

Jihočeská univerzita v Českých Budějovicích
Přírodovědecká fakulta

Možnosti využití Internetu věcí a Robot Operating System

Bakalářská práce

Michal Šenigl

Školitel: Mgr. Jiří Pech, Ph.D.

České Budějovice

2019

Šenigl, M., 2019: Možnosti využití Internetu věcí a Robot Operating System. [Possible utilization of the Internet of Things and Robot Operating System. Bc. Thesis, in Czech.] – 36 p., Faculty of Science, University of South Bohemia, České Budějovice, Czech Republic.

Anotace:

Tato práce prezentuje možnosti využití Internetu věcí a Robot Operating System, zejména v oblasti automatizace a optimalizace procesů. Zaměřuje se na možná řešení nadcházejících problémů jako je rostoucí populace a globální oteplování s pomocí internetu věcí a hodnotí, zda je v dané situaci Robot Operating System přínosem. Zároveň obsahuje model chytré domácnosti jako prezentaci možností Robot Operating System.

Annotation:

Title: Possible utilization of the Internet of Things and Robot Operating System

This thesis presents possibilities of using the Internet of Things and Robot Operating System, especially in the field of automation and process optimization. It focuses on possible solutions to upcoming challenges such as growing population and global warming using the Internet of Things and assesses whether the Robot Operating System is beneficial in given situation. It also includes a model of smart home as presentation of Robot Operating System capabilities.

Prohlašuji, že svoji bakalářskou práci jsem vypracoval samostatně pouze s použitím pramenů a literatury uvedených v seznamu citované literatury.

Prohlašuji, že v souladu s § 47b zákona č. 111/1998 Sb. v platném znění souhlasím se zveřejněním své bakalářské práce, a to v nezkrácené podobě elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách, a to se zachováním mého autorského práva k odevzdanému textu této kvalifikační práce. Souhlasím dále s tím, aby toutéž elektronickou cestou byly v souladu s uvedeným ustanovením zákona č. 111/1998 Sb. zveřejněny posudky školitele a oponentů práce i záznam o průběhu a výsledku obhajoby kvalifikační práce. Rovněž souhlasím s porovnáním textu mé kvalifikační práce s databází kvalifikačních prací Theses.cz provozovanou Národním registrem vysokoškolských kvalifikačních prací a systémem na odhalování plagiátů.

České Budějovice, dne 9.12.2019

Poděkování

Mé poděkování patří Mgr. Jiřímu Pechovi, Ph.D. za trpělivost, ochotu, zapůjčení nástrojů a odborné vedení, které mi v průběhu zpracování bakalářské práce věnoval.

Obsah :

1	Úvod.....	1
1.1	Robot Operating System	1
1.2	Definice pojmů z oblasti ROS.....	2
1.3	Průběh komunikace uzlů ROS.....	3
1.4	Úvod do IoT.....	4
1.5	Definice IoT.....	5
1.6	Definice pojmu „Thing“.....	5
2	Cíle práce.....	6
3	Integrace ROS do internetu věcí.....	6
3.1	Výhody ROS.....	7
3.1.1	Licence.....	7
3.1.2	Programovací jazyky.....	7
3.2	Bezpečnost ROS.....	8
3.3	Přínosy IoT.....	9
3.3.1	Osobní zařízení.....	9
3.3.1.1	Chytrá domácnost.....	9
3.3.1.2	Chytré doplňky.....	10
3.3.2	Zemědělství	10
3.3.2.1	Důvod pro IoT v zemědělství	11
3.3.2.2	Data-Driven Agriculture.....	11
3.3.2.3	DDA v kombinaci s umělou inteligencí.....	11
3.3.2.4	Precizní Agrikultura.....	12
3.3.3	Výroba.....	13
3.3.3.1	Industry 4.0.....	13
3.3.3.2	ROS Industrial.....	13
3.3.4	Výzkum.....	14
3.3.4.1	Enviromentální monitoring.....	14
3.3.5	Infrastruktura.....	14
3.3.5.1	Smart city.....	15
3.3.5.2	Doprava.....	15

4	Funkční model z oblasti IoT.....	16
4.3	Použitý hardware.....	16
4.3.1	Raspberry Pi.....	16
4.3.2	Senzor teploty a vlhkosti DHT-22.....	17
4.3.3	Relé SRD-5VDC-SL-C.....	17
4.3.4	Krokový motor 28BYJ-48 a drive ULN2003.....	18
4.3.5	Vývojová deska Arduino CH34.....	18
4.3.6	Vlastní senzor kvality osvětlení.....	18
4.4	Konstrukce modelu.....	20
4.2.1	Instalace základního OS.....	20
4.4.1	Instalace nástavby ROS.....	21
4.4.2	Vytvoření balíčku v ROS.....	21
4.4.3	Regulace teploty v domácnosti.....	22
4.4.3.1	Publisher.....	22
4.4.3.2	Zapojení senzoru a zpracování dat.....	23
4.4.3.3	Subscriber.....	24
4.4.3.4	Rqt graf.....	25
4.4.4	Ovládání a regulace osvětlení.....	26
4.4.4.1	Ovládání relé.....	26
4.4.4.2	Výroba plošného spoje.....	27
4.4.4.3	PWM.....	27
4.4.4.4	Krokový motor.....	28
5	Diskuze.....	29
5.3	Hodnocení aplikace ROS napříč obory IoT.....	29
5.4	Hodnocení ROS v chytré domácnosti.....	31
6	Závěr.....	32
	Reference.....	33
	Obrázky.....	35
	Grafy.....	35
	Tabulky.....	35
	Přílohy.....	36

1. Úvod

S rostoucí světovou populací je nutné uvažovat o možnostech optimalizace procesů pro zvýšení produkce jak potravin a energetické produkce, tak spotřebních doplňků. Zároveň je nutné zvážit environmentální dopady ekonomiky na životní prostředí a hledat udržitelné metody pro další rozvoj lidstva.

Zavedení internetu věcí nám umožnilo získávat data z reálného světa neuvěřitelnou rychlostí. S rostoucím počtem zařízení připojených k internetu a přesností senzorů roste velikost dostupných datasetů. Tak se stává interpretace a využití získaných dat náročnější. Využitím moderních technologií jako je real-time analytika a umělá inteligence založená na principu neuronové sítě, lze získávat výsledky téměř ihned a aplikovat je pro zvyšování efektivity procesů.

Robot Operating System je přínosem pro získávání těchto dat i následnou reakci v reálném světě na základně výsledků. To otevírá možnosti v oblasti automatizace a optimalizace s využitím internetu.

Pro demonstraci zapojení Robot Operating System do každodenního života byl vytvořen a popsán model systému pro internet věcí, který je součástí této práce. Takzvaná chytrá domácnost má potenciál ilustrovat možnosti ovládní a regulace v reálném světě, to z ní dělá ideální příklad.

1.1 Robot Operating System

Robot Operating System (dále jen ROS) byl počat v roce 2007 jako projekt společnosti Willow Garage, ale dokonce roku se k projektu připojilo přes 20 dalších institucí, to znamená, že ROS byl od počátku plánován jako multi-účelový systém.[1]

ROS není opravdovým operačním systémem, jak by název mohl naznačovat, ale jedná se pouze o nastavbu operačních systémů na bázi Linuxu jako Ubuntu, Debian a dalších operačních systémů. Je open-source a užívá se pro vývoj a simulace převážně robotů, ale i dalších zařízení internetu věcí.

Obsahuje služby, které jsou očekávány od operačního systému, a to abstrakci hardware, ovládání nízko-úrovňových zařízení, implementuje často užívané funkce, předává zprávy mezi procesy. Poskytuje také nástroje a knihovny pro získávání, vytváření, psaní a spouštění kódu na více zařízeních.[2]

1.2 Definice pojmů z oblasti ROS

ROS je znám aplikací kombinace takzvaných nodes, messages, a topics. S pomocí těchto nástrojů lze vytvořit síť předávající pouze hodnoty důležité pro dosažení/udržení požadovaného stavu dané soustavy.

Nodes (v překladu uzle), často jednoduché programy, které řeší malou podmnožinu problémů, jsou výpočetní částí ROS.

Zprávy vysílá **publisher**, typ uzlu odesílající informace. Nejčastěji se jedná o senzory, jejichž hodnota se mění na základě změn v okolí, ale může to být například i sám uživatel pomocí počítačového interface, nebo fyzického ovládání.

Pro příjem dat se využívá uzel zvaný **subscriber**, pomocí kterého lze data přijímat a využít je pro ovládání motorů, světel, ohřevu a mnoho dalších zařízení.

Messages (v překladu zprávy) jsou přesně zadané datové struktury. Podporují základní typy proměných (integer, floating point, boolean a další), stejně jako pole z nich složená. [1]

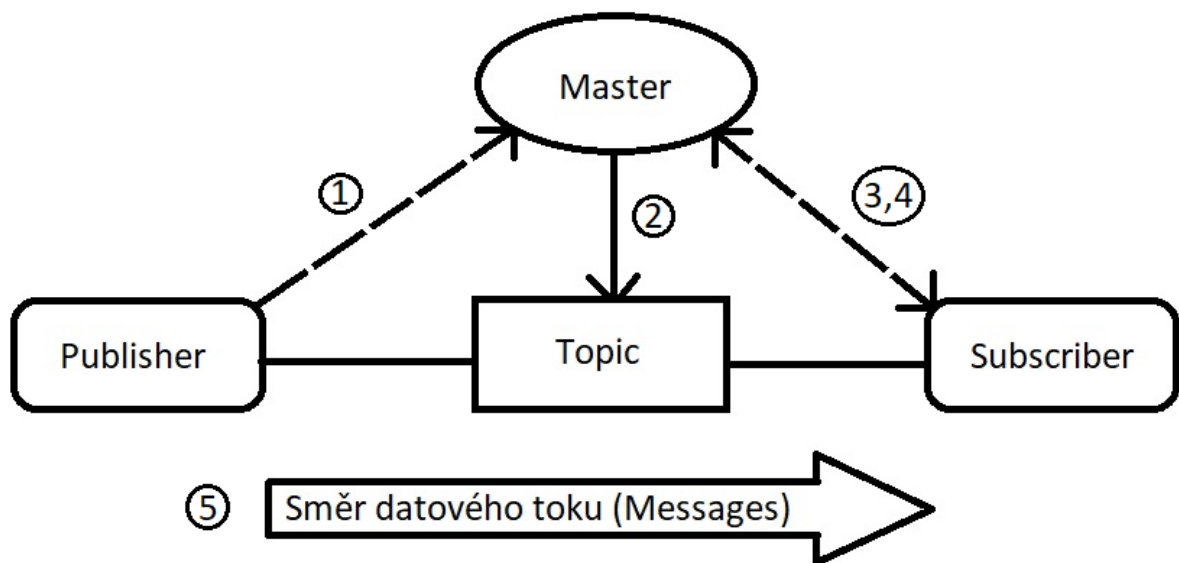
Pokud uzel potřebuje vyslat, nebo přijmout zprávu využije takzvaných **topics** (v překladu témata), fungují jako mapa datových toků.

Publisher si obecně není vědom, že existuje subscriber a naopak, takže mohou být na různých zařízeních. Pro jeden topic lze mít mnoho publisherů, a/nebo subscriberů. Zároveň jeden uzel může být propojen s mnoha topics. [1]

Díky modulárnosti lze propojit několik zařízení, která se mohou spolupracovat a dělit se o nároky na výpočetní kapacitu. Pomocí „**rqt grafu**“, ten ukládá registrační informace o topics a tím umožní správné propojení uzlů.[3]

Master, hlavní bod ROS systému uděluje ID („jméno“) registrovaným uzlům a ukládá výpočetní graf, tak určuje funkci sítě. Bez mastera nemůžou ostatní prvky sítě komunikovat, to ovšem neznamená, že skrz prochází data. Pouze „ukazuje“ odkud kam topic předává messages. [3]

1.3 Průběh komunikace uzlů ROS



Obrázek 1) Princip navázání komunikace mezi uzly ROS

Propojení uzlů probíhá takto, publisher získá data, například prostřednictvím senzoru, data zpracuje a vyšle své ID masteru s žádostí o vytvoření topic (1 na schématu). Master ověří ID a vytvoří požadovaný topic (2 na schématu), publisher mezitím uloží data do topic, ale subscriber zatím není připojen, takže přenos dat neprobíhá.

Subscriber si vyžádá informaci o topics (3 na schématu). V případě, že existuje topic, který chce subscriber odebírat, master odešle údaje pro připojení k topic (na schématu 4).

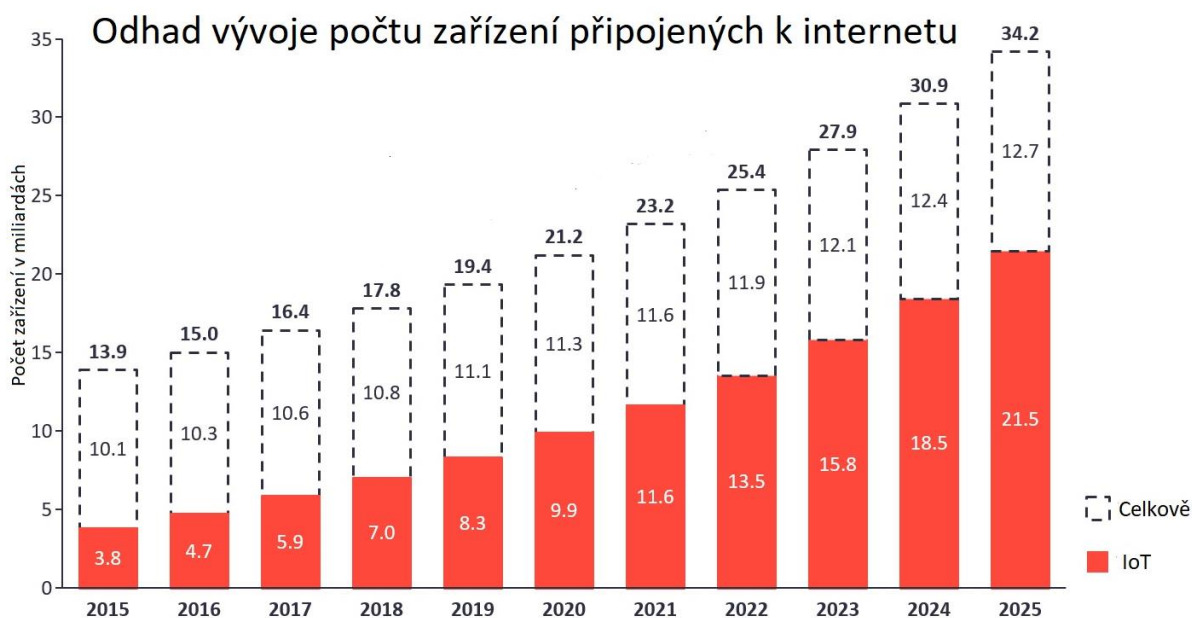
Po propojení uzlů skrz topic mezi nimi začíná přenos publikovaných dat (5 na schématu). Mezi masterem a uzly v tuto chvíli neprobíhá přenos dat.

1.4 Úvod do IoT

„Internet of Things“ (dále jen IoT) propojuje zařízení vybavená adekvátním software pomocí již existujících sítí a umožňuje jim výměnu dat a vzdálenou kontrolu. To otevírá mnoho možností v oblasti automatizace repetitivních úkonů, a celkového zvýšení kvality života, díky snižování nároků na obsluhu, případně, až naprosté automatizace. Tím se vyřazuje lidský prvek, čímž se snižují náklady a zároveň se vyřazuje možnost lidské chyby, a tak roste celková kvalita služeb.

Výraz byl poprvé použit v roce 1999, kdy se dostali do prominence Radio Frequency Identification (dále jen RFID) čipy, užívané pro zvýšení efektivity zásobovacích řetězců s pomocí internetu. [4] Dodnes neexistuje jednotně uznávaná definice tohoto výrazu, a to z důvodu, že pokrývá velmi široké spektrum zařízení, pasivních i aktivních a s každým dnem roste integrace internetu do běžného života.

Je odhadováno, že v roce 2020 bude počet zařízení připojených k internetu přes 21 miliard (Graf1) a inteligentní zařízení překročí hranici 10 miliard kusů. [5] Hodnota trhu za hardware pro IoT aplikace se přiblíží hodnotě 80 miliard USD. [6]



Graf 1) Odhad vývoje počtu zařízení s přístupem k internetu společnosti IoT Analytics

V grafu (Graf 1) nejsou do kategorie IoT započítány osobní, firemní počítače a telefonní linky, které se nezapojují do ovládání prvků reálného světa a tím pádem fungují jen jako rozhraní mezi člověkem a internetem. [5]

1.5 Definice IoT

Dodnes jedna z nejčastěji citovaných definic IoT z práce výzkumné skupiny S.Hallera:

“A world where physical objects are seamlessly integrated into the information network, and where the physical objects can become active participants in processes. Services are available to interact with these „smart objects,, over the Internet, query their state and any information associated with them, taking into account security and privacy issues.“ [7]

V překladu: „Svět, kde se reálné objekty jednoduše zapojí do informační sítě a kde se reálné objekty mohou stát aktivním účastníkem procesů. Služby (např. mobilní aplikace) jsou schopné interakce s těmito „inteligentními objekty“ přes internet, a tázat se na jejich stav a veškeré informace s nimi spojené, s ohledem na otázky bezpečnosti a soukromí.“

1.6 Definice pojmu „Thing“

Pod pojem „Things“ (z Internet of Things) dnes spadá velké množství zařízení, aby zapadali do kontextu IoT je nutné, aby dokázali komunikovat s jinými zařízeními i bez lidské interakce.

Například smartphone by většina populace považovala za nástroj ovládaný člověkem, ale pod povrchem smartphone z internetu přijímá a vysílá zpět data několikrát denně, i když se jedná o jednoduché aktualizace, nebo informace jako je poloha podle Globálního Polohového Systému (dále jen GPS).

Tyto informace mohou velmi usnadnit život, například v případě ztráty telefonu se dá snadno dohledat, nebo mobilní aplikace jako je Uber společnost využívající GPS technologii k revoluci v městské dopravě, snížením potřeby pro jednotlivce vlastnit auto, díky kompetitivním cenám a pohodlí oproti integrované dopravě. [8]

Na druhou stranu je třeba se s rostoucím počtem takzvaných „things“ zaměřit na negativa této technologie, převážně v otázkách soukromí. Z toho důvodu je veřejnost nutné poučit jak o výhodách, tak nevýhodách IoT, které jsou často ignorovány.

2. Cíle práce

Cílem teoretické části této práce je popsat princip fungování, definovat výrazy a určit výhody a bezpečnostní rizika spojená s ROS, dále prostudovat a zhodnotit přínos a efektivitu různých aplikací IoT systémů napříč odvětvími, a určit pro které aplikace je ROS efektivním řešením problému.

Cílem praktické části je sestavení funkčního modelu model z oblasti IoT, takzvané chytré domácnosti. Ukázka a popsání operace ROS v rozličných případech užití, a to:

- 1) Shromáždění a interpretace dat ze senzorů (osvětlení a teplota) do použitelného formátu v rámci uzle (publisher).
- 2) Navázání spolehlivé komunikace mezi uzly, užitím takzvaných topics.
- 3) Reakce uzlu (subscriber) na hodnoty přijaté ze senzorů, pro potřeby řízení vytápění a klimatizace.
- 4) Regulace výkonu zařízení systému s pomocí pulzní šířkové modulace.
- 5) Vzdálené ovládání aktivních prvků domácnosti jako jsou žaluzie a osvětlení.

3. Integrace ROS do internetu věcí

ROS byl vytvořen jako kooperace mezi několika institucemi, každá měla svoje vlastní potřeby a tak se z ROS stal multi-účelový nástroj, dokáže ovládat širokou škálu zařízení a senzorů.

Drtivá většina IoT zařízení funguje na některé z distribucí linuxu. To dává ROS možnost zapojit se téměř do každé sekce IoT s minimální nutností modifikace již existujících balíčků a knihoven, které jsou často open-source a zdarma stejně jako sám ROS. Integrace ROS systémů do IoT přináší určité výhody, ale i bezpečnostní rizika.

3.1 Výhody ROS

ROS byl navrhnut jako nástroj pro rychlý vývoj, díky tomu že je open-source obsahuje mnoho užitečných nástrojů z dalších open-source projektů.[9]

Díky modulárnosti lze stáhnout jen požadované balíčky a ušetřit tak místo v úložišti. Dnes existuje přes 3000 komunitních balíčků [10]. ROS nebyl vytvořen pro specifický úkol, to mu dává široké možnosti uplatnění v robotice i IoT aplikacích.

3.1.1 Licence

Jádro ROS je distribuováno pod standartní open-source licencí BSD. Tato licence je zdarma a otevřená pro všechny s nenáročnými pravidly, umožňuje dokonce komerční využití v closed-source projektech.[10]

Zatím, co jádro ROS je pod licencí BSD, mnoho balíčků má jiné licenční podmínky. Proto je vyžadováno, aby každý ROS balíček specifikoval své vlastní podmínky užití. To umožňuje rychlou identifikaci balíčků s potřebnou licencí.[10]

3.1.2 Programovací jazyky

Dnes existuje mnoho programovacích jazyků, každý má své výhody i nevýhody. Programátoři volí jazyk, ve kterém programují, podle osobních preferencí, proto byl ROS vyvinut jazykově nezávislý. To umožňuje programování v mnoha jazycích, s oficiální podporou C++, Python a Lisp. Experimentálně lze ROS programovat i v Java a Lua. [2]

Prostředí jsou vyvinutá i pro další varianty C a Octave, ale jedná se o port, ne nativní podporu.[1]

Aby ROS podporoval širokou škálu jazyků, využívá, takzvaný Interface Definition Language (IDL), který popisuje přenos messages po síti a v neutrálním jazyce. Například třířádkový IDL dokáže splnit funkci, která v Pythonu zabere 91 řádek. [1]

3.2 Bezpečnost ROS

Bezpečnost robotických systémů je maximálně důležitá a s každým rokem roste jak počet senzorů připojených k internetu, tak jejich možnosti interpretovat okolní svět. To působí obavy v oblasti soukromí.

Zároveň stoupá počet aktuátorů (akčních prvků), které mají možnost ovlivňovat reálný svět, to způsobuje obavy ve směru bezpečnosti reálného světa, například semaforey se mohou stát terčem útoku.

V roce 2012 byl na konferenci o internetové bezpečnosti DEF CON 20 proveden experiment, kdy se hackeři pokusili dostat do ROS systému. Zkušený uživatel ROS během experimentu odhalil slabinu, která umožňovala vkládání messages do operujícího ROS. Tak se účastníkovi podařilo ovládat motor robota bez přístupového hesla.[11]

Další účastník pomocí znalosti IDL a s užitím programu WIRESHARK zachytávat zprávy předávané mezi uzly, tak mohl získávat citlivá data, či převzít kontrolu většiny funkcí systému. Náročnost útoku byla popsána jako triviální. [11]

Ve výchozím nastavení hlavní server ROS odpovídá na požadavky jakéhokoli zařízení v síti, které se k němu může připojit. Kterýkoli hostitel v síti může publikovat nebo odebírat topics, nebo upravovat parametry sítě. Pokud se neautorizovanému uživateli podaří připojit k ROS masteru, může získávat citlivé informace, jako jsou data ze senzorů nebo záznamy z kamer.[12]

Této zranitelnosti využil na přelomu 2017/2018 tým Brownovy univerzity, když členové týmu pomocí veřejně dostupného algoritmu prohledali internet třikrát během dvou měsíců na základním portu ROS, 11311. Při každém skenování našli více než 100 nezabezpečených ROS jader. [13]

Existují sice metody na obranu před těmito typy útoků, jako posílání dat pouze přes uzavřenou síť, případně firewall, ovšem z veřejně dostupných údajů vyplývá, že open-source projekty jsou zranitelné, útočník může totiž mít celkovou znalost systému z běžně dostupných zdrojů a využít tak zranitelností, kterým se nelze vyhnout bez kompromisů v oblasti funkčnosti, nebo uživatelského komfortu.

3.3 Přínosy IoT

IoT propojuje různá zařízení pomocí již existujících sítí, to umožňuje snadné a rychlé zavedení nových produktů, jednoduchou instalaci a relativně nízké náklady na pořízení a zprovoznění systémů. Díky těmto kvalitám IoT snadno proniká do všech oblastí běžného života od osobních zařízení až po infrastrukturu.

3.3.1 Osobní zařízení

Prominence IoT v sektoru osobních zařízení s každým rokem roste. Téměř každý dnes již vlastní smartphone od dětí až po lidi v rozvojových zemích a to jednotlivci umožňuje přístup k internetu, a díky tomu lze ovládat několik zařízení najednou s minimální náročností obsluhy.

3.3.1.1 Chytrá domácnost.

Aplikace „smart home“ (chytrá domácnost) se při budování moderního domu stává samozřejmostí. V novostavbách se zabudovává přímo do stavby, ale existuje i mnoho zařízení použitelných externě i ve starších domech. Umožňuje kontrolu nad domácností od ovládání světel, žaluzií a dalších tradičně manuálních zařízení, přes chytré domácí spotřebiče až po bezpečnostní systémy. Všechny tyto úkony lze samozřejmě plnit s využitím ROS.

S pomocí chytrého systému snadněji optimalizovat spotřebu komodit a tím snižovat emise oxidu uhličitého a zároveň náklady v době, kdy ceny za energie rostou, stejně jako environmentální uvědomění se toto stává hlavním lákadlem chytré domácnosti. [14]

V kombinaci s umělou inteligencí lze ovládat domácí pomocníky jako je například Alexa od firmy Amazon hlasovými příkazy, výhodou je, že není nutné hledat například dálkový ovladač, klávesnici či jiné ovládání, které může v místnosti být nepraktické až působit neesteticky.

3.3.1.2 Chytré doplňky

Moderním přístupem je integrovat elektroniku, do oblečení či doplňků. Například hodinky, brýle, ale i šperky se mohou zapojit do světa IoT, výhodou toho přístupu je to, že integruje technologii do každodenního života bez nutnosti nosit další externí zařízení.

Velká část těchto zařízení klade hlavní důraz na uživatelskou přívětivost, díky tomu se mohou do IoT zapojit i méně technicky zdatní lidé, například senioři. ROS v tomto oboru nemá velké uplatnění, většinou se jedná o jednoduché systémy, zavedení ROS by zbytečně zvyšovalo komplexitu systémů.

3.3.2 Zemědělství

Ve světě je odhadem 795 milionů lidí, kteří nemají dostatek potravin pro zdravý a udržitelný životní styl, do roku 2050 bude nutné zvýšit produkci potravin až o 70%, pokud chceme žít ve světě, kde lidé netrpí hladem. [15]

ROS systémy by mohli být přínosem pro takzvanou Data-Driven agrikulturu, díky možnosti shromažďovat velké množství dat. V případně precizní agrikultury je možné, aby ROS řídil zavlažovací systémy.

3.3.2.1 Důvod pro IoT v zemědělství

Tak proč je tedy nutné uvažovat o tom, jak téměř zdvojnásobit produkci potravin? V roce 2050 je odhadovaná populace země překročí hranici 10 miliard lidí, růst se začíná zpomalovat až stagnovat v rozvinuté části planety, ovšem s vývojem ekonomiky subsaharské části Afriky, se její populace odhadem během následujících 30 let zdvojnásobí. [16] Nárůst světové spotřeby jídla je nevyhnutelný. IoT v tomto případě, díky optimalizaci procesu, který zvyšuje výnos na jednotku plochy, to může být klíčové pro dosažení tohoto cíle bez odlesňování pro získání nové orné půdy.

3.3.2.2 Data-Driven Agriculture

Inteligentní systémy pro agrikulturu, nebo-li „Data-Driven Agriculture“ (dále jen DDA), byly poprvé navrženy jako řešení již v 80. letech minulého století, ale dodnes nejsou široce rozšířená, hlavním důvodem je vysoká pořizovací cena těchto systémů. [15]

Vysoká cena je zapříčiněna hlavně potřebou vytvořit vlastní síť, nebo využít satelit, případně mobilní síť na přenos dat a tím pádem je nutné zapojit další strany, které si účtují vlastní náklady plus profit. Dosah Wi-Fi sítí, užívaných většinou IoT zařízení k propojení s uživatelem, je neadekvátní.

Dalším faktorem je nutnost generovat vlastní elektrickou energii solárním panelem. Baterie jsou nevhodné pro jejich nízkou životnost, což znamená nutnost měnit je v pravidelných intervalech. [15] Kabely jsou též nevhodné, je téměř nemožné vyhnout se přerušení kabelu při obdělávání pole a zároveň by délka kabelů výrazně navýšila dobu konstrukce a pořizovací náklady.

3.3.2.3 DDA v kombinaci s umělou inteligencí

Řešením problému s cenou a složitou instalací systémů je kombinace IoT a umělé inteligence, prvním krokem je nákladné získávání dat. Jak již bylo řečeno získávat přesná data o vlhkosti půdy, pH, obsahu živin a dalších údajů vyžaduje velké množství nákladných

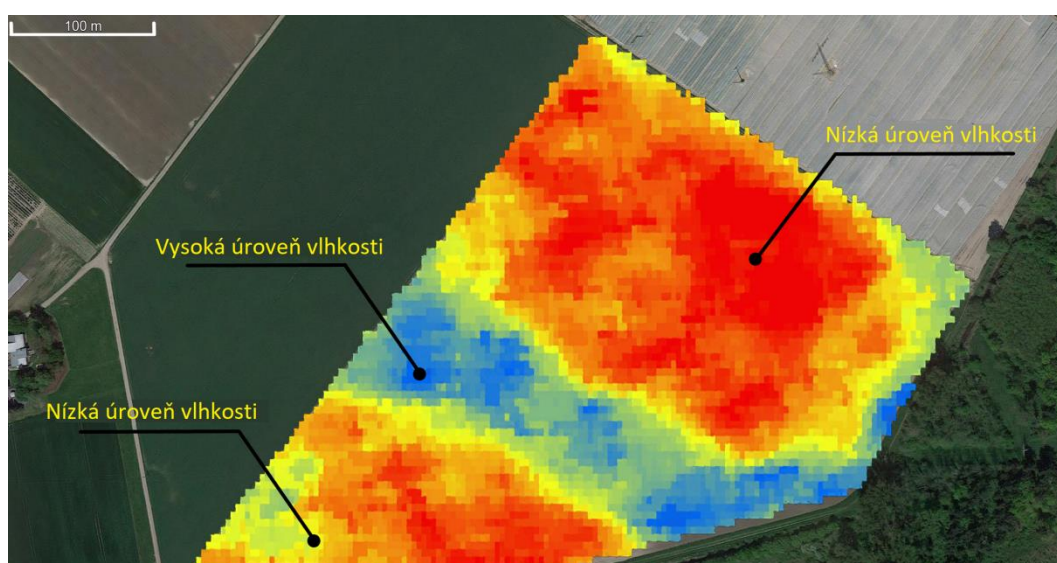
senzorů, ale díky využití umělé inteligence a dronů, které zároveň získávají fotky zájmové oblasti, lze neuronovou sít naučit odhadovat tyto proměnné s vysokou přesností, i po relativně malém počtu data setů, které sít neuronová sít zpracuje. [17]

Tento proces stačí udělat jednou, pro každou plodinu. Později lze dronem za méně než 1000 USD, s asistencí minimálního počtu senzorů na zemi, kontrolovat plochu několika kilometrů čtverečních. ROS v tomto případě může jak ovládat dron, tak vyhodnocovat výsledky a reagovat na ně.

3.3.2.4 Precizní Agrikultura

Široké zapojení technologie IoT do farmaření umožní v následujících letech přejít na model precizní agrikultury, umožňující zakročit na specifické části pole podle potřeby plodin v dané lokalitě. Například dnes se pole hnojí rovnoměrně, ale část hnojiva zůstane nevyužitá a skrze znečištění vody se dostává do přirozeného koloběhu života a tím zatěžuje celý ekosystém.

Využitím technologie umělé inteligence dokáže dron rozpoznat kde se plodinám daří méně, než je očekáváno, podle velikosti a hustoty vegetace, a zda se jedná o nedostatek vody (Obrázek 2), živin či jde o parazita pomocí barvy, tvaru rostlin a dalších znaků. [16]



Obrázek 2) Mapa vlhkosti pole vytvořená pro potřeby precizní agrikultury

3.3.3 Výroba

Výroba je prvním odvětvím, kde se aplikace IoT široce rozšířila. Optimalizace je zde klíčová při velkosériové výrobě, kdy se marže pohybuje v řádu procent. Kompetitivní prostředí snižuje ceny, a tak každé ušetřené procento může znamenat rozdíl mezi profitem a bankrotem.

3.3.3.1 Industry 4.0

Industriální IoT (dále jen IIoT) umožnil takzvanou čtvrtou industriální revoluci, zavedením chytré továrny do praxe. Chytrá továrna využívá moderní technologie ke zvýšení produktivity při snížení nákladů.

S pomocí senzorů a sítě, která propojuje všechny stroje výrobního procesu, dokáže systém pochopit dění v továrně a vytvořit virtuální simulaci procesu výroby, sledovat každý produkt od příjmu na sklad do expedice pomocí RFID nebo čárových kódů. Tyto simulace taktéž umožňují prediktivní údržbu zařízení, real-time změnu atributů výroby a statistické hodnocení výrobního procesu. To minimalizuje šanci na chyby ve výrobě a výrazně zvyšuje kvalitu produkce při snížení nákladu za vadné kusy.

IIoT zároveň účinně reguluje využití energie, pomocí senzorů sbírá data a vyhodnocuje efektivitu spotřeby. Dokáže vzdáleně ovládat osvětlení a další spotřebiče. Očekává se, že zařízení IoT budou integrována do všech forem zařízení spotřebovávajících energii (vypínače, zásuvky, žárovky, televizory a další) a budou schopna komunikovat s dodavatelskou společností, aby účinně vyvážila výrobu energie a spotřebu energie. [18]

3.3.3.2 ROS Industrial

Je rozšířením ROS pro industriální využití na automatizaci výroby a robotiku. Balíčky ROS Industrial obsahují ovládací rozhraní pro běžné průmyslové manipulátory, chapadla, senzory a sítě zařízení. Poskytuje také softwarové knihovny pro automatickou

kalibraci senzorů, plánování pohybů manipulačních paží a vývojové nástroje vytvořené specificky pro industriální prostředí. [19]

3.3.4 Výzkum

Díky celosvětové konektivitě pomocí internetu a ohromnému počtu senzorů je IoT ideální platforma pro shromažďování dat, následnou analýzou je možné zlepšovat procesy, kontrolovat kvalitu a vytvářet modely na základě výsledů měření.

IoT dnes hlavním zdrojem takzvaných Big Data, jedná se o ohromné množství dat z různých senzorů a čidel, využití ROS je jasné, exceluje totiž ve shromažďování dat. Často neuspořádaná data se analyzují pomocí algoritmů, nebo umělé inteligence, a umožňují tak vytváření relativně přesných předpovědí. S velikostí datasetu, při stejné kvalitě vstupu, roste přesnost.

3.3.4.1 Enviromentální monitoring

Aplikace IoT v tomto odvětví je především v zájmu ochrany obyvatelstva. Lze sledovat kvalitu vody a ovzduší, povodňovou aktivitu, ale i nebezpečné chemikálie v půdě. V případě ohrožení může ihned varovat obyvatelstvo a příslušné orgány o lokaci a závažnosti situace 24 hodin denně.

Dalším využitím je odhadování a sledování pohybu divoké zvěře, umožňuje efektivní studii a ochranu, bez narušení přirozeného koloběhu lidskou přítomností. Využití ROS je pro tuto aplikaci vhodné, je zdarma a umožňuje rychlou přestavbu prototypu v případě poškození zvěří či povětrnostními podmínkami.

3.3.5 Infrastruktura

Podobně jako ve výrobě i zde je IoT velmi rozšířeno. Monitorování a ovládání infrastruktury je dnes klíčové, zvláště pro metropolitní oblasti, kde je vysoká koncentrace

populace. ROS je vhodný pro tyto aplikace díky tomu, že může být distribuován na více zařízeních a monitorovat tak velkou plochu.

3.3.5.1 Smart city

Neboli chytré město je koncept zapojení celého města do systému IoT instalací mnoha různých senzorů napříč městem.

Jeden z těchto projektů probíhá ve španělském městě Santander, toto město s populací přibližně 180 000 zavedlo městskou aplikaci, která je připojena k více než 10 000 senzorům, které umožňují služby jako vyhledávání parkovacích míst, monitorování životního prostředí, počasí a dále. Naopak informace získané pomocí smartphonu pomáhají obchodníkům informovat zákazníky o výhodných nabídkách prostřednictvím algoritmu založeného na chování města, jehož cílem je maximalizovat dopad každého oznámení.[20]

Využití ROS v kontextu chytrého města nabízí několik pozitivních přínosů. Mezi ně patří například to, že ROS dokáže ovládat mnoho různých i starších typů zařízení. S modernizací měst přibývají nová zařízení IoT a zároveň je vyžadována zpětná kompatibilita.

3.3.5.2 Doprava

IoT v dopravě je jedním z nejlepších nástrojů pro snížení dopravního zatížení v hustě obydlených částech světa, jako jsou města a městské aglomerace. S masivním nárůstem počtu operujících aut jsou dopravní zácpy nevyhnutelné, ale pomocí senzorů a kamer může chytrý systém řídit dopravu optimálním způsobem.

ROS se využívá při vývoji prototypů aut s autopilotem, to umožní propojení a sdílení lokace cíle. Tak bude možné optimalizovat dopravu daleko efektivněji než s použitím kamer.

V případě hromadné dopravy jsou prvky IoT neocenitelným přínosem. Některé dnešní systémy dopravy jsou již plně automatizované, například Singapurský Mass Rapid Transit (dále jen MRT) je automatický vlakový systém, který přepravil v roce 2018 přibližně 3,3 milionu lidí denně.[21]

4. Funkční model z oblasti IoT

Jedním z cílů práce bylo zkonstruování IoT modelu chytré domácnosti a demonstrovat tak možnosti ROS v oblasti domácí automatizace. Model je založený na platformě Raspberry Pi. Obsahuje 2 druhy senzorů, digitální (DHT-22) pro měření teploty a vlhkosti a analogový (vlastní tvorba) pro regulaci osvětlení pomocí Pulse width modulation (dále jen PWM). Dále bude obsahovat aktivní prvky jako relé (vzdálené spuštění), krokový motor na ovládání žaluzií a LED diody simulující určité aplikace.

4.1 Použitý hardware

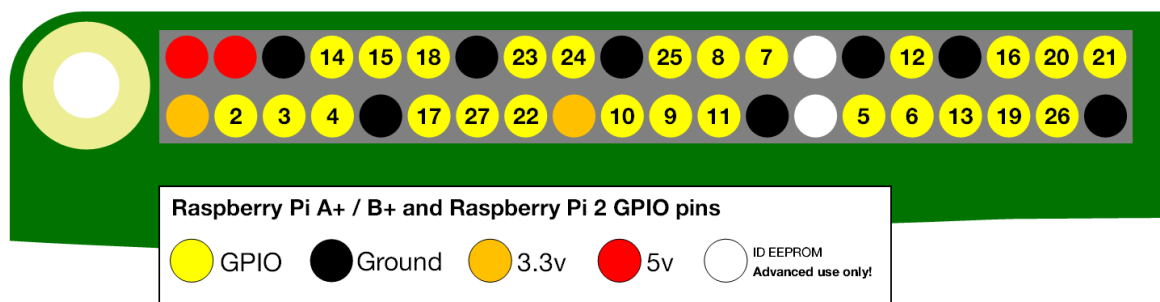
ROS má širokou podporu různého hardware, díky komunitním balíčkům je vhodný pro tvorbu prototypů ze všech odvětví IoT. Ovládací knihovny umožňují ovládání jak prvků pasivních pro sběr dat například senzory a kamery, tak pro akční členy jako motory a spínače.

Hardware použitý pro sestavení tohoto modelu je střední až nižší třídy. Jedná se o čínské výrobky a jejich kvalita je nestálá. V kontextu cílů této práce, kdy se jedná pouze o demonstrační model, nikoli zařízení pro využití v reálném světě, je kvalita dostačující. Pro průmyslovou aplikaci je třeba hardware vyšší kvality.

4.1.1 Raspberry Pi

Jedná se o jeden z nejrozšířenějších jednodeskových počítačů na bázi linuxu ve světě. Díky dostatečnému výkonu (4 jádrový ARM procesor a 1Gb RAM), relativně nízké ceně a široké dostupnosti je ideální pro amatérské aplikace ROS do světa IoT.

Díky GPIO (General Purpose Input/Output) rozhraní je Raspberry schopné komunikovat se širokým spektrem senzorů a akčních prvků. Na modelu 2 a vyšších se nachází 40 pinů (Obrázek3) z toho je 28 GPIO (GPIO 0-27), nedoporučuje se používat piny GPIO 0-1. Zbytek pinů buď plní funkci uzemnění, nebo napájení 3.3 a 5 voltů.



Obrázek 3) Rozložení GPIO pinů na Raspberry Pi

4.1.2 Senzor teploty a vlhkosti DHT-22

Pro měření teploty a vlhkosti byl zvolen senzor DHT22 pro jeho jednoduchost a nízkou cenu a zároveň poměrně vysokou přesnost měření. Při měření vzniká chyba, která se v tomto případě dá snadno zanedbat, jelikož průměrný člověk nepozná rozdíl půl stupně Celsia.

Maximální chyby deklarované výrobcem v datasheetu jsou pro relativní vlhkost $\pm 2\%$ a pro teplotu méně než ± 0.5 stupně Celsia. Důležité jsou i rozsahy měření, a to 0-100% pro relativní vlhkost a -40 až 80 stupně Celsia pro teplotu. [22]

Uvedené maximální chyby jsou na krajích rozsahů a v tomto případě, kdy je teplotní rozsah cca 15-30 stupňů Celsia jsou chyby výrazně nižší.

4.1.3 Relé SRD-5VDC-SL-C

Relé je nejjednodušší typ dálkového spínače, fungují na principu elektromagnetu, který po dosažení stanoveného napětí propojí obvod. Jedná se o on/off typ ovládání.

Technické specifikace SRD-5VDC-SL-C jsou vhodné pro většinu aplikací v domácnosti, napájení 5V DC, maximální spínané AC napětí je 250V a maximální spínaný AC výkon je 750VA.

4.1.4 Krokový motor 28BYJ-48 a drive ULN2003

Krokové motory umožňují přesné ovládání prvků reálného světa. Do kontextu této práce zapadá jako akční prvek soustavy řídicí žaluzie.

Krokový motor 28BYJ-48 v základním krokování potřebuje 4 kroky na jedno otočení s převodním poměrem 1 ku 64, to znamená že, při otočení o 360 stupňů udělá motor 256 (4*64) kroků, minimální prodleva mezi kroky je 2ms. Doporučený napájecí proud je 5V a maximální provozní proud je 500mA. [23]

Drive krokového motoru ULN2003 podporuje motor 28BYJ-48 a usnadňuje ovládání. To probíhá pomocí kombinace logických jedniček a nul na čtyřech řídicích GPIO pinech.

4.1.5 Vývojová deska Arduino CH340

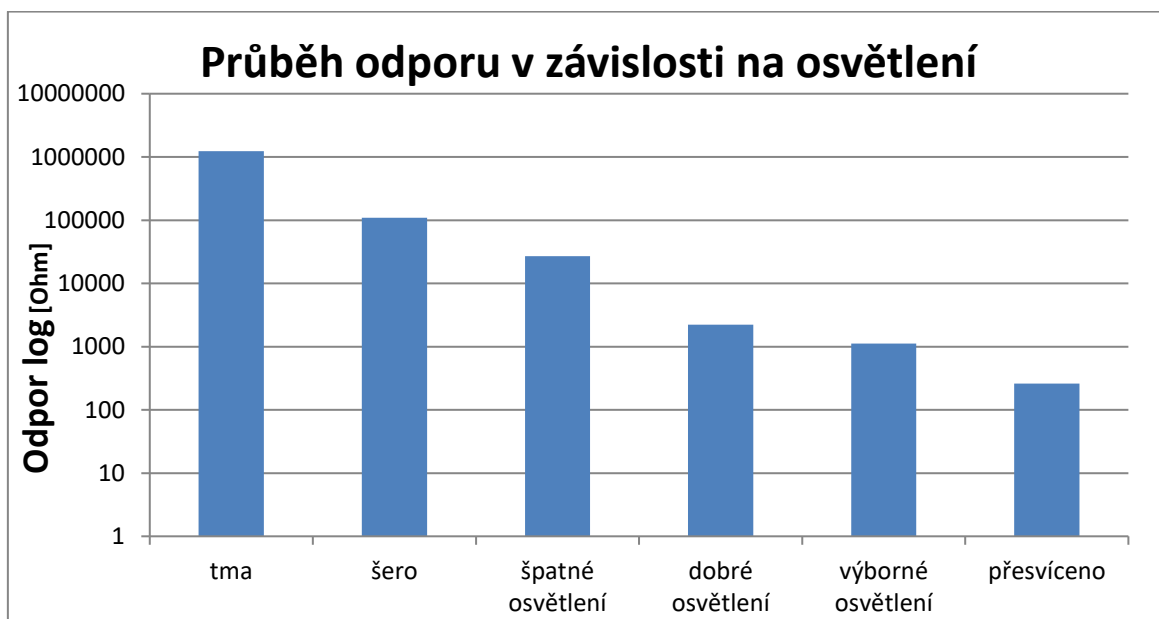
Platforma Arduino (CH340 je klon) založená na procesoru ATmega328p zapadá do kontextu této práce díky tomu, že obsahuje A/D převodník nutný pro zpracování dat z analogového senzoru. Jedná se o 10-ti bitový převodník, to znamená, že dostupný rozsah je 0-1023 (2^{10}). Pro analogovou hodnotu 5V získáme digitální hodnotu 1023, přesnost převodu je v tomto případě $\pm 0.00488V$ neboli necelých 5mV.

Použití samostatného A/D převodníku je preferovaný způsob. V tomto případě bylo dostupné Arduino zdarma a zároveň umožní demonstrovat jednoduchou komunikaci s ROS na Raspberry Pi pomocí GPIO pinů.

4.1.6 Vlastní senzor kvality osvětlení

Pro ovládání regulace osvětlení a ukázky operace analogového zařízení byl vytvořen jednoduchý senzor pro zhodnocení kvality osvětlení. Hlavním prvkem je fotorezistor PGM5516, který splňoval požadavky na změnu odporu v závislosti na osvětlení v místnosti.

Dále proběhlo měření odporů v závislosti na světelných podmínkách, naneštěstí v době produkce senzoru nebyl dostupný funkční luxmetr z toho důvodu bylo použito subjektivní měřítko. Pouze hodnota „výborné osvětlení“ byla naměřena ve výrobní hale splňující normy pro osvětlení pracoviště, podle ČSN EN 12464-1.



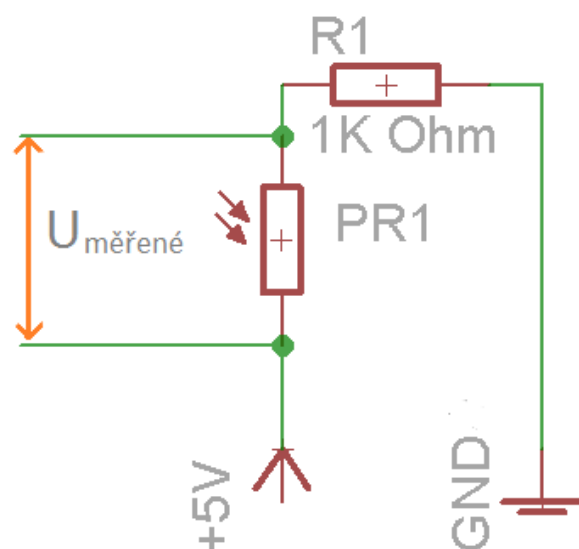
Graf 2) Závislost odporu PGM5516 na kvalitě osvětlení

Měřením byla určena hodnota, při výborném osvětlení má fotorezistor odpor 1020 Ohm. To umožnilo navrhnout velmi jednoduchý obvod, kde R1 je rovno 1K Ohm, díky čemuž PR1 spadá do tolerance $\pm 2\%$ (hodnota 1020 Ohm) pro dosažení výborného osvětlení.

Měření ukázalo, že reálná hodnota R1 je v tomto případě 1002 Ohm.

Senzor je v podstatě jednoduchý napěťový dělič (Graf2), který po zapojení do 5V zdroje udává analogově hodnoty mezi 0,12V a 4,93V. Kde hodnota 4,93V interpretuje naprostou tmou a 0,12V maximální osvětlení, kterého se podařilo dosáhnout.

Díky vhodně navrženému děliči udává ideální osvětlení hodnotu $\sim 2.46V$, což je přibližně jedna polovina maximálního výstupního napětí.



Obrázek 4) Schéma vlastního senzoru na základě fotorezistoru PGM5516

4.2 Konstrukce modelu

Konstrukce probíhala v několika etapách, podle dostupnosti součástí, doba doručení výrobků z Asie je nevyspíatelná. Na první pokus bylo použito Orange Pi Pc, čínský klon Raspberry Pi. Po problémech s kvalitou napájecího konektoru a kompatibilitou verze operačního systému (dále jen OS) vyžadované jako základ pro ROS, bylo zapůjčeno Raspberry Pi jihočeské univerzity jako základ pro tento model. Jako základní OS byla použita verze odlehčená verze Debian, pro Raspberry Pi známá jako Raspbian.

4.2.1 Instalace základního OS

Instalace OS pro Raspberry Pi probíhá na SD kartu, je vhodné zvolit kartu třídy 10 pro zvýšenou rychlost čtení a zápisu dat. Před instalací byla karta naformátována.

ROS podporuje Raspbian (Debian) jakožto jeden z OS, na které může být nástavbou. Instalace je nenáročná, po vyhledání vhodné verze je nutné balíček rozbalit. Ve složce, kam byl balíček rozbalen, se nachází takzvaný image operačního systému, ve formátu IMG. Pro instalaci na SD kartu je nutné použít externí program.

Win32 Disk Imager je freeware program pro Windows umožňující převádět image do spustitelné formy na SD kartě. Po nahrání lze Raspberry Pi připojit k síti a ověřit zda bootuje OS.

4.2.2 Instalace nastavy ROS

Do systému Raspberry Pi se ze stolního počítače připojuje pomocí Secure Shell protokolu (dále jen SSH), je nutné znát port a heslo k zařízení. Po připojení se zobrazí příkazový řádek. Užitý komunikační SSH software byl freeware MobaXterm.

Podrobný postup instalace ROS se nachází v dokumentaci pro ROS. [24] Ve zkratce instalace se provádí pomocí příkazové řádky, po aktualizaci OS je nutné zvolit požadovanou distribuci ROS, kinetic v tomto případě, a dále následovat sekvenci příkazů, která stáhne potřebné balíčky, knihovny a prostředí ROS.

Existuje několik možností instalace. Plná, obsahuje mnoho užitečných nástrojů pro vývoj, simulace robotů a má vlastní grafické prostředí. Odlehčená verze má pouze jádro a balíčky pro vytváření komunikace mezi uzly. Model je založen na jednodeskovém počítači a neobsahuje display, proto je odlehčená verze vhodnější.

4.2.3 Vytvoření balíčku v ROS

S pomocí Catkinu (oficiální systém pro budování balíčků pro ROS) byl, ve složce *pi/ros_catkin_ws/rsc*, vytvořen balíček „smarthome“ za použití příkazu *catkin_create_pkg*, který vyžaduje jméno a list knihoven potřebných pro daný balíček.

Ukázka kódu: *catkin_create_pkg smarthome rospy std_msgs*

Knihovna *rospy* umožňuje programování v pythonu, v případě programování v C++ se užívá *roscpp*. Knihovna *std_msgs* dává uzlům schopnost komunikovat pomocí messages standartních typů.

4.2.4 Regulace teploty v domácnosti

Regulace teploty je jedním z nejčastějších využití pro IoT v domácnosti. Systém regulace teploty založený na systému ROS spravuje jak chlazení, tak vytápění na základě dat ze senzoru DHT-22.

4.2.4.1 Publisher

V této aplikaci bude publisher zpravovat ovládání senzoru DHT-22 a interpretaci výsledků. Následně bude odesílat data pomocí messages, hodnoty jsou s desetinou čárkou, proto je nutné použít `Float64`.

Ukázka kódu pro import z knihovny: `from std_msgs.msg import Float64`

Definice publishera probíhá v rámci funkce `publisher()`. Definuje se jméno topic pro který publikuje (`dht22`), typ message (`Float64`) a počet měření uložených v bufferu (`queue_size`). Rate udává frekvenci publikování v Hertzech a zároveň frekvenci vzorkování senzorem.

Ukázka kódu:

```
def publisher():  
    pub = rospy.Publisher('dht22', Float64, queue_size=1)  
    rate = rospy.Rate(1)  
    msg_to_publish = Float64()
```

Další částí je `while` smyčka, která běží, dokud není provoz ROS ukončen. Ta obsahuje proměnou (`float64_to_publish`), do které se ukládají data ze senzoru (teplota). Messages typu `float64` mají pouze jeden typ pole a to data.

Ukázka kódu:

```
while not rospy.is_shutdown():  
    float64_to_publish = data_ze_senzoru
```

```

msg_to_publish.data = float64_to_publish
pub.publish(msg_to_publish)

rate.sleep()

```

V hlavní části pak stačí pouze inicializovat uzel, v tomto případě pouze s jednou proměnou a to jménem (*senzor_pub*). Zavoláním funkce `publisher()` proběhne inicializace uzlu. To lze ověřit v prostředí ROS pomocí příkazu: `roscat list`

Ukázka kódu:

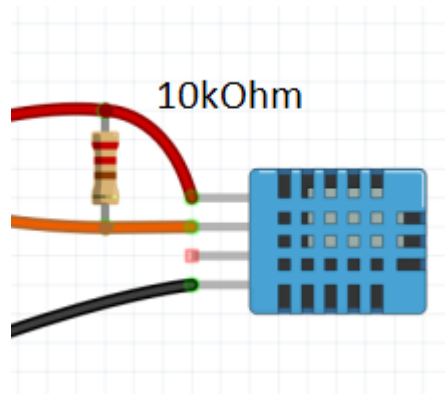
```

if __name__ == '__main__':
    rospy.init_node('senzor_pub')
    publisher()

```

4.2.4.2 Zapojení senzoru a zpracování dat

Číslo pinu	Barva	Zapojení
1	Červená	Vcc
2	Oranžová	DATA
3	-----	Null
4	Černá	GND



Tabulka 1) Zapojení pinů senzoru DHT22

Obrázek 4) Zapojení pull-up rezistoru mezi Vcc a DATA

Na senzoru DHT-22 se nachází piny, ale používají se pouze 3 a to Vcc (3,3-6V DC) neboli napájení, GND neboli země a Data pin, který je I/O. V případě nečinnosti má hodnotu logické jedničky díky 10kOhm pull-up rezistoru mezi Vcc a Data pinem. Signál, který udává, kdy započít měření je 1ms logické nuly, po přijetí odešle senzor zpátky potvrzení o provedení měření a zahájí se přenos dat.

Díky rozmanitosti knihoven pro python existuje jednoduchá knihovna od Adafruit pro interpretaci přijatých dat. Nachází se na open-source platformě Github , knihovnu lze stáhnout pomocí příkazu: `git clone https://github.com/adafruit/Adafruit_Python_DHT.git`

Pro získání hodnot je nutné importovat knihovnu, definovat data port a specifický typ senzoru. Tak později umožní snadnou aplikaci. Hodnoty se dají získat pomocí jednoho řádku kódu v hlavní části uzlu.

Ukázka kódu:

```
import Adafruit_DHT

DHT_SENSOR = Adafruit_DHT.DHT22

DHT_PIN = 4

def ziskat_data():

    vlhkost, teplota = Adafruit_DHT.read_retry(DHT_SENSOR, DHT_PIN)

    return teplota
```

4.2.4.3 Subscriber

Struktura všech uzlů je víceméně podobná i pro subscriber byla vytvořena funkce. Definuje topic, který odebírá (*dht22*), typ message (*Float64*), ale tentokrát se jako třetí parametr uvádí funkce (*callback_f*), pomocí které lze extrahovat data z message.

Ukázka kódu:

```
def subscriber():

    sub = rospy.Subscriber('dht22', Float32, callback_f)

    rospy.spin()
```

Zároveň se ve funkci (*callback_f*) nachází ovládání dvou LED diod, pro simulaci výhřevu a chlazení domácnosti. Modrá značí aktivaci chlazení, červená výhřevu v případě,

že se teplota dostane mimo určené meze. Signál, který spouští diodu, je vhodný například na spínání relé, MOSFET a dalších.

Ukázka kódu:

```
def callback_f(message):  
teplota = message.data  
if teplota > max_t:  
    GPIO.output(18, GPIO.HIGH)  
    GPIO.output(17, GPIO.LOW)  
elif teplota < min_t:  
    GPIO.output(17, GPIO.HIGH)  
    GPIO.output(18, GPIO.LOW)  
else:  
    GPIO.output(18, GPIO.LOW)  
    GPIO.output(17, GPIO.LOW)
```

Poslední částí je inicializace uzlu (*senzor_sub*) a zavolání funkce definující topic, který chce odebírat.

Ukázka kódu:

```
if __name__ == '__main__':  
    rospy.init_node('senzor_sub')  
    subscriber()
```

4.2.4.4 Rqt graf

Pomocí příkazu *roslun rqt_graph rqt_graph* lze vyvolat graf (Obrázek 5), který schematicky znázorňuje propojení uzlů pomocí topics. Je ideální pro potřeby vývoje, ve složitém systému umožňuje rychlé vyhledávání způsobu propojení uzlů.



Obrázek 5) Rqt graf ukazuje spojení mezi publisherem a subscriberem pomocí topicu

4.2.5 Ovládání a regulace osvětlení

Kontrola osvětlení probíhá pomocí uzlu, který je zároveň publisher (*regulator*). Ovládání mikroprocesoru a relé probíhá přímo v uzlu. Pro ovládání úhlu krokového motoru existuje subscriber (*motor*), který pomocí topicu (*krok*) obdrží message formátu Integer. Tato hodnota udává počet kroků po směru hodinových ručiček, při zadání záporné hodnoty se motor otáčí proti směru.

Propojení mikroprocesoru ATmega328p s platformou ROS probíhá pomocí jednoduché on/off logiky (bool). Mikroprocesor sleduje hodnoty ze senzoru kvality osvětlení. V případě, že obdrží signál logické jedničky na určený pin, spustí se systém osvětlení s regulací.

4.2.5.1 Ovládání relé

Ovládání relé je triviální záležitostí, má pouze 3 piny napájení, uzemnění a vstup. Relé se skládá z cívky navinuté na jádru feromagnetického materiálu, přivedení logické jedničky na vstup sepne elektromagnet, ten propojí obvod.

Ovládací program je také velmi jednoduchý, GPIO pin zvolený jako vstup do relé je nastaven jako výstupní. Pro aktivaci relé se GPIO nastaví na *HIGH*, pro deaktivaci na *LOW*. To lze provádět voláním jednoduché funkce.

Ukázka kódu:

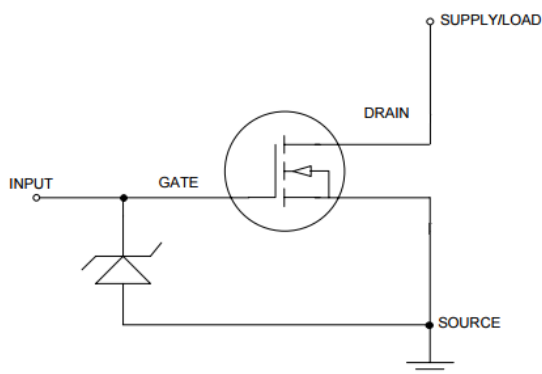
```
def rele_off():  
    GPIO.output(22, GPIO.LOW)  
def rele_on():  
    GPIO.output(22, GPIO.HIGH)
```

4.2.5.2 Výroba plošného spoje

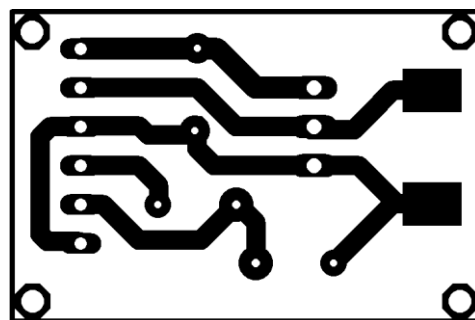
Pro potřeby regulace osvětlení byl vytvořen plošný spoj (dále jen PCB). Obsahuje senzor kvality osvětlení stejně jako spínací prvek MOSFET (Metal Oxide Semiconductor Field Effect Transistor). Poslední částí je ochranná Zenerova dioda proti přepětí. Ta se nachází mezi piny MOSFETu, specificky Gate a Source. (Obrázek 6)

Výroba začíná návrhem, existuje několik vhodných PCB programů. Zvolen byl jeden z nejpoužívanějších Autodesk EAGLE. Obsahuje mnoho nástrojů, které umožňují efektivní návrh PCB.

Návrh vzoru (Obrázek 7) byl přenesen na základ PCB desky pomocí papíru pro přenos toneru, zahřátím na teplotu ~180 stupňů Celsius po dobu jedné minuty. Reakce v nádrži s chloridem železitým rozpustila přebytečnou měď. Po odstranění toneru, který s chloridem nereaguje, zůstane na PCB požadovaný vzor. Posledním krokem bylo vyvrtání potřebných otvorů a připájení součástek.



Obrázek 6) Zapojení MOSFET



Obrázek 7) Návrh pcb

4.2.5.3 PWM

Pwm modulaci obstarává ATmega328p v obsahuje 3 timery. Ty lze využít k mnoha aplikacím s vysokou náročností na přesnost v oblasti času, například PWM.

Zvolen byl 8-bitový Timer2, pomocí kombinace pre-scaleru a předplnění lze dosáhnout přesně požadované frekvence modulace, pro toto užití 20KHz. Tato frekvence

je mimo oblast lidského slyšení. Změnou bodu, ve kterém dojde k overflow, lze regulovat dobu, po kterou je přítomno napětí a tím regulovat výkon.

PWM signál mikroprocesoru nemá dostatečný výkon na napájení osvětlení, proto je nutné jej přivést na spínací prvek (MOSFET), který obstará spínání vyššího napětí a proudu.

Programování tohoto mikroprocesoru není cílem této práce, proto se ukázka kódu nachází v přílohách (příloha 1).

4.2.5.4 Krokový motor

Krokový motor se používá pro precizní polohování, v tomto případě žaluzií. Ovládán je kombinací hodnot low a high ze 4 GPIO pinů, které zpracovává ovladač ULN2003 do impulsy dané délky ve správném pořadí. Užitím půl krokování lze zdvojnásobit počet kroků na jednu otáčku alternací mezi nabuzením jedné a dvou cívek motoru a zdvojnásobit tak rozlišení.

Program nastaví všechny 4 piny na výstup, dále je nutné definovat sekvenci, která udává pořadí impulsu z pinů.

Ukázka kódu:

```
ridici_pin = [7,11,13,15]
for pin in ridici_pin:
    GPIO.setup(pin, GPIO.OUT)
    GPIO.output(pin, 0)
sekvence = [ [1,0,0,0],
              [1,1,0,0],
              [0,1,0,0],
              [0,1,1,0],
              [0,0,1,0],
              [0,0,1,1],
              [0,0,0,1],
```

[1,0,0,1]

Sekvence má 8 členů a každý z nich má 4 proměnné. Pro převedení do formy, které ovladač motoru rozumí, je vhodná smyčka for. Ta umožní iteraci sekvence a zároveň ovládá spínání GPIO pinů.

Ukázka kódu:

for i in range (pocet_kroku):

for krok in range (8):

for pin in range (4):

GPIO.output (ControlPin[pin], sekvence[krok][pin])

time.sleep (0.002)

5. Diskuze

ROS se ukázal jako efektivní řešení mnoha problémů v robotice a IoT. Modulárnost umožňuje využívání již existujícího kódu a počet komunitních balíčků roste každým dnem. Tak se rozrůstá i spektrum aplikace ROS ve světě IoT. Po prostudování tématu IoT a sestrojení IoT modelu je jasné, že ROS má i své nevýhody.

Následuje hodnocení užitečnosti IoT v několika oborech kritických pro další rozvoj lidstva a možnosti, kterými se může ROS zapojit do tohoto procesu.

5.1 Hodnocení aplikace ROS napříč obory IoT

V případě osobních zařízení je IoT nepopiratelným přínosem. Smartphone umožňuje lidem získat jakoukoli volně dostupnou informaci v řádu sekund, určení polohy s přesností na několik metrů a celkově se zapojit do světa IoT. Takzvané chytré doplňky mají často

senzory, které měří funkce těla Toho jde využít u aplikací například ve zdravotnictví. ROS je ovšem v tomto případě zbytečně komplikovaný a vyžaduje jako základ linux. Mnoho z těchto doplňků obsahuje pouze jednoduchý mikroprocesor naprogramovaný v assembleru. U osobních zařízení zároveň vzniká nebezpečí úniku dat. (viz. Kapitola 3.2)

V případě hromadné dopravy IoT výrazně zvyšuje bezpečnost a kvalitu služeb. V dopravním sektoru umožňuje například sledování zásilky. Tak vznikají nové příležitosti jako je zavedení systému Just-In-Time, který eliminuje potřebu skladování ve výrobě. Studie uvádí, že použitím takzvaných chytrých semaforů v kombinaci s městskou sítí kamer a senzorů lze zredukovat čas strávený v zácpách až o 66% a celkovou dobu cesty až o 67% v ideálním případě. [25] ROS lze na tyto aplikace použít, ale existuje mnoho již existujících systémů vytvořených přímo pro specifickou aplikaci v dopravě. Tak se snižuje náročnost obsluhy a zvyšuje efektivita a bezpečnost.

Díky možnosti vyrábět dnes senzory ve velkém množství za nízkou cenu se otevírají nové možnosti zlepšování modelů predikce přírodních katastrof, jako tsunami a zemětřesení. Široké zavedení IoT do této oblasti bude revoluční. V kombinaci s prediktivní schopností umělé inteligence může takovýto systém zachránit mnoho lidí. ROS je ideální pro aplikace jako je tato, jedná se o nový koncept, takže neexistují řešení na míru. Zároveň je potřeba velké množství dat a ROS je na to ideální.

Aplikace IoT do průmyslu během posledních let značně zvýšila produktivitu. Pro velkovýrobu není ROS příliš vhodné jako hlavní systém. Dnes existují takzvané modulární linky, do kterých lze vložit nový modul a spustit funkci během několika hodin, díky kompatibilitě jak software, tak hardware. Odpadá tak nutnost vývoje a přesně definovaný postup instalace velmi snižuje nároky na obsluhu. ROS Industrial umožňuje vývoj systému jako je tento, ale opravdovým přínosem je využití automatizace v dříve cenově nevýhodných aplikacích. To rozšiřuje možnosti automatizace i na malosériovou produkci. Rychlé vytváření prototypů je jednou z největších výhod ROS v průmyslu.

Zvyšování produkce jídla bude nutné (viz. Kapitola 3.3.2.1), ROS by mohlo být řešením toho problému. Dokáže získávat data pro DDA (viz. Kapitola 3.3.2.2) a aplikovat výsledky s pomocí precizní agrikultury (viz. Kapitola 3.3.2.4). Díky tomu může aplikovat specifické dávky hnojiv a vody na různé části pole. Takto se dá výrazně snížit zatížení ekosystému, nejen snížením užívaných hnojiv a pesticidů, ale zároveň snížením spotřeby vody při zvyšování zisku na jednotku plochy. Studie ukazují, že využitím systému jako je

tento, lze zvýšit produkci až o 45% při snížení nároků na vodní zdroj o 35,7% v suchém klimatu. [26] Dnes jeden z nejefektivnějších systémů FarmBeats, využívá pro přenos dat televizní signál. Takzvané bílé (volné) kanály, které jsou v rurálních oblastech často dostupné. Signál je nízkofrekvenční, díky tomu snáze penetruje skrz porost a má několika násobně větší dosah než Wi-Fi. ROS je zároveň velmi vhodný pro využití v rozvojové části světa. Je zdarma, to znamená, že vzdělaný člověk může vytvořit vlastní systém jen za cenu hardware a úsilí.

5.2 Hodnocení ROS v chytré domácnosti

Práce na projektu ukázala, že ROS je proto toto užití výborným nástrojem, při dodržení určitých bezpečnostních opatření. Dokáže řídit domácí systémy, optimalizovat spotřebu a další.

Na otázku kolik energie lze ušetřit není jednoduchá odpověď, ale většina studií se shoduje, že je možné ušetřit 5-20% na účtech za energii. [27] Optimalizace probíhá s pomocí automatických světel, užívání nočního proudu na ohřev vody a adaptivního vytápění. S chladnějším počasím roste efektivita, protože vytápění zužitkuje průměrně 43% spotřebované energie, klimatizace pouze 8%. [28]

Z hlediska zvýšení kvality života je chytrá domácnost také přínosem. Původní plán byl připojit systém ovládání žaluzií k budíku a umožnit mu tak roztáhnou žaluzie v požadovaný čas před spuštěním, což umožní příjemnější probuzení. Dálková aktivace a ovládání je taktéž výhodou. Například spuštění vytápění podle pozice z GPS při přiblížení se k domu na určitou vzdálenost může být vítanou možností v zimních měsících. Automatizace domácích spotřebičů umožňuje například, aby pračka informovala uživatele o dokončení programu praní a tak dále.

Pro hodnocení aplikace vytvořeného modelu chytré domácnosti v reálném světě je nutné zvážit poměr cena/výkon. Pro vytvoření modelu byly zvoleny nejlevnější dostupné součástky (viz. Kapitola 4.1). To se negativně odráží na aplikovatelnosti v reálném světě snížením spolehlivosti a zvýšením uživatelské náročnosti. Zároveň nebyla provedena optimalizace a tak není dosaženo plného potenciálu systému. Model ovšem plní požadovanou funkci a tak se dá říci, že aplikován být může.

Pro zapojení ROS do domácí sítě je nutné zvážit bezpečnostní rizika (viz. Kapitola 3.2). Zde je list opatření doporučených pro základní zabezpečení ROS:

- 1) Změna všech základních hesel.
- 2) Zablokovat port 11311 pomocí firewallu.
- 3) Pro propojení routeru a ROS jádra neužívat WI-FI, ale ethernet kabel.

6. Závěr

ROS a internet věci celkově nám umožnili posunout hranice efektivity procesů a mají potenciál v tomto trendu pokračovat do budoucna. Velké množství dat nashromážděné za několik posledních let nám umožní vyvíjet systémy založené na umělé inteligenci, v kombinaci s roboty, kteří mohou operovat ROS, bude možné automatizovat i méně repetitivní úkony na takové úrovni, že nebude nutný lidský dohled a celosvětová produktivita extrémně naroste. Je možné, že jednoho dne až na hranici, kdy bude manuální práce minulostí, ale samozřejmě to zabere čas.

Čas, který někteří lidé na této planetě nemusí mít, pokud se rozvoj lidstva značně nezpomalí, bude nutné znatelně zvýšit produkci jídla, pro aplikaci v agrikultuře je ROS užitečným nástrojem. S pomocí systémů precizní agrikultury se daří výrazně zvyšovat výnos na jednotku plochy a tak bude možná nakrmit rostoucí populaci bez potřeby odlesňování.

Environmentální uvědomění roste s každým dnem a systémy optimalizace od domácnosti, přes dopravu až energetické sítě mohou snížit emise v řádu procent a nejenom bránit globálnímu oteplování, ale zároveň redukovat náklady na energie. Tyto systémy stejně jako popsany model mohou fungovat na bázi ROS.

Do budoucna bude potřeba mnoho vzdělaných lidí, kteří díky svému zápalu pro technologii umožní, aby se toto stalo realitou.

Reference:

- 1) Quigley M., Gerkey B., Conley K., Faust J., Foote T., Leibs J., Berger E., Wheeler R., Andrew N. (2009) „ROS: an open-source Robot Operating System“
- 2) ROS dokumentace část „Introduction“ <http://wiki.ros.org/ROS/Introduction>
- 3) ROS dokumentace část „ROS Concepts“ <http://wiki.ros.org/ROS/Concepts>
- 4) Ashton Kevin (červen 2009) „That „Internet of Things“ Thing.“ Vydání „RFID Journal“
- 5) Lasse Lueth Knud (srpen 2018) „State of the IoT 2018: Number of IoT devices now at 7B – Market accelerating“
- 6) Nordrum Amy (květen 2016). "Popular Internet of Things Forecast of 50 Billion Devices by 2020 Is Outdated"
- 7) Haller S., Karnouskos S. and Schroth C. (září 2008) "The Internet of Things in an Enterprise Context," Vydání „Future Internet“
- 8) Hughes Tony (červenec 2017) „The Effect of Ride-Sharing on the Auto Industry“ Vydání „Moody's analytics risk perspectives volume IX“
- 9) Vaughan R.T., Gerkey B.P. (2007) „Reusable robot code and the Player/Stage Project“ Vydání „Software Engineering for Experimental Robotics“
- 10) ROS dokumentace část „Is ROS For Me?“ <https://www.ros.org/is-ros-for-me/>
- 11) McCleana Jarrod, Stullb Christopher, Farrarc Charles, Mascarenas David (2012) „A Preliminary Cyber-Physical Security Assessment of the Robot Operating System“
- 12) ROS dokumentace část „Security“ <http://wiki.ros.org/Security>
- 13) DeMarinis N., Tellex S., Kemerlis V., Konidaris G., Fonseca R. (červenec 2018) „Scanning the Internet for ROS: A View of Security in Robotics Research“
- 14) Bhati Abhishek, Hansen Michael, Chan Ching Man (2016) „Energy conservation through smart homes in a smart city: A lesson for Singapore households“
- 15) Chandra Ranveer, (červenec 2018), „ How data-driven farming could transform agriculture“ přednáška „TEDx University of Rochester“
- 16) Porubcansky Mark, (červen 2015), „795 million people don't have enough to eat — why that's actually good news“
- 17) Kapetanovic Zerina, Chandra Ranveer, Chakraborty Tusher , Nelson Andrew (červenec 2019) „FarmBeats: Improving Farm Productivity Using Data-Driven Agriculture“
- 18) Parello J. Claise B. Schoening B. Quittek J. (září 2014) „Energy Management Framework“
- 19) ROS Industrial dokumentace část „Description“ <https://rosindustrial.org/about/description>
- 20) Rico Juan (duben 2014) „Going beyond monitoring and actuating in large scale smart cities". Přednáška NFC & Proximity Solutions – WIMA Monaco.
- 21) Christopher Tan (únor 2019) „Bus, train trips hit record high last year“ Vydání „Singapore Straits Times“
- 22) Datasheet senzoru DHT-22
<https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf>
- 23) Datasheet pro motor 28BYJ-48 a drive ULN2003
<http://eeshop.unl.edu/pdf/Stepper+Driver.pdf>

- 24) ROS dokumentace část „Instalation on Raspberry Pi
<http://wiki.ros.org/ROSBerryPi/Installing%20ROS%20Kinetic%20on%20the%20Raspberry%20Pi>
- 25) Chong Hon Fong, Kiat Ng Danny Wee (leden 2017) „Development of IoT device for traffic management system“
- 26) Almarshadi M. H., Ismail S. M., (2011) „Effects of precision irrigation on productivity and water use efficiency under different irrigation methods in arid climates“
- 27) Hart Jane K., Martinez Kirk (květen 2015) „Toward an environmental Internet of Things" Vydání „Earth & Space Science“
- 28) U.S. Energy Information Administration (2015) „2015 Residential Energy Consumption Survey“

Obrázky:

Obrázek 1) Obrázek by vytvořen s pomocí programu Microsoft Paint.

Obrázek 2) Obrázek je vlastnictvím Spatial Business Integration. (http://www.spatial-business-integration.com/wp-content/uploads/2017/05/SBI_SARfields_Soil-moisture-zones_TSX_161006.jpg)

Obrázek 3) Obrázek je součástí dokumentace pro Raspberry Pi část GPIO. (<https://www.raspberrypi.org/documentation/usage/gpio/>)

Obrázek 4) Obrázek je součástí dokumentace pro senzor DHT-22. (<https://arduino-shop.cz/docs/produkty/0/56/1455993178.pdf>)

Obrázek 5) Obrázek je *rqt_graph* vytvořeného systému uzlů.

Obrázek 6) Obrázek je upravená verze z datasheetu pro MOSFET https://www.semtech.com/uploads/documents/transient_voltage_protection_of_mosfets.pdf

Obrázek 7) Obrázek vytvořen s pomocí programu Autodesk EAGLE

Grafy:

Graf 1) Graf a data na něm uvedená jsou produktem IoT Analytics. (<https://iot-analytics.com/wp/wp-content/uploads/2018/08/Number-of-global-device-connections-2015-2025-Number-of-IoT-Devices.png>)

Graf 2) Naměřená data jsou vlastní, graf byl vytvořen s pomocí programu Microsoft Excel.

Tabulky:

Tabulka 1) Tabulka byla vytvořena s pomocí programu Microsoft Excel.

Příloha 1: Program pro tvorbu PWM signálu na ATmega328p

```
#include <avr/interrupt.h>
ISR(TIMER2_OVF_vect)
{
  TCNT2 = 155;
  PORTB = B00000000;
}
ISR(TIMER2_COMPA_vect)
{
  PORTB = B00000010;
}
void setup()
{
  pinMode(9, OUTPUT);
  noInterrupts();
  CLKPR=0x80;
  CLKPR=0x00;
  ASSR=0x00;
  TCCR2A=0x00;
  TCCR2B=0x02;
  TCNT2=0x00; //predplneni
  TIMSK0=0x00; //timer1 vypnout interrupty
  TIMSK1=0x00; //timer2 -//-
  TIMSK2=0x03; // timer3 8bit zapnout overflow a compA
  interrupts();
}
void loop()
{
  num = 450-analogRead(A1);
  Serial.println(num);
  if (num>0)
  {
    OCR2A = 255;
  }
  else if (num<-500)
  {
    OCR2A = 155;
  }
  else
  {
    OCR2A = 155+(500+num)/5;
  }
}
```