

Zadání maturitní práce s obhajobou

**Číslo a název**: 01 - termoregulační podložka pod hrnek

**Jméno žáka**: Josef Jindra

**Konzultant**: Ing. Luděk Fedurca

**Oponent**: Bc. Jakub Malý

**Datum zadání**: 2. prosinec 2022

**Datum odevzdání**: 1. dubna 2023

**Doba obhajoby**: 15 minut

**Zadání:**

Navrhněte chytrý držák nápojů s regulací teploty pomocí Peltierova článku. Zařízení bude přes sériovou linku komunikovat s obslužnou aplikací v PC. Aplikace bude obsahovat aktuální stav zařízení a historii požadované a skutečné teploty za posledních 7 dnů v minutových intervalech.

**Způsob zpracování:**

● **Tištěná forma**: rozsah dokumentace 15–20 stran textu; v obálce s chlopněmi nebo pevná vazba; součástí práce bude úvodní obálka, zadání práce, harmonogram a prohlášení o souhlasu se zadáním práce, samostatnosti zpracování práce a použitím legálního software

● **Digitální forma**: kopie práce a pracovní soubory, dokumentace, prezentace na přiloženém CD nebo DVD v papírové obálce s jednoduchým HTML rozcestníkem; soubory v alternativních formátech

● **Model projektu**: vytvořte funkční model, který bude simulovat zadání

**Počet vyhotovení:** 1

**Formální úprava práce:**

**Písmo:** velikost 12

**Font:** Calibri, Arial nebo Times New Roman (zvolený font dodržte v celé práci)

**Řádkování:** 1,5

**Vzdálenost mezi odstavci:** 6 b.

**Okraje:** horní a dolní 25 mm, levý (vnitřní) 40 mm, pravý (vnější) 20 mm

**Zarovnání odstavce:** do bloku

**Číslování stránek:** vpravo dolu

**Začátek hlavní kapitoly:** vždy na nové straně

Dodržení typografických pravidel hladké sazby

1

**Hodnocení:**

1. Splnění zadání

2. Plnění plánu práce a účast na konzultacích

3. Aktuálnost a přínosnost tématu

4. Odborná úroveň práce, kvalita zpracování práce, použité prostředky 5. Zpracování dokumentace – typografie, zdroje, struktura, rozsah…

6. Dodržení ČSN ISO 690 a ČSN ISO 690-2 – bibliografické citace dokumentů 7. Hodnocení modelu a prezentace

**Podpis žáka** ……………………………………………………………………………. **Podpis konzultanta** ……………………………………………………………………………. **Podpis ředitele školy** …………………………………………………………………………….

2

| **Harmonogram práce MZ** |
| --- |

| **Třída:** 4. I |
| --- |
| **Studijní obor:** 18-20-M/01 Informační technologie |
| **Jméno studenta:** Josef Jindra |
| **Konzultant:** Ing. Luděk Fedurca |
| **Číslo a název úlohy:** 01 - termoregulační podložka pod hrnek |

| **Plán práce** | |
| --- | --- |
| **Týden** | **Práce** |
| 16.1.-21.1 2023 | osnova závěrečné práce, náplň kapitol prezentace prototypu |
| 6.2.-10.2 2023 | prezentace funkčnosti minimálního prototypu, 50% dokumentace |
| 27.2.-3.3. 2023 | cvičná obhajoba, finální text k revizi |

**Ve Štětí dne ………………………………………**

**Podpis ………………………………………**

| **Kontrola plnění plánu práce** | | |
| --- | --- | --- |
| **Datum** | **Poznámky** | **Podpis** |
|  |  |  |
|  |  |  |
|  |  |  |

3

**Prohlášení**

**Třída:** 4. I

**Studijní obor:** 18-20-M/01 Informační technologie

**Jméno žáka:** Josef Jindra

**Konzultant:** Ing. Luděk Fedurca

**Číslo a název úlohy:** 01 - termoregulační podložka pod hrnek

***Čestné prohlášení o souhlasu se zadáním maturitní práce***

Prohlašuji, že jsem se seznámil s obsahem zadání maturitní práce. Souhlasím se zadaným tématem.1 Ve Štětí dne 2. 12. 2022 Podpis: ……………………

***Čestné prohlášení o samostatnosti zpracování maturitní práce***

Prohlašuji, že jsem odevzdanou maturitní práci vypracoval samostatně a uvedl jsem všechny použité zdroje. Uvědomuji si, že prokáže-li se opak, může být má práce hodnocena jako nedostatečná.

Ve Štětí dne 2. 12. 2022 Podpis: ………………………………

***Čestné prohlášení o použití legálního softwarového vybavení***

Prohlašuji, že veškeré programové vybavení, které bylo použito při řešení této maturitní práce, bylo užito v souladu s jeho licencí.

Ve Štětí dne 2.12.2022 Podpis: ……………………………………

1 V případě nesouhlasu se zadáním maturitní práce se v den zadávání žák písemně obrátí na ředitele VOŠ, SPŠ, SOŠS a CR s odůvodněním svého nesouhlasu.

4

1. Obsah

Obsah

1. Obsah................................................................................................................................................. 5 2. Popis projektu.................................................................................................................................... 7 3. Hardware ........................................................................................................................................... 7 3.1. Napájení ......................................................................................................................................... 7 3.1.1. Jak je zařízení napájeno.............................................................................................................. 7 3.2. H-můstek ........................................................................................................................................ 7 3.2.1. Zakomponování do projektu ...................................................................................................... 7 3.3. Peltierův článek.............................................................................................................................. 7 3.3.1. Zakomponování do projektu ...................................................................................................... 8 3.4. Chlazení .......................................................................................................................................... 8 3.4.1. Hliníkový chladič......................................................................................................................... 8 3.4.2. Ventilátor FFC0612DE-PP01 ....................................................................................................... 8 3.4.3. Zakomponování do projektu ...................................................................................................... 8 3.5. Arduino Orion................................................................................................................................. 8 3.5.1. Zakomponování do projektu ...................................................................................................... 9 3.6. Me 7-segment serial Display.......................................................................................................... 9 3.6.1. Zakomponování do projektu ...................................................................................................... 9 3.7. Me RGB LED ................................................................................................................................... 9 3.7.1. Zakomponování do projektu ...................................................................................................... 9 3.8. Me RJ25 Adapter.......................................................................................................................... 10 3.8.1. Zakomponování do projektu .................................................................................................... 10 3.9. Me Temperature sensor............................................................................................................... 10 3.9.1. Zakomponování do projektu .................................................................................................... 10 4. 3D tisk .............................................................................................................................................. 10 4.1. Materiály na 3D Tisk..................................................................................................................... 11 4.1.1. PETG.......................................................................................................................................... 11 4.1.2. PLA............................................................................................................................................ 11 4.1.3. ABS............................................................................................................................................ 11 4.1.4. Filaflex....................................................................................................................................... 11 4.1.5. ASA............................................................................................................................................ 11 4.2. Zakomponování do projektu ........................................................................................................ 11 5. Knihovny .......................................................................................................................................... 12

5

5.1. Python .......................................................................................................................................... 12 5.1.1. Threading.................................................................................................................................. 12 5.1.2. Tkinter....................................................................................................................................... 12 5.1.3. Seriál......................................................................................................................................... 12 5.1.4. Time.......................................................................................................................................... 12 5.1.5. Requests................................................................................................................................... 13 5.1.6. Serial.tools.list\_ports................................................................................................................ 13 5.2. Arduino C...................................................................................................................................... 13 5.2.1. MeOrion.h ................................................................................................................................ 13 5.2.2. Arduino.h.................................................................................................................................. 13 6. Popis kódu........................................................................................................................................ 14 6.1. Python .......................................................................................................................................... 14 6.2. Arduino C...................................................................................................................................... 17 7. Vývoj ................................................................................................................................................ 19 7.1. Hlavní myšlenka............................................................................................................................ 19 7.2. začátek.......................................................................................................................................... 19 7.3. Finální produkt............................................................................................................................. 19 8. Zdrojový kód .................................................................................................................................... 20

6

2. Popis projektu

Smart cup neboli Chytrý hrnek je projekt, který umožní uživateli ohřát nebo ochladit nápoj dle jeho přání. Uživatel si nejprve nastaví požadovanou teplotu nápoje pomocí Pythonovské aplikace, která bude propojena s Arduinem. Arduino bude sloužit jako řídící jednotka, která bude přijímat data o nastavené teplotě z aplikace. Pro ohřev nebo ochlazení nápoje bude využit Peltierův článek, který umožní regulaci teploty v rozmezí 10 až 70 stupňů. Peltierův článek bude umístěn pod hrnkem a bude poháněn pomocí napájecího zdroje.

3. Hardware

3.1.Napájení

Projekt je napájen 12V zdrojem stejnosměrného napětí o minimálním napětí 4A tudíž příkonem 48W. Zařízení je k napájení připojen konektorem 5,5 \*2,1mm pro 12V.

3.1.1. Jak je zařízení napájeno

Zařízení je k napájení připojen konektorem 5,5 \*2,1mm pro 12V. které se rozděluje na dva hlavní kanály do arduina Orion a do H-můstku kde se následně proud posílá jak do Peltierova článku, tak do ventilátoru chladiče.

3.2.H-můstek

H-můstek je tvořen čtyřmi spínači, které jsou propojeny tak, aby tvořily podoby písmene H. Každý spínač může být buď zapnutý (zkratuje své dva terminály) nebo vypnutý (odděluje své dva terminály). Tento obvod umožňuje vytvořit dva vstupy (A a B) pro ovládání polarity napájení, což umožňuje výměnu plus a mínus.

3.2.1. Zakomponování do projektu

Do H-můstku byl přiveden elektrický proud z 12V zdroje zařízení poté posílá proud do ventilátoru chladiče nebo do Peltierova článku kde v tomto případě pomocí signálu z arduina přepíná plus a mínus kvůli změnám vlastností tohoto článku, a to ohřívaní nebo chlazení.

3.3.Peltierův článek

Peltierův článek (někdy nazývaný také termoelektrický článek) je elektrické zařízení, které využívá Peltierův jev k vytváření teplotního rozdílu mezi dvěma páry vodičů. Peltierův jev spočívá v tom, že když elektrický proud protéká spojením dvou různých kovů, tak se na spojeních vytvoří teplotní rozdíl. Peltierův článek je složen z několika vrstev polovodičů, které jsou spojeny do série a připojeny k napájecímu zdroji. Když se na Peltierův článek přivede napětí, začne protékat elektrický proud, což způsobí přenos tepla z jedné strany článku na druhou. Výhodou je že strany, které buď chladí nebo ohřívají se dají měnit pomocí změny toku elektrického proudu, a to za plus na minus a z mínusu na plus.

7

3.3.1. Zakomponování do projektu

Peltierův článek je zde použit na chlazení nebo ohřev nápoje umístěném v hrnku, samotný článek je umístěn pod hrnkem a je napájen z H-můstku který na základě signálu přepíná směr toku elektrického proudu. Článek je schopen nápoj v hrnku ochladit nebo ohřát nápoj, a to od deseti až sedmdesáti stupňů Celsia.

Pro tento projekt byl zvolen Peltierův článek 40x40mm TEC1-12706.

3.4.Chlazení

3.4.1. Hliníkový chladič

Hliníkový chladič neboli pasivní chladič je zařízení používané ke snižování teploty v elektronických zařízeních, jako jsou počítače, automobily a další spotřebiče. Hliníkové chladiče jsou vyrobeny z hliníkových slitin, které jsou lehké, odolné proti korozi a mají vynikající tepelnou vodivost. Hliníkový chladič se skládá z hliníkového bloku s množstvím žeber, které zvyšují povrch chlazení a umožňují rychlejší odvádění tepla do okolního vzduchu. Tento blok je připojen ke zdroji tepla, jako je například procesor v počítači, pomocí tepelné pasty nebo tepelného rozhraní. Tepelná pasta nebo rozhraní umožňuje vytvoření co nejlepšího kontaktu mezi chladičem a zdrojem tepla.

3.4.2. Ventilátor FFC0612DE-PP01

FFC0612DE-PP01 je ventilátor s průměrem 60 mm, který je určen pro použití v elektronice a dalších oblastech, kde je potřeba výkonný a spolehlivý chlazení. Ventilátor má hladké a tiché lopatky s nízkým šumem, které jsou vyrobeny z kvalitního plastu a jsou odolné vůči korozi. Ventilátor je poháněn motorem s vysokou účinností, který zajistí vysoký průtok vzduchu a vynikající chlazení. Tento ventilátor má nízký napěťový rozsah, což znamená, že může být napájen napětím 12 V. Má vysokou rychlost otáčení a dosahuje až 5300 otáček za minutu, což zajistí vysoký objem vzduchu.

3.4.3. Zakomponování do projektu

Bylo zde využito pasivně-aktivní chlazení na principu toho že ventilátor který je umístěn ve větrném tunelu který začíná komolým kuželem který postupně zmenšuje průměr tunelu ve kterém je umístěn pasivní chladič, tím že se zmenšuje průměr tunelu se nám zvyšuj efektivita proudícího vzduchu a dochází tak k lepšímu ochlazení chladiče tudíž i Peltierova článku.

3.5.Arduino Orion

Makeblock Arduino Orion je základní deska pro programování robotických projektů společnosti Makeblock je založený na čipu ATmega328P. Jedná se o kompatibilní náhradu za desku Arduino UNO nebo Mega. Makeblock Orion má v sobě integrovaný mikrokontrolér, který umožňuje připojení různých senzorů a ovládání motorů. Deska také obsahuje osm konektorů RJ25 pro připojení senzorů dva porty pro připojení motorů a pro propojení s počítačem používá konektor USB Micro. Desku lze napájet 12V. Makeblock Orion je snadno programovatelný pomocí mnoha programovacích jazyků, včetně Scratch, Python a Arduino IDE. Díky tomu je vhodný pro různé uživatele od začátečníků až po pokročilé uživatele s programovacími zkušenostmi.

8

3.5.1. Zakomponování do projektu

Zařízení přijme data přes sériovou linku, a hodnotu zobrazí na display, přičemž si data od uživatele porovná s daty z teploměru a vyhodnotí, zda má chladit nebo ohřívat na základě vyhodnocení se pošle 5V buď na pin 9 nebo 3 tím se sepne Peltierův článek. Pro chlazení je zde nastaveno že na pinu 11 jde stále 5V tudíž je ventilátor vždy zapnutý. Po vyhodnocení se zařízení rozhoduje, jakou barvou má rozsvítit Led panel, modrá nápoje je moc studený, zelená nápoj je připraven a červená nápoj je moc horký, Arduino každou minutu posílá data aktuální teploty do počítače.

3.6. Me 7-segment serial Display

Me 7-segment Serial Display je displej s 4 číslicemi, který se používá v různých projektech pro zobrazení číselných hodnot. Displej je vybaven 4 vysokými svítivostí 7-segmentovými LED displeji, každý s vlastním řadičem. Display komunikuje pomocí sériové sběrnice a může být snadno připojen k desce Makeblock Orion nebo jiným mikrokontrolerům s UART rozhraním. Displej je kompatibilní s většinou mikrokontrolerů, jako jsou Arduino, Raspberry Pi a další. Displej je ideální pro použití v různých projektech, jako jsou hodiny, měřiče teploty, časovače a další projekty, které vyžadují zobrazení číselných hodnot.

3.6.1. Zakomponování do projektu

Display je zapojen do Arduina přes konektor RJ25 a vyobrazuje hodnoty které přijdou na již zmíněnou desku po sériové lince.

3.7.Me RGB LED

Me RGB LED je komponenta pro rozšíření systému Makeblock, která obsahuje tři samostatné LED diody – červenou, zelenou a modrou. Tento modul lze snadno připojit k desce Makeblock Orion nebo jiným mikrokontrolerům. Každá z LED diod v modulu Me RGB LED lze ovládat nezávisle a umožňuje tak různé barevné kombinace. Celková barva, kterou vytváří modul, se může snadno měnit programově, což umožňuje rychlé a snadné vytvoření různých barevných efektů. Modul je snadno připojitelný a umožňuje snadné použití v různých projektech, jako jsou osvětlení, indikátory stavu, nebo jako součást robotických projektů pro signalizaci různých stavů. Modul poskytuje rozmanitost a flexibilitu při tvorbě barevných efektů a zajišťuje kvalitní a spolehlivé osvětlení.

3.7.1. Zakomponování do projektu

Podle stavu hrnku, kdy se porovnávají hodnoty nastavené od uživatele se rozsvítí led panel, červená nápoj je horký zelené nápoj má požadovanou teplotu a modrá nápoj je studený. Nastavení RGB je zapsáno v desítkové soustavě.

9

3.8.Me RJ25 Adapter

Me RJ25 Adapter je komponenta pro rozšíření systému Makeblock, která umožňuje připojení senzorů k desce Makeblock Orion nebo jiným mikrokontrolerům pomocí kabelů RJ25. Adapter má na sobě 6 portů které jsou propojeny s konektorem RJ25, které umožňují snadné připojení senzorů Makeblock pomocí kabelů s konektory RJ25 bez nutnosti svařování nebo jiných složitých úkonů.

3.8.1. Zakomponování do projektu

Přes modul Me RJ25 Adapter je připojen teploměr, který má pouze konektor plus, data a gnd.

3.9.Me Temperature sensor

Me temperature sensor je senzor pro měření teploty v systému Makeblock. Tento senzor je založen na čipu TMP36, který je snadno použitelný a má vysokou přesnost měření. Me temperature sensor je snadno připojitelný k desce Makeblock Orion nebo jinému mikrokontroleru pomocí kabelu s konektorem RJ25, což umožňuje snadné připojení a použití. Senzor může být použit pro měření teploty v rozsahu -40°C až +125°C s přesností ±1°C. Me temperature sensor poskytuje digitální výstup, který lze snadno přečíst pomocí programování v různých jazycích, jako je například Arduino nebo Scratch. Me temperature sensor může být použit v různých projektech, jako jsou například inteligentní domy, systémy řízení prostředí nebo jako součást robotických projektů pro monitorování teploty různých prostor. Senzor poskytuje snadné řešení pro měření teploty v různých aplikacích a umožňuje snadnou integraci do systému Makeblock.

3.9.1. Zakomponování do projektu

Teploměr Me temperature sensor sbírá hodnotu teploty nápoje tak že je do kapaliny ponořen, poté co získá data teploty pošle je na Arduino, které už je dále pošle na počítač, přičemž rozhodne, zda chladit či ohřívat. Senzor je připojen do Me RJ25 Adapter, který je následně připojen do Arduina.

4. 3D tisk

3D tisk je moderní technologie, která umožňuje vytvářet fyzické objekty z digitálních modelů pomocí vrstvení materiálů. Většina 3D tisků využívá tzv. aditivní výroby, kdy jsou jednotlivé vrstvy materiálu postupně nanášeny na sebe, dokud není vytvořen kompletní objekt. Tento proces začíná vytvořením

digitálního modelu, který může být vytvořen pomocí CAD software, nebo lze použít 3D skenování pro získání dat z již existujícího objektu. Poté, co je digitální model hotov, je připraven k tisku. Během tiskového procesu se materiál, obvykle plast, rozehřeje a rozpustí. Poté se extruduje z trysky na platformu, kde postupně vytváří vrstvu po vrstvě. Tento proces se opakuje, dokud není hotov celý objekt. 3D tisk může být použit v mnoha různých oblastech, od průmyslové výroby po výrobu hraček nebo módních doplňků. 3D tisk umožňuje vytvářet objekty, které by byly jinak velmi obtížné nebo dokonce nemožné vyrobit pomocí tradičních metod. Díky rychlému vývoji technologie a zlepšení přesnosti a rychlosti tisku se 3D tisk stává stále více dostupný a užitečný pro širokou škálu aplikací.

10

4.1.Materiály na 3D Tisk

Filamenty jsou základním materiálem pro 3D tisk, který se používá k vytváření fyzických objektů. Zde jsou některé z nejpopulárnějších filamentů používaných v 3D tisku:

4.1.1. PETG

PETG (Polyethylentereftalat Glycol): PETG je termoplastický materiál, který se používá v 3D tisku kvůli své vysoké odolnosti vůči nárazům, prasknutí a UV záření. PETG je také recyklovatelný a netoxický, což z něj dělá ideální materiál pro tiskové aplikace, které vyžadují odolnost a výdrž.

4.1.2. PLA

PLA (Polylaktid): PLA je dalším velmi populárním materiálem pro 3D tisk. Tento materiál je biologicky rozložitelný, snadno se tiskne a je k dispozici v mnoha různých barvách. PLA má nízkou teplotu tání, což znamená, že pro jeho tisk není potřeba vysoce výkonná tiskárna.

4.1.3. ABS

ABS (Akrilonitril-butadien-styren): ABS je termoplastický polymer, který je velmi odolný proti nárazům, prasknutí a opotřebení. Tento materiál je také velmi odolný vůči teplu, což ho činí ideálním pro tisk větších a robustnějších objektů.

4.1.4. Filaflex

Filaflex: Filaflex je elastický materiál, který se používá k tisku pružných a gumových objektů. Tento materiál má vysokou flexibilitu a elasticitu, což ho činí ideálním pro výrobu sportovních doplňků, jako jsou např. chrániče nebo elastické pásy.

4.1.5. ASA

ASA (Akrylonitril-styren-akrylát): ASA je materiál, který se podobá ABS, ale má větší odolnost proti UV záření. Tento materiál je také odolný proti teplu a chemikáliím, což ho činí ideálním pro venkovní použití, jako jsou např. zahrádkářské potřeby, dopravní značky nebo průmyslové součástky.

4.2.Zakomponování do projektu

Pro tento projekt byl vytvořen větrný tunel pro chlazení a kryt celého zařízení, a to z matriálu PETG při teplotě 245° C

11

5. Knihovny

Knihovny v programování jsou kolekce předem napsaného kódu, který může být použit v jiných programech. Tyto knihovny obsahují funkce, procedury a třídy, které mohou usnadnit programování tím, že poskytují hotové řešení pro určité úkoly. Používání knihoven umožňuje programátorům šetřit čas tím, že nemusí psát kód pro každou jednotlivou část programu. Místo toho mohou využít hotové funkce, které jsou součástí knihovny. Knihovny také umožňují programátorům sdílet svůj kód s ostatními, což může zlepšit efektivitu vývoje softwaru. Existuje mnoho knihoven pro různé programovací jazyky a oblasti programování. Například pro jazyk Python existuje mnoho knihoven pro práci s daty, grafické rozhraní, webové aplikace, strojové učení a mnoho dalšího.

5.1.Python

5.1.1. Threading

Knihovna threading v Pythonu umožňuje programátorům vytvářet a pracovat s vlákny (threads) v jejich aplikacích. Vlákna jsou samostatné sekvence instrukcí, které mohou běžet současně s jinými vlákny v rámci stejného procesu. Tímto způsobem mohou programy vykonávat více úloh současně a zlepšovat výkon aplikace.

5.1.2. Tkinter

Knihovna tkinter v Pythonu je standardní knihovna pro tvorbu grafického uživatelského rozhraní (GUI). Tkinter poskytuje sadu widgetů, které umožňují programátorům vytvářet uživatelské rozhraní pro své aplikace. knihovna obsahuje různé typy widgetů, jako jsou tlačítka, pole pro vstup textu, záložky, seznamy a další. Tyto widgety lze kombinovat, aby vytvářely složitější uživatelské rozhraní. Tkinter také poskytuje metody pro reagování na uživatelské interakce s widgety, například kliknutí na tlačítko nebo změnu hodnoty v poli pro vstup textu. Programátor může definovat funkce, které budou vyvolány po těchto interakcích.

5.1.3. Seriál

Knihovna serial v Pythonu poskytuje jednoduché rozhraní pro komunikaci se sériovým portem. Tato knihovna umožňuje programátorům číst a zapisovat data z nebo do sériového portu, což je užitečné při práci s různými zařízeními, jako jsou například senzory, motory, roboti a další. Pro čtení a zápis dat z/do sériového portu knihovna serial poskytuje několik metod. Metoda read() umožňuje programátorovi přečíst data ze sériového portu, zatímco metoda write() umožňuje programátorovi zapsat data do sériového portu. Knihovna serial také obsahuje metody pro čtení a zápis řetězců, seznamů a binárních dat.

5.1.4. Time

Knihovna time v Pythonu poskytuje funkce pro práci s časem. Tato knihovna umožňuje programátorům měřit čas, zpoždění a řídit tok programu. Knihovna time obsahuje funkce, jako jsou například time(), sleep() a clock(). Metoda time() vrací aktuální čas jako počet sekund od 1. ledna 1970. Tato metoda je užitečná pro měření času běhu programu a časových intervalů mezi dvěma událostmi. Metoda sleep() umožňuje programátorovi pozastavit běh programu na určitý počet sekund. Metoda clock() vrací čas procesoru použitého pro běh programu.

12

**5.1.5.** Requests

Knihovna Requests v Pythonu je knihovna pro snadnou práci s HTTP požadavky. Knihovna Requests umožňuje programátorům snadno posílat HTTP požadavky (např. GET, POST, PUT, DELETE), získávat informace o stavu požadavku, zpracovávat odpovědi na požadavky a pracovat s cookies a autentizací.

5.1.6. Serial.tools.list\_ports

Knihovna serial v Pythonu poskytuje základní nástroje pro práci s sériovými porty. Knihovna serial.tools.list\_ports patří do podbalí nástrojů, které poskytují informace o sériových portech k dispozici na počítači. Funkce list\_ports() je metoda, která umožňuje programátorovi získat seznam všech dostupných sériových portů na počítači. Tento seznam obsahuje informace o každém sériovém portu, jako například jeho název, popis, výrobce a ID zařízení.

5.2.Arduino C

5.2.1. MeOrion.h

Knihovna MeOrion pro Arduiono je knihovna určená pro desku Makeblock Orion, která slouží jako řídící deska pro robotické projekty. Tato knihovna poskytuje jednoduché rozhraní pro programování desky Makeblock Orion a zahrnuje funkce pro ovládání různých periferií, jako jsou motory, senzory a displeje. Další důležitou třídou v knihovně je třída MeWire, která umožňuje programátorovi komunikovat s různými I2C zařízeními připojenými k desce Makeblock Orion. Tato třída poskytuje funkce pro čtení a zápis dat z různých I2C zařízení, jako jsou senzory teploty, vlhkosti a světla.

5.2.2. Arduino.h

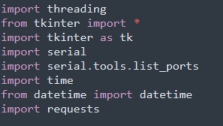
Arduino.h je základní knihovna pro Arduinu v jazyce C++. Tato knihovna obsahuje základní funkce a třídy pro práci s mikrokontroléry Arduino, jako jsou například funkce pro práci s pinovým nastavením, časovači, sériovou komunikací a mnoho dalších. Mezi základní funkce knihovny Arduino.h patří funkce pinMode(), digitalWrite() a digitalRead(). Tyto funkce umožňují programátorovi nastavit režim vstupu nebo výstupu pro jednotlivé piny na mikrokontroléru, a také číst a zapisovat digitální signály na těchto pinech. Další důležitou součástí knihovny Arduino.h jsou funkce pro práci s analogovými signály, jako je funkce analogRead(). Tato funkce umožňuje programátorovi číst hodnoty analogových signálů z analogových vstupů na mikrokontroléru.

13

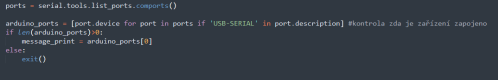
6. Popis kódu

6.1.Python

Tento kód slouží k ovládání chytrého hrnku pro nápoje. S využitím knihoven Tkinter, serial, serial.tools.list\_ports, time, datetime a requests umožňuje uživateli nastavit teplotu nápoje a ovládat ji pomocí tlačítek na grafickém rozhraní. Kromě toho také sbírá data o teplotě nápoje a odesílá je na online platformu ThingSpeak.



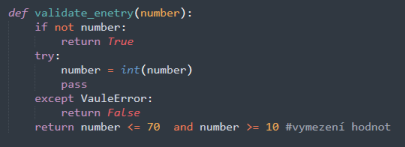
Knihovna tkinter slouží k vytvoření grafického rozhraní pro uživatele, serial k práci s sériovými porty pro komunikaci s chytrým hrnkem, serial.tools.list\_ports k získání seznamu dostupných sériových portů, time pro časování úloh a datetime pro zpracování dat a času. Knihovna requests pak slouží k odesílání dat na ThingSpeak.

Následuje kontrola, zda je chytrý hrnek připojen. Pokud ne, program se ukončí. 

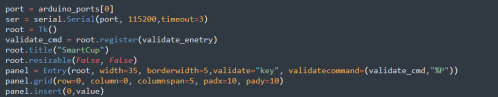
Proměnná ports získává seznam dostupných sériových portů. Poté se vytvoří seznam arduino\_ports, který obsahuje pouze ty porty, jejichž popis obsahuje řetězec "USB-SERIAL". Pokud seznam arduino\_ports není prázdný, získá se první prvek tohoto seznamu jako proměnná message\_print. Pokud je seznam arduino\_ports prázdný, program se ukončí.

14

Následuje definice funkce validate\_entry(number), která kontroluje vstupní hodnotu v poli pro nastavení teploty. Pokud se hodnota nedá převést na celé číslo, funkce vrátí False. Pokud je hodnota mimo rozsah mezi 10 a 70, funkce vrátí False. Jinak vrátí hodnotu True.



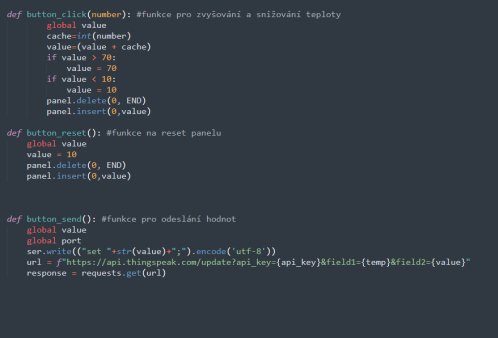
Následuje inicializace sériového portu pro komunikaci s Arduinem pomocí knihovny "serial". Dále se vytvoří okno aplikace pomocí knihovny Tkinter. Následně se vytvoří pole pro vstup teploty a validace, a tlačítka pro ovládání teploty.



Funkce "task" využívá nekonečné smyčky while k neustálému odesílání teploty a požadované teploty nápoje do ThingSpeak API. Nejprve se inicializují globální proměnné "urlt" a "ser". Poté se provede reset výstupního bufferu a odešle se příkaz "write;" na Arduino. Po krátké prodlevě se zkontroluje, zda je něco k dispozici v sériovém portu. Pokud ano, přečte se hodnota teploty a odešle se příkaz pro aktualizaci hodnoty na ThingSpeak API. Poté se smyčka

15

Zde jsou funkce pro ovládání teploty a odesílání hodnot na Arduino . Funkce button\_click(number) zvyšuje nebo snižuje teplotu podle hodnoty number, kterou dostane jako argument, a poté ověřuje, zda nová hodnota teploty je mezi 10 a 70 stupni. Pokud je hodnota mimo tuto rozmezí, nastaví se na nejbližší platnou hodnotu. Poté vkládá novou hodnotu do panelu. Funkce button\_reset() nastavuje teplotu na výchozí hodnotu 10 a vkládá ji do panelu. Funkce button\_send() odesílá hodnoty teploty a nastavené teploty na server ThingSpeak a Arduino .

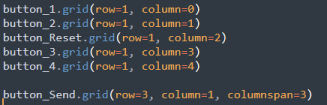


Zde se vytváří šest tlačítek s různými vlastnostmi, které jsou poté umístěny na okně aplikace pomocí metody grid(). Každé tlačítko má textovou etiketu, šířku a výšku (v pixelech) a funkci command, která se spustí po stisknutí tlačítka. Tlačítko button\_1 má text "+1" a zvyšuje teplotu o 1 stupňů Celsia. Jeho funkce command volá funkci button\_click() s argumentem 1. Tlačítko button\_2 má text "+10" a zvyšuje teplotu o 10 stupňů Celsia. Jeho funkce command volá funkci button\_click() s argumentem 10. Tlačítko button\_3 má text "-1" a snižuje teplotu o 1 stupňů Celsia. Jeho funkce command volá funkci button\_click() s argumentem -1. Tlačítko button\_4 má text "-10" a snižuje teplotu o 10 stupňů Celsia. Jeho funkce command volá funkci button\_click() s argumentem -10. Tlačítko button\_Send má text "Send" a odesílá aktuální teplotu pomocí funkce button\_send(). Tlačítko button\_Reset má text "Reset" a resetuje teplotu na výchozí hodnotu 10 stupňů Celsia pomocí funkce button\_reset().

16

Tato část kódu definuje umístění tlačítek v uživatelském rozhraní pomocí funkce grid() z modulu tkinter. Každé tlačítko je umístěno v určitém řádku a sloupci pomocí parametrů row a column. Například button\_1 je umístěno v prvním řádku a prvním sloupci (row=1, column=0), button\_2 je umístěno v prvním řádku a druhém sloupci (row=1, column=1) atd.

Také se zde používá parametr columnspan, který říká, kolik sloupců by mělo dané tlačítko zabrat. Například button\_Send je umístěno v řádku 3 a zabírá 3 sloupce (columnspan=3), protože by mělo být větší než ostatní tlačítka.

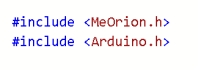


root.mainloop() spustí hlavní smyčku událostí pro tkinterové okno root. Tato smyčka čte vstup od uživatele (např. kliknutí na tlačítko)

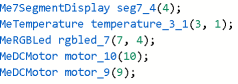
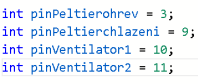


6.2.Arduino C

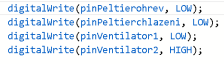
Tento kód obsahuje definice globálních proměnných a objektů z knihovny MeOrion.h pro komunikaci s hardwarovými komponentami připojenými k desce Me Orion.



Konkrétně jsou definovány 2 pinové proměnné pro Peltierův článek (používané pro ochlazování nebo ohřev), 2 pinové proměnné pro ventilátor, objekty pro 7segmentový displej (se zobrazováním čísel), teplotní čidlo a RGB LED diodu.

17

V metodě setup() jsou nastaveny výstupní piny a nastavení počátečního stavu (LOW pro všechny piny, kromě pinu pro ventilátor, který je nastaven na HIGH).



V metodě loop() se program nejprve snaží načíst příchozí data ze sériového portu. Pokud je k dispozici nějaký příkaz, tak program zkontroluje, zda se jedná o "write" a pokud ano, odešle přes sériovou linku hodnotu teploty, kterou získá z teplotního čidla. Pokud se jedná o "set", tak program zpracuje číslo, které je za příkazem "set" a přiřadí ho do proměnné "number".



Následně se program stará o řízení teploty pomocí Peltierových článků a ventilátorů. Pokud je naměřená teplota vyšší než hodnota v proměnné "number" + 1, tak se spustí ohřev a LED dioda se přepne na červenou. Pokud je teplota nižší než hodnota v proměnné "number" - 1, tak se spustí chlazení a LED dioda se přepne na modrou. Pokud je naměřená teplota v rozmezí hodnoty v proměnné "number" +/- 1, tak se vypne Peltierův článek a zelená LED dioda svítí zeleně.

18

7. Vývoj

7.1.Hlavní myšlenka

Tento projekt je aplikace pro kontrolu teploty nápojů v chytrém kelímku pomocí Arduina a Pythonu. Hlavní myšlenka projektu je vytvořit uživatelské rozhraní pro snadnou změnu teploty nápoje v kelímku a zároveň odeslat tuto teplotu a čas měření na cloudovou platformu ThingSpeak pro další analýzu dat. K tomuto účelu byla použita knihovna tkinter pro tvorbu GUI, knihovny serial a serial.tools.list\_ports pro práci s Arduinem, knihovna requests pro odesílání dat na ThingSpeak a knihovna threading pro spuštění vlákna pro průběžné měření a odesílání dat na ThingSpeak. Celkově je projekt zaměřen na jednoduchost použití a přehlednost uživatelského rozhraní.

7.2. začátek

Před zahájením práce na tomto projektu jsem si musel získat několik informací. První věc, kterou jsem musel zjistit, byla práce s moduly, které byly importovány pomocí několika knihoven, jako například threading, tkinter, serial, time a requests. Také jsem musel zjistit, jak získat API klíč pro ThingSpeak, který se používá k odesílání dat z projektu na platformu ThingSpeak. Dále jsem musel najít a vybrat správný port pro připojení vašeho zařízení. Nakonec jste museli definovat funkce, které provádějí akce v programu, jako například funkci validate\_entry pro validaci vstupních hodnot a funkce pro zvyšování a snižování teploty.

7.3. Finální produkt

Tento projekt se nazývá "SmartCup" a jedná se o programování mikrokontroléru Arduino, který je připojen k teploměru. Cílem projektu je vytvořit inteligentní podšálek, který udržuje nápoj na optimální teplotě a upozorní uživatele, když je nápoj připraven k pití. Programování zahrnuje použití knihoven jako jsou: threading, tkinter, serial, serial.tools.list\_ports, time, datetime a requests. Po spuštění programu se zjistí, zda je Arduino připojen, a pokud ano, je nastavena komunikace s ním. Uživatel může nastavit požadovanou teplotu a poté jsou tlačítky zajištěny funkce, které umožňují zvyšování a snižování teploty, odesílání nastavené teploty a resetování panelu s teplotou. V programu je také obsažen kód, který periodicky odesílá data (aktuální teplotu a nastavenou teplotu) na online platformu ThingSpeak, která uživateli umožňuje sledovat stav čajové konvice v reálném čase. Celkově je projekt "SmartCup" praktický a užitečný pro každodenní použití.

19

8. Zdrojový kód

Odkazy na GitHub:

Python aplikace: https://github.com/Hurbixcz/Smart-cup/blob/main/Samrtcup.py 

Arduino: https://github.com/Hurbixcz/Smart-cup/blob/main/Smartcup..mblock

20