



ZÁPADOČESKÁ
UNIVERZITA
V PLZNI

Fakulta elektrotechnická
Katedra teoretické elektrotechniky

BAKALÁŘSKÁ PRÁCE

Řízení peristaltického čerpadla

ZÁPADOČESKÁ UNIVERZITA V PLZNI

Fakulta elektrotechnická

Akademický rok: 2018/2019

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Filip PRŮCHA**

Osobní číslo: **E16B0047P**

Studijní program: **B2612 Elektrotechnika a informatika**

Studijní obor: **Elektrotechnika a energetika**

Název tématu: **Návrh a realizace řízení peristaltického čerpadla**

Zadávací katedra: **Katedra elektromechaniky a výkonové elektroniky**

Z á s a d y p r o v y p r a c o v á n í :

1. Uveďte způsoby řízení lineárních čerpadel malých rozměrů.
2. Navrhněte systém řízení čerpadla a proveďte simulaci.
3. Vyberte vhodnou platformu a systém realizujte.
4. Srovnajte výsledky modelu a experimentu na reálném zařízení.



Rozsah grafických prací: **podle doporučení vedoucího**

Rozsah kvalifikační práce: **30 - 40 stran**

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

Seznam odborné literatury:

1. **HUSÁK, Miroslav. Mikrosenzory a mikroaktuátory. Praha: Academia, 2008. Gerstner. ISBN 978-80-200-1478-8.**

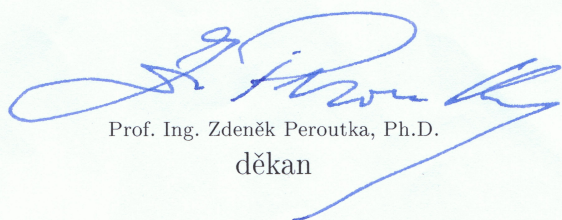
Vedoucí bakalářské práce:

Ing. Iveta Petrášová

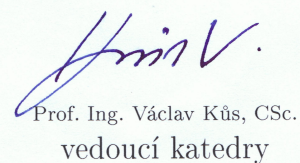
Katedra teoretické elektrotechniky

Datum zadání bakalářské práce: **5. října 2018**

Termín odevzdání bakalářské práce: **13. června 2019**



Prof. Ing. Zdeněk Peroutka, Ph.D.
děkan



Prof. Ing. Václav Kůs, CSc.
vedoucí katedry

V Plzni dne 5. října 2018

Abstrakt

Bakalářská práce se zabývá návrhem řízení lineárního peristaltického čerpadla, využívajícího pěti elektromagnetických aktuátorů k čerpání kapaliny. V rámci práce byl navrhnut a vytvořen tištěný spoj sloužící k řízení výše zmíněného čerpadla. Po realizaci řídicí elektroniky byla provedena měření průtoků čerpadla a průběhu napětí na aktuátoru při spínání.

Klíčová slova

Čerpadlo, hydrostatické čerpadlo, objemové čerpadlo, peristaltické čerpadlo, lineární peristaltické čerpadlo, řízení čerpadla, arduino, spínací cyklus

Abstract

The bachelor thesis deals with the design of linear peristaltic pump control using five electromagnetic actuators for pumping liquid. In the work there was designed and created printed circuit board used to control the above mentioned pump. After the control electronics were realized, the flow rates of the pump and the voltage on the actuator during switching were measured.

Keywords

Pump, hydrostatic pump, positive displacement pump, peristaltic pump, linear peristaltic pump, pump control, arduino, driving of linear peristaltic pump

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně, s použitím odborné literatury a pramenů uvedených v seznamu, který je součástí této bakalářské práce.

Dále prohlašuji, že veškerý software, použitý při řešení této bakalářské práce, je legální.

V Plzni dne 13. června 2019

Filip Průcha

.....

Podpis

Poděkování

Tímto bych rád poděkoval mé vedoucí práce paní Ing. Ivetě Petrášové, za odborné vedení práce, věcné připomínky a vstřícný přístup v průběhu konzultací. Dále bych rád poděkoval panu prof. Ing. Pavlu Karbanovi, Ph.D. za možnost podílet se na řešení této zajímavé problematiky. V neposlední řadě děkuji rodičům za poskytovanou podporu v průběhu mého studia.

Obsah

| | |
|--|-----------|
| 1. Čerpadla | 10 |
| 1.1. Hydrodynamická čerpadla | 11 |
| 1.2. Hydrostatická čerpadla | 12 |
| 2. Peristaltické čerpadlo | 13 |
| 2.1. Rotační peristaltická čerpadla | 13 |
| 2.1.1. Řízení | 14 |
| 2.2. Lineární peristaltická čerpadla | 15 |
| 2.2.1. Řízení | 16 |
| 3. Arduino | 17 |
| 3.1. Arduino Uno | 17 |
| 4. Realizace čerpadla | 20 |
| 4.1. Návrh tištěného spoje | 21 |
| 4.1.1. Zdroj 5 V | 23 |
| 4.1.2. OLED displej | 23 |
| 4.1.3. Spínání cívek | 23 |
| 4.1.4. Tlačítka | 24 |
| 4.1.5. Svorkovnice | 25 |
| 4.2. Výroba DPS | 25 |
| 4.3. Uživatelské rozhraní | 27 |
| 4.4. Měření průtoku média | 28 |
| 4.4.1. Spínací cyklus 1 aktuátor | 29 |
| 4.4.2. Spínací cyklus 2 aktuátory | 31 |
| 4.4.3. Spínací cyklus 3 aktuátory | 33 |
| 4.4.4. Spínací cyklus ob jeden aktuátor | 35 |
| 4.5. Simulace průběhu napětí na elektromagnetickém aktuátoru | 36 |
| 4.5.1. Simulace | 37 |
| 4.6. Zhodnocení | 39 |
| 5. Závěr | 40 |
| A. Program | 42 |

Úvod

Tato bakalářská práce se zabývá návrhem řízení lineárního peristaltického čerpadla. Práce je rozdělena do dvou stěžejních částí. První část je teoretická a popisuje možnosti čerpání kapalin a možný způsob řízení čerpadel. Druhá část je poté zaměřena na realizaci návrhu řízení pro již vyrobené peristaltické čerpadlo.

Důvodem zkoumání je vyvinout mikročerpadlo pro medicínu. Práce vznikla za účelem vyrobení modelu čerpadla, na kterém se prakticky ověří vhodnost tohoto typu čerpadla pro další výzkum.

Peristaltické čerpadlo je poměrně specifickým čerpadlem. V současnosti pro toto čerpadlo, resp. pumpu, lze najít uplatnění v lékařských oborech a farmacii, ale také v potravinářských, kosmetických, chemických a stavebních provozech. Peristaltická čerpadla jsou inspirována peristaltickým pohybem orgánů v zažívacím traktu živého organismu, například střev, která se postupně stahují a uvolňují, tím dochází k pohybu potravy ve střevech. Na obdobném principu fungují tato čerpadla.

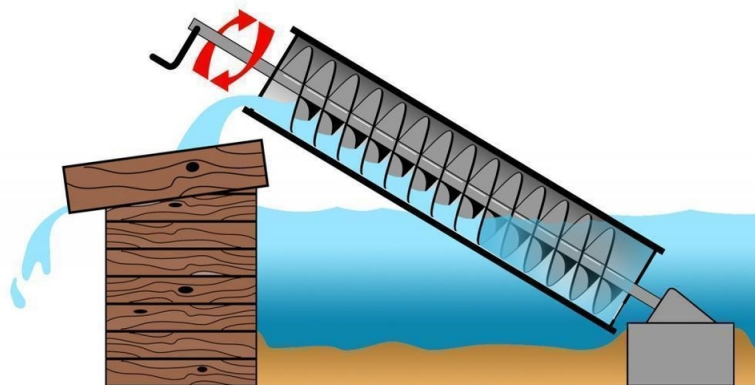
V první kapitole je popsán základní princip čerpadel a jejich dělení dle principu, pomocí něhož přeměňují mechanickou energii na hydraulickou. Druhá kapitola je již zaměřena na peristaltická čerpadla, která lze řídit pomocí Arduina, které je popsáno v další kapitole.

V poslední kapitole je blíže popsáno již realizované peristaltické čerpadlo, pro něž je realizováno řízení. V této kapitole je dále popsán realizovaný návrh a porovnává úspěšnost s ohledem na experimentální měření, následně navazuje poslední část věnovaná simulaci s možnými návrhy na zlepšení realizace spínaného obvodu.

1. Čerpadla

Čerpadlo je mechanický stroj, který je používán k transportu kapaliny (nebo jiného média) z místa **A** na místo **B**. Funguje na principu přeměny mechanické energie na kinetickou, potenciální, nebo tlakovou energii čerpaného média. Díky tomuto lze kapalinu přepravovat z níže položeného místa na místo výše položené nebo na výtlaku čerpadla získat kapalinu o vyšším tlaku, než byl tlak dopravované kapaliny na sání čerpadla.

Tento stroj známe již ze starověku, kdy byl používán k přepravě vody na zavlažování nebo do dopravy center měst. Například staří Řekové čerpali vodu do akvaduktů pomocí takzvaného Archimédova šroubu (Obrázek 1.1), což je šikmo uložená šneková hřídel ve žlabu nebo v trubce, jejímž otáčením dochází k čerpání kapaliny. V té době byla pro pohon čerpadla využívána zvířecí nebo lidská síla. Dnes se již k předávání potřebné energie vodě používá motorů nebo jiných aktuátorů.

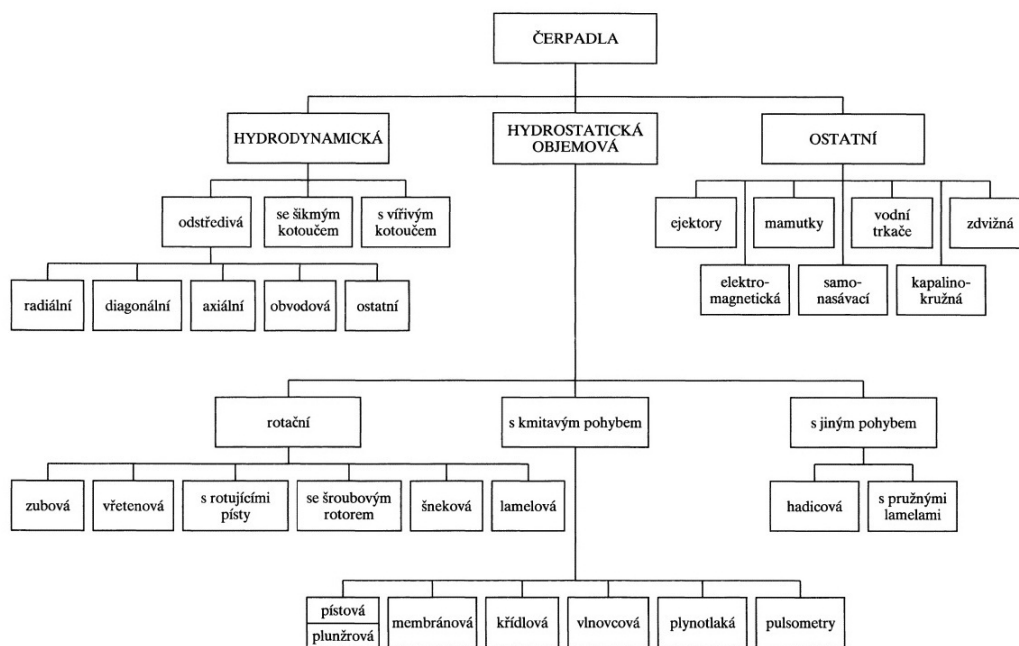


Obrázek 1.1.: Archimédův šroub [3]

Čerpadla jsou hojně využívána v řadě oblastí lidské činnosti, jsou implementována prakticky do všech technologických procesů a provozů. V průmyslu jsou čerpadla tak hojně využívána, že je to řadí v žebříčku nejpoužívanějších strojů na druhé místo, hned za elektromotory.

Základní dělení čerpadel je provedeno podle principu, jakým čerpadla přeměňují mechanickou energii na energii hydraulickou, a to zpravidla buď rotačním pohybem nebo pomocí lineárního posuvu nebo na základě jiného principu. Na základě tohoto principu rozdělujeme čerpadla na [1], [5], [2]:

- **Hydrodynamická**
- **Hydrostatická (Objemová)**
- **Ostatní**



Obrázek 1.2.: Schéma základního rozdělení čerpadel [4]

1.1. Hydrodynamická čerpadla

Hydrodynamická čerpadla neboli odstředivá přeměňují mechanickou energii pohonu na hydraulickou tak, že nejdříve prostřednictvím pracovního elementu (oběžného kola) přemění mechanickou energii na kinetickou energii kapaliny, a ta se poté transformuje v rozváděcím kole na hydraulickou energii (tlakovou energii). Říkáme tedy, že mechanická práce a hydraulická energie se transformují nepřímo. Můžeme konstatovat, že tato čerpadla se používají pro větší objemy a nízké tlaky .

Díky dvojí přeměně energie jsou hydrodynamická čerpadla oproti hydrostatickým čerpadlům méně účinná. Výhodou jsou jejich menší rozměry a hmotnost, čerpadla pracují ve větších otáčkách. To má za následek nižší pořizovací náklady. Konstrukční nevýhodou je potřeba naplnění pracovní komory čerpaným médiem, čerpadlo není schopno si kapalinu samo nasát. Výjimku tvoří samonasávací čerpadla, například čerpadla pro hasičské stříkačky.

Tento typ čerpadel má širokou škálu uplatnění, lze jej najít v domácnostech (čerpadlo ve studni nebo v bazénu), ale i v průmyslu a energetice (oběhová čerpadla v jaderné elektrárně).

Hydrodynamická čerpadla můžeme dělit podle směru výtoku z oběžného kola na [1], [5], [2]:

- Radiální
- Axiální
- Diagonální

1.2. Hydrostatická čerpadla

Hydrostatická čerpadla, také nazývána objemová, přeměňují mechanickou energii pohonu na hydraulickou prostřednictvím pístu (nebo jiného pracovního elementu). U této přeměny se uplatňuje Pascalův zákon. Říkáme tedy, že mechanická práce a hydraulická energie se vzájemně transformují přímo.

U hydrostatických čerpadel rovněž vycházejí při stejných výkonech větší rozměry než u hydrodynamických, tudíž jsou i dražší. Výhodou oproti hydrodynamickým čerpadlům je jejich vysoká účinnost.

Objemová čerpadla dále dělíme na:

- Rotační čerpadla - zubová, vřetenová, šneková apod.
- Čerpadla s kmitavým pohybem - pístová, membránová apod.
- Čerpadla s jiným pohybem - peristaltická, s pružnými lamelami.

V praxi mají hydrostatická čerpadla využití v mnoha odvětvích, lze je nalézt, stejně jako u hydrodynamických čerpadel, v běžném životě (olejové čerpadlo v automobilu), ale také v průmyslu (pístové čerpadlo pro přesné dávkování) [1], [5], [2]. Podrobněji jsou tato čerpadla rozebrána v práci Jiřího Mušky [6].

2. Peristaltické čerpadlo

Jak je zobrazeno na obrázku 1.2, peristaltická čerpadla řadíme dle jejich fyzikálního principu mezi hydrostatická čerpadla. Podle dělení konstrukčních řešení čerpadel patří peristaltické čerpadlo mezi hadicová čerpadla, proto jsou peristaltická čerpadla také nazývána hadicová. Za zmínku stojí rovněž skutečnost, že výraz „peristaltický“ je použit z medicíny, neboť funkce peristaltického pohybu lidských orgánů, konkrétně střev, je srovnatelná s funkcí, jakou je dopravováno čerpané medium v peristaltických čerpadlech [9].

Hlavní výhodou použití peristaltických čerpadel je zaručená čistota, resp. nemožnost znečištění dopravované kapaliny. Toho je dosaženo za prvé tím, že dopravované médium přijde do kontaktu pouze s vnitřní stěnou hadice (zamezí se tedy jakémukoliv dotyku s jinou komponentou čerpadla), za druhé tím, že hadice je vyrobena z chemicky neutrálního materiálu. Mezi hlavní nevýhody patří omezená životnost použité hadičky, z důvodu velkého mechanického namáhání při opakovaném stlačování hadičky. Z této příčiny musí být použit pomaloběžný stroj [9].

Za tímto účelem byl tento typ čerpadel naprojektován a je úspěšně používán v laboratořích, lékařství, chemickém a potravinářském průmyslu a dalších odvětvích. Kromě bezúnikovosti se dále vyznačují přesným konstantním průtokem. Čerpadla jsou spolehlivá pro malé průtoky a tlaky při otáčkách 20 až 200 min⁻¹ [9].

Peristaltická čerpadla jsou dělena podle použitého designu na:

- Rotační peristaltická čerpadla
- Lineární peristaltická čerpadla

2.1. Rotační peristaltická čerpadla

Kapalina je u tohoto čerpadla dopravována pomocí pružné hadice, která je stlačována rotujícími kladkami. Provedení je dvou nebo i více kladkové.

Princip čerpání rotačního peristaltického čerpadla můžeme ukázat pomocí obrázku 2.1. Cyklus rotace můžeme rozdělit na čtyři kroky:

Krok 1

Peristaltické čerpadlo se otáčí a vytváří těsnění na sací straně čerpadla. Uzavírá také **modrou** část kapaliny a začíná ji tlačít čerpadlem. **Červená** část kapaliny je vytlačována z výstupní části čerpadla. Jakmile se hadička začne zotavovat, **zelená** část média je nasávána do čerpadla [7].

Krok 2

Jak se čerpadlo otáčí, stále tlačí **modrou** oblast tekutiny. **Červená** část média je již zcela mimo čerpadlo. **Zelený** úsek je stále nasáván do čerpadla [7].

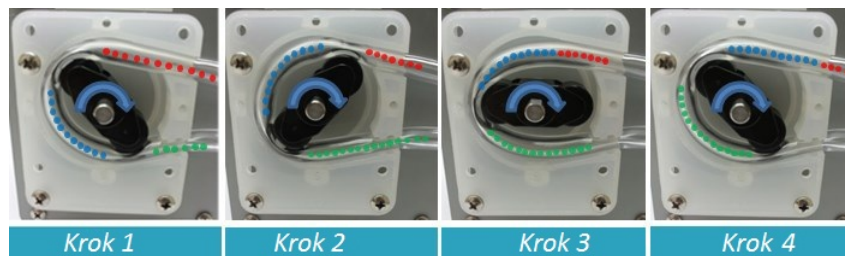
Krok 3

Jak se rotor čerpadla otáčí stále vytlačuje **modré** a i **červené** médium. **Zelený** úsek je stále nasáván do čerpadla [7].

Krok 4

Modré a **červené** médium je vytlačováno ven z čerpadla. **Zelená** část je již uzavřena z obou stran [7].

Tento sací a výtlačný cyklus se neustále opakuje. Výsledkem opakování tohoto cyklu je kontinuální přeprava kapaliny za minimálních tlakových rázů.



Obrázek 2.1.: Princip rotačního peristaltického čerpadla [7]

2.1.1. Řízení

Malá rotační peristaltická čerpadla jsou nejčastěji poháněna stejnosměrnými motory. Jedna z možností, jak lze řídit průtok čerpadla, je změnou otáček. V rovnici 2.1, která je odvozena z náhradního schématu stejnosměrného stroje, jsou zobrazeny parametry které ovlivňují otáčky.

- **Změna odporu R** - Do obvodu kotvy připojíme odpor. Tento způsob řízení není příliš běžný z důvodu zvýšení Jouleových ztrát.
- **Změna magnetického toku Φ** - U motorů které mají budící vinutí, lze změnou budícího proudu změnit magnetický tok.
- **Změna napětí U_d** - Tento způsob řízení je nejpoužívanější z důvodu malých ztrát a plynulé regulace.

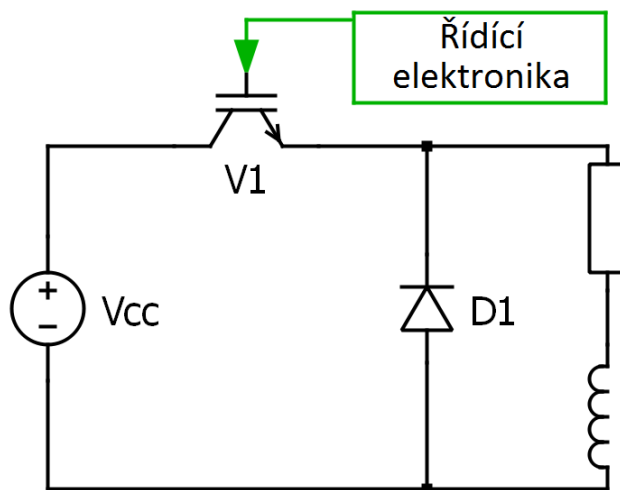
$$\omega = \frac{U_d}{k \cdot \Phi} - \frac{R \cdot M}{(k \cdot \Phi)^2} \quad (2.1)$$

Pulzní měnič - Schéma na obrázku 2.2 zobrazuje principiální fungování snižujícího měniče napětí. Je-li tranzistor $V1$ otevřený, protéká přes něj proud, který prochází přes zátěž zpět do zdroje. Je-li uzavřen, proud přes něj neprochází a antiparalelní dioda $D1$ uzavírá obvod, energie akumulovaná na indukčnosti se tak může vybit.

Střední hodnotu napětí na zátěži lze spočítat z následujícího vztahu:

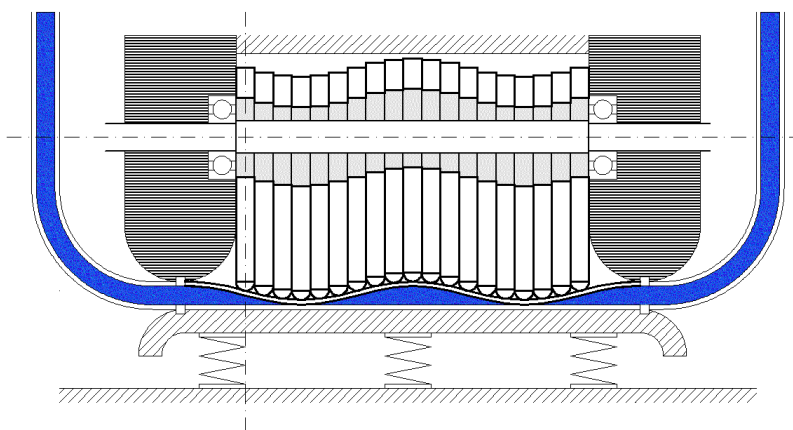
$$U_z = \frac{T_1}{T} \cdot U = z \cdot U \quad (2.2)$$

kde U_z je střední hodnota napětí na zátěži, U je napětí zdroje, T_1 je doba sepnutí, T je celková doba a z je poměrná doba sepnutí ($0 \leq z \leq 1$). Spínací frekvence ovlivňuje zvlnění výstupního proudu, se zvyšující se frekvencí se zvlnění snižuje. mikroprocesoru



Obrázek 2.2.: Schéma pulzního měniče

2.2. Lineární peristaltická čerpadla



Obrázek 2.3.: Schéma lineárního peristaltického čerpadla [9]

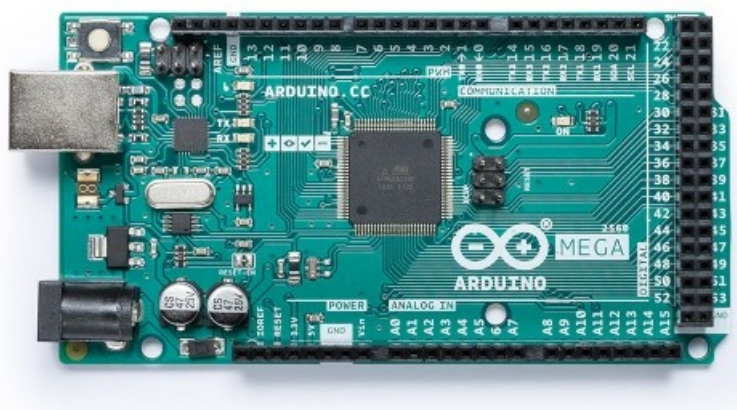
Lineární peristaltická čerpadla dopravují tekutinu pomocí pružné hadice s použitím vlnění. V lineárním peristaltickém čerpadle pracovní elementy rytmicky stlačují konkrétní úsek pružné trubky, tím vyvolávají pohyb kapaliny. Na rozdíl od rotačních peristaltických čerpadel lineární peristaltická čerpadla obvykle nepoužívají válce nebo kluzné kontaktní prvky. Možné řešení lineárního čerpadla je na obrázku 2.3 [9].

2.2.1. Řízení

Řízení je realizováno pomocí způsobu a rychlosti spínání, výběr způsobu spínání záleží na počtu dostupných aktuátoru. Aktuátor je spínán pomocí stejnosměrného spínače, principiálně se jedná o snižující pulzní měnič, schéma zapojení je stejné (obrázek 2.2). Podrobněji jsou spínací cykly popsány v kapitole 4.4. Stejnosměrné spínače lze ovládat přímo pomocí mikroprocesoru nebo pomocí nějaké vývojové platformy jako je Arduino.

3. Arduino

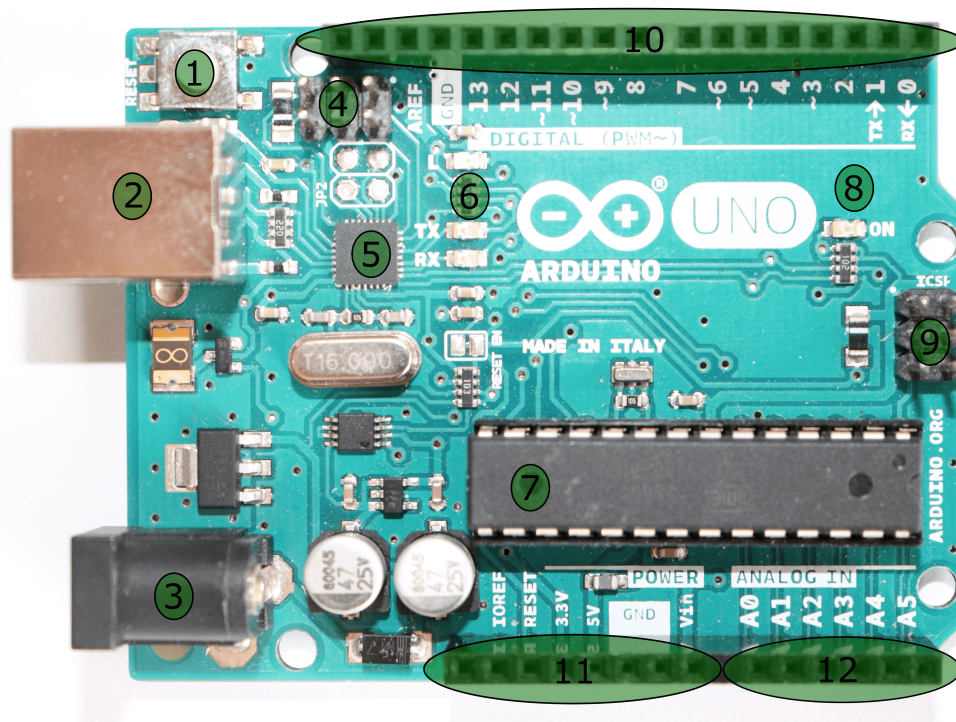
Arduino původně vzniklo pro studenty jako levnější a jednodušší náhrada k drahým vývojovým platformám jako byl například BASIC Stamp. Jedná se o Open Source projekt. To znamená, že veškerý software i hardware je volně dostupný. Existuje mnoho vývojových desek, mezi nejpoužívanější se řadí Arduino UNO pro jednodušší projekty a Arduino Mega 2560 (obrázek 3.1) pro výkonově náročnější projekty. [12]



Obrázek 3.1.: Arduino MEGA 2560 [14]

3.1. Arduino Uno

Arduino UNO patří k nejpoužívanějším vývojovým deskám v současnosti především díky nízké ceně, dostatečnému výkonu pro většinu aplikací a malým rozměrům. Další výhodou je mnoho prodávaných rozšiřujících shieldů pro tento typ Arduina. Arduino UNO je vývojová deska založena na mikroprocesoru ATmega328. Arduino UNO má 14 digitálních I/O pinů a 6 analogových vstupů. [10]



Obrázek 3.2.: Arduino UNO

1) Reset

Jedná se o bílý mikrospínač, který po zmáčknutí vyšle impuls do mikroprocesoru a dojde ke spuštění programu opět od začátku. [12]

2) Konektor USB

Můžeme se setkat s konektorem USB typu B nebo micro USB konektorem u novějších desek z důvodu úspory místa. Slouží ke komunikaci s PC a k programování mikroprocesoru. [12]

3) Napájecí konektor

Napájení lze řešit několika způsoby, pokud není Arduino napájeno z USB konektoru lze použít vestavěný lineární stabilizátor napětí. Vstupní napětí lze přivést přes napájecí konektor desky anebo přes pin Vin, tato možnost bývá využita za předpokladu že navrhované zařízení bude napájeno z baterie. Poslední možností napájení je připojení externího 5V stabilizátoru napětí přes piny GND a 5V.

Doporučené vstupní napětí je v rozmezí od 7 V do 12 V. Absolutní limit napětí pro napájení desky je od 6 V do 20 V. Napětí pod 7 V může způsobit kolísání napětí. Napětí nad 12 V způsobí jeho přehřátí z důvodu nedostatečného chlazení. [13]

4) ICSP konektor

Slouží k externímu programování USB-serial převodníku. U modelů, které mají převodník implementovaný v hlavním procesoru tento konektor není. Většina uživatelů nepotřebuje tento konektor. [12]

5) USB-serial převodník

Jedná se o mikroprocesor ATMEGA16U2, který se stará o komunikaci hlavního mikroprocesoru s PC. [12]

6) Indikační diody L, Rx a Tx

LED diody Rx a Tx slouží ke kontrole uživateli, že probíhá sériová komunikace. Dioda označená L je připojena k pinu 13, s ní lze zkontrolovat funkčnost arduina bez dalších HW úprav. [12]

7) Procesor

V Arduinu UNO je použit mikroprocesor ATmega328 navržený firmou Atmel z řady megaAVR. Tento procesor má 32 kB paměti, z které 0,5 kB zabírá Bootloader, slouží k programování Arduina a spuštění programu. K mikroprocesoru je připojen 16MHz krystal. Tento typ mikrokontroléru zvládá až 20 milionů instukcí za sekundu. [12] [14]

8) Indikační Led dioda ON

Indikuje napájení desky [12].

9) ISCP konektor

Slouží k externímu programování mikroprocesoru, používají ho i některé shieldy. [12]

10) Digitální piny

Arduino UNO obsahuje celkem 14 digitálních pinů, lze je použít jako vstupní nebo výstupní. Arduino UNO má celkem 6 pinů které lze použít k PWM (pulzně šířková modulance). Tyto piny mohou dodávat proud o hodnotě 20mA, pokud dojde k navýšení proudu nad hodnotu 40mA, dojde k poškození vývojové desky. [12]

11) Piny

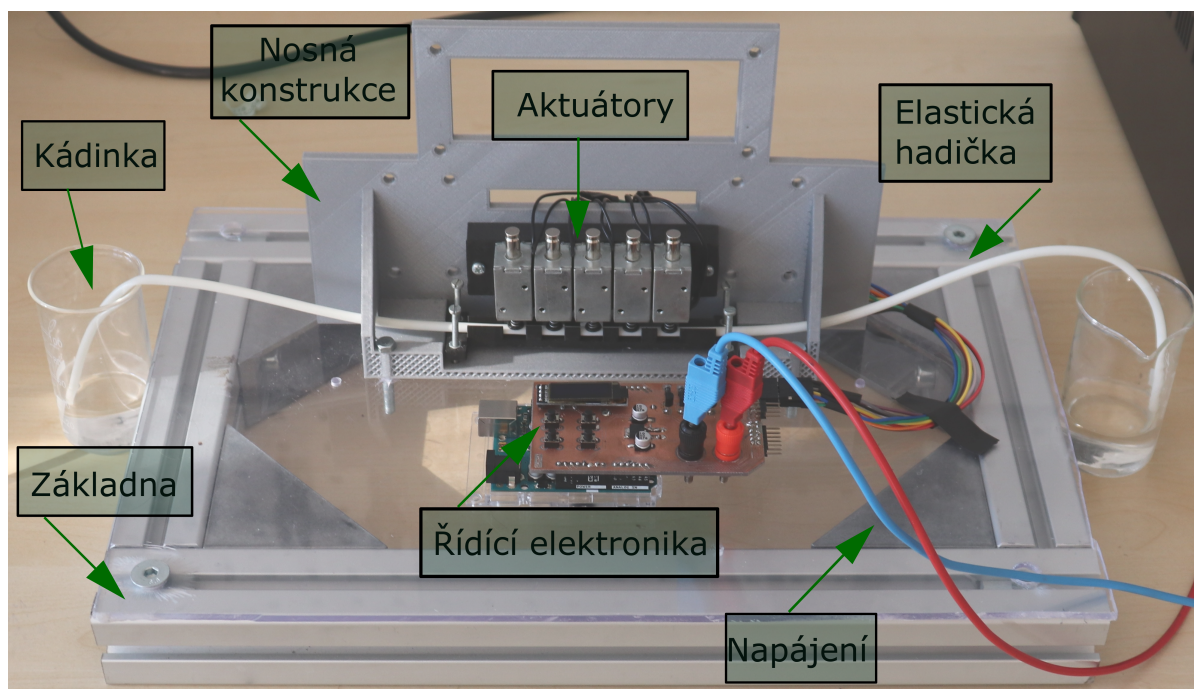
Tyto piny jsou převážně napájecí (5V 3,3V GND) a pomocné (RESET IOREF) [12]

12) Analogové vstupy

Na desce Arduino UNO nalezneme 6 analogových pinů sloužících ke čtení výstupů analogových součástí. [12]

4. Realizace čerpadla

Měření průběhu napětí na elektromagnetickém aktuátoru (žlutý průběh



Obrázek 4.1.: Fyzický model lineárního peristaltického čerpadla

V současnosti je ve zdravotnictví problém s přesným dávkováním účinné léčivé látky na konkrétní místo. Například při chemoterapii se toxické látky vpravují do celého organismu místo přímo do zasažených míst od nádorových bujení. Z tohoto důvodu byly zkoumány různé typy čerpadel. V poslední fázi by konečným produktem mělo být miniaturní čerpadlo, které se po naplnění léčivem laparoskopicky voperuje na inkriminované místo, kde by postupně uvolňovalo danou léčivou látku.

V prvním kroku vývoje, kterým se zabývá tato bakalářská práce, byl vytvořen model lineárního peristaltického čerpadla, viz obrázek 4.1. Tento model byl vytvořen za účelem praktického odzkoušení funkčnosti peristaltického čerpadla. Reálný model se skládá z několika částí:

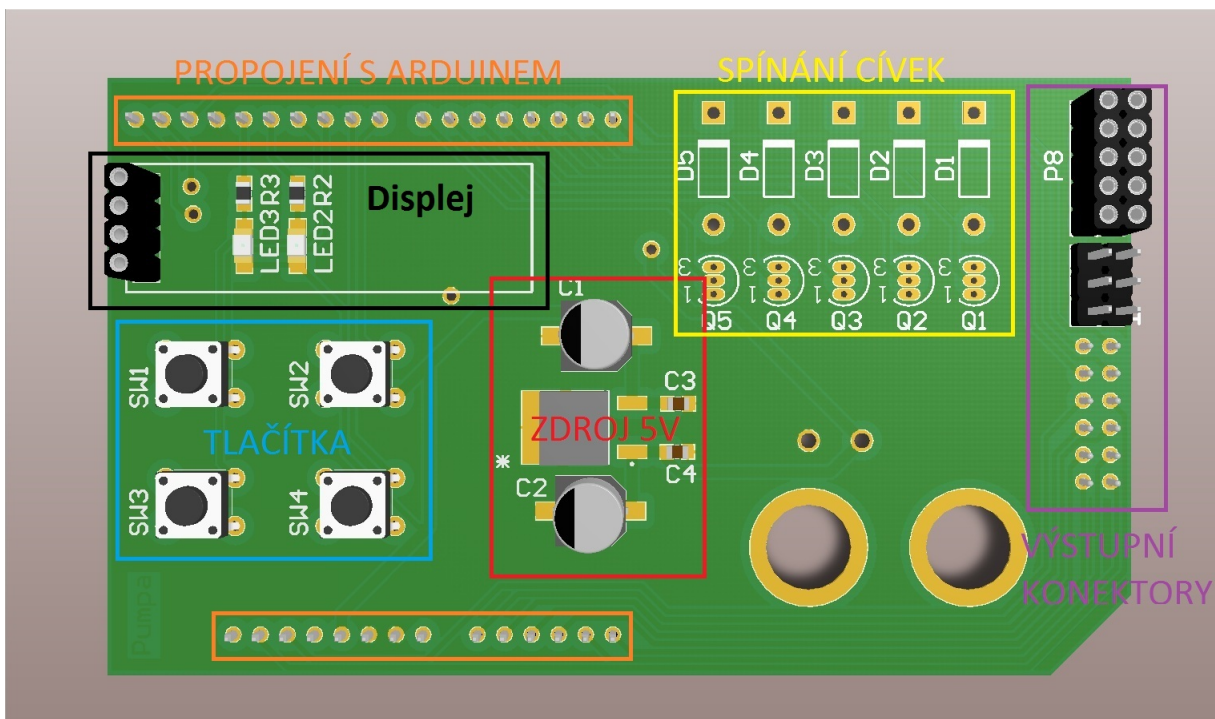
- **Základna** - Z hliníkových profilů a "L" spojek byla vytvořena obdélníková konstrukce, na kterou bylo připevněno plexisklo. Tato konstrukce základny byla použita kvůli zvýšení stability čerpadla.
- **Nosná konstrukce** - Hlavní konstrukční části čerpadla byly vytištěny z plastu pomocí 3D tisku. Na této konstrukci jsou připevněny aktuátory a žlab pro vedení hadičky.

- **Aktuátory** - O čerpání se stará 5 elektromagnetických aktuátorů, které při sepnutí vtahují své jádro a tím dojde k přitlaku hadičky.
- **Hadička** - V této aplikaci nebyl problémem chemické kompatibility čerpaného média se stěnou hadičky, protože čerpané medium je voda. Problémem se ukázalo nalezení dostatečně měkké hadičky tak, aby ji byly elektromagnetické aktuátory schopné stlačit.
- **Řídící elektronika** - Jako řídicí obvod bylo použito vývojové platformy Arduino UNO, k němuž byla připojena navržená deska plošného spoje, starající se o spínání cívek a uživatelský interface.

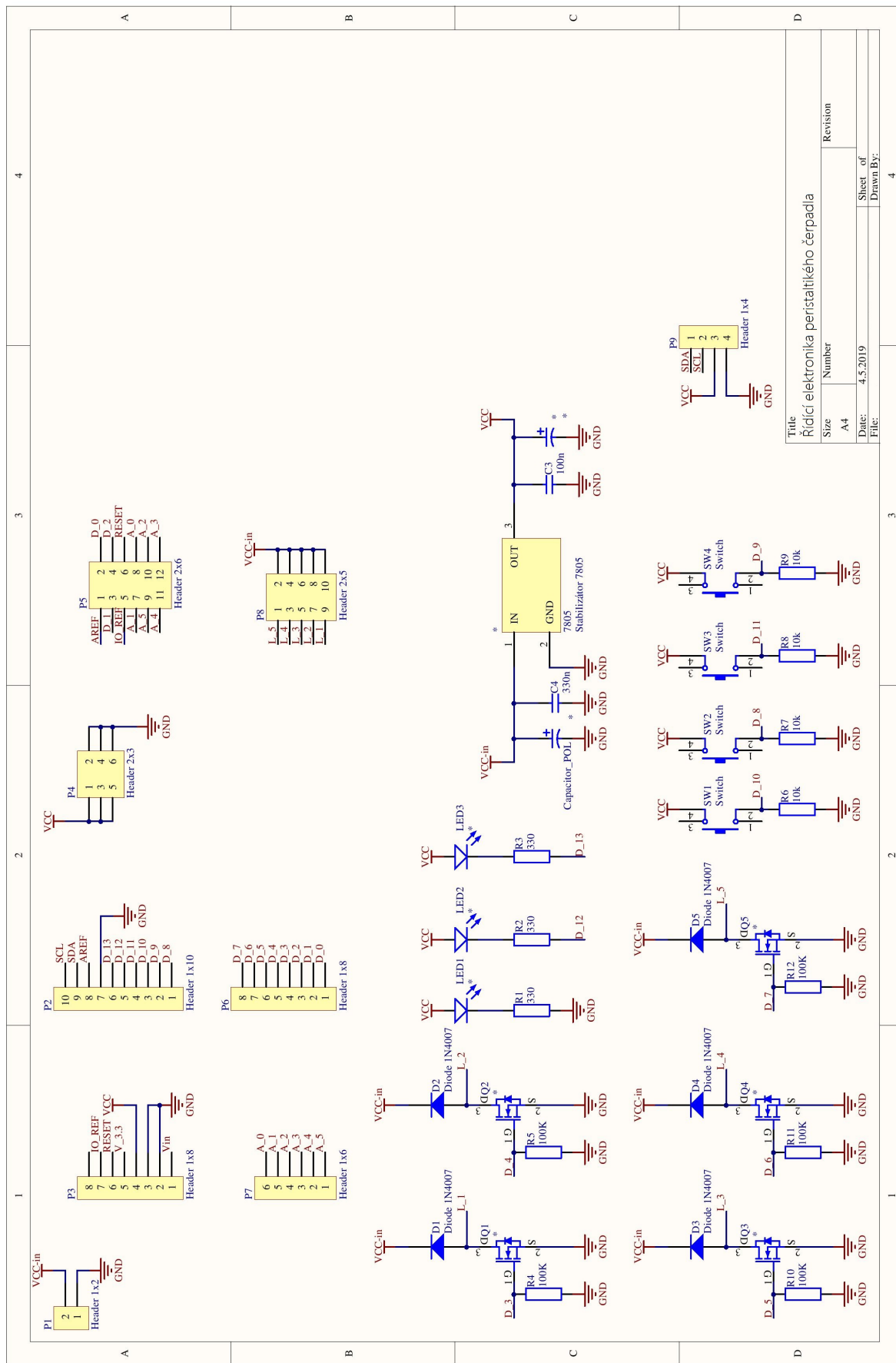
4.1. Návrh tištěného spoje

Deska plošného spoje (zkratka DPS) byla navržena v programu Altium Designer. Tento software je určen k návrhu jednovrstvých, dvouvrstvých i vícevrstvých plošných spojů. Na internetu jsou rozsáhlé knihovny s pouzdry mnoha součástek. Tento software má funkci Wizard, která byla využita i při návrhu této desky. Tato funkce pomáhá s návrhem vlastního pouzdra součástky. Mezi výhody lze zařadit velmi intuitivní ovládání.

Na obrázku 4.2 je zobrazen počítačový model navrženého tištěného spoje z programu Altium Designer. Schéma navržené řídicí desky je na obrázku 4.3. V následujících bodech této kapitoly jsou popsány jednotlivé části podrobněji.



Obrázek 4.2.: 3D model DPS v prostředí Altium designer



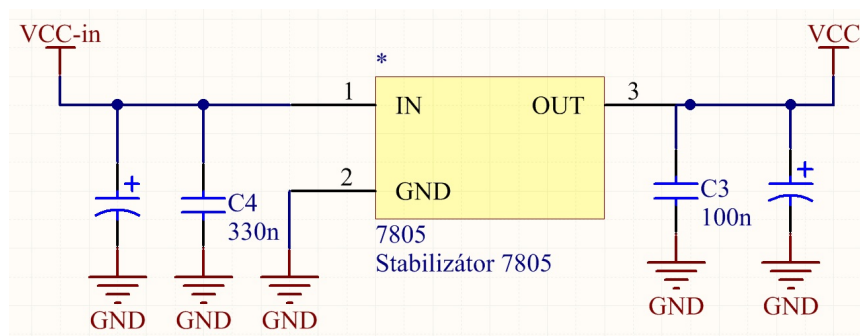
Obrázek 4.3.: Schéma řídicí elektroniky peristaltického čerpadla

4.1.1. Zdroj 5 V

Vzhledem k tomu, že napájecí napětí cívek přesahuje maximální napětí pro integrovaný stabilizátor napětí na desce Arduino UNO, tak tento stabilizátor nemohl být použit.

Pro navržený napěťový zdroj byl zvolen stabilizátor napětí 7805 v pouzdru DPAK o výstupním napětí 5 V a maximálním vstupním napětí 32 V. Maximální výstupní proud je 1 A. Schéma navrženého zdroj napětí je na obrázku 4.4.

Dle datasheetu jsou ke stabilizátoru připojeny 2 vyhlazovací elektrolytické kondenzátory o kapacitě 10 μF a 2 keramické kondenzátory o kapacitě 100 nF a 330 nF.



Obrázek 4.4.: Schéma stabilizátoru 5 V

4.1.2. OLED displej

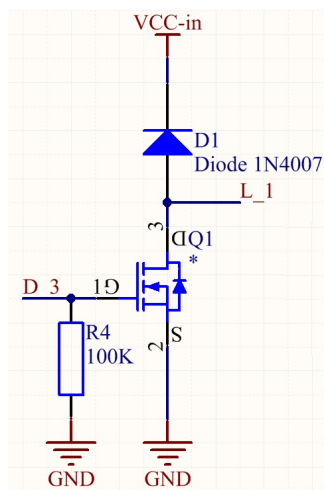
Jako zobrazovací jednotka byl vybrán OLED (z anglicky **O**rganic **L**ight **E**mitting **D**iode) displej, který s Arduinem UNO komunikuje na sběrnici I²C (jedná se o sériovou sběrnici), což je výhodné kvůli počtu potřebných vodičů. Displej je připojen k Arduino pomocí dvou datových a dvou napájecích vodičů.

U tohoto typu displeje svítí jednotlivé pixely, proto není nutno podsvěcovat celý displej. Druhá výhoda je nižší spotřeba oproti klasickým LCD displejům. Použitý displej má rozlišení 128 x 32 pixelů a velikost 2,3 cm.

4.1.3. Spínání cívek

Schéma zapojení je znázorněno na obrázku 4.5. Ke spínání cívek byly použity unipolární MOSFET tranzistory BS170 v pouzdru TO92. Na Gate je přiveden signál z pinu Arduina. Po přivedení logické 1 dojde k sepnutí tranzistoru, potažmo cívky. Mezi Gate a zemí je připojen rezistor o hodnotě 100 k Ω . Tento odpor slouží k definování logické úrovně. Source je připojeno přímo k zemi.

Aktuátor je připojen mezi vstupní napětí a Drain. K zátěži je antiparalelně připojena dioda SB160. Tato dioda slouží k omezení napěťových špiček vznikajících na induktivní zátěži použitého aktuátoru.

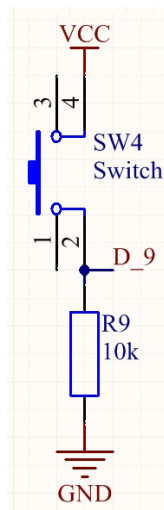


Obrázek 4.5.: Schéma tranzistorového spínače

4.1.4. Tlačítka

Pro ovládání čerpadla uživatelem jsou na tištěném spoji 4 mechanické mikrospínače. Schéma zapojení je uvedeno na obrázku 4.6. Odpor připojený mezi zem a tlačítko o hodnotě 10 k Ω slouží k určení logické hodnoty, říká se mu pull-down rezistor.

Za předpokladu, že bychom nepoužili tento rezistor, tak elektromagnetické rušení v okolí vodiče způsobí, že naměřená hodnota je dostatečně velká tak, aby ji Arduino vyhodnotilo jako logickou 1.

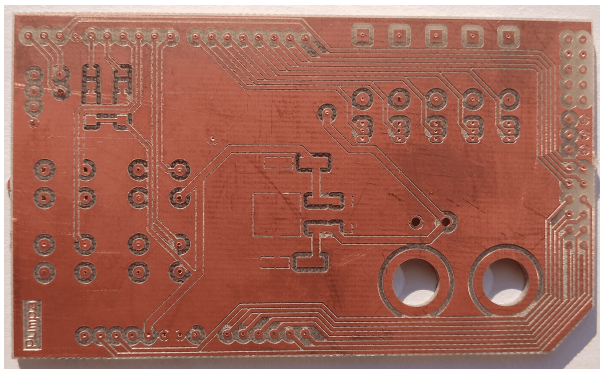


Obrázek 4.6.: Schéma tranzistorového spínače

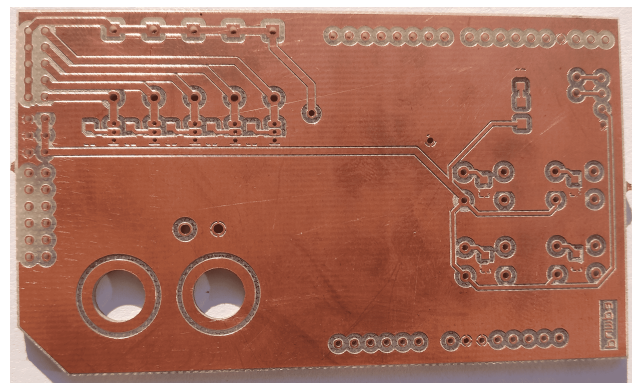
4.1.5. Svorkovnice

Na obrázku 4.2 můžeme vidět nad blokem displej a pod tlačítka konektory, ty slouží k propojení s Arduinem UNO. Vpravo od bloku spínání cívek jsou výstupní konektory. Horní konektor slouží k připojení 5 cívek, na celou spodní řadu konektoru je připojeno vstupní napětí. K horním pinům jsou připojeny jednotlivé tranzistory. Na prostředním konektoru jsou 3 páry napájecích pinů. V horní řadě je připojeno +5V a vedle GND (zem). Ke spodnímu konektoru jsou vyvedeny nepoužité piny z Arduina.

4.2. Výroba DPS



(a) Vrchní strana



(b) Spodní strana

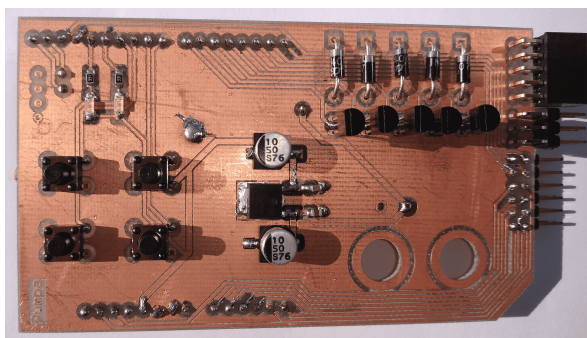
Obrázek 4.7.: Vyfrézovaná DPS

Navržený tištěný spoj byl vyroben z důvodů větší přesnosti pomocí CNC frézy. Z důvodu zvolené technologie výroby nebylo provedeno chemické, a poté ani galvanické pokovení vyvrtaných otvorů.

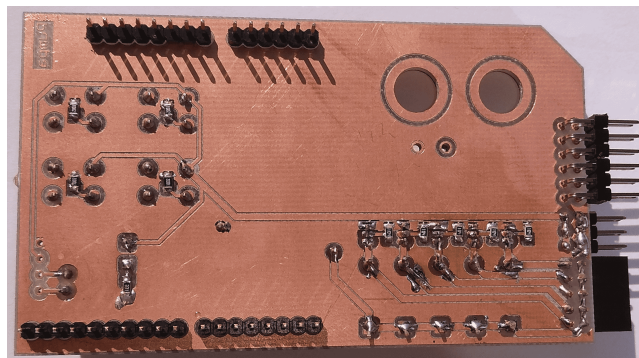
Po vyfrézování tištěného byl spoj zkontrolován, zda byl vyroben správně, poté byl zčištěn. Na obrázku 4.7 je zobrazena deska plošného spoje po očištění.

Postup realizace:

- 1) Do vyvrtaných otvorů byly zapájeny vodiče, tím vznikly potřebné vodivé spoje mezi vrchní a spodní vrstvou.
- 2) Osazení SMD součástky z obou stran.
- 3) Osazení THT součástky z obou stran.
- 4) V posledním kroku osazení napájecími konektory a OLED displej.



(a) Vrchní strana



(b) Spodní strana

Obrázek 4.8.: Osazená DPS

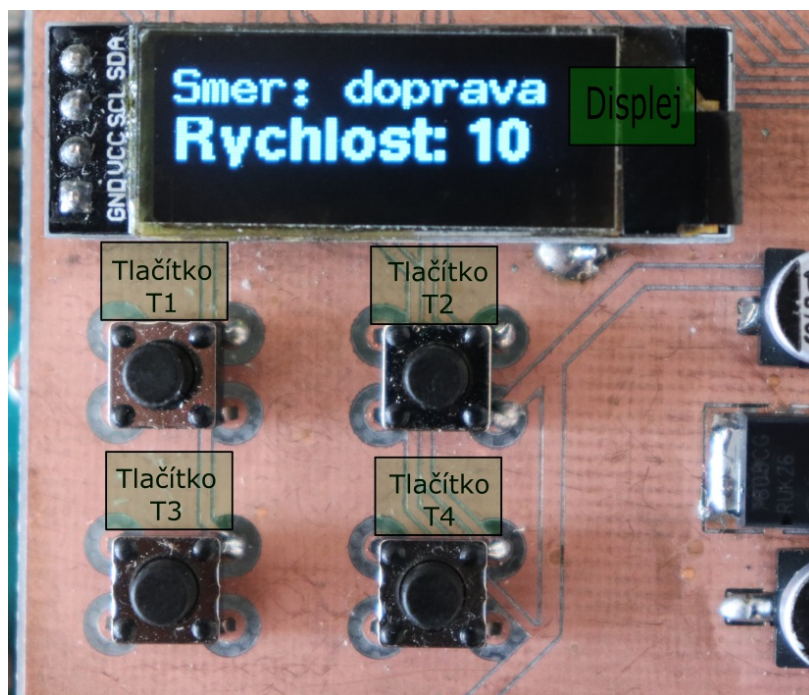
Plně osazená deska použitými součástkami, které jsou v tabulce 4.1, je zobrazena na obrázku 4.8.

| Název součástky | Hodnota | Typ pouzdra | počet |
|----------------------------|----------------|-------------|-------|
| Stabilizátor napětí | 7805 | DDPAK | 1x |
| Keramický kondenzátor | 100 nF | 805 | 1x |
| Keramický kondenzátor | 330 nF | 805 | 1x |
| Elektrolytický kondenzátor | 10 uF | E | 2x |
| MOSFET tranzistor | BS170 | TO92 | 5x |
| Schottkyho dioda | SB160 | DO41 | 5x |
| Mikrospínač | TC-0104-T | - | 4x |
| Odpor | 330 Ω | 805 | 3x |
| Odpor | 10 k Ω | 805 | 4x |
| Odpor | 100 k Ω | 805 | 5x |
| LED dioda | zelená | 1206 | 3x |
| Konektory | - | - | - |

Tabulka 4.1.: Použité součástky

Arduino UNO bylo naprogramováno ve vývojovém prostředí Arduino IDE. Celý program je v příloze této bakalářské práce. Pro OLED displej byla použita knihovna U8glib.h [15].

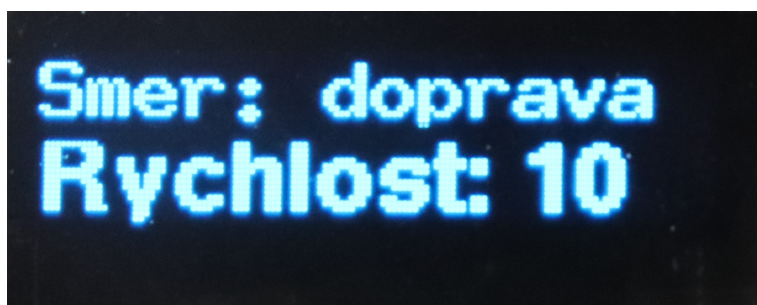
4.3. Uživatelské rozhraní



Obrázek 4.9.: Uživatelské rozhraní

Na obrázku 4.9 je zobrazeno uživatelské rozhraní. Po připojení napájení se zobrazí první menu (obrázek 4.10). V tomto menu pomocí tlačítka T2 zvyšujeme rychlost, tlačítkem T4 snižujeme rychlost a pomocí tlačítka T1 spouštíme čerpání, pokud je čerpadlo v chodu, tak jej lze jen vypnout. Tlačítkem T3 můžeme měnit směr čerpání nebo pokud jej podržíme přibližně na dvě sekundy, se dostat do druhého menu. V tomto menu nastavujeme cyklus spínání :

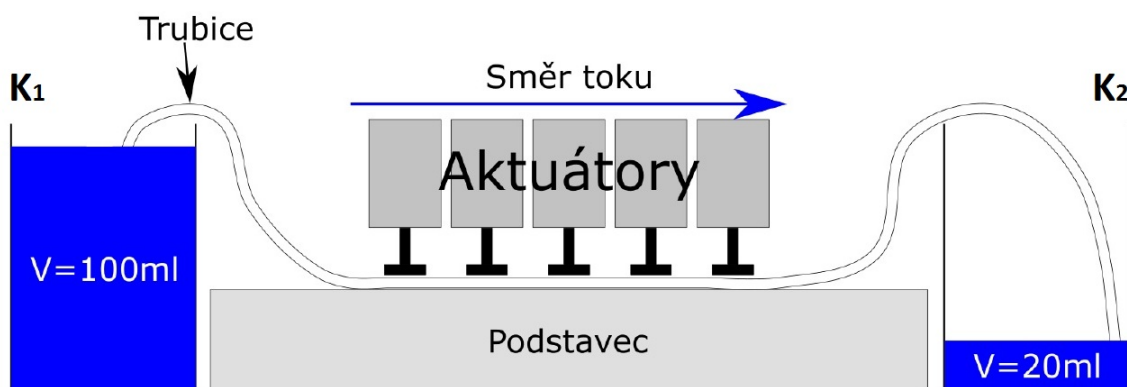
- Spínání 3 aktuátorů.
- Spínání 2 aktuátorů.
- Spínání 1 aktuátorů.
- Spínání ob jeden aktuátor.



Obrázek 4.10.: Zobrazení na displeji

4.4. Měření průtoku média

Na obrázku 4.11 je zobrazeno principiální schéma a počáteční podmínky měření. V kádince K_1 je objem kapaliny $V \doteq 100$ ml a v kádince K_2 je objem $V \doteq 20$ ml. Trubička musí být plně zaplavena, z důvodu neschopnosti čerpadla nasát čerpané médium.



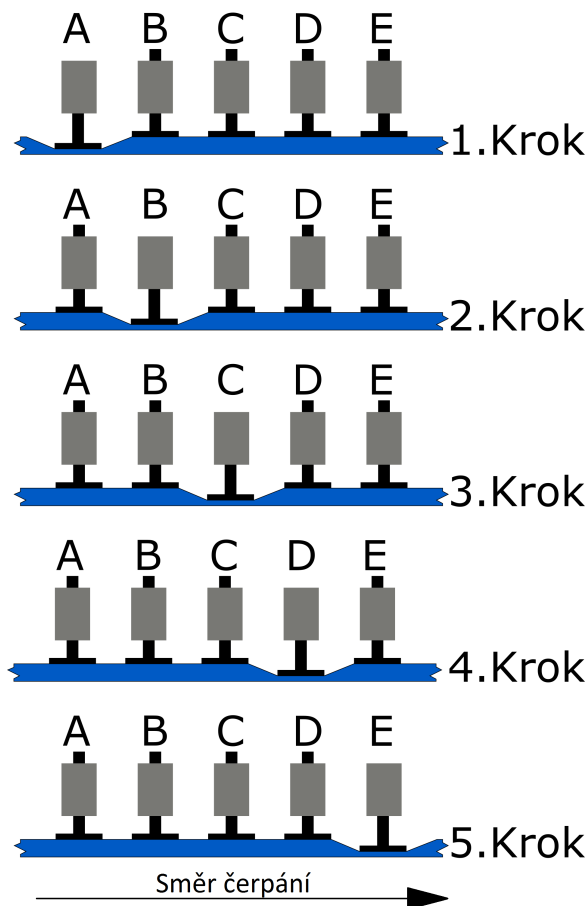
Obrázek 4.11.: Schéma peristaltického čerpadla [6]

Postup měření:

1. Nastavíme spínací cyklus a rychlost spínání.
2. Nastavení počátečního stavu.
3. Vložíme do kádinky K_2 indikační plovák pro následnou počítačovou analýzu.
4. Zapneme nahrávání na kameru.
5. Spustíme čerpání.
6. Videozáznam analyzujeme pomocí softwaru Tracker.

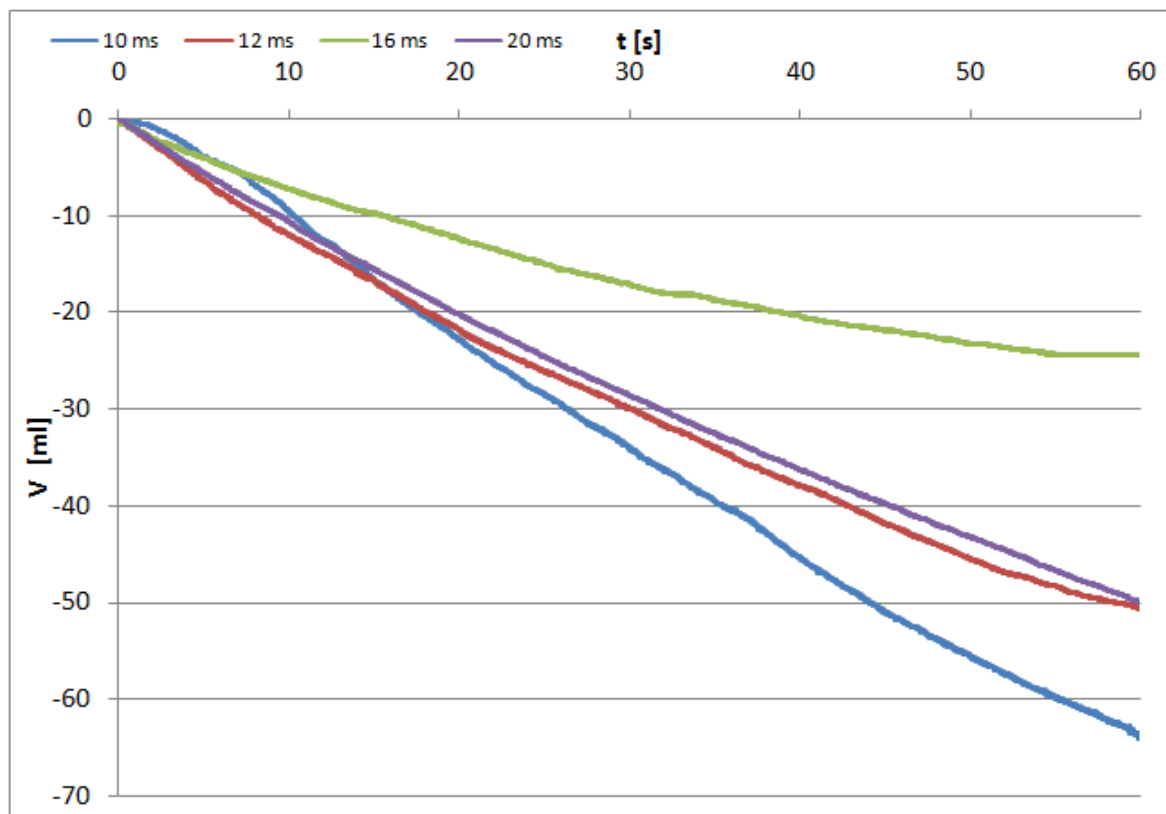
Měření bylo provedeno pro čtyři spínací cykly. První měření bylo pro spínací cyklus se spínáním 1 aktuátorem, tento cyklus byl měřen pro rychlosti 10, 12, 16 a 20 ms. U druhého měření se spínaly 2 aktuátory pro časy 12, 16, 20 a 25 ms. Ve třetím měření byly spínány 3 aktuátory, zde byly změřeny jen dva průběhy, pro 25 a 50 ms. Čtvrtý spínací cyklus "ob jeden", zde bylo provedeno jedno měření pro průběh 25 ms. Spínací cykly jsou popsány na následujících stranách.

4.4.1. Spínací cyklus 1 aktuátor



Obrázek 4.12.: Cyklus spínání

Na obrázku 4.12 je zobrazen spínací diagram pro stav, kdy je spínána jen jedna cívka z pěti. V 1. kroku je aktuátor **A** sepnut a aktuátory **B**, **C**, **D**, **E** jsou rozepnuty. V přechodu na 2. krok je cívka **A** rozepínána a cívka **B** spínána. Cívka **B** je téměř okamžitě sepnuta (přibližně 2 ms) a cívice **A** trvá okolo 8 ms, než-li se energie vybijí a cívka se rozepne. Tento časový rozdíl způsobuje, že cívka **A** působí jako zábrana pro vytlačovanou kapalinu cívkou **B** a kapalina je vytlačena ve směru čerpání. Po rozepnutí cívky **A** dojde k nasátí čerpané kapaliny. Tento děj se opakuje pro další kroky. Celý tento cyklus je realizován ve smyčce pěti po sobě jdoucích kroků, které se opakují.

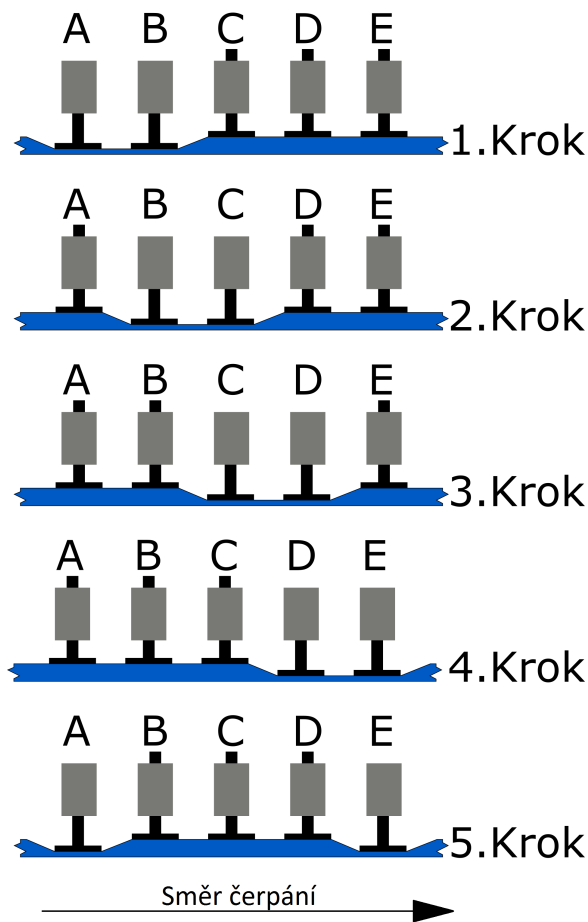


Obrázek 4.13.: Průběh průtoku při 1 aktuátoru

Měření

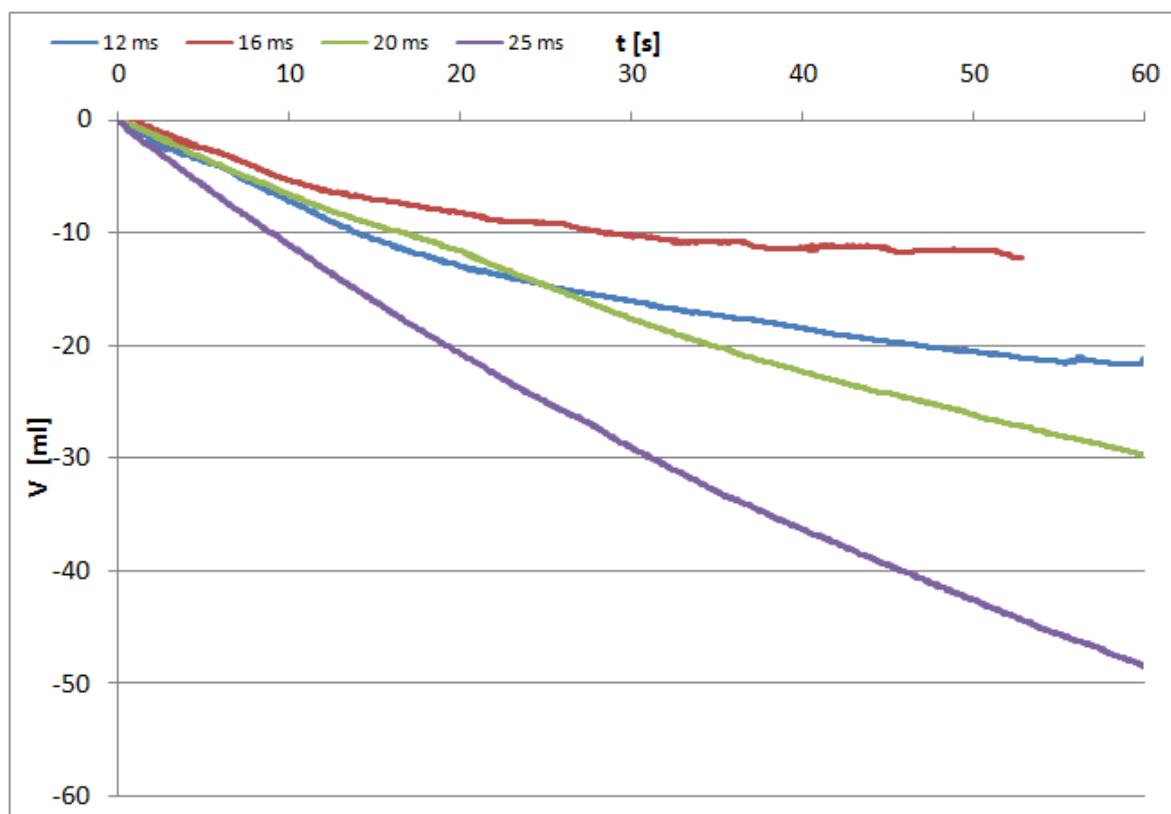
Graf 4.13 zobrazuje časový průběh změny objemu kapaliny v měřené kádince pro spínací cyklus "1 aktuátor". Měření proběhlo pro rychlosti 10, 12, 16 a 20 ms. Spínací rychlosti 12 a 20 ms mají téměř identické přečerpané množství kapaliny, zhruba 50 ml. Průběh s 10 ms byl nejrychlejší, za 1 min přečerpal 62 ml. Nejméně efektivní byla spínací rychlost 16 ms, nedocházelo k čerpání média, dokonce docházelo k brzdění vyrovnání hladiny.

4.4.2. Spínací cyklus 2 aktuátory



Obrázek 4.14.: Cyklus spínání

Na obrázku 4.14 je zobrazen spínací diagram pro stav, kdy jsou spínány dvě cívky současně. V 1. kroku jsou aktuátory **A**, **B** sepnuty a aktuátory **C**, **D**, **E** jsou rozepnuty. V přechodu na 2. krok je cívka **A** rozepínána a cívka **C** spínána. Sepnutím cívky **C** dojde k vytlačení kapaliny ve směru čerpání. Rozepnutím cívky **A** dojde k nasátí v témže směru. Tento děj se opakuje pro další kroky. Celý tento cyklus je realizován ve smyčce pěti po sobě jdoucích kroků, které se opakují.

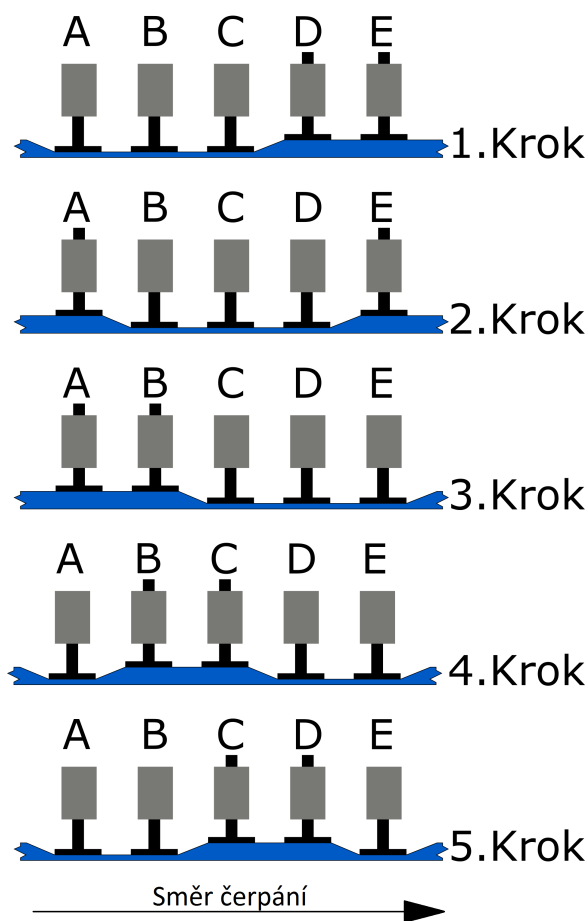


Obrázek 4.15.: Průběh průtoku při 2 aktuátorech

Měření

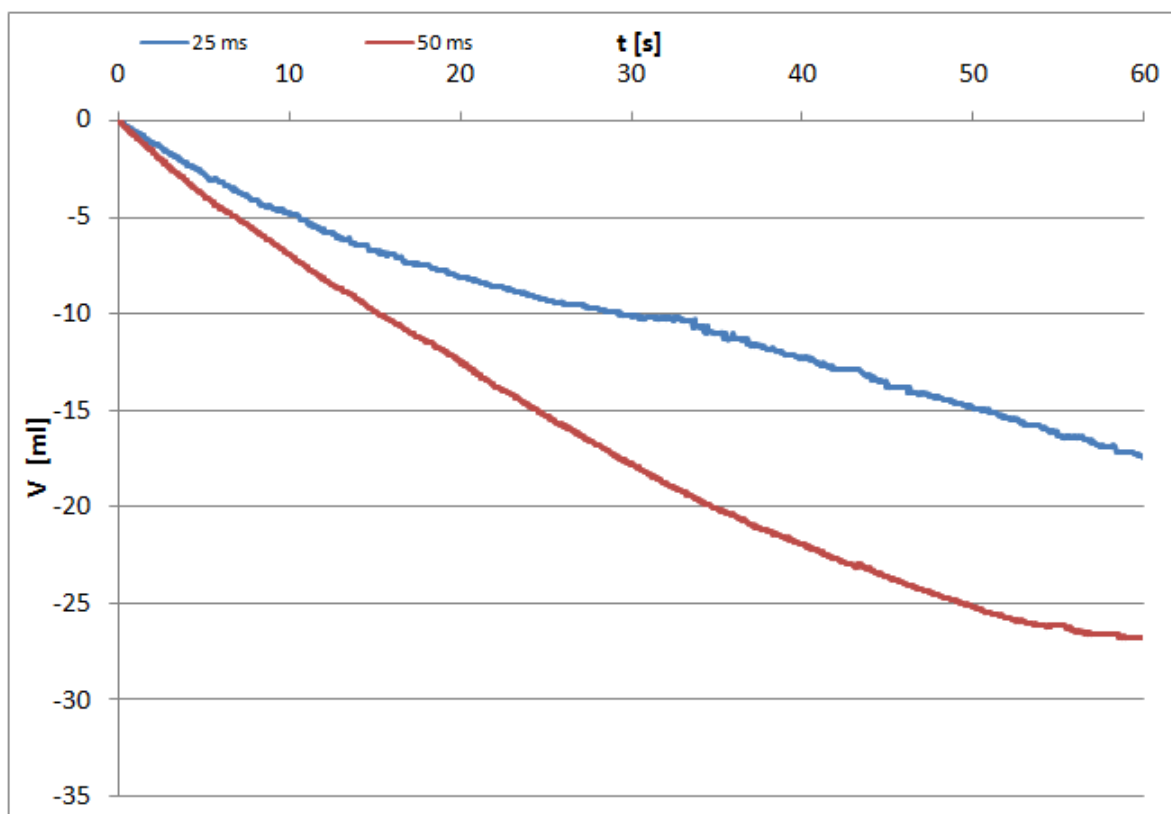
Graf 4.15 zobrazuje časový průběh změny objemu kapaliny v měřené kádince pro spínací cyklus "2 aktuátory". Měření proběhlo pro rychlosti 12, 16, 20 a 25 ms. Pro rychlosti menší než 12 ms zůstávají již cívky permanentně sepnuté. Lze říci, že pro rychlosti 12, 16 a 20 ms čerpadlo nefunguje dle očekávání, protože nečerpá. Se spínací rychlostí 25 ms čerpadlo odčerpá za 1 min přibližně 50 ml.

4.4.3. Spínací cyklus 3 aktuátory



Obrázek 4.16.: Cyklus spínání

Na obrázku 4.16 je zobrazen spínací diagram pro stav, kdy jsou spínány tři cívky současně. V 1. kroku jsou aktuátory **A**, **B**, **C** sepnuty a aktuátory **D**, **E** jsou roze-pnuty. V přechodu na 2. krok je cívka **A** rozepínána a cívka **D** spínána. Sepnutím cívky **D** dojde k vytlačení kapaliny ve směru čerpání. Rozepnutím cívky **A** dojde k nasátí v témže směru. Tento děj se opakuje pro další kroky. Celý tento cyklus je realizován ve smyčce pěti po sobě jdoucích kroků, které se opakují.

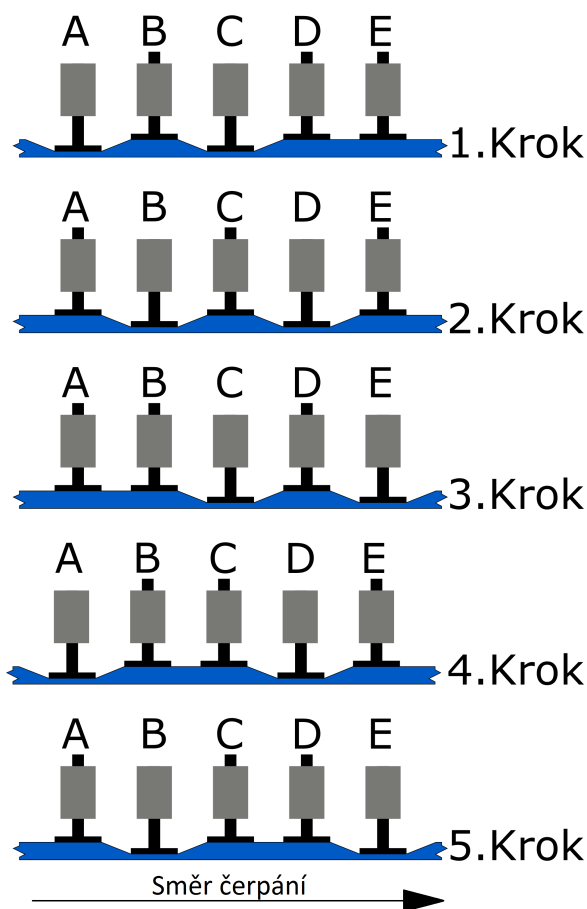


Obrázek 4.17.: Průběh průtoku při 3 aktuátorech

Měření

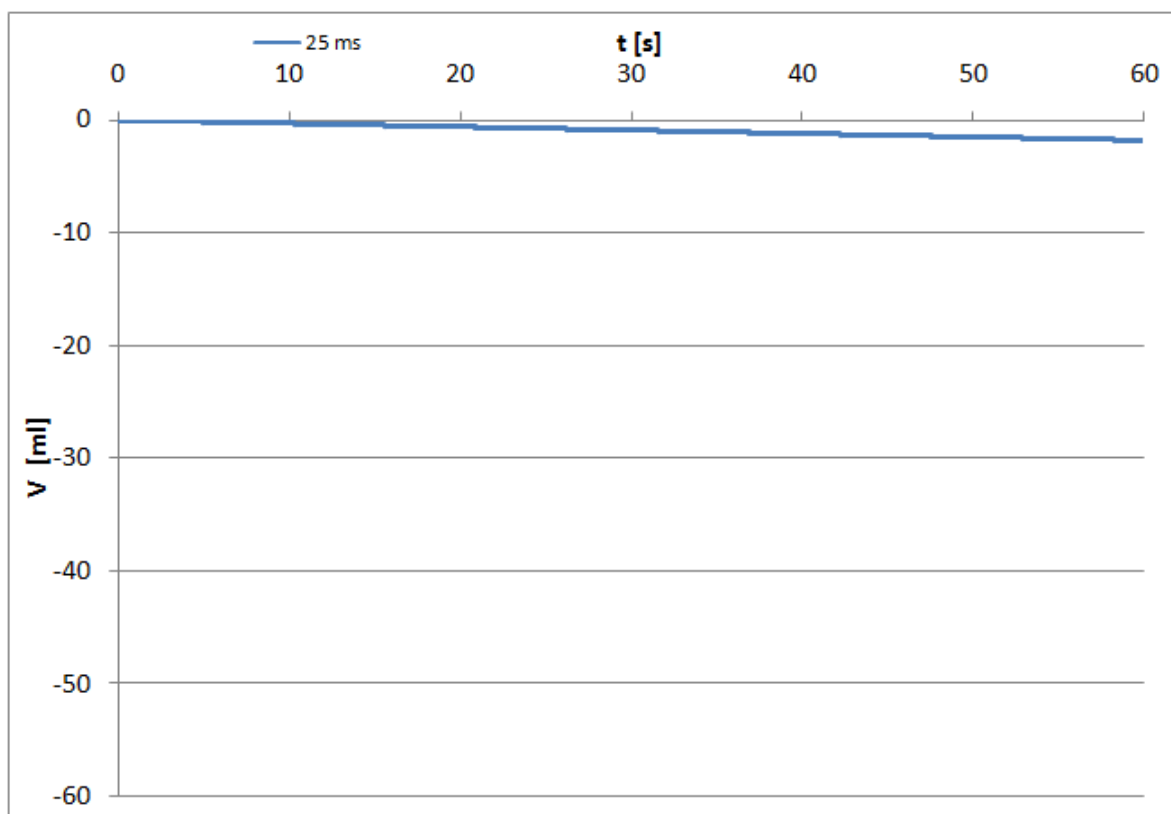
Graf 4.17 zobrazuje časový průběh změny objemu kapaliny v měřené kádince pro spínací cyklus "3 aktuátory". Měření proběhlo jen pro rychlosti 25 a 50 ms. Pro rychlosti menší než 25 ms zůstávají již cívky permanentně sepnuté. Vyšší rychlosti než 50 ms nemají smysl, čerpadlo nefunguje.

4.4.4. Spínací cyklus ob jeden aktuátor



Obrázek 4.18.: Cyklus spínání

Na obrázku 4.18 je zobrazen spínací diagram pro stav, kdy jsou spínány dvě cívky, které mají mezi sebou cívku, která není sepnutá. V 1. kroku jsou aktuátory **A**, **C** sepnuty a aktuátory **B**, **D**, **E** nesepnuty. V přechodu na 2. krok jsou cívky **A**, **C** rozepínané a cívky **B**, **D** spínané. Po sepnutí aktuátoru **B** jsou aktuátory **A**, **C** stále sepnuté, mezi těmito dvěma cívkami se zvýší hydraulický tlak. Sepnutím cívky **D** dojde k vytlačení kapaliny ve směru čerpání. Rozepnutím cívky **A** dojde k výtlaku čerpané kapaliny proti směru čerpání. Tento děj se opakuje pro další kroky. Celý tento cyklus je realizován ve smyčce pěti po sobě jdoucích kroků, které se opakují.



Obrázek 4.19.: Průběh průtoku při 2 aktuátorech s mezerou

Měření

Graf 4.17 zobrazuje časový průběh změny objemu kapaliny v měřené kádince pro spínací cyklus "ob jeden aktuátor". Měření proběhlo jen pro jednu rychlost (25 ms) z důvodu nefunkčnosti spínacího cyklu, nedochází totiž k vytlačení kapaliny.

4.5. Simulace průběhu napětí na elektromagnetickém aktuátoru

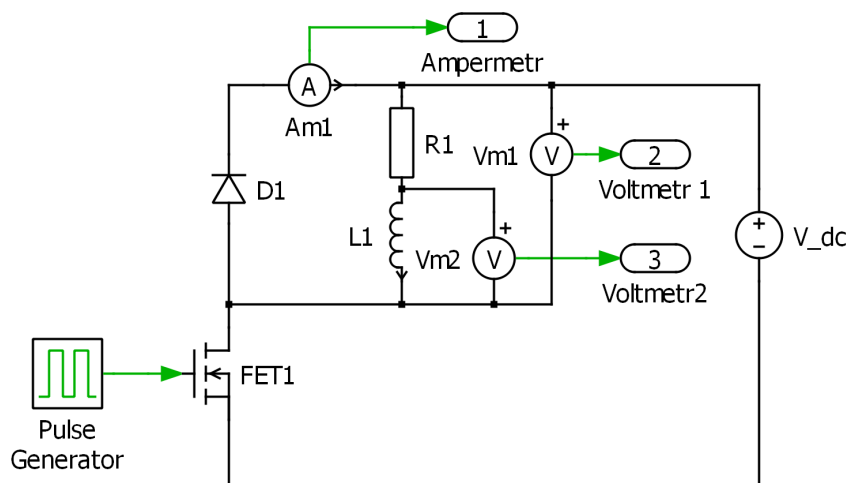
Ke stlačování pružné hadičky je použit lineární solenoid. Na obrázku 4.20 je zobrazen průběh napětí, když je obvod rozepínán a nemá ochranou antiparalelní diodu. Cívka byla napájena 18 V. Napětí po odpojení cívky bylo -24 V a přechodný děj trval 1,2 ms. Parametry tohoto solenoidu jsou:

- Odpor - $R = 60 \Omega$. Byl zjištěn z datasheetu.
- Počet závitů - $n = 2160$ závitů. Byl zjištěn z datasheetu.
- Indukčnost - $L = 50$ mH. Byla změřena RLC metrem.



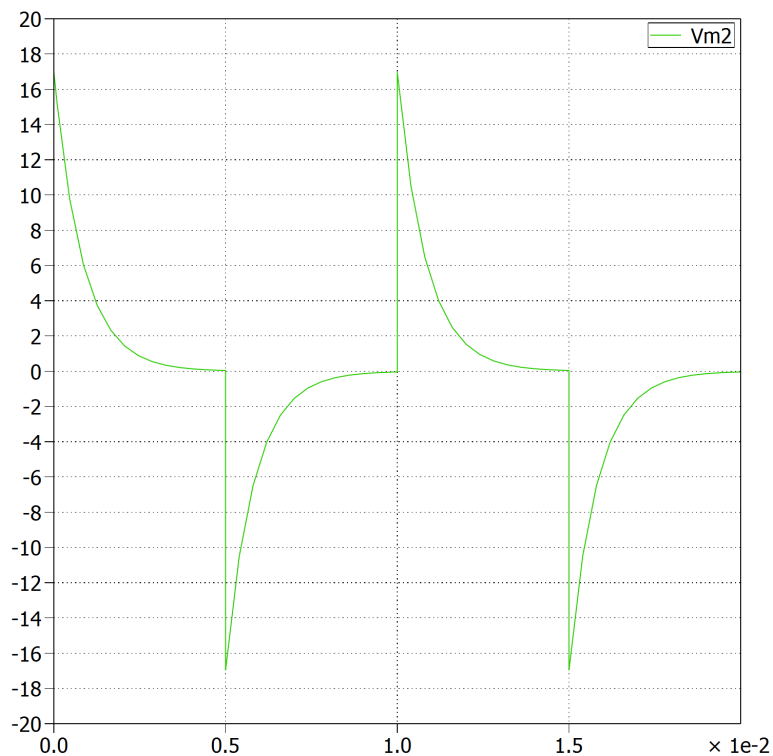
Obrázek 4.20.: Průběh napětí na solenoidu

4.5.1. Simulace



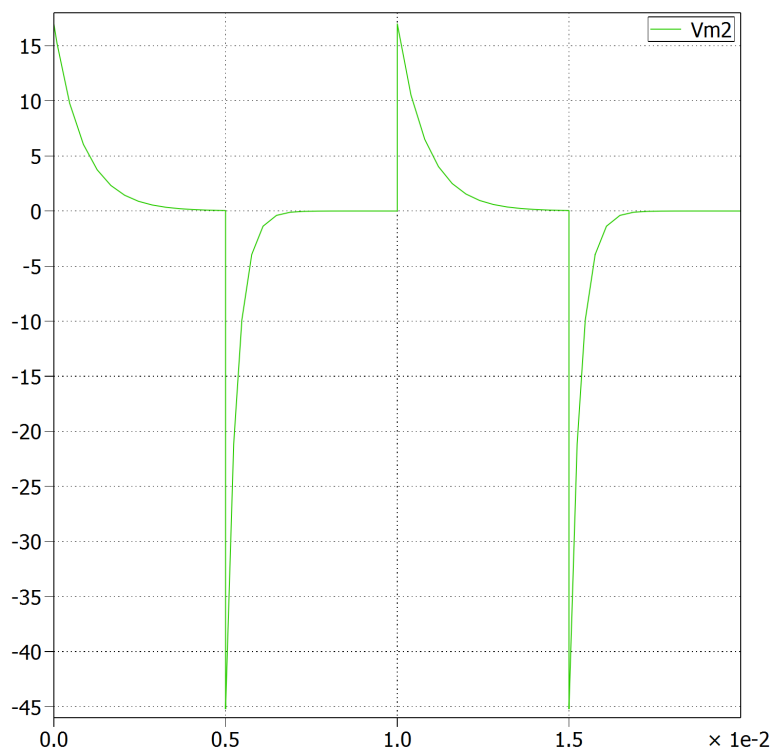
Obrázek 4.21.: Schéma obvodu pro simulaci průběhu napětí

Na obrázku 4.21 je základní schéma pro simulaci průběhu napětí na aktuátoru. Napájecí napětí bylo zvoleno 17 V, spínací frekvence 100 Hz. Simulace byla provedena v prostředí Simulink s nastavbou PLEX. Na následujícím obrázku je zobrazen průběh napětí. Ze simulace vyplývá, že vybití aktuátoru je potřeba minimálně 5 ms.



Obrázek 4.22.: Průběh napětí na aktuátoru

Ke zefektivnění vybíjení aktuátoru by mohlo vést připojení odporu v sérii s diodou. Nejvhodnější parametry měl vybíjecí odpor o velikosti 100Ω . Tato simulace je zobrazena na obrázku 4.23. Dle simulace se aktuátor vybije za 2 ms.



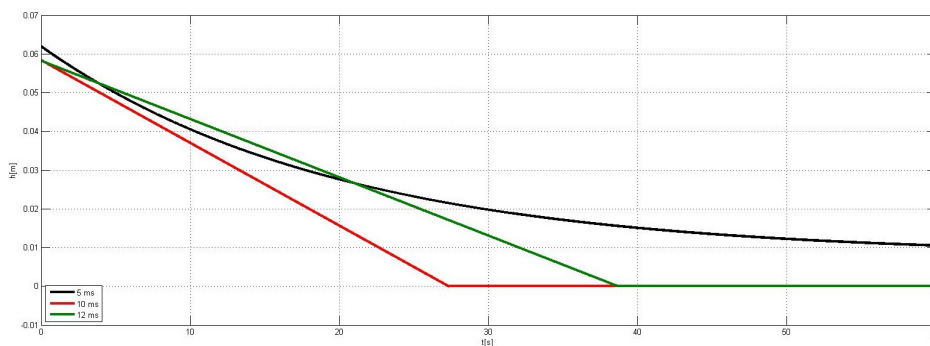
Obrázek 4.23.: Průběh napětí na aktuátoru s vybíjecím odporem

4.6. Zhodnocení

V prvním prototypu čerpadla byly aktuátory spínány pomocí relé, z tohoto důvodu byla energie cívky vybita výbojem mezi kontakty. V další fázi byla relátka nahrazena MOSFET tranzistory, ty měli mít výhodu lepších spínacích vlastností oproti relé. Pro tuto variantu však nastal problém s vybíjením energie v indukčnosti. Díky antiparalelně zapojené Schotkydo diodě dojde po rozepnutí MOSFET tranzistoru k rychlému uzavření proudové dráhy, bohužel ale vybíjení energie při spínací frekvenci 10 ms je delší než tento interval, což má za následek trvalé sepnutí daného aktuátoru. K rychlejšímu vybíjení indukčnosti by mohlo pomoci připojení dodatečného vybíjecího odporu v sérii s diodou, tato varianta má však za následek zvýšení napětí, z tohoto důvodu je potřeba dostatečně dimenzovat spínací MOSFET tranzistor.

Tato problematika má za následek nižší průtoky oproti spínání relé. V grafu 4.24 jsou znázorněny tři průběhy, při cyklu z kapitoly 4.4.2. Hodnoty byly získány z měření pana Jiřího Mušky [6]. Při spínání aktuátorů pomocí cívek jsme se dostali na rychlost průtoku 200 ml/min, to je přibližně čtyři krát více, než-li u spínání pomocí tranzistorů.

Velkou nevýhodou relé je jejich mechanické namáhání při těchto spínacích frekvencích.



Obrázek 4.24.: Průběh průtoku při 2 aktuátorech spínané pomocí relé

5. Závěr

Tato bakalářská práce se zabývá řízením lineárního peristaltického čerpadla. Práce je rozdělena do dvou hlavních částí.

V první části jsou popsány možnosti čerpání kapalin a řízení peristaltických čerpadel malých rozměrů. Druhá část se zabývá vytvořením řídicí elektroniky lineárního peristaltického čerpadla stávajícího se z pěti elektromagnetických aktuátorů.

V prvním kroku byl sestaven fyzický model čerpadla, zjištěny napěťové a proudové poměry pro daný elektromagnetický aktuátor, z důvodu výběru vhodných polovodičových spínačů. Následně byla v softwaru pro tvorbu tištěných spojů navržena řídicí elektronika pro daný typ čerpadla.

Po vytvoření řídicí elektroniky a úspěšně provedeném oživení proběhly kontrolní měření průběhu elektrických veličin pomocí osciloskopu.

Posledním stěžejním měřením bylo měření rychlosti průtoku kapaliny. Z důvodu delší vybíjecí doby elektromagnetických aktuátorů, došlo při spínacích cyklech a frekvencích, které byly ověřeny dřívější realizací, kdy aktuátory byly spínané pomocí relé, k trvalému sepnutí aktuátorů a tudíž k zastavení průtoku. Tento stav je nežádoucí proto bylo v této práci navrženo zlepšení v podobě přidání vybíjecího odporu do série s diodou. Dle simulace, která byla z tohoto důvodu provedena je navrhovaný odpor $R = 100 \Omega$, přidáním tohoto odporu se sníží doba potřebná k vybití o 8 ms.

Největší rychlost průtoku čerpadla, které bylo spínáno tranzistorem je pro spínací cyklus "1 aktuátor" s nastavenou rychlostí spínání 10 ms, takto nastavené čerpadlo má rychlost průtoku 63 ml/min. Spínací cyklus "ob jeden" je navržen tak, že nemá čerpat. Pokud by jsem zrychlili vybíjení tak spínací cyklus "3 aktuátory" by mohl fungovat dobře a rychle.

Nejrychlejší průtok má ale spínací cyklus "2 aktuátory", který je spínán pomocí relé s časem 10 ms, takto spínané čerpadlo dosahuje rychlosti průtoku 200 ml/min.

Literatura

- [1] BLÁHA, Jaroslav a BRADA, Karel *Hydraulické stroje*. 1. vyd. Praha: SNTL, 1992. 747 s. Technický průvodce; Sv. 70. Česká matice technická; Roč. 1992, č. 436. ISBN 80-03-00665-1.
- [2] Pump - Wikipedia. [online]. 2019 [cit. 2019-06-01]. Dostupné z: <https://en.wikipedia.org/wiki/Pump>
- [3] Archimedes Screw [online]. 2019 [cit. 2019-06-01]. Dostupné z: <https://pinshape.com/items/40542-3d-printed-archimedes-screw>
- [4] ČSN 11 0000 *Rozdělení a terminologie čerpadel* Praha: Český normalizační institut, 1994.
- [5] JANALÍK, Jaroslav *HYDRODYNAMIKA A HYDRODYNAMICKÉ STROJE* [online]. 2019 [cit. 2019-06-02]. Dostupné z: <https://old.vscht.cz/uchi/ped/hydroteplo/materialy/skripta.Janalik.pdf>
- [6] MUŠKA, Jiří *Lineární peristaltické čerpadlo*. Plzeň, 2019. Bakalářská práce. Západočeská univerzita v Plzni. Fakulta elektrotechnická. Vedoucí práce Prof. Ing. Pavel Karban, Ph.D.
- [7] GRAYLINE LLC *Peristaltic Pump Tubing Material Selection Whitepaper* [online]. 2019 [cit. 2019-05-26]. Dostupné z: <https://www.graylineinc.com/whitepapers/peristaltic-pump-tubing.html>
- [8] Linear Peristaltic Pump. In: Li D. (eds) *Encyclopedia of Microfluidics and Nanofluidics*. Springer, Boston, MA
- [9] Peristaltic pump - Wikipedia. [online]. 2019 [cit. 2019-06-01]. Dostupné z: https://en.wikipedia.org/wiki/Peristaltic_pump
- [10] *Arduino UNO* [online]. 2019 [cit. 2019-05-14]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardUno>
- [11] *Arduino MEGA 2560* [online]. 2019 [cit. 2019-06-10]. Dostupné z: <https://store.arduino.cc/mega-2560-r3>
- [12] VODA, Zbyšek, *programujeme Arduino* [online]. 2019 [cit. 2019-05-14]. Dostupné z: <http://arduino.cz/programujeme-arduino/>
- [13] *Arduino Power, Current, and Voltage Limitations* [online]. 2019 [cit. 2019-05-14]. Dostupné z: <http://robotics.lib-ionimoub.gr/?p=715>
- [14] *ATmega328* [online]. 2019 [cit. 2019-05-14]. Dostupné z: <https://cs.wikipedia.org/wiki/ATmega328>

- [15] Knihovna pro OLED displej. Dostupné z: <https://navody.arduino-shop.cz/navody-k-produktum/oled-i2c-displej-128x32.html>

A. Program

```
1 #include "U8glib.h"
2 U8GLIB_SSD1306_128X32 u8g(U8G_I2C_OPT_NONE);
3
4 int TL[4] = { 10 , 8, 11, 9};
5
6 const int T1 = 3;
7 int T[5] = { T1 , T1 + 1, T1 + 2, T1 + 3, T1 + 4 };
8
9 bool D[5] = { true, true, true, false, false};
10
11 const int D1 = 13;
12 const int D2 = 12;
13
14 bool start = false;
15 bool smer1 = true;
16 bool krokovani = false;
17
18 int smer2 = 0;
19 int t = 10;
20 int cerp = 1;
21 int menu = 0;
22
23 int delkaStisku = 0;
24
25 String zprava = "Smer:_doprava";
26 String zprava1 = "Rychlost:_";
27
28 void setup() {
29   for (int i = 0; i < 4; i++) {
30     pinMode(TL[i], INPUT);
31   }
32
33   for (int i = 0; i < 5; i++) {
34     pinMode(T[i], OUTPUT);
35   }
36
37   pinMode(D1, OUTPUT);
38   pinMode(D2, OUTPUT);
39 }
40
41 void loop() {
```

```
42  if (start == true)
43  {
44      switch (cerp) {
45          case 1:
46              pump();
47              break;
48          case 2:
49              pump();
50              break;
51          case 3:
52              pump3();
53              break;
54          case 4:
55              pump2();
56              break;
57          default:
58              zprava = "restartuj";
59              zprava1 = "Chyba";
60              break;
61      }
62
63  } else {
64      if (menu == 0) {
65          menu1();
66      } else {
67          menu2();
68      }
69      u8g.firstPage();
70      do {
71          draw(zprava, zprava1);
72      } while ( u8g.nextPage() );
73  }
74 }
75
76 void menu1() {
77     TL1();
78
79     if ((digitalRead(TL[1])) == HIGH && (t < 50))
80     {
81         t++;
82         delay(200);
83     }
84
85     if ((digitalRead(TL[3])) == HIGH && (t > 0))
86     {
87         t--;
88         delay(200);
89     }
```

```
90
91  if ((digitalRead(TL[2])) == HIGH)
92  {
93      delkaStisku ++;
94      if (delkaStisku == 4)
95      {
96          menu = 1;
97          delkaStisku = 0;
98      }
99
100     smer1 = !smer1;
101     fncSmer();
102     delay(200);
103 } else {
104     delkaStisku = 0;
105 }
106
107 zprava1 = "Rychlost:␣";
108 zprava1 += t;
109 }
110
111 void menu2() {
112     if ((digitalRead(TL[1])) == HIGH && (cerp < 6))
113     {
114         cerp++;
115         delay(200);
116     }
117
118     if ((digitalRead(TL[3])) == HIGH && (cerp > 0))
119     {
120         cerp--;
121         delay(200);
122     }
123     if ((digitalRead(TL[2])) == HIGH && (cerp == 5))
124     {
125         krokovani = !krokovani;
126         delay(200);
127     }
128
129     if (cerp == 0)
130     {
131         cerp = 5;
132     }
133     if (cerp == 6)
134     {
135         cerp = 1;
136     }
137
```

```
138  if ((digitalRead(TL[0])) == HIGH)
139  {
140      menu = 0;
141      fncSmer();
142  } else {
143      zprava = "Zpusob_cerpani";
144      switch (cerp) {
145          case 1:
146              zprava1 = "3_aktuator";
147              D[0] = true;
148              D[1] = true;
149              D[2] = true;
150              D[3] = false;
151              D[4] = false;
152              break;
153          case 2:
154              zprava1 = "2_aktuator";
155              D[0] = false;
156              D[1] = false;
157              D[2] = false;
158              D[3] = true;
159              D[4] = true;
160              break;
161          case 3:
162              zprava1 = "objeden";
163              D[0] = true;
164              D[1] = false;
165              D[2] = true;
166              D[3] = false;
167              D[4] = false;
168              break;
169          case 4:
170              zprava1 = "1_aktuator";
171              D[0] = true;
172              D[1] = false;
173              D[2] = false;
174              D[3] = false;
175              D[4] = false;
176
177              break;
178          case 5:
179              zprava = "Delka_kroku_1s";
180              if (krokovani) {
181                  zprava1 = "Ano";
182                  t = 1000;
183              } else {
184                  zprava1 = "Ne";
185                  t = 10;
```

```
186     }
187     break;
188     default:
189         zprava1 = "Chyba";
190         break;
191 }
192 }
193 }
194
195 void pump() {
196     for (int j = 0; j < 5; j++) {
197         for (int i = 0; i < 5; i++) {
198             if (smer2 == 1)
199                 {
200                     digitalWrite(T[4 - i], D[i]);
201                 } else {
202                     digitalWrite(T[i], D[i]);
203                 }
204             }
205
206             if (j < 2)
207                 {
208                     D[0 + j] = !D[0 + j];
209                     D[3 + j] = !D[3 + j];
210                 } else {
211                     D[0 + j] = !D[0 + j];
212                     D[j - 2] = !D[j - 2];
213                 }
214             delay(t);
215         }
216     TL1();
217 }
218
219 void pump2() {
220     for (int j = 0; j < 5; j++) {
221         for (int i = 0; i < 5; i++) {
222             if (smer2 == 1)
223                 {
224                     digitalWrite(T[4 - i], D[i]);
225                 } else {
226                     digitalWrite(T[i], D[i]);
227                 }
228             }
229
230             if (j < 4)
231                 {
232                     D[0 + j] = !D[0 + j];
233                     D[1 + j] = !D[1 + j];
```

```
234     } else {
235         D[0 + j] = !D[0 + j];
236         D[j - 4] = !D[j - 4];
237     }
238     delay(t);
239 }
240 TL1();
241 }
242
243 void pump3() {
244     for (int j = 0; j < 5; j++) {
245         for (int i = 0; i < 5; i++) {
246             if (smer2 == 1)
247                 {
248                     digitalWrite(T[4 - i], D[i]);
249                 } else {
250                     digitalWrite(T[i], D[i]);
251                 }
252             }
253
254     if (j<2){
255         D[0 + j] = !D[0 + j];
256         D[1 + j] = !D[1 + j];
257         D[2 + j] = !D[2 + j];
258         D[3 + j] = !D[3 + j];
259     }else if (j==2){
260         D[2] =! D[2];
261         D[3] =! D[3];
262         D[4] =! D[4];
263         D[0] =! D[0];
264     }else if (j==3){
265         D[3] =! D[3];
266         D[4] =! D[4];
267         D[0] =! D[0];
268         D[1] =! D[1];
269     }else if (j==4){
270         D[4] =! D[4];
271         D[0] =! D[0];
272         D[1] =! D[1];
273         D[2] =! D[2];
274     }
275
276     delay(t);
277 }
278 TL1();
279 }
280
281 void draw(String text, String text1) {
```



```
282 u8g.setFont(u8g_font_9x15B);
283 u8g.setPrintPos(0, 12);
284 u8g.println(text);
285
286 u8g.setFont(u8g_font_helvB14);
287 u8g.setPrintPos(0, 30);
288
289 int var = 0;
290 u8g.print(text1);
291 }
292
293 void fncSmer() {
294   if ((smer1 == true))
295     {
296       zprava = "Smer:_doleva";
297       smer2 = 1;
298     } else {
299       zprava = "Smer:_doprava";
300       smer2 = 0;
301     }
302 }
303
304 void TL1() {
305   if ((digitalRead(TL[0])) == HIGH)
306     {
307       delay(60);
308       if ((digitalRead(TL[0])) == HIGH)
309         {
310           start = !start;
311         }
312       if (start == false)
313         {
314           for (int i = 0; i < 5; i++) {
315             digitalWrite(T[i], false);
316           }
317         }
318       delay(200);
319     }
320 }
```