

**ČESKÉ VYSOKÉ
UČENÍ TECHNICKÉ
V PRAZE**

**FAKULTA
STROJNÍ**



**DOMÁCÍ AUTOMATIZACE
S RASPBERRY PI**

2017

**MAREK
MACH**

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Mach** Jméno: **Marek** Osobní číslo: **408926**
Fakulta/ústav: **Fakulta strojní**
Zadávající katedra/ústav: **Ústav přístrojové a řídicí techniky**
Studijní program: **Strojírenství**
Studijní obor: **Informační a automatizační technika**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Domácí automatizace s Raspberry Pi

Název bakalářské práce anglicky:

Home automation with Raspberry Pi

Pokyny pro vypracování:

- Zprovoznění senzorů a rádiového vysílače přes Raspberry Pi.
- Zprovoznění vlastního serveru a databáze na Raspberry Pi.
- Napsání jednoduché webové stránky kde bude možné informace číst.
- Naprogramovat Rasp. aby v případě definovaných vstupů upozornilo.

Seznam doporučené literatury:

NORRIS, Donald. "Raspberry Pi: projekty. 1. vyd. Brno." (2015).

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Cyril Oswald, U12110.3

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Ing. Matouš Cejnek, U12110.3

Datum zadání bakalářské práce: **19.04.2017** Termín odevzdání bakalářské práce: **16.06.2017**

Platnost zadání bakalářské práce: _____

Podpis vedoucí(ho) práce

Podpis vedoucí(ho) ústavu/katedry

Podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací.
Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

Datum převzetí zadání

Podpis studenta

Poděkování

Rád bych poděkoval vedoucímu mé práce, Ing. Cyrilovi Oswaldovi, který mi ochotně radil při plnění stanovených úkolů a přinášel náměty na další možný vývoj, jenž by vedl k úspěšnému dokončení úkolu.

Dále děkuji Pavle a Jitce za korekturu práce a Ondřejovi za cenné připomínky.

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně s tím, že její výsledky mohou být dále použity podle uvážení vedoucího bakalářské práce jako jejího spoluautora. Souhlasím také s případnou publikací výsledků bakalářské práce nebo její podstatné části, pokud budu uveden jako její spoluautor.

V Praze dne

.....

Podpis

Anotační list

Název Práce: Domácí automatizace s Raspberry Pi

Klíčová slova: Akční členy; Automatizace; Chytrá domácnost; Cloud; Internet of Things; Python; Raspberry Pi; Senzory; Server; Smart Home

Abstrakt:

Cílem této práce je představit možnosti prostředků, pomocí kterých se dá vytvořit systém domácí automatizace a navrhnout konkrétní řešení realizace.

Bakalářská práce obsahuje v první části rešerši na téma možností prostředků pro vytvoření automatické domácnosti, a to zejména z oblasti reportování stavů, zjištěných pomocí senzorů, a případného upozornění na neobvyklé nebo nebezpečné stavy. V druhé části práce obsahuje popis realizace domácího reportovacího zařízení pomocí Raspberry Pi, vlastního cloud serveru a jednoduché webové stránky pro čtení údajů ze senzorů.

Abstract:

The goal of this thesis is to present the possibilities of how one can create a system of home automation and to suggest a concrete solution for realization.

This bachelor thesis in the first part contains the research on the possibilities of means for creation of an automated household, especially in the area of reporting statuses detected by sensors, and eventual warning of unusual or dangerous situations. In the second part of the thesis, there is a description of the implementation of the domestic reporting device using Raspberry Pi, private cloud server and a simple web page for reading sensor data.

Obsah

1. Úvod.....	4
1.1. Slovo úvodem.....	4
1.2. Volba tématu	4
1.3. Cíl práce	5
2. Teoretická část.....	6
2.1. Hardware	6
2.1.1. Řídící moduly	6
2.1.2. Senzory	12
2.1.3. Akční členy	14
2.2. Software.....	18
2.2.1. Operační systém.....	18
2.2.2. Server	18
2.2.3. Programovací jazyk	19
3. Praktická část.....	20
3.1. Hardware	20
3.1.1. Řídící jednotka	20
3.1.2. Senzory	21
3.1.3. Doplnky	23
3.1.4. Kompletace senzorů	23
3.2. Software.....	25
3.2.1. Operační systém.....	25
3.2.2. Language of Things.....	25
3.2.3. Server	27
3.2.4. Programovací jazyk	27
3.3. Naprogramování řídicího systému	28
3.3.1. Realizovaný postup přípravy operačního systému	28
3.3.2. Příprava softwaru pro komunikaci se senzory	29
3.3.3. Spuštění vlastního serveru	30
3.3.4. Naprogramování užitečných rozšíření.....	31
3.4. Závěr praktické části	33
4. Závěr	36
5. Reference	38
I. Přílohy.....	43
I.I. Znovuspuštění programu.....	43
I.II. Odesílání sms	44
I.III. Čtení senzorů se zápisem do databáze	45

Seznam obrázků

Obrázek 1 Arduino [10].....	7
Obrázek 2 Raspberry Pi [16]	8
Obrázek 3 Intel Edison [17].....	9
Obrázek 4 Periferie Tecomat Foxtrot [20]	10
Obrázek 5 Bezdrátový přijímač PLC [25]	13
Obrázek 6 Ovládací relé [27]	14
Obrázek 7 Bezdrátový fyzický spínač [30]	15
Obrázek 8 Active Buzzer [32]	16
Obrázek 9 Nasazení bezdrátové hlavice ventilu [35]	17
Obrázek 10 Chytrá hlavice [36].....	17
Obrázek 11 Srovnání modelů Raspberry [43].....	20
Obrázek 12 Rádiový modul [60].....	21
Obrázek 13 Teplotní senzor PrivateEyePi [61]	22
Obrázek 14 Senzor Wireless Things [62]	22
Obrázek 15 Princip magnetického senzoru [49]	22
Obrázek 16 Magnetický senzor [50]	23
Obrázek 17 Senzor zaplavení [51]	23
Obrázek 18 Detail místa pro baterii [52].....	24
Obrázek 19 Detail kompletního senzoru [53].....	24
Obrázek 20 Schéma Language of Things [54]	25
Obrázek 21 Datové pakety	26
Obrázek 22 Vlastní sestavení systému	33
Obrázek 23 Instalovaný magnetický senzor.....	33
Obrázek 24 Instalovaný senzor záplavy	34
Obrázek 25 Instalované teplotní čidlo	34

Seznam tabulek

Tabulka 1 Srovnání řídicích modulů.....	12
Tabulka 2 Ukázka paketů	26

Seznam zkratek

IoT	Internet of Things
NB-IoT	NarrowBand IoT
SD	Secure Digital
RAM	Random-Access Memory
Wi-Fi	Wireless Fidelity
GPS	Global Positioning System
HDMI	High-Definition Multi-media Interface
USB	Universal Serial Bus
PLC	Programovateľný automat
CIB	Common Installation Bus
PC	Personal Computer
GPIO	General Purpose Input/Output
LTE	Long Term Evolution
UART	Universal Synchronous / Asynchronous Receiver and Transmitter
AES	Advanced Encryption Standard
SQL	Structured Query Language
HDD	Hard Disk Drive
OTP	One-Time Programmable
SoC	System on a chip
API	Application Programming Interface

1. Úvod

1.1. SLOVO ÚVODEM

Éra internetu věcí pomalu, ale jistě přichází [1]. Internet věcí (IoT) je označení pro propojení zařízení, které pro svůj provoz nevyžaduje síťové připojení, do internetové sítě. Obyčejná lednička se pak, po připojení senzorů a zapojení do vyššího celku připojeného na internetovou síť, stává autonomně fungujícím zařízením, které pozná, které potraviny docházejí, a je samo schopné je objednat i s dopravou až ke dveřím.

Internet věcí se tak stal trendem, o kterém se mluví již dlouhou dobu a výrobci stále slibují zcela autonomní domácnosti (jeden z prvních pokusů o postavení takového domu je Push Button Manor z padesátých let [2]). Z postupu tohoto trendu se ovšem zdá, že změny nepřijdou nárazově ihned s dostupnou technologií, ale spíše se do našich domácností plíží postupně krůček za krůčkem, jak se tyto technologie stávají cenově dostupnější a rozšířenější. [3]

Základem chytré domácnosti jsou senzory připojené k vyhodnocovacímu zařízení a akční členy, vykonávající nějakou činnost na povel řídicího zařízení. Systémy chytré domácnosti se dělí například na náročné, na míru dělané systémy, a zpravidla na stavebnicové systémy. Na míru dělané systémy mají komplexním způsobem vyřešit snímání a řízení celého domu. Nevýhodou tohoto systému je samozřejmě poměrně vysoká cena, ale také to, že se bude nejspíše jednat o silně uzavřený systém, a nikoliv o otevřený a rozšiřitelný produkt, který by se do budoucna dále rozšiřoval o produkty a moduly dalších výrobců. Stavebnicové produkty jsou postavené na bezdrátových senzorech a akčních členech, které umí proměnit i standardní byt či dům v chytrou verzi. Nedisponují komplexitou výše zmíněných systémů ani většinou neumožňují zabudování zařízení přímo do domu, ale umísťují se na nebo do vnitřního vybavení. Výhodou je relativně rychlá realizace bez zasahování do vedení nebo stavebních prvků.

1.2. VOLBA TÉMATU

Volbu tématu velice ovlivnila vlastní potřeba po určitých prvcích chytré domácnosti, která se nedala uspokojit produkty dostupnými na trhu. Většina je poměrně drahá a možnosti individuální rozšiřitelnosti nulové. A přitom možnosti a výhody, které z takové domácnosti plynou, jsou značné a omezené prakticky pouze vlastním nápadem, jak je využít. Komu by se nehodila možnost vypnout přívod vody ve chvíli, kdy detektor záplavy zahlásí únik vody? Možnost přivolat hasiče, když z domu odjede, a je detekován kouř? Vzbudit spící lidi při úniku plynu? Upozornit na otevřená vrata garáže? Zavřít ventil topení při otevřených oknech? ... Toto a mnoho dalšího lze dosáhnout s jednoduchými senzory, vyhodnocovacím počítačem a připojením k internetu.

Ovšem ruku v ruce s nápady, co všechno lze takto řešit, přišla i ekonomická otázka: Když to potřebuji já, tak nejspíš i někdo jiný. Zrodil se tedy projekt, který by mohl lidem pomoci si takový systém postavit svépomocí. Toto jsem si ovšem neuvědomil jen já, ale i další lidé a společnosti. Od roku 2016 se v České republice začalo testovat NB-IoT¹, a tak se i korporátní operátor připravuje na vytvoření mobilní sítě pro IoT [4].

¹ Více v kapitole Senzory mobilní sítě

Nejen korporátní hráči, ale i spousta malých výrobců a podniků se snaží proniknout na tento čerstvý trh, s kterým je, z mého pohledu, zřejmá analogie s mladým internetovým trhem v devadesátých letech. Řekl bych, že nyní nastala ideální doba, kdy trh ještě neovládají dominantní hráči, nikdo nemá zřejmou technologickou převahu, doba přeje otevřeným řešením a objevují se služby, které se dají na IoT navázat (například dovážka potravin až do domu a podobně).

1.3. CÍL PRÁCE

Hlavním cílem této práce je poskytnout náhled do možností technologií IoT, a to zejména do oblasti senzorů a jejich vyhodnocování. Práce má čtenáři poskytnout ucelený přehled nejvýznamnějších řešení v oblasti s popisem výhod a nevýhod jednotlivých produktů. Práce se věnuje především takovým řešením, které je možné uživatelsky realizovat a nezabývá se tak pokročilými komplexními systémy.

Dílčím cílem je pak návrh vlastního řešení pomocí vybraných produktů z oblasti IoT, a to zejména zprovoznění senzorů a jejich připojení k řídicí jednotce. Na řídicí jednotce spuštění servere a databáze a tím umožnění čtení dat pomocí webové stránky.

2. Teoretická část

2.1. HARDWARE

2.1.1. Řídící moduly

Procesní jednotka je mozkiem celého systému automatizované domácnosti. Zpracovává data pomocí předem připraveného programu, vyhodnocuje data přijatá ze senzorů přes přijímač a dále může přes různé periferie, na základě programu nebo příkazu, vykonávat činnost. Procesní jednotka také, po připojení vhodných periférií jako klávesnice, obrazovka a podobně, slouží jako uživatelský vstup do systému. U některých druhů procesních jednotek jsou možná další využití a jednotka pak může sloužit jako osobní počítač nebo multimediální centrum.

Jednodeskové počítače

Jednodeskové počítače jsou počítače na jedné, většinou poměrně malé desce plošných spojů. Na této desce jsou umístěny všechny potřebné komponenty od procesoru, přes RAM a flash paměť až po několik vstupních a výstupních rozhraní. Jednodeskové počítače většinou nedisponují rozhraním pro pevný disk a jako paměť jim slouží SD karta. Toto řešení paměti přináší výhodu kompaktních rozměrů, ale přináší problémy spojené s provozem SD karty. SD karty mají omezenou životnost zápisů a tento problém je pak třeba řešit (například softwarově, využitím části RAM, nebo hardwarově, připojením pevného disku). Tento problém podrobněji rozeberu v druhé části práce.

Tyto počítače většinou běží pod operačním systémem Linux, což je pro vlastní řešení chytré domácnosti podstatná výhoda, zejména kvůli tomu, že se jedná o otevřený a svobodný systém se značnou podporou komunity.

Další výhodou je relativně malá spotřeba energie a příznivá cena. Mezi nevýhody pak patří nižší výpočetní výkon, citlivost na přehřívání (neobsahují aktivní chlazení) a nevhodnost jako vývojový prostředek. Programy je efektivnější vytvářet například na výkonnějších stolních počítačích a do jednodeskového počítače pouze nahrát kód.

Jednodeskových počítačů je na trhu velmi mnoho a já představím tři nejvýznamnější: Arduino, Raspberry Pi a Intel Edison.

Orientační cena jednodeskových počítačů jsou stovky korun. Cena celkového řešení pak v řádu nižších desetitisíců. [5] [6]

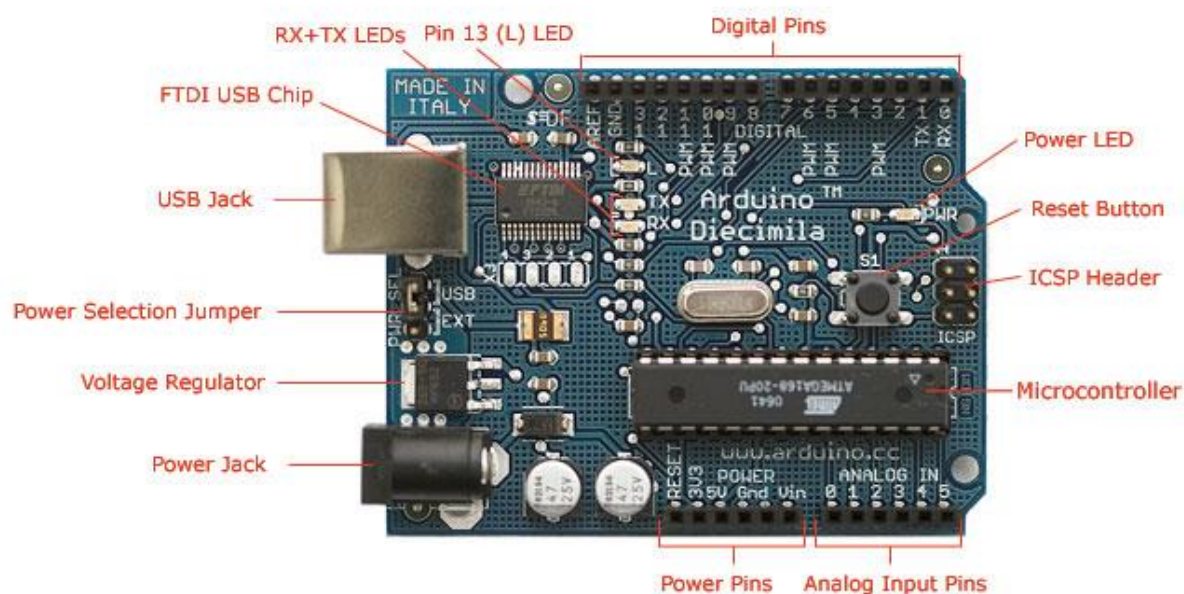
Arduino

Arduino není počítač v pravém slova smyslu, ale jedná se o mikrokontroler. Jako takový se hodí spíše pro jednodušší úlohy. Hodí se výborně pro úlohy typu jednoduššího řízení a regulace. Pokud by cílem bylo pouze přečíst data ze senzorů a někam je vypsát, byl by to zřejmě nejlepší kandidát. Pro hovoří úžasná jednoduchost a spolehlivost, podpora velké komunity. K Arduino se dá také připojit velké množství periferních zařízení (takzvané shiely), jako je například rozšíření o ethernet, Wi-Fi, GPS nebo rozhraní pro ovládání motorů.

Jak bylo zmíněno, jedná se mikrokontroler, a není tak zamýšlen jako plnohodnotný počítač. Kód je vyvíjen na jiném zařízení a je pouze nahrán do paměti Arduino (většinou ve formě nekonečné smyčky). Nelze ho tedy použít jako náhradu stolního

počítače ani jako domácí multimediální centrum. Složitější aplikace by pak také mohly dosáhnout limitů tohoto přístroje.

Programy pro Arduino jsou psány pomocí jazyka Arduino, který je založen na jazyce Wiring, jež vychází z C/C++. Arduino je od základu vyvíjen jako učební pomůcka, a tak disponuje jednoduchým a čistým programovacím prostředím, které je multiplatformní, a je tak možno vyvíjet nejen ve Windows, ale i v Linuxu nebo Mackintoshi OSX. [7] [8] [9]



Photograph by SparkFun Electronics. Used under the Creative Commons Attribution Share-Alike 3.0 license.

Obrázek 1 Arduino [10]

Raspberry Pi

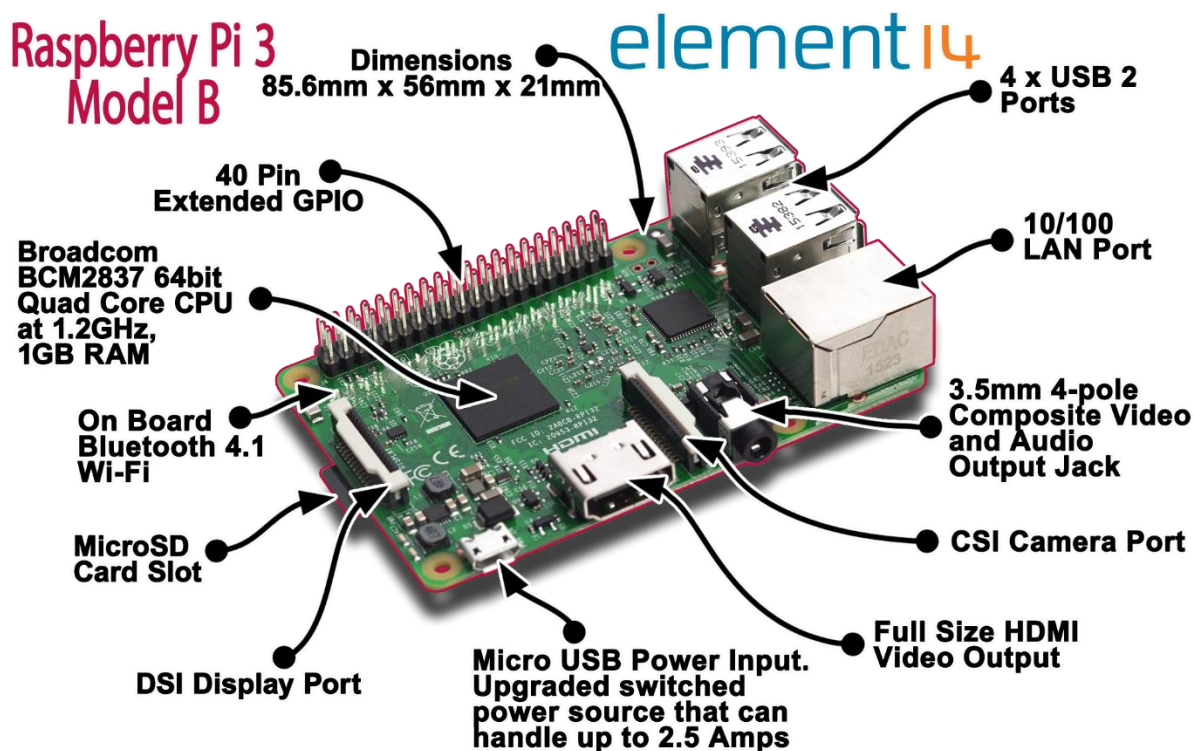
Raspberry je již plnohodnotným počítačem. Je tak na něj možné nainstalovat různé operační systémy, ovšem nejčastěji běží pod taktovkou Raspbianu, odvozeniny Linuxu. [11] Další výhodou je možnost vybrání vhodného zařízení z celé řady, od nejmenší a nejlevnější varianty Zero až po nejvýkonnější a nejdražší variantu 3. Výkony tohoto jednodeskového počítače se již, zejména ve variantě 3, blíží slabším stolním počítačům a je tak možno jej využít jako plnohodnotnou náhradu za stolní počítač nebo jako multimediální centrum. K tomu je také Raspberry vybavené HDMI výstupem a ve variantě 3 i Wi-Fi modulem.

Výhodou tohoto řešení je nejlepší komunitní podpora, příznivá cena a schopnost zvládnout prakticky jakoukoliv očekávanou aplikaci. Na internetu je obrovské množství programů, napsaných pro Raspberry Pi, a značná část je dostupná zdarma. Existuje tak nesčetné množství předpřipravených řešení, která lze integrovat do stávajícího systému.

Nevýhodou pak může být například to, že v případě využití jako serveru je nutno paměťový disk připojit pomocí poněkud pomalého USB 2.0, a sdílení souborů tak bude nejspíše limitováno propustností USB můstku. V případě náročnějšího využití je také možný problém s přehříváním, protože počítač nedisponuje chladičem. Chladič

lze samozřejmě přidat, ale přijdeme o kompaktní rozměry a v případě použití aktivního chladiče stoupne spotřeba energie.

Protože se jedná o plnohodnotný počítač, je možné pro psaní programů využít prakticky jakéhokoliv jazyka, ale komunitně nejpoužívanějším, a také implicitně podporovaným, je jazyk Python (v případě nejpoužívanějšího Raspbianu). [12] [13] [14] [15]

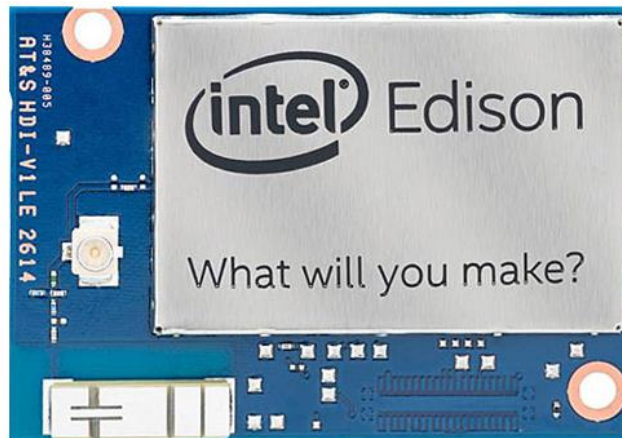


Obrázek 2 Raspberry Pi [16]

Intel Edison

Výjimečnost tohoto řešení spočívá v jeho velikosti, která odpovídá zhruba SD kartě, a je tak předurčeno pro přenosná zařízení. Lze ho snadno přestavět, například do formy tabletu, a mít přenosný ovládací a řídicí prvek pro celý systém. Nevýhodou pak je to, že kvůli minimalismu nedisponuje tolika výstupními rozhraními, a takovýto systém bude nutno doplnit o další prvek. Zde se nabízí perfektní spolupráce jednoduchého mikrokontroleru Arduina, který se bude starat o příjem zpráv z periférií a ukládání na server a Edisonu, který bude díky Wi-Fi připojení spojen se systémem bezdrátově a bude mít na starost všechny nastavbové činnosti.

Je však třeba mít na paměti, že výkon tohoto „drobečka“ sice dostačuje na řízení chytré domácnosti, ale je bohužel nedostatečný na multimediální centrum.



Obrázek 3 Intel Edison [17]

Programovatelné automaty

Programovatelné automaty (PLC) jsou rozměrově poměrně malé počítače, určené k průmyslovému využití a primárně slouží k automatizaci procesů v reálném čase. K výhodám určitě patří značná spolehlivost a připravenost periférií k napojení na další prvky. PLC jsou většinou předpřipraveny pro digitální komunikaci pomocí vstupů a výstupů a existuje velké množství periférií připravených prakticky pro všechny automatizační potřeby. Systém postavený na PLC by měl být v obecné rovině velice spolehlivý a robustní s dlouhou životností.

Orientační cena automatu je řádově 10 000 Kč a cena celkového řešení pak v řádu desítek až stovek tisíc korun.

PLC systémy různých výrobců nabízejí různé funkčnosti v různých cenových hladinách, ale v obecné rovině se příliš neliší a pro představu o možnostech systému založeném na PLC postačí jeden příklad za všechny:

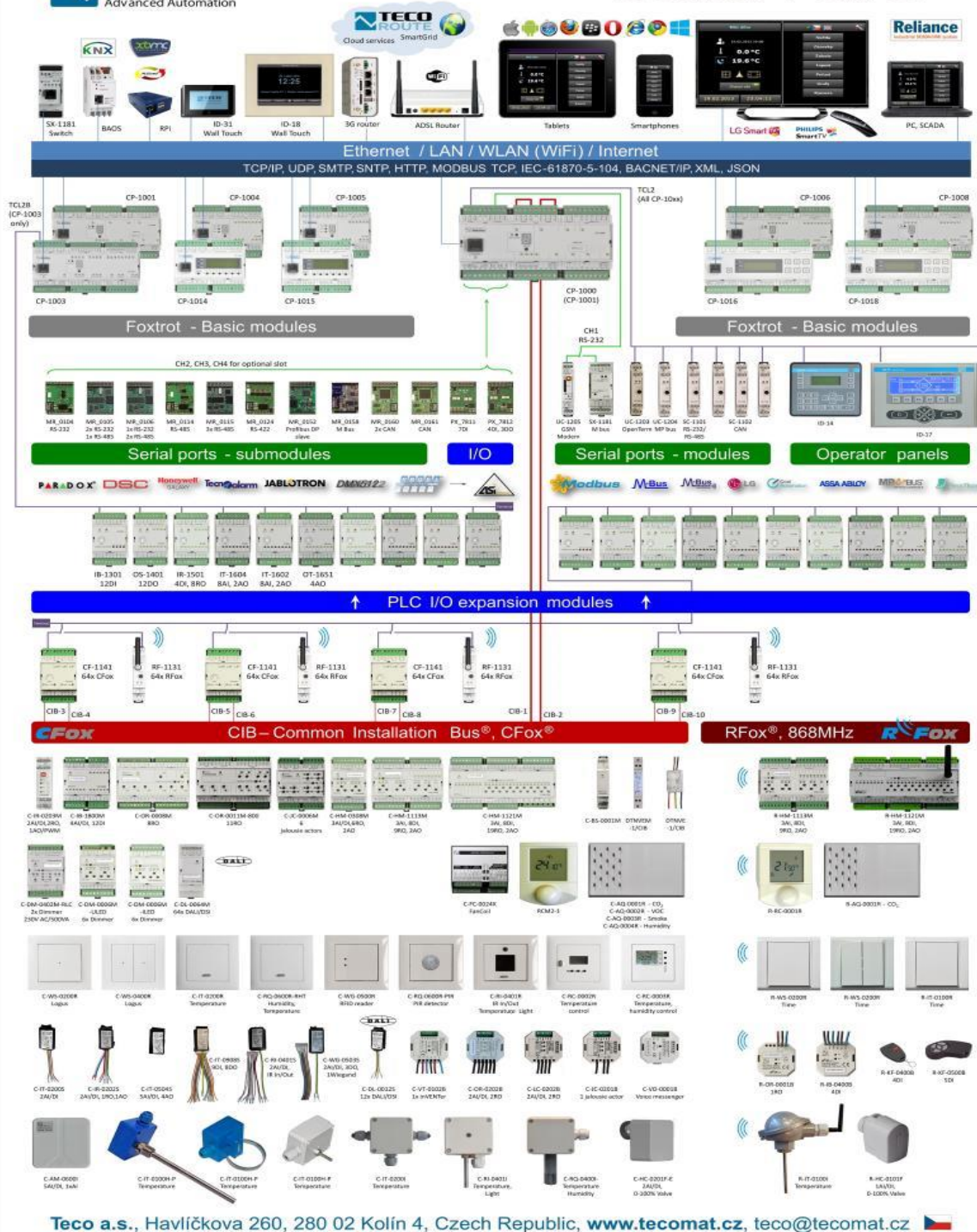
Tecomat Foxtrot

Tento poměrně malý modulární řídicí a regulační systém od společnosti Teco a.s. je zaměřen právě na chytré elektroinstalace a inteligentní budovy. Tento systém je díky jeho ceně a průmyslovým možnostem předurčen pro velice složité a komplexní instalace, které je schopen spolehlivě pokrýt.

Systém je díky sběrnici CIB (Common Installation Bus) možné rozšířit o libovolný elektroinstalační prvek CFox. Na univerzalitě pak přidává duplexní bezdrátový systém Fox, který umožňuje zapojení bezdrátových prvků.

Programování probíhá v prostředí Mosaic respektující normu IEC61131-3. a je tedy možno využít až 4 druhy jazyků. Ovšem programování PLC, a zvláště složitějších systémů, vyžaduje více zkušeností a nenabízí tak intuitivní prostředí jako dříve představené procesní jednotky.

Velkou výhodou představuje celá řada předpřipravených periférií umožňujících proměnit budovu ve skutečně inteligentní. [18] [19]



Obrázek 4 Periferie Tecomat Foxtrot [20]

Výhodou a zároveň nevýhodou je průmyslový původ i podstata tohoto systému. Systém je připraven a schopen ovládat nejrůznější prvky od světel až ke klimatizaci. Může být poměrně jednoduše zapojen do bezpečnostních prvků budov, a dokonce být využit k vizualizaci a mapování procesů probíhajících v budově. Ovšem cenou za tyto možnosti, za spolehlivost a rozšiřitelnost, je cena, která je násobně vyšší než u ostatních možností. Jedná se také o velice specificky zaměřený produkt, a tak jej není

možno tak obecně využívat. Pro vlastní cloud, multimediální centrum i náhradu PC by pak bylo nutno najít jiné řešení.

Osobní počítač

Na rozdíl od specificky zaměřeného PLC je osobní počítač nejuniverzálnější řešení. Je schopen všeho, čeho jsou schopny jednodeskové počítače a do jisté míry je schopen i většiny činností PLC. Daní za univerzálnost je pak to, že nevyniká v ničem. Spotřeba energie bude vyšší, je ho třeba rozšířit o doplňkové moduly a softwarové vybavení, aby byl schopen komunikovat s dalšími periferiemi, a protože není specificky zaměřen, nenabízí z hlediska IoT ani zdaleka tak velkou komunitní základnu s mnoha vývojáři a hotovými projekty na téma IoT jako jednodeskové počítače.

Na druhou stranu může velice jednoduše sloužit jako domácí server, cloud, i multimediální centrum, a pochopitelně zastane roli domácího počítače.

Orientační cena počítače jsou jednotky tisíc korun a cena celkového řešení pak v řádu desetitisíců korun.

Předpřipravené stavebnicové systémy

Předpřipravené stavebnicové systémy umožňují nejrychlejší realizování domácí automatizace. V základu většinou obsahují centrální jednotku, několik magnetických a teplotních čidel, kamerové vybavení, čidla pohybu a podobně. Obsahují i software připravený pro takové aplikace.

Nicméně jejich programové vybavení a druhy periférií jsou dané a většinou je není možné žádným způsobem rozšířit či upravit.

Orientační cenou řídicí jednotky stavebnicového systému jsou vyšší tisíce korun a cena celkového řešení v řádu desetitisíců.

Podobné systémy dnes již nabízí velké množství výrobců a velmi se liší jak cenou, tak schopnostmi. Za všechny představím produkt Fibaro.

Fibaro

Mozkem tohoto systému je centrální jednotka zajišťující komunikaci s periferiemi, a která dokonce slouží jako multimediální centrum. Po připojení do sítě umožní ovládání přes počítač nebo telefon. K centrální jednotce se pak bezdrátově připojují další prvky jako například detektor kouře, záplavy nebo světla, bezdrátová zásuvka, bezdrátový ventil topení, detektor pohybu nebo magnetický senzor.

Fibaro komunikuje pomocí bezdrátového otevřeného protokolu Z-Wave, který byl vyvinut právě pro domácí automatizaci. Z-Wave vytváří topologii sítě Mesh², a tak je každý článek schopen sledovat a řídit práci jiných modulů, což ovšem znamená, že všechny moduly musí být zároveň vysílači a zároveň přijímači. Díky otevřenosti tohoto standardu by mělo jít připojit jakékoliv certifikované zařízení a systém tak rozšířit. [21]

² http://automa.cz/cz/casopis-clanky/bezdratove-site-typu-mesh-2005_12_30826_1141/

Tabulka 1 Srovnání řídicích modulů

Řídicí modul	Orientační cena ³	Multimedia	Vlastní programy	Použitelné jako server
Arduino	700	Ne	Omezeně	Ne
Raspberry	1200	Ano	Ano	Ano
Intel Edison	1250	Omezeně	Ano	Ne
Tecomat Foxtrot	16000	Ne	V rámci PLC	Omezeně
PC	10000	Ano	Ano	Ano
Fibaro	16000	Ano	Omezeně	Ne

2.1.2. Senzory

Dle [22] je senzor prvkem měřicího přístroje, na který měřená veličina přímo působí. V užším slova smyslu znamená technickou součástku, která měří veličinu a převádí ji na elektrický signál. Elektrický signál je pak vhodný k přenášení do dalších zařízení.

Senzory jsou pro systém domácí automatizace klíčovým prvkem. Poskytují procesní jednotce informace o aktuálním stavu systému a tím poskytují řídicímu algoritmu informace potřebné k funkci programu. Senzory mohou být drátové nebo bezdrátové, pro systém domácí automatizace jsou mnohem vhodnější bezdrátové senzory. Bezdrátové senzory je možné instalovat bez nutnosti zásahů do elektroinstalace domu. Nevýhodou těchto senzorů je pak vyšší cena, způsobená navýšením o vysílač, a také provoz na baterii, kterou je nutno vyměňovat.

Monnit

Firma Monnit se zaměřuje na produkci bezdrátových monitorovacích řešení s ohledem na příznivou cenu. Senzory Monnit jsou postaveny kolem Wireless gateway (brána), která tvoří síťový uzel pro shromažďování informací z bezdrátových senzorů a posílá je do softwaru procesní jednotky.

Senzorů je dostupné velké množství od měření teploty a vlhkosti, přes detekci otevřených dveří a oken až po únik plynu či detekci obsazeného sedadla. Cena těchto senzorů je ve střední třídě a zpracování, společně s technickou podporou jsou na dobré úrovni. Nicméně se jedná o poměrně uzavřený systém a je tak nutností zakoupit i poměrně hodně podpůrných produktů. Zejména již zmíněnou bránu, pak také software, jehož cena se liší dle využití a je možné jej také za poplatek odemknout a vysílat naměřené hodnoty do dalších programů.

Cílovou skupinou těchto senzorů se snadno mohou stát instituce, které ocení spolehlivost a robustnost systému a využijí pokročilé softwarové vybavení. [23]

Privateeyepi

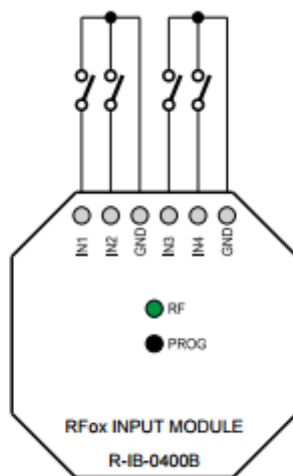
Firma Privateeyepi je zaměřena na velice levná zařízení určená právě pro jednodeskové počítače a domácí využití. Díky tomu, že je firma zaměřena na jednodeskové počítače, nabízí velké množství návodů, jak se senzory pracovat, a dokonce spravuje vlastní online systém domácího monitorování a zabezpečení. Systém je bohužel uzavřený a poměrně omezený, nicméně je schopen plně zabezpečit jednoduché aplikace. Oproti firmě Monnit nejsou senzory na tak

³ Velice záleží na použitém modelu

profesionální úrovni (je nutné je před spuštěním složit a spájet) a jejich vzhled je také poněkud strohý. Nicméně cena je přibližně poloviční. Rozsah produktů také není velký. Dostupné jsou senzory teploty, vlhkosti, pohybu a magnetický detektor. Senzory se bezdrátově připojují k jednodeskovému počítači pomocí rádiového modulu, který se připojí na GPIO piny. Software zajišťující provoz je volný a otevřený a je tak vhodný pro individuální aplikaci, při níž je možné ho rozšířit o vlastní nápady a kód. Využívaný jazyk je Python. [24]

Senzory pro PLC

PLC jsou schopny pracovat téměř s jakýmkoliv drátovým průmyslovým senzorem a pro práci s bezdrátovými zařízeními slouží takzvané vstupní a výstupní moduly. Do těchto modulů se připojí libovolná čidla, jejichž signál je pak přenášen rádiovým signálem do PLC, které se již postarají o vyhodnocení dle programu.



Obrázek 5 Bezdrátový přijímač PLC [25]

Toto řešení je na jednu stranu výhodné, protože drátových senzorů je nesmírné množství v nejrůznějších variantách, cenových hladinách i schopnostech a dává tak systému postaveném na PLC nesmírnou variabilitu, která se také od průmyslového řešení očekává. Nevýhodou pak je, že se z počátku jedná o nehotovu elektrickou součástku a je vhodné ji umístit do krabičky a vyřešit umístění na požadované místo.

Senzory mobilní sítě

Tyto senzory jsou hudbou blízké budoucnosti. Jak je uvedeno v úvodu, tak se již v České republice připravuje spuštění sítě NB-IoT vhodné právě pro tyto senzory. NB-IoT síť je druh mobilní sítě uzpůsobené pro připojování senzorů IoT. Její hlavní výhodou je, že vznikne úpravou softwaru LTE sítě (vyhradí se část pásma). Důvod této úpravy je zejména v energetické náročnosti. Zařízení v klasické síti se často hlásí i když „nic nepotřebuje“. Také každému datovému přenosu předchází komunikace otevírání kanálu a podobně se i kanál uzavírá. U jednoduchého IoT senzoru je tak vysláno mnohem víc dat, než je třeba, což znamená větší nároky jak na síť, tak zejména na spotřebu elektrické energie v baterii senzoru. Výhodou této technologie je to, že není vázána na dosah rádiového signálu, ale senzory odesílají informace přímo do mobilní sítě. K procesní jednotce se pak tyto informace dostanou přes internetové připojení. Nevýhodou je to, že každý senzor má vlastní připojení do sítě, což systém prodražuje kvůli pravidelným poplatkům operátorovi.

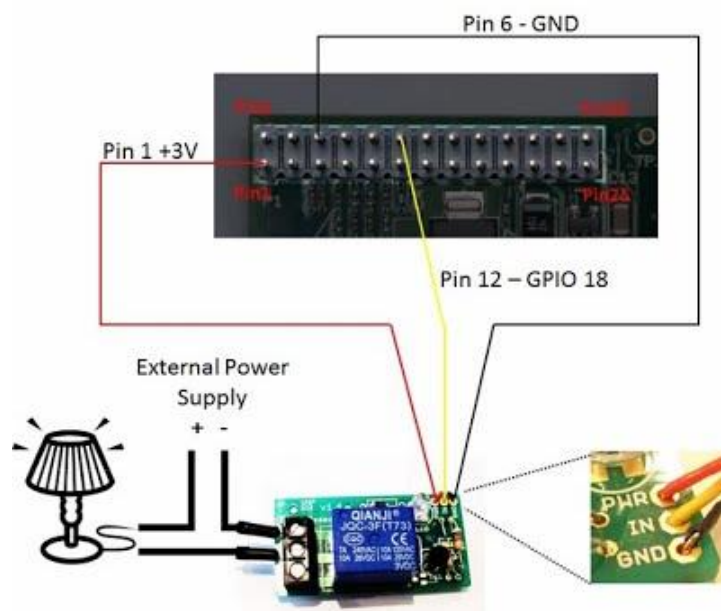
Tyto senzory ovšem nebyly ještě oficiálně představeny, a tak není možné přinést bližší informace.

2.1.3. Akční členy

Akční členy jsou dle [26] členy ovlivňující akční veličinu. Mají tedy za úkol vykonat určitou činnost na pokyn z řídicí jednotky za účelem změny stávajícího stavu. Může se jednat například o změnu teploty, průtoku tekutiny, nebo třeba přepnutí snímače. V této části uvedu několik nejzajímavějších akčních členů vhodných pro IoT.

Vypínač

Jeden ze základních akčních členů umožňuje vypínat a zapínat elektrický přístroj sepnutím nebo rozepnutím elektrického okruhu. Hodí se zejména na rozsvěcení a zhasínání světel na dálku, ale je možné ho také využít například na dálkové spuštění klimatizace a podobně. Zde bych rád upozornil všechny čtenáře, aby v případě, že nedisponují potřebnou elektrotechnickou kvalifikací, upřednostnili řízení nízkonapěťových obvodů z důvodu vlastní bezpečnosti.



Obrázek 6 Ovládací relé [27]

Ze schématu výše vyplývá, že se relé připojuje přímo k GPIO pinům procesní jednotky. Toto řešení ovšem není příliš efektivní, protože vyžaduje natažení kabeláže ke každému řízenému obvodu. Nabízí se několik možností, jak tento problém vyřešit. V případě, že bude třeba vytvořit ještě nějaký další návazný systém, jako třeba stahování rolet a podobně, doporučuji použít nějaký mikrokontroler jako třeba Arduino. Pak může mít každá místnost jedno Arduino, na které budou napojeny elektrické obvody světel rolet a podobně. Celé to bude jako systém bezdrátově připojeno k nějakému výkonnějšímu výpočetnímu jádru, které se postará o řízení celého systému. V případě, že bude cílem pouze řídit světelné okruhy, je možné využít nějakého předpřipraveného bezdrátového zařízení, které se postará o komunikaci, např. Wi-Fi Remote Switch Controller.

Další zajímavou možností je opravdový fyzický vypínač na dálkové ovládní. Jedná se vlastně o dálkově ovládané zařízení, schopné fyzicky stisknout většinu vypínačů. Touto cestou je tak možné dálkově spínat běžná zařízení, u kterých by nebyla možná nebo efektivní přestavba. Hodí se tak například na spuštění PC, kávovaru nebo třeba větráku. [28] [29]



Obrázek 7 Bezdrátový fyzický spínač [30]

Siréna

Další užitečnou součástí je siréna. Sirény mohou sloužit jako poplach zabezpečovacího systému a tím upozornit okolí, jako například sousedy, na nebezpečí. Nicméně tento druh systému přesahuje rozsah této práce, a tak spíše uvedu vhodnější a velice malou součástku: Active Buzzer. Tato malá siréna, dokáže produkovat poměrně hlasitý zvuk, který ovšem v žádném případě nebude rušit sousedy, a je tak možné ji využít jako upozornění na nestandardní situaci, jako jsou otevřené vchodové dveře, garážová vrata, únik plynu a podobně. [31]



Obrázek 8 Active Buzzer [32]

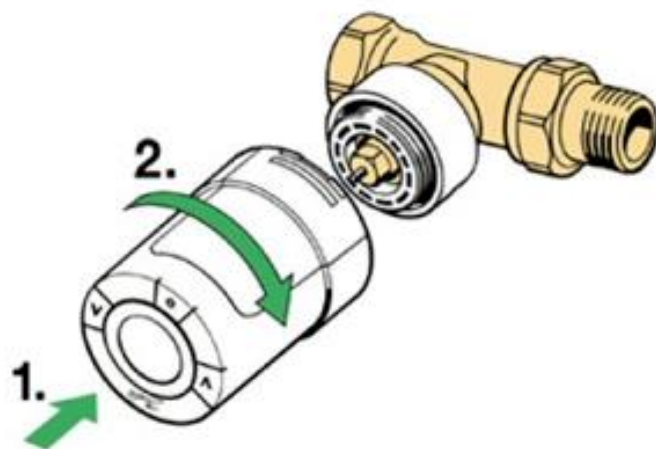
Ovládání rolet

Ovládání elektrických rolet se velice podobá bezdrátovému spínači, ale je přece jenom o něco komplexnější. Základ je podobný, ale je nutné ho rozšířit o koncové snímače a zajistit spolehlivý chod, aby se zabránilo poškození motoru rolet. Některé z těchto modulů umí i například naklápět žaluzie v případě jejich využití nebo měřit spotřebu energie a podobně. [33]

Ovládání ventilu topení

V podobě automatického ovládání topného systému přichází možnost díky IoT kontinuálně šetřit spotřebu energií. Bezdrátová hlavice se nasadí jednoduše na ventil topení a připojí se k řídicí jednotce. Hlavici je pak možno ovládat jak dálkově, tak automaticky dle programu. Díky tomu je možné automatizovat vytápění tak, aby se teplota večer v ložnici začala snižovat, ale v koupelně zvyšovat. V noci pak teplota klesla na teplotu optimální pro spánek, k ránu se začalo topit první v koupelně a na základě plánu i v ostatních místnostech – například ve všední den, kdy všichni odejdou do práce se ráno ohřeje pouze koupelna a kuchyň. Ostatní místnosti až odpoledne a o víkendu by se automaticky začalo topit všude. Obdobně by to mohlo fungovat při odjezdu na dovolenou – v domě by se udržovala pouze nízká teplota a topit by se začalo až těsně před příjezdem. Díky tepelným sensorům je navíc možné změřit tepelnou dynamiku domácnosti a na základě venkovní teploty vypočítat dobu potřebnou k vytopení na požadovanou teplotu.

Hlavice u většiny výrobců zobrazují aktuální teplotu a umí se zavřít při detekci otevřeného okna. Další schopností je ochrana proti zamrznutí nebo pravidelné otáčení ventilem jako ochrana proti zatuhnutí. [34]



Obrázek 9 Nasazení bezdrátové hlavice ventilu [35]



Obrázek 10 Chytrá hlavice [36]

2.2. SOFTWARE

Software neboli počítačové vybavení, se dělí na systémový a aplikační software. Aplikační software je program, se kterým uživatel pracuje anebo zajišťuje řízení nějakého stroje či externího procesu. V kontextu této práce půjde zejména o vlastní systém nastíněný v druhé části této práce, zatímco zde popíši některé programovací jazyky, ve kterých lze tyto programy napsat. Systémový software je pak program zajišťující chod počítače a jeho komunikaci s okolím. Zde půjde zejména o operační systém a programové vybavení serveru.

2.2.1. Operační systém

Operační systém je základní programové vybavení každé procesní jednotky. Jedná se o velice komplexní software, který má na starosti umožnění plnohodnotné práce s procesní jednotkou. Jedná se zejména o přidělování systémových zdrojů, zajištění ovládacího rozhraní anebo zajištění aplikačního rozhraní.

U některých procesních jednotek je operační systém navrhnut přímo pro daný výrobek (Arduino, stavebnicové systémy, PLC). U dalších jako například u Intel Edisonu nebo samozřejmě u PC je možné operační systém zvolit. Volba je pak zejména mezi Windows⁴ anebo různými variantami Linuxu. Volba by pak měla být založena zejména na možnostech periferních zařízení a jejich kompatibilitou s daným operačním systémem. Celkové srovnání Linuxu a Windows přesahuje rozsah této práce, ale pro IoT systém zde popsany, jsou vhodné oba.

2.2.2. Server

K vlastnímu domácímu serveru vedou dvě základní cesty, které je dokonce možné využívat paralelně.

První z nich je využít hotové programové vybavení od nějakého developera a to implementovat do svého systému. Toto je vhodné řešení například pro osobní cloud, který díky tomu umožní spravovat a editovat dokumenty na dálku a přitom si zachovat dobrou bezpečnost. K těmto programům pak bývá dostupná aplikace pro různé operační systémy a je tak zajištěn jednoduchý přístup z ostatních zařízení (počítač, mobil, tablet ...) Takovým softwarem je například ownCloud, a přesto, že se jedná o open-source program, tak je jeho funkčnost na vysoké úrovni, a funkčností je velice podobný masově užívanému Dropboxu. Také umožňuje práci s dokumenty, jak to známe například z Google Drive. Aplikace je pak dostupná pro Windows, MacOS, Linux, Android a iOS. Další volbou jsou Synthing nebo Seafile, možnosti těchto systémů jsou dost podobné a uživatelům doporučuji vyzkoušet všechny a zvolit si uživatelsky příjemnější program. [37] [38] [39]

Druhou možností je navrhnout a vytvořit vlastní server, což je cesta delší, ale flexibilnější a je možné naprogramovat přesně to, co je třeba. Toto řešení je pak vhodné například pro vlastní webovou stránku s aktuálním stavem řízeného systému. Zde je si třeba zvolit softwarový server – většinou půjde o Apache HTTP Server, na kterém běží celosvětově většina stránek. Nicméně je samozřejmě možné využít i jiné. Rozhodnutí by mělo stát zejména na vývojáři, podle toho, co mu vyhovuje nejvíce.

⁴ Windows u jednodeskových počítačů ovšem není plná desktopová verze, ale Windows 10 IoT. Tento operační systém je vhodný pro domácí automatizaci ale není ho možné použít jako náhrada stolního PC.

Druhým krokem je pak zvolení interpreta. Nejčastější volbou bude PHP, které je specializované na webové stránky a má nativní podporu napříč databázovými systémy, hostingovými službami a podobně – PHP se stalo v podstatě standardem. Další významnou výhodou je nesmírné množství dostupných projektů a kódů, které je možné využít. Drobné nevýhody pak spatřuji v občas nepřehledném programování, například v nekonzistentnosti pojmenování skupin a funkcí, nejednotném pořadí parametrů a podobně. Alternativně lze samozřejmě psát i v jiných jazycích, jako například Python nebo Ruby, které poslední dobou útočí na pozici PHP [40].

2.2.3. Programovací jazyk

Programovací jazyk jako prostředek pro zápis algoritmů ovládající celý systém je velice podstatnou částí systému. Toto odpadá jen při využití stavebnicových systémů, kde je vše naprogramováno a není třeba, a většinou ani možné, do programu zasahovat. Podobná situace je PLC, kde automat podporuje různá programovací rozhraní. Program se sestaví většinou v PC a pouze se nahraje do PLC.

Některé procesní jednotky, především mikrokontrolery, mají vlastní programovací jazyky, většinou vycházející ze známějších jazyků. Například Arduino a jeho Wiring.

U osobního počítače nebo třeba Raspberry jsou pak možnosti větší a developer si může zvolit programovací jazyk dle své libosti. Pro práci s IoT se dle mého názoru hodí nejvíce Python, kvůli velkému množství dostupných a použitelných kódů a projektů. Také výrobci různých periférií pro IoT vydávají návody většinou v Pythonu (např. <http://www.projects.privateeyepi.com/home/install-privateeyepi>). Python je také velice jednoduchý a klade důraz na produktivitu programátora. Z mých vlastních zkušeností plyne, že právě v Pythonu se dají programy vyvinout nejrychleji. Další možností je určitě JAVA, která patří k nejpoužívanějším jazykům [41] a z toho plyne silná komunitní podpora. Výhodou je také multiplatformní přenosnost. JAVA pak bude velmi výhodná například pro programování další aplikace pro mobil a může běžet i na webové stránce. Nakonec uvedu C++, který je z uvedených programů nejkomplexnější a dává prostor pro vytváření velkých a robustních programů. Ovšem vzhledem k předpokládané jednoduchosti systému je tato komplexnost spíše na škodu a povede k pomalejšímu vývoji.

Obecně se dá říct, že všechny programovací jazyky zvládnou to, co je potřeba a záleží spíše na programátorovi, který jazyk je mu bližší. Také je třeba se zamyslet nad tím, jaké budou použity periferní zařízení a podívat se, jestli výrobce doporučuje pro komunikaci nějaký programovací jazyk či, jestli nejsou již hotové programy pro ovládání daného zařízení. [15]

3. Praktická část

Tato část práce popisuje hardwarové a softwarové prostředky, použité pro vytvoření realizované chytré domácnosti za pomoci Raspberry Pi. Dále pak nástin programového řešení.

Mezi hlavní funkce systému patří:

- Hlídání otevřených garážových dveří a jejich světelná, případně zvuková signalizace.
- Vzdálený přístup k vlastnímu serveru a uložišti.
- Detekce různých parametrů domácnosti (teplota, otevřená okna, případně dveře, detekce záplav)

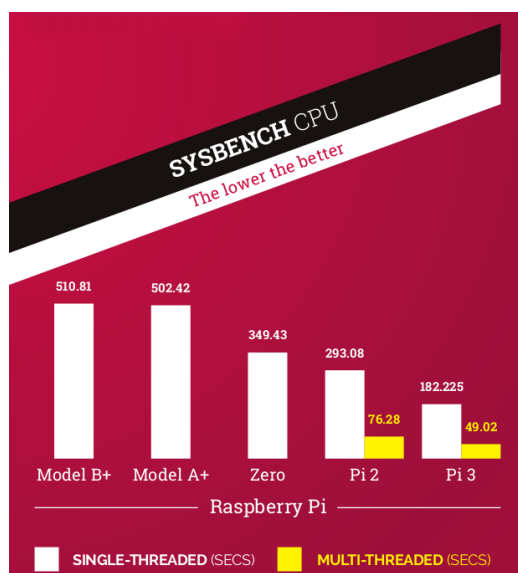
3.1. HARDWARE

3.1.1. Řídící jednotka

Raspberry Pi

Jako základní hardware jsem zvolil Raspberry Pi, pro jeho jednoduchou rozšiřitelnost o další externí prvky a plný Linux. Podle [42], operační systém, na rozdíl od Arduina, který je pouze mikrokontrolerem, umožňuje vyvíjení složitějších programů. Zároveň je Raspberry mnohem rychlejší a podporuje přímé připojení obrazovky, myši, klávesnice a dalších periférií.

Pro využití projektu jsem zvolil nejvyšší model Raspberry Pi 3, zejména z důvodu integrované Wi-Fi a Bluetooth. Jak uvádí [42], Raspberry Pi 3 je také o mnoho rychlejší než předchozí verze. Vzhledem k plánovanému využití také pro procházení webu a domácího multimediálního centra je toto zrychlení velice užitečné.

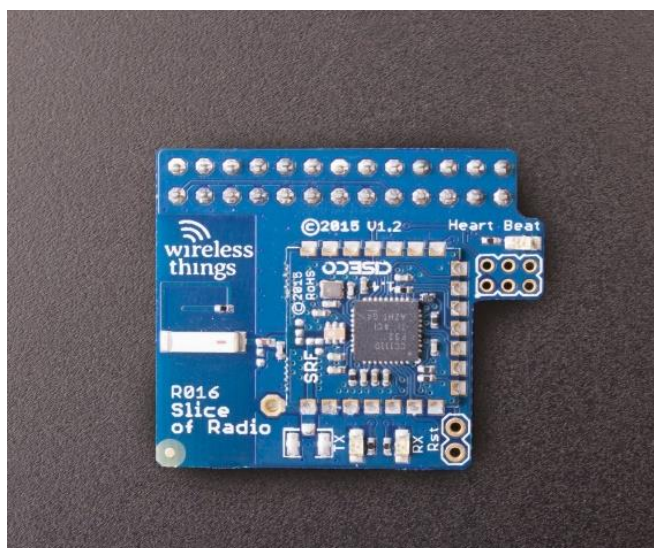


Obrázek 11 Srovnání modelů Raspberry [43]

3.1.2. Senzory

Rádiový přijímač

Jako rádiový přijímač jsem zvolil Slice of Radio [44] od společnosti Ciseco, které umožňuje připojení Raspberry k velkému počtu senzorů od různých výrobců a je dokonce možné si senzor poskládat sám, např. podle Privateeyepi [45]. Díky rádiovému přijímači bude možné vytvořit bezdrátovou domácí síť senzorů, bez nutnosti propojování kabely. Tento rádiový přijímač komunikuje pomocí sériového portu (UART @ 9600bps) a nepotřebuje žádné ovladače. Díky jeho nízkému profilu je možné ho umístit, i s Raspberry Pi, do krabičky. V případě nedostatečného signálu, při nutnosti pokrýt větší prostor je možné jednoduše připájet anténu. Přenos podporuje 128 bitové šifrování a je tak odolný proti sledování.



Obrázek 12 Rádiový modul [60]

Snímač teploty

Jako snímač teploty je použit Wireless Ambient Temperature Sensor SB-CA-White a Wireless Temperature Sensor. Oba teploměry využívají odporový teploměr a rádiový přenos jednoduché šifrované textové zprávy. [46] [47]



Obrázek 13 Teplotní senzor PrivateEyePi [61]

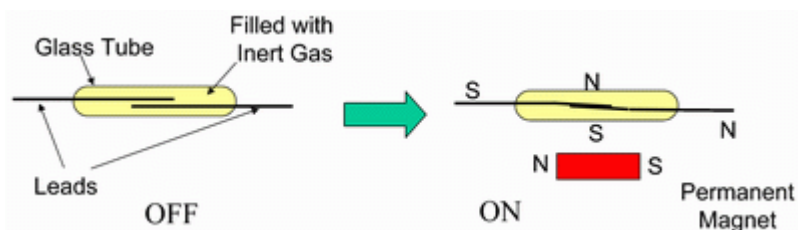


Obrázek 14 Senzor Wireless Things [62]

Oba tyto senzory jsou založeny na rádiovém vysílači, který umožní spárování s rádiovým přijímačem, umístěným na Raspberry Pi a jednoduchém odporovém teploměru NTCLE100E3103JB0. [48]

Snímač polohy

Snímač polohy je ze stejné série jako snímač teploty. Díky jednoduchému magnetickému snímání umožňuje spolehlivou práci. Senzor funguje na principu klasického Reedova senzoru. Jedná se tedy o mechanický spínač ovládaný magnetickým polem.



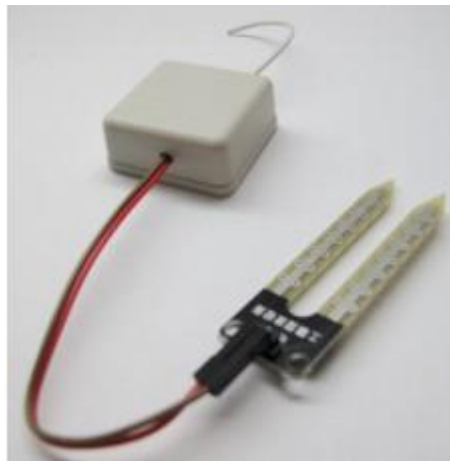
Obrázek 15 Princip magnetického senzoru [49]



Obrázek 16 Magnetický senzor [50]

Snímač zaplavení

Snímač vlhkosti pro detekci úniku kapalin, zejména z pračky a myčky, případně z koupelny a kuchyně obecně. Detektor funguje na principu změny vodivosti mezi dvěma elektrodami.



Obrázek 17 Senzor zaplavení [51]

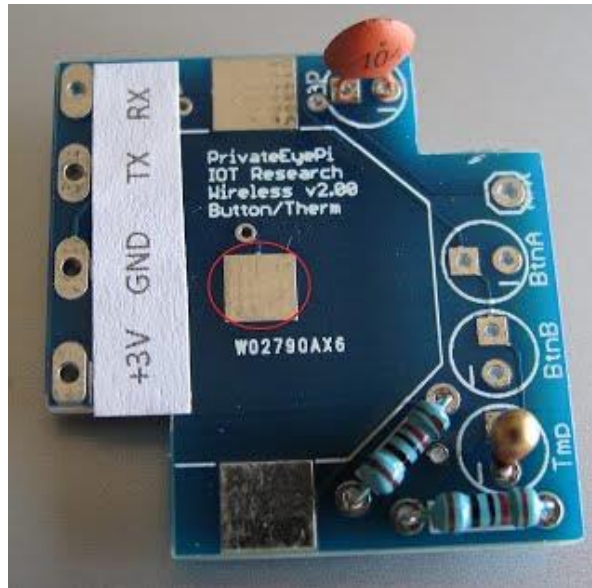
3.1.3. Doplnky

Doplňky systému pro pohodlné ovládání a vylepšenou vizuální stránku.

- Krabička pro Raspberry
- Bezdrátová klávesnice a myš
- SD karta

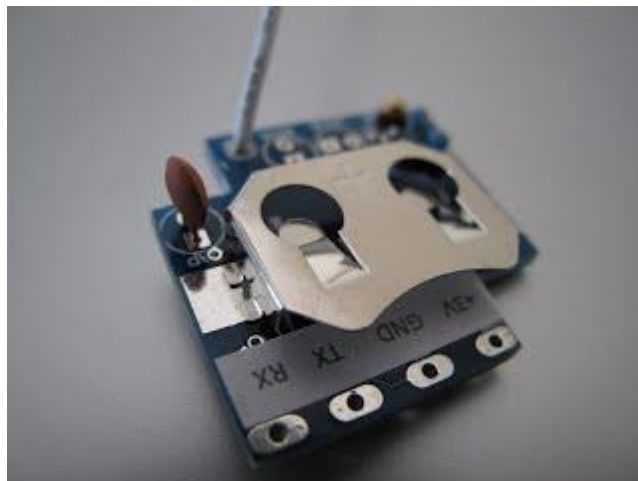
3.1.4. Kompletace senzorů

Senzory se dodávají ve stavebnicovém provedení a je nutné je před použitím zprovoznit. K rádiovému vysílači je nutné připájet dvojici odporů a jeden kondenzátor dle doporučení výrobce, které je přibaleno spolu s vysílačem nebo dle fotografie níže.



Obrázek 18 Detail místa pro baterii [52]

K vysílači lze pak připojit odporový teploměr a další senzor (magnetický, vodivostní). Místo označené kolečkem slouží pro baterii, takže je nutné na vyznačených místech připájet držák na CR2032 baterku. Nakonec je třeba vše uzavřít do pouzdra a senzor umístit na požadované místo. Baterie by měla vydržet odhadem dva roky provozu, a senzor je sám schopen reportovat napětí v baterii, a tím upozornit na pokles pod únosnou mez.



Obrázek 19 Detail kompletního senzoru [53]

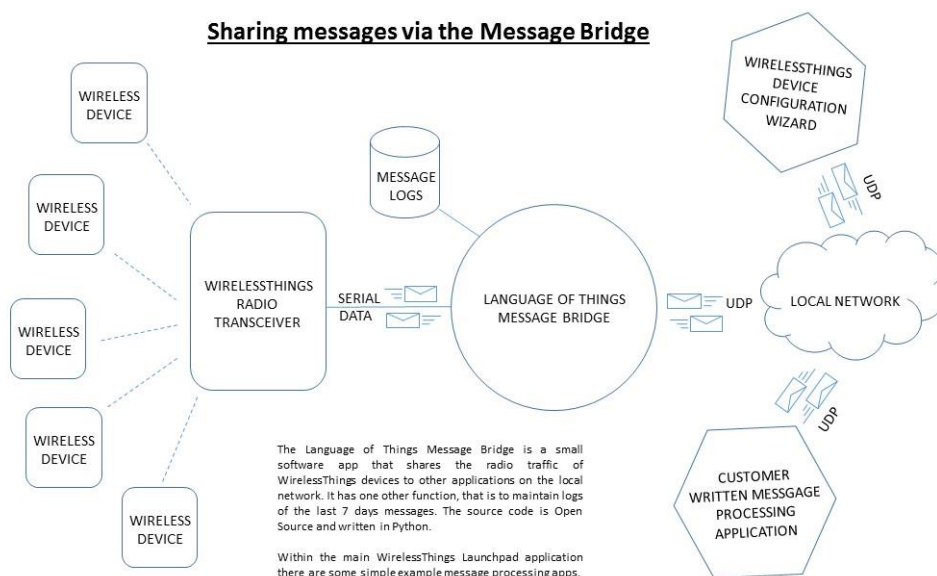
3.2. SOFTWARE

3.2.1. Operační systém

Jako operační systém bude použit operační systém Raspbian s několika úpravami. Tyto úpravy umožní přístup k sériovému portu, přes který komunikují senzory.

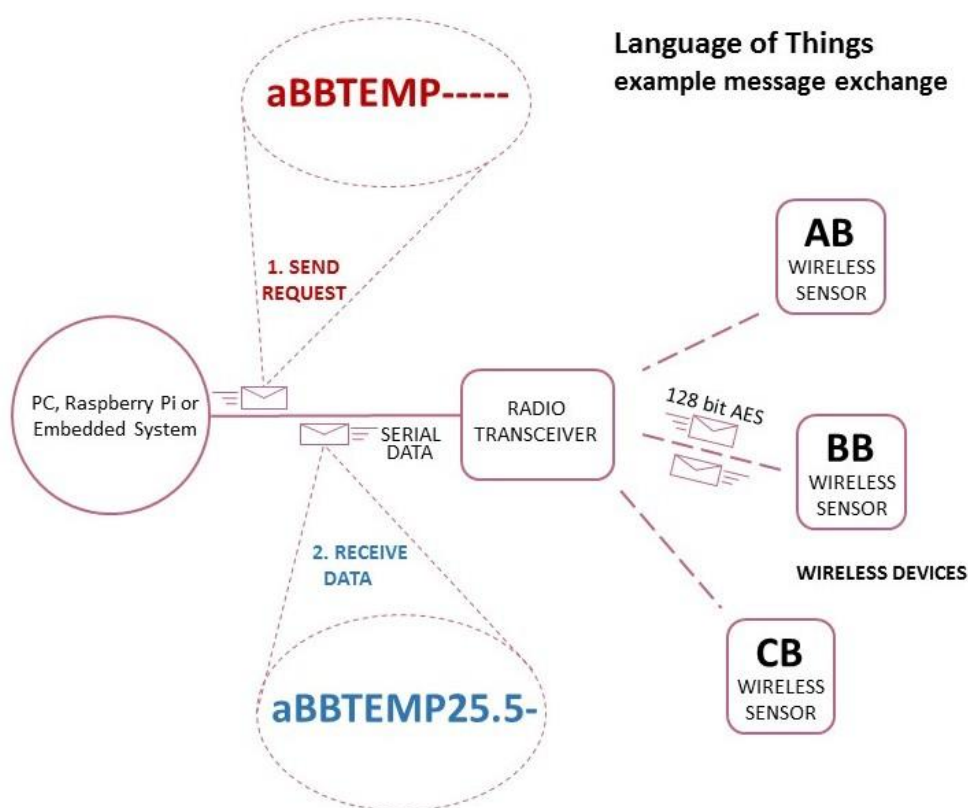
3.2.2. Language of Things

„Řeč věcí“, která spojuje senzory a Raspberry, je velice jednoduchá a účelná.



Obrázek 20 Schéma Language of Things [54]

Senzory většinu času stráví v klidném módu, kde nic neměří, jen šetří baterii. V daných intervalech se probudí, změří požadovanou veličinu (teplotu, vlhkost, ...), a přenesou informaci do přijímače. Přenos je šifrován 128 bitovým AES, takže se nedá jednoduše odposlouchávat, a proto je poměrně bezpečný.



Obrázek 21 Datové pakety

Z ukázky je patrné, že jazyk přenosu je velice jednoduchý a intuitivní, dá se v podstatě „dekódovat“ i bez návodu. Takže „Abbtemp-----“, znamená začátek paketu, BB – ID senzoru. Senzor může mít ID AA-ZZ, což umožňuje hodně zapojených senzorů současně. TEMP – příkaz, v tomto případě požadavek na přečtení teploty, -----zbytek znaků se vyplní pomlčkami, datový paket má vždy 12 znaků.

Tabulka 2 Ukázka paketů

Zpráva	Příklad	Popis
aXXTEMP	aXXTEMP29.5-	Senzor reportuje teplotu.
aXXAWAKE	aXXAWAKE----	Senzor reportuje připravenost.
aXXBATT	aXXBATT2.95-	Senzor reportuje aktuální stav baterie.
aXXSLEEPING	aXXSLEEPING-	Senzor reportuje, že přešel do úsporného módu.
aXXSTARTED	aXXSTARTED--	Zařízení bylo zapnuto nebo restartováno.
aXXBATTLOW	aXXBATTLOW-	Baterie je téměř vybitá.

3.2.3. Server

Raspberry Pi v roli serveru přináší několik problémů, ale na druhou stranu silně rozšiřuje možnosti využití, zejména o vzdálený přístup.

Jak uvádí [55], základním problémem je, že SD karta není připravena na to, aby se na ni neustále zapisovala a přepisovala data. V takových případech se většinou uvádí možnost překrytí části úložiště RAM diskem, ovšem to by vedlo na sníženou ukládací kapacitu. Já jsem se rozhodl pro externí pevný disk, na kterém server poběží. Zároveň se tím vytvoří poměrně velký prostor pro vlastní server, i na ukládání nejrůznějšího obsahu, který pak bude přístupný vzdáleně.

Software pro server

Webový server bude realizován pomocí serveru Apache, protože předpokládaný jednoduchý server obsahuje všechny potřebné funkce, a je zároveň celosvětově nejrozšířenější [56]. Má tedy nejdostupnější technickou podporu v podobě milionů uživatelů.

Jako interpret jsem zvolil PHP, ze stejného důvodu jako Apache: pro jednoduché účely má zaručeně všechny potřebné funkce a je celosvětově nejrozšířenější [57]. A nakonec jako databázový server jsem zvolil MySQL, dotazování v této multiplatformní databázi je realizováno pomocí nejrozšířenějšího databázového jazyka SQL. MySQL pak bylo vyvíjeno s důrazem na rychlost, a to za cenu některých zjednodušení, což ovšem vzhledem k jednoduchosti samotné aplikace nijak nevadí.

Vlastní cloud pak realizuji pomocí softwaru ownCloud, který umožní vzdálenou práci s dokumenty.

3.2.4. Programovací jazyk

Pro programování aplikace chytré domácnosti jsem se rozhodl využít programovacího jazyka Python. Python je víceparadigmatický, takže umožňuje při psaní programů používat nejen objektově orientované paradigma, ale i procedurální, a v omezené míře i funkcionální. Následně použije, které se v dané chvíli hodí nejlépe. Python má díky tomu vynikající vyjadřovací schopnosti. Kód programu bývá ve srovnání s jinými jazyky krátký a dobře čitelný. [58]

Pro naprogramování webového serveru bude použit PHP, jak již bylo zmíněno výše.

3.3. NAPROGRAMOVÁNÍ ŘÍDICÍHO SYSTÉMU

3.3.1. Realizovaný postup přípravy operačního systému

Postup, který jsem využil při realizaci je následovný: Nejprve je třeba stáhnout zvolený operační systém do PC (já jsem použil čistý a nejnovější Raspbian) a pomocí vhodného softwaru zapsat obraz na SD disk (použití programu Etcher). Poté SD disk umístit do Raspberry, připojit myš, klávesnici, monitor a nakonec napájecí USB kabel. Raspbian poté sám provede prvotní instalaci s nastavením, a Raspberry je připraveno k používání.⁵

Nyní je třeba připravit bootování z HDD. V bootovací konfiguraci je nutné nastavit *enable USB boot mode*. To nastaví bit v OTP (one time programmable) a umožní bootovat z USB.⁶ Dalším krokem bude příprava adresáře `/boot` a aktuálních bootovacích souborů příkazem:

```
$ sudo apt-get update && sudo apt-get upgrade
```

a spuštění *USB boot mode* příkazem

```
echo program_usb_boot_mode=1 | sudo tee -a /boot/config.txt
```

nakonec restartování příkazem:

```
sudo reboot
```

Nyní je třeba nastavit sériový port. Nejprve je nutné spustit konfigurační nástroj příkazem:

```
raspi-config
```

V položce *Interfacing Options* následovanou *P6 Serial* je třeba nastavit *No* na otázku "Would you like a login shell to be accessible over serial?" a *YES* na otázku "Would you like the serial port hardware to be enabled?" Poté *Exit and reboot*. Vzhledem k tomu, že je použito Raspberry Pi 3, tak je třeba vypnout Bluetooth, aby sériový port fungoval. Tedy:

```
Edit boot.txt
```

```
sudo nano /boot/cmdLine.txt
```

a zcela dole přidat následující parametr:

```
dtoverlay=pi3-disable-bt
```

Poté znovu restartovat.

Systém v pořádku naběhl a bootuje z HDD. Systém se zdá být funkční i bez SD karty a správně zobrazuje navýšení paměti. V mém případě téměř 1 TB.

⁵ Defaultní přihlašovací údaje jsou pi a heslo raspberry, v zájmu bezpečnosti silně doporučuji změnit tyto údaje na nějaké bezpečnější.

⁶ Vzhledem k použití OTP v SoC se jedná o trvalé nastavení a nelze ho změnit.

3.3.2. Příprava softwaru pro komunikaci se senzory

Jako první krok je zapotřebí stáhnout a nainstalovat software výrobce určený pro komunikaci příkazem:

```
cd /home
sudo wget -N www.privateeyepi.com/downloads/install.sh
sudo sh install.sh
```

Poté nainstalovat knihovny pro Python příkazem:

```
apt-get install python-serial
```

Do Raspberry se stáhne několik programů pro čtení senzorů a také možnost číst stav senzorů na webu. Jedná se o velice základní možnosti a více na <http://projects.privateeyepi.com>. V MySQL vytvořím databázi *hodnoty* se sloupci ID a hodnota, a jako primární klíč zvolíme ID.

Důležitá je následující sekvence umožňující čtení připojených senzorů:

```
def main():
    port = '/dev/ttyAMA0' # Deklarace proměnných.
    baud = 9600

    # Otevření sériového portu podle proměnných výše.
    ser = serial.Serial(port=port, baudrate=baud)

    sleep(0.2)
    print "Počkejte max 5 min na vyslání dat z vysílačů"

    while True:
        while ser.inWaiting():
            # Přečte znak
            char = ser.read()
            # Zkontroluje, že jde o začátek LLAP message
            if char == 'a':
                # začne načítat paket
                llapMsg = 'a'

                # Přečte dalších 11 znaků
                # Načte celý paket
                llapMsg += ser.read(11)

                # Rozdělí zprávu na data a ID
                devID = llapMsg[1:3]
                data = llapMsg[3:]

                print "ID zařízení : " + devID
                print "Data : " + data

                # Nahraje do databáze aktuální hodnoty
                cursor.execute("UPDATE hodnoty SET Hodnota=% WHERE ID=%s",
                    (data, devID))

            sleep(0.2)
```

3.3.3. Spuštění vlastního serveru

ownCloud

Službu ownCloud využijí jako můj vlastní cloud pro soubory, které bude možno vzdáleně spravovat a upravovat. Data je pak možné sdílet s ostatními nebo se synchronizovat například s notebookem.

Prvním krokem je stažení Release key a přidání ho mezi důvěryhodné:

```
cd /tmp  
wget http://download.opensuse.org/repositories/isv:ownCloud:community/Debian_8.0/Release.key  
apt-key add - < Release.key
```

Nyní stažení a instalace ownCloud:

```
echo 'deb  
http://download.opensuse.org/repositories/isv:ownCloud:community/Debian_8.0/' >>  
/etc/apt/sources.list.d/owncloud.list  
apt-get update  
apt-get install ownCloud
```

Během instalace ownCloud se nainstaluje i MySQL, a tak je třeba vytvořit správcovské heslo. Poté se přihlásit do MySQL a vytvořit databázi:

```
mysql --defaults-file=/etc/mysql/debian.cnf  
CREATE DATABASE owncloud;  
CREATE USER owncloud@localhost IDENTIFIED BY 'moje_heslo_do_mysql_serveru';  
GRANT ALL PRIVILEGES ON owncloud.* TO owncloud@localhost;  
flush privileges;  
quit
```

Defaultně využívá ownCloud složku `/var/www/owncloud/data`. V případě, že je umístění třeba změnit, provede se to pomocí příkazů:

```
mkdir /var/owndata  
chown www-data:www-data /var/owndata  
chmod 750 /var/owndata
```

Cloud již běží, a je možné se k němu přihlásit tak, že do prohlížeče zadáme IP adresu/ownCloud tedy například:

```
http://192.168.1.100/owncloud.
```

Při prvním spuštění se musí nastavit umístění složky s daty a využívanou databází. Nakonec umožníme přístup přes SSL:

```
a2enmod ssl  
a2ensite default-ssl  
service apache2 restart
```

Vlastní web

Webovou stránku lze psát v mnoha různých redakčních systémech. Já zvolil WordPress, znovu pro jeho masové rozšíření. V MySQL jsem vytvořil databázi s názvem WordPress a přešel do adresáře /var/www. V něm jsem vytvořil podsložku web. K vytvořené stránce se pak lze dostat na adrese IP/web. Dále jsem stáhnul WordPress příkazem:

```
$ sudo wget https://cs.wordpress.org/wordpress-4.1-cs_CZ.tar.gz
$ sudo tar -zxvf wordpress-4.1-cs_CZ.tar.gz
```

Poté už instalace probíhá v prohlížeči, nastaví se požadované údaje a po zadání adresy se objeví web a testovací příspěvek.

Web je pomocí WordPressu možno upravovat dle libosti. Zde jen uvedu nejdůležitější kód pro výpis z databáze, kam se ukládají data ze senzorů:

```
$hodnoty = Db::queryAll('
    SELECT *
    FROM hodnoty
');
echo('<h2>Hodnoty</h2><table border="1">');
foreach ($hodnoty as $u)
{
    echo('<tr><td>' . htmlspecialchars($u['ID']));
    echo('</td><td>' . htmlspecialchars($u['Hodnota']));
    echo('</td></tr>');
}
echo('</table>');
```

Tento PHP kód načte databázi a vypíše definované sloupce (v tomto případě ID a Hodnotu)

3.3.4. Naprogramování užitečných rozšíření

Odesílání SMS

Odeslání SMS v případě definovaných vstupů je užitečná schopnost systému. Je tak možné odeslat výstražné SMS v případě detekce nebezpečné situace. Například je tak možné upozornit na zapomenuté otevřené dveře nebo vrata, poslat výstrahu na únik vody z pračky a podobně.

Tuto funkci je možné realizovat například přes SMS bránu. Já jsem zvolil službu sms.sluzba.cz, kde je i dostupný skript pro odesílání. [59] Je třeba se na webu zaregistrovat a nabít kredit. Cena odeslání zprávy je 0,58 Kč. Nejprve je nutné nainstalovat api službu:

```
$ pip install smssluzbacz-api # stáhne api službu
```

A poté je již možné odesílat SMS příkazem:

```
from smssluzbacz_api.lite import SmsGateApi # naimportuje bránu
api = SmsGateApi('login', 'password', timeout=2, use_ssl=True) # login do systému
api.send('123456789', 'zpráva', use_post=True) # definuje odesílanou zprávu
```

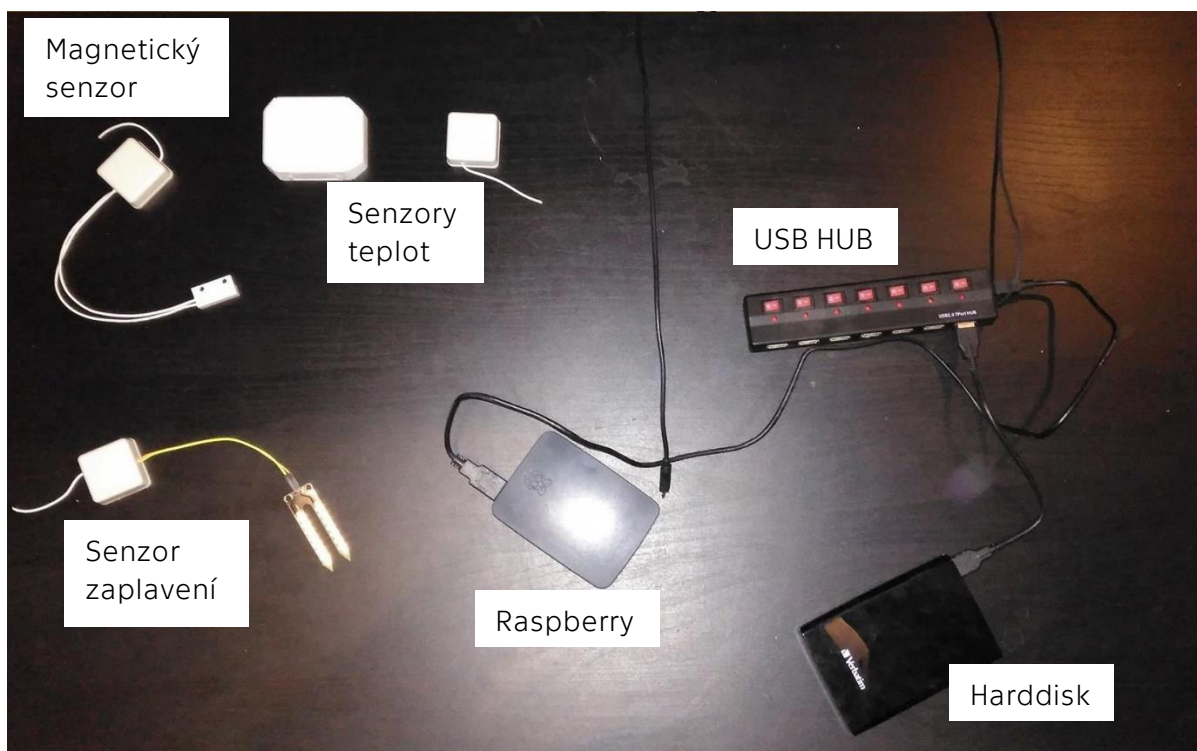
Znovuspuštění čtení programu při restartování Raspberry Pi

Pro hladký chod je výhodné nastavit cron job tak, aby při restartování raspberry došlo k znovuspuštění hlavního programu. Toho dosáhneme spuštěním následujícího programu jako crona.

```
process_name= "sensor.py" # jméno znovu spouštěného procesu  
tmp2 = os.popen("ps -Af").read()
```

```
if process_name not in tmp2[:] and globals.auto_rfsensor:  
    newprocess="nohup python %s/%s &" % (globals.install_directory, process_name)  
    os.system(newprocess) # pokud není proces spuštěn, spustí ho
```

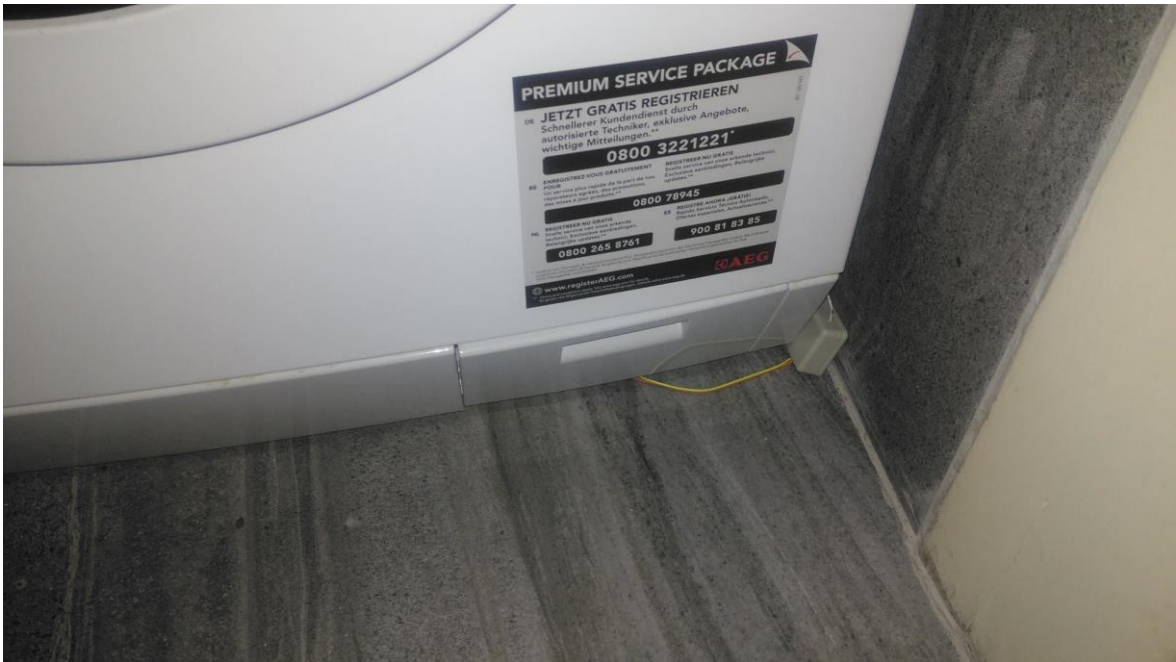
3.4. ZÁVĚR PRAKTICKÉ ČÁSTI



Obrázek 22 Vlastní sestavení systému



Obrázek 23 Instalovaný magnetický senzor



Obrázek 24 Instalovaný senzor záplavy



Obrázek 25 Instalované teplotní čidlo

System se podařilo v pořádku spustit a je připraven na aplikaci do reálné domácnosti. Všechny senzory při testování vracely správné hodnoty (referenční hodnoty byly určeny pomocí domácího teploměru a hodnoty binárních senzorů metodou fyzického vyzkoušení). Magnetický senzor vrací údaj o změně prakticky okamžitě, zatímco senzor zaplavení vykazoval prodlevu v řádu sekund při prvním ponoření a zhruba

poloviční prodlevu při následných ponořeních v rychlém sledu. Tento senzor se tedy nejvíce jako vhodný při aplikaci automatizování rychle tekoucí vody, jako je například robotické zalévání květin, protože by mohlo dojít vlivem zpoždění k přelití. Na druhou stranu například měření hladiny vany a následné uzavření ventilu by mělo být uskutečnitelné díky tomu, že se vana nenapouští rychle a je tak zmíněné zpoždění možné kompenzovat. Další možná aplikace je detekce unikající vody například z pračky nebo ventilu. Tuto aplikaci jsem ověřil a při vylití vody v oblasti senzoru došlo po pár vteřinách k ohlášení úniku. Při této aplikaci by bylo vhodné aplikovat také ovládaný ventil pro uzavření přítoku a zamezení škodám.

Výborně funguje vlastní cloud, který při testování splnil všechny požadované úlohy. Zejména pak umožnil editování souborů z jiného počítače, stahování souborů do mobilu a sdílení jednotlivých souborů pomocí odkazu.

Webová stránka splnila jednoduchý požadavek na reportování stavu senzorů a představuje největší potenciál pro další vývoj. Jako další kroky bych doporučoval zabezpečení proti útoku, umožnění ovládání Raspberry z webové stránky a umožnění nahrávání obrazu z webkamery přímo na web pro možnost vizuální kontroly domácnosti.

4. Závěr

Vlastní domácnost je jedním z pilířů lidského života a mnoho lidí si přeje automatizovat různé činnosti v domácnosti nebo mít alespoň možnost být informován o aktuální stavu a upozorněn na hrozící nebezpečí. Tato touha nezůstala nepovšimnuta jak mezi výrobci elektroniky, tak mezi internetovou komunitou.

Tato práce v první části přináší přehled prostředků, pomocí kterých lze takovou domácnost vytvořit. Jsou popsány jak nejvýznamnější hardwarové prvky, jako řídicí jednotky, senzory a akční členy, tak prvky softwarového systému, jako operační systém, server a programovací jazyk. Tím byl splněn primární cíl práce, a sice přinést ucelený přehled prostředků, kterými je možno vytvořit automatizovanou domácnost. V práci se také úspěšně povedlo shrnout nevýznamnější výhody a nevýhody těchto prvků a připravit tak dobrý podklad pro následné rozhodování, které z nich zvolit.

Prostředky jsem hodnotil z různých úhlů, ovšem s největším důrazem na vhodnost pro postavení automatizované domácnosti svépomocí. Nejdůležitějšími parametry tedy byly cena, komunitní podpora a otevřenost. Tyto parametry nejlépe splnil Raspberry Pi jako řídicí jednotka. Tato jednotka se pohybuje v nižší cenové hladině vůči ostatním hodnoceným systémům a je schopna plnit všechny požadované úkoly včetně multimediálního centra. Nabízí také masovou komunitní podporu zaměřenou na práci se senzory a nabízí, zvláště pro domácí využití, většinu potřebných programů zdarma. V případě, že není požadována funkce multimediálního centra nebo náhrady PC, je možné zvolit nižší verzi a ušetřit prostředky.

Jako nejvhodnější senzory se jeví ty od Privateeyepi, které jsou cenově nejdostupnější, ale je nutné je před použitím spájet a složit. Jejich spolehlivost vlivem nutnosti pájet obvod může být nižší, ale já jsem během krátkodobého testování žádné poruchy nezaznamenal. Porouchaný senzor je možné softwarově velice snadno odhalit (buď se pravidelně nehlásí nebo hlásí nesmyslné, případně stále stejné hodnoty).

V případě požadavku na komplexnější systém bych doporučil postavit automatizovanou domácnost na základě prostředků s protokolem Z-Wave. Tyto prostředky jsou ve vyšší cenové hladině, ale nabízejí větší rozsah výrobků a díky síti Mesh jsou schopné pokrýt větší plochu.

Softwarové prostředky jsou velice vyrovnané, co se týče schopností pro tuto aplikaci. Z vývojového hlediska lehce vede Linux díky otevřenosti systému a možnosti nainstalovat plný systém na jednodeskový počítač. Programovací jazyk Python se pak jeví výhodnější pro rychlý návrh a naprogramování systému, zatímco jazyk C pro komplexnější finální program. Jazyk Java je pak možné s výhodami použít při požadavku na multiplatformost.

V druhé části práce byl pak splněn sekundární cíl práce – návrh vlastního řešení pomocí vybraných produktů z oblasti IoT, a to zejména zprovoznění senzorů a jejich připojení k řídicí jednotce. Vlastní řešení bylo realizováno pomocí řídicí jednotky Raspberry Pi 3. Komunikace byla realizována rádiovými moduly od společnosti Ciseco a vlastní měření bylo provedeno senzory od společnosti PrivateEyePi. Softwarově je tento systém postavený na operačním systému Raspbian, databázi MySQL a serveru Apache s interpretem PHP.

Celý systém byl úspěšně krátkodobě otestován a vše běželo bezproblémově. Sensory správně četly požadované hodnoty a vyslaly je pomocí rádiového signálu do řídicí jednotky, kde byly úspěšně dekodovány a uloženy do databáze. Webová stránka pak úspěšně četla data z databáze a zobrazovala je na webu.

Systém v tomto stavu je připraven plnit nejrůznější činnosti automatizované domácnosti a je možné ho integrovat do domácnosti. Základní schopnosti tohoto systému jsou čtení stavu senzorů a jejich zobrazování, spuštění akčních členů v případě definovaných vstupů, schopnost fungovat jako multimediální centrum a možnost náhrady domácího PC. Mezi další schopnosti patří vlastní cloud pro soubory, provoz vlastního serveru, který umožní provoz vlastních webových stránek a je možné ho upravit a zprovoznit vlastní mailovou schránku. Díky softwaru třetích stran je možné webovou stránku upravovat v redakčním systému, prohlížet a spravovat soubory v cloudu v propracovaném prostředí na většině platform a ovládat některé akční členy pomocí Wi-Fi a aplikace v mobilu.

Další doporučené kroky pro vytvoření spolehlivě fungujícího systému domácí automatizace jsou: test dlouhodobého provozu, naprogramování uživatelského rozhraní umožňující jednoduché přidávání a ovládání prvků a realizování připojení mimo lokální síť – tento krok silně doporučuji spojit se zabezpečením celého systému proti napadení zvenčí.

5. Reference

- [1] „Přichází éra Internetu věcí. Ovlivní život každého z nás,“ 19 11 2014. [Online]. Available: <http://www.forbes.cz/prichazi-revolucni-trend-ktery-zmeni-zivot-kazdeho-z-nas/>. [Přístup získán 17 5 2017].
- [2] M. Hickey, „At Home with the Push Button Manor,“ 18 12 2006. [Online]. Available: <https://techcrunch.com/2006/12/18/at-home-with-the-push-button-manor/>. [Přístup získán 29 05 2017].
- [3] M. Zouzalík, „Kdy budeme mít chytré domy?,“ *21. století*, č. 19.11.2004, 2004.
- [4] Vodafone, „Vodafone si pro rozšíření služby Internet věcí vybral technologii NB-IoT,“ 29 6 2016. [Online]. Available: <https://www.vodafone.cz/o-vodafonu/o-spolecnosti/pro-media/tiskove-zpravy/detail/vodafone-si-pro-rozsireni-sluzby-internet-veci-vyb/>. [Přístup získán 17 5 2017].
- [5] A. Vojáček, „Základní typy jednodeskových počítačů - Embedded SBC,“ 21 1 2007. [Online]. Available: <http://automatizace.hw.cz/clanek/2007012101>. [Přístup získán 3 6 2017].
- [6] L. Čížek, „ZAŘÍZENÍ PRO SBĚR A REPREZENTACI DAT MĚŘENÝCH JEDNOÚČELOVÝM STROJEM,“ 2016. [Online]. Available: https://otik.uk.zcu.cz/bitstream/11025/23075/1/bp_cizek.pdf. [Přístup získán 3 6 2017].
- [7] [Online]. Available: <https://arduino.cz/>. [Přístup získán 10 05 2017].
- [8] M. Selecký a M. Herodek, *Arduino: uživatelská příručka*, Brno: Computer Press, 2016.
- [9] J. Blum, *Exploring Arduino: tools and techniques for engineering wizardy*, Indianapolis: Wiley, 2013.
- [10] [Online]. Available: <https://www.arduino.cc/en/uploads/Main/ArduinoDiecimilaComponents.jpg>. [Přístup získán 21 05 2017].
- [11] N. P. Shashank Sharma, „5 of the most popular Raspberry Pi distros,“ 13 12 2015. [Online]. Available: <http://www.techradar.com/news/software/5-of-the-most-popular-raspberry-pi-distros-1292537>. [Přístup získán 3 6 2017].
- [12] [Online]. Available: <https://www.raspberrypi.org/>. [Přístup získán 10 05 2017].
- [13] E. Upton, G. Halfacree a J. Goner, *Raspberry Pi: uživatelská příručka*, Brno: Computer Press, 2016.

- [14] S. Monk, Programming the Raspberry Pi: getting started with Python, New York: McGraw-Hill, 2013.
- [15] D. NORRIS, Raspberry Pi: projekty, Brno, 2015.
- [16] [Online]. Available:
<https://www.element14.com/community/servlet/JiveServlet/showImage/102-80899-15-252356/Pi3+Breakout+Feb+29+2016.png>. [Přístup získán 10 05 2017].
- [17] [Online]. Available: <https://software.intel.com/sites/default/files/managed/74/18/iot-edison-hero-image.png>. [Přístup získán 10 05 2017].
- [18] Co je Tecomat Foxtrot?, „Co je Tecomat Foxtrot?,” [Online]. Available:
http://www.tecomat.com/clanek__388__co-je-tecomat-foxtrot__.html. [Přístup získán 3 6 2017].
- [19] L. Urban, „Tecomat Foxtrot - nový modulární PLC od firmy Teco,” *Automa*, pp. 18-19, 10 2007.
- [20] [Online]. Available:
http://www.tecomat.com/wpimages/fotoprod/1200__panel__Foxtrot.jpg. [Přístup získán 10 05 2017].
- [21] [Online]. Available: <https://www.mojefibaro.cz/>. [Přístup získán 10 05 2017].
- [22] C. Ing. František Vdoleček, TECHNICKÁ MĚŘENÍ, Brno: VUT FSI v Brně , 2002.
- [23] [Online]. Available: <http://www.monnit.com/>. [Přístup získán 10 05 2017].
- [24] [Online]. Available: <https://ha.privateeyepi.com/store/index.php?route=common/home>. [Přístup získán 10 05 2017].
- [25] [Online]. Available:
http://www.tecomat.com/wpimages/other/DOCS/cze/PRINTS/Cat__Foxtrot-CZ-datasheets/Foxtrot-CZ-R-IB-0400B.pdf. [Přístup získán 10 05 2017].
- [26] R. W. R. F. Jiří Tůma, Základy automatizace, Ostrava: Vysoká škola báňská – Technická univerzita Ostrava , 2007.
- [27] [Online]. Available: http://projects.privateeyepi.com/_/rsrc/1468849493480/home/on-off-project/circuit%20%28640x550%29.jpg?height=343&width=400. [Přístup získán 10 05 2017].

- [28] [Online]. Available: https://ha.privateeyepi.com/store/index.php?route=product/product&path=66&product_id=130. [Přístup získán 10 05 2017].
- [29] [Online]. Available: <https://www.kickstarter.com/projects/switchbot/switch-bot-the-worlds-smallest-remote-robot/faqs>. [Přístup získán 10 05 2017].
- [30] [Online]. Available: <https://res.cloudinary.com/indiegogo-media-prod-cld/image/upload/v1481219741/tcxctw1cvadctziszsq5i.png>. [Přístup získán 5 06 2017].
- [31] [Online]. Available: https://ha.privateeyepi.com/store/index.php?route=product/product&path=65&product_id=64. [Přístup získán 10 05 2017].
- [32] [Online]. Available: <https://ha.privateeyepi.com/store/image/cache/data/buzzer2-228x228.jpg>. [Přístup získán 10 05 2017].
- [33] [Online]. Available: <https://www.mojefibaro.cz/doplňky/zaluziový-modul/>. [Přístup získán 10 05 2017].
- [34] [Online]. Available: <https://www.mojefibaro.cz/produkty/termostaticka-hlavice/>. [Přístup získán 10 05 2017].
- [35] [Online]. Available: <https://www.mojefibaro.cz/wp-content/uploads/2015/09/danfoss-install-420x300.png?x81775>. [Přístup získán 10 05 2017].
- [36] [Online]. Available: https://homeautomationfordummies.files.wordpress.com/2015/01/014g0012_4-rid.jpg. [Přístup získán 10 06 2017].
- [37] [Online]. Available: <https://owncloud.org/>. [Přístup získán 10 05 2017].
- [38] [Online]. Available: <https://syncthing.net/>. [Přístup získán 10 05 2017].
- [39] [Online]. Available: <https://www.seafile.com/en/home/>. [Přístup získán 10 05 2017].
- [40] L. Jelínek, „PHP programovacím jazykem nové generace?“, 3 9 2014. [Online]. Available: <https://www.linuxexpres.cz/novinky/php-programovacim-jazykem-nove-generace>. [Přístup získán 26 5 2017].
- [41] „TIOBE Index for May 2017,“ [Online]. Available: <https://www.tiobe.com/tiobe-index/>. [Přístup získán 26 5 2017].
- [42] M. Simon, „Arduino Vs. Raspberry Pi: Which Is The Right DIY Platform For You?“, 07 05 2014. [Online]. Available: <http://readwrite.com/2014/05/07/arduino-vs-raspberry-pi->

projects-diy-platform/.

- [43] [Online]. Available: <http://f4b1.com/files/img/old/raspberry-pi-3-vs-pi-2-vs-pi-zero-vs-a-vs-b-benchmark-sysbench.png>. [Přístup získán 10 05 2017].
- [44] [Online]. Available: <https://www.wirelessthings.net/slice-of-radio-wireless-rf-transciever-for-the-raspberry-pi>. [Přístup získán 10 01 2016].
- [45] [Online]. Available: <http://projects.privateeyepi.com/home/home-alarm-system-project/wireless-projects/wireless-relay-project>. [Přístup získán 10 05 2017].
- [46] [Online]. Available: <https://www.wirelessthings.net/wireless-temperature-senzor>. [Přístup získán 10 01 2016].
- [47] [Online]. Available: http://ha.privateeyepi.com/store/index.php?route=product/product&path=66&product_id=115. [Přístup získán 10 05 2017].
- [48] [Online]. Available: <http://www.vishay.com/docs/29049/ntcle100.pdf>. [Přístup získán 10 05 2017].
- [49] [Online]. Available: http://automatizace.hw.cz/images/magnet_sensors/smallreed_solomon.gif. [Přístup získán 21 05 2017].
- [50] [Online]. Available: http://ha.privateeyepi.com/store/image/cache/data/rf-mini/IMG_5000-80x80.JPG. [Přístup získán 10 05 2017].
- [51] [Online]. Available: [http://ha.privateeyepi.com/store/image/cache/data/rf-mini/IMG_5060%20\(632x640\)-228x228.jpg](http://ha.privateeyepi.com/store/image/cache/data/rf-mini/IMG_5060%20(632x640)-228x228.jpg). [Přístup získán 10 05 2017].
- [52] [Online]. Available: http://www.projects.privateeyepi.com/_/rsrc/1468849513279/home/home-alarm-system-project/wireless-projects/wireless-temperature-sensor-2-0/battery%20clip%20%28640x630%29.jpg?height=315&width=320. [Přístup získán 10 05 2017].
- [53] [Online]. Available: http://www.projects.privateeyepi.com/_/rsrc/1468849510573/home/home-alarm-system-project/wireless-projects/wireless-temperature-sensor-2-0/IMG_5017%20%28640x480%29.jpg?height=240&width=320. [Přístup získán 10 05 2017].
- [54] [Online]. Available: https://www.wirelessthings.net/media/wysiwyg/message_bridge_diagram.jpg. [Přístup získán 10 01 2016].

- [55] M. Krúpa, „52) RAMDisk,“ 10 6 2014. [Online]. Available: <http://www.astromik.org/raspi/52.htm>.
- [56] Netcraft, „November 2005 Web Server Survey,“ 7 12 2005. [Online]. Available: http://news.netcraft.com/archives/2005/11/07/november__2005__web__server__survey.html.
- [57] „Usage of server-side programming languages for websites,“ [Online]. Available: https://w3techs.com/technologies/overview/programming__language/all. [Přístup získán 16 3 2017].
- [58] Python, „Comparing Python to Other Languages,“ [Online]. Available: <https://www.python.org/doc/essays/comparisons/>. [Přístup získán 29 05 2017].
- [59] [Online]. Available: https://sms.sluzba.cz/sms__gate/doc. [Přístup získán 10 05 2017].
- [60] [Online]. Available: https://www.wirelessthings.net/media/catalog/product/cache/1/small__image/195x/9df78eab33525d08d6e5fb8d27136e95/r/0/r016__art.jpg. [Přístup získán 10 01 2016].
- [61] [Online]. Available: [http://ha.privateeyepi.com/store/image/cache/data/rf-mini/IMG_4963%20\(640x480\)-80x80.jpg](http://ha.privateeyepi.com/store/image/cache/data/rf-mini/IMG_4963%20(640x480)-80x80.jpg). [Přístup získán 10 05 2017].
- [62] [Online]. Available: https://www.wirelessthings.net/media/catalog/product/cache/1/small__image/195x/9df78eab33525d08d6e5fb8d27136e95/l/0/I0001.jpg. [Přístup získán 10 01 2016].

I. Přílohy

I.I. ZNOVUSPUŠTĚNÍ PROGRAMU

```
import os  
import globals
```

```
globals.init()
```

```
process_name = "sensor.py" # změnit na jméno programu  
tmp2 = os.popen("ps -Af").read()
```

```
if process_name not in tmp2[:] and globals.auto_rfsensor:  
    newprocess="nohup python %s/%s &" % (globals.install_directory, process_name)  
    os.system(newprocess)
```

I.II. ODESÍLÁNÍ SMS

```
$ pip install smssluzbacz-api
```

A poté je již možné odesílat SMS příkazem:

```
from smssluzbacz_api.lite import SmsGateApi  
api = SmsGateApi('login', 'password', timeout=2, use_ssl=True)  
api.send('123456789', 'zpráva', use_post=True)
```

I.III. ČTENÍ SENZORŮ SE ZÁPÍSEM DO DATABÁZE

```
import serial
import time
import sys
import MySQLdb as mysql
import pymysql as mysql
from time import sleep

conn = mysql.connect(host='localhost',user='root',password'',db='hodnoty')
curs = conn.cursor()
conn.autocommit(True)

def main():
    # Deklarace proměnných.
    port = '/dev/ttyAMA0'
    baud = 9600

    # Otevření sériového portu podle proměnných výše.
    ser = serial.Serial(port=port, baudrate=baud)

    sleep(0.2)

    print "Počkejte max 5 min na vyslání dat z vysílačů"

    while True:
        while ser.inWaiting():
            # Přečte znak
            char = ser.read()
            # Zkontroluje, že jde o začátek LLAP message
            if char == 'a':
                # začne načítat paket
                llapMsg = 'a'

                # Přečte dalších 11 znaků
                # Načte celý paket
                llapMsg += ser.read(11)

                # Rozdělí zprávu na data a ID
                devID = llapMsg[1:3]
                data = llapMsg[3:]

                print "ID zařízení : " + devID
                print "Data : " + data

                cursor.execute("UPDATE hodnoty SET Hodnota=%s WHERE ID=%s",
                    (data, devID))

                sleep(0.2)

if __name__ == "__main__":
    main()
```