

Řízení vytápění inteligentního domu

Vedoucí práce:
Ing. Richard Klein

Autor práce:
Zimmermann Štěpán

Brno 2013

Tímto bych chtěl poděkovat své rodině za veškerou podporu, své přítelkyni za trpělivost při hledání pravopisných chyb a především vedoucímu mé práce za spoustu užitečných rad a věcných připomínek, které mi byly velice nápomocné.

Čestné prohlášení

Prohlašuji, že jsem tuto práci: **Řízení vytápění inteligentního domu** vypracoval/a samostatně a veškeré použité prameny a informace jsou uvedeny v seznamu použité literatury. Souhlasím, aby moje práce byla zveřejněna v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů, a v souladu s platnou *Směrnicí o zveřejňování vysokoškolských závěrečných prací*.

Jsem si vědom/a, že se na moji práci vztahuje zákon č. 121/2000 Sb., autorský zákon, a že Mendelova univerzita v Brně má právo na uzavření licenční smlouvy a užití této práce jako školního díla podle § 60 odst. 1 Autorského zákona.

Dále se zavazuji, že před sepsáním licenční smlouvy o využití díla jinou osobou (subjektem) si vyžádám písemné stanovisko univerzity o tom, že předmětná licenční smlouva není v rozporu s oprávněnými zájmy univerzity, a zavazuji se uhradit případný příspěvek na úhradu nákladů spojených se vznikem díla, a to až do jejich skutečné výše.

V Brně dne 25. května 2013

Abstract

Zimmermann Š. Management of heating a house. Bachelor thesis. Brno: Mendel University, 2015.

Bachelor thesis deals with the design and subsequent automation of heating a house. Another part is processing output using a web application.

Keywords

heating, photovoltaic collector, heat pump, storage tank.

Abstrakt

Zimmermann Š. Řízení vytápění rodinného domu. Bakalářská práce. Brno: Mendelova univerzita v Brně, 2015.

Bakalářská práce se zabývá návrhem a následnou automatizací vytápění rodinného domu. Další částí je zpracování výstupu pomocí webové aplikace.

Klíčová slova

vytápění, fotovoltaiický kolektor, tepelné čerpadlo, akumulční nádrž.

Obsah

1	Úvod a cíl práce	15
1.1	Úvod.....	15
1.2	Cíl práce.....	15
2	Teoretická část	16
2.1	Rešerše	16
2.1.1	Seznámení s měřicími prvky.....	16
2.1.2	Raspberry Pi.....	16
2.1.3	My SQL.....	16
2.1.4	PHP	17
2.1.5	Tepelné čerpadlo.....	17
2.2	Solární kolektory	17
2.3	Tepelné čerpadlo vzduch – voda.....	18
2.4	Akumulační nadrž.....	19
2.4.1	Výměníky.....	19
2.5	TME	20
2.6	TX20ETH	21
2.7	Raspberry pi.....	21
2.8	Q8	22
2.8.1	Vlastnosti.....	22
2.8.2	Ovládání:.....	22
2.8.3	Základní funkce.....	23
2.9	PHP.....	23
2.9.1	Proč PHP	23
2.9.2	Nevýhody PHP.....	24
2.10	Mysql.....	24
2.11	WYSIWYG Web Builder.....	24
3	Návrh	25
3.1	Výchozí stav.....	25

3.2	Návrh automatizované soustavy	26
3.3	Návrh měřicí soustavy	27
3.3.1	Údaje o teplotě.....	27
3.3.2	Údaje o výrobě/spotřebě elektrické energie.....	27
3.3.3	Měření rychlosti a směru větru.....	27
3.4	Návrh řídicí soustavy	28
3.5	Návrh software	28
3.5.1	MySQL databáze.....	28
3.5.2	PHP skripty	30
3.5.3	Návrh vizualizace	30
4	Praktická část	33
4.1	Instalace operačního systému Raspbian	33
4.2	Nastavení systému Raspbian	33
4.3	Nastavení quida Q8.....	35
4.3.1	Síť.....	36
4.3.2	Odesílání	36
4.4	Nastavení teploměrů TME	37
4.5	Nastavení anemometru	39
4.6	PHP skripty	39
4.6.1	Skripty pro sběr dat.....	39
4.6.2	Skripty pro práci nad daty	41
4.6.3	Skripty pro vytváření výstupů.....	44
4.7	Demon cron.....	45
4.8	Vytvoření webové stránky	46
4.9	Finální zapojení a uspořádání.....	47
5	Ekonomické zhodnocení	48
5.1	Vzniklé úspory	48
5.2	Návrh návratnosti investice.....	48
6	Diskuse	49
6.1	Jsou v plánu nějaká vylepšení?	49

Obsah	11
6.2 Jakých úspor se pomocí zavedení systému dosahuje?.....	49
6.3 Bylo by možné systém udělat levněji?	49
7 Závěr	50
8 Literatura	51
A Ukázka kódu pro vytvoření grafu	53

Seznam obrázků

Obr. 1	Schéma fotovoltaického článku	18
Obr. 2	Schéma tepelného čerpadla vzduch-voda.....	19
Obr. 3	TME multi	20
Obr. 4	TX20ETH s venkovním čidlem	21
Obr. 5	Raspberry pi	22
Obr. 6	I/O modul Q8	23
Obr. 7	Automatizovaná soustava	27
Obr. 8	ERD návrh databáze	29
Obr. 9	Návrh vizualizace webové stránky s aktuálním stavem	31
Obr. 10	Návrh vizualizace webové stránky s historií	32
Obr. 11	fstab	34
Obr. 12	Nastavení sítě u Q8	36
Obr. 13	Nastavení odesílání u Q8.....	36
Obr. 14	Základní nabídka teploměru TME	37
Obr. 15	Výpis po nastavení modulu TME.....	38
Obr. 16	Výpis jednoho čidla po nastavení.....	38
Obr. 17	Crontab	46
Obr. 18	Zapojení rozvodné skříně.....	47

Seznam tabulek

Tab. 1	Tabulka s výpočtem tepelné ztráty akumulční nádrže.	26
Tab. 2	Tabulka s pořizovací cenou.	48

1 Úvod a cíl práce

1.1 Úvod

V dnešní době jsou již téměř všichni lidé na zemi obklopeni nespočtem různých věcí. Všechny tyto věci potřebují, ať už ke svému vzniku, případně ke svému dalšímu provozu, stále větší množství energie.

Naše konzumní společnost, která se stále více rozrůstá, řeší případné nedostatky energie tak, že zvýší její produkci. Naskytá se ovšem otázka, zdali je tento růst udržitelný, a pokud ano, tak jak dlouho?

Jisté je jenom jedno a to, že dělat věci tak, jak jsme zvyklí dnes už možná zítřka nebo za padesát let nebude možné, a to z jednoho prostého důvodu. Téměř celá naše spotřeba energie je momentálně zaštiťována díky těžbě a spotřebě fosilních paliv, jejichž zásoby nejsou nekonečné.

Jediným řešením, které je v této době realizovatelné s ohledem na naši technickou vyspělost, je začít nahrazovat výrobu energie z fosilních paliv energií ze zdrojů obnovitelných. S ohledem na výkonost těchto zdrojů bude také třeba drasticky snížit naši spotřebu ve všech odvětvích naší činnosti.

Jelikož mi není osud naší planety ani tvorů, kteří na ní žijí úplně lhostejný, rád bych přispěl touto prací ke zlepšení situace. Z tohoto důvodu jsem si vybral toto téma a rozhodl se, snížit energetickou náročnost našeho domu.

1.2 Cíl práce

Cílem této práce je řídit systém pro snížení a optimalizaci energetické náročnosti rodinného domu při vytápění a výrobě teplé užitkové vody. Součástí tohoto systému budou solární kolektory, tepelné čerpadlo, akumulární nádrž, potřebné měřicí a řídicí prvky.

Dále bych rád sbíral naměřená data, která se mohou později hodit, ať už mě kvůli zpětnému hodnocení, či k dalšímu vědeckému nebo nekomerčnímu použití, které bude řešit podobnou problematiku.

V neposlední řadě vytvořím webovou aplikaci. Pomocí této aplikace budu nasbíraná data vyhodnocovat a zobrazovat ve srozumitelné podobě například formou různých grafů. V této podobě budu moci výsledky své práce lépe prezentovat a zpřístupnit je široké veřejnosti.

2 Teoretická část

2.1 Rešerše

2.1.1 Seznámení s měřicími prvky

Pro vypracování mé bakalářské práce bude potřeba nasbírat velké množství dat. Tyto informace budu sbírat pomocí různých čidel a měřících přístrojů, přes teplotní čidla, průtokoměry a čítače elektrických impulzů. Tato čidla a další řídicí pomůcky byly zakoupeny od společnosti Papouch s.r.o..

Tyto přístroje bude nutné dobře zapojit a nastavit tak, aby vracely správné hodnoty ve správný čas. Za tímto účelem bude nutné prostudovat dokumentaci těchto produktů. Dokumentace se nachází přímo na oficiálních webových stránkách výrobce (papouch). Tyto materiály jsou velmi pěkně zpracované a obsáhlé, proto budou pro tuto část mé práce naprosto dostačující.

Pro nastavení některých komponent budou zapotřebí externí programy. Dokumentace k těmto programům se také nachází na výše zmíněné stránce.

2.1.2 Raspberry Pi

Pro řízení bude použito zařízení, které se jmenuje Raspberry Pi. Jedná se o malý počítač, na který se dá nainstalovat operační systém Linux. V mém případě se bude jednat o upravenou a zeštíhlenou verzi, vytvořenou speciálně pro tento počítač, která se jmenuje Raspbian.

Na oficiálních stránkách tohoto systému (raspbian) se nachází dokumentace a fórum, kde jsou popsány přesné návody jak operační systém nainstalovat. V případě, že bych potřeboval nějaké další informace, můžu použít jakoukoli knihu či internetový zdroj o operačním systému Linux z distribuce Debian, protože příkazy budou z 80% stejné nebo alespoň hodně podobné.

2.1.3 My SQL

Data budou ukládána do MySQL databáze. Na toto téma bylo sepsáno velké množství knih. Já sám vlastním knihu „Naučte se MySQL za 21 dní“ (Maslakowski, 2001). Tato kniha obsahuje 21 lekcí, které slouží pro základní pochopení práce s tímto jazykem.

Jako další zdroj by bylo určitě dobré uvést web (mysql) kde se nachází kompletní dokumentace funkcionality jazyka v anglickém jazyce. Samozřejmě existují i další internetové stránky a hlavně diskuze, které se zabývají touto problematikou, nicméně tyto dva zdroje by měly být pro potřeby mé bakalářské práce dostačující.

2.1.4 PHP

Jako komunikační prostředek mezi měřicími prvky a databází jsem si zvolil programovací jazyk PHP. Stejně jako u programovacího jazyku MySQL i zde existuje velké množství literatury. Byla mi doporučena kniha „Mistrovství v PHP 5“ (Gutmans, A. Bakken, S. S. Rethans, D.). V této knize se jde od úplných základů, až po pokročilou problematiku tohoto programovacího jazyka.

Na internetu se dá nalézt obrovské množství stránek a fór o této problematice. Za zmínku určitě stojí stránka (php). Tato stránka je brána jako oficiální dokumentace tohoto programovacího jazyka. Dá se tu najít popis jakékoli funkce, která v PHP existuje.

2.1.5 Tepelné čerpadlo

Součástí mé automatizované soustavy je i tepelné čerpadlo. I když ho nebudu zapojovat či nastavovat, určitě je vhodné si problematiku nastudovat, pro pochopení celku. Pro tyto účely mi poslouží kniha „Stavíme tepelné čerpadlo“ (ŽERAVÍK, A.).

2.2 Solární kolektory

„Fotovoltaický jev objevil *Alexandre Edmond Becquerel* v roce 1839. Vzájemným působením slunečního záření a hmoty dochází k pohlcování fotonů a uvolňování elektronů, v polovodiči pak vznikají volné elektrické náboje, elektron-díra, které jsou už jako elektrická energie odváděny ze solárního článku přes regulátor dobíjení do akumulátoru, nebo ke spotřebiči.

Sluneční světlo se skládá z fotonů - částic solární energie. Když foton dopadne na fotovoltaický článek, může být odražen, pohlcen, nebo může skrz tento článek projít. Jenom pohlcené fotony produkují elektřinu. V takovém případě je energie fotonu předána elektronu v atomu článku, jenž je z polovodičového materiálu.

Dopadne-li na fotovoltaický článek světelný paprsek, uvolní světlo některé elektrony z použitého materiálu. Tyto volné elektrony se elektrickým polem oddělí, takže v jedné vrstvě křemíku vznikne přebytek elektronů a ve druhé jejich nedostatek.

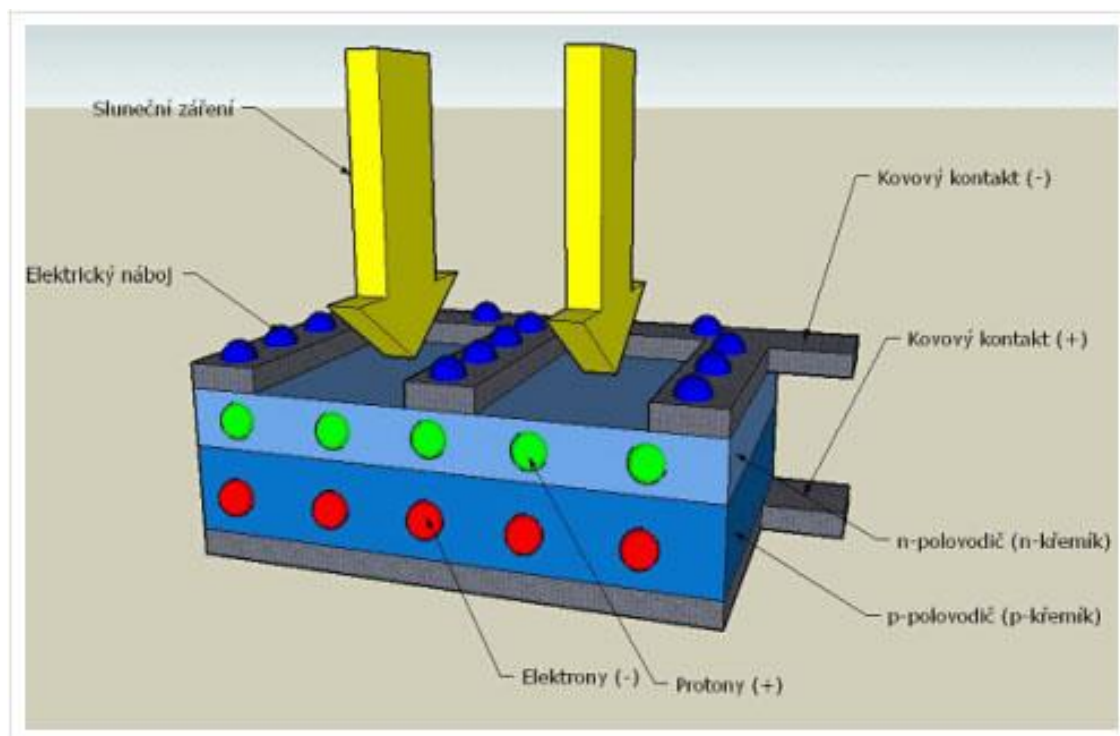
Propojením obou stran článku přes nějaký spotřebič vznikne uspořádaný tok elektronů, který nazýváme elektrický proud. Jeho velikost závisí od druhu materiálu.

Fotovoltaický sluneční článek je tedy polovodičová dioda mající velkou plochu, spodní celoplošný kovový kontakt (reflektor) a vrchní kovový kontakt (mřížku, hřeben) zabírající velmi malou plochu (4-8% plochy článku), aby nestínil.

Teoretická účinnost v případě článku z krystalického křemíku je okolo 30%. Praktická účinnost přeměny energie světelné na elektrickou je však u křemíkových článků pouze 4 až 13%, tedy dost nízká, což je hlavní nevýhodou získávání „sluneční“ elektrické energie pomocí fotovoltaických článků obecně. U článků z arzeni-

du galia (GaAs) bylo zatím dosaženo účinnosti 23%. Dnes jsou nejvíce rozšířeny fotovoltaické články na bázi křemíku (Si), neboť křemík má řadu výhod. Je hojně zastoupen v zemské kůře dokonce jako druhý nejrozšířenější prvek.

Výkon fotovoltaického panelu o ploše 1 m² může být v našich klimatických podmínkách až 150 W.“ (Hobza, 2009)



Obr. 1 Schéma fotovoltaického článku

2.3 Tepelné čerpadlo vzduch – voda

V rámci mé bakalářské práce bude použito tepelné čerpadlo fungující na principu vzduch – voda. Dále si popíšeme, jak takové čerpadlo funguje.

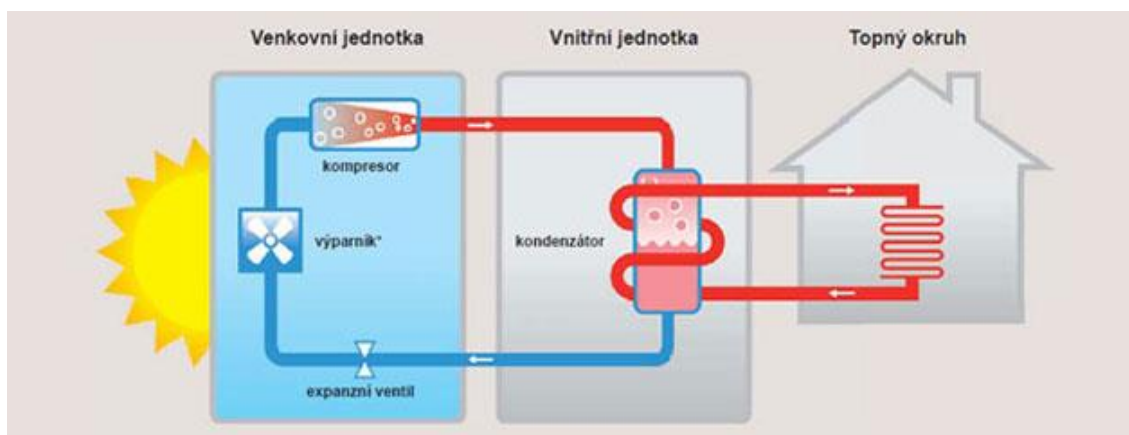
Na výparník tepelného čerpadla se přivádí vzduch, ze kterého se získává teplo. Tento typ má široké využití a vyžaduje nejméně náročnou investici. Mezi klady patří snadná instalace, kterou je možné provést téměř na všech místech bez potřeby dodatečných stavebních úprav. Topný faktor je závislý na teplotě vzduchu a v zimním období, kdy se teploty pohybují kolem -12 °C, je nutné zapínat další tzv. bivalentní zdroj energie.

Nevýhodami jsou vyšší provozní náklady a kratší životnost než je tomu u ostatních druhů tepelných čerpadel. Tvorba námrazy na výparníku, jejíž odstranění vyžaduje další energii a tím negativně ovlivňuje topný faktor. Náchylnost k tvorbě rušivého hluku při vyšších otáčkách ventilátoru.

Některá čerpadla bývají rozdělena na dva díly (vnitřní část a venkovní část), toto čerpadlo nazýváme oddělené tepelné čerpadlo, odborně split. Venkovní část obsa-

huje výparník, ventilátor a expanzivní ventil a je spojena s vnitřní částí, kde jsou obvykle umístěny zbývající díly. Tyto části jsou spojeny izolovanými měděnými trubkami, proto je toto řešení dražší než kompaktní provedení. Výhodou je, že kompresor, který je vydatným zdrojem hluku, je umístěn uvnitř objektu.

Kompaktní řešení tepelného čerpadla, kde jsou všechny prvky zařízení uvnitř jednoho celku, který je napojený na topný systém. Tento typ čerpadla může být umístěn venku, kde je nebezpečí zamrznutí a ovlivňování výkonu kolísáním teplot, nebo uvnitř objektu ve sklepě nebo v podkroví. Zde vzniká nutnost přivádět venkovní vzduch k výparníku pomocí potrubí. (Žeravík, 2003)



Obr. 2 Schéma tepelného čerpadla vzduch-voda

2.4 Akumulační nádrž

V dnešní době, kdy nejsou vyvinuty dostatečně výkonné akumulátory, se jako nejrozmumnější řešení jeví energii, která není okamžitě spotřebována, ukládat v podobě tepla, které se potom využije k vytápění objektu či přehřívání užitkové vody před jejím finálním ohřevem a požadovanou teplotu.

Pro přeměnu elektrické energie na energii tepelnou budou použity topné spirály, které se budou spínat a rozpínat podle aktuální výroby a spotřeby elektrické energie. Jednotlivé topné spirály mají příkon 500 W a celkem jich bude použito osm. Díky těmto topným spirálám bude akumulační nádrž ukládat až 4 kW nespoteřebované energie bez nutnosti ji vracet zpátky do rozvodné sítě. Po jejím uložení je žádoucí, aby energie v nádrži zůstala. Proto je nádrž zaizolována 5 cm skelné vaty, na níž je ještě 20 cm polystyrénu.

Pro uložení vyrobené energie bude použita akumulační nádrž o rozměrech 340 cm x 170 cm x 100 cm. Díky těmto rozměrům pojme objem 5700 l. Na ohřátí tohoto množství vody o 1 °C bude zapotřebí přibližně 8,8 kWh energie.

2.4.1 Výměníky

- Výměník pro teplou užitkovou vodu – výměník, který bude odebírat z akumulační nádrže teplo pro ohřev užitkové vody. Výměník bude vyrobený

z mědi a jeho rozměry budou 50 m na délku s průměrem 22 mm. Jeho přibližná celková plocha tedy bude 3,5 m².

- Výměník pro podlahové vytápění – výměník, který bude odebírat teplo pro vytápění celého rodinného domu. Výměník bude vyrobený z mědi a jeho rozměry budou 2 x 25 m na délku s průměrem 22 mm. Jeho přibližná celková plocha tedy bude 3,5 m².

2.5 TME

TME je jednoduché teplotní čidlo s rozhraním Ethernet. Senzor má rozpětí měření mezi -55°C až +125°C. Měřenou hodnotu je možné číst několika způsoby, mimo jiné i z interních webových stránek. Komunikace probíhá různými protokoly z rodiny TCP/IP. Díky tomu lze vybrat způsob vhodný pro danou aplikaci. Teplota je vždy k dispozici přímo ve stupních Celsia. Nejsou nutné přepočty pomocí matematických funkcí, které vyjadřují tepelnou závislost. Umožňuje i komunikaci SNMP protokolem (UDP), přes MODBUS TCP, také umí odeslat e-mail při překročení nastavených mezí, apod. Teplotu z TME lze snadno vložit do vlastních webových stránek pomocí metody GET. Výsledek měření je možné číst také z XML souboru nebo jej odesílat jako HTTP POST v XML formátu ke zpracování webové službě. (Papouch, 2012)



Obr. 3 TME multi

2.6 TX20ETH

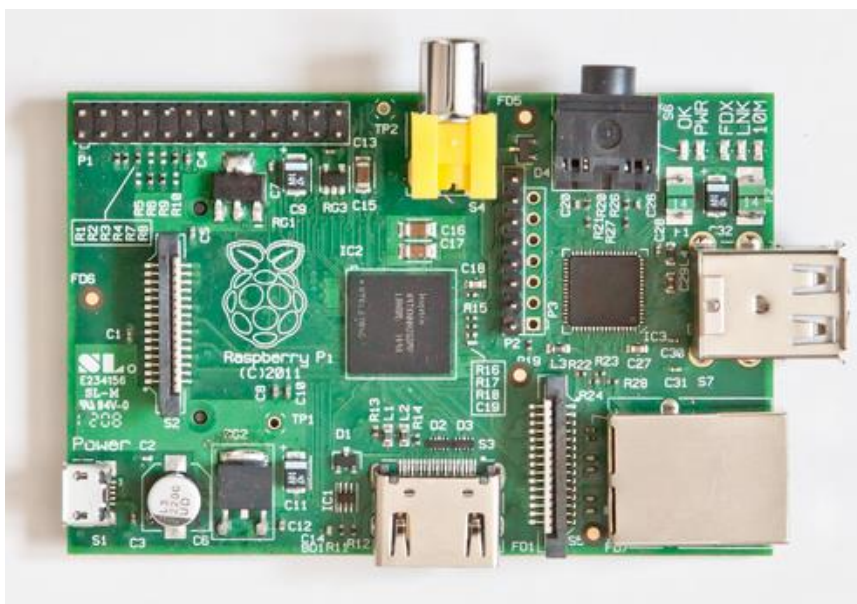
TX20ETH je ethernetový anemometr, který je konstruován na měření směru a rychlosti větru. Připojuje se přímo k počítačové síti (Ethernetu). Naměřené údaje jsou k dispozici jak pro lidskou, tak pro strojovou obsluhu. Směr a rychlost větru je vidět na interním webovém rozhraní, které nebude v mé práci využito. Při překročení mezí umí TX20ETH upozornit obsluhu e-mailem. Pokud jde o strojovou obsluhu, umí TX20ETH komunikovat datovým kanálem po TCP, protokoly SNMP a MODBUS TCP. Také umí odesílat hodnoty na vzdálený server pomocí jednoduchých požadavků HTTP GET. Hodnoty z TX20ETH i nastavení jsou snadno dostupné ve formátu XML. (Papouch, 2009)



Obr. 4 TX20ETH s venkovním čidlem

2.7 Raspberry pi

Raspberry pi je malý počítač o trochu větší než platební karta. Výpočetní výkon zajišťuje procesor ARM1176JZF-S s taktem 700 MHz a 512 MB operační paměti. Data a operační systém jsou uloženy na paměťové kartě. Jako operační systém můžou být použity systémy linuxového typu. K Raspberry pi je možné kromě napájení připojit klávesnici a myš. Obrazový výstup je zajištěn pomocí HDMI slotu. Pro připojení k síti se používá klasický konektor pro kroucenou dvoulinku.



Obr. 5 Raspberry pi

2.8 Q8

Quido ETH je I/O modulů s digitálními vstupy, výstupy a teploměrem. Vstupy jsou určeny pro připojení napětí nebo kontaktu – rozlišují dva stavy (0 a 1). Výstupy jsou relé s přepínacím kontaktem. Teploměr může být připojen na kabelu délky až 15 metrů a je schopen měřit teploty v rozsahu až od -55 do +125 °C. Quida se připojuje přímo k počítačové síti LAN (Ethernetu) a lze je kompletně ovládat a spravovat přes webové rozhraní.

2.8.1 Vlastnosti

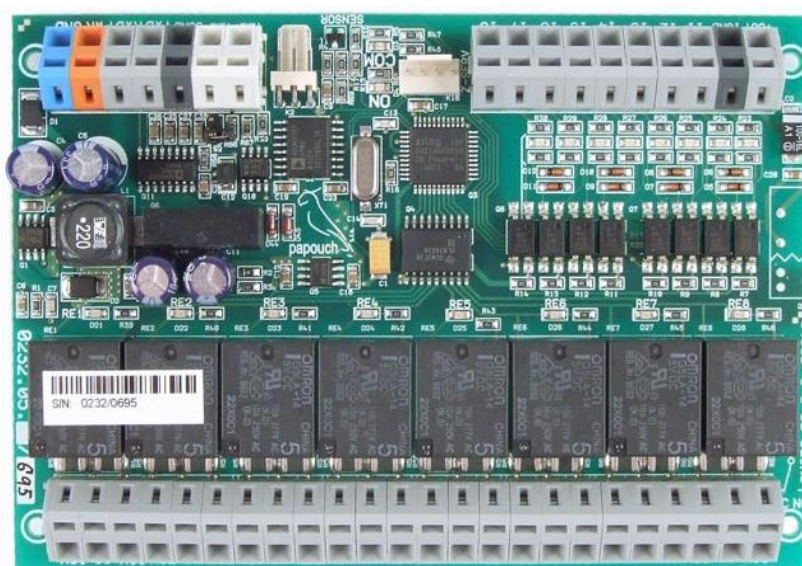
- Galvanicky oddělené digitální vstupy pro napětí nebo kontakt.
- Výstupy typu přepínací kontakt relé.
- Teploměr s měřicím rozsahem -55 až +125 °C.
- Připojení přímo k počítačové síti LAN (Ethernetu).
- Indikace zapnutí, komunikace a stavu vstupů a výstupů kontrolkami.
- Napájení z externího zdroje 8 až 30 V.

2.8.2 Ovládání:

- Standardními protokoly: MODBUS TCP2, SNMP, HTTP GET, Spinel2.
- Pro všechny moderní webové prohlížeče včetně OS iPhone a Android.
- Zabezpečení přístupu zvlášť pro prohlížení a zvlášť pro konfiguraci.
- Volitelné stavové symboly vstupů a výstupů.
- Softwarem Wix.

2.8.3 Základní funkce

- Čtení aktuálního stavu vstupů.
- Automatické odesílání informace (email, HTTP GET, ...) o změně na vstupech.
- Počítání impulsů na vstupech nebo počítání změn stavu vstupu (pro prvních 60 vstupů).
- Ovládání výstupních relé s přepínacím kontaktem.
- Nastavování výstupů na definovanou dobu.
- Teploměr: měření teplot -55 až +125 °C.
- Funkce hlídání teploty (sepnutí relé při různých pohybech teploty, také sepnutí na nastavenou dobu při dosažení zadané teploty, odeslání e-mailu nebo HTTP GET). (Papouch, 2015)



Obr. 6 I/O modul Q8

2.9 PHP

2.9.1 Proč PHP

- PHP je relativně jednoduché na pochopení.
- PHP má syntaxi velmi podobnou jazyku C a je tedy většině vývojářů dost blízký.
- PHP podporuje širokou řadu souvisejících technologií, formátů a standardů.
- Je to otevřený projekt s rozsáhlou podporou komunity.
- Dají se najít knihovny či zdroje již hotového kódu k okamžitému použití nebo funkční PHP aplikace. Podstatná část z hotového kódu je šířena pod nějakou svobodnou licencí a dá se použít ve vlastních projektech.
- PHP dobře komunikuje s webovým serverem Apache.

- PHP snadno komunikuje s databázemi, jako je MySQL, PostgreSQL a řada dalších.
- PHP je multiplatformní a lze jej provozovat s většinou webových serverů a na většině dnes existujících operačních systémů.
- PHP podporuje mnoho existujících poskytovatelů webhostingových služeb.

2.9.2 Nevýhody PHP

- PHP je interpretovaný, ne kompilovaný jazyk.
- kdokoli má přímý přístup k serveru, může nahlédnout do vašich PHP skriptů.
- Podpora objektového programování není v PHP na moc dobré úrovni.
- Protože je PHP aktivně vyvíjen, v budoucích verzích jazyka se mohou některé funkce změnit nebo se mohou chovat jinak než dosud.

(Zajíc, 2004)

2.10 Mysql

MySQL je velmi populární databáze. Podle mnoha zdrojů je to rovněž velmi rychlá databáze. Nemá však tolik funkcí a možností jako některé konkurenční databázové systémy. Vybrat si vhodnou databázi je tedy klasický kompromis mezi rychlostí softwaru a jeho schopnostmi.

Pro účely mé práce budou základní funkce MySQL dostačující. Proto není žádný důvod pro použití komplexnějších systémů jako například PostgreSQL.

2.11 WYSIWYG Web Builder

WYSIWYG Web Builder je vizuální webový editor pro vytváření webových stránek. WYSIWYG je akronym anglické věty „What you see is what you get“, česky „co vidíš, to dostaneš“. Tato zkratka označuje způsob editace dokumentů v počítači, při kterém je verze zobrazená na obrazovce vzhledově totožná s výslednou verzí dokumentu.

Jinými slovy WYSIWYG znamená, že hotové stránky se zobrazí přesně tak, jak bylo navrženo. Můžete si vytvořit webové stránky bez znalostí HTML. Vytváření probíhá jednoduše umístováním jednotlivých prvků na požadované místo. To znamená, že jednotlivé prvky se usazují na obrazovku přímo do zvoleného místa a WYSIWYG Web Builder následně sám zajistí vytvoření odpovídajícího zdrojového HTML kódu.

WYSIWYG Web Builder podporuje vkládání bloků textu, obrázků, tabulek, prvků Java, ActiveX, Flash, Windows Media apod. přímo do internetových stránek. Svou práci si můžete ihned prohlédnout v náhledu. Web Builder poskytuje plnou kontrolu nad obsahem a vzhledem webových stránek. Součástí programu je také FTP klient, jehož prostřednictvím se hotové stránky rovnou umístí na web.

(WYSIWYG Web Builder,2012)

3 Návrh

3.1 Výchozí stav

Předtím než jsem začal s touto prací, již byly některé komponenty nainstalovány a využívány. Jedná se především o fotovoltaické kolektory, tepelné čerpadlo a akumulční nádrž.

Fotovoltaické kolektory byly instalovány v červenci roku 2012. Bohužel v této době již nebyly udíleny státní dotace na výrobu elektrické energie tímto způsobem. Díky tomu se výkupní cena v dnešní době pohybuje ve výši 0,40 Kč/kWh. Při spotřebování vyrobené energie zaplatíme 0,90 Kč/kWh a při jejím nákupu z rozvodné sítě 3,30 Kč/kWh. Z toho vyplývá, že spotřebovat energii, kterou si sami vyrobíme nás vyjde na 1,30Kč/kWh, což je podstatně lepší než ji za 0,40 Kč prodat a poté ji opět za 3,30 Kč nakoupit.

Dalším již instalovaným úsporným zařízením bylo tepelné čerpadlo, které bylo nainstalováno v říjnu roku 2013. V této době bylo připojeno přes malý výměník přímo s okruhem podlahového topení a řízeno pomocí ekvitermní regulace dodávané výrobcem zařízení. Princip ekvitermní regulace spočívá v tom, že je nastavena požadovaná teplota v místnosti, kterou se přístroj snaží udržovat. Tím dochází k častému vypínání a zapínání přístroje. Tento stav se ani zdaleka neblíží stavu ideálnímu, protože při poklesu teploty se přístroj zapne na vysoký výkon, aby co nejrychleji místnost vytopil. Při tomto krátkodobém vysokém výkonu se tepelné čerpadlo ničí a nedosahuje takového poměru výkonu k příkonu jako by mohlo, kdyby běželo delší dobu při nižším příkonu. Další nepříjemný jev, který díky tomu nastává, je především během zimních měsíců kondenzace vodní páry ze vzduchu na venkovním výměníku a jeho následným namrzáním. V případě že tento stav nastane, se čerpadlo přepne na opačný chod, aby se odmrazilo, čímž vznikají další náklady na jeho provoz a podstatně se tím snižuje jeho účinnost.

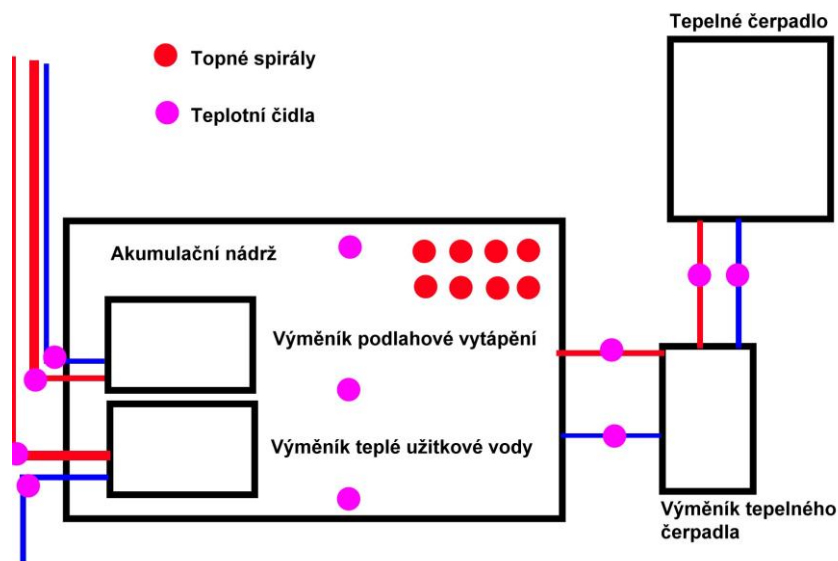
Poslední komponentou byla akumulční nádrž, která nebyla využívána. K nádrži by bylo dobré alespoň uvést její tepelnou izolaci, která je vypočtena v následující tabulce.

Tab. 1 Tabulka s výpočtem tepelné ztráty akumulční nádrže.

Součinitel tepelné vodivosti izolace:	λ	0,033	W/m.K
Tloušťka izolace:	d	250	mm
Součinitel přestupu tepla na vnějším povrchu nádrže:	α_e	10	W/m ² .K
Součinitel přestupu tepla na vnitřním povrchu nádrže:	α_i	lze zanedbat	
$U = \frac{1}{\frac{1}{\alpha_i} + \frac{d}{\lambda} + \frac{1}{\alpha_e}}$			
Součinitel prostupu tepla izolace:	U	0,130	W/m ² .K
Teplota v místnosti:	t _i	20	°C
Teplota vody v nádrži:	t _v	35	°C
Povrch nádrže:	S	22	m ²
Tepelné ztráty:	Q	43	W
$Q = U \times A \times \Delta t$			
Tepelné ztráty za den:		1,03	kWh

3.2 Návrh automatizované soustavy

Pro zvýšení efektivity systému bude zapotřebí zajistit ukládání energie vyrobené pomocí fotovoltaických článků. Pro uložení bude použita akumulční nádrž, která bude za tímto účelem doplněna o topné spirály. Při instalovaném výkonu fotovoltaických článků o 4,4 kW, bude dostačující, použít osm spirál o příkonu 500 W. Dalšího významného zlepšení by se mělo dosáhnout připojením nádrže do celého systému, především propojením nádrže s tepelným čerpadlem. Díky tomu že bude tepelné čerpadlo ohřívat větší množství vody, se nám povede odstranit většinu nepříznivých jevů, které jsou s jeho používáním spjaté. Návrh soustavy je naznačen v obrázku (obr. 1).



Obr. 7 Automatizovaná soustava

3.3 Návrh měřicí soustavy

Aby mohl systém fungovat a měl se podle čeho řídit, musí nějakým způsobem získávat data. Tato data budou získávána pomocí různých čidel.

3.3.1 Údaje o teplotě

Tato data budou získávána pomocí jedenácti teplotních čidel rozmístěných po automatizované soustavě podle (Obr. 1). Jednotlivá čidla budou zapojena sériově a přivedena do sběrnice (TME). Odtud již bude možné získaná data vzít a poslat si použitím metody http GET k dalšímu zpracování. Další teplotní čidlo bude umístěno mimo objekt pro snímání venkovní teploty. Toto čidlo bude připojeno k modulu Q8.

3.3.2 Údaje o výrobě/spotřebě elektrické energie

Měření této veličiny bude probíhat pomocí počítání impulzů. V mém případě budu počítat náběhy jednotlivých impulzů za určitý čas. Dále bude potřeba zjistit, kolik impulzů dá dohromady 1 kW elektrické energie. Potom již jednoduše přepočítám pomocí vzorce a získám výslednou výrobu/spotřebu elektrické energie. Jako čítač impulzů bude možné použít modul Q8.

3.3.3 Měření rychlosti a směru větru

Posledním měřicím prvkem použitým v systému je anemometr, který je umístěn na střeše objektu. Data naměřená pomocí tohoto přístroje, budou posílána metodou

http GET. Poznatky získané tímto přístrojem sice přímo neovlivňují soustavu, avšak budou použity při zvažování případného rozšíření soustavy o větrnou elektrárnu.

3.4 Návrh řídicí soustavy

Řídicí soustava bude sestavena ze tří prvků a to malého počítače Raspberry pi, spínacího modulu Q8 a webového serveru, na kterém poběží MySQL databáze a výsledná webová aplikace.

Na Raspberry pi poběží operační systém linux Raspbian dále webový server apache2 a bude zde nainstalováno PHP. Dále bude potřeba nastavit statickou IP adresu, aby se k přístroji dalo přistupovat vzdáleně a mohl být umístěn v rozvodné skříni. Nakonec již bude stačit nahrát PHP skripty a správně nastavit daemon cron, aby spouštěl skripty, když bude potřeba.

Raspberry pi bude posílat instrukce modulu Q8, který se bude starat o spínání topných těles a tepelného čerpadla. K modulu Q8 budou muset být přidána ještě spínací relátka, aby bylo možné spirály spínat, aniž by došlo k poškození modulu.

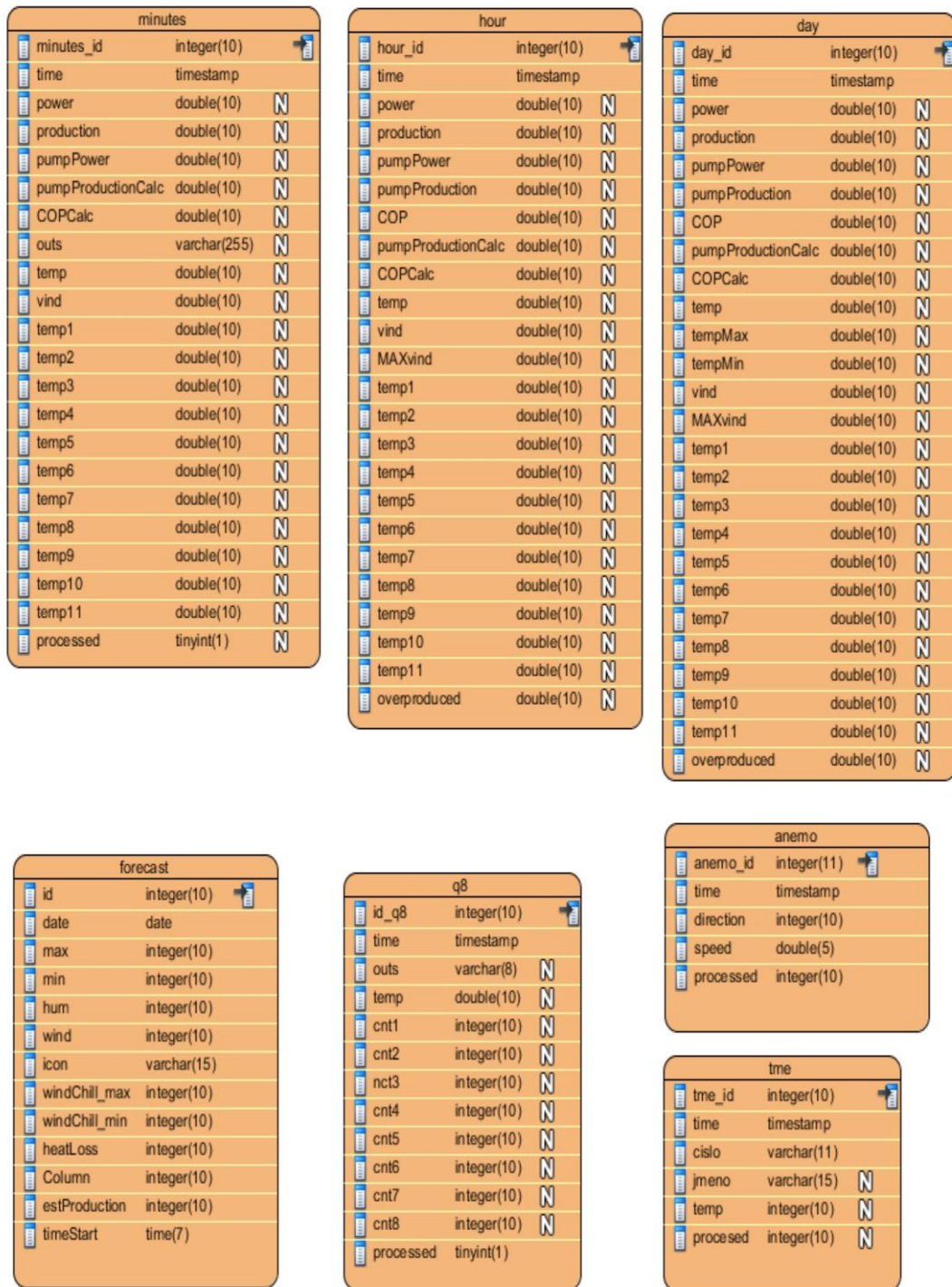
Poslední věcí, kterou bude Raspberry pi obsluhovat bude komunikace s databázovým serverem, od kterého si bude brát data, určená k řízení či vkládat naměřená data určená k archivaci pro další využití.

3.5 Návrh software

Použitý software by se dal rozčlenit do 3 stěžejních částí. První bude MySQL databáze pro skladování naměřených dat dále PHP skripty, které budou zajišťovat funkčnost systému a v poslední řadě vyrobení uživatelsky přijatelné formy výstupu pomocí programu Wysiwyg Web Builder.

3.5.1 MySQL databáze

Databáze bude umístěna na vzdáleném serveru a její návrh bude uveden v následujícím obrázku pomocí erd diagramu.



Obr. 8 ERD návrh databáze

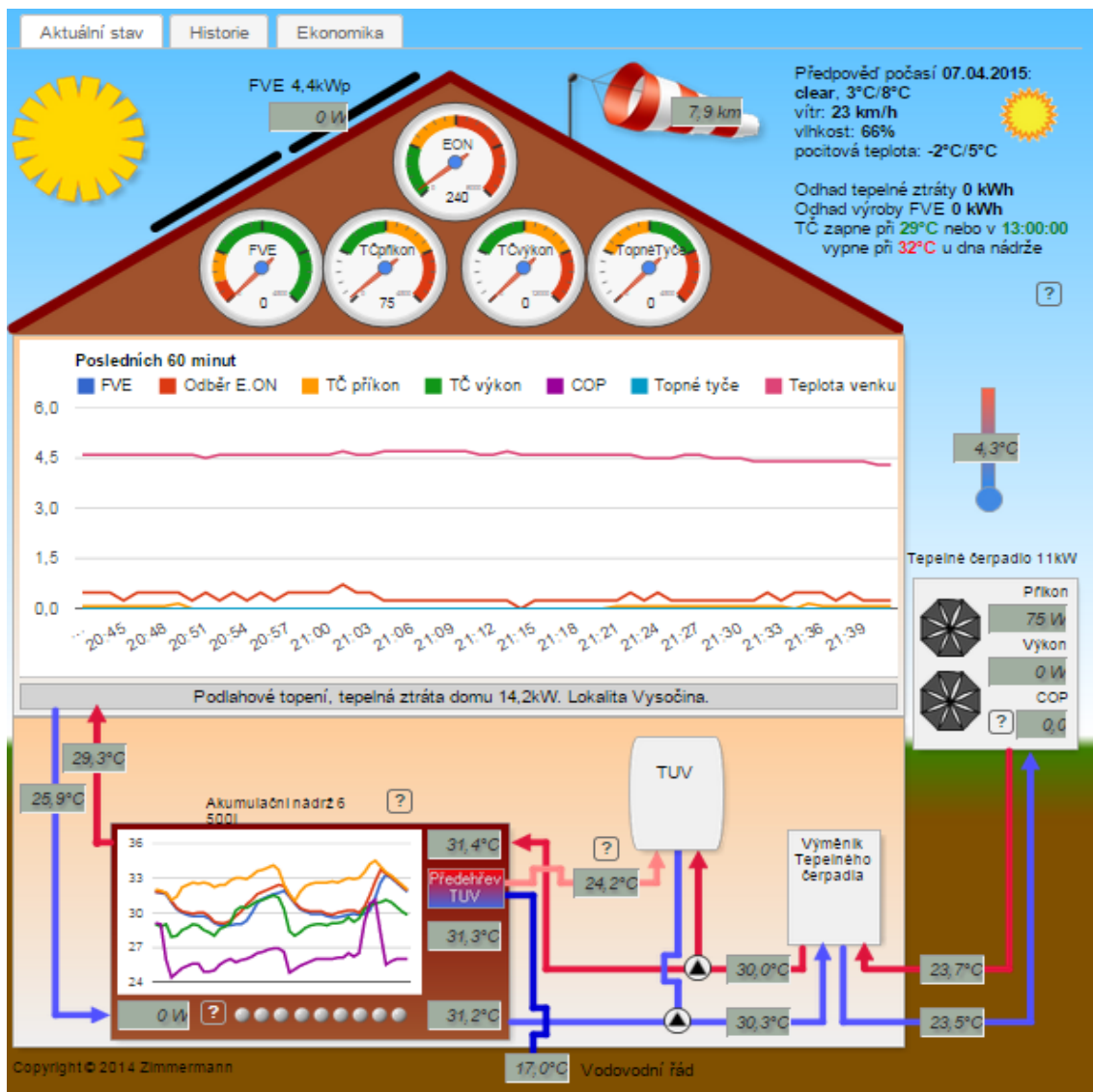
3.5.2 PHP skripty

Tato část mojí práce bude zdaleka tou nejzdlouhavější a nejobtížnější. Pro plnění databáze daty budou určeny skripty *q8.php*, *sum_min.php*, *sum_hod.php*, *sum_day.php*, které budou pracovat s naměřenými daty a budou zajišťovat jejich uložení do databáze. Tyto data budou doplněna ještě pomocí skriptu *anemo.php* o data naměřená pomocí anemometru a skriptem *forecast.php*, kde bude umístěn parser, pomocí něhož bude z internetové *api.wunderground.com* získávána předpověď počasí. Pomocí skriptu *delete.php* bude jednou denně databáze čištěna od přebytečných dat, která již byla zpracována. Všechny tyto skripty budou umístěny na raspberry pi a budou volány v pravidelných časových intervalech pomocí *de*-*annou* *cron*.

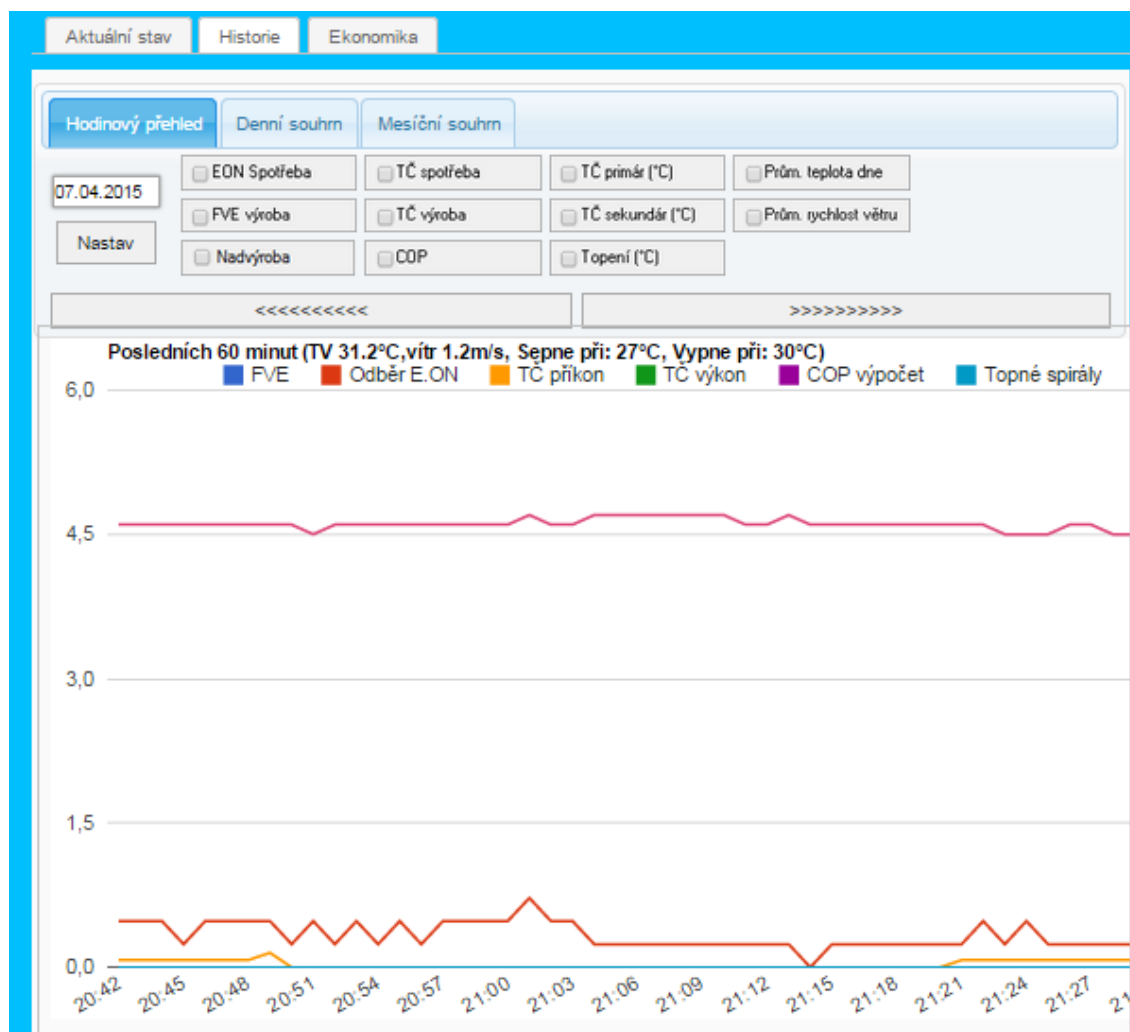
Další skupina skriptů se bude starat o to, aby byla naměřená data převedena do grafů, díky čemuž budou daleko přehlednější. Tyto skripty budou čtyři a bude se jednat o *graf_nadrz.php*, *graf_min.php*, *graf_hod.php*, *graf_day.php*. Tyto skripty budou umístěny na webovém serveru a budou volány přímo webovou aplikací.

3.5.3 Návrh vizualizace

Webovou aplikaci jsem se rozhodl dělat pomocí programu *Wysivyg Web Builder*, což mi značně ulehčí práci s psáním *html* kódu a jeho stylováním pomocí *css*. Návrh vizualizace se bude skládat ze dvou částí. Na úvodní stránce budou aktuální naměřená data. Dále se bude moci přepnout do záložky historie, kde bude mít uživatel možnost procházet grafy za zvolené časové období s daty, které si sám zvolí. Grafický návrh bude zobrazen v následujících dvou obrázcích.



Obr. 9 Návrh vizualizace webové stránky s aktuálním stavem



Obr. 10 Návrh vizualizace webové stránky s historií

4 Praktická část

V této části mé bakalářské práce bude popsáno veškeré nastavování instalace a zapojení, které bylo třeba učinit pro to, aby byl systém plně funkční.

4.1 Instalace operačního systému Raspbian

Na začátku je dobré si přichystat veškeré vybavení, které budeme potřebovat, takže si seženeme klávesnici a myš do USB 2.0, dále monitor či televizi se vstupem pro hdmi konektor, síťový kabel a dostatečně velkou a pokud možno i rychlou paměťovou kartu typu SD. Poté si na nějakém jiném počítači stáhneme operační systém, jaký budeme chtít instalovat, v mém případě Raspbian. Systém se stáhne v iso formátu, takže budeme potřebovat ještě externí program pro nahrání tohoto obrazu na SD kartu. Program se jmenuje Win32DiskImager. Po jeho nainstalování a spuštění, vybereme cestu ke staženému iso souboru s operačním systémem, vložíme SD kartu a vybereme volbu *write*. Po dokončení této operace je již systém nainstalovaný. Kartu vložíme do raspberry pi a připojíme napájení. Po chvíli se objeví tabulka se základním nastavením. Zde nic měnit nebudeme, jenom zvolíme volbu *expand_rootfs*, díky které se nám systémový oddíl na kartě rozšíří na celou její kapacitu. Poté zvolíme volbu *Finish* a systém se restartuje. Po restartu se objeví příkazová řádka a systém je připraven k použití. Pokud bychom se z nějakého důvodu někdy potřebovali dostat zpět na nastavovací obrazovku, stačí zadat příkaz *raspi-config*.

4.2 Nastavení systému Raspbian

V této části si popíšeme veškeré kroky, které jsou nutné udělat k tomu, aby na raspberry bylo schopné komunikovat s SQL databází bylo na něm možné spouštět PHP skripty, a také k němu vzdáleně přistupovat protože určitě nebudeme chtít mít v rozvodné skříni monitor myš a klávesnici.

První věc, která je po instalaci potřeba udělat je zavolat 2 příkazy *apt-get upgrade* a *apt-get update*. Tím si zajistíme, že náš systém bude v aktuálně nejnovější verzi a všechno bude fungovat tak, jak má.

Dále bude potřeba nainstalovat webový sever apache . Ten si stáhneme a nainstalujeme pomocí příkazu *apt-get install apache2 -y*. Pro ověření že vše proběhlo v pořádku, se můžeme přepnout do operačního systému a spustit webový prohlížeč, kam zadáme localhost. Pokud se nám zobrazí stránka, na které bude napsáno „It works!“, všechno proběhlo v pořádku.

Další komponentou, kterou je třeba nainstalovat je PHP. PHP nainstalujeme jednoduše pomocí příkazů *apt-get install php5 libapache2-mod-php5 -y*. PHP si můžeme otestovat například tak, že si vytvoříme nějaký jednoduchý PHP skript, který nám bude pomocí příkazu *echo* vypisovat „Hello world“. Ten zkopírujeme do složky */var/www/*, a poté si ho zavoláme z internetového prohlížeče. Pokud na obra-

zovce uvidíme „Hello world“, vše je v pořádku a můžeme přistoupit k dalšímu kroku.

Dalším krokem bude nastavení statické ip adresy. Kdyby se adresa přidělovala dynamicky, tak bychom k raspberry pi nemohli vzdáleně přistupovat, a to z toho důvodu že bychom nevěděli kam. Příkazem `nano /etc/network/interfaces` se dostaneme do síťového konfiguračního souboru. Zde nás bude zajímat třetí řádek, kde je napsáno `iface eth0 inet dhcp`. Tento řádek přepíšeme na: `iface eth0 inet static`. Dále přidáme následující řádky:

Address + ip adresa na kterou chceme raspberry pi přidělit

Netmask + maska sítě

Network + adresa sítě

Broadcast + broadcast sítě

Gateway + výchozí brána sítě.

Potom konfigurační soubor uložíme a ukončíme textový editor nano. Změny se projeví až po restartu zařízení. Zařízení můžeme restartovat příkazem `reboot`. Pro kontrolu můžeme zadat následující příkaz `ip a s`. Uvidíme výpis, ve kterém bude položka s ip adresou, která by měla být stejná jako ta, kterou jsme zadávali do konfiguračního souboru. Pro jistotu můžeme ještě zkusit ping na nějakou stránku, která není v naší síti například `ping www.seznam.cz`. Tímto příkazem si ověříme, že nám správe funguje DNS server. Pro ukončení pingu zmačkneme `ctrl + c`.

Při vývoji tohoto systému se mi několikrát stalo, že SD karta přestala fungovat a narušil se její bootovací sektor. Potom došlo třeba k výpadku proudu a raspberry se již nespustilo. SD karta má omezený počet zápisů, takže je dobré omezit jakékoli zapisování na minimum. Nicméně operační systém na kartu sám zapisuje logovací soubory. Naším cílem bude, aby se tak nadále nedělo, a proto přesměrujeme zápis těchto souborů do operační paměti. Aby se tak stalo, zadáme příkaz `nano /etc/fstab`. Do tohoto souboru vložíme následující příkazy:

```

GNU nano 2.2.6                                     File: /etc/fstab
proc          /proc        proc          defaults      0      0
/dev/mmcbk0p1 /boot        vfat          defaults      0      2
/dev/mmcbk0p2 /            ext4          defaults,noatime 0      1
tmpfs /tmp tmpfs        defaults,noatime,nosuid,size=100m 0 0
tmpfs /var/tmp tmpfs        defaults,noatime,nosuid,size=30m 0 0
tmpfs /var/log tmpfs        defaults,noatime,nosuid,size=100m 0 0
tmpfs /var/run tmpfs        defaults,noatime,nosuid,size=2m 0 0
#tmpfs /var/spool/mqueue tmpfs defaults,noatime,nosuid,mode=0700,gid=12,size=30m 0 0

# a swapfile is not a swap partition, so no using swapon|off from here on, use
# dphys-swapfile swap[on|off] for that

```

Obr. 11 fstab

Soubor uložíme a provedeme restart zařízení. Po restartu už by se nemělo na SD kartu nic samovolně ukládat.

Ted' když je všechno nainstalované a nastavené by bylo dobré, vytvořit si zálohu pro případ že karta přestane fungovat. Pro vytvoření zálohy lze použít stej-

ný program jako pro instalaci systému. Po jeho spuštění zadáme cestu, kam chceme zálohu vytvořit a zvolíme volbu *read*. Bohužel záloha je o trochu větší než zálohovaná karta, proto jsem instalaci prováděl na kartu menší, než jsem měl v úmyslu používat. Zálohu 4 Gb karty nahrajete na kartu 8 Gb, a pokud nechcete přijít o nevyužitě místo, jednoduše zadáme příkaz *raspi-config* a zvolíme volbu *expand_rootfs*. Tím se nám systémový oddíl se rozšíří na celou její velikost.

4.3 Nastavení quida Q8

Připojíme stejnosměrné napájecí napětí z rozsahu 8 až 30 V ke svorkám plus k oranžové svorce a mínus k modré. Quido má integrovanou ochranu proti poškození přepólováním napájecího napětí. Po připojení napájení se na desce rozsvítí kontrolka.

Nyní připojíme quido k počítačové síti. Quido je z výroby nastavené na síť s maskou 255.255.255.0. Pokud máte jiný rozsah sítě, můžete toto nastavení změnit použitím programu Ethernet configurator. Tento program je ke stažení na internetových stránkách výrobce. Po nastavení správné ip adresy se již budeme moci dostat do webového rozhraní, kde je možné provést další potřebná nastavení. Nastavení je uspořádáno do následujících deseti panelů:

1. Síť – konfigurace síťového rozhraní, maska, IP adresa, TCP kanál a portu Modbusu
2. Zabezpečení – nastavení bezpečnostních hesel pro přístup do webu
3. E-maily – odesílání emailů o aktuálním stavu a změnách
4. SNMP – integrace do Vašeho síťového SNMP manageru
5. Odesílání – odesílání stavů na Váš server pomocí HTTP GET + nastavování stavů
6. Vstupy – konfigurace vstupů, obrázků, jejich jmen, počítadel
7. Výstupy – konfigurace výstupů, jejich jmen, hlídání teploty, obrázků
8. Teploměry – hlídání teploty
9. Ostatní – název zařízení, teplotní jednotky, výběr jazyka webu
10. Info – informace o zařízení

Pro nás budou nejdůležitější dvě části, a to především síť a odesílání, které si trochu víc přiblížíme.

4.3.1 Sít'

Sít'	Zabezpečení	E-maily	Odesílání	Vstupy	Výstupy	Teploměry	Ostatní	Info	
Nastavení sítě									
IP adresa zařízení	10.0.0.98								
Maska sítě	255.0.0.0								
IP adresa brány	0.0.0.0								
Port webového rozhraní	80								
IP adresa DNS serveru	0.0.0.0								
Lokální port pro TCP/UDP spojení	10001								
Port pro MODBUS TCP	502								
Režim zařízení?									
Komunikační režim	WEB ▼								
Vzdálené zařízení pro TCP/UDP spojení?									
Vzdálená IP adresa	0.0.0.0								
Vzdálený port	0								
								Reset	Uložit

Obr. 12 Nastavení sítě u Q8

4.3.2 Odesílání

Sít'	Zabezpečení	E-maily	Odesílání	Vstupy	Výstupy	Teploměry	Ostatní	Info
Nastavení pro HTTP GET								
Adresa webového serveru	10.0.0.100							
Port webu	80							
Adresář skriptů na serveru	mereni/							
Název skriptu	q8.php							
Periodické odesílání aktuálních hodnot	0							
Odesílat GET při změnách	<input checked="" type="checkbox"/>							
Příklad: HTTP GET z Quido ETH 8/8: www.server.net/script.php?mac=00-20-4A-B4-8D-F7&name=Office&ins=01101010&outs=00100010&tempS=0&tempV=21.8&cnt2=235&cnt6=126								
Tip: Pokud posíláte HTTP GET na server v jiné síti, je třeba mít také správně nastavenou IP adresu brány na záložce Sít'.								
								Uložit

Obr. 13 Nastavení odesílání u Q8

4.4 Nastavení teploměrů TME

Po připojení teploměru k napájení a síti bude nutné změnit IP adresu. Zařízení připojíme přímo k jednomu počítači kříženým kabele a použijeme program Ethernet configurator, kterým jsme nastavovali quido Q8. Po nastavení adresy připojíme teploměr k síti a můžeme začít nastavovat jednotlivá čidla. Čidla se zapojují sériově, ale pro jejich nastavení je musíme připojovat jedno po druhém, protože jsou všechna z výroby nastavena na stejné číslo. Pro jejich nastavení použijeme program, který si stáhneme od výrobce. Nachází se v souboru tme.zip. Potom co pro všechna čidla nastavíme různá čísla, je můžeme připojit všechna do série. Poté se připojíme k zařízení pomocí programu PuTTY. Zde zvolíme protokol Telnet a zadáme port 9999. Poté se nám zobrazí základní nabídka (obr.14).

```
Change Setup:
 0 Server configuration
 1 Device 1 Ethernet
 2 Modules
 7 factory defaults
 8 exit without save
 9 save and exit      Your choice ?
-----
```

Obr. 14 Základní nabídka teploměru TME.

Zde zvolíme volbu 1 která slouží pro nastavení zařízení TME. Po potvrzení volby se dostaneme do nastavovacího dialogu. Pokud něco splete musíme celým nastavením projít znovu protože zde není možné nic vracet ani mazat. Po úspěšném nastavení vypadá výsledek následovně (viz obr.15)

```

***** Device settings *****
Device name       : (Regulace)
Use TCP server (0) or UDP (2) or Web (3)
Mode             : (3)
Source Port      : (10001)
Destination Port : (0)
Destination IP   : 10.0.0.0
HTTP Port (internal WEB) : (80)
Sending period Email or Trap (19282) min:

--- SNMP ---

Read community   : (er@HSLd)
Write community  : (7)
Enabled Trap Y/N : (N)

--- SMTP (e-mail) ---

Enabled Email Y/N : (N)

--- HTTP and SOAP settings ---

GUID             : (10-0-0-99)
URL path (HTTP GET) : ( )
Remote HTTP port  : (80)
Remote HTTP server IP : 10.0.0.100
Script address    : (mereni/tme.php)
Sending period HTTP : (10) s

Send HTTP SOAP(Y) or GET(N) : (N)
-----

```

Obr. 15 Výpis po nastavení modulu TME

Po tomto nastavení bude třeba ještě nastavit jednotlivá čidla. Proto abychom tento krok mohli provést, zvolíme volbu 2. Po jejich nastavení bychom měli vidět výpis (viz obr.16).

```

Module 1       : Enable
Module 1 name  : (VyparnikTepla)
Module 1 type  : (TQS)
Module 1 adres HEX: (01)
Module 1 watch : (temperature)
Module 1 max value : (+125.0 C)
Module 1 min value : (-055.0 C)
Module 1 hys   : (+000.0 C)
-----

```

Obr. 16 Výpis jednoho čidla po nastavení

Poté se můžeme připojit k zařízení přímo z webového prohlížeče a měli bychom všechna nastavená čidla vidět i s aktuálními naměřenými hodnotami.

4.5 Nastavení anemometru

Po zapojení anemometru bude opět třeba změnit rozsah sítě. To uděláme tak, že přístroj připojíme napřímo k nějakému počítači a změníme základní výrobní nastavení pomocí programu Ethernet configurator. Potom již bude možné se připojit pomocí webového prohlížeče a všechna ostatní nastavení již provést zde. Opět nás bude především zajímat nastavení sítě a odesílání. Nastavení bude podobné jako u modulu Q8 s tím rozdílem, že se GET nebude odesílat při změně, ale pravidelně jednou za minutu.

4.6 PHP skripty

Tato část mé práce se zabývala implementací PHP skriptu. Zcela jistě se jedná o část, která je nejvíce časově náročná. Skripty použité v mé bakalářské práci by se daly rozdělit do tří částí podle toho, jakou funkcionalitu zajišťují. Skripty z první části se starají o odchycení dat a jejich uložení do databáze. Druhá skupina skriptů provádí nad těmito daty operace a vytváří z nich data nová. Třetí skupina se potom stará o převedení dat do uživatelsky přívětivější podoby grafu.

4.6.1 Skripty pro sběr dat

Tyto skripty jsou umístěné na raspberry pi. Zde čekají, dokud jim nepřijde http GET. Když se tak stane, spustí se, převezmou si data, která jim byla zaslána a uloží je do databáze.

Skriptu *q8.php* přijde zpráva ve tvaru: *www.server.net/script.php?mac=00-20-4A-B4-8D-F7&name=Office&ins=01101010&outs=00100010&tempS=0&tempV=21.8&cnt2=235&cnt6=126*, kterou následně zpracuje následující PHP kód:

```
if(!empty($_GET))
```

Touto podmínkou si ověříme, zdali GET nepřišel prázdný. Pokud ano, nebudeme dělat nic. Pokud není prázdný, vezmeme si z něj potřebná data následujícím kódem.

```
$udaje["outs"]=$_GET["outs"];  
$udaje["tempv"]=$_GET["tempV"];  
$udaje["cnt1"]=$_GET["cnt1"];  
$udaje["cnt2"]=$_GET["cnt2"];  
$udaje["cnt3"]=$_GET["cnt3"];  
$udaje["cnt4"]=$_GET["cnt4"];  
$udaje["cnt5"]=$_GET["cnt5"];
```

```
$udaje["cnt6"]=$_GET["cnt6"];
$udaje["cnt7"]=$_GET["cnt7"];
$udaje["cnt8"]=$_GET["cnt8"];
```

Tímto si do proměnné udaje uložíme data z proměnné GET. Ve chvíli kdy už máme data, můžeme vkládat do databáze. Vložení provedeme následujícím SQL dotazem.

```
mysql_query("INSERT INTO q8 (outs,temp,cnt1,cnt2,cnt3,cnt4,cnt5,cnt6,cnt7,cnt8) VALUES
('.$udaje['outs'].','.$udaje['temp'].','.$udaje['cnt1'].','.$udaje['cnt2'].','.$udaje['cnt3'].','.$udaje['cnt4'].','.$udaje['cnt5'].','.$udaje['cnt6'].','.$udaje['cnt7'].','.$udaje['cnt8'].')")
```

Dalšími skripty z této skupiny jsou *tme.php* a *anemo.php*. Skriptu *tme.php* a *anemo.php* přichází zpráva v podobném tvaru jako u skriptu Q8. Zpracování těchto dat již nebudu popisovat, protože je totožné jako u skriptu q8.php.

Posledním skriptem, který přináší do systému nějaká nová data je skript *parser.php*. Tento skript parsuje předpověď počasí z XML souboru který si stáhne na webových stránkách <http://api.wunderground.com>. Celý příkaz pro stažení a uložení do proměnné potom bude vypadat následovně:

```
$promenna = simplexml_load_string (file_get_contents
("http://api.wunderground.com/api/8a81de11b00b08cd/forecast /lang:CZ/q/trebic.xml"));
```

Tento xml soubor obsahuje předpovědi na tři dny, počínaje dneškem. Nás bude zajímat vždy jen předpověď na zítřek, proto si xml soubor prohledáme pomocí foreach cyklu a výběr ještě omezíme pomocí podmínky jenom na ten den, který nás zajímá. Kód bude vypadat následovně:

```
foreach($promenna -> forecast -> simpleforecast -> forecastdays ->forecastday as $key=> $day){
if($i == 2){
$max = $day->high->celsius;
$min = $day ->low->celsius;
$hum = $day -> avehumidity;
$wind = $day -> avewind -> kph;
$den = $day -> date ->day;
$mesic = $day -> date -> month;
$rok = $day -> date -> year;
$icon = $day -> icon;
} $i++; }
```


Teď již máme data z předpovědi počasí téměř nachystaná. Ještě bude potřeba proměnné `den`, `mesic`, `rok` přeformátovat do tvaru tak, jak ho zná databáze. Výsledek bude v proměnné `date` a získáme ho následujícím kódem:

```
$mesic = str_pad($mesic, 2, "0", STR_PAD_LEFT);  
$den = str_pad($den, 2, "0", STR_PAD_LEFT);  
$date = $rok."-".$mesic."-".$den;
```

Teď již máme všechny proměnné pro vložení záznamu do databáze. O veškerá data získaná pomocí skriptů z této skupiny se již dále budou starat skripty, které si popíšeme v následující podkapitole.

4.6.2 Skripty pro práci nad daty

V této části se budeme zabývat skripty, které budou také běžet na Raspberry pi. Jejich volání bude probíhat voláním v pravidelných intervalech pomocí demonu `crone`. Tyto skripty se budou starat o data v databázi, zajišťovat jejich mazání průměrování a přesuny mezi jednotlivými tabulkami. Jedná se o skripty `sum_min.php`, `sum_hod.php`, `sum_day.php` a `delete.php`.

Nejvíce rozsáhlým skriptem z této skupiny je `sum_min.php`. Tím že je volán každou minutu, se totiž nabízí i k samotnému řízení celé soustavy. Rychlejší reakce systému na změny by byly zbytečné, protože teplota vody se skokově nemění. Výroba a spotřeba elektrické energie sice ano, ale pokaždé, když mrak zastíní fotovoltaické kolektory, které důsledkem toho, přestanou vyrábět elektrickou energii.

Pro řízení bude nejprve potřeba získat data. Ty si vytáhneme z tabulky `q8`, protože s nimi budeme dále procovat. Tyto data získáme následujícím SQL dotazem:

```
$result = mysql_query("SELECT id_q8,cnt1,cnt2,cnt3,time,outs FROM q8 WHERE time > (NOW() -  
INTERVAL 1 MINUTE)");  
$predMinutou = mysql_fetch_array($result,MYSQL_ASSOC);
```

Poté si budeme muset zjistit, kolik impulzů za minutu přišlo. Toho docílíme tak, že projdeme proměnnou `predMinutou` cyklem `for`, a pokud se hodnota z předchozího průchodu liší od stávající, navýšíme si počítadlo o jejich rozdíl. Problém může nastat při vynulování čítače po dosažení maximální hodnoty. Proto je zde podmínka, že musí být následující číslo větší nebo stejné než předchozí. Pokud podmínka neprojde, znamená to, že se počítadlo vynulovalo a přijdeme o jeden impulz. V dalším průchodu již bude vše fungovat tak jak má. Počítadlo se nuluje po dosažení hodnoty 65 535, takže je tento jeden impulz zanedbatelný. PHP kód řešící tuto problematiku potom vypadá následovně:

```

for ($i = 0 ; $i < mysql_num_rows($result); $i++) {
    $dotazNow = mysql_fetch_array($result,MYSQL_ASSOC);
    if ( $dotazNow['cnt1'] >= $k ) {
        $m = $dotazNow['cnt1'] - $k;
        $dum = $dum + $m;
    }
    $k = $dotazNow['cnt1'];
}

```

Poté, co známe počet impulzů, můžeme je jednoduše přepočítat na elektrickou energii. Přepočet je v následujícím kódu:

```

$spotreba = $spotreba * 4 * 60;
$kolektor = $kolektor * 60;
$spotrebaCerpadlo = $spotrebaCerpadlo *250/200*60;

```

Pokud známe proměnné spotreba, kolektor a outs máme dostatek informací k tomu, abychom mohli začít spínat topné spirály. To bude probíhat ve třech krocích. V prvním kroku si zjistíme, jaká je přibližná aktuální spotřeba. Spočítáme aktuálně sepnuté spirály plus spotřebu tepelného čerpadla, plus přibližnou konstantu spotřeby domu.

```

$wat = $x * 500 + 750+ $spotrebaCerpadlo;

```

V proměnné x je počet aktuálně sepnutých spirál. Tento počet jsem získal tak, že jsem si vzal proměnnou outs a pomocí case ji procházel stav po stavu, dokud jsem nenalezl ten, který byl totožný s předlohou. Podle stavu proměnné outs jsem již schopný říci, kolik topných těles je aktuálně sepnuto.

V dalším kroku rozhodneme, kolik spirál sepneme nebo rozepneme. Připínat budeme vždy maximálně jednu spirálu, rozepnout je však můžeme všechny. Tento problém budeme řešit následujícím kódem:

```

$x = 0;
if ( $spotreba < 1 AND $kolektor > $wat ) {
    $x = 1 ; }

```

Stejným způsobem pokračujeme až do konce, kdy proměnné x přiřadíme -9.

V tuto chvíli už víme, kolik spirál připnout nebo odepnout. Ovládání probíhá pomocí příkazu:

```
file_get_contents("http://10.0.0.98/set.xml?type=A&sts=1000xxxx");
```

Kde 1 znamená sepnuto 0 rozepnuto a x ponechání beze změny. Přiložím ukázkou kódu, která je zkrácená a opět se stejným způsobem pokračovalo, až do případu kdy byly všechny stavy sepnuté.

```
switch ($outs) {  
  
    case 0000:  
  
        if ($x > 0) {  
  
            file_get_contents("http://10.0.0.98/set.xml?type=A&sts=1000xxxx");  
  
            }  
  
            break;  
  
        case 1000:  
  
            if ($x > 0 ) {  
  
                file_get_contents("http://10.0.0.98/set.xml?type=A&sts=0100xxxx");  
  
                }  
  
            if ($x < 0) {  
  
                file_get_contents("http://10.0.0.98/set.xml?type=A&sts=0000xxxx");  
  
                }  
  
            break;
```

Dále budeme potřebovat získat údaje z teplotních čidel. Ty budeme selektovat z tabulky tme. Ta se plní každou minutu, proto může dotaz vypadat následovně:

```
$result = mysql_query("SELECT temp FROM tme WHERE cislo = 1 ORDER BY tme_id DESC LIMIT  
1 ");  
  
$T1 = mysql_fetch_array($result,MYSQL_ASSOC);
```

Stejný dotaz bude pro všech jedenáct čidel. Změní se pouze parametr, u atributu číslo.

Podle získaných teplot se řídí tepelné čerpadlo. To je spínáno a rozpínáno pomocí následujícího kódu:

```
if($T5<=30.7){
```

```

file_get_contents("http://10.0.0.98/set.xml?type=A&sts=xxxx0xxx");
}

if($T5>31.6){
file_get_contents("http://10.0.0.98/set.xml?type=A&sts=xxxx1xxx");
}

```

Ted' už si stačí připravit data z předpovědi počasí. Tato data máme uložena v tabulce forecast. Zajímat nás bude pouze poslední záznam. Po tom, co tato data získáme, můžeme udělat záznam do tabulky minutes. Sql insert bude vypadat následovně:

```

mysql_query("INSERT INTO minutes (power, production, pumpPower, pumpProductionCalc, COPCalc, outs, temp, vind, temp1, temp2, temp3, temp4, temp5, temp6, temp7, temp8, temp9, temp10, temp11) VALUES (". $spotreba. ", ". $kolektor. ", ". $spotrebaCerpadlo. ", ". $vykoncerpadlovypocet. ", ". $vykonost. ", ". $dotazPosledni[outs]. ", ". $T. ", ". $vitr. ", ". $T1. ", ". $T2. ", ". $T3. ", ". $T4. ", ". $T5. ", ". $T6. ", ". $T7. ", ". $T8. ", ". $T9. ", ". $T10. ", ". $T11. ")");

```

Tímto bych skript *sum_min.php* opustil. Dalším skriptem je skript *sum_hod.php*. Ten čerpá hodnoty, ze stejných tabulek jako skript minutový. Data by se mohla brát i z minutové tabulky, ale takto dosáhneme vyšší přesnosti. Data, která tento skript zpracovává, jsou označena jako zpracovaná a následně budou smazána skriptem *delete.php*. Skript *sum_day.php* už pouze průměruje data z tabulky hour a vkládá je do tabulky day.

4.6.3 Skripty pro vytváření výstupů

V této části je několik skriptů, které se zabývají vykreslováním naměřených dat do grafů. Všechny jsou si hodně podobné, proto funkčnost popíši pouze na jednom z nich. Všechny tyto skripty budou volány přímo framem ve kterém se budou zobrazovat. Pro vykreslení grafu byla použita technologie google charts. Celé to funguje tím způsobem, že odešlete data v určitém formátu a vrátí se vám krásný graf. V mém případě to bylo trochu složitější, protože jsem si data, která se budou v grafu zobrazovat, chtěl sám vybrat. Díky tomu bude graf přehlednější a můžu se podívat jen na to, co mě zajímá. Prvně bude tedy nutné si podle vybraných věcí vytvořit proměnné či řetězce, které potom odešleme ke zpracování. Bude následovat ukázka kousku kódu, který se touto problematikou zabývá.

```

$promenna2 = "Year";

$promenna = array();

```

```
if(!empty($_POST)){  
    if(!empty($_POST['Kalendar'])){  
        $kalendar = $_POST['Kalendar'];  
    }  
    if(!empty($_POST["spotreba"])){  
        $spotreba = $_POST["spotreba"];  
        $promenna2 =$promenna2.",'Odběr E.ON'";  
        $promenna[] = 'spotreba';  
    }  
    if(!empty($_POST["vyroba"])){  
        $vyroba = $_POST["vyroba"];  
        $promenna2 =$promenna2.",'FVE'";  
        $promenna[] = 'vyroba';  
    }  
}
```

Po naplnění těchto dvou proměnných, kde proměnná2 obsahuje legendu k výslednému grafu a pole promenna stringi s názvy proměnných. Selectovat budeme všechny sloupce z tabulky, ke které je graf určen. Samotné volání grafu přiložím do příloh.

4.7 Démon cron

Démon cron bude sloužit k automatickému volání php skriptu. Bude se jednat především o skripty, které se budou starat o databázi. Skript *sum_min.php* se bude volat každou minutu, *sum_hod.php* každou hodinu a *sum_day.php* jednou denně. Dále pak *delete.php* jednou denně ve tři hodiny a patnáct minut, kdy není skoro žádná spotřeba a žádná výroba elektrické energie, a nejsou spuštěny žádné jiné skripty. Posledním je *praser.php*, který se spouští v jedenáct hodin večer, kdy je předpověď na další den co nejpřesnější. Cron funguje tak, že se vytvoří zápis do jeho tabulky. Do tabulky se dostaneme příkazem *crontab -e*. V mém případě tabulka vypadá následovně.

```
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow  command
* * * * * /usr/bin/php /var/www/mereni/sum_min.php
0 * * * * /usr/bin/php /var/www/mereni/sum_hod.php
0 1 * * * /usr/bin/php /var/www/mereni/day.php
15 3 * * * /usr/bin/php /var/www/mereni/delete.php
0 23 * * * /usr/bin/php /var/www/mereni/parser.php
```

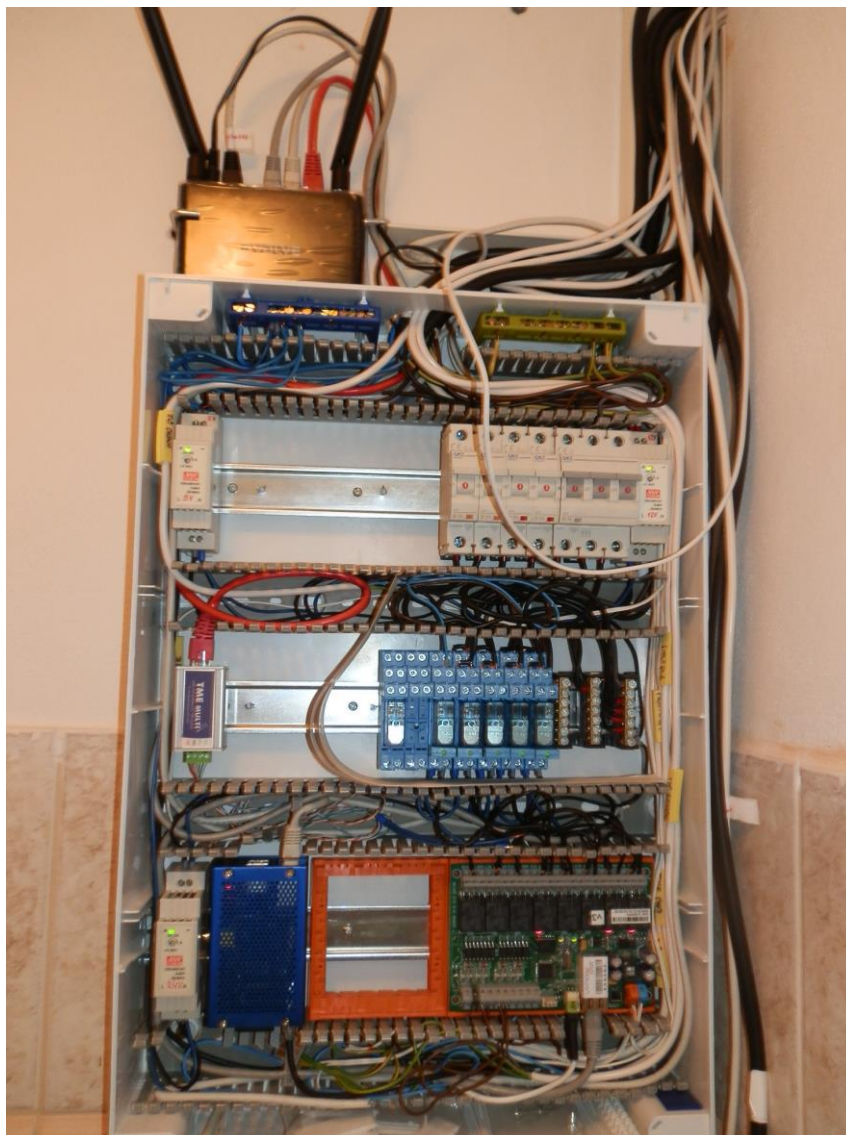
Obr. 17 Crontab

4.8 Vytvoření webové stránky

Webová stránka byla vytvořena v programu WYSIWYG Web Builder . Jedná se o pěkně udělaný intuitivní program. Celá moje práce byla přetáhnout obrázky na pozici, kde jsem je chtěl mít. V nabídce jsou již hotová menu, takže si stačilo vybrat a přetáhnout na pozici. V místech, kde jsem chtěl mít své grafy, jsou umístěny frejmy. Program obsahuje i různé budíky či teploměry. Krásně se tu dá vytvořit i formulář pro odeslání dat. Po přetáhnutí všech komponent na své místo, je možné na jakoukoli komponentu dvakrát poklepat myší a vyskočí vám okno s veškerým nastavením, které je k ní dostupné. U menu si například nastavíte jeho chování při klepnutí na tlačítko či jen po přejetí myši. Dále si nastavíte požadovaný vzhled či stránky, na které nás menu odkáže. U formuláře si nastavíte data, která má odeslat při zaškrtnutí checkboxu a kam data odeslat při stisknutí tlačítka. Framu stačí pouze nastavit stránku, která se v něm má zobrazovat. Když všechno vypadá tak, jak chceme a je správně nastaveno stačí kliknout na tlačítko publish a program nám již sám vytvoří plně funkční HTML stránku, kterou umístí na web.

4.9 Finální zapojení a uspořádání

Ve chvíli kdy je všechno nastavené a připravené k použití už mi zbývalo jenom všechno uspořádat do rozvodné skříně, aby bylo vše na jednom místě a přehledně zapojené. Fotka rozvodné skříně je na následující fotce (obr.18)



Obr. 18 Zapojení rozvodné skříně.

5 Ekonomické zhodnocení

Tab. 2 Tabulka s pořizovací cenou

Název	počet	Cena	Celkem
TME – Ethernetový teploměr	1	2 990 Kč	2 990 Kč
Quido ETH 8/8	1	4 400 Kč	4 400 Kč
Teploměr	11	1 200 Kč	13 200 Kč
Raspberry pi	1	880 Kč	880 Kč
Wifi router	1	1 000 Kč	1 000 Kč
Anemometr s Ethernetem	1	4 900 Kč	4 900 Kč
Relátka finder 10A	8	200 Kč	1 600 Kč
Práce	200	70 Kč	14 000 Kč
Celkem			42 970 Kč

5.1 Vzniklé úspory

Jedna z úspor systému je to že spotřebovává vyrobenou energii namísto jejího vrácení do rozvodné sítě. Touto cestou vzniká roční úspora přibližně 3 000 Kč.

Dalším úsporným opatřením je nahrazení systému řízení tepelného čerpadla, které má mnoho pozitivních efektů. Prvním z nich je ten, že čerpadlo je spouštěno jenom jednou denně namísto předchozích deseti až patnácti vypnutí a zapnutí. Díky tomu může běžet celou svou pracovní dobu na nižší konstantní příkon. Tím dochází k odstranění namrzání venkovního výměníku a čerpadlo má celkově lepší poměr příkon/výkon. Další pravděpodobný efekt této změny bude prodloužení životnosti tepelného čerpadla o 30 – 50%. Když vezmeme v úvahu všechny tyto faktory, tak díky této změně můžeme ročně ušetřit až 13 000 Kč.

5.2 Návratnost investice

Celý systém řízení i s odvedenou prací vyšel na 42 970 Kč. Roční úspory, které zavedením systému vznikají, se pohybují mezi 10 000 – 15 000 Kč. Z toho vyplývá, že návratnost instalovaného systému je někde mezi třemi až čtyřmi roky s ohledem na vnější jevy.

6 Diskuse

6.1 Jsou v plánu nějaká vylepšení?

System řízení je kompletní a nebudou v něm prováděny žádné změny. V případě potřeby se mohou změnit nastavené hodnoty pro řízení, ale to je pouze malá úprava, která nevyžaduje žádné velké zásahy do kódu.

System bych do budoucna rad rozšířil o další položku ve webovém výstupu. Moje představa je taková, že pod položkou ekonomika, kterou mám již předpřipravenou, bude přepočít zabývající se návratnosti investice a počítadlem úspor. To že to v systému zatím není ničemu nevadí, jde spíše jenom o zajímavost.

6.2 Jakých úspor se pomocí zavedení systému dosahuje?

Zavedením systému se dosahuje poměrně vysokých úspor za vytápění a ohřev užitkové vody. Výše úspor je poměrně hodně závislá na aktuálním počasí, proto jsou uvedené hodnoty pouze orientační. Oproti stavu před zavedením systému, jsou roční úspory okolo 10 000 – 15 000 Kč.

6.3 Bylo by možné systém udělat levněji?

Řídící systém vyšel přibližně na 10 000 Kč. Tato cena by šla snížit, kdyby nebyla použita zařízení od firmy Papouch s.r.o. Na druhou stranu se nákupem těchto komponent ušetřilo hodně práce.

7 Závěr

Cílem této práce je řídit systém pro snížení a optimalizaci energetické náročnosti rodinného domu při vytápění a výrobě teplé užitkové vody. Tato řídicí část byla splněna. Systém je již v provozu a jeho provoz je bezproblémový. Podařilo se i dosáhnout nemalých úspor při vytápění a ohřevu teplé užitkové vody.

Další částí práce bylo vytvoření webové stránky na které by byla naměřená data prezentována. I tato část bakalářské práce byla splněna. Na webové stránce proto můžete vidět nejen aktuální data ale i data naměřená v minulosti. Je možné si vybrat zdali chceme historii prohlížet po dnech, měsících či se rovnou podívat na vývoj dat ze celý rok. Pomocí formuláře si můžeme vybrat jen ta data, která nás zajímají, což je dobré pro zvýšení přehlednosti grafu. Webová stránka je k nahlédnutí na adrese www.zimmermann.cz/regulace.

8 Literatura

HOBZA, OTAKAR. 2009. *Fotovoltaika.* ., [Http://nadrevo.blogspot.cz/](http://nadrevo.blogspot.cz/) [online]. [cit. 2015-05-07]. Dostupné z: <http://nadrevo.blogspot.cz/2009/02/slunecni-kolektory-druhy-ucinnost.html>

PAPOUCH S.R.O. 2015. *Quido-eth.pdf* [online]. [cit. 2015-05-07]. Dostupné z: http://www.papouch.com/cz/website/mainmenu/clanky/vyberte-si/io-pro-ethernet-usb-rs485-rs232/quido-eth.pdf/_downloadFile.php

PAPOUCH S.R.O. 2012. *Tme.pdf* [online]. [cit. 2015-05-07]. Dostupné z: http://www.papouch.com/cz/shop/product/tme-ethernetovy-teplomer/tme.pdf/_downloadFile.php

PAPOUCH S.R.O. 2009. *Tx20eth.pdf* [online]. [cit. 2015-05-07]. Dostupné z: http://www.papouch.com/cz/shop/product/tx20eth-ethernetovy-anemometr/tx20eth.pdf/_downloadFile.php

WYSIWYG WEB BUILDER. 2012. *Http://www.wysiwygwebbuilder.cz/* [online]. [cit. 2015-05-07]. Dostupné z: <http://www.wysiwygwebbuilder.cz/index.php>

ZAJÍC, PETR. 2004. *PHP - Historie a budoucnost.* [Http://www.linuxsoft.cz/](http://www.linuxsoft.cz/) [online]. [cit. 2015-05-07]. Dostupné z: http://www.linuxsoft.cz/article.php?id_article=171

ŽERAVÍK, ANTONÍN. *Stavíme tepelné čerpadlo: [návratnost i za jeden rok]*. 1. vyd. Pře-
rov: Antonín Žeravík, 2003, 311 s. ISBN 80-239-0275-X.

Přílohy

A Ukázka kódu pro vytvoření grafu

```
$mesic2 = substr($kalendar, 0, 2);

$rok2 = substr($kalendar, 3, 4);

$scass = $rok2."".$mesic2;

$result = mysql_query("SELECT * FROM day WHERE EXTRACT(YEAR_MONTH FROM time)=$scass.");

$radku = mysql_num_rows($result);

echo" <html>

<head>

<meta charset='UTF-8'>

<script type='text/javascript' src='https://www.google.com/jsapi'></script>

<script type='text/javascript'>

google.load('visualization', '1', {packages:['corechart']});

google.setOnLoadCallback(drawChart);

function drawChart() { ";

echo("var data = google.visualization.arrayToDataTable([";

echo("[". $promenna2."],");

for ($i=0; $i<$radku; $i++) {

$now = mysql_fetch_array($result,MYSQL_ASSOC) ;

$spotreba = $now['power']/1000 ;

$vyroba = $now['production']/1000 ;

$scspotreba = $now['pumpPower']/1000;

$scvyroba = $now['pumpProduction']/1000;

$stemp = $now['temp'];

$scas2 = $now['time'];

$mesic = substr($scas2, 5, 2);

$den = substr($scas2, 8, 2);
```

```
$hod = substr($cas2, 11, 2);
$cas = $den;
$cas = intval($cas);
$overproduced = $now['overproduced']/1000;
$COP = $now['COP'];
$vind = $now['vind'];

    echo "[".$cas;

    foreach($promenna as $p){

    echo (",".$p);

    }

    echo"]";

    if ($i<($radku-1)) {
echo(",");
}
}

echo("]");

$neco = $mesic."/". $rok2;
echo "    var options = { title: '$neco',

        chartArea: { left: 40,width:'95%',height:'85%', right:5},

        legend:{position:'top',alignment:'center'},

        curveType: 'function'

    };

    var chart = new google.visualization.LineChart(document.getElementById('chart_div'));

    chart.draw(data, options);

}

</script>
</head>
```

```
<body>
  <div id='chart_div' style='width: 740px; height: 630px;'></div>
</body>
</html> ";
mysql_close($con);
?>
```