

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA ELEKTROTECHNICKÁ



System pro měření spotřeby energií a dalších veličin v budově

Diplomová práce

Studijní obor: Inteligentní budovy

Vypracoval: **Bc. Miroslav Hašek**

Vedoucí práce: **Ing. Pavel Mlejnek, Ph.D.**

Datum: **11.5.2015**



Akademický rok 2014-15

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta elektrotechnická
Katedra měření

ZADÁNÍ DIPLOMOVÉ PRÁCE

Student: **Bc. Miroslav Hašek**
Studijní program: **Inteligentní budovy**

Název tématu česky: **Systém pro měření spotřeby energií a dalších veličin v budově**

Název tématu anglicky: **System for energy measurement and other quantities in building**

Pokyny pro vypracování:

Navrhněte a realizujte měřicí modul na DIN lištu, který bude schopen vyčítat údaje o spotřebě elektrické energie, plynu a vody. Déle bude tento modul schopen získávat údaje o teplotách a vlhkostech z externích senzorů. V modulu implementujte vhodné komunikační rozhraní, které umožní zapojení do rozsáhlejšího systému MaR. Vyvinutý modul doplňte displejem, který bude přehledně zobrazovat naměřené údaje. Modul bude všechny hodnoty také logovat do databáze a prezentovat v integrovaném webserveru. Modul vyzkoušejte ve vhodném objektu.

Seznam odborné literatury:

- [1] Merz, H., Hansemann, T., Hübner, Ch.: Automatizované systémy budov. Praha: Grada Publ. a.s. 2008. ISBN 978-80-247-2367-9.
- [2] Haasz, V., Sedláček, M.: Elektrická měření: přístroje a metody. ČVUT 2005. ISBN 80-01-02731-7
- [3] ČSN EN 62056-21 Měření elektrické energie – Výměna dat pro odečet elektroměru, řízení tarifu a regulaci zátěže – Část 21: Přímá místní výměna dat. Březen 2004

Vedoucí diplomové práce: Ing. Pavel Mlejnek, Ph.D.

Datum zadání diplomové práce: 15. ledna 2015

Platnost zadání do¹: 31. srpna 2016

L.S.

Doc. Ing. Jan Holub, Ph.D.
vedoucí katedry

Prof. Ing. Pavel Ripka, CSc.
děkan

Anotace

Diplomová práce se zabývá návrhem měřicího modulu pro vyčítání údajů o spotřebě v domácnosti. V úvodní části práce jsou stručně popsány způsoby měření spotřeby elektřiny, plynu a vody s ohledem na návrh měřicího systému. Dále je popsána programovatelná deska Arduino YÚN a sběrnice, které jsou použity pro komunikaci se senzory a dalšími systémy. Praktická část je věnována návrhu a realizaci měřicího modulu. Blíže jsou také popsány jednotlivé způsoby vyčítání údajů o spotřebě včetně měření teploty, relativní vlhkosti a osvětlení.

Klíčová slova

měření spotřeby energie, měření teploty, měření relativní vlhkosti, Arduino, měřicí modul, sběr a ukládání dat

Annotation

The diploma thesis deals with the design of the measuring module aimed at reading the household consumption data. The ways of measuring the electricity, gas, and water consumption are described in the introductory part of the work in view of the design of the measuring system. Further on there is a description of the programmable board Arduino YÚN together with the bus used for the communication with the sensors and other systems. The practical part is focused on the design and implementation of the measuring module. The individual ways of reading the consumption data are dealt with in a greater detail including the measurements of temperature, relative humidity, and lighting.

Key words

energy consumption measurements, measurements of temperature and relative humidity, Arduino, measuring module, data collecting and saving

Čestné prohlášení autora práce

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne

.....

Podpis autora práce

Poděkování

Na tomto místě bych rád poděkoval Ing. Pavlu Mlejnkovi, Ph.D. za záštitu nad touto prací, cenné rady a připomínky.

Obsah

1	Monitoring spotřeby energie a dalších veličin	9
1.1	Měření spotřeby elektřiny	9
1.2	Měření spotřeby plynu	15
1.3	Měření spotřeby vody	16
2	Arduino	19
2.1	Arduino YÚN - hardware	20
2.2	Software	22
2.3	Arduino - Shields	23
3	Sběrnice a protokoly	25
3.1	UART	25
3.2	RS232	25
3.3	RS485	26
3.4	I2C	26
3.5	1-Wire	27
3.6	Ethernet	28
3.7	Bezdrátový přenos	29
4	Senzory	30
5	Měřicí modul	31
5.1	Architektura systému	31
5.2	Návrh měřicího modulu	32
5.3	Velikost měřicího modulu a výběr krabičky	32
5.4	Návrh napájení	33
5.5	Impulzní, binární a analogové vstupy	34
5.6	Reléové výstupy	35
5.7	Ovládací tlačítka s I/O expandérem	36
5.8	LCD displej	38
5.9	Převodník UART - RS232	39
5.10	Převodník UART - RS485	39
5.11	1-Wire driver	40
5.12	Obvod reálného času	40
5.13	Bezdrátový přijímač	41
5.14	LED signalizace	41
5.15	Výroba DPS	42
5.16	Přední panel	43
5.17	Finální modul	44
6	Měření spotřeby energie a dalších veličin	45
6.1	Vyčítání spotřeby pomocí LED indikující spotřebu elektřiny	45
6.2	Vyčítání spotřeby plynu	46
6.3	Vyčítání spotřeby vody	48

6.4	Měření teploty	48
6.5	Měření relativní vlhkosti	51
6.6	Měření osvětlení	52
7	Software	55
7.1	Arduinu YÚN - MCU	55
7.2	Arduinu YÚN - Linuxová část	61
8	Závěr	65
9	Příloha 1 - OBIS kódy	70

Seznam obrázků

1	Lokální měření spotřeby elektřiny (převzato z [1])	9
2	Analogový elektroměr (převzato z [2])	10
3	Shunt (převzato z [3])	11
4	Statický elektronický elektroměr (převzato z [4])	11
5	Využití LED diody indikující spotřebu	12
6	Elektroměrová optická hlavice	12
7	Měřicí transformátory proudu (převzato z [5])	15
8	Membránový plynoměr (převzato z [6])	16
9	Impulzní snímač otáček (převzato z [7])	16
10	Lopatkový vodoměr (převzato z [8])	17
11	Objemový vodoměr (převzato z [8])	17
12	Vodoměr s impulzním výstupem (převzato z [9])	18
13	Logo Arduino (převzato z [10])	19
14	Arduino YÚN (převzato z [11])	19
15	Arduino YÚN - komunikace mezi MCU a SoC (převzato z [11])	20
16	Stavové LED (převzato z [11])	21
17	ATmega32u4 - zapojení I/O pinů (převzato z [12])	21
18	Ukázka programu v Arduino IDE	23
19	I2C sběrnice(převzato z [13])	26
20	I2C sběrnice - datový rámec(převzato z [13])	26
21	1-wire sběrnice (převzato z [14])	27
22	1-wire, zápis a čtení(převzato z [14])	27
23	Inteligentní senzor - blokové schéma (převzato z [15])	30
24	Konfigurace V2	31
25	Měřicí modul	32
26	Krabička pro měřicí modul (převzato z [16])	33
27	Schéma napájení (převzato z [17])	33
28	Schéma napájení	34
29	Galvanicky oddělené vstupy	34
30	Binární a analogové vstupy	35
31	Reléové výstupy	36

32	Použití relé (převzato z [18])	36
33	Popis pinů PCF8574 (převzato z [19])	37
34	Tlačítka a PCF8574	38
35	LCD displej s expandérem	38
36	Schéma převodníku	39
37	Schéma převodníku UART - RS485	40
38	1-wire driver	40
39	Schéma RTC obvodu	40
40	RTC obvod	41
41	Schéma bezdrátového přijímače	41
42	LED signalizace	42
43	Obrazec spodní DPS	42
44	Obrazec vrchní DPS	43
45	Foto spodní DPS (s vloženým Arduino YÚN) a vrchní DPS s LCD displejem	43
46	Přední panel měřicího modulu	44
47	Foto měřicího modulu	44
48	Schéma obvodu pro snímání impulzů z elektroměru	45
49	Umístění sondy na elektroměru	45
50	Výpis dat z elektroměru	46
51	Schéma switch debounceru	46
52	Blokové schéma hallova senzoru (převzato z [20])	47
53	Analogový číselník plynoměru	47
54	Možné umístění optosenzoru na plynoměru	48
55	Schéma pro vyčítání impulzů z plynoměru	48
56	Teplotní závislost odporu termistoru (převzato z [21])	49
57	Připojení termistoru a senzoru LM35 k analogovému vstupu	49
58	Senzor teploty LM35	50
59	DS18B20 (převzato z [22])	50
60	Schéma zapojení senzorů DS18B20 (převzato z [22])	51
61	DHT11 (převzato z [23])	51
62	Připojení senzoru DHT11 k mikrokontroléru (převzato z [23])	52
63	BPW21 (převzato z [24])	52
64	Poměrná spektrální citlivost lidského oka a fotosenzoru (převzato z [24])	53
65	TSL 2561 (převzato z [25])	53
66	Blokové schéma TSL 2561 (převzato z [25])	53
67	Poměrná spektrální citlivost TSL 2561 (převzato z [25])	54
68	Piny I2C sběrnice měřicího modulu	54
69	Výpis dat na LCD displej	58
70	Výpis naměřených dat	62
71	Graf spotřeby elektrické energie - 24hod	63
72	Webová vizualizace	64

Seznam tabulek

1	Rámec s žádostí	13
2	Rámec s žádostí - význam jednotlivých bajtů	13
3	Rámec s identifikací	13
4	Rámec s identifikací - význam jednotlivých bajtů	13
5	Rámec s potvrzením a žádostí o data	13
6	Rámec s potvrzením - význam jednotlivých Bytů	14
7	Rámec s daty	14
8	Rámec s daty - význam jednotlivých bajtů	14
9	Popis OBIS kódů	14
10	Základní parametry Arduino YÚN	20
11	Tabulka I/O pinů Arduino	22
12	Formát MAC rámce	28
13	Tabulka pinů PCF8574	37
14	Nastavení adresy expandéru	37
15	Význam pinů LCD displeje	39

Úvod

V souvislosti se zvyšujícím se komfortem bydlení roste i poptávka po hospodárném využití energetických zdrojů. Důvody, proč tomu tak je, mohou být různé. Pro někoho je to otázka finančních úspor v souvislosti se zvyšující se cenou energií, pro druhého zase ekologický aspekt hospodárného využití energie. Základním předpokladem pro efektivní využití energetických zdrojů je detailní monitoring jejich toků, a to jak v makroměřítku, tak i v mikroměřítku, jako například v objektu pro bydlení.

Tato diplomová práce se zabývá především monitoringem spotřeby energií v domácnostech. Konvenční monitoring spotřeby energie probíhá především za účelem fakturace nakoupené energie od konkrétního distributora. Čtvrtletní nebo měsíční faktura však pro detailní monitoring a optimalizaci spotřeby nestačí. Diplomová práce si klade za cíl vytvořit systém pro monitoring spotřeby energií a dalších medií, který bude umět v reálném čase zaznamenávat údaje o spotřebě.

Motivací k sepsání této práce přispěl zájem o open-source vývojovou desku Arduino a přesvědčení, že poskytuje dostatek prostoru pro návrh a realizaci měřicího modulu.

V úvodní části práce jsou stručně popsány způsoby měření spotřeby elektřiny, plynu a vody s ohledem na návrh měřicího systému. Následuje kapitola zabývající se popisem programovatelné desky Arduino YÚN. Další kapitola se věnuje popisem sběrnic, které jsou použity pro komunikaci se senzory a dalšími systémy.

Praktická část je věnována návrhu a realizaci měřicího modulu. Blíže jsou také popsány jednotlivé způsoby vyčítání údajů o spotřebě včetně měření teploty, relativní vlhkosti a osvětlení. V závěru práce je popsána softwarová část měřicího modulu.

1 Monitoring spotřeby energie a dalších veličin

V současné době existuje mnoho systémů pro měření a fakturaci elektrické energie. Některé z nich jsou autonomní, jiné pak součástí systémů inteligentního řízení budovy. Co se týká průmyslového řešení velkých objektů je situace a nabídka podstatně bohatší než u objektů typu rodinného domu či bytu.

Velké budovy dnes již zpravidla obsahují více či méně kvalitní energetický management budovy. Tyto systémy se označují pod pojmem BEMS. Starají se o měření spotřeby energií, měření dalších hodnot, které spotřebu ovlivňují a dále vhodně řídí tok energií a potřebných medií. Pro domácnosti se používá termín HEMS a slouží podobně jako BEMS pro efektivnější využití energie podporující úspory.[26]

Systémy pro měření spotřeby energie jsou často integrované do některého typu inteligentní elektroinstalace, ať už jde o monitoring postavený na decentralizovaném systému inteligentní elektroinstalace KNX, nebo o monitoring postavený na některém z uzavřených typů inteligentních elektroinstalací (např. Tecomat, Ego-n, Loxone).

U autonomních systémů bývá ve většině případů sledována pouze jedna veličina (např. spotřeba elektřiny). Ve velké míře se pak využívá různých lokálních měřičů spotřeby energie (pro konkrétní spotřebič).



Obrázek 1: Lokální měření spotřeby elektřiny (převzato z [1])

Vyjimku tvoří některé autonomní systémy pro komplexní měření spotřeby elektřiny, plynu v domácnostech, jako jsou například moduly SDS od společnosti onlinetechnology.cz, nebo otevřený projekt Open Energy Monitor, z kterého jsem vycházel při návrhu měřícího modulu.

1.1 Měření spotřeby elektřiny

Pro měření spotřeby elektrické energie se využívá elektroměrů různých druhů. Elektroměry fungují v zásadě jako wattmetry integrující výkon v čase. Pro okamžitý výkon střídavého proudu platí následující vzorec.

$$p(t) = i(t) \cdot u(t) \quad (1)$$

Spotřebovaná elektrická energie (práce vykonaná elektrickým proudem) je pak dána integrací výkonu v čase (vzorec 2). Spotřeba elektrické energie je nejčastěji udávána v kilowatthodinách (kWh), wattsekundách (Ws) atd.[27, 28]

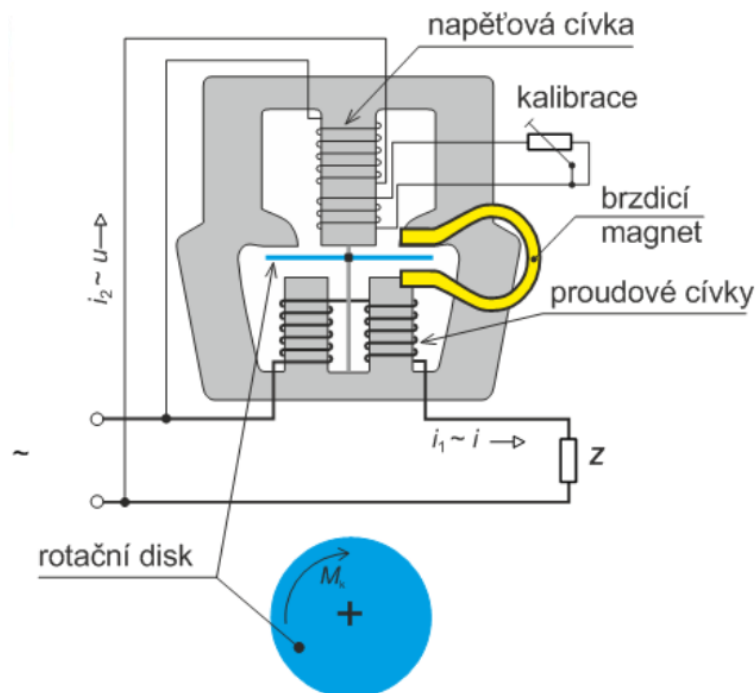
$$W = \int P(t) \cdot dt \quad (2)$$

1.1.1 Elektroměry

podle principu měřících soustav dělíme elektroměry na:

- analogové indukční elektroměry
- elektronické statické elektroměry

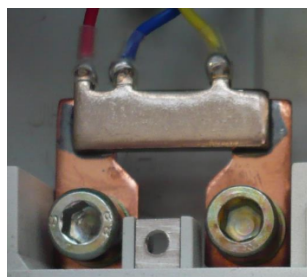
Analogové elektroměry fungují na principu elektromagnetické indukce podobně jako asynchronní motor s kotvou nakrátko. Základní prvky analogového elektroměru - dvojice elektromagnetů (napět'ový a proudový), díky své konstrukci vytvářejí točivé magnetické pole působící na hliníkový otočný kotouč umístěný ve vzduchové mezeře. V něm se indukují vířivé proudy, které ho uvádějí do pohybu. Principiální schéma analogového elektroměru lze vidět na obrázku 2.[27, 28]



Obrázek 2: Analogový elektroměr (převzato z [2])

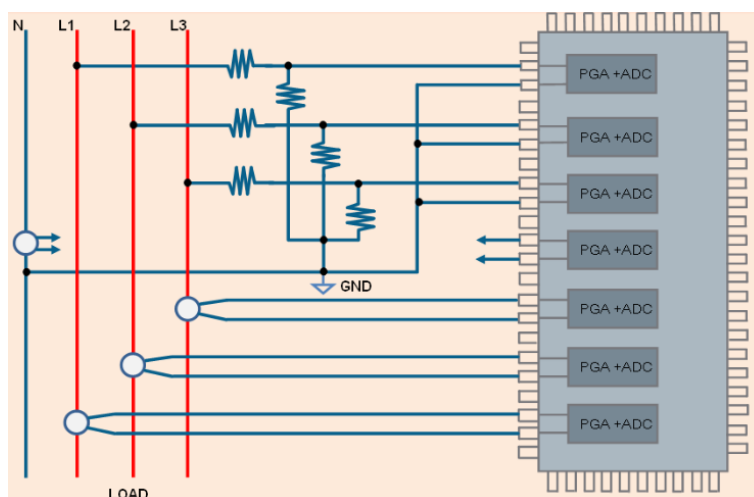
Elektronické statické elektroměry jsou založeny na principu, který nevyžaduje převod na mechanický pohyb. Prvky měřicí soustavy používané u statických elektroměrů jsou:

- Hallova sonda v zapojení analogové násobičky.
- Měřicí transformátory proudu.
- Bočníky (shunt).



Obrázek 3: Shunt (převzato z [3])

Příklad třífázového elektroměru s měřicími transformátory proudu je na obrázku 4. Pro měření napětí se pak využívá odporový dělič. Měřené veličiny z měřicích transformátorů a odporových děličů jsou přivedeny na vstup mikrokontroléru, který obsahuje obvody pro zesílení signálu (operační zesilovače) a pro převod na digitální signál (AD převodník). Výkon, resp. energie je pak vypočítána numericky v mikrokontroléru.[28]

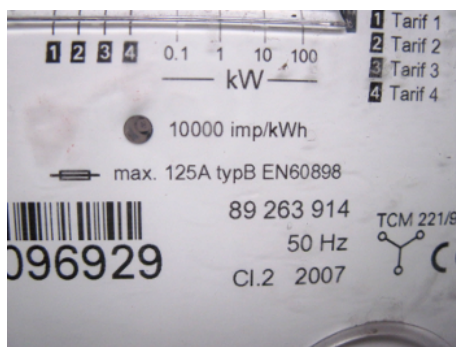


Obrázek 4: Statický elektronický elektroměr (převzato z [4])

1.1.2 Možnosti vyčítání informací o spotřebě z elektroměru

Pro měření spotřeby elektrického proudu přímo z elektroměru napojeného na distribuční soustavu elektrické energie jsou k dispozici v zásadě tři možnosti.

- Vyčítání dat z elektroměru pomocí LED indikující spotřebu elektrické energie
 - hodnota převodní konstanty bývá udána na těle elektroměru poblíž LED a její hodnota obvykle nabývá hodnot od 200 imp/kWh do 10000 imp/kWh
 - vhodným snímacím prvkem, například fototranzistorem, lze zaznamenávat impulzy



Obrázek 5: Využití LED diody indikující spotřebu

- Vychytání dat z elektroměru pomocí elektroměrové optické hlavice komunikující podle normy ČSN EN 62056-21.
- jedná se o specifický protokol, který bude blíže popsán v kapitole 1.1.3.



Obrázek 6: Elektroměrová optická hlavice

- Pomocí vyvedeného rozhraní S0
 - vyžaduje distributorem schválený galvanicky oddělený spínač pro snímání impulzů elektroměru
 - nutný zásah technika distributora el. energie

1.1.3 Měření spotřeby pomocí optického rozhraní

Tato možnost umožňuje sledovat kromě celkové spotřeby elektrické energie ještě další parametry, jako například zatížení jednotlivých fází. Nicméně záleží na distributorovi elektrické energie, které z těchto údajů zpřístupní pro uživatelský odečet. Standard totiž definuje několik uživatelských úrovní a vyšší úrovně jsou tak chráněny heslem.

1.1.4 Norma ČSN EN 62056-21

Norma vycházející z normy IEC 62056-21 a popisuje způsob komunikace s elektroměry, především pak popis komunikačního protokolu.

Základní vlastnosti komunikačního protokolu jsou:

- poloduplexní asynchronní přenos

- přenosová rychlost 300 bps (výchozí - lze navýšit až na 19200 bps)
- 1 start bit, 7 datový bitů, 1 stop bit, sudá parita

Norma připouští pět různých komunikačních módů (A-E), které se liší způsobem předávání dat (master-slave, client-server atd.).

V této diplomové práci byla testována komunikace s elektroměrem pomocí módu E, který je založen na komunikaci client (řídící systém) - server (elektroměr). Data se předávají pomocí datových rámců a komunikaci iniciuje vždy řídící systém.

Na začátku komunikace vyšle řídící systém elektroměru datový rámec s žádostí o identifikaci.

/	?	Adresa zařízení	!	CR	LF
2Fh	3Fh	(max. 32 znaků)	21h	0Dh	0Ah

Tabulka 1: Rámec s žádostí

Znak	Význam
/	Startovní znak
?	Povel s požadavkem na přenos
!	Koncový znak
CR	Ukončovací znak - návrat vozíku
LF	Ukončovací znak - posun o řádku

Tabulka 2: Rámec s žádostí - význam jednotlivých bajtů

Položka Adresa zařízení je nepovinná (pokud komunikace probíhá pouze s jedním elektroměrem). Na žádost o identifikaci odpoví elektroměr rámcem s možnostmi nastavení rychlosti, komunikačního módu a identifikací.

/	X	X	X	Z	\	W	Identifikace	CR	LF
2Fh	(A-Z)	(A-Z)	(A-Z)	(0-9)	5Ch	-	(max. 16 znaků)	0Dh	0Ah

Tabulka 3: Rámec s identifikací

Znak	Význam
/	Startovní znak
X,X,X	Označení výrobce elektroměru
Z	Určení přenosové rychlosti
\	Součástí identifikace (nepovinný znak)
W	Součástí identifikace (nepovinný znak)
Identifikace	Znaková identifikace elektroměru
CR	Ukončovací znak - návrat vozíku
LF	Ukončovací znak - posun o řádku

Tabulka 4: Rámec s identifikací - význam jednotlivých bajtů

Poté je potřeba nastavit novou komunikační rychlost (bajt Z) a zaslat žádost o data.

ACK	V	Z	Y	CR	LF
06h	(0-9)	(0-9)	(0-9)	0Dh	0Ah

Tabulka 5: Rámec s potvrzením a žádostí o data

Znak	Význam
ACK	Potvrzovací znak
V	Řídící znak protokolu (0 - procedura normálního protokolu)
Z	Přenosová rychlost
Y	Řídící znak protokolu (0 - odečet dat)
CR	Ukončovací znak - návrat vozíku
LF	Ukončovací znak - posun o řádku

Tabulka 6: Rámec s potvrzením - význam jednotlivých Bytů

Následně elektroměr vyšle rámec s daty.

STX	Datový blok	!	CR	LF	ETX	BCC
02h	(data)	21h	0Dh	0Ah	03h	(kontrolní znak)

Tabulka 7: Rámec s daty

Znak	Význam
STX	Znak začátku rámce
Datový blok	Datový blok s měřenými hodnotami
!	Koncový znak
CR	Ukončovací znak - návrat vozíku
LF	Ukončovací znak - posun o řádku
ETX	Koncový znak bloku
BCC	Kontrolní znak bloku

Tabulka 8: Rámec s daty - význam jednotlivých bajtů

Datový blok se skládá z datových řádků, kde každý řádek obsahuje jednoznačné identifikační číslo změřené veličiny (OBIS kód), hodnotu veličiny a jednotku veličiny. OBIS kód jednoznačně definuje, co která hodnota znamená. Vzhledem k tomu, že OBIS kódy mohou využívat i jiné měřiče spotřeby, je standardem definováno šest různých identifikátorů. Elektroměry obvykle využívají pouze tři identifikátory.

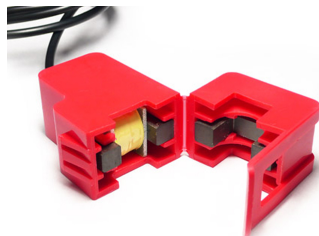
První znak každého řádku určuje buď fyzikální veličinu (označeno číslem - proud, napětí, energie, úroveň), nebo událost (označeno písmenem). Druhý znak definuje výsledek výpočtu množství podle specifického algoritmu (minimum, maximum). Třetí znak pak upřesňuje typ měření (tarif, číslo fáze). Vysvětlení některých OBIS kódů, viz tabulka 9. Další pak v příloze 1.[29, 28]

OBIS kód	Popis
21.8.0	Kladná činná energie z fáze L1
41.8.0	Kladná činná energie z fáze L2
61.8.0	Kladná činná energie z fáze L3
C.2.1	Poslední změna parametrů elektroměru (vynulování čítače)
C.1.0	Sériové číslo elektroměru
0.0.0	Adresa zařízení
1.8.0	Celková dodaná činná energie
1.8.1	Celková dodaná činná energie, tarif T1
C.7.0	Celkový výpadek energie - čítač

Tabulka 9: Popis OBIS kódů

1.1.5 Další alternativní způsoby měření spotřeby elektřiny

Orientační měření spotřeby lze také provést pomocí měřicích proudových transformátorů, které se připevní na jednotlivé fáze měřené sítě. Na výstup měřicího proudového transformátoru lze pak připojit definovaný odpor, na němž lze měřit výstupní napětí (I/U převodník). Napětí je nutné navzorkovat vhodnou vzorkovací frekvencí.



Obrázek 7: Měřicí transformátory proudu (převzato z [5])

Pro výpočet okamžitého výkonu se musí dále měřit i napětí sítě. Okamžitou hodnotu napětí lze měřit na malém transformátoru, který je připojen do sítě. Lze opět použít AD převodník s vhodnou vzorkovací frekvencí. Z převodového poměru transformátoru pak lze vypočítat okamžitou hodnotu napětí, která se využije pro výpočet okamžitého výkonu. Do výpočtu se však také musí započítat fázové posuvy způsobené měřicími transformátory, ty lze změřit experimentálně. Tento způsob měření spotřeby elektřiny lze však brát pouze jako orientační. Více o tomto způsobu měření energie viz [30][28].

1.2 Měření spotřeby plynu

V souvislosti se snahou vyjadřovat spotřebu plynu v energetických jednotkách, není jediným kritériem spotřebované množství plynu v m^3 . Snaha je, aby spotřebitel platil přímo za energii obsaženou v zemním plynu. Standardně instalované plynoměry nicméně udávají spotřebu v objemových jednotkách a přepočítání na energii provede distributor plynu. Množství energie dodané odběrateli je vypočteno podle vzorce 3, kde V_p je objem plynu v m^3 dodaný odběrateli, k_p je přepočtový objemový koeficient a H_s^{15} je účtované spalné teplo zemního plynu dodaného za vykazované období. [31, 28]

$$Q = V_p \cdot k \cdot H_s^{15} \text{ [kWh]} \quad (3)$$

Spalné teplo H_s^{15} udává množství tepla, které lze získat dokonalým spálením určitého množství plynu se vzduchem při teplotě 15 °C. Z dlouhodobých měření odpovídá průměrná hodnota spalného tepla tranzitního plynu 10,5 kW/m³. Přesný výpočet spalného tepla se provádí podle ČSN ISO 6976. [31]

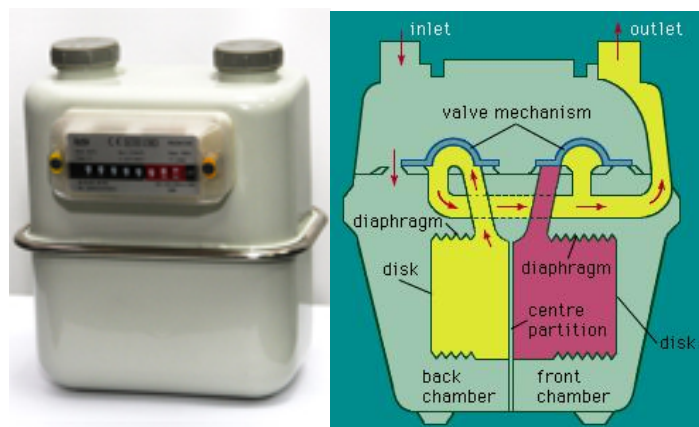
Přepočtový objemový koeficient se vypočítá dle vzorce 4.[31, 28]

$$k = \frac{T_v}{T_p} \cdot \frac{p_b + \Delta p_p}{p_v} \cdot \frac{z_v}{z_p} \text{ [-]} \quad (4)$$

Kde T_v je vztažná teplota zemního plynu, T_p - provozní teplota zemního plynu před plynoměrem, p_b - barometrický tlak vzduchu v místě odběru, závislý na nadmořské výšce, Δp_p - přetlak zemního plynu před plynoměrem, p_v - vztažný tlak zemního plynu, z_v - kompresní faktor při vztažných podmínkách, z_p - kompresní faktor při provozních podmínkách. Za normálních provozních podmínek je $k = 1$. [31, 28]

1.2.1 Princip membránového plynoměru

Membránový plynoměr měří spotřebu plynu v objemových jednotkách. Mechanismus se skládá ze dvou komor, kde každá komora je rozdělena membránou. Tím jsou vytvořeny čtyři měrné prostory, které jsou spojeny šoupátkovými rozvody s měřicím ústrojím. Měřicí ústrojí pak převádí pohyb šoupátek na rotační pohyb. Objem plynu, který představuje jednu otáčku (jeden pracovní cyklus) se nazývá cyklický objem a je udáván v dm^3 na štítku plynoměru. Rotační pohyb měřicího ústrojí je poté přes definovaný převod spřažen s rotačním číselníkem, který udává spotřebu s rozlišením na $0,001 \text{ m}^3$. [32, 33]



Obrázek 8: Membránový plynoměr (převzato z [6])

1.2.2 Měření spotřeby plynu pomocí magnetického kontaktu

Tento princip vyčítání spotřeby využívá té skutečnosti, že na posledním kolečku analogového bubínkového číselníku je umístěn neodymový magnet. Díky tomu lze snímat otáčky například pomocí jazýčkového magnetického kontaktu. Na trhu se prodává snímač (viz obrázek 9), který lze připevnit k plynoměru. Problémem tohoto snímače však je jeho vysoká cena a ve většině případů nutnost instalace technikem daného distributora. Některé plynoměry mají totiž zaplombovaný výřez pro vložení snímacího prvku.



Obrázek 9: Impulzní snímač otáček (převzato z [7])

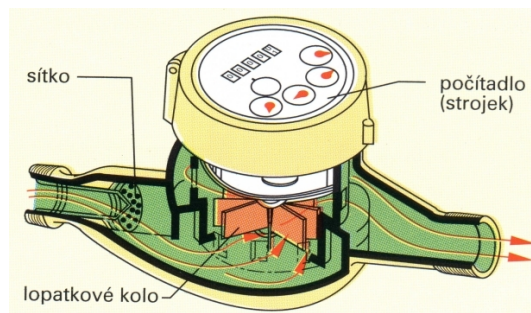
1.3 Měření spotřeby vody

Spotřeba vody se měří a fakturuje v objemových jednotkách (m^3). Vodárenské podniky měří spotřebu vody za pomoci vodoměrů různých druhů. Pro domácí monitoring spotřeby vody lze využít i průtokoměry různých konstrukcí, nicméně ty buď vyžadují zásah do vodoinstalace daného objektu, nebo představují příliš vysokou investici. Snahou tedy je odečítat spotřebu vody přímo z fakturačních vodoměrů vodárenských podniků.

Vodoměry používané pro měření spotřeby vody v běžné bytové výstavbě se dle konstrukce dělí na lopatkové a objemové.

1.3.1 Lopatkové vodoměry

Hlavní součástí lopatkového vodoměru je lopatkové kolo, které se vlivem protékající vody uvádí do pohybu. Pohyb tohoto kola se pak přenáší na počítadlo objemu. Lopatkové vodoměry se pak dále dělí na suchoběžné (počítadlo oddělené od pitné vody) a mokroběžné (počítadlo ponořeno do pitné vody). Pro studenou pitnou vodu se nejčastěji používají mokroběžné vodoměry. Suchoběžné vodoměry se zpravidla používají na teplou vodu pro kalorimetrická měření. Nevýhodou lopatkových vodoměrů je to, že vyžadují určitý minimální průtok pro pohyb lopatkového kola (cca 50 l/h).[8, 28]

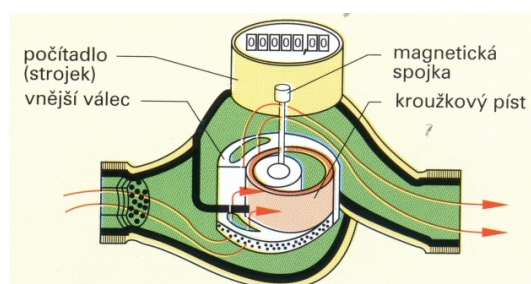


Obrázek 10: Lopatkový vodoměr (převzato z [8])

1.3.2 Objemové vodoměry

Objemové vodoměry se využívají tam, kde je potřeba měřit i velmi malé průtoky (od 4 l/h). Hlavní součástí objemového vodoměru je kroužkový píst, který se tlakem vody naplní a poté se uvede do pohybu do místa, kde dojde k jeho vyprázdnění. Otáčivý pohyb pístu se pak přenáší na objemové počítadlo. [8, 28]

I přes lepší vlastnosti objemového vodoměru se ve většině případů využívá lopatkových vodoměrů, a to vzhledem k jejich nižší ceně.



Obrázek 11: Objemový vodoměr (převzato z [8])

1.3.3 Vodoměry s impulzním výstupem

Nejsnazší způsob získávání údajů o spotřebě je za pomoci vodoměru s impulzním výstupem. Jedná se nejčastěji o lopatkový vodoměr. V principu se jedná o spřažení spínacího kontaktu a analogového počítadla vodoměru. Nadřazeným systémem je pak možné čítat jednotlivé impulzy a vyhodnocovat tak spotřebu vody. Pro co nejpřesnější měření je vhodné volit počítadlo s převodní konstantou alespoň 1 imp/10l, ideálně pak 1 imp/1 l. Vodoměr s impulzním výstupem si lze nechat nainstalovat daným distributorem vody. [28]



Obrázek 12: Vodoměr s impulzním výstupem (převzato z [9])

1.3.4 Měření spotřeby vody pomocí web kamery a softwaru na rozpoznávání obrazu

Měření spotřeby vody pomocí impulzního výstupu je vhodné pro orientační měření. Každý chybně načtený impulz výrazně zvyšuje chybu měření absolutní hodnoty spotřebované vody.

Pro absolutní měření spotřeby vody z vodoměru je možné využít kameru umístěnou nad vodoměrem a pomocí softwaru na rozpoznávání obrazu vyčíst data o spotřebě. Softwarem, který se pro tuto činnost hodí, je například knihovna Open CV, nebo Python Image Processing (PIP). Pro tuto činnost v zásadě vystačí běžná USB web kamera. Software pak může běžet na zařízení typu Raspberry Pi.[28]

2 Arduino

Arduino je open-source platforma založená na zařízení jež obsahuje mikrokontrolér, vstupně výstupní piny a vlastní vývojové prostředí. Pro programování využívá vlastní vývojové prostředí, které je odvozeno z jazyka C. Výhoda této platformy je možnost snadného vývoje prototypových zařízení. V neposlední řadě je výhodou široce zaměřená komunita uživatelů využívající Arduino. Arduino může být široce využito v mnoha aplikacích jako je robotika, automatizace atd. Logo Arduino Open-Source komunity je na obrázku 13.



Obrázek 13: Logo Arduino (převzato z [10])

Jelikož je Arduino otevřený projekt, existuje nepřehledné množství variant desek Arduino. Schémata jsou volně dostupná, a tak není problém vyrobit si vlastní desku určenou přímo na míru dané aplikace.

Další možností je vyzkoušení aplikace na platformě Arduino a nahrání výsledného programu do mikrokontroléru, jež bude využit v konkrétním zařízení. V tomto případě se tedy využije Arduino jako programátor pro dané čipy ATiny, AVR atd.

Oficiální distribuce prototypových desek arduino nabízí mnoho variant, které se liší použitým mikrokontrolérem, počtem vstupů, výstupů, velikostí paměti a jejich rozměry.

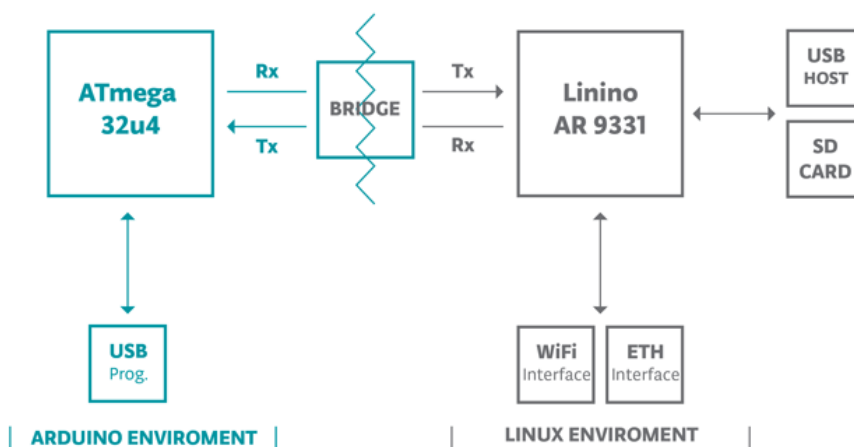
V následujícím textu bude blíže popsána varianta vývojové desky Arduino YÚN (viz obrázek 14), která je součástí měřicího modulu, ostatní verze se v některých parametrech mohou lišit.[34]



Obrázek 14: Arduino YÚN (převzato z [11])

2.1 Arduino YÚN - hardware

Arduino YÚN je vývojová deska, jenž obsahuje mikrokontrolér s ATmega32u4 a počítač typu SoC na jedné desce. Komunikaci mezi mikrokontrolérem a SoC obstarává interface nazývaný Bridge.



Obrázek 15: Arduino YÚN - komunikace mezi MCU a SoC (převzato z [11])

Základní technické parametry zařízení jsou uvedeny v tabulce 10. Díky tomu, že se jedná o otevřený projekt, jsou všechna schémata volně k dispozici ke stažení. Schéma zapojení Arduino YÚN je (ke stažení viz [35]).

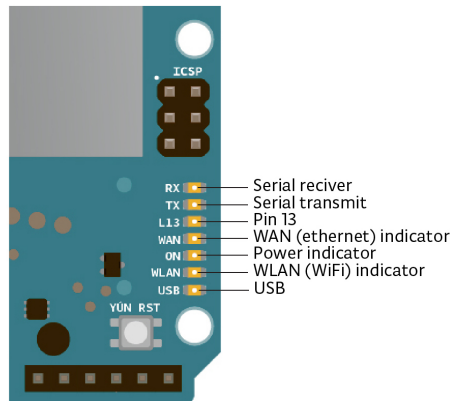
Tabulka 10: Základní parametry Arduino YÚN

Mikroprocesor	ATmega32u4
Pracovní napětí	5 V
Digitální I/O piny	20
PWM kanálů	7
Analogové vstupy	12
DC proud přes I/O piny	40 mA
DC proud přes 3.3 V piny	50 mA
Flash paměť	32 KB (ATmega32u4), 4 KB využity pro bootloader
SRAM	2,5 KB (ATmega32u4)
EEPROM	1 KB (ATmega32u4)
Oscilátor	16 MHz
Procesor pro Linux OS	Atheros AR9331, MIPS, 400MHz
Pracovní napětí	3,3 V
RAM	64 MB DDR2
Flash paměť	16 MB
Ethernet	IEEE 802.3 10/100Mbit/s
WiFi	IEEE 802.11b/g/n
USB	Type-A 2.0
Čtečka paměťových karet	Micro-SD
Orientační spotřeba	300 mA

Arduino YÚN se v základní konfiguraci napájí 5 V zdrojem. Zdroj může být připojen buď přes micro USB konektor, nebo přes vstupní piny oznažený jako Vin a GND. Arduino lze případně rozšířit o PoE

napájecí modul.

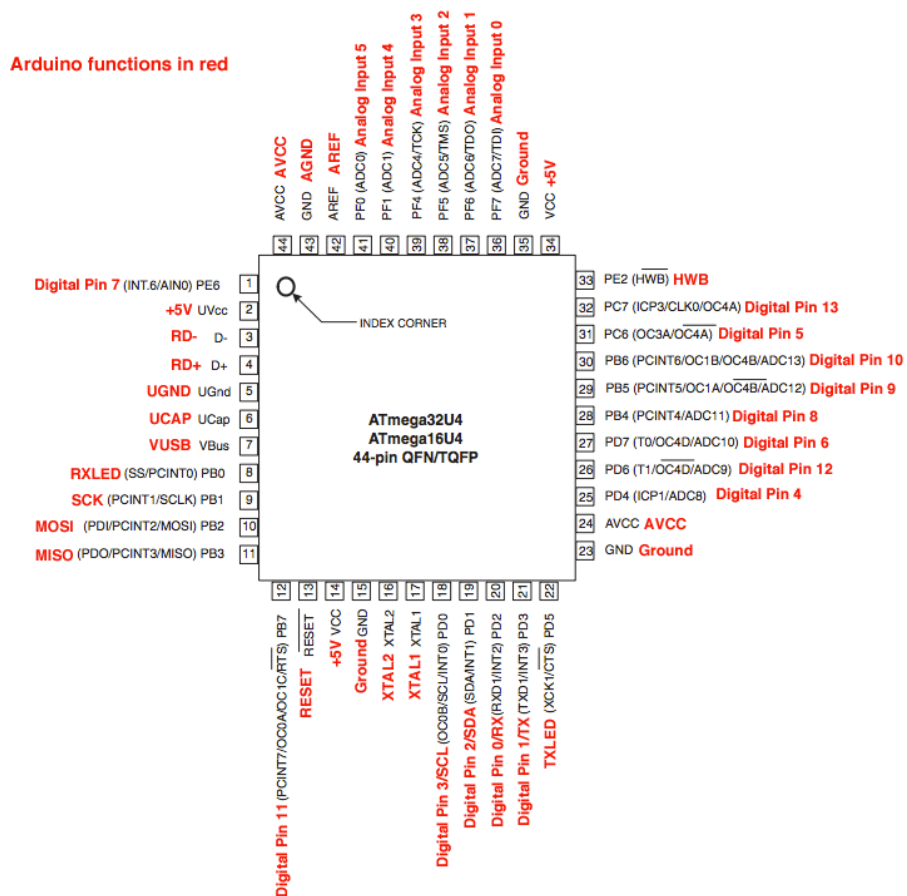
Přímo na desce Arduina se nachází stavové diody, jejichž význam je patrný z obrázku 16.



Obrázek 16: Stavové LED (převzato z [11])

Vstupy a výstupy

Mikrokontrolér obsahuje 20 digitálních vstupně/výstupních pinů (7 z nich může být použito pro výstup PWM signálu a 12 z nich může být použito pro analogové vstupy). Základní zapojení vstupů a výstupů samotného mikrokontroléru ATmega 32u4 lze vidět na obrázku níže.



Obrázek 17: ATmega32u4 - zapojení I/O pinů (převzato z [12])

Každý z 20 pinů může být využit jako digitální vstup, nebo výstup operující s TTL logikou. Kromě toho mají některé piny přiřazenou speciální funkci, viz tabulka 11. Každým pinem může procházet maximální proud o velikosti 40 mA. K jednotlivým pinům lze softwarově připojit pull up rezistory o velikosti 20-50 kOhm, které slouží pro udržení definované logické úrovně. [36, 34]

Tabulka 11: Tabulka I/O pinů Arduino

Označení	Číslo pinu	Popis
Serial(Rx, Tx)	0, 1	sériové rozhraní UART
TWI (SDA,SCL)	2, 3	rozhraní pro I2C komunikaci
External interrupts	0, 1, 2, 3, 7	externí hardwarové přerušení
PWM	3, 5, 6, 9, 10, 11, 13	8-bitová pulzně šířková modulace(PWM)
SPI	ICSP konektor	rozhraní SPI
LED	13	LED dioda připojená k pinu 13
Analog inputs	A0-A5, A6-A11(z digitálních vstupů)	analogový vstup, 10 - bitové rozlišení (1024 hodnot)
AREF		referenční pro analogové vstupy
Reset		pro reset mikrokontroléru

Arduino obsahuje 10 bitový AD převodník pro celkem 6 vstupů (A0-A5). Převádí tedy vstupní napětí v rozsahu 0-5 V na digitální číslo typu integer nabývající hodnoty 0-1023. Čtení analogového vstupu se provádí příkazem `analogRead()`, kdy se do závorek vkládá číslo analogového vstupu.[37, 34]

Arduino YÚN nemá klasický analogový výstup, ale výstup ve formě signálu PWM. PWM umožňuje změnou střídavy měnit střední hodnotu výstupního napětí. Toho se nejčastěji využívá u regulace otáček DC motorů, regulaci jasu LED čipů a v mnoha dalších aplikacích.

2.2 Software

Vývojáři Arduino IDE software se inspirovali nástrojem nazývaným Wiring (programové prostředí, vývojové prostředí a prototypová deska s AVR mikrokontrolérem) a projektem Processing (programovací jazyk a IDE), který byl vyvinut na MIT, a to především pro výuku programování.

Vlastní Arduino IDE je napsáno v jazyce Java a lze spustit pod jakýmkoliv operačním systémem. Programovací jazyk v Arduino IDE je podobný jazyku C. Vlastnímu programu pro Arduino se pak v Arduino komunitě říká Sketch.

Arduino IDE obsahuje kromě textového editoru pro psaní programu také funkce pro nastavení komunikace s deskou Arduino, kontrolu programu, funkci pro nahrání programu do mikroprocesoru a okno Serial monitor pro sériovou komunikaci s Arduinem.

Příklad programu pro blikání LED diody napsaný v Arduino IDE je na obrázku 18. [34]



Obrázek 18: Ukázka programu v Arduino IDE

Softwarové knihovny

V programech pro Arduino se může využívat veliké množství knihoven - uživatelských programových balíčků, které obsahují předprogramované funkce (třídy) a dokáží tak usnadnit práci při psaní programu. Některé základní knihovny jsou připraveny k použití přímo v Arduino IDE a další lze doinstalovat, případně si vytvořit pro konkrétní aplikaci knihovnu vlastní. Například knihovna TFT library slouží pro obsluhu TFT displejů. GSM library pak pro obsluhu GSM modulů. Existuje i knihovna pro obsluhu různých protokolů jako je Modbus, či CAN, dále pak knihovny pro obsluhu inteligentních senzorů, apod. [34]

Přidání knihovny do programu se provede pomocí příkazu `include` se jménem knihovny.

2.3 Arduino - Shields

Výhodou platformy Arduino je možnost rozšíření o další desky, nazývané shields které mohou doplnit určitou funkci (komunikaci, měření atd.). Arduino shields jsou většinou konstruované tak, aby se daly snadno připojit k základní desce Arduino, vždy je však potřeba zkontrolovat kompatibilitu s danou verzí Arduina, případně pro danou verzi shield upravit. Pro Arduino shields většinou existuje i knihovna pro Arduino IDE, která se stará o programovou implementaci. [34]

Arduino shields, ať už oficiálních, tak i neoficiálních, je celá řada. Například GSM shield umožňuje arduinu využívat mobilní síť GSM. Aplikace mohou být různé, například odesílání naměřených dat prostřednictvím SMS, nebo dálkově zapínat/vypínat spotřebiče v domácnosti. Mezi další běžně dostupné shieldy patří:

- RFID/NFC shield - slouží k čtení bezkontaktních karet a čipů.
- Xbee shield - doplní Arduino o bezdrátový modul Xbee.
- TFT touch shield - umožňuje rozšíření aplikace o dotykový displej.
- Motor shield - slouží k ovládání krokových motorků.

3 Sběrnice a protokoly

V této kapitole se zaměřím na komunikační protokoly a sběrnice, především však ty, které lze využít v měřicím modulu s deskou Arduino a pro komunikaci se senzory.

V praxi je často potřeba kombinovat různé sběrnice a protokoly. Důležitým předpokladem pro návrh je tedy znalost základních komunikačních protokolů a sběrnic. Pojem sběrnice a protokol můžeme definovat takto: sběrnice je skupina signálových vodičů pro sdílení a řízení dat mezi dvěma a více elektronickými zařízeními. Jejich přenos se pak řídí stanoveným protokolem. [26, 34]

Vytvořit nebo implementovat funkční sběrnici a protokol je velmi složité. Z tohoto důvodu se vytváří model vrstev, kdy každá vrstva má na starosti určitou funkční část a dochází tak k dekompozici problému. Nejznámější referenční model se nazývá OSI model, který definuje následujících sedm vrstev:

- Fyzická vrstva - stará se o přenos jednotlivých bitů
- Linková vrstva - zabezpečuje spojení a přenos rámců
- Síťová vrstva - stará se o směrování v síti a síťové adresování, zabezpečuje přenos paketů
- Transportní vrstva - zajišťuje přenos dat mezi koncovými uzly
- Relační vrstva - zajišťuje sestavení, řízení a zrušení relací
- Prezentační vrstva - zajišťuje kódování znaků
- Aplikační vrstva - obsahuje jádra aplikací [38]

Zřídka jsou vrstvy OSI modelu v daném komunikačním standardu implementovány všechny a mnohdy pracují jen s některými vrstvami. Například sběrnice RS 232 a RS 485 komunikuje pouze na fyzické vrstvě a o zabezpečení přenosu se pak stará některý z komunikačních protokolů, např. Modbus. [34]

3.1 UART

UART je základní a nejjednodušší způsob komunikace po sériové lince. Jedná se o asynchronní přenos, takže vysílač i přijímač má vlastní generátor hodinového signálu. Je možné u něj řídit rychlost hodinového signálu, délku paketu, počet stop bitů, paritu. UART používá pouze dva piny: Rx pro příjem a Tx pro vysílání. Klidová úroveň na lince je log 1. Vysílání začne tzv start bitem, kdy dojde ke změně stavu z log 1 na log 0. Poté následuje vysílání daného bajtu ve formě jednotlivých bitů, vysílání je pak zakončené stop bitem ve formě log 1.[39, 34]

Arduino pracuje s úrovní signálu 5 V. Arduino využívá také sériovou linku pro komunikaci s počítačem přes USB. Standardně obsahuje Arduino jednu sériovou linku, další sériové linky lze implementovat pomocí softwarové knihovny.

3.2 RS232

Sběrnice RS 232 slouží pro spojení typu bod - bod. Podporuje synchronní i asynchronní komunikaci. Původně byla sběrnice RS232 určena pro komunikaci mezi terminálem (DTE) a modemem (DCE), v současné době se však nejvíce používá pro spojení dvou DTE zařízení. [40, 34]

Maximální doporučená délka kabelu je 15 m. Rychlost sběrnice může být maximálně 115 kbit/s v závislosti na délce a kvalitě vedení. Pro přenos bitů se používá inverzní logika, kdy napětí -3 V až -25 V reprezentuje log. 1 a napětí +3 V až +25 V reprezentuje log. 0. [40, 34]

Pro komunikaci elektroměrové optické hlavice s PC nebo mikrokontrolérem Arduino je potřeba převodník RS232 na USB, nebo převodník RS232 na UART. Pro převodníky RS232 na USB se vyrábějí převodníky s FTDI čipy v kompaktním pouzdře. Vzhledem k masivní výrobě těchto převodníků a příznivé ceně se nevyplatí kusová výroba a je výhodnější si kompletní převodník zakoupit.

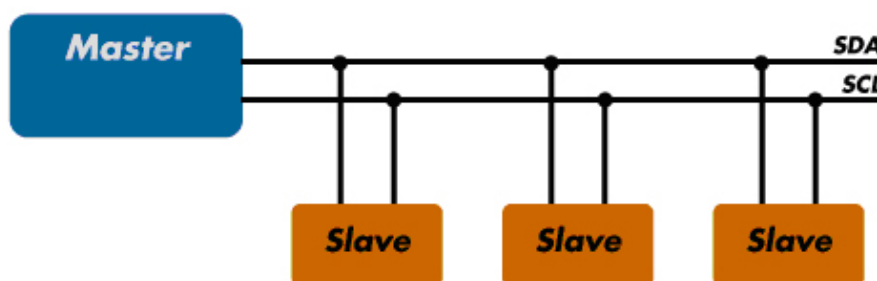
3.3 RS485

Jedná se o sběrnice rozhraní, jež se ve velkém množství využívá v průmyslu pro sériovou komunikaci mezi jednotlivými zařízeními. Komunikace je na principu master - slave. Na tento typ rozhraní je možno připojit maximálně 32 zařízení a jeho délka může být maximálně 1,2 km. Na každý konec tohoto rozhraní je třeba připojit zakončovací odpor 120 Ohm (terminátor). Kabel musí být veden od jednoho zařízení k druhému s maximální délkou odbočky od sběrnice 25 cm. Z hlediska provozu se jedná o poloduplexní asynchronní přenos s maximální rychlostí 10 Mb/s v závislosti na délce a kvalitě vedení. Napětí -2 V až -6 V reprezentuje log. 1 a napětí 2 V až 6 V reprezentuje log. 0. [40, 34]

Toto rozhraní využívají některé komunikační protokoly používané v automatizaci budov jako například Modbus.

3.4 I2C

Sběrnice I2C se používá pro komunikaci mezi integrovanými obvody. Sběrnice se skládá z dvojice vodičů SDA a SCL, ke kterým se v praxi přidává signálová zem. Zařízeními s I2C komunikací se někdy také říká TWI devices. Komunikace probíhá, jak je naznačeno v obrázku 19, v režimu master-slave, kdy jedno master zařízení řídí veškeré datové toky na sběrnici, díky tomu nedochází na sběrnici ke kolizím. [41, 13, 34]



Obrázek 19: I2C sběrnice(převzato z [13])

Vodič SDA (serial data) slouží pro obousměrný přenos dat a vodič SCL (serial clock) slouží pro přenos signálových hodin. Oba piny SCL a SDA jsou připojeny přes pull up rezistory, které zajistí, že v klidovém stavu bude na obou pinech log 1.

Každé zařízení I2C obsluhuje sedmibitovou adresu, takže na jednu sběrnici lze připojit až 128 zařízení. V praxi však jsou některé adresy rezervovány pro speciální funkce (např. broadcast). [13, 34]

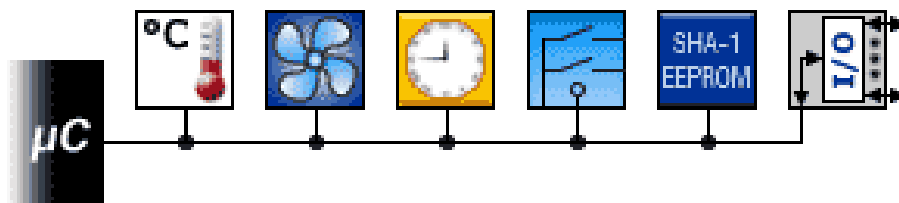


Obrázek 20: I2C sběrnice - datový rámeček(převzato z [13])

Formát přenášeného rámce je zobrazen na obrázku 20. Vysílání na sběrnici začíná start bitem, který vypadá tak, že signál SDA přejde do log 0 přičemž signál SCL zůstane nějakou dobu (v závislosti na přenosové rychlosti) v log 1. Přenášená hodnota bitu je pak platná s náběžnou hranou signálu SCL. Po start bitu následuje adresa slave zařízení, s kterým chce master komunikovat a dále bit určující, zda má zařízení Master data vysílat nebo přijímat, signál ACK pak slouží pro potvrzení příjmu dat. [13, 34]

3.5 1-Wire

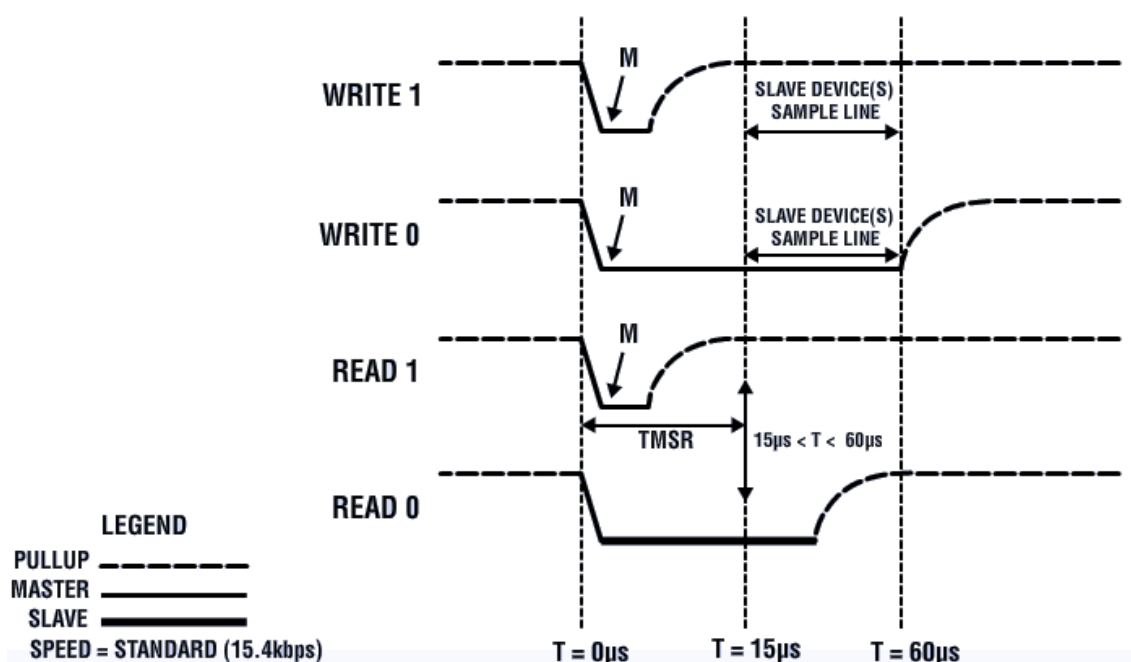
Sběrnice 1-Wire je navržena firmou Dallas Semiconductor. Tato sběrnice umožňuje připojit na jeden signálový vodič a zem několik senzorů a dalších obvodů, jak naznačuje obrázek 21.



Obrázek 21: 1-wire sběrnice (převzato z [14])

Nejnámější aplikací 1-Wire jsou přístupové systémy pomocí iButton klíčenky. Firma Dallas nabízí několik integrovaných obvodů a senzorů využívající 1-Wire komunikaci, například AD převodníky, čtečky a paměti EEPROM pro přístupové systémy atd.

Každé 1-Wire zařízení má z výroby v paměti ROM unikátní 64 bitovou adresu, která toto zařízení od ostatní na sběrnici odlišuje. Komunikace po sběrnici probíhá v režimu master - slave. Master, pokud chce vysílat, zahajuje spojení reset bitem, kdy stáhne úroveň datového vodiče do log 0 na dobu 480 μ s. Pak sběrnici uvolní a naslouchá, pokud je na sběrnici nějaké zařízení, tak opět stáhne datový vodič do log 0 na dobu 60 - 240 μ s. Poté probíhá komunikace v časových slotech, kdy v jednom časovém slotu 60-120 μ s je přenesena informace o velikosti jeden bit, mezi časovými sloty musí být mezera min. 1 μ s.[14, 34]



Obrázek 22: 1-wire, zápis a čtení(převzato z [14])

3.6 Ethernet

Ethernet je technologie nejčastěji používaná v lokálních počítačových sítích (LAN) vznikajících od počátku sedmdesátých let. Ethernet je standardizován pomocí norem řady IEEE802.3, které definují jednotlivé varianty lišící se přenosovou rychlostí, použitým přenosovým médiem a řízením přístupu k mediu.

3.6.1 Řízení přístupu k přenosovému mediu

Nejčastěji se využívá metoda CSMA/CD. Tato metoda funguje tak, že stanice která chce vysílat sleduje přenosový kanál, pokud je v klidu, tak začne vysílat rámeček. Pokud během vysílání zjistí stanice příchod cizího signálu (od stanice která začala vysílat ve stejném časovém intervalu), tak nastane kolize. Kolize se řeší tak, že stanice vyšle informaci „jam“ o 32 bitech, že došlo ke kolizi a všechny stanice se tak odmlčí a každá stanice vygeneruje náhodný časový interval po kterém začne znovu vysílat. Aby byla kolize detekovatelná musí mít rámeček minimální délku 64 B (max 1518 B). Při opakované kolizi se pak generovaný časový interval, při kterém stanice nevysílá, exponenciálně zvětšuje. Těchto pokusů však provede maximálně 16, při dalším pokusu dojde k zahazení rámce. [42, 34]

Formát rámce se skládá z několika částí, viz tabulka 12.

Tabulka 12: Formát MAC rámce

7	1	6	6	2	Proměnná délka		4
Preamble	SD	DA	SA	Length	Information	Pad	FCS

- Preamble slouží k synchronizaci přijímače
- SD (start delimiter) umožňuje rozpoznat začátek rámce
- DA (destination address) cílová adresa
- SA (source address) adresa odesílatele
- Length určuje délku rámce
- Information obsahuje samotná data
- Pad (padding) zajišťuje minimální délku rámce
- FCS - CRC - detekce chyb, kontrolní součet [38]

3.6.2 Ethernet v průmyslových aplikacích

Ethernet původně vznikl jako kancelářský systém. Nicméně díky jeho masivnímu rozšíření se začal používat i pro průmyslové aplikace, přestože pro to není příliš vhodný. Nevýhodou je jeho nedeterminismus, z čehož vyplývá neschopnost práce v reálném čase.

Z tohoto důvodu se začaly využívat metody, které pomocí nadřazeného protokolu, nebo způsobu komunikace zabrání vzniku kolize (např. master-slave komunikace, Token-Passing, Modbus TCP/IP, Ethernet Powerlink ad.). [38, 34]

3.7 Bezdrátový přenos

Pro přenos informací ze senzorů do centrální vyhodnocovací jednotky nelze vždy zabezpečit standardní kabelové připojení. Pro tyto účely lze využít některou z běžně používaných bezdrátových technologií. V některých případech je vhodné, aby bezdrátová technologie měla malou spotřebu energie s ohledem na bateriové napájení. Dále je potřeba vybrat technologii takovou, která bude mít požadovaný dosah s dobrou prostupností skrz stavební materiály budovy.

Vhodných bezdrátových technologií pro komunikaci se senzory v budově je například technologie Xbee. Perspektivní v tomto směru je i technologie EnOcean, která umožňuje bezdrátovou a bez bateriovou komunikaci, kdy samotný senzor je energeticky soběstačný (Energy Harvesting).

Další možností je využití samostatných rádiových modulů komunikujících s mikroprocesorem. V tomto směru je s ohledem na cenu perspektivní rádiový modul RFM12B.

4 Senzory

Senzory jsou prvky, který převádějí informaci obsaženou v jistém typu energie na informaci s jiným typem energie, kterou dokážeme patřičně zpracovat.[43]

V tomto konkrétním případě mají za úkol zpracovat informace z měřičů energií a dále pak získat stavové informace o teplotě, vlhkosti atd.

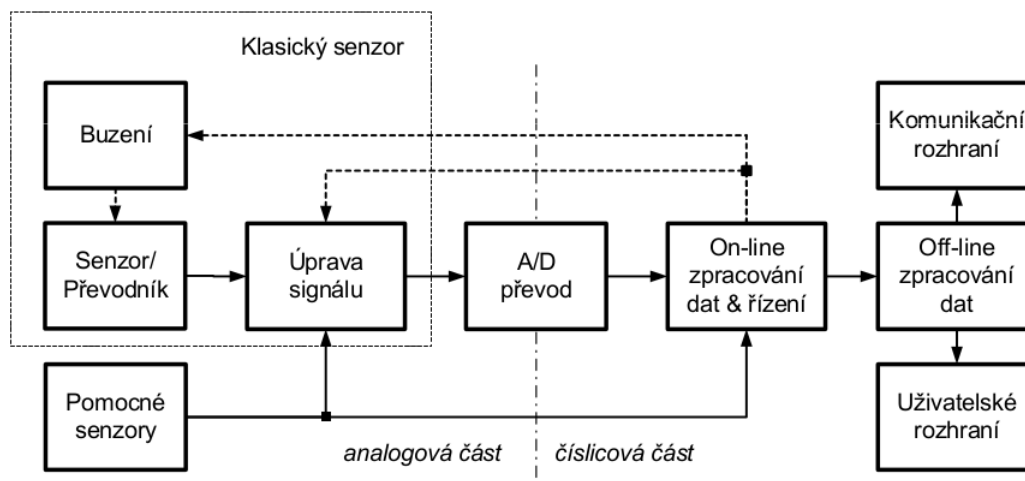
Z pohledu zpracování signálů dělíme senzory na tyto tři druhy.:

- analogové - s nelineární charakteristikou
- analogové - s lineární charakteristikou
- digitální - inteligentní senzory, někdy označované jako smart senzory [43, 34]

První varianta s nelineárním vztahem mezi vstupní a výstupní veličinou je z pohledu senzoru nejjednodušší. příkladem takové součástky může být například termistor. Obvykle se pak pro výpočet výsledné hodnoty hledá vhodná funkce, která bude v potřebném rozsahu odpovídat dané charakteristice senzoru.[43, 34]

Druhá varianta senzoru s lineární charakteristikou již ke své funkci obvykle potřebuje pomocný obvod, který provede potřebnou kompenzaci nelinearity senzoru. Existuje více metod pro linearizaci, jejímž výsledkem je pak lineární závislost vstupní hodnoty na hodnotě výstupní.[43]

Poslední varianta inteligentního senzoru (označovaný také jako smart senzor) je z hlediska implementace do nadřazených řídicích systémů nejvýhodnější. Principiální blokové schéma inteligentního senzoru je na obrázku 23.[34]



Obrázek 23: Inteligentní senzor - blokové schéma (převzato z [15])

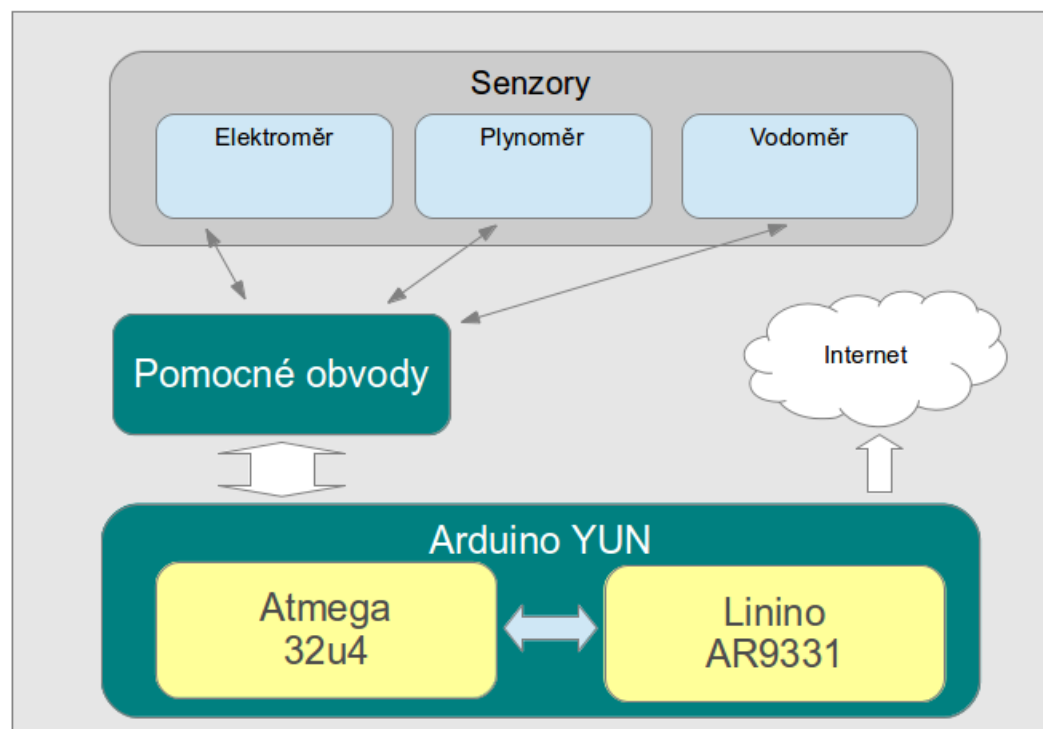
Základním prvkem inteligentního senzoru je A/D převodník, který převede již upravený analogový signál ze senzoru na signál digitální, který pak může být dále zpracován do vhodné podoby tak, aby se data dala vhodně přenést konkrétním komunikačním protokolem. Dále inteligentní prvek může obsahovat pomocné senzory, které slouží pro kompenzaci naměřené hodnoty vzhledem k vnějším podmínkám.[34]

Aplikací konkrétních senzorů se budu zabývat v následujících kapitolách.

5 Měřicí modul

5.1 Architektura systému

Architektura celého systému vychází z Projektu 2, kde jsem se zabýval testováním systému pro vyčítání informací o spotřebě na open-source platformách Arduino a Raspberry Pi (viz Projekt 2 [28]). Tyto platformy jsem si vybral vzhledem k jejich snadné implementaci, nízké ceně a široké komunitě vývojářů. Systém je navržen tak, aby v případě potřeby, byl snadno překonfigurovatelný ať už po stránce hardwarové, tak i softwarové. Během návrhu a testování jsem vyzkoušel různé možnosti konfigurací. Výsledná konfigurace je vyobrazena na obrázku 24, vychází z požadavku na malé rozměry měřicího modulu, vestavěného webového serveru a dostatku potenciálu pro softwarové řešení. Základní prvky systému jsou tvořeny:

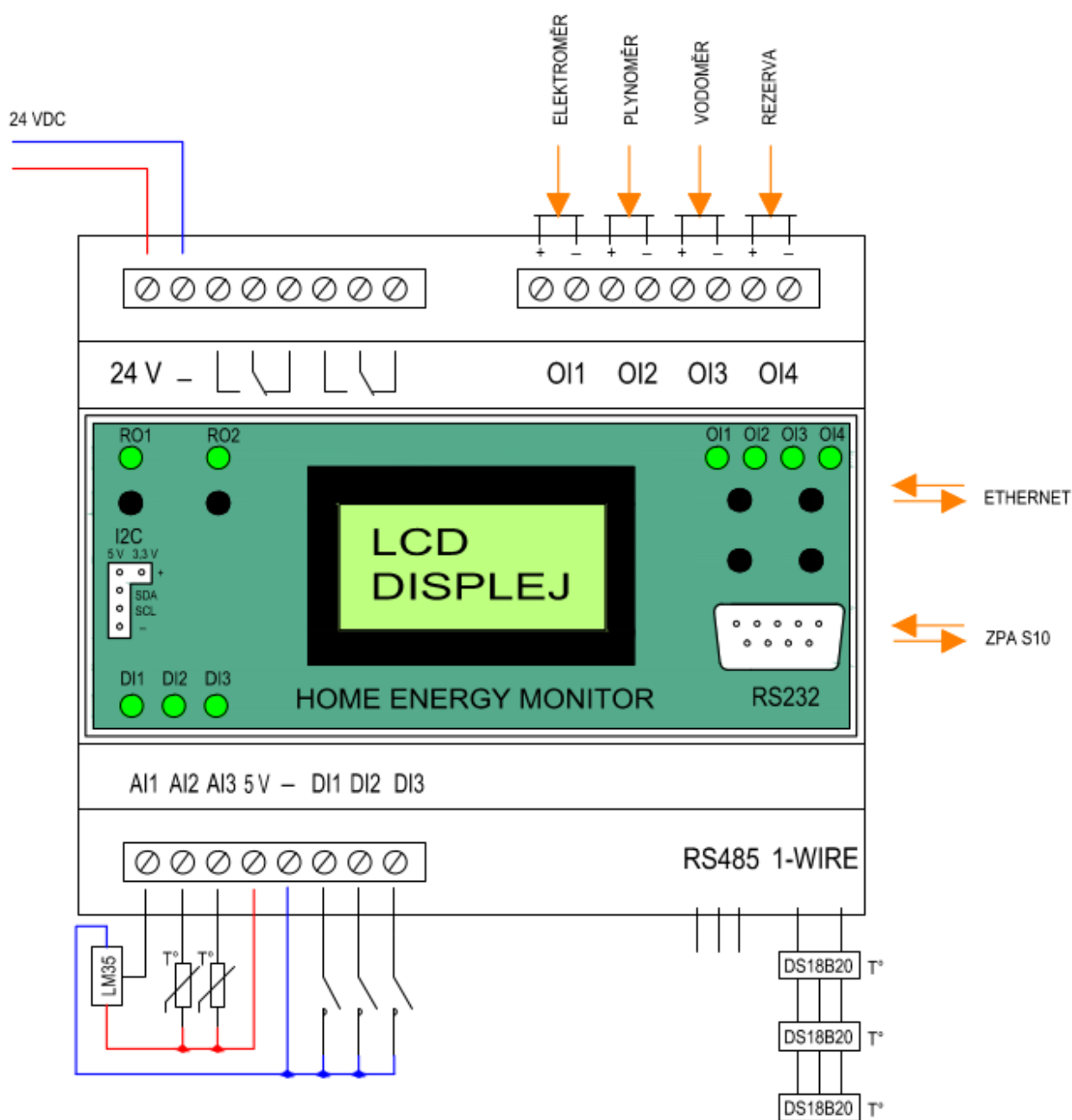


Obrázek 24: Konfigurace V2

- Sensory s pomocnými obvody .
 - vyčítání impulzů z elektroměru, plynoměru, vodoměru
 - měření teploty, relativní vlhkosti, osvětlení
- Sběrníkovými systémy.
 - 1-Wire pro měření teploty, relativní vlhkosti
 - I2C pro vnitřní obvody a další senzory
- Měřicím modulem
 - mikrokontrolerová část (Arduino YÚN)
 - SoC část s operačním systémem (Arduino YÚN)
 - pomocné obvody

5.2 Návrh měřicího modulu

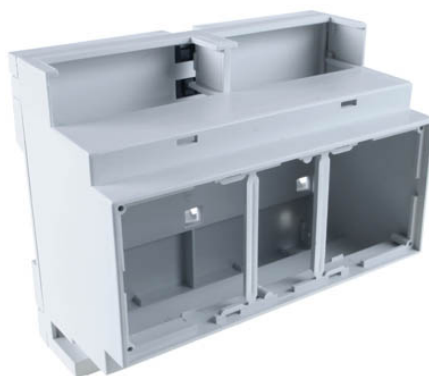
Ideový výkres měřicího modulu je vidět na obrázku 25. V dalších kapitolách postupně popíši návrh jednotlivých částí měřicího modulu, výběr komponent a praktickou realizaci.



Obrázek 25: Měřicí modul

5.3 Velikost měřicího modulu a výběr krabičky

Velikost modulu byla ovlivněna několika parametry a omezeními. Pevně dané jsou rozměry Arduina YÚN jenž jsou 73 x 53 mm. Konstrukce krabičky pak byla zvolena tak, aby umožňovala připojení na DIN lištu do standardní rozvodnicové skříně společně s dalšími moduly. Tyto krabičky se vyrábí v jmenovitých velikostech v násobcích 1M (1 modul), kde písmeno M označuje modul o velikosti 17,5 mm. Pro měřicí modul byla vybrána krabička Modulbox H53 o šíři 6M.

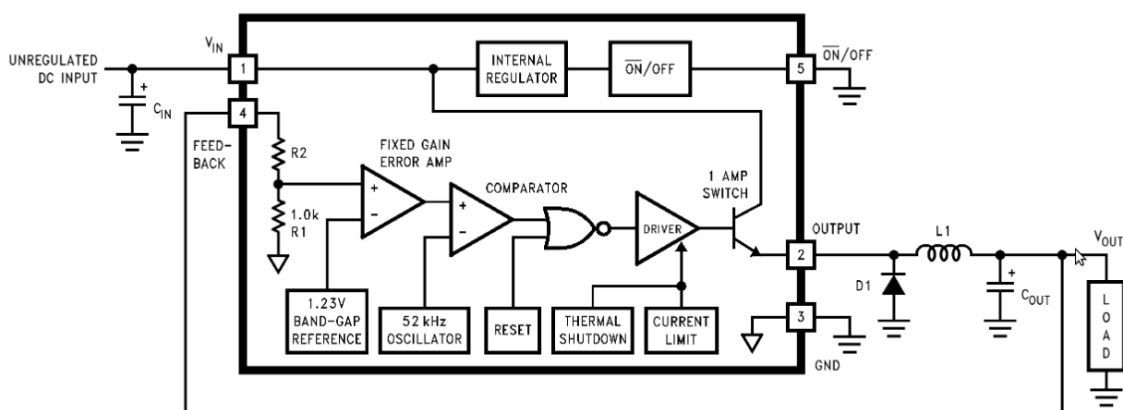


Obrázek 26: Krabička pro měřicí modul (převzato z [16])

Do této krabičky jsou umístěny dvě DPS. První o velikosti 102 x 85 mm, která obsahuje vstupní piny (male) pro nasunutí Arduino YÚN. Druhá o velikosti 102 x 58 mm, která bude usazena ve vrchní části krabičky.

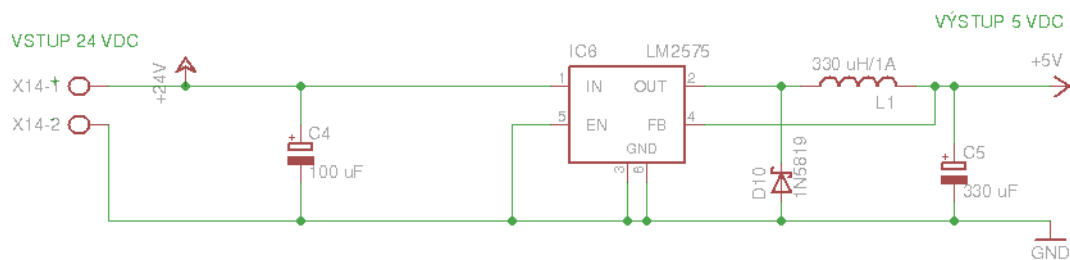
5.4 Návrh napájení

Napájení celého zařízení bude stejnosměrným zdrojem o velikosti 24 V/1A. Napětí 24 V přivedené na svorky slouží pro napájení reléových cívek a elektroměrové optické hlavice. Dále je napětí 24 V přivedeno na vstup DC/DC spínaného (step-down) měniče. Výstupní napětí 5 V slouží pro napájení Arduina, pomocných obvodů a senzorů. Napájecí část využívá obvod LM2575, jehož výhodou je minimum přidružených součástek. Blokové schéma obvodu LM2575 je na obrázku 27.



Obrázek 27: Schéma napájení (převzato z [17])

Schéma napájení vycházející z katalogového zapojení je na obrázku 28. Na výstupu je vzhledem k vysoké frekvenci přepínání měniče (až 50 kHz) použita schottkyho dioda. Výstupní tlumivka o velikosti 330 uH je dimenzována na 1 A, což je pro danou aplikaci dostatečné. [28]

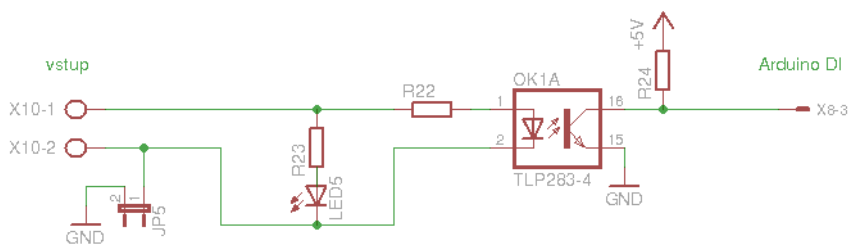


Obrázek 28: Schéma napájení

Přímo na desce Arduina je měnič 5 V na 3.3 V pro napájení vnitřních i vnějších obvodů. Toto napětí je také vyvedeno na výstupní pin. měřicí modul tak bude obsahovat možnost napájení některých senzorů (např. čidlo osvětlení) napětím o velikosti 3,3 V.

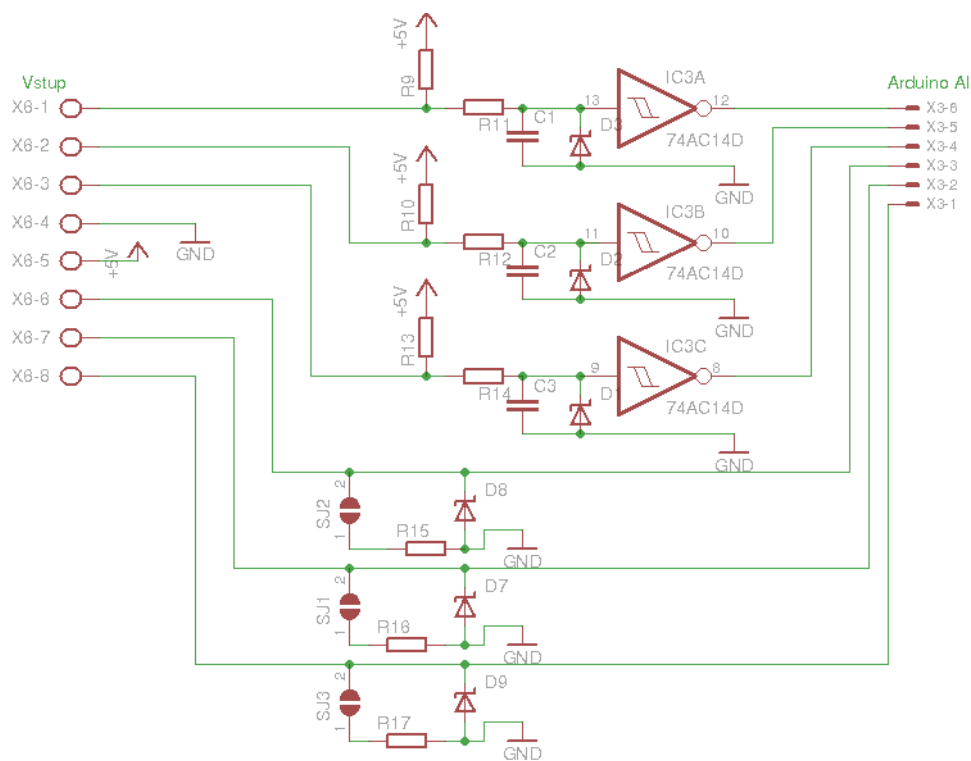
5.5 Impulzní, binární a analogové vstupy

Impulzní vstupy pro senzory měřičů energie jsou galvanicky oddělené od binárních vstupů Arduina. Schéma zapojení galvanicky oddělených impulzních vstupů pomocí optočlenu TLP283 je na obrázku 29. Je-li senzor napájen z měřicího modulu umožňuje propojka JP5 (propojení s GND) ovládat vstup jedním drátem. Měřicí modul obsahuje čtyři takovéto vstupy.



Obrázek 29: Galvanicky oddělené vstupy

Binární vstupy slouží pro připojení tlačítek, magnetických jazýčkových kontaktů atd. Schmittův klopný obvod, rezistory a kondenzátor slouží pro odstranění záchvěvů při přepnutí kontaktu. Jako Schmittův klopný obvod je použit integrovaný obvod 74LS14. Transil na vstupu slouží jako ochrana proti přepětí.

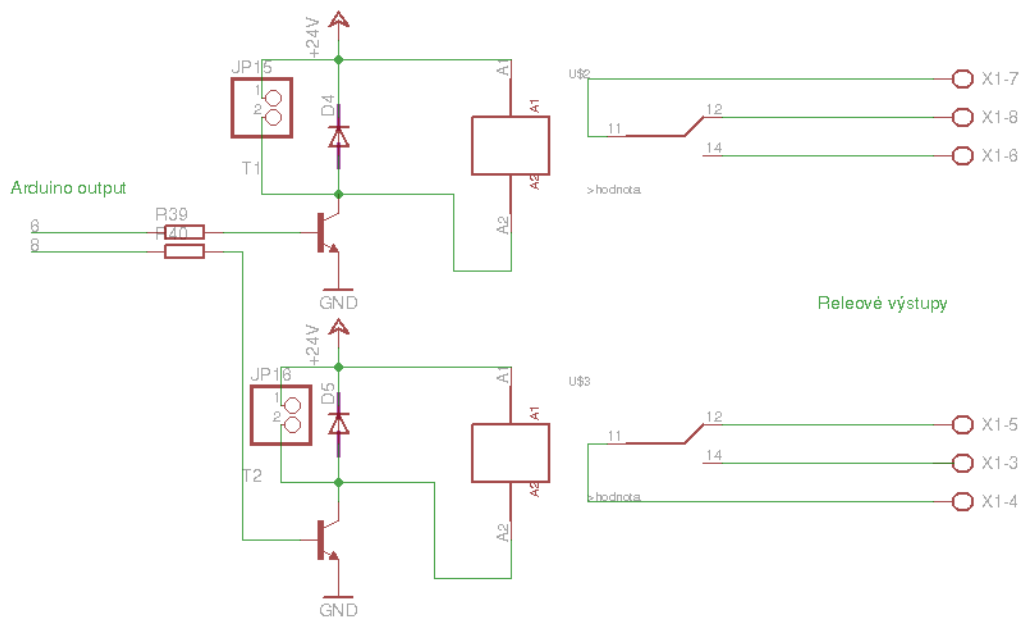


Obrázek 30: Binární a analogové vstupy

Také analogové vstupy jsou vybaveny transily pro ochranu proti přepětí. Propojky SJ slouží pro vřazení 10 kOhm rezistoru mezi vstup a GND. Vřazení propojky je vhodné za předpokladu, že je potřeba na vstup připojit termistor, nebo fotorezistor. Pro jiné aplikace jako například - měření teploty pomocí senzoru LM35, je výhodné nechat propojku vyřazenou. [28]

5.6 Reléové výstupy

Měřicí modul obsahuje kromě vstupů také dva univerzální binární reléové výstupy. Ty mohou být využity například pro spínání alarmu (havarijní funkce) nebo pro časové spínání cirkulačního čerpadla. Arduino má výstupní hodnotu napětí ve stavu log 1 odpovídající 5 V. Proud výstupem je omezen hodnotou 40 mA. Větší hodnota proudu by mohla obvod poškodit. Z tohoto důvodu je potřeba realizovat zapojení, které bude schopno spínat daný kontakt relé. Schéma zapojení reléových výstupů je na obrázku 31.



Obrázek 31: Reléové výstupy

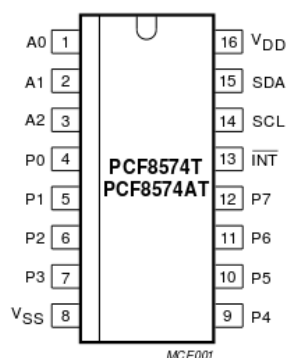
Relé bylo vybráno s ohledem na rozměry, typ FINDER 34.51.7.024.0010. Jedná se o relé s DC cívkou 24 V, odpor cívky je 3350Ω a maximální spínaný proud 6A. Paralelně k cívce relé je v závěrném směru vřazena dioda, která působí jako ochrana proti napět'ovým špičkám. Dále jsou paralelně vyvedeny piny, které slouží k LED signalizaci sepnutého stavu na předním panelu měřicího modulu.



Obrázek 32: Použité relé (převzato z [18])

5.7 Ovládací tlačítka s I/O expandérem

Měřicí modul obsahuje na předním panelu tlačítka pro manuální spínání reléových výstupů a dále pak tlačítka pro změnu režimu zobrazení informací na LCD displeji. Arduino YÚN obsahuje omezený počet I/O pinů. Pokud je potřeba navýšit množství pinů, lze použít I/O expandér PCF8574, který se připojí na I2C sběrnici. Obvod PCF8574 obsahuje celkem 8 I/O pinů. Těchto expandérů lze připojit na jednu I2C sběrnici maximálně 16, tzn. lze rozšířit Arduino až o 128 I/O pinů. Měřicí modul tak lze dále rozšířit o jednotku reléových výstupů, jednotku binárních vstupů atd. Popis pinů obvodu PCF8574 je na obrázku 33. Význam jednotlivých pinů pak v tabulce 13.



Obrázek 33: Popis pinů PCF8574 (převzato z [19])

Tabulka 13: Tabulka pinů PCF8574

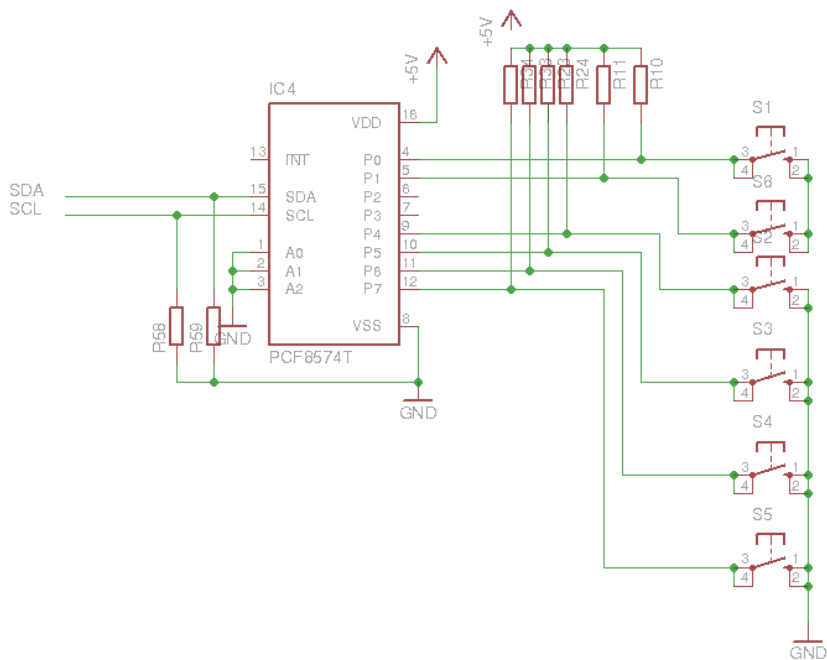
A0 - A2	adresové vstupy
SDA,SCL	I2C sběrnice
P0 - P7	I/O piny
Vdd	napájení +
Vss	napájení, GND

Adresa expandéru je stanovena pomocí pinů A0-A2. Číselné hodnoty, kterých může nabývat, jsou zobrazeny v tabulce 14.

Tabulka 14: Nastavení adresy expandéru

A2	A1	A0	Adresa (dec)	Adresa (hex)
0	0	0	32	20
0	0	1	33	21
0	1	0	34	22
0	1	1	35	23
1	0	0	36	24
1	0	1	37	25
1	1	0	38	26
1	1	1	39	27

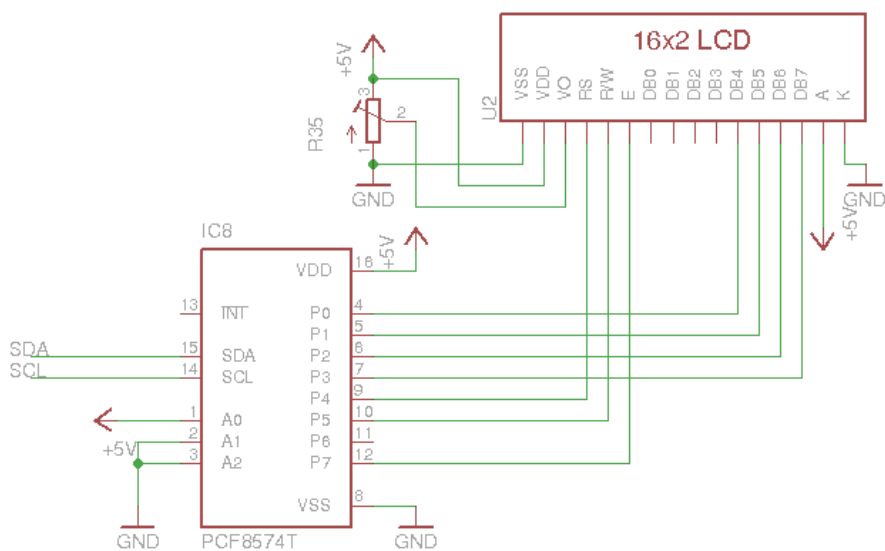
Schéma zapojení vstupních tlačítek přes expandér PCF8574 je na obrázku 34. Ke vstupním pinům expandéru je zapojen pullup odpor pro definování logické úrovně.



Obrázek 34: Tlačítka a PCF8574

5.8 LCD displej

měřicí modul obsahuje displej pro zobrazování základních údajů. K tomuto byl vybrán LCD displej s paralelní komunikací Winstar WH0802 . Displej má dva řádky po osmi znacích. Přímá paralelní komunikace mezi displejem a Arduinem by zabírala zbytečně mnoho pinů. Proto je i v tomto případě použit I2C expandér PCF8574. Schéma zapojení LCD displeje s expandérem PCF8574 je na obrázku 35.



Obrázek 35: LCD displej s expandérem

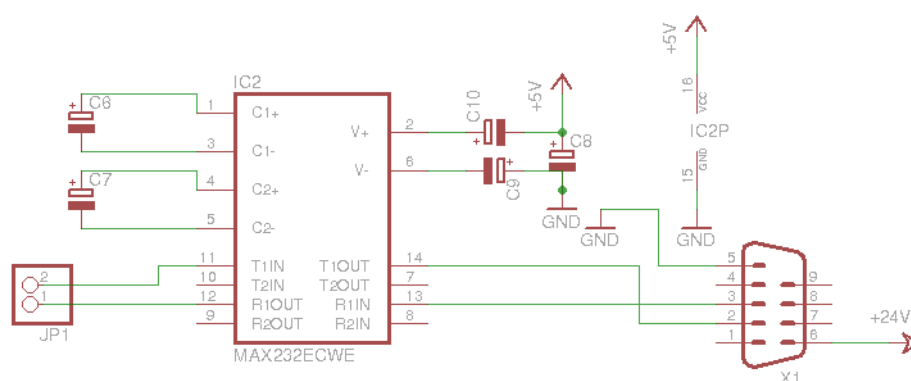
Z obrázku je patrné, že není potřeba propojovat všechny piny LCD displeje, význam jednotlivých pinů LCD displeje je v tabulce 15. Trimr slouží k nastavení kontrastu displeje.

Tabulka 15: Význam pinů LCD displeje

Označení pinu řadiče displeje	Význam	Číslo pinu I2C expandéru
RS	Definuje, zda je zadán znak nebo příkaz	P4
R/W	Určuje, zda se jedná o čtení nebo zápis	P5
E	Povoluje přístup k jednotlivým pinům DB	P4
DB4 - DB7	Bity datového znaku	P0 - P3
Vss	GND	-
Vdd	Napájení + 5 V	-
V0	Pin pro řízení kontrastu	-
A	Anoda (+) podsvícení LED	-
K	Katoda (-) posvícení LED	-

5.9 Převodník UART - RS232

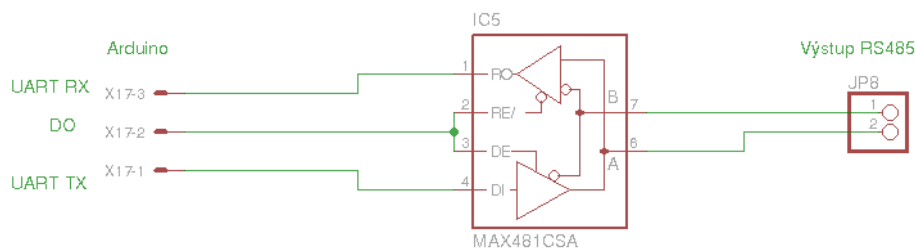
Schéma převodníku je na obr. 36. Hlavní součástí je v tomto schématu obvod MAX232. Obvody elektroměrové optické hlavice potřebují pro svou činnost napájecí napětí 24 V. Konektor pro připojení elektroměrové optické hlavice byl zvolen Cannon 9 na přední části měřicího modulu.



Obrázek 36: Schéma převodníku

5.10 Převodník UART - RS485

Pro začlenění měřicího modulu do rozsáhlejšího systému MaR obsahuje modul sběrnici RS485. Arduino YÚN obsahuje pouze jednu sériovou linku. Další sériová linka je vytvořena pomocí obecných I/O pinů a softwarové knihovny SoftwareSerial. Komunikační protokol komunikující po této lince pak je Modbus. Schéma zapojení převodníku UART na RS485 s pomocí obvodu MAX481 je na obrázku 36.

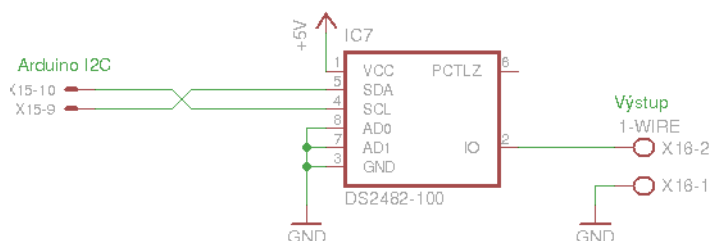


Obrázek 37: Schéma převodníku UART - RS485

5.11 1-Wire driver

Arduino umožňuje připojit 1-Wire senzory přímo na některý z I/O pinů. Obsluhu senzorů pak řeší softwarová knihovna v Arduino. Toto řešení je nejsnazší, nicméně má několik nevýhod mezi které patří: přímé galvanické připojení na vstupní piny, nemožnost řízení proudu do sběrnice („aktivní pullup“).

Dalším možným řešením, které bylo použito, je využít 1-Wire driver DS2482, který umožňuje komunikovat s 1-Wire senzory pomocí I2C sběrnice. Toto řešení umožňuje oproti předchozímu připojit 1-Wire senzory na delší vzdálenost (řádově 200-300 m). Schéma zapojení 1-Wire driveru je na obrázku 38. Piny AD0 a AD1 pak definují adresu I2C zařízení. [28]

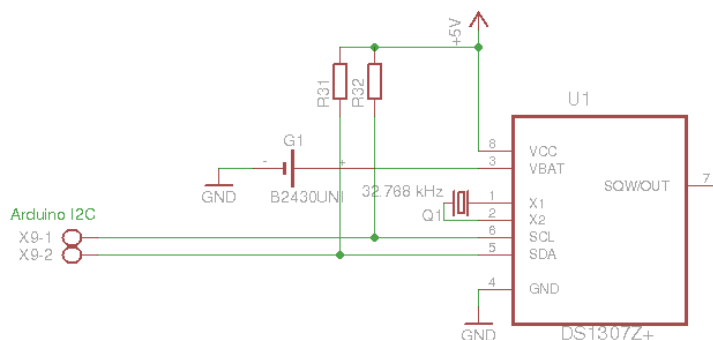


Obrázek 38: 1-wire driver

5.12 Obvod reálného času

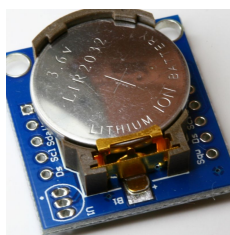
Obvod reálného času (RTC) umožní logovat naměřené hodnoty s časovou značkou. Časovou značku by bylo možné získat z linuxové části desky Arduino YÚN, to však vyžaduje, aby zařízení bylo po každém restartu připojené do internetu a mělo tak přístup k NTP serveru.

RTC obvod využívá integrovaný obvod DS1307, který umožňuje komunikovat pomocí I2C sběrnice. K obvodu DS1307 je připojen krystal a baterie, která udržuje čas i po výpadku napájení.



Obrázek 39: Schéma RTC obvodu

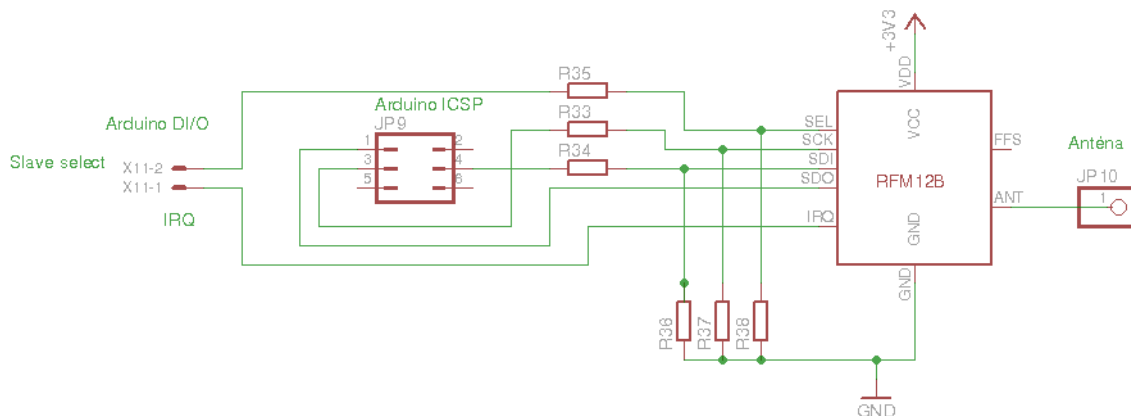
Na trhu lze sehnat již hotový kompaktní RTC obvod (viz obrázek 40). Tento obvod je možno volitelně umístit dovnitř nebo vně měřicího modulu.



Obrázek 40: RTC obvod

5.13 Bezdrátový přijímač

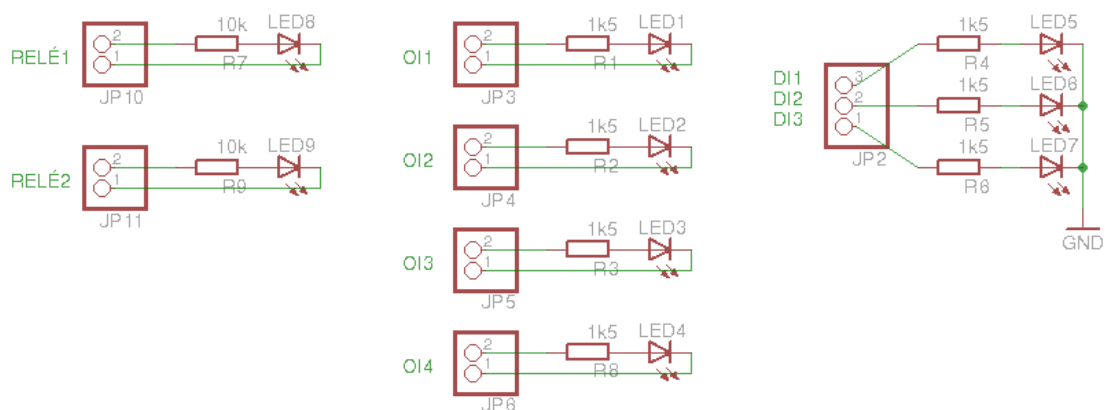
Navrhovaný systém předpokládá, že jsou senzory k měřicímu modulu s Arduinem připojeny kabelem (např. JYSTY 2x2x0,8) hvězdicově, senzory teploty pak s libovolnou topologií. V některých případech však není technicky možné přivést k senzoru kabel. Pro tyto případy bude možné využít bezdrátový přenos pracující v pásmu 868 MHz. Pro bezdrátový přenos je využít přijímač RFM12B, který umožňuje komunikovat rychlostí až 115 kb/s na vzdálenost cca 100 m. Obvod RFM12B je připojen k Arduinu přes rozhraní SPI. Komunikaci pak obstarává softwarová knihovna určená pro tento obvod. Schéma zapojení bezdrátového přijímače je na obrázku 41. Pro frekvenci 868 MHz vychází 1/2 vlná anténa délky 164 mm. Anténa se připojuje přes konektor na boku přístroje [28]



Obrázek 41: Schéma bezdrátového přijímače

5.14 LED signalizace

Pro signalizaci stavu sepnutí jednotlivých vstupů a výstupů jsou na vrchní DPS umístěny nízkopříkonové LED. Schéma zapojení je na obrázku 42. Propojení mezi vrchní a spodní DPS je realizováno flexibilními vodiči a řadovými oboustrannými kolíky s roztečí pinů 2,54 mm.

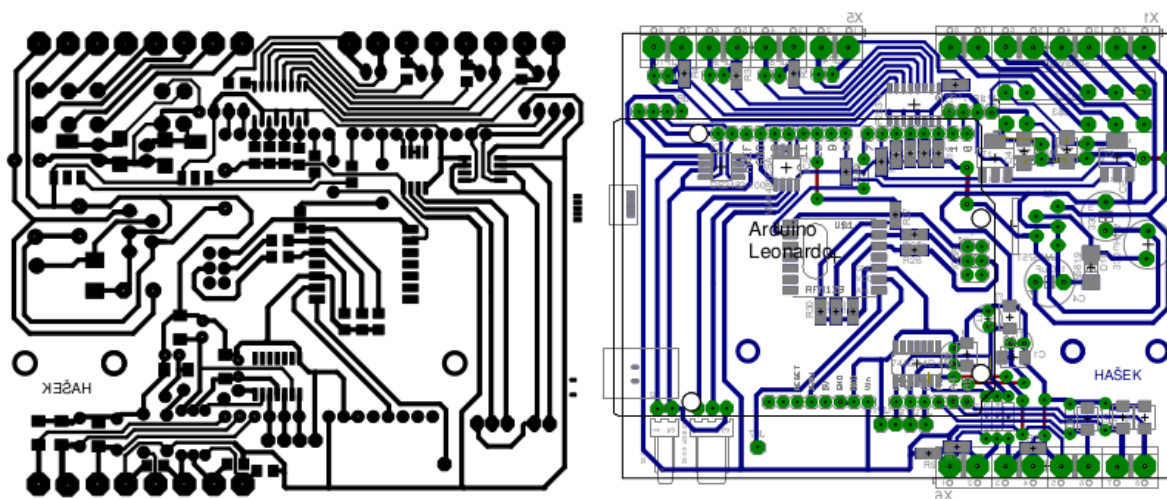


Obrázek 42: LED signalizace

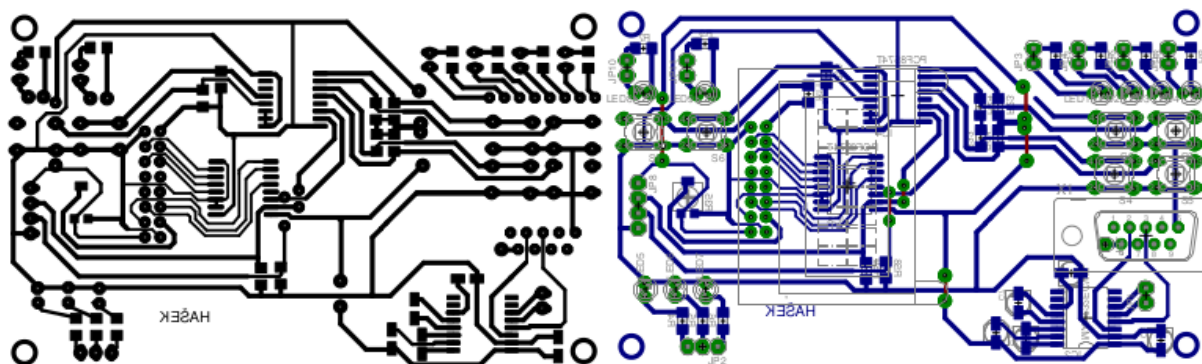
5.15 Výroba DPS

Měřicí modul se skládá ze dvou desek plošných spojů umístěných nad sebou. DPS byly tvořeny v návrhovém softwaru Eagle. Spodní deska obsahuje šroubové svorky pro vstupy a výstupy. Dále pak tvoří základnu pro Arduino YÚN. Obsahuje napájení, vstupní a výstupní obvody, převodník UART-RS485 a 1-Wire driver. Vrchní deska obsahuje LCD displej, tlačítka a obvod převodníku UART-RS232.

Obrazce desek jsou vidět na následujícím obrázku.

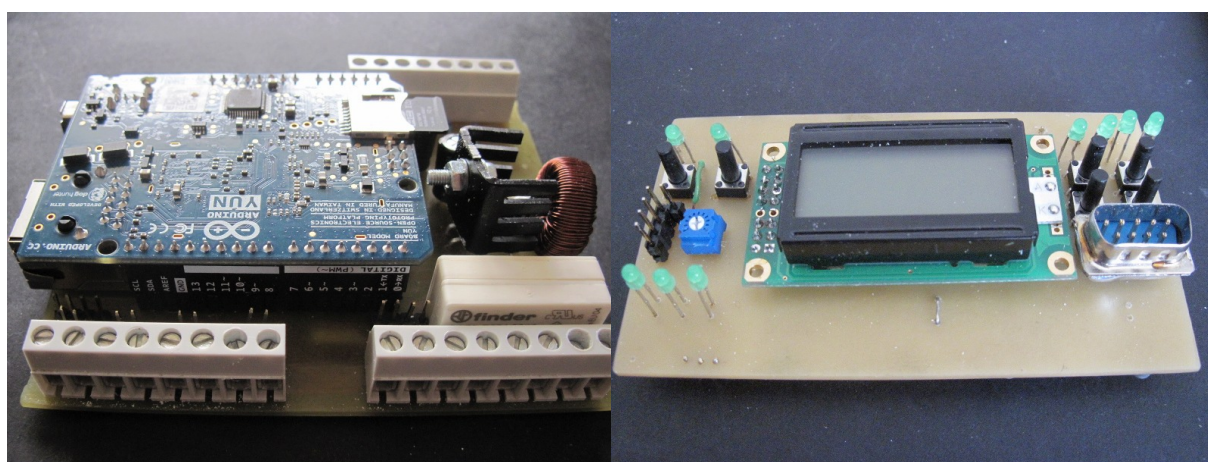


Obrázek 43: Obrazec spodní DPS



Obrázek 44: Obrazec vrchní DPS

Pro přenesení motivu na DPS byla použita metoda fotocesty. Osazené desky jsou vidět na obrázku 45. Spodní DPS již má na sobě nasunuto Arduino YÚN.



Obrázek 45: Foto spodní DPS (s vloženým Arduino YÚN) a vrchní DPS s LCD displejem

5.16 Přední panel

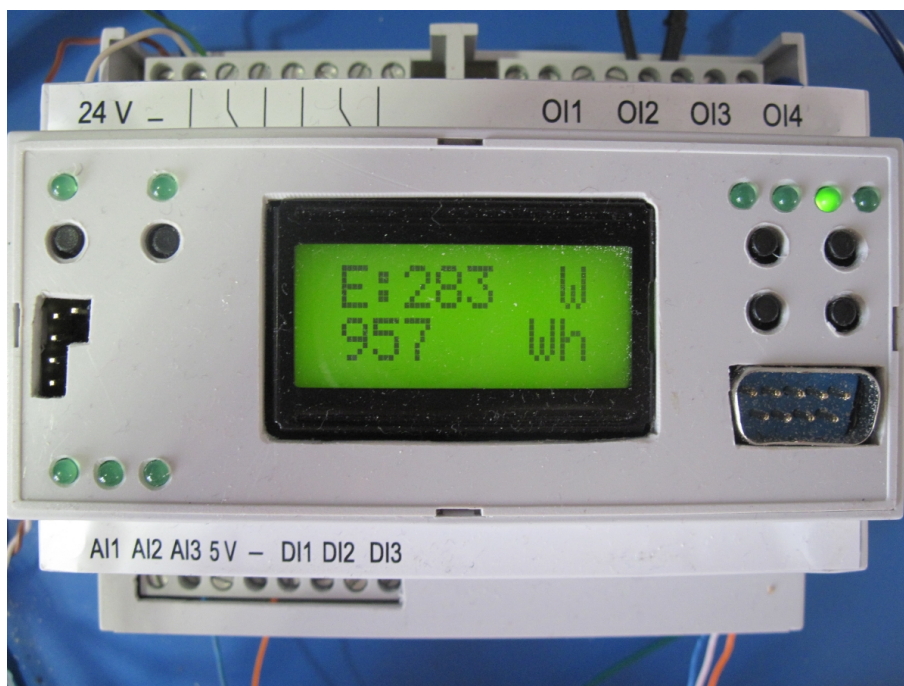
Přední panel obsahuje:

- Diody pro signalizaci sepnutých relé.
- Tlačítka pro ovládání výstupních relé.
- Diody pro signalizaci stavu opticky oddělených vstupů.
- Diody pro signalizaci stavu binárních vstupů.
- Tlačítka pro ovládání LCD displeje.



Obrázek 46: Přední panel měřicího modulu

5.17 Finální modul



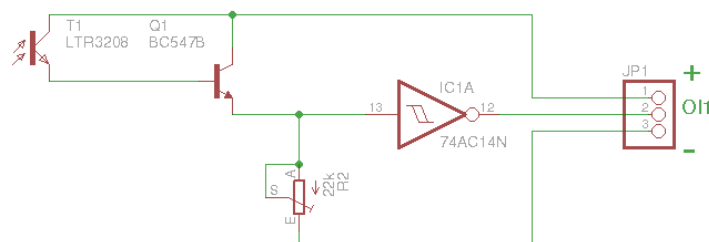
Obrázek 47: Foto měřicího modulu

6 Měření spotřeby energie a dalších veličin

V této kapitole se budu věnovat připojení jednotlivých senzorů a obvodů pro měření spotřeby elektřiny, plynu a vody k měřicímu modulu. Jednotlivé snímače měřičů spotřeby jsou konstruovány s ohledem na konkrétní měřicí přístroje a některé parametry se tak mohou mezi různými měřiči spotřeby lišit (převodní konstanta, citlivost snímacího prvku atd.).

6.1 Vyčítání spotřeby pomocí LED indikující spotřebu elektřiny

Tato metoda využívá pulzující LED umístěné v těle elektroměru. Testovaný elektroměr má převodní konstantu 10000 imp/kWh. Při takto vysoké převodní konstantě je impuls široký v řádu jednotek milisekund. Pro snímání je použit fototranzistor. Schéma zapojení obvodu pro zaznamenání impulsu je na obrázku 48. Výstup z fototranzistoru spíná NPN tranzistor. Za ním se nachází Schmittův klopný obvod, který má odfiltrout případné falešné sepnutí z fototranzistoru. [28]



Obrázek 48: Schéma obvodu pro snímání impulsů z elektroměru



Obrázek 49: Umístění sondy na elektroměru

6.1.1 Vyčítání spotřeby pomocí elektroměrové optické hlavičky

Komunikace s elektroměrovou optickou hlavičkou probíhá pomocí sběrnice RS232 měřicího modulu. Pro vyčítání dat je použit program Pydlms, který je volně ke stažení (viz [44]). Výstup z programu ve formě výpisu získaných dat je vidět na obrázku 50.

```

root@Arduino:~# python dlms.py
  identifier: /LGZ4\2ZMD120AMt.G03

41.8.0: 0001176 [kWh]
C.2.1: 07-07-11 06:48
  C.1: 89263914
    0.0: T096929
  0.2.4: 4001
C.8.0: 03477910
  0.2.1: PRE-7EZ12200AP09
  0.2.0: G03
  0.3.3: 00000250
C.8.4: 00000002
C.8.2: 00000002
C.8.3: 00000001
62.8.0: 0000000 [kWh]
C.8.1: 03477905
  F.F: 00000000
  1.8.4: 0000001 [kWh]
  1.8.0: 0015376 [kWh]
61.8.0: 0003464 [kWh]
  1.8.2: 0000001 [kWh]
  1.8.3: 0000001 [kWh]
42.8.0: 0000000 [kWh]
21.8.0: 0010735 [kWh]
22.8.0: 0000000 [kWh]
  C.7.1: 00000034
  C.7.0: 00000032
  C.7.3: 00000034
  C.7.2: 00000033
  1.8.1: 0015374 [kWh]
  2.8.0: 0000001 [kWh]

```

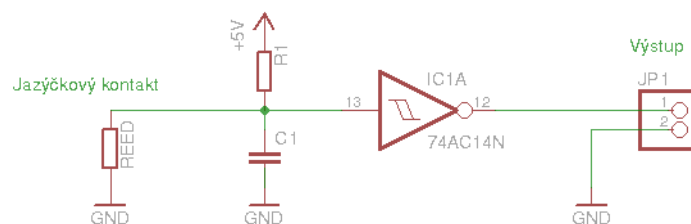
Obrázek 50: Výpis dat z elektroměru

6.2 Vyčítání spotřeby plynu

Vyčítání spotřeby plynu z plynoměru využívá toho, že na posledním otočném číselníku je umístěn neodmýmový magnet a úzká odrazová ploška. Z této skutečnosti vycházejí možné způsoby zachycení impulsu pro další zpracování.

6.2.1 Vyčítání spotřeby plynu pomocí jazýčkového kontaktu

Pro snímání lze místo výše zmíněného snímače využít obyčejný jazýčkový kontakt. Pro odstranění zákmitů se však musí využít buďto HW nebo SW řešení zákmitů při spínání (debouncing). K tomuto účelu lze, podobně jako u snímání impulsů elektroměru, využít Schmittův klopný obvod. Časová konstanta T se přibližně vypočítá $T = R \cdot C$.



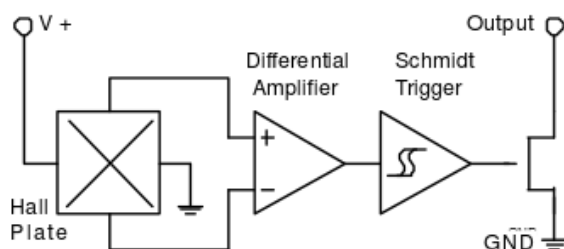
Obrázek 51: Schéma switch debounceru

Rozhodujícím kritériem po použití tohoto způsobu je přístupnost výřezu pro vložení snímacího prvku. Snímání přes průhledný kryt vykazovalo vzhledem ke vzdálenosti značnou chybovost. [28]

6.2.2 Vyčítání spotřeby pomocí Hallova senzoru

Hallův senzor je založen na měření hallova napětí, které vznikne důsledkem vzájemného působení proudu procházejícího polovodičovou destičkou a magnetického pole, které na destičku působí.

Hallův senzor lze využít podobně jako v předcházejícím případě díky neodýmovému magnetu na posledním analogovém otočném číselníku. Pro tento účel se v zásadě hodí senzor s hallovou sondou, který již obsahuje obvody pro vyhodnocení. Díky těmto obvodům pak lze snímat pouze impulzy. Blokové schéma takového senzoru je vidět na obrázku 52.

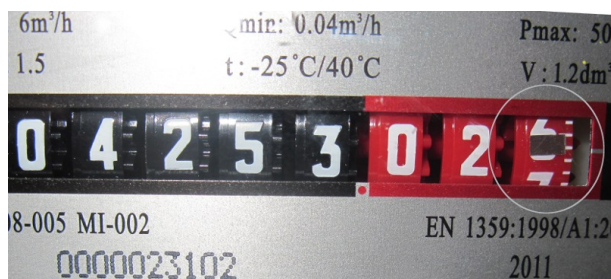


Obrázek 52: Blokové schéma hallova senzoru (převzato z [20])

Testování tohoto senzoru probíhalo pomocí multimetru a Arduino desky s AD převodníkem. Testované senzory nebyly dost citlivé na to, aby je magnet plynoměru přes plexi průhledný kryt dokázal vybudit. Pokud jsem je testoval na běžný magnet, tak vše fungovalo vždy do určité vzdálenosti (v závislosti na velikosti magnetu). Řešením by tedy mohlo být použití citlivějšího senzoru (~1 mT).[28]

6.2.3 Vyčítání spotřeby plynu pomocí optosenzoru

Metoda založená na vyčítání spotřeby pomocí optosenzoru využívá odrazové plošky umístěné na posledním analogovém číselníku. Tato ploška se nejčastěji nachází pod číslem 6 (viz obrázek 53).



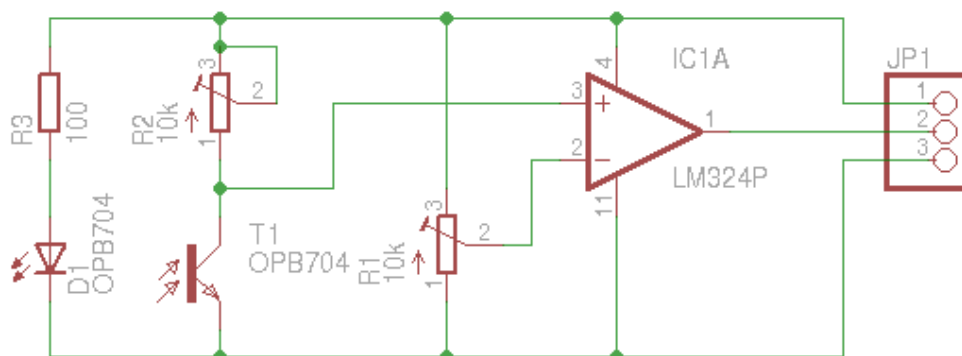
Obrázek 53: Analogový číselník plynoměru

Snímají se počty otáček, z kterých lze pak vyhodnotit množství spotřebované energie. Pokud číselník neobsahuje odrazovou plošku, tak obvykle obsahuje jednu, či více čárových značek, které je možné snímat obdobným způsobem. Pro snímání odrazové plošky jsem použil optosenzor OPB704, který obsahuje fotodiodu a fototranzistor v jednom pouzdře. Možný způsob připevnění pomocí modelovací hmoty je vidět na obrázku 54.



Obrázek 54: Možné umístění optosenzoru na plynoměru

Pro vyhodnocení impulsu je pak zapotřebí připojit výstupní signál fototranzistoru k operačnímu zesilovači v zapojení ve funkci komparátoru. Komparátor pak vyhodnocuje napěťové hladiny na svých vstupech a když dojde k překročení této hladiny, objeví se na výstupu impuls. Schéma zapojení obvodu pro vyhodnocení odrazové plošky je na obrázku 55.[28]



Obrázek 55: Schéma pro vyčítání impulsů z plynoměru

6.3 Vyčítání spotřeby vody

Pro měření spotřeby vody je v měřicím modulu připraven opticky oddělený vstup. Pro snímání spotřeby se tak počítá s vodoměrem s impulzním výstupem.

V průběhu zpracování práce byla také testována možnost absolutního snímání hodnot z vodoměru pomocí USB webkamery a zařízení Raspberry Pi. Obraz z webkamery se podařilo snímat a ukládat v pravidelných intervalech.

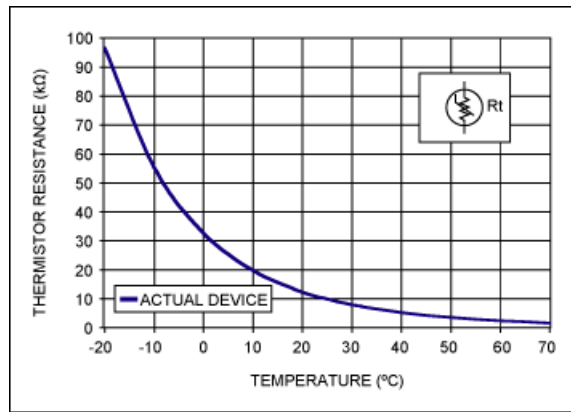
Realizace softwaru pro následné zpracování obrazu by však vydalo na samostatnou práci.

6.4 Měření teploty

Měření teploty v interiéru a exteriéru budovy umožní lépe vyhodnocovat data o spotřebě elektrické energie, plynu atd. v souvislosti s vytápěním objektu. Díky tomu je možné v kombinaci s dalšími regulačními systémy vhodně optimalizovat spotřebu. Měřit teplotu pomocí měřicího modulu lze několika způsoby.

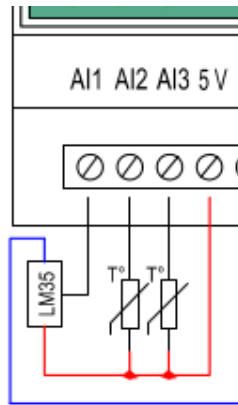
6.4.1 Termistor

První varianta je cenově nejdostupnější a využívá ke své funkci NTC termistor, což je součástka, u které odpor se zvyšující se teplotou klesá (negativní teplotní koeficient).



Obrázek 56: Teplotní závislost odporu termistoru (převzato z [21])

Nevýhodou je jeho nelineární charakteristika. Připojení termistoru k analogovému vstupu měřicího modulu je na obrázku 57.



Obrázek 57: Připojení termistoru a senzoru LM35 k analogovému vstupu

měřicí modul měří hodnotu napětí na děliči tvořeného termistorem a rezistorem 10 kOhm. Napětí je měřeno v rozsahu 1024 hodnot (0-1023). Pro výpočet odporu termistoru z naměřeného napětí je potřeba použít vzorec pro dělič napětí.

$$R_T = R_b * \left(\frac{1024}{N_{vzorku}} - 1 \right) \quad (5)$$

Pro výpočet teploty pak můžeme použít převodní tabulku, nebo vzorec pro závislost odporu na teplotě. Vzorec závislosti teploty na odporu termistoru je:

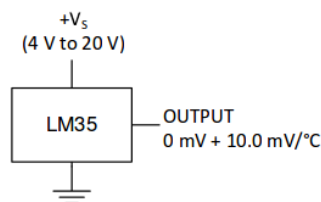
$$R_T = R_0 * e^{-B \left(\frac{1}{T_0} - \frac{1}{T} \right)} \quad (6)$$

Z předcházejícího vzorce vyjádřená teplota T je uvedena níže. Funkce logaritmu je zahrnuta v programové knihovně „math.h” - vývojového prostředí Arduino IDE.

$$T = \frac{1}{\frac{1}{T_0} + \frac{1}{B} * \log \left(\frac{R_T}{R_0} \right)} \quad (7)$$

6.4.2 Senzor LM35

Další možnou variantou je použití převodníku teplota-napětí, například LM35. Zapojení je patrné z obrázku 58.



Obrázek 58: Senzor teploty LM35

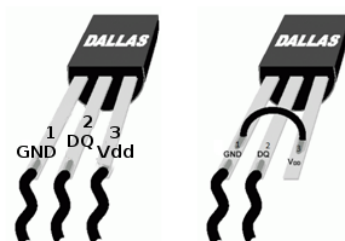
Výhodou je výsledná lineární charakteristika senzoru. Převodní koeficient senzoru LM35 je 10 mV/°C. měřicí modul opět vzorkuje signál v rozsahu 1024 hodnot. Referenční napětí je 5000 mV.

Teplota je vypočítána na základě vzorce:

$$T = \frac{U_{ref}}{1024 * K_{coef}} * N_{vzorku} \quad (8)$$

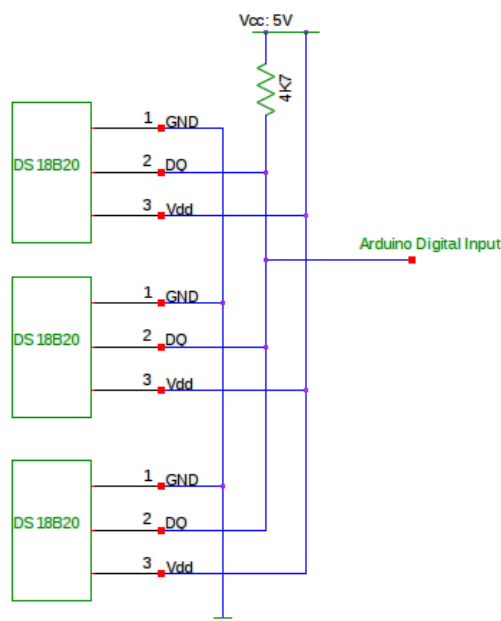
6.4.3 Senzor DS18B20

Inteligentní senzor teploty DS18B20 od firmy Dallas komunikuje pomocí 1-Wire sběrnice (blíže viz kapitola 3.5). Teplotní rozsah tohoto senzoru je od -55 °C do +125 °C s přesností 0,5 °C. Pro konverzi využívá 12 bitový AD převodník. Výhoda tohoto senzoru je jeho snadná implementace, sběrniceová topologie a to, že měřená veličina je přímo v jednotkách °C. Senzor DS18B20 lze provozovat ve dvou režimech (normal mode a parasit mode). [34, 28]



Obrázek 59: DS18B20 (převzato z [22])

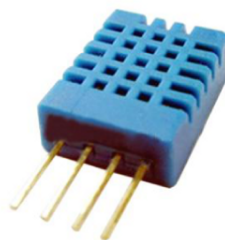
V normálním módu se připojují k nadřazenému systému všechny tři vývody inteligentního senzoru (GND - , Vdd + , Dq pro data), zatímco v parazitním módu se využívají pouze dva vývody senzoru - Vdd je spojen s GND a Dq, slouží jak pro data, tak pro parazitní napájení senzoru . Zapojení s parazitním módem lze využít na krátkou vzdálenost cca 10 m, zatímco s normálním módem cca 30 m. Příklad zapojení tří senzorů v normálním módu viz obrázek 60.



Obrázek 60: Schéma zapojení senzorů DS18B20 (převzato z [22])

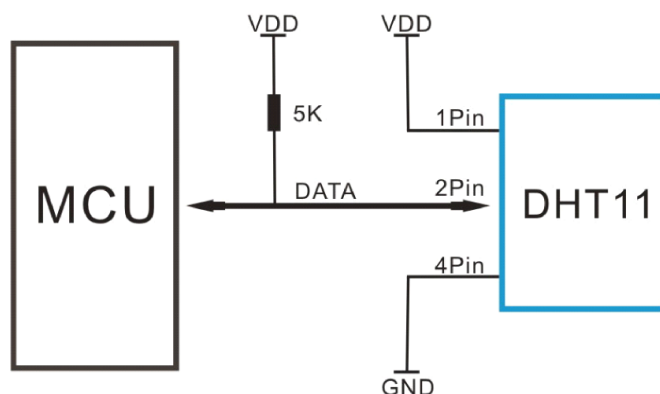
6.5 Měření relativní vlhkosti

Relativní vlhkost se měří zpravidla za účelem udržení hygienického komfortu v daném prostředí. Využívá se především v klimatizačních systémech a v systémech úpravy vzduchu. Lze tak například automaticky spínat ventilátor v koupelně na základě změny vlhkosti vzduchu apod. Senzorů relativní vlhkosti je celá řada. Jeden z nich je například senzor DHT11 (viz obrázek 61), který měří relativní vlhkost a teplotu. Tento senzor lze připojit na sběrnici 1-Wire.



Obrázek 61: DHT11 (převzato z [23])

Inteligentní senzor využívá pro měření teploty NTC termistor a pro měření relativní vlhkosti (RH) kapacitní prvek. Hodnota RH je měřena s přesností 5% (DHT22 s přesností 2%). O převodní tabulku a komunikaci na sběrnici se stará 8bitový mikroprocesor. Pro připojení senzoru na vzdálenost delší jak 20 m se doporučuje zařadit do obvodu Pull-Up rezistor o velikosti 5 kOhm tak, jak je naznačeno na obrázku 62. [23]



Obrázek 62: Připojení senzoru DHT11 k mikrokontroléru (převzato z [23])

Hodnota napájecího napětí se musí pohybovat mezi 3 - 5,5 V. Pro komunikaci s deskou Arduino je vytvořena knihovna DHTlib, pomocí níž lze snadno zprostředkovat komunikaci se senzorem. Maximální spotřeba senzoru činí 2,5 mA.[23, 34]

6.6 Měření osvětlení

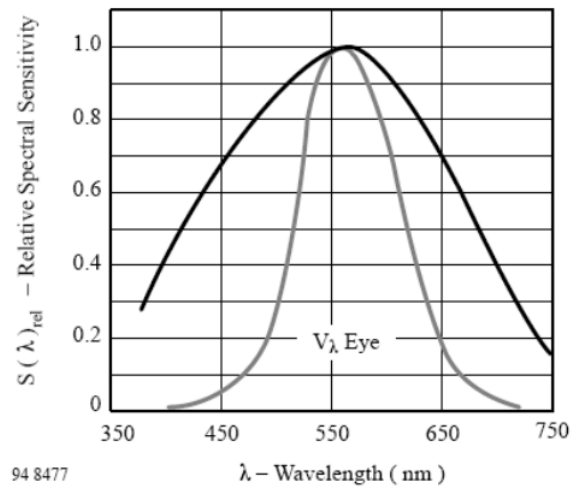
Intenzita osvětlení je fotometrická veličina definovaná jako světelný tok dopadající na určitou plochu. Je tedy podílem světelného toku a plochy. Značí se E. Její jednotkou je lux (lx).

Pro měření osvětlení lze použít stejně jako pro měření teploty, analogový prvek, například fotorezistor.

Stejně tak lze využít prvek s lineární charakteristikou, například fotodiodu BPW21 (obrázek 63), která obsahuje korekční filtr pro vlnové délky odpovídající citlivosti lidského oka. Porovnání poměrné spektrální citlivosti lidského oka a fotosenzoru je na obrázku 67.



Obrázek 63: BPW21 (převzato z [24])



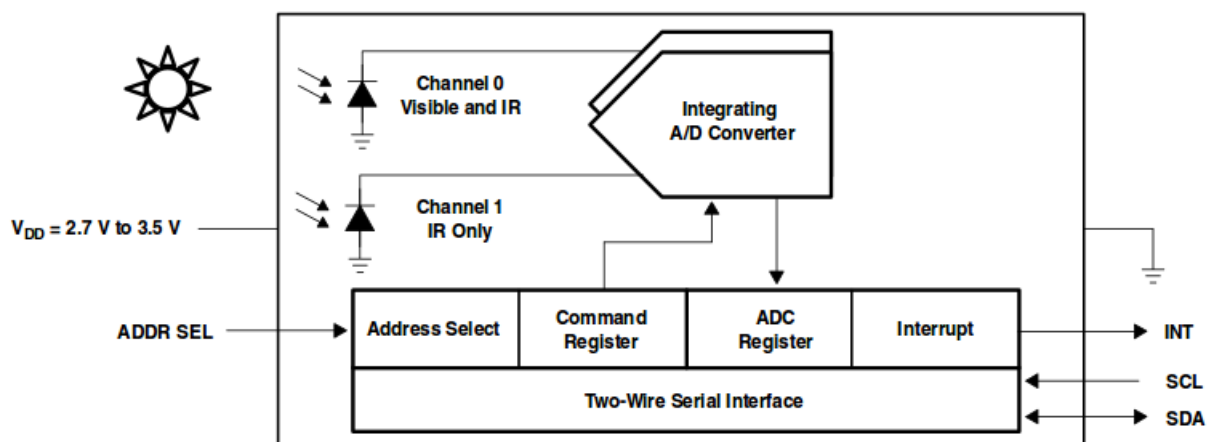
Obrázek 64: Poměrná spektrální citlivost lidského oka a fotosenzoru (převzato z [24])

Pro měření osvětlení v budově je však výhodnější využít některý z inteligentních senzorů. Snímaná hodnota osvětlení je pak přímo v jednotce intenzity osvětlení - lux. Jedním z nich je senzor intenzity osvětlení TSL 2561 komunikující prostřednictvím I2C sběrnice.



Obrázek 65: TSL 2561 (převzato z [25])

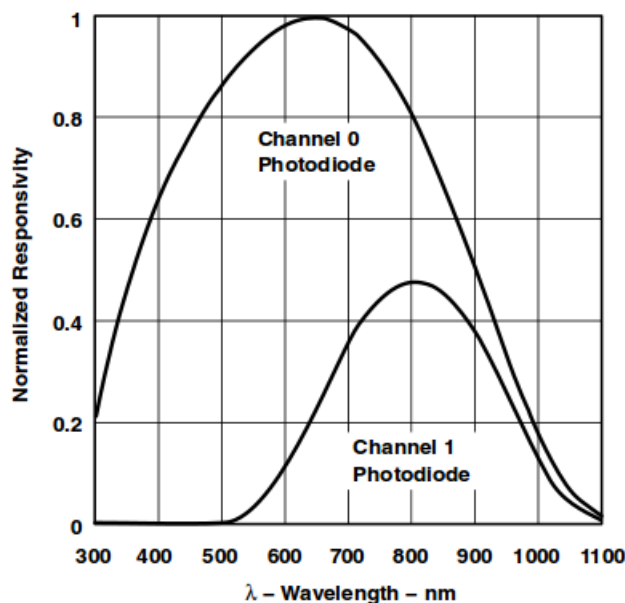
Tento senzor se v praxi především používá pro automatické nastavení podsvětlení displejů u mobilních telefonů a tabletů, a to na základě měření intenzity osvětlení dopadajícího na displej. Nicméně tento senzor lze vhodně využít i pro automatizaci v budově. Blokové schéma senzoru lze vidět na obrázku 66.



Obrázek 66: Blokové schéma TSL 2561 (převzato z [25])

Senzor obsahuje dvě fotodiody. První fotodioda připojená na Channel 0 je citlivá pro vlnovou délku

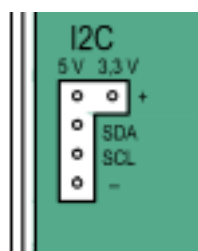
odpovídající viditelnému a infračervenému spektru, zatímco fotodioda připojená na Channel 1 je citlivá pouze pro vlnové délky odpovídající infračervené oblasti. Analogová hodnota z obou fotodiód prochází přes A/D převodník. Druhá fotodioda slouží pro kompenzaci IR složky záření z digitálního výstupu. Graf závislosti poměrné spektrální citlivosti na vlnové délce obou fotodiód se nachází na obrázku 67.



Obrázek 67: Poměrná spektrální citlivost TSL 2561 (převzato z [25])

Mikroprocesor pak na základě dat z obou fotodiód provede aproximaci pro spektrální citlivost odpovídající lidskému oku.

Napájecí napětí senzoru by mělo být v rozmezí 2,7 až 3,6 V. Připojení Arduina se senzorem TSL2561 se provede propojením odpovídajících pinů na předním panelu měřicího modulu - napájecích pinů 3,3 V a GND, dále pak datových pinů I2C sběrnice SDA a SCL. Pro tento senzor existuje knihovna pro Arduino IDE, a tak lze snadno tento senzor programově implementovat. [34]



Obrázek 68: Piny I2C sběrnice měřicího modulu

7 Software

Softwarová část měřicího modulu je rozdělená na program běžící v MCU a další programy umístěné v linuxové části desky Arduino YÚN. SW část v MCU slouží pro obsluhu senzorů, vstupů a výstupů. Dále také zpracovává informace o aktuální spotřebě, atd. V linuxové části měřicího modulu je program pro vyčítání dat z elektroměru a další programy určené pro následnou vizualizaci. Úseky programu jsou vypsány v následujících podkapitolách. Vzhledem k délce programu a přehlednosti práce jsou vypsány jen ty, které jsou důležité pro jeho pochopení.

7.1 Arduino YÚN - MCU

Na začátku programu jsou načteny potřebné softwarové knihovny pomocí příkazu `#include`.

```
#include <OneWire.h> // knihovna pro komunikaci s 1-Wire senzory
#include <DallasTemperature.h> // knihovna pro 1-wire senzory teploty
#include <SimpleModbusSlave.h> // knihovna pro komunikaci protokolem Modbus RTU
#include <Bridge.h> // knihovna pro interface mezi mikrokontrolérem a procesorem
#include <Process.h> // knihovna která umožňuje spustit proces v linuxové části
#include <YunServer.h> //knihona umožňující Arduino pracovat jako server
#include <YunClient.h>//knihona umožňující Arduino pracovat jako klient
#include <FileIO.h> //knihovna pro práci se sobory (např. na SD kartě)
#include <PinChangeInt.h> // knihovna pro změnu pinu přerušeni
#include <dht11.h> //knihovna pro senzory vlhkosti
#include <SimpleModbusSlave.h> //knihovna pro implementaci Modbus protokolu
#include <SoftwareSerial.h> // knihovna pro definování pinů ser.linky
#include <Wire.h> // knihovna pro I2C komunikaci
#include <LiquidCrystal_I2C.h> // knihovna pro obsluhu lcd displeje
#include <math.h> //knihovna pro matematické operace
#include <Adafruit_Sensor.h> // knihovna pro senzory spol. Adafruit
#include <Adafruit_TSL2561_U.h> // knihovna pro obsluhu I2C senzoru osvětlení
```

Poté jsou definovány jednotlivé globální proměnné.

```
...
#define I2C_ADR_LCD 0x21 // adresa I2C expandéru
#define OPTO_VSTUP1 4 // opto vstup 1 na pinu 4
#define OPTO_VSTUP2 5
...
#define RS485RX 11
#define RS485CONTROL 12
#define RS485TX 13
...
unsigned long pocitadloImpEl=0; //čítač impulzů elektroměru
unsigned long pocitadloImpPl=0; //čítač impulzů plynoměru
...
```

Následuje funkce Setup (), která slouží pro inicializaci jednotlivých pinů a komunikačních sběrnic.

```
void Setup () {
  ...
  pinMode(OPTO_VSTUP1, INPUT); // nastavení vstupní proměné OPTO_VSTUP1
  pinMode(RELE1, OUTPUT); // nastavení výstupní proměné RELE1
  ...
  PCintPort::attachInterrupt(PREKUSENI, countInterrupts, FALLING); //nastavení
  signálu přerušení
  ...
  lcd.begin (8,2); // inicializace LCD displeje
  lcd.setCursor(0,0); // nastavení kurzoru na pozici (0,0) - (sloupec, řádek)
  lcd.print("START..."); //
  ...
  sensors.begin(); // inicializace 1-Wire sběrnice
  ...
  Bridge.begin(); // inicializace rozhraní mezi MCU a SoC
  server.listenOnLocalhost(); // informuje server, že naslouchá na portu 5555
  ..
}
```

7.1.1 Hardwarové přerušení

Následující funkce se vykoná, dojde-li k hardwarovému přerušení. Přerušení může být vyvolané některým z opticky oddělených vstupů měřicího modulu. Pokud dojde k přerušení, testuje se, který ze vstupů toto přerušení vyvolal a následně dojde inkrementaci čítače impulzů.

```
...
void countInterrupts() {
  unsigned long pulseTime = millis(); // zaznamenání okamžiku přerušení
  if (digitalRead(OPTO_VSTUP1)==LOW) { // obrácená logika
    pocitadloImpEl ++; //čítač impulzů
    intervalEl=pulseTime-lastTime; // výpočet časového intervalu mezi dvěma impulzy
    lastTime = pulseTime;
  }
  ...
}
```

7.1.2 Výpočet spotřeby, příkonu a dodaného množství

V následujícím bloku je ukázka příkazů pro výpočet spotřeby a příkonu elektrické energie. Příkazy pro výpočet spotřeby plynu a vody jsou řešeny obdobným způsobem. Výpočet aktuálního příkonu elektrické energie se provede na základě hodnoty časového rozdílu mezi dvěma impulzy. U měření aktuálního příkonu ve formě plynu se čítají impulzy po dobu pěti minut a následně je dodaná energie podělena časovým intervalem.

7.1.3 Měření teploty a relativní vlhkosti

```
...
spotrebaEl = pocitadloImpEl*10; // [Wh]
power = (3600000.0 / (intervalEl))/10;
// přepočítání na aktuální příkon [W], číslo 10 zahrnuje konstantu 10 imp/Wh
...
```

Příkazy pro měření teploty pomocí termistoru:

```
...
int sensorValue = analogRead(AI2); // přečtení hodnoty AD převodníku
float R=10000*(1024.0f/float(sensorValue)-1); // výpočet odporu termistoru
T=(1.0f/(1.0f/298.15f+(1.0f/3977)*log(R/10000)))-273.15; //výpočet teploty
...
```

Příkazy pro načtení hodnoty teploty 1-Wire senzoru DS18B20:

```
...
sensors.requestTemperatures(); //žádost o načtení teplot
float teplota2=sensors.getTempCByIndex(0); //přečtení teploty ze senzoru s
indexem 0
...
```

Příkazy pro načtení teploty vlhkosti ze senzoru DHT11:

```
...
senzor_vlhkosti.read(ONE_WIRE_BUS); // přečtení hodnot
vlhkost = senzor_vlhkosti.humidity; // vlhkost
teplotaInt = senzor_vlhkosti.temperature; // teplota
...
```

7.1.4 Obsluha tlačítek a LCD displeje

Příklad části programu pro obsluhu tlačítek na předním panelu. Nejprve je potřeba zahájit komunikaci s I2C expandérem.

```

...
Wire.requestFrom(32,1);// zahájení komunikace s expandérem PCF8574
if(Wire.available())
{
iInput = Wire.read(); // přečtení bajtu
}
if (iInput==254) // TL1 _ pro RELE1
{
if(currentMillis - previousMillis2 >= intervalTl) // podmínka složící pro
odstranění záchvěvů při stisku tl
{
previousMillis2 = currentMillis;
boolean hodnota=digitalRead(RELE1); // přečte aktuální stav výstupu
if (hodnota==0) digitalWrite(RELE1, 1); // příkaz pro přepnutí relé
if (hodnota==1) digitalWrite(RELE1, 0); // příkaz pro přepnutí relé
}
}
...

```

Příkazy pro obsluhu LCD displeje:

```

...
void lcdDisplej (void){
switch(lcdStav){ // přepínání výpisu položky
case 0:
lcd.setCursor(0,0); //nastavení kurzoru displeje
lcd.print("E:"); lcd.print(int(power));
lcd.setCursor(7,0);
lcd.print("W");
lcd.setCursor(0,1);
lcd.print(spotrebaE1);
lcd.setCursor(6,1);
lcd.print("Wh");
break;
}
...

```



Obrázek 69: Výpis dat na LCD displej

7.1.5 Komunikace s webovým serverem

Oboustranou komunikaci s webovým serverem zajišuje knihovna YunServer a YunClient(). Z webové stránky je možné ovládat výstupní relé a číst aktuálně naměřená a vypočtená data. Zápis dat a požadavek na přečtení vstupu z webové stránky se provádí skrz hypertextový odkaz.

```
...
void digitalCommand(YunClient client) {
int pin, value;
pin = client.parseInt(); // přečte číslo pinu
if (client.read() == '/') {
value = client.parseInt(); // přečte hodnotu, která se má zapsat na výstup
digitalWrite(pin, value); // zapíše hodnotu
}
else {
value = digitalRead(pin); // pokud za příkazem není žádná hodnota, dojde k
přečtení hodnoty vstupu
}
}
...
```

Formát hypertextového odkazu je následující:

```
http://arduino.local/arduino/digital/6 // pro čtení hodnoty na pinu 6
http://arduino.local/arduino/digital/6/0 // pro zápis hodnoty 0 na pin 6
http://arduino.local/arduino/digital/6/1 // pro zápis hodnoty 1 na pin 6
```

Analogicky je řešeno i čtení vypočtených a změřených hodnot.

7.1.6 Odeslání dat pro generování grafů

Data se odesílají do linuxové části Arduino YÚN, kde následně probíhá generování grafů. Odeslání dat probíhá pomocí příkazu Bridge.put() v pravidelném časovém intervalu.

```
...
char buf_rh[8]; // definice lokální proměnné buf_rh
...
dtostrf(vlhkost,1,2,buf_rh); // konvertuje číslo obsažené v proměnné vlhkost na
char
Bridge.put("humidity", buf_rh); // odešle hodnotu do linuxové části
...
```

7.1.7 Záznam hodnot s časovou značkou na SD kartu

Tato funkce každý den uloží denní spotřebu energií a vody. K záznamu přidá časovou značku. Poté dojde k vynulování čítačů.

```
...
DateTime now = rtc.now(); // přečtení aktuálního času z RTC obvodu
if(now.day()>denZaznam || ((now.day()<denZaznam) && (nowDay()==1))) { // pokud
je další den
File script = FileSystem.open("/sda1/data.txt",FILE_APPEND); // otevření souboru
pro zápis
if (script) { // pokud je soubor otevřen zapišou se následující hodnoty
script.print(spotrebaEl);
script.print("|");
script.print(spotrebaPlyn);
script.print("|");
script.print(spotrebaVoda);
script.print("|");
script.print(now.year(), DEC);
script.print(now.month(), DEC);
script.print(now.day(), DEC);
script.close();
}
denZaznam=now.day;
impEl=0; // vynuluj čítač
impPlyn=0; // vynuluj čítač
impVoda=0; // vynuluj čítač
}
...
```

7.1.8 Implementace protokolu Modbus

Pro komunikaci s rozsáhlejším systémem MaR je měřicí modul vybaven sběrnici RS485. Komunikační protokol byl zvolen Modbus. Pro Arduino existuje knihovna simpleModbus (viz [45]), která je vhodná pro softwarovou implementaci měřicího modulu. Měřicí modul je nastaven jako slave (server).

```

...
enum { // definování jednotlivých registrů, číslování od 0
EL_VAL,
PL_VAL,
...
HOLDING_REGS_SIZE // číslo posledního registru - určuje délku
...
void setup()
modbus_configure(&Serial1, 9600, SERIAL_8N2, 1, 2, HOLDING_REGS_SIZE,
holdingRegs); //definování ser. komunikace
modbus_update_comms(9600, SERIAL_8N2, 1);
...
modbus_update();
holdingRegs[EL_VAL]=spotrebaEl;
holdingRegs[PL_VAL]=spotrebaPl;
...

```

7.2 Arduinu YÚN - Linuxová část

Výchozí nastavení IP adresy Arduina YÚN je 192.168.240.1. Základní nastavení (změna IP adresy, přístupové údaje atd.) lze provést pomocí webového prohlížeče. Pro práci s příkazovým řádkem linuxové distribuce je vhodné nastavit komunikaci přes SSH klienta (program Putty).

Vestavěná flash paměť pro data má pouze 16 MB. Z tohoto důvodu je vhodné zvětšit paměť pomocí microSD paměťové karty. Pro tento účel slouží program YunDiskSpaceExpander, který se spouští přímo v Arduino IDE.

Pro zprovoznění webového serveru, komunikaci s MCU, generování grafů atd. je potřeba, příkazem `opkg install`, nainstalovat následující aplikace.

```

opkg install openssh-sftp-server
opkg install php5 php5-cgi
opkg install php5-mod-mysql
opkg install pyopenssl
opkg install python-openssl
easy_install pyserial
opkg install kmod-usb-serial-ftdi

```

7.2.1 Program pro komunikaci s elektroměrovou optickou hlavicí

Během testování se nepodařilo softwarově zprovoznit převodník UART-RS232. Proto bylo využito náhradní řešení za použití převodníku USB-RS232. Pro vyčítání dat byl upraven program Pydlms, který je volně ke stažení. Úprava spočívala v zápisu hodnot do souboru na SD kartě. Program aktualizuje tabulku naměřených hodnot každých deset minut. Soubor s naměřenými hodnotami je možné zobrazit pomocí php skriptu webové vizualizace. Výpis naměřených hodnot viz obrázek 70.

Výpis dat

Elektroměr: /LGZ4\2ZMD120AMt.G03

Datum: 7. 5. 2015, 22:13

Kód	Popisek	Hodnota
41.8.0	Kladná činná energie z fáze L2	0001299 [kWh]
C.2.1	Datum nulování čítače	07-07-11 06:48
C.1	Seriové číslo elektroměru	89263914
0.0	Adresa zařízení	T096929
C.8.0	Celkový pracovní čas	03682204
0.2.1	ID elektroměru	PRE-7EZ12200AP09
0.2.0	Software ID	G03
62.8.0	Záporná činná energie z fáze L3	0000000 [kWh]
F.F	Chybový status	00000000
1.8.4	Kladná činná energie, tarif T4	0000001 [kWh]
1.8.0	Celková dodaná činná energie	0016446 [kWh]
61.8.0	Kladná činná energie z fáze L2	0003691 [kWh]
1.8.2	Kladná činná energie, tarif T2	0000001 [kWh]
1.8.3	Kladná činná energie, tarif T3	0000001 [kWh]
42.8.0	Záporná činná energie z fáze L2	0000000 [kWh]
21.8.0	Kladná činná energie z fáze L1	0011457 [kWh]
22.8.0	Záporná činná energie z fáze L1	0000000 [kWh]
C.7.1	Časová značka události, výpadek fáze L1	00000035
C.7.0	Časová značka události, celkový výpadek napájení	00000033
C.7.3	Časová značka události, výpadek fáze L3	00000035
C.7.2	Časová značka události, výpadek fáze L2	00000034
1.8.1	Celková dodaná činná energie, tarif T1	0016444 [kWh]
2.8.0	Celková záporná energie	0000001 [kWh]

Obrázek 70: Výpis naměřených dat

7.2.2 Generování grafů

Grafy jsou generovány pomocí nástroje RRDtool.

Příkaz pro vytvoření databáze dat pro graf spotřeby elektrické energie vypadá následovně:

```
rrdtool create elpower.rrd --step 60
DS:elpo:COUNTER:600:0:50 COUNTER
RRA:AVERAGE:0.5:1:1440
```

Příkaz create vytvoří databázi elpower.rrd. Příkaz – step 60 určuje krok (v sekundách) jakým se budou zapisovat nové hodnoty. DS značí popis proměnné, která se bude zaznamenávat. COUNTER znamená, že se jedná o proměnnou, jejíž hodnota neustále narůstá (inkrementální čítač). Pro vykreslení grafu pak vypočítává s rozdíl dvou předchozích hodnot. Příkaz RRA určuje způsob ukládání dat. Vytvoření databáze pro graf spotřeby plynu a vody je zcela analogické. Pro měření stavové hodnoty (teplota, vlhkost) se místo proměnné COUNTER použije proměnná GAUGE.

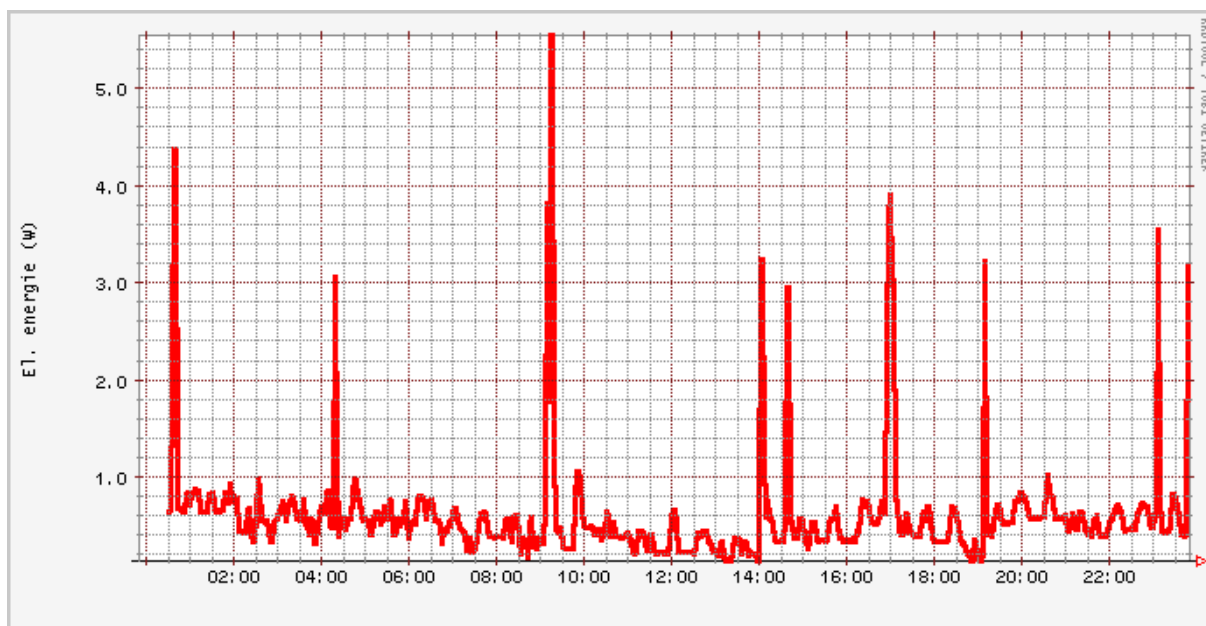
Pro vkládání dat do databáze slouží příkaz rrdupdate.


```
rrdupdate elpower.rrd N:elpower
```

Generování grafů probíhá pomocí následujícího příkazu.

```
rrdtool graph /mnt/sda1/arduino/www/elpower3.png  
-a, PNG, -w, 600, -h, 300, -r, --vertical-label, El. energie (W), --title,  
Mereni spotreby el. energie - 24h,  
DEF:elpo=elpower.rrd:elpo:AVERAGE,  
CDEF:elpos=elpo,60,* ,  
LINE2:elpos#ff0000:
```

Za příkazem `graph` se nachází cesta k souboru, kam se bude ukládat graf. Následují příkazy definující velikost grafu a jeho popisky. Příkaz `DEF` označuje proměnou, která se bude používat. `CDEF` slouží pro úpravu dat uložených v databázi - v tomto případě budou hodnoty vynásobeny šedesáti. `LINE2` značí, že křivka bude mít tloušťku 2px. Graf spotřeby elektrické energie vygenerovaný prostřednictvím RRDtools je na obrázku 71.

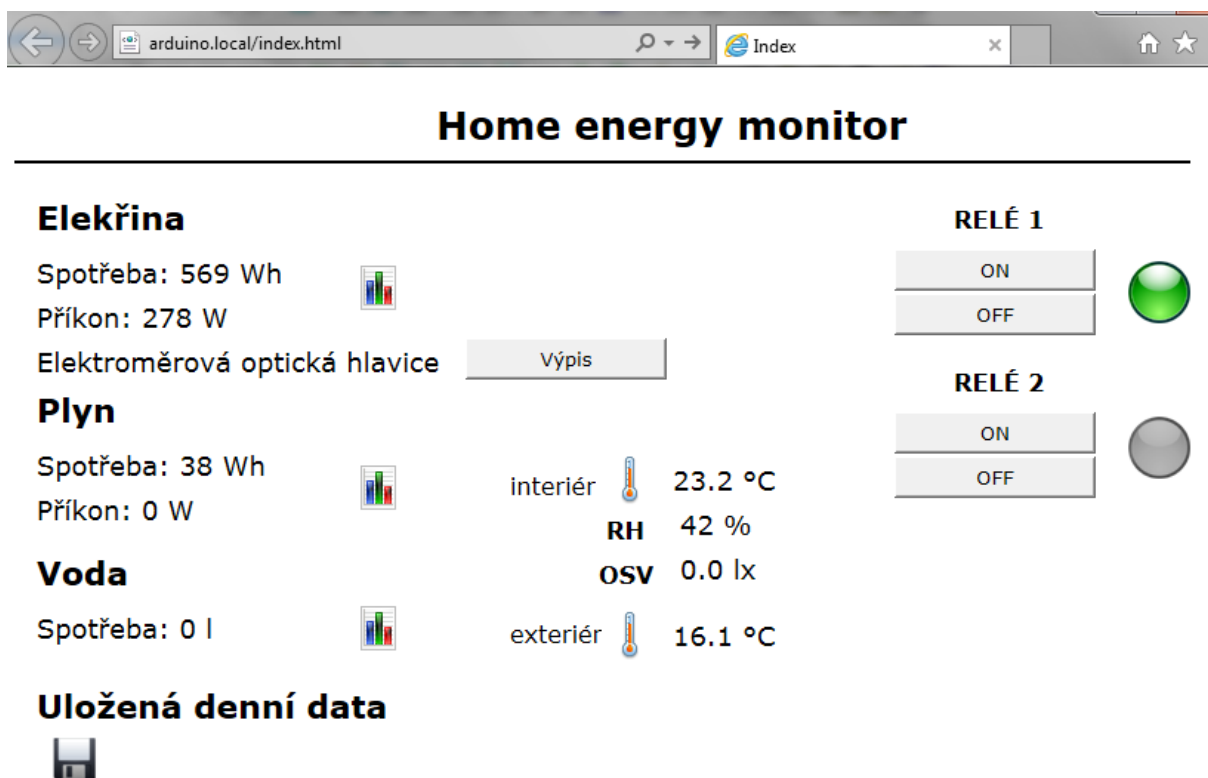


Obrázek 71: Graf spotřeby elektrické energie - 24hod

Příkazy pro vkládání dat a příkazy pro vykreslování grafů jsou součástí programu napsaného v programovacím jazyce python. Pro příjem dat z mikrokontrolerové části je použita knihovna `bridge`.

7.2.3 Vizualizace naměřených dat

Měřicí modul obsahuje vizualizaci naměřených dat pomocí vestavěného webservru. Ukázka vizualizace viz obrázek 72. Způsob předávání dat mezi mikrokontrolérem a webovým servrem byl popsán v kapitole 7.1.5. Z úvodní stránky lze přejít na grafy průběhu spotřeby (viz obrázek 71) a na stránku s výpisem údajů elektroměrové optické hlavice. Z webové vizualizace lze také ovládat výstupní relé měřicího modulu. Dále je možné stáhnout soubor, kam se zaznamenává denní spotřeba elektřiny, plynu a vody.



Obrázek 72: Webová vizualizace

8 Závěr

Cílem této práce bylo navrhnout a realizovat měřicí modul pro monitoring spotřeby energií a měření dalších veličin. Navržený měřicí modul těchto cílů dosahuje.

V teoretické části práce jsou popsány možnosti měření spotřeby elektřiny, plynu a vody s ohledem na návrh měřicího modulu. Část práce byla věnována základnímu popisu použitých senzorů a sběrnic. Samostatná kapitola pak byla věnována vývojové desce Arduino YÚN, která tvoří jádro měřicího modulu.

Praktické realizaci měřicího modulu předcházelo testování různých možností vyčítání údajů z fakturačních měřičů spotřeby elektřiny a plynu. Výsledky testování sloužily jako podklad pro návrh měřicího modulu. Poté následovala výroba a osazení potřebných DPS.

Pro vyčítání spotřeby elektřiny byla využita metoda snímání impulzů LED umístěné v těle elektroměru. Dále je také využita metoda odečtu pomocí elektroměrové optické hlavičky, komunikující s měřicím modulem přes převodník USB-RS232. Pro vyčítání spotřeby z plynoměru se nejlépe osvědčila metoda využívající optozávoru. Pro měření spotřeby vody je počítáno s instalací vodoměru s impulzním výstupem, pro který je v měřicím modulu připraven vstup.

Měření teploty je realizováno pomocí termistoru a senzorů připojených na 1-Wire sběrnici. Bohužel se nepodařilo do odevzdání této práce softwarově implementovat 1-Wire driver DS2482. Pro komunikaci s 1-Wire senzory byl tedy využit pin původně určený pro binární vstup. Pro měření vlhkosti byl použit senzor DHT11. Dále byla také doplněna možnost měření osvětlení.

Vizualizace naměřených dat probíhá přes LCD displej měřicího modulu, kdy lze tlačítka přepínat sledovanou veličinu, a přes webové rozhraní měřicího modulu. Modul dále obsahuje dva reléové výstupy pro univerzální použití. Tyto výstupy lze ovládat tlačítka na měřicím modulu a přes webové rozhraní.

Další možné rozšíření měřicího modulu může být v oblasti bezdrátové komunikace se senzory. Měřicí modul je na toto rozšíření připraven v podobě hardwarové implementace bezdrátového komunikačního modulu RFM12B.

Seznam použitých zkratk

BEMS	Building Energy Management System - systémy pro úsporu energií v budovách
HEMS	Home Energy Management Systems - systémy pro úsporu energií v domácnostech
LED	Light-Emitting Diode - světlo emitující dioda
AD	Analog-Digital - analogově číslicový převodník
PWM	Pulse Width Modulation - pulzně šířková modulace
USB	Universal Serial Bus - univerzální sériová sběrnice
TTL	Transistor-Transistor Logic - tranzistorově-tranzistorová logika
IDE	Integrated Development Environment - vývojové prostředí
MIT	Massachusetts Institute of Technology
LAN	Local Area Network - lokální síť
GSM	Groupe Spécial Mobile - systém pro mobilní komunikaci
RFID	Radio Frequency Identification - identifikace na rádiové frekvenci
NFC	Near Field Communication
OSI	Open Systems Interconnection - otevřený referenční komunikační model
UART	Universal Asynchronous Receiver/Transmitter - univerzální asynchronní sériové rozhraní
DTR	Data Terminal Ready
DSR	Data Set Ready
DPS	Deska plošných spojů
FTDI	Future Technology Devices International
I2C	Inter-Integrated Circuit - vnitřní integrovaná sběrnice
SDA	Synchronous Data - datový kanál
SCL	Synchronous Clock - hodinový signál
TWI	Two Wire Interface - dvoudrátová sběrnice
CSMA/CD	Carrier Sense Multiple Access with Collision Detection - mnohonásobný přístup s detekcí kolize
SoC	System on Chip

Reference

- [1] Měřič spotřeby voltcraft energy. URL
<http://www.conrad.cz/meric-spotreby-voltcraft-energy-check-3000-cz.k122181>.
- [2] P.Ripka. Měření spotřeby energie a medií - monitorování a ovládání. URL
http://measure.feld.cvut.cz/system/files/files/cs/vyuka/predmety/A5M38SZS/07_monitorovani%20spotreby_IB_2015.pdf.
- [3] Petr Voborník. Výzkum spolehlivosti statických elektroměrů - diplomová práce, 2013. URL
https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=67797.
- [4] Radomír Kozub. Smart metering: Cesta od mechanického elektroměru k chytrým sítím, 12 2012. URL
http://www.crr.vutbr.cz/system/files/brozura_10_1212.pdf.
- [5] Non-invasive ac current sensor. URL <http://www.elecfreaks.com/store/noninvasive-ac-current-sensor-sct013-30a-max-p-88.html>.
- [6] Gas meter. URL <http://www.britannica.com/EBchecked/topic/226453/gas-meter>.
- [7] Nízkofrekvenční vysílač impulsů in-z61, . URL
http://www.jet-service.cz/elster/prospekty/in_z61_cz.pdf.
- [8] Hans a kol. Nestlé. *Příručka zdravotně technických instalací*. Sobotáles, 2003.
- [9] Water meter and arduino. URL
<http://www.instructables.com/id/Water-meter-Arduino-Aquameterduino-/>.
- [10] Arduino community logo. URL <http://arduino.cc/en/Trademark/CommunityLogo>.
- [11] Arduino yun. URL
<http://arduino.cc/en/Main/ArduinoBoardYun?from=Products.ArduinoYUN>.
- [12] Atmega 32u4-arduino pin mapping, . URL
<http://arduino.cc/en/Hacking/PinMapping32u4>.
- [13] Pavel Tišnovský. Komunikace po sériové sběrnici i2c, 2009. URL
<http://www.root.cz/clanky/komunikace-po-seriove-sbornici-isup2supc/>.
- [14] Maxim. 1-wire tutorial. URL
<http://www.maximintegrated.com/products/1-wire/flash/overview/index.cfm>.
- [15] Systémy pro sběr a přenos dat, přednáška z předmětu a5m38spd. inteligentní senzory., . URL
<http://measure.feld.cvut.cz/system/files/files/cs/vyuka/predmety/A5M38SPD/tema3.pdf>.
- [16] Enika- modulbox. URL <http://www.enika.cz/cz/soucastky-pro-elektroniku/krabicky-a-pouzdra/na-din-listu/modulbox/6-m.html?vyrobek=102&jazyk=cz>.
- [17] Lm2575 - datasheet, . URL <http://www.ti.com/lit/ds/symlink/lm1575.pdf>.
- [18] Datasheet - relay finder. URL
<http://www.gme.cz/img/cache/doc/634/546/relef3451-24-datasheet-1.pdf>.

- [19] Datasheet pcf8574, . URL http://www.nxp.com/documents/data_sheet/PCF8574.pdf.
- [20] Melexis datasheet, section 3, . URL <http://datasheet.octopart.com/MLX90248ESE-EBA-000-RE-Melexis-datasheet-17019364.pdf>.
- [21] Ray Wisman. Msp430 polling, 2012. URL <http://homepages.ius.edu/RWISMAN/C335/HTML/msp430Polling.HTM>.
- [22] Ds18b20 temperature sensing. URL <http://openenergymonitor.org/emon/buildingblocks/DS18B20-temperature-sensing>.
- [23] Sensor teploty a vlhkosti dht11, 2012. URL <http://www.elektronovinky.cz/soucastky/sikovny-senzor-teploty-vlhkosti-dht11>.
- [24] Měření osvětlení. laboratorní úloha z předmětu x34ses. URL <http://www.micro.feld.cvut.cz/home/X34SES/cviceni/Navody%20na%20cviceni/06.Mereni%20osvetleni.pdf>.
- [25] Tsl 2561 luminosity sensor, 2014. URL <https://learn.adafruit.com/downloads/pdf/tsl2561.pdf>.
- [26] Bohumír Garlík. *Inteligentní budovy*. BEN - technická literatura, Praha, 1 edition, 2012.
- [27] Haasz; Sedláček. *Elektrická měření: přístroje a metody*. Česká technika - nakladatelství ČVUT, 2 edition, 2005.
- [28] Hašek Miroslav. *Projekt 2 (A5M99PR2) - Systém pro měření spotřeby energie v budově*. 2015.
- [29] Popis obis kódu pro protokol normy iec 62056. URL http://www.promotic.eu/cz/pmdoc/Subsystems/Comm/PmDrivers/IEC62056_OBIS.htm.
- [30] Open energy monitor. URL <http://openenergymonitor.org/emon/>.
- [31] Josef Fík. Účtování spotřeby zemního plynu v kilowatthodinách. URL <http://www.tzb-info.cz/2012-uctovani-spotreby-zemniho-plynu-v-kilowatthodinach>.
- [32] Tomáš Hlinčík. Vliv okolní teploty na údaj plynoměru, 5 2008. URL http://tresen.vscht.cz/kap/data/studentska_cinnost/obhajoba_semestralniho_projektu_2008/38_vliv_okolni_teploty_na_udaj_plynomeru_soubor.pdf.
- [33] Měření průtoku, . URL <http://www.maryshfmmi.webzdarma.cz/mppak.htm>.
- [34] Hašek Miroslav. *Projekt 1 (A5M99PR1) - Návrh systému domácí automatizace*. 2014.
- [35] Arduino - schematic, 2014. URL <http://www.arduino.cc/en/uploads/Main/arduino-Yun-schematic.pdf>.
- [36] Arduino leonardo, . URL <http://arduino.cc/en/Main/ArduinoBoardLeonardo>.
- [37] Vlastimil Slinták. Arduino dokumentace - vstupně výstupní piny. URL <http://docs.uart.cz/docs/io-piny/>.
- [38] Systémy pro sběr a přenos dat, přednáška z předmětu a5m38spd. referenční model osi., . URL <http://measure.feld.cvut.cz/system/files/files/cs/vyuka/predmety/A5M38SPD/tema4.pdf>.

- [39] J. Fischer. Návrh vestavěných systémů, přednáška 6, 2012. URL <http://measure.feld.cvut.cz/system/files/files/cs/vyuka/predmety/A5M38SPD/tema4.pdf>.
- [40] Systémy pro sběr a přenos dat, přednáška z předmětu a5m38spd, . URL <http://measure.feld.cvut.cz/cs/system/files/files/cs/vyuka/predmety/A5M38SPD/tema6.pdf>.
- [41] Wire library, 2013. URL <http://arduino.cc/en/reference/wire>.
- [42] Systémy pro sběr a přenos dat, ethernet, přednáška z předmětu a5m38spd. URL <http://measure.feld.cvut.cz/system/files/files/cs/vyuka/predmety/A5M38SPD/tema7.pdf>.
- [43] Miroslav Husák. *Mikrosenzory a mikroaktuátory*. Gerstner, 1 edition, 2008.
- [44] Program pydlms, 2011. URL <https://github.com/bsdphk/PyDLMS/blob/master/dlms.py>.
- [45] Modbus rtu library, 2014. URL https://drive.google.com/folderview?id=0B0B286tJkafVSENVcU1RQVBfSzg&usp=drive_web.

9 Příloha 1 - OBIS kódy

(příloha je přiložena zvlášť)