

# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ  
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

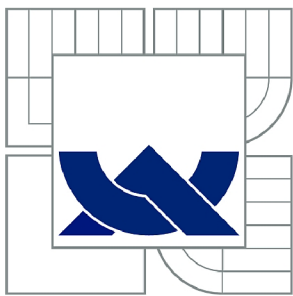
STANICE PRO LOGOVÁNÍ TEPLoty A VLHKOSTI

BAKALÁŘSKÁ PRÁCE  
BACHELOR'S THESIS

AUTOR PRÁCE  
AUTHOR

TOMÁŠ VILÍMEK

BRNO 2015



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ**

**ÚSTAV TELEKOMUNIKACÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF TELECOMMUNICATIONS

## **STANICE PRO LOGOVÁNÍ TEPLoty A VLHKOSTI**

TEMPERATURE AND HUMIDITY MEASURING STATION

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**TOMÁŠ VILÍMEK**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. ZOLTÁN SZABÓ, Ph.D.**

BRNO 2015



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav telekomunikací

# Bakalářská práce

bakalářský studijní obor  
Teleinformatika

**Student:** Tomáš Vilímek

**ID:** 120542

**Ročník:** 3

**Akademický rok:** 2014/2015

**NÁZEV TÉMATU:**

## Stanice pro logování teploty a vlhkosti

### POKYNY PRO VYPRACOVÁNÍ:

Cílem semestrální práce je návrh a vývoj zařízení, který bude schopen měřit a ukládat naměřené hodnoty vlhkosti vzduchu a teploty ve více bodech v prostoru. Na displeji přístroje bude možno zobrazit aktuální, minimální i maximální hodnoty ze všech senzorů a dále naměřená data přenést do osobního počítače pro další zpracování. Přístroj realizujte a správnou funkci ověřte.

### DOPORUČENÁ LITERATURA:

[1] MATOUŠEK, D. Práce s mikrokontroléry ATMEL AVR - ATmega16. 1. vyd. Praha : BEN - technická literatura, 2006. 320 s. ISBN 80-7300-174-8.

[2] SLINTÁK, S. Arduino a USB. [online]. [cit. 2014-12-08]. Dostupné z URL: <<http://uart.cz/394/arduino-a-usb/>>.

**Termín zadání:** 9.2.2015

**Termín odevzdání:** 2.6.2015

**Vedoucí práce:** Ing. Zoltán Szabó, Ph.D.

**Konzultanti bakalářské práce:**

**doc. Ing. Jiří Mišurec, CSc.**

*Předseda oborové rady*

### UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **ABSTRAKT**

Bakalářská práce s názvem „Stanice pro logování teploty a vlhkosti“ je zaměřená na návrh zařízení, které bude schopno měřit, uchovávat a zobrazovat data o teplotě a vlhkosti z připojených senzorů. V teoretické části jsou popsány jednotlivé použité součásti zařízení jako LCD modul, senzory, modul reálného času, ukládání dat, komunikace s počítačem a některé možné alternativy senzorů. V praktické části je shrnuto řešení a implementace jednotlivých částí, schéma zapojení, popis funkcí a obsluhy zařízení a rozbor programového vybavení zařízení.

## **KLÍČOVÁ SLOVA**

Teplota, Vlhkost, Mikrokontroler, Atmel, ATmega, RTC, Arduino, LCD

## **ABSTRACT**

Bachelor thesis entitled „Temperature and humidity measuring station“ is focused on design of a device, that will be able to measure, save and display information about temperature and humidity from connected sensors. There is described used components such as LCD display, sensors, RTC, data saving, communication with personal computer and description of alternative sensors. Practical part contains implementation of components, scheme, manual for the device and description of device firmware.

## **KEYWORDS**

Temperature, Humidity, Microprocessor, ATmel, ATmega, RTC, Arduino, LCD

VILÍMEK, Tomáš *Stanice pro logování teploty a vlhkosti*: bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2015. 44 s. Vedoucí práce byl Ing. Zoltán Szabó, Ph.D.



## PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Stanice pro logování teploty a vlhkosti“ jsem vypracoval(a) samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor(ka) uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil(a) autorská práva třetích osob, zejména jsem nezasáhl(a) nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom(a) následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno .....

.....

podpis autora(-ky)

## PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce panu Ing. Zoltánu Szabóovi za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno .....

.....

podpis autora(-ky)

# OBSAH

<b>Úvod</b>	<b>10</b>
<b>1 Teoretická část</b>	<b>11</b>
1.1 Vývojový kit - Arduino Uno . . . . .	11
1.2 Arduino IDE . . . . .	11
1.3 Mikrokontroler . . . . .	13
1.3.1 Mikrokontroler ATmega328 . . . . .	15
1.4 Sériový převodník . . . . .	15
1.4.1 Rozhraní UART . . . . .	16
1.4.2 Rozhraní USB . . . . .	16
1.5 Měření teploty a vlhkosti . . . . .	17
1.5.1 Měření teploty . . . . .	17
1.5.2 Měření vlhkosti . . . . .	21
1.6 Zobrazení informací . . . . .	22
1.7 Obvod reálného času . . . . .	23
1.8 Datové úložiště . . . . .	23
1.8.1 Technologie Flash . . . . .	23
1.8.2 SPI protokol . . . . .	24
<b>2 Praktická část</b>	<b>25</b>
2.1 Blokové schéma . . . . .	25
2.2 Napájení zařízení . . . . .	26
2.3 Programové vybavení . . . . .	27
2.3.1 Hlavní smyčka . . . . .	27
2.3.2 Měření a převod hodnot z čidel . . . . .	30
2.3.3 Ukládání hodnot . . . . .	31
2.3.4 Komunikace s PC . . . . .	33
2.3.5 Nahrání programové obsluhy . . . . .	33
2.4 Obsluha a konstrukce zařízení . . . . .	34
2.5 Obslužný PC program . . . . .	35
<b>3 Závěr</b>	<b>37</b>
<b>Literatura</b>	<b>38</b>
<b>Seznam symbolů, veličin a zkratk</b>	<b>40</b>
<b>Seznam příloh</b>	<b>41</b>

<b>A Schéma zapojení</b>	<b>42</b>
<b>B Návrh plošného spoje</b>	<b>43</b>

# SEZNAM OBRÁZKŮ

1.1	Arduino Uno . . . . .	11
1.2	LCD Keypad Shield . . . . .	12
1.3	Datalogger shield . . . . .	12
1.4	Arduino IDE . . . . .	13
1.5	Von Neumannova architektura . . . . .	14
1.6	Harvardská architektura . . . . .	14
1.7	ATMega328 [2] . . . . .	15
1.8	FT232RL [6] . . . . .	16
1.9	Schéma zapojení USB rozhraní [7] . . . . .	17
1.10	Schéma LM335 [12] . . . . .	20
1.11	Čidlo vlhkosti . . . . .	22
1.12	Alfanumerický LCD displej . . . . .	22
1.13	DS1307 [5] . . . . .	23
1.14	Vlastnosti flash [17] . . . . .	24
1.15	SPI sběrnice [14] . . . . .	24
2.1	Blokové schéma . . . . .	25
2.2	Regulátor proudu[18] . . . . .	26
2.3	Odpojovač baterie . . . . .	27
2.4	Vývojový diagram . . . . .	28
2.5	Zapojení čidla LM335[12] . . . . .	30
2.6	Zapojení čidla HIH-4000-001[20] . . . . .	31
2.7	Graf závislosti Výstupního napětí na % relativní vlhkosti[13] . . . . .	32
2.8	Programování zařízení . . . . .	34
2.9	Zkonstruované zařízení . . . . .	35
2.10	Zobrazení vlhkosti . . . . .	36
2.11	Zobrazení teploty 1. čidla . . . . .	36
2.12	Obslužný program . . . . .	36
A.1	Schéma zapojení . . . . .	42
B.1	Návrh plošného spoje vrchní strana . . . . .	43
B.2	Návrh plošného spoje spodní strana . . . . .	44

# SEZNAM TABULEK

# ÚVOD

Tato práce je zaměřená na návrh a vývoj zařízení, které bude měřit a ukládat naměřené hodnoty vlhkosti a teploty vzduchu z více míst. Tyto hodnoty bude zařízení zobrazovat na displeji a po připojení k počítači, naměřená data bude schopna přenést pomocí běžného rozhraní USB. K tomu bude potřeba vytvořit jednoduchou obslužnou aplikaci, která data stáhne a následně zpracuje do požadované podoby.

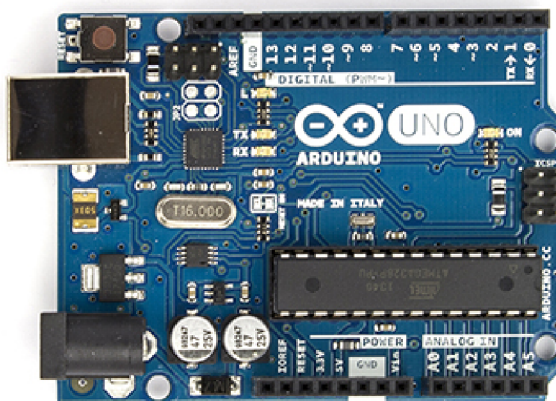
Zařízení bude založeno na mikrokontroleru ATmega od firmy Atmel a vyvinuto pomocí vývojového kitu Arduino Uno a příslušných shieldů a komponentů. Součástí vývoje bude návrh desky plošného spoje a s vhodným rozmístěním komponentů a následné otestování zařízení.

# 1 TEORETICKÁ ČÁST

V teoretické části jsou popsány jednotlivé použité součásti stanice, jejich funkce a jejich základní vlastnosti.

## 1.1 Vývojový kit - Arduino Uno

Arduino Uno obr. 1.1 je otevřená vývojová platforma založená na mikrokontroleru ATmega328 s frekvencí 16MHz a vývojovém prostředí, které vychází z prostředí Wiring. Může být využito pro stavbu vlastních samostatných zařízení nebo stavbu vlastního zařízení spolupracujícím s počítačem. Arduino má 20 vstupně-výstupních vývodů, z nichž některé lze použít jako výstup pro PWM průběh, dále pak analogové vývody. S počítačem komunikuje přes USB rozhraní, z kterého je i napájeno, případně lze napájet z externího adaptéru nebo z baterie. Arduino lze rozšířit o tzv. shieldy, které lze jednoduše měnit, případně i spojovat a lze s nimi navrhnout a otestovat téměř cokoli. K vytvoření stanice byly potřeba celkem dva shieldy (LCD Keypad Shield obr. 1.2 a Datalogger shield obr. 1.3). [1] [2]



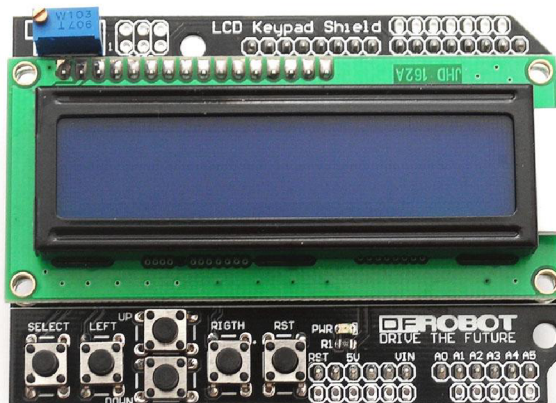
Obr. 1.1: Arduino Uno

## 1.2 Arduino IDE

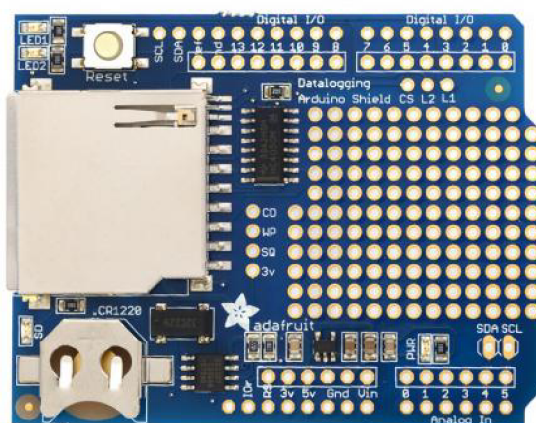
Aplikace pro vývojový kit Arduino Uno se vyvíjí pomocí jednoduchého vývojového prostředí Arduino IDE obr. 1.4. Toto vývojové prostředí obsahuje textové pole pro tvorbu aplikací, základní vzorové aplikace včetně základních knihoven potřebných k vývoji. Prostředí je napsáno v jazyce Java a díky tomu ho lze spustit na libovolném operačním systému, na kterém běží Java Virtual Machine. Programovací



jazyk, pro psaní aplikací, vychází z prostředí Wiring a je velmi podobný jazyku C/C++. [1]



Obr. 1.2: LCD Keypad Shield



Obr. 1.3: Datalogger shield



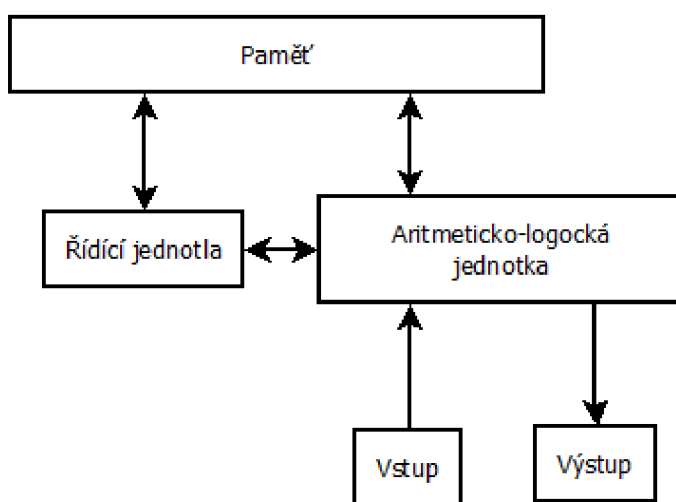
Obr. 1.4: Arduino IDE

## 1.3 Mikrokontroler

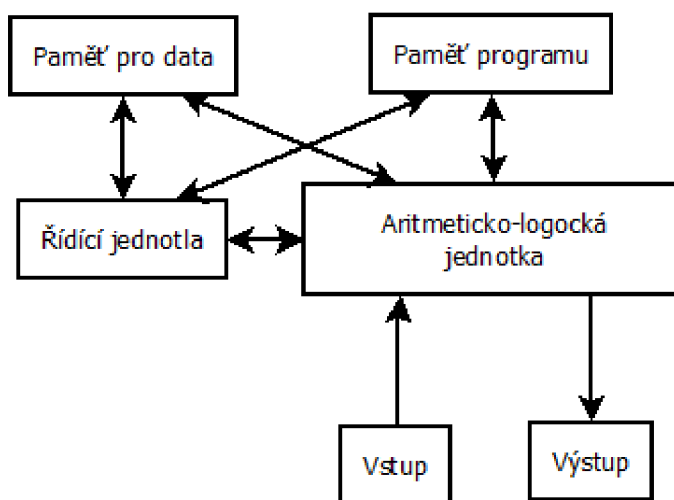
Mikrokontroler se skládá z logických obvodů - řídicí jednotka, aritmeticko-logická jednotka, paměť a obvody vstupu/výstupu. Jeho činnost je řízena programem uloženým v paměti. Při běhu programu, kdy mikrokontroler vykonává činnost udávanou programem (aritmeticko-logické operace, přesun dat, atd.), nezávisle provádí čin-

nosti, které jsou nezávislé na programu (přerušeni - vnější nebo vnitřní, kontrola napájení, A/D převody, atd.). Tyto činnosti mohou ovlivnit běh programu.

Architekturu mikrokontrolerů lze rozdělit podle sdílení paměti programu a dat. Von Neumannova architektura obr. 1.5 má společnou paměť pro program i data, operace se provádějí sekvenčně a Harvardská architektura obr. 1.6 má paměť zvlášť pro program a zvlášť pro data, operace se provádějí paralelně. Dále pak podle instrukční sady jakou mikrokontroler používá. Instrukční sada RISC mikrokontrolerů je zaměřena na jednoduché, vysoce optimalizované strojové instrukce stejné délky. Naopak instrukční sada CISC je zaměřena na komplexní instrukce různé délky a různé doby provedení, které by šly naprogramovat pomocí jednodušších instrukcí. [2] [3]



Obr. 1.5: Von Neumannova architektura

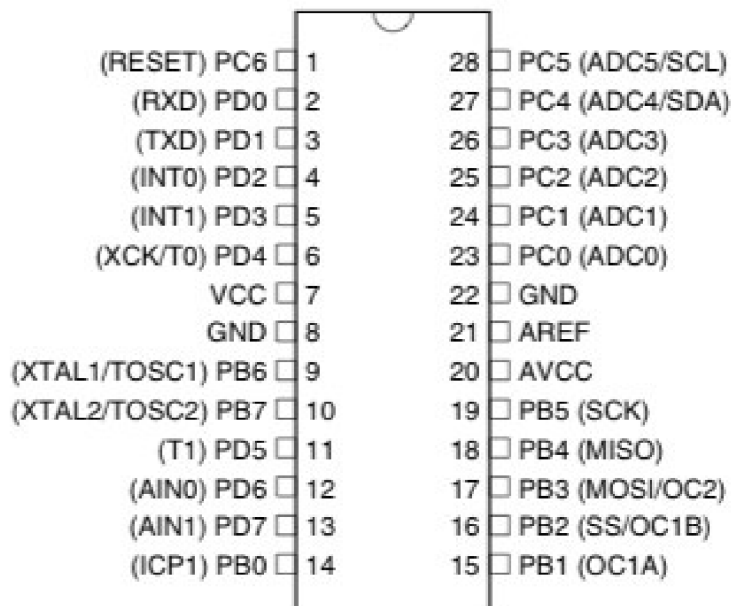


Obr. 1.6: Harvardská architektura

### 1.3.1 Mikrokontroler ATmega328

Mikrokontroler ATmega328 obr. 1.7 patří do rodiny Atmel AVR. Tyto mikrokontrolery jsou založeny na Harvardské architektuře s redukovanou instrukční sadou (RISC). Dále pak obsahuje zejména časovače, A/D převodníky, komparátor, sériovou linku, programovou paměť flash o velikosti 32kB (0,5kB je obsazeno bootlo-aderem, který je potřebný pro nahrání a spuštění vlastního programu) a operační paměť RAM o velikosti 2kB. ATmega328 má 14 digitálních vstupně/výstupních vývodů, z nichž jich 6 lze použít jako výstup PWM průběhu, a 6 analogových vstupů, které jsou vzorkovány 10-bit A/D převodníkem.

Pro správnou funkčnost mikrokontroleru, musí být časován hodinovým signálem. Hodinový signál lze získat připojením vnějšího krystalu o frekvenci 16 MHz s kondenzátory na příslušné vývody mikrokontroleru nebo využít vnitřní kalibrovaný oscilátor o frekvenci 8 MHz. [1] [2] [3]

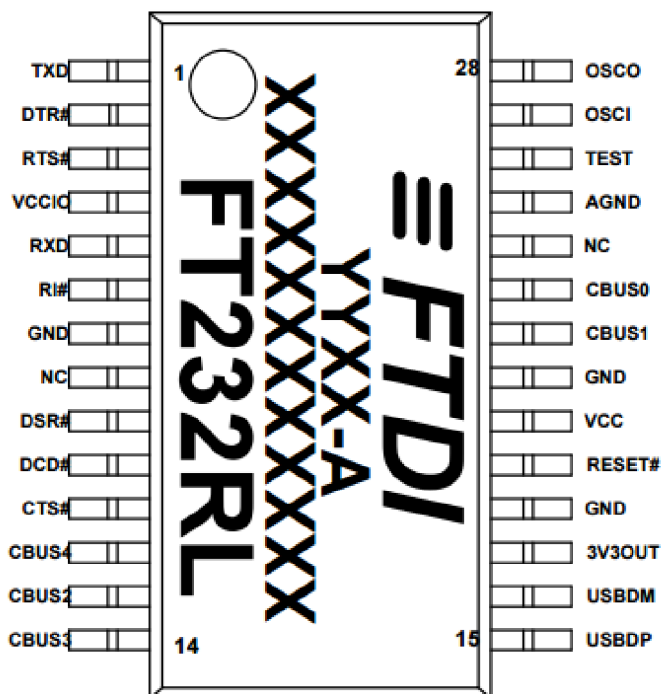


Obr. 1.7: ATmega328 [2]

## 1.4 Sériový převodník

Aby mohl mikrokontroler jednoduše komunikovat s počítačem pomocí USB rozhraní, je nutné vyřešit převod mezi sériovou linkou mikrokontroleru UART a USB rozhraním. K tomu lze využít převodník FT232RL 1.8 naprogramovaný jako USB-UART

převodník. Převodník FT232RL je pak k hlavnímu mikrokontroleru ATmega328 připojen přes UART rozhraní a po připojení k počítači se zařízení hlásí jako standardní sériová linka. [1] [4] [6]



Obr. 1.8: FT232RL [6]

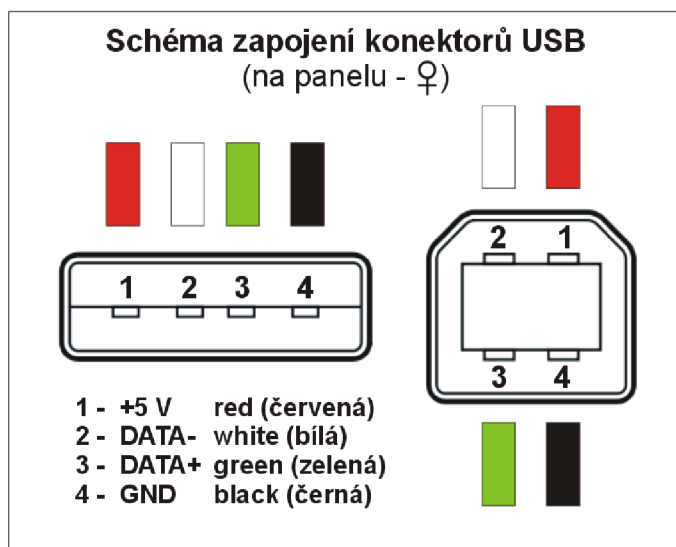
### 1.4.1 Rozhraní UART

Rozhraní UART je asynchronní sériové rozhraní, které se používá ke komunikaci mezi přístroji. Pro komunikaci jsou zapotřebí vodiče Rx a Tx, přes která se odesílají a přijímají data. Jelikož se jedná o asynchronní sériový přenos, musí být vysílač i přijímač vlastní generátor hodinového signálu. Generátor hodinového signálu musí mít na obou stejný kmitočet a tento kmitočet lze řídit a tím měnit rychlost přenosu. [4]

### 1.4.2 Rozhraní USB

Rozhraní USB je universální sériová sběrnice přes kterou lze provést obousměrnou komunikaci různých zařízení s počítačem. USB je popisováno jako zřetěžená hvězdicová architektura, ale každé zařízení komunikuje s hostitelským portem, jako kdyby mělo vyhrazené spojení. Komunikace probíhá přes USB hub, každý hub může mít maximálně sedm portů. Huby se mohou řetězit až do sedmé úrovně nebo maximálně

do 127 portů. Rozhraní USB má čtyři vodiče, z nichž dva jsou datové a zbylé dva jsou určena k napájení zařízení o napětí 5 V. Přenosová rychlost může dosahovat až 480 Mb/s. [7] [15]



Obr. 1.9: Schéma zapojení USB rozhraní [7]

## 1.5 Měření teploty a vlhkosti

V této kapitole jsou uvedeny základní metody měření teploty a vlhkosti, na kterých jsou založeny měřicí senzory. Na základě vlastností těchto metod, jsou pak vybrány a popsány senzory použité pro tento projekt.

### 1.5.1 Měření teploty

Teplota je veličina, která charakterizuje, zda látka při tepelném kontaktu s jinou látkou bude či nebude v tepelné rovnováze (zda bude či nebude přijímat nebo předávat teplo). Jedná se o skalární veličinu, která je vhodná k popisu ustálených makroskopických systémů. Teplota souvisí s kinetickou energií částic látky a je klíčovou veličinou pro popis tepelných jevů. Teplota je základní fyzikální veličinou soustavy SI a její jednotkou je kelvin (K), vedlejší jednotkou stupeň Celsia (°C). Nejnižší možná teplota je absolutní nula ( $0\text{K} = -273,15\text{°C}$ ). [8] [9]

Měřit teplotu lze několika způsoby a podle typu výstupu z čidla lze rozdělit způsoby na tyto skupiny:

## Mechanické měření

Pro mechanické měření teploty se používají zejména bimetalické teploměry. Tyto teploměry jsou vyrobeny ze dvou podélně spojených pásků kovů o různé tepelné roztažnosti. Vlivem změny teploty se pásek deformuje. Míra deformace je pak přenášena přímo nebo převodem na vstup. Tento způsob měření je pro digitální zpracování není příliš vhodný, především kvůli velké nepřesnosti, způsobené stárnutím a oxidací kovů. [8] [9]

## Analogové měření

- **Termistor**

Termistory lze realizovat ve velmi malých rozměrech. Jejich měřicí rozsah je obvykle od  $-60\text{ }^{\circ}\text{C}$  do  $200\text{ }^{\circ}\text{C}$ . Termistory vynikají velmi dobrou teplotní citlivostí, přesností měření dosahující  $\pm 0,1\text{ }^{\circ}\text{C}$  a nízkou cenou. Jejich nevýhoda je malý rozsah měřených teplot, kdy při větším rozsahu teplot se díky nelineárním průběhu měření vyskytují velké chyby měření. Avšak pro měření teplot vzduchu by termistory byly více než dostačující. Jejich nelineární závislost odporu termistorů lze vyjádřit exponenciální funkcí. [9]

$$R_t = R_{t_0}^{-B_T(\frac{1}{t_0 t})}, \quad (1.1)$$

kde je

$R_t$  odpor termistoru při teplotě  $t[\Omega]$ ,

$R_{t_0}$  odpor termistoru při teplotě  $t_0[\Omega]$ ,

$B_T$  materiálová konstanta $[-]$ ,

- **Termoelektrický článek.**

U termoelektrických článků se k převodu teploty na elektrické napětí využívá termoelektrického jevu. Přiložíme-li k sobě dva různé kovy, vznikne na nich rozdíl napětí (konstantní stykový potenciál). Tento potenciál je značně závislý na teplotě a tato závislost lze vyjádřit vztahem:

$$U_E = A + B\Delta t + C\Delta t^2, \quad (1.2)$$

kde je

$U_E$  termoelektrické napětí  $[V]$ ,

$A, B, C$  konstanty závislé na použitých kovech $[-]$ ,

$\Delta t$  rozdíl teplot mezi kovy  $[K]$ .

Aktivní (měřicí) konec termočlánku se nazývá horký a druhý konec studený. Studený konec termočlánku je nutné udržovat na konstantní teplotě. Výsledné

termoelektrické napětí je pak dáno rozdílem napětí mezi horkým a studeným koncem. Při dodržení konstantní teploty studeného konce, bude výsledné termoelektrické napětí přímo úměrné teplotě na horkém konci. Závislost napětí na teplotě není lineární, k lineární závislosti se blíží pouze v určitém teplotním intervalu, a jejich přesnost není vysoká. Jejich požití je zejména při speciálních vysokoteplotních aplikacích. [9]

- **Odporové senzory**

Odporové senzory jsou založeny na závislosti elektrického odporu vodičů nebo polovodičů na teplotě. Odpor senzoru se stanoví podle Ohmova zákona

$$R = \frac{U}{I} [\Omega] \quad (1.3)$$

při konstantním měřícím proudu  $I$  je měřen úbytek napětí na senzoru. Odporové senzory vynikají dlouhodobou stálostí a velmi širokým rozsahem měřitelných teplot ( $-200\text{ }^{\circ}\text{C}$  až  $1000\text{ }^{\circ}\text{C}$ ). Avšak závislost odporu odporových sensorů není lineární a díky velice malé změně odporu, při jednotkovém skoku teploty jsou tyto senzory bez dodatečných obvodů nevhodné pro velmi přesná měření, ale pro měření teploty vzduchu dostačující. Základní odporové snímače jsou platinové s odporem  $100\ \Omega$  při  $0\text{ }^{\circ}\text{C}$ . Závislost odporu odporových sensorů lze vyjádřit polynomem 3. řádu. [9]

$$\rho_t = \rho_{t_0} [1 + at_1 + bt_1^2 + c(t_1 - 100)], \quad (1.4)$$

kde je

$\rho_t$  měrný el. odpor při teplotě  $t_1$  [ $\Omega m$ ],

$\rho_{t_0}$  měrný el. odpor při teplotě  $0\text{ }^{\circ}\text{C}$  [ $\Omega m$ ],

$a$  lineární teplotní koeficient odporu [ $K^{-1}$ ],

$b$  kvadratický teplotní koeficient odporu [ $K^{-2}$ ],

$c$  mocninný teplotní koeficient 3.řádu, pro teploty do  $0\text{ }^{\circ}\text{C}$  [ $K^{-3}$ ].

- **Polovodičové senzory**

Senzory založené na polovodičích využívají Zenerova jevu. Teplotní dioda pracuje v závěrném směru a stabilizuje napětí na hodnotu danou teplotou. Největší předností polovodičových sensorů je jejich lineární průběh měření. Napětí je přímo úměrné absolutní teplotě a jejich charakteristika je dána pouze strmostí. Není třeba linearizace, ale pouze přesné reference se kterou se stabilizované napětí na diodě srovnává. Jejich nevýhodou je větší odchylka v celém rozsahu měřených teplot ( $-55\text{ }^{\circ}\text{C}$  až  $150\text{ }^{\circ}\text{C}$ ). Tyto obvody je nutné kalibrovat a kalibrace z výroby většinou probíhá pro  $25\text{ }^{\circ}\text{C}$ . Odchylka lze řešit dalším vývodem z čidla, přes který se kompenzují negativní vlivy na výslednou hodnotu.



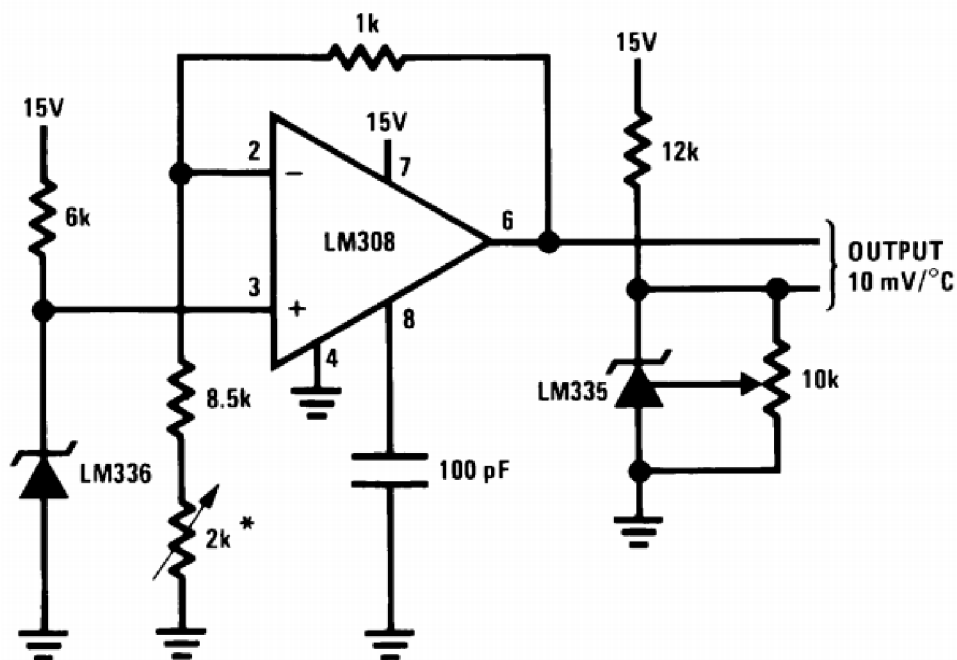
Díky své lineární závislosti teploty na napětí jsou pro potřeby projektu, i přes nižší, ale dostačující přesnost, ideální právě tyto senzory. [9]

## Digitální měření

Pro digitální měření se zpravidla používají polovodičové senzory, které mají lineární závislost napětí na teplotě. Díky snadné integraci tohoto průběhu lze do pouzdra snímače jednoduše přidat číslicový modulátor, který plní funkci AD převodníku. Výstupní signál pak nabývá binárních hodnot, které si mikrokontroler pouze přečte a zpracuje bez nutnosti jakéhokoli mezipřevodu. Komunikace s tímto senzorem by výhradně probíhala po sériové lince. Tyto senzory není potřeba kalibrovat a jejich přesnost je dána typem obvodu. [9]

## Teplotní čidlo LM335

Teplotní čidlo LM335 obr. 1.10 obsahuje dvě diody, které využívají Zenerova jevu. První dioda (LM335) mění své stabilizované napětí dle teploty a druhá (referenční dioda LM336) se snaží svou referenční hodnotu držet s co nejmenší hodnotou. Rozdílový zesilovač následně hodnoty napětí diod odečte a jejich rozdíl dá na výstup. Při použití referenčního napětí 5 V, pak rozdíl 0 V značí teplotu 0 °C. [9] [12]



Obr. 1.10: Schéma LM335 [12]

## 1.5.2 Měření vlhkosti

Vlhký vzduch je směsí vzduchu, tvořeného cca 78% dusíku, 21% kyslíku, 0,03% oxidu uhličitého a 0,97% ostatních plynů a vodní páry. Vlhkost vzduchu je dána obsahem vodních par, které jsou ve vzduchu vždy, ale v proměnlivém množství. Vlhkost vzduchu můžeme vyjádřit několika hodnotami. Měrná vlhkost vzduchu je skutečné množství vodní páry ve vzduchu při dané teplotě. Maximální vlhkost je největší množství vody, které je vzduch za dané teploty a tlaku schopen pojmout. Relativní vlhkost vzduchu je poměr měrné ku maximální vlhkosti za dané teploty a je udávána v procentech. Vzhledem k organismu živočichů se jedná o hodnotu výstižnou, a proto se vlhkost vyjadřuje nejčastěji v této hodnotě. [10] [11]

Pro měření vlhkosti vzduchu je mnoho zařízení pracujících na různých principech, ale pouze několik z nich je vhodná pro následný převod na elektrickou veličinu.

### Odporové vlhkoměry

Odporové vlhkoměry jsou založeny na změně elektrického odporu. Senzor využívá elektrolytické vodivosti mezi destičkami, které jsou iontovou solí. Různou koncentrací této soli lze ovlivnit rozsah měření relativní vlhkosti. Velikost odporu je závislá nejen na vlhkosti, ale i na teplotě, proto je nutné měřit teplotu a soustavu stabilizovat. Senzor nelze použít při vyšších teplotách a je citlivý na orosení. [11]

### Kapacitní vlhkoměry

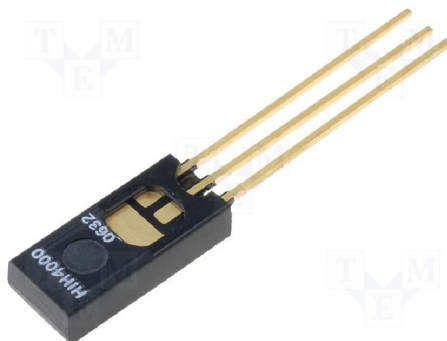
Konstrukce je velmi podobná odporovým sensorům. Základ senzoru je kondenzátor, jehož dielektrikum tvoří tenká vrstvička materiálu reagující na vlhkost okolí. Při změně vlhkosti okolí se velmi rychle mění jeho elektrický odpor a kapacita, čili impedance. Tyto senzory jsou citlivé, přesné a také se vyrábějí s integrovaným zpracováním signálu, kdy na výstupu není hodnota kapacity, ale například elektrické napětí, které je vhodnější na zpracování, případně i digitální hodnota. [11]

### Vlhkoměry s vyhříváními termistory

Senzory využívají závislost tepelné vodivosti vzduchu na jeho vlhkosti. V senzoru jsou zabudovány dva stejné termistory, z nichž jeden je hermeticky uzavřen a druhý je přístupný okolí. Průchodem proudu se termistory zahřívají a jejich ochlazení závisí na jejich okolí. Senzor pak poskytuje signál úměrný absolutní vlhkosti. [11]

## Čidlo vlhkosti

Jako čidlo vlhkosti bylo vybráno čidlo HIH-4000-001 1.11 od společnosti Honeywell. Čidlo je konstruováno jako kapacitní vlhkoměr a jeho výstupem je napětí úměrné relativní vlhkosti okolí. Rozsah měření je 0-100% relativní vlhkosti s přesností +/- 0,3%. [13]



Obr. 1.11: Čidlo vlhkosti

## 1.6 Zobrazení informací

Pro zobrazení naměřených údajů byl zvolen alfanumerický LCD displej o velikosti 2 řádků a 16 sloupců, který vyniká dobrou čitelností a nízkou spotřebou. Jeho další předností je zabudované vlastní podsvícení, které zaručuje dobrou čitelnost i za sníženého osvětlení. Toto podsvícení lze hardwarově či softwarově vypínat a zapínat, případně měnit i jas a tím regulovat spotřebu. Jednotlivé znaky jsou tvořeny 35 body uspořádané v matici 5x7 znaků. Znaky jsou na displej zadávány pomocí ASCII znaků, ale pomocí programové lze definovat vlastní znaky. [16]

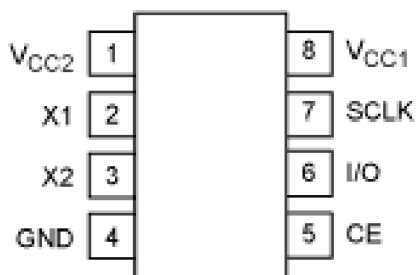


Obr. 1.12: Alfanumerický LCD displej

## 1.7 Obvod reálného času

Úkolem obvodu reálného času je udržovat údaj o aktuální čase. Napájení obvodu je dvojí - přímo ze zařízení a ze záložní baterie (zpravidla nižší napětí). Pokud je napájení ze zařízení odpojeno, nelze s obvodem komunikovat. Po odpojení napájení údaj o čase se smaže. Když je připojen k napájení, počítá impulzy z externího oscilátoru a ukládá aktuální hodnoty data a času. V tomto režimu má velice nízký příkon cca 500nA. S obvodem lze komunikovat pomocí  $I^2C$  sběrnice.

Vybraný modul DS1307 obr. 1.13 od firmy Dallas umožňuje číst časové údaje s přesností jednotek vteřin až po rok. Přesný čas je odvozen od vnějšího krystalu a časový údaj je zálohován baterií. Mikrokontroler s modulem RTC komunikuje přes  $I^2C$  sběrnici, která se skládá ze tří vodičů - SCLK(hodinový signál), I/O(data), CE(zápis, čtení). [5]



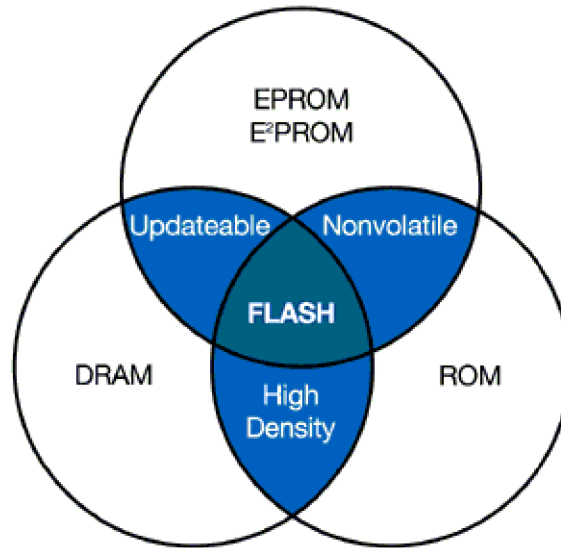
Obr. 1.13: DS1307 [5]

## 1.8 Datové úložiště

Jelikož mikrokontroler disponuje velice omezenou pamětí, musí být k němu připojena externí paměť, kam lze uložit naměřená data z čidel. Proto byla vybrána SD paměťová karta, která je velmi rozšířená a levná. SD karta je založena na technologii flash a s mikrokontrolerem komunikuje přes SPI protokol.

### 1.8.1 Technologie Flash

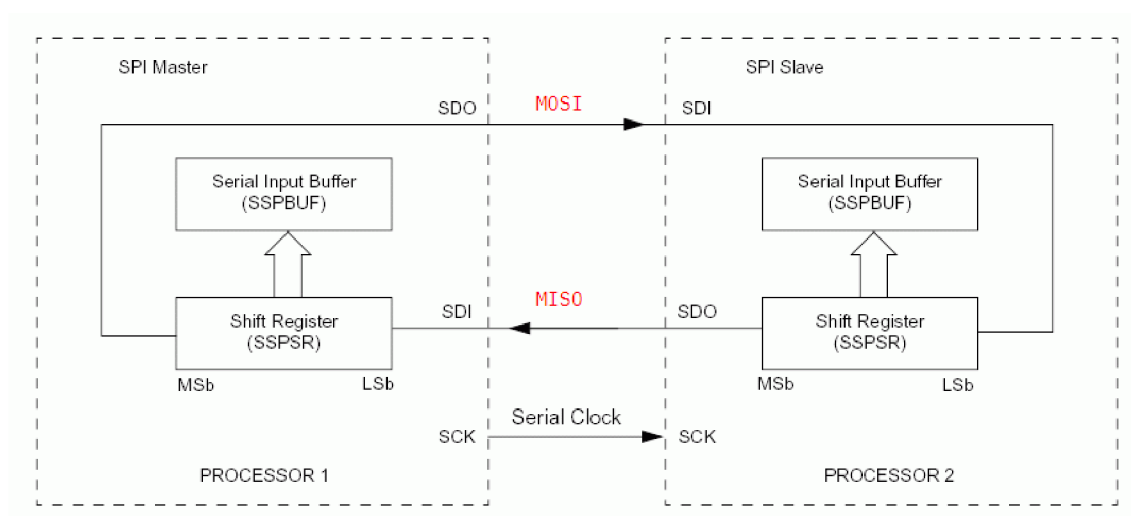
Technologie flash 1.14 využívá paměti EEPROM a zahrnuje v sobě výhody několika druhů pamětí - náhodnou přístupnost, velkou kapacitu a po odpojení napájení zachování dat. Lze ji programovat i zabudovanou v zařízení a neobsahuje žádné pohyblivé části. Paměti jsou rozměrově malé a jsou odolné vůči vnějšímu prostředí. [17]



Obr. 1.14: Vlastnosti flash [17]

## 1.8.2 SPI protokol

SPI 1.15 je sériová periferní sběrnice ke komunikaci mikrokontroleru s ostatními komunikujícími uzly. Komunikace probíhá na společné sběrnici, ke které je připojeno dvě a více komunikujících uzlů, a využívá čtyř vodičů. Jeden uzel je v roli takzvaného řadiče sběrnice neboli v režimu *Master* a ostatní uzly jsou v režimu *Slave*. Uzel pracující jako *Master* generuje hodinový signál (vodič SCK), rozvedený do všech uzlů, tím je umožněn obousměrný synchronní přenos dat po vodičích MISO (*Master in Slave out*) a MOSI (*Master out Slave in*). Uzel *Master* vybírá uzel *Slave*, se kterým bude komunikovat, pomocí signálu SSEL (*Slave select*). [14]



Obr. 1.15: SPI sběrnice [14]

## 2 PRAKTICKÁ ČÁST

V praktické části jsou uvedeny základní součásti zařízení, jeho napájení, programová obsluha a programová obsluha na PC ke komunikaci se zařízením.

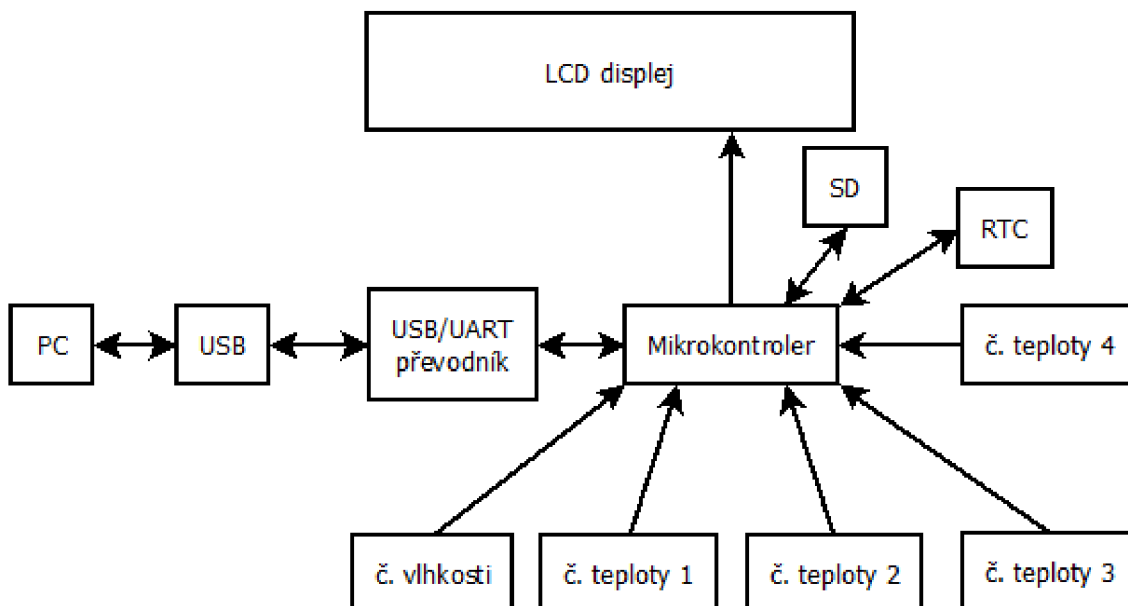
Řídicí obvod je mikrokontroler ATmega328 komunikující s LCD displejem, úložným prostorem na SD kartě, RTC modulem reálného času, jednotlivými čidly teploty a vlhkosti, pomocí USB/UART převodníku FT232RL komunikuje s PC a pomocí tlačítek s obsluhující osobou.

Navržené zařízení se nejdříve pomocí Arduino Uno, příslušných shieldů a čidel otestuje a poté se zkonstruuje vlastní jednoduché zařízení, které bude složeno jen z potřebných modulů a bude kompaktnější. Zařízení bude mít vlastní primární i sekundární napájení, jednoduché ovládání a možnost připojení čidel přes konektory jack. Výsledný obvod A.1 a navržený plošný spoj B.1 B.2 je v příloze.

### 2.1 Blokové schéma

Základní blokové schéma obvodu obr. 2.1, celé schéma zapojení zařízení obr. A.1 lze prohlédnout v příloze. V blokovém schématu jsou zobrazeny základní součásti zařízení a šipkami jsou znázorněny komunikační toky mezi těmito součástmi.

Srdcem zařízení je mikrokontroler, přes který jde veškerá komunikace. LCD displej po stisku uživatelského tlačítka zobrazí aktuální naměřené údaje. USB/UART



Obr. 2.1: Blokové schéma

převodník se stará o komunikaci s PC a mikrokontroleru, zejména při načítání údajů uložených na SD kartě.

## 2.2 Napájení zařízení

Zařízení je primárně napájeno pomocí čtyř 1,2 V baterií typu AA a dále je možno zařízení připojit k napájecímu adaptéru, které má výstupní napětí 5 V maximálně však 6 V. Toto napětí zaručí, že nedojde k vážnějšímu poškození akumulátoru dalším nabíjením a následným přebíjením. K tomu lze využít i standardní nabíječky na telefony/tablety, USB výstup počítače nebo použít napájení pomocí solárních článků.

Aby nedošlo k poškození baterií, jejichž kapacita je 2100 mAh, nadměrným proudem, tak zařízení obsahuje proudový regulátor tvořený obvodem LM7805 2.2, který omezuje vstupní proud do zařízení na přibližně 1 A a pomocí vztahu

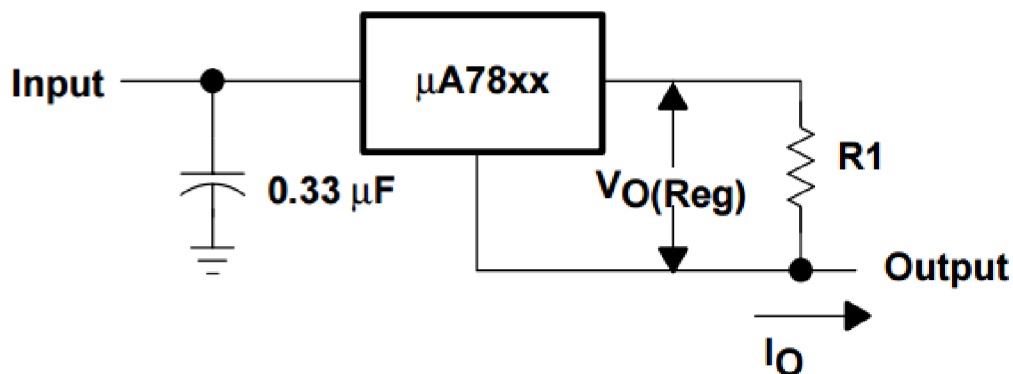
$$I_0 = \frac{U_0}{R_1} + I_B, \quad (2.1)$$

kde je

- $I_0$  výstupní proud [A],
- $U_0$  výstupní napětí [V],
- $R_1$  nastavovací odpor [ $\Omega$ ],
- $I_B$  rozdílový proud [A],

dopočítáme potřebný odpor  $R_1$ ,

$$R_1 = \frac{5}{(1 - 0,008)} = 5,04\Omega \rightarrow 5,6\Omega$$

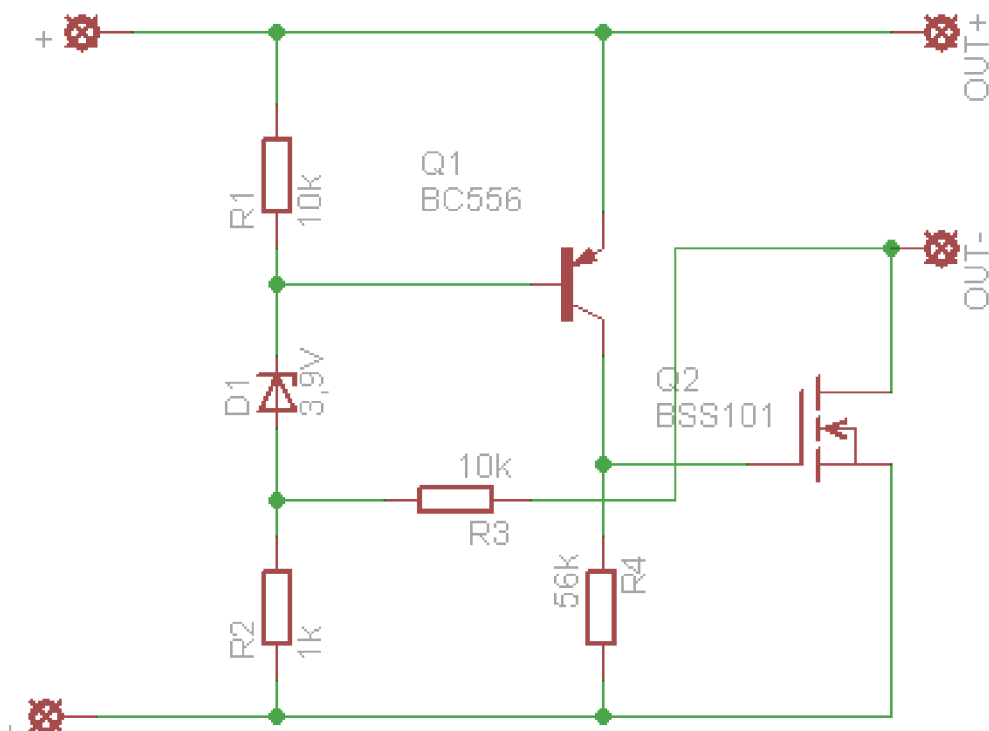


Obr. 2.2: Regulátor proudu[18]

a výsledný vstupní proud tedy bude

$$I_0 = \frac{U_0}{R_1} + I_B = \frac{5}{5,6} + 0,008 = 0,9 \text{ A.}$$

Dále pak zařízení obsahuje odpojovač baterie 2.3, který ochrání baterie proti nadměrnému vybití. Po připojení napájení odpojovač při dosažení napětí zenerovy diody (3,9 V) sepne bipolární tranzistor a tím i druhý unipolární tranzistor a dochází k dobíjení baterie a zároveň napájení zařízení. Skokové vypnutí/zapnutí odpojovače zajišťuje rezistor  $R3$ . [21]



Obr. 2.3: Odpojovač baterie

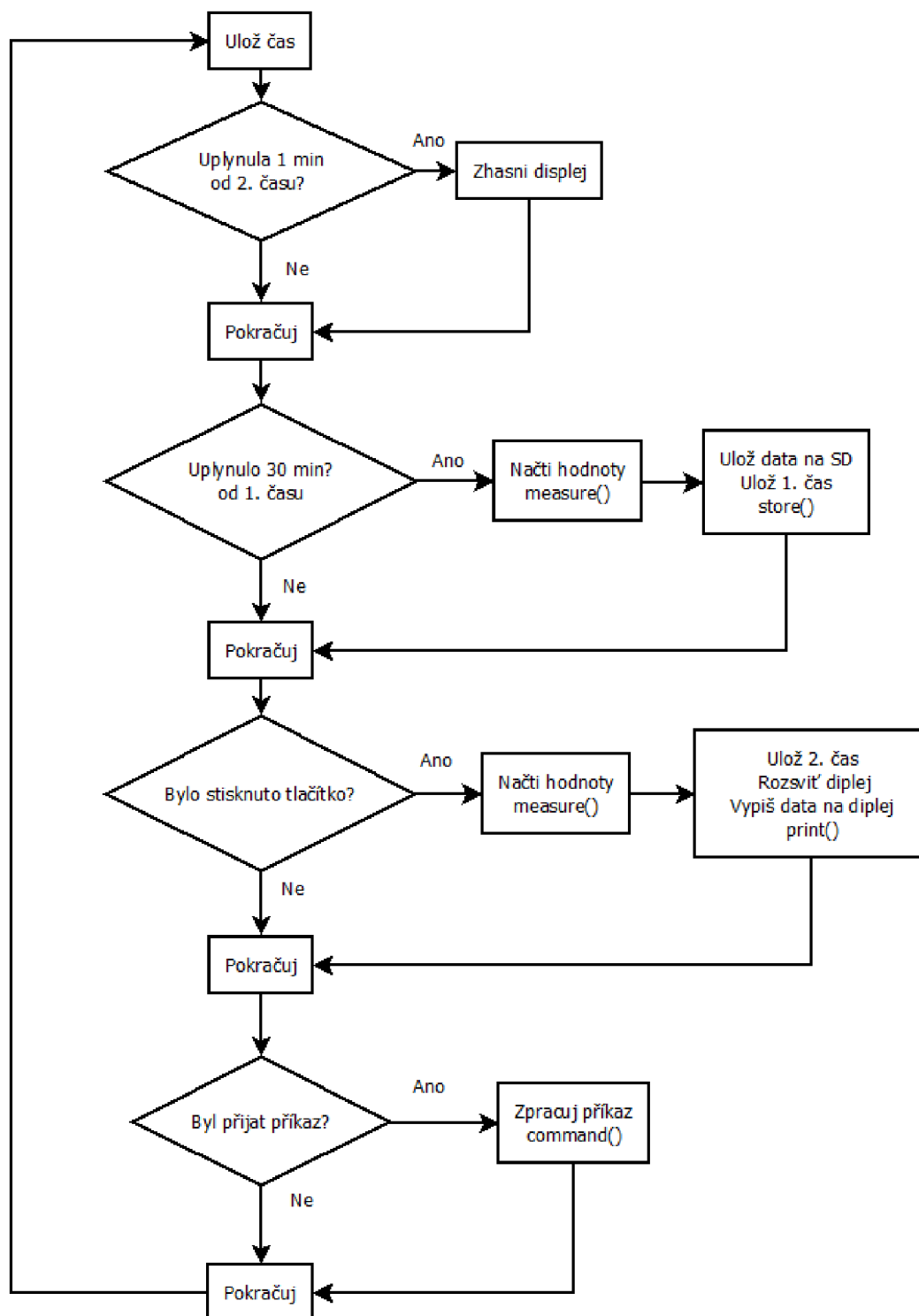
## 2.3 Programové vybavení

Programové vybavení zařízení je vytvořeno ve vývojovém prostředí Arduino IDE. Celý zdrojový kód, lze prohlédnout v příloze. Výsledný program je nahrán pomocí vývojového kitu Arduino Uno do zařízení pomocí ISP rozhraní. Struktura programu je zobrazena na vývojovém diagramu obr. 2.4.

### 2.3.1 Hlavní smyčka

Po inicializaci jednotlivých zařízení a modulů se spustí nekonečná smyčka, ve které neustále testují čtyři stavy.





Obr. 2.4: Vývojový diagram

V prvním stavu kontroluje, zda po stisku uživatelského tlačítka uběhla jedna minuta. Pakliže ano, uživatel již aktivně nevyužívá zařízení a mikrokontroler vypne podsvícení displeje. Ve druhém stavu kontroluje uplynutí časového intervalu

(30 minut) ukládání dat. Mikrokontroler porovnává uložený čas ve funkci *store()*; s aktuálním časem a každých 30 minut pak mikrokontroler uloží data na SD kartu a změní čas posledního uložení. Ve třetím stavu kontroluje stisk uživatelského tlačítka, kdy po stisknutí tlačítka načte data ze senzorů a zobrazí je na displeji. Po každém stisku tlačítka se mění údaje na displeji, dle zvoleného senzoru a uloží se čas rozsvícení displeje, který je sledován ve stavu 1. V posledním stavu mikrokontroler zjišťuje, zda nebylo navázáno spojení s počítačem a pokud ano, zpracuje přijatý příkaz.

Ukázka kódu:

```
void loop() {
  currentMillis = millis();
  if (currentMillis - previousLogMillis >= logInterval) {
    store();}
  if (currentMillis - previousLightMillis >= lightInterval) {
    analogWrite(light, 0);
    lcd.clear();}
  int btn = analogRead(A0);
  long btnPushed;
  if (btn > 800)
    btn_push = false;
  else {
    if (!btn_push) {
      analogWrite(light, 128);
      previousLightMillis = millis();
      btnPushed = millis();
      index = ++index % 5;
      measure();
      print();
      btn_push = true;}
    if(currentMillis - btnPushed >= 3000 && btn_push){
      sensor_data[index][1] = 100;
      sensor_data[index][2] = -100;}}
  while (Serial.available()) {
    command();}}
```

### 2.3.2 Měření a převod hodnot z čidel

Jelikož jsou použita analogová čidla, je potřeba převést napětí na stupně Celsia případně na procenta relativní vlhkosti. ATmega328 disponuje 10-bit A/D převodníkem (1024 vzorků), na jednotlivé převodníky jsou připojena čidla, a při referenčním napětí 4,8 V je velikost jednoho vzorku cca 4,688 mV. [12][13][19][20]

U čidla teploty LM335 připadá výstupní napětí 10 mV/K, získaná hodnota teploty je v Kelvinech a odečtením  $-273,15^{\circ}\text{C}$  získáme výslednou hodnotu ve stupních Celsia. Zapojení čidla dle 2.5.

Výsledný odpor  $R_1$  musí zajistit aby čidlem protékal maximální proud 2 mA při teplotě  $25^{\circ}\text{C}$  (298,13 K) a referenčním napětí 4,8 V. Napětí na výstupu čidla při teplotě  $25^{\circ}\text{C}$  musí být

$$298,13\text{ K} * 10\text{ mV/K} = 2,98\text{ V},$$

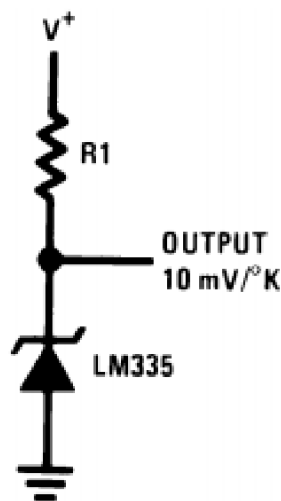
dále pak úbytek napětí na čidle

$$4,8\text{ V} - 2,98\text{ V} = 1,82\text{ V},$$

čímž dostáváme výsledný odpor

$$R_1 = 1,82\text{ V}/0,002\text{ A} = 910\ \Omega \rightarrow 1\text{ k}\Omega.$$

Podle zapojení čidla vlhkosti 2.6 volíme odpor  $86\text{ k}\Omega$ . Od výstupního napětí z čidla vlhkosti po odečtení hodnoty offsetu 0,958 V, kdy připadá 30,68 mV na % relativní vlhkosti 2.7, získáme výslednou hodnotu vlhkosti při teplotě  $25^{\circ}\text{C}$ . Pro jiné



Obr. 2.5: Zapojení čidla LM335[12]

teploty musíme vlhkost kompenzovat a komu využijeme vztah:

$$RH_{true} = \frac{RH_{sensor}}{1,0546 - 0,00216T}, \quad (2.2)$$

kde je

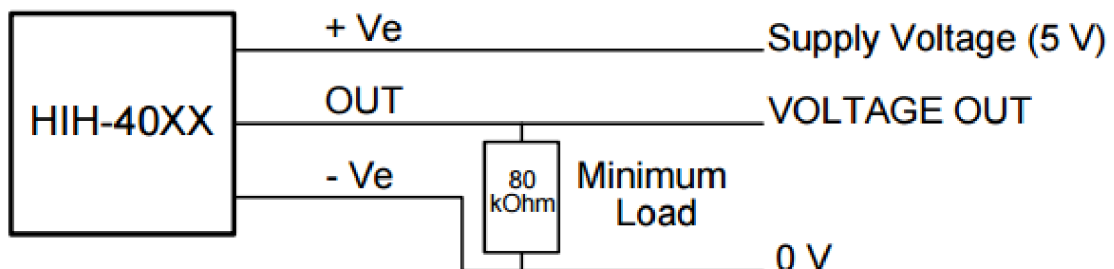
$RH_{true}$  výsledná relativní vlhkost [%],  
 $RH_{sensor}$  naměřené relativní vlhkost [%],  
 $T$  naměřená teplota [ $^{\circ}C$ ].

Ukázka kódu:

```
const float const_temp = 4800 / 1024.0 / 10;  
const float const_humm = 4.8 / 1024.0;  
void measure() {  
    float data = analogRead(A1);  
    data *= const_humm;  
    data -= 0.958;  
    data *= 30.680;  
    sensor_data[0][0] = data;  
    data = analogRead(A2);  
    data *= const_temp;  
    sensor_data[1][0] = data - 273.15; }  
    sensor_data[0][0] = sensor_data[0][0] /  
(1.0546-0.00216*sensor_data[1][0]); }
```

### 2.3.3 Ukládání hodnot

V daném intervalu, který je kontrolován v hlavní smyčce, mikrokontroler načte aktuální čas z RTC modulu a k němu připojí naměřené hodnoty z čidel, které jsou uloženy v lokální proměnné a výsledný řetězec uloží na SD kartu. Aby byl dodržen

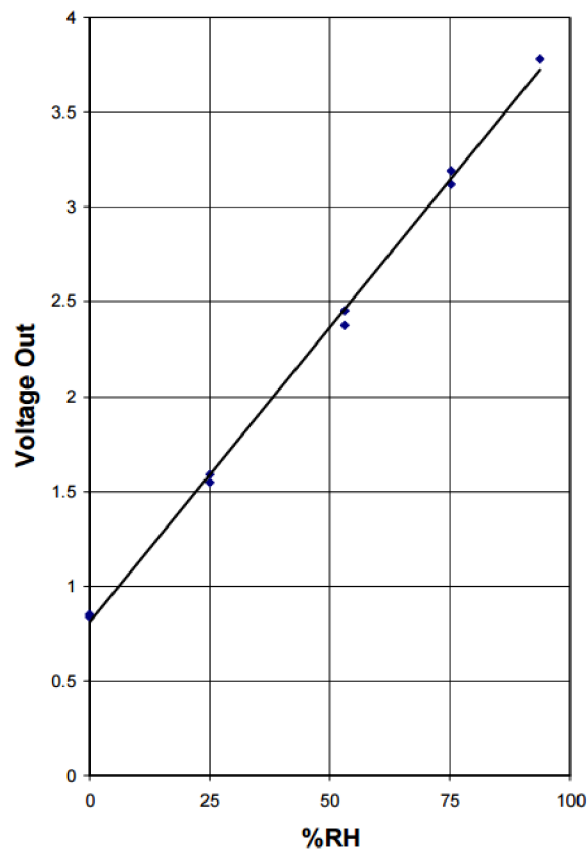


Obr. 2.6: Zapojení čidla HIH-4000-001[20]

interval ukládání hodnot musí být uložen aktuální čas běhu zařízení, tento čas je pak testován v hlavní smyčce.

Ukázka kódu:

```
void store() {
  previousLogMillis = millis();
  measure();
  String string = "";
  DateTime now = RTC.now();
  string+=String(now.day()+". "+String(now.month()+ " "
+String(now.hour()+":"+String(now.minute()+ " ");
  Serial.print(string);
  myFile = SD.open("data.txt", FILE_WRITE);
  if (myFile) {
    for (int i = 0; i < 5; i++) {
      for (int j = 0; j < 3; j++) {
```



Obr. 2.7: Graf závislosti Výstupního napětí na % relativní vlhkosti[13]

```

        string += String(sensor_data[i][j]);
        string += " ";}}
myFile.println(string);
myFile.close();
} else {
    Serial.println("Chyba pri zapisu do souboru na SD store()");}}

```

### 2.3.4 Komunikace s PC

Mikrokontroler neustále hlídá stav sériové linky, zda nebylo navázáno spojení a v případě navázaného spojení testuje přijatý příkaz. Mikrokontroler obsluhuje pouze příkazy odeslání uložených dat na SD kartě, případně mazání dat pro uvolnění místa aby se při dalším stahování dat zbytečně znovu nenačítala i stará stažená data.

Ukázka kódu:

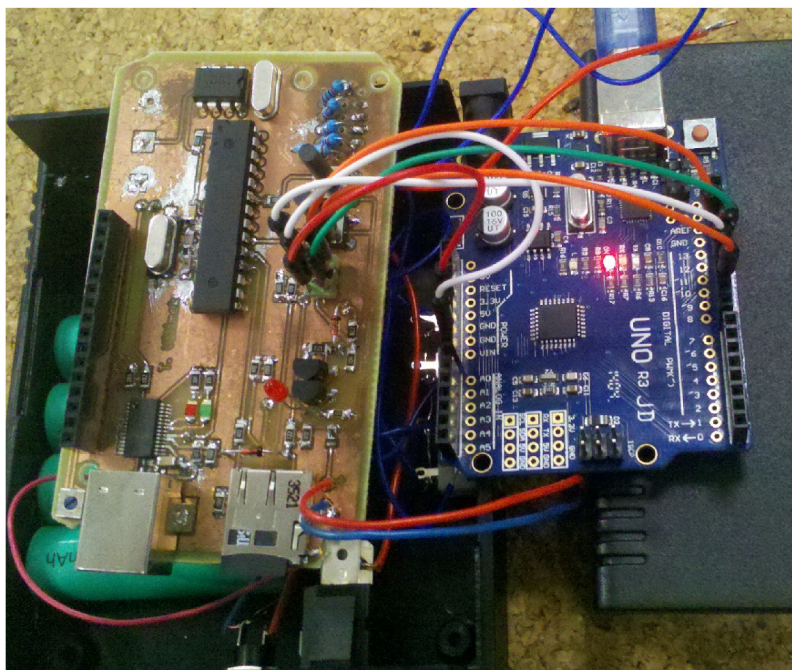
```

void command() {
    int val = Serial.parseInt();
    if (val == 1) {
        myFile = SD.open("data.txt");
        if (myFile) {
            while (myFile.available()) {
                Serial.write(myFile.read());}
            myFile.close();}
        else
            Serial.println("Nepodaril se otevrit soubor!");}
    else if (val == 2)
        SD.remove("data.txt");
    else
        Serial.println("Neplatny prikaz!");}

```

### 2.3.5 Nahrání programové obsluhy

K nahrání programové obsluhy je zapotřebí nejdříve do mikrokontroleru tzv. bootlo-ader, který je potřebný pro běh aplikací vytvořených pomocí Arduino IDE. K nahrání postačí vývojový kit Arduino Uno, který se připojí přes ISP rozhraní se zaří-zením 2.8. Kit musí být připojen k počítači a přes vývojové prostředí Arduino IDE je do zařízení nahráno vše potřebné. Po nahrání bootlo-aderu je možné programo-ovou obsluhu nahrát opět přes ISP rozhraní, případně pomocí počítače přes rozhraní USB. [22]



Obr. 2.8: Programování zařízení

## 2.4 Obsluha a konstrukce zařízení

Po otestování všech potřebných komponentů bylo vytvořeno samostatné zařízení. Jednotlivé komponenty byly osázeny na navrženou a vyrobenou DPS a deska vložena do vhodně upravené (otvory pro displej a konektory) konstrukční krabičky. V této krabičce jsou zabudovány baterie k napájení zařízení včetně jejich ochranných obvodů (odpojovač a proudový regulátor) a čidlo vlhkosti. Zařízení 2.9 dále disponuje dvěma tlačítky, jedním přepínačem, konektory jack k připojení teplotních čidel, napájecím konektorem a USB rozhraním k propojení zařízení s počítačem.

Přepínač slouží pouze k zapnutí/vypnutí zařízení, první tlačítko (select) slouží k zobrazení aktuálně naměřených hodnot na displeji 2.10 2.11 a dále k výběru požadovaného čidla, jehož naměřené hodnoty se mají zobrazit. Po stisknutí tlačítka výběru se zobrazí naměřená data, na které momentálně ukazuje proměnná *index* a zároveň se zobrazí i minimální a maximální naměřené hodnoty daného čidla. Po držení tohoto tlačítka po dobu 3s dojde k vynulování minimálních a maximálních naměřených hodnot příslušného čidla (po vynulování se zobrazí hodnoty dalšího čidla).

Druhé tlačítko (reset) slouží k případnému resetu zařízení při zamrznutí nebo pro nastavení začátku logování, kdy se nesmažou již naměřená data, která jsou již uložena na SD kartě, ale umožní ručně nastavit ukládání dat na konkrétní čas (například začátek ukládání dat přesně v poledne). Dále po stisknutí tlačítka reset se

vynulují minimálních a maximálních naměřené hodnoty čidel. Při vybití baterií pod napětí 3,7 V se rozsvítí červená signalizační LED dioda a po nepřipojení zařízení k přídatnému napájení dojde k odpojení baterií od zátěže aby nedošlo k jejich poškození vlivem nadměrného vybití.

Nastavení hodin zařízení na aktuální čas proběhne automaticky po připojení zařízení k počítači přes USB rozhraní a tento čas pak zařízení udržuje až do vybití druhé záložní baterie, která napájí pouze modul reálného času.

## 2.5 Obslužný PC program

Obslužný program obr. 2.12 byl vytvořen pomocí vývojového prostředí Microsoft Visual Studio 2012 a je založen na platformě Microsoft .NET Framework verze 4 a napsán v programovacím jazyce C/C++. Program disponuje grafickým rozhraní a umožňuje základní obsluhu zařízení.

Je určen pro načtení dat ze zařízení a po jejich získání, data uložit do vybraného souboru v daném umístění. Program podporuje více rychlostí pro komunikaci se zařízením, které v základu komunikuje rychlostí 19200 baudů, ale je možné ho změnit přehráním programové obsluhy zařízení. Dále pak umožňuje výběr portu na kterém bude komunikovat se zařízením (lze tak komunikovat s více zařízením). A v případě potřeby smazat data uložená v zařízení.



Obr. 2.9: Zkonstruované zařízení

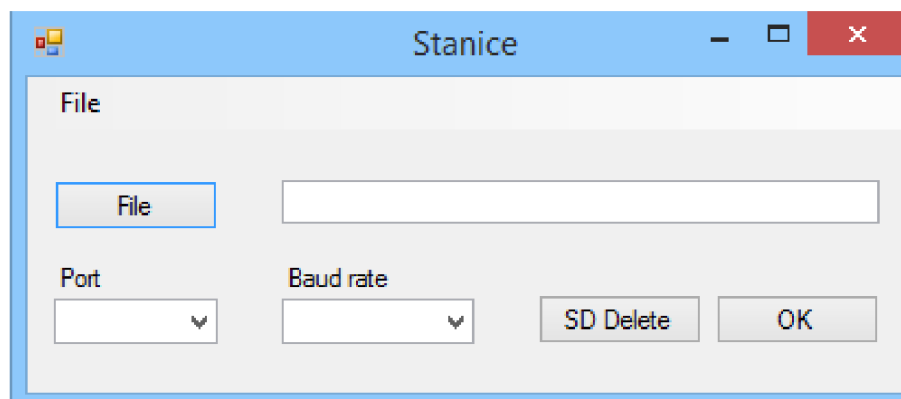




Obr. 2.10: Zobrazení vlhkosti



Obr. 2.11: Zobrazení teploty 1. čidla



Obr. 2.12: Obslužný program

Program stáhne data ze zařízení a vytvoří textový soubor na zvoleném umístění. K textovému souboru přepíše hlavičku a každý další řádek odpovídá jednomu záznamu.

Výsledný soubor může vypadat například takto:

```
---Date--- Humm% - Min - Max - Tmp1°C Min - Max Tmp2°C Min - Max Tmp3°C Min - Max Tmp4°C Min - Max
12.5. 15:10 33.53 33.53 33.53 25.68 25.68 25.68 24.70 24.70 24.70 31.54 31.54 31.54 26.17 26.17 26.17
12.5. 15:40 32.48 32.48 33.53 25.19 25.19 25.68 25.68 24.70 25.68 29.10 29.10 31.54 25.68 25.68 26.17
12.5. 16:10 28.88 28.88 33.53 26.17 25.19 26.17 24.70 24.70 25.68 29.10 29.10 31.54 26.65 25.68 26.65
12.5. 16:40 33.83 28.88 33.83 25.19 25.19 26.17 25.19 24.70 25.68 29.10 29.10 31.54 25.19 25.19 26.65
...
```

### 3 ZÁVĚR

V této práci jsou uvedeny a popsány jednotlivé hlavní komponenty zařízení, jejich způsoby komunikace, potřebné nástroje k vytvoření programové obsluhy mikrokontroleru a obslužného programu pro počítač. Dále jsou zde shrnuta problematika měření teploty a vlhkosti, výhody a nevýhody jednotlivých metod měření senzorů těchto veličin a blíže popsány a odůvodněny vybrané metody pro tento projekt.

V praktické části je uvedeno blokové schéma zařízení, návrh napájení zařízení, zapojení měřících čidel, získávání naměřených hodnot a jejich převod z elektrických veličin na odpovídající veličiny. Dále pak struktura programové obsluhy s ukázkami hlavních funkcí a jejich činností, popis a obsluha zařízení.

Zařízení se povedlo úspěšně navrhnout a odsimulovat na vývojovém kitu Arduino Uno a příslušných shieldech, kde byla ověřena funkčnost jednotlivých modulů a komponentů. Tyto komponenty a moduly byly pak ručně osázeny na navrženou a vyrobenou DPS. Následně bylo úspěšně zkonstruováno samostatné zařízení, které bylo testováno a vzniklé konstrukční a funkční problémy byly zcela vyřešeny.

Výsledné zařízení pak lze využít pro shromažďování údajů o příslušné místnosti, bytu či domu a jejich prezentaci, po příslušné úpravě, například na webu. Podle naměřených dat by poté šlo regulovat topení, klimatizaci, případně i odvětrávání, buď podle okamžité situace nebo s danou predikcí, která bude vycházet ze statistik nashromážděných údajů.

## LITERATURA

- [1] *Arduino Project*. [online]. [cit. 2014-12-08]. Dostupné z URL: <<http://arduino.cc/en/Main/Products>>.
- [2] *Microcontroller with 4/8/16/32K Bytes In-System Programmable Flash data-sheet*. [online]. [cit. 2014-12-08]. Dostupné z URL: <<http://www.atmel.com/Images/doc8161.pdf>>.
- [3] MATOUŠEK, D. *Práce s mikrokontroléry ATMEL AVR - ATmega16*. 1. vyd. Praha : BEN - technická literatura, 2006. 320 s. ISBN 80-7300-174-8.
- [4] SLINTÁK, S. *Arduino a USB*. [online]. [cit. 2014-12-08]. Dostupné z URL: <<http://uart.cz/394/arduino-a-usb/>>.
- [5] SLINTÁK, S. *Arduino a USB*. [online]. [cit. 2014-12-08]. Dostupné z URL: <<http://uart.cz/383/arduino-a-obvod-realneho-casu/>>.
- [6] *Future Technology Devices International Ltd. FT232R USB UART IC* [online]. [cit. 2014-12-08]. Dostupné z URL: <[http://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS\\_FT232R.pdf](http://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT232R.pdf)>.
- [7] *USB 2.0 Specification*. [online]. [cit. 2014-12-08]. Dostupné z URL: <[http://www.usb.org/developers/docs/usb20\\_docs/](http://www.usb.org/developers/docs/usb20_docs/)>.
- [8] *Teplota a její měření*. [online]. [cit. 2014-12-08]. Dostupné z URL: <<http://www.tzb-info.cz/3115-teplota-a-jeji-mereni>>.
- [9] SNOPEK, P. 59 s. *Přesný termostat* [online]. [cit. 2014-12-08]. Dostupné z URL: <[http://www.urel.feec.vutbr.cz/web\\_documents/ukazkove\\_prace/Snopek/Snopek.pdf](http://www.urel.feec.vutbr.cz/web_documents/ukazkove_prace/Snopek/Snopek.pdf)>.
- [10] *Vlhkost vzduchu a její měření* [online]. [cit. 2014-12-08]. Dostupné z URL: <<http://www.tzb-info.cz/3137-vlhkost-vzduchu-a-jeji-mereni>>.
- [11] *Měření vlhkosti vzduchu* [online]. [cit. 2014-12-08]. Dostupné z URL: <<http://www.micro.feld.cvut.cz/home/X34SES/cviceni/Navody%20na%20cviceni/07.Mereni%20vlhkosti.pdf>>.
- [12] *LM135/LM235/LM335, LM135A/LM235A/LM335A Precision Temperature Sensors*. [online]. [cit. 2014-12-08]. Dostupné z URL: <<http://www.ti.com/lit/ds/symlink/lm335.pdf>>.
- [13] *Humidity Sensors HIH-4000 Series*. [online]. [cit. 2014-12-08]. Dostupné z URL: <<http://www.phanderson.com/hih-4000.pdf>>.

- [14] *Externí sériové sběrnice SPI a I2C*. [online]. [cit. 2014-12-08]. Dostupné z URL: <<http://www.root.cz/clanky/externi-seriove-sbernice-spi-a-i2c/>>.
- [15] *Universální sériová sběrnice (USB)*. [online]. [cit. 2014-12-08]. Dostupné z URL: <<http://www.root.cz/clanky/universalni-seriova-sbernice-usb/#ic=serial-box&icc=text-title>>.
- [16] *LCM MODULE TC1602A-01T*. [online]. [cit. 2014-12-08]. Dostupné z URL: <<http://www.adafruit.com/datasheets/TC1602A-01T.pdf>>.
- [17] ČALOUN, P. *Hlasový výstup elektronického systému* [online]. [cit. 2014-12-08]. 30 s. Dostupné z URL: <[http://petrcaloun.sweb.cz/files/Project\\_Caloun\\_PIC\\_SD.pdf](http://petrcaloun.sweb.cz/files/Project_Caloun_PIC_SD.pdf)>.
- [18] *μA7800 SERIES POSITIVE-VOLTAGE REGULATORS* [online]. [cit. 2015-05-13]. Dostupné z URL: <<https://www.sparkfun.com/datasheets/Components/LM7805.pdf>>.
- [19] *The LM335 Temperature Sensor* [online]. [cit. 2013-05-138]. Dostupné z URL: <<http://web.mit.edu/rec/www/workshop/lm335.html>>.
- [20] *Humidity Sensors HIH-4000 Series* [online]. [cit. 2015-05-13]. Dostupné z URL: <<http://sensing.honeywell.com/honeywell-sensing-hih4000-series-product-sheet-009017-5-en.pdf?name=HIH-4000-001>>.
- [21] *Odpojovač zátěže při podpětí* [online]. [cit. 2015-05-13]. Dostupné z URL: <<http://www.zajic.cz/odpojovac/odpojovac.htm>>.
- [22] *Using an Arduino as an AVR ISP* [online]. [cit. 2015-05-13]. Dostupné z URL: <<http://www.arduino.cc/en/Tutorial/ArduinoISP>>.

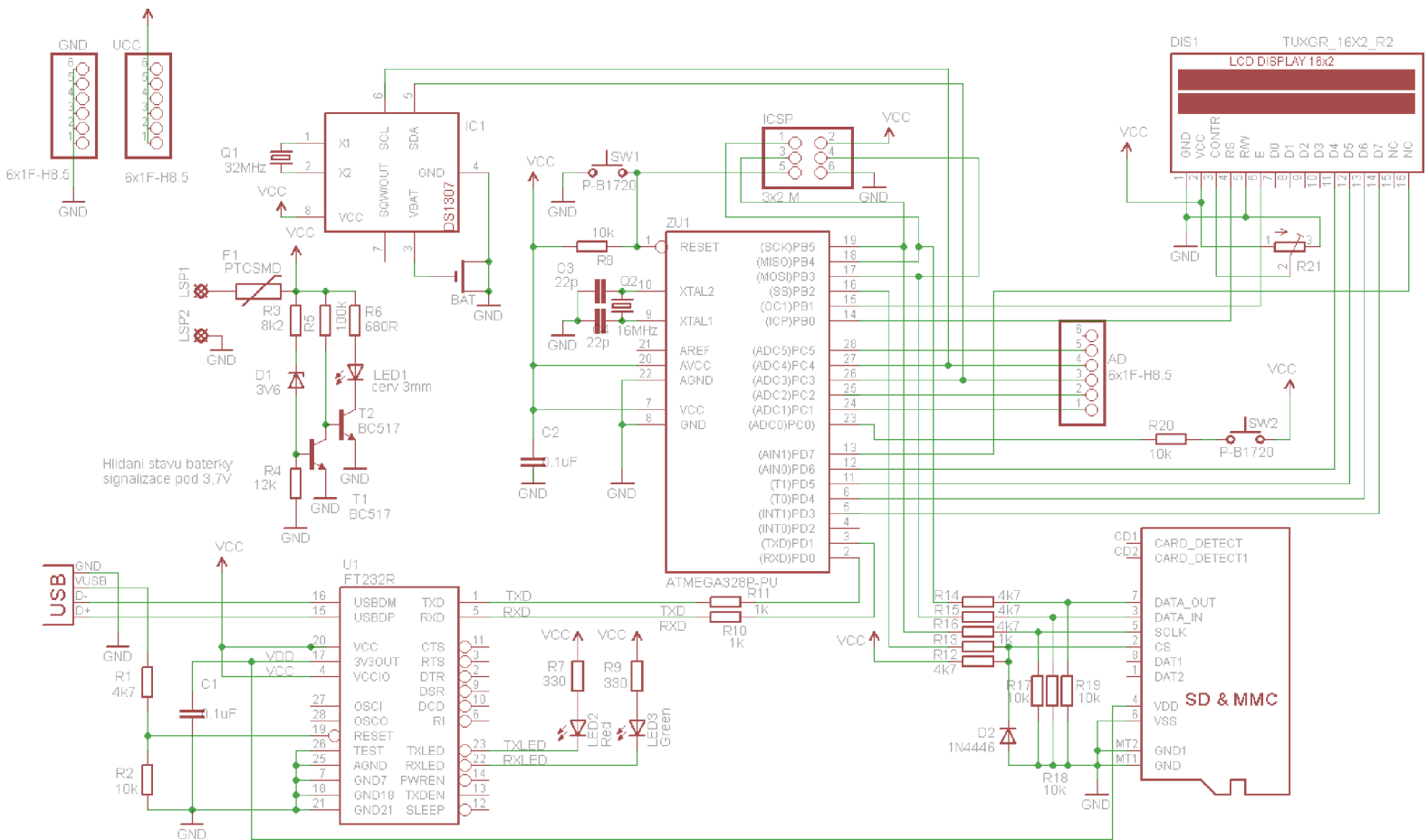
## SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

PWM	Pulsně šířková modulace – Pulse Width Modulation
USB	Universální sériová sběrnice – Universal Serial Bus
RAM	Paměť s přímým přístupem – Random Access Memory
ISP	Možnost programování za běhu – In System Programming
RISC	Redukovaná instrukční sada – Reduced Instruction Set Computing
CISC	Kompletní instrukční sada – Completed Instruction Set Computer
UART	Asynchronní sériové rozhraní – Universal Asynchronous Receiver Transmitter
LCD	Displej z tekutých krystalů – Liquid Crystal Display
ASCII	Americký standartní kód pro výměnu informací – American Standard Code for Information Interchange
SD	Paměťová karta typu SD – Secure Digital
EEPROM	Elektricky mazatelná paměť typu ROM – Electrically Erasable Programmable Read-Only Memory
A/D	Analogově/Digitální – Analog/Digital
SPI	Sériová periferní sběrnice – Serial Peripheral Interface
IDE	Integrované vývojové prostředí – Integrated Development Environment
RTC	Modul reálného času – Real Time Clock
ROM	Paměť pouze pro čtení – Read Only Memory
PCB	Printed Circuit Board – Deska plošných spojů
DPS	Printed Circuit Board – Deska plošných spojů

# SEZNAM PŘÍLOH

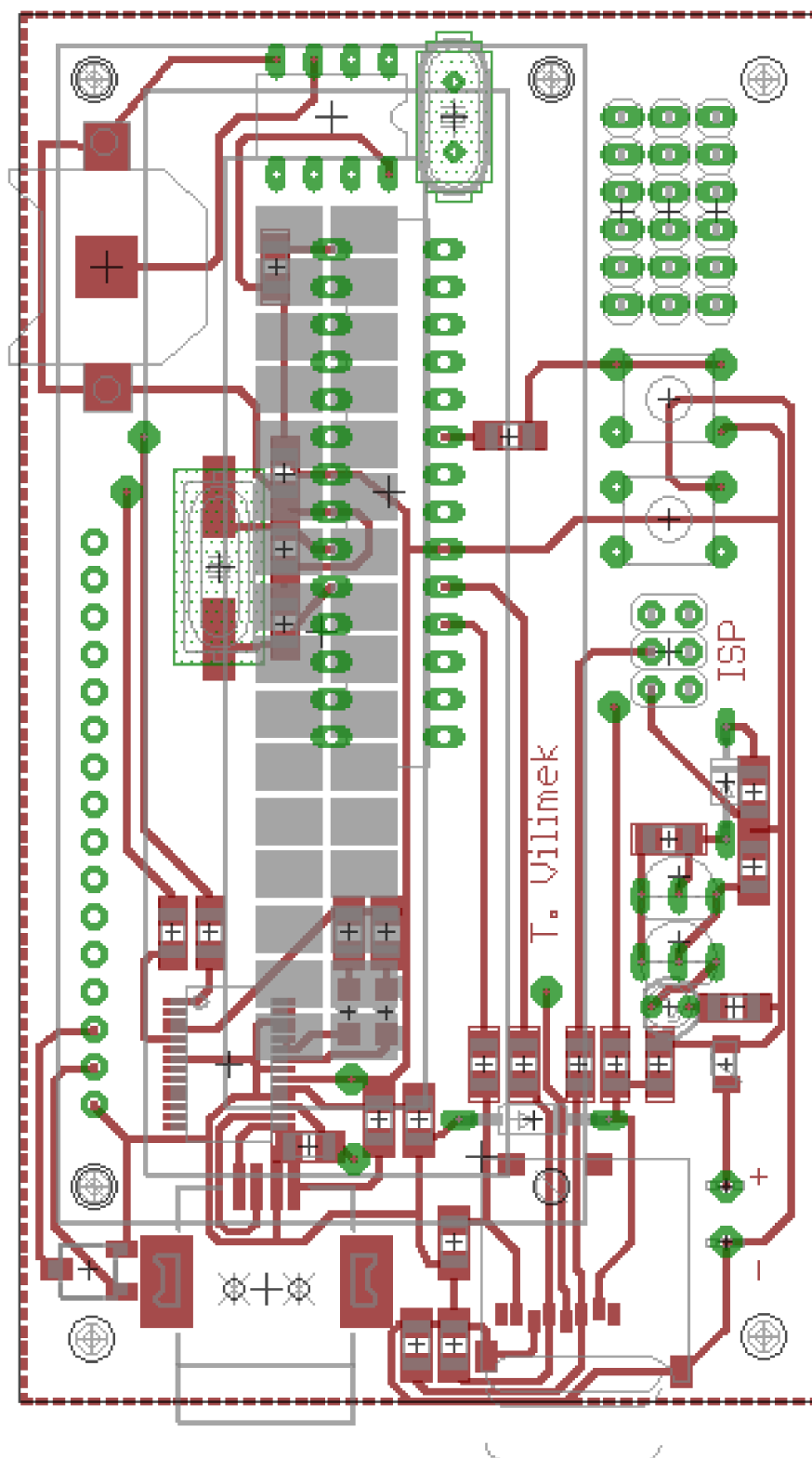
A Schéma zapojení	42
B Návrh plošného spoje	43

# A SCHÉMA ZAPOJENÍ



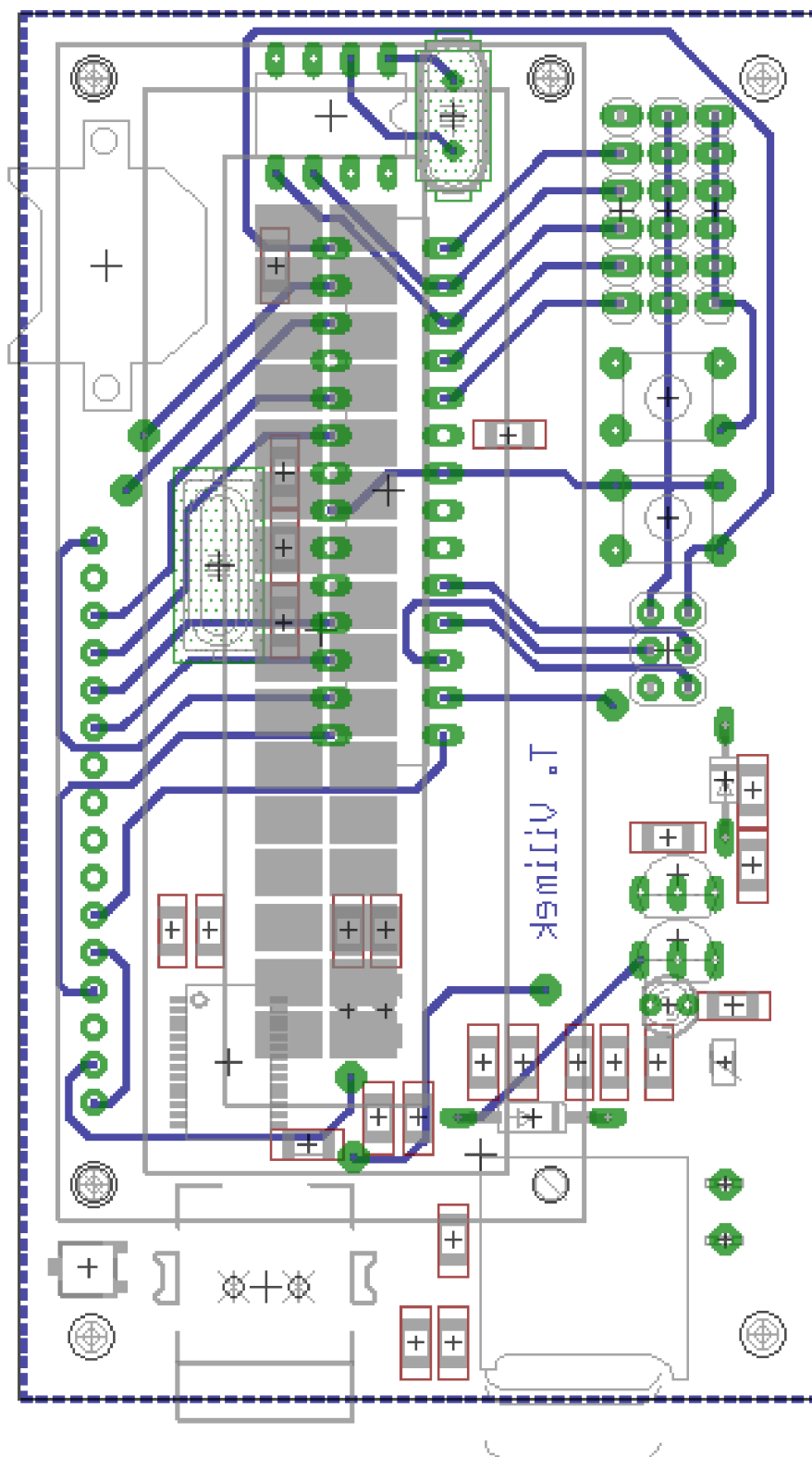
Obr. A.1: Schéma zapojení

## B NÁVRCH PLOŠNÉHO SPOJE



Obr. B.1: Návrh plošného spoje vrchní strana





Obr. B.2: Návrh plošného spoje spodní strana