

Automatický míchač nápojů

Aleš Horváth

Bakalářská práce
2020



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

Ústav informatiky a umělé inteligence

Akademický rok: 2019/2020

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: **Aleš Horváth**
Osobní číslo: **A17122**
Studijní program: **B3902 Inženýrská informatika**
Studijní obor: **Softwarové inženýrství**
Forma studia: **Prezenční**
Téma práce: **Automatický míchač nápojů**
Téma práce anglicky: **An Automatic Drink Maker**

Zásady pro vypracování

1. Popište existující konstrukce zařízení na automatické míchání nápojů.
2. Proveďte návrh vlastního zařízení na automatické míchání nápojů.
3. Návrh hardwarově realizujte.
4. Sestavte řídicí systém s využitím platformy Arduino.
5. Implementujte software pro řídicí mikro počítač.

Rozsah bakalářské práce:

Rozsah příloh:

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

Seznam doporučené literatury:

1. CATSOULIS, John. Designing embedded hardware. 2nd ed. Sebastopol, CA: O'Reilly, 2005, xvi, 377 p. ISBN 0596007558.
2. LADMAN, Josef. Elektronické konstrukce pro začátečníky. Praha: BEN – technická literatura, 2001. ISBN 80-730-0015-6.
3. MARGOLIS, Michael. Arduino cookbook. 2nd ed. Sebastopol, Calif.: O'Reilly, 2012, xx, 699 p. ISBN 1449313876.
4. NOVÁK, Petr. Mobilní roboty: pohony, senzory, řízení. Praha: BEN – technická literatura, 2005. ISBN 80-7300-141-1.
5. PINKER, Jiří. Mikroprocesory a mikro počítače. 1. vyd. Praha: BEN – technická literatura, 2004, 159 s. ISBN 80-7300-110-1.

Vedoucí bakalářské práce:

Ing. Jan Dolinay, Ph.D.

Ústav automatizace a řídicí techniky

Datum zadání bakalářské práce: 28. listopadu 2019
Termín odevzdání bakalářské práce: 15. května 2020



doc. Mgr. Milan Adámek, Ph.D.
děkan

prof. Mgr. Roman Jašek, Ph.D.
ředitel ústavu

Ve Zlíně dne 9. prosince 2019

Prohlašuji, že

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

Prohlašuji,

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

Ve Zlíně, dne 6. 8. 2020

.....
podpis diplomanta

ABSTRAKT

Cílem práce bylo sestavit zařízení pro automatické míchání nápojů. Řídicí systém zajišťuje dávkování obsahu nádrže do kelímku, ovládání výběru druhu nápoje pomocí enkodéru, čištění jednotlivých nádrží, zavodňování při doplnění nádrže a osvětlení nádrží. Samostatné nalévání jednotlivých druhů nápojů a kompletní možnosti lze kontrolovat a sledovat pomocí LCD displeje. Dále je realizován automatický posuv sklenice pod určitou nádrž, automatická kalibrace posuvníku na výchozí pozici při zapnutí zařízení i po zvolení míchaného drinku. Tento posuv je realizován pomocí krokového motoru a ozubeného řemenu. Systém je vytvořen na platformě Arduino z důvodu lehkosti programování a připojování dalších periférií. Zařízení bylo podrobena delšímu testování a funguje dle očekávání.

Klíčová slova: Arduino, řídicí systém, míchač, nápoj, dávkování, pojezd, nádrž, posuvník

ABSTRACT

The aim of this bachelor thesis was to assemble a device for automatic mixing of drinks. The control system ensures dosing of the contents of the tank into the cup, control of the selection of the type of beverage by means of an encoder, cleaning of individual tanks, irrigation when refilling the tank and lighting of the tanks. Separate pouring of individual types of drinks and complete options can be controlled and monitored using the LCD display. Furthermore, there is an automatic sliding of the cup under a certain tank, automatic calibration of the slider starting position when the device is switched on and after selecting the mixed drink. This positioning is implemented using a stepper motor and a toothed belt. The system is based on the Arduino platform because of the ease of programming and connecting other peripherals. The device has been tested for a long period of time and it works as expected.

Keywords: Arduino, control system, mixer, drink, dosage, gear, tank, slider

Poděkování Tímto bych chtěl srdečně poděkovat vedoucímu této bakalářské práce, a to panu Ing. Janu Dolinayovi, Ph.D. za veškerou pomoc a ochotu spojenou s bakalářskou prací. Dále bych chtěl poděkovat mé rodině za podporu během celého studia a mému otci za pomoc a rady ohledně konstrukce automatického míchače.

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

OBSAH

1 ÚVOD	9
TEORETICKÁ ČÁST	10
1. TEORETICKÝ ÚVOD	11
1.1. Existující konstrukce zařízení na automatické míchání nápojů	11
1.1.1. BARSYS 2.0.....	12
1.1.2. THE BARTESIAN.....	12
1.1.3. MIXOLOGIQ	13
1.1.4. SOMABAR.....	14
1.1.5. DRINKWORKS.....	14
1.2. PLATFORMA ARDUINO.....	15
1.2.1. Typy desek Arduino	15
1.2.1.1. ARDUINO UNO.....	16
1.2.1.2. ARDUINO NANO	16
1.2.1.3. ARDUINO MINI	17
1.2.1.4. ARDUINO LEONARDO	17
1.2.1.5. ARDUINO MICRO	18
1.2.1.6. ARDUINO MEGA2560	18
1.2.1.7. ARDUINO LILYPAD.....	19
PRAKTICKÁ ČÁST	20
2 NÁVRH VLASTNÍHO ŘÍDICÍHO SYSTÉMU	21
2.1 Mechanický návrh provedení zařízení.....	22
2.2 Zvolené komponenty vlastního návrhu zařízení	22
2.2.1 Relé modul 8 kanálů.....	23
2.2.2 Krokový motor SX17-1005LQCEF	23
2.2.3 Vývojová deska Arduino Mega2560.....	24
2.2.4 Motorový driver A4988.....	24
2.2.5 RGB LED prsten	25
2.2.6 Rotační enkoder	25
2.2.7 Posuvník sklenice.....	26
2.2.8 Nádrž	26
2.2.9 Držák pro nádrž.....	27
2.2.10 Elektromagnetický ventil.....	27
2.2.11 Čerpadlo	28
2.2.12 Zdroj napájení	28
2.2.13 LCD displej.....	29
2.3 Zapojení jednotlivých komponent.....	30

2.4	Programová část.....	32
2.4.1	Vývojový diagram.....	33
2.4.2	void setup()	33
2.4.3	void loop()	34
2.4.4	void enkoder().....	35
2.4.5	void kalibrace()	35
2.4.6	void led().....	36
2.4.7	void alko().....	37
2.4.8	void nealko()	37
2.4.9	void VelkeDavky()	38
2.4.10	void MaleDavky()	39
2.4.11	void cistení()	39
2.4.12	void zavodnění()	40
2.4.13	void nulovánípozice().....	41
2.4.14	void vpravo1().....	41
2.4.15	void vlevo1().....	42
2.4.16	void cervena1()	42
2.4.17	void modra1().....	42
2.4.18	void nalevani1(int x)	43
2.4.19	void Menu()	43
2.4.20	void NazvyMenu()	44
2.4.21	void probiha().....	44
2.4.22	void odebrat().....	45
2.4.23	void hotovo().....	45
2.5	Kontrola funkcionality zařízení	46
	ZÁVĚR	47
	SEZNAM POUŽITÉ LITERATURY	48
	SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ	51
	SEZNAM OBRÁZKŮ	52
	SEZNAM TABULEK	53
	SEZNAM PŘÍLOH	53

1 ÚVOD

Míchání nápojů je v dnešní době populární věc. Na trhu se začíná objevovat čím dál více přístrojů, které jsou automatické a vyžadují minimální zásah od uživatele. Jedná se pouze o údržbu a zásobování přístroje. Míchače jsou od každého výrobce jiné. Liší se zejména technologicky a konstrukčně. Takové přístroje mohou mít 2 způsoby míchání nápojů. Rozlišují se na statické a pohyblivé. U statického typu míchání se sklenice neposouvá a je pouze do ní nalévána tekutina. U pohyblivého míchání se sklenice posouvá po ose X a jezdí pod jednotlivé láhve, pod kterými se zastaví a naleje do sklenice určité množství druhu tekutiny.

Automatické míchače poslední dobou jdou ve vývoji dopředu, avšak pořád se na trhu nevyskytuje výrobek, který by bylo možné uživatelsky rozšiřovat bez nutnosti odborného zásahu. Cena dostupných automatických míchačů se pohybuje okolo 20 tisíc korun. Jedná se však o nejlevnější modely, které nabízejí uživateli hodně omezený výběr míchaných drinků.

Tato práce má za cíl sestavit a naprogramovat míchač, který bude možné snadno upravit podle požadavků uživatele. Vytvořený míchač se ovládá pomocí enkodéru. Celý přístroj komunikuje s uživatelem pomocí LCD displeje. Veškeré elektronické součástky jsou řízeny Arduinem, který se stará o veškerý chod míchače nápojů. Uživatel má možnost zvolení míchaného nápoje, samotných nápojů bez míchání a v poslední řadě také čištění, které je důležité z důvodu udržování minimální hygieny.

Teoretická část bakalářské práce se zabývá popsáním existujících konstrukcí automatických míchačů. Další část práce je tvořena popisem vlastního zařízení na automatické míchání nápojů. Dále teoretická část obsahuje popsání použitých komponentů, které byly vybrány pro sestavení tohoto přístroje. Dále je popsána řídicí mikropočítačová vývojová deska Arduino, která tvoří řídicí jednotku míchače.

Praktická část bakalářské práce se zabývá hardwarové realizaci automatického míchače, do které se řadí i sestavený řídicí systém automatického míchače. Dále se zabývá popsáním jednotlivých funkcí, které zajišťují funkčnost automatického míchače.

TEORETICKÁ ČÁST

1. TEORETICKÝ ÚVOD

Pro míchání nápojů je potřebné si zvolit správný výběr ingrediencí, ze kterých se míchané nápoje skládají. Základem drinků jsou většinou alkoholické nápoje, ke kterým se přidá nealkoholický nápoj. Pro každého uživatele jsou požadavky drinků jiné. Tím se liší i celkový počet nápojů, ze kterých se drink skládá. Velikost každého míchače je individuální a závislá na počtu míchaných nápojů.

V dnešní době se v baru nachází barmani, kteří připraví drink na přání zákazníka. V následující praktické části bakalářské práce budu popisovat konstrukci a realizaci automatického zařízení na míchání drinků, které bude usnadňovat manuální práci pro namíchání drinku bez znalosti míchání.

1.1. Existující konstrukce zařízení na automatické míchání nápojů

Z důvodu velkého zájmu mezi lidmi je možné v dnešní době míchat drinky s minimální pomocí uživatele. Veškerá práce k výrobě míchaného drinku je zautomatizována. Dostupnost těchto zařízení je na trhu s každým dnem vyšší. V České republice zatím není žádný výrobce, který by používal pro míchání drinků pohyb sklenice. Na tuzemském trhu jsou pouze výrobci automatů statických. Přední výrobci těchto míchačů, nabízí svým klientům zařízení, které umožňuje funkce, např. měření aktuální teploty nápojů, bezdrátovou komunikaci s mobilním zařízením přes aplikaci, chlazení nápojů, pojezd sklenice pro dávkování jednotlivých druhů nápojů, LCD displej pro komunikaci s uživatelem s možností ovládní tlačítka nebo případně pomocí dotykové obrazovky. Dále obsahují snímače, které jsou nezbytné pro nejlepší zajištění fungování zařízení. Na zahraničním trhu se nachází i zařízení, které jsou levnější, avšak nemusí být tak vybavené například snímači nebo dotykovou obrazovkou. Při výběru je nutné zjistit si o jednotlivých výrobcích, a jejich výrobců bližší specifikace a vybírat podle možností a potřeb uživatele.

1.1.1. BARSYS 2.0

Výrobce Barsys patří mezi přední výrobce automatických míchačů nápojů. Uživateli nabízí pouze bezdrátovou komunikaci pomocí druhého zařízení. Není tak možné ovládat míchač bez mobilního telefonu. Pro připojení podporuje zařízení s operačním systémem iOS od společnosti Apple a také Android. Barsys 2.0 umožňuje míchání až z 8 nápojů. Dále je vybaven podsvícením sklenice. Jedná se pohyblivý míchač, který pracuje na principu posunu sklenice po ose X. Doba namíchání jednoho drinku trvá v rozmezí 15-30 sekund. Výrobce udává ve specifikacích spotřebu 50W. Cena tohoto produktu se pohybuje okolo 1000 \$. [1]



Obrázek 1 Barsys 2.0 [1]

1.1.2. THE BARTESIAN

Dalším výrobcem je společnost Bartesian. Zabývá se výrobou statického míchače nápojů. Pro míchání drinků si vybírá pouze ze 4 nápojů (ingrediencí). Není zde možnost bezdrátové komunikace pomocí druhého zařízení. Volbu nápoje si uživatel vybírá pomocí vložení (kapsle) do přístroje, na kterém se nachází čárový kód. Zařízení tento kód přečte, uživatel si zvolí na dotykovém displeji sílu drinku a přístroj následně drink připraví během několika sekund. Pokud uživatel chce mít možnost míchání více drinků, musí si „kapsle“ s čárovým kódem od výrobce koupit za danou částku 14,99 \$. Samotné zařízení stojí kolem 350 \$. [2]



Obrázek 2 The Bartesian [2]

1.1.3. MIXOLOGIQ

Mixologiq patří vybaveností mezi nejlepší automatické míchače nápojů. Uživatel má na výběr z 300 naprogramovaných drinků. Drink trvá zařízení umíchat za 30 sekund. Komunikace mezi uživatelem a zařízením probíhá za pomoci dotykového displeje. Firma má sídlo ve Francii. Cenu zařízení na oficiálních stránkách výrobce neuvádí. Na neoficiálních zdrojích je uvedena cena okolo 6000 \$. Mixologiq chladí své láhve pomocí lednice, která je součástí zařízení. Láhve si uživatel vloží do lednice a vloží hadičku do každé lahve, takhle je zajištěno nalévání obsahu láhve do sklenice. [3]



Obrázek 3 Mixologiq [3]

1.1.4. SOMABAR

Mezi další statické automatické míchače patří výrobce Somabar, který nabízí zařízení s možností míchání drinků z 6 nápojů. Do zařízení se nepřipojují láhve přímo od výrobců nápojů, ale přímo od výrobce Somabar. Do lahví se nalejí nápoje na míchání a následně se tyto lahve připojí do zařízení. Drinky je možné si vybrat pomocí aplikace přes tablet nebo mobilní zařízení. Opět zde chybí komunikace přímo ze zařízení jako i u některých podobných zařízeních. Výrobce pochází z USA. Somabar uvádí ve specifikacích zařízení rychlost umíchaní drinku pod 10 sekund a uvádí, že zařízení udělá míchaných nápoj 10x rychleji než člověk rukama. [4]



Obrázek 4 Somabar [4]

1.1.5. DRINKWORKS

Drinkmaker od společnosti Drinkworks, funguje na principu vložení kapsle s již namíchaným drinkem do zařízení. Poté uživatel stiskne tlačítko a drink se vylije do sklenice, zařízení během procesu nalévání drink z kapsle ochladí a následně obsah vylije do sklenice. Zařízení používá pro komunikaci s uživatelem LCD displej, na kterém vypisuje, jaká fáze míchání se zrovna provádí. [5]



Obrázek 5 Drinkmaker [5]

1.2. PLATFORMA ARDUINO

Arduino je volně dostupná platforma se svým vývojovým prostředím. Pochází z Itálie a poprvé bylo světu představeno v roce 2005. Společnost si dávala za cíl vytvořit jednoduchou platformu určenou zejména pro studenty, kterým umožní rychlý vývoj a jednoduché použití. Během 5 let společnost zaznamenala obrovský zájem na trhu a prodala za tuto dobu přes 120 tisíc kusů. Arduino je založeno na mikrokontrolerech ATmega od společnosti Atmel. Je přizpůsobenou pro jednoduché připojení elektronických součástek, např. krokové motory, led diody, LCD displejů a senzorů, bluetooth modulů. Během posledních let se na trhu začaly objevovat kopie, tzv. klony, vyráběné v Asii. Z důvodu ceny kopie začali lidé spíše kupovat tyto neoriginální vývojové desky. [6]

1.2.1. Typy desek Arduino

Mikrokontroléry vývojových desek Arduino, jsou uživatelsky programovatelné přes USB rozhraní. Vývojová deska může být napájena pomocí USB kabelu z PC, které dodává do Arduino přímo 5V, nebo přímo napájena zdrojem přes souosý konektor, kterým může být připojeno napětí 6V - 15V. Za konektorem se nachází stabilizátor, který stabilizuje napětí na 5V. Ve většině případů se připojuje napájení 12V, které je v daném rozmezí od výrobce. [7]

1.2.1.1. ARDUINO UNO

Arduino Uno je v dnešní době nejpoužívanější vývojová deska od firmy Arduino. Na desce obsahuje mikrokontroler ATmega328 od firmy Atmel. Pracuje s frekvencí 16 MHz. Doporučené vstupní napětí udává výrobce v rozmezí 7–12V. Uživatel má možnost použití 6 analogových pinů a 14 digitálních pinů, ze kterých může být 6 pinů použito s PWM (pulzně šířková modulace). Váha celé desky je 25g. Rozměry desky jsou 69 * 53 mm. [8]



Obrázek 6 Arduino Uno [8]

1.2.1.2. ARDUINO NANO

Arduino Nano je osazen mini USB portem, kterým uživatel nahrává do mikrokontroleru program. Logické úrovně výstupu operují s napětím 5V. Maximální napájecí napětí se pohybuje v rozsahu 6-20V. Doporučený rozsah napájecího napětí je 7-12V. Na desce je celkem 30 pinů určených jako výstup. 14 pinů je digitálních (z toho 6 s podporou PWM výstupu) a 16 analogových. Na desce je osazen krystal o frekvenci 16 MHz. [9]



Obrázek 7 Arduino Nano [9]

1.2.1.3. ARDUINO MINI

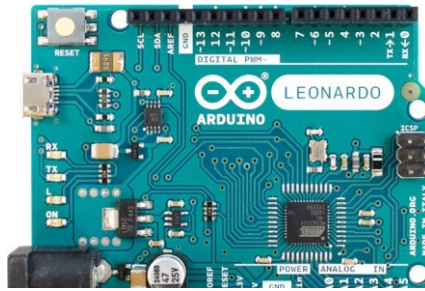
Arduino Mini je určeno pro pokročilejší uživatele převážně z důvodu, že na desce není napájen přímo konektor pro nahrávání softwaru do mikrokontroleru. Tuto desku je možné programovat použitím USB programátoru. Na desce se nachází mikrokontroler od firmy Atmel typu ATmega328P s frekvencí 16 MHz. Mini vyniká oproti jeho ostatním vrstevníkům svými rozměry, které jsou 33 * 18 * 6 mm. Na desce se dále nachází tlačítko pro reset mikrokontroléru. Na výstupu pinů může být maximální proud 150mA, přičemž je možné odebírat z jednoho pinu maximálně 20mA(z některých 40mA). Omezení je tím pádem dvojnásobné. Deska je napájena 5V. Na vývojové desce se nachází ochrana proti přetížení a obrácené polaritě napájecího napětí. [10]



Obrázek 8 Arduino Mini [10]

1.2.1.4. ARDUINO LEONARDO

Arduino Leonardo má stejný mikrokontroler na desce jako verze Micro. Je nástupcem verze Uno. Liší se v proudovém zatížení na pin. U verze Leonardo je možné zatížit pin až 40mA, u vývojové desky Micro je tato zatížitelnost poloviční 20 mA. Na desce je resetovací tlačítko pro přerušení aktuálně vykonávaných procesů. Komunikace je zajištěna díky portu micro USB. [11]



Obrázek 9 Arduino Leonardo [12]

1.2.1.5. ARDUINO MICRO

Arduino Micro svojí strukturou a stavbou vychází z verze Arduino Leonardo. Má jeho stejné schopnosti a liší se hlavně ve velikosti, která je menší než u verze Leonardo. Na desce je umístěn mikrokontroler ATmega32U4 od výrobce Atmel. Na jejím vývoji se podílela firma Adafruit, která vyrábí např. RGB LED prsteny. Dále na desce opět zajišťuje frekvenci krystalový oscilátor o frekvenci 16 MHz. Verze Micro, nabízí 20 digitálních pinů pro nastavení vstupů/výstupů, z nichž je možné 7 pinů nastavit pro PWM výstup a 12 jako analogový vstup. Pro komunikaci není potřeba externího programátor, jelikož na desce je USB port, který slouží ke komunikaci. Na vývojové desce je umístěno restartovací tlačítko. [13]



Obrázek 10 Arduino Micro [14]

1.2.1.6. ARDUINO MEGA2560

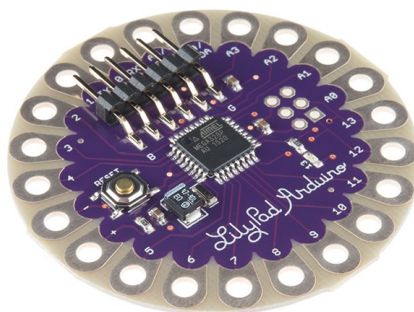
Arduino Mega2560 řídí mikrokontroler ATmega2560. Jedná se největší verzi od Arduino. Nabízí připojení až 52 vstupních/výstupních. 14 pinů je možné použít jako PWM a 16 jako analogové vstupy. Dále obsahuje krystalový oscilátor s frekvencí 16 MHz a resetovací tlačítko. [15]



Obrázek 11 Arduino Mega2560 [16]

1.2.1.7. ARDUINO LILYPAD

Arduino Lilypad je malá vývojová deska kruhového tvaru. Na desce je k dispozici 14 digitálních vstupních/výstupních pinů z toho 6 analogových a 6 pro PWM. Na Lilypad se nenachází USB převodník, pro naprogramování je zapotřebí použití převodníku. Napájecí napětí je v rozsahu 2,7-5,5V. [17]



Obrázek 12 Arduino LilyPad [18]

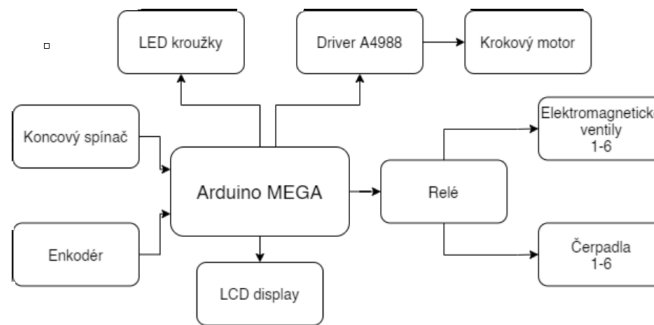
PRAKTICKÁ ČÁST

2 NÁVRH VLASTNÍHO ŘÍDICÍHO SYSTÉMU

Jelikož na tuzemském trhu nejsou automatické míchače k dispozici, bylo potřeba celý kompletní systém navrhnout. Byla možná pouhá inspirace ze zahraničního trhu. Vlastní návrh automatického míchače nabízí tyto možnosti:

- Ovládání pomocí enkodéru pro volbu míchaného nápoje
- Zavodnění vybrané nádrže pro přesnější míchání nápojů
- Čištění zvolené nádrže z hygienických důvodů
- Kalibrace posuvníku s kelímkem na začáteční pozici při zvolení drinku
- Výběr 1 drinku z 6 nabízených (označených jako Napoj 1 až Napoj 6)
- Výběr ze sekce alkoholických nápojů
 - Vodka
 - Jägermeister
 - Rum
- Výběr ze sekce nealkoholických nápojů
 - Pomerančový džus
 - Jablečný džus
 - Energetický nápoj
- Výpis aktuální volby na LCD displej
 - Procházení jednotlivých sekcí pro výběr
 - V době míchání výpis uživateli „Připravujeme“
 - Při dokončení vypsání hlášky „Odeberte napoj“

Blokové schéma řídicího systému automatického míchače:

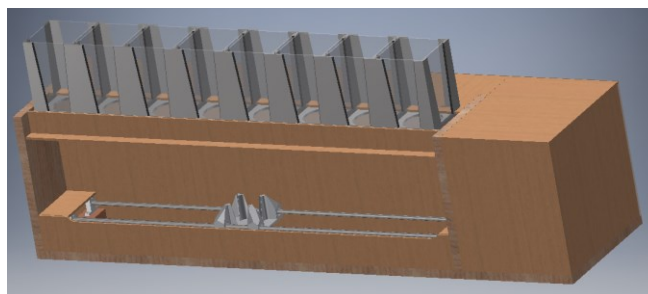


Obrázek 13 Blokové schéma řídicího systému

Z blokového schéma lze vidět principiální řízení. Arduino jako řídicí jednotka reaguje na podnět enkodéru (volby uživatele) a vypisuje na LCD displeji aktuální výběr. Po výběru následuje rozsvícení RGB LED prstenu červenou barvou, poté posuvník s kelímkem se posouvá pod červeně svítící RGB LED prsten. Po posunu pod danou nádrž začne probíhat nalévání obsahu z nádrže v určitém množství. Tento obdobný cyklus se opakuje v závislosti na míchání nápoje. Po namíchání veškerých ingrediencí se posuvník vrátí do začátečního bodu s již připraveným drinkem určeným pro odebrání.

2.1 Mechanický návrh provedení zařízení

V obrázku je znázorněn posuvník sklenice s nádržemi, který je ze spodní strany připojen k ozubenému řemenu. Posun zajišťuje krokový motor, který podle směru otáčení pohybuje s ozubeným řemenem. Posuvník se tímto způsobem posouvuje pod určitou nádrž a poté se provede nalévání nápoje do kelímku.[19]



Obrázek 14 Mechanický návrh zařízení

2.2 Zvolené komponenty vlastního návrhu zařízení

Následující praktická část je věnována popisu použitých komponent a součástí v hardwarové realizaci.

2.2.1 Relé modul 8 kanálů

Relé modul s 8 kanály (relé) disponuje s pracovním napětím 5V, kterým je spínáno. Výrobce udává že, na každé relé může být připojeno až 230V AC (střídavého napětí) nebo 30V DC (stejnoseměrného napětí). Relé modul má široké možnosti použití. Jeho hlavní využití je pro všechna řízení MCU (mikrokontroléru) v průmyslovém odvětví, dále pro inteligentní domácnost a spousty dalších. Při propojení s platformou Arduino se každé relé spíná pomocí logické úrovně „0“. Tímto dochází k propojení kontaktu v relé. Modul je poměrně velký, jeho rozměry jsou 5.3 x 13.5 x 1,7 cm (Š x D x V). [20]



Obrázek 15 Relé modul 8 kanálů [21]

2.2.2 Krokový motor SX17-1005LQCEF

Krokový motor pochází od společnosti Microcon. Jeho délka kroku činí $1,8^\circ$ s tolerancí $\pm 0,1^\circ$ na krok. Pro jeho výkon je vhodný na automatizační pojezd menších rozměrů. V tomto případě na posun posuvníku se sklenicí. Váhu dle výrobce je 300g. Přídržný výkon má 0,5 Nm. Další jeho významné uplatnění je v 3D tiskárnách, kde slouží pro posuv os. Jedná se o dvoufázový krokový motor. [22]



Obrázek 16 Krokový motor [23]

2.2.3 Vývojová deska Arduino Mega2560

Jedná se o desku, která se vyznačuje velkým počtem I/O pinů. Své uplatnění najde u větších projektů, kde se nachází více komponent pro řízení nebo snímání. Na desce jsou otvory pro uchycení distančních sloupků a tak je možné desku uchytit na potřebném místě. [24]



Obrázek 17 Arduino Mega2560 [25]

2.2.4 Motorový driver A4988

Pro fungování driveru je zapotřebí napájení 5V pro samotný mikrokontroler a 12V pro napájení připojeného krokového motoru. Dále se na driveru nachází 4 piny pro připojení dvoufázového krokového motoru. Motorový driver je dodáván s chladičem, z důvodu chlazení mikrokontroleru. Výstupní proud je udáván výrobcem maximálně 2A a vstupní napětí pro krokový motor může být až 36V. Jedná se o nejpoužívanější typ motorových driverů v 3D tiskárnách. [26]



Obrázek 18 Motorový driver A4988 [27]

2.2.5 RGB LED prsten

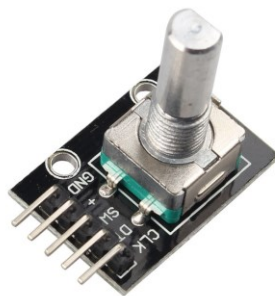
RGB LED prsten slouží k osvětlení. Je napájen 5V. Prsten má v sobě integrovaný driver pro komunikaci s platformou Arduino. Na desce prstenu se nachází 4 vývody. 1 pro vstupní data „IN“, 2 pro výstupní data „OUT“ (pro připojení více LED prstenů do série) a následně 2 vývodu pro napájení „5V“ a „GND“. [28]



Obrázek 19 RGB LED prsten [29]

2.2.6 Rotační enkoder

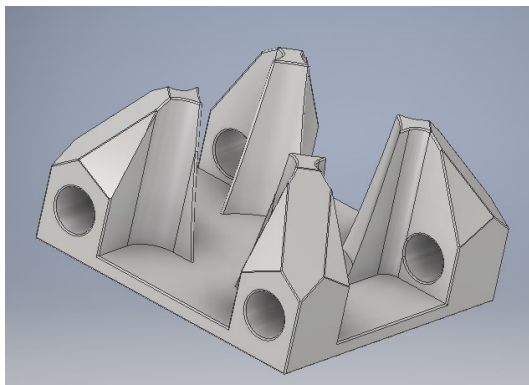
Existují 2 typy, absolutní a inkrementální. Absolutní dává přesnou polohu ve stupních a Inkrementální udává, o kolik kroků bylo hřídelí posunuto. V případě použití míchače se jednalo o inkrementální. Na desce se nachází 5 pinů pro ovládání. „GND“ – uzemnění, „VCC“ - napájení (3,3 – 5V), „SW“ – pro stisk tlačítka po stisknutí se napětí změní na nízké. Zbylé 2 piny (DT a CLK) slouží jako výstup, dle kterých reaguje řídicí jednotka na posun hřídele. [30]



Obrázek 20 Rotační enkoder [31]

2.2.7 Posuvník sklenice

Jedná se o komponentu, do které je umístěn kelímek. V posuvníku jsou udělány otvory pro 4 lineární ložiska z důvodu klouzavého pohybu. Tato nezbytná součást zařízení byla navržena v programu Autodesk Inventor a následně vytisknuta na 3D tiskárně. Pro vytisknutí byl zvolen filament typu PLA od společnosti Gembird.



Obrázek 21 Posuvník sklenice

2.2.8 Nádrž

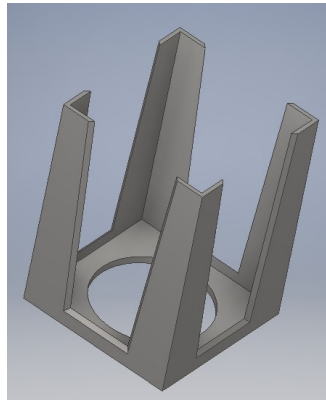
Nádrž je spojena pomocí akvaristického lepidla značky Den Braven, které se používá k lepení akvárií. Pro stěny nádrže bylo použito plexisklo o tloušťce 3mm. Na dně se nachází vývod, kterým je zajištěna výpust' tekutiny. Rozměry jsou 100 x 100 x 150 mm (š x d x v). Rozměr nádrže byl zvolen s ohledem na váhu celého zařízení ale také z důvodu rozumného doplňování obsluhou, v provozních podmínkách.



Obrázek 22 Nádrž

2.2.9 Držák pro nádrž

Pro upevnění nádrže a také kvůli vzhledové stránce bylo nutné vytvořit držák pro samotnou nádrž. Jeho návrh byl vytvořen v programu Autodesk Inventor a následně vytisknut na 3D tiskárně. Na spodní straně se nachází kulatý otvor z důvodu vývodky nádrže a kvůli vložení RGB LED prstenu, který je umístěn z důvodu podsvícení nádrže.



Obrázek 23 Držák na nádrž

2.2.10 Elektromagnetický ventil

Pro uchování obsahu v nádrži bylo nutné připojit elektromagnetický ventil. Je k dispozici ve dvou verzích, „normálně otevřený“ a „normálně zavřený“. Jeho napájecí napětí je 12V. Pro návrh míchače byl zvolen typ „normálně zavřený“. [32]



Obrázek 24 Elektromagnetický ventil [33]

2.2.11 Čerpadlo

Čerpadlo má napájecí napětí 12V. Jeho rozměry jsou 90 x 40 x 35mm (d x š x v). Jeho váhu udává výrobce 106g. Maximální sací výška jsou 2m. Za minutu vyčerpá množství tekutiny v rozmezí 1,5 – 2l. Pracovní proud se pohybuje v 0,5 až 0,7 A. Používá se v kávovarech, myčkách nádobí, domácích spotřebičích atd. [34]



Obrázek 25 Čerpadlo [35]

2.2.12 Zdroj napájení

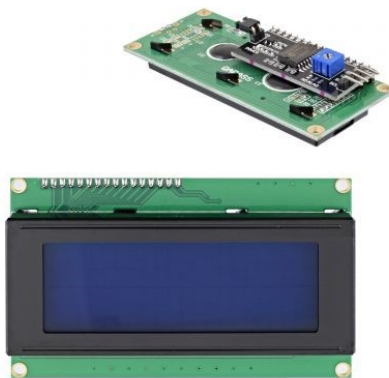
Z důvodu síťového zapojení bylo potřeba zvolit zdroj z 230V na 12V a dokáže dodávat proud o hodnotě 40 A. Výkon tohoto zdroje je 480W. Většina elektronických komponentů s tímto napětí pracují. Ostatní jsou na 5V napájení, pro které byl použit regulátor 7805, který stabilizuje napětí z 12V na 5V. [36]



Obrázek 26 Zdroj napájení 480W [37]

2.2.13 LCD displej

LCD displej komunikuje pomocí sběrnice I2C. Displej je zapojen čtyřmi piny, přičemž 2 jsou určeny pro komunikaci a 2 pro připojení napájení, VCC a GND. Komunikační piny se nazývají SDA a SCL. Pin SDA slouží pro data a pin SCL je určen pro hodinový signál. Napájecí napětí je uvedeno 5V. Samotný displej je podsvícený modrou barvou a písmena tvoří barva bílá. Pro doladění kontrastu se na desce ze zadní části nachází potenciometr, kterým lze po jeho otočení doladit viditelnost písmen. [38]



Obrázek 27 LCD Displej [39]

2.3 Zapojení jednotlivých komponent

Následující tabulky shrnují zapojení jednotlivých pinů mezi řídicí jednotkou a ostatními komponenty viz tabulka 1 a tabulka 2, ve kterých se nachází všechny podstatné informace, které jsou nezbytné pro správné zapojení zařízení.[40]

Součástka	Pin součástky	Pin v Arduino Mega	Sběrnice +5V	Sběrnice +12V	Sběrnice "-"	Relé
LCD display	SCL	21				
	SDA	20				
	VCC	5V				
	GND	GND				
Enkodér	CLK	A3				
	DT	A2				
	SW	A1				
	VCC	5V				
	GND	GND				
Driver A4988	DIRECTION	22				
	STEP	23				
	ENABLE	4				
	VMOT			+		
	GND				-	
	VDD	5V				
	GND	GND				
Relé deska 1	GND				-	
	IN1	38				
	IN2	40				
	IN3	42				
	IN4	44				
	IN5	46				
	IN6	48				
	IN7	50				
	IN8	52				
VCC			+			
Relé deska 2	GND				-	
	IN1	39				
	IN2	41				
	IN3	43				
	IN4	45				
	IN5	47				
	IN6	49				
	IN7	51				
	IN8	53				
VCC			+			

Tabulka 1 Zapojení součástek

Součástka	Pin součástky	Pin v Arduino Mega	Sběrnice +5V	Sběrnice +12V	Sběrnice "-"	Relé
LED kruh 1	VCC		+			
	DATA	25				
	GND				-	
LED kruh 2	VCC		+			
	DATA	24				
	GND				-	
LED kruh 3	VCC		+			
	DATA	27				
	GND				-	
LED kruh 4	VCC		+			
	DATA	26				
	GND				-	
LED kruh 5	VCC		+			
	DATA	29				
	GND				-	
LED kruh 6	VCC		+			
	DATA	28				
	GND				-	
Koncový spínač	VCC		+			
	DATA	32				
Čerpadlo 1	VCC			+		relé1 IN1
	GND				-	
Čerpadlo 2	VCC			+		relé1 IN2
	GND				-	
Čerpadlo 3	VCC			+		relé1 IN3
	GND				-	
Čerpadlo 4	VCC			+		relé1 IN4
	GND				-	
Čerpadlo 5	VCC			+		relé1 IN5
	GND				-	
Čerpadlo 6	VCC			+		relé1 IN6
	GND				-	
Elektromagnetický ventil 1	VCC			+		relé2 IN1
	GND				-	
Elektromagnetický ventil 2	VCC			+		relé2 IN2
	GND				-	
Elektromagnetický ventil 3	VCC			+		relé2 IN3
	GND				-	
Elektromagnetický ventil 4	VCC			+		relé2 IN4
	GND				-	
Elektromagnetický ventil 5	VCC			+		relé2 IN5
	GND				-	
Elektromagnetický ventil 6	VCC			+		relé2 IN6
	GND				-	

Tabulka 2 Zapojení součástek

2.4 Programová část

Vývoj samotného softwaru pro zařízení byl prováděn ve vývojovém prostředí Arduino IDE. Tato platforma je podporována na operačních systémech Windows, MAC OS a Linux. Zhotovený software se nahrává do řídicí jednotky „Arduino“ pomocí sériové sběrnice USB. [41]

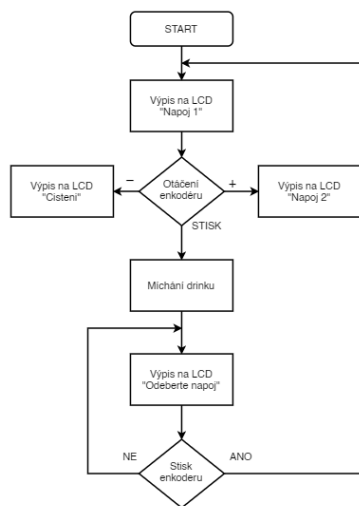
Program se dělí na několik částí:

- Defínování vstupů a výstupů pro veškeré použité komponenty
- Smyčka pro počáteční nastavení pinů „void setup()“
- Hlavní smyčka „void loop()“
- Nadefinované funkce pro usnadnění psaní kódu

Mezi funkce patří ovládání krokového motoru pro zajištění posunu pojezdu se sklenicí, podsvícení jednotlivých nádrží, kalibrace počáteční pozice z důvodu bezpečnosti zařízení, nalévání nápojů pro míchání, výpis informací uživateli na LCD displej, ovládání pomocí enkoderu a v poslední řadě možnost čištění a zavodňování nádrží.[42]

2.4.1 Vývojový diagram

V dílčím kroku „Míchání drinku“ jsou zahrnuty funkce pro kalibraci posuvníku na počáteční pozici, podsvícení jednotlivých nádrží, posuny pod dané nádrže s následným naléváním nápojů do sklenice, včetně výpisu hlášek na LCD displej. Pomocí točení enkoderu se uživatel posouvá v menu na další možnosti, kde může opět pomocí stisku potvrdit svůj výběr.



Obrázek 28 Ukázka vývojového diagramu

2.4.2 void setup()

Po definování pinů následuje jejich nastavení. Jestli se jedná o vstupní pin, který snímá nebo se jedná o výstupní pin.

```

void setup() {

  // MOTOR
  pinMode(stepPin, OUTPUT);
  pinMode(dirPin, OUTPUT);
  pinMode(enable, OUTPUT);
  digitalWrite(enable, HIGH);
  //INICIALIZACE DISPLEJE
  lcd.init(); //inicializace lcd
  lcd.backlight(); //kontrast displeje

  //NASTAVENI LED PRSTENU
  ledka1.begin();
  ledka1.setBrightness(20); //nastaveni jasu prvnio LED prstenu
  ledka2.begin();
  ledka2.setBrightness(20);
  ledka3.begin();
  ledka3.setBrightness(20);
  ledka4.begin();
  ledka4.setBrightness(20);
  ledka5.begin();
  ledka5.setBrightness(20);
  ledka6.begin();
  ledka6.setBrightness(20);

  //NASTAVENI ENKODERU PRO PINY NA VSTUPY S POVOLENYM pullup rezistorem
  pinMode (CLK, INPUT_PULLUP);
  pinMode (DT, INPUT_PULLUP);
  pinMode (SW, INPUT_PULLUP);

  // NASTAVENI PINU LED PRSTENU JAKO VYSTUP
  pinMode(LED1, OUTPUT);
  pinMode(LED2, OUTPUT);
  pinMode(LED3, OUTPUT);
  pinMode(LED4, OUTPUT);
}
  
```

Obrázek 29 Ukázka funkce void setup()

2.4.3 void loop()

Jedná se o hlavní programovou funkci. Zde se pouze provádí přičítání nebo odčítání proměnných pomocí enkoderu. Pokud je některá z proměnných v určitém rozmezí hodnot provádí se volaná funkce. Na tomto principu funguje hlavní část programu. Na první proměnné je znázorněn popis funkčnosti. U ostatní je tento princip stejný. [44]

```
void loop() {
  enkoder(); //enkoder musi porad byt volan z duvodu tocení enkoderem

  if (velke >= 0 && velke <= 6) //pokud promenna velke nabyva hodnoty v rozmezi 0-6
  {
    VelkeDavky();           //probihani funkce Velkedavky
  }
  if (male >= 0 && male <= 5)
  {
    MaleDavky();
  }

  if (menu >= 0 && menu <= 11)
  {
    Menu();
    NazvyMenu();
  }

  if (nealk >= 0 && nealk <= 5)
  {
    nealko();
  }

  if (alk >= 0 && alk <= 5)
  {
    alko();
  }

  if (clean >= 0 && clean <= 8)
  {
    cisteni();
  }

  if (voda >= 0 && voda <= 8)
  {
    zavodneni();
  }
}
```

Obrázek 30 Funkce void loop()

2.4.4 void enkoder()

Tato funkce má za úkol reagovat na ovládání enkoderu. Jeho možnosti jsou otáčení vlevo, vpravo a jeho stisk pro výběr. Při točení enkoderem proti směru hodinových ručiček dochází k odečítání hodnot proměnných o hodnotu „-1“ a při otáčení enkoderu po směru hodinových ručiček (vpravo) dojde k přičítání hodnot proměnných o hodnotu „+1“.

```
void enkoder()
{
    if (jednou != 1) //pocatecni provedeni funkci Menu() a NazvyMenu() pri zapnuti zarizeni
    {
        Menu();
        NazvyMenu();
        jednou++;
    }
    aktualniStav = digitalRead(CLK); // cteni pinu CLK
    // pokud se predchozi a soucasny stav CLK lisi, znamena to, ze doslo k pulzu
    if (aktualniStav != predchoziStav)
    {
        // pokud se stav DT lisi od stavu CLK, znamena to, ze se enkoder otaci ve smeru hodinovych rucicek
        if (digitalRead(DT) != aktualniStav)
        {
            menu++; //pricitani hodnoty u promenne menu +1
            //Menu();
            //NazvyMenu();
            nealk++;
            alk++;
            velke++;
            male++;
            clean++;
            voda++;
        }
        else
        {
            menu--; //odecitani hodnoty u promenne menu -1
            //Menu();
            //NazvyMenu();
            nealk--;
            alk--;
            velke--;
            male--;
            clean--;
            voda--;
        }
    }
    predchoziStav = aktualniStav; //nastaveni promenne predchoziStav na hodnotu aktualniStav
}
```

Obrázek 31 Funkce void enkoder()

2.4.5 void kalibrace()

Z důvodu zajištění menšího rizika problému mezi uživatelem a zařízením byla naprogramována funkce „kalibrace()“. Při programování této funkce nebylo potřeba volit v konstrukci zařízení senzoru. Jedná se pouze o počáteční start pojezdu, jelikož pohyb je zajištěn krokovým motor, který je řízen přesným počtem kroků. Bez tohoto programového ošetření by mohlo dojít ke špatnému posunu pojezdu se sklenicí a obsah nádrže při čerpání by byl naléván mimo prostor sklenice.

```

void kalibrace()
{
  if (n != 1) // podmínka pro vykonání kalibrace
  {
    for (int i = 0; i < 4000; i++) //cyklus pro posouvání pojezdu
    {
      pozice = digitalRead(koncak); //čtení hodnoty koncového spínače
      if (pozice == LOW) //pokud není pojezd na začátku kde je koncový spínač
      {
        digitalWrite(enable, LOW); //nastavení pinu pro povolení řízení driveru A4988
        digitalWrite(dirPin, LOW); //nastavení pinu pro otáčení krokového motoru ve směru hodinových ručiček
        vpravo(); //volání funkce pro otočení doprava
        //}
      }
      if (pozice == HIGH)
      {
        digitalWrite(enable, HIGH); //nastavení pinu pro nemožnost řízení driveru A4988
        break; //vyskočení ze smyčky
      }
    }
    Serial.print("Kalibrace provedena"); //vypis hlasky na seriovou linku
    n++; //pricenti hodnoty promenne n +1
  }
}

```

Obrázek 32 Funkce void kalibrace()

2.4.6 void led()

Tato funkce slouží pro zapnutí všech LED prstenců pod nádržemi při spuštění zařízení. Její použití je ve funkci „setup()“. Všechny LED prstence se rozsvítí na modrou barvu. Tímto je zajištěn pěkný efekt nádrží a při večerních hodinách, kdy je méně světla, dodává zařízení pěkný vzhled.

```

void led()
{
  if (c != 1) //podmínka pro rozsvícení všech LED prstenců na modrou barvu
  {
    for (int i = 0; i < pocet24; i++) { // cyklus pro nastavení všech LEDEK na PRSTENCU
      ledka1.setPixelColor(i, ledka1.Color(0, 0, 255)); // LED1- modra
      ledka2.setPixelColor(i, ledka2.Color(0, 0, 255)); // LED2- modra
      ledka3.setPixelColor(i, ledka3.Color(0, 0, 255)); // LED3- modra
      ledka4.setPixelColor(i, ledka4.Color(0, 0, 255)); // LED4- modra
      ledka5.setPixelColor(i, ledka5.Color(0, 0, 255)); // LED5- modra
      ledka6.setPixelColor(i, ledka6.Color(0, 0, 255)); // LED6- modra
    }
    ledka1.show(); //rozsvícení LED prstenu 1
    ledka2.show(); //rozsvícení LED prstenu 2
    ledka3.show(); //rozsvícení LED prstenu 3
    ledka4.show(); //rozsvícení LED prstenu 4
    ledka5.show(); //rozsvícení LED prstenu 5
    ledka6.show(); //rozsvícení LED prstenu 6
    Serial.print("ZAPNUTO"); //vypis na seriovou linku ZAPNUTO
    c++; //pricenti promenne c o hodnotu +1
  }
}

```

Obrázek 33 Funkce void led()

2.4.7 void alko()

Ve funkci „alko()“ se nachází menu pro výběr samostatného alkoholického nápoje. Uživatel má na výběr ze tří druhů. Poté má volbu výběru množství. Pro alkoholické nápoje jsou na výběr malé dávky z funkce „MaleDavky()“.

```
void alko()
{
    switch (alk) //vytvoření switche pro procházení v menu
    {
        case 0:
            alk = 4;
            break;
        case 1:
            lcd.setCursor(4, 0); //nastavení kurzoru na 4 znak zleva na 0 radek
            lcd.print("Drink Maker"); //vypsání hlasky na LCD
            lcd.setCursor(2, 2); //nastavení kurzoru na 2 znak zleva na 2 radku
            lcd.print("<< Vodka >>"); //vypsání hlasky na LCD
            stiskTlacitka = digitalRead(SW); //ctení pinu tlacitka
            if (stiskTlacitka == 0) //pokud je promenna ma log. hodnotu 0 provede se nasledujici kod
            {
                delay(200); //zpozdení 200ms
                lcd.clear(); //vymazání displeje
                volba2 = 1; //pomocna promenna pro menu MaleDavky
                male = 1; //přirazení hodnoty 1 pro promennou male
                alk = 400000; //nastavení promenne mimo funkcní rozmezi ve smyčce loop
            }
            break;
        case 2:
            lcd.setCursor(4, 0);
            lcd.print("Drink Maker");
            lcd.setCursor(2, 2);
            lcd.print("<< Rum >>");
            stiskTlacitka = digitalRead(SW);
            if (stiskTlacitka == 0)
            {
                delay(200);
                lcd.clear();
                volba2 = 2;
                male = 1;
                alk = 400000;
            }
            break;
    }
}
```

Obrázek 34 Ukázka funkce alko()

2.4.8 void nealko()

Funkce „nealko()“ je obdobná jako funkce „alko()“. Na výběr má uživatel ze tří druhů nealkoholických nápojů a to je jablko, pomeranč a energetický nápoj. Po výběru nápoje je uživatel přesměrován do funkce „VelkeDavky()“.

```

void nealko()
{
    switch (nealk)
    {
        case 0:
            nealk = 4;
            break;
        case 1:
            lcd.setCursor(4, 0);           //nastaveni kurzoru na 4 znak na 0 radku
            lcd.print("Drink Maker");     //vypis hlasky na lcd
            lcd.setCursor(2, 2);         //nastaveni kurzoru na 2 znak na 0 radku
            lcd.print("<< Jablko >>");    //vypis hlasky na lcd
            stiskTlacitka = digitalRead(SW); //cteni pinu tlacitka
            if (stiskTlacitka == 0)      //pokud je stisknuto tlacitko provede se nasledujici usek kodu
            {
                delay(200);              //zpozdeni 0,2s
                lcd.clear();             //vymazani lcd
                volbal = 1;              //nastaveni hodnoty 1 do promenne volbal
                velke = 1;               //nastaveni hodnoty 1 do promenne velke
                VelkeDavky();            //provedeni funkce VelkeDavky()
                nealk = 400000;          //nastaveni promenne mimo funkcní rozmezi ve smyčce loop
                nealko();                //provedeni funkce nealko
            }
        break;
    }
}

```

Obrázek 35 Ukázka funkce void nealko()

2.4.9 void VelkeDavky()

Do funkce „VelkeDavky()“ je uživatel přesměrován z menu nealkoholických nápojů. Zde si může vybrat, jaké množství chce. Množství není přesné, jelikož není pro snímání v zařízení zapojen průtokoměr, který by měřil množství nalité tekutiny. Po vybrání množství a potvrzení je provedeno nalití nápoje do kelímku. Na výběr má uživatel 100,150,200 a 250 ml.

```

void VelkeDavky()
{
    switch (velke)
    {
        case 0:
            velke = 5;
            break;
        case 1:
            lcd.setCursor(4, 0);           //nastaveni kurzoru na 4 znak na 0 radek
            lcd.print("Drink Maker");     //vypis hlasky na lcd
            lcd.setCursor(2, 2);         //nastaveni kurzoru na 2 znak na 2 radek
            lcd.print("<< 100 ml >>");    //vypis hlasky na lcd
            stiskTlacitka = digitalRead(SW); //cteni pinu tlacitka
            if (stiskTlacitka == 0)      //pokud bylo stisknuto provede se nasledujici usek kodu
            {
                if (volbal == 1)
                {
                    probiha();           //provedeni funkce probiha()
                    nulovanipozice();    //provedeni funkce nulovani()
                    kalibrace();         //provedeni funkce kaibrace()
                    delay(2000);         //zpozdeni 2s
                    cervena5();          //provedeni funkce ls
                    delay(1000);         //zpozdeni 1s
                    vlevo4();            //provedeni funkce vlevo4()
                    delay(500);          //zpozdeni 0,5s
                    nalevani5(5500);     //provedeni funkce nalevani5()
                    delay(100);          //zpozdeni 0,1s
                    modra5();            //provedeni funkce modra5()
                    delay(500);          //zpozdeni 0,5s
                    vpravo4();           //provedeni funkce vpravo4()
                    delay(500);          //zpozdeni 0,5s
                    odebrat();           //provedeni funkce odebrat()
                }
            }
        }
}

```

Obrázek 36 Ukázka funkce void VelkeDavky()

2.4.10 void MaleDavky()

Ve funkci „MaleDavky()“ si uživatel zvolí množství pro alkoholický nápoj. Je možné si vybrat ze tří množství, kterými jsou 20 ml, 40 ml a 50 ml. Poté dojde k nalití alkoholického nápoje.

```
void MaleDavky()
{
    switch (male)
    {
        case 0:
            male = 4;
            break;
        case 1:
            lcd.setCursor(4, 0);           //nastavení kurzoru na 4 znak na 0 radku
            lcd.print("Drink Maker");     //vypis hlasky na lcd
            lcd.setCursor(2, 2);         //nastavení kurzoru na 2 znak na 2 radku
            lcd.print("<< 20 ml >>");    //vypis hlasky na lcd
            stiskTlacitka = digitalRead(SW); //cteni pinu tlacitka
            if (stiskTlacitka == 0)      //pokud doso ke stisknutí tlacitka provede se nasledujici kod
            {
                if (volba2 == 1)        //pokud je splnena podminka provede se nasledujici kod
                {
                    probiha();          //provedeni funkce probiha()
                    nulovanipozice();  //provedeni funkce nulovanipozice()
                    kalibrace();        //provedeni funkce kalibrace()
                    delay(2000);        //zpozdeni 2s
                    cervena2();         //provedeni funkce cervena2()
                    delay(1000);        //zpozdeni 2s
                    vlevol();           //provedeni funkce vlevol()
                    delay(500);         //zpozdeni 0,5s
                    nalevani2(1100);    //provedeni funkce nalevani2()
                    delay(100);         //zpozdeni 0,1s
                    modra2();           //provedeni funkce modra2()
                    delay(500);         //zpozdeni 0,5s
                    vpravol();          //provedeni funkce vpravol()
                    delay(500);         //zpozdeni 0,5s
                    odebrat();          //provedeni funkce odebrat()
                }
            }
    }
}
```

Obrázek 37 Ukázka funkce void MaleDavky()

2.4.11 void cistení()

Pro udržování čistoty nádrží a hadic, byla vytvořena funkce „cistení()“. Uživatel s v hlavním menu vybere možnost „Cistení“, po kterém se objeví v menu s výběrem jednotlivých nádrží. Následně je nutné, aby uživatel vložil pod vybranou nádrž nádobu. Pokud uživatel stlačí enkoder, sepne se elektromagnetický ventil, který umožní chod tekutiny hadicí a poté dojde k zapnutí čerpadla. Obsluha drží tlačítko do doby, dokud není nádrž včetně hadic prázdná. Takhle je vypuštěn kompletní obsah nádrže například při ukončení zařízení.

```

void cisteni()
{
    switch (clean) //vytvoreni switche pro menu cisteni s promennou clean
    {
        case 0:
            clean = 7;
            break;
        case 1:
            lcd.setCursor(4, 0); //nastaveni kurzoru na 4 znak zleva na 0 radek
            lcd.print("Drink Maker"); //vypis hlasky na LCD
            lcd.setCursor(2, 2); //nastaveni kurzoru na 2 znak zleva na 2 radku
            lcd.print("<< Nadrz 1 >>"); //vypis hlasky na LCD
            stiskTlacitka = digitalRead(SW); //cteni hodnoty na pinu
            if (stiskTlacitka == 0) //pokud je hodnota na pinu rovna log.hodnote 0 provede se nasledujici
            {
                digitalWrite(ventill, LOW);
                digitalWrite(cerpadlol, LOW);
            }
            else //provede se pokud není splněna podmínka
            {
                digitalWrite(cerpadlol, HIGH); //cerpadlo vypnuto
                digitalWrite(ventill, HIGH); //ventil uzavren
            }
            break;
    }
}

```

Obrázek 38 Ukázka funkce void cisteni()

2.4.12 void zavodneni()

Funkce „zavodneni()“ slouží pro odvodnění hadic, které vedou od nádrže. Pomocí této funkce dojde k zavodnění hadic, aby bylo zajištěno nejpřesnější dávkování, jelikož nalévání probíhá za pomoci určitého času, za který je nalito určité množství tekutiny do kelímku. Uživatel si zvolí v menu pro zavodňování nádrž a stiskem enkoderu potvrdí svou volbu. Poté se posune posuvník s kelímkem pod zvolenou nádrž a dojde k zapnutí čerpadla a otevřením elektromagnetického ventilu. Tyto součástky jsou zapnuty na určitou dobu, která by měla zajistit zavodnění hadic.

```

void zavodneni()
{
    switch (voda) // vytvoreni switche pro prochazeni jako v menu
    {
        case 0:
            voda = 7; // pri tocení enkoderem proti smeru hodinových rucicek, když promenna voda ma hodnotu 0 vrati se na case 7,
            //aby uzivatel nemusel znovu vickrat tocit aby se dostal na konec
            break; //vyskoceni z case
        case 1:
            lcd.setCursor(4, 0); //nastaveni lcd kurzoru na 4 znak zleva na nultem radku
            lcd.print("Drink Maker"); //vypsani hlasky na lcd
            lcd.setCursor(2, 2); //nastaveni lcd kurzoru na 2 znak zleva na druhem radku
            lcd.print("<< Nadrz 1 >>"); //vypsani hlasky na lcd
            stiskTlacitka = digitalRead(SW); //cteni pinu SW
            if (stiskTlacitka == 0) //pokud je enkoder stisknut provede se nasledujici kod v podmince
            {
                probiha(); //zavolani funkce probiha()
                delay(100); //zpozdeni 100 ms
                cervenal(); //volani funkce cervenal()
                delay(500); //zpozdeni 500 ms
                nalevanil(2500); //provedeni funkce nalevanil() s casem 2500ms
                delay(500); //zpozdeni 500 ms
                modral(); //volani funkce modral()
                delay(500); //zpozdeni 500 ms
                hotovo(); //volani funkce hotovo()
                voda = 400000; //nastaveni promenne voda mimo funkcní rozmezi ve funkci loop()
                a = 1; //nastaveni promenne a pro funkcní rozmezi funkce NazvyMenu()
                menu = 1; //nastaveni promenne menu pro funkcní rozmezi funkce Menu()
                NazvyMenu(); //volani funkce NazvyMenu()
                Menu(); //volani funkce Menu()
            }
            break; //vyskoceni z case
    }
}

```

Obrázek 39 Ukázka části funkce void zavodneni()

2.4.13 void nulovanipozice()

Ve funkci „nulovanipozice()“ dochází jen k nastavení proměnné „n“ na hodnotu 0. Jelikož ve funkci „kalibrace()“ je tato proměnná velmi důležitá. Pokud proměnná „n“ má hodnotu 0 a následně je zavolána funkce „kalibrace()“, je provedeno posunutí pojezdu na začátek dráhy.

```
void nulovanipozice()
{
    n = 0; //nastaveni promenne n na hodnotu 0
}
```

Obrázek 40 Funkce void nulovanipozice()

2.4.14 void vpravo1()

Tato funkce slouží k posunu posuvníku směrem vpravo při pohledu z přední strany zařízení. Je pojmenována vpravo1(). Zajišťuje posun posuvníku o jednu nádrž. Ve zdrojovém kódu se nachází další funkce, které zajišťují větší posun posuvníku a liší se pouze počty kroků ve smyčce. Nazývají se vpravo2(), vpravo3(), vpravo4(), vpravo5() a vpravo6().

```
void vpravo1()
{
    digitalWrite(enable, LOW); //povoleni pinu enable pro moznost rizeni driveru
    delay(100); //zpozdeni 0,1s
    digitalWrite(dirPin, LOW); //nastaveni pinu pro otaceni ve smeru hod. rucicek
    for (int x = 1; x < 721; x++) //cyklus pro provedeni poctu kroku
    {
        digitalWrite(stepPin, HIGH); //nastaveni pinu na log 1
        delayMicroseconds(800); //zpozdeni 0,8ms pro rychlost kroku
        digitalWrite(stepPin, LOW); //nastaveni pinu na log 0
        delayMicroseconds(800); //zpozdeni 0,8ms pro rychlost kroku
    }
    delay(100); //zpozdeni 0,1s
    digitalWrite(enable, HIGH); //zakazani moznosti rizeni driveru
    delay(2000); //zpozdeni 2s
}
```

Obrázek 41 Funkce void vpravo1()

2.4.15 void vlevo1()

Funkce je obdobná jako u „vpravo1()“, liší se pouze změnou hodnoty na pinu „dirPin“, kde je nastavena její hodnota na „HIGH“, tedy na logickou 1. Další funkce pro větší posun směrem doleva jsou vlevo2(), vlevo3(), vlevo4(), vlevo5() a vlevo6().

```
void vlevo1()
{
    digitalWrite(enable, LOW);           //povoleni k rizeni driveru
    delay(100);                          //zpozdeni 0,1s
    digitalWrite(dirPin, HIGH);          //nastaveni pinu na log.1 pro toceni proti smeru hod.rucicek
    for (int x = 1; x < 721; x++)        //cyklus for s poctem kroku motoru
    {
        digitalWrite(stepPin, HIGH);     //nastaveni pinu na log. 1
        delayMicroseconds(800);          //zpozdeni pro rychlost kroku 0,8ms
        digitalWrite(stepPin, LOW);      //nastaveni pinu na log. 0
        delayMicroseconds(800);          //zpozdeni pro rychlost kroku 0,8ms
    }
    delay(100);                          //zpozdeni 0,1s
    digitalWrite(enable, HIGH);          //zakazani rizeni driveru
    delay(2000);                          //zpozdeni 2s
}
```

Obrázek 42 Funkce void vlevo1()

2.4.16 void cervena1()

Funkce slouží k rozsvícení RGB LED prstenu na červenou barvu pod první nádrž, než se posune pod nádrž posuvník s kelímkem. Pro ostatní rozsvícení RGB LED prstenu do červené barvy slouží funkce cervena2(), cervena3(), cervena4(), cervena5() a cervena6().

```
void cervena1()
{
    for (int i = 0; i < pocet24; i++)      // cyklus pro nastaveni barvy každé ledky
    {
        ledkal.setPixelColor(i, ledkal.Color(255, 0, 0)); // nastaveni cervene barvy ledky
    }
    ledkal.show();                        // rozsviceni 24 ledek
}
```

Obrázek 43 Funkce void cervena1()

2.4.17 void modra1()

Jedná se o stejnou funkci jako v případě funkce na rozsvícení RGB LED prstenu na červenou barvu. Dalšími funkcemi jsou stejnojmenné názvy pouze se změnou čísla modra2(), modra3(), modra4(), modra5() a modra6().

```
void modra1()
{
    for (int i = 0; i < pocet24; i++)      // cyklus pro nastaveni barvy každé ledky
    {
        ledkal.setPixelColor(i, ledkal.Color(0, 0, 255)); // LED1- modra
    }
    ledkal.show();                        // rozsviti se LED1
}
```

Obrázek 44 Funkce void modra1()

2.4.18 void nalevani1(int x)

Poslední ukázanou funkcí je „nalevani1(int x)“. Zde si stačí zvolit parametr funkce, kterým je čas nalévání. Jedná se o nalévání pod první nádrž. Dalšími jsou nalevani2(), nalevani3(), nalevani4(), nalevani5() a nalevani6().

```
void nalevani1(int x)
{
    digitalWrite(cerpadlol, LOW);    //spusteni cerpadla
    digitalWrite(ventill, LOW);     //spusteni elektromagnetickeho ventilu
    delay(x);                        //doba nalevani podle delky
    digitalWrite(ventill, HIGH);    //vypnuti ventilu
    digitalWrite(cerpadlol, HIGH);  //vypnuti elektromagnetickeho ventilu
    delay(1000);                    //zpozdeni 1s
}
```

Obrázek 45 Funkce void nalevani1(int x)

2.4.19 void Menu()

Jedná se o výchozí menu, kterým uživatel prochází při zapnutí zařízení. Zde má na výběr ze šesti nápojů. Ve funkci jsou už volány veškeré potřebné funkce k namíchání nápoje.

```
void Menu() {
    switch (menu)
    {
        case 0:
            a = 10;                //nastavena hodnota promenne pro nazvy menu - napoje
            menu = 10;             //nastavena hodnota promenne pro menu
            break;
        case 1:
            a = 1;                //nastavena hodnota promenne pro nazvy menu - napoje
            stiskTlacitka = digitalRead(SW); //cteni pinu tlacitka
            if (stiskTlacitka == 0) //podminka pokud je tlacitko stisknuto
            {
                probiha();        //volani funkce probiha()
                nulovaniPozice(); //volani funkce nulovaniPozice()
                kalibrace();       //volani funkce kalibrace()
                delay(2000);       //zpozdeni 2s
                cervena6();        //volani funkce cervena6() - rozsviceni pod nadrž 6 do cervene barvy
                delay(2000);       //zpozdeni 2s
                vlevo5();          //volani funkce vlevo5() - posunutí o 5 nadrž vlevo
                delay(500);        //zpozdeni 0,5s
                nalevani6(11000); //volani funkce nalevani6() - nalevani z nadrž 6
                delay(100);        //zpozdeni 0,1s
                modra6();          //volani funkce modra6() - rozsviceni pod nadrž 6 do modre barvy
                delay(500);        //zpozdeni 0,5s
                cervena2();        //volani funkce cervena2() - rozsviceni pod nadrž 2 do cervene barvy
                delay(1000);       //zpozdeni 1s
                vpravo4();         //volani funkce vpravo4() - posunutí o 4 nadrž vpravo
                delay(500);        //zpozdeni 0,5s
                nalevani2(2750); //volani funkce nalevani2() - nalevani z nadrž 2
                delay(100);        //zpozdeni 0,1s
                modra2();          //volani funkce modra2() - rozsviceni pod nadrž 6 do modre barvy
                delay(500);        //zpozdeni 0,5s
                vpravo1();         //volani funkce vpravo1() - posunutí o 1 nadrž vpravo
                delay(500);        //zpozdeni 0,5s
                odebrat();        //volani funkce odebrat()
            }
        break;
    }
}
```

Obrázek 46 Ukázka funkce void Menu()

2.4.20 void NazvyMenu()

Slouží pouze k vypisování hlášek na LCD displej. Je vykonávána s funkcí Menu().

```
void NazvyMenu()
{
    switch (a)
    {
        case 1:
            lcd.setCursor(4, 0);           //nastaveni kurzoru na 4 znak na 0 radku
            lcd.print("Drink Maker");      //vypis hlasky na LCD
            lcd.setCursor(2, 2);           //nastaveni kurzoru na 2 znak na 2 radku
            lcd.print("<< Napoj 1 >>");    //vypis hlasky na LCD
            break;
        case 2:
            lcd.setCursor(2, 2);           //nastaveni kurzoru na 2 znak na 2 radku
            lcd.print("<< Napoj 2 >>");    //vypis hlasky na LCD
            break;
        case 3:
            lcd.setCursor(2, 2);           //nastaveni kurzoru na 2 znak na 2 radku
            lcd.print("<< Napoj 3 >>");    //vypis hlasky na LCD
            break;
        case 4:
            lcd.setCursor(2, 2);           //nastaveni kurzoru na 2 znak na 2 radku
            lcd.print("<< Napoj 4 >>");    //vypis hlasky na LCD
            break;
        case 5:
            lcd.setCursor(2, 2);           //nastaveni kurzoru na 2 znak na 2 radku
            lcd.print("<< Napoj 5 >>");    //vypis hlasky na LCD
            break;
        case 6:
            lcd.setCursor(2, 2);           //nastaveni kurzoru na 2 znak na 2 radku
            lcd.print("<< Napoj 6 >>");    //vypis hlasky na LCD
            break;
        case 7:
            lcd.setCursor(2, 2);           //nastaveni kurzoru na 2 znak na 2 radku
            lcd.print("<< Nealko >>");     //vypis hlasky na LCD
            break;
        case 8:
            lcd.setCursor(2, 2);           //nastaveni kurzoru na 2 znak na 2 radku
            lcd.print("<< Alko >>");       //vypis hlasky na LCD
            break;
    }
}
```

Obrázek 47 Ukázka funkce void NazvyMenu()

2.4.21 void probiha()

Tato funkce je volána ve smyčce jako první, aby uživatel dostal informaci vypsáním na LCD a věděl, že jeho drink se již připravuje.

```
void probiha()
{
    lcd.clear();                          //vymazani LCD displeje
    lcd.setCursor(4, 0);                  //nastaveni kurzoru na 4 znak na 0 radku
    lcd.print("Drink Maker");            //vypis hlasky na LCD
    lcd.setCursor(4, 2);                  //nastaveni kurzoru na 4 znak na 2 radku
    lcd.print("Pripravujeme");          //vypis hlasky na LCD
}
```

Obrázek 48 Funkce void probiha()

2.4.22 void odebrat()

Funkce „odebrat()“ je používána na konci nalévání pro podání informace uživateli, že jeho nápoj je již připraven k odebrání, při odebrání musí potvrdit stisknutím tlačítka.

```
void odebrat()
{
  lcd.setCursor(4, 0);           //nastaveni kurzoru na 4 znak na 0 radku
  lcd.print("Drink Maker");     //vypis hlasky na LCD
  lcd.setCursor(3, 2);         //nastaveni kurzoru na 3 znak na 2 radku
  lcd.print("Odeberte napoj"); //vypis hlasky na LCD
  while (digitalRead(SW) != 0) //cekani na potvrzeni stisknutim tlacitka
  {
    menu = 12;                 //nastaveni mimo funkcní rozsah
  }
  delay(200);                 //zpozdeni 0,2s
}
```

Obrázek 49 Funkce void odebrat()

2.4.23 void hotovo()

Tato funkce se používá při zavodňování nádrží, kde po zavodnění je následně vypsána uživateli informace.

```
void hotovo()
{
  lcd.clear();                 //vymazani LCD displeje
  lcd.setCursor(4, 0);         //nastaveni kurzoru na 4 znak na 0 radku
  lcd.print("Drink Maker");   //vypis hlasky na LCD
  lcd.setCursor(4, 2);         //nastaveni kurzoru na 4 znak na 2 radku
  lcd.print("Dokonceno");     //vypis hlasky na LCD
  while (digitalRead(SW) != 0) //cekani na potvrzeni stisknutim tlacitka
  {
    menu = 12;                 //nastaveni mimo funkcní rozsah
  }
  delay(200);                 //zpozdeni 0,2s
}
```

Obrázek 50 Funkce void hotovo()

2.5 Kontrola funkcionality zařízení

Většina konstrukce zařízení byla zkonstruována ze dřeva. Elektronické součástky a komponenty byly převážně kupovány ze zahraničního obchodu. Většina z nich pochází z Číny. Důvodem koupi ze zahraničního trhu byla jeho cena. Zařízení je možné stále vylepšovat, například osazením senzoru pro snímání sklenice na posuvníku, měření objemu tekutin v nádržích nebo chlazení nádrží. Jedná se o funkční prototyp automatického míchače, který na českém trhu není obvyklý. V poslední části probíhalo na zařízení manuální testování. Jeho průběh byl následovný, v první fázi byl vybrán drink a uživatelem potvrzen jeho výběr stiskem tlačítka. Poté bylo spuštěno míchání drinku, které proběhlo správně. Tímto způsobem byly vyzkoušeny všechny možnosti v zařízení. Veškeré komponenty komunikují s řídicí jednotkou dle softwaru. Zařízení je připraveno k použití a je funkční.



Obrázek 51 Celkový pohled na zařízení



Obrázek 52 Vzhled menu na displeji

ZÁVĚR

Cílem této práce bylo navrhnout a zkonstruovat zařízení pro automatické míchání nápojů. V první fázi bylo důležité navrhnout základní konstrukci, od které probíhal vývoj. Stěny konstrukce jsou z dřevěné desky. Z horní části byly připevněny držáky na nádrže. V zadní části se nachází relé moduly s elektromagnetickými čerpadly a ventily. Řídící jednotka je umístěna společně se zdrojem napájení, LCD displejem a enkodérem na boční straně zařízení. Dále se na zařízení se nachází komponenty, které bylo nutné kompletně navrhnout a následně vyrobit. To se týká posuvníku se sklenicí a držáky na nádrže. Tyto nezbytné součásti jsou vytisknuty na 3D tiskárně. Po upevnění posuvníku byl pojezd připojen k ozubenému řemenu. Po kompletním sestrojení konstrukce probíhalo hardwarové zapojení dle blokového schématu. Poté probíhalo programování řídicí jednotky. Výsledkem práce je zařízení, které umí namíchat drink dle volby uživatele. Míchač je funkčně podobný jako zařízení, která již jsou na trhu dostupná. Celková cena přístroje se pohybuje okolo 12 tisíc korun. Jedná se o prototyp, jehož konstrukce je schopna dalšího vývoje popř. rozvoje.

SEZNAM POUŽITÉ LITERATURY

- [1] Barsys. *Thebarsys.com* [online]. [cit. 2020-08-01]. Dostupné z: <https://thebarsys.com/pages/technology>
- [2] The Bartesian. *Bartesian.com* [online]. [cit. 2020-08-01]. Dostupné z: <https://bartesian.com/products/the-bartesian>
- [3] Mixologiq. *Mixologiq.com* [online]. [cit. 2020-08-01]. Dostupné z: <https://www.mixologiq.com/en/cocktails-machine-automated-mixo-two/>
- [4] Somabar. *Somabar.com* [online]. [cit. 2020-08-01]. Dostupné z: <https://www.somabar.com/prosumer>
- [5] Drinkmaker. *Drinkworks.com* [online]. [cit. 2020-08-01]. Dostupné z: <https://www.drinkworks.com/shop/product/1040197/>
- [6] ARDUINO PLATFORMA. *Wikipedia.com* [online]. [cit. 2020-08-01]. Dostupné z: <https://cs.wikipedia.org/wiki/Arduino>
- [7] Typy desek Arduino. *Navody.arduino-shop.cz* [online]. [cit. 2020-08-01]. Dostupné z: <https://navody.arduino-shop.cz/technikuv-blog/napajeni-arduina.html>
- [8] Arduino Uno. *Arduino-shop.cz* [online]. [cit. 2020-08-02]. Dostupné z: https://arduino-shop.cz/arduino/974-eses-klon-arduino-uno-r3-precise.html?gclid=EAIaIQobChMI3MjtpJnb6gIVAp3VCh0v1gQAEAQYASABEgITIVD_BwE
- [9] Arduino Nano. *Conrad.cz* [online]. [cit. 2020-08-02]. Dostupné z: <https://www.conrad.cz/arduino-nano-65250-atmega328.k1172623>
- [10] Arduino Mini. *Arduino-shop.cz* [online]. [cit. 2020-08-02]. Dostupné z: <https://arduino-shop.cz/arduino/880-arduino-mini-atmega328p-5v-16m.html>
- [11] ARDUINO LEONARDO. *Cs.wikipedia.org* [online]. [cit. 2020-08-02]. Dostupné z: https://cs.wikipedia.org/wiki/Arduino_Leonardo
- [12] Arduino Leonardo. *Store.arduino.cc* [online]. [cit. 2020-08-02]. Dostupné z: <https://store.arduino.cc/arduino-leonardo-with-headers>
- [13] Arduino Micro. *Rpishop.cz* [online]. [cit. 2020-08-02]. Dostupné z: <https://rpishop.cz/arduino/2384-531-arduino-micro.html>
- [14] Arduino Micro. *Taillieu.info* [online]. [cit. 2020-08-02]. Dostupné z: <https://www.taillieu.info/index.php/internet-of-things/webcom-arduino/404-arduino-boards-pin-mapping>
- [15] Arduino Mega2560. *Laskarduino.cz* [online]. [cit. 2020-08-02]. Dostupné z: <https://www.laskarduino.cz/arduino-mega2560-klon/>

- [16] Arduino Mega2560. Conrad.cz [online]. [cit. 2020-08-02]. Dostupné z: <https://www.conrad.cz/programovatelná-deska-arduino-mega-2560.k191790>
- [17] ARDUINO LILYPAD. Laskarduino.cz [online]. [cit. 2020-08-02]. Dostupné z: <https://www.laskarduino.cz/arduino-lilypad--atmega328p--klon-100/>
- [18] Arduino LilyPad. Sparkfun.com [online]. [cit. 2020-08-02]. Dostupné z: <https://www.sparkfun.com/products/13342>
- [19] NOVÁK, Petr. Mobilní roboty: pohony, senzory, řízení. Praha: BEN - technická literatura, 2005. Robotika. ISBN 80-7300-141-1.
- [20] Relé modul 8 kanálů. Laskarduino.cz [online]. [cit. 2020-08-02]. Dostupné z: <https://arduino-shop.cz/arduino/879-arduino-rele-modul-8-kanalu-s-optickym-oddelenim.html>
- [21] Relé modul 8 kanálů. Laskarduino.cz [online]. [cit. 2020-08-02]. Dostupné z: https://www.laskarduino.cz/8-kanalu-rele-modul-5vdc-250vac-10a/?gclid=EAIaIQobChMI0dybrCHg6gIVGLd3Ch0gYAMCEAQYAIAABEGLRdPD_BwE
- [22] Krokový motor. Umax.cz [online]. [cit. 2020-08-02]. Dostupné z: <https://www.umax.cz/krokovy-motor-nema-sx17-1005lqcef-0-5nm/>
- [23] Krokový motor. Krokovemotory.cz [online]. [cit. 2020-08-02]. Dostupné z: <http://www.krokovemotory.cz/SX17-1005LQCEFDs.pdf>
- [24] Vývojová deska Arduino Mega2560. Laskarduino.cz [online]. [cit. 2020-08-02]. Dostupné z: https://www.laskarduino.cz/arduino-mega2560-klon/?gclid=EAIaIQobChMI95Hj7sbg6gIVkLh3Ch239gZmEAQYAYABEGI8tvD_BwE
- [25] Arduino Mega2560. Røge-shop.cz [online]. [cit. 2020-08-02]. Dostupné z: <http://roge-shop.cz/produkt/klon-arduino-mega-precise-eses/?add-to-cart=7348>
- [26] Motorový driver A4988. Laskarduino.cz [online]. [cit. 2020-08-02]. Dostupné z: https://www.laskarduino.cz/a4988-driver-pro-krokovemotory/?gclid=EAIaIQobChMItaH82sng6gIVicx3Ch274A6uEAQYAIABEGJY-vD_BwE
- [27] Motorový driver A4988. Futuretech3d.com [online]. [cit. 2020-08-02]. Dostupné z: <https://futuretech3d.com/electronics/3-a4988-stepper-motor-driver.html>
- [28] RGB LED prsten. Arduino-shop.cz [online]. [cit. 2020-08-02]. Dostupné z: https://arduino-shop.cz/arduino/7691-rgb-led-kruh-16-x-neopixel-ws2812b.html?gclid=EAIaIQobChMI36fJm7L96gIV0OR3Ch31-QUNEAQYASABEGLbQvD_BwE

- [29] RGB LED prsten. Makerfabs.com [online]. [cit. 2020-08-02]. Dostupné z: <https://www.makerfabs.com/ws2812b-rgb-led-16-bit-ring.html>
- [30] Rotační enkoder. Lastminuteengineers.com [online]. [cit. 2020-08-02]. Dostupné z: <https://lastminuteengineers.com/rotary-encoder-arduino-tutorial/>
- [31] Rotační enkoder. Teczium.net [online]. [cit. 2020-08-02]. Dostupné z: <https://teczium.net/product/rotary-encoder-module-ky-040/>
- [32] Elektromagnetický ventil. Hwpro.cz [online]. [cit. 2020-08-02]. Dostupné z: https://www.hwpro.cz/oc/index.php?route=product/product&product_id=531
- [33] Elektromagnetický ventil. *Arduino-shop.cz* [online]. [cit. 2020-08-02]. Dostupné z: https://arduino-shop.cz/arduino/1066-elektromagneticky-ventil-1-2.html?gclid=EAIAIQobChMI3q2muObr6gIVyPZRCh1fRgpVEAQYAIAiABEGKgEfD_BwE
- [34] Čerpadlo. *Arduino-shop.cz* [online]. [cit. 2020-08-02]. Dostupné z: <https://arduino-shop.cz/arduino/7521-vodni-čerpadlo-se-silentbloky-dc-6-12v-r385.html>
- [35] Čerpadlo. Indiamart.com [online]. [cit. 2020-08-02]. Dostupné z: <https://www.indiamart.com/proddetail/dc-water-pump-12v-r385-22210915312.html>
- [36] Zdroj napájení. Ledme.cz [online]. [cit. 2020-08-02]. Dostupné z: <https://ledme.cz/led-napajeci-zdroje/1105-napajeci-zdroj-480w-12v.html>
- [37] Zdroj napájení 480W. Electropeak.com [online]. [cit. 2020-08-02]. Dostupné z: <https://electropeak.com/power-switching-12v-10a>
- [38] LCD displej. *Arduino-shop.cz* [online]. [cit. 2020-08-02]. Dostupné z: <https://arduino-shop.cz/arduino/1421-eses-i2c-20x4-display-pro-jednodeskove-pocitace.html>
- [39] LCD Displej. Gmelectronic.com [online]. [cit. 2020-08-02]. Dostupné z: <https://www.gmelectronic.com/blue-lcd-display-20x4-i2c>
- [40] CATSOULIS, John. Designing embedded hardware. 2nd ed. Sebastopol: O'Reilly, 2005. ISBN 0596007558.
- [41] LADMAN, Josef. Elektronické konstrukce pro začátečníky. Praha: BEN - technická literatura, 2001. ISBN 80-730-0015-6.
- [42] PINKER, Jiří. Mikroprocesory a mikropočítače. Praha: BEN - technická literatura, 2004. ISBN 80-7300-110-1.
- [43] MARGOLIS, Michael, 2012. Arduino cookbook. 2nd ed. Sebastopol: O'Reilly Media. ISBN 978-1-449-31387-6.

SEZNAM POUŽITÝCH ZKRATEK A SYMBOLŮ

mm	Milimetr
GND	Uzemnění
g	Gram
V	Volt, jednotka elektrického napětí
AC	Střídavý proud
DC	Stejnoseměrný proud
SDA	přenos dat
SCL	přenos hodinového signálu
VCC	Napájecí napětí
Mhz	Megahertz - kmitočet
PWM	Pulzní modulová šířka
ml	Mililitr
LED	Světelná dioda
IDE	Vývojové prostředí
Nm	Newton metr, jednotka momentu síly
3D	trojrozměrný prostor
°	stupeň otočení - úhel
I2C	Sériová sběrnice, dvou vodičové datové propojení
CLK	pin pro hodinový signál
DT	pin pro data
USB	Univerzální sériová sběrnice
A	Ampér, jednotka proudu

SEZNAM OBRÁZKŮ

Obrázek 1 Barsys 2.0 [1].....	12
Obrázek 2 The Bartsian [2].....	13
Obrázek 3 Mixologiq [3].....	13
Obrázek 4 Somabar [4]	14
Obrázek 5 Drinkmaker [5].....	15
Obrázek 6 Arduino Uno [8].....	16
Obrázek 7 Arduino Nano [9].....	16
Obrázek 8 Arduino Mini [10].....	17
Obrázek 9 Arduino Leonardo [12].....	18
Obrázek 10 Arduino Micro [14]	18
Obrázek 11 Arduino Mega2560 [16]	19
Obrázek 12 Arduino LilyPad [18]	19
Obrázek 13 Blokové schéma řídicího systému.....	22
Obrázek 14 Mechanický návrh zařízení.....	22
Obrázek 15 Relé modul 8 kanálů [21].....	23
Obrázek 16 Krokový motor [23].....	23
Obrázek 17 Arduino Mega2560 [25]	24
Obrázek 18 Motorový driver A4988 [27].....	24
Obrázek 19 RGB LED prsten [29].....	25
Obrázek 20 Rotační enkoder [31].....	25
Obrázek 21 Posuvník sklenice.....	26
Obrázek 22 Nádrž.....	26
Obrázek 23 Držák na nádrž.....	27
Obrázek 24 Elektromagnetický ventil [33].....	27
Obrázek 25 Čerpadlo [35]	28
Obrázek 26 Zdroj napájení 480W [37].....	28
Obrázek 27 LCD Displej [39].....	29
Obrázek 28 Ukázka vývojového diagramu.....	33
Obrázek 29 Ukázka funkce void setup()	33
Obrázek 30 Funkce void loop()	34
Obrázek 31 Funkce void enkoder().....	35
Obrázek 32 Funkce void kalibrace().....	36
Obrázek 33 Funkce void led()	36
Obrázek 34 Ukázka funkce alko()	37
Obrázek 35 Ukázka funkce void nealko()	38
Obrázek 36 Ukázka funkce void VelkeDavky().....	38
Obrázek 37 Ukázka funkce void MaleDavky()	39
Obrázek 38 Ukázka funkce void cistení()	40

Obrázek 39 Ukázka části funkce void zavodneni().....	40
Obrázek 40 Funkce void nulovanipozice().....	41
Obrázek 41 Funkce void vpravo1().....	41
Obrázek 42 Funkce void vlevo1().....	42
Obrázek 43 Funkce void cervena1()	42
Obrázek 44 Funkce void modra1()	42
Obrázek 45 Funkce void nalevani1(int x)	43
Obrázek 46 Ukázka funkce void Menu()	43
Obrázek 47 Ukázka funkce void NazvyMenu()	44
Obrázek 48 Funkce void probiha()	44
Obrázek 49 Funkce void odebrat().....	45
Obrázek 50 Funkce void hotovo()	45
Obrázek 51 Celkový pohled na zařízení.....	46
Obrázek 52 Vzhled menu na displeji	46

SEZNAM TABULEK

Tabulka 1 Zapojení součástek.....	30
Tabulka 2 Zapojení součástek.....	31

SEZNAM PŘÍLOH

PŘÍLOHA P I: CD SE ZDROJOVÝM KÓDEM