



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**

FACULTY OF INFORMATION TECHNOLOGY

**ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ**

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

**ZAŘÍZENÍ PRO MONITOROVÁNÍ A AGREGACI VELKÉHO MNOŽSTVÍ PRŮMYSLOVÝCH DAT**

A MONITORING AND AGGREGATION DEVICE FOR LARGE AMOUNTS OF INDUSTRY DATA

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**OLIVER HELD**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**doc. RNDr. PAVEL SMRŽ, Ph.D.**

BRNO 2019

## Zadání bakalářské práce



21538

Student: **Held Oliver**  
Program: Informační technologie  
Název: **Zařízení pro monitorování a agregaci velkého množství průmyslových dat**  
**A Monitoring and Aggregation Device for Large Amounts of Industry Data**  
Kategorie: Informační systémy

### Zadání:

1. Seznamte se metodami zpracování velkých dat a dostupnými řešeními pro monitoring a agregaci, schopnými nasazení v průmyslovém prostředí.
2. S využitím získaných znalostí navrhnete zařízení, vybavené vhodným procesorem, které dokáže číst data z připojených senzorů, synchronizovat čas, data předzpracovat a agregovat a pomocí síťového rozhraní odesílat na server k dalšímu zpracování.
3. V součinnosti se spolupracující firmou 4dot realizujte navržené řešení a vyhodnoťte chování systému v reálných podmínkách.
4. Vytvořte stručný plakát prezentující práci, její cíle a výsledky

### Literatura:

- dle domluvy s vedoucím

Pro udělení zápočtu za první semestr je požadováno:

- funkční prototyp řešení

Podrobné závazné pokyny pro vypracování práce viz <http://www.fit.vutbr.cz/info/szz/>

Vedoucí práce: **Smrž Pavel, doc. RNDr., Ph.D.**

Vedoucí ústavu: Černocký Jan, doc. Dr. Ing.

Datum zadání: 1. listopadu 2018

Datum odevzdání: 15. května 2019

Datum schválení: 6. listopadu 2018

## Abstrakt

Táto bakalárska práca popisuje hardvérovú a softvérovú prípravu monitorovacieho zariadenia pre zber veľkých dát v priemysle. Zariadenie je zložené z vývojového kitu BeagleBone Black, na ktoré sú pripevnené senzory na snímanie vibrácií, teploty a pretvorenia. Cieľom práce je vytvoriť zariadenie s dôrazom na nízke náklady, jednoduchosť a možnú rozšíriteľnosť. Pri zbere dát zo senzorov sa využíva zbernica I2C a vstavaný AD prevodník. Dáta sú odosielané na vzdialený server pomocou protokolu UDP. Časť práce sa venuje vytvoreniu a použitiu vlastného tenzometrického modulu. Prínosom práce je otestovanie platformy a metódy zberu, agregácie a odosielania dát.

## Abstract

This bachelor thesis describes hardware and software development of a monitoring device for the collection of industrial big data. The device is based on the development BeagleBone Black kit together with vibration sensors, temperature sensors and strain gauges. The objective of the thesis is to design a device with respect to low production cost, simplicity and possible modularity. The I2C bus and embedded AD converter are used for the collection of the data from the sensors. The data is sent to remote server using UDP protocol. Furthermore, custom strain gauge module is designed and tested. The practical contribution of the thesis is to test the platform functionality, that includes collection, aggregation and transfer of the data.

## Klíčové slová

BeagleBone Black, I2C, Linux, Debian, Tenzometrický modul, Veľké dáta, IoT, Prediktívna údržba

## Keywords

Beaglebone Black, I2C, Linux, Debian, Strain gauge module, Big Data, IoT, Predictive maintenance

## Citácia

HELD, Oliver. *Zařízení pro monitorování a agregaci velkého množství průmyslových dat*. Brno, 2019. Bakalárska práca. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce doc. RNDr. Pavel Smrž, Ph.D.

# Zařízení pro monitorování a agregaci velkého množství průmyslových dat

## Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením pána doc. RNDr. Pavla Smrža, Ph.D. Uviedol som všetky literárne pramene a publikácie, z ktorých som čerpal.

.....

Oliver Held  
13. mája 2019

## Podakovanie

Ďakujem vedúcemu bakalárskej práce doc. RNDr. Pavlovi Smržovi, Ph.D. za cenné rady, vecné pripomienky a ochotu pri konzultáciách a vypracovaní práce.

# Obsah

<b>1</b>	<b>Úvod</b>	<b>3</b>
<b>2</b>	<b>Zber dát v priemysle</b>	<b>5</b>
2.1	Internet vecí . . . . .	5
2.2	Veľké dáta . . . . .	6
2.3	Údržba . . . . .	7
2.3.1	Reaktívna údržba . . . . .	7
2.3.2	Preventívna údržba . . . . .	7
2.3.3	Prediktívna údržba . . . . .	8
2.4	Existujúce riešenia . . . . .	9
2.4.1	Siemens MindSphere + MindConnect . . . . .	9
2.4.2	SKF Multilog IMx-8 . . . . .	10
2.4.3	Diagnostický systém na hardvéri CompactRIO . . . . .	10
<b>3</b>	<b>Výber hardvéru</b>	<b>12</b>
3.1	Charakteristika komponentov . . . . .	12
3.1.1	Zbernica I2C . . . . .	12
3.1.2	Zbernica SPI . . . . .	14
3.1.3	Prevodník AD . . . . .	15
3.2	Vývojové kity . . . . .	15
3.2.1	Raspberry Pi . . . . .	15
3.2.2	BeagleBone Black . . . . .	16
<b>4</b>	<b>Návrh riešenia</b>	<b>18</b>
4.1	Vývojový kit . . . . .	18
4.2	Operačný systém . . . . .	19
4.2.1	Android . . . . .	19
4.2.2	Angstrom . . . . .	19
4.2.3	Debian . . . . .	20
4.2.4	Fedora . . . . .	20
4.2.5	Buildroot . . . . .	20
4.3	Senzory . . . . .	20
4.3.1	Snímač teploty - HTU21D . . . . .	20
4.3.2	Akcelerometer - ADXL345 . . . . .	20
4.3.3	Návrh tenzometra . . . . .	21
4.4	Návrh softvéru . . . . .	22
4.4.1	Čítanie a agregácia dát . . . . .	22
4.4.2	Odosielanie dát . . . . .	23

4.4.3	Prijímanie a ukladanie dát . . . . .	23
<b>5</b>	<b>Implementácia</b>	<b>26</b>
5.1	Hardvér . . . . .	26
5.1.1	BeagleBone Black . . . . .	27
5.1.2	HTU21D a ADXL345 . . . . .	27
5.1.3	Tenzometrický modul . . . . .	27
5.2	Softvér . . . . .	28
5.2.1	Inštalácia operačného systému . . . . .	28
5.2.2	Inicializačný skript . . . . .	28
5.2.3	Sietová konfigurácia . . . . .	29
5.2.4	Otvorenie periférií . . . . .	29
5.2.5	Vývojové prostredie . . . . .	29
5.2.6	Aplikácie pre čítanie dát . . . . .	29
5.2.7	Aplikácia pre odosielanie dát . . . . .	31
5.2.8	Serverová aplikácia . . . . .	31
5.2.9	Spúšťanie aplikácií . . . . .	31
5.2.10	Čítanie dát z I2C . . . . .	31
5.2.11	Vstavaný AD prevodník . . . . .	32
5.2.12	Zhrnutie softvéru . . . . .	33
5.3	Návod k inštalácii . . . . .	33
<b>6</b>	<b>Experimenty a vyhodnotenie</b>	<b>35</b>
6.1	Prvý experiment: Overenie agregácie a vyťaženia . . . . .	35
6.2	Druhý experiment: Porovnanie snímaču teploty . . . . .	36
6.3	Tretí experiment: Dlhodobý zber dát . . . . .	36
6.4	Štvrtý experiment: Zber dát z testovacej podložky . . . . .	37
6.5	Zhrnutie experimentov . . . . .	38
<b>7</b>	<b>Záver</b>	<b>40</b>
	<b>Literatúra</b>	<b>42</b>

# Kapitola 1

## Úvod

S vývojom priemyslu sa zvyšuje nárok na efektívnu výrobu, čo si žiada podrobnú analýzu procesov. Nevyhnutná súčasť analýzy je zber veľkého množstva dát. Pomocou zozbieraných dát je možné získať prehľad o množstve udalostí, ktoré sa udiali v minulosti. Pomocou týchto dát sme schopní odhaliť zmeny stavu. Takto nadobudnuté informácie vedú k predikcii udalostí, ktoré môžu v budúcnosti nastať.

Predikcia poruchy na priemyselných strojoch je neoceniteľná pre všetky továrne. Nečakaná odstávka stroja stojí továrne veľké množstvo financií a znemožňuje dosiahnuť dohodnuté termíny. Základnou motiváciou k vytvoreniu zariadenia, ktoré zbiera priemyselné dáta a umožňuje predpovedať poruchu je práve spomínaná možnosť predikcie poruchy a možnosť zefektívnenia výroby.

Táto práca sa zaoberá vytvorením zariadenia vhodného na nasadenie do priemyslu pre zber a agregáciu veľkého množstva priemyselných dát. Takéto zariadenie získava dáta z priemyselných senzorov. Namerané hodnoty sa po agregácii odosielať na vzdialený server, na ktorom sú nazbierané dáta pripravené na ďalšie použitie a dopomáhajú k predikcii poruchy na priemyselnom stroji. Práca je vytvorená v spolupráci s firmou, ktorá pracuje na vývoji produktov rozvíjajúcich priemyselnú údržbu.

Na začiatku práce sú vysvetlené základy potrebné pre pochopenie zberu dát v priemysle, možnosti udržania chodu výroby a prehľad existujúcich riešení danej problematiky. V nasledujúcej časti sú charakterizované komponenty, ktoré musí monitorovacie zariadenie obsahovať pre čítanie dát zo senzorov a sú porovnané vývojové kity, ktoré spĺňajú požiadavky pre účely tejto práce. Nasleduje návrh riešenia, v ktorom je zdôvodnený výber vývojového kitu, jeho operačného systému a použitých senzorov. Pri zostavovaní zariadenia je dôraz kladený na čo najnižšiu cenu všetkých jeho komponentov, keďže sa jedná o študentskú prácu. Po zvolení všetkých hardvérových častí nasleduje návrh softvéru, ktorý je schopný zberu a odoslania dát na vzdialený server. V implementačnej časti je mimo iného popísaný postup pre zhotovenie vlastného tenzometrického modulu. Nutnou súčasťou práce je testovanie a vyhodnotenie riešenia. V testovacej časti sú opísané experimenty, pomocou ktorých je dokázaná schopnosť plynulého chodu zariadenia.

Práca sa odráža od systému vyvinutého partnerskou firmou, ktoré je popísané v kapitole 2.4.3. Cieľom je napodobniť funkcionality tohoto riešenia so snahou čo najviac znížiť cenu využitého hardvéru pri zachovaní všetkej funkcionality potrebnej pre správny zber a agregáciu dát. Pri výbere komponentov sa zohľadňuje možnosť nasledujúceho rozšírenia riešenia.

Pri testovaní senzorov, ktoré sa v práci používajú sa ukázalo, že nie všetky sú vhodné pre zber priemyselných dát. Pri analýze zozbieraných dát z tenzometrického modulu BF350

sa ukázalo, že dáta neodpovedajú realite, čo zapríčinila chyba vnútorného zapojenia. Táto chyba spôsobila osciláciu výstupného signálu. Z tohoto dôvodu bolo rozhodnuté pre zhotovenie vlastného tenzometrického modulu. Postup návrhu a implementácie je v tejto práci popísaný.

Výstupom zariadenia sú agregované dáta, ktoré sú uložené v databáze na vzdialenom serveri. Takéto dáta je ďalej možné analyzovať. Správna analýza dát umožní previesť zásah údržby ešte pred poruchou. Zároveň je včasným upozornením umožnené naplánovať odstávku stroja a zamedziť neočakávanému prestoju. Vďaka údržbe je proces výroby efektívnejší a šetrí peniaze, čas aj životné prostredie.



## Kapitola 2

# Zber dát v priemysle

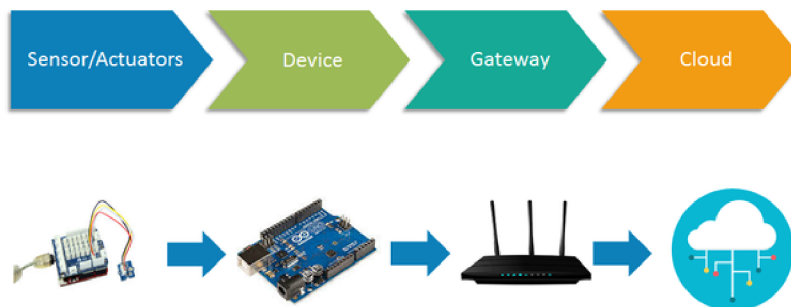
### 2.1 Internet vecí

Internet vecí, známejší ako IoT, je výraz, ktorý je v modernej informatike čím ďalej tým viac spomínaný. Pole využitia sa rozrastá. Ide o novodobý systémový trend, ktorý označuje sieť umožňujúcu digitálnym alebo mechanickým zariadeniam pripojenie bez akéhokoľvek zásahu človeka. IoT teda v informatike označuje prepojenie vstavaných zariadení, ktoré využívajú internet a bezdrôtový prenos dát. IoT prináša nové možnosti vzájomnej interakcie, nielen medzi jednotlivými systémami, ale taktiež prináša nové možnosti ovládania týchto zariadení, ich sledovania a zabezpečovania pokročilých služieb. Využitie tejto metódy má vysoké uplatnenie najmä v oblastiach, kde je potrebné zbierať a vymieňať veľké dáta. Táto kapitola čerpá z [13].

IoT si našlo veľké využitie v inteligentných domácnostiach, zdravotníctve, ale aj v priemysle, kde tento trend nachádza stále väčšie a väčšie uplatnenie z dôvodu jednoduchého a hlavne efektívneho zbierania veľkých dát. Tento systém je schopný realizácie bezproblémovej integrácie rôznych výrobných zariadení vybavených funkciami pre snímanie, identifikáciu, spracovanie, komunikáciu, ovládanie a sieťovanie. IoT prináša do sféry priemyselných aplikácií a inteligentnej výroby mnoho nových využití, ktoré sú spojené s efektívnejším chodom celého procesu výroby. Medzi tieto využitia patrí napríklad sieťová kontrola, riadenie výrobných procesov a zariadení či riadenie aktivít a situácií, ktoré vo výrobe nastávajú, či už ide o bežné monitorovanie alebo sledovanie a predvídanie rôznych porúch strojov. To v praxi pre výrobný priemysel znamená zefektívnenie výroby a taktiež posun vo výrobe nových produktov. IoT dokáže poskytovať dynamickú odozvu na požiadavku produktov a optimalizáciu výrobných reťazcov v reálnom čase prostredníctvom prepojenia strojov, senzorov a riadiacich systémov. Veľkým prínosom a kompetenciou IoT je tiež automatizácia riadenia procesov spojená s optimalizáciou bezpečnosti a ochrany zariadení, čo spočíva v možnostiach prediktívnej údržby, štatistického vyhodnocovania a meraní s cieľom maximalizovať spoľahlivosť výrobných strojov. O rozrastaní a nevyhnutnosti využívania moderných technológií na zber a vyhodnotenie veľkých dát svedčí aj odhad inštitútu *McKinsey Global*, ktorý v roku 2011 odhadoval, že Spojené Štáty potrebujú 140 000 až 190 000 nových pracovníkov s analytickými schopnosťami a 1,5 milióna manažérov a analytikov s analytickými schopnosťami na podnikateľské rozhodnutia na základe analýzy veľkých dát [15].

Problémové vo využití tohto prepojenia môžu byť rôzne existujúce štandardy pre komunikáciu rôznych skupín výrobcov, medzi ktoré patria Open Interconnect Consortium (In-

tel, Samsung, Dell, Broadcom. . . ), AllSeen Alliance(Cisco, LG, Microsoft, Qualcomm, Sharp, . . . ), Industrial Internet Consortium a OGC Sensor Web Enablement<sup>1</sup>.



Obr. 2.1: Architektúra IoT. Prevzaté z [23].

Na obrázku 2.1 je znázornená architektúra IoT. Jedná sa o kombináciu senzorov, vývojevy dosky a vzdialeného úložiska, kde sa dáta zbierajú a ukladajú.

## 2.2 Veľké dáta

Ak sa v dnešnej dobe hovorí v spojení s priemyselným sektorom o veľkých dátach, ide o extrémne veľké súbory údajov, ktoré je možné analyzovať a skúmať. Získané údaje je možné využívať v prospech efektivity výrobných priemyselných procesov. Téma veľkých dát je neoddeliteľnou súčasťou konceptu štvrtej priemyselnej revolúcie, ktorá zahŕňa mnohé odvetvia priemyslu. Pravdou je, že množstvo dát bude v budúcnosti už len pribúdať. S rastúcim množstvom dát rastú aj požiadavky a nároky ich spracovania. z iného pohľadu by sa dalo hovoriť o tom, že veľké množstvo dát existovalo už dávno pred tým, ako sa začalo o pojme veľké dáta vôbec hovoriť. Podstatné teda je, že pri tomto pojme nejde ani tak o kvantitu dát, ale o kvalitu informácií a prepojení, ktoré sme schopní z týchto dát vyťažiť. Dnešné možnosti prinášajú veľa otázok a výziev k odhaleniu využitia veľkých dát v priemysle.

Je dôležité poznamenať aj to, že mnohé spoločnosti dokážu fungovať efektívne aj s použitím malých dát. Pre priemyselné spoločnosti je dôležité to, ako vedia s množstvom dát pracovať. Využitie veľkých dát teda môže priniesť pridanú hodnotu podnikom, ktoré sú schopné s týmito dátami pracovať. Pre využitie veľkých dát však neexistuje univerzálne riešenie a tak sa musí každá spoločnosť zamyslieť nad možnosťami, ktoré jej zber a analýza veľkých dát ponúkajú.

Digitalizácia a automatizácia sa v dnešnej dobe postupne začína vzťahovať na všetky oblasti života a spoločnosti. Ani priemyselná výroba nezostáva nepovšimnutá a automatizácia sa jej začína týkať čo raz viac a viac, a to vo všetkých jej odvetviach. V súčasnej dobe je teda trendom, a dá sa povedať nevyhnutnosťou, spolupráca IT spoločností s priemyselnými podnikmi, prípadná spolupráca IT a priemyselných oddelení spoločnosti. Tento trend demonštruje dôležitosť zberu veľkých dát pre priemyselné podniky. Narastá počet podnikov, ktoré investujú do technológií zberu dát a ich analýzy. Tým sa ukazuje nevyhnutnosť zberu dát pre správne rozhodnutia, týkajúce sa výroby s cieľom čo najefektívnejších výrobných postupov a tým šetrenia množstva prostriedkov.

<sup>1</sup>Všetečka, R.: Firmy bojují o ovládnutí internetu věcí. [Online; navštívené 02.02.2019]. URL [https://www.idnes.cz/technet/pc-mac/souboj-internetu-veci-iot.A150316\\_161724\\_hardware\\_vse43](https://www.idnes.cz/technet/pc-mac/souboj-internetu-veci-iot.A150316_161724_hardware_vse43)

Mnohé podnikové procesy sú prostredníctvom riešení moderného priemyslu digitálne transformované. Tento pochod umožňuje efektívny zber údajov z výrobných prevádzok a ich následné spracovanie pre ďalšie analýzy, prípadne pre prognózy výroby. Základom zberu veľkých dát pre priemyselné účely je využitie senzorov vo výrobných procesoch. Pomocou senzorov je možný zber obrovského množstva parametrov v celom výrobnom procese. Hlavným cieľom zberu dát a automatizácie je vytvoriť prístroje čo sa nekazia, cenové nároky na ich obsluhu sú minimálne a kvalita finálnych výrobkov je vysoká a konštantná. Medzi najbežnejšie využitie zberu veľkých dát patrí monitoring vyťaženia výrobných procesov, taktiež vyhodnocovanie celkovej efektivity zariadení a v neposlednom rade je to efektívnosť využitia všetkých strojov v priemyselnej výrobe. Veľké dáta teda ponúkajú veľké množstvo možností pri riešení optimalizácie procesov a zvyšovania ich efektivity. Ide napríklad o predikciu alebo identifikáciu príčin vzniku porúch na výrobných strojoch. Možnosťou veľkých dát je taktiež určenie oblastí so zníženou kvalitou alebo efektívnosťou výroby. Na základe analýzy takto zozbieraných veľkých dát sme schopní identifikácie takýchto oblastí a hlavne, máme možnosť tieto oblasti odstraňovať. Veľké dáta sú teda analyzované a výsledky analýzy sú využité na predikciu servisných zásahov a preventívne chovanie s týmto spojené. Podniky teda dokážu ušetriť veľké množstvo prostriedkov na odstávkach z dôvodu porúch. V zhrnutí, môžeme povedať, že čím viac o zariadení zistíme z jeho správania pri chode, tým viac nám tieto informácie pomôžu pri jeho zefektívnení a zlepšení [14].

## 2.3 Údržba

V každom priemyselnom podniku sú na začiatku nasadené nové a bezchybné mechanické diely. Treba si ale uvedomiť, že nič netrvá večne. Kompletná výmena nefunkčných strojov je absurdná a finančne nemožná. O nástroje je potrebné sa starať a vykonávať inteligentnú údržbu všetkého príslušného zariadenia. Pod pojmom údržba v priemysle sa dá chápať každá činnosť, ktorá podporuje stav zariadenia. Údržba v priemysle sa dá rozdeliť do niekoľkých kategórií. Táto kapitola čerpá z [3] a [18].

### 2.3.1 Reaktívna údržba

Reaktívna údržba je špecifická tým, že zásah sa vykonáva až vo chvíli, keď dôjde k havárii. Tento druh údržby je dobrý v prípade, že havária neovplyvní výrobu a náklady na prácu a materiál sú zanedbateľné. V dnešnej dobe je snaha čo najviac znížiť neočakávané zásahy, pretože v tomto prípade je odstránenie takejto havárie veľmi nákladné. Najväčšie výdavky sú z dôvodu odstávky výroby pri oprave. Využívanie reaktívnej údržby vyžaduje veľké množstvo náhradných dielov na čo najrýchlejšiu opravu.

### 2.3.2 Preventívna údržba

Pokročilejšia je údržba preventívna, ktorej podstata je preventívna kontrola a výmena nebezpečných a často poruchových komponentov pravidelne, bez ohľadu na ich kondíciu. Pri tejto údržbe je kľúčové vyhodnotenie, ktoré procesy sú kritické, a správne navrhnutie intervalu plánovaných odstávok a preventívnych výmen komponentov. Najväčšou výhodou tejto údržby je skutočnosť, že poskytuje značné predĺženie schopnosti chodu zariadení. Značnou nevýhodou sú vysoké náklady, ktoré súvisia s pravidelným menením komponentov, ktoré sú často v dobrom stave.

### 2.3.3 Prediktívna údržba

Prediktívna údržba je analytický postup, ktorého výsledkom je predpoveď, kedy dôjde k zlyhaniu výrobného zariadenia. Umožňuje predchádzať zlyhaniu včasnou a efektívnou údržbou. Hlavným problémom je skutočnosť, že objemy dát súvisiace s výkonnosťou a funkčnosťou zariadenia stále narastajú. Prediktívna údržba je založená na spájaní zdrojov interných dát zo senzorov, čidiel a riadiacich jednotiek. Následná analýza je založená na sofistikovaných štatistických metódach. Tieto metódy sú schopné odhaliť skryté nebezpečenstvá a tendencie k poruchám. Včasné zistenie problému ponúka dostatok času na prípravu odstávky a prípravu na odstránenie poruchy. Aj keď je prediktívna údržba špecifická pre každé zariadenie, všeobecne sa zaužívajú postupy ako ich implementovať:

1. Identifikovať problémové oblasti v rámci zariadenia.
2. Vypočítať finančný dopad problému.
3. Identifikovať aké dáta je potrebné analyzovať, ktoré z nich sú k dispozícii a ktoré je potrebné začať zbierať.
4. Odhadnúť náklady na implementáciu riešenia a porovnať ich s možným prínosom.
5. Realizovať testovací projekt a overiť jeho funkčnosť.
6. Skombinovať dáta z prediktívnej údržby a začleniť ich do metód údržby spoločnosti.

Z hľadiska prediktívnej údržby sú v súčasnosti najviac rozvíjané prostriedky na online monitoring. Online monitoring má dlhú históriu, kde na začiatku išlo o klasickú vizuálnu kontrolu. Dnes sa však možnosti rozšírili a vznikli automatizované metódy, ktoré využívajú pokročilé techniky spracovania signálov založené na rozpoznávaní vzorov, neurónových sieťach, fuzzy logike a mnohých ďalších metódach. Pre získavanie týchto dát slúžia senzory pripojené na takzvané online jednotky, ktoré dáta zbierajú a šíria ich ďalej. Integrácia týchto senzorov s technikami prediktívnej údržby môže zabrániť zbytočnej výmene zariadení, ušetriť náklady a zlepšiť bezpečnosť procesov, ich dostupnosť a efektívnosť [10].

#### Online jednotky

Pod pojmom online jednotky sa v minulosti rozumeli prístroje, ktoré sú trvalo nainštalované na stroji a vykonávajú priebežné meranie, jednoduché výpočty a okamžite dochádza k vyhodnoteniu. Dnes, keď sa nároky zákazníka na presnosť skutočného stavu zvyšujú prichádzajú moderné online zariadenia, ktoré priebežné merania a výpočty odosielať do nadradeného diagnostického systému a až tam dochádza k pokročilejšiemu spracovaniu. Výsledkom pokročilejšieho spracovania sú pokročilejšie vyhodnocovacie algoritmy. Základné výhody a rozdiely moderných online diagnostických systémov oproti zastaralým sú nasledovné:

- Možnosť vykonávať komplexnejšie analýzy.
- Zapojenie umelej inteligencie, schopnosť učiť sa na základe zozbieraných dát, rozhodovať o kondícii stroja podobného typu.
- Schopnosť komunikovať cez internet, ukladať dáta na vzdialené úložisko.

## Sledované veličiny

Všeobecne je možné stav stroja sledovať pomocou rôznych fyzikálnych veličín. Najčastejšie sledovaná veličina v priemysle je teplota. Avšak, ak má byť filozofia prediktívnej údržby, ktorou je včasné zachytenie poruchy splnená, je potrebné sledovať veličiny, ktoré poskytnú informáciu o začínajúcej poruche v začínajúcom štádiu. Ideálnym spôsobom je sledovanie viacerých poruchových veličín súčasne, no to je často veľmi nákladné. Výsledkom skúmaní sa tak ukazuje ako finančne aj efektívne najvhodnejšie meranie a analýza vibrácií, ktorá môže byť doplnená o meranie teploty, opotrebovania materiálu či otáčok a akustických emisií.

## 2.4 Existujúce riešenia

V dnešnej dobe existuje veľa zariadení, ktoré dokážu zbierať dáta zo strojov. Najväčší problém týchto zariadení je vysoká cena a to, že niektoré neposkytujú plnohodnotný prístup ku všetkým surovým dátam. Prostriedky pre údržbu je potrebné voliť pre zariadenia jednotlivo s ohľadom na ekonomickú návratnosť investície. Nepretržité sledovanie si vyžadujú stroje, ktoré svojim výpadkom ohrozujú ekonomický chod fabriky. Nároky na plynulosť výroby neustále narastajú a tak aj ponuka nástrojov, ktoré zvyšujú produkciu pribúda. Schopnosť orientovať sa v ponuke a vybrať správne riešenie je dôležité.

V tejto kapitole sú popísané existujúce riešenia popredných spoločností v prediktívnej údržbe. Informácie sú získane z technických dokumentácií. Ďalej je popísaná implementácia zariadenia od firmy, s ktorou na bakalárskej práci spolupracujem. Toto zariadenie spĺňa všetky potrebné parametre, ale je príliš drahé na potreby zberu dát. Toto riešenie je vyvinuté na hardvéri *CompactRIO* od firmy National Instruments.

### 2.4.1 Siemens MindSphere + MindConnect

Siemens MindSphere [20] je operačný systém od spoločnosti Siemens založený na cloudovom systéme, ktorý umožňuje spájať zariadenia s pokročilou analytikou. Systém je schopný komunikovať s jednotlivými zariadeniami, ale aj zbierať dáta a poskytuje nástroje pre ich vyhodnotenie. Vďaka tejto analýze dát je možné predpovedať poruchy jednotlivých pripojených zariadení, riadiť výrobou a povšimnúť si problémy výroby. Na základe dát je možné v platforme MindSphere vytvárať digitálne dvojčinky výrobkov a strojov, pomocou čoho je možná simulácia chodu. Veľká výhoda operačného systému MindSphere je možnosť pripojenia existujúcich systémov od firmy Siemens. V prípade zberu nových dát, z ešte nezapojených senzorov, ponúka Siemens IoT konektor *MindConnect IoT2040*, ktorý sa využíva pre zber a prenos dát cez spomínaný operačný systém MindSphere. Toto zariadenie podporuje prenos nazbieraných dát cez zabezpečené internetové pripojenie k rôznym službám, ktoré sú založené na cloudovom riešení. Podľa výrobcov sa jedná o časovo aj nákladovo efektívne riešenie zberu a prenosu priemyselných dát. Medzi vlastnosti zariadenia patrí zber údajov pomocou väčšiny priemyselných protokolov, bezplatná aktualizácia softvéru, bezpečnostný koncept podľa priemyselného štandardu, 500 MB lokálnej vyrovnávacej pamäte a možnosť pripojenia až piatich kanálov. Na obrázku 2.2 je znázornený konektor *MindConnect IoT2040*.



Obr. 2.2: Siemens MindConnect IoT2040. Prevzaté z [20].

### 2.4.2 SKF Multilog IMx-8

Multilog Online systém IMx-8 [21], je kompaktná monitorovacia jednotka od firmy SKF. Jej obrovskou výhodou je malý rozmer, úplnosť systému a ľahká konfigurovateľnosť. Zariadenie je zobrazené na obrázku 2.3.

Zariadenie ponúka 8 analógových a 2 digitálne kanály, ktoré je možné pripojiť k rôznym zariadeniam. Hlavným využitím je zber dát z rotačných zariadení a ich prediktívna analýza. Zariadenie umožňuje prepojenie s ďalšími jednotkami IMx od firmy SKF. Dáta je možné ukladať na cloud. Pripevnenie zariadenia prebieha na DIN Rail, ako u väčšiny priemyselných zariadení. Výrobca odporúča vloženie zariadenia do rozvodovej skrine s krytím minimálne IP65 pre lepšiu ochranu v priemyselnom prostredí.

Zariadenie obsahuje 4 GB internú pamäť, vďaka tejto pamäti je možné ukladanie dát, v prípade, že je zariadenie momentálne nepripojené k internetu a ponúka možnosť spätného odoslania dát. Komunikácia prebieha pomocou Modbusu.

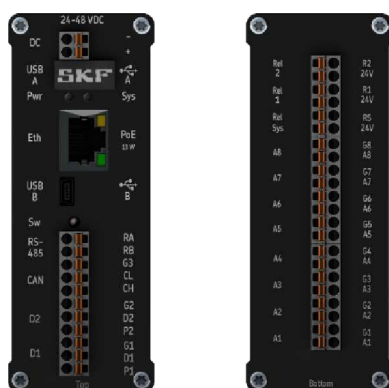
Zariadenie podporuje takzvané PoE (Power of Ethernet), ide o napájanie pomocou ethernetu čo je vyhovujúce pre priemyselné prostredie.

Výhoda zberu dát pomocou tohoto zariadenia je, že podporuje paralelne čítanie dát zo všetkých kanálov súčasne. Ukladania dát je možné periodicky, alebo pri prekročení prahovej hodnoty.

Rozsah analógových kanálov je do 40 kHz a maximálna vzorkovacia frekvencia 102,4 kHz.

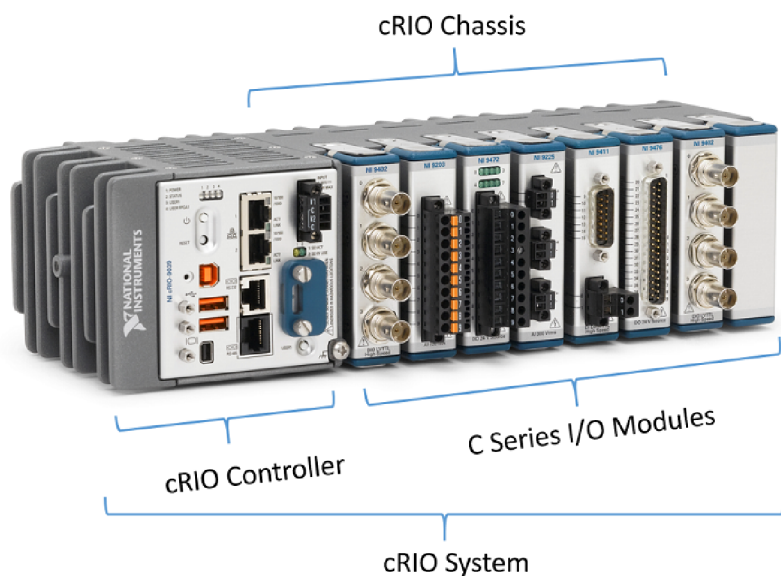
### 2.4.3 Diagnostický systém na hardvéri CompactRIO

CompactRIO [11] je priemyselný ovládač od firmy National Instruments, ktorý sa využíva pre priemyselné kontrolné systémy. CompactRIO obsahuje FPGA modul, výkonný programovateľný obvod, ku ktorému sú pripojené periférie a moduly. Systém je naprogramovaný v jazyku LabVIEW. Po základnej inicializácii systému, ktorá zahŕňa sieťovú konfiguráciu, načítanie potrebných informácií a ďalších dôležitých veličín nasleduje hlavný chod zariadenia. Pre čítanie dát z každého pripojeného modulu je zodpovedné samostatné vlákno programu. Dátový tok modulov je charakterizovaný vysokým objemom, preto prenos surových dát medzi vláknami je obsluhovaný pomocou zdieľanej pamäte s FIFO radou. Na čítanie dát z radu slúži hlavná aplikácia, ktorá zbiera dáta, agreguje ich a odosiela na vzdialený server. Diagnostická jednotka je navrhnutá pre chod pri pripojení k internetu. Pri svojom



Obr. 2.3: Multilog IMx-8. Prevzaté z [21].

chode odosiela namerané dáta na server k ďalšiemu spracovaniu. Akonáhle dôjde k výpadku internetového pripojenia, namerané dáta sa ukladajú na USB flash disk. Po obnovení pripojenia sú tieto dáta odoslané na server. K tomuto zariadeniu sú pripojené jednoosové piezoelektrické snímače vibrácií. Tieto snímače potrebujú konštantný prúd v rozsahu od 2 do 20 mA. Hlavným problémom tohoto riešenia je vysoká cena hardvéru. Preto vznikol nápad vytvoriť lacnejšie riešenie, ktorého základom je táto bakalárska práca. Na obrázku 2.4 je ovládač CompactRIO s pripojenými modulmi tvoriaci komplexný systém.



Obr. 2.4: CompactRIO. Prevzaté z [11].

## Kapitola 3

# Výber hardvéru

Základom každého monitorovacieho zariadenia je spoľahlivý hardvér. Pre vývoj monitorovacieho zariadenia bol použitý vývojový kit namiesto vytvorenia vlastnej platformy. Výhodou je dostatok ponúkaných knižníc a odborných návodov pri postupe. Cena vývojových kitov je o niečo vyššia ako zostavenie vlastného hardvéru, no pre prototyp je to ideálna voľba. V tejto dobe existuje nespočetné množstvo možností. Táto kapitola sa venuje výberu vhodného vývojového kitu, ktorý tvorí základnú časť zariadenia. Na začiatku sú spomenutí všetci možní kandidáti a ich klady, zápory a vhodnosť použitia.

Základné požiadavky sú čo najnižšia cena, podpora Linuxu, ethernet, dostatočne výkonný procesor, malé rozmery, dostatočný počet GPIO pinov, vstavaný AD prevodník a komunikačné zbernice pre komunikáciu so senzormi. Požiadavka na operačný systém Linux je vhodná, pretože tento operačný systém umožňuje jednoduchú konfiguráciu siete, časovú synchronizáciu s NTP serverom, predpripravenú komunikáciu s perifériami a prácu pri spúšťaní a synchronizácii aplikácií.

### 3.1 Charakteristika komponentov

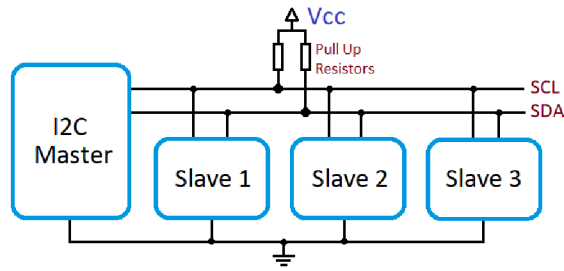
V tejto časti sú vysvetlené základné časti vývojových kitov a ich využitie pri riešení problematiky tejto práce.

#### 3.1.1 Zbernica I2C

Zbernica I2C vznikla z názvu Internal-Integred-Circuit Bus. Ako je už z názvu jasné, ide o internú dátovú zbernicu slúžiacu pre prenos dát medzi jednotlivými integrovanými obvodmi najmä v rámci jedného zariadenia. Je to dvojvodičová obojsmerná sériová zbernica od firmy *Philips Semiconductors*. Mimo napájania obsahuje vodiče označované SDA a SCL. SDA slúži ako linka na prenos dát a SCL na prenos hodinového signálu. Na obe linky je pripojený pull-up rezistor. Veľká výhoda je, že všetka komunikácia prebieha pomocou týchto dvoch spomínaných liniek, čo výrazne optimalizuje nároky na počet vstupno-výstupných pinov a zjednodušuje výsledné zapojenie. Zapojenie pomocou zbernice I2C je vidieť na obrázku 3.1. V základnej verzii poskytuje frekvenciu hodín 100 kHz, no v dnešnej dobe sú ponúkané vylepšené verzie, kde frekvencia hodín dosahuje až 1 MHz [24].

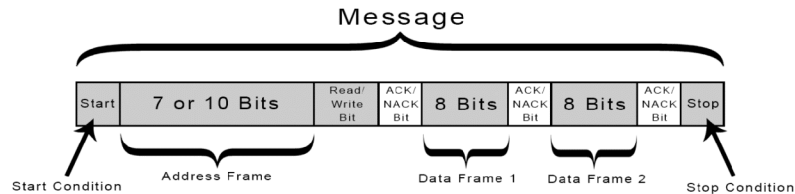
Ďalšia veľká výhoda je možnosť pripojenia viacerých čipov s rozličnou adresou na jednu spoločnú zbernicu. V tomto prípade je ale nutné mať rovnakú frekvenciu hodín pre všetky čipy.





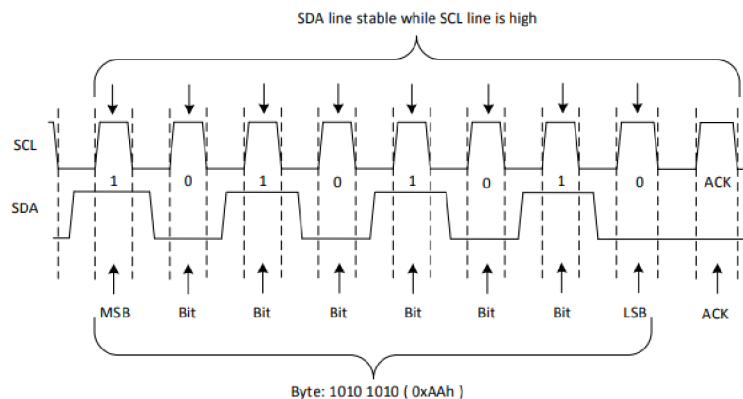
Obr. 3.1: Príklad zapojenia pomocou zbernice I2C. Prevzaté z [17].

V zapojení je vždy jeden integrovaný obvod nastavený ako MASTER a všetky ostatné ako SLAVE. Úlohou MASTER je generovanie hodinového signálu na SCL linke. Prenos dát pomocou I2C je založený na správach, ktoré sú rozdelené na rámce. Príklad takejto správy je možné vidieť na obrázku 3.2. Správa začína štart bitom, po ktorom nasleduje adresa, čítanie/zápis bit, potvrdzujúci bit, dátové rámce a stop bit [12].



Obr. 3.2: Príklad I2C správy. Prevzaté z [1].

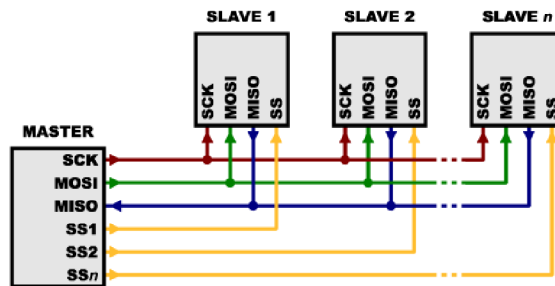
Štart a stop bit slúžia na informácie o začiatku a konci komunikácie. Rámec s adresou je sekvenca siedmich bitov, ktorá určuje adresu SLAVE, s ktorým chce MASTER komunikovať. Čítanie/zápis bit určuje smer komunikácie, teda či si MASTER vyžaduje dáta, alebo ich naopak prijíma. Potvrdzujúci bit sa odosiela za každým rámcem a informuje, či boli dáta odoslané správne, prípadne nesprávne. Potvrdzujúci bit vždy ide opačným smerom ako komunikácia. Príklad komunikácie je na obrázku 3.3.



Obr. 3.3: Príklad odosielania jedného bajtu pomocou I2C. Prevzaté z [24].

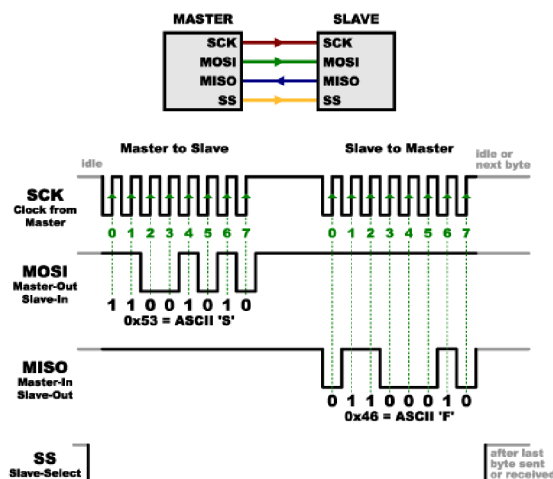
### 3.1.2 Zbernica SPI

Zbernica SPI, ktorej názov je skratkou *Serial Peripheral Interface*, je jednou zo sériových externých zberníc, ktorá slúži pre vzájomnú komunikáciu dvoch a viac komunikačných uzlov. Jeden uzol je vždy v roli MASTER a ostatné uzly fungujú ako SLAVE. Uzol vykonávajúci rolu MASTER obsahuje generátor hodinového signálu, ktorý sa zasiela všetkým ostatným uzlom, vďaka čomu je umožnený synchronný obojsmerný chod. Tento hodinový signál je prenášaný pomocou vodiča SCK. Okrem tohoto vodiča sa pri zbernici využívajú dva vodiče pre prenos dát, kde každý z nich má na starosti jeden smer dát. Tieto vodiče sa označujú MISO a MOSI. Posledný vodič potrebný pre správnu funkčnosť SPI je SS, ktorý slúži k výberu konkrétneho uzlu na komunikáciu. Zapojenie pomocou zbernice SPI je zobrazené na obrázku 3.4.



Obr. 3.4: Príklad zapojenia pomocou zbernice SPI. Prevzaté z [9].

Najväčšia výhoda SPI je jednoduchosť a to ako implementácie tak aj elektrického rozhrania, ktoré tvoria jednosmerné piny a prenosový protokol. V podstate sa jedná o niekoľko externe pripojených posuvných registrov, ktoré sa posúvajú vďaka jednotnému hodinovému signálu. Bežná hodinová frekvencia sa pohybuje v rozsahu 1 MHz až 10 MHz, avšak pri využití dostatočne malej kapacity spoja je možné použiť frekvenciu až 70 MHz. Hlavnou nevýhodou je skutočnosť, že pri SPI je možné dáta prenášať len na krátku vzdialenosť. Na obrázku 3.5 je zobrazená obojsmerná komunikácia pomocou SPI.



Obr. 3.5: Príklad komunikácie pomocou zbernice SPI. Prevzaté z [9].

### 3.1.3 Prevodník AD

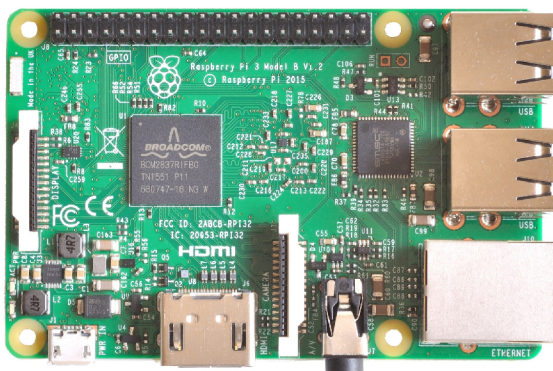
AD prevodník prevádza vstupnú analógovú veličinu na digitálnu, aby ju bolo možné spracovávať pomocou číslicových počítačov. Prevodník prevádza niekoľko činností v periodických cykloch. Vzorkovací obvod vzorkuje meranú veličinu, prevedie ju a na záver odošle prevedenú hodnotu na výstup.

## 3.2 Vývojové kity

V tejto kapitole sú opísané najznámejšie kity vhodné ako základ pre monitorovacie zariadenie, ktorému sa venuje táto bakalárska práca.

### 3.2.1 Raspberry Pi

Ako prvá a najznámejšia možnosť je Raspberry Pi, zobrazená na obrázku 3.6. Jedná sa o jednodoskový počítač vyvinutý britskou spoločnosťou *Raspberry Pi Foundation*. Najvhodnejší model tejto dosky je Raspberry Pi 3B. Doska obsahuje procesor z rodiny Cortex A5 so 4 jadrami a frekvenciou 1,2 GHz. Ďalej obsahuje 40 vstupno-výstupných pinov, 10/100 ethernetový port a pamäť RAM s kapacitou 1GB typu LPDDR2. Jeho rozmery sú 85,60 mm x 56,5 mm x 17 mm. Cena sa pohybuje okolo 40 dolárov za kus. Platforma využíva ako ukladací priestor MicroSD kartu, ktorá sa vkladá do vyvedeného slotu priamo na vývojovú dosku. Raspberry Pi ďalej disponuje konektorom na DSI displej, množstvom vstupno-výstupných pinov (známych pod skratkou GPIO), pomocou ktorých je možné dosku jednoducho rozšíriť [7].



Obr. 3.6: Vývojový kit Raspberry Pi 3B. Prevzaté z [7].

## GPIO

Raspberry Pi 3B má k dispozícii 40 GPIO pinov, ktoré sú rozložené do dvoch radov po 20, so vzdialenosťou stredov 2,54 mm. Toto rozmiestnenie sa v elektronike objavuje často a je to štandardná vzdialenosť v prototypových platformách.

Niektoré z GPIO pinov majú jednoúčelové využitie ako je možné vidieť na obrázku 3.7. Okrem univerzálnych GPIO na rozličné použitie sú dôležité napájania či už 3,3 V alebo 5 V. Piny s označením Ground označujú uzemnenie. Niektoré z pinov sú prepojené a vytvárajú určité obvody, ako napríklad piny 2 a 3 môžu slúžiť ako SDA a SCL pre zbernicu I2C.

Raspberry Pi 3 GPIO Header				
Pin#	NAME		NAME	Pin#
01	3.3v DC Power	Red	DC Power 5v	02
03	GPIO02 (SDA1 , I <sup>2</sup> C)	Blue	DC Power 5v	04
05	GPIO03 (SCL1 , I <sup>2</sup> C)	Blue	Ground	06
07	GPIO04 (GPIO_GCLK)	Green	(TXD0) GPIO14	08
09	Ground	Black	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	Green	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	Green	Ground	14
15	GPIO22 (GPIO_GEN3)	Green	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	Red	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	Purple	Ground	20
21	GPIO09 (SPI_MISO)	Purple	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	Purple	(SPI_CE0_N) GPIO08	24
25	Ground	Black	(SPI_CE1_N) GPIO07	26
27	ID_SD (I <sup>2</sup> C ID EEPROM)	Yellow	(I <sup>2</sup> C ID EEPROM) ID_SC	28
29	GPIO05	Green	Ground	30
31	GPIO06	Green	GPIO12	32
33	GPIO13	Green	Ground	34
35	GPIO19	Green	GPIO16	36
37	GPIO26	Green	GPIO20	38
39	Ground	Black	GPIO21	40

Rev. 2  
29/02/2016

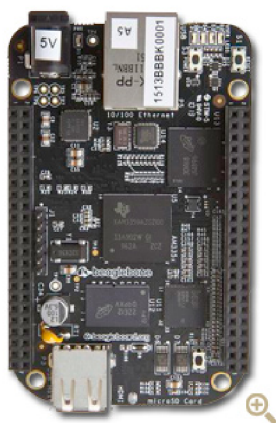
www.element14.com/RaspberryPi

Obr. 3.7: Rozloženie GPIO pinov na Raspberry Pi 3B. Prevzaté z [5].

### 3.2.2 BeagleBone Black

BeagleBone Black je lacný jednodoskový počítač s procesorom z rodiny ARM Cortex A8 s frekvenciou 1,2 GHz od firmy *Texas Instruments*. Vývojová doska BeagleBone je zobrazená na obrázku 3.8. Jeho veľkou výhodou je možnosť rozšírenia pomocou 92 pinov, rozdelených do dvoch hlavičiek, na ktoré je možné pripojiť priamo senzory, alebo celé rozširujúce moduly. Súčasťou hlavičiek je aj množstvo periférií a zberníc procesoru, vďaka ktorým je využitie tohto vývojového kitu široké [2].

Procesor je súčasťou obvodu SoC (System on Chip), v ktorom sa spolu s ním nachádza aj akcelerátor, pamäťový subsystém či real-time subsystém. Na BeagleBone je možné využívať až 6-krát UART, 2-krát SPI a 3-krát I2C. Nevýhodou však je, že nie je možné obsluhovať všetky periférie súčasne, pretože ich piny sa prekrývajú a sú konfigurovateľné na konkrétne využitie.



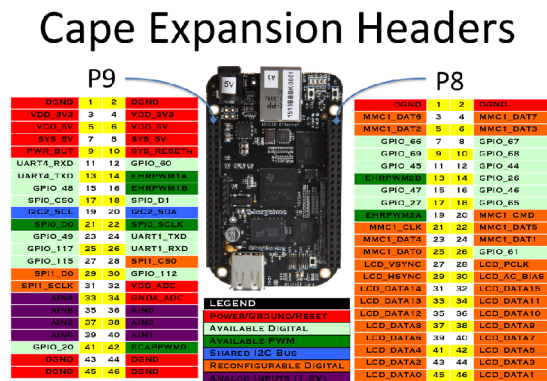
Obr. 3.8: BeagleBone Black. Prevzaté z [2].

## Napájanie

Napájanie BeagleBone Black je zaistené jednočipovým PMIC TPS65217C, ktorý môže byť napájaný cez USB port alebo DC adaptér.

## GPIO

Podobne ako u Raspberry Pi je vzdialenosť stredov 2,54 mm, ale rozdielne je rozloženie a počet pinov. Piny sú rozdelené do dvoch hlavičiek po 46 pinov. Hlavičky sú pomenované P8 a P9.



Obr. 3.9: Rozloženie GPIO pinov na BeagleBone Black. Prevzaté z [2].

Ako je vidieť na obrázku 3.9 taktiež BeagleBone Black má vyvedené napájanie 3,3 V aj 5 V, uzemnenie, všetky spomínané zbernice a periférie. Veľká výhoda oproti vyššie spomínanej platforme je prevodník AD, ktorý je priamo integrovaný vo vývojovej platforme BeagleBone Black až 7-krát aj s vlastným napájaním a uzemnením.

## AD prevodník

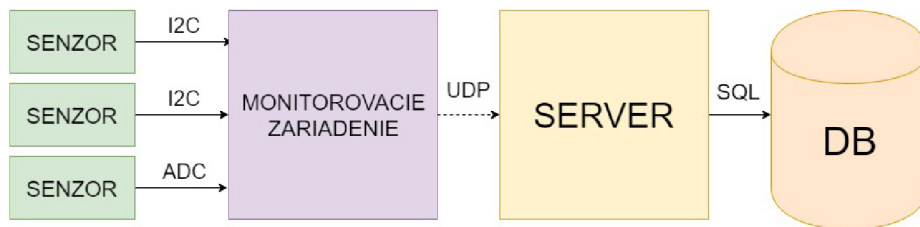
AD prevodník v BeagleBone je 7-kanálový s rozlíšením 12 bitov. Jeho vnútorná napäťová referencia je 1,8 V, čo treba brať do úvahy pri prispôbení signálu vstupnými obvodmi.

## Kapitola 4

# Návrh riešenia

V tejto kapitole je popísaný postup pri návrhu zariadenia, ktoré je možné nasadiť do priemyslu a je schopné zvládať priemyselnú záťaž v ťažkých podmienkach. Hlavné požiadavky na zariadenie sú nízka cena, rýchlosť čítania a vhodnosť nasadenia do priemyselnej výroby. Základom zariadenia je vývojová doska, na ktorú sú pripojené senzory. Zo senzorov prebieha pravidelné čítanie dát na vysokej frekvencii, výber vhodných dát a odosielanie dát na vzdialený server.

V tejto kapitole je priblížený výber jednotlivých komponentov a ich vhodnosť použitia na monitorovacie zariadenie, ktoré je možné nasadiť do priemyslu. Nachádza sa tu odôvodnenie vybranej vývojovej dosky, senzorov a návrh softvéru. Na obrázku 4.1 je zobrazená schéma navrhovaného riešenia. Celé riešenie pozostáva z troch senzorov pripojených na monitorovacie zariadenie pomocou zbernice I2C a AD prevodníku. Toto zariadenie vyhodnocuje dáta a vhodné dáta odosiela na vzdialený server. Server dáta spracuje a ukladá do lokálnej databázy.



Obr. 4.1: Schéma navrhovaného riešenia.

### 4.1 Vývojový kit

Z vyššie uvedených vývojových kitov bolo rozhodnuté pre monitorovacie zariadenie využiť BeagleBone Black. Hlavnou výhodou tohoto zariadenia je moja predošlá skúsenosť, dostatočný počet vstupno-výstupných pinov, dostatočný procesor a vnútorná pamäť. BeagleBone Black, narozdiel od Raspberry Pi, obsahuje vstavaný AD prevodník ako aj I2C či SPI zbernice, ktoré sú veľmi užitočné pri zbieraní dát zo senzorov. Jeho cena je veľmi priaznivá a rozmery sú ideálne. Ďalšou výhodou tejto dosky je, že ponúka aj priemyselnú verziu s rovnakou špecifikáciou, avšak s väčším teplotným rozsahom.

## 4.2 Operačný systém

Operačný systém je softvérový základ. Jedná sa o základ, ktorý spravuje zdroje počítača a poskytuje prístup k týmto zdrojom všetkým ďalším programom. V stručnosti ide o skupinu programov a súborov, ktoré riadia a kontrolujú činnosť hardvéru a softvéru, zaisťujú komunikáciu s užívateľom a komunikáciu s perifériami. Pri výbere operačného systému pre vývojovú dosku s jadrom Linux prichádza na možnosť viacero operačných systémov. Výhoda využitia linuxového operačného systému je práve skutočnosť, že sa jedná o otvorený operačný systém, ktorý podporuje viac užívateľský prístup. Ďalšia výhoda je jeho jednoduchá inštalácia a množstvo operačných systémov, ktoré sú prispôsobené pre vývojové dosky, s ktorými pracujem. Informácie o operačných systémoch, ktoré sú v tejto kapitole opisované sú čerpané z [6].

### 4.2.1 Android

Operačný systém Android spadá do skupiny operačných systémov s jadrom Linux. V dnešnej dobe sa tento operačný systém stáva populárny čo raz viac aj v súvislosti s vývojovou doskou BeagleBone Black. Tento operačný systém je primárne určený pre mobilné zariadenia ako sú mobilné telefóny a tablety.

V súčasnosti je pre BeagleBone Black k dispozícii niekoľko inštalačných súborov s operačným systémom Android. Existuje predstavený obraz, ktorý umožňuje zaviesť Android 4.2.2 Jelly Bean na vývojovú dosku BeagleBone Black, avšak ide len o doplnkovú možnosť pre užívateľov, ktorá nie je oficiálne podporovaná Circuitco alebo BeagleBoard.org. Pre nainštalovanie operačného systému Android 4.2.2 Jelly Bean na BeagleBone Black je nutné mať k dispozícii minimálne 4 GB microSD kartu.

Existuje open source projekt s názvom BBBAndroid. Jedná sa o aktualizovaný proces inštalácie operačného systému Android na BeagleBone Black. Ide o automatizovaný a jednoduchý spôsob pre inštaláciu AOSP Android KitKat 4.4.4 pomocou portu. Pomocou vlastného jadra Linuxu na BeagleBone Black prenáša port BBBAndroid funkciu BBB nazývanú *cape manager* do prostredia Android. Táto možnosť robí BBBAndroid vynikajúcou voľbou pre študentov ale aj profesionálov, ktorí chcú experimentovať s hardvérovým rozhraním pod operačným systémom Android.

### 4.2.2 Angstrom

Tento systém je vo vývojovej doske BeagleBone Black predinštalovaný v palubnej eMMC pamäti, ktorá slúži na uloženie systému a dát. Ide o verziu špeciálne navrhnutú pre vstavané zariadenia. V prípade potreby sú najnovšie inštalačné súbory dostupné na oficiálnych stránkach BeagleBoard.org – Latest – Images. V náväznosti na dobu kúpy BeagleBone Black, nemusí obsahovať najnovšiu verziu Angstromu a pre to je vhodné tento operačný systém pred použitím aktualizovať.

Angstrom je operačný systém vyvinutý za účelom stability a užívateľsky príjemnej distribúcie pre vstavané zariadenia. Tento operačný systém využíva *Busybox* pre mnohé kľúčové nástroje. Využitie *Busyboxu* má svoje výhody aj nevýhody. K výhodám patrí to, že vyžaduje menej úložného priestoru a má menšie nároky na pamäť, čo zlepšuje čas a výkon systému. Najväčšou nevýhodou je nekonzistentné správanie.

Angstrom používa Connam pre správu sieťových pripojení. Momentálne však nie je k dispozícii žiadna dokumentácia a taktiež nie sú poskytované manuálové údaje. To môže pri začiatku používania spôsobiť komplikácie v závislosti na úrovni skúsenosti užívateľa.

### 4.2.3 Debian

Debian je jedna z najstarších distribúcií, ktorá je nekomerčná a na jej vývoji sa podieľajú skupiny dobrovoľníkov z celého sveta. Jej základom je linuxové jadro a k nemu nabalené softvérové balíky. Výhodou využitia Debianu pre BeagleBone je, že jeho odlišnosť od bežnej verzie x86 server alebo desktop je minimálna. V okruhu linuxových užívateľov si Debian získal reputáciu kvalitného a stabilného operačného systému. Vďaka jeho obľube existuje množstvo návodov a rád od užívateľov, čo pomáha riešiť väčšinu problémov.

### 4.2.4 Fedora

Ide o projekt sponzorovaný spoločnosťou *Red Hat*. Fedora je linuxová distribúcia založená na balíčkovom systéme RPM. Produkt Fedora je open source zdrojom, ktorý je možné voľne študovať, prijímať a upravovať.

### 4.2.5 Buildroot

Buildroot je set *Makefileov*, ktoré zjednodušujú generovanie kompletného linuxového systému. Umožňuje generovať ľubovoľný súbor nástrojov kompilácie, koreňový súbor a obraz jadra. Buildroot je vhodný hlavne pre prácu so vstavanými systémami, ktoré využívajú rôzne CPU architektúry.

## 4.3 Senzory

Keďže je monitorovacie zariadenie primárne určené na zber dát z tvarovacích strojov určených k prediktívnej analýze poruchy, najdôležitejšie je snímanie vibrácií, teploty a deformácia pevných častí stroja. Preto bolo rozhodnuté použiť snímač teploty - HTU21D, akcelerometer - ADXL345, pri výbere tenzometrického modulu bola prvá voľba tenzometer - BF350-3AA, kde sa pri jeho testovaní zistilo, že kvôli chybe jeho vnútorného zapojenia výstupný signál osciluje. Na základe tejto skutočnosti bolo rozhodnuté pre vývoj vlastného tenzometrického modulu, ktorému je venovaná kapitola [4.3.3](#).

### 4.3.1 Snímač teploty - HTU21D

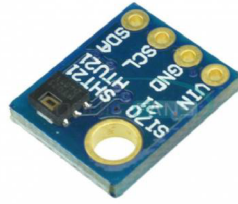
Jedná sa o miniatúrny snímač od spoločnosti *Measurment Specialities*, zobrazený na obrázku [4.2](#). Je to snímač teploty s presnosťou  $\pm 0.3$  C. V porovnaní ceny a kvality bol tento snímač najlepšia voľba. Snímač disponuje digitálnym výstupom na zbernici I2C. Zbernica je štvorvodičová, kde sú dva vodiče určené na dáta a dva sú napájacie. Adresa zariadenia na zbernici je pevne daná, čo znemožňuje pripojenia viacerých HTU21D na rovnakú zbernicu. Pre prepočet výstupu senzoru na reálnu teplotu je potrebné využiť vzorec:

$$Temp = -46,85 + 175,72 \cdot \frac{S_{Temp}}{2^{16}}$$

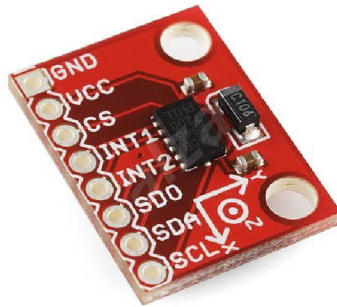
### 4.3.2 Akcelerometer - ADXL345

Akcelerometer ADXL345 je trojosí s vysokým rozlíšením a meracím rozsahom  $\pm 16g$ . Je zobrazený na obrázku [4.3](#). Na komunikáciu môže využívať zbernicu SPI aj I2C. Získané údaje odosiela vo formáte 16-bitového dvojkového komplementu.





Obr. 4.2: Snímač teploty HTU21D. Prevzaté z [19].



Obr. 4.3: Akcelerometer ADXL345. Prevzaté z [22].

### 4.3.3 Návrh tenzometra

Ako bolo zmienené na začiatku kapitoly, kvôli chybe zapojenia zakúpeného tenzometrického modulu bolo prikročené k vytvoreniu vlastného modulu. Tenzometrický modul je založený na prístrojovom zosilňovači s nastaviteľným zosilnením, ktorý zosiluje signál z tenzometrického mostu. Následne dochádza k filtrovaniu a prispôbeniu signálu na vstup AD prevodníku. Výsledné zapojenie je zobrazené na obrázku 4.4.

Zapojenie vychádza z dokumentácie prístrojového zosilňovača AD8221 [4]. Na jeho vstupy je privedené výstupné napätie tenzometrického mostu, ktorý je zložený z presných rezistorov označených na obrázku ako SG2, SG3, SG4 o hodnote  $348 \Omega$  a fóliového tenzometru SG1. Tieto hodnoty rezistorov boli zvolené na základe nízkej ceny a dobrej dostupnosti, pretože rezistory o hodnote  $350 \Omega$  sú drahé a ťažko dostupné. Týmto sa do systému zavádza počiatočný offset výstupného napätia. Tenzometrický most je budený jednosmerným napätím  $5 \text{ V}$ . Prístrojový zosilňovač má nastavené zosilnenie na hodnotu 100 pomocou rezistoru RG a má privedené referenčné napätie  $2,5 \text{ V}$  jednosmerných pre posunutie výstupného napätia do kladných hodnôt, aby bolo možné zachytiť namáhanie tenzometru na ťah aj tlak. Pre vytvorenie referenčného napätia bol použitý obvod REF3125, ktorý zo vstupných  $5 \text{ V}$  jednosmerných vytvorí precízne referenčné napätie  $2,5 \text{ V}$ . Vzťah pre výstupné napätie prístrojového zosilňovača AD8221 je opísaný v rovnici:

$$\begin{aligned}
U_A &= U_{IN} \cdot \frac{SG2}{SG2 + SG1} \\
U_B &= U_{IN} \cdot \frac{SG4}{SG4 + SG3} \\
U_{MOST} &= U_A - U_B \\
U_{AD8221} &= 100 \cdot U_{MOST} + 2,5V
\end{aligned}$$

Získaný zosilnený signál je ďalej filtrovaný filtrom typu dolná prepust' zloženým z rezistoru RF o hodnote 4,7 k $\Omega$  a keramického kondenzátora CF o kapacite 1  $\mu$ F. Vzhľadom k tomu, že deje merané tenzometrom sú pomalé, bola medzná frekvencia filtru zvolená na nízku frekvenciu 34 Hz, ktorá sa dá dosiahnuť pomocou bežne dostupných pasívnych komponentov. Filter odseparuje užitočný signál od vysokofrekvenčného šumu. Medzná frekvencia tohoto filtru je daná vzťahom uvedeným v rovnici:

$$f_{cutoff} = \frac{1}{2 \cdot \pi \cdot R \cdot C}$$

Doposiaľ opísané obvody sú následne impedančne oddelené operačným zosilňovačom ADA4807, aby nebol výstup prístrojového zosilňovača prúdovo zaťažovaný. Tento operačný zosilňovač má jeden vývod slúžiaci k jeho spusteniu, preto bol do obvodu pridaný rezistor R\_DSBL o hodnote 10 k $\Omega$ , ktorý cez jednosmerných 5 V operačný zosilňovač spustí. Na výstupe operačného zosilňovača je zapojený odporový delič, zložený z rezistorov RD1 a RD2, pre prispôsobenie úrovne meraného signálu vzhľadom k vnútornej napäťovej referencii AD prevodníka vnútri BeagleBone Black. Táto napäťová referencia má hodnotu 1,8 V, preto bol deliaci pomer odporového deliča zvolený tak, aby výstupné napätie pri nezataženom tenzometri bolo v polovici vstupného rozsahu AD prevodníka, čo je 0,9 V.

## 4.4 Návrh softvéru

Základnou požiadavkou na funkčnosť je rýchlosť a bezchybnosť. Výsledná aplikácia preto beží na strane monitorovacieho zariadenia nad operačným systémom Debian s jadrom Linux a je implementovaná v jazyku C a príjem dát prebieha na linuxovom serveri a je implementovaný v jazyku C++.

### 4.4.1 Čítanie a agregácia dát

Snahou zariadenia je čítanie dát zo senzoru maximálnou možnou rýchlosťou a ich následná agregácia. Správna agregácia dát je veľmi dôležitá. V prípade, že by sa odosieli všetky namerané dáta, bolo by ich uchovávanie a práca s nimi náročná. V prípade nesprávnej agregácie, naopak môžu chýbať dôležité informácie a výsledky analýz tak budú nesprávne. Výber dát, ktorý sa zo zariadenia odosiela, je konfigurovateľný podľa dvoch parametrov. Prvý parameter je časová hodnota. Ak sa za tento čas nevybrali žiadne dáta na odoslanie, automaticky sa jedna hodnota odošle. Ďalším parametrom je prahová hodnota senzoru. Táto prahová hodnota je špecifická pre každý druh senzoru. Po čítaní zo senzoru prebehne kontrola, či sa daná hodnota od predchádzajúcej zmenila minimálne o určenú prahovú hodnotu. V prípade splnenia tejto podmienky sa dáta označia za dôležité a sú odosielané na uloženie. Obrázok 4.5 tento výber opisuje.

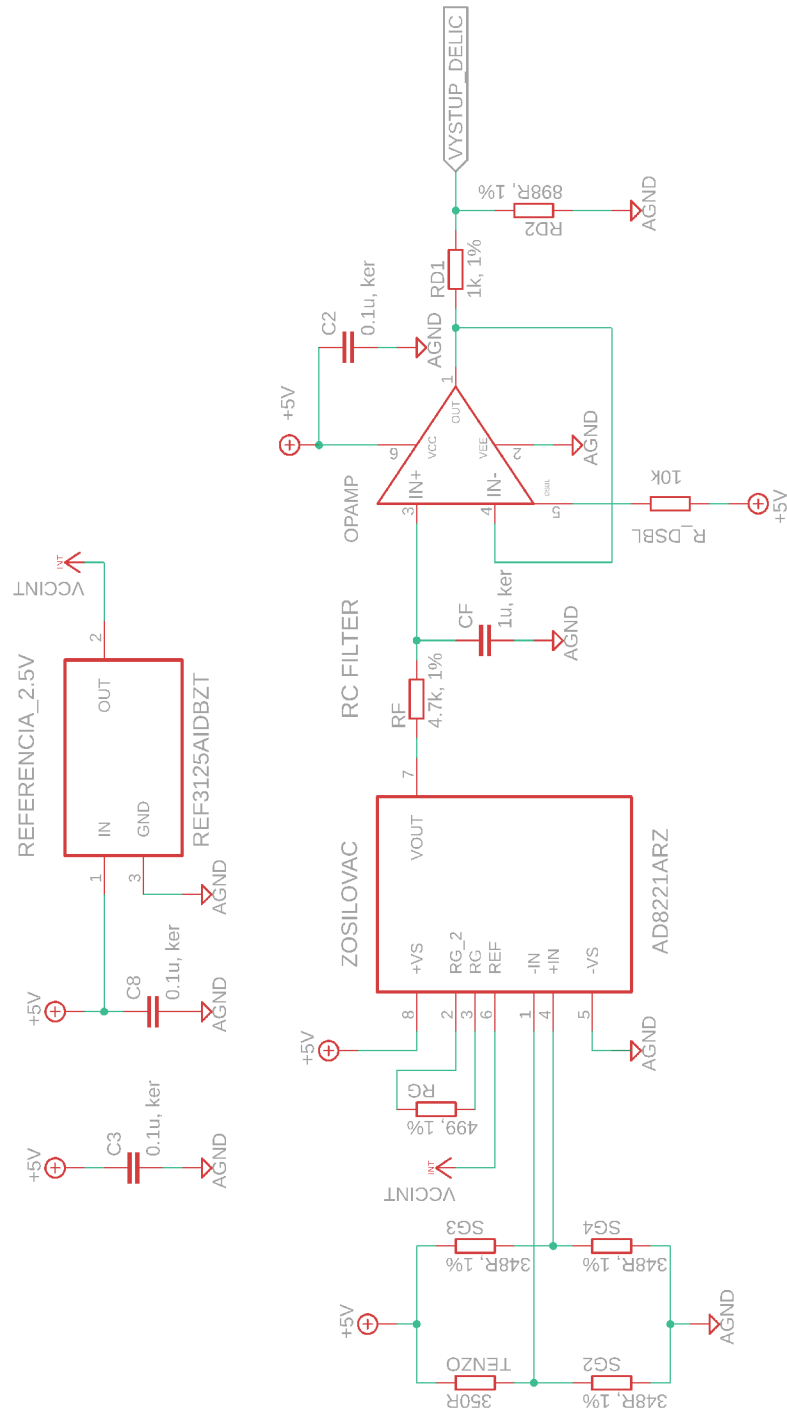
#### 4.4.2 Odosielanie dát

Pre odosielanie dát sa využíva jednoduchý komunikačný protokol, ktorý beží pod UDP. Protokol sa skladá z hlavičky, ktorá má pevnú veľkosť. V hlavičke sa prenáša jednoznačný identifikátor zariadenia, ktoré správu odosiela, verzia protokolu a čas odoslania správy. Po hlavičke nasledujú dátové vstupy, ktorých počet je premenlivý podľa počtu senzorov, z ktorých zariadenie číta dáta. Dátový vstup obsahuje identifikátor druhu senzoru a teda presne špecifikuje o aký typ dát sa jedná. Ďalej obsahuje dĺžku dát a samotné dáta. Dáta sú odosielané v štruktúre, ktorá obsahuje hodnotu a časovú značku. Protokol je zobrazený na obrázku 4.6.

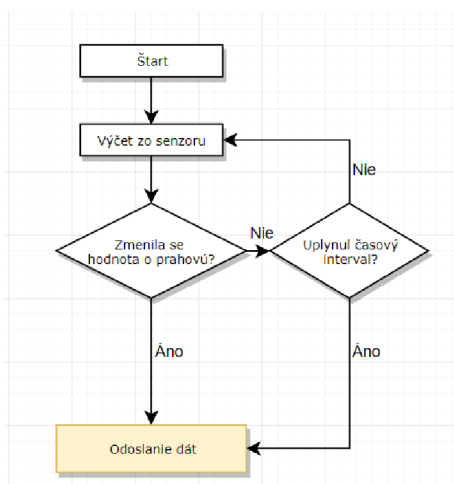
#### 4.4.3 Prijímanie a ukladanie dát

Prijímanie dát na strane serveru musí byť spoľahlivé a rýchle, preto je ideálna voľba programovací jazyk C/C++. Pre jednoduchosť zaistenia konkurentnosti je vhodné obmedziť veľkosť odosielaných dát na jeden paket.

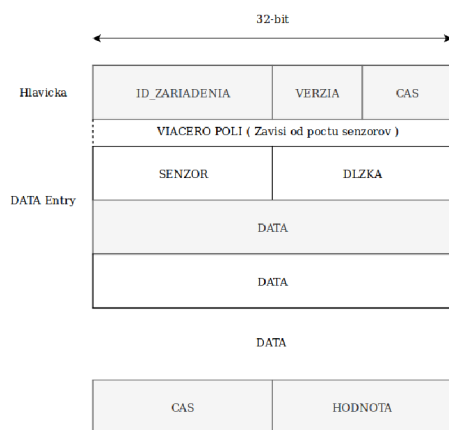
Na obrázku 4.7 je vidieť, že v prípade odoslania viacerých paketov od jedného klienta server nespozná, že sa jedná o toho istého klienta a požiadavky dokáže spracovať iba jednotlivu, čo však pri použití opisovaného protokolu nie je problém. Po prijatí správy je potrebné získané dáta uložiť do databázy.



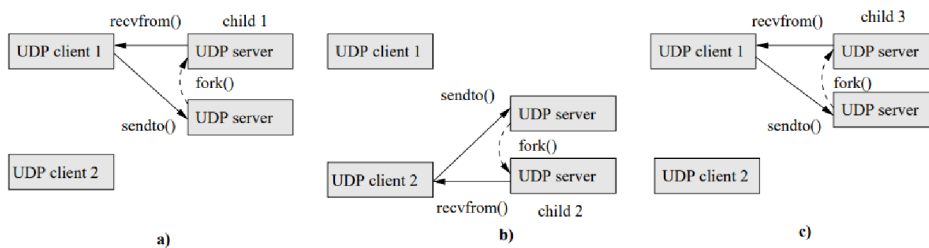
Obr. 4.4: Schéma tenzometrického modulu.  
24



Obr. 4.5: Výber vhodných dát zo senzoru.



Obr. 4.6: Komunikačný protokol.



Obr. 4.7: Jednoduchý konkurentný server UDP. Prevzaté z [16].

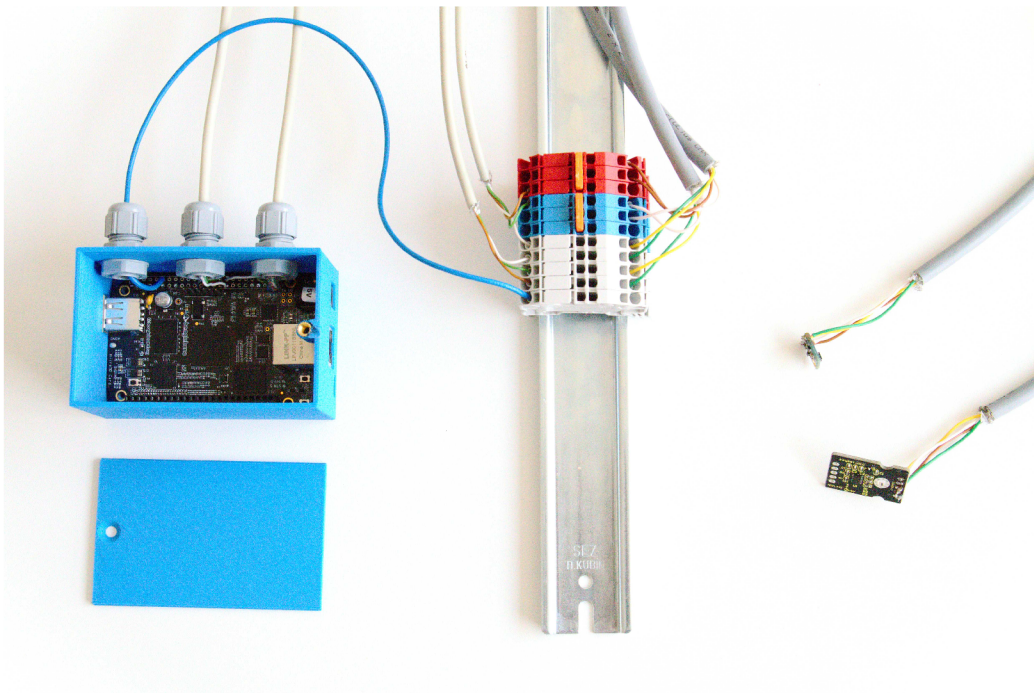
# Kapitola 5

## Implementácia

Táto kapitola je rozdelená na dve hlavné podkapitoly, pričom sa prvá časť venuje príprave všetkých zvolených komponentov na použitie. V druhej časti je popísaný postup implementácie monitorovacieho zariadenia a postup jeho inštalácie.

### 5.1 Hardvér

Výsledkom hardvérového spracovania je monitorovacie zariadenie, ktoré vydrží priemyselné podmienky. Na obrázku 5.1 je zobrazené výsledné monitorovacie zariadenie.



Obr. 5.1: Výsledné pripojenie senzorov na monitorovacie zariadenie.

### 5.1.1 BeagleBone Black

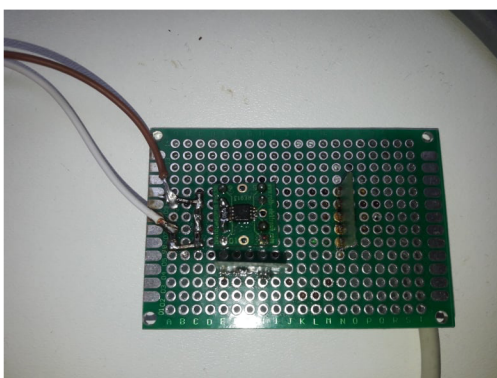
Základná úprava vývojového kitu spočíva v pripojení potrebných senzorov a vytvorení ochranného krytu, ktorý chráni BeagleBone pred poškodením. Keďže sú všetky potrebné výstupy umiestnené na hlavičke P9, ako je možné vidieť na obrázku 3.9, bolo pre správne zapojenie potrebné odinštalovať hlavicu a na vyvedené výstupy napevno pripájať vodiče. Všetky potrebné výstupy sú vyvedené z vývojového kitu cez použité priemyselné priechodky do svorkovnice umiestnenej na DIN lište. Použité vstupy sú DGND, ktorý slúži na uzemnenie, VDD\_3v3 pre napájanie 3,3 V, ktoré využívajú oba snímače pripojené na I2C zbernica, VDD\_5V pre napájanie 5 V, ktoré využíva zostavený tenzometrický modul. Ďalej sú potrebné výstupy I2C2\_SCL a I2C2\_SDA pre komunikáciu s i2c-1 perifériou a SPI0\_D0 a SPI0\_SCLK pre komunikáciu s perifériou i2c-2. Výstup AIN2 slúži ako vstup do AD prevodníku pre tenzometrický modul. Zvyšné výstupy zostali nezapojené.

### 5.1.2 HTU21D a ADXL345

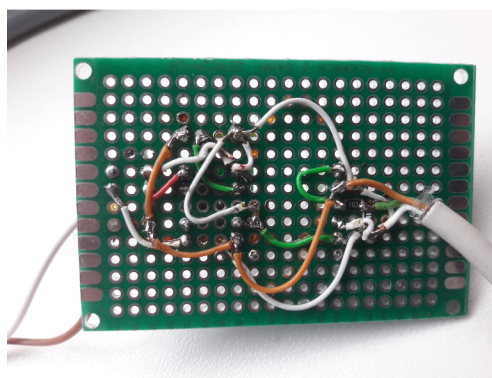
Keďže sa jedná o zakúpené senzory, ich nasadenie nežiada žiadne ďalšie rozšírenie, a tak bolo potrebné zvoliť vhodné vodiče a pripájať senzory na pripravenú svorkovnicu. Oba senzory komunikujú cez I2C, čo znamená, že ku každému senzoru vedú dva vodiče na napájanie a dva na prenos dát. Preto bolo rozhodnuté použiť štvoržilové vodiče s krútenými párami, ktoré sú odolné a vhodné na využitie v priemysle.

### 5.1.3 Tenzometrický modul

Základom tenzometrického modulu je univerzálny plošný spoj, ktorý obsahuje 280 bodov. Jedná sa o obojstrannú prototypovú dosku vyrobenú z laminátu FR4 s hrúbkou 1,6 mm, na ktorej sa nachádzajú pocínované spájkovacie body. Rozmery plošného spoja sú 60 x 40 mm. Vstupom do tenzometrického modulu sú 3 vodiče pripojené na výstupy BeagleBone a to VDD\_5V, AIN2 a DGND. Toto pripojenie je možné vidieť v pravej časti obrázku 5.3. Privedené vodiče sú pripojené podľa schémy 4.4 k operačnému zosilňovaču ADA4807. Ďalej sa na plošnom spoji nachádza prístrojový zosilňovač AD8221, tenzometrický most, obvod REF3125, filter dolnej priepustnosti a všetky ďalšie komponenty opísané v kapitole 4.3.3. Výstupom tenzometrického modulu je samotný tenzometer, ktorý tvorí poslednú časť tenzometrického mostu. Tenzometer je pripojený k plošnému spoju dvoma vodičmi. Toto pripojenie je zobrazené na ľavej časti obrázku 5.2.



Obr. 5.2: Tenzometrický modul (pohľad zhora).



Obr. 5.3: Tenzometrický modul (pohľad zdola).

## 5.2 Softvér

Táto kapitola popisuje všetky softvérové úpravy, ktoré boli vykonané na vývojovom kite a serveri pre funkčnosť práce.

### 5.2.1 Inštalácia operačného systému

Prvá časť práce s vývojovým kitom BeagleBone Black je samotná inštalácia operačného systému. Keďže jedna z požiadaviek na zariadenie bol operačný systém s jadrom Linuxu, bolo rozhodnuté pre *Debian*, voľne stiahnuteľný na oficiálnej stránke BeagleBone [2], ktorý je špeciálne upravený pre IoT zariadenia. Tento operačný systém je ponúkaný v dvoch verziách, s grafickým rozhraním alebo bez grafického rozhrania. Pre monitorovacie zariadenie nie je grafické rozhranie potrebné, tak sa rozhodlo pre druhú variantu, ktorá šetrí pamäť. Momentálne najnovšia a doporučená verzia je 9.5.2018-10-7. Po stiahnutí operačného systému je potrebné naboťovať ho na SD kartu. Po pripojení SD karty k počítaču je potrebné využiť softvér na správne nahranie operačného systému na SD kartu. Na nahranie bol použitý *usb image writer*, ktorý je voľne dostupný pre linuxové zariadenia od verzie 12.10 vyššie. Po vytvorení bootovacej SD karty je potrebné vložiť kartu do BeagleBone, ktorý je vypnutý. Ďalej je potrebné držať bootovacie tlačítko, ktoré je umiestnené v pravom hornom rohu vývojového kitu a pritom priviesť na dosku prúd pomocou USB kábla, alebo 5 V adeptéra. Po tomto kroku stačí vyčkať, kým sa nerozsvietia všetky 4 LED svetlá v pravom hornom rohu vývojového kitu. Operačný systém je úspešne nainštalovaný.

### 5.2.2 Inicializačný skript

Pretože nie všetky nastavenia, ktoré sú pre správnu funkčnosť monitorovacieho zariadenia potrebné sú predvolené, je potrebné vytvoriť inicializačný skript, ktorý sa spustí po zapnutí zariadenia a postupne sú do neho pridané všetky potrebné príkazy. Pre vytvorenie skriptu, ktorý sa v Linuxe vykoná po spustení, je potrebné vytvoriť súbor v zložke */etc/systemd/system*. Tento súbor musí byť vo formáte *meno.service*. Do tohto súboru je potrebné vpísať názov s celou cestou k programu, ktorý sa má spustiť. Na obrázku 5.4 je znázornený obsah súboru *startbp.service*, ktorý pri spustení BeagleBone spustí skript *start\_bp.sh*, ktorý je uložený v zložke */usr/bin*. Skript otvára potrebné periférie BeagleBone a viac o ňom je napísané v podkapitole 5.2.4.

```
[Unit]
Description=kod na spustenie bakalarskej prace
After=syslog.target network.target
[Service]
Type=simple
ExecStart=/usr/bin/start_bp.sh
[Install]
WantedBy=multi-user.target
```

Obr. 5.4: Spustenie inicializačného skriptu pomocou service.



### 5.2.3 Sieťová konfigurácia

Po úspešnej inštalácii operačného systému je nevyhnutná pre monitorovacie zariadenie správna sieťová konfigurácia, keďže sa všetky namerané dáta odosielaajú na vzdialený server. Keďže sa monitorovacie zariadenie pripravuje na využitie v priemysle, je pravdepodobné, že bude musieť využívať statickú IP adresu, ktorá mu bude priradená. Na konfiguráciu slúži súbor *interfaces*, ktorý sa nachádza na ceste */etc/network*. Do tohoto súboru treba vložiť konfiguráciu každého používaného rozhrania. Nastavenie statickej IP adresy je znázornené na obrázku 5.5. V prípade voľnosti je možné využiť pripojenie k sieti pomocou DHCP, ktoré je znázornené na obrázku 5.6.

```
auto eth0
iface eth0 inet static
address 172.16.23.77
netmask 255.255.255.0
gateway 172.16.23.254
dns-nameservers 8.8.8.8
```

Obr. 5.5: Konfigurácia v */etc/network/interfaces* pre statickú IP adresu.

```
auto eth0
allow-hotplug eth0
iface eth0 inet dhcp
```

Obr. 5.6: Konfigurácia v */etc/network/interfaces* pre dynamickú IP adresu.

### 5.2.4 Otvorenie periférií

Monitorovacie zariadenie využíva dve I2C periférie, no na doske je v prednastavených nastaveniach po zapnutí BeagleBone otvorená len jedna, preto je potrebné, aby sa pri spustení otvorili obe a boli pripravené na použitie. Ďalej je potrebné otvorenie periférie AD prevodníka. O to sa stará skript, ktorý sa spustí pri bootovaní BeagleBone. Tento skript zapíše do súboru */sys/devices/bone\_capemgr.9/slots* riadky *BB-I2C1* a *BB-ADC*, ktoré slúžia pre otvorenie potrebných periférií.

### 5.2.5 Vývojové prostredie

Ako vývojové prostredie pre aplikácie pre BeagleBone Black programované v jazyku C bolo zvolené Eclipse CDT. Jedná sa o plne funkčné vývojové prostredie C a C++ založené na platforme Eclipse. Výhodou je, že je voľne stiahnutelné na oficiálnej stránke [8].

Po jeho nainštalovaní je potrebné doinštalovať pomocou možnosti *Help/Install New Software..* doplnok *C/C++ Development Tools*. Po nainštalovaní doplnkov ponúka Eclipse možnosť nastavenia prekladu na konkrétny procesor, ktorý sa nachádza na vývojovej doske a tým umožňuje vytvoriť spustiteľný program priamo na počítači a na zariadenie priamo nahrat výslednú aplikáciu. K tomuto nastaveniu sa dá dostať cestou *Project/Properties/C/C++ Build/Settings*. Pre kompiláciu projektu pre BeagleBone Black je potrebné nastaviť *Command* : *arm-linux-gnueabi-gcc*.

Pre debugovanie je využitý doplnok *gdb-server*. Po jeho nainštalovaní na vývojový kit pomocou príkazu *apt-get install gdbserver* je možné nahrať spustiteľný súbor debugovať pomocou vývojového prostredia Eclipse CDT.

### 5.2.6 Aplikácie pre čítanie dát

Každý senzor je niečím špecifický, no základnú funkcionálnu čítania a odosielenia dát majú všetky senzory rovnakú. Na začiatku aplikácie sa vytvorí soket na odosielenie nameraných

hodnôt na lokálnu aplikáciu. Nasleduje otvorenie *file descriptoru* pre príslušnú perifériu senzoru. Po inicializácii nasleduje nekonečný cyklus, v ktorom sa volá funkcia na čítanie dát zo senzoru. Pseudokód tohoto algoritmu je znázornený na obrázku 5.7. Táto funkcia si na začiatku vždy zistí aktuálny čas pomocou funkcie *gettimeofday* z knižnice *time.h*. Po načítaní času prečíta prvú hodnotu zo senzoru. Po prečítaní hodnoty sa rozhoduje, či hodnota prekročila rozdiel minimálne o prahovú hodnotu, alebo prešla minúta od posledného odoslania hodnoty. V prípade splnenia podmienky sa hodnota aj s aktuálnym časom zapíše do štruktúry *sprava* a pomocou funkcie *sendudp* sa odosiela do lokálnej aplikácií na spracovanie hodnoty a neskoršie odoslanie na vzdialený server. Pseudokód je znázornený na obrázku 5.8.

```

1  /*
2     Vytvorenie socketu
3  */
4  if (( sock = socket(AF_INET, SOCK_DGRAM, 0)) < 0){
5      exit(2);
6  }
7
8  address.sin_family = AF_INET;
9  address.sin_port = htons(CISLO_PORTU);
10 inet_pton(AF_INET, IP_ADDRESS, &address.sin_addr);
11
12 /*
13     Otvorenie zbernice
14 */
15 char *bus = "CESTA_K_ZBERNICI";
16 if((file = open(bus, O_RDWR)) < 0){
17     exit(1);
18 }
19 /*
20     Nekonecny cyklus vycitania dat
21 */
22 while(1){
23     funkcia_citania_zo_senzoru();
24 }

```

Obr. 5.7: Pseudokód základu aplikácie pre čítanie dát zo senzoru.

```

1  funkcia_citania_zo_senzoru(){
2      //nacitanie casu
3      cas = gettimeofday();
4      //citanie zo senzoru
5      hodnota_zo_senzoru = read(file);
6
7      if(abs( hodnota_zo_senzoru - PREDCHADZAJUCA_HODNOTA ) > PRAHOVA_HODNOTA ||
8          (CAS_POSLEDNEJ_HODNOTY + MINUTA ) > cas){
9          send_udp(hodnota_zo_senzoru,cas);
10     }
11 }

```

Obr. 5.8: Pseudokód funkcie na čítanie zo senzoru.

### 5.2.7 Aplikácia pre odosielanie dát

Aplikácia slúži na prijímanie dát z jednotlivých senzorov. Tieto dáta spracováva a ukladá si ich do svojej štruktúry a pri naplnení podmienok na odosielanie správu odosiela na vzdialený server.

Na začiatku aplikácie prebieha inicializácia, ktorej súčasťou je vytvorenie soketu na prijímanie dát z lokálnych aplikácií, ktoré odosielajú dáta zo senzorov. Po tejto inicializácii nasleduje nekonečný cyklus prijímania dát. V tomto cykle aplikácia prijíma správu. Z tejto správy vytiahne informácie a ukladá prijaté dáta do vytvorenej štruktúry *sprava*. Táto správa sa postupne naplňa a keď dosiahne potrebnú dĺžku, tak sa prijaté dáta odosielajú na vzdialený server pomocou funkcie *sendudp*. Správa na odoslanie sa vynuluje a začína sa naplňať odznova. Funkcia *sendudp* vytvorí soket na odoslanie a odosiela pripravenú správu na vzdialený server pomocou funkcie *sendto* z knižnice *socket.h*.

### 5.2.8 Serverová aplikácia

Serverová aplikácia je implementovaná v jazyku C++. Jej hlavný účel je prijímanie správ odoslaných z monitorovacích zariadení, spracovanie správy a jej následné uloženie do databázy.

Na začiatku si aplikácia vytvorí zásobník s konštantnou veľkosťou 512 bajtov, čo je maximálna dĺžka paketu používaného protokolu. Nasleduje pripojenie na databázu, do ktorej sa prijaté dáta ukladajú. Po pripojení na databázu sa vytvorí soket, pomocou ktorého sa prijímajú dáta a ukladajú do vytvoreného zásobníku. Po inicializačnej časti nasleduje nekonečný cyklus, kde aplikácia čaká na prijatie správy. Po prijatí správy sa využije funkcia *fork* na rozdelenie programu na 2 procesy. Rodičovský proces ja pripravený na prijatie ďalšej správy a detský proces pokračuje v čítaní prijatej správy. Pomocou tejto funkcie sa zaisťuje konkurentnosť aplikácie a jej schopnosť pracovať s viacerými správami súčasne. Detský proces následne prechádza správu, rozdelí si ju na jednotlivé časti a získané dáta ukladá do databázy. Všetky dáta sú uložené do jednej tabuľky. Príklad uložených dát v databáze je zobrazený na obrázku 5.9.

### 5.2.9 Spúšťanie aplikácií

Pre spúšťanie aplikácií po inicializácii a kontrolu ich chodu sa využíva *Cron*. Jedná sa o nástroj pre administráciu systému. Cron sa spúšťa pri štarte systému a aktivuje sa pravidelne každú minútu a kontroluje svoje *crontabs*. Jedná sa o súbory, kde sú definované úlohy s časom ich spustenia. Výhodou je fakt, že Cron je súčasťou všetkých unixových systémov.

Základom funkčnosti sú štyri aplikácie, jedna pre každý senzor a hlavná aplikácia pre prijímanie dát zo všetkých senzorov, ktorá sa stará o odosielanie správ na vzdialený server. Na obrázku 5.10 je zobrazené nastavenie, ktoré spustí 10 sekúnd po boote aplikáciu na prijímanie dát a 20 sekúnd po boote aplikácie na čítanie dát. Ďalšie riadky slúžia na pravidelnú kontrolu každé 2 hodiny, či sú spustené všetky aplikácie, a v prípade že nie, sú opätovne zapnuté.

### 5.2.10 Čítanie dát z I2C

Pri pripojení senzoru na I2C zbernicu BeagleBone je potrebné otvorenie periférie, čo je vysvetlené v podkapitole 5.2.4. Po otvorení periférie je potrebné otvorenie *file descriptoru* ako je znázornené na obrázku 5.7. Cesta k súboru pre prácu s I2C sa v Debiane nachádza

id	senzor	time	value
33361	1	1553763856701	28.16925
33362	1	1553763861918	28.098755
33373	2	1553763862041	146
33374	2	1553763862084	152
33375	2	1553763862242	147
33376	2	1553763862399	152
33377	2	1553763862479	148
33378	2	1553763862596	152
33379	2	1553763862638	145

Obr. 5.9: Dáta uložené v databáze na serveri.

```
@reboot sleep 10 && /root>HelloWorld/senzorserver/senzorserver
@reboot sleep 20 && /root>HelloWorld/tempenzor/tempenzor
@reboot sleep 20 && /root>HelloWorld/accsenzor/accsenzor
@reboot sleep 20 && /root>HelloWorld/adcsenzor/adcsenzor

0 */2 * * * [ `ps -ef|grep -v grep|grep senzorserver` -eq 0 ] && /root>HelloWorld/senzorserver/senzorserver
0 */2 * * * [ `ps -ef|grep -v grep|grep tempenzor` -eq 0 ] && /root>HelloWorld/tempenzor/tempenzor
0 */2 * * * [ `ps -ef|grep -v grep|grep accsenzor` -eq 0 ] && /root>HelloWorld/accsenzor/accsenzor
0 */2 * * * [ `ps -ef|grep -v grep|grep adcsenzor` -eq 0 ] && /root>HelloWorld/adcsenzor/adcsenzor
```

Obr. 5.10: Spustenie a kontrola aplikácií pomocou Cronu.

v zložke *dev*, pri aktuálnej konfigurácii sa v zložke nachádzajú dva súbory pre I2C a to *i2c-1* a *i2c-2*. *I2c-1* slúži na prácu so snímačom teploty a *i2c-2* na prácu s akcelerometrom. Pre správnu komunikáciu treba vždy senzoru odoslať úvodnú správu I2C protokolu, v ktorej je zapísaná adresa senzoru, ako bolo spomínané pri opise zbernice I2C. Pre túto správu je určená funkcia *ioctl* z knižnice *ioctl.h*, kde ako prvý argument treba zadať *file descriptor*, potom *I2C\_SLAVE*, čo je *define* z knižnice *i2c-dev.h* a ako posledný argument adresu senzoru. Táto adresa môže byť pevne daná výrobcom a je možné ju dohľadať v datasheete senzoru, alebo existujú nástroje pre ich detekciu. V prípade oboch používaných senzorov je adresa pevne daná. Pre akcelerometer ADXL345 je to adresa *0x53* a pre snímač teploty HTU21D *0x40*. Na obrázku 5.11 je znázornený pseudokód čítania dát zo snímaču teploty opísaný v tejto kapitole.

### 5.2.11 Vstavaný AD prevodník

Podobne ako pri I2C zbernici existuje v operačnom systéme súbor, pomocou ktorého je možné čítať dáta zo vstavaného AD prevodníka na BeagleBone. Tento súbor je umiestnený v zložke */sys/bus/iio/devices/iio:device0/in\_voltage2\_raw*. Keďže má BeagleBone až 7 vstavaných AD prevodníkov číslo 2 v zložke určuje, o ktorý prevodník sa jedná. Podľa schémy je výstup tenzometru práve na AIN2. Po otvorení *file descriptoru* stačí už len čí-

```

1  /*
2  otvorenie file descriptoru
3  */
4  char *bus = "/dev/i2c-1"
5  if((file = open(bus, O_RDWR)) < 0){
6      exit(1);
7  }
8  /*
9  uvidna i2c sprava s adresou SLAVE
10 */
11 ioctl(file,I2C_SLAVE, 0x40);
12 /*
13 vyzva senzoru pre odoslanie hodnoty
14 */
15 write(file, "0xFE", 1)
16 /*
17 nactanie ziskanej hodnoty do premennej data
18 */
19 read(file,data,2);

```

Obr. 5.11: Pseudokód čítania dát snímača teploty HTU21D na I2C zbernici.

tať hodnoty, ktoré vytvorený tenzometer neustále zapisuje do súboru. Na obrázku 5.12 je znázornené čítanie z AD prevodníka.

```

1  #define ADC_PATH "/sys/bus/iio/devices/iio:device0/in_voltage2_raw"
2  /*
3  otvorenie filedescriptoru read-only
4  */
5  if((file = open(ADC_PATH, O_RDONLY) < 0){
6      exit(2);
7  }
8  /*
9  vycitanie hodnoty z adc do premennej data
10 */
11 read(file, &data, 4);

```

Obr. 5.12: Pseudokód čítania dát z AD prevodníka.

### 5.2.12 Zhrnutie softvéru

Pri spustení zariadenia, ktoré má nakonfigurované sieťové rozhranie, sa spustí inicializačný skript, ktorý otvorí potrebné periférie. Nasleduje práca Cronu, ktorý spustí tri aplikácie samostatne pre každý senzor a hlavnú aplikáciu, ktorá prijíma vhodné dáta zo senzorov a odosiela ich na vzdialený server. Aplikácie využívajú UDP sokety, zbernici I2C a vstavaný AD prevodník. Kontrola chodu aplikácií sa pravidelne opakuje opäť za pomoci Cronu.

## 5.3 Návod k inštalácii

V tejto časti práce je popísané ako postupovať pri nasadení opisovaných implementačných častí pre vytvorenie monitorovacieho zariadenia.

1. Nainštalovanie operačného systému na vývojový kit BeagleBone Black podľa návodu [5.2.1](#).
2. Nastavenie sieťovej konfigurácie podľa [5.2.3](#).
3. Vytvorenie servisu na spúšťanie inicializačného skriptu a jeho vytvorenie podľa [5.2.2](#).
4. Nahratie priložených spustiteľných aplikácií a pridanie ich automatického spustenia pomocou Cronu podľa [5.2.9](#).
5. Vyviest potrebné výstupné piny z hlavičky P9 na svorkovnicu podľa [5.1.1](#) pomocou vhodnej kabeláže.
6. Pripojenie senzorov na svorkovnicu.
7. Osadenie senzorov na sledovaný prístroj.
8. Pripojenie vývojového kitu k internetu a pripojenie napájania.
9. Spustenie serverovej aplikácie na serveri, ktorý má povolené potrebné porty.

## Kapitola 6

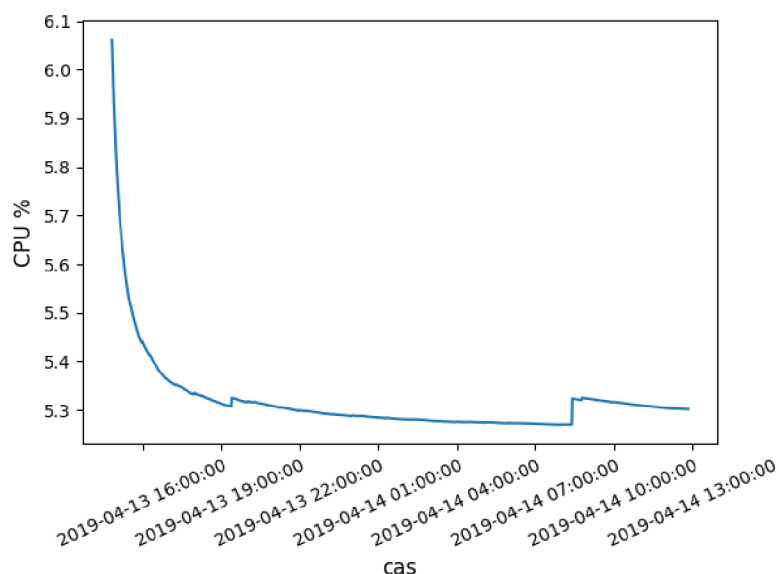
# Experimenty a vyhodnotenie

Vo fáze testovania monitorovacieho zariadenia prebehli štyri experimenty. Prvý testuje činnosť a plynulosť zariadenia po dobu 24 hodín bez záťaže. Pri experimente bez záťaže nie sú senzory osadené na prístroji, sú len voľne položené. Výstupom experimentu by mali byť takmer konštantné hodnoty dát zo sensorov. Pri tomto experimente sú okrem dát zozbierané štatistické údaje o vyťažení procesora zariadenia a veľkosť sieťového chodu. Druhý experiment spočíva v porovnaní senzoru teploty monitorovacieho zariadenia so senzorom pripojeným na zariadenie firmy, v spolupráci s ktorou je táto práca vytvorená. Sensory oboch zariadení sú umiestnené v tesnej blízkosti a snímajú dáta. Po uplynutí testovacej doby sú dáta, zozbierané oboma senzormi, porovnané. Tretí experiment vykonáva zber dát v priebehu niekoľkých dní zo všetkých sensorov pripojených na monitorovacie zariadenie. Posledný experiment porovnáva hodnoty získané z monitorovacieho zariadenia pripojeného k testovacej podložke s dátami nameranými na rovnakom testovacom nástroji na existujúcom riešení partnerskej firmy. Po prevedení experimentov nasleduje ich zhrnutie a interpretácia.

### 6.1 Prvý experiment: Overenie agregácie a vyťaženia

Monitorovacie zariadenie bolo pripojené k sieti po dobu 24 hodín, kde odosiela dáta zo všetkých troch sensorov, ktoré boli voľne položené v kancelárskych priestoroch. Predmetom skúmania tohoto experimentu je vyťaženie procesora, počet odoslaných paketov na vzdialený server. Je predpoklad, že počet paketov nebude veľký, pretože voľne položené senzory by nemali výrazne meniť namerané hodnoty a tak v rámci agregácie dát sa zaznamená len jeden údaj za minútu, kde snímač teploty zaznamenáva teplotu v miestnosti, prípadne tenzometrický modul, alebo akcelerometer zachytí šum v budove.

Sledovanie vyťaženia procesora prebiehalo pomocou vytvoreného skriptu, ktorý sa spúšťal pravidelne každú minútu pomocou Cronu. Práca s Cronom je opísaná v kapitole 5.2.9. Na volanie skriptu je využité nastavenie `* * * * * /root/cpuusage.sh`, vďaka ktorému sa skript `cpuusage.sh` spúšťa každú minútu. Skript zapíše do textového súboru `cpulog.txt` percentuálne využitie procesora a časovú značku v podobe timestampu. Využitie procesora sa nachádza v súbore `/proc/stat` a timestamp sa dá získať pomocou príkazu `date` s prepínačom `+%s`. Po uplynutí 24 hodín bol z výpisu využitia procesora `cpulog.txt` pomocou knižnice `matplotlib` v Pythone vykreslený graf, ktorý je na obrázku 6.1. Po spustení monitorovacieho zariadenia bolo využitie procesoru 6 % a postupne sa ustálilo na hodnote 5 % počas celého trvania experimentu.



Obr. 6.1: Prvý experiment: Využitie CPU.

Na sledovanie počtu a veľkosti odoslaných dát bola využitá aplikácia *ntop*, ktorá je voľne stiahnuteľná. Po dobu 24 hodín prišlo na vzdialený server 314 paketov, ktoré mali celkom veľkosť 101,83 kB. Na tomto príklade je dobre vidieť vhodnosť agregácie a fakt, že sa zbytočne neodosieli záznamy zo senzorov v dobe, keď zmena ich hodnôt neprekročila prahovú hodnotu.

## 6.2 Druhý experiment: Porovnanie snímaču teploty

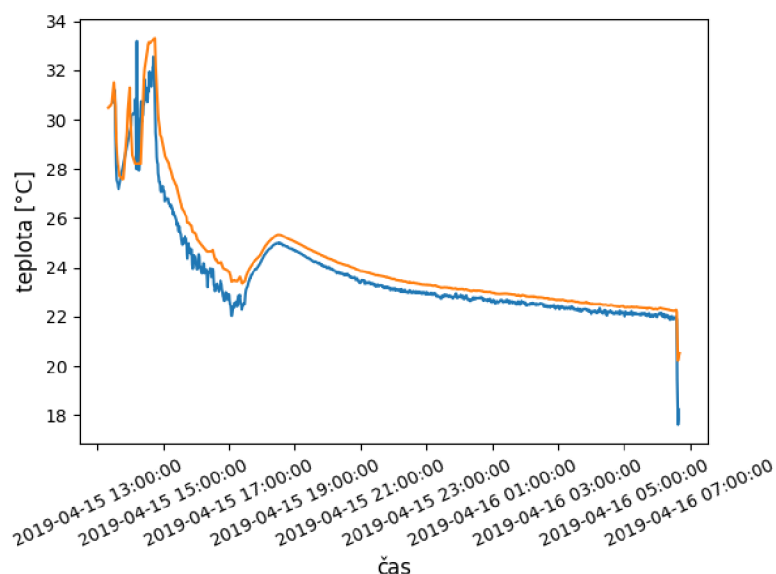
Počas tohoto experimentu bol v tesnej blízkosti snímača teploty HTU21, ktorý je pripojený na monitorovacie zariadenie, zavedený snímač teploty PT1000, pripojený na monitorovacie zariadenie od firmy National Instruments. Snímač teploty HTU21D je popísaný v kapitole 4.3.1. Predmetom skúmania bol rozdiel v zozbieraných dátach pomocou existujúceho riešenia od partnerskej firmy a monitorovacieho zariadenia. Výsledky experimentu je možné vidieť na grafe zobrazenom na obrázku 6.2. Ako je jasne vidieť z grafu, hodnoty oboch senzorov sú podobné. Oranžová farba značí hodnoty namerané zariadením CompactRIO a modrá farba patrí monitorovaciemu zariadeniu.

## 6.3 Tretí experiment: Dlhodobý zber dát

V tomto experimente prebiehalo vyčítanie dát súčasne zo všetkých senzorov po dobu niekoľkých dní. Experiment ukazuje podobnosť výsledkov počas dňa. Na grafoch na obrázkoch 6.3, 6.4 a 6.5 sú zobrazené namerané hodnoty. Ako je vidieť na obrázkoch, keďže je priebeh každého dňa skoro rovnaký, aj 24 hodinové úseky na grafoch sú podobné. Týmto experimentom bola otestovaná schopnosť plynulého zberu dát počas dlhšej časovej periódy a zároveň overenie správnosti dát na základe porovnania dát z jednotlivých dní s ostatnými.

Na obrázkoch 6.4 a 6.5 je názorne zobrazená súvislosť medzi hodnotami získanými z tenzometrického modulu a snímača teploty, ktorá je spôsobená teplotnou rozťažnosťou



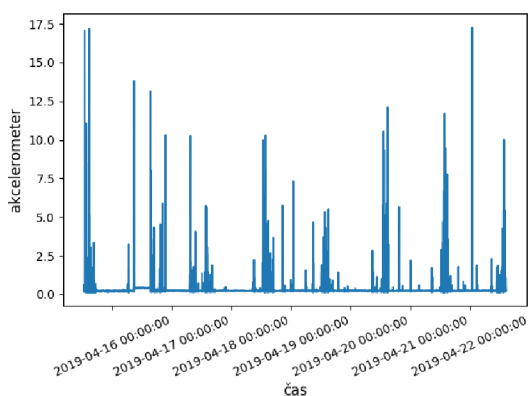


Obr. 6.2: Druhý experiment: Porovnanie dát z monitorovacieho zariadenia a CompactRIO.

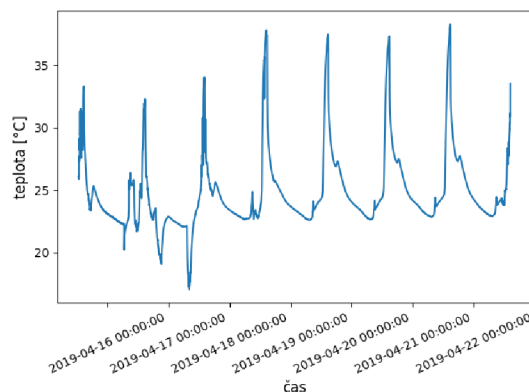
materiálu, na ktorom je nalepený tenzometer. Tento jav poukazuje na dôležitosť merania viacerých veličín zároveň, pretože zmena jednej veličiny môže spôsobiť zmenu inej.

#### 6.4 Štvrtý experiment: Zber dát z testovacej podložky

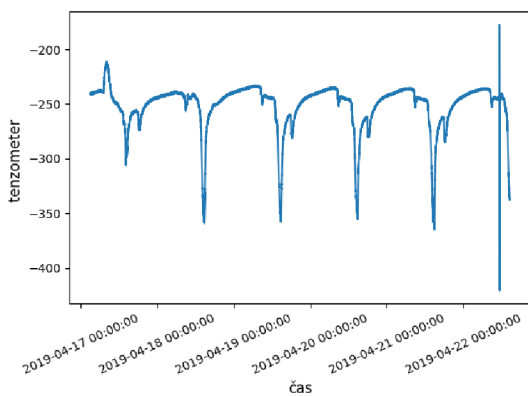
Počas tohoto experimentu boli senzory pripevnené na testovaciu podložku, ktorá ponúka možnosť vibrovania pomocou vibračných motorov a zahriatia pomocou skratu na výkonnom rezistore. Testovacia podložka striedala fázu chodu a nečinnosti. Predmetom skúmania sú zozbierané dáta zo všetkých sensorov. Na obrázku 6.9 je zobrazená testovacia podložka a k nej pripojené senzory. Na podložke sú 4 tlačítka, kde prvé slúži na zahriatie podložky a ďalšie tri slúžia na spínanie troch motorov. Zo spodnej strany podložky sú pripojené senzory monitorovacieho zariadenia. Podložka slúži na testovanie monitorovacích zariadení firmy. Jej výroba nie je súčasťou tejto práce, slúži však ako nástroj pre testovanie. V experimente sa počas jednej hodiny strieda chod podložky 1 minúta a jej nečinnosť 5 minút. Výsledky sú znázornené na obrázku 6.6, 6.7 a 6.8. Na grafoch je jasne vidieť rozdiel medzi fázami chodu a nečinnosti testovacej podložky. Na grafe teploty vždy pri spustení začala teplota narastať a vo fáze nečinnosti klesať, čo je vidieť na obrázku 6.7. Pri snímaní vibrácií po dobu jednej minúty vždy hodnoty narástli a v čase nečinnosti sa držali okolo nuly, čo je zachytené na obrázku 6.6. Tenzometrický modul pri spustení testovacej plošiny nezaznamenával vďaka RC filtru žiadnu zmenu, preto pri rozvibrovaní a zahrievaní testovacej podložky bol umelo vytváraný ťah a tlak na tenzometer, čo je vidieť na obrázku 6.8. Pre porovnanie nameraných hodnôt bol experiment zopakovaný. Na testovaciu podložku boli pripojené senzory z monitorovacieho zariadenia CompactRIO partnerskej firmy. Zozbierané dáta zo snímaču teploty sú zhodné, ale získané dáta z tenzometru a akcelerometru zachytávajú väčšie zmeny z dôvodu lepšieho upevnenia a lepšieho rozlíšenia samostatných sensorov.



Obr. 6.3: Tretí experiment: Hodnoty získané z akcelerometru.



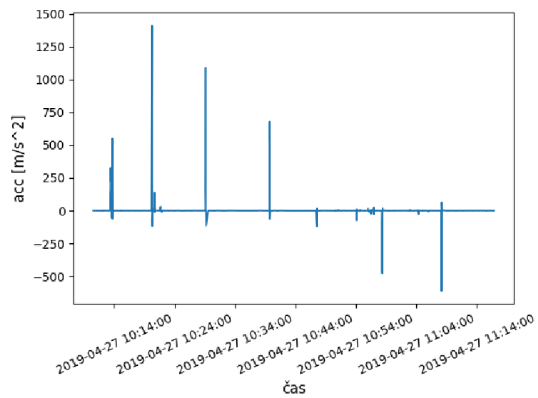
Obr. 6.4: Tretí experiment: Hodnoty získané zo snímača teploty.



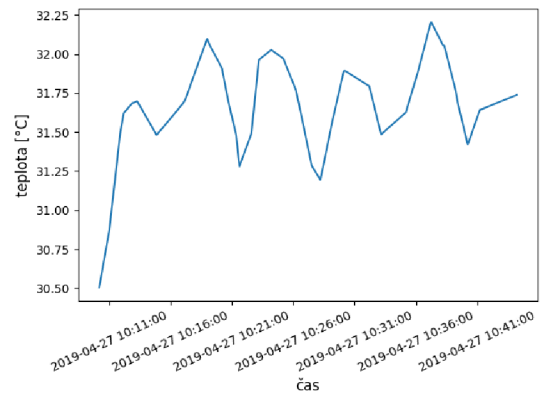
Obr. 6.5: Tretí experiment: Hodnoty získané z tenzometrického modulu.

## 6.5 Zhrnutie experimentov

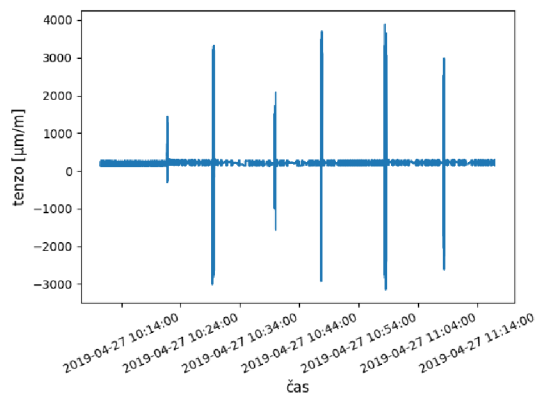
Po vykonaní všetkých experimentov je možné konštatovať, že monitorovacie zariadenie je schopné plynulého nepretržitého chodu a využitie jeho zdrojov je minimálne, čo spĺňa požiadavky k možnosti nasledujúceho rozšírenia. Nedostatky je možné vidieť v kvalite senzorov a ich zachytávaní zmien. Pre spoľahlivosť zariadenia je vhodné zvážiť využitie kvalitnejších senzorov, čo však zvyšuje výrobnú cenu. Pre účely tejto práce bol však kladený dôraz na čo najnižšie náklady. Ako veľký nedostatok sa ukázal tvar zvolených senzorov a ich napájania. V porovnaní so senzormi partnerskej spoločnosti ich tvar znemožňuje kvalitné osadenie na prístroj. Výsledky experimentov sú aj napriek tomu pozitívne. Pred nasadením monitorovacieho zariadenia do reálneho chodu továrne je potrebné globálne testovanie v priemyselných podmienkach.



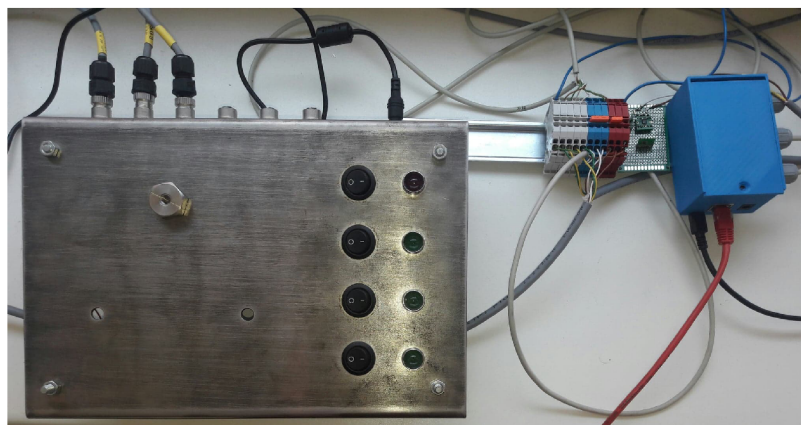
Obr. 6.6: Štvrtý experiment: Hodnoty získané z akcelerometru.



Obr. 6.7: Štvrtý experiment: Hodnoty získané zo snímača teploty.



Obr. 6.8: Štvrtý experiment: Hodnoty získané z tenzometrického modulu.



Obr. 6.9: Štvrtý experiment: Pripojenie monitorovacieho zariadenia na testovaciu podložku.

# Kapitola 7

## Záver

Cieľom práce je navrhnutie, vytvorenie a následné otestovanie monitorovacieho zariadenia pre zber veľkých dát v priemysle. Vytvorené riešenie pozostáva zo senzorov priamo pripojených na monitorovacie zariadenie. Toto zariadenie so senzormi komunikuje pomocou integrovaných zberníc a prevodníkov. V zariadení prebieha čítanie dát zo senzorov a algoritmus pre agregáciu a výber vhodných dát, ktoré sa odosielaajú na vzdialený server pomocou navrhnutého protokolu nad UDP. Zariadenie bolo vytvorené pod linuxovým operačným systémom Debian, čo vyplynulo z požiadavok partnerskej firmy. Na zariadení sú spustené aplikácie pre zber, agregáciu a odoslanie dát implementované v jazyku C. Na vzdialenom serveri beží aplikácia v roli konkurentného UDP serveru na príjem a uloženie dát do lokálnej databázy implementovaná v jazyku C++.

Prvá časť práce ozrejmuje čitateľovi praktiky zberu dát v priemysle, ich využitie, vysvetľuje časté pojmy preberanej problematiky a popisuje existujúce riešenia spoločností zaoberajúcich sa zberom dát v priemysle. V ďalšej časti sú popísané hardvérové komponenty, ktoré monitorovacie zariadenie obsahuje a sú popísané vývojové kity, medzi ktorými sa rozhodovalo pri výbere základu monitorovacieho zariadenia. Po výbere komponentov nasleduje návrh riešenia, v ktorom je zdôvodnený výber vývojového kitu, operačného systému, senzorov a softvérový návrh riešenia. V ďalšej kapitole je popísané zostavenie výsledného monitorovacieho zariadenia a implementácia jeho softvéru. Posledná kapitola sa zaoberá testovaním zariadenia, ktoré zahŕňa niekoľko experimentov a ich vyhodnotenie.

Základnou požiadavkou zariadenia bola jeho cena. Tabuľka 7.1 ukazuje náklady na vytvorenie monitorovacieho zariadenia. Celkovo vytvorenie monitorovacieho zariadenia bez prirátania nákladov na návrh, implementáciu a testovanie vyšlo pod 100 €. Táto suma sa ani z ďaleka nepribližuje cenám monitorovacích zariadení spomínaných spoločností, no je potreba podotknúť, že sa jedná o prototyp, ktorý je treba vylepšiť a rozšíriť, aby bol vhodný pre reálne využitie pri zbere veľkého množstva dát v priemysle. Pre prípadné vylepšenia by bolo vhodné rozšíriť BeagleBone o externý AD prevodník s väčším rozlíšením a analógovú elektroniku pre prispôsobenie signálu zo senzorov na jeho vstup. Všetky tieto zmeny znamenajú nárast ceny a preto v rámci tejto práce neboli využité.

Prínos tejto práce vidím v popísaní a prehľade zberu dát v priemysle, vytvorení návodu na prácu s metódami čítania dát zo senzorov pomocou jednodoskových počítačov a vytvorení dobrého základu pre kvalitné zariadenie pre zber veľkého množstva dát v priemysle.

<b>Komponent</b>	<b>Cena</b>
BeagleBone Black	50 €
HTU21D	8 €
ADXL345	5 €
Kabeláž	5 €
Rezistory, kondenzátory, svorky, DIN lišta, tenzometer	10 €
Plošný spoj	1 €
3D tlač krytu	4 €
AD8221	3 €
ADA4807	3 €
REF3125	2 €
<b>Spolu</b>	<b>91 €</b>

Tabuľka 7.1: Cena komponentov monitorovacieho zariadenia.

# Literatúra

- [1] Basics, C.: *BASICS OF THE I2C COMMUNICATION PROTOCOL*. [Online; navštívené 07.04.2019].  
URL <http://www.circuitbasics.com/basics-of-the-i2c-communication-protocol/>
- [2] Beagleboard: *BeagleBone Black*. [Online; navštívené 07.04.2019].  
URL <https://beagleboard.org/black>
- [3] David Burian, M. J.: *Trendy prediktivná diagnostiky nejen výrobních strojů*. [Online; navštívené 12.03.2019].  
URL <https://www.mmspektrum.com/clanek/trendy-prediktivna-diagnostiky-nejen-vyrobnich-stroju.html>
- [4] Devices, A.: Precision Instrumentation Amplifier-AD8221. *Online: http://www.analog.com/static/imported-files/data\_sheets/AD8221.pdf*, 2017.
- [5] Element14: *Raspberrz Pi 3 model b gpio pinblock pinout*. [Online; navštívené 27.03.2019].  
URL <https://www.element14.com/community/docs/DOC-73950/1/raspberry-pi-3-model-b-gpio-40-pin-block-pinout>
- [6] eLinux: *BeagleBone Operating Systems*. [Online; navštívené 22.03.2019].  
URL [https://elinux.org/BeagleBone\\_Operating\\_Systems](https://elinux.org/BeagleBone_Operating_Systems)
- [7] FOUNDATION, R. P.: *Raspberry Pi 3 Model B*. [Online; navštívené 07.04.2019].  
URL <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>
- [8] Fundation, E.: *CDT Downloads*. [Online; navštívené 07.04.2019].  
URL <https://www.eclipse.org/cdt/downloads.php>
- [9] Grusin, M.: *Serial Peripheral Interface (SPI)*. [Online; navštívené 07.04.2019].  
URL <https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi/all>
- [10] Hashemian, H. M.: State-of-the-art predictive maintenance techniques. *IEEE Transactions on Instrumentation and measurement*, ročník 60, č. 1, 2011: s. 226–236.
- [11] Instruments, N.: *CompactRIO Systems*. National Instruments, Marec 2019, [Online; navštívené 31.03.2019].  
URL <http://www.ni.com/cs-cz/shop/compactrio.html>
- [12] Jonathan Valdez, J. B.: *Understanding the I2C Bus*. [Online; navštívené 07.04.2019].  
URL <http://www.ti.com/lit/an/slva704/slva704.pdf>

- [13] Lee, I.; Lee, K.: The Internet of Things (IoT): Applications, investments, and challenges for enterprises. *Business Horizons*, ročník 58, č. 4, 2015: s. 431 – 440, ISSN 0007-6813, doi:<https://doi.org/10.1016/j.bushor.2015.03.008>.  
URL <http://www.sciencedirect.com/science/article/pii/S0007681315000373>
- [14] Lee, J.; Kao, H.-A.; Yang, S.: Service innovation and smart analytics for industry 4.0 and big data environment. *Procedia Cirp*, ročník 16, 2014: s. 3–8.
- [15] Manyika, J.; Chui, M.; Brown, B.; aj.: Big data: The next frontier for innovation, competition, and productivity. 2011.
- [16] Matoušek, P.: *Síťové aplikace a jejich architektura*. Akademické nakladatelství, VUTIUM, 2014, ISBN 9788021437661.
- [17] Robotics, R. P.: *Raspberry Pi I2C*. [Online; navštívené 07.04.2019].  
URL <http://tinyurl.com/y7cp6h9v>
- [18] Scheffer, C.; Girdhar, P.: *Practical machinery vibration analysis and predictive maintenance*. Elsevier, 2004.
- [19] Shop, A.: *IIC I2C Senzor Teploty a Vlhkosti HTU21D*. [Online; navštívené 07.04.2019].  
URL <https://arduino-shop.cz/arduino/1487-htu21d-i2c-senzor-teploty-a-vlhkosti.html>
- [20] Simens: *This is MindSphere!* Marec 2019, [Online; navštívené 21.03.2019].  
URL <https://new.siemens.com/global/en/products/software/mindsphere.html>
- [21] SKF: *SKF Multilog On-Line System IMx-8*. [Online; navštívené 20.03.2019].  
URL [https://www.skf.com/binary/tcm:12-289998/IMx-8%20User%20Manual\\_EN%2015V\\_090\\_00049\\_100\\_tcm\\_12-289998.pdf](https://www.skf.com/binary/tcm:12-289998/IMx-8%20User%20Manual_EN%2015V_090_00049_100_tcm_12-289998.pdf)
- [22] Sparkfun: *IIC SparkFun Triple Axis Accelerometer Breakout - ADXL345*. [Online; navštívené 07.04.2019].  
URL <https://www.sparkfun.com/products/9836>
- [23] Subhani, M.: *What is IoT architecture?* [Online; navštívené 05.02.2019].  
URL <https://www.quora.com/What-is-IoT-architecture>
- [24] telos: *I2C – What’s That?* [Online; navštívené 07.04.2019].  
URL <https://www.i2c-bus.org/>