


Řídicí systém automatické udírny

Jan Smékal

Bakalářská práce
2023

 Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav automatizace a řídicí techniky

Akademický rok: 2022/2023

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: Jan Smékal
Osobní číslo: A20325
Studijní program: B0714A150006 Aplikovaná informatika v průmyslové automatizaci
Specializace: Inteligentní systémy s roboty
Forma studia: Prezenční
Téma práce: Řídicí systém automatické udírny
Téma práce anglicky: Automatic Smokehouse Control System

Zásady pro vypracování

1. Zpracujte literární rešerši na dané téma.
2. Vyberte vhodné typy snímačů, akčních členů a řídicího mikropočítače pro danou aplikaci.
3. Proveďte hardwarový návrh řídicí jednotky udírny na bázi vybraného mikropočítače.
4. Prakticky realizujte řídicí jednotku.
5. Vytvořte obslužné programové vybavení jednotky.
6. Ověřte správnou funkci řídicí jednotky udírny.

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

Seznam doporučené literatury:

1. CAMERON, Neil. Electronics projects with the ESP8266 and ESP32: building web pages, applications, and Wifi enabled devices. Berkeley, CA: Apress L. P., 2021, 1 online zdroj (703 stran). Technology in action. Dostupné z: doi:9781484263365.
2. FROHN, Manfred. Elektronika: polovodičové součástky a základní zapojení. Praha: BEN – technická literatura, 2006. ISBN 8073001233.
3. GEORGE, Damien P. a Paul SOKOLOVSKY. MicroPython documentation [online]. 2022 [cit. 2022-11-24]. Dostupné z: <https://docs.micropython.org>.
4. JURÁNEK, Antonín a Miroslav HRABOVSKÝ. EAGLE pro začátečníky /: uživatelská a referenční příručka .. 2. vydání. Praha: BEN – technická literatura, 2007, 191 s. ISBN 80-730-0213-2.
5. ÜNSALAN, Cem, Duygun E. BARKANA a H. Deniz GURHAN. Embedded digital control with microcontrollers: implementation with C and Python. Hoboken, NJ: Wiley, 2021, 1 online resource (371 pages). Dostupné z: doi:9781119576600.
6. VLK, Václav. Udírny: konstrukce, stavba, uzení. Grada, 2007, 1 online zdroj (112 stran). ISBN 978-80-247-8318-5. Dostupné také z: <https://www.bookport.cz/AccountSaml/SignIn/?idp=https://shibboleth.utb.cz/idp/shibboleth&returnUrl=/kniha/udirny-1780>.

Vedoucí bakalářské práce: **Ing. Petr Dostálek, Ph.D.**
Ústav automatizace a řídicí techniky

Konzultant bakalářské práce: **Ing. Tomáš Dulík, Ph.D.**
Ústav informatiky a umělé inteligence

Datum zadání bakalářské práce: **16. prosince 2022**
Termín odevzdání bakalářské práce: **24. května 2023**

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



prof. Ing. Vladimír Vašek, CSc. v.r.
ředitel ústavu

Ve Zlíně dne 12. prosince 2022

Prohlašuji, že

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

Prohlašuji,

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

Ve Zlíně, dne 16.5.2023

Jan Smékal, v.r.
podpis studenta

ABSTRAKT

Bakalářská práce se zabývá návrhem a praktickou realizací řídicího systému pro automatickou udírnu. V rámci této práce je popsáno použití řídicích systémů pro automatizaci procesu uzení za účelem zefektivnění a ulehčení práce. V teoretické části je stručný popis zabývající se výběrem vhodného typu snímačů, akčních členů a mikropočítače pro danou aplikaci. V praktické části jsou popsány jednotlivé praktické úkony realizace řídicí jednotky a obslužné programové vybavení jednotky.

Klíčová slova: mikropočítač, automatická udírna, řídicí systém

ABSTRACT

The bachelor thesis deals with the design and practical implementation of a control system for an automatic smokehouse. The use of control systems to automate the smoking process in order to make the work more efficient and easier is described in this thesis. In the theoretical part, a brief description is given dealing with the selection of the appropriate type of sensors, actuators and microcomputer for the application. In the practical part, the individual practical tasks of implementing the control unit and the operating software of the unit are described.

Keywords: microcomputer, automatic smokehouse, control system

Tímto bych rád poděkoval vedoucímu bakalářské práce Ing. Petru Dostálkovi, Ph.D. za odborné vedení, cenné rady a řadu připomínek, poskytnutí literatury a svůj čas při pomoci řešení problémů. Také bych chtěl poděkovat Ing. Tomáši Dulíkovi, Ph.D. jakožto konzultantovi za pomoc při nejasnostech v rámci programování a výběru mikropočítače.

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

OBSAH

| | |
|---|-----------|
| ÚVOD..... | 9 |
| I TEORETICKÁ ČÁST | 10 |
| 1 AUTOMATICKÉ UDÍRNY | 11 |
| 1.1 TYPY AUTOMATICKÝCH UDÍREN..... | 11 |
| 2 VÝBĚR MIKROPOČÍTAČE | 13 |
| 2.1 POROVNÁNÍ MIKROPOČÍTAČŮ..... | 14 |
| 2.2 FINÁLNÍ VÝBĚR | 14 |
| 3 ESP32 DEVKIT V1 | 15 |
| 3.1 TECHNICKÉ PARAMETRY | 16 |
| 4.3 VÝBĚR RELÉ/STYKAČE | 23 |
| II PRAKTICKÁ ČÁST | 34 |
| 7 ZÁKLADNÍ KONSTRUKCE | 35 |
| 7.1 KONSTRUKCE | 35 |
| 7.1.1 Základna..... | 35 |
| 7.1.2 Generátor kouře..... | 36 |
| 7.1.3 Samotná udírna..... | 37 |
| 7.2 3D MODEL | 38 |
| 8 ELEKTROINSTALACE | 40 |
| 8.1 SILOVÁ ČÁST | 40 |
| 8.2 ŘÍDICÍ ČÁST..... | 42 |
| 9 NÁVRH ŘÍDICÍ JEDNOTKY | 45 |
| 9.1 POUŽITÉ SOUČÁSTKY | 45 |
| 9.2 NÁVRH SCHÉMATU ZAPOJENÍ | 46 |
| 9.3 NÁVRH DESKY PLOŠNÝCH SPOJŮ | 48 |
| 9.4 REALIZACE DESKY PLOŠNÝCH SPOJŮ..... | 48 |
| 10 PROGRAMOVÉ ŘEŠENÍ | 50 |
| 10.1 PROGRAM..... | 50 |
| 11 OVĚŘENÍ FUNKCE UDÍRNY | 60 |
| 11.1 OVĚŘENÍ REGULACE TEPLoty | 60 |
| 11.2 OVĚŘENÍ SPUŠTĚNÍ UDÍRNY | 61 |
| ZÁVĚR | 63 |
| SEZNAM POUŽITÉ LITERATURY..... | 64 |
| SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK | 69 |
| SEZNAM OBRÁZKŮ | 71 |

| | |
|----------------------------|-----------|
| SEZNAM TABULEK..... | 72 |
| SEZNAM PŘÍLOH..... | 73 |

ÚVOD

Uzení jako takové patří k naší kultuře už velmi dlouhou dobu a za tu dobu se z něj stalo umění, které se jako každý jiný technologický postup zdokonaluje a stále mění i když základní principy zůstávají stejné.

Uzení je velice zdoluhavý a časově náročný proces obzvláště v domácích podmínkách a zde se nabízí možnost spojit základní principy uzení s moderními technologiemi dokonce i v domácích podmínkách. V dnešní době firmy zabývající se zpracováním masa a jeho uzení využívají moderní automatické udírny, které stojí nemalé peníze a jsou náročné jak na provoz, tak na údržbu a navzdory tomu se v dnešní době vytrácí základní principy uzení, které střídá chemické uzení pomocí chemie a uzení v páře. Naším cílem je postavit zařízení, které bude postavené na základních principech uzení, ale zefektivní a ulehčí nám práci. Dnešní trh nabízí širokou škálu automatických udíren jak pro domácí, tak pro komerční použití. Většina těchto udíren, avšak nefungují tak dokonale a uživatelsky přívětivě. Uživatel stále musí znát základní principy a co a jak který typ masa vyžaduje. Naše řešení je přizpůsobeno tak aby udírnu mohl ovládat kdokoliv bez znalostí procesu, avšak bez hrozby špatného zpracování masného výrobku.

Kdysi se mikropočítače využívali na ty nejjednodušší logické úlohy, ale v dnešní době máme tak výkonné mikropočítače, které mají široké spektrum využití a můžeme je použít i pro ty nejnáročnější úlohy, některé mikropočítače dokonce obsahují vlastní operační systém jako Linux apod... Dnešní mikropočítače se využívají hojně v inteligentních domácnostech pro měření teploty, ovládání osvětlení, termostaty pro podlahové topení, bezdrátové sítě atd. možnosti jsou téměř nekonečné, záleží jen na nás. Jedná se o dostupnou alternativu oproti výkonnějším PLC, které jsou dominantní v oblasti Průmyslu, avšak cenový rozdíl je výrazný.

I. TEORETICKÁ ČÁST

1 AUTOMATICKÉ UDÍRNY

V dnešní době automatické udírny představují technologii, která umožňuje automatizaci procesu uzení potravin. Tyto zařízení jsou navrženy tak aby udržovaly konstantní teplotu a vlhkost, což je klíčové pro správný proces uzení. Automatické udírny mohou mít různé funkce a vlastnosti, které se mohou od sebe lišit jak v zavidlosti na konkrétním modelu, tak na výrobci. V této části si některé druhy blíže představíme.

1.1 Typy automatických udíren

Samozřejmě je dnes trh plný různých typů automatických udíren, které se většinou od sebe liší druhem použité technologie. Mezi nejčastější typy patří elektrické a plynové, multifunkční udírny a izolované udírny.

Je důležité si také uvědomit některé věci a to například, jaké množství chcete udit nebo kolik máte místa na její případné umístění. [1]

1.1.1 Elektrické automatické udírny

Elektrickou automatickou udírnu si můžeme představit jako troubu, která je navržena tak aby automaticky generovala trvale kouř. Kouř v udírně je poté zahříván topným tělesem umístěné vně udírny. V rámci této rešerše byl zvolen model od významného výrobce v tomto odvětví Bradley Smoker 4. Tato udírna je dodávaná jako celek s automatickým generátorem kouře, který jako palivo používá speciální udicí brikety. Tento model však funguje tím způsobem, že si na ovládacím panelu nastavíme požadovanou teplotu, délku trvání celého procesu a dobu generování kouře, tímto nastavením bude udírna regulovat teplotu. Avšak tento model není nikterak prostorný a je určený spíše na občasné uzení omezeného množství potravin. Cena takové udírny potom činí 29 990 Kč. [2]

1.1.2 Plynové automatické udírny

Jak už název napovídá plynové automatické udírny využívají pro generování tepla plyn. Avšak co je pro tento druh udíren typické je, že nemají automatický regulátor teploty a tím neumožňují nastavit požadovanou teplotu, a tedy se nehodí pro všechny metod uzení.

Z tohoto důvodu jsou plynové udírny nejméně oblíbené a téměř vůbec se neprodávají. [3]

1.1.3 Multifunkční udírny

Zde tento typ je nejvyžívanější. Poskytuje širokou škálu využití. Můžete v ní udit různé potraviny, jako je maso na různé způsoby, sýry atd. jako zdroj tepla a kouře slouží dřevo, dřevěná drť nebo piliny. Některé modely jsou vybaveny topnou spirálou. [1]

1.1.4 Izolované udírny

Většina typů udíren je omezena na použití v období od jara do podzimu. Když teploty klesnou pod 5 °C, jejich použití se obecně nedoporučuje. Avšak pokud pořídíme udírnu, která má izolaci, je možné ji využívat celoročně bez sebemenších problémů. Samozřejmě cena je vyšší než u modelů, které neobsahují izolaci. [1]

2 VÝBĚR MIKROPOČÍTAČE

Dnešní moderní doba nám nabízí mnoho možností, co se ohledu výběru vhodného mikropočítače pro naše účely týče od mikropočítačů sloužícím k jednoduchým úlohám až po mikropočítače, které mají dokonce svůj vlastní operační systém např. Raspberry Pi.

Naším cílem je vybrat mikropočítač tak aby nám nabízel co nejlepší jak poměr cena/výkon tak aby měl velký rozsah využití a funkcí za předpokladu přívětivého programového ovládaní. Mimo jednoduchou implementace je zapotřebí také zhodnotit podmínky ve kterých bude mikropočítač pracovat, a tudíž vliv vnějšího prostředí na samotný hardware. Jeden z požadavků pro naše řešení bylo, aby danou udírnu bylo možné ovládat dálkově přes webové rozhraní, a tudíž se co nejvíce vyhnout manuálnímu hardwaru, kterými jsou ovládací tlačítka, na který by mohly působit vnější vlivy. V rámci modernizace bývá dnes již standartním požadavkem vzdálené ovládaní zařízení, pokud je to možné, šetří to nemalé náklady na obsluhu, která musí být přítomná u zařízení i náklady za další hardware potřebný k realizaci ovládaní. Pro potřebu návrhu řízení byl vybrán mikropočítač ESP32 Devkit V1 od firmy Espressif Systems.

2.1 Porovnání mikropočítačů

Pro potřebu návrhu řízení byly brány v úvahu celkem dva odlišné druhy mikropočítačů, a tedy ESP32 Devkit V1 a Arduino UNO+WiFi ATmega328P+ESP8266. Jejich výhody a nevýhody jsou popsány v tabulce níže.

Tabulka 1: Porovnání ESP32 a Arduino UNO+WiFi [4] [5]

| Vlastnost | ESP32 Devkit V1 | Arduino UNO+WiFi |
|-----------|------------------------------------|----------------------------------|
| Výhody | Vyšší výkon, integrovaný Bluetooth | Široká komunitní podpora |
| | Větší paměť | Jednodušší programování |
| | Více digitálních I/O pinů | Nižší cena |
| Nevýhody | Vyšší cena | Nižší výkon |
| | Složitější programování | Absence Bluetooth |
| | Méně knihoven | Menší paměť |
| | Nekonzistentní kvalita desek | Menší počet digitálních I/O pinů |

2.2 Finální výběr

V rámci porovnání mikropočítačů s ohledem na cenu, využití, podpory webového rozhraní a fungování v pracovním prostředí i za nepříznivých teplotních podmínek a diskusi s konzultantem byl vybrán mikropočítač ESP32 Devkit V1. Zařízení odpovídá veškerým požadavkům, které jsou stanoveny na základě návrhu. Mikropočítač Arduino UNO+WiFi má nižší výkon, absence integrovaného Bluetooth, menší množství paměti, menší počet analogových vstupů a digitálních I/O pinů. Tyto faktory mohou ovlivnit schopnost Arduino UNO+WiFi správně a efektivně řídit automatickou udírnu a zajistit plynulou komunikaci přes webový server.

3 ESP32 DEVKIT V1

ESP32 DevKit V1 je výkonný a univerzální mikropočítač určený pro širokou škálu aplikací. Termín ESP32 se obvykle používá k označení čipu ESP32, ale také se ho používá k označení vývojových desek, které mají tento čip zabudovaný. Samostatné použití čipu ESP32 pro výukové účely, testování a prototypování není praktické, protože vyžaduje připojení dalších obvodů pro napájení a programování, což může být komplikované. Většinou se tedy spíše používají vývojové desky s čipem ESP32, které jsou vybaveny potřebnými obvody pro napájení a programování, připojení k počítači a piny pro připojení periférií. Tyto desky jsou také vybaveny vestavěným napájením a ovládním LED diod, anténou pro Wi-Fi signál a dalšími užitečnými funkcemi. [6]



Obrázek 1: ESP32 DevKit V1 [7]

3.1 Technické parametry

Pro souhrn technický parametrů nám poslouží tabulka, kde jsou znázorněny jen ty nejdůležitější parametry, které má jinak ESP32 velké množství.

Tabulka 2: Technické parametry ESP32 [4]

| Parametry | Popis |
|-----------------|--------------------------|
| Procesor | 2x 160 MHz Tensilica LX6 |
| Paměť RAM | 520 KB |
| Paměť flash | 4 MB |
| WiFi | 802.11 b/g/n |
| Bluetooth | V4.2 BR/EDR and BLE |
| Počet GPIO pinů | 34 |
| ADC | 12-bitový |
| PWM | 16 kanálů |
| Napájení | 2,2-3,6V DC |
| Rozměry | 18 x 22,5 mm |

3.2 Funkce a vlastnosti

ESP32 je výkonný a univerzální modul MCU s podporou Wi-Fi, Bluetooth a Bluetooth Low Energy, který je určen pro širokou škálu aplikací. Jádrem modulu je čip ESP32-D0WDQ6 s dvěma individuálně ovladatelnými CPU jádry a nastavitelnou taktovací frekvencí 80 MHz až 240 MHz. Modul má také koprocesor s nízkou spotřebou energie. Integrovanými periferními zařízeními jsou např. dotykové senzory, Hallovy senzory, SD karty, Ethernet, vysokorychlostní SPI, UART, I²S a I²C. Kombinace technologií Bluetooth, Bluetooth LE a Wi-Fi umožňuje široké využití, zatímco nízký spánkový proud 5 μ A umožňuje použití pro napájení z baterií. Modul podporuje rychlost přenosu dat až 150 Mb/s a vysoký výstupní výkon 20 dBm na anténě. FreeRTOS s LwIP je voleným operačním systémem, integrovaný je také TLS 1.2 s hardwarovou akcelerací a podporována je také bezpečná aktualizace OTA. [4]

3.3 Aplikace ESP32

ESP32 je díky své univerzálnosti ideální pro použití v mnoha aplikacích ať už pro řízení automatické udírny tak i pro chytrou domácnost, pro studijní účely nebo jako jednoduchý webový server. V případě komunity začíná být tento mikropočítač více oblíbený a využívaný pro stále více projektu, hlavně díky své dostupnosti i co se teď týče nedostatku čipů nebyl s koupí žádný problém na rozdíl od konkurenčního Arduina nebo Raspberry Pi. Na internetu můžeme najít širokou škálu projektu a možnosti využití od jednoduchých až po náročné aplikace. [8]

3.4 Způsob programování

Při výběru mikropočítače ESP32 hrála důležitou roli možnost programování v jazyce MicroPython. Tento jazyk byl zvolen jako výchozí, avšak ESP32 nabízí i další programovací možnosti, jako jsou Arduino C/C++, Espressif IDF, JavaScript a DVA. Nicméně, MicroPython byl vybrán z důvodu své jednoduchosti a uživatelské přívětivosti při programování webového rozhraní. V rámci programování poskytuje MicroPython snadno použitelné funkce a zjednodušuje vývoj webových aplikací pro ESP32. Tento jazyk umožňuje rychlejší vývoj a úpravy kódu, což je pro výzkumníky a vývojáře klíčové pro dosažení efektivity a efektivního využití mikropočítače. [8]

3.5 Historie

ESP32 je nástupce modelové řady mikropočítačů ESP8266, který má oproti svému předchůdci několik nových funkcí. [] Čip byl mezi anglicky mluvícími výrobci zpopularizován v srpnu 2014 prostřednictvím modulu ESP-01 vyráběného externím výrobcem Ai-Thinker. Tento malý modul umožňuje mikropočítačům připojovat se k sítím Wi-Fi a navazovat jednoduchá spojení TCP/IP pomocí příkazů Hayes. Z počátku však k tomuto čipu a příkazům, neexistovala téměř žádná dokumentace. Každopádně z důvodu nízké ceny a skutečnosti, že na modulu bylo jen velmi málo externích komponentů, což odkazovalo na to, že by modul mohl být vyroben s nízkými náklady, přilákaly mnoho vývojářů, kteří zkoumali modul, software a také vytvářeli dokumentaci. [9] [8]

3.6 Zapojení vývodu ESP32 GPIO

ESP32 disponuje 48 piny s mnoha funkcemi. Nicméně, na všech vývojových deskách pro ESP32 nejsou všechny piny viditelné a některá z nich by neměly být používány. Naše vývojová deska ESP32 obsahuje 36 vyvedených GPIO pinů, které jsou určeny k připojení periférií. [8]

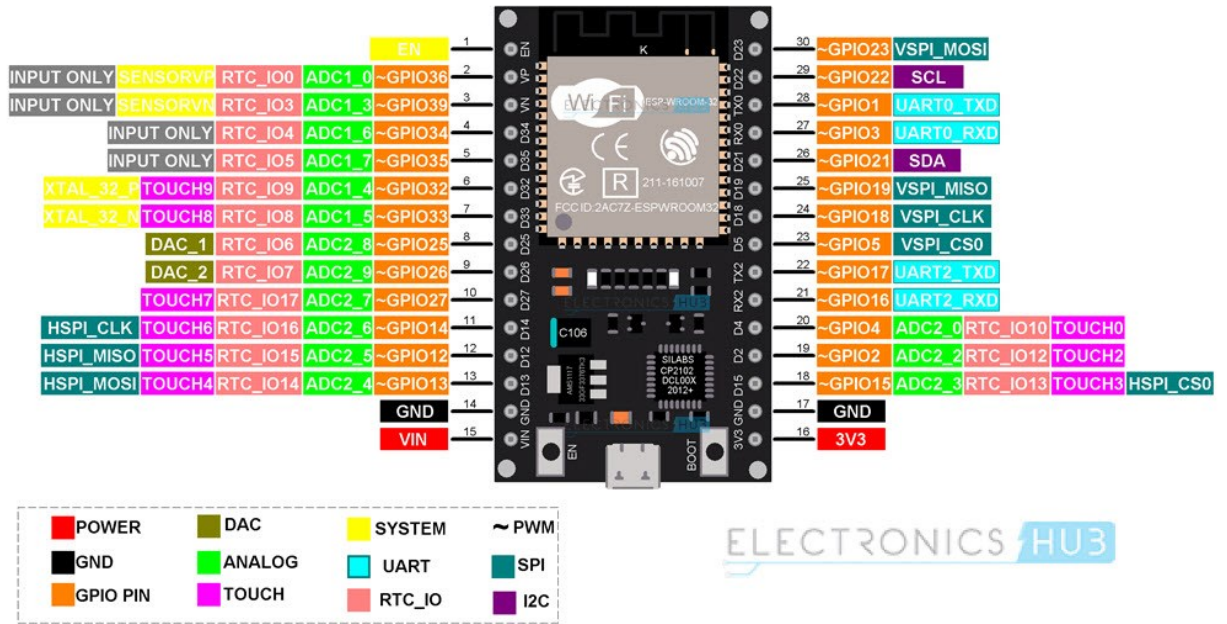
3.6.1 Napájení vývojové desky a periférií

Na všech vývojových deskách jsou obvykle dostupné napájecí piny jako 3V3, GND a VIN. Tyto piny lze využít k napájení desky, pokud není napájení zajištěno přes port USB anebo k napájení pro připojené periférie. 3V3 pin je schopný poskytnout až 500 mA a VIN pin který slouží pro externí napájení může přijímat napětí v rozmezí od 5 V až 12 V, zdroj by měl být schopný dodat alespoň 500 mA proudu, ale doporučuje se 1 A [8]

3.6.2 Výstupní vstupní piny (GPIOs)

Všechny piny GPIO mají definované své číslo, které byste měli používat pro označování. ESP32 umožňuje určit, které piny budou fungovat jako UART, I2C nebo SPI, a toho můžeme dosáhnout odpovídajícím přizpůsobením kódu.

Čip ESP32 je vybaven funkcí multiplexování, která umožňuje přiřadit jednomu pinu více funkcí. Pokud není funkce pinu programově specifikováno, budou piny automaticky nakonfigurovány podle výchozího nastavení. [8]



Obrázek 2: ESP32 DEVKIT V1 vývody. [7]

4 PERIFERIE PRO ESP32

Pro správnou funkčnost je důležité vybrat správné typy snímačů a akčních členů, které budou zajišťovat řízení dané udírny. Pro naše zadání bude potřeba vybrat vhodný teplotní snímač pro kontrolu teploty uvnitř udírny a také relé, která budou ovládat napájení topných spirál. Do této části samozřejmě zakomponujeme i prvky pro bezpečnost a napájení.

4.1 Výběr teplotního snímače

Správný výběr teplotního snímače je jedna z nejdůležitějších věcí pro efektivní řízení, co s největší přesností. Požadavky byly stanoveny na co nejjednodušší instalaci, programovatelnost s co nejvyšší přesností a spolehlivostí. V rámci širší rešerše byly v úvahu brány čtyři druhy teplotních snímačů, které se od sebe liší jak použitou technologií měření, zapojením tak i cenou. Mezi posuzovanými snímači byl DS18B20, DHT11, DHT22, PT100, PT1000, Thermistor NTC 10k Ω B3950.

Všechny čidla byla vyhodnocena na základě jejich vlastností. Základní parametry byly vypsány v tabulce níže. Jak je již patrné čidla PT100, PT1000 a Thermistor NTC 10k Ω B3950 byly nakonec vyřazeny hned na začátku výběru z důvodu analogového rozhraní, a tudíž potřeby převodníku např. modulu MAX31865, což je další modul navíc, který nám nepřinese větší výhodu při regulaci. Naše požadavky nejsou kladeny na tak velkou přesnost, které by nám tato čidla přinesla. [10] [11] [12] [13] [14] [15] [16]

Tabulka 3: Parametry jednotlivých čidel

| Teplotní snímač | Typ čidla | Rozsah teplot | Přesnost | Rozhraní | Požadavky na připojení |
|---------------------------|-----------|--------------------|----------------|-----------|-------------------------------|
| DS18B20 | Digitální | -55 °C až +120 °C | ±0,5 °C | OneWire | 1 datový pin, napájení, GND |
| DHT11 | Digitální | 0 °C až +50 °C | ±2 °C | OneWire | 1 datový pin, napájení, GND |
| DHT22 | Digitální | -40 °C až ±125 °C | ±0,5 °C | OneWire | 1 datový pin, napájení, GND |
| PT100 | RTD | -200 °C až +850 °C | ±0,15 °C | Analogový | MAX31865 modul, napájení, GND |
| PT1000 | RTD | -200 °C až +850 °C | ±0,15 °C | Analogový | MAX31865 modul, napájení, GND |
| Thermistor NTC 10kΩ B3950 | Analogový | -30 °C až +125 °C | ±1 °C až ±5 °C | Analogový | Odporový dělič, napájení, GND |

4.2 Teplotní snímač DS18B20

Pro měření teploty bylo vybrán snímač teploty DS18B20 ve voděodolné verzi.

Snímač teploty DS18B20 je digitální snímač teploty, který ke komunikaci s ESP32 využívá pouze jednu datovou linku. Tato vlastnost je známá jako „jednovodičový“ systém.

Tohle čidlo může být napájeno jak z externího zdroje nebo ze samotné linky, což se nazývá „parazitní režim“. V parazitním režimu není externí zdroj nutný, protože senzor napájíme přímo z datové linky.

Každý snímač DS18B20 má odlišný sériový kód o délce 64 bitů. Tato vlastnost umožňuje připojit k jednomu datovému vodiči více čidel. Proto můžeme získávat údaje o teplotě z více čidel pomocí jediného pinu. [17]



Obrázek 3: Teplotní snímač DS18B20 ve voděodolné verzi [18]

4.2.1 Specifikace

Tabulka 4: Specifikace teplotního senzoru DS18B20 [17]

| Parametry | Popis |
|--------------------------|------------------------|
| Komunikace | Jednovodičová sběrnice |
| Napájení | 3,0 V až 5,5 V |
| Rozsah provozních teplot | -55 °C až +125 °C |
| Přesnost | +/-0,5 °C |

4.3 Výběr relé/stykače

Pro naše potřeby je při výběru stykače nebo relé zohlednit několik faktorů jako je napětí a proud jakož i charakter aplikace. Jeden z nejdůležitějších prvků, který je třeba zohlednit je proudová zatížitelnost, topná spirála má výkon 3 kW a je tedy potřeba spočítat proud, který relé nebo stykač bude schopný bezpečně přenášet. Výpočet bude vycházet ze vzorce:

$$(1) I = \frac{P}{U} = \frac{3000 \text{ W}}{230 \text{ V}} = 13,04 \text{ A.}$$

Jak už bylo řečeno topná spirála má výkon 3 kW, přívodní napájení bude na napětí 230 V a proud bude 13,04 A. Z toho vyplývá, že jmenovitý proud relé nebo stykače by měl být vyšší než 13 A.

Samozřejmě existuje mnoho typů relé a stykačů, a každý z nich má své vlastní specifické charakteristiky. V našem případě připadá v úvahu elektromagnetické relé, které se skládá z cívky, která generuje magnetické pole, jež působí na kontakty a umožňuje jejich přepínání, nebo polovodičová Solid state relé, které jsou vhodná pro rychlejší spínání a delší životnost, protože neobsahují žádné pohyblivé součásti, které se mohou časem opotřebovat. Avšak Solid state relé mohou vyžadovat chlazení například hliníkový chladič při vyšších proudcích z důvodu výkonové ztráty. Pokud porovnáme klasické a polovodičové relé, zjistíme, že větší ovládací výkon je obvykle charakteristický pro klasické relé, zatímco výkonové ztráty na pracovních kontaktech bývají menší. Jako další připadají v úvahu stykače, jakožto elektromagneticky řízené spínače, které se často využívají jak v průmyslových aplikacích například pro řízení elektromotorů, tak i pro domácí použití. Rozdíl mezi stykačem a relé je takový, že stykače jsou navrženy pro zapínání a vypínání výkonových zařízení, avšak z důvodu mechanických částí jsou náchylnější na opotřebení. [19] [20] [21] [22] [23]

Tabulka 5: Porovnání spínacích prvků

| | | | |
|----------------|--|--|--|
| Spínací prvek | Solid state relé (SSR) | El. Magnetické relé (EMR) | Stykač |
| Princip | Elektronické prvky, jako jsou tranzistory, se používají k otevírání a zavírání obvodu. | Elektromagnetické cívky se používají k otevírání a zavírání obvodu. | Velké elektromagnetické relé, které se používá k ovládnání velkých proudů. |
| Spínání | Může rychle a často spínat, bez mechanického opotřebení. | Může být omezeno rychlostí a četností spínání kvůli mechanickému opotřebení. | Může být omezeno rychlostí a četností spínání kvůli mechanickému opotřebení, ale je konstruován pro vyšší proudy a napětí než EMR. |
| Odolnost | Neobsahuje pohyblivé části, takže se neopotřebuje | Obsahuje pohyblivé části, které se mohou opotřebovat. | Obsahuje pohyblivé části, které se mohou opotřebovat, ale je konstruován pro delší životnost než EMR. |
| Proud a napětí | Může zvládnout různé hodnoty proudu a napětí, ale může vyžadovat chlazení při vyšších hodnotách. | Může zvládnout různé hodnoty proudu a napětí, ale obecně je vhodnější pro nižší hodnoty. | Navržený k ovládnání velkých proudů a napětí. |

Je důležité si uvědomit, že v tomto případě bude probíhat regulace teploty na požadovanou hodnotu prostřednictvím spínání a vypínání tepelné spirály v krátkých proměnlivých intervalech PWM, a tedy je potřeba brát ohled na to jak často bude daný prvek spínat a jaký to bude mít vliv na jeho součásti a také je potřeba brát v potaz proudovou zatížitelnost. Z těchto důvodů je nejvhodnější řešení použít solid state relé, které nemá žádné pohyblivé součásti a na rozdíl od obyčejného elektromagnetického relé má požadovaný jmenovitý proud. [21] [24]

4.4 Relé SSR Solid State SSR-25DA 380V

Protože použité topné spirály mají výkon necelé 3 kW je potřeba pro spínání využít zvláštní druh relé nebo stykač. Z důvodu dvupolohové regulace nám nejlépe vyhovuje použití zařízení SSR Solid State SSR-25DA 380 V.

Relé SSR-25DA 380 V je polovodičové relé, které umožňuje spínat a ovládat různá elektrická zařízení bez použití mechanických částí (Jedno z kritérií). SSR-25DA využívá polovodičové prvky pro řízení proudu a napětí, což umožňuje snadnější a spolehlivější řízení spínání. To znamená, že nemá žádné pohyblivé části jako jsou kontakty nebo spínací bubny relé, což umožňuje rychlejší spínání a delší životnost.

SSR-25DA může snadno spínat zátěže s napětím až 380 V a proudem až 25 A. Kromě toho má SSR-25DA také funkce ochrany proti přetížení a přehřátí. Aby bylo zmírněno zahřívání bude relé přichyceno na pasivní chladič z hliníku. [21] [25]



Obrázek 4: Relé Solid State SSR-25DA 380 V [26]

4.5 Topná spirála

V obyčejné udírně generuje teplo pro uzení samotný oheň a z něho vytvořený kouř, každopádně v automatické udírně teplo vytváříme za pomoci tepelné spirály a dým je vytvářen samostatně v generátoru kouře v takzvaném "DymBoxu". Pro naše využití jsme použili dvouokruhovou spirálu vyseparovanou z horkovzdušné trouby.

Samotná spirála se dělí na dva okruhy, a tedy na malý okruh a velký okruh. Malý okruh má výkon 0,9 kW a velký okruh 1,8 kW. Samotná spirála je 1fázová, ale každý okruh má své vlastní napájení, a tudíž je potřeba ovládat každý kruh zvlášť.

4.6 Displej

Pro zobrazování vnitřní hodnot tepla budeme využívat OLED displej IIC I2C 1,3" s rozlišením 128x64 bodů. Tento displej byl zvolen z důvodu přijatelné ceny, kompaktních rozměrů a nižší spotřebě. V našem případě bude displej sloužit jen pro zobrazení informace o teplotě z důvodu kompletního ovládání přes webové rozhraní, a tudíž nemožnosti narušit proces uzení. [27]

4.6.1 Specifikace displeje

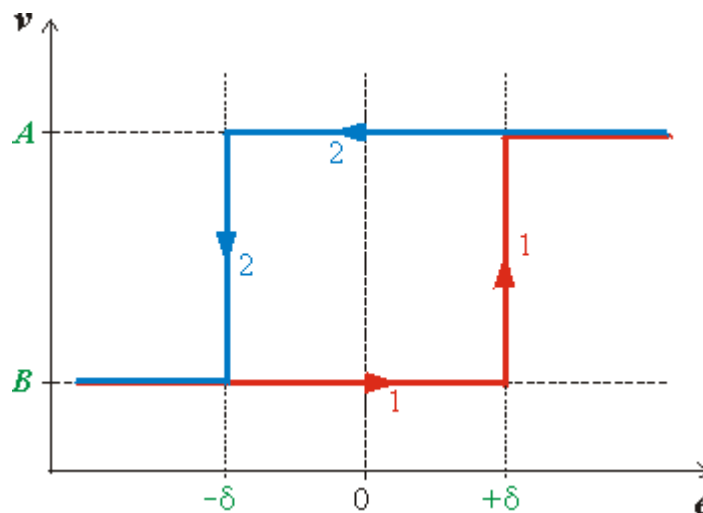
Tabulka 6: Specifikace displeje IIC I2C OLED [27]

| Parametry | Popis |
|------------------|--------------------|
| Úhlopříčka | 1,3" |
| Rozlišení | 128x64 |
| Čip | SSH1106 |
| Rozměry displeje | 29,42 x 14,7 mm |
| Napětí | 3,3 – 5 V |
| Pracovní teplota | -20 až +70 °C |
| Rozhraní | IIC |
| Barva | bílá |
| Piny | GND, VCC, SCL, SDA |

4.7 Dvoupolohová regulace

Pro regulaci teploty v automatické udírně bude použita dvoupolohová regulace, která je jedním z nejjednodušších typů regulace, což patří mezi jeho výhody. V takovém systému je regulátor nastaven tak, aby reagoval na dvě různé polohy, a to buď zapnuto – vypnuto. Princip dvoupolohové regulace je jednoduchý: pokud je hodnota regulované veličiny pod určitou nastavenou hodnotou, regulátor sepne: jakmile sledovaná hodnota překročí požadovanou hodnotu regulátor vypne. Jednou z nevýhod dvoupolohové regulace je, že dochází k častému zapínání a vypínání, což může vést k opotřebení mechanických součástí a neefektivnímu provozu. V tomto případě je potřeba nastavit hysterzezi kolem žádané veličiny.

Dvoupolohová regulace se často používá v systémech, kde není nutná příliš vysoká přesnost, například v termostatech pro vytápění domu. V našem případě by dvoupolohová regulace mohla fungovat tak, že když teplota klesne pod určitou hodnotu, regulátor sepne relé a tím zapneme samotnou spirálu. Když teplota překročí žádanou hodnotu regulátor rozepne relé a tím, vypne spirálu. Tak v případném průběhu teplot můžeme vidět oscilaci kolem žádané hodnoty. [28]



Obrázek 5: Princip práce dvoupolohového regulátoru s hysterzezí. [28]

5 POUŽITÝ SOFTWARE

Pro návrh a řízení automatické udírny bylo využito široké spektrum softwarových technologií. Využity byly jak programovací jazyky pro ESP32, tak i programy pro návrh desky plošných spojů. Použitý software je volně dostupný zdarma, případně je k dispozici zkušební verze.

5.1 Programovací jazyk MicroPython

K programování byl využit MicroPython což je verze jazyka Python 3 určená pro mikrokontrolery a vestavěné systémy. Tato verze Pythonu je velmi podobná standardní verzi jazyka, a tudíž je snazší na naučení, pokud jsme již dříve v Pythonu pracovali. V našem případě budeme využívat i implementaci Java Scriptu pro webové rozhraní. Důležité je také zmínit CSS styly využitě pro vizuální úpravu webového rozhraní. [29] [30]

5.1.1 Rozdíly mezi Pythonem a MicroPythonem

Funkce jazyka Python jsou až na výjimky dostupné také v MicroPythonu. Hlavní rozdíl mezi jazykem Python a MicroPython však spočívá v tom, že jazyk MicroPython byl vyvinut speciálně pro použití v omezeném režimu. Z tohoto důvodu nám MicroPython nenabízí kompletní standardní knihovnu, ale pouze malou část standardní knihovny Pythonu. MicroPython však obsahuje moduly pro nízkoúrovňový přístup k hardwaru, což znamená, že existují knihovny, které nám usnadňují přístup ke GPIO a interakci s nimi, takže je výhodný pro naše použití. [29]

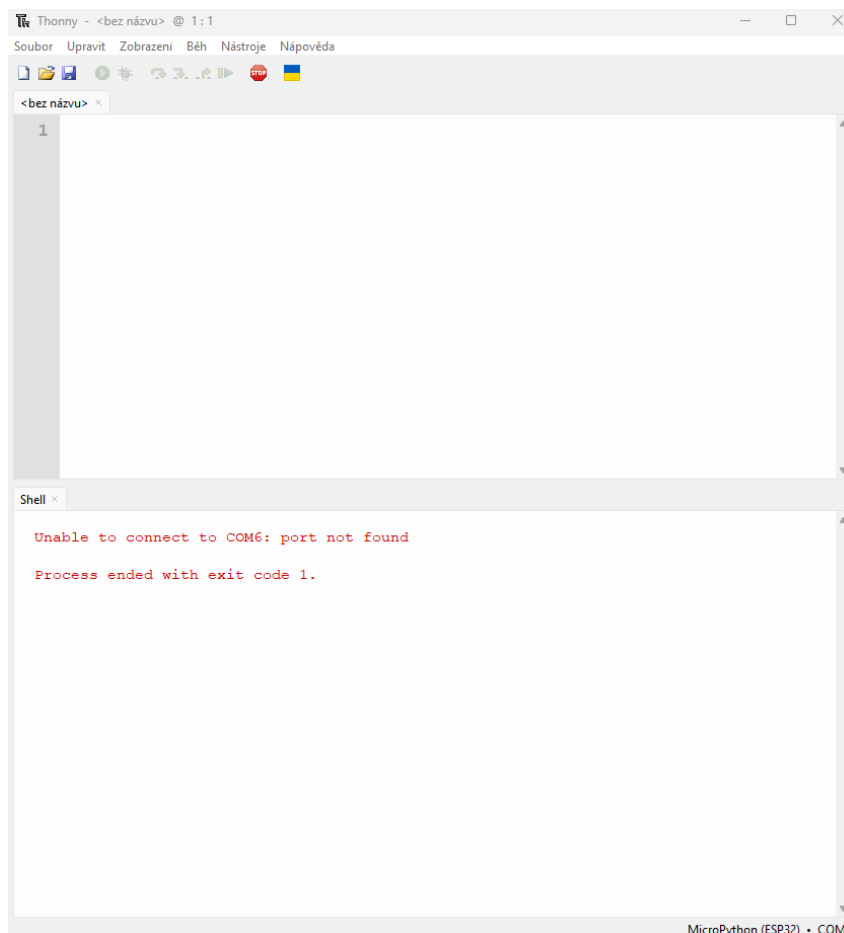
5.2 Thonny IDE

Pro programování v MicroPythonu bylo rozhodnuto o využití Thonny IDE.

Thonny je poměrně nové integrované vývojové prostředí pro učení a výuku programování, které do pracovního postupu pro začátečníky začleňuje vizualizaci programu. Nabízí hned několik základních funkcí, jako jsou různé metody procházení kódu, postupné vyhodnocování výrazu a uživatelsky přívětivé zobrazení zásobníku volání. Thonny je opensource a tudíž jej lze rozšiřovat.

Při otevření Thonnyho, se nám zobrazí dvě hlavní okna. Jedno je „Editor“ a druhé „Terminál“. Sekce Editor je oblast, kde můžeme vytvářet a upravovat soubory, tedy náš kód. Sekce terminál slouží k zadávání příkazu, které se budou provádět přímo na desce bez

nutnosti načítat nové soubory. Kromě toho terminál také prezentuje informace o stavu programu a zobrazuje zprávy týkající se chyb syntaxe. [31] [32]



Obrázek 6: Thonny IDE

5.3 EAGLE

Pro řízení udírny je samozřejmě potřeba mikrokontroler aplikovat na desku plošných spojů, která je vyrobena pro téměř celé řízení a valná část periferií je implementováno přímo na desku. K tomuto účelu byl využit software od společnosti Autocad EAGLE za použití studentské licence.

EAGLE je programovatelný softwarový nástroj pro automatizaci elektronického návrhu, který nabízí funkce pro náskres schématu a rozvržení desek plošných spojů na základě připojení ve schématu. Název EAGLE znamená Easily Applicable Graphical Layout Editor. Eagle byl vyvinutý společností CadSoft Computer a později zakoupen firmou Autodesk, který jej přidala k Fusion 360. [33]

5.4 Autodesk Inventor

I když v samotném zadání práce není samotná konstrukce udírnny, je důležité ukázat zde jak samotný návrh vzniknul a k tomuto účel byl využita studentská licence na Autodesk Inventor.

Autodesk Inventor je softwarový nástroj určený pro 3D modelování používaný především v oblasti strojírenského návrhu digitálních prototypů. Pro 3D návrh dokáže vytvářet výkresovou dokumentaci, řezy, detaily a kotacemi. Vše je navzájem propojené, což znamená, že jakékoli úpravy v modelu se automaticky promítnou do dokumentace, Díky tomu mohou uživatelé rychle identifikovat potenciální chyby v návrhu a provést úpravy ještě před výrobou. Navíc Inventor umožňuje uživatelům simulovat své strojírenské modely a tím ověřit, zda jsou funkční a bez chyb. [34]

5.5 ProfiCAD

Řídicí systém není jen o návrhu a výrobě DPS pro ESP32, ale také návrhu a zapojení jak silové přívodní části, tak řídicího rozvaděče. V tomto ohledu byl využit software ProfiCAD, který je volně dostupný na internetu.

ProfiCAD je profesionální program pro elektrotechnické návrhy a tvorbu schémat. Jedná se o uživatelsky přívětivý program pro elektrotechniky, kteří se zabývají návrhem a dokumentací elektroinstalací. Výhoda ProfiCADu je v jeho schopnosti snadno vytvářet elektrotechnická schémata a dokumentace. ProfiCAD disponuje širokou škálou symbolů a předdefinovaných šablon. Software umožňuje snadnou editaci a aktualizaci schémat, což usnadňuje případné změny v průběhu projektu. ProfiCAD je také vybaven funkcí kontroly chyb, která upozorní na případné problémy ve schématu. Samozřejmostí je export schémat v různých formátech jako PDF, DXF nebo DWG. [35], [36]

6 TECHNOLOGICKÝ PROCES UZENÍ

Samotné uzení je velice zdlouhavý proces, který má své striktní pravidla, kterých se musíme držet, aby hotový výrobek odpovídal požadavkům. Pokud by proces uzení proběhl jinak, mohlo by dojít následně k otravě jídlem a jiným problémům kterých se chceme vyvarovat.

6.1 Základní princip

Uzení jako takové je proces tepelné úpravy masa anebo také sýrů, při kterém se potravina vystavuje teplému kouři způsobenému hořením určitého druhu dřeva po určitou dobu. Samotné uzení pak potraviny vysušuje, tepelně upravuje, dále konzervuje a přidává jim specifickou chuť. Délka procesu závisí na velikosti a hmotnosti masa a požadované úrovni vysušení. Požadavky se liší podle druhu masa a konkrétního druhu potraviny. [37]

6.2 Jednotlivé potraviny v uzení

Každý hotový produkt má své specifické podmínky na technologický proces, který se liší v závislosti na druhu masa, jakou chceme, aby měl výrobek chuť a zda bude uzený anebo sušený.

6.2.1 Vepřové klobásy

U vepřových klobás můžeme využít více technologických procesů, a tedy uzení studeným kouřem, teplým kouřem a horkým kouřem. V této udírně je využito nejčastěji uzení studeným kouřem.

Uzení vepřových klobás teplým kouřem začíná tak, že se udírna rozežřeje na 70 °C, klobásy se vloží dovnitř a nechají se otevřená dvířka, aby klobásy oschly, po uplynutí 30 minut se dvířka zavřou a udíme po dobu 6 hodin. Ke tzv. kouřové chuti je použita štěpka Olše v generátoru kouře. [38]

6.2.2 Čabajské klobásky

Pro uzení pravé Čabajské klobásy se používá mnoho postupů a je tedy možnost jí udit na více způsobů.

V naší udírně bude využit tradiční postup, a tedy uzení studeným kouřem, při kterém se klobáska konzervuje a vytváří se fermentace. Udírna se vyhřeje na teplotu 23 °C a udíme 36 hodin. V generátoru kouře se použijí bukové piliny. Další sušení probíhá poté mimo udírnu a necháváme sušit cca 2 týdny. Delší proces by mohl mít za následek ztvrdnutí klobás. [39]

6.2.3 Vepřová krkovice

Pro uzení Krkovice se používá mnoho postupů a je tedy možnost jí udit na více způsobů.

Maso před uzením musí být cca 12 hodin v pokojové teplotě cca 20 °C, poté se vloží do udírny a nechá se nahřívat při teplotě 70 °C, po 2 hodinách sepne generátor kouře a začíná proces konzervování kouře tzv. uzení, štěpka se může používat z ovocných stromů peckovin, olšová a nejvíc se využívá buková. Kilo Vepřové krkovice se udí 12 h, aby to nebylo přeuzené je v mase digitální teploměr, který snímá teplotu uvnitř masa, kde podle teploty zjistíme, zda je už maso hotové. Zde je využito uzení teplým kouřem. [40]

II. PRAKTICKÁ ČÁST

7 ZÁKLADNÍ KONSTRUKCE

V rámci této práce není předmětem samotná konstrukce udírny nýbrž její řízení, avšak je vhodné zmínit konstrukční základ, který je velice důležitý a bez toho by samotná udírna vůbec nebyla.

Udírna byla sestrojena podle vzoru starší klasické udírny, která je nejvíce rozšířena a pouze byla zvolena robustnější konstrukce, lepší izolace a potřebné úpravy pro správnou funkčnost, jak generátoru kouře, tak topných spirál.

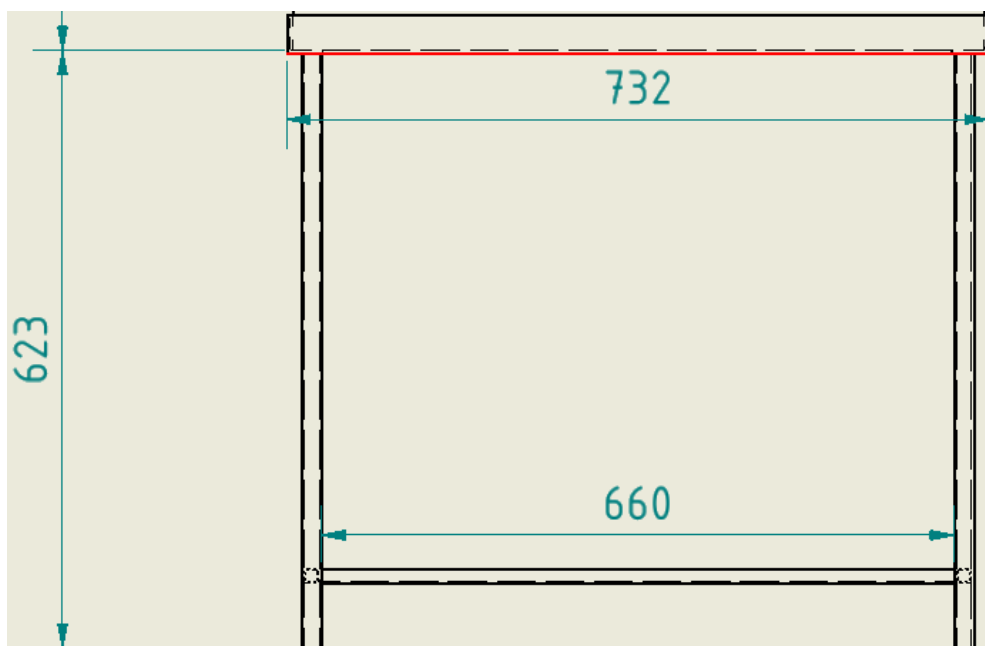
7.1 Konstrukce

Udírna je konstruována na tři části které, jsou v konečné kompletaci poskládané dohromady i z důvodu snadnější manipulace a efektivity.

7.1.1 Základna

Pro konstrukci základny byl využit jelek ISO S235JR o rozměrech 20 x 20 x 2 mm. Ocelová základna poskytuje dostatečnou pevnost a stabilitu pro nesení těžké konstrukce samotné udírny a zároveň je odolná vůči vlivům prostředí.

Ve vrchní části základny, která má tvar L je posazena udírna a samotná vrchní část stojí na čtyřech podpěrných sloupcích, které jsou mezi sebou spojeny, aby se zvýšila stabilita a pevnost konstrukce.



Obrázek 7: Výkres základny

7.1.2 Generátor kouře

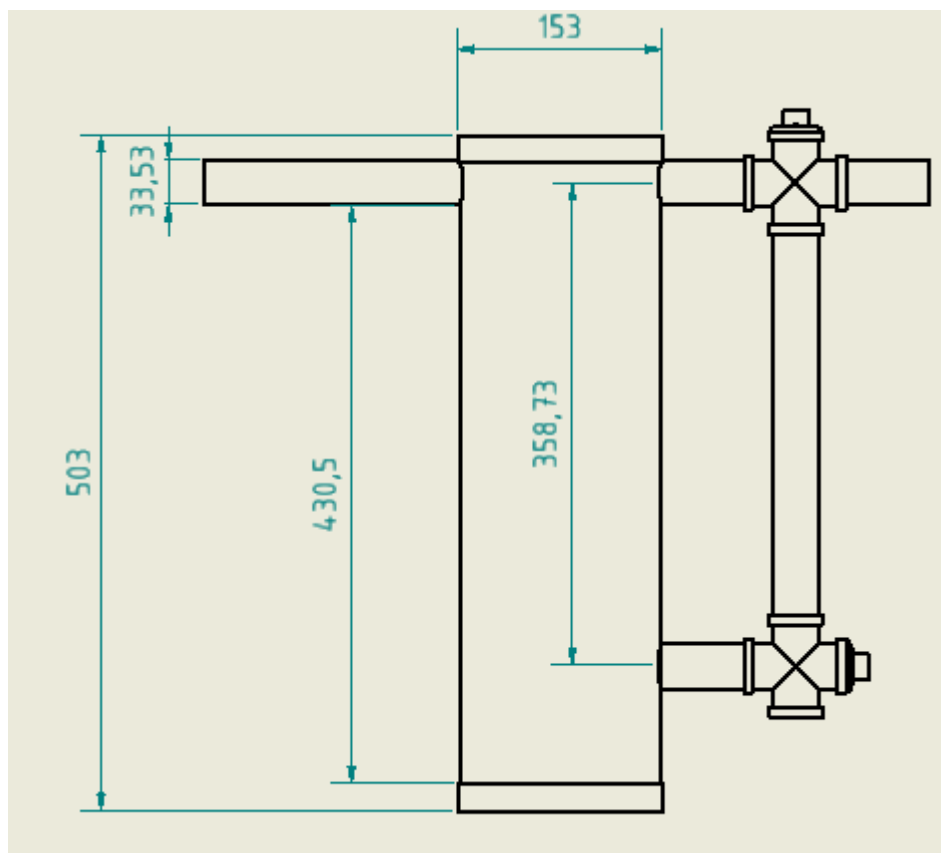
Generátor kouře je klíčovým prvkem pro vytváření kouře v udírně. Konstrukce tohoto generátoru se skládá z několika důležitých prvků.

Základem konstrukce je silnostěnná trubka, která slouží jako nosný prvek pro ostatní součásti a jako zásobník pro štěpku. Do této trubky jsou umístěny další prvky, jako je například ventilátor s napětím 12 V, který slouží k vytváření proudu vzduchu.

Dalším důležitým prvkem je separátor kondenzátu, který slouží k oddělení nežádoucích látek. Tento prvek se nachází v dolní části trubky.

Fixírka, neboli trubička připojená k ventilátoru která vhání kouř ze zapálené štěpky do udírny.

Posledním důležitým prvkem konstrukce je rošt ze slitiny mědi od firmy Ampco Metal. Tento rošt díky velkému podílu mědi udržuje teplotu a zabraňuje tak srážení vody. Díky tomu zůstává štěpka suchá a kouř se vytváří kontinuálně.



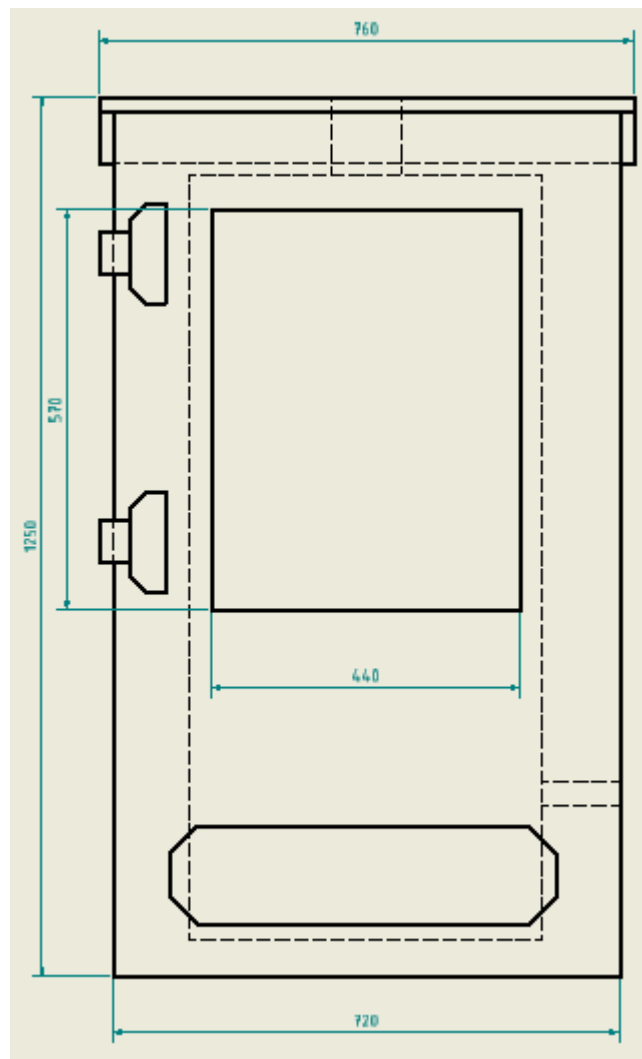
Obrázek 8: Výkres generátoru kouře

7.1.3 Samotná udírna

Velice klíčová je konstrukce samotné udírny, aby byly vytvořeny co nejvíce ideální podmínky. Pro dosažení těchto podmínek se využívá smrkových desek a dvojité stěny.

Dvojitá stěna je tvořena smrkovými deskami, mezi kterými je izolace z čedičové vaty a potažená aluminiovou fólií. Tloušťka této izolace dosahuje 10 cm, což zajišťuje dokonalou tepelnou izolaci a tím i konstantní teplotu uvnitř udírny, což je klíčové pro správnou úpravu masa.

Pro vytváření tepla se na dně udírny nachází spirála. Nad touto spirálou je umístěna nerezová nádoba, která slouží k zachytávání odkapávajícího tuku z masa. V této nádobě se nachází písek, který zabraňuje vznícení tuku.



Obrázek 9: Výkres samostatné udírny

7.2 3D model

Pro lepší představu udírny, byl vypracován 3D model za použití programu Autodesk Inventor. Model odpovídá reálným rozměrům a finální podobě automatické udírny. Byl vypracován svépomocí za použití oficiální knihovny Inventoru, díly, které byly použity z webu jsou ozdrojovány.

Prvním krokem při kreslení konstrukce udírny bylo vytvoření 2D výkresu na papír, který zahrnoval podrobné rozměry pro každou součást udírny. Tento výkres sloužil jako plán pro výrobu jednotlivých součástí udírny.

Po dokončení 2D výkresu byl vytvořen 3D model udírny. Zde se začalo s vytvořením základních tvarů pro jednotlivé součásti udírny, jako jsou dvířka, stěny, základna, na které je udírna umístěna. Poté byly přidávány detaily, jako jsou panty, rozvaděče, zásuvky, samotný generátor kouře, kde bylo využito přírub, nádoba na odpad a ventilátor. [41]

V průběhu tvorby 3D modelu byly použity také různé nástroje a funkce Inventoru, jako jsou například funkce pro kruhové výseče, vazby a nastavování úhlu, což umožnilo vytvořit přesné a detailní modely každé součásti udírny. Na závěr byla vytvořena výkresová dokumentace.



Obrázek 10: 3D model automatické udirny

8 ELEKTROINSTALACE

Elektroinstalace automatické udírny je nejdůležitější prvek v rámci této práce. Obsahuje elektrické součástky a zařízení, které zajišťují bezpečný, spolehlivý a efektivní provoz výrobního procesu. Elektroinstalace se dělí na dvě hlavní části: napájecí část a řídicí část.

8.1 Silová část

Napájecí část elektroinstalace slouží k napájení všech prvků nezbytných pro provoz udírny. Mezi tyto prvky patří topná spirála, zdroj pro mikropočítač nebo napájení ventilátoru. Kompletní návrh silové části probíhal na základě získaných zkušeností z praxe jako elektrotechnik se zaměřením na silnoproudá zařízení a vlastněním vyhlášky 50/1978 Sb. §6 [42].

8.1.1 Hlavní napájení

Hlavní napájení a celá silová část byla zapojena podle schéma, které bylo vytvořeno v programu ProfiCad na základě požadavků pro správné zapojení a funkčnosti všech komponentů s ohledem na bezpečnost. V prvním kroku je potřeba vytvořit nový soubor do kterého bude možnost vložit komponenty a schématické značky. Po vložení všech komponentů slouží k propojení a kresby vedení funkce „Spoj“. Pro přehlednost byly upraveny barvy jednotlivých vodičů a označeny příslušnou schématickou značkou. Při návrh je důležité popsat do jakého místa jsou vyvedeny jednotlivé kabely, aby bylo zřejmé k čemu tento vývod slouží a kam dále pokračuje. Ve schématu jsou zobrazeny schématické značky jednofázových jističů, proudového chrániče, SSR relé a svorkovnic pro PE a N. Celé schéma je k nahlédnutí v příloze viz PŘÍLOHA P IV.

K napájení je použito jednofázové vedení kabelem CYSY 3x2,5 AC 230 V 50 Hz, což znamená, že využívá pouze jedné fáze, neutrálního vodiče a ochranného vodiče. Napětí mezi fází a neutrálním vodičem je 230 V. Napájení je přivedeno do hlavního rozvaděče, kde je přívodní kabel připojený na svorkovnici a dále vyveden na kombinovaný proudový chránič a do dvou dalších jednofázových jističů o hodnotě 16 A, které jsou dále vyvedeny na zásuvky připevněné na konstrukci.

Kombinovaný proudový chránič s jističem má hodnotu 16 A. Tato hodnota byla zvolena po výpočtu hodnoty proudu pro topnou spirálu. Hodnota proudu se dá vypočítat ze vztahu, který již byl použit výše.



Obrázek 11: Proudový chránič s jističem 16 A [43]

Z proudového chrániče jsou přivedeny fázové vodiče na vstupní svorkovnice SSR relé 25-DA 380 V, poté jsou z výstupní svorky vedeny fázové vodiče společně s neutrálním a ochranným vodičem vedeny k topné spirále opět vodičem CYSY 3x2,5. Každé relé SSR 25-DA 380 V je napojeno na jeden okruh topné spirály, dohromady má topná spirála dva okruhy. Oba okruhy využijeme v případě potřeby ohřevu na vyšší teplotu a však v případě potřeby nižší teploty využijeme jen jeden okruh a druhý zůstane nesepnutý a v tedy je jednodušší regulace a nižší odběr. Obě relé jsou připevněné na žebrovaném hliníkovém chladiči.

Z hlavního napájení jsou dále vyvedeny dva kabely, kdy jeden napájí řídicí rozvaděč s ESP32 a druhý slouží k napájení zásuvek na konstrukci a ventilátoru na generátoru kouře. Kabeláž je v obou případech stejná, a to CYSY 3x2,5. Níže se nachází obrázek rozvaděče s hlavním napájením.



Obrázek 12: Hlavní napájení

Na obrázku se nachází dvě relé SSR 25-DA 380 V sepnutém stavu, jak nám indikuje informační dioda, jističe a proudový chránič jsou též zapnuté, a tudíž je celý okruh řízení pod napětím. Hlavní přívod ze sítě je přiveden ze spodní strany rozvaděče to stejné platí pro oba kabely, které jsou zde instalační lištou vedeny k topné spirále. Instalační lišty jsou zde použity jak z estetického důvodu, tak i jako ochrana kabeláže před mechanickým poškozením. Pro zvýšení ochrany před vnějšími vlivy byly použity na vývodech tzv. geovývody.

8.2 Řídicí část

Schéma řídicí části stejně jak silové části je navrhnuté v programu ProfiCad na základě požadavků pro správné zapojení a funkčnosti všech komponentů s ohledem na bezpečnost. V prvním kroku je potřeba vytvořit nový soubor do kterého bude možnost vložit komponenty a schématické značky podle použitých komponentů. Po vložení všech komponentů slouží k propojení a kresby vedení funkce „Spoj“. Pro přehlednost byly upraveny barvy jednotlivých vodičů a označeny příslušnou schématickou značkou. Při návrh

je důležité popsat do jakého místa jsou vyvedeny jednotlivé kabely, aby bylo zřejmé k čemu tento vývod slouží a kam dále pokračuje. Ve schématu jsou zobrazeny schématické značky svorkovnice, zdroje stejnosměrného napětí s volitelnou hodnotou. Ve schématu je umístěn komponent, který představuje DPS s ESP32. Celé schéma je k nahlédnutí v příloze viz PŘÍLOHA P V.

Řídicí část je umístěna v druhém rozvaděči nad rozvaděčem se silovou částí. Řídicí rozvaděč obsahuje mikropočítač ESP32 osazený na desce plošných spojů, napájení pro 12 voltový zdroj a OLED displej, který je umístěný na vnitřní straně odnímatelného krytu s výřezem, aby šlo na displej vidět.

Do rozvaděče vede jeden přívodní kabel, který napájí 12 V zdroj, z něhož jde přívod na napájení desky plošných spojů, na které je osazené ESP32 a také přívod fáze na relé umístěné též na desce plošných spojů, toto relé později spíná ventilátor umístěný na generátoru kouře. Z rozvaděče je dále vyvedeno teplotní čidlo DS18B20 pro snímání teploty připojené k desce s ESP32, které vede do vnitřního prostoru samotné udírny. Mezi další vývody patří vývod ze svorkovnice směrem k relé SSR 25-DA 380 V, který slouží ke spínání, na svorkovnici je přivedeno stejnosměrné napětí 12 V.



Obrázek 13: Řídící rozvaděč.

V rámci vnitřních propojení byly využity jednotlivé vodiče CYA 1 z důvodu nízkého zatížení nebylo třeba využívat silnější vodiče. Na obrázku je dále dočasně připojený microUSB kabel sloužící pro komunikaci s ESP32 a počítačem, avšak tento kabel je zde umístěn jen z důvodu ladění a bude odstraněn.

9 NÁVRH ŘÍDICÍ JEDNOTKY

Pro správnou funkčnost a možnost zapojení všech požadovaných periférií bylo zapotřebí navrhnout schéma a poté desku plošných spojů, kterou následně po objednání všech potřebných komponentu vyrobíme. Pro návrh samotného schématu a desky plošných spojů byl využit program EAGLE od společnosti Autodesk s edukační licencí.

9.1 Použité součástky

Tabulka 7: součástky pro výrobu řídicí jednotky

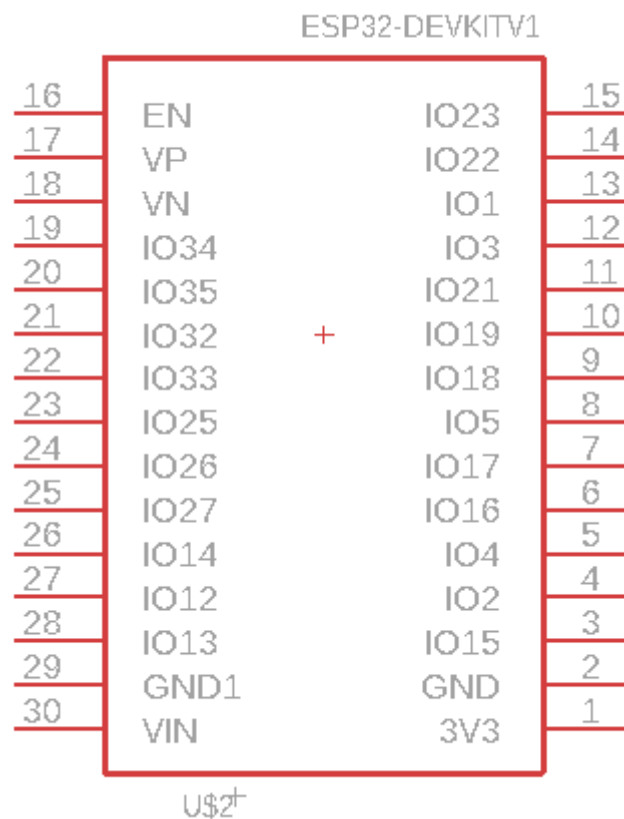
| Název | Popis | Počet kusů |
|---------------------------|---|------------|
| Metalizovaný rezistor 2kΩ | Metalizovaný rezistor s odporem 2 kΩ | 10 ks |
| CK 100n-63V | Keramický kondenzátor 100 nF | 10 ks |
| CE 100u/35VIT | Elektrolytický kondenzátor 100 uF | 5 ks |
| 1N4007 | Dioda 1N4007 | 13 ks |
| L7805CV-DG | Lineární stabilizátor napětí s výstupním napětím DC 5 V | 2 ks |
| V71102 BLACK | Chladič | 1 ks |
| FEPCU150X100 | Fotocuprexit 150x150 | 1 ks |
| OLED displej 1.3“ | OLED displej I2C, bílý | 1 ks |
| KLS 301-5,00 | Svorkovnice 2pól, rozteč 5 mm | 6 ks |
| KLS 300-5,00 | Svorkovnice 3pól, rozteč 5 mm | 5 ks |
| CONFFLY BL110G-V8,5 | Dutinková lišta 1x10pin | 2 ks |
| CONFFLY BL104G-V8,5 | Dutinková lišta 1*4pin | 1 ks |
| KLS S1G08C | Kolíková lišta 1x8pin | 1 ks |
| ESP32 DevKit V1 | Vývojová deska ESP32 | 1 ks |
| Relé RSB2A080P7 | Paticové relé 230 V DC, 8 A | 2 ks |

V tabulce jsou uvedeny všechny potřebné součástky nutné k výrobě řídicí jednotky, z důvodu možných chyb byly některé součástky nakoupeny po více kusech, které později mohou sloužit k případným opravám nebo k jinému projektu.

9.2 Návrh schématu zapojení

Pro návrh schématu vytvoříme v program EAGLE nový projekt, který si libovolně pojmenujeme dle našich „deska“, poté zvolíme vytvořit nové schéma, kde se již dostaneme do editoru a můžeme začít kreslit schéma.

V dalším kroku vyhledáme všechny potřebné součástky a ty postupně rozmístíme na plochu kde je poté všechny propojíme funkcí Net. Nesmíme zapomenout pojmenovat všechny součástky a samozřejmě správně definovat hodnoty a jednotky. Důležitým krokem je buď vytvořit knihovnu s mikropočítačem ESP32 DevKit V1, případně ji stáhnout z internetu a upravit podle sebe. V návrhu je implementovaný Napěťový regulátor z důvodu napájení ESP32, snímače teploty a OLED displeje, všechny tyto součástky vyžadují napětí 5 V. Napětí z přírodních svorek tedy 12 V bylo přivedeno na výstupní svorky pro obě relé SSR 25-DA 380 V.



Obrázek 14: ESP32 DevKit V1 v programu EAGLE

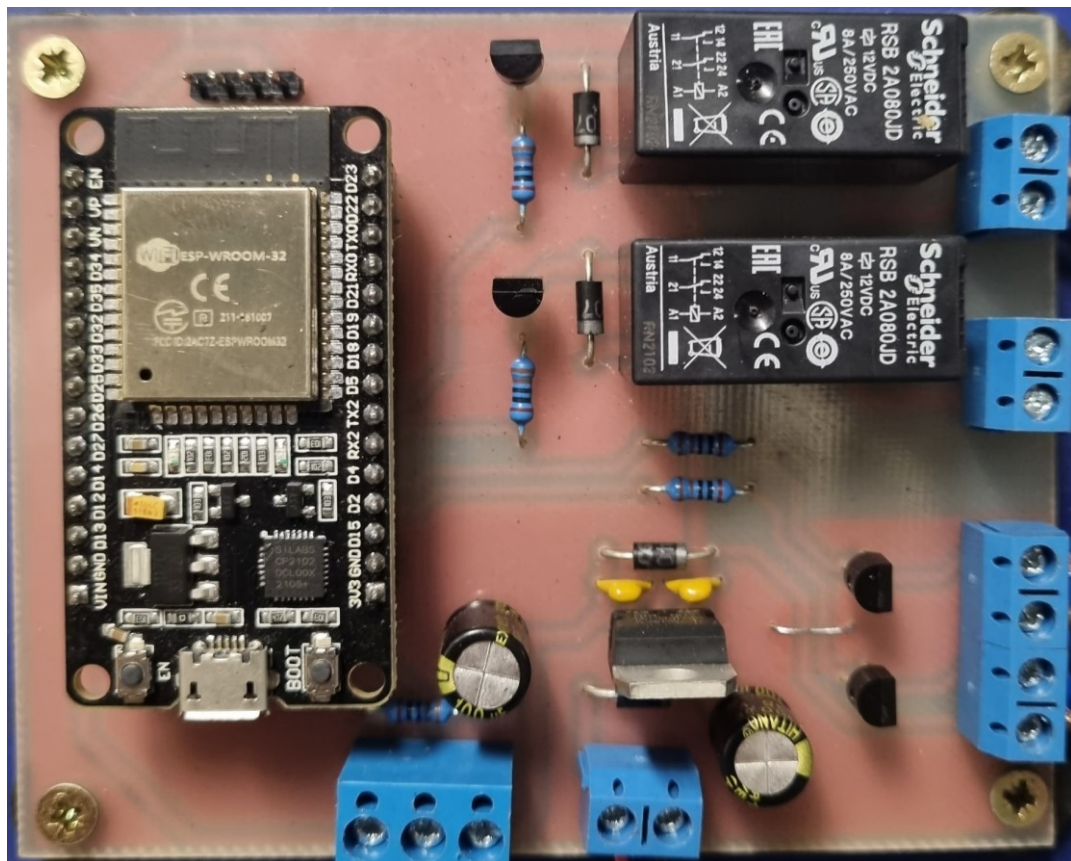
Na obrázku můžeme vidět ESP32 DevKit V1, který má specifikované všechny piny a korespondují s datasheetem od výrobce. Pro ovládání displeje pomocí ESP32 jsou použity piny IO22, IO21, na kterých jsou dostupné signály I2C sběrnice. Piny IO17 a IO16 slouží pro ovládání dvou relé 4052 Schneider, kde výstupy jsou proudově posíleny tranzistory. Jedno relé se stará o spínání ventilátoru pro generátor kouře a druhý bude využit v budoucích optimalizacích a vylepšeních. Piny IO2 a IO15 jsou připojeny přes tranzistory, které tyto výstupy proudově posilují a zároveň umožňují spínat 12 V řídicí napětí na vstupní svorky pro SSR 25-DA 380 V, V neposlední řadě je použit pin IO4, který slouží pro připojení teplotního čidla DS18B20 a také piny VIN a GND pro připojení napájení ESP32.

9.3 Návrh desky plošných spojů

Po již vytvořeném schématu přichází na řadu návrh desky plošných spojů, pro přepnutí do editoru desky plošných spojů klikneme na tlačítko v listě v levém rohu a přepneme se na Board. Zde už před sebou vidíme editor, předdefinovaný prostor, do kterého se budou umisťovat součástky, které jsou momentálně rozloženy mimo desku. V našem případě není rozměr desky velký, jen pouhých 100x80 mm, ale i při těchto rozměrech se podařilo součástky jednoduše umístit na desku podle potřeby. Po rozmístění všech součástek začneme pomocí funkce RouteAirWire kreslit plošné spoje a propojovat součástky podle schématu, samozřejmě ještě potřeba určit si šířku spoje podle velikosti proudu, který daným spojem bude protékat. V neposlední řadě nesmíme zapomenout v každém rohu nakreslit otvory pro možnost desku poté nainstalovat do rozvaděče. Desku plošných spojů společně se schématem bude poté možno dohledat v příloze viz PŘÍLOHA P I, PŘÍLOHA P II.

9.4 Realizace desky plošných spojů

Po konečném návrh desky plošných spojů přichází na řadu finální výroba, kde se budou vytvářet plošné spoje na fotocuprexitu o rozměrech 100*80 mm. Po vyleptání spojů a vyvrtání otvorů pro piny jednotlivých součástek mohou být součástky rozmisťovány do určených otvorů a pomocí mikropájkky, cínu a kalafuny začít pájet součástky k desce. Po úspěšném připájení všech součástek je potřeba ověřit funkčnost desky připojením napájecího zdroje.



Obrázek 15: Zrealizovaná deska plošných spojů

10 PROGRAMOVÉ ŘEŠENÍ

Kapitola se zabývá tvorbou programového vybavení řídicího mikropočítače udírny. Jeden z důvodů, proč byl zvolen ESP32 DevKit V1 je možnost programování v jazyce MicroPython, což je verze programovacího jazyku Python 3 pro mikropočítače.

Pro programové řešení bylo využito programovací prostředí Thonny IDE, které je volně dostupné na internetu a doporučované jako jedno z nejlepších možných řešení. V rámci programování byly použity dostupné knihovny, které má v sobě MicroPython již implementované tak i knihovny dostupné na internetu.

10.1 Program

Tento kód je určený pro ovládání a monitorování udírny pomocí mikropočítače ESP 32 DevKit V1. Program řídí tři relé a zobrazuje informace na OLED displeji I2C 1,3“. Kromě toho implementuje webový server, který umožňuje ovládání udírny pomocí webového prohlížeče. Když se spustí, tento kód zařídí, že ESP32 DevKit V1 bude ovládat externí prvky na základě aktuálně spuštěného programu.

Kód byl rozdělen do vybraných částí a popsán, jak každá část funguje a k čemu slouží. Některé části jsou vynechány jako například čtení teploty a v elektronické příloze bude k nalezení kompletní kód v MicroPythonu.

10.1.1 Boot.py

ESP32 obsahuje v MicroPythonu skript boot.py který se spustí jako první a může sloužit k nastavení a importování potřebných knihoven pro správnou funkčnost. V tomto případě sice boot.py importuje určité knihovny, ale většina knihoven je importována až ve skriptu main.py z důvodu přehlednosti a z důvodů využití jiných tříd v každém skriptu. Boot.py je zde tedy spíše testovací a doplňkový skript, který zajišťuje připojení k WiFi síti a práci se síťovými komponentami, import knihovny „esp“ pro práci s ESP32 čipem, správu paměti a garbage collector, který slouží pro uvolnění nepoužité paměti.

```
ssid = '02-Internet-858'  
password =   
  
station = network.WLAN(network.STA_IF)  
  
station.active(True)  
station.connect(ssid, password)  
  
while station.isconnected() == False:  
    pass  
  
print('Jsme online ')  
print(station.ifconfig())
```

Obrázek 16: Funkce pro připojení k WiFi síti

Na obrázku výše je část kódu ze skriptu boot.py která má funkci pro připojení k WiFi síti v domácích podmínkách, tudíž hned po zapnutí se ESP32 připojí k WiFi síti, pro kontrolu při ladění programu ještě vypíšeme na terminál text 'Jsme online ' a tím je ověřeno, že je ESP32 úspěšně připojeno k síti, dále vypíše na terminál IP adresu ESP32.

10.1.2 Main.py

Skript main.py je hlavní skript, který se stará o běh celého programu. Jak bylo již zmíněno většina důležitých knihoven je importováno právě zde a zde probíhá i hlavní inicializace pinů pro použití v programu. Níže jsou popsány části kódu a vysvětleno k čemu slouží.

10.1.2.1 Import knihoven

Import knihoven je důležitý pro komunikaci s hardwarovými součástkami ESP32 DevKit V1 a pro implementaci webového rozhraní. Na obrázku jsou importovány knihovny, které jsou potřebné. Většina knihoven je již implementována v MicroPythonu avšak jedna knihovna musela být ručně nahrána do ESP32 a poté importována do skriptu.

```
#Import důležitých knihoven
import socket
import machine
import onewire, ds18x20
import time
from machine import Pin, SoftI2C
import _thread
import sh1106
import json
```

Obrázek 17: Import knihoven

Knihovna socket, která poskytuje nízkoúrovňové síťové rozhraní pro komunikaci pomocí BSD socketů. Umožňuje vytvářet TCP a UDP spojení, posílat a přijímat data a spravovat síťové servery. V tomto případě je využita pro vytvoření webového serveru, jehož prostřednictvím lze ovládat provoz udírny.

Knihovna machine slouží pro práci s hardwarovými komponentami ESP32, jako jsou GPIO piny, komunikace s I2C pro OLED displeje. V tomto případě je knihovna použita pro nastavení výstupních pinů relé, inicializaci I2C pro OLED displej a pro komunikaci s DS18B20 což je teplotní čidlo.

Knihovna onewire poskytuje základní funkci pro komunikaci přes OneWire sběrnici a knihovna ds18x20 je určena pro čtení teploty teplotního čidla DS18B20.

Knihovna time je určena pro práci s časem. V tomto projektu je využita pro zpoždění mezi čtením teploty a aktualizací stavu relé.

Knihovna Pin je třída z knihovny machine, slouží k nastavení a ovládní GPIO pinů. SoftI2C je třída pro implementaci I2C komunikace, která slouží k nastavení pinů pro komunikaci s I2C zařízením v našem případě OLED displeje.

Knihovna _thread slouží pro práci s vlákny v MicroPythonu. Zde je použita pro vytvoření dvou paralelních vláken: jedno pro naslouchání příchozích webových požadavků a druhé

pro řízení udírny a aktualizaci OLED displeje. Tato knihovna je velice důležitá a bez ní by nebylo možné tak efektivně ovládat udírnu. [44]

Knihovna `sh1106[]` slouží pro práci s OLED displejem s řadičem SH1106. Z důvodu mnoha různých výrobců tohoto displeje není jednoduché najít správnou knihovnu pro inicializaci displeje. Tato knihovna tedy umožňuje inicializaci displeje, nastavení kontrastu, výpis textu. Knihovna `sh1106` je jako jediná byla použita z internetu a nahrána zvlášť do ESP32, protože MicroPython tuto knihovnu nemá v sobě implementovanou. Poslední import knihovny, kterou máme je `json`, která je zde využita pro převod dat mezi objekty micropythonu a řetězci JSON. V našem program vykonává funkci pro odesílání dat o teplotě a zbývajícím čase na webovou stránku bez obnovení prohlížeče. [45]

10.1.2.2 Inicializace periférií a nastavení OLED displeje

Důležitá součást kódu je samozřejmě inicializace a nastavení všech komponent, které jsou využívány jako piny, na kterých jsou připojeny relé, čidlo a sběrnice pro OLED displej přes I2C komunikaci.

```
# Nastavení pinů
rele1 = Pin(2, Pin.OUT)
rele2 = Pin(15, Pin.OUT)
rele3 = Pin(17, Pin.OUT)

# Nastavení teplotního čidla DS18B20
dat = machine.Pin(4)
ds = ds18x20.DS18X20(OneWire.OneWire(dat))
roms = ds.scan()

# Nastavení OLED displeje
i2c = SoftI2C(scl=Pin(22), sda=Pin(21))
display = sh1106.SH1106_I2C(128, 64, i2c)
```

Obrázek 18: Nastavení pinu pro relé, teplotního čidla a OLED displeje

V této části kódu se nastavují piny pro `rele1`, `rele2` a `rele3`. Na těchto pinech jsou připojené relé, na `rele1` je připojeno jedno ze dvou SSR 25-DA 380 V, přímo tohle relé ovládá menší okruh topné spirály, zatímco `rele2` spíná druhé SSR 25-DA 380 V které je připojeno na větší okruh topné spirály. V neposlední řadě je zde `rele3`, které spíná ventilátor pro generátor kouře.

10.1.2.3 Funkce web_page()

Tato část kódu je velice důležitá protože funkce `web_page()` vrací HTML kód webové stránky. Na webové stránce se nachází tlačítka přes které volíme udící programy nebo program ukončujeme, pokud tedy uživatel klikne na jedno z těchto tlačítek, webový prohlížeč se přesměruje na URL definovanou pomocí `location.href`. URL obsahuje název programu, který slouží jako příkaz. Celé toto HTML tedy slouží jako dynamické uživatelské rozhraní. Z důvodu rozsahu kódu, bude celý kód k nahlédnutí v příloze.

Kód ještě obsahuje samozřejmě další prvky, kde se nastavují parametry např. pro responzivní design, aby se stránka správně zobrazovala na různých zařízeních i když ve většině případů bude využit mobilní telefon. V tomto kódu bylo také využito knihovny `json` pro implementaci JavaScriptu z důvodu zobrazení aktuální teploty, zbývajících času, a taky zvoleného programu. Tato funkce je velice důležitá, protože se stará o pravidelné zasílání GET požadavku na `‘/data‘`. JavaScript také posílá POST požadavky na server při kliknutí na tlačítka. Je také potřeba zmínit využití CSS stylů dostupné z internetu. [46]



Obrázek 19: Uživatelské rozhraní na webové stránce

10.1.2.4 *listen_for_clients(lock)*

Další důležitá funkce, která naslouchá příchozím klientům a zpracovává jejich požadavky z webového rozhraní. U této funkce je důležité, že se spouští jako samostatné vlákno, je to z důvodu, aby funkce mohla pracovat nezávisle na dalších částech programu. Zjednodušeně řečeno tato funkce zajišťuje komunikaci mezi webovým prohlížečem a ESP32.

Tato funkce na základě požadavků například, klobasa, krkovice, bucek spustí program pro uzení. Tyto požadavky mohou být odeslány pouze pokud žádný jiný program neběží. Tohle ošetření bylo použito z důvodů neustálého spouštění zvoleného programu kvůli

implementaci JavaScriptu. V kódu je definice funkce se parametrem lock, který slouží pro synchronizaci vláken. Kód obsahuje deklaraci globálních proměnných, které jsou sdíleny mezi vlákny. While true je nekonečná smyčka, která zajišťuje, že funkce bude stále naslouchat a zpracovávat příchozí klienty. Conn, addr = s.accept() čeká na připojení klienta a přiřazuje spojení a adresu do proměnných conn a addr. Request = conn.recv(1024) přijímá požadavek od klienta o velikosti až 1024 bajtů. Lock.acquire() získá zámek pro synchronizaci vláken. Tímto zajišťuje, že jen jedno vlákno bude přistupovat ke globálním proměnným najednou. V kódu dále jsou podmínky pro zpracování různých požadavků např. pokud je v požadavku konkrétní URL, nastaví se globální proměnné podle zvoleného programu. Lock.release() zase uvolňuje zámek pro další vlákna a umožňuje tedy přistupovat ke zdrojům. Jak je vidět v ukázce kódu na konci je ošetření výjimek typu OSError, která často narušovala průběh programu. [44] [47]


```
def listen_for_clients(lock):
    global program, program_start_time, display_state, current_temperature, program_started

    while True:
        try:
            conn, addr = s.accept()
            print('Client connected from:', addr)
            request = conn.recv(1024)
            request = str(request)
            print('Přijatý požadavek:', request)

            lock.acquire() # Synchronizace vláken

            if not program_started and request.find('/klobasa') != -1:
                program = "klobasa"
                program_start_time = time.time()
                display_state = "running_program"
                program_started = True

            if not program_started and request.find('/krkovice') != -1:
                program = "krkovice"
                program_start_time = time.time()
                display_state = "running_program"
                program_started = True

            if not program_started and request.find('/bucek') != -1:
                program = "bucek"
                program_start_time = time.time()
                display_state = "running_program"
                program_started = True

            if not program_started and request.find('/rolada') != -1:
                program = "rolada"
                program_start_time = time.time()
                display_state = "running_program"
                program_started = True

            if not program_started and request.find('/gouda') != -1:
                program = "gouda"
                program_start_time = time.time()
                display_state = "running_program"
                program_started = True

            if not program_started and request.find('/parenica') != -1:
                program = "parenica"
                program_start_time = time.time()
                display_state = "running_program"
                program_started = True

            if request.find('/ukoncit') != -1:
                program = None
                program_start_time = 0
                display_state = "end_program"
                program_started = False

            lock.release() # Uvolnění zámku

            if request.find('/data') != -1:
                elapsed_time = time.time() - program_start_time
                remaining_time = max(program_duration[program] - elapsed_time, 0) if program else 0
                data = get_temperature_time_data()
                conn.send('HTTP/1.1 200 OK\n')
                conn.send('Content-Type: application/json\n')
                conn.send('Connection: close\n\n')
                conn.sendall(json.dumps(data))
            else:
                response = web_page()
                conn.send('HTTP/1.1 200 OK\n')
                conn.send('Content-Type: text/html\n')
                conn.send('Connection: close\n\n')
                conn.sendall(response)
        except OSError as e:
            if e.args[0] == 104: # ECONNRESET
                print("Spojení přerušeno klientem.")
            else:
                print("OSError:", e)
        finally:
            conn.close()
```

Obrázek 20: listen_for_client()

10.1.2.5 Main(lock)

Tato část programu je hlavní funkce řídící logiku udírny. Ta měří teplotu v udírně pomocí teplotního čidla DS18B20. Dále pak na základě výběru programu uživatele ovládá relé, která regulují teplotu v udírně. Funkce obsahuje několik programů, které mají přednastavenou dobu trvání a cílovou teplotu. V průběhu provozu udírny se na OLED displeji zobrazuje aktuální teplota, zbývající čas a název spuštěného programu. Ve zkratce funkce obsahuje deklaraci proměnných, inicializaci proměnné pro čas spuštění rele3, proměnnou pro uchování času poslední aktualizace displeje, Nekonečnou smyčku while True pro provádění akcí v závislosti na zvoleném programu, čtení aktuální teploty a ukládání do proměnné nebo pauzu mezi měřením teploty. Z důvodu rozsáhlosti této části kódu nebude vložen obrázek s ukázkou, ale bude možné nahlédnout do přílohy.

10.1.2.6 Update_display()

Je část kódu kde tato funkce zajišťuje aktualizaci OLED displeje s informacemi o teplotě, aktuálním programu a zbývajícím čase. Tato funkce má čtyři parametry a tedy temperature, program, remainig_time, display_state. Funkce nejdříve smaže celý displej nastavením všech pixelů na 0 k tomuto je využít příkaz display.fill(0). Funkce se na základě hodnoty display_state rozhodne, jaké informace zobrazí na displeji. Pokud je tedy display_state rovno select_program, zobrazí se na displeji text Vyberte program, tento text se zobrazuje buď při zapnutí ESP32 nebo po ukončení programu. Další, co funkce obsahuje je, že pokud je display_state rovno running_program, zobrazí se na displeji aktuální teplota a název aktuálního programu a také zbývající čas do konce programu. V neposlední řadě, pokud je display_state rovno end_program zobrazí se na displeji text Konec programu a po deseti vteřinách se znovu objeví Vyberte program. Na konci funkce je ještě ošetření, pokud dojde k chybě během funkce OSError chyba se zachytí a vypíše se do terminálu zpráva Chyba při aktualizaci displeje: Tato funkce je užitečná jak při ladění displeje, tak v případě vzniku nečekané chyby na displeji, avšak tato chyba nenaruší průběh uzení.

```
def update_display(temperature, program, remaining_time, display_state):
    try:
        display.fill(0)

        if display_state == "select_program":
            display.text("Vyberte program", 0, 32, 1)
        elif display_state == "running_program":
            display.text("Teplota:", 0, 0, 1)
            display.text("{:.1f} C".format(temperature), 0, 16, 1)

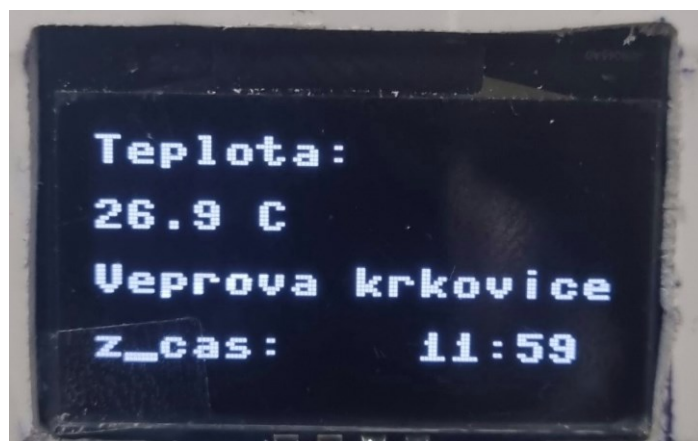
            program_names = {
                "klobasa": "Veprova klobasa",
                "krkovice": "Veprova krkovice",
                "bucek": "Bucek",
                "rolada": "Rolada",
                "gouda": "Gouda",
                "parenica": "Parenica",
            }

            if program in program_names:
                display.text(program_names[program], 0, 32, 1)

            if program in ["klobasa", "krkovice", "bucek", "rolada", "gouda", "parenica"]:
                display.text("z_cas:", 0, 48, 1)
                display.text(f"{remaining_time}", 80, 48, 1)
            elif display_state == "end_program":
                display.text("Konec programu", 0, 32, 1)
            display.show()
    except OSError as e:
        print("Chyba při aktualizaci displeje:", e)
```

Obrázek 21: Funkce update_display

Na dalším obrázku je ukázka, jak se zobrazují informace na OLED displeji konkrétně zde program Vepřová krkovice, která má nastavenou teplotu na 70 °C a čas uzení 12 hodin.



Obrázek 22: Zobrazení hodnot na OLED displeji

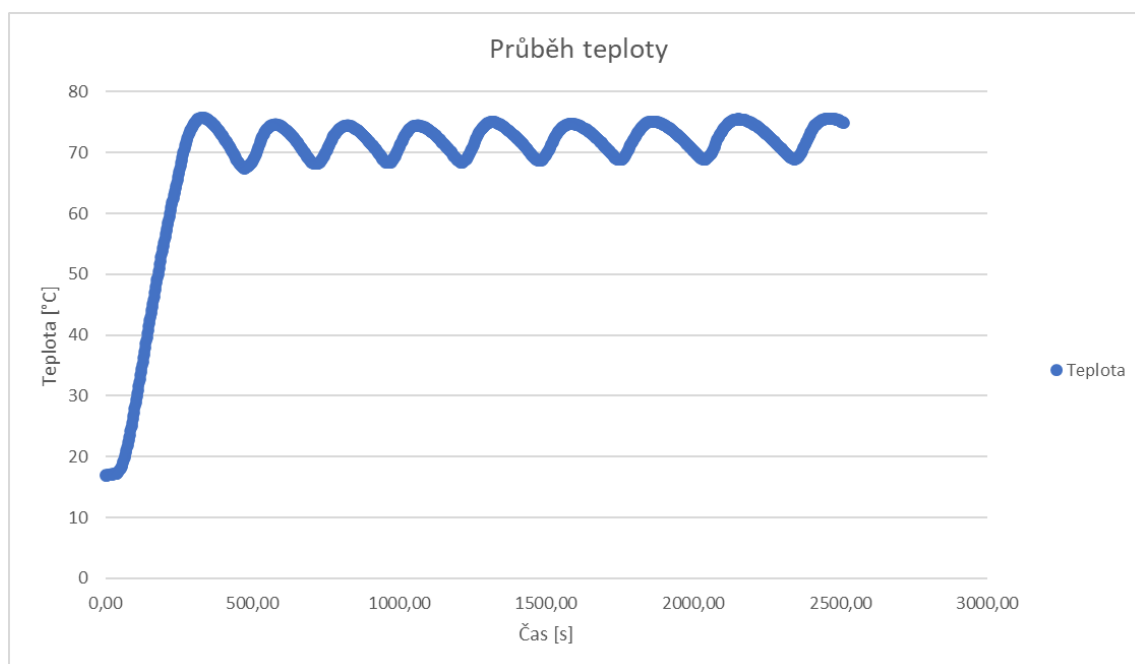
11 OVĚŘENÍ FUNKCE UDÍRNY

Kapitola se zabývá ověřením funkčnosti řízení automatické udírny. Cílem bylo nejen ověřit, zda udírna pracuje podle předem stanovených parametrů, ale také zjistit, jak dobře dokáže udržovat žádanou teplotu nebo efektivně spínat relé. Také je potřeba ověřit uživatelské rozhraní na webu, OLED displeji a interakci s uživatelem.

Před uvedením do provozu bylo potřeba ověřit správnou funkčnost všech komponentů, z tohoto důvodu byl napsán jednoduchý program, který byl určený pro spínání a rozpínání relé a tím ověřit, zda vše funguje. Po sepnutí relé bylo pomocí multimetru ověřeno, zda se na vstupních i výstupních svorkách nachází správné hodnoty napětí. Důležité bylo také ověřit funkčnost teplotního senzoru DS18B20, kde byl také vytvořen obslužný program, který testoval, zda teplotní čidlo funguje dle požadavků a výstupní hodnoty odpovídají. Ověřena byla i stabilita WiFi sítě, avšak z důvodů ošetření programu, pokud nastane odpojení od sítě, zvolený program dále pokračuje až do konce, a tedy nehrozí ohrožení procesu.

11.1 Ověření regulace teploty

Nejdůležitější bylo ověřit, zda udírna bude udržovat žádanou teplotu, která byla definována pro každý udící program. K tomu budou využita data z teplotního snímače DS18B20, vzorkovací frekvence byla nastavena na 2 sekundy a délka celého měření byla 2512 sekund.



Obrázek 23: Průběh žádané teploty.

Před začátkem byla udírna otevřená, aby se vnitřní teplota srovnala s teplotou okolí, jakmile teplota v udírně dosáhla teploty okolí, spustí předefinovaný udící program. Počáteční teplota byla 16,94 °C a žádaná teplota nastavena na 70 °C. Udírna dosáhla žádané teploty přibližně za pět minut s tím, že po vypnutí relé teplota stále roste na určitou hodnotu, než začne zase klesat. Jakmile teplota klesla pod žádanou hodnotu relé sepnula. Z grafu je patrná funkčnost dvoupolohové regulace.

11.2 Ověření spuštění udírny

Po nastavení všech potřebných parametrů o ověření všech periférií bylo potřeba ověřit správnou funkčnost udírny jako celku. Prvním krokem je potřeba připojit přívodní kabel do sítě, poté zapnout všechny potřebné jističe a kombinovaný proudový chránič. Po chvíli se rozsvítí OLED displej umístěný na ovládacím rozvaděči a vypisuje text „Zvolte program“ v tu chvíli je možné pomocí mobilního telefonu nebo počítače ovládat automatickou udírnu. Na webových stránkách se zobrazí ovládací panel pro volbu programů, aktuální teplota, zbývající čas a zvolený program. V tomto případě se teplota měří nezávisle na programu, a tedy v nekonečné smyčce po 5 sekundách snímá teplotu. Po zvolení udícího programu se aktualizuje webová stránka a zobrazí se zvolený program a odpočet zbývajícího času. Teplota se stále aktualizuje. Po zvolení programu se také aktualizuje OLED displej, který zobrazuje aktuální teplotu, zvolený program a zbývající čas. V závislosti na zvoleném programu ESP32 sepne SSR relé a snaží se dosáhnout žádané hodnoty a poté se na této hodnotě udržovat. Po uběhnutí předem definovaného času ESP32 sepne relé, které ovládá ventilátor v generátoru kouře. Při běhu programu se všechny údaje zobrazené jak na webovém rozhraní, tak na OLED displeji každých pět sekund obnovují. Po skončení programu se na OLED displeji vypíše text „Konec programu“ a po 10 vteřinách se znovu zobrazí text „Zvolte program“ v tu chvíli je udírna připravena na volbu dalšího udícího programu. Na webovém rozhraní se údaje vyresetují krom teploty. Ověřit také bylo potřeba tlačítko ukončit program, které jde zde z důvodu buď špatné volby programu a v tedy je můžeme vypnout a znovu zvolit správný program nebo z důvodu už hotového výrobku, aby nedošlo k přerušení. Tato situace může nastat například při uzení Vepřové krkovice, v mase je během uzení implementovaný vpichový digitální teploměr, jakmile má maso uvnitř požadovanou teplotu, je zapotřebí udírnu vypnout. V rámci budoucí optimalizace je v plánu ošetření i tohoto jevu a implementace vpichového teploměru do samotného řízení udírny. Ověřeny byly také jisticí prvky a proudový chránič, u proudového chrániče máme možnost

otestovat funkčnost tlačítkem test, kdy po stisknutí tlačítka proudový chránič fungoval dle požadavků.

Během udícího programu nedošlo k žádným neočekávaným chybám, vše fungovalo podle nastavených parametrů a zapojení. Díky pasivnímu chlazení pro SSR relé nedocházelo k příliš velkému zahřívání po dobu běhu programu.



Obrázek 24: Kompletní sestava automatické udírny

ZÁVĚR

Tato bakalářská práce se zabývá návrhem a sestavením řízení pro automatickou udírnu ovládanou pomocí webového rozhraní, které běží na ESP32 DevKit V1. V rámci práce byla provedena teoretická analýza automatických udíren, a to včetně výběru vhodného mikropočítače a periférií pro připojení k ESP32. V další kapitole se zabývá výběrem softwaru pro návrh DPS a programové řešení. V další části je zaměřena na obecné základy principu uzení a správné přípravy na tento proces.

V praktické části práce byly popsány základní konstrukční vlastnosti a model automatické udírny. Byl vytvořen kompletní model udírny s využitím programu Autodesk Inventor. Následovala kapitola věnovaná elektroinstalaci a implementaci řízení do udírny. Byly popsány použité součástky, návrh a výroba řídicí jednotky a rozvržení a zapojení elektroinstalace.

Předposlední kapitola se zaměřila na tvorbu a popis programového řešení v programovacím jazyce MicroPython. Bylo potřeba naprogramovat a nastavit všechny potřebné udící programy, vizualizaci informací na OLED displeji a dynamické webové rozhraní pro volbu programů. Všechny definované programy obsahují požadované parametry jako je žádaná teplota, čas uzení nebo dobu provozu generátoru kouře. Tímto způsobem by měly všechny parametry odpovídat požadavkům kladeným na správný proces uzení.

Poslední kapitola této práce se zaměřuje na ověření funkčnosti a zpracování automatické udírny jak celku. Bylo nutné ověřit, zda průběh teploty během udícího programu odpovídá dvoupolohové regulaci a zda teplotní čidlo DS18B20 správně funguje a ESP32 dokáže s těmito daty pracovat. Součástí ověření bylo také testování funkčnosti všech komponentů včetně SSR 25DA 380 V, relé RSB2A080P7 a elektrického zapojení obecně včetně bezpečnostních prvků.

Všechna schémata jsou přiložena v přílohách práce, které lze se zdrojovými kódy pro ESP32 a 3D modelem nalézt také na přiloženém CD.

Hlavním přínosem této práce je spojení všech nabytých znalostí získaných během studia a také získání nových dovedností v oblasti programování mikropočítačů, praktické tvorby DPS a návrhu požadovaného modelu zařízení. Práce představuje komplexní a funkční systém pro automatizaci procesu uzení. Výsledky této práce mohou být inspirací pro další vývoj a zdokonalování automatizovaných procesů v oblasti potravinářského průmyslu.

SEZNAM POUŽITÉ LITERATURY

- [1] CO VYBRAT. *Jak vybrat elektrickou udírnu* [online]. In: . [cit. 2023-05-14]. Dostupné z: <https://www.covybrat.cz/nejlepsi-udirna/>
- [2] Elektrická digitální udírna Bradley Smoker 4. In: *Grilovani.cz* [online]. Pardubice [cit. 2023-05-14]. Dostupné z: <https://www.grilovani.cz/udirna-bradley-smoker-4-digital/>
- [3] Plynové udírny. In: *Grilovani.cz* [online]. Pardubice [cit. 2023-05-14]. Dostupné z: <https://www.grilovani.cz/plynove-udirny/>
- [4] ESP32 Series Datasheet. In: *Espressif Systems* [online]. Shanghai, 2023 [cit. 2023-05-14]. Dostupné z: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf
- [5] Arduino Uno WiFi. In: *Arduino Documentation* [online]. 2023: Arduino [cit. 2023-05-14]. Dostupné z: <https://docs.arduino.cc/retired/boards/arduino-uno-wifi>
- [6] TOMEŠEK, Jiří. Lekce 1 - Seznámení s ESP32. In: *ITnetwork* [online]. Praha: ITnetwork, 2023 [cit. 2023-03-24]. Dostupné z: <https://www.itnetwork.cz/hardware-pc/arduino/esp32/seznameni-s-esp-32>
- [7] Getting Started with ESP32 | Introduction to ESP32. In: *ElectronicsHub* [online]. 2023 [cit. 2023-05-14]. Dostupné z: <https://www.electronicshub.org/getting-started-with-esp32/>
- [8] Getting Started with the ESP32 Development Board. In: *RandomNerdTutorials* [online]. RandomNerdTutorials, 2013 [cit. 2023-05-14]. Dostupné z: <https://randomnerdtutorials.com/getting-started-with-esp32/>
- [9] ESP8266. In: *Wikipedia* [online]. [cit. 2023-05-14]. Dostupné z: <https://en.wikipedia.org/wiki/ESP8266>
- [10] *DS18B20* [online]. Rev 6. Maxim Integrated Products, 2019 [cit. 2023-05-14]. Dostupné z: <https://www.analog.com/media/en/technical-documentation/datasheets/DS18B20.pdf>
- [11] *DHT11 Humidity & Temperature Sensor* [online]. Mouser Electronics [cit. 2023-05-14]. Dostupné z: <https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf>

- [12] LIU, Thomas. *Digital-output relative humidity & temperature sensor/module DHT22 (DHT22 also named as AM2302)* [online]. Aosong Electronics Co.,Ltd [cit. 2023-05-14]. Dostupné z: <https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf>
- [13] Snímače teploty Pt100 - princip, zapojení, třídy přesnosti. In: *PROFESS* [online]. Plzeň: PROFESS, spol. s r.o. [cit. 2023-05-14]. Dostupné z: https://www.profess.cz/cs/pci/odporove_snimace_teploty
- [14] *Cable Sensor, Pt100/Pt1000* [online]. UK: IVO [Baumer IVO GmbH & Co. KG], 2008 [cit. 2023-05-14]. Dostupné z: <https://pdf1.alldatasheet.com/datasheet-pdf/view/1491698/IVO/PT1000.html>
- [15] *NTC thermistor* [online]. Verze 1.0. ECLIPSE s.r.o, 2017 [cit. 2023-05-14]. Dostupné z: <https://dratek.cz/docs/produkty/1/1490/1488979094.pdf>
- [16] *MAX31865* [online]. Rev 3. Maxim Integrated, 2015 [cit. 2023-05-14]. Dostupné z: https://www.laskakit.cz/user/related_files/max31865.pdf
- [17] MicroPython: DS18B20 Temperature Sensor with ESP32 and ESP8266. In: *RandomNerdTutorials* [online]. RandomNerdTutorials [cit. 2023-05-14]. Dostupné z: <https://randomnerdtutorials.com/micropython-ds18b20-esp32-esp8266/>
- [18] DS18B20 Digitální vodotěsné čidlo teploty 1m. In: *LaskaKit* [online]. Rychnov nad Kněžnou: LaskaKit, 2023 [cit. 2023-05-14]. Dostupné z: https://www.laskakit.cz/dallas-digitalni-vodotesne-cidlo-teploty-ds18b20-1m/?gclid=Cj0KCQiA3eGfBhCeARIsACpJNU8wi3SclqEM37j73QAyhX0UrgMjApclA1A5XLMsyYy_gh9qbpjENPYaAvcxEALw_wcB
- [19] Elektrický proud. In: *Wikina* [online]. Wikina [cit. 2023-05-14]. Dostupné z: http://www.wikina.cz/a/Elektrick%C3%BD_proud
- [20] REDAKCE ELECTREND. Stykače – co byste o nich měli vědět. In: *Electrend* [online]. Electrend, 2019 [cit. 2023-05-14]. Dostupné z: <https://www.legrand.cz/electrend/zabezpeceni/stykace-co-byste-o-nich-meli-vedet/>
- [21] Solid state relé. In: *Wikipedia* [online]. Wikipedia [cit. 2023-05-14]. Dostupné z: https://cs.wikipedia.org/wiki/Solid_state_rel%C3%A9
- [22] Elektromagnetické relé. In: *Wikipedia* [online]. Wikipedia [cit. 2023-05-14]. Dostupné z: https://cs.wikipedia.org/wiki/Elektromagnetick%C3%A9_rel%C3%A9

- [23] Stykač. In: *Wikipedia* [online]. Wikipedia [cit. 2023-05-14]. Dostupné z: <https://cs.wikipedia.org/wiki/Styka%C4%8D>
- [24] Pulzně šířková modulace. In: *Wikipedia* [online]. Wikipedia [cit. 2023-05-14]. Dostupné z: https://cs.wikipedia.org/wiki/Pulzn%C4%9B_%C5%A1%C3%AD%C5%99kov%C3%A1_modulace
- [25] *Solid State Relé* [online]. Verze 1.0. ECLIPSERA s.r.o, 2017 [cit. 2023-05-14]. Dostupné z: <https://dratek.cz/docs/produkty/0/85/1466628529.pdf>
- [26] Relé SSR Solid State SSR-25DA 380V. In: *Drátek.cz* [online]. ECLIPSERA [cit. 2023-05-14]. Dostupné z: <https://dratek.cz/arduino/1344-rele-ssr-solid-state-24v-380v-25a-ssr-25da-pro-pid-regulator-teploty.html>
- [27] *IIC I2C OLED display 1,3" 128x64 Bílý* [online]. Verze 1.0. ECLIPSERA, 2017 [cit. 2023-05-14]. Dostupné z: <https://dratek.cz/docs/produkty/0/735/1527497643.pdf>
- [28] Dvoupolohová regulace. In: *Ústav počítačové a řídicí techniky VŠCHT Praha* [online]. Praha, 2013 [cit. 2023-05-14]. Dostupné z: uprt.vscht.cz/ucebnice/mrt/F3/F3k36-dvrg.htm
- [29] Getting Started with MicroPython on ESP32 and ESP8266. In: *RandomNerdTutorials* [online]. RandomNerdTutorials, 2013 [cit. 2023-05-14]. Dostupné z: <https://randomnerdtutorials.com/getting-started-micropython-esp32-esp8266/>
- [30] ACSANY, Philipp. HTML and CSS for Python Developers. In: *Real Python* [online]. Real Python, 2012 [cit. 2023-05-14]. Dostupné z: <https://realpython.com/html-css-python/>
- [31] ANNAMAA, Aivar. Introducing Thonny, a Python IDE for learning programming. In: *The ACM Digital Library* [online]. The ACM Digital Library, 2023 [cit. 2023-05-14]. Dostupné z: <https://dl.acm.org/doi/10.1145/2828959.2828969>
- [32] Getting Started with Thonny MicroPython (Python) IDE for ESP32 and ESP8266. In: *RandomNerdTutorials* [online]. RandomNerdTutorials, 2013 [cit. 2023-05-14]. Dostupné z: <https://randomnerdtutorials.com/getting-started-thonny-micropython-python-ide-esp32-esp8266/>

- [33] EAGLE (program). In: *Wikipedia* [online]. Wikipedia [cit. 2023-05-14]. Dostupné z: [https://en.wikipedia.org/wiki/EAGLE_\(program\)](https://en.wikipedia.org/wiki/EAGLE_(program))
- [34] Autodesk Inventor. In: *3D inventorblog* [online]. Adeon CZ, 2012 [cit. 2023-05-14]. Dostupné z: <https://www.inventor3dblog.cz/inventor/>
- [35] ProfiCAD. In: *ProfiCAD* [online]. Coto Technology, 2003 [cit. 2023-05-16]. Dostupné z: <https://www.proficad.cz/>
- [36] JEDLIČKA, Václav. ProfiCAD – software pro tvorbu elektrotechnické dokumentace. In: *ElektroPrůmysl.cz* [online]. 2023 [cit. 2023-05-16]. Dostupné z: <https://www.elektroprumysl.cz/software/proficad-rychly-system-pro-tvorbu-elektrotechnicke-dokumentace>
- [37] Uzení. In: *Wikipedia* [online]. Wikipedia [cit. 2023-05-14]. Dostupné z: <https://cs.wikipedia.org/wiki/Uzen%C3%AD>
- [38] Jak udit klobásy teplým kouřem?. In: *GrilyKrby.cz* [online]. GrilyKrby.cz, 2015 [cit. 2023-05-14]. Dostupné z: <https://www.grilykrby.cz/otazky-a-odpovedi/jak-udit-klobasy-teplym-kourem>
- [39] Metody uzení - studený, teplý nebo horký kouř?. In: *Grilovani.cz* [online]. Grilovani.cz, 2008 [cit. 2023-05-14]. Dostupné z: <https://www.grilovani.cz/zakladni-rady-a-tipy/metody-uzeni-studený--teply-nebo-horky-kour-/>
- [40] PONEŠOVÁ, Anita. Krkovice z domácí udírny: jak ji naložit a vyudit bez rychlosoli. In: *Proženy.cz* [online]. Proženy.cz [cit. 2023-05-14]. Dostupné z: <https://www.prozeny.cz/clanek/krkovice-z-domaci-udirny-jak-ji-nalozit-a-vyudit-bez-rychlosoli-78544>
- [41] Wall socket single SIEMENS IP44. In: *GrabCad.com* [online]. GrabCad.com, 2023 [cit. 2023-05-14]. Dostupné z: https://grabcad.com/library/wall-socket-single-siemens-ip44-1?fbclid=IwAR3utk6B1M5Vp2FwWlOM23sMQaQqZ3VMDEP34_9qiLEG_ZOzYy6bTvFvicA
- [42] BLAŽKOVÁ, Ivana. Kvalifikace pracovníků v elektrotechnice dle vyhlášky č. 50/1978 Sb. In: *BOZPinfo.cz* [online]. Výzkumný ústav bezpečnosti práce, 2002 [cit. 2023-05-14]. Dostupné z: <https://www.bozpinfo.cz/kvalifikace-pracovniku-v-elektrotechnice-dle-vyhlasky-c-501978-sb>

- [43] Proudový chránič s jističem 16A PFL7-16/1N/B/003 Eaton. In: *Palmat.cz* [online]. Palmat.cz, 2023 [cit. 2023-05-14]. Dostupné z: https://www.palmat.cz/proudovy-chranic-s-jisticem-16a-pfl7-16-1n-b-003-eaton/?gclid=CjwKCAjwjYKjBhB5EiwAiFdSfsFJEsonoVzXCjIh_ZFCMlnojDnYX9ZIQQG4gwyjVko5sZahQ16rkBoC5w4QAvD_BwE
- [44] GEORGE, Damien P. a Paul SOKOLOVSKY. *MicroPython Documentation* [online]. V1.20.0. 2023 [cit. 2023-05-15]. Dostupné z: <https://docs.micropython.org/en/v1.20.0/micropython-docs.pdf>
- [45] SH1106: sh1106.py. In: *GitHub.com* [online]. GitHub, 2023 [cit. 2023-05-15]. Dostupné z: <https://github.com/robert-hh/SH1106/blob/master/sh1106.py>
- [46] Font Awesome Free 6.1.0. In: *Fontawesome.com* [online]. Fonticons, 2022 [cit. 2023-05-14]. Dostupné z: <https://cdn.jsdelivr.net/npm/font-awesome@6.1.0/css/all.min.css>
- [47] GEORGE, Damien P. a Paul SOKOLOVSKY. *MicroPython documentation* [online]. In: . [cit. 2023-05-15]. Dostupné z: <https://docs.micropython.org>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

| | |
|--------------------------|---|
| I/O | Input/Output |
| LED | Light Emitting Diode |
| RAM | Random Access Memory |
| GPIO | General Purpose Input/Output |
| BLE | Bluetooth Low Energy |
| BR/EDR | Basic Rate/Enhanced Data Rate |
| ADC | Analog-to-Digital Converter |
| PWM | Pulse Width Modulation |
| DC | Direct Current |
| MCU | Microcontroller Unit |
| CPU | Central Processing Unit |
| SD card | Secure Digital card |
| SPI | Serial Peripheral Interface |
| UART | Universal Asynchronous Receiver-Transmitter |
| I ² S | Inter-IC Sound |
| I ² C/IIC/I2C | Inter-Integrated Circuit |
| FreeRTOS | Free Real-Time Operating System |
| LwIP | Lightweight Internet Protocol |
| TLS | Transport Layer Security |
| OTA | Over-The-Air |
| TCP/IP | Transmission Control Protocol/Internet Protocol |
| GND | Ground |
| VIN | Voltage IN |
| USB | Universal Serial Bus |
| RTD | Resistance Temperature Detector |

| | |
|-------|--|
| NTC | Negative Temperature Coefficient |
| SSR | Solid state relé |
| EMR | Electromechanical Relay |
| VVC | Supply Voltage |
| SCL | Serial Clock |
| SDA | Serial Data |
| CSS | Cascading Style Sheets |
| IDE | Integrated Development Environment |
| EAGLE | Easily Applicable Graphical Layout Editor |
| PDF | Portable Document Format |
| DXF | Drawing Exchange Format |
| DWG | Drawing |
| ISO | International Organization for Standardization |
| 3D | Three-Dimensional |
| 2D | Two-Dimensional |
| DPS | Deska plošných spojů |
| CYSY | Označení kabeláže |
| AC | Alternating Current |
| OLED | Organic Light-Emitting Diode |
| CYA | Označení vodiče |
| IP | IP Address |
| BSD | Berkeley Software Distribution |
| UDP | User Datagram Protocol |
| URL | Uniform Resource Locator |
| HTML | Hypertext Markup Language |

SEZNAM OBRÁZKŮ

| | |
|---|----|
| Obrázek 1: ESP32 DevKit V1 [7]..... | 15 |
| Obrázek 2: ESP32 DEVKIT V1 vývody. [7] | 19 |
| Obrázek 3: Teplotní snímač DS18B20 ve voděodolné verzi [18] | 22 |
| Obrázek 4: Relé Solid State SSR-25DA 380 V [26] | 26 |
| Obrázek 5: Princip práce dvoupolohového regulátoru s hysterezí. [28]..... | 28 |
| Obrázek 6: Thonny IDE..... | 30 |
| Obrázek 7: Výkres základny | 35 |
| Obrázek 8: Výkres generátoru kouře | 36 |
| Obrázek 9: Výkres samostatné udírny | 37 |
| Obrázek 10: 3D model automatické udírny | 39 |
| Obrázek 11: Proudový chránič s jističem 16 A [43]..... | 41 |
| Obrázek 12: Hlavní napájení | 42 |
| Obrázek 13: Řídicí rozvaděč..... | 44 |
| Obrázek 14: ESP32 DevKit V1 v programu EAGLE..... | 47 |
| Obrázek 15: Zrealizovaná deska plošných spojů..... | 49 |
| Obrázek 16: Funkce pro připojení k WiFi síti | 51 |
| Obrázek 17: Import knihoven | 52 |
| Obrázek 18: Nastavení pinu pro relé, teplotního čidla a OLED displeje | 53 |
| Obrázek 19: Uživatelské rozhraní na webové stránce | 55 |
| Obrázek 20: listen_for_client()..... | 57 |
| Obrázek 21: Funkce update_display | 59 |
| Obrázek 22: Zobrazení hodnot na OLED displeji | 59 |
| Obrázek 23: Průběh žádané teploty. | 60 |
| Obrázek 24: Kompletní sestava automatické udírny | 62 |

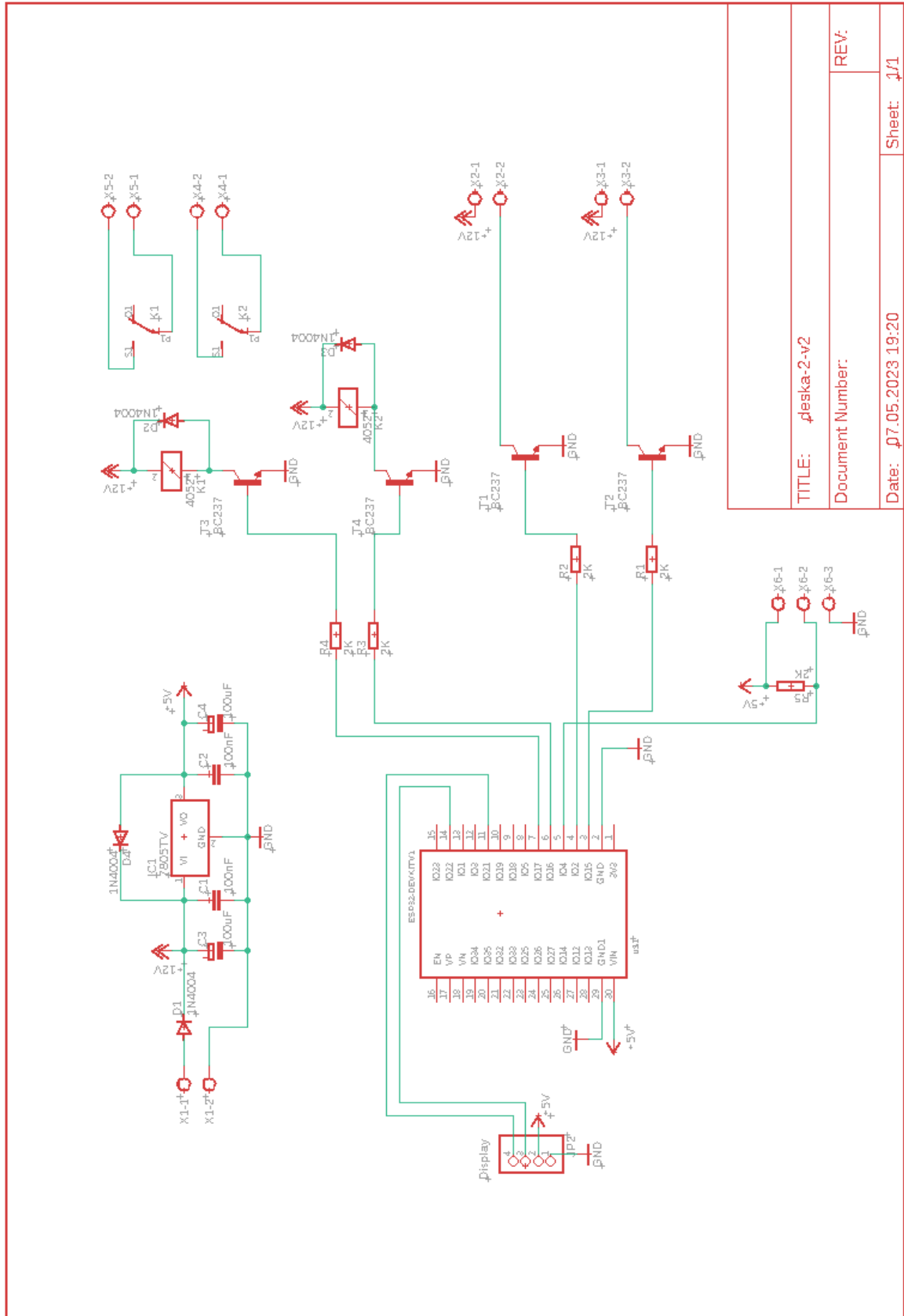
SEZNAM TABULEK

| | |
|--|----|
| Tabulka 1: Porovnání ESP32 a Arduino UNO+WiFi [4] [5] | 14 |
| Tabulka 2: Technické parametry ESP32 [4]..... | 16 |
| Tabulka 3: Parametry jednotlivých čidel | 21 |
| Tabulka 4: Specifikace teplotního senzoru DS18B20 [17] | 22 |
| Tabulka 5: Porovnání spínacích prvků | 24 |
| Tabulka 6: Specifikace displeje IIC I2C OLED [27] | 27 |
| Tabulka 7: součástky pro výrobu řídicí jednotky | 45 |

SEZNAM PŘÍLOH

| | |
|----------------|--------------------------|
| Příloha P I: | Schéma řídicí jednotky |
| Příloha P II: | DPS řídicí jednotky |
| Příloha P III: | Osazení řídicí jednotky |
| Příloha P IV: | Schéma hlavního napájení |
| Příloha P V: | Schéma řídicí části |

PŘÍLOHA P I: SCHÉMA ŘÍDICÍ JEDNOTKY



TITLE: deska-2-v2

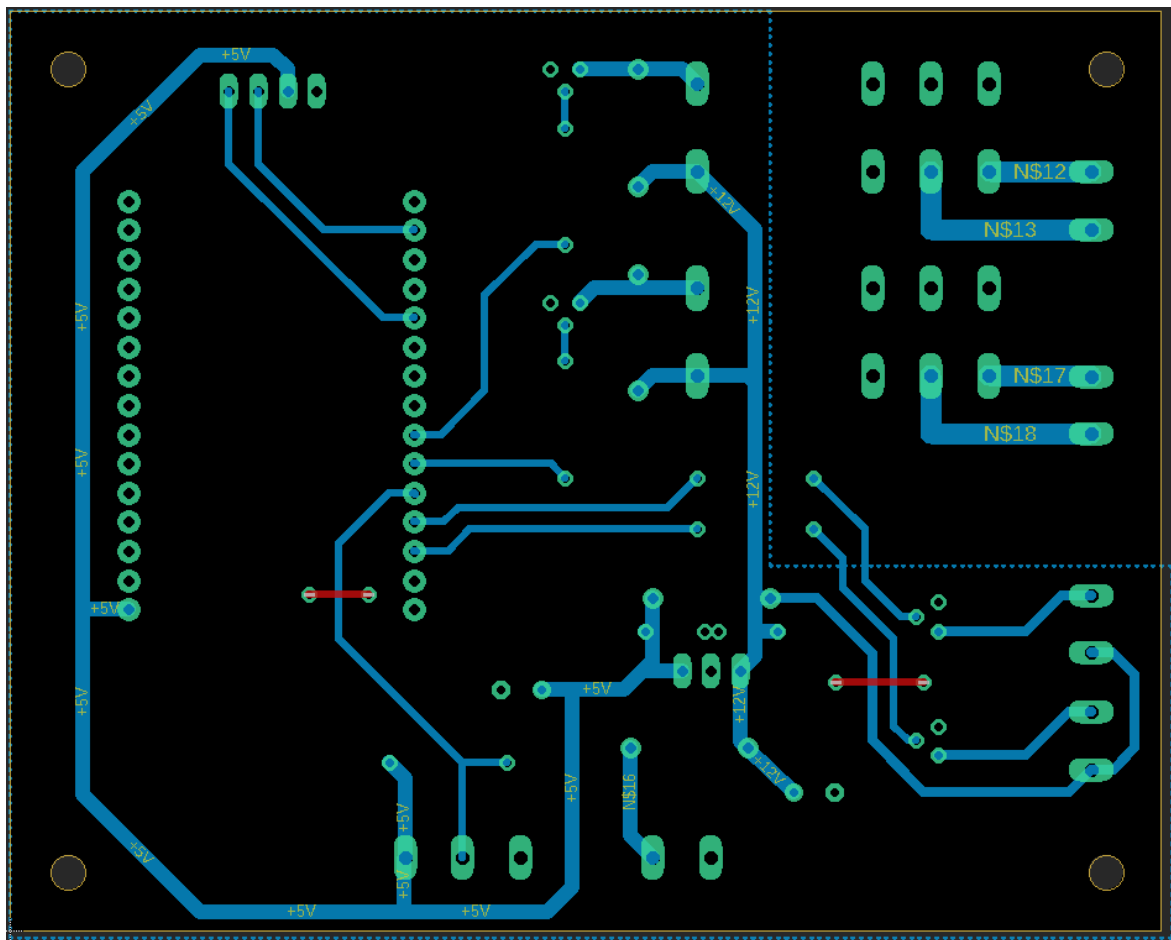
Document Number:

REV:

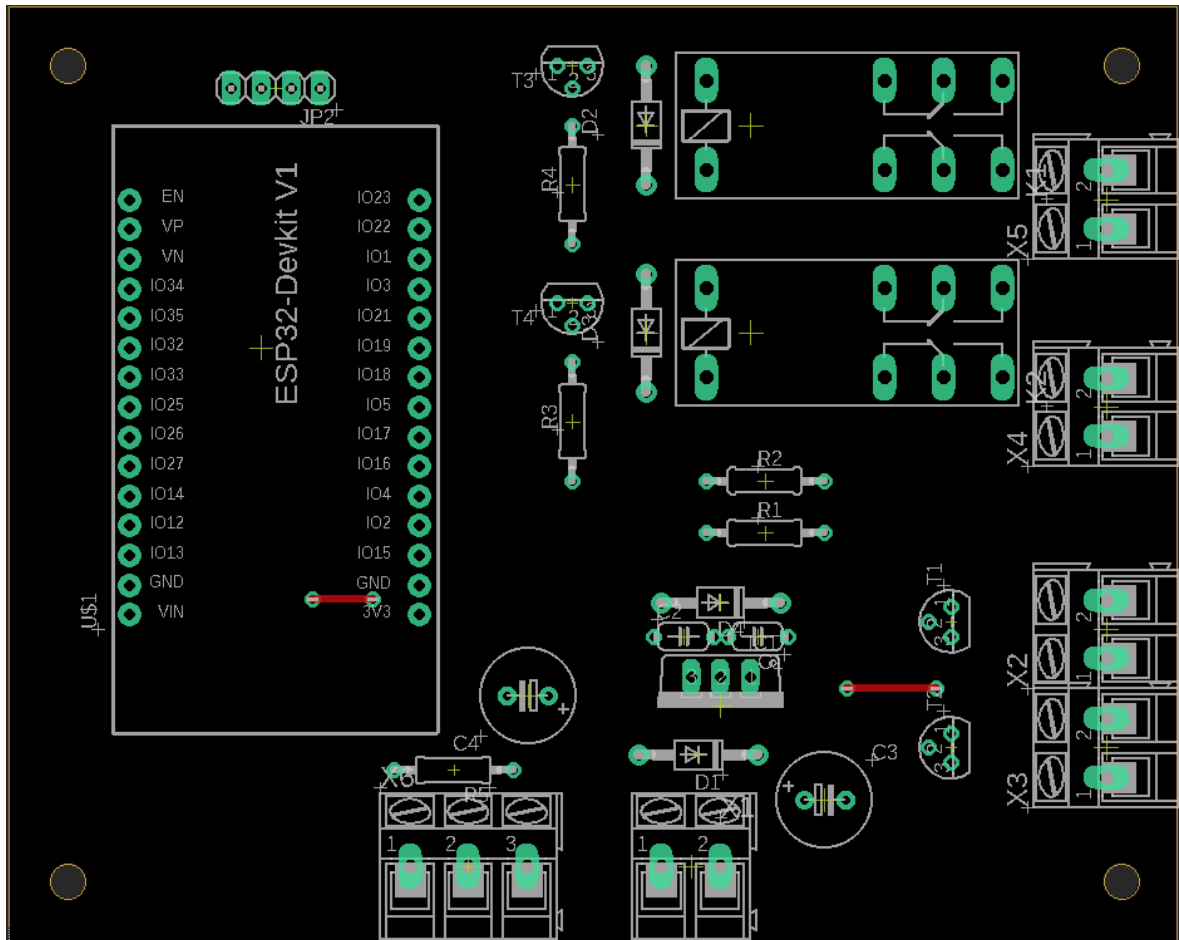
Date: 07.05.2023 19:20

Sheet: 1/1

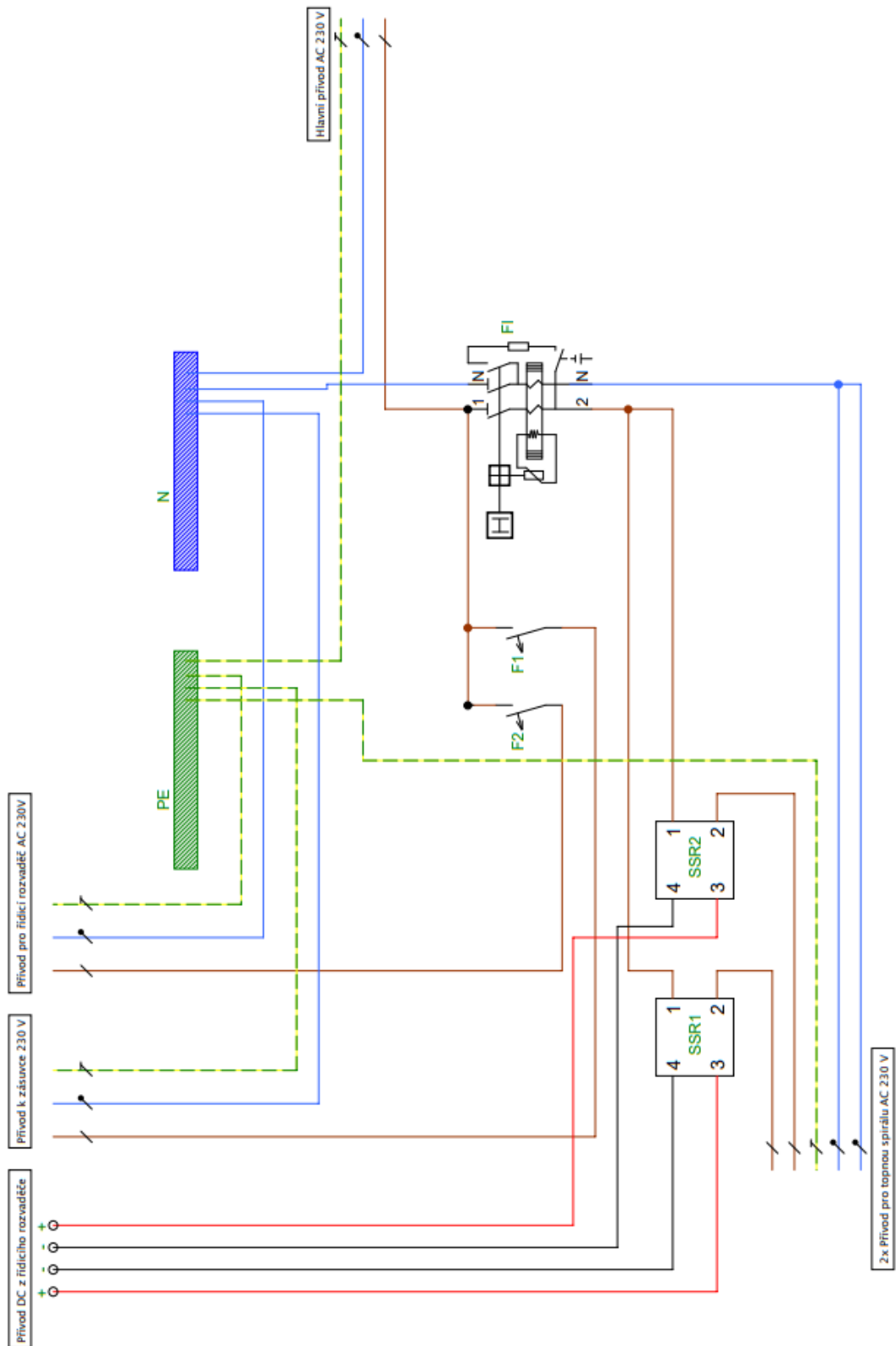
PŘÍLOHA P II: DPS ŘÍDICÍ JEDNOTKY



PŘÍLOHA P III: OSAZENÍ ŘÍDICÍ JEDNOTKY



PŘÍLOHA P IV: SCHÉMA HLAVNÍHO NAPÁJENÍ



PŘÍLOHA P V: SCHÉMA ŘÍDICÍ ČÁSTI

