

Gymnázium, Praha 7, Nad Štolou 1

MATURITNÍ PRÁCE

Využití minipočítačů a open-source hardware ve výuce

Autor práce: Eliška Rojíčková

Vedoucí práce: Bc. Martin Sourada

Třída: 6S

2021/2022

Abstrakt

Tato maturitní práce je zaměřená na využití minipočítačů a open-source hardware ve výuce. Výhodou využití open-source hardware ve výuce je jeho otevřenost systému. Studenti vidí veškeré procesy, které běží na pozadí, takže můžou lépe pochopit co se právě děje. Práce je rozdělená na tři kapitoly, v první je řešeno jednotlivých možností využití minipočítačů a open-source hardware ve výuce. V druhé a třetí kapitole je pak podrobněji rozebrané využití Raspberry Pi při měření fyzikálních experimentů, konkrétně při měření koeficientů charakterizujících NTC termistor. Součástí práce je popis uživatelského prostředí pro měření experimentu kalibrace NTC termistoru.

My final thesis is focused on the use of minicomputers and open-source hardware in education. The advantage of using them is the openness of the system. Students can see all processes that run in the background, so they can easily understand what is going on. The paper is divided into three chapters. The first chapter explores the possibilities of using minicomputers and open-source hardware in education. In the second and the third chapter, the use of Raspberry Pi for measuring physical experiments is discussed in more detail. Specifically, the measurement of coefficients characterizing the NTC thermistor. Part of the thesis is a description of the user environment for measuring the NTC thermistor calibration experiment.

Čestné prohlášení

Prohlašuji, že jsem maturitní práci vypracovala samostatně, použila jsem pouze podklady uvedené v příloženém seznamu a postup při zpracování a dalším nakládání s prací je v souladu se zákonem č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů.

V Praze dne 30. března 2022

vlastnoruční podpis

Poděkování

Děkuji vedoucímu této práce Bc. Martinovi Souradovi za odborné vedení, ochotu a trpělivost, kterou mi poskytoval v průběhu psaní této práce. A všem co se podílejí na spoluvytváření open-source komunity.

Obsah

| | |
|---|-----------|
| Úvod | 6 |
| 1 Možnosti využití open-source hardware ve výuce | 7 |
| 1.1 Open-source hardware | 7 |
| 1.2 Raspberry Pi | 7 |
| 1.2.1 Raspberry Pi jako měřič fyzikálních experimentů | 8 |
| 1.3 Arduino | 8 |
| 1.4 3D tisk ve výuce | 8 |
| 2 Kalibrace NTC termistoru pomocí RPi | 9 |
| 2.1 Naměřené výsledky | 12 |
| 3 Popis praktické části | 12 |
| 3.1 Třída Thermistor | 13 |
| 3.2 Měření teploty a napětí | 13 |
| 3.3 Výpočet lineární regrese | 14 |
| Závěr | 16 |
| Seznam použité literatury | 17 |
| Seznam obrázků | 19 |
| Seznam výpisů kódu | 20 |

Úvod

Moderní technologie mají velký vliv na život všech a u mládeže snad ještě větší než u ostatních. Proto je potřeba zapojovat technologie do výuky a zatraktivnit ji tak studentům. Pro mnohé společnosti zabývající se open-source hardware je právě vzdělání mladé generace velmi důležité. Výhodou zapojení minipočítačů do výuky je jejich dostupnost, nejlevnější modely se pohybují okolo sta korun, a jejich všestrannost, není potřeba se omezovat jen na hodiny informatiky, ale využívat je můžeme například i v hodinách fyziky.

Pro tuto maturitní práci jsou určeny celkem tři cíle. Prvním je rozebrat možnosti využití open-source hardware a minipočítačů ve výuce. Prvním cílem se zabývá kapitola číslo jedna. Druhý cíl je navrhnout experiment, který využívá konkrétní minipočítač nebo open-source hardware zmíněný v první kapitole. Zároveň popsat fyzikální pozadí daného experimentu a uvést postup jeho provedení. Druhému cíli se věnuje druhá kapitola. Posledním cílem je naprogramovat grafickou aplikaci s jednoduchým uživatelským prostředím pro ovládání experimentu a zobrazení výsledků. Třetím cílem se zabývá praktická část¹ a kapitola tři, kde je popsána.

Jako hlavní cíle práce byly stanoveny:

- Provést rešerši dostupných minipočítačů a open-source hardware využitelného pro výuku.
- Navrhnout experiment pro který lze konkrétní hardware využít.
- Naprogramovat jednoduchou grafickou aplikaci pro ovládání experimentu.

¹Praktická část je dostupná na [1] ve větvi main.

1 Možnosti využití open-source hardware ve výuce

1.1 Open-source hardware

Open-source hardware je hardware, jehož dokumentace je veřejně dostupná. Protože je známa veškerá jeho specifikace, je možné ho libovolně programovat. Zároveň si ho můžeme doma sami vyrobit z běžně dostupných součástek. Častější je, že si uživatelé základní zařízení koupí a následně ho modifikují pro své potřeby. Zajímavější je komunita, která kolem open-source kultury vznikla. Lidé si sdílejí nápady na využití a vylepšení, často se i autoři hardware inspiroují nápady od uživatelů a zakomponují je do dalších verzí. Stejným způsobem funguje i vyměňování zkušeností s open-source hardware ve výuce. Zářným příkladem je komunita, která vznikla kolem projektu Prusa pro školy ohledně využití 3D tiskáren, viz kapitola 1.4. Další velká výhoda používání open-source hardware ve výuce je ta, že jsou vidět veškeré procesy na pozadí. Tudíž žáci mohou lépe pochopit co se zrovna děje a jak to celé funguje.

1.2 Raspberry Pi

Raspberry Pi je malý jednodeskový počítač. Poprvé byl představen v roce 2011 britskou nadací Raspberry Pi Foundation. Jeho primárním cílem je podpořit výuku informatiky na školách a seznámit studenty s možnostmi, které přináší minipočítače. Raspberry Pi Foundation zároveň zastřešuje spoustu programů spojující studenty a učitele se zájmem o technologie. Mezi jejich programy patří například Astro Pi Challenge, který nabízí studentům spustit jejich vlastní program na Mezinárodní vesmírné stanici. Dále například vydává časopis Hello World o počítačích a digitální tvorbě určený pro pedagogy.

Při používání Raspberry Pi se žáci neučí pouze programovat, ale zároveň si osvojí základy elektrotechniky. RPi se hodí pro ovládání jednoduchých i složitějších obvodů. Na desce RPi se vyskytuje 40 GPIO (= General-Purpose Input Output) pinů, díky kterým můžeme na RPi připojit senzory a měřiče jako například senzor vlhkosti, pohybu nebo teploměr. Ty zajišťují nekonečné možnosti tvoření vlastních projektů. Navíc díky otevřené specifikaci je možné si vymýšlet vlastní senzory. Mezi nejpopulárnější patří výroba vlastní meteostanice, sestavení a naprogramování robota či autíčka na dálkové ovládání. Jako u většiny open-source hardware se i okolo Raspberry Pi vytvořila komunita, kde si nadšenci radí a sdílejí svoje nápady. Zároveň jich hodně publikuje Raspberry Pi Foundation na svých stránkách, kde jsou detailně zpracované nejen pro potřeby učitelů ale i žáků. [2] [3] [4]

1.2.1 Raspberry Pi jako měřič fyzikálních experimentů

Jedeno z možných využití Raspberry Pi ve výuce, a to přesněji v hodinách fyziky, je ho zapojit do elektrického obvodu jako multimetr. Pokud je cílem pomocí něj měřit napětí, je potřeba na GPIO piny připojit AD/DA převodník, který převádí signál z analogového na digitální a naopak. Další možnost je opět připojit na RPi speciální senzory, které ale také povětšinou fungují, tak, v nich vzniká elektrický proud nebo napětí, jako například teploměr nebo laser pro měření vzdálenosti. Pomocí RPi lze pak měřit i celkem složité fyzikální jevy, které pomocí ručních měřičů nejsou změřitelné. Navíc se data můžou na RPi dále zpracovávat nebo odesílat do databáze. Mezi zajímavé pokusy patří například matematické kyvadlo [5], kde se pomocí laseru měří perioda kmitání, což je mnohem přesnější, než snaha o měření stopkami. Z naměřených dat pak lze vypočítat přibližnou hodnotu tíhového zrychlení. Nebo lze celkem jednoduše sestavit obvod pro měření Seebeckova jevu [6], který měří napětí, které se generuje na daném termoelektrickém článku. Následně z dat lze vypočítat seebeckovy koeficienty pro dané materiály.

1.3 Arduino

Arduino je stejně jako Raspberry Pi open-source jednodeskový počítač, ale jeho ovládání je ještě snazší. Oproti RPi se k Arduino nepřipojuje monitor, klávesnice ani myš. Místo toho se kód píše ve vývojovém prostředí na jiném počítači a následně pošle na dané Arduino, které je potřeba s prostředím spojit. Společnost nabízí program Arduino Education, který pro žáky připravil několik sad pro experimentování s Arduinem. Od sad pro úplné začátečníky, až pro sady zaměřené na vysokoškolské studenty. Jejich set Education Starter Kit se stal finalistou v soutěži 'BETT Awards 2022' v kategorii 'Hardware - AV - VR/AR - Robotics or Digital Device'. Jedná se o sadu určenou pro žáky základních škol, v které se nachází základní komponenty pro osm studentů. Zároveň je k sadě online materiál na jedenáct lekcí. Jeho cena se pohybuje okolo šesti tisíc korun. [7] [8]

1.4 3D tisk ve výuce

Další z open-source hardware, které lze využít ve výuce jsou 3D tiskárny. Ne všechny patří mezi open-source hardware, ale tiskárny firmy Prusa Research, která i vede program pro školy, mezi ně patří. Při zapojení 3D tiskáren do výuky se žáci naučí celý nový proces, od vytvoření modelu, přes jeho zpracování pro tisk až po samotnou obsluhu 3D tiskárny. Využití 3D tiskáren ve výuce se nemusí nutně jen omezovat na hodiny informatiky. Na těch

se žáci naučí s 3D tiskárnou pracovat, ale vytisknuté modely lze využít napříč osnovami. V dějepise si můžou ukázat modely vzácných artefaktů, ke kterým se jinak nedostanou, v biologii můžou pracovat s různými ukázkami buněk, virů a jiných organismů. Nebo na hodiny chemie lze vytisknout modely atomů a vazeb a následně tvořit jednotlivé molekuly. Josef Průša vybudoval svojí firmu na základě Open-source filosofie a na stejné myšlence založil i program Prusa pro školy. Funguje na principu, že každá škola, ať už základní, střední nebo vysoká, či jiná instituce zabývající se mimoškolními aktivitami může zažádat o zapůjčení 3D tiskárny na dobu tří měsíců. Na základě open-source myšlenky školy veřejně sdílí projekty, které vymyslí, s veškerými podklady pro ostatní. Pokud žáci vypracují zajímavý projekt, firma Prusa Research škole tiskárnu věnuje. Velká výhoda open-source přístupu je, že učitelé se můžou inspirovat jinými projekty a zapojit je do výuky. [9]

2 Kalibrace NTC termistoru pomocí RPi

V Praktické části se věnuji jedné z možností jak využít Raspberry Pi pro měření fyzikálních experimentů. Zvolila jsem měření charakteristik termistoru.

NTC termistor je polovodičová součástka, kde s rostoucí teplotou klesá odpor, oproti tomu v PTC termistoru odpor roste s rostoucí teplotou. Negastor, jak je NTC termistor občas nazýván, funguje na principu, že s rostoucí teplotou roste energie termistoru, díky tomu dochází k excitaci nosičů náboje z valenčního do vodivostního pásu, tím pádem roste množství volných nosičů náboje a odpor klesá. Odpor ovšem klesá nelineárně podle rovnice (1), kde A je konstanta závislá na tvaru a materiálu termistoru a B je teplotní koeficient. [10]

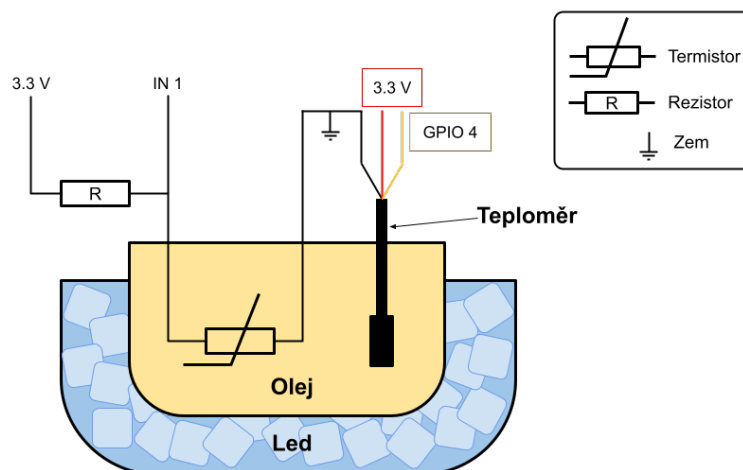
$$R = A \cdot e^{\frac{B}{T}} \quad (1)$$

Pokud rovnici zlogaritmujeme dostaneme lineární závislost viz rovnice (2). Graf lineární závislosti dostaneme, když na osu y budeme vynášet $\ln R$ a na osu x hodnotu $\frac{1}{T}$.

$$\ln R = \frac{B}{T} + \ln A \quad (2)$$

Termistor charakterizují dva koeficienty a to koeficient B (někdy označována jako β) a odpor R_0 což je odpor při teplotě 25°C . Experiment se věnuje měření koeficientů pro daný termistor a následné porovnání s hodnotami nalezenými ve specifikaci.

Obvod se skládá z termistoru, ochranného rezistoru a teploměru viz obrázek 1.



Obrázek 1: Schéma zapojení obvodu

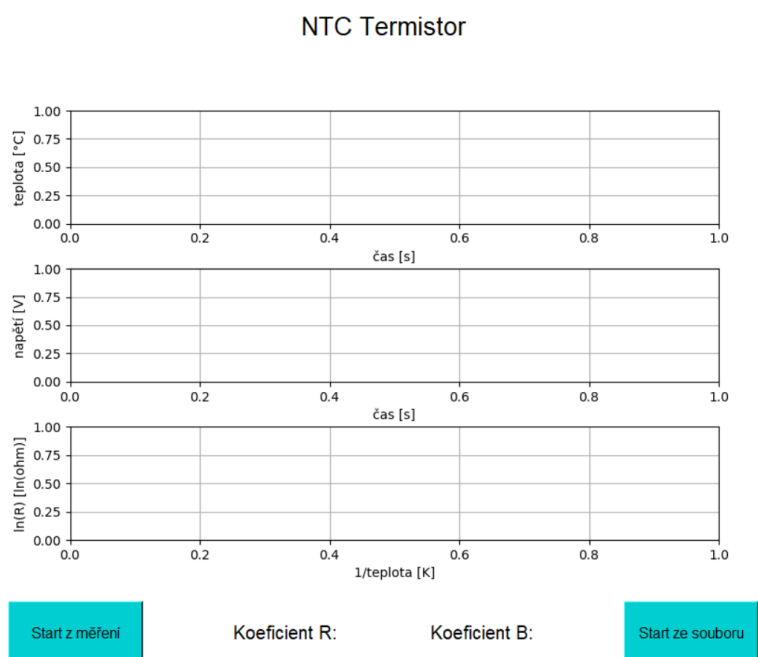
Zapojení lze rozdělit na dva nezávislé obvody. První obvod měří pomocí Raspberry Pi napětí nad termistorem. Jelikož je ochranný odpor a termistor zapojený jako dělič napětí lze vypočítat velikost odporu termistoru R_2 pomocí vztahu (3), kde U_2 je napětí, měřené nad termistorem, R_1 je hodnota ochranného odporu a U je celkové napětí (v našem zapojení konstantě 3,3 V). Hodnota ochranného rezistoru R by měla být přibližně stejně velká jako odpor termistoru.

$$R_2 = \frac{U_2 \cdot R_1}{U - U_2} \quad (3)$$

Druhý obvod je pouze zapojení teploměru, který připojíme pomocí pinu GPIO 4. V závislosti na druhu teploměru je někdy potřeba zapojit ochranný odpor i v tomhle obvodu. Změnu teploty termistoru zajišťuje ochlazování oleje ledem okolo. Olej je potřeba před

experimentem ohřát na teplotu okolo 60°C , aby byla teplotní změna větší ale zároveň aby nebyl olej moc horký a proto nebezpečný. Olej využíváme, protože kapalina ve které plave termistor a teploměr musí být nevodivá aby jsme obvod nezkratovali.

Po zapojení obvodu a nainstalování dodatečných knihoven pro správné fungování kódu je potřeba spustit kód. Zobrazí se uživatelské prostředí viz obrázek 2. Skládá se z dvou ovládacích tlačítek. Levé pro spuštění se zapojeným obvodem, kdy se budou data měřit z reálného experimentu. Pravé spustí kód s tím, že se data budou brát z csv souboru, kde jsou hodnoty naměřené z předchozího měření.

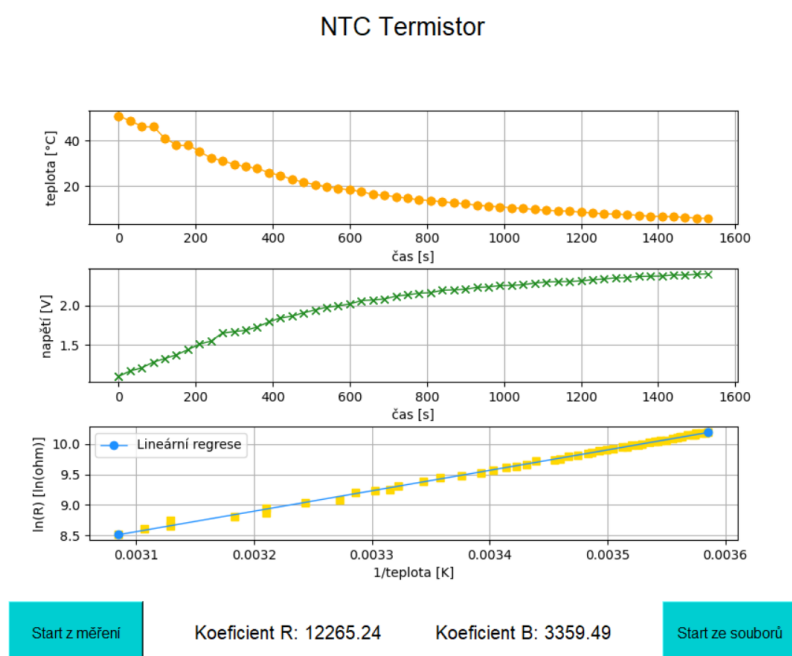


Obrázek 2: Uživatelské prostředí

Kromě tlačítek, jsou zde také osy souřadnic pro tři grafy. Na prvním se bude zobrazovat závislost teploty na čase, na druhém závislost napětí nad termistorem na čase a na třetím závislost inverzní teploty na zlogaritmovaném odporu termistoru. Třetí graf je nejdůležitější pro měření koeficientů B a R_0 , protože zobrazuje vztah (2). Jelikož se jedná o lineární závislost, lze naměřená data proložit přímkou. Koeficient B je roven směrnici přímky a pomocí interpolace lze vypočítat odpor při teplotě 25°C neboli R_0 .

2.1 Naměřené výsledky

Pro samotný experiment jsem použila termistor modelu NTCLE100E3 s prvním pruhem hnědým, druhým červeným, třetím oranžovým a čtvrtým zlatým. Podle specifikace by měly být hodnoty koeficientů $R_0 = 12\,000\,\Omega \pm 5\,\%$ a $B = 3\,740\,\text{K} \pm 2\,\%$ [11]. Na obrázku 3 lze vidět, že závislost teploty ani napětí na čase není lineární. Zároveň tam jsou vidět hodnoty koeficientů, které mi vyšly, $R_0 = 12\,265,24\,\Omega$ a $B = 3\,359,49\,\text{K}$. Koeficient R_0 mi vyšel s přesností na 2,2 %, což se vešlo do odchylky udávané ve specifikaci, zatímco koeficient B vyšel s přesností na 10 %, což je výrazně vyšší odchylka, než je specifikaci. Chyby v měření nastaly nejspíš nerovnoměrným ochlazováním oleje, popřípadě, že celkové napětí, které jsem dodávala do obvodu z RPi nemuselo být přesně 3,3 V, jak jsem s ním poté počítala.



Obrázek 3: Naměřená data

3 Popis praktické části

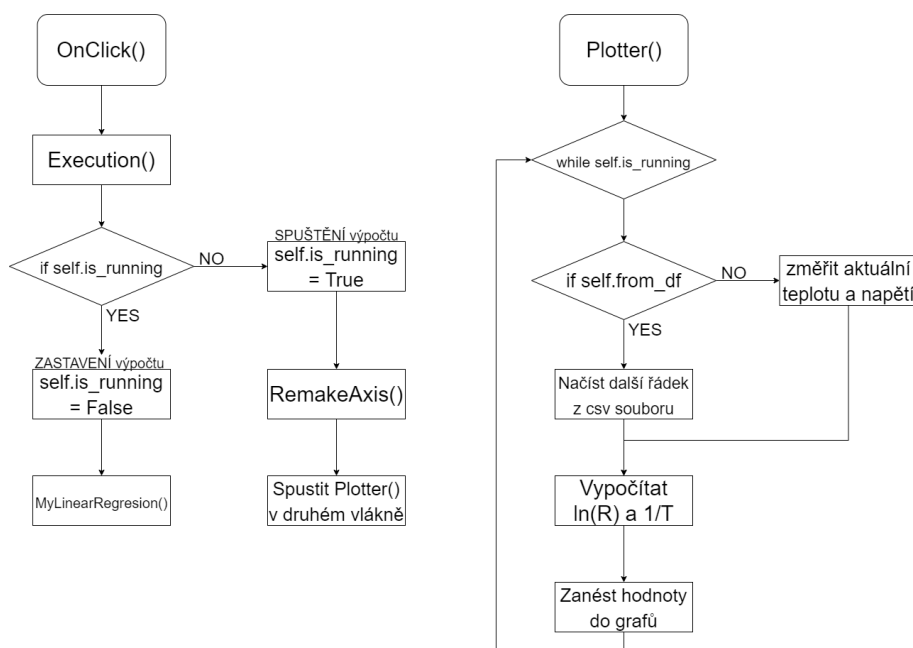
Hlavní část kódu se nachází v souboru `gui.py`, přesněji jde o třídu `Thermistor`². Dále třída využívá funkce `read_voltage` a `read_temperature` z souborů `napeti.py` a `teplota.py`. Ty slouží pro měření aktuální teploty a napětí, které následně vrací.

²Kód praktické části je dostupný na [1] ve větvi main.

3.1 Třída Thermistor

Třída **Thermistor** obsahuje sedm metod, které zajišťují nejenom vytvoření uživatelského prostředí, ale i zpracování dat a jejich kreslení do grafů. Na vývojovém diagramu viz obrázek 4 lze vidět jak na sebe jednotlivé metody navazují.

Celý program běží ve dvou vláknech. První vlákno obhospodařuje stisknutí libovolného tlačítka, které spouští nebo zastavuje výpočet ve druhém vlákně. V druhém vlákně běží cyklus, který načítá data a zakresluje je do grafů.



Obrázek 4: Vývojový diagram

3.2 Měření teploty a napětí

Teplotní senzor po nahrání příslušných ovladačů, vytvoří virtuální zařízení, které najde na adrese `'/sys/bus/w1/devices'`. Jeho název se může lišit, ale teploměry připojené na tuhle sběrnici začínají vždy řetězcem `"28"`. Ve funkci `read_temperature_raw()` daný soubor přečteme a jednotlivé řádky uložíme do proměnných `valid` a `temp`. Výsledek může vypadat například takhle, s tím, že první řádek odpovídá proměnné `valid` a druhý proměnné `temp`.

```

01 4b 46 7f ff 07 10 f6 : crc=f6 YES
01 4b 46 7f ff 07 10 f6 t=20562

```

Pomocí funkce `read_temperature()` poté ověříme, že došlo ke korektnímu načtení hodnot, což se pozná podle toho, že první řádek obsahuje `YES`, pak upravíme data, tak aby šlo s teplotou dále pracovat. Aktuální teplota se nachází v druhém řádku na konci za `t=` a je ještě potřeba ji vydělit tisícem. Aktuální teplota změřená v příkladu nahoře je tedy $20,562^{\circ}\text{C}$

Na měření napětí využívám metodu `read_adc_voltage()` třídy `ADCDACPi`, která je součástí `ABElectronics_Python_Libraries`. Je dostupná na [12].

3.3 Výpočet lineární regrese

V metodě `MyLinearRegression()` využívám `linear_model` z knihovny `sklearn`. Jelikož pro lineární regresi je potřeba dvoudimenzionální pole, musíme jako první na data použít funkci `reshape(-1, 1)`, která nezmění hodnoty dat. Jelikož prokládáme data přímkou, stačí nakreslit spojnici mezi krajními body. Funkční hodnotu pro oba body lze získat pomocí funkce `lm.predict()`. Stejná funkce je využita v interpolaci pro zjištění odporu při teplotě 25°C . Koeficient B je roven směrnici přímky, proto ho získáme pomocí vlastnosti `lm.coef_`.

```

1
2 def MyLinearRegression(self):
3     """
4     --- Metoda pro lineární regresi ---
5     - Proložení posledního grafu přímkou
6     - Vypočítání koeficientů  $R_{25}$  s~B
7         - B = směrnice přímky
8         -  $R_{25}$  = interpolovaný odpor pro teplotu 25 °C
9     """
10    y = self.dataframe_write['Zlogaritmovaný
11    ↪ odpor'].values.reshape(-1, 1)
12    x = self.dataframe_write['Převrácená teplota'].values.reshape(-1,
13    ↪ 1)
14
15    lm = linear_model.LinearRegression().fit(x, y)
16    x2 = [min(x), max(x)]
17    y2 = lm.predict(x2)
18
19    T_25 = 25 + self.kelvin
20    invert_T_25 = np.array([1/(T_25)])
21    log_R_25 = lm.predict(invert_T_25.reshape(-1, 1))
22    koef_R_25 = round(np.exp(log_R_25[0][0]), 2)
23
24    koef_B = round(lm.coef_.tolist()[0][0], 2)
25    self.text_R.config(text=f'Koeficient R: {koef_R_25}')
26    self.text_B.config(text=f'Koeficient B: {koef_B}')
27
28    self.ax3.plot(x2,y2, marker='o', lw=1, label='Lineární regrese',
29    ↪ color="dodgerblue")
30    self.ax3.legend()
31    self.graph.draw()

```

Výpis kódu 1: Metoda MyLinearRegression()

Závěr

Hlavní cíle této maturitní práce byly:

- Provést rešerši dostupných minipočítačů a open-source hardware využitelného pro výuku.
- Navrhnout experiment pro který lze konkrétní hardware využít.
- Naprogramovat jednoduchou grafickou aplikaci pro ovládání experimentu.

Veškeré cíle se podařilo naplnit v jednotlivých kapitolách, které se jim věnovaly. Experiment, kterému se práce věnovala v druhém cíli, byla kalibrace NTC termistoru, kdy se pomocí Raspberry Pi měřila teplota a napětí, což jsou veličiny potřebné pro vypočítání koeficientů B a R_0 charakterizující NTC termistor. Časová náročnost celého experimentu je méně než hodina a půl, tudíž je vhodný pro využití při laboratorních pracích ve fyzice. Pro experiment je potřeba sestavit obvod, zahřát olej přibližně na teplotu 60°C a vytvořit led, což obojí je možné ve školních laboratořích. Dále je potřeba základní vybavení pro Raspberry Pi tj. monitor, myš a klávesnice a základní elektrotechnické součástky. Experiment se nejvíce hodí pro třetí ročníky, kdy se ve fyzice probírá elektřina, respektive polovodiče. Kapitálové náklady na jednu sadu, tj. RPi, AD/DA převodník, teplotní senzor a termistor, na které můžou pracovat tři až čtyři studenti, činí přibližně 600 Kč. Provozní výdaje experimentu jsou zanedbatelné.

Naměřené výsledky provedeného experimentu jsou následovné: teplotní koeficient B vyšel s přesností na 10 %, což je oproti specifikaci celkem nepřesné, zatímco odpor R_0 vyšel s přesností na 2,2 %, což se vešlo do rozmezí udávaného ve specifikaci.

Seznam použité literatury

1. ROJÍČKOVÁ, Eliška. *Kalibrace Termistoru* [online]. GitHub, 2022 [cit. 2022-03-29]. Dostupné z: https://github.com/ERojickova/Kalibrace_Termistoru.
2. PŘISPĚVATELÉ WIKIPEDIE. Raspberry Pi Foundation. In: *Wikipedie: Otevřená encyklopedie* [online]. 2021. Datum poslední úpravy: 28. 12. 2021, 09:34 UTC [cit. 2022-03-24]. Dostupné z: https://en.wikipedia.org/w/index.php?title=Raspberry_Pi_Foundation&oldid=1062418034.
3. PŘISPĚVATELÉ WIKIPEDIE. Raspberry Pi. In: *Wikipedie: Otevřená encyklopedie* [online]. 2022. Datum poslední úpravy: 19. 03. 2022, 20:29 UTC [cit. 2022-03-24]. Dostupné z: https://cs.wikipedia.org/w/index.php?title=Raspberry_Pi&oldid=21052761.
4. FOUNDATION, Raspberry Pi. *Teachers and educators* [online]. 2022 [cit. 2022-03-24]. Dostupné z: <https://web.archive.org/web/20220324033056/https://www.raspberrypi.org/teach/>.
5. PŘISPĚVATELÉ WIKIPEDIE. Matematické kyvadlo. In: *Wikipedie: Otevřená encyklopedie* [online]. 2021. Datum poslední úpravy: 8. 08. 2021, 13:12 UTC [cit. 2022-03-30]. Dostupné z: <https://cs.wikipedia.org/w/index.php?title=Matematické%20kyvadlo&oldid=20331828>.
6. PŘISPĚVATELÉ WIKIPEDIE. Termoelektrický jev. In: *Wikipedie: Otevřená encyklopedie* [online]. 2022. Datum poslední úpravy: 28. 03. 2022, 09:06 UTC [cit. 2022-03-30]. Dostupné z: <https://cs.wikipedia.org/w/index.php?title=Termoelektrický%20jev&oldid=21082859>.
7. PŘISPĚVATELÉ WIKIPEDIE. Arduino. In: *Wikipedie: Otevřená encyklopedie* [online]. 2022. Datum poslední úpravy: 13. 02. 2022, 07:45 UTC [cit. 2022-03-24]. Dostupné z: <https://cs.wikipedia.org/w/index.php?title=Arduino&oldid=20933612>.
8. ARDUINO. *Arduino Education: Empower scientists and artists of the future* [online]. 2022 [cit. 2022-03-24]. Dostupné z: <https://web.archive.org/web/20220301031400/https://www.arduino.cc/education>.
9. PRUSA RESEARCH, a.s. *Prusa pro školy* [online]. 2021 [cit. 2022-03-23]. Dostupné z: <https://web.archive.org/web/20210516154230/https://proskoly.prusa3d.cz/vyzva/>.
10. JANÍČEK, Jan. Polovodičové snímače teploty - termistory. *Automa* [online]. 2015, č. 1, s. 28–29 [cit. 2022-03-23]. Dostupné z: <https://web.archive.org/web/20220323220222/https://automa.>

- cz/cz/casopis-clanky/polovodicove-snimace-teploty-termistory-2015_01_53323_7922/.
11. VISHAY. *NTCLE100E3 Datasheet (PDF) - Vishay Siliconix* [online]. 2016 [cit. 2022-03-22]. Dostupné z: <https://pdf1.alldatasheet.com/datasheet-pdf/view/1002979/VISHAY/NTCLE100E3.html>.
 12. DOREY, Andrew. *ABElectronicsPythonLibraries* [online]. GitHub, 2020 [cit. 2022-03-27]. Dostupné z: https://github.com/abelectronicsuk/ABElectronics_Python_Libraries.

Seznam obrázků

| | | |
|---|----------------------------------|----|
| 1 | Schéma zapojení obvodu | 10 |
| 2 | Uživatelské prostředí | 11 |
| 3 | Naměřená data | 12 |
| 4 | Vývojový diagram | 13 |

Seznam výpisů kódu

| | | |
|---|---------------------------------------|----|
| 1 | Metoda MyLinearRegression() | 15 |
|---|---------------------------------------|----|