

UNIVERZITA PARDUBICE

Fakulta elektrotechniky a informatiky

Dávkovač pájecí pasty

Richard Schellrich

Bakalářská práce

2024

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: 2022/2023

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Richard Schellrich**
Osobní číslo: **I20060**
Studijní program: **B0714P060001 Aplikovaná elektrotechnika**
Téma práce: **Dávkovač pasty**
Zadávající katedra: **Katedra elektrotechniky**

Zásady pro vypracování

Cílem práce je návrh elektroniky pro ruční dávkovač pasty. Dávkovač pasty tlačí závitovou tyčí kartuši pasty (v podstatě injekční stříkačka) a vytlačí pastu ven. Teoretická část práce popíše teorii kolem páječích past (druhy, jejich vlastnosti atp.) dále rozbor možností pohonu pístu (závitové tyče), problematiku dávkování viskózních kapalin/past. Praktická část bude obsahovat návrh, konstrukci a ověření elektroniky dávkovače pasty s uživatelským rozhraním a režimy kontinuálního tlačení s retrakcí po ukončení dávky, automatickými dávkami a možností uživatelsky přepínat, případně nastavovat jednotlivé režimy.

Rozsah pracovní zprávy: **30-50**
Rozsah grafických prací:
Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam doporučené literatury:

- [1] VÁŇA, V. Mikrokontroléry ATMEL AVR: popis procesoru a instrukční soubor. Praha: BEN technická literatura, 2003. 336 s. ISBN 978-80-7300-083-0.
- [2] VÁŇA, V. Mikrokontroléry ATMEL AVR: programování v jazyce C. Praha: BEN technická literatura, 2003. 216 s. ISBN 978-80-7300-102-0.
- [3] VLACH, J. Řízení a vizualizace technologických procesů. Praha: BEN technická literatura, 2002. 160 s. ISBN 978-80-86056-66-X.
- [4] BRTNÍK, B. Základní elektronické obvody. Praha: BEN technická literatura, 2011. 156s. ISBN 978-80-7300-408-8
- [5] RIPKA, P.; TIPEK, A. Master Book of Sensors. Praha : BEN, 2003. ISBN 0-12-752184

Vedoucí bakalářské práce: **Ing. Pavel Rozsival**
Katedra elektrotechniky

Datum zadání bakalářské práce: **15. listopadu 2022**
Termín odevzdání bakalářské práce: **12. května 2023**

Ing. Zdeněk Němec, Ph.D. v.r.
děkan

L.S.

Ing. Jan Pidanič, Ph.D. v.r.
vedoucí katedry

V Pardubicích dne 31. ledna 2023

Prohlášení autora

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Beru na vědomí, že v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, a směrnicí Univerzity Pardubice č. 9/2012, bude práce zveřejněna v Univerzitní knihovně a prostřednictvím Digitální knihovny Univerzity Pardubice.

V Pardubicích dne 10. 05. 2024

Richard Schellrich

Poděkování

Rád bych poděkoval své rodině a přátelům, kteří mě vždy podporovali ve studiu. Dále bych chtěl poděkoval svému vedoucímu práce Ing. Pavlovi Rozsivalovi za konzultace k práci.

Anotace

Cílem práce je návrh elektroniky pro ruční dávkovač pasty. Dávkovač pasty tlačí závitovou tyčí kartuši pasty (v podstatě injekční stříkačka) a vytlačí pastu ven. Teoretická část práce popíše teorii kolem pájecích past (druhy, jejich vlastnosti atp.) dále rozbor možností pohonu pístu (závitové tyče), problematiku dávkování viskózních kapalin/past. Praktická část bude obsahovat návrh, konstrukci a ověření elektroniky dávkovače pasty s uživatelským rozhraním a režimy kontinuálního tlačení s retrakcí po ukončení dávky, automatickými dávkami a možností uživatelsky přepínat, případně nastavovat jednotlivé režimy.

Klíčová slova

Pájecí pasta, dávkovač, elektronika, Arduino, řízení, pájení, desky plošných spojů

Title

solder paste dispenser

Annotation

The goal of the thesis is the design of electronics for a manual paste dispenser. The paste dispenser pushes the paste cartridge (essentially a syringe) with a threaded rod and pushes the paste out. The theoretical part of the work will describe the theory of soldering pastes (types, their properties, etc.), further analysis of piston drive options (threaded rods), the issue of dispensing viscous liquids/pastes. The practical part will include the design, construction and verification of the electronics of the paste dispenser with a user interface and modes of continuous pushing with retraction after the end of the dose, automatic doses and the possibility of user switching or setting individual modes.

Keywords

solder paste, dispenser, electronics, Arduino, control, soldering, printed circuit boards

Obsah

Seznam zkratk	9
Seznam obrázků	10
Úvod	11
1 Teoretický základ	12
1.1 Vlastnosti a složení pájecí pasty	12
1.1.1 Olovnaté pájky	12
1.1.2 Bezolovnaté pájky	12
1.2 Tavidla	13
1.2.1 Složení tavidel	13
1.3 Metody aplikace pájecí pasty	14
1.3.1 Sítotisk	14
1.3.3 Dávkovače	15
1.4 Proces pájení.....	15
1.4.1 Pájení vlnou	15
1.4.2 Pájení vlečením	16
1.4.3 Pájení ponorem.....	16
1.5 Proces přetavením.....	16
1.5.1 Ohřevem infračervenými zářiči	16
1.5.2 Ohřevem horkým plynem.....	17
1.5.4 Ohřevem pomocí laseru.....	18
2 Přehled existujících řešení	18
2.1 Pneumatické dávkovače	18
2.2 Hydraulické dávkovače	19
2.3 Elektrické dávkovače.....	19
3 Parametry ovlivňující dávkování	19
4 Komponenty	19
4.1 Řízení periferií.....	19
4.1.1 Arduino Uno	20
4.2 Periferie	20
4.2.1 Elektromotory	21
5 Výroba a realizace prototypu	23

5.1 Návrh a konstrukce elektroniky.....	24
5.2 Fyzická konstrukce	26
5.3 Programové vybavení	27
5.4 Testování výrobku	28
Závěr	29
Literatura	30
Příloha A – Schéma obvodu.....	33
Příloha B – Kód programu	34

Seznam zkratek

SMD	Surface Mount Device (Součástka pro povrchovou montáž)
SMT	Surface Mount Technology (Technologie povrchové montáže)
THT	Through Hole Technology (Technologie Prokovových Děr)
USB	Universal Serial Bus (Univerzální sériová sběrnice)
DPS	Deska Plošných Spojů
IR	InfraRed (infračervené)
BLDC	Brushless DC electric motor (Bezkartáčový DC elektromotor)
BLAC	Brushless AC electric motor (Bezkartáčový AC elektromotor)
DC	Direct Current (Stejnsměrný proud)
AC	Alternating Current (Střídavý proud)
A/D	Analogové/Digitální
BGA	Ball grid array (Kuličky uspořádané do mřížky)
IDE	Integrated Development Environment (Vývojové prostředí)
I2C	Inter-Integrated Circuit (multi masterová sériová sběrnice)
LCD	Liquid Crystal Display (Displej z tekutých krystalů)
GND	Ground (uzemění)
PLA	Polylactic acid (kyselina polymléčná)

Seznam obrázků

Obrázek 1 - Woodův kov[15].....	12
Obrázek 2 - Kalafuna[2].....	13
Obrázek 3 - SMT planžeta[4].....	14
Obrázek 4 - Ruční dávkovač pájecí pasty[16].....	15
Obrázek 5 - Vlna z procesu pájení vlnou[5].....	16
Obrázek 6 - Infračervená pájecí stanice	17
Obrázek 7 - Pec pro přetavení horkým plynem[6]	17
Obrázek 8 - Laserová pájecí stanice[17]	18
Obrázek 9 - Pneumatický dávkovač pájecí pasty[8]	18
Obrázek 10 - Vývojová deska Arduino Uno[9].....	20
Obrázek 11 - Rozdělení elektromotorů[18].....	21
Obrázek 12 - Rotor krokového motoru[13].....	22
Obrázek 13 - Krokový motor s budičem[20].....	24
Obrázek 14 - Schéma modulu budiče.....	24
Obrázek 15 - Schéma modulu displeje s I2C převodníkem	25
Obrázek 16 - Modul LCD displeje[19]	25
Obrázek 17 – Základní stanice	26
Obrázek 18 – Ruční část dávkovače.....	26
Obrázek 19 – Arduino IDE.....	27

Úvod

V dnešním světě automatizace se klade důraz na masovou výrobu elektroniky a nedílnou součástí je i způsob pájení. Ve velkém měřítku se využívají SMD komponenty, u kterých se používá pájecí pasta na připevnění k DPS.

Tato bakalářská práce se zaměřuje na problematiku dávkování pájecí pasty a návrh elektroniky pro ruční dávkovač pájecí pasty. Cílem práce je vytvořit zařízení, které umožní přesné dávkování pájecí pasty s důrazem na cenu a jednoduchost konstrukce, využitím krokového motoru se závitovou tyčí pro pohon kartuše pasty. Výhodou tohoto systému je jednoduchost oproti např. pneumatickému pohonu.

Práce se skládá z teoretické části, která se věnuje vlastnostem a složením pájecích past i tavidel, problematice nanášení na desku plošných spojů a procesu přetavení. Praktická část se skládá z návrhu, konstrukce a ověření elektroniky dávkovače pasty.

1 Teoretický základ

1.1 Vlastnosti a složení pájecí pasty

Pájecí pasta se skládá z miniaturních kuliček pájky a tavidla. Tyto kuličky jsou většinou složeny z dvousložkových či tříložkových slitin, které určují jejich teplotu tání, mechanickou odolnost, nebo smáčivost. Tyto pájky můžeme rozdělit na olovnaté a bezolvnaté.

1.1.1 Olovnaté pájky

Olovnaté pájky jsou slitiny olova (Pb) a cínu (Sn). Olovo je měkký a tvárný materiál, a to se pozitivně projevuje v kombinaci s cínem. Tato slitina je po zatuhnutí pružná a dostatečně pevná. Provedení těchto pájek je v několika variantách podle poměru olova a cínu, například Sn63Pb37, kde tato pájka obsahuje 63 % cínu a 37 % olova či Sn60Pb40, která má obsah z 60 % cínu a 40% olova. Vlastnosti pájky se drobně liší podle poměrů prvků v nich obsažených. Teplota tání se u těchto slitin pohybuje okolo 180°C.

Jeden z nevšedních druhů těchto pájek je takzvaný Woodův kov. Tato pájka je složena z 12,5 % cínu, 25 % olova, 50 % bismutu a 12,5 % kadmia. Je velmi specifická pro svou nízkou teplotu tání, která se pohybuje okolo 70°C. Nevýhodou tohoto kovu je obsah toxických látek.
[1]



Obrázek 1 - Woodův kov[15]

1.1.2 Bezolvnaté pájky

Bezolvnaté pájky jsou slitiny především z cínu a jiných kovů. V dnešní době jsou výhradně používaným typem pájek, jelikož olovnaté pájky byly v EU zakázány pro komerční využití v roce 2006. V roce 2018 byl k tomuto zákazu vypsán dodatek Evropským parlamentem, který zakazuje plošný prodej pájek, které obsahují více než 0,1 % olova fyzickým osobám. Bezolvnaté pájky mají podobné vlastnosti jako ty olovnaté, ačkoliv v jistých odvětvích je jejich používání nevhodné, kvůli nižší životnosti a spolehlivosti. Mezi tyto odvětví se řadí zdravotnictví, kosmonautika nebo armáda.

Dále tyto pájky můžeme opět rozdělit na několik druhů. Ku příkladu pájka typu Sn99,3Cu0,7. Teplota tání této pájky se pohybuje okolo 227°C. Z důvodu vyšší teploty tání je pájka náchylnější k oxidaci a také je křehčí díky obsahu mědi oproti olovným pájkám. Ani smáčivost této pájky není tak dobrá jako u olovnatých.

Mezi další bezolovnaté pájky můžeme zařadit pájku Sn96,5Ag3Cu0,5 s přidaným obsah stříbra, které snižuje teplotu tání na 217°C. Stříbro taktéž zlepšuje smáčivost této pájky a v neposlední řadě zlepšuje pevnost spoje.[1]

1.2 Tavidla

Tavidla se mohou vyskytovat v různých formách, v případě pájecích past je tavidlo smícháno přímo s kovovými zrny. Vlastnosti tavidla a jeho množství, které je přidáno k pájecím zrnům, udávají vlastnosti vzniklé pájecí pasty. Používají se k odstranění nečistot oxidů z pájecích ploch. Zároveň podporují roztékavost pájky a smáčivost pájeného povrchu. V neposlední řadě tavidla slouží i ke zkvalitnění přenosů tepla od zdroje do pájeného spoje.



Obrázek 2 - Kalafuna[2]

1.2.1 Složení tavidel

Tavidla můžeme rozdělit také do třech základních skupin podle složení.

- Pryskyřičná
- Organická
- Anorganická

Základem pryskyřičných tavidel je pryskyřice, která může být buď ve formě kalafuny, nebo se může jednat o pryskyřici umělou. Pryskyřice bývá většinou rozpuštěna v etylalkoholu nebo xylynu. Pokud bychom uvažovali pouze tuto látku, bylo by tavidlo velmi málo aktivní. Proto se přidávají k této látce ještě kyseliny nebo amin hydrochloridy, které určují míru aktivity těchto tavidel. Typickým příkladem může být kyselina salicylová nebo jantarová.

Organická tavidla byla vyvinuta jako důsledek omezení některých chemických látek. Hlavním kritériem bylo, aby byly zbytky tavidla po procesu pájení jednoduše odstraněny. To znamená, aby se tyto látky daly odstranit vodou. Tato tavidla jsou velmi účinná, ovšem vlastnosti, které mají jejich použití omezuje. Zbytky pájky zůstávají po pájení tekuté a mohou vyvolat korozi.

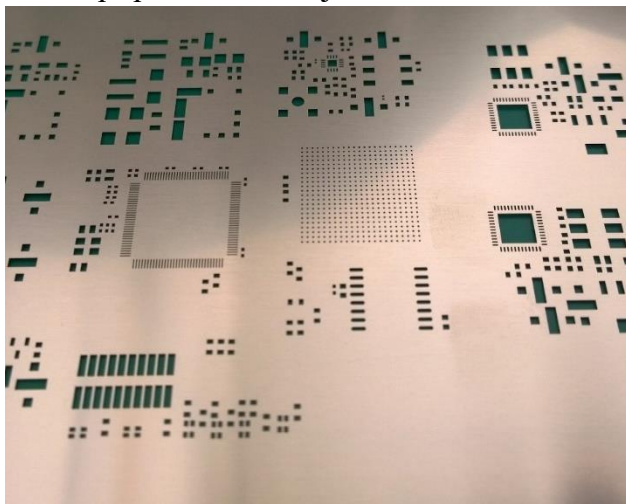
Anorganická tavidla se používají pouze ve speciálních případech. Jejich základem jsou například soli nebo kyseliny, které obsahují, stejně jako pryskyřičná tavidla, aktivátor. Tím je například čpavek, kyselina fosforečná nebo chlorid amonný. Právě z důvodu použití těchto agresivních rozpouštědel je práce s těmito látkami obtížnější a povrch po pájení musí být od těchto tavidel důkladně očištěn. Neочиštěné tavidlo by mohlo mít špatný vliv na okolní pájené spoje i na celý výrobek.[3]

1.3 Metody aplikace pájecí pasty

Metody nanášení můžeme rozdělit na dva druhy, podle typu fungování celého systému. Pájecí pastu můžeme aplikovat pomocí dávkovače nebo tzv. sítotiskem.

1.3.1 Sítotisk

Sítotisk je forma nanášení pasty skrze planžetu. Planžeta je tenký kovový plát, ve kterém jsou vytvořeny přesné díry podle pouzder komponentů. Tato planžeta se přikládá k DPS aby na ní lícovala s pájecími ploškami. Poté se po planžetě přemaže pájecí pasta, která zůstane na pájecích ploškách. Takto připravená deska je vhodná k osazování.



Obrázek 3 - SMT planžeta[4]

1.3.3 Dávkovače

Dávkovače jsou v podstatě injekční stříkačky, které nanášejí pastu na pájecí plošky postupně. Na trhu jsou k dostání jak ruční dávkovače, tak i automatické. Ruční dávkovače jsou přesné pouze podle toho, jak přesný je uživatel, jelikož nemají žádnou elektroniku na hlídání průtoku či celkovém vytlačeném množství pasty. Jejich výhodou je především cena. Dalšími dávkovači jsou dávkovače automatické, které jsou přesnější ve výtlaku pasty díky elektronice díky čemuž jsou dražší. Jejich použití je nevhodné pro velkovýrobu kvůli rychlosti nanášení. Dále si rozebereme dávkovače v další kapitole.



Obrázek 4 - Ruční dávkovač pájecí pasty[16]

1.4 Proces pájení

Proces pájení můžeme rozdělit na tři základní druhy a ty jsou rozepsané v následujících kapitolách

1.4.1 Pájení vlnou

Pájení vlnou je velmi rozšířenou metodou. Pomocí vlny lze pájet nejen součástky THT, ale i SMD. Pro povrchovou montáž je využitelné z metod pájení tekutou pájkou právě pájení vlnou.

Před samotným procesem pájení vlnou se musí v případě povrchové montáže součástky k desce nejprve pomocí lepidla přilepit, aby je vlna nesmyla. Důležité je, aby lepidlo zajistilo dostatečnou pevnost spoje, a také aby lepidlo nebylo elektricky vodivé.

Při samotném pájení vlnou je roztavená pájecí slitina ve vyhřívané vaně poháněna čerpadlem, které vytváří vlnu. Deska plošného spoje následně prochází nad touto vlnou s malým sklonem (asi 4-8°) a dochází jejímu smáčení roztavenou pájkou. Tím jsou kontaktní plošky se součástkami zapájeny.



Obrázek 5 - Vlna z procesu pájení vlnou[5]

1.4.2 Pájení vlečením

Deska je vlečena po klidné hladině s roztavenou pájecí slitinou ve vyhřívané vaně. Pájení vlečením se tedy liší od pájení vlnou především v tom, že pájecí slitina je poháněna čerpadlem jen minimálně. Liší se také ve sklonu, kde se u pájení vlnou pohyboval od 4° do 8°, zatímco dopravník DPS je u pájení vlečením rovnoběžně s nádobou. Vzhledem k porovnání s metodou pájení vlnou se pájení vlečením příliš nevyužívá.

1.4.3 Pájení ponorem

Stejně jako pájení vlečením nenachází pájení ponorem větší využití. Deska je ponořena do roztavené pájecí slitiny a následně vyjmuta. Využití je pouze v malých sériích nebo při výrobě prototypů. Tato metoda je naprosto nedostačující pro součástky velmi náchylné na teplo, či jejich pouzdra.

1.5 Proces přetavením

Pájení přetavením je výhodné pro součástky s malou roztečí vývodů (typicky integrované obvody), které by klasickým pájením buďto nemohly být zapájeny (BGA čipy), nebo by pro ně takovéto přetavení nedávalo smysl v rámci velkovýroby (zvýšená tvorba můstků mezi blízkými spoji). Existuje několik možností pájení přetavením, v následujících kapitolách jsou rozepsané.

1.5.1 Ohřevem infračervenými zářiči

Plošný spoj je nejprve předehřátý a následně je dopravníkem přesunut pod zářiče. Nevýhodou pájení IR zářiči je nerovnoměrné prohřívání desky, protože část záření není pohlceno, ale odraženo. Záleží na mnoha aspektech povrchu jako jsou materiál, barva,

drsnost nebo úhel dopadajícího záření. Z tohoto důvodu dochází k nerovnoměrnému prohřívání, což vede k rozdílné kvalitě pájených spojů.



Obrázek 6 - Infračervená pájecí stanice

1.5.2 Ohřevem horkým plynem

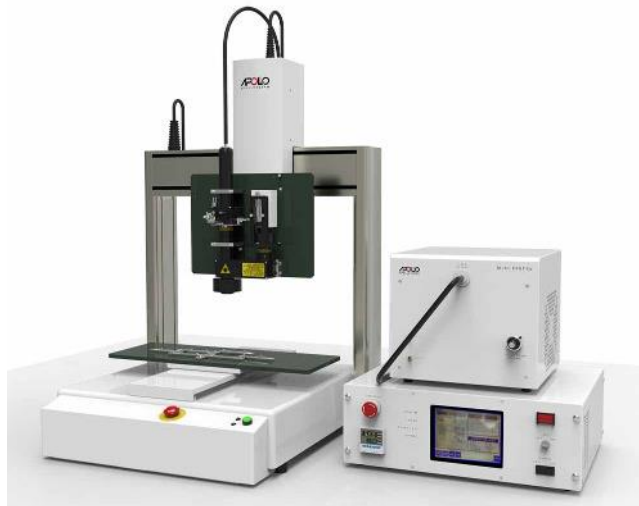
Pro ohřev horkým plynem se nejčastěji užívá vzduch, ale může být použit například i dusík, který zároveň zabraňuje oxidaci na pájených spojkách. Ohřátý plyn je hnán ventilátory na dopravník desek plošných spojů. Výhodou této metody je oproti ohřevu IR zářiči lepší prohřátí desky z důvodu menší závislosti teploty na aspektech povrchu (např. materiál, barva, drsnost). Nevýhodou tohoto procesu je delší čas, za který nastane přetavení.



Obrázek 7 - Pec pro přetavení horkým plynem[6]

1.5.4 Ohřevem pomocí laseru

Laserový svazek je směřován na jednotlivé spoje, čímž se ohřívá pájecí pata na teplotu, kdy dojde k jejímu přetavení. Tato metoda je využívána především k pájení součástek, které není možné připájet vlnou nebo přetavením kvůli náchylnosti součástky na teplotu. Pájení laserem je tedy spíše metodou pro osazení jednotek součástek, pro osazení celé desky plošného spoje se nevyužívá z důvodů časových, ale i ekonomických.[7]



Obrázek 8 - Laserová pájecí stanice[17]

2 Přehled existujících řešení

2.1 Pneumatické dávkovače

Pneumatické dávkovače jsou dávkovače, které posouvají píst kartuše skrze stlačený vzduch, jenž je dopravován trubičkou ze základny. Výhodou tohoto systému je jednoduchost konstrukce kartuše, která na sobě má pouze pneumatický píst a v podstatě nekonečně velký rozsah, na kterém může píst jezdit. Nevýhodou se však stává základní vlastnost plynů a tou je stlačitelnost. Díky tomu nelze vytvořit precizní systém založený na pneumatickém pohonu.



Obrázek 9 - Pneumatický dávkovač pájecí pasty[8]

2.2 Hydraulické dávkovače

Hydraulické jsou dosti podobné pneumatickým. V zásadě je místo plynu použita kapalina. Nevýhodou oproti pneumatickému systému je dimenzování vedení pro kapalinu, která by mohla vytéct. Výhodou však je, že kapaliny jsou nestlačitelné a dají se díky tomu udělat velmi precizní hydraulické systémy.

2.3 Elektrické dávkovače

Elektrickým dávkovačem je myšlen dávkovač, který elektromotorem přímo pohybuje pístem v kartuši bez složitého řízení stlačeného vzduchu nebo hydraulické kapaliny. Přesnost tohoto řešení závisí na typu použitého motoru a použitém řízení tohoto motoru. Výhodou tohoto systému je pak z mého pohledu jednoduchost a proveditelnost. Nevýhodou se stává váha kartuše, jelikož obsahuje i motor s převodovým mechanismem.

3 Parametry ovlivňující dávkování

Dávkování viskózních kapalin, jako je pájecí pasta, představuje výzvu v oblasti přesnosti a konzistence. Vysoká viskozita pasty může komplikovat pohyb pístu a vést k nekonzistentním výsledkům dávkování. Díky různým viskozitám pájecích past, je třeba zvolit vhodný mechanismus pro přesné dávkování. Regulace tlaku a průtoku pájecí pasty je dalším klíčovým faktorem, který ovlivňuje proces dávkování. Uvedené informace naznačují, že důkladná analýza viskozity a vhodná regulace tlaku a průtoku jsou nezbytné pro dosažení optimálních výsledků dávkování pájecí pasty. V reálném prostředí je tato analýza velmi náročná, jelikož konzistenci samotné pájecí pasty ovlivňuje několik faktorů, ať už to jsou teplota, složení, či časová degradace materiálu. V podstatě pro naše použití nemá taková analýza význam, jelikož náš výrobek je určen jen pro neprofesionální použití.

4 Komponenty

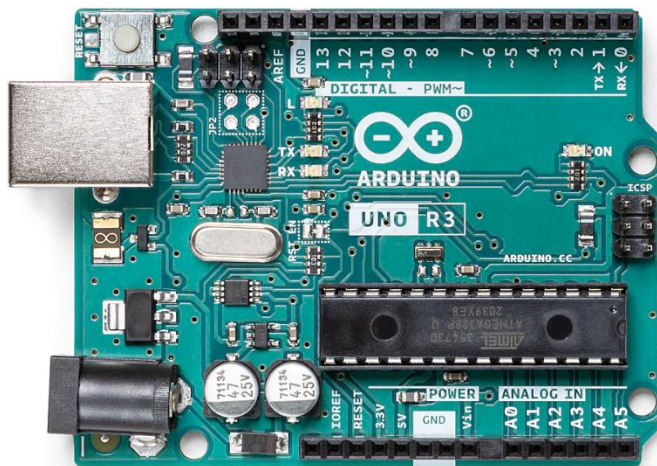
Tato kapitola se zabývá všeobecnými znalostmi o potřebných součástkách k výrobě, jejich výhodami a nevýhodami. Dále výběrem nejvhodnějších součástek a zařízení a také možnostmi řízení celého systému.

4.1 Řízení periferií

Řízení periferií systému můžeme provádět různými způsoby. Ať už to jsou hradlová pole, či přes mikrokontrolér. Mikrokontrolér je integrovaný obvod, který se skládá z procesoru, paměti, A/D převodníků, komparátorů a vstupně/výstupních portů, které lze libovolně programovat. V dnešní době jsou základní mikrokontroléry produkovány v takové míře, že jejich cena se stala zanedbatelnou a díky jejich všestrannosti použití se staly nejpoužívanějším řídicím komponentem. V dnešní době se na trhu objevuje celá řada vývojových desek s těmito mikrokontroléry. Mezi asi nejznámější můžeme zařadit desku Uno od firmy Arduino.

4.1.1 Arduino Uno

Arduino Uno je vývojová deska od firmy Arduino, kterou řídí mikrokontrolér ATmega328 od firmy Atmel. Mezi její hlavní přednosti můžeme vyzdvihnout její příznivou cenu a jednoduchost programování, přítomnost napájecího konektoru, USB port, dostačující počet pinů (14 digitálních vstupně výstupních a 6 analogových vstupních, které lze použít taky jako digitální) a dostačující velikost flash paměti. Pro takovou aplikaci naprosto dostačující.



Obrázek 10 - Vývojová deska Arduino Uno[9]

4.2 Periferie

Periferiemi se rozumí součástky a zařízení připojené k řídicímu obvodu, mezi takové můžeme zařadit všechny druhy senzorů či výkonových prvků. Řídicí obvod načte informace získané ze senzorů, které následně zpracuje podle definované logiky a přešle na požadovaný výstup. Periferie připojené na výstup řídicího obvodu jsou signalizační zařízení, jako jsou stavové LED diody a displeje, či zařízení vykonávající práci jako jsou elektromotory.

4.2.1 Elektromotory

Elektromotory jsou základní točivé stroje. Elektromotory můžeme rozdělit na několik druhů, ať už je budeme dělit podle napájecího napětí, počtu fází, či podle vnitřní konstrukce. Základními konstrukčními prvky motoru je rotor a stator, které se skládají z magnetů a cívek. Rotor je rotující část elektromotoru, zatímco stator je statická část elektromotoru.

Elektrické nářadí			
Komutátorový motor	Komutátorový motor	BLDC	PMSM (BLAC)
Rotor + komutátor (kotva)	Rotor + komutátor (kotva)	Rotor: perm. magnet	Rotor: perm. magnet
Stator: perm. magnet	Stator: plechy s vinutím	Stator: plechy s vinutím	Stator: plechy s vinutím

Obrázek 11 - Rozdělení elektromotorů[18]

Střídavé motory

Střídavé motory můžeme ještě rozdělit na synchronní a asynchronní. Synchronní motor se vyznačuje tím, že se otáčí současně s rotací magnetického pole statoru, takto konstruované motory mohou pracovat i jako generátory. Asynchronní motor se pro změnu vyznačuje tím, že se rotor zpožďuje za rotací magnetického pole statoru. Takové motory jsou levné, mají jednoduchou konstrukci a dají se dobře regulovat.[10]

Stejnoseměrné motory

Stejnoseměrné motory jsou motory pracující se stejnosměrným proudem a napětím. Tyto motory můžeme dále rozdělit na kartáčové a bezkartáčové. Kartáčové motory mají uhlíky přiléhající na komutátor. Jsou konstrukčně velmi jednoduché a nepotřebují k provozu žádný elektrický obvod. Bezkartáčové motory mají rotor tvořen permanentními magnety a stator cívkami, ke kterým je připojen střídač. Střídač je speciální obvod, který převádí stejnosměrné napětí resp. stejnosměrný proud na střídavé napětí resp. střídavý proud.[11][12]

Krokové motory

Krokové motory jsou speciálním druhem stejnosměrného motoru. Tyto motory se netočí, ale pootáčí po jednotlivých krocích (v určitých úhlech). Délka jednotlivých kroků je určena výrobcem daného motoru. Ke každému krokovému motoru je ještě zapotřebí budič, který generuje impulsy v určitém sledu a délce. Krokové motory se dají dále rozdělit podle typu buzení rotorů.



Obrázek 12 - Rotor krokového motoru[13]

- **Pasivní točivé krokové motory**

Jsou také nazývány jako drápkové motory. Rotor těchto motorů je tvořen vyniklými póly z magneticky měkkého materiálu, kdy jednotlivé póly rotoru jsou vtahovány pod póly statoru, do míst s nejsilnějším magnetickým polem, a to díky rozdílu v magnetické vodivosti vzduchu vůči „drápkům“ vyniklých pólů rotoru. Po vtažení do magnetického pole statoru jsou oba zúčastněné drápky rotoru v této pozici pak i nadále drženy, jako pod elektromagnetem, dokud řídicí obvod nenabudí jiný pól statoru, ideálně s nejmenším úhlem k překroku.

- **Aktivní točivé krokové motory**

Jsou také nazývány jako motory s aktivním rotorem. Rotor těchto motorů je tvořen permanentním magnetem, po obvodu rotoru se střídají severní a jižní póly budících magnetů.

- **Hybridní točivé krokové motory**

Taktéž jsou nazývány jako aktivní krokové motory s axiálně orientovaným permanentním magnetem. Jde o speciální druh aktivních krokových motorů, jejichž rotor je tvořen axiálně uloženým permanentním magnetem, na jehož koncích jsou umístěny feromagnetické pólové nástavce. Jedná se tedy o jakousi kombinaci obou předcházejících typů krokových motorů.

- **Lineární krokové motory**

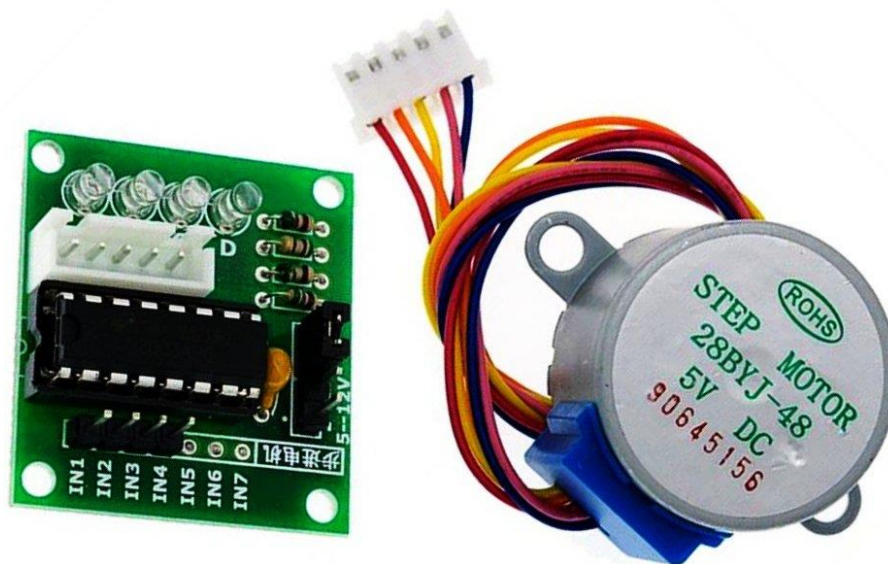
Tyto motory mají neperiodický posuvný pohyb. Stator tohoto motoru je rozvinut do úsečky, má tedy možnost pouze konečného pohybu na jednu stranu. Rotor se netočí, nýbrž posouvá, a je proto potřeba řešit pojezdy, místo ložisek.[13]

5 Výroba a realizace prototypu

V této části práce se zaměří na konkrétní implementaci návrhu a konstrukce ručního dávkovače pájecí pasty s navrženou elektronikou. Tato část bude rozdělena do několika podkapitol, které budou obsahovat postup od návrhu až po testování zařízení.

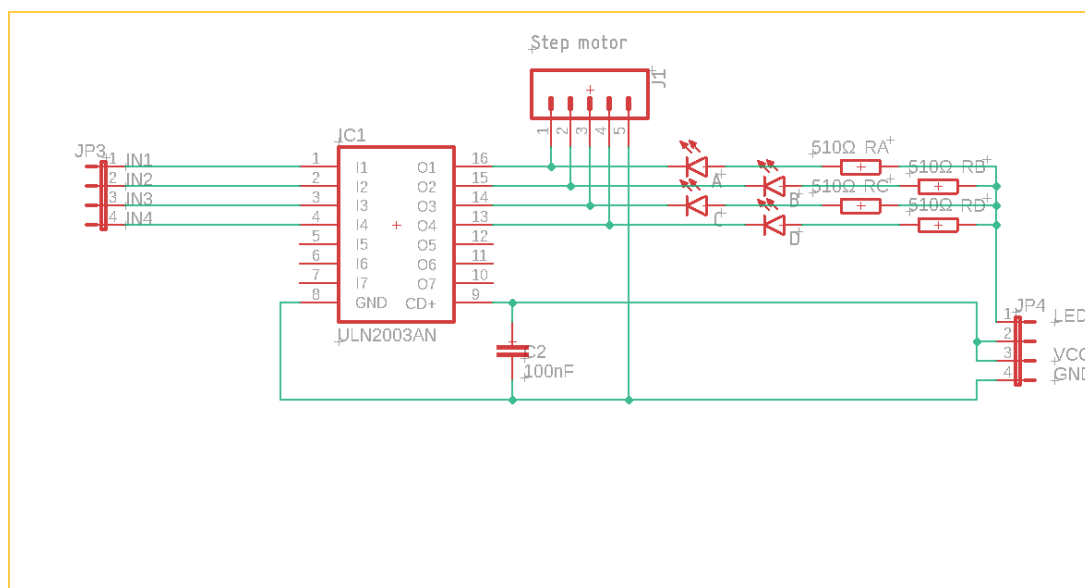
5.1 Návrh a konstrukce elektroniky

Navrhovanou variantou je použití krokového motoru pro pohyb závitové tyče namontované k injekční stříkačce s pastou. Byl vybrán motor 28BYJ-48, jelikož je napájen napětí 5V jako zbytek periférií.



Obrázek 13 - Krokový motor s budičem[20]

Ve stanici je připojen k modulu budiče pro krokové motory, který mimo jiné ochraňuje řídicí obvod od indukčních špiček spojených s použitím indukční zátěže. Tento modul je řízen 4bitovou logikou.

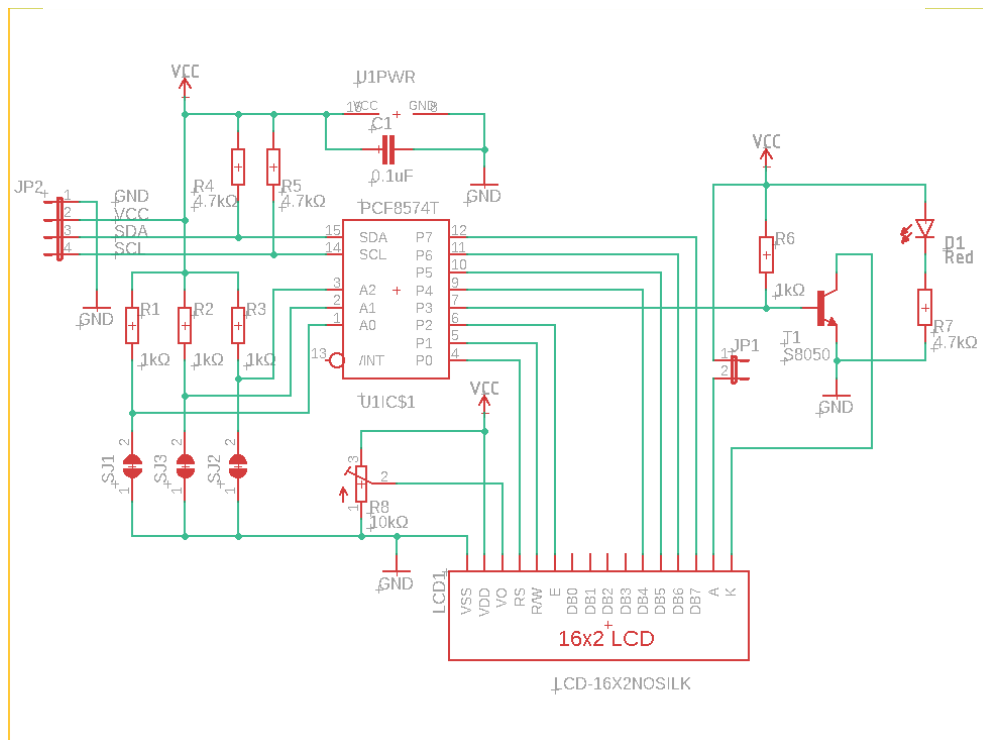


Obrázek 14 - Schéma modulu budiče

O řízení celého zařízení se stará Arduino UNO. Pro indikaci stavu a vybírání jednotlivých přednastavených módů je použit modul dvouřádkového displeje 16x2 znaky s rozhraním I2C. K Arduino jsou ještě připojeny čtyři tlačítka na ovládání celého zařízení. Tlačítka jsou přímo připojena mezi vstupní piny Arduino a GND. Veškeré schémata jsou mnou překreslená pomocí programu Autodesk Eagle.



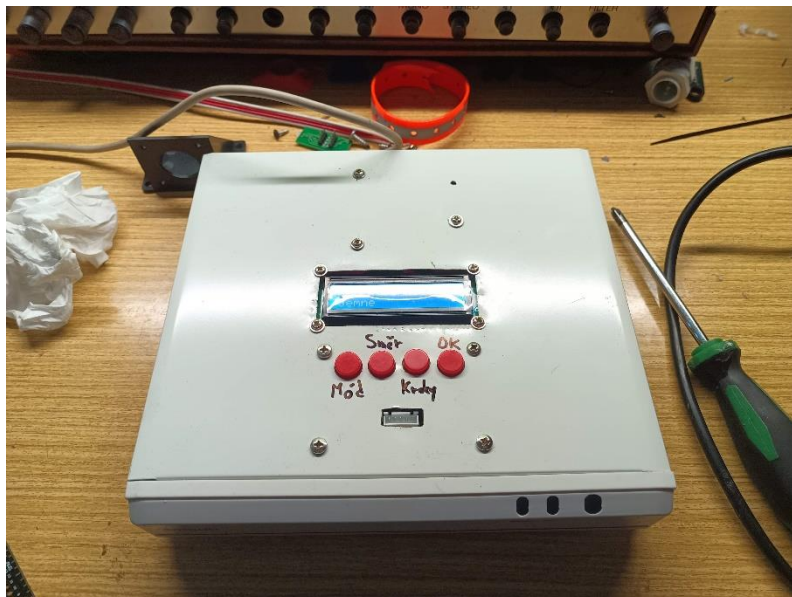
Obrázek 16 - Modul LCD displeje[19]



Obrázek 15 - Schéma modulu displeje s I2C převodníkem

5.2 Fyzická konstrukce

Fyzická konstrukce držáku motoru je vyhotovena na 3D tiskárně z dostupných modelů na internetu. Materiál plastových dílů je z PLA, který je asi nejméně náročný na vytisknutí a je také relativně levný. Zapouzdření stanice celého ovládání s displejem a Arduinem je řešeno pomocí kovové krabičky z tzv. šuplíkových zásob. Konstrukce je možná nevzhledná, ale plní funkci.



Obrázek 17 – Základní stanice

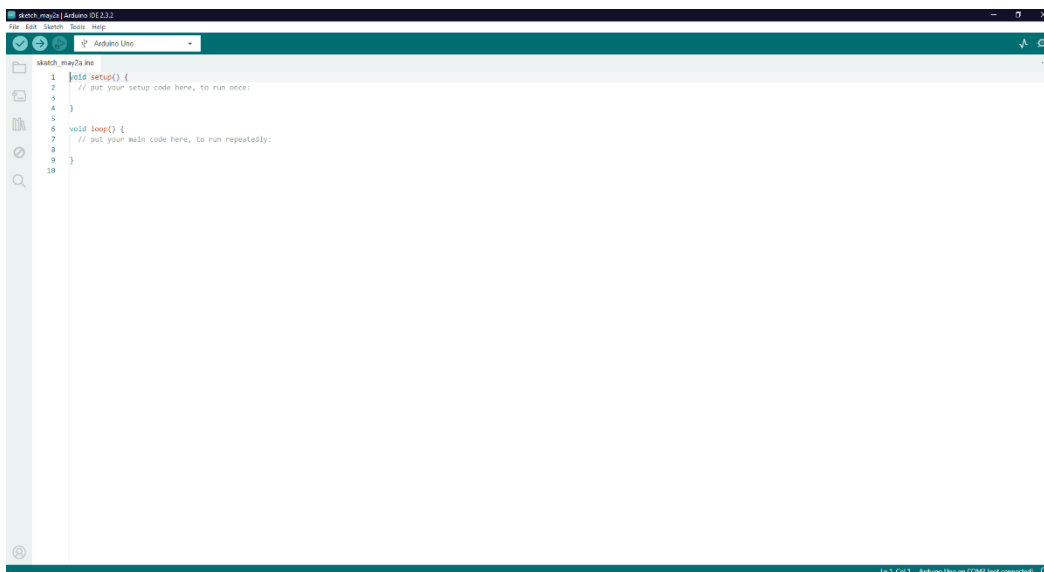


Obrázek 18 – Ruční část dávkovače

5.3 Programové vybavení

Po zapojení modulů a sestavení konstrukce jsem začal s psáním programu. Program byl psán v prostředí Arduino IDE. Toto prostředí je vyvíjeno přímo společností Arduino. V tomto prostředí se programuje pomocí jazyka Wiring, který je postaven na bázi jazyků C a C++.

[14]



Obrázek 19 – Arduino IDE

Program je vytvořen tak, aby dal uživateli na výběr mezi třemi nejpoužívanějšími pouzdry v elektronice, kterými jsou 0603, 0805 a 1206. Dále je zde na výběr mezi jemnými a hrubými kroky krokového motoru. Tyto hodnoty jsou deklarovány jako proměnné v záhlaví programu pro případnou rychlou změnu a dále jsou zde deklarovány proměnné pro jednotlivé kroky a prodlevy. Poté v části setup inicializovány všechny potřebné periferie, knihovny a nastavení vstupních a výstupních pinů. Dále je zde vytvořeno několik void funkcí na obsluhu jednotlivých pohybů, případně souboru pohybů, či podprogramy pro vykreslování textů na displeji. Je zde i funkce loop, ve které program cykluje a čeká na zmáčknutí tlačítek. Po zmáčknutí tlačítka *OK* se program přesune na nastavenou sekvenci úkonů podle předtím nastavených parametrů pomocí tlačítek *Mód*, *Směr*, *Kroky*. Tato sekvence může být například vytlačení pasty ven ze stříkačky pro pouzdro 0805 hrubě. Zároveň se veškeré vybrané parametry vykreslují na displeji, po dokončení operace se na displeji vykreslí hláška *Hotovo*. Zde je úryvek z kódu pro onu verzi 0805 ven hrubě. Zbytek se nachází dále v dokumentu v sekci příloh.

case 1:

```
lcd.clear();  
lcd.print("Dopredu Hrube 0805");  
lcd.setCursor(0, 1);  
lcd.print("Pracuji");
```

```

while(i<KrokHrube0805)
{
    krokDopreduHrube();
    delay(casMotorHrube);
    i++;
}
if(i == KrokHrube0805)
{
    i = 0;
}
lcd.setCursor(0, 1);
lcd.print("Hotovo  ");
vypnutiMotoru();
break;

```

5.4 Testování výrobku

Hotový výrobek byl testován na správné dávkování pro jednotlivé pouzdra. Ačkoliv bylo zařízení testované, není možné s přesností nastavit vytlačované množství díky už zmiňovaným parametrům jako jsou teplota či složení a degradace pasty. Nastavené parametry jsou pouze orientační a je třeba je přenastavit pokaždé s použitím jiné pasty, nebo při rozdílném prostředí dávkování. Zde jsou přednastavené počty kroků, které lze snadno měnit pomocí vývojového prostředí.

```

// Proměnné pro nastavení kroků krokového motoru pro každý mód
int KrokJemne0603 = 1000;
int KrokJemne0805 = 2000;
int KrokJemne1206 = 3000;
int KrokJemneFreeUse = 5000;

int KrokHrube0603 = 500;
int KrokHrube0805 = 1000;
int KrokHrube1206 = 2000;
int KrokHrubeFreeUse = 2500;

```

Dále se zde může měnit i časová prodleva mezi kroky pomocí proměnných *casMotorJemne* a *casMotorHrube*.

Závěr

Psaná část bakalářské práce je rozdělena do několika kapitol. První kapitola se věnuje teorii okolo pájecích past, tavidel a jejich nanášení. Druhá kapitola se zaměřuje na již použitá řešení a jejich výhody a nevýhody. Ve třetí kapitole byly rozebrány problémové faktory při nanášení past. Kapitola čtvrtá se věnuje možnostmi použití různých součástek, jejich stručným popisem a porovnáním výhod a nevýhod jejich použití. A v páté kapitole s dostáváme ke konstrukci a odzkoušení navrhovaného řešení.

Cílem této bakalářské práce bylo vytvoření ručního dávkovače pájecí pasty s uživatelským rozhraním. Dávkovač má displej, který uživateli ukazuje výběr mezi jednotlivými přednastavenými režimy. Tyto režimy dávají na výběr z pohybu krokového motoru či z výběru několika přednastavených SMD pouzder a kontinuální retrakcí.

Celé zařízení je postaveno na desce Arduino Uno a periferie se skládají z modulů k této vývojové desce. O zobrazení informací se stará dvouřádkový displej a celé zařízení se ovládá pomocí čtyř tlačítek. Napájení je řešeno pomocí USB portu. Fyzická konstrukce se skládá z ruční jednotky a stanice. Ve stanici je zapouzdřena veškerá elektronika s ovládáním, ruční jednotka obsahuje pouze nádobu na pájecí pastu a krokový motor na vytlačování pasty.

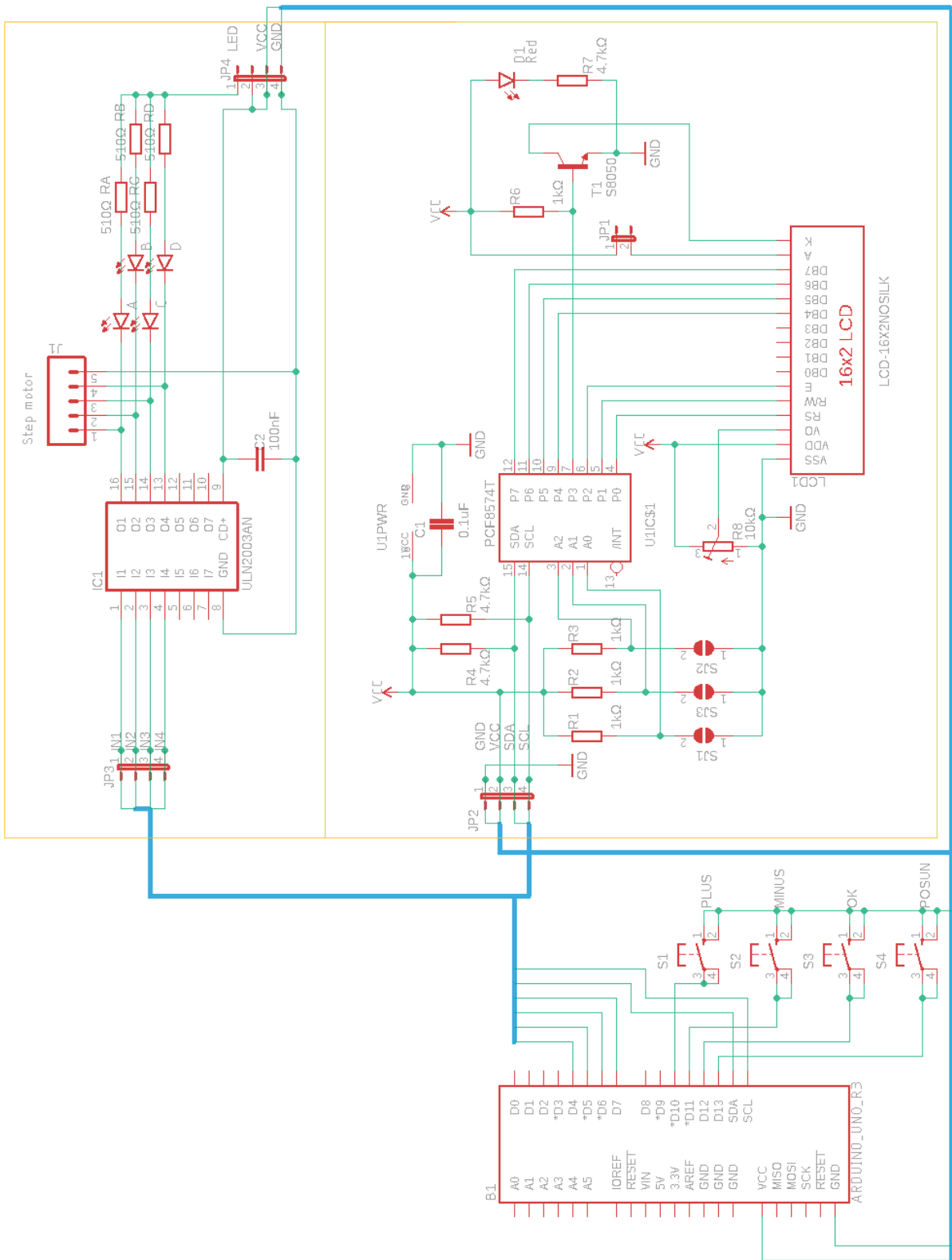
Literatura

- [1] MACHALICKÝ, Vít. *Pájení v elektrotechnice* [online]. Praha, 2020 [cit. 2024-04-25]. Dostupné z: https://dspace.cvut.cz/bitstream/handle/10467/92812/F3-BP-2020-Machalicky-Vit-Pajeni_v_elektrotechnice.pdf?sequence=-1&isAllowed=y. Bakalářská práce. ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE. Vedoucí práce Ing. Ivana Beshajová Pelikánová, Ph.D.
- [2] NUBA Kalafuna Grand - 50g. In: *GME.cz* [online]. [cit. 2024-04-30]. Dostupné z: https://img.gme.cz/detail_big/1200/800/eshop_data/eshop_data/1/745-017/pctdetail.745-017.1.jpg
- [3] BÍM, Jiří. *Kvalita pájených a lepených spojů v elektrotechnice* [online]. Praha, 2018 [cit. 2024-04-25]. Dostupné z: <https://dspace.cvut.cz/bitstream/handle/10467/76502/F3-BP-2018-Bim-Jiri-Kvalita%20pajenych%20a%20lepenych%20spoju%20v%20elektrotechnice.pdf?sequence=-1&isAllowed=y>. Bakalářská práce. ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE. Vedoucí práce Ing. Ivana Beshajová, Ph.D.
- [4] SMT planžeta. In: *Awos.cz* [online]. [cit. 2024-04-30]. Dostupné z: <https://www.awos.cz/data/filecache/c3/SMT-planzety.jpg>
- [5] Schließung von Kirsten Soldering. In: *All-electronics.de* [online]. [cit. 2024-04-30]. Dostupné z: https://www.all-electronics.de/files/upload/post/all/wp-content/uploads/2017/08/kirsten-soldering_jetwave.jpg
- [6] Hotflow 4. In: *PBT* [online]. 2017 [cit. 2024-04-30]. Dostupné z: <https://www.pbt.cz/getattachment/cf784e55-2617-4973-ba04-43f939f8b8e3/Hotflow-4.aspx>
- [7] KORDÍK, Jiří. *Metody pro osazování desek plošných spojů* [online]. Praha, 2019 [cit. 2024-04-25]. Dostupné z: <https://dspace.cvut.cz/bitstream/handle/10467/85067/F3-BP-2019-Kordik-Jiri-Bakalarska%20prace%20KOS.pdf?sequence=-1&isAllowed=y>. Bakalářská práce. ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE. Vedoucí práce Ing. Ivana Beshajová Pelikánová, Ph.D.

- [8] Dávkořač materiálu. In: *Rs-online.com* [online]. [cit. 2024-04-30]. Dostupné z: https://res.cloudinary.com/rsc/image/upload/bo_1.5px_solid_white,b_auto,c_pad,dpr_2,f_auto,h_399,q_auto,w_710/c_pad,h_399,w_710/Y2498138-01?pgw=1
- [9] Arduino Uno Rev3. In: *Store.arduino.cc* [online]. [cit. 2024-04-30]. Dostupné z: https://store.arduino.cc/cdn/shop/products/A000066_03.front_1000x750.jpg?v=1629815860
- [10] ŠANDA, Dobroslav. *Elektropohon pro malé domácí spotřebiče* [online]. Praha, 2014 [cit. 2024-04-28]. Dostupné z: <https://dspace.cvut.cz/bitstream/handle/10467/24728/F3-BP-2014-Sanda-Dobroslav-prace.pdf?sequence=3&isAllowed=y>. Bakalářská práce. ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE. Vedoucí práce Ing. Vít Hlinovský, CSc.
- [11] Střídač. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2024-04-28]. Dostupné z: <https://cs.wikipedia.org/wiki/St%C5%99%C3%ADda%C4%8D>
- [12] Bezkartáčový stejnosměrný motor. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2024-04-28]. Dostupné z: https://cs.wikipedia.org/wiki/Bezkart%C3%A1%C4%8Dov%C3%BD_stejnosm%C4%9Brn%C3%BD_motor
- [13] Krokový motor. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2024-04-28]. Dostupné z: https://cs.wikipedia.org/wiki/Krokov%C3%BD_motor
- [14] Arduino. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2024-04-28]. Dostupné z: <https://cs.wikipedia.org/wiki/Arduino>
- [15] Woodův kov. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2024-04-30]. Dostupné z: https://cs.wikipedia.org/wiki/Wood%C5%AFv_kov
- [16] MANUAL SOLDER PASTE DISPENSER JBC DPM-B. In: *P2jtechnology.cz* [online]. [cit. 2024-04-30]. Dostupné z: https://www.p2jtechnology.cz/764958-thickbox_default/manual-solder-paste-dispenser-jbc-dpm-b.jpg
- [17] *Laser Soldering* [online]. In: . [cit. 2024-04-30]. Dostupné z: <https://www.apollo-seiko-europe.com/laser-soldering>

- [18] NAGYPÁL, Herman. Jaký motor je ve vašem nářadí? In: *Hermantools* [online]. 2019 [cit. 2024-04-29]. Dostupné z: <https://www.hermantools.cz/blog/jaky-motor-je-ve-vasem-naradi>
- [19] IIC I2C displej LCD 1602, 16 x 2 LCD znaků - Modrý modul. In: *Dratek.cz* [online]. [cit. 2024-05-02]. Dostupné z: https://dratek.cz/photos/produkty_gal/f/3/3118.jpg?m=1502871455
- [20] Modul driveru s ULN2003 + krokový motor 28BYJ-48. In: *Kondik.cz* [online]. [cit. 2024-04-30]. Dostupné z: https://cdn.myshoptet.com/usr/www.kondik.cz/user/shop/big/41438-3_yxb315-1.jpg?656e4551

Příloha A – Schéma obvodu



Příloha B – Kód programu

```
#include <LiquidCrystal_I2C.h>

#define PinMotoru1 4
#define PinMotoru2 5
#define PinMotoru3 6
#define PinMotoru4 7
#define tlacitkoMODPin 13
#define tlacitkoSMERPin 12
#define tlacitkoJEMNOSTPin 11
#define tlacitkoOKPin 10
LiquidCrystal_I2C lcd(0x27, 16, 2);

int vybranyMod = 0; // 0: 0603, 1: 0805, 2: 1206, 3: Free Use
int smerMotoru = 0; // 0: dopredu, 1: dozadu
int jemnostMotoru = 0; // 0: jemne, 1: hrube
int step_number = 0; //pocitadlo jednotlivych kroku krok. motoru
int casMotorJemne = 1; //delay pro jednotlivlive kroky
int casMotorHrube = 5;
int casTlacitko = 300; //delay mezi stisknutími tlačítek
int i = 0;

// Proměnné pro nastavení ovládání krokového motoru pro každý mód
int KrokJemne0603 = 1000;
int KrokJemne0805 = 2000;
int KrokJemne1206 = 3000;
int KrokJemneFreeUse = 5000;

int KrokHrube0603 = 500;
int KrokHrube0805 = 1000;
int KrokHrube1206 = 2000;
int KrokHrubeFreeUse = 2500;

void setup()
{
  pinMode(tlacitkoMODPin, INPUT_PULLUP);
  pinMode(tlacitkoSMERPin, INPUT_PULLUP);
  pinMode(tlacitkoJEMNOSTPin, INPUT_PULLUP);
  pinMode(tlacitkoOKPin, INPUT_PULLUP);
  pinMode(PinMotoru1, OUTPUT);
  pinMode(PinMotoru2, OUTPUT);
  pinMode(PinMotoru3, OUTPUT);
  pinMode(PinMotoru4, OUTPUT);

  lcd.begin(16, 2);
  lcd.clear();
}
```

```

    lcd.setBacklight(HIGH);
    lcd.print("Vyber mod/smer/jemnost:");
    lcd.setCursor(0, 1);
    aktualizujDisplejModu();
    aktualizujDisplejSmeru();
    aktualizujDisplejJemnosti();
}

void loop()
{
    if (digitalRead(tlacitkoMODPin) == LOW)
    {
        zmenitMod(1);
        delay(casTlacitko); // Zpoždění pro zabránění rychlé změny módu
    }
    else if (digitalRead(tlacitkoSMERPin) == LOW)
    {
        zmenitSmer(1);
        delay(casTlacitko); // Zpoždění pro zabránění rychlé změny módu
    }
    else if (digitalRead(tlacitkoJEMNOSTPin) == LOW)
    {
        zmenJemnost(1);
        delay(casTlacitko); // Zpoždění pro zabránění rychlé změny módu
    }
    else if (digitalRead(tlacitkoOKPin) == LOW)
    {
        zacniTocit();
        delay(casTlacitko); // Zpoždění pro zabránění rychlé změny módu
    }
    else
    {
    }
}

void zacniTocit()
{
    switch(jemnostMotoru)
    {
        case 0:
            switch(smerMotoru)
            {
                case 0:

```

```

switch(vybranyMod)
{
case 0:
  lcd.clear();
  lcd.print("Dopredu Jemne 0603");
  lcd.setCursor(0, 1);
  lcd.print("Pracuji");
  while(i<KrokJemne0603)
  {
    krokDopreduJemne();
    delay(casMotorJemne);
    i++;
  }
  if(i == KrokJemne0603)
  {
    i = 0;
  }
  lcd.setCursor(0, 1);
  lcd.print("Hotovo ");
  vypnutiMotoru();
  break;
case 1:
  lcd.clear();
  lcd.print("Dopredu Jemne 0805");
  lcd.setCursor(0, 1);
  lcd.print("Pracuji");
  while(i<KrokJemne0805)
  {
    krokDopreduJemne();
    delay(casMotorJemne);
    i++;
  }
  if(i == KrokJemne0805)
  {
    i = 0;
  }
  lcd.setCursor(0, 1);
  lcd.print("Hotovo ");
  vypnutiMotoru();
  break;
case 2:
  lcd.clear();
  lcd.print("Dopredu Jemne 1206");
  lcd.setCursor(0, 1);
  lcd.print("Pracuji");
  while(i<KrokJemne1206)
  {
    krokDopreduJemne();

```

```

    delay(casMotorJemne);
    i++;
}
if(i == KrokJemne1206)
{
    i = 0;
}
lcd.setCursor(0, 1);
lcd.print("Hotovo ");
vypnutiMotoru();
break;
case 3:
lcd.clear();
lcd.print("Dopredu Jemne Free Use");
lcd.setCursor(0, 1);
lcd.print("Pracuji");
while(i<KrokJemneFreeUse)
{
    krokDopreduJemne();
    delay(casMotorJemne);
    i++;
}
if(i == KrokJemneFreeUse)
{
    i = 0;
}
lcd.setCursor(0, 1);
lcd.print("Hotovo ");
vypnutiMotoru();
break;
}
break;
case 1:
switch(vybranyMod)
{
case 0:
lcd.clear();
lcd.print("Dozadu Jemne 0603");
lcd.setCursor(0, 1);
lcd.print("Pracuji");
while(i<KrokJemne0603)
{
    krokDozaduJemne();
    delay(casMotorJemne);
    i++;
}
if(i == KrokJemne0603)
{

```

```

    i = 0;
}
lcd.setCursor(0, 1);
lcd.print("Hotovo ");
vypnutiMotoru();
break;
case 1:
lcd.clear();
lcd.print("Dozadu Jemne 0805");
lcd.setCursor(0, 1);
lcd.print("Pracuji");
while(i<KrokJemne0805)
{
    krokDozaduJemne();
    delay(casMotorJemne);
    i++;
}
if(i == KrokJemne0805)
{
    i = 0;
}
lcd.setCursor(0, 1);
lcd.print("Hotovo ");
vypnutiMotoru();
break;
case 2:
lcd.clear();
lcd.print("Dozadu Jemne 1206");
lcd.setCursor(0, 1);
lcd.print("Pracuji");
while(i<KrokJemne1206)
{
    krokDozaduJemne();
    delay(casMotorJemne);
    i++;
}
if(i == KrokJemne1206)
{
    i = 0;
}
lcd.setCursor(0, 1);
lcd.print("Hotovo ");
vypnutiMotoru();
break;
case 3:
lcd.clear();
lcd.print("Dozadu Jemne Free Use");
lcd.setCursor(0, 1);

```

```

    lcd.print("Pracuji");
    while(i<KrokJemneFreeUse)
    {
        krokDozaduJemne();
        delay(casMotorJemne);
        i++;
    }
    if(i == KrokJemneFreeUse)
    {
        i = 0;
    }
    lcd.setCursor(0, 1);
    lcd.print("Hotovo  ");
    vypnutiMotoru();
    break;
}
break;
break;
case 1:
    switch(smerMotoru)
    {
        case 0:
            switch(vybranyMod)
            {
                case 0:
                    lcd.clear();
                    lcd.print("Dopredu Hrube 0603");
                    lcd.setCursor(0, 1);
                    lcd.print("Pracuji");
                    while(i<KrokHrube0603)
                    {
                        krokDopreduHrube();
                        delay(casMotorHrube);
                        i++;
                    }
                    if(i == KrokHrube0603)
                    {
                        i = 0;
                    }
                    lcd.setCursor(0, 1);
                    lcd.print("Hotovo  ");
                    vypnutiMotoru();
                    break;
                case 1:
                    lcd.clear();
                    lcd.print("Dopredu Hrube 0805");

```

```

lcd.setCursor(0, 1);
lcd.print("Pracuji");
while(i<KrokHrube0805)
{
    krokDopreduHrube();
    delay(casMotorHrube);
    i++;
}
if(i == KrokHrube0805)
{
    i = 0;
}
lcd.setCursor(0, 1);
lcd.print("Hotovo ");
vypnutiMotoru();
break;
case 2:
lcd.clear();
lcd.print("Dopredu Hrube 1206");
lcd.setCursor(0, 1);
lcd.print("Pracuji");
while(i<KrokHrube1206)
{
    krokDopreduHrube();
    delay(casMotorHrube);
    i++;
}
if(i == KrokHrube1206)
{
    i = 0;
}
lcd.setCursor(0, 1);
lcd.print("Hotovo ");
vypnutiMotoru();
break;
case 3:
lcd.clear();
lcd.print("Dopredu Hrube Free Use");
lcd.setCursor(0, 1);
lcd.print("Pracuji");
while(i<KrokHrubeFreeUse)
{
    krokDopreduHrube();
    delay(casMotorHrube);
    i++;
}
if(i == KrokHrubeFreeUse)
{

```



```

        i = 0;
    }
    lcd.setCursor(0, 1);
    lcd.print("Hotovo ");
    vypnutiMotoru();
    break;
}
break;
case 1:
switch(vybranyMod)
{
case 0:
lcd.clear();
lcd.print("Dozadu Hrube 0603");
lcd.setCursor(0, 1);
lcd.print("Pracuji");
while(i<KrokHrube0603)
{
    krokDozaduHrube();
    delay(casMotorHrube);
    i++;
}
if(i == KrokHrube0603)
{
    i = 0;
}
lcd.setCursor(0, 1);
lcd.print("Hotovo ");
vypnutiMotoru();
break;
case 1:
lcd.clear();
lcd.print("Dozadu Hrube 0805");
lcd.setCursor(0, 1);
lcd.print("Pracuji");
while(i<KrokHrube0805)
{
    krokDozaduHrube();
    delay(casMotorHrube);
    i++;
}
if(i == KrokHrube0805)
{
    i = 0;
}
lcd.setCursor(0, 1);
lcd.print("Hotovo ");
vypnutiMotoru();

```

```

break;
case 2:
lcd.clear();
lcd.print("Dozadu Hrube 1206");
lcd.setCursor(0, 1);
lcd.print("Pracuji");
while(i<KrokHrube1206)
{
    krokDozaduHrube();
    delay(casMotorHrube);
    i++;
}
if(i == KrokHrube1206)
{
    i = 0;
}
lcd.setCursor(0, 1);
lcd.print("Hotovo ");
vypnutiMotoru();
break;
case 3:
lcd.clear();
lcd.print("Dozadu Hrube Free Use");
lcd.setCursor(0, 1);
lcd.print("Pracuji");
while(i<KrokHrubeFreeUse)
{
    krokDozaduHrube();
    delay(casMotorHrube);
    i++;
}
if(i == KrokHrubeFreeUse)
{
    i = 0;
}
lcd.setCursor(0, 1);
lcd.print("Hotovo ");
vypnutiMotoru();
break;
}
break;
}
}

void krokDopreduHrube()
{

```

```

switch(step_number){
  case 0:
    digitalWrite(PinMotoru1, HIGH);
    digitalWrite(PinMotoru2, LOW);
    digitalWrite(PinMotoru3, LOW);
    digitalWrite(PinMotoru4, LOW);
    break;
  case 1:
    digitalWrite(PinMotoru1, LOW);
    digitalWrite(PinMotoru2, HIGH);
    digitalWrite(PinMotoru3, LOW);
    digitalWrite(PinMotoru4, LOW);
    break;
  case 2:
    digitalWrite(PinMotoru1, LOW);
    digitalWrite(PinMotoru2, LOW);
    digitalWrite(PinMotoru3, HIGH);
    digitalWrite(PinMotoru4, LOW);
    break;
  case 3:
    digitalWrite(PinMotoru1, LOW);
    digitalWrite(PinMotoru2, LOW);
    digitalWrite(PinMotoru3, LOW);
    digitalWrite(PinMotoru4, HIGH);
    break;
}

step_number++;
if(step_number > 3)
{
  step_number = 0;
}
}

```

```

void krokDopreduJemne()
{
  switch(step_number){
    case 0:
      digitalWrite(PinMotoru1, HIGH);
      digitalWrite(PinMotoru2, LOW);
      digitalWrite(PinMotoru3, LOW);
      digitalWrite(PinMotoru4, LOW);
      break;
    case 1:
      digitalWrite(PinMotoru1, HIGH);
      digitalWrite(PinMotoru2, HIGH);
      digitalWrite(PinMotoru3, LOW);
      digitalWrite(PinMotoru4, LOW);

```

```

        break;
    case 2:
        digitalWrite(PinMotoru1, LOW);
        digitalWrite(PinMotoru2, HIGH);
        digitalWrite(PinMotoru3, LOW);
        digitalWrite(PinMotoru4, LOW);
        break;
    case 3:
        digitalWrite(PinMotoru1, LOW);
        digitalWrite(PinMotoru2, HIGH);
        digitalWrite(PinMotoru3, HIGH);
        digitalWrite(PinMotoru4, LOW);
        break;
    case 4:
        digitalWrite(PinMotoru1, LOW);
        digitalWrite(PinMotoru2, LOW);
        digitalWrite(PinMotoru3, HIGH);
        digitalWrite(PinMotoru4, LOW);
        break;
    case 5:
        digitalWrite(PinMotoru1, LOW);
        digitalWrite(PinMotoru2, LOW);
        digitalWrite(PinMotoru3, HIGH);
        digitalWrite(PinMotoru4, HIGH);
        break;
    case 6:
        digitalWrite(PinMotoru1, LOW);
        digitalWrite(PinMotoru2, LOW);
        digitalWrite(PinMotoru3, LOW);
        digitalWrite(PinMotoru4, HIGH);
        break;
    }

    step_number++;
    if(step_number > 6)
    {
        step_number = 0;
    }
}

void krokDozaduHrube()
{
    switch(step_number){
        case 0:
            digitalWrite(PinMotoru1, LOW);
            digitalWrite(PinMotoru2, LOW);
            digitalWrite(PinMotoru3, LOW);
            digitalWrite(PinMotoru4, HIGH);
            break;

```

```

    case 1:
    digitalWrite(PinMotoru1, LOW);
    digitalWrite(PinMotoru2, LOW);
    digitalWrite(PinMotoru3, HIGH);
    digitalWrite(PinMotoru4, LOW);
    break;
    case 2:
    digitalWrite(PinMotoru1, LOW);
    digitalWrite(PinMotoru2, HIGH);
    digitalWrite(PinMotoru3, LOW);
    digitalWrite(PinMotoru4, LOW);
    break;
    case 3:
    digitalWrite(PinMotoru1, HIGH);
    digitalWrite(PinMotoru2, LOW);
    digitalWrite(PinMotoru3, LOW);
    digitalWrite(PinMotoru4, LOW);
    break;
}

step_number++;
if(step_number > 3)
{
step_number = 0;
}
}

void krokDozaduJemne()
{
switch(step_number){
case 0:
digitalWrite(PinMotoru1, LOW);
digitalWrite(PinMotoru2, LOW);
digitalWrite(PinMotoru3, LOW);
digitalWrite(PinMotoru4, HIGH);
break;
case 1:
digitalWrite(PinMotoru1, LOW);
digitalWrite(PinMotoru2, LOW);
digitalWrite(PinMotoru3, HIGH);
digitalWrite(PinMotoru4, HIGH);
break;
case 2:
digitalWrite(PinMotoru1, LOW);
digitalWrite(PinMotoru2, LOW);
digitalWrite(PinMotoru3, HIGH);
digitalWrite(PinMotoru4, LOW);
break;

```

```

    case 3:
    digitalWrite(PinMotoru1, LOW);
    digitalWrite(PinMotoru2, HIGH);
    digitalWrite(PinMotoru3, HIGH);
    digitalWrite(PinMotoru4, LOW);
    break;
    case 4:
    digitalWrite(PinMotoru1, LOW);
    digitalWrite(PinMotoru2, HIGH);
    digitalWrite(PinMotoru3, LOW);
    digitalWrite(PinMotoru4, LOW);
    break;
    case 5:
    digitalWrite(PinMotoru1, HIGH);
    digitalWrite(PinMotoru2, HIGH);
    digitalWrite(PinMotoru3, LOW);
    digitalWrite(PinMotoru4, LOW);
    break;
    case 6:
    digitalWrite(PinMotoru1, HIGH);
    digitalWrite(PinMotoru2, LOW);
    digitalWrite(PinMotoru3, LOW);
    digitalWrite(PinMotoru4, LOW);
    break;
    }

    step_number++;
    if(step_number > 6)
    {
        step_number = 0;
    }
}

void vypnutiMotoru()
{
    digitalWrite(PinMotoru1, LOW);
    digitalWrite(PinMotoru2, LOW);
    digitalWrite(PinMotoru3, LOW);
    digitalWrite(PinMotoru4, LOW);
}

void zmenitMod(int mod)
{
    vybranyMod = (vybranyMod + mod + 4) % 4; // Přepínání mezi módy (0, 1, 2,
3)
    aktualizujDisplejModu();
}

```

```

void zmenitSmer(int smer)
{
    smerMotoru = (smerMotoru + smer + 2) % 2; // Přepínání mezi módy (0, 1)
    aktualizujDisplejSmeru();
}
void zmenJemnost(int jemnost)
{
    jemnostMotoru = (jemnostMotoru + jemnost + 2) % 2; // Přepínání mezi módy
(0, 1)
    aktualizujDisplejJemnosti();
}

void aktualizujDisplejModu()
{
    lcd.clear();
    lcd.print("Vyber mod:");
    lcd.setCursor(0, 1);
    if (vybranyMod == 0)
    {
        lcd.print("0603");
    }
    else if (vybranyMod == 1)
    {
        lcd.print("0805");
    }
    else if (vybranyMod == 2)
    {
        lcd.print("1206");
    }
    else if (vybranyMod == 3)
    {
        lcd.print("Free Use");
    }
}

void aktualizujDisplejSmeru()
{
    lcd.clear();
    lcd.print("Vyber smer:");
    lcd.setCursor(0, 1);
    if (smerMotoru == 0)
    {
        lcd.print("Dopredu");
    }
    else if (smerMotoru == 1)
    {
        lcd.print("Dozadu");
    }
}

```

```
    }  
  }  
  void aktualizujDisplejJemnosti()  
  {  
    lcd.clear();  
    lcd.print("Vyber jemnost:");  
    lcd.setCursor(0, 1);  
    if (jemnostMotoru == 0)  
    {  
      lcd.print("Jemne");  
    }  
    else if (jemnostMotoru == 1)  
    {  
      lcd.print("Hrube");  
    }  
  }  
}
```