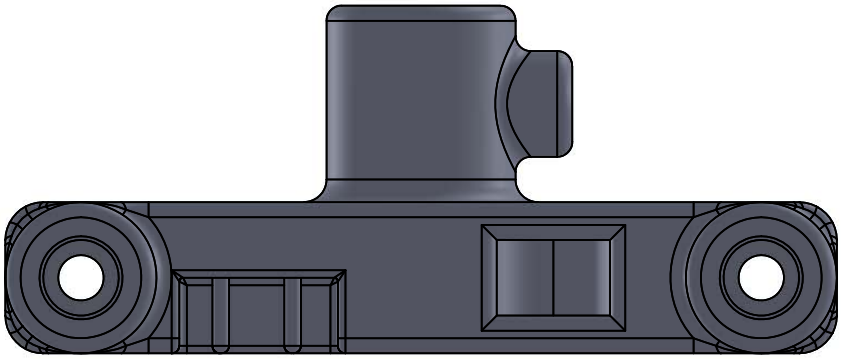
	NÁZEV	PODPIS	DATUM	MATERIÁL:	NÁZEV:	
NAVRHL	Bc. Tomáš Trenz		25.4.2018	ABS	Model vozíku pojezdu	A4
SCHVÁLIL						
VÝROBA					MĚŘÍTKO: 1:1	LIST 1 Z 1 LISTŮ

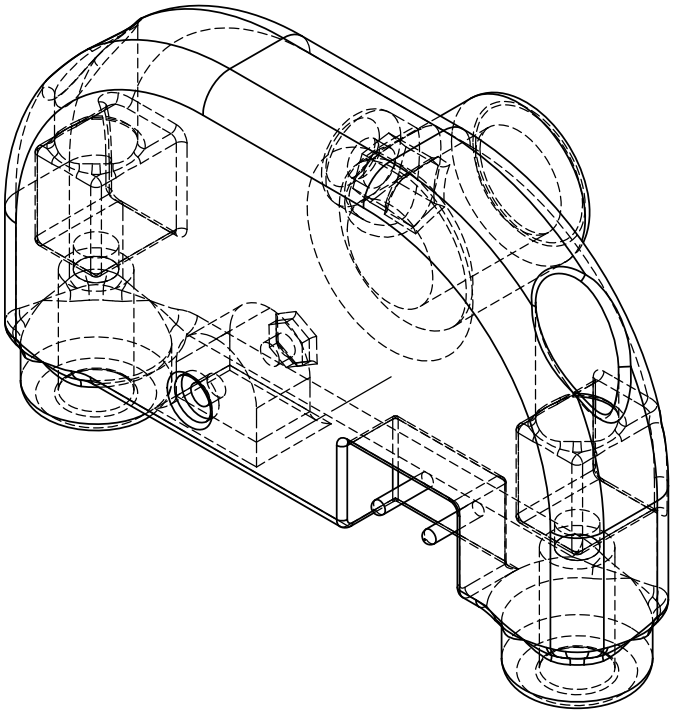


	NÁZEV	PODPIS	DATUM	MATERIÁL:	NÁZEV:	
NAVRHL	Bc. Tomáš Trenz		15.5.2018	ABS	Vozík posunu	A4
SCHVÁLIL						
VÝROBA					MĚŘÍTKO: 1:2	LIST 1 Z 1 LISTŮ

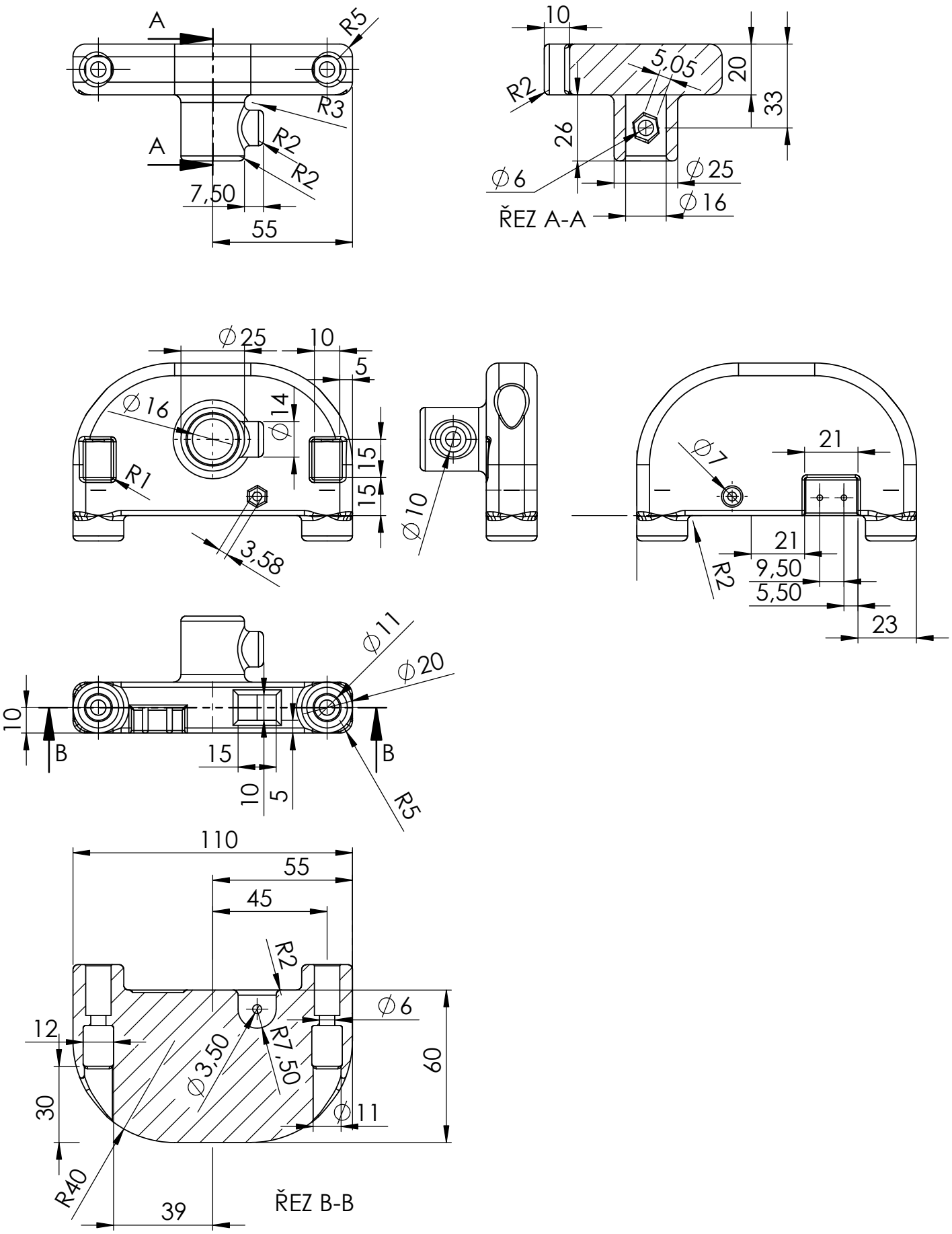








	NÁZEV	PODPIS	DATUM	MATERIÁL:	NÁZEV:	
NAVRHL	Bc. Tomáš Trenz		25.4.2018	ABS	Model zakončení dráhy	A4
SCHVÁLIL						
VÝROBA					MĚŘÍTKO: 1:1	LIST 1 Z 1 LISTŮ



	NÁZEV	PODPIS	DATUM	MATERIÁL:	NÁZEV:	
NAVRHL	Bc. Tomáš Trenz		15.5.2018	ABS	Zakončení dráhy	A4
SCHVÁLIL						
VÝROBA					MĚŘÍTKO: 1:2	LIST 1 Z 1 LISTŮ

B OBSAH PŘILOŽENÉHO CD

Na přiloženém disku se nachází PDF verze semestrální práce.

Dále je zde adresářové struktury se zdrojovými kódy a projekty vytvořenými v programu Atmel Studio verze 7.0. Program pro vývojovou desku Arduino Due lze kompilovat v Atmel Studiu i v Arduino IDE. ale vyžaduje instalaci knihoven umístěných v příloze. Program pro Wemos D1 lze v Atmel Studiu pouze prohlížet. Pro kompilaci je nutné využít Arduino IDE a je také nutné mít nainstalované knihovny.

Ve složce Library jsou uloženy knihovny, které byly použity při kompilaci.

Složka Schema obsahuje projekt a dokumenty programu Eagle 5.11.0.

Modely dílů pro 3D tisk jsou umístěny v složce pojmenované Modely.

```
/ ..................................... kořenový adresář přiloženého CD
├── due ..................... složka projektu Atmel Studia 7.0 pro Arduino Due
│   ├── due ..................................... složka zdrojových souborů
│   │   ├── _vm ..................................... složka souborů Visual Micro
│   │   ├── Debug ..................................... složka souborů kompilátoru
│   │   ├── Device_Startup ..................... složka knihovny rozdělení paměti
│   │   ├── buttons_driver.cpp ................... zdrojový soubor ovladače tlačítek
│   │   ├── buttons_driver.h ................... hlavičkový soubor ovladače tlačítek
│   │   ├── controller.cpp ........................... zdrojový soubor kontroléru
│   │   ├── controller.h ........................... hlavičkový soubor kontroléru
│   │   ├── defines.h ............................... hlavičkový soubor konstant
│   │   ├── due.cppproj ..................................... projekt Atmel Studia
│   │   ├── due.ino ..................................... projekt Arduino studia
│   │   ├── I2C_drive.cpp ........................... zdrojový soubor I2C rozhraní
│   │   ├── I2C_drive.h ........................... hlavičkový soubor I2C rozhraní
│   │   ├── oled_driver.cpp ..................... zdrojový soubor ovladače displeje
│   │   ├── oled_driver.h ..................... hlavičkový soubor ovladače displeje
│   │   ├── opto_switch.cpp ..................... zdrojový soubor ovladače spínače
│   │   ├── opto_switch.h ..................... hlavičkový soubor ovladače spínače
│   │   ├── sensors_driver.cpp ................... zdrojový soubor ovladače senzorů
│   │   ├── sensors_driver.h ................... hlavičkový soubor ovladače senzorů
│   │   ├── stepper_driver.cpp ......... zdrojový soubor ovladače krokového motoru
│   │   ├── stepper_driver.h ......... hlavičkový soubor ovladače krokového motoru
│   │   └── systick_hook.cpp ............... zdrojový soubor události 1ms časovače
│   └── due.atsln ..................................... projekt Atmel Studia 7.0
└── D1 ............................ složka projektu Atmel Studia 7.0 pro Wemos D1
    ├── _vm ..................................... složka souborů Visual Micro
    ├── Debug ..................................... složka souborů kompilátoru
    ├── Controller.cpp ............................... zdrojový soubor kontroléru
    ├── Controller.h ............................... hlavičkový soubor kontroléru
    ├── D1.cppproj ..................................... projekt Atmel Studia
    ├── D1.atsln ..................................... projekt Atmel Studia 7.0
    ├── D1.ino ..................................... projekt Arduino studia
    └── Defines.h ................................... hlavičkový soubor konstant
```

- I2C_driver.cpp ..... zdrojový soubor I2C rozhraní
- I2C_driver.h ..... hlavičkový soubor I2C rozhraní
- WEB_driver.cpp ..... zdrojový soubor webového rozhraní
- WEB_driver.h ..... hlavičkový soubor webového rozhraní
- Wifi_driver.cpp ..... zdrojový soubor Wifi rozhraní
- Wifi_driver.h ..... hlavičkový soubor Wifi rozhraní

- Knihovny ..... složka knihoven pro kompilaci
  - Adafruit_SSD1306.zip ..... knihovna OLED displeje
  - Adafruit-GFX-Library.zip ..... grafická část OLED knihovny
  - ArduinoWebSockets.zip ..... Knihovna pro WebSocket
  - StepperDriver-master.zip ..... knihovna ovladače krokového motoru
- Modely ..... složka s 3D modelů pro SolidWorks 2014
  - Ukonceni.SLDPRT ..... soubor s 3D modelem ukončení dráhy
  - Vozik.SLDPRT ..... soubor s 3D modelem vozíku pro fotoaparát
  - Zacatek.SLDPRT ..... soubor s 3D modelem hlavní části dráhy
- Schema ..... složka s projektem Eagle 5.11.0
  - eagle ..... projekt Eagle 5.11.0
  - schema.brd ..... rozvržení a spoje prototypové desky
  - schema.sch ..... schéma prototypové desky
- diplomova_prace.pdf ..... Diplomová práce