



Středoškolská technika 2023

Setkání a prezentace prací středoškolských studentů na ČVUT

Stavba meteorologické stanice na platformě Arduino

Magdaléna Svobodová

Gymnázium Botičská
Botičská 1, 128 01 Praha 2

Poděkování

Ráda bych poděkovala mému školiteli Ing. Jakobovu Šafaříkovi za cenné rady a připomínky k mé práci. Také bych ráda poděkovala mému externímu školiteli Ing. Jiřímu Bednaříkovi za pomoc při meteorologické části mé práce a také zpřístupnění profesionální meteorologické stanice Churáňov a jejích dat.

Anotace

Cílem mé práce bylo sestavení amatérské meteorologické stanice na platformě Arduino a následné porovnání naměřených dat za časový úsek 24 hodin. Měřila jsem teplotu, tlak a relativní vlhkost. Hodnoty jsem porovnávala s profesionálními údaji z meteorologické stanice Churáňov. Měření probíhalo každou minutu pomocí čidel BMP280 a DS18B20. Výsledky byly zasílány pomocí Wi-Fi sítě na můj počítač. Hodnoty jsem následně zpracovala do tabulek a grafů a zjistila tak přesnost amatérských čidel oproti profesionálním přístrojům. Z výsledků mé práce vyplívá, že čidla neměří tak přesně, jako uvádějí výrobci a při velkých teplotních změnách za kratší časový úsek za profesionálními údaji zaostávají.

Abstract

The goal of my thesis was to build an amateur meteorological station on the Arduino platform and then compare the measured data, which were from period of 24 hours. I measured temperature, pressure, and relative humidity. Then I compared this data with professional data from the Churáňov meteorological station. Measurements were made every minute using the BMP280 and DS18B20 sensors. The results were sent via Wi-Fi to my computer. After that I processed the data into tables and graphs and found out the accuracy of amateur sensors compared to professional devices. The results of my work show that the sensors do not measure as accurately as stated by the manufacturers, and in the case of large temperature changes in a shorter period of time, they lag behind the professional data.

Obsah

Úvod.....	6
Přehled literatury	7
Meteorologie	7
Teplota.....	7
Tlak.....	8
Vlhkost vzduchu	9
Meteorologická stanice	10
Arduino	12
Metodika	15
Pomůcky	15
Postup.....	16
Hardware	16
Software	16
Měření	17
Zpracování dat	17
Výsledky a diskuse	18
Závěr.....	21
Seznam literatury	22
Přílohy.....	23

Úvod

Tématem mé práce je stavba meteorologické stanice na platformě Arduino. Mohlo by se zdát, že s meteorologickými stanicemi v každodenním životě nepřijdeme do styku, pravdou ale je, že na jejich bezchybnou práci spoléháme každý neustále. Při výběru adekvátního oblečení pro nadcházející den se všichni obracíme na předpovědi počasí s cílem vyhnout se kupříkladu nepříjemným deštivým překvapením. Právě za předpovědi počasí stojí složité matematické modely, které využívají naměřených dat na meteorologických stanicích. Člověk byl pozorováním okolního světa fascinován odpradáвна a počasí nebylo výjimkou. Staletí sledování neviditelné atmosféry kolem nás přinesly mnoho způsobů, jak změny v atmosféře pozorovat. Progresivní vývoj meteorologických zařízení zapříčinil, že v dnešním světě existuje mnoho alternativ, jak děje měřit. Dražší varianty přístrojů se nachází na profesionálních stanicích, kde umožňují měřit přesné hodnoty, které jsou následně využity pro výpočet předpovědi počasí. Na pouhé pozorování světa kolem nás jsou přístupné levnější alternativy pro běžnou populaci. Nasnadě je ale otázka, jak moc se tyto dva typy liší v oblasti přesnosti. Jsou odchylky, které výrobci amatérských měřících přístrojů uvádějí, pravdivé? Je vůbec možné se na amatérská zařízení spolehnout? Právě na tyto otázky se ve své práci pokusím odpovědět. Jako amatérskou variantu využiji prostředí Arduina, ve kterém se pokusím zprovoznit funkční meteorologickou stanici, která bude měřit různé druhy teplot a atmosférický tlak. Následně z údajů vypočítám relativní vlhkost. Mým druhým cílem je grafické porovnání mých údajů a dat z profesionální meteorologické stanice za stejný časový úsek. Chci tedy zjistit přesnost dat, kterých jsou levnější čidla kompatibilní s Arduinem schopná dosáhnout a jestli se takovýto způsob měření atmosférických dějů a jevů vyplatí.

Přehled literatury

Meteorologie

Meteorologie je vědní obor zabývající se složením, vlastnostmi, ději a jevy probíhajícími v zemské atmosféře. (1) Atmosféra Země neboli ovzduší je plynný obal planety. Její dosah od zemského povrchu je přibližně 30 km. Tato sféra je tvořena různými plyny, které souhrnně označujeme jako vzduch. Dále se v atmosféře nachází kapalné i pevné částice, z nichž převažující část utváří atmosférický aerosol. (2)

Meteorologie se obecně zajímá o procesy a jevy v troposféře. Troposféra je nejnižší vrstvou atmosféry Země, pokud ji rozdělíme vertikálně podle průběhu teploty vzduchu s výškou. Tento typ dělení ovzduší je jedním ze základních, a kromě troposféry do něj spadá také stratosféra, mezosféra a termosféra. Hranici troposféry je poměrně těžké určit jednotně pro celou Zemi. Záleží totiž na mnoha faktorech. Jedním z těchto kritérií je oblast. Na pólech se jako tloušťka troposféry udává 7 až 9 km. Na rovníku sahá až do výšky 18 km. Dalším faktorem je roční doba, v zimě totiž dosahuje troposféra obecně níže než v létě. Záleží také na tom, jaká je celková povětrnostní situace, v cyklónách (oblastích s nízkým atmosférickým tlakem) například dosahuje níže než v anticyklónách (oblastích s vysokým atmosférickým tlakem). Dělíme ji na dolní, střední a mezní troposféru. Typickým znakem troposféry je všeobecné klesání teploty s výškou. V průměru se jedná o přibližně 0,65 °C na každých 100 m výšky. V troposféře je soustředěno zhruba 75 % celkové hmotnosti atmosféry. V této sekci se také vyskytuje naprostá většina vody obsažená v ovzduší, a proto je troposféra místem, ve kterém vznikají mlhy, různé druhy oblaků, bouře a srážky. Neustále se v ní vertikálně promíchává vzduch a rychlost proudění vzduchu s přibývajícím výškou roste.

V České republice výzkum a provoz v oblasti meteorologie zajišťuje Český hydrometeorologický ústav (zkráceně ČHMÚ). ČHMÚ mimo jiné také organizuje sběr, přenos, zpracování a archivaci dat. (2)

Teplota

Teplota je jednou ze základních fyzikálních veličin. Symbolem teploty je t a její jednotkou je Kelvin K, což je jedna ze základních jednotek a spadá do Mezinárodní soustavy jednotek SI. (3) Udává nám velikost kinetické energie termického pohybu molekul. V meteorologii se nejčastěji měří teplota vzduchu nebo půdy. Údaje těchto veličin se uvádí hlavně ve stupních Celsiovy teplotní stupnice (jednotka stupeň Celsia °C). Rovnice 1 popisuje vztah mezi Celsiovou a Kelvinovou teplotní stupnicí. Celsiova stupnice rozděluje interval mezi bodem mrznutí a bodem varu čisté vody při normálním tlaku vzduchu 1 013,25 hPa na 100 dílků. (2)

$$T(^{\circ}\text{C}) = T(\text{K}) - 273,15$$

Rovnice 1 Vztah mezi
Celsiovou a Kelvinovou
teplotní stupnicí

Teplota vzduchu je jednou z meteorologických veličin, která udává tepelný stav ovzduší. Teplotu vzduchu měříme teploměrem, okolo kterého může volně proudit vzduch a je chráněn před přímým slunečním světlem. V České republice se teplota vzduchu na přízemních stanicích měří ve výšce 2 m v meteorologické budce nebo radiačním krytu. Konkrétním typem teploměrů je dvojice suchého a vlhkého teploměru.

Suchý teploměr je jeden ze dvou rtuťových teploměrů. Tento typ kapalinového teploměru je založen na faktu, že kapalina, v tomto případě rtuť, má jinou teplotní roztažnost než nádoba, ve které se tekutina nachází. Společně s vlhkým teploměrem tvoří suchý teploměr psychrometr. V meteorologickém významu je teplota suchého teploměru teplotou vzduchu.

Vlhký teploměr je druhým teploměrem psychrometru. Vlhkou teplotou označujeme teplotu, které původně nenasycený vzduch nabude po nasycení vodní parou. Rozlišujeme dva základní typy, podle toho, jestli proces proběhne adiabaticky, tedy bez výměny tepla mezi soustavou a okolím, nebo izobaricky, tedy beze změny tlaku. V meteorologii vlhkou teplotou míníme zkráceně teplotu vlhkého teploměru. Její velikost se v ideálním případě blíží teplotě izobarické. Vlhký teploměr je pokryt tkaninovým obalem (někdy označujeme jako punčošku). Tato látka je namočená v nádobce s čistou, destilovanou vodou a zajišťuje tak, že povrch teploměru je neustále pokryt filmem čisté vody. (2)

Tlak

Tlak v kapalinách a plynech je fyzikální skalární veličina, která nám udává stav v libovolném místě kapaliny a plynu v klidu. Jeho symbolem je p a jednotkou Pascal (značíme Pa). (3) Tlak je podle Svobody (3) síla F působící kolmo na rovinnou plochu kapaliny či plynu, kdy pro plochu je určena veličina obsah S . Rovnice 2 popisuje vztah pro výpočet tlaku.

$$p = \frac{F}{S} \quad \text{Rovnice 2 Vztah pro výpočet tlaku}$$

V meteorologii je definován speciální typ tlaku, kterým je atmosférický tlak. Značíme ho jako p_a , jeho jednotka je stejná jako u tlaku Pa. (3) V meteorologii se ale nejčastěji používá jednotka hPa. (1) Je způsoben působením atmosféry. Tíhovou silou jsou totiž všechny částice vzduchu přitahovány k zemskému povrchu. Proto je celá atmosféra spoutána se Zemí a koná s ní otáčivý pohyb. Tímto působením dostáváme atmosférickou tíhovou sílu značenou jako F_a . Právě touto silou je vyvolán atmosférický tlak. (3) Ve své práci Novák (1) definuje atmosférický tlak takto: „Tlak vzduchu (též atmosférický tlak) je síla působící v daném místě atmosféry kolmo na libovolně orientovanou jednotkovou plochu vyvolaná tíhou vzduchového sloupce sahajícího od hladiny, ve které se tlak zjišťuje, až k horní hranici atmosféry.“

Abychom mohli naměřené hodnoty atmosférického tlaku určit jako nízké či vysoké, je zapotřebí si zavést nějakou normální hodnotu tlaku, kterou je normální tlak vzduchu. Jeho hodnota je 1 013,25 hPa. Jedná se o stanovenou hodnotu tlaku, která byla zavedena jako průměrná hodnota tlaku vzduchu při mořské hladině na 45° zeměpisné šířky, při teplotě 15 °C a tíhovém zrychlení 9,80665 m s⁻². (2)

Atmosférická tlak s přibývajícím výškou klesá. Pokud těleso posuneme výše, bude nad ním menší sloupec vzduchu, a proto na něj bude působit menší síla. (1) Zpravidla se tlak zmenší o 8 hPa na každých 100 m výšky, ale pouze do výšky 1 km nad zemským povrchem. (4) Pro porovnání různých velikostí tlaků z jiných výšek je potřeba je sjednotit na stejnou výšku, aby byly vidět reálné tlakové útvary a nejenom ty způsobené rozdílnou polohou. Standartně se proto naměřené hodnoty na různých meteorologických stanicích redukuje na stejnou výšku, a to zpravidla na střední výšku hladiny moře. Díky tomu je pak možné vytvářet meteorologické mapy přízemního tlakového pole. (2)

Vlhkost vzduchu

Vlhkost vzduchu popisuje, jaké množství vodní páry se ve vzduchu nachází. (2) Tato pára se do vzduchu dostává vypařováním vody z velkých vodních ploch jako jsou například moře, jezera apod. Tento plyn se pak v atmosféře ochlazuje a kondenzuje zpět na vodu, nebo někdy přímo sublimací z páry na led. Takto se vytváří oblaka, poté za určitých kritérií vznikají srážky a tímto způsobem se voda vrátí zpět na zemský povrch. Výpar definujeme jako stav, kdy se více molekul vypařuje z povrchu kapaliny, než se přeměňuje z plynného na kapalné nebo pevné skupenství. (3)

Relativní vlhkost vzduchu (r) neboli poměrná vlhkost vzduchu je jednou z charakteristik vlhkosti vzduchu. Dalšími charakteristikami jsou například absolutní vlhkost, tlak vodních par či měrná vlhkost vzduchu. Relativní vlhkost vzduchu popisuje stupeň, do kterého je vzduch nasycen vodními parami. Vyjadřujeme ji v procentech. (2) Suchý vzduch má relativní vlhkost 0 %, vzduch nasycený vodní párou má hodnotu 100 %. Pro člověka se ideální relativní vlhkost vzduchu pohybuje mezi 50 % a 70 %. (3)

Jedním ze způsobů, jak změřit relativní vlhkost vzduchu je pomocí psychrometru (viz **Chyba! Nenalezen zdroj odkazů.**). Toto zařízení je tvořeno dvěma stejnými teploměry, nejčastěji rtuťovými. První teploměr nazýváme suchý teploměr (na **Chyba! Nenalezen zdroj odkazů.** vlevo). Ten měří teplotu vzduchu. Druhým teploměrem je vlhký teploměr (na **Chyba! Nenalezen zdroj odkazů.** vpravo), na který je navlečena látková punčoška namočená v destilované vodě, která na teploměru tvoří film vody. Tento film se vypařuje v případě, že relativní vlhkost vzduchu je nižší než 100 %. Voda si od teploměru odebírá teplo, které potřebuje, aby se vypařila. Toto teplo je také ukazatelem jedné z charakteristik vlhkosti vzduchu, a to sytostním doplňkem. Jedná se o množství vodní páry, která je potřeba dodat, aby se vzduchu stal při konstantní teplotě nasyceným. Teplota vlhkého teploměru je většinou nižší než teplota teploměru suchého. Výjimkou je situace, kdy je vzduch vodní parou nasycen (například pokud je hustá mlha), to se mohou teploty suchého a vlhkého teploměru rovnat. Při záporných teplotách může být vlhká teplota dokonce vyšší než suchá (led tvořící se z punčošky předává teplo teploměru, aby mohl své skupenství přeměnit z kapalného na pevné). Relativní vlhkost vzduchu se určuje pomocí údajů suchého a vlhkého zařízení.

Rovnice 3 vyjadřuje Magnusův vzorec, který lze pro výpočet relativní vlhkosti z dat psychrometru využít. Jedná se o vzorec popisující závislost tlaku nasycené vodní páry (e_s) nad rovinným vodním povrchem na teplotě vzduchu. Tlak nasycené vodní páry při teplotě 0 °C (e_{s0}) je roven 6,1 hPa a T ve vzorci označuje teplotu ve stupních Celsia. (2)

$$e_s = e_{s0} 10^{\frac{7,45T}{235+T}}$$

Rovnice 3 Magnusův vzorec (2)

Pro výpočet tlaku vodní páry (e) je zapotřebí psychrometrického vzorce, který zachycuje Rovnice 4. Z Magnusova vzorce lze dosadit hodnotu nasycené vodní páry (e_s). A označuje psychrometrický koeficient, který je závislý na rychlosti vzduchu. Veličina p je tlak vzduchu. Ve vzorci se také nachází dva druhy teplot naměřené psychrometrem. T je teplota vzduchu, kterou udává suchý teploměr a T' je teplota vzduchu udávaná vlhkým teploměrem. Psychrometrický vzorec se využívá pro tvorbu psychrometrických tabulek, ze kterých lze vyčíst různé parametry týkající se vlhkosti vzduchu na základě dat naměřených psychrometrem. (2)

$$e = e_s - Ap(T - T')$$

Rovnice 4 Psychrometrický vzorec (2)

Pro výpočet relativní vlhkosti vzduchu stačí hodnoty tlaku nasycené vodní páry a tlaku vodní páry dosadit do vzorce podle Rovnice 5. (2)

$$r = \frac{e}{e_s}$$

Rovnice 5 Vzorec pro výpočet relativní vlhkosti (2)

Rozlišujeme dva typy psychrometrů. Prvním jsou uměle ventilované aspirační, typicky Assmannův psychrometr. Teploměry se nachází v kovových trubkách, jimiž uměle proudí vzduch. Toto zařízení dovoluje měřit teplotu i pokud je vystavenou slunečnímu záření. Druhým typem je uměle neventilovaný psychrometr Augustův, který je zobrazen na **Chyba! Nenalezen zdroj odkazů..** Tento přístroj se nachází v meteorologických budkách a je tvořen dvěma teploměry, z nichž ten vlhký je obalen punčoškou, která je namočená do destilované čisté vody v nádobce, která je připevněná pod přístrojem. Z tohoto zařízení se teploty odečítají při pravidelných srovnávacích měřeních a také v případě výpadku automatizovaného systému na meteorologických stanicích jako záložní. (2)



Obr. 1: Psychrometr (není-li uvedeno jinak, obrázek autorka)

Meteorologická stanice

Meteorologická stanice je místo, kde probíhají meteorologická pozorování na základě dohodnutých postupů. Meteorologické pozorování je definováno jako sbírání údajů o jednom nebo více meteorologických prvcích nebo jevech. Pozorování probíhá v pozorovacím termínu (určitém časovém úseku). Meteorologické pozorování probíhá zpravidla na stálých meteorologických stanicích. Výchozím předpokladem je odpovídající vybavení a to technické, personální a komunikační. (2)

Meteorologické stanice můžeme rozdělit na základě různých hledisek podle zaměření, charakteru dat, umístění apod. Základním rozdělením by ale mohlo být podle úrovně na profesionální stanice a amatérské stanice. (2)

Profesionální stanice jsou stanice, na kterých měří většinou profesionální pozorovatelé s požadovanou klasifikací. (2) Pozorovatelé zde obsluhují stanice v nepřetržitém či kombinovaném provozu (část dne obsluhuje pozorovatel, část dne probíhá měření automaticky). Do sítě stanic ČHMÚ standartně řadíme také šest stanic Armády ČR a dvě stanice Ústavu fyziky atmosféry AV ČR (zkráceně ÚFA AV ČR). Stanice jsou automatizovány (tzn. vybaveny automatickými měřicími přístroji). Za profesionální stanice jsou považovány ty, které tvoří zprávu SYNOP (stanice pak označujeme jako synoptické). (5) (6) Jedná se o meteorologický kód, který obsahuje všechny potřebné naměřené údaje k tomu, aby bylo

možné vykreslit synoptickou mapu. Každou hodinu tedy všechny stanice vyšlou jednu synoptickou zprávu do předpovědního centra ČHMÚ v Praze. (2)

Amatérskými stanicemi míníme stanice určené k soukromým účelům, jejichž majitelé jsou neprofesionálové. (7)

Meteorologická stanice Churáňov (její fotografie je na Obr. 2) je jednou z profesionálních synoptických stanic ČHMÚ. Byla založena v roce 1952 a od té doby nepřetržitě sleduje a měří meteorologické jevy v oblasti Churáňova na Šumavě. (8)



Obr. 2: Meteorologická stanice Churáňov

Jedním z mnoha vybavení stanice je meteorologická budka (viz Obr. 3). Jedná se o bílou plastovou skříňku ve výšce 2 metrů nad zemí. Díky své struktuře umožňuje proudění vzduchu okolo přístrojů, a přitom zabraňuje jiným vnějším činitelům (sluneční záření, déšť apod.) aby zařízení poničila. Stěny jsou z dvojitých žaluzií, dno je vytvořeno z drátěného síta a stavbu ochraňuje dvojitá střeška. Její dvířka jsou směřována na sever. (2) V budce je standardně umístěno několik přístrojů, hlavními jsou vlasový vlhkoměr, lihové teploměry a dva rtuťové teploměry sloužící jako psychrometry. V dnešní době se zařízení využívají pouze na porovnávací měření a data se odečítají z automatizovaného vybavení stanice. (8)



Obr. 3: Meteorologická budka

Arduino

Arduino je vývojová platforma. Míjíme tím základnu, která existuje v hardwarové (pojem hardware zahrnuje technické vybavení počítače) i softwarové (software jsou programy a aplikace v počítači) podobě. Pracují na ní programy a všechny komponenty počítače. Konkrétně k vývojové neboli prototypové platformě řadíme také programovací jazyk (nástroj pro tvorbu systémů a aplikací) a textové editory. (9)

Arduino je jedno z nejrozšířenějších vývojových platforem na světě. (10) Označujeme ho za open source platformu (11), což znamená, že jeho zdrojový kód a schémata hardwaru jsou volně přístupná veřejnosti. Zdrojový kód je umístěný v jednom nebo více textových souborech ve chvíli, kdy zaznamenáváme text počítačového programu do zvoleného programovacího jazyka, kdy buďto dochází k jeho interpretaci anebo se ze něj stane nástroj, pomocí kterého můžeme vytvořit samostatně spustitelný soubor (9) Arduino je založeno na mikroprocesorech ATmega od společnosti Atmel. K dispozici je vedle samostatných programovatelných desek také software programovací prostředí. (11)

Arduino pochází z Itálie, kde se v roce 2005 pracovníci z Interaction Design Institute rozhodli, že vytvoří jednoduchou a levnou verzi vývojového setu, která měla konkurovat drahé desce BASIC Stamp. Jejich set měl být určen převážně pro studenty, kteří si desku BASIC Stamp z finančních důvodů nemohli pořizovat. Jelikož se Arduino mezi studenty stalo populárním, tvůrci se ho rozhodli zpřístupnit i celému světu. Nejenom že levně prodávali vlastní desky, ale také zdarma sdíleli návody a schémata spojená s jejich výrobky. Od té doby prošlo Arduino nespočtem změn a neustále probíhá jeho vývoj a zdokonalování. Kvůli jeho vlastnosti opensource také vznikají kopie a klony. Jedná se o neoficiální verze hardwaru. (10)

Oproti ostatním platformám poskytuje Arduino mnoho výhod, což z něj ve výsledku dělá jednu z nejpobulárnějších variant. Je cenově dostupné, má velké množství periférií (počítačová zařízení připojená k počítači, která nejsou jeho součástí, rozdělujeme na vstupní a výstupní na základě toho, jestli přenášíme data do nebo z počítače k uživateli (9)) a má přehledné programovací prostředí. Důležitým pozitivním faktorem je také velmi aktivní komunita uživatelů, kteří zajišťují zpřístupňování velkého množství zdrojových kódů a knihoven. (11) Knihovnou míjíme konkrétní funkce a procedury, ze kterých je vytvořen jeden soubor. Smyslem knihovny je usnadnění programátorovi tvorbu zdrojového kódu, protože může využít již vytvořený kód v jiném programu. (9)

Hardware

Hardwarem míjíme všechno technické vybavení počítače. Zjednodušeně řečeno jsou to veškeré fyzické součásti, na které si můžeme sáhnout. Může to být například pevný disk či základní deska. Periferie je specifický typ hardwaru. (9)

Mezi Hardware patří základní deska, která je jednou z nejdůležitějších dílů systémů Arduina. Je zprostředkovatelem napájení, přenáší data a všechny komponenty propojuje. Nezbytnou součástí Arduina je mikroprocesor. (10) Mikroprocesor je elektronický obvod, který čte a provádí operace. Přijímá a zpracovává informaci a na základě programu na výstupu nastaví požadované hodnoty. Základní desky mají mnoho typů a liší se na základě jejich různorodých vlastností. Některé mají například schopnost komunikovat přes Wi-Fi, což je bezdrátová technologie určená pro domácí síť nebo mobilní zařízení. (9)

Pro měření různých jevů využíváme senzory, což jsou obecně čidla, která dokážou převést naměřenou veličinu na elektrický signál. Čidla komunikují po takzvaných komunikačních rozhraních neboli sběrnicích. Těch existuje mnoho druhů, obecně se ale jedná o způsoby komunikace mezi vstupním a výstupním zařízením, v případě čidel je čidlo vstupním a displej výstupním zařízením. (9)

Pro zobrazení dat přímo na stanici je vhodné připojit LCD displej. Jedná se o výstupní zařízení schopné zobrazovat informace (například naměřené údaje).

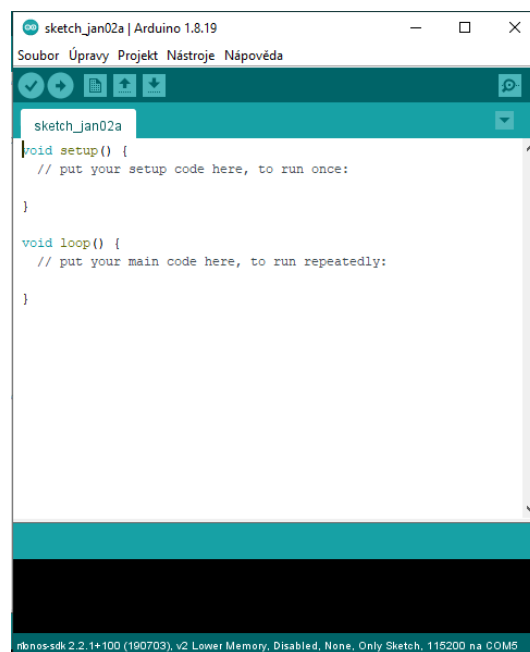
Na samotné propojení všech částí se používají propojovací kabely.

Na napájení a propojení s počítačem se standardně využívá kompatibilitnost základní desky s USB kabelem přes USB port. Port je rozhraním pro připojení periférií, připojujeme standardně pomocí konektorů. USB míníme rozhraní, které dovoluje připojit další hardware zařízení k desce. Jeho hlavním smyslem při vývoji bylo standardizovat porty a nahradit tím dříve masivně používané sériové nebo paralelní porty. (9)

Prvkem, který také řadíme do hardwaru a je nezbytným, je kontaktní pole, které jednotlivé součástky propojuje. Existují i pole, která jsou tzv nepájivá, což znamená, že je můžeme využívat bez letování. Jedná se o způsob spojování součástek pájecím materiálem. To umožňuje rychlejší a jednodušší práci při konstruování prototypů obvodů. Propojení součástek je zařízení pomocí kontaktních hřebínků, které jsou zasunuté zespodu. (9)

Software

Arduino se programuje v Arduino IDE (integrované vývojové prostředí), což je označení pro programové vybavení určené pro programátory nebo vývojáře. Prostředí je možné vidět na Obr. 4. Je napsáno v jazyce Java, které provádí kompilaci neboli překlad, z programovacího jazyka do binárního kódu procesoru. Tento kód je poté přes USB rozhraní a převodník z USB do sériové linky schopen nahrát kód přímo do mikroprocesoru. Programovacím jazykem může být C nebo C++. Hlavní výhodou je ale specifická podpora jazyka Wiring. Jedná se o knihovnu, která umožňuje uživateli používat další knihovny přímo určené pro součástky, se kterými programátor pracuje. Tyto knihovny jsou volně přístupné a jejich hlavním cílem je usnadnění psaní zdrojového kódu, protože aniž bychom věděli specifické funkce například čidla, máme možnost s tímto čidlem pracovat a pouze zadávat potřebné odkazy na funkce v knihovnách. (11) (9) (10)



Obr. 4: Arduino IDE

Jak je možné vidět na Obr. 4, obslužný program pro Arduino je rozdělen do dvou základních částí a to funkce setup a funkce loop. První funkcí, neboli blokem, je setup. Do složených závorek se v této části píše kód, ve kterém se inicializují použité knihovny a nastavují se jejich parametry. Setup je spuštěn pouze jednou a to na začátku (při připojení k napájení, zmáčknutí tlačítka restart na základní desce nebo při nahrání kódu do Arduina. Druhou částí je již zmíněný

loop. Do jeho složených závorek se píše kód, který se bude neustále opakovat dokud zařízení neodpojíme od zdroje napájení. (10) (11)

Na Obr. 4 si můžeme také všimnout textu za dvojitými lomítky. Jedná se o komentáře, jejichž obsah program ignoruje. Používají se pro psaní poznámek a slouží k větší přehlednosti kódu. (10)

Metodika

Cílem práce bylo sestavit meteorologickou stanici na platformě Arduino. Tato stanice měla měřit teplotu a tlak a následně naměřená data zaznamenat. Ze suchého a vlhkého teploměru jsem poté vyrobila psychrometr a z dat jsem měla dopočítat relativní vlhkost. Naměřená data jsem měla srovnat s daty za stejný časový úsek z profesionální meteorologické stanice Churáňov a zjistit tak přesnost mých čidel. Odchytky od profesionálních hodnot jsem poté měla srovnat s odchylkami, které uvádí výrobci čidel. V následující kapitole popíšu, jaké pomůcky jsem ke stavbě stanice potřebovala, postup, jakým jsem stanici sestavila, samotné měření a zpracování dat do tabulek.

Pomůcky

Nejdůležitější pomůckou potřebnou pro mou práci byla stavebnice Arduino.

K realizaci byla potřebná základní deska. Těch existuje mnoho typů, pro tuto práci jsem použila typ ESP8266 12-E NodeMCU Kit. Mimo jiné jsem ho zvolila hlavně kvůli jeho možnosti komunikovat přes připojení Wi-Fi sítě s mým počítačem. Znamenalo to tedy, že jsem byla schopna automaticky ukládat aktuální naměřené hodnoty v minutovém intervalu.

Pro propojení jednotlivých částí jsem využila nepájivého kontaktního pole určeného pro Arduino.

Tabulka č.1 :Specifika čidel BMP280 a DS18B20

Typ čidla	BMP280	DS18B20
Popis	Senzor pro teplotu a barometrický tlak	Digitální teplotní čidlo
Rozpětí možného měření	od 300 hPa do 1 100 hPa od 0 °C do 65 °C	od -10 °C do +85 °C
Přesnost uvedená výrobcem	100 Pa, 1 °C	0,5 °C
Komunikační rozhraní	I2C	OneWire
Napětí	1,8 V až 3,6 V	3 V až 5,5 V
Výrobce	Bosh Sensortec	Analog Devices (dříve Maxim, předtím Dallas)
Ostatní	-	vodotěsné pouzdro TO-92 nutnost připojení odporu o velikosti 4700 Ω

Na samotné měření bylo potřebné čidlo, které pak předávalo data procesoru na základní desce, který je následně odesílal do mého počítače. Rozhodla jsem se měřit teplotu a tlak s tím, že teplota byla měřena na dvou typech čidel. Prvním typem bylo čidlo BMP280, které měřilo teplotu a tlak. Jelikož jsem ale také chtěla měřit suchou a vlhkou teplotu, bylo zapotřebí, aby druhý typ čidla byl voděodolný. Tento požadavek splňuje čidlo D18B20 měřící pouze teplotu. Abych mohla odděleně zaznamenávat suchou a vlhkou teplotu, potřebovala jsem tedy 2 čidla

tohoto stejného typu. Na jedno z nich jsem ale navlékla speciální savou punčošku zapůjčenou na meteorologické stanici Churáňov. Při měření byl volný konec punčošky konstantě ponořen do destilované vody. Výrobce tohoto typu čidla vyžadoval zapojení rezistoru o velikosti 4700 Ω . Specifika obou čidel zachycuje

Tabulka č.1.

Stanice byla schopna ukazovat aktuální údaje jak na mém počítači, tak přímo na stavebnici. Toho jsem docílila připojením I2C LCD displeje.

Potřebnou součástí jsou také konektory. Na spojení jsem využila propojovací kabely typu Dupont.

Pro propojení s počítačem jsem využila vlastnosti desky, která má v sobě zabudované připojení k Wi-Fi. Program je napsán v jazyce C++ (pomocí Wiring) ve vývojovém prostředí Arduino 1.8.19.

Postup

Postup je značně zjednodušen díky faktu, že k jednotlivým čidlům existují detailní návody a knihovny na jejich snadné použití včetně zdrojového kódu.

Hardware

Jako zdroj informací o zapojení jednotlivých součástek jsem použila doporučené zapojení od výrobců čidel a měřících modulů. Schéma Hardwaru (Příloha 1) je nakresleno v programu Fritzing. Stanici jsem posléze podle schématu postavila.

Pro připojení jsem využila dva typy sběrnic. Prvním typem byla sběrnice I2C, pomocí které jsem připojila čidlo BMP280 a LCD displej. Sběrnice OneWire mi propojila dvě čidla stejného typu, a to 18B20. Konkrétní zapojení pinů na desce je popsáno v Tabulka č.2. Využila jsem toho, že k jedné sběrnici může být připojeno více periférií. K napájení jsem použila USB napájení 5 V a zabudovaný stabilizátor 3,3 V na základní desce. Moduly byly napájeny při napětí 3,3 V.

Jako zdroj napájení při měření jsem využila powerbanku o kapacitě 2000 mA. To mi umožnilo měřit minimálně 24 hodin v kuse.

Tabulka č.2 : Zapojení pinů na desce

Pin	Název	Popis
D1	SCL (I2C)	Připojení pomocí sběrnice I2C <ul style="list-style-type: none">• čidla BMP280• displej LCD
D2	SDA (I2C)	Připojení pomocí sběrnice I2C <ul style="list-style-type: none">• čidla BMP280• displej LCD
D4	Data (OneWire)	Připojení pomocí sběrnice OneWire <ul style="list-style-type: none">• 2 čidla 18B20

Software

Jako vývojové prostředí jsem využila k tomu určené Arduino 1.8.19. To sloužilo k napsání, zkompilování a nahrání programu do procesoru. K samotnému sepsání programu jsem využila dostupných knihoven a jejich příkladů, které dodávají výrobci.

Program fungoval následovně: Po připojení napájení se spustil proces Setup, kde se nastavily vstupy a výstupy připojených pinů dle Tabulka č.2. Dále se nastavily následující hodnoty: rychlost komunikace sériové linky (v mém případě rychlost $112\,500 \text{ bit} \cdot \text{s}^{-1}$), parametry pro LCD displej a parametry pro jednotlivá čidla. Poté začala část Loop. V ní se nejdříve naměřil tlak a teplota pomocí čidla BMP280 a naměřené hodnoty se zobrazily na LCD displeji a poslaly se po sériové lince. Dále program naměřil pomocí identických čidel 18B20 suchou a vlhkou teplotu, znovu je ukázal na displeji a odeslal. Nakonec naměřené hodnoty uložil na webu. Při měření na meteorologické stanici se data měřila každou minutu. Blokové schéma celého Softwaru je nakresleno v programu Visio (viz

Příloha 2).

Měření

Měření probíhalo na profesionální meteorologické stanici Churáňov. Mojí stanici jsem vložila do meteorologické budky. Mimo jiné se v této budce měří suchá a vlhká teplota, takže je možné říct, že podmínky pro mou a profesionální stanici byly totožné. Sběr dat trval přibližně 24 hodin. Po ukončení měření jsem získala také profesionální data za stejný časový úsek pro porovnání údajů. Pro srovnávání relativní vlhkosti a teploty vzduchu jsem využila dat z profesionálního přístroje HMP45A&HMP45D, který využívá sensoru HUMICAP 180. Obě dvě zařízení jsou od výrobce Vaisala.

Zpracování dat

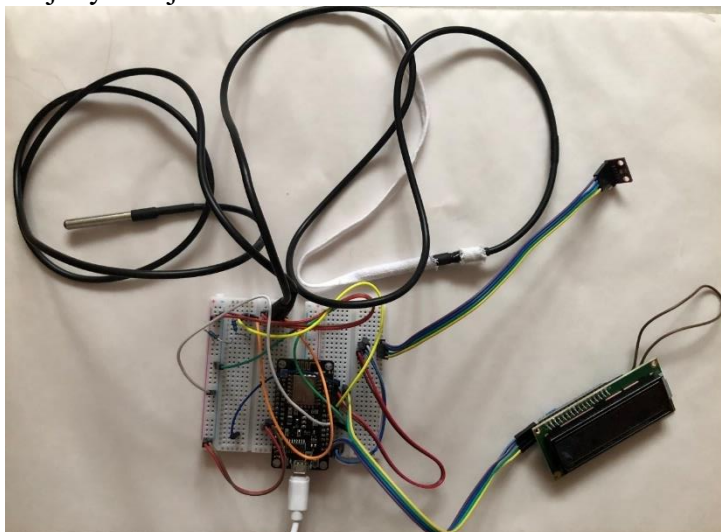
Pro zpracování naměřených dat do tabulek a grafů jsem využila tabulkový editor Microsoft Excel.

Jelikož jsem měřila tlak ve výšce o 8 metrů menší, než kde byl měřený tlak profesionální stanice Churáňov, byla potřeba naměřená data přepočítat na jinou výškovou hladinu abych mohla data srovnávat. Protože se tlak zmenší přibližně o 8 hPa na 100 m přidané výšky, přičetla jsem k hodnotám udávaným profesionální meteorologickou stanicí 0,65 hPa.

K dopočítání mojí relativní vlhkosti jsem využila 3 vzorce popsané v Rovnice 3, Rovnice 4 a Rovnice 5.

Výsledky a diskuse

Prvním cílem mé práce bylo sestavit funkční meteorologickou stanicí na platformě Arduino. Výsledná stanice (viz Obr. 5) dokázala měřit tlak, suchou a vlhkou teplotu. Na displeji ukazovala aktuální naměřené hodnoty a data byla zaslána na pomoci Wi-Fi sítě do mého počítače. Použitý zdrojový kód je v Příloha 3.



Obr. 5: Sestavená meteorologická stanice

Mým druhým cílem bylo měřit data na profesionální meteorologické stanici Churáňov po delší časový úsek. Má data naměřená za přibližně 24 hodin jsem poté zpracovala společně s daty z profesionálních zařízení do tabulky. Ukázkou, jak tabulka vypadá, zachycuje Tabulka č.3.

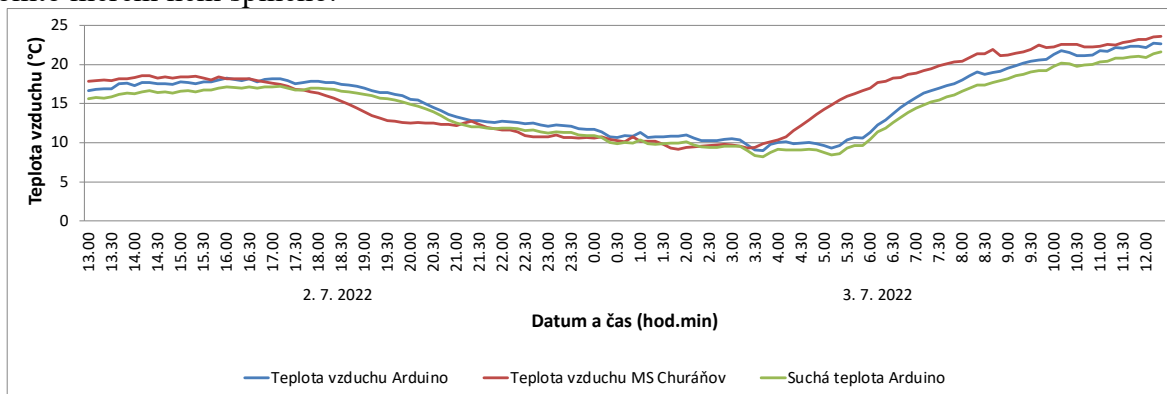
Tabulka č.3 : Ukázka tabulky naměřených a vypočítaných výsledků

Datum měření	Čas měření	Teplota Arduino (°C)	Tlak A (hPa)	Suchá teplota Arduino (°C)	Vlhká teplota Arduino (°C)	Teplota MS (°C)	Tlak MS (přepočtený) (hPa)	Relativní vlhkost MS (%)	Relativní vlhkost Arduino (%)
2. 7. 2022	13.05	16,6	899,08	15,63	10,63	17,84	898,50	51,19	51,85
2. 7. 2022	13.06	16,6	899,12	15,63	10,69	17,84	898,50	51,19	52,38
2. 7. 2022	13.07	17,0	899,05	15,75	11,00	17,84	898,50	51,19	54,25
2. 7. 2022	13.08	17,0	899,05	15,88	11,06	17,84	898,50	51,19	53,81
2. 7. 2022	13.09	16,8	899,03	15,75	11,06	17,84	898,50	51,19	54,78
2. 7. 2022	13.10	16,9	899,01	15,88	11,13	17,95	898,48	49,67	54,43
2. 7. 2022	13.11	17,2	899,00	16,00	11,25	17,95	898,48	49,67	54,60
2. 7. 2022	13.12	17,2	899,04	15,88	11,38	17,95	898,48	49,67	56,65
2. 7. 2022	13.13	16,8	898,95	15,75	11,06	17,95	898,48	49,67	54,78
2. 7. 2022	13.14	16,8	898,95	15,75	10,94	17,95	898,48	49,67	53,71
2. 7. 2022	13.15	16,5	898,95	15,63	10,88	17,95	898,48	49,67	54,08

Legenda: Arduino = údaj z stanice postavené na platformě Arduino, MS = údaj naměřený na profesionální stanici Churáňov

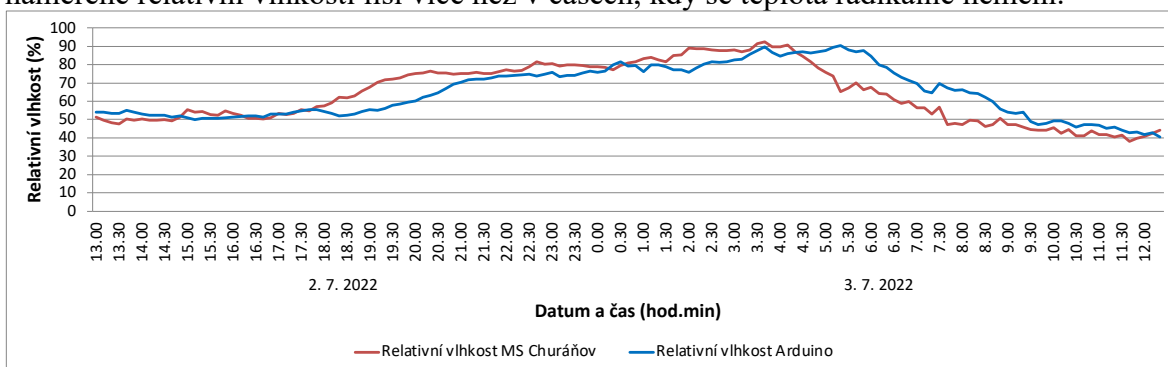
Pro větší názornost a viditelnost rozdílu dat jsem vytvořila graf z naměřených hodnot.

Na Graf 1 jsou znázorněny celkem 3 druhy dat. Prvním, značeným modře, je teplota vzduchu naměřená čidlem BMP280 a zeleně je vykreslena teplota naměřená čidlem DS18B20, které měřilo suchou teplotu. Červeně jsou označena data teploty vzduchu z profesionální meteorologické stanice. Jak na grafu můžeme vidět, odchylky amatérských čidel od profesionálního jsou poměrně velké. Ve chvílích, kdy se teplota nemění nijak rychle, amatérská data kopírují ta profesionální poměrně dobře, takže zde by se problém dal vyřešit kalibrací čidel. Od 18. do 21. hodiny, kdy se ale teplota začne rychle měnit, můžeme pozorovat velké odchylky. Stejný jev je vidět od 4. do 9. hodiny. Lze z toho usoudit, že čidla špatně reagují na rychlou změnu teplot a za reálnou teplotou zaostávají. Výrobce čidla BMP280 uvádí, že odchylka by neměla překročit 1 °C, což ale, jak je viditelné z grafu, v mnoha případech není pravdou. U čidla DS18B20 uvádí výrobce maximální odchylku dokonce 0,5 °C, což opět při tomto měření není splněno.



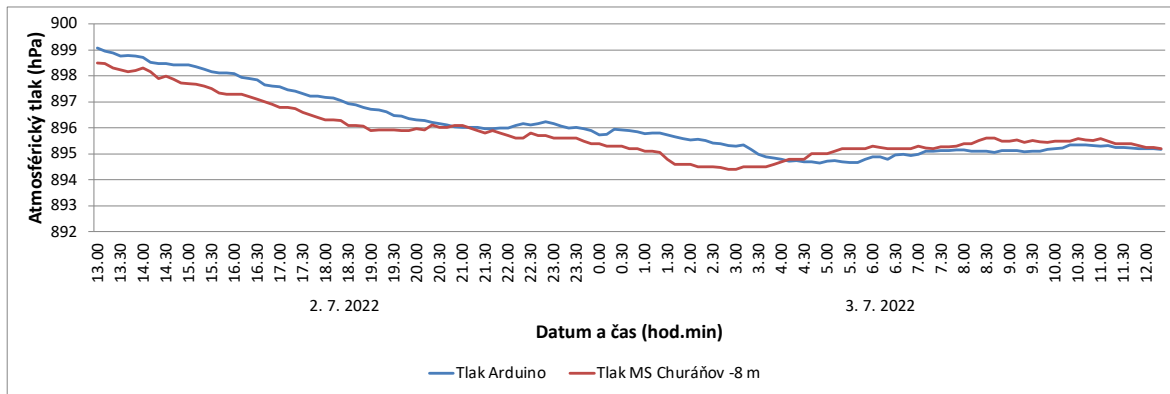
Graf 1: Grafické srovnání naměřených dat teploty vzduchu

Graf 2 znázorňuje data, která vychází z psychrometru a byla z nich dopočítána relativní vlhkost. Jelikož se ve vzorci pro výpočet relativní vlhkosti uplatňuje suchá teplota z čidla DS18B20, o kterém víme, že se od profesionálního čidla značně liší, je jasné, že i v případě relativní vlhkosti bude značná odchylka pozorovatelná. V grafu můžeme ale také pozorovat, jak je relativní vlhkost na teplotě závislá a ve stejných úsecích, jako v Graf 1, se od profesionálně naměřené relativní vlhkosti liší více než v časech, kdy se teplota radikálně nemění.



Graf 2: Grafické srovnání naměřených a vypočítaných dat relativní vlhkosti vzduchu

Graf 3 graficky znázorňuje naměřený atmosférický tlak. Z grafu můžeme vyčíst, že hodnoty naměřené pomocí čidla BMP280, které jsou zakresleny v grafu modře, se nijak značně neliší od profesionálních dat zbarvených červeně. Sice se od uvedené odchylky výrobcem 1 hPa v některých momentech amatérské čidlo od profesionálního vzdaluje, ale tento rozdíl by bylo možné zmenšit pomocí kalibrace čidla.



Graf 3: Grafické srovnání naměřených hodnot atmosférického tlaku

Závěr

V této práci jsem nejdříve sestavovala amatérskou meteorologickou stanici na platformě Arduino, která dokázala měřit teplotu, tlak a následně se vzorcem dala dopočítat relativní vlhkost. Na LCD displeji se zobrazovaly aktuální naměřené hodnoty. Celé realizaci značně dopomohl fakt, že je veřejnosti zpřístupněno mnoho návodů a knihoven, kterých může bezplatně využít.

Ve druhé části jsem měřila data na profesionální meteorologické stanici Churáňov, ze které jsem si poté vzala profesionální data za stejný časový úsek 24 hodin a pomocí tabulek a grafů jsem data porovnávala. Při měření teploty a relativní vlhkosti jsem zjistila, že čidla neměří s takovou přesností, jako udávají výrobci zařízení a odchylky jsou v některých místech poměrně velké. Bylo ale možné pozorovat, že při malých teplotních změnách se data z amatérských čidel sice vzdalují od profesionálních dat, ale průběh poměrně dobře kopírují. V budoucnu by se odchylka dala omezit kalibrací amatérských čidel. Při velkých teplotních změnách, které nastávají například ráno či večer, ale amatérské údaje zaostávají za profesionálními a ukazují velmi nepřesné hodnoty. Zde by již kalibrace nepomohla. Myslím si, že by se tato čidla hodila například na měření teplot v domech, kde je teplota poměrně konstantní.

Senzor tlaku měřil poměrně přesné hodnoty a naměřená data se tolik nelišila od profesionálních. Jelikož je ale tento senzor na stejném zařízení jako senzor teploty, a to čidlu BMP280, pro využití tohoto čidla by bylo zapotřebí nejdříve vyřešit problémy, které se týkají senzoru teploty. Výsledky přesnosti čidel potvrdili mou hypotézu, že jsou profesionální údaje mnohem přesnější než amatérské.

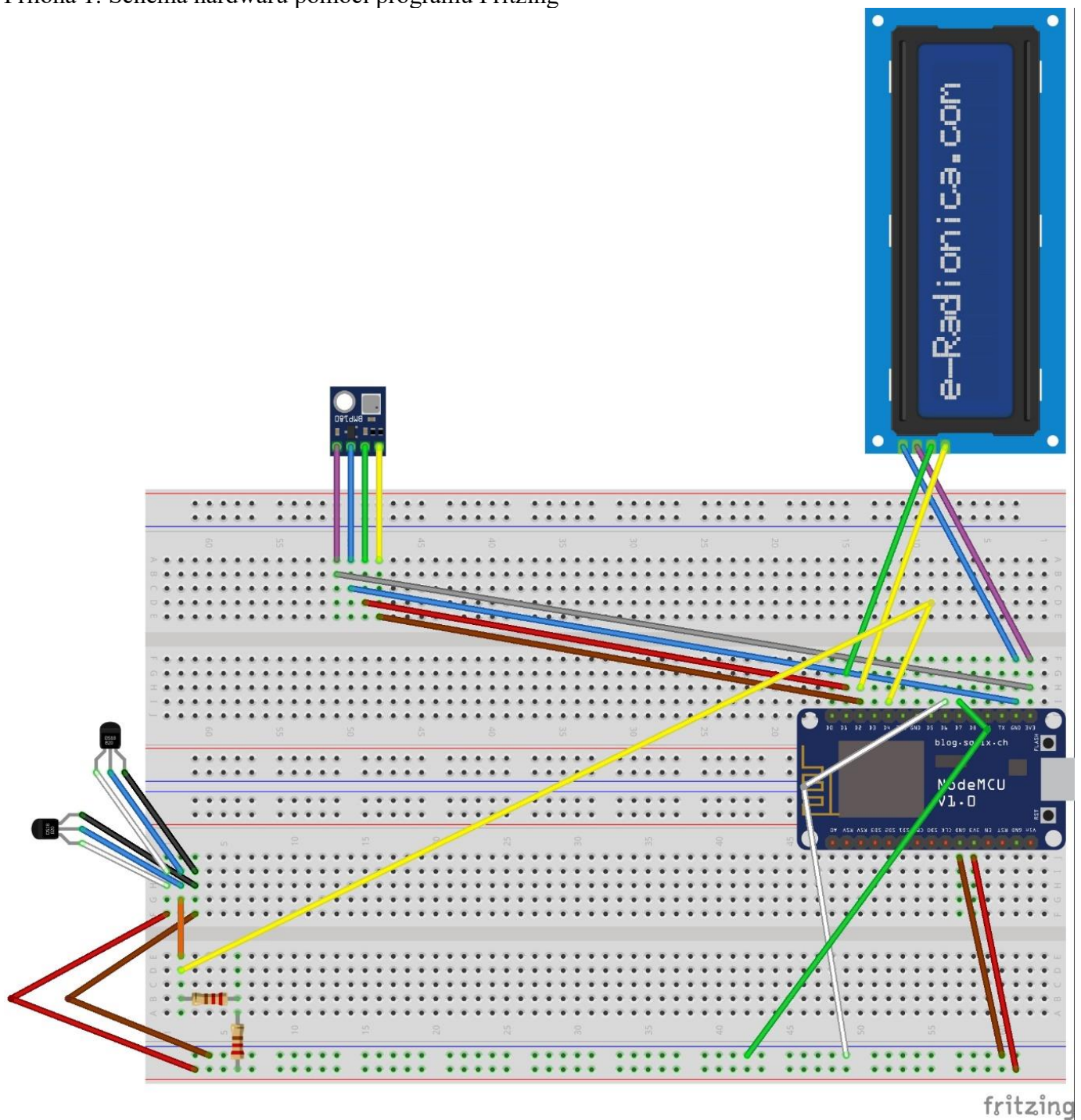
Osobně mi tato práce přinesla mnoho nových informací a zkušeností, ke kterým bych se bez ní nedostala. Dozvěděla jsem se mnohé o meteorologii a různorodých způsobech, jak se dají jevy v atmosféře měřit a pozorovat. Naučila jsem se základy programování v prostředí Arduina, se kterým jsem do té doby neměla žádné zkušenosti. Zlepšila jsem se také v používání tabulkového editoru Microsoft Excel. Bylo velmi zajímavé pozorovat počasí kolem mě a jsem ráda za tuto příležitost se těmto tajemným jevům alespoň lehce přiblížit.

Seznam literatury

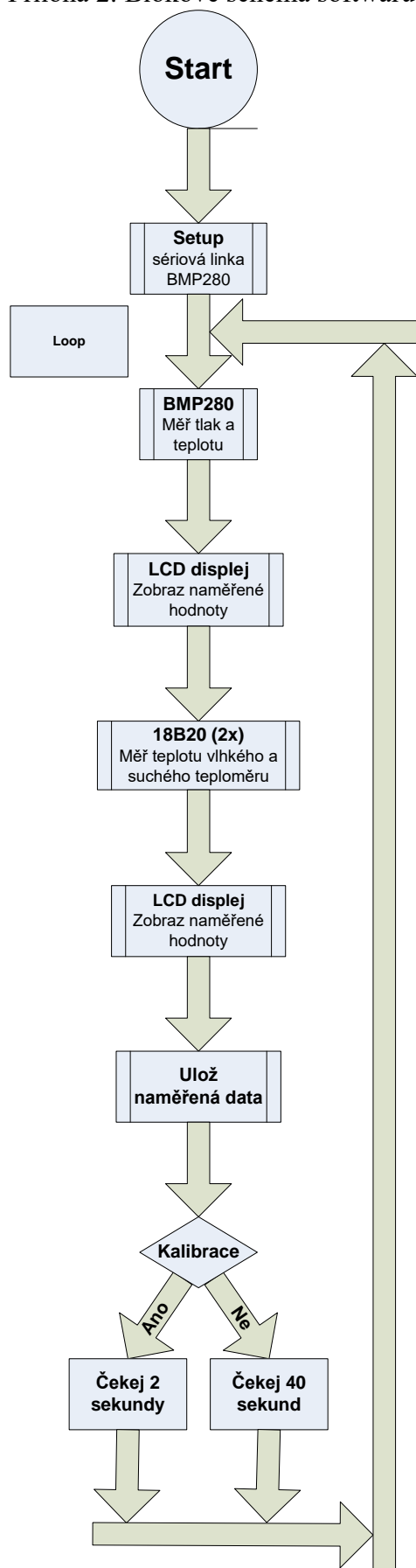
1. NOVÁK, Martin. *Meteorologie a ochrana prostředí: úvod do meteorologie a klimatologie*. Ústí nad Labem: Univerzita Jana Evangelisty Purkyně, Fakulta životního prostředí, 2004. ISBN 80-7044-597-1.
2. Česká meteorologická společnost [online]: Elektronický meteorologický slovník (eMS) [cit 18.12.2022]. Dostupné z: <http://slovník.cmes.cz>
3. SVOBODA, Emanuel. *Přehled středoškolské fyziky*. 6., upravené a doplněné vydání. Praha: Prometheus, 2019. ISBN 978-80-7196-475-9.
4. LIPINA, Pavel. Meteorologická staniční síť ČHMÚ v roce 2020. *Meteorologické zprávy* [online]. Praha: Český hydrometeorologický ústav, 2021, **74**(5), 149–164 [cit. 2022-12-30]. Dostupné z: https://www.chmi.cz/files/portal/docs/reditel/SIS/casmz/assets/2017/chmu_mz_5-17.pdf
5. LIPINA, Pavel. Členění meteorologické staniční sítě ČHMÚ a horské meteorologické stanice v Česku. *Meteorologické zprávy* [online]. Praha: Český hydrometeorologický ústav, 2017, 2017-10-31, **70**(5), 134–142 [cit. 2022-12-30]. Dostupné z: https://www.chmi.cz/files/portal/docs/reditel/SIS/casmz/assets/2017/chmu_mz_5-17.pdf
6. Meteorologická stanice. In: *Wikipedie: otevřená encyklopedie* [online]. San Francisco (CA): Wikimedia Foundation, 2022, 2022-05-06 [cit. 2022-12-18]. Dostupné z: https://cs.wikipedia.org/wiki/Meteorologická_stanice
7. STANĚK, Josef. *50 let meteorologické stanice Churáňov*. Praha: Český hydrometeorologický ústav, 2002. ISBN 80-85813-98-x.
8. IT Slovník. *It-slovník.cz* [online]. Vídeň: Kamil Havlíček, 2022 [cit. 2022-12-30]. Dostupné z: <https://it-slovník.cz/>
9. VODA, Zbyšek. *Průvodce světem Arduina*. Bučovice: Martin Stríž, 2015. ISBN 978-80-87106-90-7.
10. DITTRICH, Slavomír. *Sběr dat z meteorologických stanic* [online]. Praha, 2016 [cit. 2022-12-30]. Dostupné z: <https://dspace.cvut.cz/handle/10467/63048>. Diplomová práce. České vysoké učení technické v Praze.
11. Nepájivé pole. In: *Wikipedie: otevřená encyklopedie* [online]. San Francisco (CA): Wikimedia Foundation, 2020, 2020-07-13 [cit. 2023-01-01]. Dostupné z: https://cs.wikipedia.org/wiki/Nepájivé_pole

Přílohy

Příloha 1: Schéma hardwaru pomocí programu Fritzing



Příloha 2: Blokové schéma softwaru z programu Vizio



Příloha 3 Zdrojový kód pro amatérskou stanici

bmp280_18B20_test_esp8266_v1

This is a library for the BMP280 humidity, temperature & pressure sensor

Designed specifically to work with the Adafruit BMP280 Breakout

----> <http://www.adafruit.com/...651>

These sensors use I2C or SPI to communicate, 2 or 4 pins are required to interface.

Adafruit invests time and resources providing this open source code, please support Adafruit and open-source hardware by purchasing products from Adafruit!

Written by Limor Fried & Kevin Townsend for Adafruit Industries.
BSD license, all text above must be included in any redistribution

*****/

```
#include <Arduino.h>
#include <ESP8266WiFi.h>
#include <Wire.h>
#include <SPI.h>
#include <Adafruit_BMP280.h>

/*
  zapojeni pinu
  OneWire ds(2); // on D4 (a 4.7K resistor is necessary)
  kalibrace pinMode(13, INPUT_PULLUP); //D7 = pin 13
  wifi pinMode(12, INPUT_PULLUP); //D7 = pin 13
*/

#define pinKalibrace 13 //D7
#define pinWiFi 12 //D6

//kalibrovaní
char isKalibrace () {
  int d7 = digitalRead(pinKalibrace);
  if (d7 > 0)
  {
    return 0;
  } else {
    return 1;
  }
}

//wifiOFF
//na vstupu D6 testujeme zda je log 0 (WifiOn) nebo log 1 (Wifi PFF)
char isWifiOFF () {
  int d6 = digitalRead(pinWiFi);
  if (d6 > 0)
  {
    return 0;
  } else {
    return 1;
  }
}

//18B20
#include <OneWire.h>
byte p = 0;
byte pMax = 0;
float pMereni[8];
String pAdresy[8];
byte k;

//OneWire ds(5); // on pin 2 (a 4.7K resistor is necessary)
OneWire ds(2); // on pin4 = gpio2 (a 4.7K resistor is necessary)

void mereni18B20() {

  byte i;
  byte present = 0;
  byte type_s;
  byte data[12];
  byte addr[8];
  float celsius, fahrenheit;
  Serial.print("mereni start ");

  while (1 == 1) {
    if (!ds.search(addr)) {
      Serial.println(WiFi.localIP());
      Serial.println("No more addresses.");
      Serial.println();
    }
  }
}
```

```

ds.reset_search();
p = 0;
delay(250);
Serial.print("mereni end ");
return;
}

p++;
if (p > pMax) {
  pMax = p;
};

String s;
Serial.println();
Serial.print("ROM =");
s = printAddress(addr);
pAdresy[p] = s;
Serial.print(s);

if (OneWire::crc8(addr, 7) != addr[7]) {
  Serial.println("CRC is not valid!");
  return;
}
Serial.println();

// the first ROM byte indicates which chip
switch (addr[0]) {
  case 0x10:
    Serial.println(" Chip = DS18S20"); // or old DS1820
    type_s = 1;
    break;
  case 0x28:
    Serial.println(" Chip = DS18B20");
    type_s = 0;
    break;
  case 0x22:
    Serial.println(" Chip = DS1822");
    type_s = 0;
    break;
  default:
    Serial.println("Device is not a DS18x20 family device.");
    return;
}

ds.reset();
ds.select(addr);
ds.write(0x44, 1); // start conversion, with parasite power on at the end

delay(1000); // maybe 750ms is enough, maybe not
// we might do a ds.depower() here, but the reset will take care of it.

present = ds.reset();
ds.select(addr);
ds.write(0xBE); // Read Scratchpad

Serial.print(" Data = ");
Serial.print(present, HEX);
Serial.print(" ");
for ( i = 0; i < 9; i++) { // we need 9 bytes
  data[i] = ds.read();
  Serial.print(data[i], HEX);
  Serial.print(" ");
}
Serial.print(" CRC=");
Serial.print(OneWire::crc8(data, 8), HEX);
Serial.println();

int16_t raw = (data[1] << 8) | data[0];
if (type_s) {
  raw = raw << 3; // 9 bit resolution default
  if (data[7] == 0x10) {
    // "count remain" gives full 12 bit resolution
    raw = (raw & 0xFFF0) + 12 - data[6];
  }
} else {
  byte cfg = (data[4] & 0x60);
  // at lower res, the low bits are undefined, so let's zero them
  if (cfg == 0x00) raw = raw & ~7; // 9 bit resolution, 93.75 ms
  else if (cfg == 0x20) raw = raw & ~3; // 10 bit res, 187.5 ms
  else if (cfg == 0x40) raw = raw & ~1; // 11 bit res, 375 ms
  //// default is 12 bit resolution, 750 ms conversion time
}
celsius = (float)raw / 16.0;
Serial.print(" Temperature = ");
Serial.print(celsius);
pMereni[p] = celsius;

```

```

    Serial.println(" Celsius");
}

}

//dodani dispalye
#include <LiquidCrystal_I2C.h>

//pouziva GPIO4 - D2 - SDA ; GPIO5 - D1 - SCK
LiquidCrystal_I2C lcd(0x27, 20, 4); // set the LCD address to 0x27 for a 16 chars and 2 line display

void setupLCD()
{
    lcd.init();           // initialize the lcd

    // Print a message to the LCD.
    lcd.backlight();
    lcd.setCursor(3, 0);
    lcd.print("Hello, world!");

void lcdShow(String radek1, String radek2) {
    lcd.setCursor(0, 0);
    lcd.print(" ");
    lcd.setCursor(0, 0);
    lcd.print(radek1);

    lcd.setCursor(0, 1);
    lcd.print(" ");
    lcd.setCursor(0, 1);
    lcd.print(radek2);
}

char const hex_chars[16] = { '0', '1', '2', '3', '4', '5', '6', '7', '8', '9', 'A', 'B', 'C', 'D', 'E', 'F' };
// function to print a device address
String printAddress(byte deviceAddress[8]) {
    String str = "";
    for (uint8_t i = 0; i < 8; i++) {
        char const byte = deviceAddress[i];
        str += hex_chars[ ( byte & 0xF0 ) >> 4 ];
        str += hex_chars[ ( byte & 0x0F ) >> 0 ];
    }
    return str;
}

//cidlo BMP280
#define BMP_SCK (13)
#define BMP_MISO (12)
#define BMP_MOSI (11)
#define BMP_CS (10)

Adafruit_BMP280 bmp; // I2C
//Adafruit_BMP280 bmp(BMP_CS); // hardware SPI
//Adafruit_BMP280 bmp(BMP_CS, BMP_MOSI, BMP_MISO, BMP_SCK);

//wifi
const char* ssid = "SUMAVA_BRIDGE"; // "DETI";
const char* password = "JiEr2010";

const char* host = "js2002.g6.cz";
const String url = "/SUMAVA_data_in_2.php?Location=S&Debug=ON";
String response ;
String wifiResponse ;

WiFiServer server(80);

void wget ( String url ) {
    //test zda odesilat data
    if (isWifiOFF() == 1) {
        Serial.print("Posilani dat vypnuto ");
        return;
    }
}

// We start by connecting to a WiFi network

String inf ;

if (WiFi.status() != WL_CONNECTED) {
    Serial.print("Connecting to ");

```

```

Serial.println(ssid);

WiFi.begin(ssid, password);
while (WiFi.status() != WL_CONNECTED) {
  delay(500);
  Serial.print(".");
}
Serial.println("");
Serial.println("WiFi connected");
};
Serial.println("IP address: ");
Serial.println(WiFi.localIP());

Serial.print("connecting to ");
Serial.println(host);

// Use WiFiClient class to create TCP connections
WiFiClient client;
const int httpPort = 80;
wifiResponse = "OK - Data ulozena";
if (!client.connect(host, httpPort)) {
  Serial.println("connection failed");
  response = "connection failed";
  wifiResponse = "CHYBA - Data neuulozena";
  return ;
}

// We now create a URI for the request

Serial.print("Requesting URL: ");
Serial.println(url);

// This will send the request to the server
client.print(String("GET ") + url + " HTTP/1.1\r\n" +
  "Host: " + host + "\r\n" +
  "Connection: close\r\n\r\n");
delay(1000);
response = "";

// Read all the lines of the reply from server and print them to Serial
Serial.println("Respond:");
while (client.available()) {
  String line = client.readStringUntil('\r');
  Serial.print(line);
  response = response + line + "\r";
}

Serial.println();
Serial.println("closing connection");
}

void setup() {
  //Serial.begin(9600);
  Serial.begin(115200);
  while (!Serial) delay(100); // wait for native usb
  //mam zapnout WiFi?
  pinMode(pinWiFi, INPUT_PULLUP); //D6 = pin 12
  //kalibrui? NE => merim 1x z minutu ANO => merim 6x za minutu
  pinMode(pinKalibrace, INPUT_PULLUP); //D7 = pin 13

  Serial.println(F("setupLCD"));
  setupLCD();

  Serial.println(F("BMP280 test"));
  unsigned status;
  status = bmp.begin(BMP280_ADDRESS_ALT, BMP280_CHIPID);
  //status = bmp.begin();
  if (!status) {
    Serial.println(F("Could not find a valid BMP280 sensor, check wiring or "
      "try a different address!"));
    Serial.print("SensorID was: 0x"); Serial.println(bmp.sensorID(), 16);
    Serial.print("  ID of 0xFF probably means a bad address, a BMP 180 or BMP 085\n");
    Serial.print("  ID of 0x56-0x58 represents a BMP 280,\n");
    Serial.print("  ID of 0x60 represents a BME 280.\n");
    Serial.print("  ID of 0x61 represents a BME 680.\n");
    while (1) delay(10);
  } else {
    Serial.print("SensorID was: 0x"); Serial.println(bmp.sensorID(), 16);
  }
}

/* Default settings from datasheet. */
bmp.setSampling(Adafruit_BMP280::MODE_NORMAL, /* Operating Mode. */
  Adafruit_BMP280::SAMPLING_X2, /* Temp. oversampling */
  Adafruit_BMP280::SAMPLING_X16, /* Pressure oversampling */
  Adafruit_BMP280::FILTER_X16, /* Filtering. */

```

```

Adafruit_BMP280::STANDBY_MS_500); /* Standby time. */
}

void loop() {

String payload;
int httpCode;
String s;
char i;
char x[20];
float X;
String url1;
String inf1;
String inf2;

Serial.println("SUMAVA");

Serial.print(F("Temperature = "));
inf1 = "Mereni";
inf2 = "Cidlo BMP280";
lcdShow(inf1, inf2);
delay(2000);
X = bmp.readTemperature();
Serial.print(X);
Serial.println(" *C");
snprintf(x, 6, "%f", X);
url1 = "&T51=";
url1 = url1 + x;
inf1 = String(x) + " C";

X = bmp.readPressure();
Serial.print(F("Pressure = "));
Serial.print(X);
Serial.println(" Pa");
snprintf(x, 12, "%f", X);
url1 = url1 + "&P51=";
url1 = url1 + x;
url1 = url + url1;
inf2 = String(x) + " Pa";

lcdShow(inf1, inf2);
delay(5000);
//18b20
// float celsius;
float temperatureC;
String Adresa;
inf1 = "Mereni";
inf2 = "Cidlo 18B20";

lcdShow(inf1, inf2);
delay(2000);

mereni18B20 ();
s = "";
for (k = 1; k <= pMax; k++) {
// Search the wire for address
if (pAdresy[k] ) {
temperatureC = pMereni[k];
Adresa = pAdresy[k];
Serial.print(temperatureC);
Serial.print("C, ID = ");
Serial.println(Adresa);
//s = hostPage2;
s = s + "&ID6" + String(k) + "=" + Adresa + "&T6" + String(k) + "=" + String(temperatureC);
//read data
};
};
Serial.println(s);
url1 = url1 + s;
inf1 = "T1 = " + String(pMereni[1]) + "C";
inf2 = "T2 = " + String(pMereni[2]) + "C";
lcdShow(inf1, inf2);
delay(5000);

Serial.println("*****");
Serial.println(url1);

inf1 = " Ukladani dat";
inf2 = "";
lcdShow(inf1, inf2);
delay(2000);

if (isWifiOFF () == 0) {
wget(url1);
Serial.println("*****");
}
}

```

```
//Serial.println(response);
//Serial.println("*****");

Serial.println();
inf2 = wifiResponse;
} else {
  inf2 = "WiFi je vypnuta";
}
lcdShow(inf1, inf2);
delay(5000);

if (isKalibrace() == 0) {
  inf1 = " Konec mereni";
  inf2 = "pauza 45 sec";
  lcdShow(inf1, inf2);
  Serial.println("deep sleep");
  ESP.deepSleep(45e6); // 45e6 is 45 seconds
} else {
  inf1 = " Konec mereni";
  inf2 = "pauza 2 sec";
  lcdShow(inf1, inf2);

  Serial.println("kalibrace");
  delay(2000); //cekej 2 sekundy
}
}
```