

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

DIPLOMOVÁ PRÁCE

Brno, 2017

Bc. Martin Štábl



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

## ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY

DEPARTMENT OF CONTROL AND INSTRUMENTATION

## DATALOGGER PRO STAVEBNÍ KONSTRUKCE

DATALOGGER OR BUILDING CONSTRUCTIONS

### DIPLOMOVÁ PRÁCE

MASTER'S THESIS

### AUTOR PRÁCE

AUTHOR

Bc. Martin Štábl

### VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. Zdeněk Bradáč, Ph.D.

BRNO 2017

# Diplomová práce

magisterský navazující studijní obor **Kybernetika, automatizace a měření**

Ústav automatizace a měřicí techniky

**Student:** Bc. Martin Štábl

**ID:** 154888

**Ročník:** 2

**Akademický rok:** 2016/17

**NÁZEV TÉMATU:**

## Datalogger pro stavební konstrukce

### POKYNY PRO VYPRACOVÁNÍ:

Navrhněte a realizujte malý autonomní datalogger s nízkou spotřebou pro měření teploty, vlhkosti a dalších parametrů ve stavebních konstrukcích. Systém navrhněte tak, aby k němu bylo možné připojit několik měřicích čidel. Systém postavte na základě mikrokontrolérů, vybavte bateriovým napájením a GSM komunikací. Navrhněte a realizujte HW, oživte a otestujte. Vytvořte komplexní programové vybavení pro mikrokontroléry a pro nadřazenou centrální stanici. Demonstrujte správnou funkci.

1. Proved'te literární rešerši
2. Navrhněte koncepci systému a obvodové schéma
3. Realizujte DPS, osad'te a oživte
4. Vytvořte programové vybavení, které otestujte a demonst'rujte
5. Zhodno'te řešení a demonst'rujte jeho funkčnost

### DOPORUČENÁ LITERATURA:

Pavel Herout: Učebnice jazyka C, KOPP, 2004, IV. přepracované vydání, ISBN 80-7232-220-6

Dle pokynů vedoucího práce.

**Termín zadání:** 6.2.2017

**Termín odevzdání:** 15.5.2017

**Vedoucí práce:** doc. Ing. Zdeněk Bradáč, Ph.D.

**Konzultant:**

**doc. Ing. Václav Jirsík, CSc.**  
*předseda oborové rady*

### UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové 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

Tato práce se zabývá návrhem a realizací jednoúčelového autonomního zařízení pro sběr dat. V práci jsou uvedeny postupy, kterými se k návrhu přistupuje. Výběr senzorů vyhovujících úloze, pro kterou je určen. Pohled na zařízení v porovnání s obdobnými přístroji na trhu a jejich funkčnost. Základní koncepce navrhovaného dataloggeru. Práce obsahuje hardwarové a softwarové vybavení tohoto zařízení.

## KLÍČOVÁ SLOVA

Datalogger, sběr dat, senzory vlhkosti a teploty, mikrokontrolér

## ABSTRACT

This work deal with developing dedicated autonomous device for data collection. The work describes the approaches how to design that. Solve choosing of the sensor taking into his properties for this device. And regard on the comparison with similar devices on market according behavior and function. General concept of the suggested datalogger. The thesis contains hardware and software equipment of this device.

## KEYWORDS

Datalogger, data collection, humidity and temperature sensors, mikrocontroller

ŠTÁBL, Martin. *Datalogger pro stavební konstrukce*. Brno, 2017, 48 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav automatizace a měřicí techniky. Vedoucí práce: doc. Ing. Zdeněk Bradáč, CSc.

## PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Datalogger pro stavební konstrukce“ jsem vypracoval(a) samostatně pod vedením vedoucího diplomové 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 diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové 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 diplomové práce panu doc. Ing. Zdeňku Bradáčovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno .....

.....

podpis autora(-ky)

# OBSAH

<b>Úvod</b>	<b>9</b>
<b>1 Dataloggery</b>	<b>10</b>
1.1 Nabídka běžně vyráběných dataloggerů . . . . .	10
1.2 Funkce dataloggeru . . . . .	12
<b>2 Součásti pro použití</b>	<b>13</b>
2.1 Senzor teploty a vlhkosti . . . . .	13
2.2 Měření průtoku vzduchu . . . . .	14
2.2.1 Termický anemometr [13] . . . . .	15
2.2.2 Tranzistorový anemometr . . . . .	15
2.2.3 Wind sensor [15] . . . . .	16
2.3 Mikrokontrolér . . . . .	16
2.3.1 MSP430-G-2553 . . . . .	16
2.4 Komunikace . . . . .	18
<b>3 Koncept systému</b>	<b>20</b>
<b>4 Návrh zařízení</b>	<b>22</b>
4.1 Schéma . . . . .	22
4.2 Návrh desky plošných spojů . . . . .	23
<b>5 Hardwarové vybavení</b>	<b>25</b>
5.1 Komunikační modul . . . . .	26
5.2 Napájení . . . . .	26
<b>6 Programové vybavení</b>	<b>28</b>
6.1 I <sup>2</sup> C sběrnice . . . . .	28
6.1.1 Implementace I <sup>2</sup> C . . . . .	29
6.2 UART sběrnice . . . . .	29
6.3 SPI a mikroSD karta . . . . .	30
6.4 Hlavní program . . . . .	32
<b>7 Zhodnocení zařízení</b>	<b>34</b>
7.1 Cena . . . . .	34
7.2 Spotřeba . . . . .	34
7.3 Poznámky k dalším verzím . . . . .	35
<b>8 Závěr</b>	<b>36</b>

Literatura	37
Seznam symbolů, veličin a zkratk	39
Seznam příloh	40
A Dokumentace dataloggeru	41
B Příklady zdrojových kódů	45
C Obsah přiloženého CD	48



# SEZNAM OBRÁZKŮ

1.1	Srovnávané typy Dataloggerů, a.) Testo 174H [4], b.) Comet R3120 [3], c.) TFA 30.5022 [5], d.) GAR195 [5] . . . . .	11
2.1	Sensory vlhkosti a teploty, a.) Bosch BME280 [11], b.) Sensirion SHT20 [12], TI HDC1080 [6] . . . . .	13
2.2	Linearizovaný tranzistorový anemometr [14] . . . . .	15
2.3	Wind Sensor Rev. C od Modern device [15] . . . . .	16
2.4	Vývojová sada Launchpad [17] . . . . .	17
2.5	Mikrokontrolér MSP430-G-2553 [Foto: Herbert K.] . . . . .	17
2.6	Funkční schéma mikrokontroléru [1] . . . . .	18
2.7	blokové schéma SIM900 od firmy SIMcom . . . . .	19
3.1	Koncepční schéma navrhovaného dataloggeru . . . . .	20
4.1	Pinout a pohled na LaunchPad [17] . . . . .	22
4.2	Pohled na grafické rozvržení vodivých vrstev v DPS . . . . .	24
5.1	Pohled na osazenou desku dataloggeru . . . . .	25
5.2	GSM modul od Pandatron.cz [16] . . . . .	26
5.3	Napájecí modul s modulem LM2596 . . . . .	27
6.1	Průběh odesílání dat po I <sup>2</sup> C sběrnici (autor: Marcin Floryan) . . . . .	28
6.2	Zapojení dvou modulů UART . . . . .	30
6.3	Diagram úrovní komunikace mezi mikrokontrolérem a mikroSD kar- tou [19] . . . . .	30
6.4	Vývojový diagram hlavního programu dataloggeru . . . . .	33
A.1	Schéma zapojení dataloggeru . . . . .	42
A.2	Osazovací plán - rozmístění součástek na desce . . . . .	43
A.3	Vyrobená deska - pohled ze strany součástek . . . . .	44
A.4	Vyrobená deska - zadní strana . . . . .	44

# ÚVOD

Tato práce se věnuje návrhu autonomního zařízení pro sběr dat, také jinak dataloggeru. Tento datalogger je určen pro stavební konstrukce a je tedy kladen důraz na nízkou spotřebu, aby mohl pracovat delší dobu bez nutnosti kontroly či obsluhy. Předpokládáné je využití bateriového napájení pro lepší mobilitu zařízení a možnost umístit ho i do starších konstrukcí bez elektrické přípojky. Menší důraz kladu na cenovou dostupnost, aby mohlo tohle zařízení konkurovat dnešní nabídce.

Jelikož se jedná o jednoúčelové zařízení, je důležité správně zvolit veličiny, které bude schopno měřit. Volba těchto veličin musí být s ohledem na důležitost pro stavební konstrukce a směrodatnost pro operátora. Proto si stanovím základní veličiny, a zdůvodním jejich důležitost právě pro stavební konstrukce.

Následně prohlédneme nabídku procesorů, abychom mohli vybrat, který nám svými parametry vyhovuje pro řešení této úlohy a splnění základních požadavků stanovených v zadání. Zhodnotíme i možnosti programování tohoto procesoru, abychom se vyhnuli obtížím s drahými programovacími programy a složitostem, které jsou vzhledem k jedno-účelnosti zařízení nevhodné.

Dále si ukážeme některé dostupné senzory a zhodnotíme možnost využití v našem přístroji. Snímače nalezneme pro všechny veličiny, které jsme zvolili jako důležité pro naši úlohu.

# 1 DATALOGGERY

V této kapitole se podíváme na možnosti dnes vyráběných dataloggerů, jejich cenovou relaci. Vyberu pár dataloggerů, abych mohl stanovit průměrné parametry takovýchto zařízení a měli jsme tak představu, co se očekává od dataloggeru pro stavební konstrukce. Závěrem této kapitoly bych shrnul vlastnosti, které bych chtěl realizovat na mém zařízení.

## 1.1 Nabídka běžně vyráběných dataloggerů

Dnešní běžně vyráběné zařízení pro sběr dat, se často specializují na průmyslové využití např. farmaceutický, potravinový, chemický průmysl atd. Levnější jedno-účelová zařízení, které mám zadáno jsou spíše výjimkou a sériové se moc nevyrábí. Kvůli využití těchto zařízení právě v farmaceutických firmách jsou na tyto zařízení kladený nároky na různé certifikace a přesnosti. Právě proto často neřeší úlohu jednoduchosti bez nutnosti velké přesnosti, což u stavebních konstrukcí můžeme tolerovat. Zde dávám porovnání nejpodobnějších dataloggerů mému zadání. Výčet parametrů a přesností je deklarován výrobcí. Uvádím jen pár nejdůležitějších parametrů jako je rozsah a přesnost měření, kapacita paměti, typ baterie. Navíc uvádím základní funkce těchto zařízení.

### TESTO 174H [4]

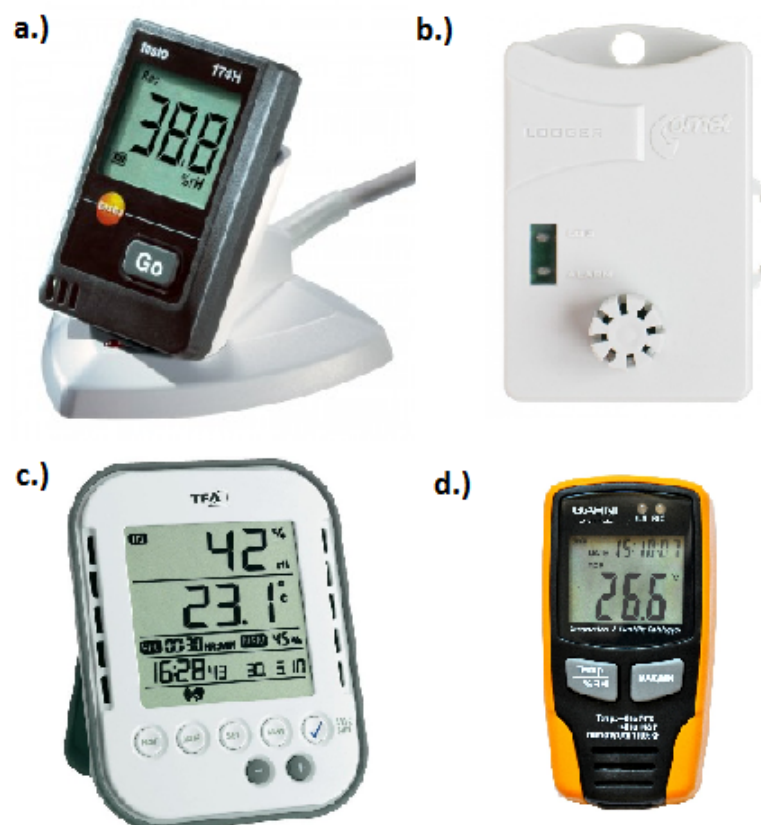
- Rozsah teploty:  $-20 \dots 70$  °C, přesnost  $\pm 0.5$  °C
- Rozsah vlhkosti:  $0 \dots 100$  % rv, přesnost 3 %
- Celková kapacita paměti: 16000 hodnot teploty (v necyklickém záznamu)
- Napájení: 2x CR2032 (lithiová knoflíková baterie)
- Cena: 5000 Kč

### COMET R3120 [3]

- Rozsah teploty:  $-30 \dots 80$  °C, přesnost  $\pm 0.4$  °C
- Rozsah vlhkosti:  $5 \dots 95$  % rv, přesnost  $\pm 2.5$  %
- Celková kapacita paměti: 32000 hodnot teploty (v necyklickém záznamu)
- USB, RS232, Ethernet nebo GSM modem (nutno dokoupit)
- Napájení: Lithiová baterie 3,6 V, rozměr AA
- Cena: 5000 Kč

### TFA 30.5022 [5]

- Teploměr:  $0 \dots 50$  °C, přesnost 1 °C
- Rozsah vlhkosti:  $0 \dots 100$  % rv, přesnost 3 %



Obr. 1.1: Srovnávané typy Dataloggerů, a.) Testo 174H [4], b.) Comet R3120 [3], c.) TFA 30.5022 [5], d.) GAR195 [5]

- Celková kapacita paměti: 50000 posledních měření
- Přístroj nelze propojit s PC
- Napájení: 3x baterie, typ AAA
- Cena: 2000 Kč

#### **GAR195 [5]**

- Teploměr: -40 ... 70 °C, přesnost 1 °C
- Rozsah vlhkosti: 0 ... 100 % rv, přesnost 3 %
- Celková kapacita paměti: 32700 posledních měření
- USB, nelze jinak rozšířit
- Napájení: lithiová baterie 3,6 V
- Cena: 2500 Kč

## 1.2 Funkce dataloggeru

Z přehledu běžně používaných dataloggerů zde vypíši funkce, na které by se nemělo zapomenout při realizaci mého přístroje. Jde o funkce, které nemají všechny dataloggery, avšak z hlediska funkčnosti a použitelnosti mého zařízení by bylo vhodné je použít. Dále i funkce které jsou u zařízení tohoto druhu brány jako samozřejmost.

### Základní funkce

- Hlídaní zaplnění paměti
- Alarm (oznámení překročení nastavených limitů)
- Detekce vybité baterie
- Detekce chyby (chyby senzorů, chyby programu atd.)
- Zachování dat při ztrátě energie (energeticky nezávislá paměť)
- Volitelný interval měření

### Pokročilé funkce

- Časový plán měření - nestejněměrné měření v čase, např. pokud nás zajímá více teplota přes noc.
- Cyklický záznam - možnost zapnutí funkce cyklického záznamu, neboli při zaplnění paměti, nová hodnota přepíše nejstarší (paměť typu fronta FIFO<sup>1</sup>).
- Stažení dat do PC přes USB port.
- Přepínatelné stupnice měření teploty (Celsiova °C nebo Fahrenheitova °F)
- Ochrana heslem proti cizímu zásahu - vhodné při použití v místech, která nejsou hlídána.
- Modulární komunikace - více možností při výběru komunikace (možnost změny GSM modulu, Ethernet, Bluetooth, WiFi), zadáním je komunikace pomocí GSM, ovšem rozšiřitelnost může být výhodou.
- Připojitelné sondy - možnost připojit sondy, pro měření dalších veličin či pro měření na více místech.

---

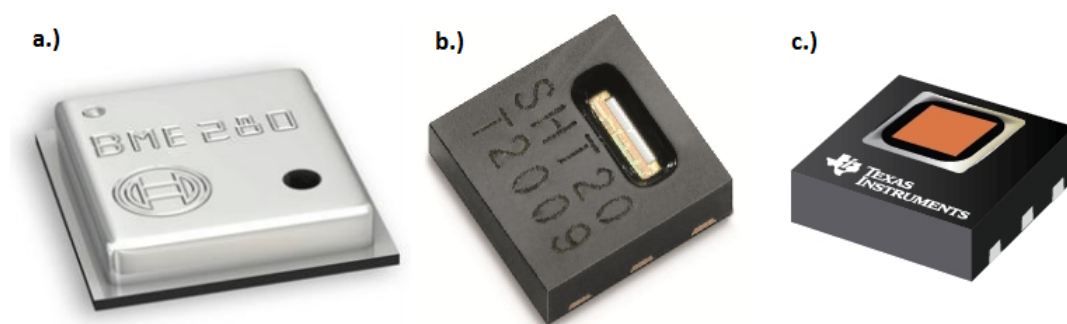
<sup>1</sup>first in - first out

## 2 SOUČÁSTI PRO POUŽITÍ

Přichází na řadu kapitola o druzích snímačů, které jsou v nabídce dnešních výrobců. V kapitole posoudíme rozsahy, přesnosti, ceny atd., podle potřeb posoudím možnost použití v navrhovaném zařízení. Dále si ukážeme další důležitá zapojení, které budou potřeba k realizaci některých funkcí zmíněných v předchozí kapitole.

### 2.1 Senzor teploty a vlhkosti

Zde uvádím porovnání pár snímačů, které v dnešní době vynikají v oboru měření teploty a vlhkosti. První dva snímače jsou téměř shodné, jedná se pouze o jinou verzi u nichž HDC1010 má navíc vyvedeny dva adresovací piny, což umožňuje připojení více těchto senzorů na jednu sběrnici.



Obr. 2.1: Sensory vlhkosti a teploty, a.) Bosch BME280 [11], b.) Sensirion SHT20 [12], TI HDC1080 [6]

#### Texas Instrument HDC1080 [6]

Rozsah teploty	-40 až 125
Přesnost měř. teploty	$\pm 0,2$ °C
Rozsah vlhkosti	0 až 100 %
Přesnost měř. vlhkosti	$\pm 2$ %
Nap. napětí	2,7 až 5,5 V
Spotřeba	1,3 $\mu A$
Rozhraní	I <sup>2</sup> C
Cena	85 Kč

### **Texas Instrument HDC1010 [7]**

Rozsah teploty	-40 až 85
Přesnost měř. teploty	$\pm 0,2$ °C
Rozsah vlhkosti	0 až 100 %
Přesnost měř. vlhkosti	$\pm 2$ %
Nap. napětí	2,7 až 5,5 V
Spotřeba	1,3 $\mu A$
Rozhraní	I <sup>2</sup> C s měnitelnou adresou
Cena	100 Kč

### **Bosch BME280 [11]**

Rozsah teploty	-40 až 80
Přesnost měř. teploty	$\pm 1$ °C
Rozsah vlhkosti	0 až 100 %
Přesnost měř. vlhkosti	$\pm 3$ %
Nap. napětí	1,7 až 3,6 V
Spotřeba	1,8 $\mu A$
Rozhraní	I <sup>2</sup> C a SPI
Cena	300 Kč

### **Sensirion SHT20 [12]**

Rozsah teploty	-40 až 125
Přesnost měř. teploty	$\pm 0,3$ °C
Rozsah vlhkosti	0 až 100 %
Přesnost měř. vlhkosti	$\pm 3$ %
Nap. napětí	2,1 až 3,6 V
Spotřeba	0,9 $\mu A$
Rozhraní	I <sup>2</sup> C
Cena	150 Kč

## **2.2 Měření průtoku vzduchu**

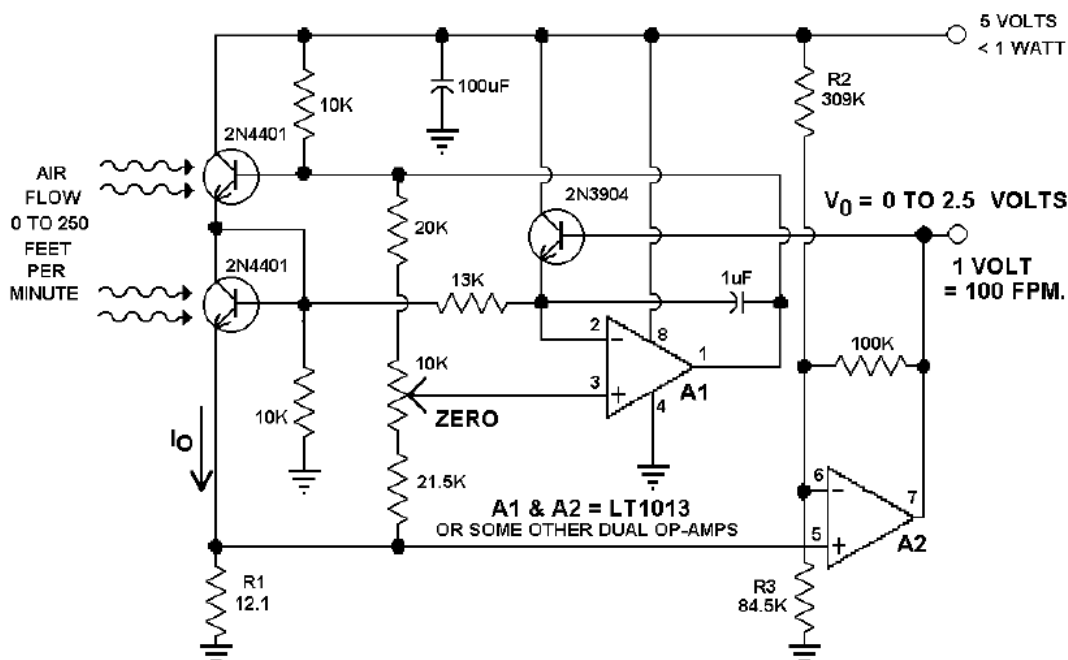
Pro ověření funkčnosti klimatizace či ventilace stavební konstrukce jako důležitého parametru uvažuji o použití jednoduchého senzoru průtoku. V tomto případě stačí pouze informativní ukazatel zda funguje či nefunguje ventilace, po případě proloženo pár stupni. Nejde tedy o přesné měření, ovšem stále se snažíme, aby naše zařízení nebylo energeticky úsporné a zároveň nenáročné na údržbu. Pokusím se najít nejvhodnější anemometr pro tuto úlohu.

### 2.2.1 Termický anemometr [13]

Nejjednodušší způsob, jak měřit rychlost proudění vzduchu je pravděpodobně termický neboli zchlazovací anemometr. Princip tohoto anemometru spočívá v zchlazení drátku (odporu) nahřátého procházejícím proudem. Proud vzduchu se následně dá určit právě z úbytku energie, který nastal zchlazením. Tento způsob neurčuje směr větru ale pouze sílu. Z důvodu zahřívání je náročnější na spotřebu. Tyto anemometry se používají spíše pro měření průtoku při konstantní teplotě měřené látky, tedy spíše pro kapaliny v uzavřených systémech.

### 2.2.2 Tranzistorový anemometr

Jednoduchým řešením by mohl být tranzistorový anemometr s vyhříváním tranzistorem. Využívá závislosti rychlosti větru a teplotní impedance tranzistoru, který představuje teplotní senzor. Je energeticky úspornější a robustnější než klasické "hot-wire" metody. Kompenzuje teplotu okolního prostředí. Jak jde vidět na obrázku 2.2, jeho výstupem je analogová hodnota, která se při napájení 5-ti V pohybuje v rozmezí 0 až 2,5 V. Tato hodnota je vhodná i pro mikroprocesory s napájením 3,3 V a není ani potřeba použití napěťového děliče.

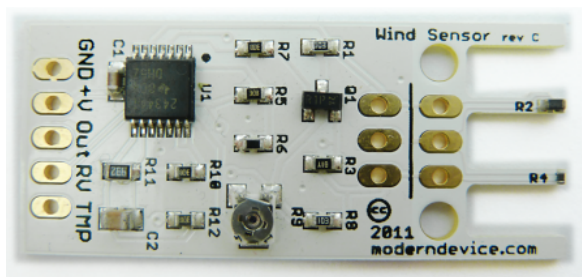


Obr. 2.2: Linearizovaný tranzistorový anemometr [14]



### 2.2.3 Wind sensor [15]

Jako poslední možnost bych mohl využít Wind sensor. Tenhle senzor je vyvinut právě pro malé projekty s velkou citlivostí, kde ostatní používané anemometry stojí až 6ti násobek ceny tohoto senzoru. Má analogový výstup a měří až do 60 *mph* (zhruba 95 *kmh*).



Obr. 2.3: Wind Sensor Rev. C od Modern device [15]

## 2.3 Mikrokontrolér

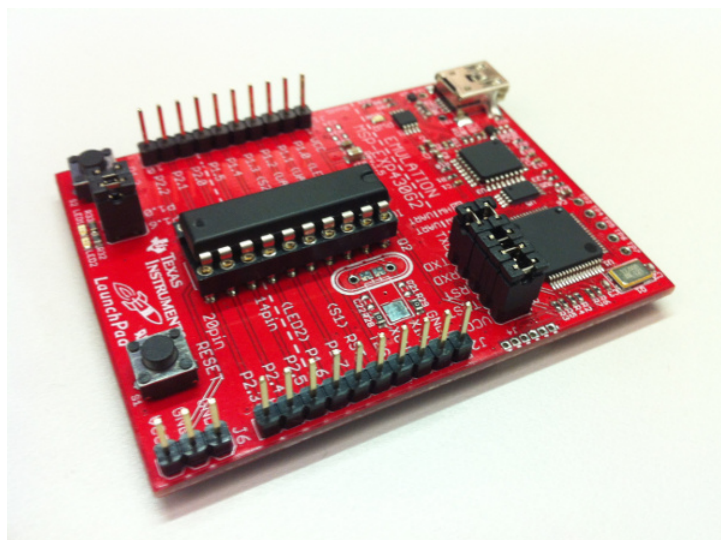
Nejdůležitější částí a vlastně mozkiem celého zařízení bude mikrokontrolér. Je těžké vybrat z množství na trhu nabízených mikrokontrolérů ten pravý. Já jsem s ohledem na spotřebu a dostupnost vybral mikrokontrolér společnosti Texas Instrument rodiny MSP430.

Tyto mikrokontroléry se vyznačují svou velice nízkou spotřebou v aktivním i standby modu. Je vyvinut pro použití v bateriově poháněných zařízeních. Kvůli své jednoduchosti je jejich typické využití právě pro jednoúčelové přístroje s jednoduchými čidly fyzikálních veličin. Vhodně zpracovává i analogové signály čidel, díky zabudovanému 10-ti bitovému ADC. Zvládá i zpracování dat a komunikaci s nadřazenými jednotkami. Pro komunikaci disponuje USI, SPI a I<sup>2</sup>C.

Předností této řady procesorů je hlavně dostupnost tzv. Launchpadu [2] přímo od TI jde vlastně o vývojový kit pro programování a debugging.

### 2.3.1 MSP430-G-2553

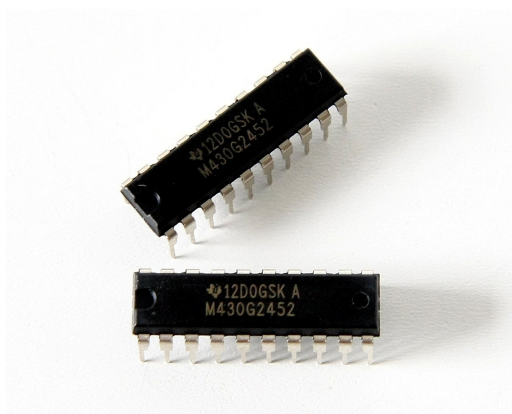
Z nabízených procesorů k téhle vývojové sadě jsem vybral jako dostačující mikrokontrolér MSP430G2553 od firmy Texas Instrument. Tento procesor by mohl v dostatečné míře splňovat všechny požadavky na zadané zařízení a zvládnout veškerou potřebnou obsluhu spojenou s funkcí dataloggeru.



Obr. 2.4: Vývojová sada Launchpad [17]

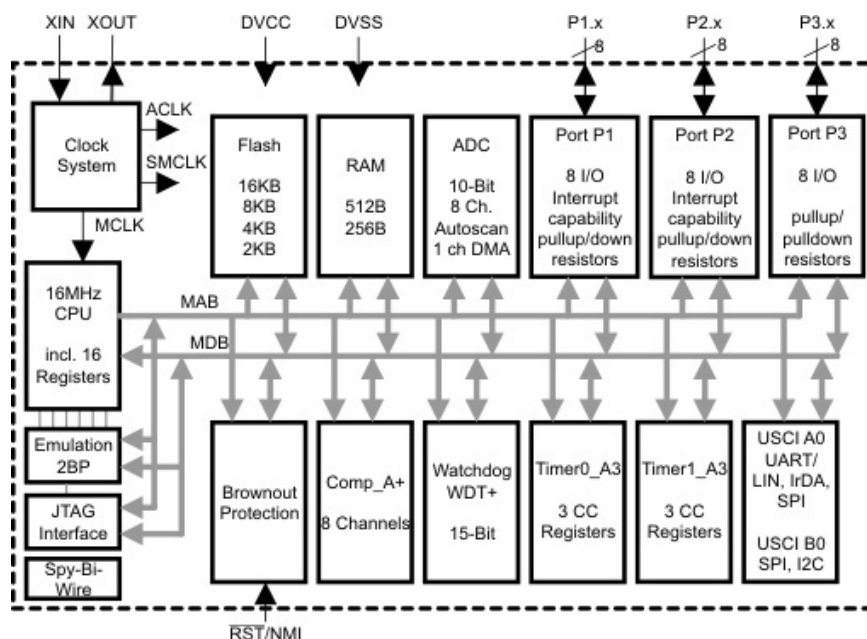
### Parametry

- 16-ti bitový, architektury RISC <sup>1</sup>
- Nízké napájecí napětí 1,8 až 3,6 V (vhodné pro bateriové použití)
- Nízká spotřeba aktivní režim 330  $\mu A$  a standby režim 0,7  $\mu A$
- 16 KB Flash paměti (nevolatilní paměť) a 512B RAM
- 2x 16-ti bitový časovač
- seriové rozhraní USI podporující UART, SPI a I<sup>2</sup>C
- 10-ti bitový ADC převodník a 8 analogových vstupů
- Pracovní teplota -40 to 85 °C



Obr. 2.5: Mikrokontrolér MSP430-G-2553 [Foto: Herbert K.]

<sup>1</sup>Reduced Instruction Set Computing - mikroprocesory s redukovanou instrukční sadou



Obr. 2.6: Funkční schéma mikrokontroléru [1]

## 2.4 Komunikace

Pro komunikaci bude v zařízení použit GSM modul, tak aby byl datalogger použitelný v místech bez nutnosti internetového připojení a aby nebyl závislý na dalších nadřazených zařízeních. GSM Standard poskytuje samostatnost mého zařízení, avšak také možnost ovládání pomocí mobilního telefonu skrze SMS. Zařízení by mohlo být dobře ovladatelné při neočekávaných situacích a o těchto situacích informovat uživatele prakticky kdekoliv, díky dnešnímu pokrytí mobilními sítěmi.

K propojení tohoto komunikačního modulu se používá sériové komunikace UART, kterou vybraný mikrokontrolér podporuje.

Jedním kritériem, které by mohlo ve výběru pomoci je to, jestli chceme, aby datalogger využíval pouze textové zprávy ke komunikaci s uživatelem nebo jestli budeme využívat i datové služby, tedy komunikaci internetovou.

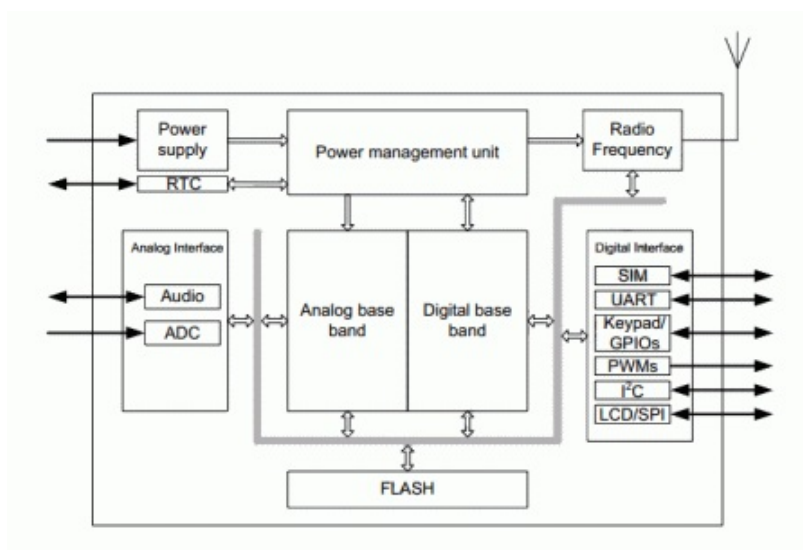
### SMS modul

S tímto modulem by bylo zařízení značně levnější, jak pořizovací cenou tak i provozně, jelikož by nepotřebovalo žádný datový tarif. Bylo by však nutné pro získání naměřených dat dojít k zařízení a stáhnout data do počítače, což by mohlo být v určitých ohledech nevhodné z pohledu uživatelského použití. Zařízení ovšem může mnohem lépe využít právě datové přenosy pro zobrazení měřených výsledků.

## GSM modul

Aby zařízení mohlo komunikovat s uživatelem i přes internetové připojení, bude potřeba použít GSM modul s podporou nejen sms zpráv. Pár jednoduchých GSM modulů s podporou datových služeb, aby kompaktnosti se hodily do mého projektu, se dá najít. Musí být však možné je napájet napětím do 3,6 V, abychom nemuseli používat vyšší napětí než je napětí procesoru.

Jeden modul, který jsem vybral, je modul splňující moje požadavky. Rozsah napájecího napětí má od 3,4 až 4,5 V, je SIM900 od firmy SIMcom. Má i nižší spotřebu a to okolo 1,5 mA v úsporném režimu.



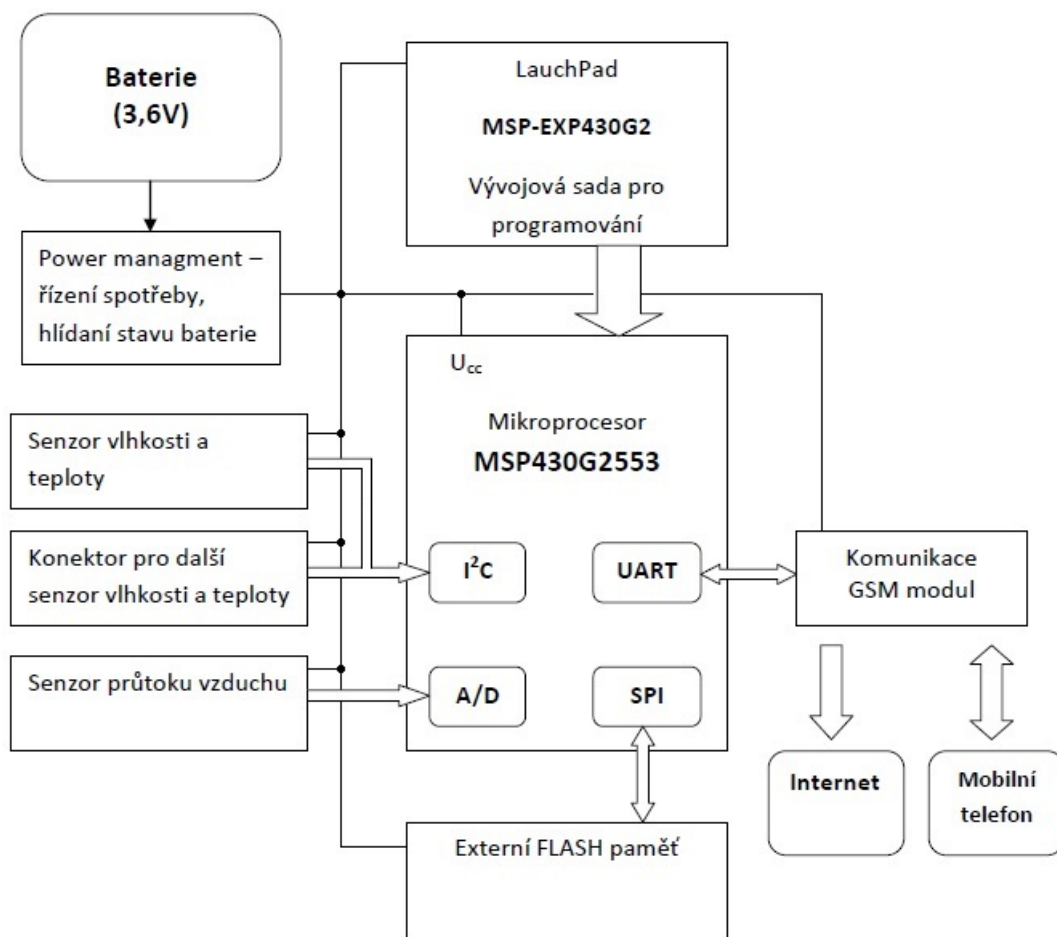
Obr. 2.7: blokové schéma SIM900 od firmy SIMcom

### 3 KONCEPT SYSTÉMU

Celé zařízení bude navrženo tak, aby mohlo pracovat autonomně bez zásahu obsluhy. Aby bylo dosaženo co nejdelší pracovní doby přístroje, jsou použity části s co nejmenší spotřebou. Díky úspornosti by zařízení mohlo pracovat dlouhou dobu samostatně bez zásahu, jen v krajních případech by mělo upozornit obsluhu pomocí SMS. Tyto krajní případy by mohly být např.

- Vybití baterie
- Překročení nastavených teplotních limitů
- Překročení nastavených limitů vlhkosti
- Naplnění paměti zařízení

Přes GSM by mělo být možné získat naměřená data a nejlépe je zobrazit přes webové rozhraní. Takové webové rozhraní by mohlo případně zobrazovat teplotní trendy a další statistické údaje. Typ komunikace by mohl být u našeho zařízení zaměnitelný z jiným než přes GSM např. WiFi, Ethernet.



Obr. 3.1: Konceptní schéma navrhovaného dataloggeru

Na vstupech budou umístěny senzor vlhkosti a teploty, dále pak průtokový senzor a vhodně bude přidána možnost připojit pomocí konektoru další senzor vlhkosti a teploty, pro měření v od sebe vzdálených místech konstrukce. Senzor průtoku vzduchu bude vhodné taky umístit mimo krabici samotného dataloggeru, aby se dalo umístit senzor přímo před klimatizaci či ventilaci.

Mikrokontrolér MSP430G, který bude jádrem celého dataloggeru bude sbírat data ze všech senzorů a ukládat je do nevolatilní paměti, z důvodu možné ztráty dat při vybití baterie. U bateriového zařízení bude potřeba použití paměti typu FLASH<sup>1</sup> pro ukládání naměřených dat bez obav ze smazání dat vybitím baterie nebo pro větší variabilitu řešení na microSD kartu připojenou skrze SPI rozhraní. Tyto paměti jsou nezávislé na napětí a jsou elektrický či softwarevě mazatelné. Procesor má integrováno 16 KB FLASH paměti pro program, to se ještě externí FLASH paměti pro možnost uložení nastavení či měřených hodnot.

Pro celé zařízení vybereme baterii typu Li-ion<sup>2</sup> s co nejvhodnější kapacitou (s ohledem na velikost a cenu) pro toto použití s nominálním napětím okolo 3,6 V. Tuto baterii budeme pomocí elektroniky hlídat co se týče stavu, abychom mohli upozornit uživatele před vybitím. Další části zařízení bude, odpojovat baterii v kritickém stavu, aby nedošlo k nadměrnému vybití akumulátoru. Při nadměrném vybití může dojít k nevratnému poškození baterie.

Li-ion baterie ovšem nebude vhodná při použití připojitelného GSM modulu. GSM modul má nárazově vyšší odběry, které by mohly baterii poškodit a také má vyšší nároky na stabilitu napětí. V tomto případě by bylo vhodnější použití větší olovnaté baterie s nominálním napětím 12V a napájením přes spínaný měnič stejnosměrného napětí na požadovanou hodnotu.

---

<sup>1</sup>Nevolatilní elektricky programovatelná paměť s libovolným přístupem

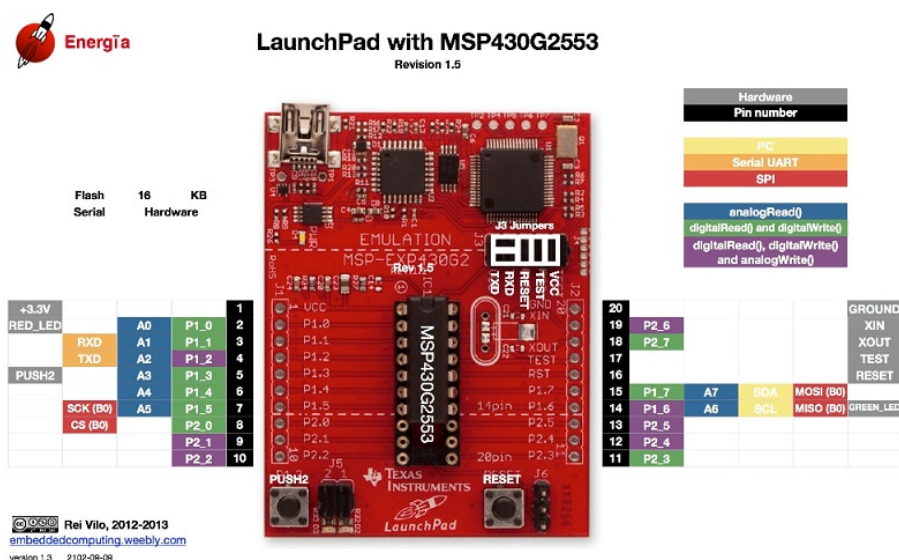
<sup>2</sup>Lithium-iontová baterie - nejpoužívanější typ akumulátoru

## 4 NÁVRH ZAŘÍZENÍ

Po návrhu koncepce systému přichází na řadu realizace reálného přístroje. K jeho sestrojení je potřeba vytvořit schéma v trasovacím programu, který je uzpůsoben k návrhu desek plošných spojů. V našem případě jsme využili nejpoužívanější program EAGLE firmy Autodesk [18], který nabízí omezenou studentskou verzi.

### 4.1 Schéma

Pro nakreslení schématu jsem vycházel z uspořádání vnějších pinů mikrokontroleru, pro který jsem navíc využil vývojovou sadu MSP-EXP430G2 Rev1.5 4.1 [17], umožňující jednoduché programování a debugging. V mém zařízení budu dle 3.1 vy-



Obr. 4.1: Pinout a pohled na LaunchPad [17]

užívat UART pro bezdrátovou komunikaci přes GSM modul, následně pro ukládání dat na SD kartu SPI sběrnici a I2C ke sběru dat ze všech senzorů. Mikroprocesor MSP430G2553 má ke komunikaci určeny dvě sériové rozhraní USCI<sup>1</sup>. Jedno rozhraní bude použito pro implementaci UART<sup>2</sup> a druhé pro SPI rozhraní k SD kartě. Pro I2C použiji softwarovou implementaci a tedy připojím všechna zařízení k dvěma volným pinům (GPIO<sup>3</sup>). Jde o rozšiřující EEPROM paměť, modul reálných hodin a senzor teploty. Další piny vybírám dle možnosti připojení k analogově-číslíkovému

<sup>1</sup>Universal serial communication interface - univerzální rozhraní pro sériovou komunikaci

<sup>2</sup>Universal asynchronous receiver and transmitter - zařízení pro asynchronní sériovou komunikaci

<sup>3</sup>General purpose input/output - vstup/výstup k běžnému použití

převodníku, tedy čtení analogové hodnoty hladiny nabití baterie a analogový výstup anemometru. Nesmí se zapomenout na napěťový dělič pro snížení zatížení vstupních pinů. K zjištění hladiny baterie jsem použil pro dělič potenciometr, aby bylo možné měnit dělicí poměr dle potřeby.

Ještě je potřeba přidat pull-up rezistory ke sběrnici I<sup>2</sup>C a adresovací přechody ke změně adresy paměti. Ještě jsem přidal pro možnou obměnu a univerzálnost v návrhu Jumperové propojky a výstupní piny pro připojení dalšího teplotního senzoru a anemometru. Konektory pro připojení napájení jsem zvolil klasické šroubovací svorky. K propojení desky s LaunchPadem a GSM modulem slouží dutinkové lišty.

V posledním kroku návrhu jsem přidal ke každému integrovanému obvodu, mezi napájení a zem, blokovací kondenzátor o velikosti 100 nF, a k vstupnímu konektoru větší elektrolytický kondenzátor 100  $\mu$ F (elektrolyt je vhodný díky svému nízkému sériovému odporu).

Takto vytvořené schéma najdete v příloze [?].

## 4.2 Návrh desky plošných spojů

Návrh samotného osazení desky je závislý na umístění jednotlivých komponent. Důležité je dodržet rozteče dutinkových lišt pro LaunchPad a pro GSM modul. Velikost desky jsem určil tak, aby nebyla širší než samotný LaunchPad a velikost byla co nejekonomičtější. Všechny blokovací kondenzátory se umísťují co nejblíže jednotlivým obvodům. Obezřetně je potřeba pohlídat i směry jednotlivých konektorů, aby byly snadno dostupné či nebyl úplně vyloučen přístup, například kdyby byl otočen konektor pro SD kartu.

Tato deska je dvouvrstvá (červená a modrá barva na obrázku 4.2) s rozlitou zemí. Rozměr desky je 56 × 77 mm a má v rozích díry pro distanční sloupky k umístění do krabičky.

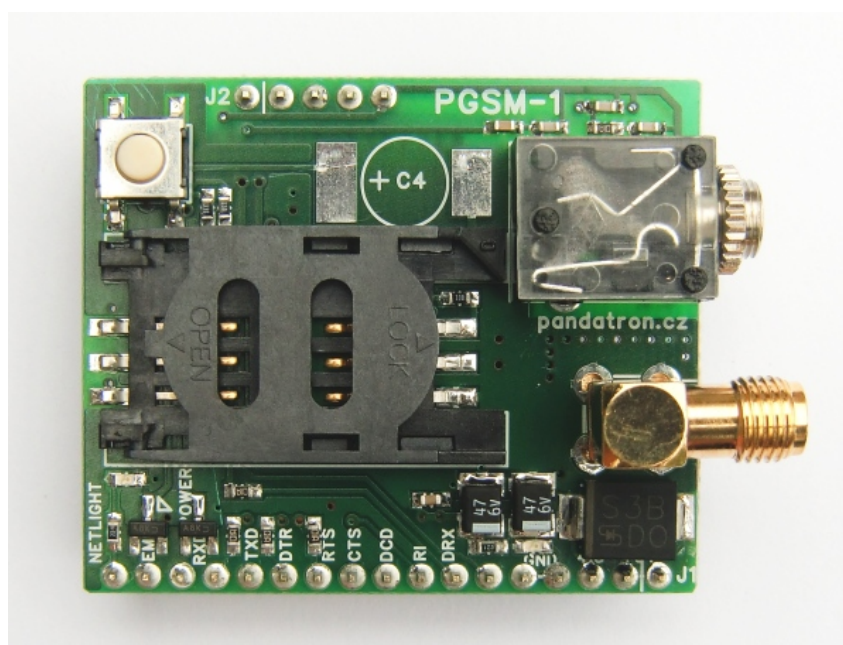






## 5.1 Komunikační modul

Pro GSM komunikaci mého zařízení jsem vybral GSM modul **PGSM-SIM900**, což je modul využívající čip SIM900 společnosti SimCOM [16]. Jde o čtyř-pásmový GSM modul podporující datové služby TCP/IP, HTTP, SMTP a FTP. Tím je vhodný pro různé druhy datové komunikace. Nabízí se TCP, UDP nebo odesílání na internet HTTP. Dále i podpora krátkých textových zpráv či e-mailu pomocí SMTP. Exceluje také svou nízkou spotřebou ve standby módu. Jednoduchostí a právě spotřebou je ten správný pro datalogger.

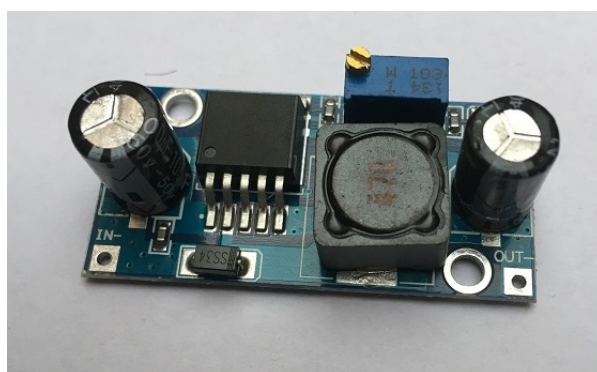


Obr. 5.2: GSM modul od Pandatron.cz [16]

Modul je říditelný pomocí UART sběrnice, probíhá v textové podobě pomocí AT příkazů v kódování ASCII [20].

## 5.2 Napájení

Pro napájení jsem zvolil stejnosměrný měnič napětí s vysokým dodávaným proudem, kvůli špičkovým proudům GSM modulu při inicializaci. Použitý modul LM2596, má nastavitelné výstupní napětí. I přes vysokou účinnost (více jak 90%), má vyšší odběr naprázdno, je ovšem elektronicky odpojitelný, tím jde spotřebu rapidně snížit.



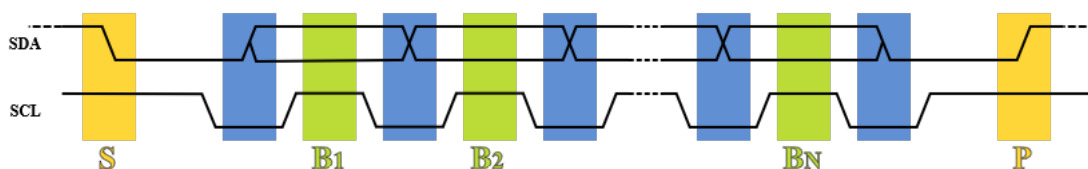
Obr. 5.3: Napájecí modul s modulem LM2596

## 6 PROGRAMOVÉ VYBAVENÍ

Program je navržen tak, aby byl přehledný a jednotlivé části byly použitelné pro realizaci i dalších projektů. Jednotlivé funkce mají tedy svou knihovnu, kde jsem zavedl makra a proměnné potřebné k řešení dané problematiky. Snažil jsem se vše pojmenovávat jasně, aby byl program čitelný i pro externího uživatele.

### 6.1 I<sup>2</sup>C sběrnice

I<sup>2</sup>C sběrnice je sériová sběrnice umožňující dvousměrný přenos dat pouze pomocí dvou vodičů. První SDA (*serial data*) pro data a druhý SCL (*serial clock*) pro hodiny. Tato sběrnice se používá hlavně k připojení nízko-rychlostních periférií jako jsou právě senzory, paměti, obvody reálného času atd. Sběrnice umožňuje připojení až 128 zařízení, ta ovšem musí mít každé svou jedinečnou adresu. Sběrnice neumožňuje plně duplexní přenos, v jeden okamžik nemůže vysílat více než jedno zařízení [10].



Obr. 6.1: Průběh odesílání dat po I<sup>2</sup>C sběrnici (autor: Marcin Floryan)

Klidový stav na sběrnici je v *log. 1*, inicializace komunikace probíhá start podmínkou, jak je vidět na obrázku 6.1 nastavením SDA do *log. 0* a potom SCL do *log. 0*. Následně všechna zařízení poslouchají prvních 8 bitů adresy (7mi bitová adresa a LSB určuje, zda je požadován zápis či čtení). Po těchto bitech musí přijít od slave zařízení acknowledge (ACK), který dává master stanici informaci o připravenosti zařízení, se kterým chce komunikovat. Takto po jednotlivých 8-mi bitech probíhá další komunikace.

Protože mnou použitý mikrokontrolér disponuje pouze dvěma USCI rozhraními pro sériovou komunikaci, které jsem při realizaci použil ke komunikaci pro SPI sběrnici a pro UART, musel jsem implementovat I<sup>2</sup>C softwarově na dvou volných GPIO pinech. Jelikož tato sběrnice a komunikace po ní není nějak rychlá, budou rychlosti změn na těchto pinech dostačující k její realizaci.

### 6.1.1 Implementace I<sup>2</sup>C

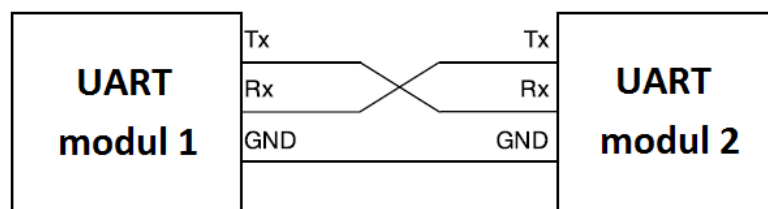
Jako první jsem k implementaci vytvořil základní funkce pro komunikaci po této sběrnici. Jde o start/stop podmínku, zaslání/příjem bytu dat a nakonec příjem/odeslání ACK bitu (viz příloha B.1). Z těchto základních funkcí jsem následně implementoval funkce rozšířené k správě a čtení z jednotlivých čipů. Tyto funkce mají návratovou hodnotu *0 SUCCESS* nebo *-1 FAILURE*. Všechny funkce vychází z komunikačních specifikací uvedených v katalogových listech jednotlivých čipů a implementovaných pomocí základních funkcí dle I<sup>2</sup>C specifikace.

#### Popis vybraných funkcí

- `int writeEEPROM(const int pointer, const int data);`  
Pomocí I<sup>2</sup>C sběrnice zapíše na adresu *pointer* v připojené paměti EEPROM data o velikosti 16 bitů.
- `int setAlarm(uchar acc);`  
Nastaví alarm na čipu DS3231, který vyvolá změnu na výstupu pinu INT. Ten je snímán na sestupnou hranu k probuzení mikrokontroléru z nízkopříkonového módu. Povolení alarmu je nastaveno ve funkci `setupRTC()`. Vstupními parametry jsou makra *SECONDS*, *MINUTES* a *HOURS*, podle toho v jakých jednotkách je požadován interval přerušení.
- `int resetAF();`  
Vynuluje alarm flag ve status registru DS3231 k umožnění dalšího přerušení po splnění podmínky k alarmu.
- `int getTime(uchar* min, uchar* hour, uchar* date, uchar* month);`  
Vrátí aktuální čas a datum dle DS3231 obvodu. Všechny hodnoty data a času jsou kódovány v BCD kódu (binárně kódované dekadické číslo). Nutno číst po jednotlivých znacích (4 bitech).
- `int getTemp_Hum(int address, long* temperature, long *humidity);`  
Z adresou definovaného zařízení (podporované jsou pouze HDC1080 a HDC1010 <0x40 - 0x43>) sejme aktuální hodnotu a převede jí výpočtem z digitálního rozlišení do setin stupňů (z důvodu náročnosti floating-point aritmetiky).

## 6.2 UART sběrnice

Jde o dvouvodičové sériové asynchronní zařízení, které zprostředkovává sériovou komunikaci. Je to často používané rozhraní mezi procesory a periférie. Běžně se nachází u všech mikrokontrolérů. Je velice rozšířené, neboť umožňuje plně duplexní přenos, jeden vodič pro příjem *Rx* a druhý pro odesílání *Tx*. U zapojení je důležité tedy překřížit vodiče, zapojit *Tx* do *Rx* a naopak, jako na obrázku 6.2.



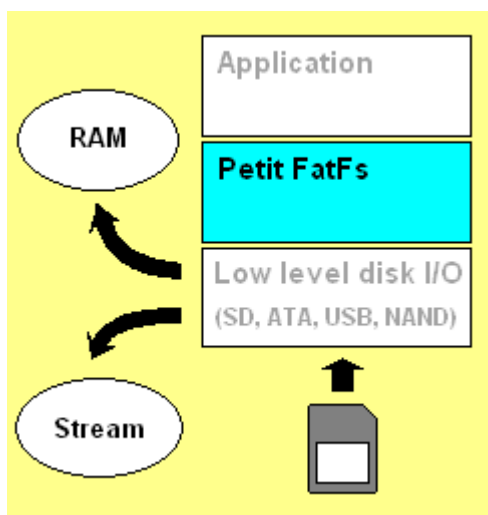
Obr. 6.2: Zapojení dvou modulů UART

Klidový stav na sběrnici je *log. 1*, jelikož se jedná o asynchronní komunikaci, pro zahájení se použije start podmínka. Před komunikací musí obě strany vědět konfiguraci komunikace, ty jsou dané specifikací. Obecně jde o rychlost přenosu, počet přenášených bitů najednou a počet a typ paritních bitů.

### 6.3 SPI a mikroSD karta

Naměřené údaje je možné ukládat na mikroSD kartu, pro kterou je umístěn konektor přímo na desce. SD karta představuje v tomto případě vysokokapacitní zařízení, které je nezávislé na napětí a je jednoduše připojitelné k počítači. SD karty využívají souborový systém FAT32, pro přístup k takovýmto souborům je použita volně dostupná knihovna Petite FatFs [19]. Jedná se o knihovnu pro malé mikroprocesory s omezenou pamětí a velikostí RAM. Součástí je i nízko-úrovňový přístup k disku.

Ke komunikaci přes SPI rozhraní je použita knihovna Výrobce mikrokontroléru.



Obr. 6.3: Diagram úrovní komunikace mezi mikrokontrolérem a mikroSD kartou [19]

Na obrázku 6.3 vidíme, jak funguje komunikace mezi mikrokontrolérem a soubory na SD kartě. Jako první je využita sběrnice SPI, která zprostředkovává připojení paměťové karty s procesorem. Je tedy komunikačním kanálem mikroprocesoru a nízkoúrovňové vstupně/výstupní vrstvy. Pro samotnou zprávu souboru je použita výše zmíněná knihovna Petite FatFs [19], ta obsahuje základní funkce pro zprávu souborového systému. Tato knihovna díky své malé velikosti a nenáročnosti má značné omezení, na které je třeba při programování brát zřetel. Jedná se hlavně o funkci pro zápis, autor ChaN na svých stránkách [19] uvádí tyto

- Nelze vytvořit soubor, využívá se existujících souborů.
- Funkce `pf_write` nemůže rozšířit velikost souboru.
- Zápis může začít nebo skončit pouze v jednotlivých sektorech. (Prázdné musí být doplněny nulami.)
- Zápis může probíhat i do souboru určeného pouze pro čtení.

Z tohoto důvodu musí být na kartě vytvořen prázdný soubor o specifické velikosti pro zápis naměřených dat. A musí být hlídány hranice sektorů, aby nedocházelo ke ztrátě dat přepisem či chybným uložením.

### Popis vybraných funkcí

- `void spi_initialize( void );`  
Inicializuje sériovou sběrnici SPI, nastaví pin CS (*chip select*, provede potřebné nastavení USCI rozhraní.
- `DSTATUS disk_initialize (void);`  
Inicializuje komunikaci s připojenou paměťovou kartou. Vrací status o úspěšnosti pokusu o připojení.
- `FRESULT pf_mount(FATFS *fs);`  
Připevní nový souborový systém pro organizaci disku procesorem. Vstupem je ukazatel na nový souborový systém uchovávající důležité informace k práci s diskem. Návrátová hodnota vypovídá o úspěšnosti či chybě.
- `FRESULT pf_open (const char *path);`  
Funkce k otevření existujícího souboru v aktuálním souborovém systému. Vstupním parametrem je celý název souboru i s příponou. Návrátová hodnota vypovídá o úspěšnosti či chybě.
- `FRESULT pf_write (const void* buff, WORD btw, WORD* bw);`  
Funkce pro zápis přímo do souboru. Jako parametry přijímá ukazatel na Buffer s daty k zápisu, požadovaný počet zapsaných znaků a počet znaků, které byly zapsány.

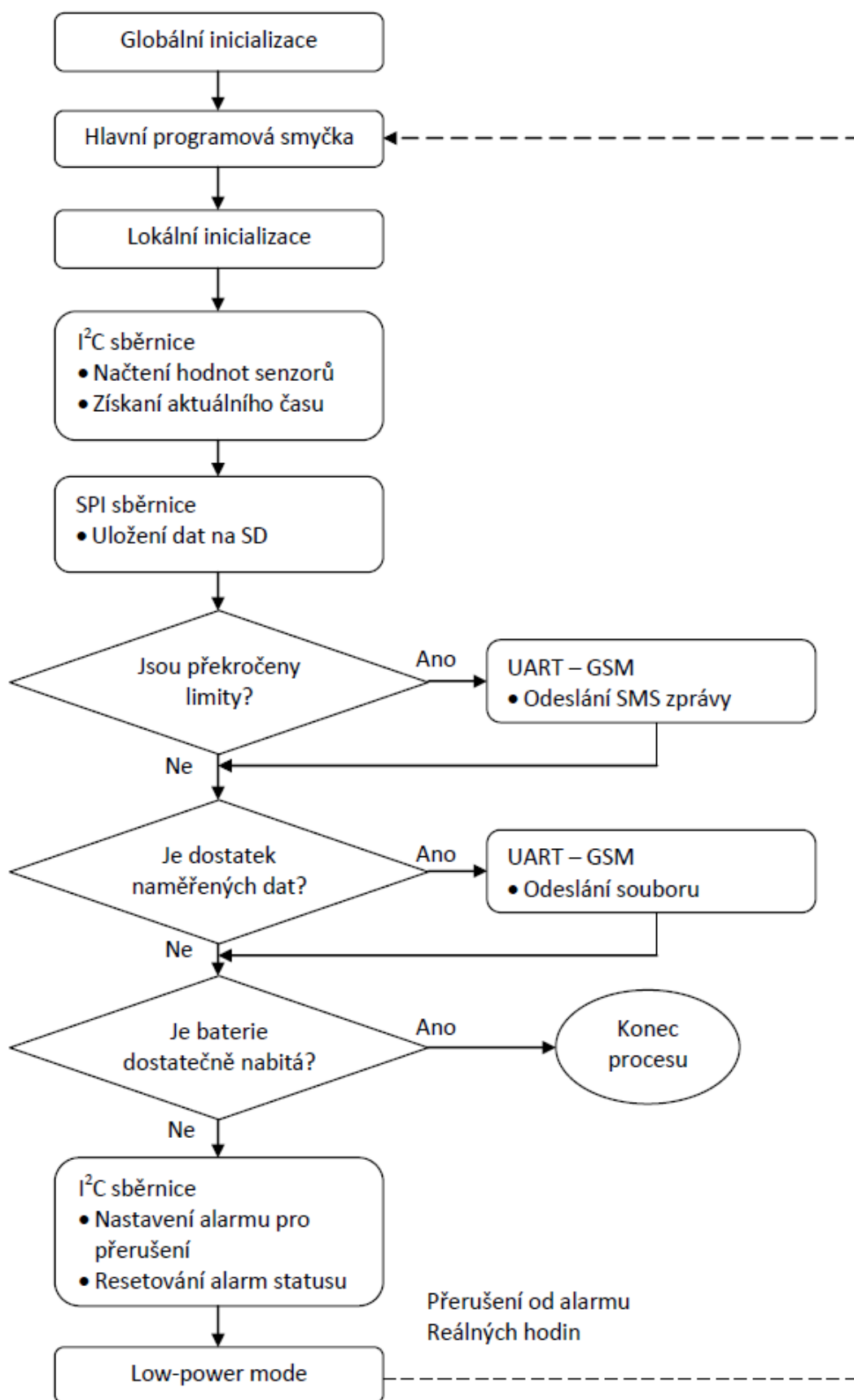


## 6.4 Hlavní program

Pro snadnější práci jsem si vytvořil vývojový diagram pro hlavní program mikrokontroléru. Začátek je věnován inicializaci globálních proměnných a nastavení registrů pro přerušení vstupy a výstupy. Následně jsem umístil nekonečnou smyčku `while`, která zajistí průběh programu. Jako první inicializuji lokální proměnné, které je třeba obnovit po probuzení CPU. Pak snímám měřené veličiny po sběrnici I<sup>2</sup>C, tj. načtu hodnoty vlhkosti teploty z HDC1080, čas z obvodu reálného času DS3231. Vše s přesností, která může být brána ještě jako směrodatná a pomocí SPI a knihoven k obsluze souborového systému FAT32 na SD kartu.

Po těchto krocích testuji podmínky nabitosti baterie, kontroly zda nejsou naplněny soubory k odeslání nebo překročení nestavených limitních hodnot teploty. Tyto podmínky podněcují mikrokontrolér k použití připojeného GSM komunikačního modulu. Na dané podmínky reaguje předepsaným způsobem, buďto odesláním SMS zprávy nebo odeslání souboru přes e-mail či na HTTP server. Pouze na podmínku vybití baterie navíc uspí procesor a ukončí tím proces sběru dat. GSM modul je ovládat pomocí hardwarového rozhraní USCI na kterém běží UART. Poté je ovládat GSM at příkazy [20].

Jako poslední nastavení přerušení od obvodu reálných hodin. Ty prvně nastavím a resetuji Flag značící proběhlé přerušení, následně uspím procesor do Modu 4 (nejúspornější) a nechám procesor čekat na definované přerušení k opakování měřící smyčky, které je vyvoláno buďto hodinami dle alarmu nebo tlačítkem na vývojové desce.



Obr. 6.4: Vývojový diagram hlavního programu dataloggeru

## 7 ZHODNOCENÍ ZAŘÍZENÍ

V této kapitole bych rád zhodnotil výsledné zařízení z na začátku vzpomenutých kritérií, pro přehled.

### 7.1 Cena

Cena na výrobu desky je její komponenty nepřevyšuje částku 3000,- Kč, takže požadavek ceny se podařilo splnit. Cena je bez baterie a krabičky. Cena by mohla být snížena výrobou více kusech.

Tab. 7.1: Seznam součástek osazených na desce

Název	Cena v Kč <sup>1</sup>
LaunchPad	300
MCU	70
GSM modul	900
Senzory	300
Windsensor	450
Drobný materiál	200
Napájecí modul	50
Výroba desky	600
Celkem	2970

### 7.2 Spotřeba

Spotřeba jak je vidět v tabulce 7.2, spotřeba je ovlivněna hlavně aktivním režimem ve kterém jsou aktivní právě energeticky náročné prvky. V režimu spánku nepřesahuje spotřebu 2 až 3 mA. Proto je potřeba hledět na omezení právě energetický náročných prvků. Například zasíláním naměřených dat až po nasbírání určitého množství. Při měření dvakrát denně a odesílání dat pouze při naplnění souboru jednou měsíčně by mohlo zařízení vydržet na baterii o kapacitě 4000 mAh až jeden rok. Požadavek na nízkou spotřebu je tedy taky splněn. Spotřeba napájecího modulu naprázdno je velice vysoká a neumožnila by dlouhodobou aplikaci na tomto zařízení. Ovšem tento modul jde elektronicky odpojit a tím spotřeba rapidně klesne. Ovšem procesor by musel být napájen zvlášť, což by nebyl z ohledem na jeho nízkou energetickou náročnost žádný problém. Tyto poznatky jsem uvedl v závěru práce 7.3.

Tab. 7.2: Spotřeba jednotlivých prvků

Název	Proudová zátěž (v $\mu\text{A}$ )
Mikrokontrolér (aktivní)	220
Mikrokontrolér (standby)	1
GSM modul (standby)	1500
GSM modul (aktivní)	150000
Senzory	3
Windsensor	20000 až 40000
Napájecí modul(spotřeba naprázdno)	5000
Napájecí modul(klidový stav)	80

## 7.3 Poznámky k dalším verzím

Zde uvedu pár poznámek k vývoji dalších verzí tohoto zařízení na které jsem přišel během tvorby této diplomové práce.

- Lepší přístup k tlačítkům umístěným na vývojové desce. Neboť je možné vyvolat přerušení manuálně právě tlačítkem, to je ovšem špatně přístupné.
- Oddělené napájení GSM modulu a mikrokontroléru s ostatními perifériemi. GSM modul je energeticky velice náročný a potřebuje speciální napájení pomocí stejnosměrného měniče napětí, který má vysoký odběr naprázdno. Toho by šlo zamezit odpojováním tohoto napájecího modulu, pokud by byl procesor napájen jiným zdrojem energie.
- Další tlačítka na navrženou desku pro rozšířené ovládání celého zařízení např. bezpečné odpojení SD karty bez ztráty dat.
- Implementovat více druhu datové komunikace k rozmanitosti použití.

## 8 ZÁVĚR

V této diplomové práci jsem prozkoumal nabídku dataloggerů podobných mému zadání a uvedl jejich příklady k porovnání. Z tohoto průzkumu jsem udělal přehled funkcí běžně nabízených těmito přístroji. Všechny jsem uvedl pro využití k další práci a vývoji mého dataloggeru.

Nejdříve jsem hledal možnosti použití různých senzorů pro měření vlhkosti a teploty. Z hlediska ceny, přesnosti, spotřeby energie a dostupnosti k použití v mikroprocesorem řízeném zařízení. //Udělal jsem malý přehled s pár příklady z dnešní nabídky na trhu, převážně těch nejpoužívanějších. Dále se zabývám problematikou měření průtoku vzduchu. Diskutuji hlavně použití takzvaného "hot-wire" měření, a možnosti jeho rozšíření k zvýšení přesnosti.

K těmto senzorům jsem vybral mikroprocesor, který exceluje hlavně svou spotřebou a dostačujícími parametry pro sestavení jednoduchého autonomního dataloggeru ze zadání. Z hlediska programování uvádím nabízenou vývojovou sadu výrobce, pomocí které se dá programovat a ladit tento mikrokontrolér. Pro komunikaci jsem vybral GSM modul kompatibilní s tímto procesorem, zapojitelný přes UART.

Hlavním cílem práce bylo navrhnout a naprogramovat autonomní datalogger s nízkou spotřebou a velkou variabilitou. V práci o postupu tohoto návrhu píší a snažím se vše vysvětlit na příkládaných obrázcích. Závěrem popisuji programové vybavení tohoto zařízení a jeho funkce.

Zařízení je navrženo co nejmodulárněji, aby byly jeho jednotlivé části nahraditelné jak programově, tak hardwarově a jednotlivé části projektu byly přenositelné na jiná zařízení s co nejmenší nutností změn.

# LITERATURA

- [1] Texas Instruments Incorporated: *MSP430G2x53, MSP430G2x13 Mixed Signal Microcontroller (Rev. J)* [online]. Dallas: duben 2011, poslední aktualizace květen 2013 [cit. 20. 12. 2016].  
Dostupné z URL: <<http://www.ti.com/lit/ds/symlink/msp430g2553.pdf>>.
- [2] Texas Instruments Incorporated: *User's Guide: MSP-EXP430G2 LaunchPad™ Development Kit* [online]. Dallas: červen 2010, poslední aktualizace březen 2016 [cit. 20. 12. 2016].  
Dostupné z URL: <<http://www.ti.com/lit/ug/slau318g/slau318g.pdf>>.
- [3] COMET system, s.r.o.: [online][cit. 11. 11. 2016].  
Dostupné z URL: <<http://www.cometsystem.cz/produkty/dataloggery/>>.
- [4] TESTO, s.r.o.: [online][cit. 11. 11. 2016].  
Dostupné z URL: <<http://www.merici-pristroje.eu/>>.
- [5] I & CS spol. s r.o.: [online][cit. 11. 11. 2016].  
Dostupné z URL: <<http://www.e-pristroje.cz/vlhkomery-dataloggery.html/>>.
- [6] Texas Instruments Incorporated: *HDC1080 Low Power, High Accuracy Digital Humidity Sensor with Temperature Sensor* [online]. Dallas: listopad 2015, poslední aktualizace 20. 8. 2016 [cit. 13. 12. 2016].  
Dostupné z URL: <<http://www.ti.com/lit/ds/symlink/hdc1080.pdf>>.
- [7] Texas Instruments Incorporated: *HDC1010 Low Power, High Accuracy Digital Humidity Sensor with Temperature Sensor (Rev. A)* [online]. Dallas: květen 2016, poslední aktualizace srpen 2016 [cit. 6. 1. 2017].  
Dostupné z URL: <<http://www.ti.com/lit/ds/symlink/hdc1010.pdf>>.
- [8] Maxim Integrated Products, Inc.: *DS3231M Real-time clock* [online]. Sunnyvale: 10. 6. 2014, poslední aktualizace 15. 3. 2015 [cit. 20. 1. 2017].  
Dostupné z URL: <<https://datasheets.maximintegrated.com/en/ds/DS3231M.pdf>>.
- [9] Atmel Corporation: *2-Wire Serial EEPROM AT24C32* [online]. San Jose: 7. 3. 2003 [cit. 18. 1. 2017].  
Dostupné z URL: <<http://www.atmel.com/images/doc0336.pdf>>.

- [10] NXP Semiconductors N.V: I<sup>2</sup>C-bus specification and user manual (Rev. 6) [online]. 4. 4. 2014 [cit. 6. 3. 2017].  
Dostupné z URL: <[http://cache.nxp.com/documents/user\\_manual/UM10204.pdf](http://cache.nxp.com/documents/user_manual/UM10204.pdf)>.
- [11] Bosch Sensortec GmbH: *BME280: Final data sheet* [online]. Reutlingen: 26. 10. 2015 [cit. 20. 12. 2016].  
Dostupné z URL: <[https://ae-bst.resource.bosch.com/media/\\_tech/media/datasheets/BST-BME280\\_DS001-11.pdf](https://ae-bst.resource.bosch.com/media/_tech/media/datasheets/BST-BME280_DS001-11.pdf)>.
- [12] Sensirion AG: *Datasheet SHT20* [online]. Staefa: květen 2014 [cit. 20. 12. 2016].  
Dostupné z URL: <[https://www.sensirion.com/fileadmin/user\\_upload/customers/sensirion/Dokumente/2\\_Humidity\\_Sensors/Sensirion\\_Humidity\\_Sensors\\_SHT20\\_Datasheet\\_V4.pdf](https://www.sensirion.com/fileadmin/user_upload/customers/sensirion/Dokumente/2_Humidity_Sensors/Sensirion_Humidity_Sensors_SHT20_Datasheet_V4.pdf)>.
- [13] Howard Hodson: *Hot-Wire Anemometers* [online]. University of Cambridge, Cambridge [cit. 26. 12. 2016].  
Dostupné z URL: <<http://www-g.eng.cam.ac.uk/whittle/current-research/hph/hot-wire/hot-wire.html>>.
- [14] Woodward, S.: *Self-heated transistor digitizes airflow*. EDN, Chapel Hill: 14. 3. 1996, poslední aktualizace 30. 9. 2006 [cit. 26. 12. 2016].
- [15] Modern Device: *Wind Sensor Rev. C* [online]. Providence: 2011 [cit. 26. 12. 2016].
- [16] SimCOM: *The GSM/GPRS modul for M2M applications* [online]. [cit. 1. 4. 2017].  
Dostupné z URL: <<http://pandatron.cz/shop/ds/SIM900D.pdf>>.
- [17] Energia Project: *Guide to the MSP430 LaunchPad (MSP-EXP430G2)* [online]. Providence: 2011 [cit. 26. 12. 2016].  
Dostupné z URL: <[http://energia.nu/Guide\\_MSP430LaunchPad.html](http://energia.nu/Guide_MSP430LaunchPad.html)>.
- [18] Autodesk Inc.: [online][cit. 15. 1. 2017].  
Dostupné z URL: <<https://www.autodesk.com/products/eagle/overview/>>.
- [19] ChaN: *Petit FatFs - FAT file system module R0.02a* [online]. 2010 [cit. 15. 4. 2017].  
Dostupné z URL: <[http://elm-chan.org/fsw/ff/00index\\_p.html](http://elm-chan.org/fsw/ff/00index_p.html)>.
- [20] Ubinetic: *AT command set for GSM*. 2001 [cit. 21. 4. 2017].

## SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

<i>mph</i>	míle za hodinu (jednotka rychlosti)
<i>kmh</i>	kilometry za hodinu (jednotka rychlosti)
ADC	Analog-digital converter (analogově-číslicový převodník)
USI	Universal serial interface - víceúčelové sériové rozhraní
USCI	Universal serial communication interface - univerzální rozhraní pro sériovou komunikaci
SPI	Serial Peripheral Interface - sériové periferní rozhraní
I <sup>2</sup> C	Inter-Integrated Circuit - multi-masterová počítačová sériová sběrnice
GPIO	General purpose input/output - vstup/výstup k běžnému použití
BCD	Binary coded decimal - binárně kódované dekadické číslo
GSM	Groupe SpécialMobile - Globalní Systém pro mobilní komunikaci, nejpoužívanější standard
UART	Universal asynchronous receiver and transmitter - zařízení pro asynchronní sériovou komunikaci
RISC	Procesory s redukovanou instrukční sadou
FLASH	Nevolatilní elektricky programovatelná paměť s libovolným přístupem
Li-ion	Lithium-iontová baterie - nejpoužívanější typ akumulátoru



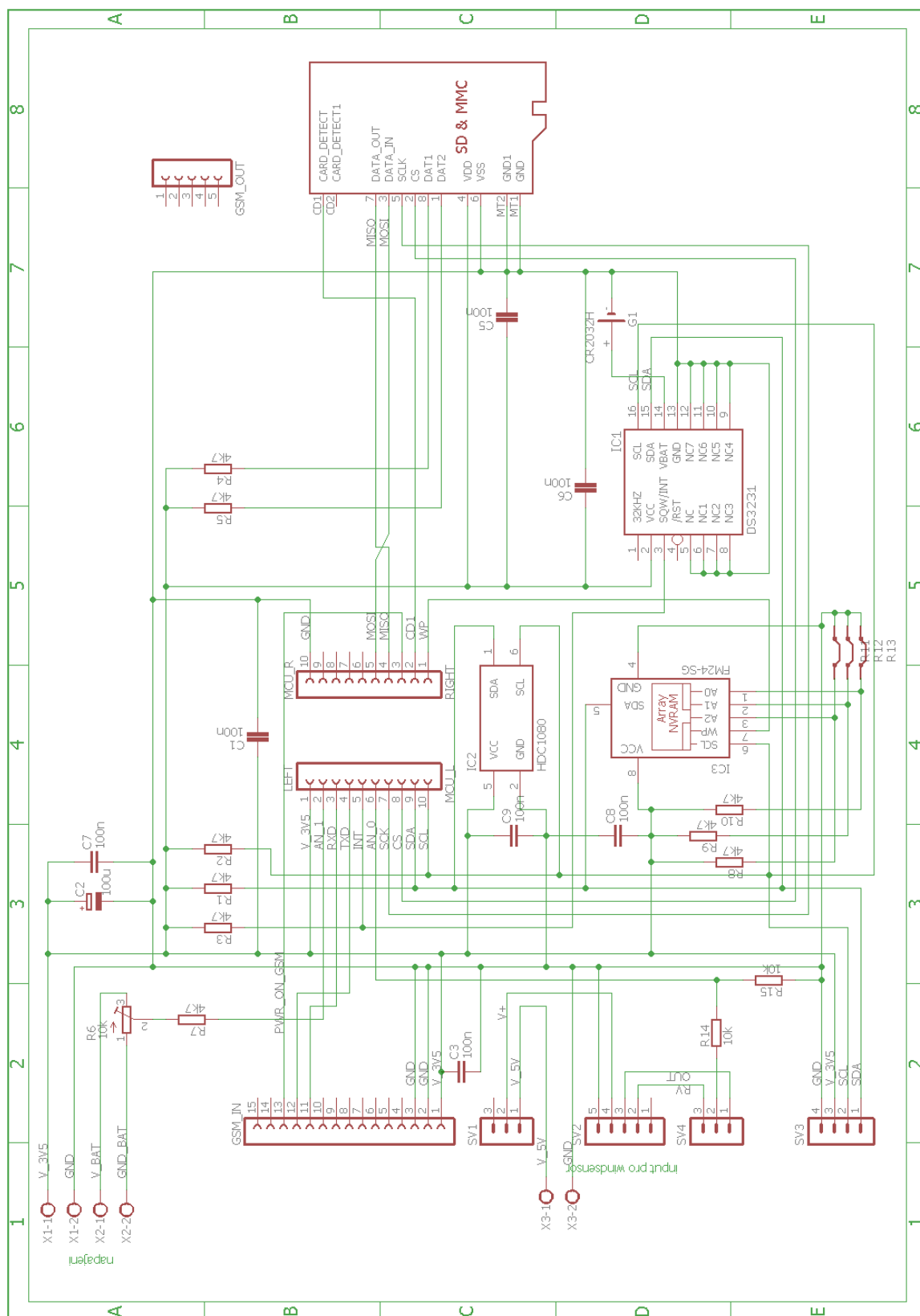
# SEZNAM PŘÍLOH

A Dokumentace dataloggeru	41
B Příklady zdrojových kódů	45
C Obsah přiloženého CD	48

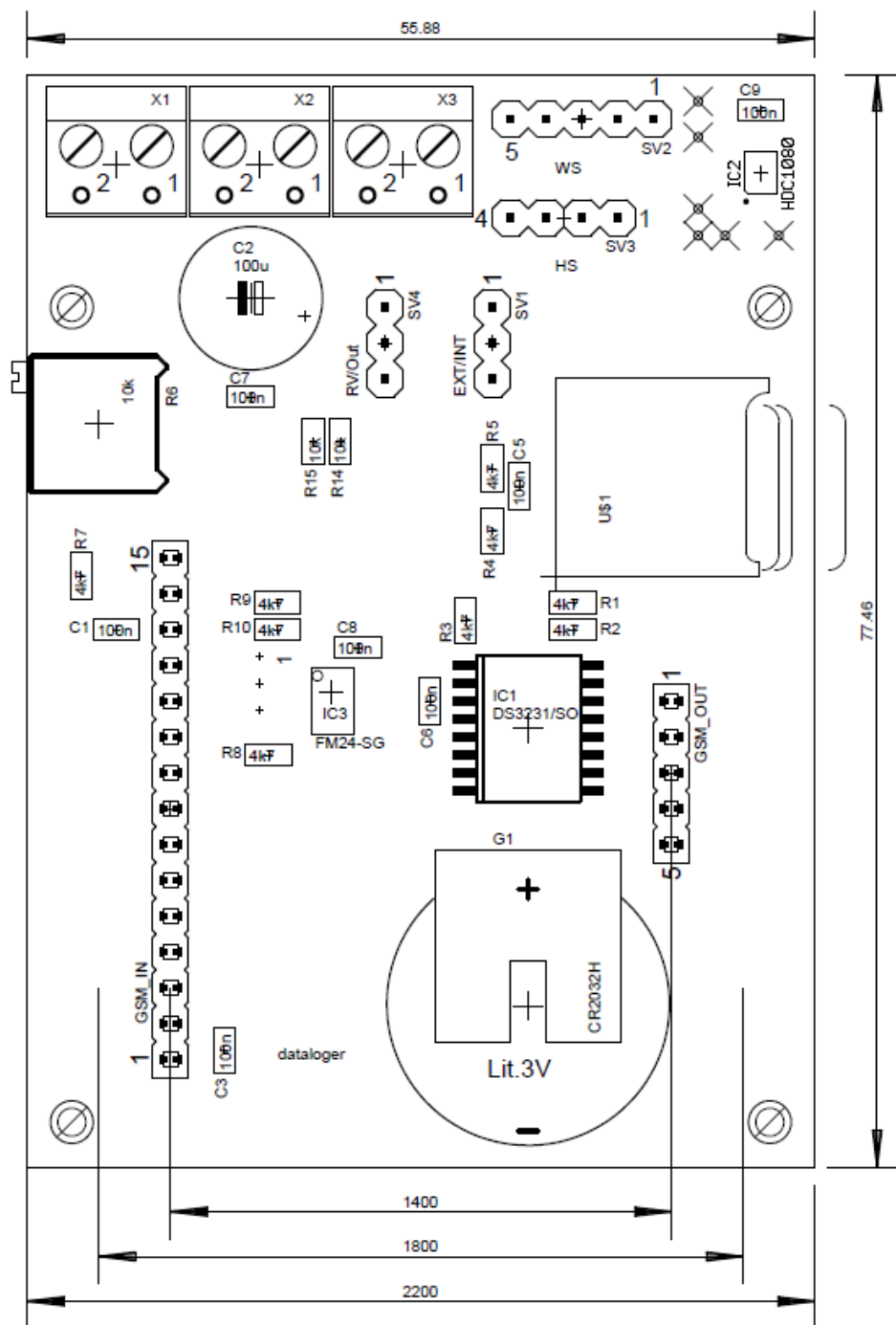
## A DOKUMENTACE DATALOGGERU

Tab. A.1: Seznam součástek osazených na desce

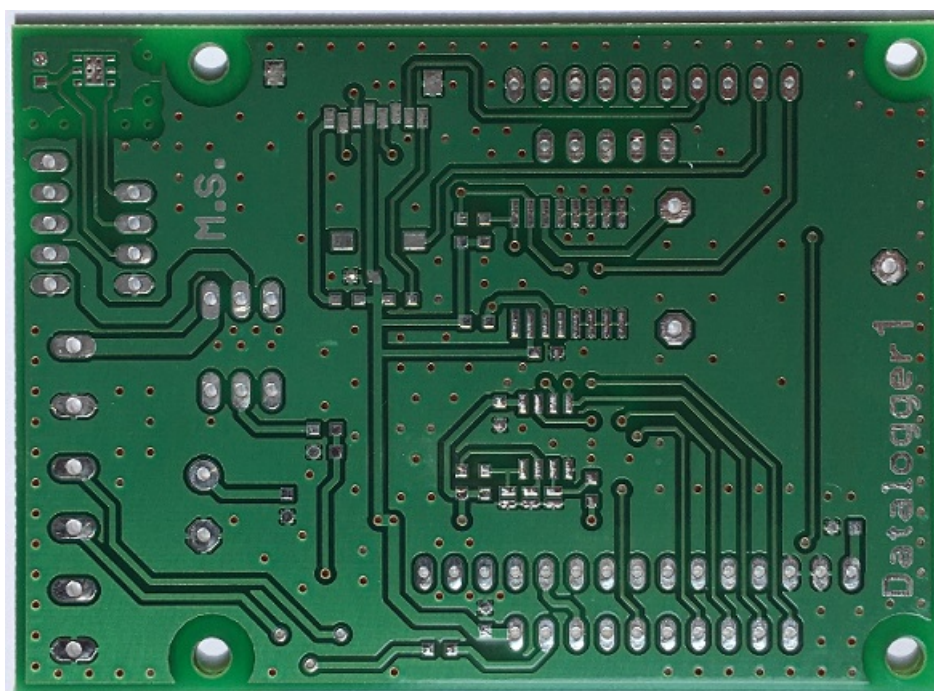
Značka	Název	Hodnota	Package
C1	C0603	100 nF	0603
C2	C	100 $\mu$ F	RM 5
C3 - C9	C0603	100 nF	0603
G1	CR2032		Horizontal
GSM_IN	FE15-1	15 pin	
GSM_OUT	FE05-1	5 pin	
IC1	DS3231M		SO16
IC2	HDC1080		WSON6
IC3	EEPROM	24C32N	SO8
MCU_L	FE10-1	10 pin	
MCU_R	FE10-1	10 pin	
R1 - R5	R0603	4k7	0603
R6	R-Trim	10k	
R7 - R10	R0603	4k7	0603
R11 - R13	Jumper		
R14 - R15	R0603	10k	0603
SV1 - SV4	MAxx-1		
U1	MICROSD		
SV1 - SV4	MAxx-1		
X1 - X3	W237-102		



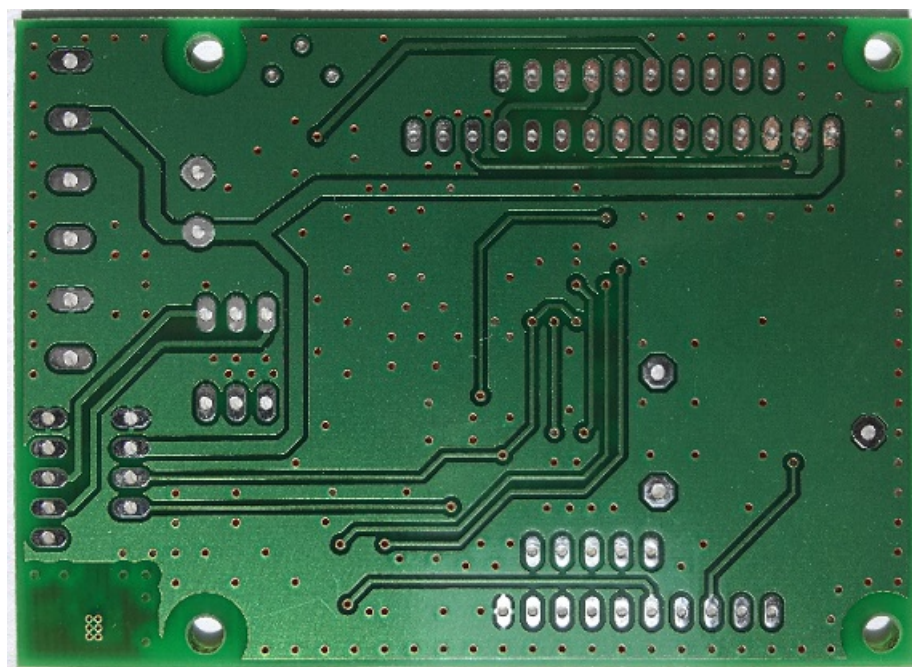
Obr. A.1: Schéma zapojení dataloggeru



Obr. A.2: Osazovací plán - rozmístění součástek na desce



Obr. A.3: Vyrobená deska - pohled ze strany součástek



Obr. A.4: Vyrobená deska - zadní strana

## B PŘÍKLADY ZDROJOVÝCH KÓDŮ

Výpis B.1: Příklad implementace I<sup>2</sup>C sběrnice v C.

```
1 // Send byte to slave (one by one)
2 void sendByte(void) {
3     PIN_DIR |= SDA;    // set SDA as output
4     bitCounter = 0;
5     while(bitCounter < 8) {
6         (txData & BIT7) ? (PIN_OUT |= SDA) : (PIN_OUT &= ~SDA);
7         PIN_OUT |= SCL; // after SDA flip SCL
8         txData <<= 1;    // shift transfer buffer
9         bitCounter++;
10        PIN_OUT &= ~SCL; // SCL low
11    }
12    PIN_OUT |= SDA;    // set SDA high
13    PIN_DIR &= ~SDA;
14 }
15 // Master acknowledge for slave
16 void sendAck(unsigned char ack) {
17     ackFlag = ack;
18     PIN_DIR |= SDA;    // SDA output
19     (ackFlag) ? (PIN_OUT &= ~SDA) : (PIN_OUT |= SDA);
20     PIN_OUT |= SCL;    // after SDA flip SCL
21     PIN_OUT &= ~SCL;   // flip SCL again
22     PIN_OUT |= SDA;    // set SDA high
23     PIN_DIR &= ~SDA;
24 }
25 // Start condition of I2C bus
26 void start_i2c(void) {
27     PIN_OUT |= SCL;    // SCL high
28     PIN_DIR |= SDA;    // SDA output
29     PIN_OUT &= ~SDA;   // SDA low
30     PIN_OUT &= ~SCL;   // SCL low
31     PIN_OUT |= SDA;    // SDA high
32     PIN_DIR &= ~SDA;
33     usdelay(1000);    // delay 1000 us
34 }
```

## Výpis B.2: Příklad hlavní smyčky programu

```

1      // Main loop for measuring
2  while(1) {
3      time = interval;
4      getTime(&minutes,&hour,&date,&month);
5      getTemp_Hum(0x00,&temp1,&humi1);
6      measCount++;
7
8      ADC10CTL0 |= ENC + ADC10SC;      // start ADC
9      __bis_SR_register(CPUOFF + GIE); // LPM0 for ADC
10     if (ADC10MEM < 0x100)
11         airFlow = 25;      // Clear P1.0 LED off
12     else if (ADC10MEM < 0x200)
13         airFlow = 50;
14     else if (ADC10MEM < 0x300)
15         airFlow = 75;
16     else
17         airFlow = 99;
18
19     /*if(temp1 > highLimit || temp1 < lowLimit){
20         Line = 0;
21         sprintf(Line, "Text zprávy", temp1/100);
22         sendMessage(Line);
23     }*/
24
25     if(offset > 32768){
26         /*sprintf(Line, "LOG%d.CSV", (measCount%1088));
27         sendFile(Line);*/
28         sprintf(Line, "LOG%d.CSV", (measCount%1088)+1);
29         res = pf_open(Line); // open file
30         offset = 0;
31         if(res != FR_OK)
32             LPM4;
33     }
34     if(diskRes == RES_OK){
35         if(offset == 0){
36             strcat(Line, "#;DATE;TIME;TEMP;HUM;AFLOW\r\n");
37             res = pf_write(Line, strlen(Line), &bw);
38             offset += bw;
39             measCount++;
40         }

```

### Výpis B.3: Příklad hlavní smyčky programu - pokračování

```

1      sprintf(Line, "%04d;%x.%x.;%x:%x;%02d,%02d;%02d;%02d\r\n",
2          measCount,
3          date, month,
4          hour, minutes,
5          temp1/100, temp1-temp1/100*100, humi1,
6          airFlow);
7      res = pf_write(Line, strlen(Line), &bw);
8      offset += bw;
9      if( (measCount%17) == 0 ){
10         res = pf_write(0, 0, &bw);
11         res = pf_lseek(offset+100);
12     }
13     } // end if
14     P1IE |= BIT3;           // P1.3 interrupt enabled
15     do{
16         resetAF();
17         LPM4;
18     }while((--time)!=0);
19     P1IE &= ~BIT3; // P1.3 interrupt disabled
20     } // end while
21 } // end main()

```



## C OBSAH PŘILOŽENÉHO CD

```
/ ..... kořenový adresář přiloženého CD
├── doc ..... dokumentace k dataloggeru
├── src
│   ├── thesis ..... Zdrojový kód práce ve formátu Latex
│   └── app ..... Zdrojové kódy pro mikrokontrolér [CSS 7.1]
└── text
    └── stabl_martin_2017.pdf ..... hlavní soubor diplomové práce
```