

**EKONOMICKÁ UNIVERZITA V BRATISLAVE
FAKULTA HOSPODÁRSKEJ INFORMATIKY**

Evidenčné číslo: 103004/I/2021/36103157977991172

**VYTVORENIE IoT SIETE Z MODULOV ARDUINO A
RASPBERRY**

Diplomová práca

**EKONOMICKÁ UNIVERZITA V BRATISLAVE
FAKULTA HOSPODÁRSKEJ INFORMATIKY**

**VYTVORENIE IoT SIETE Z MODULOV ARDUINO A
RASPBERRY**

Diplomová práca

Študijný program:	Informačný manažment
Študijný odbor:	Ekonómia a manažment
Školiace pracovisko:	Katedra aplikovanej informatiky
Vedúci záverečnej práce:	Ing. Mgr. Peter Schmidt, PhD.



Ekonomická univerzita v Bratislave
Fakulta hospodárskej informatiky

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Marek Sekan
Študijný program: informačný manažment (Jednoodborové štúdium, inžiniersky II. st., denná forma)
Študijný odbor: ekonómia a manažment
Typ záverečnej práce: Inžinierska záverečná práca
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: Vytvorenie IoT siete z modulov Arduino a Raspberry
Anotácia: V DP autor vytvorí automatizovaný systém správy mikroklimy skleníka, kde bude sledovať potrebné parametre pomocou snímačov. Namerané hodnoty vyhodnotí riadiaca jednotka, ktorá bude reagovať príslušnými opatreniami.
Vyžadujú sa znalosti z HW, OS, DSD a programovania

Vedúci: Ing. Mgr. Peter Schmidt, PhD.
Katedra: KAI FHI - Katedra aplikovanej informatiky FHI
Vedúci katedry: Ing. Mgr. Peter Schmidt, PhD.

Dátum zadania: 28.10.2019

Dátum schválenia: 28.10.2019

Ing. Mgr. Peter Schmidt, PhD.
vedúci katedry

Čestné vyhlásenie

Čestne vyhlasujem, že diplomovú prácu som vypracoval samostatne s použitím iba uvedenej odbornej literatúry, ktorá sa nachádza v zozname použitej literatúry.

V Bratislave dňa 4.5.2021

.....

Podpis študenta

PodĎakovanie

Rád by som sa touto cestou chcel poďakovať svojmu školiťovi, pánovi Ing. Mgr. Petrovi Schmidtovi, PhD. za jeho ústretový prístup, odborné rady, vecné pripomienky, podporu a trpezlivosť pri vypracovávaní tejto práce a návrhu a realizácii zariadenia.

ABSTRAKT

SEKAN, Marek: *Vytvorenie IoT siete z modulov Arduino a Raspberry* – Ekonomická univerzita v Bratislave. Fakulta hospodárskej informatiky; Katedra aplikovanej informatiky. – Vedúci záverečnej práce: Ing. Mgr. Peter Schmidt, PhD. – Bratislava: FHI EU, 2021, 65 s.

Cieľom záverečnej práce je prakticky využiť dve najznámejšie platformy pre Internet vecí. Práca je rozdelená do piatich kapitol a obsahuje 25 obrázkov. Prvá kapitola sa venuje charakteristike pojmov, ktoré súvisia s Internetom vecí a sú dôležité na pochopenie tohto pojmu. Ďalej sa v tejto kapitole definujú siete, ktoré sa používajú pri IoT. Následne sú v prvej kapitole predstavené aj platformy, vhodné na vývoj IoT zariadení. V druhej kapitole je uvedený cieľ práce. V rámci tretej kapitoly sú predstavené zvolené platformy, snímače, prvky a komunikačné technológie, v závere kapitola obsahuje schému zapojenia. Štvrtá kapitola je zameraná na zostavenie projektu, zapojenie modulov, ich programovanie či testovanie zariadenia, na záver je v kapitole ukážka hotoveného zariadenia. Finálna kapitola obsahuje diskusiu o možných návrhoch na vylepšenie zariadenia.

Kľúčové slová: Raspberry Pi, Arduino, Internet vecí, návrh, analýza, realizácia, snímač

ABSTRACT

SEKAN, Marek: *Creating IoT network from Arduino and Raspberry modules*. – University of Economics in Bratislava. Faculty of Economic informatics; Department of Applied Informatics. – Consultant: Ing. Mgr. Peter Schmidt, PhD. – Bratislava: FHI EU, 2021, 65 s.

The aim of the final thesis is to show how to use two of the most interesting platforms for the Internet of Things in practice. The thesis is divided into 5 chapters and includes 25 pictures. The first chapter is dedicated to the characteristics and related concepts, which are important for understanding the term Internet of Things. Next are defined networks which are used for IoT and introduced platforms suitable for development of IoT devices. In the second chapter there is the aim of the thesis. In the third chapter there are introduced chosen platforms, sensors, elements and communication technologies, at the end of the chapter there are introduced a circuit diagram. The fourth chapter is about building the project, connection of modules, programming and testing the device, at the end of the chapter there is an example of final functional device. In the final chapter there is a discussion about different improvement of the device.

Key words: Raspberry Pi, Arduino, Internet of Things, design, analysis, implementation, sensor

Obsah

Zoznam obrázkov	10
Zoznam skratiek.....	11
Úvod.....	12
1 Súčasný stav riešenej problematiky	13
1.1 Charakteristika pojmov súvisiacich s IoT	13
1.2 Internet of Things	15
1.3 5G sieť	19
1.4 WiFi sieť	21
1.5 Technológia Bluetooth	21
1.6 Siete používané pre IoT	22
1.6.1 Zigbee	22
1.6.2 Thread	23
1.6.3 Z-Wave	23
1.6.4 NarrowBand-IoT	24
1.6.5 SigFox	24
1.6.6 LoRaWAN	24
1.7 Platforma Raspberry Pi	25
1.7.1 Komponenty na doske Raspberry Pi	25
1.7.2 GPIO	27
1.7.3 Sense HAT modul	28
1.7.4 Raspberry Pi 4 Model B – najnovší model	29
1.7.5 Podobné platformy	30
1.8 Platforma Arduino	31
1.8.1 Typy Arduino dosiek	32
1.8.2 Sériová komunikácia	33

1.8.3	GPIO piny	34
1.8.4	Shieldy	34
1.9	Programovací jazyk Python	36
2	Cieľ práce.....	38
3	Metodika práce	39
3.1	Voľba modulov	39
3.2	Voľba snímačov	41
3.2.1	PIR snímač Hc-Sr501	41
3.2.2	Snímač DHT11	42
3.2.3	Snímač DHT22	42
3.3	Výber ďalších prvkov.....	43
3.4	Voľba komunikačnej technológie	43
3.5	Schéma zapojenia.....	44
4	Výsledky práce	46
4.1	Zostavenie projektu.....	46
4.1.1	Zapojenie časti s modulom Raspberry Pi	47
4.1.2	Zapojenie časti s modulom Arduino	48
4.2	Programovanie modulu Raspberry Pi	49
4.3	Programovanie modulu Arduino.....	54
4.4	Prepojenie modulov Arduino a Raspberry Pi.....	57
4.5	Testovanie zariadenia.....	58
4.6	Ukážka zhotoveného zariadenia.....	59
5	Diskusia	62
	Záver	63
	Zoznam použitej literatúry	64

Zoznam obrázkov

Obr. 1: Porovnanie Machine-to-Machine a IoT.....	16
Obr. 2: Predpokladaný vývoj pripojených IoT zariadení	18
Obr. 3: Kamerový modul pre Raspberry Pi	26
Obr. 4: Vstupné a výstupné hroty GPIO – schéma.....	27
Obr. 5: Modul Sense	29
Obr. 6: Banana Pi M5	30
Obr. 7: Orange Pi 4b	31
Obr. 8: Klon dosky Arduino Mega 2560	32
Obr. 9: Doska Arduino Yún.....	33
Obr. 10: Motor shield.....	35
Obr. 11: Modul Raspberry Pi 4 Model B.....	40
Obr. 12: Modul UNO neznámeho výrobcu	40
Obr. 13: Vrchný a spodný pohľad na PIR snímač	41
Obr. 14: Snímač DHT11 s LED kontrolkou	42
Obr. 15: Snímač AM2302 DHT22	43
Obr. 16: Schéma zapojenia projektu.....	45
Obr. 17: Schéma prepojenia plôch na nespájkovanom poli.....	46
Obr. 18: Zapojenie prvej časti projektu	48
Obr. 19: Zapojenie druhej časti projektu	49
Obr. 20: Prostredie programu Thonny Python IDE	50
Obr. 21: Prostredie programu Arduino IDE	55
Obr. 22: E-mailová notifikácia	59
Obr. 23: Predný pohľad na náš prototyp.....	60
Obr. 24: Vrchný pohľad na prototyp	60
Obr. 25: Spustenie prototypu	61

Zoznam skratiek

BD – Big Data

EDGE – Enhanced Data Rates for GSM Evolution

GPIO – General-purpose input/output

GPRS – General Packet Radio Service

GPS – Global Positioning System

GSM – Global System for Mobile Communications

HDMI – High-Definition Multimedia Interface

IDE – Integrated Development Environment

IoT - Internet of things (internet vecí)

LED – Light-Emitting Diode

LTE – Long Term Evolution

OS – Operačný systém

PAN – Personal Area Network

PIR – Pasiv Infrared Detector

PWM – Pulse width modulation

RFID – Radio Frequency IDentifikacion

SaaS – Software as a Service

SIM – Subscriber Identity Module

SoC – System on a Chip

USB – Universal Serial Bus

WiFi – Wireless Fidelity

Úvod

Internet vecí, tento pojem počujeme z každej strany. Tento pojem pre každého z nás znamená niečo iné, avšak každodenne sa s ním stretávame, aj keď o tom niekedy vôbec netušíme. Veľké percento ľudí by dokázalo využiť v dennom živote zariadenia fungujúce na princípe IoT.

Krízu, ktorá spôsobila celosvetová pandémia, sa dotkla množstva ľudí a firiem. Tak ako každá kríza, aj celosvetová pandémia má nielen negatívne dopady ale aj menšie výhody. Spoločnosť si začína uvedomovať požiadavky modernej doby, ktorá vyžaduje využitie informačných technológií na maximum, s cieľom znížiť výdavky alebo mobilitu ľudí. Zníženie výdavkov aj mobility ľudí nám umožňuje aj Internet vecí, ktorý plne podporuje automatizáciu.

Zmyslom tejto diplomovej práce bude oboznámenie sa s Internetom vecí, dôležitými pojmami, ktoré súvisia s Internetom vecí, opis platforiem, spôsob komunikácie medzi zariadeniami, porozumenie princípu fungovania a v štvrtej kapitole aj praktické zhotovenie IoT zariadenia, ktoré sme si v predošlej kapitole navrhli.

1 Súčasný stav riešenej problematiky

V dnešnej dobe je možné si zaobstarať rôzne smart zariadenia, ktoré sú vhodné na bezdrôtovú komunikáciu. Aktuálne sa pre IoT používajú komunikačné štandardy LTE, WiFi alebo Bluetooth. Avšak vo svete a už aj na Slovensku sa spúšťajú nové mobilné siete piatej generácie, 5G siete. Siete pre IoT sú na území Slovenskej republiky budované viacerými operátormi, medzi ktorých patrí napríklad spoločnosť Orange, Telekom, Towercom a Slovanet. [10]

1.1 Charakteristika pojmov súvisiacich s IoT

S pojmom Internet vecí sa nám spájajú dôležité pojmy, ktoré spolu úzko súvisia. Jedným z týchto dôležitých pojmov je Big Data. Pod slovným spojením Big Data rozumieme údaje, ktoré môžu byť štruktúrované ale aj neštruktúrované. V prípade BD nie je dôležitý význam získaných dát, ale ide o spôsob, akým sa s nimi zaobchádza. BD sa z hľadiska uľahčovania obchodných rozhodnutí, aby sa mohli formulovať ideálne stratégie pre podnik, zhromažďujú, a neskôr sa štruktúrujú s ich následnou analýzou. Keďže objem dát neustále v aktuálnej situácii stúpa, tak práve BD nám pomáha tieto dáta analyzovať a ukladať. Vďaka tomu, že tieto dáta vieme zhromažďovať a analyzovať, tak môžeme aplikovať zmeny vo firme, ako napríklad zníženie nákladov či úspora času, ktorý môžeme použiť na vývoj nových produktov, ktoré môžu prísť častejšie. [10]

Ďalším pojmom, súvisiacim s IoT, je pojem Cloud computing. Pojem Cloud computing si zadefinujeme ako model, ktorý umožňuje pohodlný prístup na požiadanie k výpočtovým zdrojom odkiaľkoľvek. Na zabezpečenie týchto zdrojov je potrebná len minimálna námaha manažmentu alebo interakcia poskytovateľa. Servisné modely na báze cloudu si opíšeme postupne:

- Software as a Service – Cloudová infraštruktúra poskytovateľa obsahuje aplikácie, ktoré používa spotrebiteľ. Ide o tenkého klienta, čiže rozhranie, kedy sú aplikácie dostupné z klientovho zariadenia. Infraštruktúra funguje na cloude u poskytovateľa, včítane sietí, serveru, operačného systému a úložného priestoru, pričom spotrebiteľ ju nespravuje. Na báze SaaS cloudu je založená napríklad služba Gmail od spoločnosti Google, pričom si väčšina

Ľudí nevedomuje, že používaním tejto služby využíva Cloud computing. [10]

- Platform as a Service – Spotrebiteľovi je poskytnutá infraštruktúrna platforma, ktorá je predinštalovaná. Môže ísť napríklad o Java development, .NET runtime a pod.. Zákazník nespravuje jadro infraštruktúry, ktoré obsahuje sieť, servery, operačné systémy a úložiská. Spravuje však nasadené aplikácie, konfiguračné nastavenia pre svoj hosting a prostredie aplikácií.
- Infrastructure as a Service – Tento model poskytuje zákazníkovi výpočtové zdroje, sieť, spracovanie a ukladanie dát. Následne si zákazník na túto infraštruktúru môže prevádzkovať po nasadení ľubovoľný softvér. Zákazník má síce kontrolu nad OS a aplikáciami, ktoré nasadzuje, avšak základnú infraštruktúru pri tomto modeli nespravuje a nekontroluje. Nad sieťovými komponentami môže mať zákazník kontrolu, avšak len nad vybranými typmi komponentov. [8]

Ďalší pojem, ktorý je dôležitý spomenúť je Outsourcing. Pojem sa chápe ako spojenie anglických slov out & source, čo v preklade znamená slová von a zdroj. Avšak pre správne pochopenie pojmu si pojem zapíšeme ako Outside Resource Using, čiže používanie vonkajších zdrojov podniku. Outsourcing však v podniku vyvoláva váhanie v stratégii Make or Buy (Vyrobenie alebo Nákup).

V prípade, ak chce spoločnosť dosiahnuť rozšírenie IoT vo svojom podniku, tak vďaka Outsorcingu ho dosiahne s investíciou nižších nákladov. Často ide o presun časti aktivít vykonávaných v podniku vo vlastnej réžii do správy externého podniku, ktorý tieto aktivity podniku predáva. V prípade Outsorcingu ide tiež o účinnejšie testovanie IoT skôr, ako sa vedenie spoločnosti rozhodne implementovať daný model. Rozvoj IoT sa stretáva s problémami, ktoré Outsourcing pomáha vyriešiť, ide napríklad o nedostatok informácií, prípadne o zdroje. Subjekt, ktorý sa stará o Outsorcingovanie návrhu, však má iný pohľad na Internet vecí, a zaujíma sa o možné zmeny a príležitosti, s ktorými podnik vôbec nepočíta alebo sa o ne nezaujíma. [10]

1.2 Internet of Things

Internet vecí je aktuálne na začiatku svojej éry, avšak už sa začína používať vo väčšej miere, keďže sa s ním začíname stretávať v každodennom živote. Virtuálne a fyzické objekty si vymieňajú rôzne údaje cez internet. Toto prepojenie medzi virtuálnym a reálnym svetom vkladá do sveta nové možnosti, keďže zariadenia skonštruované na princípe IoT, po zozbieraní všetkých potrebných dát, vedia fungovať takmer alebo aj plne autonómne. Pointou Internetu vecí je to, že sa z bežne používaných „hlúpych“ zariadení stanú smart zariadenia.

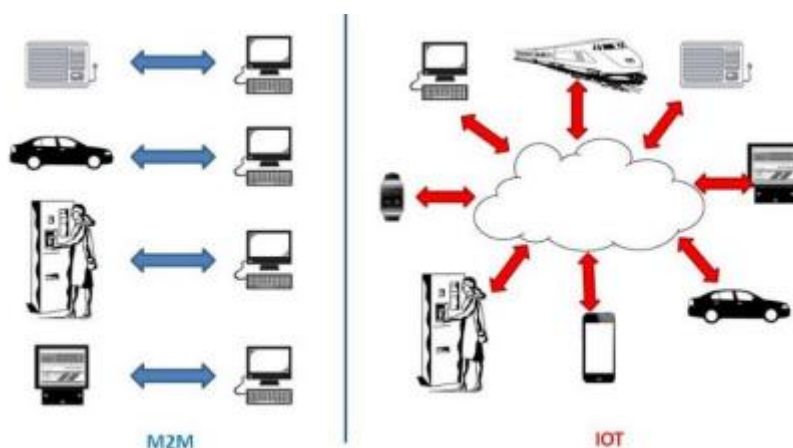
Internet vecí je možné definovať ako sieť fyzických objektov, obsahujúcich komunikačnú technológiu, ktorá je zabudovaná a ktorá slúži na snímanie údajov alebo interakciu s vonkajším prostredím. [4]

Pomocou internetu môže Internet vecí spojiť celý veľký online svet priamo s ľuďmi. K dosiahnutiu tohto cieľa sú používané rôzne snímače, ktoré zbierajú dáta zo smart objektov z fyzického sveta. Zozbierané údaje sú po získaní nahrávané do cloudu, kde sa stanú Big Datami. Následne sú tieto údaje integrované a použité na vývoj inteligentných systémov. Z tohto dôvodu je IoT jednou z hlavných technológií, ktoré vedú štvrtú priemyselnú revolúciu. V inteligentných systémoch sú vytvárané silné hodnoty, ktoré využívajú inteligentné technológie, vrátane IoT, pre dynamický inteligentný svet. Inteligentné systémy sú neustále vyvíjané pre spracovanie Big Data, ktoré sú získané prostredníctvom Internetu vecí.

V čase prvého rozvoja IoT, v roku 2000, bola vyvinutá technológia RFID, ktorá bola dôležitá najmä pre riadenie zásob a logistiku. Technológia RFID bola aplikovaná predovšetkým z dôvodu, že sa vďaka nej podarilo znížiť náklady na distribúciu a výrobu výrobkov. RFID bola tiež využívaná na sledovanie polohy výrobkov, ktoré boli dodávané pomocou lokalizačných informačných systémov. Technológia sa neustále vyvíjala, až sa vyvinula do aplikácií typu machine-to-machine. Aplikácie typu machine-to-machine umožňujú monitoring, riadenie a komunikáciu so zariadeniami so vzdialenou aplikačnou infraštruktúrou, na ktorú používa komunikačné kanály. Machine-to-machine komunikácia sa v najnovšej verzii rozšírila do sveta internetu. IoT pomohol vyvinutiu internetu do svojej tretej generácie – pokročilejšia forma machine-to-machine komunikácie. Prvá generácia bola označovaná ako Internet počítačov (IoC – Internet of Computers) a druhá generácia,

ktorá bola zameraná viac na zákazníkov, bola označovaná ako Internet ľudí (IoP – Internet of People). Predpokladá sa však, že okolo roku 2030 sa internet vyvinie do štvrtej generácie, ktorá bude označovaná ako Internet všetkého (IoE – Internet of Everything). [11]

Pri machine-to-machine komunikácii, kam spadajú napríklad zariadenia ako platobné terminály, automaty, kamerové a požiarne systémy, mýtny systém, sú používané aj SIM karty. Počet aktívnych SIM kariet na Slovensku presiahol v máji 2020 jeden milión. [7]



Obr. 1: Porovnanie Machine-to-Machine a IoT

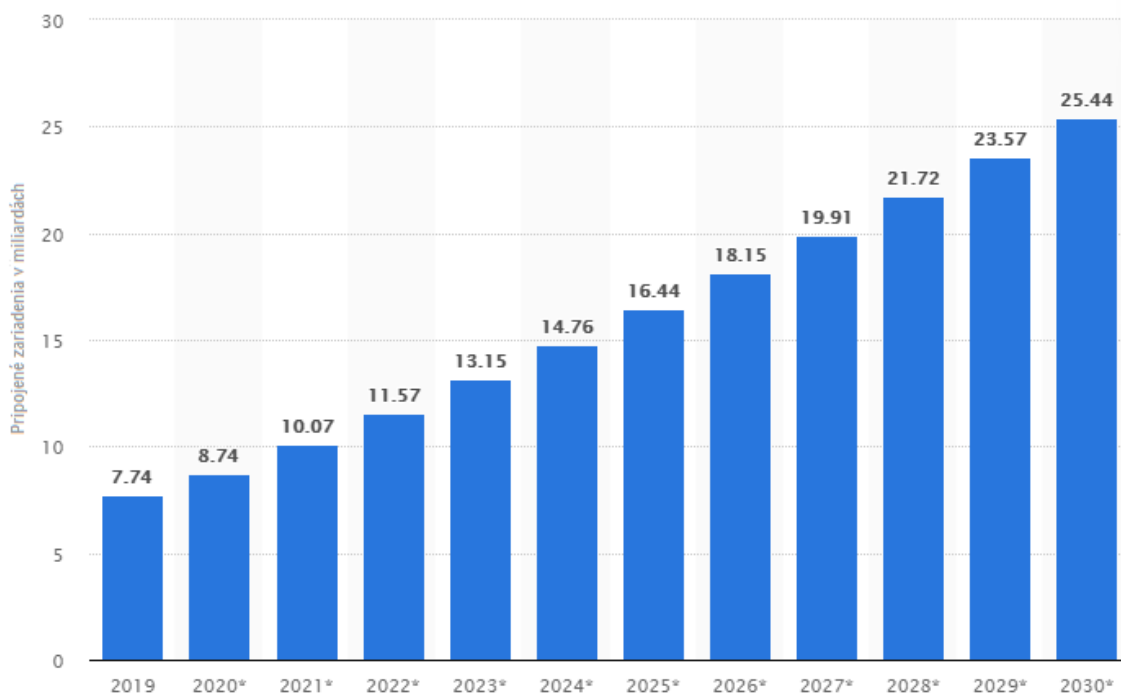
Zdroj: <https://joycheruvathoor.com/2015/10/31/iot-and-m2m-are-they-the-same/>

Pojem Internet vecí spomenul v roku 1999 ako prvý Kevin Ashton počas svojej prezentácie. Za vyše 20 rokov od prvej zmienky o tomto pojme sa Internet vecí stal bežným pojmom, ktorému rozumie skoro každý obyvateľ planéty. Najčastejšie si ľudia spájajú tento pojem s inteligentnou domácnosťou. Pod inteligentnou domácnosťou rozumieme domácnosť, ktorá vie vykonávať určité rozhodnutia samostatne a niektoré pomocou rozhodnutia človeka, avšak vzdialeným prístupom. Môže ísť napríklad o zapnutie či vypnutie klimatizácie pri dosiahnutí určitej teploty alebo vlhkosti vzduchu, automatické zapínanie svetiel po zotmení alebo počas dovolenky, alarmový systém, ktorý hlási majiteľovi neželaný pohyb v jeho nehnuteľnosti, ale môže ísť napríklad aj o zapnutie vetrania a kontaktovanie bezpečnostných zložiek v prípade, že v byte snímače zaregistrujú únik plynu alebo požiar.

IoT používajú nielen domácnosti ale aj firmy. Najčastejším príkladom používania Internetu vecí vo firme je v spojení s GPS sledovanie presnej polohy služobných vozidiel.

Medzi rokmi 2008 a 2009 bol podľa spoločnosti CISCO počet svetovej populácie prekročený počtom zariadení pripojených k internetu. Počítače, ktoré sa k internetu pripájali ako prvé, neskôr nahradili smartfóny. [10]

Podľa aktuálnych predpovedí sa počet IoT zariadení do internetu na celom svete do roku 2030 stonásobí. V roku 2020 sa najväčší počet aktívnych zariadení nachádzal v Číne, pričom išlo o 3,17 miliárd zariadení. Zariadenia IoT sa používajú vo všetkých druhoch odvetví, ale aj na spotrebiteľských trhoch. Spotrebiteľský trh predstavoval v roku 2020 približne 60 percent zariadení, ktoré sú pripojené k Internetu vecí. Predpokladá sa, že počas nasledujúcich desiatich rokov sa toto percento výrazne nezmení. S počtom viac ako sto miliónov pripojených IoT zariadení patria medzi hlavné odvetvia výroba a distribúcia elektriny a plynu, vodárne, hospodárenie s odpadom, maloobchod, veľkoobchod, skladovanie a preprava tovaru a v neposlednom rade vládne inštitúcie. Aktuálne sa predpokladá, že do roku 2030 stupne počet IoT zariadení vo všetkých odvetviach o viac ako 8 miliárd zariadení. Najväčším podielom v segmente spotrebiteľov dominujú internetové a mediálne zariadenia, medzi ktoré patria smartfóny a tablety, pričom sa počet týchto zariadení zvýši na viac ako osem miliárd. Za týmito zariadeniami potom pokračujú v počte jednej miliardy pripojených zariadení autonómne vozidlá, infraštruktúra IT, inteligentná sieť, sledovanie a monitorovanie majetku. [6]



Obr. 2: Predpokladaný vývoj pripojených IoT zariadení

Zdroj: www.statista.com/statistics/1183457/

V Internete vecí hrajú dôležitú a kritickú úlohu snímače. Snímače zbierajú dáta a ukladajú ich na internet, do cloudu. Snímače môžu byť zabudované do existujúcich fyzických zariadení alebo môžu existovať ako samostatné externé zariadenia. Snímače zachytávajú rôzne dáta, ide napríklad o pohyb alebo polohu. Tiež môže ísť o dôležité údaje pre ľudí, vrátane teploty, vlhkosti, kvality ovzdušia v atmosfére, hluk alebo svetelný smog. Internet vecí poskytuje vzdialenú kontrolu pre klimatizáciu, vykurovanie a osvetlenie v budovách. Z tohto dôvodu je nevyhnutné mať nainštalované rôzne snímače, ktoré poskytujú reálne dáta o dianí. Pre komunikáciu medzi snímačmi a zariadeniami, ktoré rozhodujú o úlohách preto zohráva dôležitú úlohu sieťové rozhranie. Pri Internete vecí sem patria siete ako PAN, WPAN, WiFi, všetky generácie mobilných sietí, Bluetooth, satelitné siete ako GPS, Glonass, Galileo, sériová komunikácia alebo mikrovlnná sieť. Rýchlosť prenosu dát je závislá na zvolenej technológii. V prípade, ak sa nevyžaduje náročný prenos dát (časté a objemné dáta), tak je možné použiť akékoľvek pripojenie. V prípade náročnejšieho prenosu sa volí technológia, ktorá spĺňa všetky požiadavky na kvalitu a rýchlosť prenosu. Predpokladá sa, že Internet vecí bude mať v budúcnosti veľký vplyv na každodenný život, ale aj na všetky odvetvia priemyslu. Tento vplyv bude z dôvodu, vývoja a neustále narastajúceho využívania niektorých technológií, medzi ktoré patria bezdrôtové

siete, komunikačné moduly, snímače a smart terminály. Významné zmeny sa prenesú hlavne do odvetví ako zdravotníctvo, vzdelávanie, výroba alebo distribúcia.

Svetová populácia aktuálne žije v hyperkonektivite. Táto konektivita znamená to, že ľudia a nehmotné veci sú prepojené prostredníctvom IoT. Internet vecí aktuálne naozaj vedie éru superfúzie. Táto superfúzia vytvára hodnoty ekonomické, etické a sociálne, ktoré sú spájané s priemyselným odvetvím, ktoré je vyjadrované ako produktívny obchodný model.

Ako sme vyššie spomínali, tak Internet vecí bol prostredníctvom technológie RFID a Machine-to-machine vyvinutý pre jednoduchú komunikáciu medzi objektmi a zariadeniami. IoT však tiež úplne nové typy služieb a produktov, ktoré firmy v minulosti nepredpokladali. V roku 1969, kedy začali vznikať prvé uzly, ktoré začínali tvoriť internet (v tej dobe ARPANET), sa nepredpokladal masívny rozmach siete a informačná revolúcia vo svete. Internet vecí však pokračuje v nie menšej miere v týchto stopách. [11]

1.3 5G sieť

Aby sme jasne pochopili termín 5G sieť, tak si musíme opísať podrobný plán vývoja smerom ku 5G komunikácii.

- **Pred 1G:** Išlo o bezdrôtové komunikácie, ktoré boli hlasovo-orientované, pričom využívali analógové systémy so single-sideband moduláciou (modulácia s jedným postranným pásmom).
- **1G:** V prípade prvej generácie sietí išlo o všetku bezdrôtovú komunikáciu, ktorá bola hlasovo-orientovaná. Spoločnosť Bell Labs sa v roku 1966 rozhodla prijať analógový systém pre vysokokapacitný mobilný systém. Analógový systém firma zvolila z dôvodu, že v tom čase bola výroba digitálnych rádiových systémov príliš drahá, a preto firma zvolila analógový systém s FM rádiom. V roku 1983 sa zvolil názov amerického bunkového systému, išlo o AMPS (Advanced Mobile Phone Services – v preklade: Pokročilá služba mobilných telefónov).
- **2G:** Nástup sietí druhej generácie prišiel v roku 1990. Tak ako pri predchádzajúcich generáciách išlo tiež o hlasovo-orientovanú bezdrôtovú komunikáciu. Išlo však už o digitálny systém, preto môžeme povedať, že prechod zo sietí prvej generácie na druhú znamenal migráciu

z analógového systému na digitálny. V Európe vznikol systém GSM, a v Severnej Amerike systém IS-54 (tento systém bol neskôr označený za nežiadúci a od jeho používania sa upustilo).

- **2.5G:** Túto generáciu sietí poznáme od roku 1995, pričom v Európe išlo o rozšírenie systému GSM na GPRS a EDGE. V Spojených štátoch bol prijatý systém CDMA (viacnásobný prístup s rozdelením kódu). Pri sieťach 2.5G išlo o bezdrôtovú komunikáciu najmä pre vysokokapacitný prenos hlasu s obmedzenými dátovými službami (napr. SMS správy) a teda nešlo už iba o hlasovo-orientovanú komunikáciu ako pri predchádzajúcich troch generáciách.
- **3G:** Bezdrôtová komunikačná platforma disponuje hlasovými aj dátovými schopnosťami. Túto generáciu sietí poznáme od roku 1999, pričom pracuje v režime duplexu s frekvenčným delením a aj časovom duplexe. Prechodom zo sietí druhej generácie na tretiu generáciu systémov bol zavŕšený vývin z hlasovo-orientovaných systémov na systémy, ktoré sú dátovo orientované.
- **4G:** Od roku 2013 sa vo svete začali vo veľkom používať siete štvrtej generácie, keďže tieto siete priniesli niekoľkonásobne vyššie rýchlosti. Pri sieti štvrtej generácie ide vlastne o vysokorýchlostný dátový tok plus systém pre hlas. Pre siete štvrtej generácie poznáme dva druhy systémov. Jeden, ktorý bol vyvinutý v Spojených štátoch amerických, WiMAX. Systém WiMAX využíva ortogonálny frekvenčne delený multiplex, ktorý je vyvinutý zo siete WiFi. Druhým systémom je systém LTE, ktorý bol vyvinutý po systéme WiMAX. Oba systémy používajú šírku pásma 20 MHz. Prechod zo sietí tretej generácie na siete 4G znamenal posun od nízkych dátových rýchlostí na vysokorýchlostný dátový tok, vhodný pre mobilné video a dostupnejší internet. [9]
- **5G:** Siete piatej generácie boli spustené už v roku 2019 v Južnej Kórei, no na Slovensku sa po testovacej prevádzke 5G sieť spustila do prevádzky až v roku 2020. Piata generácia sietí prináša rýchlejšiu odozvu, niekoľkonásobne vyššie prenosové rýchlosti a nižšiu spotrebu energie.

Pri sieťach piatej generácie je predpoklad, že do nej bude zapojených približne 50 miliárd zariadení. Existuje predpoklad, že všetko, čo môže mať prospech z pripojenia, bude

pripojené. Vozidlá, ktoré sú vyrobené po roku 2018, musia mať na Európskom trhu systém eCall, ktorý automaticky po dopravnej nehode kontaktuje bezpečnostné zložky. Tieto systémy potrebujú k svojej činnosti tiež mobilnú sieť, aby vedeli bezpečnostným zložkám odoslať svoju aktuálnu polohu. Táto funkcia pomôže zachrániť život pri dopravnej nehode v situácii, keď sa nehoda stane niekde, kde nie je častý prechod iných vozidiel alebo keď sú účastníci vážne zranení tak, že si nevedia privolať pomoc sami. Pri podobných systémoch, kedy sa pomocou mobilných zariadení napr. vo vozidlách vyžaduje komunikácia v reálnom čase na veľkú vzdialenosť, je vyžadovaná čo najnižšia odozva, kratšia ako jednu milisekundu. [9]

1.4 WiFi sieť

Veľmi často sa pri IoT zariadeniach nainštalovaných hlavne v domácnostiach, prípadne vo firmách používa sieť WiFi, pomocou ktorej získané údaje odosielať do cloudu. Najčastejšie sa v domácnostiach pripájajú k internetu zariadenia ako televízor, práčka, sušička, chladnička či umývačka riadu. Tieto zariadenia vedia komunikovať v lokálnej sieti medzi sebou, no najčastejšie sa tieto zariadenia pripájajú k serverom, ktoré spravujú výrobcovia prostredníctvom Outsorcingu. Vďaka pripojeniu k serveru si vie majiteľ zariadenia zistiť stav zariadenia, či si nevyžaduje údržbu, v akom stave sa práve nachádza prací cyklus alebo koľko času zostáva do konca umývacieho cyklu v umývačke riadu, prípadne môže ísť o vzdialené spustenie klimatizácie. Výhodou použitia WiFi je to, že drvivá väčšina domácností disponuje internetovým pripojením a z nich skoro každá domácnosť má aj bezdrôtovú sieť WiFi. Ďalšou výhodou je, že pri používaní IoT zariadenia alebo aplikácie nie je potrebná dodatočná investícia, aby boli zariadenia pripojené do siete. Pri používaní WiFi siete je možné odosielať objemnejšie dáta, keďže táto sieť je vhodná na prenos väčšieho objemu dát.

1.5 Technológia Bluetooth

Bluetooth od verzie 4.0 podporuje technológiu Low Energy, kedy bola maximálna prenosová rýchlosť definovaná na 1Mb/s. Od tejto verzie sa na menej náročné prenosy spolu s nízkou energiou začala technológia Bluetooth používať aj pri zariadeniach pre Internet vecí. V roku 2016 bola predstavená verzia technológie 5.0, pričom sa jej maximálna

prenosová rýchlosť zvýšila dvojnásobne, na 2 Mb/s. V roku 2019 bola predstavená zatiaľ najnovšia verzia tejto technológie, a to Bluetooth 5.2. V tejto verzii bola vylepšená správa energie, možnosť prenášať väčšie množstvo informácií cez menšiu šírku pásma, a teda pri nižšom výkone. Táto verzia priniesla tiež novinku, ktorou je protokol Enhanced Attribute, vďaka ktorému je umožnená paralelná transakcia medzi dvoma zariadeniami súčasne. [1] Z dôvodu, že technológia Bluetooth má nižší dosah (vo verzii 5.0 počas testov až 800 metrov), tak je veľmi často používaná pri IoT zariadeniach, ktoré nevyžadujú komunikáciu na veľkú vzdialenosť. Najčastejšie sa stretávame s použitím technológie pri nositeľných zariadeniach, medzi ktoré patria smart hodinky, náramky, bezdrôtové slúchadlá a reproduktory. Veľmi často sa technológia používa aj v domácnostiach v rámci PAN siete, môže ísť napríklad o smart LED osvetlenie v domácnostiach.

1.6 Siete používané pre IoT

V tejto kapitole si predstavíme najznámejšie a najpoužívané siete, ktoré sa používajú pre IoT. Pri jednotlivých sieťach si predstavíme ich základnú charakteristiku, možnosti ich využitia a ich výhody či prípadné nevýhody oproti konkurenčným sieťam.

1.6.1 Zigbee

Sieť Zigbee je bezdrôtová technológia, ktorá bola vyvinutá ako voľný, otvorený globálny štandard, ako vhodné riešenie potrieb lacných bezdrôtových sietí pre Internet vecí, vyžadujúci nízky výkon. Zigbee operuje na rádiovkej špecifikácii IEEE 802.15.4, pričom funguje v nelicencovaných frekvenčných pásmach s hodnotami 868 MHz, 900 MHz a 2,4 GHz. Protokol Zigbee bol navrhnutý spôsobom, aby poskytoval bezdrôtové dátové riešenie, ktoré je ľahko použiteľné a je charakterizované spoľahlivými a bezpečnými architektúrami siete. Sieť Zigbee vo svojej aktuálnej verzii 3.0 po pripojení k doméne IP vytvára možnosť monitorovania a kontroly nad zariadeniami v sieti LAN alebo WAN, vrátane internetu a prináša naozaj Internet vecí.

Zigbee protokol zahŕňa nasledujúce funkcie:

- nízka latencia,

- podpora sieťových topológií ako: point-to-point, point-to-multipoint alebo MASH,
- nízky pracovný cyklus, vďaka ktorému je poskytovaná vyššia výdrž batérie v zariadení,
- šifrovanie AES (128-bitové) pre bezpečné dátové pripojenia.

Vďaka nízkej spotrebe energie umožňuje sieť Zigbee zariadeniam, ktoré ju využívajú, pracovať dlhé roky na batériu, pre množstvo monitorovacích a riadiacich aplikácií. Najčastejšie sa používa pri automatickom odčítaní pomerových meračov tepla, ovládanie osvetlenia, pri lekárskejších prístrojoch alebo pri systémoch riadenia vzduchotechniky vo firmách. [14]

1.6.2 Thread

Sieťový protokol Thread vznikol pred šiestimi rokmi, no má vysoký potenciál bez problémov priviesť protokol IP k sieťovým zariadeniam s nízkou spotrebou energie v oblasti inteligentných domov a budov. Protokol má potenciál prepájať kamerový systém a zámky dverí s bezpečnostnými riešeniami zabezpečenia dátovej transportnej vrstvy, ktoré sa používa aj pri transakciách platobnými kartami. Thread používa platformu OpenThread, čiže Open-source platformu, vďaka ktorej protokol Thread získava silu komunity vývojárov, ktorí neustále zdokonaľujú tento protokol. Tento protokol využíva rovnakú rádiovú špecifikáciu ako sieť Zigbee, čiže tieto siete sú veľmi podobné. [12]

1.6.3 Z-Wave

Z-Wave je tiež bezdrôtový komunikačný protokol, ktorý sa používa najmä v inteligentných domácich sieťach a umožňuje prepojeným zariadeniam vymieňať riadiace príkazy a dáta. Sieť Z-Wave sa skladá zo zariadení IoT, hlavného radiča (domáci smart rozbočovač), ktorý je ako jediné zariadenie v sieti Z-Wave pripojené k internetu. K rozbočovaču môže byť pripojených až 231 zariadení. Protokol pracuje vo frekvenčnom pásme 868,42 MHz, vďaka čomu sa vyhýba rušeniu v pásme 2,4 GHz, ktoré je používané

technológiami Bluetooth a WiFi. Sieť ponúka prenosovú rýchlosť 9,6 kilobajtov za sekundu, 40 alebo 100 kilobajtov za sekundu. Protokol Z-Wave tiež zhŕňa šifrovanie AES (128-bitové) a podporu protokolu IP verzie 6. Oproti sieti Zigbee ponúka sieť Z-Wave až trojnásobne väčší dosah, až 29 metrov. [15]

1.6.4 NarrowBand-IoT

Sieť patrí k úzkopásmovým bezdrôtovým technológiám LPWAN. Sieť je určená výhradne iba na prenos dát. Na území Slovenskej republiky túto sieť prevádzkuje spoločnosť Telekom, pričom medzi výhody siete NarrowBand-IoT patria napríklad nízke náklady na prevádzku, nízka energetická náročnosť, celosvetová štandardizácia, stabilita, bezpečnosť, dobrý dosah signálu a licencované frekvenčné pásmo. [10]

1.6.5 SigFox

Sieť SigFox sa využíva na odpočty spotrebovaných energií, meraní zrážok v meteostaniciach, logistike a podobne. Veľkosť odoslanej správy je obmedzená na 96 bitov, a počet správ, ktoré môžu byť odoslané, je 140 na jeden deň. Vďaka nízkej cene čipu modemu (max. 2 eurá) a ročného stropu na prenesené dáta (približne 9 eur), sa radí táto sieť medzi nízkonákladové. Pokrytie sieťou SixFox je v rámci 60 krajín až 5 miliónov km². [10]

1.6.6 LoRaWAN

Frekvenčné pásmo, v ktorom je sieť prevádzkovaná, je 868 MHz. Pokrytie siete je v zastavanej časti niekoľko kilometrov, na voľnom priestranstve to sú desiatky km. Komunikácia prostredníctvom siete je šifrovaná na úrovniach. [10]

1.7 Platforma Raspberry Pi

Raspberry Pi je malý a lacný počítač vo veľkosti približne platobnej karty. Tento počítač je vhodný napríklad na prehľadávanie internetu, hranie nenáročných hier alebo na programátorský tréning pri písaní vlastných programov, no je veľmi vhodný aj na vytváranie vlastných logických obvodov. Napriek svojim malým rozmerom je to plnohodnotný počítač, ktorý zvláda všetky bežné požiadavky. Platforma Raspberry Pi sa zrodila z myšlienky podporiť celosvetové počítačové vzdelávanie, ktoré bude finančne dostupné. Tvorcovia vytvorili v roku 2012 niekoľko tisíc testovacích dosiek, ktoré boli takmer hneď vypredané. V nasledujúcich rokoch dosia celosvetovo počet milión dodaných dosiek. Tieto dosky sa stali úspešné v domácnostiach, školách, kanceláriách, dátových centrách, no dostali sa aj do vesmíru rámci kozmických balónov.

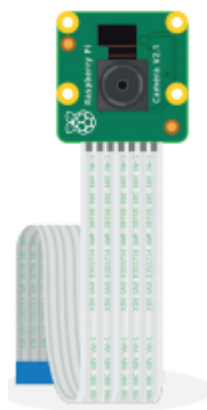
Prvý model Raspberry Pi bol Raspberry Pi model B, od ktorého boli neskôr odvodené ďalšie modely, ktoré mali buď zlepšené hardvérové vlastnosti alebo zmenšené rozmery, vhodné pre rôzne prípady použitia. Menšia verzia, Raspberry Pi Zero, je plnohodnotná verzia pôvodného modelu, avšak namiesto USB typu A má porty USB B typu micro (alebo typu C) a nižšie energetické nároky. Veľmi dôležité je spomenúť tiež to, že všetky modely Raspberry Pi sú kompatibilné, čo zaručuje, že program, ktorý je písaný pre jeden z množstva modelov bude fungovať úplne rovnako na inom modeli.

1.7.1 Komponenty na doske Raspberry Pi

Raspberry Pi sa skladá z viacerých komponentov, ktoré si teraz opíšeme. Najdôležitejším komponentom je pod kovovým krytom SoC, čo je integrovaný obvod, v ktorom je obsiahnutá väčšina, prípadne všetky súčasti počítača, ako sú napríklad procesor, pamäť, grafický procesor. Hneď vedľa SoC nájdeme čip, ktorý obsahuje pamäť RAM (pamäť s náhodným prístupom). Pamäť slúži na dočasné uloženie programu, s ktorým sa pracuje a pri strate napájania je pamäť vymazaná. Ďalej sa na doske nachádza tzv. rádio čip, komponent, vďaka ktorému je Raspberry schopné komunikovať bezdrôtovo. Najnovšie Raspberry obsahuje v tomto rádio čipe WiFi, ktorým sa pripája k LAN alebo k Internetu, a Bluetooth, ktorý slúži na komunikáciu s periférnymi zariadeniami. Medzi ďalšie komponenty umiestnené na doske patrí čip – USB radič, ktorý sa stará o prevádzku 4 USB portov, ktoré sa nachádzajú na doske. Hneď vedľa neho je ďalší čip, ktorý ovláda Ethernet

port, ktorý sa nachádza vedľa USB portov a umožňuje káblové pripojenie do komunikačnej siete prostredníctvom krútenej dvojlinky. Pod napájacím konektorom (v najnovšej verzii je doska Raspberry Pi napájaná cez port USB typu C) sa nachádza integrovaný obvod správy napájania.

Raspberry Pi obsahuje viacero portov, medzi ktoré patria štyri porty USB, vďaka ktorým je možné pripojiť periférne zariadenia ako klávesnica alebo myš k modulu. Polovica z portov je typu USB 2.0 a polovica USB 3.0, ktoré poskytujú vyššiu prenosovú rýchlosť. Tak ako sme vyššie spomínali, doska obsahuje aj port RJ-45 na pripojenie Ethernetového kábla. Za USB portami je ďalej 3,5 mm konektor, ktorý okrem zvukového prenosu zvláda aj video prenos a s kompatibilným displejom a adaptérom dokáže zobrazovať video výstup. V blízkosti 3,5 mm jacku sa nachádza konektor na pripojenie kamerového modulu.



Obr. 3: Kamerový modul pre Raspberry Pi

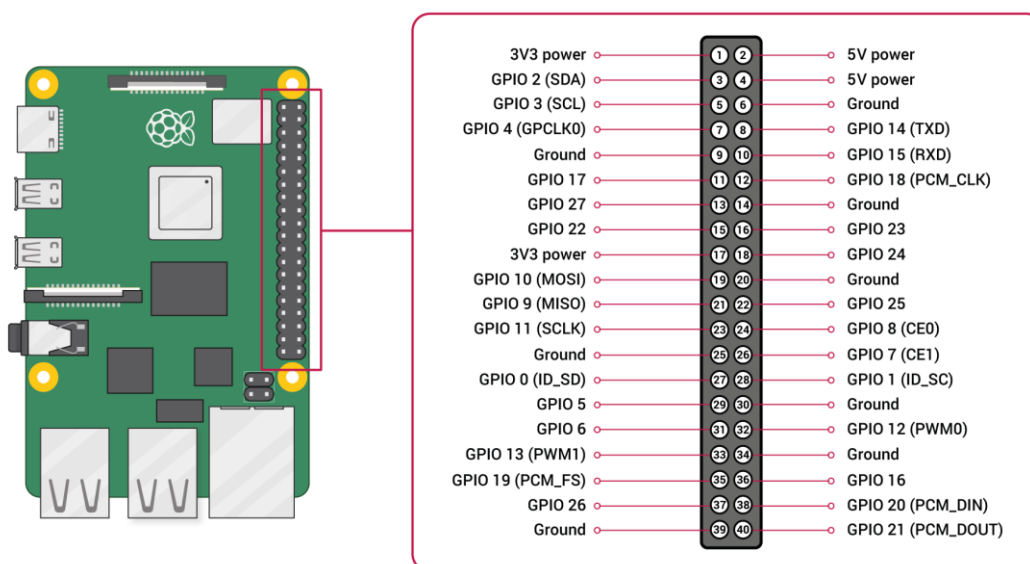
Zdroj: Raspberry Pi – Užívateľská príručka

Na doske najnovšieho modelu, Raspberry Pi 4 Model B, sa nachádzajú dva micro-HDMI porty, ku ktorým sa pripájajú displeje alebo monitory – je možné ich používať naraz. Na tomto modeli sa nachádza nový napájací port typu USB-C, ktorý sa používa v moderných smartfónoch štandardne a preto je možné používať nabíjačku zo smartfónu ako napájací zdroj, no odporúča sa používať oficiálny napájací zdroj z dôvodu možných vyšších odberov prúdu, ktoré bežná nabíjačka zo smartfónu nemusí stíhať dodávať. Na hornej časti nájdeme dva rady kovových hrotov, ktoré trčia z dosky. Tieto hroty sú označené ako GPIO, čo znamená univerzálny vstup a výstup. Tieto hroty sú veľmi dôležité pri konštruovaní IoT aplikácií, pretože slúžia na komunikáciu s LED, tlačidlami, snímačmi a podobne. V blízkosti týchto hrotov sa nachádzajú ďalšie štyri, ktoré slúžia na napájanie pomocou ethernetového kábla, čiže nie je potrebné napájať dosku samostatným zdrojom napätia. Posledný konektor,

ktorý sa nachádza na vrchnej strane dosky, je konektor na pripojenie dotykového displeja. Pomocou tohto konektora sa prenášajú dotyky, displej musí byť pripojený pomocou HDMI rozhrania. Na spodnej časti dosky sa nachádza micro-SD šachta, do ktorej sa vkladá micro-SD karta, ktorá slúži ako úložisko. Na tomto úložisku je tiež nahraný operačný systém, no slúžia taktiež na ukladanie dát.

1.7.2 GPIO

V hornej časti dosky sa nachádzajú porty na univerzálny vstup a výstup, označovaný ako GPIO. Ide o 40 vytŕčajúcich kovových hrotov, ktoré sa používajú na pripojenie snímačov alebo LED. Tieto hroty je možné používať ako vstup alebo aj výstup.



Obr. 4: Vstupné a výstupné hroty GPIO – schéma

Zdroj: <https://www.raspberrypi.org/documentation/usage/gpio/>

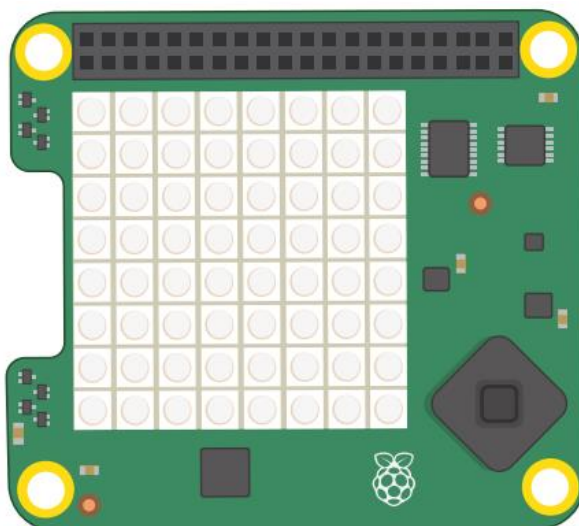
Výstupy označené ako 5V power dodáva napätie 5 voltov na výstup stále, počas celej doby, čo je zariadenie pripojené k zdroju napájania. Označenie 3V3 power znamená, že na týchto hrotoch je stále napätie 3,3 voltov, keď je Raspberry Pi pripojené k zdroju. Ground, je záporný pól napájania, ktorý uzatvára elektrický obvod. Hroty, ktoré v schéme majú označenie GPIO sa používajú ako vstupno/výstupné hroty, ktoré sa definujú v programe,

ktorým sú ovládané podľa našej vôle, okrem portov GPIO 0 a GPIO 1, pretože tieto porty sú rezervované na použitie hardvéru, ktorý sa pripája k doske zvrchu (HAT) alebo na iné príslušenstvo. [3]

1.7.3 Sense HAT modul

Ide o multifunkčný doplnok, ktorý obsahuje joystickový ovládač, maticu 8x8 LED diód, ktoré disponujú farbami zo škály RGB, pričom je možné ich naprogramovať tak, že môžu vydávať akúkoľvek farbu. Sense HAT modul tiež obsahuje šesť zabudovaných snímačov:

- **Barometer:** Tento snímač meria tlak vzduchu. Často sa používa na predpoveď počasia, keďže pri zvýšenom atmosférickom tlaku je predpoklad, že počasie bude slnečné a oblačnosť bude minimálna, naopak zníženie tlaku predpovedá, že sa oblačnosť zvýši a zvyšuje sa aj pravdepodobnosť zrážok. Okrem predpovede počasia sa používa na sledovanie klesania a stúpania v horách, keďže v inej nadmorskej výške je aj iný atmosférický tlak a iná hustota vzduchu.
- **Magnetometer:** Snímač meria silu magnetického poľa Zeme, pričom pomáha sledovať pohyb celého Sense HAT modulu. Snímač je možné použiť ako kompas, na určenie svetových strán, na detekciu kovových predmetov alebo elektrického poľa.
- **Akcelerometer:** Akcelerometer meria silu akcelerácie vo viacerých smeroch.
- **Gyroskop:** Používa sa na meranie zmien uhla v priebehu času, čiže uhlovú rýchlosť. Snímač zisťuje, ako sa otáča modul vzhľadom na Zemský povrch.
- **Snímač vlhkosti:** Údaje o relatívnej vlhkosti sú vhodné na predpoveď počasia, kedy vieme predpokladať, blížiac sa zrážky.



Obr. 5: Modul Sense

HAT Zdroj: Raspberry Pi – Užívateľská príručka

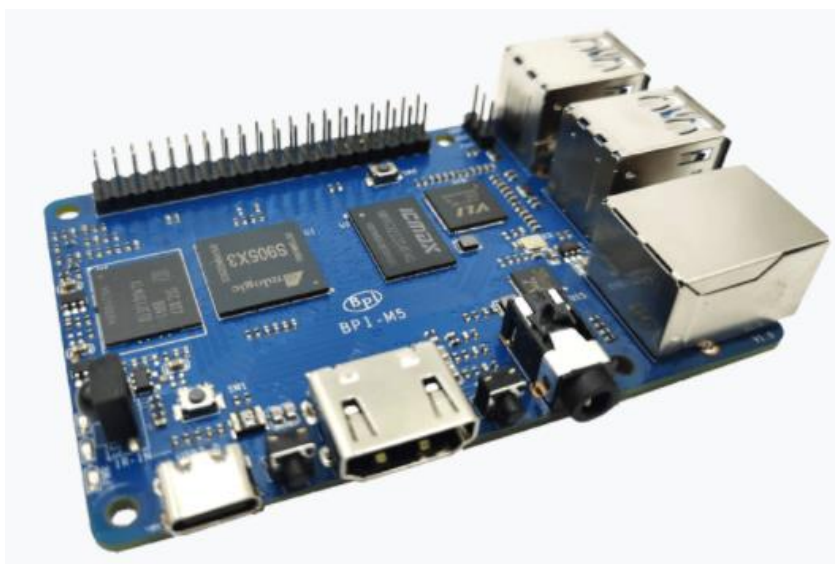
1.7.4 Raspberry Pi 4 Model B – najnovší model

Rýchlosť aj výkon najnovšieho modelu oproti predchádzajúcej verzii rapídne narástli, pričom sa zariadenie výkonovo dotahuje na desktopové zariadenia. Najnovší model je energeticky úspornejší a tichší. Model verzie 4 obsahuje okrem iného aj gigabitový ethernet port, čiže po pripojení k pevnej sieti dokáže rýchlosť prenosu dosiahnuť až 1 gigabit za sekundu, samozrejmosťou je aj bezdrôtové pripojenie prostredníctvom siete WiFi alebo Bluetooth. Pre rýchlejší prenos dát boli upgradované aj USB porty, ich aktuálny počet je 4, z toho sú dva vo verzii 3.0 a dva vo verzii USB 2.0. Pamäť RAM je na výber od 2 GB po 8 GB. Ďalšou zmenou je použitie USB-C portu na napájanie, pričom sa odporúča použiť zdroj s výkonom 15 W. Zmena sa dotkla aj zobrazovacieho portu, pričom pôvodný HDMI port bol nahradený dvomi micro-HDMI portami, čo umožňuje pripojiť k modulu až 2 zobrazovacie monitory. [3]

1.7.5 Podobné platformy

Keďže Raspberry Pi vytvorilo vo svete menšiu revolúciu na trhu s jednodoskovými počítačmi, tak sa okrem lacných napodobnenín z Číny objavili aj ďalšie moduly, ktoré sa aj názvom približovali úspešnej verzii.

Medzi prvé môžeme zaradiť aj úspešnú kópiu - platformu Banana Pi. V každej verzii sa snažili vývojári predbehnúť Raspberry Pi, ktoré malo v komunite určité nedostatky. V prvej verzii obsahovala platforma dvojjadrový procesor, ethernetový port s rýchlosťou 1 Gb/s, SATA konektor, vďaka ktorému bola platforma vhodnejšia k domácim serverom. V ďalšej verzii o túto výhodu prišla, ale dostala štvorjadrový procesor s taktom 1 GHz. V tretej verzii sa konektor SATA vrátil, procesor dostal až osem jadier. Cena je oproti platforme Raspberry Pi väčšia, kvôli vyššiemu výkonu. [2]



Obr. 6: Banana Pi M5

Zdroj: <http://www.banana-pi.org/m5.html>

Ďalšou kópiou, ktorá vznikla ako reakcia na úspech platformy Raspberry Pi, je ďalšie ovocie, tentokrát ide o Orange Pi. Pri platforme Orange Pi existuje asi 27 modelov, pričom cena týchto modelov je nižšia. Model Orange Pi 4 B obsahuje 6-jadrový procesor s frekvenciou až 2.0 GHz, 4 GB pamäte RAM, 16 GB úložisko, vstavenú komunikáciu WiFi a Bluetooth, Ethernet port, HDMI port, Display port, USB porty a 40 GPIO pinov. [13]



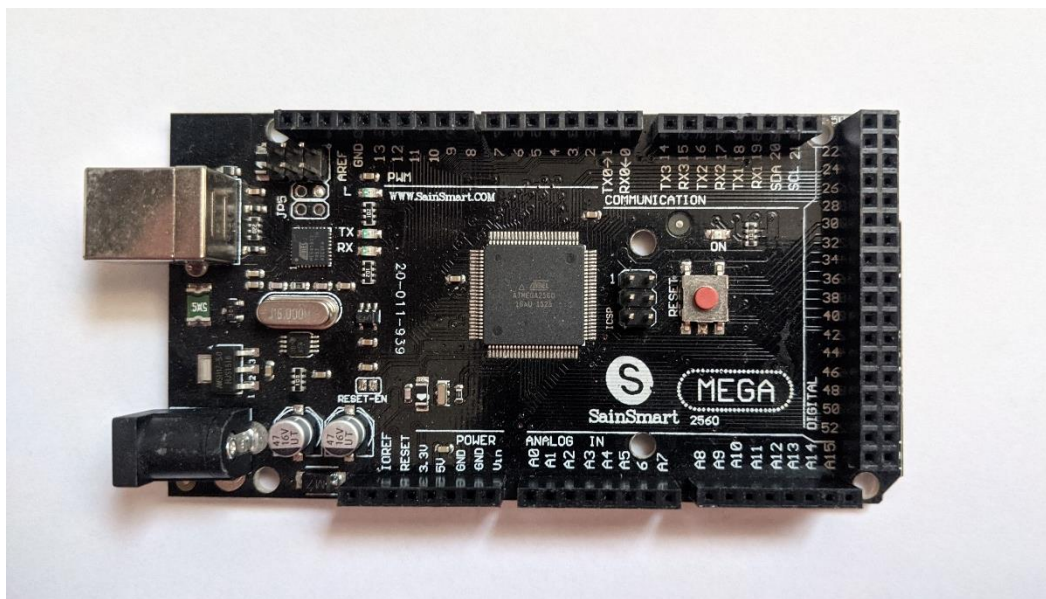
Obr. 7: Orange Pi 4b

Zdroj: <https://all3dp.com/2/orange-pi-4b-board-review-specs/>

1.8 Platforma Arduino

Arduino nie je mikropočítač, tak ako Raspberry Pi, ale ide o mikrokontrolér. Pod pojmom Arduino rozumieme open-source platformu, ktorá je založená na báze jednoduchého vstupu a výstupu, ale taktiež vývojové prostredie pre túto platformu. Vývojové prostredie používa programovací jazyk Processing, ktorý je použitý preto, lebo zahŕňa vlastný editor. Arduino dosky sa dajú zakúpiť zmontované a pripravené k použitiu, no pre pokročilejších je možné si dosku zostaviť samostatne.

Vývoj Arduina započal v roku 2005 v Taliansku, pričom dôvodom na vývoj bola skutočnosť, že do tej doby existovali na trhu iba drahé dosky BASIC Stamp. Počas šiestnástich rokov od začiatku vývoja sa predali státisíce vývojových dosiek. Keďže Arduino je platforma typu open-source, tak okrem oficiálnych dosiek je možné si zaobstaráť aj dosky od iných výrobcov. Napriek tomu, že ide o neoficiálne a neoriginálne typy vývojových dosiek, tak klony sú plne funkčné a spolupracujú s vývojovým prostredím rovnako ako originálne dosky. [10]



Obr. 8: Klon dosky Arduino Mega 2560

Zdroj: Vlastné spracovanie

1.8.1 Typy Arduino dosiek

Na každej doske sa nachádza hlavná časť, procesor. Výrobcom procesoru je spoločnosť Atmel. Na trhu existujú rôzne verzie dosiek.

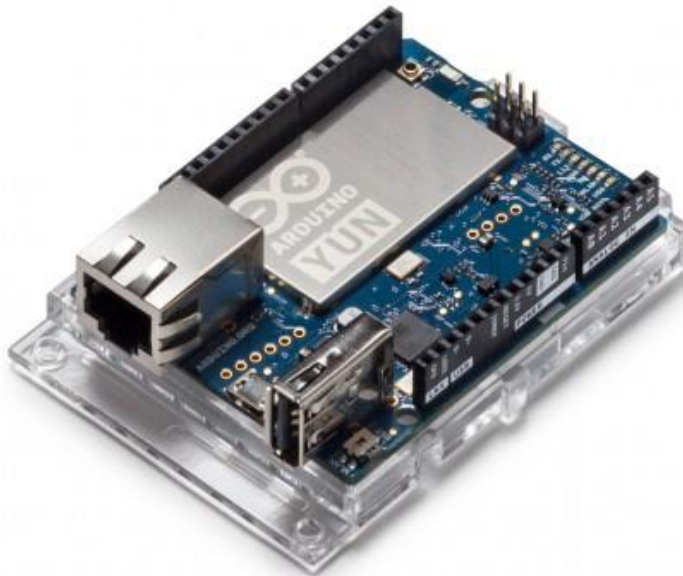
Najpoužívanejšou doskou je Arduino Uno, ktorá obsahuje USB port, pomocou ktorého sa do nej nahráva program. Existuje aj jej modifikácia, Arduino Ethernet, ktorá obsahuje Ethernetový port namiesto USB portu.

Na dosku Arduino Uno v nadväznosti vznikla doska Arduino Yún. Oproti verzii Uno obsahuje navyše čip Atheros AR9331, na ktorom je možné spustiť odľahčenú verziu OS Linux Linino, microUSB port, pre požiadavky OS aj klasický USB port a Ethernet port.

Ďalšou podobnou doskou ako Uno je aj doska Arduino Leonardo. Rozdiel medzi týmito doskami je len v použitom čipe a microUSB konektore, pričom Leonardo používa čip ATmega32u4 (Arduino Uno používa čip ATmega328P).

Pri ďalšom vývoji, kedy sa začali do dosiek dostávať výkonnejšie čipy, ktoré sú aj rozmernejšie, bolo potrebné zväčšiť aj dosku. Doska Arduino Mega 2560 vznikla predĺžením základnej dosky, Arduino Uno. Keďže sa zväčšila plocha dosky, tak sa na dosku dostalo aj viac pinov. Arduino Mega 2560 je vďaka väčšiemu čipu a tým pádom aj väčšej

pamäti pre ukladaný program často používaná pre väčšie projekty, napr. robotické autíčka alebo projekty, v ktorých sa používa dotykový displej.



Obr. 9: Doska Arduino Yún

Zdroj: <https://store.arduino.cc/>

Pri projektoch, ktoré si vyžadujú čo najmenšie rozmery sa používajú dosky Arduino Mini, Arduino Micro a Arduino Nano. Najmenšia doska, verzia Mini, však nemá USB port a preto na nahratie programu do dosky je potrebné použiť externý prevodník. Ako odpoveď na tento nedostatok bola vyvinutá doska Arduino Nano, ktorá už obsahovala mini USB port, takže odpadla potreba použitia prevodníku. Doska Arduino Micro vie vďaka čipu ATmega32u4 odosielať príkazy do počítača, ktoré simulujú posun myši alebo stláčanie kláves na klávesnici a používajú sa na projekty herných ovládačov. [10]

1.8.2 Sériová komunikácia

Nahrávanie programu do dosky prebieha prostredníctvom USB. V prípade, ak doska, na ktorú chceme nahráť program neobsahuje prevodník, tak je potrebné ho pripojiť medzi Arduino a USB kábel.

Okrem nahrávania programu do dosiek je sériová komunikácia využívaná aj na komunikáciu medzi Arduino a počítačom, prípadne mobilom, kam sa odosielať namerané hodnoty, aby boli čitateľné, prípadne na odosielanie príkazov zo strany užívateľa. Okrem komunikácie pomocou USB konektora je možné na sériovú komunikáciu použiť piny z dosky, ktoré majú označenie Tx a Rx. Pomocou týchto pinov je možné pripojiť Bluetooth mobil alebo ďalšiu dosku Arduino. [10]

1.8.3 *GPIO piny*

Digitálne piny, ktoré sú na doske označené číslicami, prípadne textom DIGITAL alebo DIGITAL PWM môžu mať dva stavy. Stav LOW a HIGH. Ak používame tieto piny ako digitálne vstupy, tak v prípade, že napätie na pine je väčšie ako 3 volty, tak sa vyhodnotí, že na vstupe je hodnota HIGH (vysoká). Ak je veľkosť napätia nižšia ako 2 volty, tak sa vyhodnotí, že na vstupe je hodnota LOW (nízka). V prípade, že tieto piny využijeme ako digitálne výstupy, tak v prípade nastavenia hodnoty LOW je na pine výstupné napätie 0 voltov a v prípade nastavenej hodnoty HIGH je na pine napätie 5 voltov.

Pri pinoch, ktoré sú označené ako PWM (alebo DIGITAL PWM) Arduino generuje PWM signál, ktorý je náhradou analógového signálu, pričom sa na výstupnom pine strieda napätie o hodnote 5 V a 0 V.

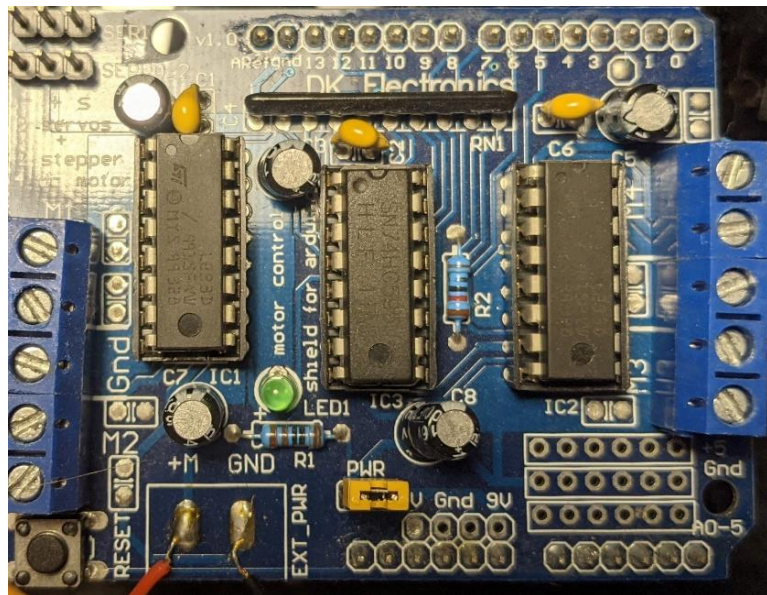
V prípade, že chceme k Arduino pripojiť snímač, ktorý používa analógový výstup, použijeme analógové vstupy, ktorými dosky Arduino disponujú. Tieto vstupy sú priamo na doske označené veľkým písmenom A a poradovým číslom vstupu. [10]

1.8.4 *Shieldy*

Pre rozšírenie funkčnosti dosky Arduino je možné použiť rôzne druhy shieldov. Pod pojmom shield rozumieme modulárne dosky plošných spojov, ktoré sa pripájajú na dosku Arduino a dodávajú doske rozšírené možnosti použitia.

Základné shieldy poznáme:

- Ethernet shield – obsahuje port RJ-45 na prenos dát do rýchlosti 100 Mb/s a slot na microSD kartu.
- Motor shield – použiteľnosť Arduina je možné rozšíriť na používanie krokových motorčekov, ktoré môžu byť použité na otváranie okien, pohyb autíčka alebo pohyb robota. Pri použití shieldu je možné použiť až štyri motorčeky s napätím 5 až 12 voltov.



Obr. 10: Motor shield

Zdroj: Vlastné spracovanie

- WiFi shield – podpora štandardu 802.11 b/g a pracovná frekvencia je 2,4 GHz, shield obsahuje vstavanú anténu a rovnako ako Ethernet shield slot na microSD kartu.
- GSM shield – vhodný na jednoduchšie IoT projekty, podporuje komunikáciu prostredníctvom GPRS a SMS správ.
- Relé shield – používaný v zapojeniach, v ktorých sa ovláda elektrický obvod, používa sa na automatické ovládanie zariadení, ktoré sú pripojené v inom elektrickom obvode ako doska Arduino. [10]

1.9 Programovací jazyk Python

Programovací jazyk Python vznikol v roku 1991 a nachádza sa medzi top desiatimi programovacími jazykmi. Aj keď sa to nezdá, ale programy a služby napísané v Pythone používame denne. Ide napríklad o YouTube, Google, Netflix a ďalšie.

Jazyk Python vieme nájsť v rôznych výpočtových prostrediach, pričom ide napríklad o:

- príkazový riadok na monitore alebo v termináli,
- grafické rozhranie používateľa, čo zahŕňa aj web,
- mobilné zariadenia (smartfóny a pod.),
- web na stranách servera aj klienta,
- vstavané zariadenia,
- na populárnych veľkých stránkach na backendových serveroch,
- v cloude.

Použitie Pythonu ako programovacieho jazyka je možné od jednorazových skriptov až po systémy, ktoré obsahujú milióny riadkov napísaného programu. Python je univerzálny programovací jazyk, ktorý je na vysokej úrovni. Dizajn tohto jazyka je veľmi dobre čitateľný, čo je veľká výhoda pri jeho použití z dôvodu, že počítačový program sa napíše iba jedenkrát, no následne ho číta a reviduje veľké množstvo ľudí. Táto vlastnosť okrem iného výrazne uľahčuje učenie a zapamätanie si postupností. Oproti statickému jazyku má Python výhodu v tom, že program zaberá oveľa menej miesta. Vďaka štúdii, ktorá preukázala, že väčšina programátorov produkuje približne rovnaký počet riadkov kódu za deň, pričom nezáleží, o aký programovací jazyk ide, dokáže zvýšiť efektivitu približne dvojnásobne. Od ostatných programovacích jazykov sa Python odlišuje tým, že ľudí baví programovať v jazyku Python najmä z toho dôvodu, že im v iných jazykoch chýba nejaká vlastnosť Pythonu.

Nie vždy je Python najvhodnejším jazykom pre úplne každú situáciu. Jedným z dôvodov je to, že predvolene nie je nainštalovaný na každom zariadení. Python je síce pre väčšinu aplikácií rýchly, no pri niektorých náročnejších aplikáciách nemusí byť dostatočne rýchly, ako sa od neho očakáva. Ide napríklad o program, ktorý využíva väčšinu času výpočtom vecí na centrálnom procesore. Takýto program môže fungovať rýchlejšie v jazyku Java, C alebo C++.

Veľmi dôležité je spomenúť najväčší problém, s ktorým je možné sa stretnúť pri programovaní jazyku Python. Ide o problém, že existujú dve verzie Pythonu, Python 2 a Python 3. Python 2 je býva predinštalovaný na počítačoch s OS Linux a na počítačových zariadeniach Apple. Ide o veľmi dobrý jazyk, no nie dokonalý. Tak, ako vo viacerých oblastiach, aj pri programovacích jazykoch sa objavujú ľahko opraviteľné kozmetické chyby, no aj ťažko opraviteľné. V prípade opravy tých ťažko opraviteľných chýb sa dostávame do situácie, že sú opravy nekompatibilné. Aplikácie napísané po oprave chýb nebudú fungovať v starom systéme Python, a naopak, staré programy napísané pred opravou nebudú funkčné v novom systéme. Guido van Rossum, tvorca Pythonu sa spolu s ďalšími vývojármi rozhodli spojiť ťažké opravy dokopy, a nazvať ich Python 3. V súčasnej verzii je na trhu dostupná už deviata aktualizácia Pythonu 3, verzia 3.9.0. [5]

2 Cieľ práce

Cieľom diplomovej práce je poskytnutie komplexného úsudku k danej problematike. Taktiež sme si vytýčili cieľ navrhnuť praktické využitie konceptu Internetu vecí. Na jeho dosiahnutie je potrebné náš cieľ rozdeliť na niekoľko čiastkových cieľov.

Na úvod je potrebné definovať pojmy, ktoré úzko súvisia s Internetom vecí, základnú myšlienku Internetu vecí a použitých technológií, keďže neskôr budeme vychádzať z teoretického základu.

Ako prvé si určíme, čo od nášho projektu očakávame, ktorú zo sietí chceme využiť na komunikáciu zariadenia s používateľom. Nasledovať bude krok, v ktorom bude potrebné si zadať, aké hodnoty chceme získavať, aké snímače použijeme na získanie týchto hodnôt a z finančného hľadiska aj spôsob simulácie iných zariadení.

Pre vhodný spôsob simulácie iných zariadení potrebujeme dostatočne ukázať, že nami zhotovené zariadenie bude schopné jednoznačne ukázať, že krok, ktorý sa snažíme simulovať aj nastal, to znamená, že simulácia je úspešná. Taktiež je pri simulácii dôležité ukázať, kedy bude simulácia ukončená.

Po dôslednej analýze a zohľadnení našich požiadaviek sme si zvolili, že použijeme dve najznámejšie platformy, Arduino a Raspberry Pi. Pre simuláciu iných zariadení sme z finančného hľadiska a jednoznačnej identifikovateľnosti zvolili použitie LED kontroliek, ktoré splnili naše kritériá.

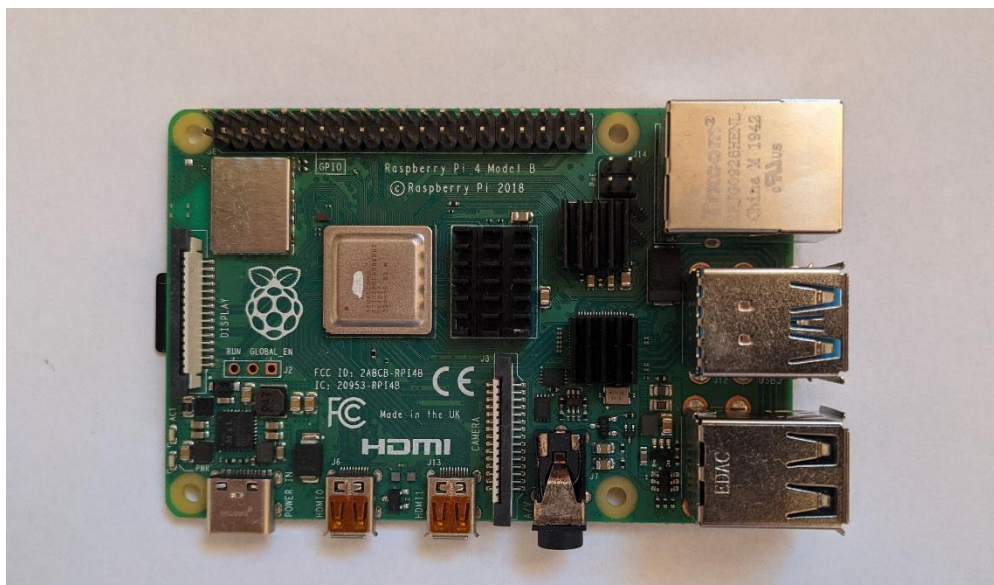
3 Metodika práce

Na základe oboznámenia sa s problematikou Internetu vecí z dostupnej literatúry a s ohľadom na ustavičnú diskusiu a neustálym rozpínaním sa IoT vo svete ohľadom problematiky sme sa usilovali v našej práci zdôrazniť základné princípy Internetu vecí, ktoré vytvárajú základ pre zhotovenie praktickej časti záverečnej práce. Pri praktickej časti sme mimo využitia dostupnej literatúry využili aj praktické skúsenosti s programovaním v rôznych programoch, ktoré som získal v priebehu štúdia. Inšpirovali sme sa aktuálnou situáciou, ktorá vznikla v súvislosti s pandémiou koronavírusu, kedy automatizácia aj pri základných úkonoch získala dôležité svoje miesto v našich životoch. Po analýze situácie sme dospeli k úsudku, že náš návrh na automatizáciu má praktické využitie, pričom by v dnešnej dobe mohol priblížiť podstatu IoT rôznym ľuďom.

3.1 Voľba modulov

Na praktické vyhotovenie našej práce sme si z dôvodu širokej dostupnosti, jednoduchšieho rozširovania a rôznych druhov hardvéru vybrali dve platformy, Raspberry Pi a Arduino. Vďaka tomu, že obe platformy sú značne rozšírené, majú vysokú podporu v internetových komunitách, čo je v prípade určitej nejasnosti vhodné, keďže v rámci komunit nie je zložité požiadať o radu. Taktiež vďaka tomu, že ide o dve najrozšírenejšie platformy, potrebné snímače a hardvérové súčiastky sú dostupnejšie a sú kompatibilné.

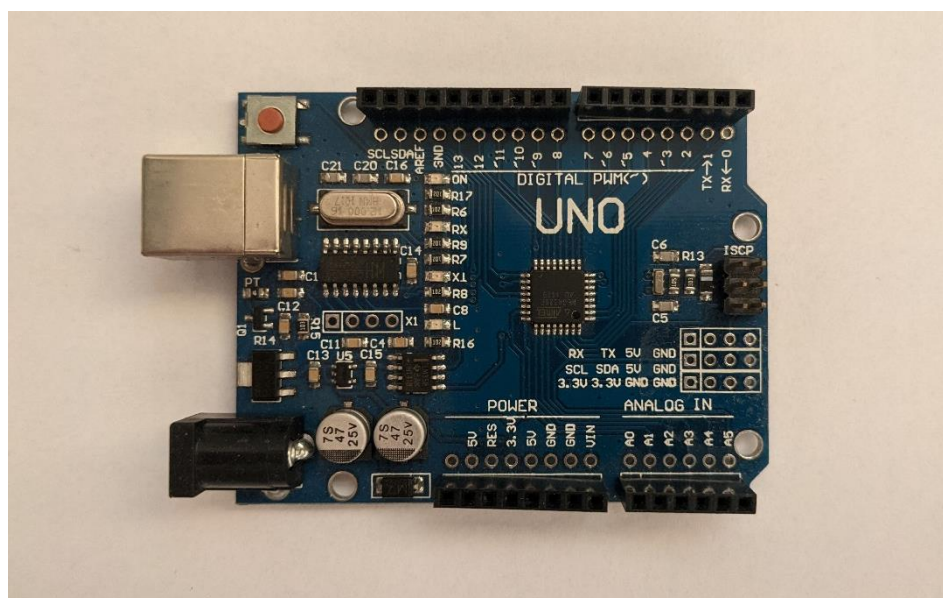
Z platformy Raspberry Pi sme si vybrali najnovší modul, Raspberry Pi 4 Model B s kapacitou pamäte RAM 2 GB, ktorá je pre náš projekt kapacitne postačujúca. Prostredníctvom programu Raspberry Pi Imager sme si na microSD kartu museli nahráť operačný systém Raspbian v najnovšej verzii. Raspbian je odvodený zo systému Debian.



Obr. 11: Modul Raspberry Pi 4 Model B

Zdroj: Vlastné spracovanie

Z platformy Arduino sme si zvolili dosku Uno, ktorej vhodnosť pre náš projekt sa osvedčila už pri iných projektoch. Naša doska UNO nie je originálna doska Arduino Uno, ale ide o dosky neznámeho výrobcu, pričom aj na tejto doske sa nachádza čip ATmega328P, ktorý vyrába spoločnosť Atmel.



Obr. 12: Modul UNO neznámeho výrobcu

Zdroj: Vlastné spracovanie

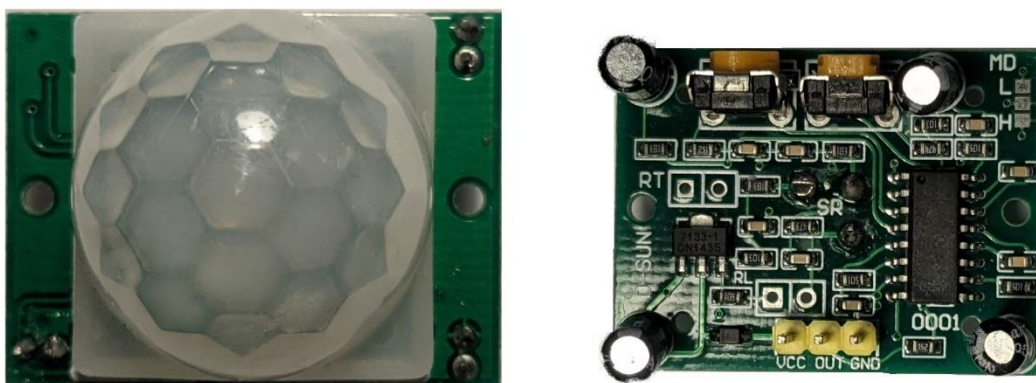
3.2 Voľba snímačov

Pre projekt, ktorý sme sa rozhodli skonštruovať sme usúdili, že na použitie budú vhodné snímače na sledovanie teploty, vlhkosti a pohybu. Snímače DHT11 a DHT22 sme sa rozhodli použiť na meranie teploty a vlhkosti vzduchu. Na sledovanie pohybu sme použili snímač PIR od spoločnosti D-Sun.

3.2.1 PIR snímač Hc-Sr501

Snímač PIR funguje na princípe detekcie infračervenej vlnovej dĺžky žiarenia, ktorá nie je viditeľná ľudským okom. Snímač je možné nastaviť tak, že v prípade, že zaznamená infračervené žiarenie, tak si ho môže interne „zapamätať“ na minimálne tri sekundy a maximálne na päť minút.

Pracovné napätie snímača je 3V až 12V pri jednosmernom prúde. Vstup na napájanie je na snímači označený ako VCC. Pod označeným GND rozumieme záporný pól napájania, pin OUT, ktorý sa nachádza v strede medzi VCC a GND, je digitálny výstup. Na pin OUT je v prípade zachyteného pohybu odosielaná hodnota HIGH, čo znamená 3,3V. V prípade, že pohyb zachytený nebol alebo bol ukončený, hodnota na pin-e je LOW, čiže 0 voltov.



Obr. 13: Vrchný a spodný pohľad na PIR snímač

Zdroj: Vlastné spracovanie

3.2.2 Snímač DHT11

V prípade snímača DHT11 ide o digitálny snímač, ktorý je schopný merať teplotu od 0°C až po 50°C, pričom odchýlka merania je $\pm 2^\circ\text{C}$. Tento snímač dokáže okrem merania teploty merať aj vlhkosť okolitého vzduchu v rozmedzí od 20% po 90% relatívnej vlhkosti. Presnosť merania je nižšia ako pri meraní teploty, odchýlka sa uvádza v rozmedzí 5% relatívnej vlhkosti. Samotný snímač má štyri piny, ktoré sú označené ako VCC – kladný pól napájania, v rozmedzí 3 až 5,5V, GND – záporný pól napájania, DATA – digitálny výstup a NC – z anglického Not Connected, čiže nezapojený. Z dôvodu obmedzenia rušenia sa k výstupnému portu DATA odporúča pripojiť rezistor s hodnotou 1800 Ω , čo sme v našom prípade nemuseli spraviť, keďže náš snímač disponuje vstavaným rezistorom. Na našom snímači sa okrem vstavaného rezistoru nachádza aj LED indikátor, ktorý signalizuje, že snímač je aktívny a meria teplotu.



Obr. 14: Snímač DHT11 s LED kontrolkou

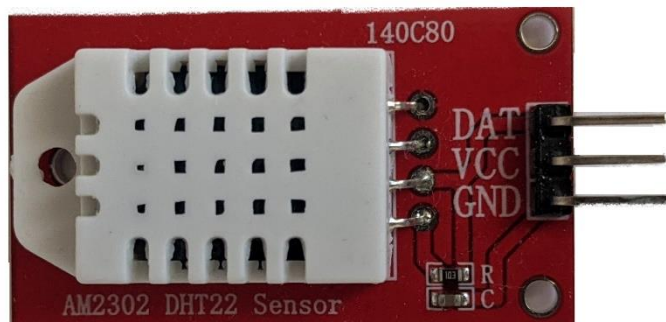
Zdroj: Vlastné spracovanie

3.2.3 Snímač DHT22

Snímač DHT22 je taktiež digitálny snímač, ktorého prednosťou oproti snímaču DHT11 je väčší rozsah pri meraní hodnôt a zároveň aj nižšia odchýlka, čiže hodnoty sú presnejšie a bližšie k realite. Snímač disponuje taktiež štyrmi vývodmi, ktoré majú označenia VCC, DATA, NC a GND. Zapojenie je rovnaké, pričom je rozdiel iba pri napájaní, kedy má snímač minimálnu požiadavku na vstupné napätie 3,3V.

Snímač má rozsah merania teploty od -40° po $+80^\circ$ Celzia, pričom odchýlku má iba 0,5 stupňa. Rozsah merania relatívnej vlhkosti vzduchu má od 0% do 100% pri odchýlke

$\pm 2\%$ relatívnej vlhkosti. V našom projekte sme si zvolili použitie snímača DHT22 s číslom modelu AM2302, ktorý má na doske plošných spojov už integrovaný rezistor, ktorý obmedzuje rušenie.



Obr. 15: Snímač AM2302 DHT22

Zdroj: Vlastné spracovanie

3.3 Výber ďalších prvkov

Z dôvodu vyšších nákladov a zlej dostupnosti zariadení, sme sa rozhodli simulovať použité zariadenia pomocou LED diód. Tie sme použili tak, že nami zvolené platformy Arduino alebo Raspberry Pi vyšlú pokyn na otvorenie okna alebo spustenie vetráku tak sa rozsvieti príslušná LED kontrolka, ktorá indikuje, že sa vykonal impulz, ktorý sme naprogramovali. V našom prípade sme si zvolili štyri LED diódy, pričom každá je inej farby, aby sme vedeli rozoznať, o aký pokyn sa jedná. K LED kontrolke pripojenej k doske Arduino sme nepotrebovali okrem samotnej LED diódy žiadne ďalšie prvky, no pri platforme Raspberry Pi sme ku každej LED dióde pripojili rezistor s hodnotou $470\ \Omega$. Tento odpor sme pripojili z dôvodu ochrany LED diódy pred jej zničením.

3.4 Voľba komunikačnej technológie

V našom projekte sme okrem hardvérových častí, ktoré budú nainštalované na mieste použitia, rozmýšľali nad voľbou komunikačnej technológie, ktorou budeme vzdialene informovať používateľa o dianí. Z dôvodu, že doska Raspberry Pi 4 má integrovaný Bluetooth a WiFi čip, sme zúžili výber na tieto dve komunikačné technológie. Po analýze

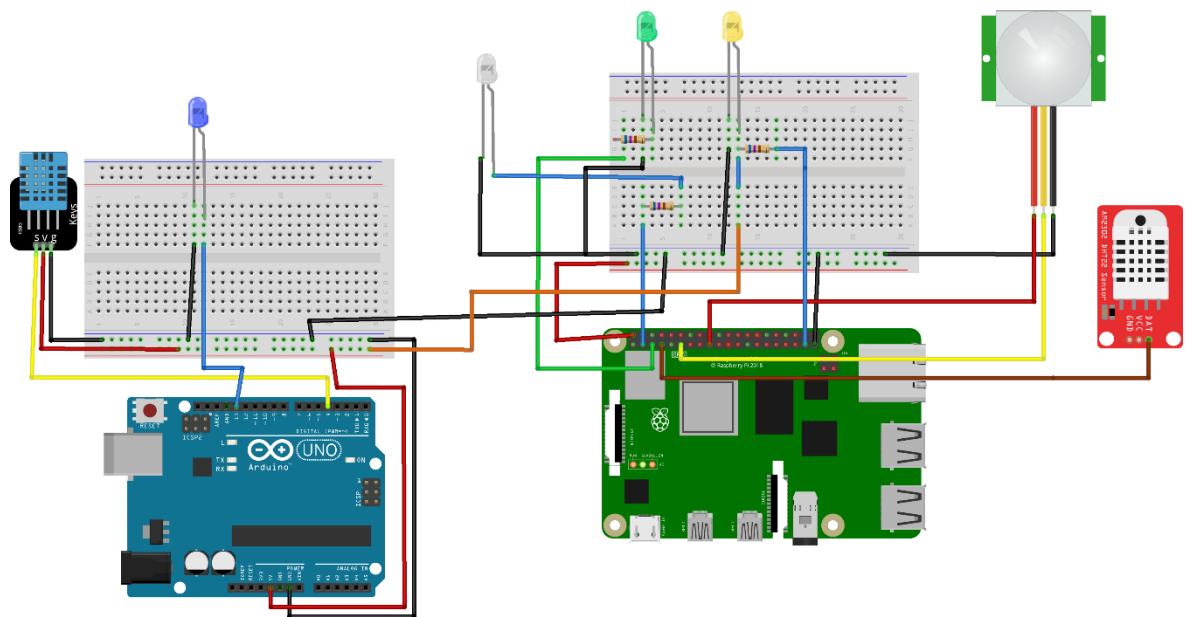
sme dospeli k záveru, že využijeme WiFi pripojenie, ktoré je bezplatné a dokážeme pomocou neho odosielať e-mailly. Na odosielanie e-mailov sme si zvolili službu Gmail, pričom v nastaveniach služby bolo potrebné nastaviť prístup k menej zabezpečeným službám.

3.5 Schéma zapojenia

Po analýze a výbere platformy a vhodných snímačov sme sa pustili do návrhu zapojenia. Na zapojenie sme plánovali použiť nespájkované pole, tzv. breadboard, ktoré nám umožňuje rýchle a nedeštrukčné úpravy. Z tohto dôvodu sme toto zapojenie realizovali aj pri schéme zapojenia, aby nám schéma sedela s reálnym zapojením.

Na vytvorenie schémy sme použili program Fritzing, pričom sme do programu museli naimportovať knižnice, ktoré obsahujú naše snímače v grafickom zobrazení. Všetky súčiastky sme pripojili k breadboardu, ktorý sme následne prepojili s doskami Arduino a Raspberry.

Pre lepšiu orientáciu sme použili dve nespájkované polia, pre každú dosku vlastné, aby sme ich vedeli samostatne skontrolovať a taktiež v prípade chýb vedeli zistiť, kde sa nachádzajú a ako ich eliminovať. Schémy sme upravovali, pokiaľ sme sa nedostali k ideálnemu zapojeniu.



fritzing

Obr. 16: Schéma zapojenia projektu

Zdroj: Vlastné spracovanie

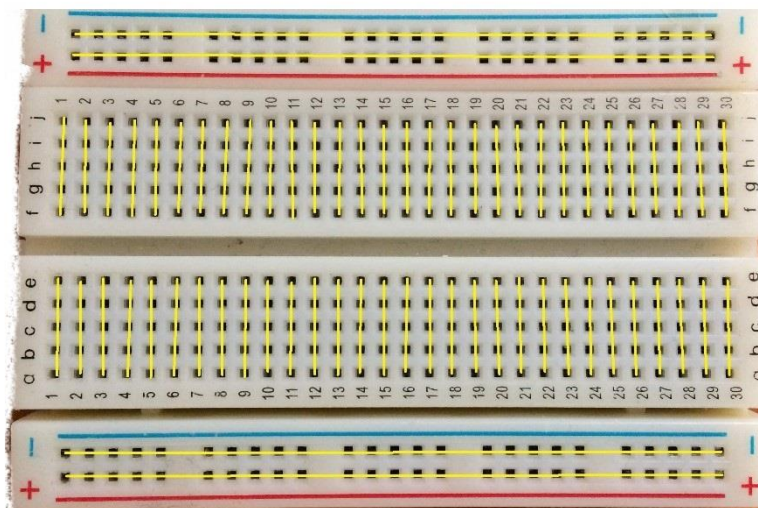
4 Výsledky práce

Po tom, čo sme si zvolili platformy a v rámci platforiem dosky, komunikačnú technológiu, snímače a zvyšné prvky, tak sme si navrhli praktickú časť zadania. V návrhu sme použili vhodné moduly, snímače a prvky, ktoré spĺňali naše požiadavky tak, aby bola zabezpečená funkčnosť nášho konceptu. Zvolili sme si dva snímače teploty, pre každú použitú platformu samostatný snímač, PIR snímač a LED kontrolky.

4.1 Zostavenie projektu

Pri zostavovaní nášho projektu, resp. pri prototypu projektu sme sa rozhodli využiť nespájkované pole. Tento typ zapojenia sme zvolili z toho dôvodu, že v prípade potreby vieme rýchlo a celkom jednoducho zmeniť zapojenie, prípadne z jedného napájania vieme použiť viacero vývodov.

Schéma zapojenia kontaktov v nespájkovanom poli je uvedená na obrázku č. 16, ktorý sa nachádza nižšie:



Obr. 17: Schéma prepojenia plôch na nespájkovanom poli

Zdroj: Vlastné spracovanie

4.1.1 Zapojenie časti s modulom Raspberry Pi

Pri začiatku prepájania jednotlivých súčiastok sme postupovali spôsobom, že sme si k modulu Raspberry Pi pripojili breadboard. Z GPIO portov na doske Raspberry Pi sme si pripojili napájanie (stále napätie +5V, ktoré je na kontakte GPIO č. 2) na kladný pól breadboardu. Po tomto kroku sme na breadboard umiestnili LED diódy. Ku každej anóde LED diódy sme umiestnili rezistor s odporom 470 Ω . Katódy sme prepojili káblikom k negatívnemu pólu na breadboade. Po tomto kroku sme prepojili nezapojenú časť rezistorov k doske Raspberry Pi, pričom sme využili porty označené ako GPIO 2 – pre bielu LED, GPIO 3 – pre zelenú LED a GPIO 26 – pre žltú LED.

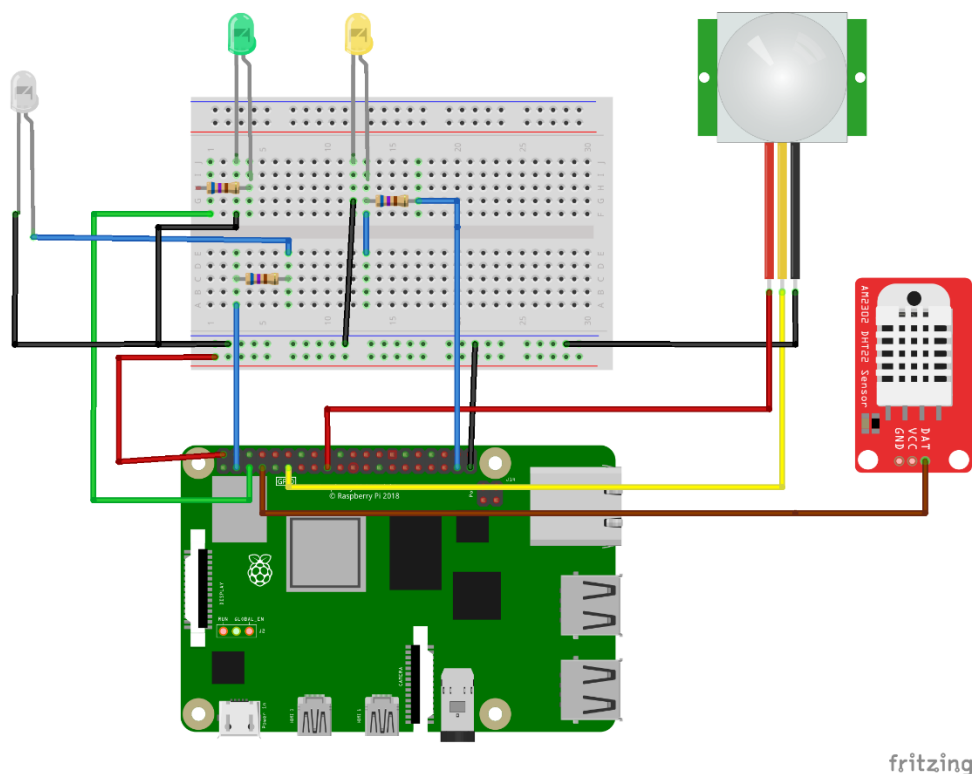
Bielu LED sme použili ako signalizáciu pri zachytení pohybu a zároveň ako simuláciu spustenia plašiča zvery prípadne zapnutie fotopasce.

Zelenú LED sme využili na simulovanie otvorenia okna a/alebo spustenie ventilácie v prípade, že teplota vzduchu stúpne nad nami nastavenú hodnotu.

Žltá LED nám indikuje, že sme vyslali signál na zapnutie modulu Arduino.

Po zapojení LED sme pokračovali a pripojili PIR snímač. PIR snímač má tri piny, ktoré sme pripojili nasledovne: k pinu označenému ako VCC sme pripojili kladný pól, ku GND záporný pól z breadboardu a pin OUT sme prepojili s portom GPIO 17, ktorý sa nachádza na doske Raspberry Pi.

Keď sme mali zapojený PIR snímač, tak sme sa pustili do zapojenia posledného prvku k doske Raspberry Pi, snímača DHT22. Snímač sme zapojili pinmi VCC a GND ku nespájkovanému poľu. Pin DATA sme pripojili priamo k doske na pozíciu GPIO 4.

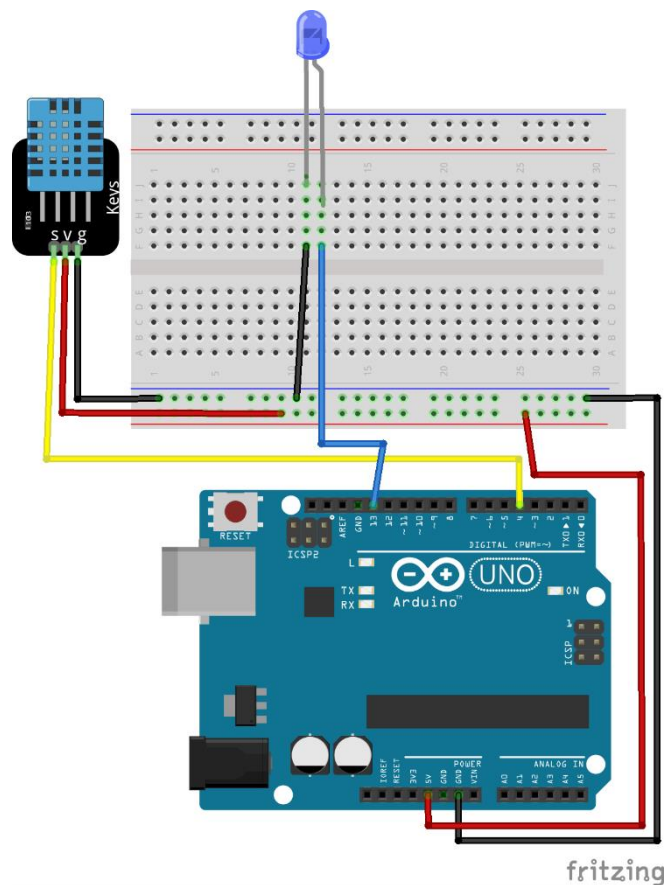


Obr. 18: Zapojenie prvej časti projektu

Zdroj: Vlastné spracovanie

4.1.2 Zapojenie časti s modulom Arduino

Po zapojení všetkých častí v module Raspberry Pi sme si obdobným spôsobom zapojili aj modul Arduino. Z dôvodu ľahšej manipulácie sme použili tiež nespájkované pole. Dosku Arduino sme s breadboardom spojili tak, že sme prepojili kladný pól breadboardu s portom na doske Arduino, ktorá má označenie 5V. Záporný pól sme prepojili s portom GND. Na kladný a záporný pól na breadboarde sme pripojili teplotný snímač DHT11, pričom nezapojený pin DATA sme pripojili priamo k doske na PWM pin s číslom 4. Po pripojení snímača sme pripojili aj LED diódu simulujúcu spustenie zavlažovania. LED diódu sme pripojili tak, že jej katódu sme pripojili k zápornému pólu na nespájkovanom poli a katódu sme pripojili k pinu 13 na doske Arduino.

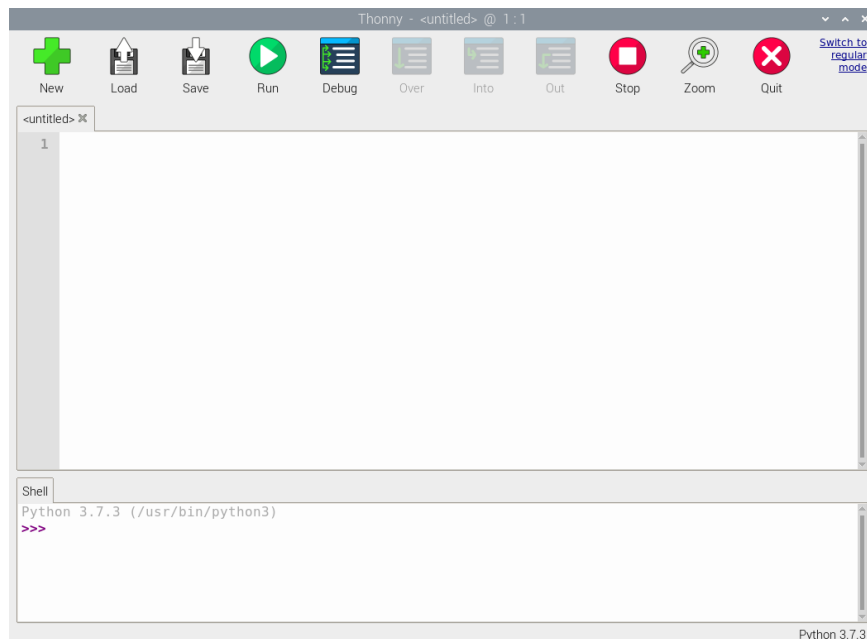


Obr. 19: Zapojenie druhej časti projektu

Zdroj: Vlastné spracovanie

4.2 Programovanie modulu Raspberry Pi

Po prepojení hardvérových častí na module Raspberry Pi sme sa dostali k programovaniu. V prípade platformy Raspberry Pi sa programuje priamo v mikropočítači, takže sme si ho pripojili k monitoru, zapojili si myš, klávesnicu a modul sme pripojili k zdroju napájania. Po úvodnom nastavení počítača sme sa dostali na úvodnú obrazovku. V menu sme si vybrali program Thonny Python IDE, ktorého prostredie po spustení vyzeralo nasledovne:



Obr. 20: Prostredie programu Thonny Python IDE

Zdroj: Vlastné spracovanie

Po otvorení programu sme pristúpili k písaniu programu, pričom na začiatku sme potrebovali nainštalovať knižnice a zadať prístup k e-mailovej službe. Na prístup k e-mailovej službe sme použili SMTP server s portom 587. Následne sme potrebovali napísať prihlasovacie meno a heslo do e-mailu, z dôvodu ochrany pred zneužitím sme tieto údaje v ukážke kódu zmenili.

```
import smtplib
import RPi.GPIO as GPIO
import time
import Adafruit_DHT #dht_knižnica_snímč teploty
SMTP_SERVER = 'smtp.gmail.com' #E-mailovy server
SMTP_PORT = 587 #Server Port
GMAIL_USERNAME = 'vzorovy_mail@gmail.com' #e-mail
odosielatela
GMAIL_PASSWORD = 'Heslo123' #heslo do e-mailu odosielatela
```

Ďalším krokom sme si zadefinovali GPIO porty, ktoré sme si podľa predchádzajúceho schematickeho návrhu zvolili. Ako vstupný pin pre informáciu z PIR snímaču sme si zvolili pin s číslom 11 a ako výstupné porty sme použili porty s označením 3, 5 a 37. Ďalej sme zadefinovali pin 4 ako vstup zo snímača teploty. Pri tejto časti sme si

museli dávať pozor, keďže v prípade programovania s GPIO portami sa používajú poradové čísla portov, no v schéme GPIO sa používajú aj označenia GPIO portov, ktoré sa nezhodujú. Z dôvodu prehľadnosti kódu a ďalšej práce s premennými sme si v poznámke označili, kde sú pripojené.

```
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BOARD)
GPIO.setup(11, GPIO.IN) #vstup z PIR snímaču
GPIO.setup(3, GPIO.OUT) #biela LED
GPIO.setup(5, GPIO.OUT) #zelená LED
GPIO.setup(37, GPIO.OUT) #zltá LED
DHT_SENSOR = Adafruit_DHT.DHT22
DHT_PIN = 4
```

Po zadefinovaní premenných a sme si vytvorili triedu `Emailer`. V tejto triede sme si vytvorili hlavičku e-mailu, v ktorej sme si zadefinovali premenné, ktoré použijeme neskôr. Po vytvorení hlavičky sme napísali časť, ktorá sa pripája k serveru a následne časť, ktorá má na starosť prihlásenie do služby. Ako posledný krok v triede sme vytvorili časť, ktorá odosiela e-maily a ukončuje komunikáciu so serverom.

```
class Emailer:
    def sendmail(self, recipient, subject, content):

        #Vytvorenie hlavičky
        headers = ["From: " + GMAIL_USERNAME, "Subject: " +
subject, "To: " + recipient,
                    "MIME-Version: 1.0", "Content-Type:
text/html"]
        headers = "\r\n".join(headers)

        #Pripojenie k serveru
        session = smtplib.SMTP(SMTP_SERVER, SMTP_PORT)
        session.ehlo()
        session.starttls()
        session.ehlo()

        #Prihlásenie do Gmail-u
        session.login(GMAIL_USERNAME, GMAIL_PASSWORD)
```

```

#Odoslanie Email-u a ukončenie komunikácie
session.sendmail(GMAIL_USERNAME, recipient, headers +
"\r\n\r\n" + content)
session.quit

```

Po zápise triedy sme si zadefinovali, že premenná *sender* zahŕňa triedu *Emler* pomocou príkazu:

```
sender = Emler()
```

V nasledujúcom kroku sme si určili podmienky, ktoré si z dôvodu lepšej prehľadnosti opíšeme postupne. Ako prvú podmienku sme si určili, že ak je na výstupe z čidla pohybu hodnota LOW (t.j. 0V) tak si na konzolu vypíšeme *Bez pohybu* a nastavíme na výstupný pin 3 hodnotu 0V, čo v našom prípade znamená, že sa nevykoná žiadna akcia, čiže LED simulujúca akciu je zhasnutá. V prípade, že výstup zo snímača bude s hodnotou HIGH tak odošleme e-mail na adresu prijímateľa. V tejto časti kódu sme si zvolili text predmetu správy a jej obsah. Predmet správy sme si zvolili výstižne: *Zachytený pohyb na záhradke!*. Do obsahu správy sme okrem informácie o pohybe pridali aj informáciu o dátume a čase zachyteného pohybu. Pre kontrolu odoslania e-mailu sme si na konzolu nechali vypísať informáciu *E-mail odoslaný a Pohyb!*. Taktiež po zachytení pohybu je na výstupný pin č. 3 privedené napätie, ktoré spustí nami zvolenú akciu. Akcia je v našom prípade simulovaná tiež LED kontrolkou, ktorá sa zasvieti. Na tento pin môže byť pripojená fotopasca, ktorá do e-mailu odošle fotografiu, o aký pohyb sa jedná, spustenie svetla na vyplašenie zvery alebo spustenie plašiča divej zvery.

```

while True:
    i=GPIO.input(11)

    if i==0:                                #Ak je výstup z čidla LOW
        print ("Bez pohybu"),i
        GPIO.output(3, 0) #Turn OFF LED
        time.sleep(1)
    elif i==1:                              #Ak je výstup z čidla HIGH
        sendTo = 'testovaci_email@gmail.com' #e-mail
        prijimatela
        emailSubject = "Zachyteny pohyb na zahradke!"

```

```

        emailContent = "Na zahradke bol zachyteny pohyb v
case: " + time.ctime()
        sender.sendmail(sendTo, emailSubject, emailContent)
        print("E-mail odoslany")
        print ("Pohyb!"), i
        GPIO.output(3, 1)  #Zapnutie bielej LED
        time.sleep(30)
    elif i!=0:
        GPIO.output(3, 1)  #Vypnutie bielej LED
        time.sleep(30)

```

V poslednej časti nášho programu sme si zadefinovali podmienku zo snímača teploty a vlhkosti. Ako hraničnú hodnotu pre teplotu sme si zvolili 28°C, keďže optimálna teplota v skleníku je približne 25°C. V prípade, že teplota dosiahne alebo presiahne nami zvolenú hraničnú hodnotu, tak na výstupný pin číslo 5 vyšleme signál, ktorý zabezpečí otvorenie okna a/alebo spustenie ventilátora. V našom prípade sme si funkčnosť nasimulovali rozsvietením LED kontrolky zelenej farby. Pre overenie nám program do konzoly vypíše namerané hodnoty teploty a vlhkosti a zároveň informáciu, že okno je otvorené. Ak teplota klesne pod 28°C, tak na pin systém prestane vysielat' signál, LED kontrolka zhasne a na konzole sa nám vypíše informácia o zatvorení okna.

```

elif temperature >=28:
    print ("Teplota: {0:0.1f}*C  Vlhkosť:
{1:0.1f}%".format(temperature, humidity))
    GPIO.output(5, 1)  #otvorenie okna (+3.3V na PIN 5)
    print("Okno otvorene")
else:
    GPIO.output(5, 0)  #zatvorenie okna (0V na PIN 5)
    print("Okno zatvorene")
    print ("Teplota: {0:0.1f}*C  Vlhkosť:
{1:0.1f}%".format(temperature, humidity))

```

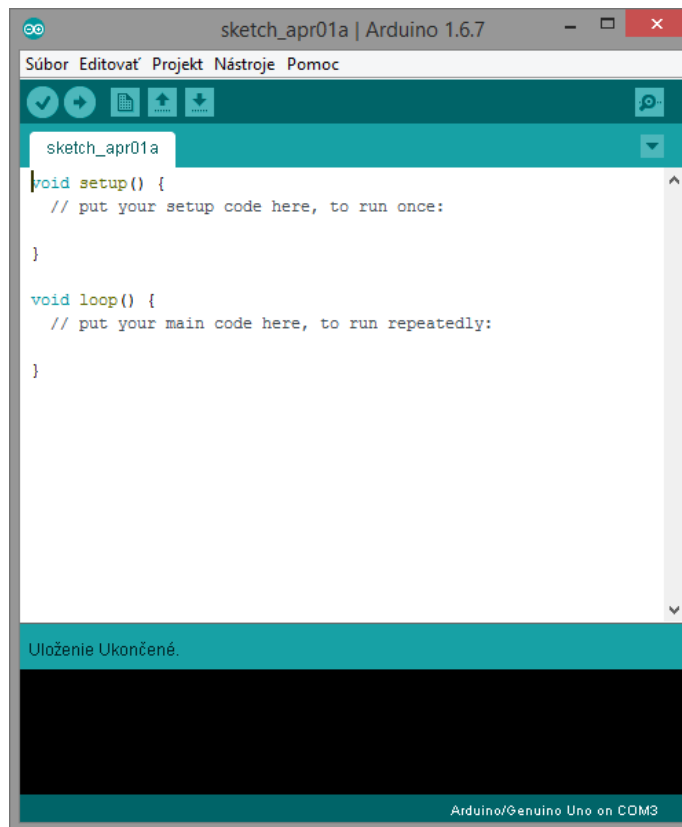
Ďalej sme pokračovali v meraní relatívnej vlhkosti vzduchu, ktorú vieme využiť na to, aby sme zvolili hraničné hodnoty, ktoré nám dajú informáciu o tom, či je potrebné začať zavlažovať alebo nie. Nami zvolená hodnota 60% znamená, že v prípade, ak klesne vlhkosť pod túto hodnotu, zapneme napájanie doske Arduino, ktorá podľa svojho snímača rozhodne, či je potrebné začať zavlažovanie alebo nie. Táto časť programu vyzerá nasledovne:

```
if humidity < 60:
    GPIO.output(37, 1)  #spustenie arduina (+3.3V na PIN
37)
    print("Arduino spustene")
else:
    GPIO.output(37, 0)  #vypnutie arduina (0V na PIN 37)
    print("Arduino vypnute")
```

Počas programovania sme narazili aj na drobné problémy. Najhlavnejší problém bol v tom, že v programovacom jazyku Python záleží na správnom odsadení jednotlivých prvkov kódu, čo nám robilo v prípade rôznych úprav problémy, no nakoniec sa nám ich podarilo opraviť a vyriešiť.

4.3 Programovanie modulu Arduino

Po naprogramovaní hlavnej časti s modulom Raspberry Pi sme pokračovali v práci a začali sme programovať druhú časť práce, modul Arduino. Pre programovanie tejto časti sme použili aplikáciu Arduino IDE vo verzii 1.6.7, ktorá sa používa na programovanie, kompiláciu a následné nahratie programu do dosky Arduino. Prostredie tohto programu vyzerá nasledovne:



Obr. 21: Prostredie programu Arduino IDE

Zdroj: Vlastné spracovanie

Na začiatku programovania sme si zadefinovali premenné, pričom pre výstup sme si zadefinovali pin číslo 13. Tak ako pri časti s modulom Raspberry Pi, aj tu sme využili LED diódu ako kontrolku, resp. simuláciu spustenia zavlažovania. Ďalej sme si zvolili pomocnú premennú, ktorej sme priradili hodnotu 0. Po zadefinovaní premenných sme zahrnuli knižnicu pre snímač teploty DHT11. Po zahrnutí knižnice sme zadefinovali pin 4 ako vstupný pin z čidla teploty a taktiež sme potrebovali určiť, o aký typ snímaču ide.

```
int ledPin = 13;           //zvolenie portu 13 pre anodu
LED
int val = 0;               // premenná pre čítanie
#include <DHT.h>
#define DHTPIN 4
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
```

V ďalšej časti programu sme si zadeklarovali výstup, pričom sme deklaráciu umiestnili do časti void setup (), ktorá sa vykoná iba raz, a to po spustení zariadenia. Najprv

sme si ledPin zadeklarovali ako výstup. Následne sme si nastavili program tak, aby nám hneď po spustení zariadenia vykonal kontrolu, že snímač teploty funguje korektne.

```
void setup() {  
    pinMode(ledPin, OUTPUT);          // deklaracia LED ako vystup  
    Serial.begin(9600);  
    dht.begin();  
    float h = dht.readHumidity();  
    float t = dht.readTemperature();  
    float f = dht.readTemperature(true);  
  
    if (isnan(h) || isnan(t) || isnan(f)) {  
        Serial.println("Chyba citania z cidla DHT11");  
        delay(4000);  
    }  
    else
```

Ďalej sme postupovali tak, že ak je snímač funkčný a údaje z neho vieme prečítať, tak namerané hodnoty teploty a relatívnej vlhkosti vzduchu zobrazíme na konzole, na výstup nastavíme hodnotu LOW a počkáme štyri sekundy. Na začiatku tejto časti kódu sme si zadefinovali premenné pre teplotu a vlhkosť. Časť kódu, obsahuje aj premennú, ktorá vypočítava hodnoty namerané snímačom, konkrétne ide o tieto dve premenné:

```
float hif = dht.computeHeatIndex(f, h);  
float hic = dht.computeHeatIndex(t, h, false);
```

Časť, ktorá vypisuje na konzolu namerané hodnoty:

```
Serial.print("Teplota: ");  
Serial.print(hic);  
Serial.print(" *C \n");  
Serial.print("Vlhkost ");  
Serial.print(h);  
Serial.println(" %");  
digitalWrite(ledPin, LOW);  
delay(4000);
```

Po uplynutí času porovnáme, či nameraná hodnota je väčšia ako 50% relatívnej vlhkosti, tak sa zavlažovanie nespustí, pričom ak je nameraná hodnota nižšia, tak spustíme zavlažovanie. Táto časť programu sa tiež vykoná iba raz.

```
if (h > 55) {  
    digitalWrite(ledPin, LOW);  
}  
else  
    digitalWrite(ledPin, HIGH);      //spustenie  
zavlažovania
```


Keďže predchádzajúca časť nášho kódu sa vykoná iba raz, museli sme si vytvoriť ešte podmienku aj v časti void loop (), ktorá sa vykonáva opakovanie. V prípade, ak sa dostanem na koniec časti void loop, tak sa program začne od tejto časti vykonávať znova, teda v slučke. V tejto časti je program veľmi podobný tomu, ktorý sa nachádza v časti, ktorá sa vykonáva iba na začiatku, no v podmienke hodnoty na zopnutie zavlažovania sme si zvolili o niečo vyššiu hodnotu relatívnej vlhkosti.

```
if (h < 55) {  
    digitalWrite(ledPin, HIGH);           //zapnutie  
    zavlažovania  
    delay(4000);  
}  
else  
    digitalWrite(ledPin, LOW);  
}
```

Po ukončení programovania bolo nutné nahráť náš kód do dosky Arduino, pričom na začiatku je potrebná kompilácia programu. Počas kompilácie sa projekt skontroluje, či je validný. V prípade, že kompilácia neprebehne korektne, tak nám program Arduino IDE vypíše dole hlášku a pokúsi sa nás naviesť na chybu. Ak kompilácia prebehne v poriadku, tak môžeme pristúpiť k nahratiu programu do dosky.

Pre nahratie programu do dosky Arduino bolo potrebné v programe, v časti Nástroje a ďalej Doska zvoliť dosku, ktorú používame. V časti Nástroje bolo potrebné zvoliť aj sériový port, ktorým je doska pripojená k počítaču. Keď sme splnili všetky predchádzajúce podmienky, tak sme program nahrali do dosky.

4.4 Prepojenie modulov Arduino a Raspberry Pi

Po naprogramovaní oboch častí projektov sme pristúpili k spojeniu oboch modulov. Modul Arduino je pripojený k nespájkovanému poľu, ktorého kladný pól je prepojený s pinom 26 na doske Raspberry Pi. Týmto prepojením sme dosiahli funkčnosť, ktorú sme si zvolili ako vyhovujúcu.

Prepojenie sme museli realizovať až po naprogramovaní modulu Arduino, keďže sme potrebovali nahráť program do dosky, ale v prípade že by sme mali prepojenie

zrealizované počas nahrávania programu, tak by sme mohli poškodiť modul Raspberry Pi tým, že by sme na jeho výstupný pin priviedli napätie z dosky Arduino.

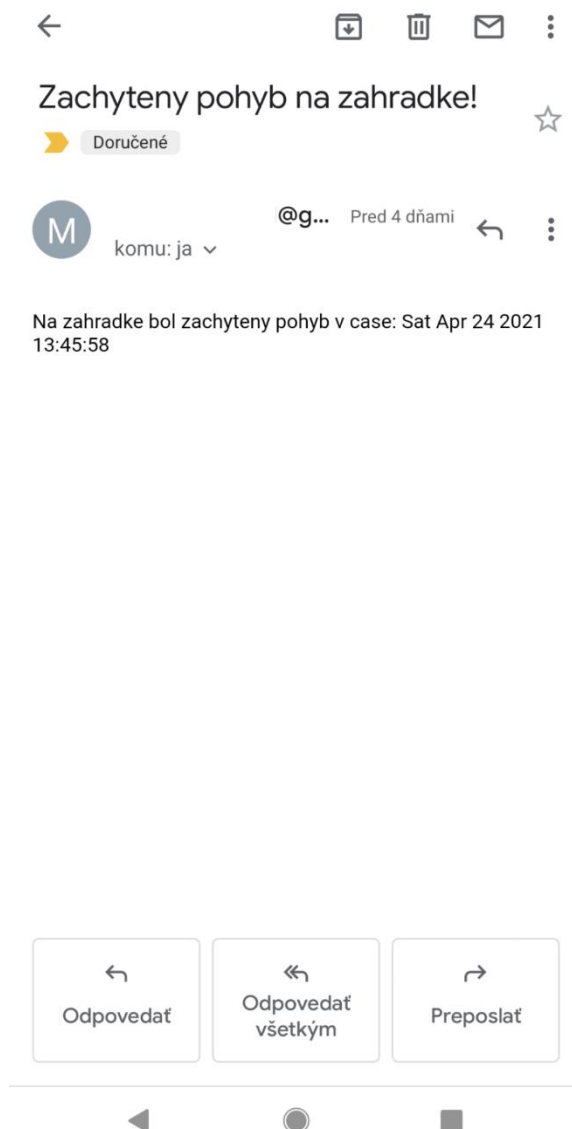
4.5 Testovanie zariadenia

Keď sme zariadenia prepojili, tak sme mohli pristúpiť k testovaniu funkčnosti. Počas testovania sme museli upraviť kód, aby zariadenie spĺňalo naše predstavy. Počas testovania sme v našom kóde narazili na menšie problémy, pričom hlavný problém sme mali v tom, že časť, ktorá beží na platforme Arduino nám spúšťala zavlažovanie aj v prípade, kedy sme nechceli, respektíve nám spúšťala a zatvárala zavlažovanie na striedačku, aj keď podľa vypisovaných hodnôt sa malo zavlažovanie spustiť a čakať na zvýšenie vlhkosti. Tento problém sme opravili takým spôsobom, že sme si na začiatku programu v časti void setup () spustili podmienku, ktorá overila, či je potrebné okamžite spustiť zavlažovanie alebo to potrebné nie je.

Po vyriešení problémov sme ešte vymenili LED kontrolky, ktoré nám simulujú vykonávanie činností, tak, aby ku každej simulovanej činnosti prislúchala jedna farba LED diódy a vedeli sme tak s určitosťou povedať, ktorá činnosť sa práve vykonáva.

Ďalej sme sa v rámci testovania venovali nastaveniu citlivosti pohybového snímača tak, aby bola intenzita priestoru snímania pohybu dostatočná, aby snímač zachytil každý pohyb.

Počas testovania zariadenia sme overili doručenie e-mailovej notifikácie, ktorá fungovalo bezchybne. E-mailová notifikácia obsahovala všetky údaje, ktoré sme od nej očakávali a informovala nás o dianí v skleníku.

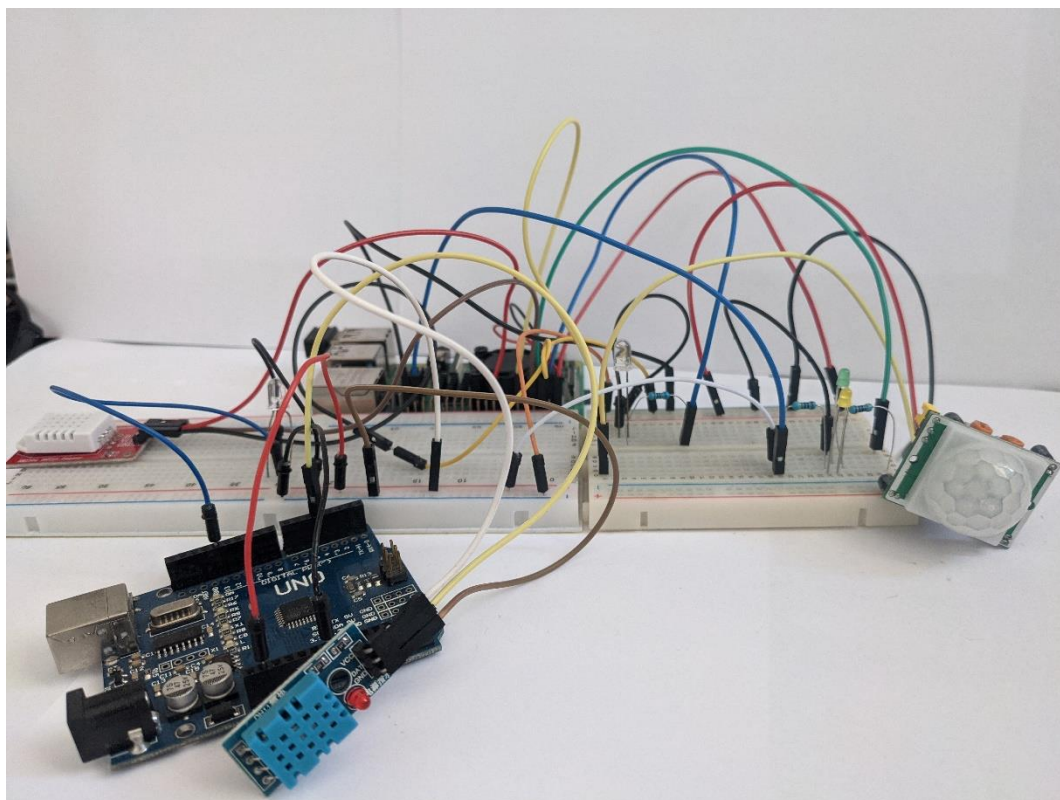


Obr. 22: E-mailová notifikácia

Zdroj: Vlastné spracovanie

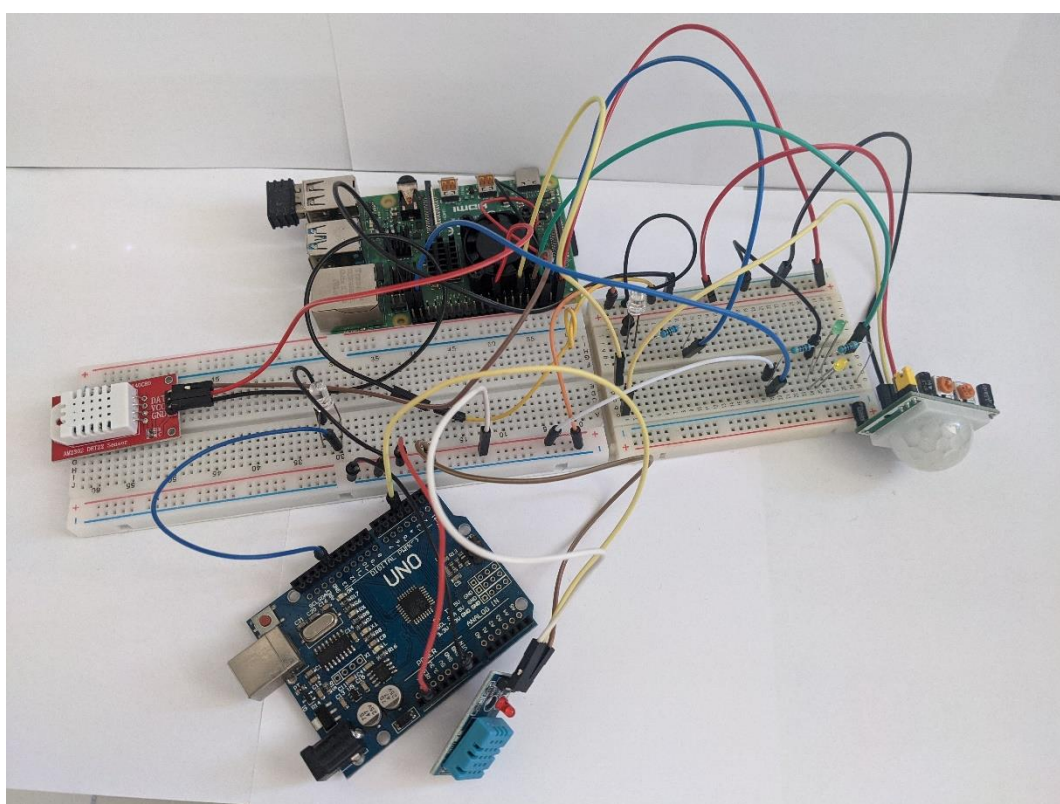
4.6 Ukážka zhotoveného zariadenia

Po vykonaní predchádzajúcich krokov sme dospeli k zostavenému prototypu zariadenia, ktoré sme navrhli. Náš prototyp vyzerá takto:



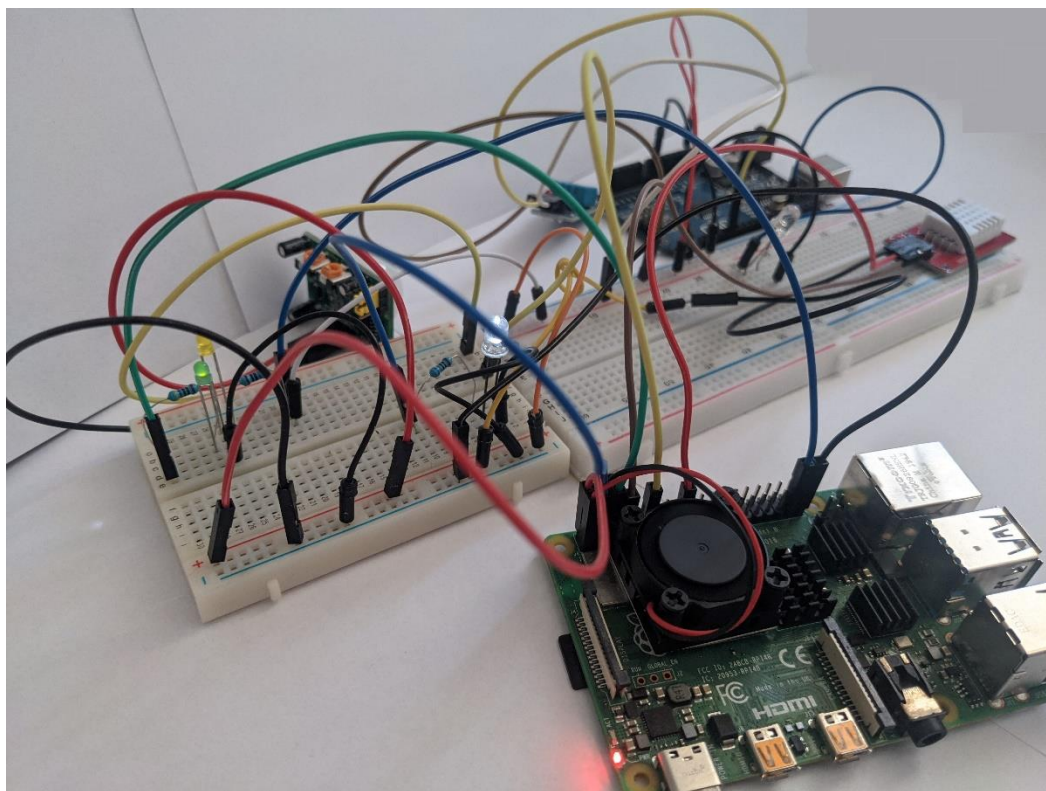
Obr. 23: Predný pohľad na náš prototyp

Zdroj: Vlastné spracovanie



Obr. 24: Vrchný pohľad na prototyp

Zdroj: Vlastné spracovanie



Obr. 25: Spustenie prototypu
Zdroj: Vlastné spracovanie

5 Diskusia

V tejto kapitole si preskúmame možnosti, ako by sme nami skonštruované zariadenie vedeli vylepšiť. V našej práci sme si skonštruovali prototyp zariadenia, ktoré nám poskytuje informácie získané pomocou snímačov a automatizuje vykonávanie základných činností spojených s vhodnou mikroklimou skleníka. Ako hlavnú úpravu, ktorú by sme mohli navrhnúť, je použitie dosky plošných spojov, ktorou by sme nahradili nespájkované pole. Do dosky plošných spojov by sme prispájkovali snímače a LED kontrolky, ktoré by slúžili na kontrolu funkčnosti zariadenia v prípade, že by sme nepotrebovali naďalej simulovať iné zariadenia.

Z hľadiska využiteľnosti by sme mohli naše zariadenie rozšíriť o viac snímačov, ktoré by nám priniesli viac dát, ktoré by sme vedeli využiť. Ďalej by sme naše zariadenie mohli rozšíriť aj o dotykový displej, na ktorom by sa zobrazovali namerané údaje, a taktiež by sme si vedeli pomocou pár klikov zmeniť nastavené hraničné hodnoty.

Taktiež je dôležité riešiť napájanie zariadenia. V našom testovacom prípade sme používali ako zdroj napätia nabíjačku na mobil, avšak táto možnosť je vhodná len v prípade, ak je v blízkosti skleníka elektrická zásuvka. V prípade, ak zásuvka nie je k dispozícii, je možné použiť dve alternatívne možnosti napájania. Jednou z možností je použitie akumulátorov s vysokou kapacitou, ktoré by však bolo treba dostatočne včas nabiť, prípadne vymeniť za nabité. Ďalšou možnosťou je použitie kombinácie akumulátorov s fotovoltickými článkami. Táto kombinácia je najvhodnejšou alternatívou z toho dôvodu, že by nebol potrebný ľudský zásah do automatizovaného systému, keďže fotovoltické články by dokázali udržiavať akumulátory v nabitom stave. Taktiež ide o ekologické riešenie, nakoľko energia získaná zo slnka je obnoviteľným zdrojom a tým pádom nie je vyčerpateľná.

Na ovládanie zavlažovania by sme vedeli použiť elektromagnetické ventily, ktoré sú vhodné na projekty automatického zavlažovania. Tieto ventily je možné zakúpiť v cenovej hladine od cca piatich eur, pričom možnosti napájania sú od 5 voltov až po napájanie 220 voltov. Existujú dva druhy ventilov. Jeden druh funguje tak, že v prípade, že je bez napájania, tak ostáva uzatvorený, ďalší druh naopak, v prípade, že je bez napájania, tak je ventil otvorený. V našom prípade by sme použili ventil, ktorý je bez napájania uzavretý. Taktiež v prípade vyššieho počtu ventilov alebo použitia ventilov s vyšším spínaným napätím by bolo potrebné použiť shield, ktorý by ovládal tieto ventily.

Záver

Cieľom tejto diplomovej práce bolo prakticky využiť platformy Raspberry Pi a Arduino pre potenciál Internetu vecí. V našom prípade sme si vybrali vyhotovenie zariadenia na správu mikroklimy skleníka na platformách Raspberry Pi a Arduino.

V prvotnej časti sa upriamili na charakteristiku pojmov, ktoré s našou problematikou súvisia. Neskôr sme si vysvetlili pojem Internet vecí, prešli sme si vývoj mobilných sietí až po najnovšiu generáciu. Taktiež sme sa v tejto časti venovali používaným sieťam, pričom sme si okrem iného uviedli aj ich výhody či nevýhody. Nasledovali sme opisom platforiem, ktoré sme si zvolili, možnosti ich rozšírenia a spomenuli sme aj ďalšie podobné platformy. Ku koncu prvej kapitoly sme si opísali aj programovací jazyk Python, v ktorom sa programovala časť zariadenia.

V ďalšej kapitole sme si ujasnili cieľ diplomovej práce, čo znamenalo v našom prípade vytvorenie zariadenia na správu mikroklimy skleníka. V následnej časti práce sme si podrobne opísali nami zvolené časti. Po voľbe častí sme si načrtli schému zapojenia, podľa ktorej sme následne zariadenie zostavili, naprogramovali jednotlivé časti a spojili ich do jedného celku. Po prepojení zariadení sme pristúpili k testom zariadenia, počas ktorých sme vychytali chyby a vykonali úpravy, ktoré sme považovali za potrebné.

Nami navrhnuté zariadenie, ktorého prototyp sme si aj prakticky zhotovili, dokáže automaticky spravovať mikroklimu skleníka pomocou dvoch snímačov DHT, ktoré merajú teplotu vzduchu. Zariadenie vyhodnocuje získané dáta a podľa tohto vyhodnotenia dokáže otvárať okná alebo spúšťať cirkuláciu vzduchu. Okrem ovládania snímania teploty vzduchu naše zariadenie sníma aj vlhkosť vzduchu, na základe ktorej dokáže ovládať spúšťanie zavlažovania pomocou elektromagnetických ventilov.

V závere sa dá konštatovať, že sa nám podarilo skonštruovať zariadenie na správu mikroklimy skleníka, ktoré okrem automatizácie jednotlivých požadovaných krokov aj informuje používateľa prostredníctvom internetu o zachytenom pohybe. Keďže v dnešnej dobe, kedy vo svete zúri pandémia koronavírusu, je naše zariadenie napríklad v prípade povinnej karantény vhodné na zabezpečenie správnej mikroklimy v skleníku, ktorá je dôležitá pre správny rast zasadených rastlín.

Zoznam použitej literatúry

- [1] Bluetooth 5.1 vs 5.2 [online]. [cit. 2021-2-20]. Dostupné na: <https://audio46.com/blogs/headphones/bluetooth-5-1-vs-5-2>
- [2] Do nového roku s novým jednodeskovým počítačom – ale ktorým? [online]. [cit. 2021-2-27]. Dostupné na: <https://www.root.cz/clanky/do-noveho-roku-s-novym-jednodeskovym-pocitacem-ale-kterym/>
- [3] Halfacree, Gareth. The official Raspberry Pi Beginner's Guide. 4. vyd. 2020. 248 s. ISBN: 978-1-912047-73-4
- [4] Internet Of Things [online]. [cit. 2021-1-25]. Dostupné na: <https://www.gartner.com/en/information-technology/glossary/internet-of-things>
- [5] Lubanovic, Bill. Introducing Python. 2. vyd. 2015. 476 s. ISBN: 978-1-449-35936-2
- [6] Number of Internet of Things (IoT) connected devices worldwide from 2019 to 2030 [online]. [cit. 2021-2-7]. Dostupné na: <https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/>
- [7] Operátori zverejnili čísla o SIM kartách, ktoré využívajú stroje [online]. [cit. 2021-2-6]. Dostupné na: https://camit.sk/sk/novinky/20453_operatori-zverejnili-cisla-o-sim-kartach-ktore-vyuzivaju-stroje
- [8] RAFAELS, Ray. Cloud Computing: From Beginning to End. 2015. 152 s. ISBN: 978-1511404587
- [9] Rodriguez, Jonathan. FUNDAMENTALS OF 5G MOBILE NETWORKS. 2015. 336 s. ISBN: 9781118867525
- [10] Sekan, Marek. VYUŽITIE PLATFORMY ARDUINO PRE IoT. Bratislava : s.n., 2019. 103004/B/2019/36103157977991172
- [11] The Internet of Things in a Smart Connected World [online]. [cit. 2021-2-6]. Dostupné na: https://www.researchgate.net/publication/326752232_The_Internet_of_Things_in_a_Smart_Connected_World

- [12] Thread protocol simplifies IoT security [online]. [cit. 2021-2-21]. Dostupné na: <https://www.embedded.com/thread-protocol-simplifies-iot-security/>
- [13] What's Orange Pi 4B? [online]. [cit. 2021-2-27]. Dostupné na: <http://www.orangepi.org/Orange%20Pi%204B/>
- [14] Zigbee Wireless Mesh Networking [online]. [cit. 2021-2-21]. Dostupné na: <https://www.digi.com/solutions/by-technology/zigbee-wireless-standard>
- [15] Z-Wave [online]. [cit. 2021-2-21]. Dostupné na: <https://internetofthingsagenda.techtarget.com/definition/Z-Wave>