



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ**
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ
DEPARTMENT OF TELECOMMUNICATIONS

IMPLEMENTACE KOMUNIKAČNÍCH PROTOKOLŮ PRO IOT S VYUŽITÍM ROZŠIŘUJÍCÍHO MODULU UNIPI PRO RASPBERRY PI

IMPLEMENTATION OF IOT COMMUNICATION PROTOCOLS UTILIZING UNIPI MODULE FOR RASPBERRY PI

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

PhDr. Jan Krejčí

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. Pavel Mašek

BRNO 2017



Diplomová práce

magisterský navazující studijní obor **Telekomunikační a informační technika**
Ústav telekomunikací

Student: PhDr. Jan Krejčí

ID: 187017

Ročník: 2

Akademický rok: 2016/17

NÁZEV TÉMATU:

Implementace komunikačních protokolů pro IoT s využitím rozšiřujícího modulu UniPi pro Raspberry Pi

POKyny pro vypracování:

Teoretická část diplomové práce bude zahrnovat seznámení s modulem UniPi pro embedded zařízení Raspberry Pi. Dále bude provedena analýza možností implementace komunikačních protokolů pro Internet věcí (IoT) s využitím UniPi. Na základně podporovaných protokolů bude navrhnut scénář s využitím reálných senzorů (měřicích zařízení), kdy rozšiřující modul UniPi bude figurovat v roli přijímače M2M (Machine-to-Machine) dat od senzorů. V praktické části bude provedena implementace komunikačního protokolu Wireless M-BUS s cílem umožnit příjem šifrovaných dat (šifrovací algoritmus AES) a jejich následnou vizualizaci.

DOPORUČENÁ LITERATURA:

- [1] BOSWARTHICK, David, Omar ELLOUMI a Olivier HERSENT. 2012. M2M communications: a systems approach. Hoboken, N.J.: Wiley, xxiii, 308 p. ISBN 978-1-119-99475-6.
- [2] MONK, Simon. 2013. Programming the Raspberry Pi: getting started with Python. New York: McGraw-Hill. ISBN 00-718-0783-7.

Termín zadání: 1.2.2017

Termín odevzdání: 24.5.2017

Vedoucí práce: Ing. Pavel Mašek

doc. Ing. Jiří Mišurec, CSc.
předseda oborové rady

Konzultant:

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Předkládaná diplomová práce je zaměřena na implementaci protokolu Wireless M-Bus do embedded zařízení RaspberryPi za pomocí rozšiřující desky UniPi. Protokol je implementován v jazyce Python a s Wireless M-Bus zařízeními komunikuje pomocí komunikačního modulu IQRF připojeného na sběrnici UART. Teoretická část práce se zaměřuje na přehled embedded zařízení pro IoT, možnosti jejich rozšíření, popisuje danou rozšiřující desku i Wireless M-Bus komunikační modul. Podrobněji se zaměřuje na vrstvy protokolu Wireless M-Bus, čímž poskytuje základy potřebné pro porozumění praktické části. Teoretickou část uzavírá přehled vyčítaných zařízení včetně popisu jejich datových jednotek. V praktické části je provedena implementace aplikace pro vyčítání dat z Wireless M-Bus senzorů a jejich následnou vizualizaci. Aplikace je schopna vyčítat i zařízení umožňující šifrovaný přenos.

KLÍČOVÁ SLOVA

Bonega, EN 13757-4, Google Charts, IIoT, IQRF TR-27D-WMB, Kamstrup, Neuron, Python, RaspberryPi, UniPi, Weptech, Wireless M-Bus, ZPA

ABSTRACT

Presented diploma thesis is focused on the implementation of Wireless M-Bus protocol to embedded device RaspberryPi with expansion board UniPi. The protocol is implemented in Python with Wireless M-Bus devices communicating via IQRF transceiver connected to the UART bus. The theoretical part is focused on an overview of embedded devices for the IoT, the possibility of their expansion. Further, the UniPi expansion board and Wireless M-Bus transceiver are detailed. First part of the thesis focuses on the Wireless M-bus layers, which provides a basic knowledge for understanding the practical part. The theoretical part concludes overview of captured devices including a description of their data units. In the practical part is the implementation of the sample application for receiving data from a Wireless M-Bus sensors. The application is able to read data from devices transmitting encrypted communication.

KEYWORDS

Bonega, EN 13757-4, Google Charts, IIoT, IQRF TR-27D-WMB, Kamstrup, Neuron, Python, RaspberryPi, UniPi, Weptech, Wireless M-Bus, ZPA

KREJČÍ, Jan *Implementace komunikačních protokolů pro IoT s využitím rozšiřujícího modulu UniPi pro RaspberryPi*: diplomová práce. Brno: Vysoké učené technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2017. 109 s. Vedoucí práce byl Ing. Pavel Mašek

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Implementace komunikačních protokolů pro IoT s využitím rozšiřujícího modulu UniPi pro RaspberryPi“ jsem vypracoval(a) samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor(ka) uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil(a) autorská práva třetích osob, zejména jsem nezasáhl(a) nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom(a) následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno
.....
podpis autora(-ky)

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Pavlu Maškovi za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno
.....
podpis autora(-ky)



Faculty of Electrical Engineering
and Communication
Brno University of Technology
Purkynova 118, CZ-61200 Brno
Czech Republic
<http://www.six.feec.vutbr.cz>

PODĚKOVÁNÍ

Výzkum popsaný v této diplomové práci byl realizován v laboratořích podpořených z projektu SIX; registrační číslo CZ.1.05/2.1.00/03.0072, operační program Výzkum a vývoj pro inovace.

Brno

.....
podpis autora(-ky)

OBSAH

| | |
|--|-----------|
| Úvod | 15 |
| 1 Internet věcí | 16 |
| 1.1 Spotřebitelský Internet věcí | 16 |
| 1.2 Průmyslový Internet věcí | 16 |
| 1.2.1 Průmysl 4.0 | 17 |
| 2 Embedded zařízení pro IoT | 18 |
| 2.1 Arduino | 18 |
| 2.1.1 Arduino Duemilanove | 19 |
| 2.1.2 Arduino Uno | 19 |
| 2.1.3 Arduino Leonardo | 19 |
| 2.1.4 Arduino Mega | 20 |
| 2.1.5 Arduino Due | 20 |
| 2.1.6 Arduino Mini | 20 |
| 2.1.7 Arduino Micro | 20 |
| 2.1.8 Arduino Nano | 21 |
| 2.1.9 Arduino Fio | 21 |
| 2.1.10 Arduino MKR1000 | 21 |
| 2.1.11 Lilypad Arduino | 21 |
| 2.1.12 Arduino Yun | 22 |
| 2.2 Arduino klony | 22 |
| 2.2.1 Freeduino | 22 |
| 2.2.2 LABduino | 22 |
| 2.2.3 Arduelo Libero | 22 |
| 2.2.4 Bare Bones Board | 23 |
| 2.2.5 Freaduino | 23 |
| 2.2.6 Runtime | 23 |
| 2.2.7 Nanode | 23 |
| 2.2.8 Seeeduino | 23 |
| 2.2.9 Teensy | 24 |
| 2.2.10 Diavolino | 24 |
| 2.2.11 Boarduino | 24 |
| 2.3 RaspberryPi | 24 |
| 2.3.1 RaspberryPi | 25 |
| 2.3.2 RaspberryPi 2 | 26 |
| 2.3.3 RaspberryPi 3 | 26 |

| | | |
|----------|---|-----------|
| 2.3.4 | RaspberryPi Zero | 26 |
| 2.4 | RaspberryPi klony | 26 |
| 2.4.1 | BananaPi | 27 |
| 2.4.2 | OrangePi | 27 |
| 2.4.3 | CubieBoard | 28 |
| 2.4.4 | UpBoard | 29 |
| 2.4.5 | PINE64 | 29 |
| 2.4.6 | HardKernel Odroid | 30 |
| 2.4.7 | BeagleBoard | 31 |
| 2.5 | Intel | 32 |
| 2.5.1 | Intel Galileo | 32 |
| 2.5.2 | Intel Edison | 32 |
| 2.6 | AMD Gizmo | 33 |
| 3 | Rozšiřující deska UniPi | 35 |
| 3.1 | UniPi v1 | 36 |
| 3.2 | UniPi v2 - Neuron | 37 |
| 3.3 | Srovnání obou verzí | 41 |
| 3.4 | Sběrnice na UniPi | 42 |
| 3.4.1 | UART | 42 |
| 3.4.2 | SPI | 43 |
| 3.4.3 | RS-485 | 43 |
| 3.4.4 | I2C | 43 |
| 3.4.5 | 1Wire | 44 |
| 3.4.6 | GPIO | 44 |
| 3.5 | Software pro UniPi | 44 |
| 4 | Komunikační modul Wireless M-Bus | 46 |
| 4.1 | Obecný popis modulu TR-72D-WMB | 46 |
| 4.2 | Komunikační módy | 47 |
| 4.3 | Komunikační protokol | 48 |
| 5 | Wireless M-Bus protokol | 50 |
| 5.1 | Princip komunikace | 50 |
| 5.2 | Režimy přenosu | 51 |
| 5.3 | Struktura zasílaných dat | 53 |
| 5.4 | Popis jednotlivých vrstev | 54 |
| 5.4.1 | Fyzická vrstva Wireless M-Bus | 54 |
| 5.4.2 | Linková vrstva Wireless M-Bus | 55 |
| 5.4.3 | Aplikační vrstva Wireless M-Bus | 59 |

| | | |
|-------------------|---|-----------|
| 5.5 | Šifrování dat | 64 |
| 5.5.1 | Šifrovací algoritmus DES | 64 |
| 5.5.2 | Šifrovací algoritmus AES | 64 |
| 5.5.3 | Inicializační vektor | 65 |
| 5.5.4 | Šifrovací klíč | 66 |
| 5.5.5 | Určení šifrovaných dat | 67 |
| 5.5.6 | Princip dešifrování | 68 |
| 5.5.7 | Kontrola rozšifrování dat | 68 |
| 6 | Wireless M-Bus zařízení | 69 |
| 6.1 | Weptech OMST-868A | 69 |
| 6.2 | Bonega | 72 |
| 6.3 | Kamstrup | 74 |
| 6.4 | ZPA | 77 |
| 7 | Návrh implementace | 80 |
| 7.1 | Výběr OS | 80 |
| 7.2 | Výběr programovacího jazyka | 81 |
| 7.3 | Nastavení komunikačního modulu a čidla | 81 |
| 7.4 | Zajištění dedikovaného běhu | 81 |
| 7.5 | Zajištění podpory šifrování | 82 |
| 7.6 | Zpracování dat | 82 |
| 7.6.1 | Nešifrovaný přenos | 82 |
| 7.6.2 | Šifrovaný přenos | 82 |
| 7.7 | Zajištění uložení dat | 84 |
| 7.8 | Zajištění vizualizace dat | 85 |
| 7.9 | Struktura aplikace | 85 |
| 7.9.1 | Start programu v rámci operačního systému | 85 |
| 7.9.2 | Start programu z pohledu aplikace | 87 |
| 7.9.3 | Základní kontrola a cyklus příjmu dat | 87 |
| 7.9.4 | Dešifrování dat | 87 |
| 7.9.5 | Parsování dat | 88 |
| 7.9.6 | Uložení dat | 89 |
| 7.9.7 | Ošetření výjimek | 89 |
| 7.10 | Export dat | 90 |
| 7.11 | Vizualizace dat | 90 |
| 8 | Závěr | 93 |
| Literatura | | 94 |

| | |
|---|------------|
| Seznam symbolů, veličin a zkratek | 99 |
| Seznam příloh | 103 |
| A Přehled parametrů jednotlivých jednodeskových počítačů | 104 |
| B Ukázka zachycených dat | 105 |
| C Ukázka vizualizace dat | 106 |
| D Obsah přiloženého DVD | 109 |

SEZNAM OBRÁZKŮ

| | | |
|------|---|----|
| 1.1 | Schéma odvětví Průmyslu 4.0 | 17 |
| 2.1 | Arduino Duemilanove, Uno a Leonardo | 19 |
| 2.2 | Arduino Mega a Due | 20 |
| 2.3 | Arduino Mini, Micro a Nano | 21 |
| 2.4 | Arduino Fio a MKR1000 | 21 |
| 2.5 | Lilypad Arduino a Arduino Yun | 22 |
| 2.6 | Bare Bones Board a Freaduino | 23 |
| 2.7 | Diavolino a Boarduino | 24 |
| 2.8 | RaspberryPi prvních verzí | 26 |
| 2.9 | RaspberryPi následujících verzí | 26 |
| 2.10 | BananaPi BPI-M2 a OrangePi Plus2 | 28 |
| 2.11 | CubieBoard1 a UpBoard1 | 29 |
| 2.12 | PINE A64+ 2GB a HardKernel Odroid-C2 | 30 |
| 2.13 | BeagleBone Black [34] | 31 |
| 2.14 | Intel Galileo [35] | 32 |
| 2.15 | Arduino board pro Intel Edison | 33 |
| 2.16 | Intel breakout board | 33 |
| 2.17 | AMD Gizmo | 34 |
| 3.1 | UniPi v1 [42] | 36 |
| 3.2 | Blokové schéma UniPi v1 | 37 |
| 3.3 | UniPi rozšiřující deska | 38 |
| 3.4 | Unipi Neuron [43] | 39 |
| 3.5 | Detaily UNiPi desky | 40 |
| 3.6 | Detaily vnitřního uspořádání UniPi Neuronu S103 | 40 |
| 3.7 | Blokové schéma UniPi v2 | 41 |
| 3.8 | UART rámec | 42 |
| 3.9 | Zpětná kompatibilita GPIO konektoru | 45 |
| 4.1 | Modul IQRF TR-72DA-WMB [46] | 46 |
| 4.2 | Blokové schéma modulu TR-72D-WMB [46] | 47 |
| 4.3 | Přehled typu modulu dle antény [46] | 47 |
| 4.4 | Různé módy dle použité topologie [46] | 48 |
| 5.1 | Princip kódování Manchester | 55 |
| 5.2 | Princip algoritmu AES v módu CBC | 65 |
| 5.3 | Obecné schéma dešifrování AES-128 CBC | 68 |
| 6.1 | Čidlo Weptech OMST-868A [58] | 69 |
| 6.2 | Sada Bonega [59] | 72 |
| 6.3 | Kamstrup Multical 402 [60] | 74 |

| | | |
|-----|---|-----|
| 6.4 | ZPA ZE.310 [61] | 77 |
| 7.1 | Model zvolené SqLite 3 databáze | 84 |
| 7.2 | Vývojový diagram aplikace pro vyčítání dat | 86 |
| 7.3 | Snímek obrazovky vizualizační aplikace | 92 |
| C.1 | Vizualizace měření elektroměrem ZPA (interval 24 hodin) | 106 |
| C.2 | Vizualizace měření vodoměry Bonega (interval 24 hodin) | 107 |
| C.3 | Vizualizace měření čidlem Weptech (interval 24 hodin) | 108 |

SEZNAM TABULEK

| | | |
|------|--|----|
| 1.1 | Porovnání průmyslového a spotřebitelského IoT [4, 5] | 16 |
| 3.1 | Porovnání modelů UniPi NEURON dle I/O [43] | 39 |
| 3.2 | Varianty modelů UniPi NEURON dle CPU a RAM [43] | 39 |
| 5.1 | Popis standardu EN-13757 [48] | 50 |
| 5.2 | Režimy přenosu WM-Bus protokolu [48] | 52 |
| 5.3 | Formát datové jednotky [51] | 53 |
| 5.4 | Formát datové jednotky protokolu Wireless M-Bus [51] | 54 |
| 5.5 | Zkrácený formát datové jednotky [51] | 54 |
| 5.6 | Formát datové jednotky po přijetí modulem IQRF | 54 |
| 5.7 | Tabulka kódování 3 ze 6 [54] | 56 |
| 5.8 | Identifikace typu zařízení | 57 |
| 5.9 | Kódování CI-Pole | 58 |
| 5.10 | Hodnoty Status pole | 59 |
| 5.11 | Struktura dat aplikáční vrstvy | 60 |
| 5.12 | Kódování DIF Pole | 60 |
| 5.13 | Kódování rozšiřujícího bitu DIF pole | 60 |
| 5.14 | Kódování funkčního pole DIF pole | 61 |
| 5.15 | Kódování Data pole DIF pole | 61 |
| 5.16 | Kódování DIFE Pole | 61 |
| 5.17 | Kódování VIF Pole | 61 |
| 5.18 | Kódování rozšiřujícího bitu VIF pole | 62 |
| 5.19 | Kódování Data pole VIF pole | 63 |
| 5.20 | Kódování VIFE Pole | 63 |
| 5.21 | Kódování data ve formátu G | 64 |
| 5.22 | Kódování data a času ve formátu F | 64 |
| 5.23 | Formát inicializačního vektoru | 66 |
| 5.24 | Význam bitů pole ConfigurationWord | 67 |
| 6.1 | Telegram ze zařízení Weptech 868A [58] | 70 |
| 6.2 | Konfigurace intervalu zasílání pomocí DIP přepínače [58] | 71 |
| 6.3 | Přehled nastavení čidla [58] | 72 |
| 6.4 | Telegram z modulu Bonega [59] | 73 |
| 6.5 | Telegram ze zařízení Kamstrup Multical 402 [60] | 76 |
| 6.6 | Telegram ze zařízení ZPA ZE.302 [61] | 78 |
| 7.1 | Rozklíčovaný zachycený paket | 83 |

SEZNAM UKÁZEK ZDROJOVÝCH KÓDŮ

| | | |
|-----|--|----|
| 4.1 | Komunikace s modulem přes sériový port | 49 |
| 5.1 | Implementace vyčítání uložení MSB | 53 |
| 5.2 | Implementace F a G formátu | 62 |
| 5.3 | Sestavení inicializačního vektoru | 65 |
| 5.4 | Implementace AES | 68 |
| 5.5 | Ověření kontrolní sekvence AES | 68 |
| 7.1 | Implementace AES dešifrování | 88 |
| 7.2 | Ukázka parsování dat | 88 |
| 7.3 | Ukázka ukládání dat | 89 |
| 7.4 | Ukázka exportu dat | 90 |
| 7.5 | Ukázka vizualizace dat | 90 |
| 7.6 | Ukázka parametrizace vizualizovaných dat | 91 |

ÚVOD

Fenoménem dneška je propojování Internetu věcí (IoT - Internet of Things), služeb (IoS - Internet of Services) a lidí (IoP - Internet of People) a s ním související vývoj komunikací stroj-stroj (M2M - Machine to Machine), člověk-stroj (H2M - Human to Machine) nebo člověk-člověk (H2H - Human to Human). Internet věcí, služeb a lidí se rozšiřuje závratným tempem a proniká tak do odvětví, ve kterých se rostoucím tempem využívají komunikační nízkovýkonové (embedded) zařízení a roste potřeba rozšíření těchto zařízení o nové komunikační protokoly a technologie. Vzniknou sítě založené na propojených zařízeních, které budou schopny samostatné výměny informací, vyvolání potřebných akcí v reakci na momentální podmínky a vzájemné nezávislé kontroly. Senzory, přístroje a IT systémy budou vzájemně propojeny a budou na sebe pomocí standardních komunikačních protokolů vzájemně reagovat a analyzovat data, aby mohly předvídat případné chyby či poruchy, konfigurovat samy sebe a v reálném čase se přizpůsobovat změněným podmínkám [1, 2].

Tato práce vychází z požadavku na implementaci Wireless M-Bus protokolu do produktu UniPi NEURON. K tomuto účelu bylo zvoleno nízkovýkonové (embedded) zařízení RaspberryPi a jeho rozšiřující modul UniPi. Pro M2M komunikaci byl zvolen protokol Wireless M-Bus, jelikož je jedním z nejrozšířenějších a navíc je založen na protokolu M-Bus, který je osvědčený a velmi rozšířený (měření a regulace topných systémů, plynu, odběru vody a elektrické energie). V teoretické části práce jsou popsány jednotlivé rodiny jednodeskových počítačů a jejich vlastnosti, popis rozšiřujících desek UniPi a samotného komunikačního modulu pro Wireless M-Bus a popis komunikačního protokolu Wireless M-Bus. Teoretickou část uzavírá přehled vyčítaných měřících zařízení protokolu Wireless M-Bus.

Praktická část se zaměřuje na implementaci Wireless M-Bus protokolu v zařízení RaspberryPi pomocí rozšiřujícího modulu UniPi a komunikačního modulu Wireless M-Bus. Tato implementace vyčítání dat ze vzdálených zařízení je realizována v jazyku Python a následně jsou získaná data vizualizována pomocí Google Chart API [3].

1 INTERNET VĚCÍ

Cílem Internetu věcí (IoT - Internet of Things) je propojení zařízení, systémů a služeb za účelem poskytnutí více dat, která mohou být převedena na informace a informace potom na znalosti, které mohou být následně aplikovány. Princip IoT je tedy sběr dat, ty jsou následně uložena a analyzována a poté dojde ke sdílení výsledků. V rámci IoT se vytvořily dva hlavní směry, průmyslový Internet věcí (iIoT - Industry IoT) a spotřebitelský Internet věcí (cIoT - Customer IoT) [4, 5]. Rozdíly obou směrů jsou shrnutы v Tab. 1.1.

1.1 Spotřebitelský Internet věcí

Spotřebitelský Internet věcí se zaměřuje na spotřebitelská zařízení, IT a telekomunikační zařízení a další. Jsou zde využívána zařízení zjednodušující každodenní život pomocí automatizace v domácnosti, chytrých zařízení nebo pomocí nositelné elektroniky. Hlavní výhodou je zvýšení uživatelského zážitku (QoE - Quality of Experience).

1.2 Průmyslový Internet věcí

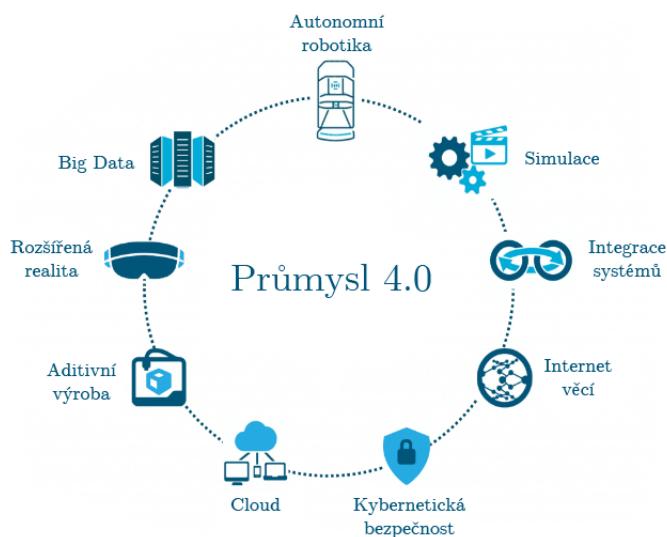
Průmyslový Internet věcí vychází z M2M (Machine-to-Machine) a rozšiřuje komunikaci o možnost uložení, analýzy a zobrazení dat. Jedná se o IoT zařízení a systémy, které jsou používány v průmyslových odvětvích, jako jsou průmyslová automatizace, energetický průmysl a zdravotnictví. Hlavním zaměřením je efektivnější využívání zdrojů, snížení provozních nákladů, zvýšení efektivity či bezpečnosti. V praxi může sloužit například pro zajištění bezpečnosti pracovníků či automatizaci údržby.

Tab. 1.1: Porovnání průmyslového a spotřebitelského IoT [4, 5]

| | Spotřebitelský IoT | Průmyslový IoT |
|------------|--|--|
| Zaměření | Spotřebitel. | Průmysl. |
| Zařízení | Chytré zařízení a nositelná elektronika. | Stroje, zařízení a průmyslová automatizace. |
| Důležitost | Nejedná se o životně důležité systémy. | Jedná se o životně důležité systémy. |
| Využití | Zvýšení uživatelského zážitku. | Lepší využívání zdrojů, snížení provozních nákladů, zvýšení efektivity či bezpečnosti. |

1.2.1 Průmysl 4.0

Současný trend digitalizace a s ní související automatizace výroby je označován jako Průmysl 4.0. Koncept vychází z dokumentu, který byl představen na veletrhu v Hannoveru v roce 2013. Předpokládá se, že v horizontu následujících 10 až 15 let nastane příchod čtvrté průmyslové revoluce, která přinese radikální změnu ve srovnání s nynějším výrobním procesem. Podle této myšlenky vzniknou chytré továrny, které budou využívat kyberneticko-fyzikální systémy. Ty převezmou opakující se a jednoduché činnosti, které do té doby vykonávali lidé. Má zahrnovat kompletní (viz Obr. 1.1) digitalizaci, robotizaci a automatizaci většiny současných lidských činností pro zajištění větší rychlosti a efektivity výroby přesnějších, osobitějších, spolehlivějších a levnějších produktů, současně pro efektivnější využití materiálů a ekologičtější průmysl i lidský život.



Obr. 1.1: Schéma odvětví Průmyslu 4.0

Na průmyslové úrovni má jít o nahrazení manuální lidské práce robotizací, současné manuální zadávání výrobních dat a postupů má být nahrazeno automatickým elektronickým předáváním informací mezi jednotlivými výrobními komponentami a materiály. Významné změny mají i ve spojitosti s automatizovaným průmyslem nastat v oblasti domácností a běžného bydlení, kde mají být jednotlivé domácí systémy vzájemně elektronicky propojeny a jejich vzájemná koordinovaná spolupráce bude maximalizovat efektivitu a současně minimalizovat spotřebu médií.

V reflexi na tento trend v září 2015 vydalo Ministerstvo průmyslu a obchodu Národní iniciativu Průmysl 4.0 [1], podle které bude revoluce příležitostí pro růst a konkurenceschopnost českých firem a České republiky vůbec.

2 EMBEDDED ZAŘÍZENÍ PRO IOT

V současnosti existuje velké množství zařízení v roli výpočetní jednotky, využitelných pro chytrou domácnost či Internet věcí. Tato kapitola představí nejznámější z nich, popíše jejich možnosti, uvede možnosti připojení senzorů a zmíní jejich nedostatky.

Mezi nejznámější nízkovýkonové (embedded) zařízení patří open-source Arduino (Kap. 2.1), RaspberryPi (Kap. 2.3) a jejich klony (Kap. 2.2 a 2.4). Poté budou zmíněny desky předních firem výrobců procesorů Intel (Kap. 2.5) a AMD (Kap. 2.6) a v neposlední řadě budou představeny desky firmy CubieBoard (Kap. 2.4.3), HardKernel (Kap. 2.4.6) a další.

Z výše uvedených byly vybrány jen nízkovýkonová zařízení, využitelná pro chytrou domácnost či Internet věcí, s vyvedenými GPIO piny a dostatečnou dokumentací.

Jejich vlastnosti jsou přehledně shrnuty v příloze A.

2.1 Arduino

Ardiuno je skupina několika jednodeskových počítačů založených na mikrokontrolerech. Nejedná se však o klasický stolní počítač IBM PC, ale o prototypovací desku, ke které se spíše jak ovládací zobrazovací periferie připojují senzory, moduly, serva a displeje. Projekt je od svého počátku šířen jako open-source, příručka jazyka a externí knihovny jsou pak šířeny pod licencí Creative Commons.

Výrobce těchto desek vytvořil vývojové prostředí shodné pro všechny produkty Ardiuno. To se nazývá Arduino IDE, je dostupné zdarma na webu výrobce a podporuje jazyk Wiring [6], což je upravená verze jazyka C. Prostředí zároveň obsahuje i Serial Monitor, který slouží k oboustranné sériové komunikaci mezi Arduinem a PC. Alternativou ještě může být prostředí Processing [7] využívající stejnojmenný jazyk, umožňující vytváření grafických multiplatformních aplikací.

Na deskách bývá několik diod, resetovací tlačítko, různé přídavné sběrnice (UART, I2C), konektory pro ICSP programování, napájecí konektor, oscilátor a obvod zprostředkovávající komunikaci po USB.

Arduino podporuje připojení rozšiřujících karet. Ty se u Arduina nazývají Shieldy, mají převážně stejný tvar jako deska Arduina a připojují se pomocí dlouhých pinů. Zabírají celou plochu, ale většina z nich dále zpřístupňuje GPIO (General Purpose Input/Output) piny, lze je tedy skládat na sebe. Stejně jako Arduino desek existuje i celá řada shieldů. Samozřejmě lze k Arduinu připojit i samotné moduly nebo senzory, přímým připojením na dané piny. Je však třeba dbát na to, že Arduino pracuje s 5 V logikou, zatímco například RaspberryPi pracuje s 3,3 V logikou.

2.1.1 Arduino Duemilanove

Arduino Duemilanove je vývojová jednoprocesorová deska s mikroprocesorem AT-Mega168 od firmy Atmel, tedy platformě Atmel AVR. Parametry zařízení jsou: ATmega168 s 16 MHz krystalem, 16 KB flash, 1 KB SRAM (Static Random Access Memory), 512 B EEPROM (Electrically Erasable Programmable Read-Only Memory). Konektivita: 14 digitálních vstupně/výstupních pinů, z toho 6 z nich může být využito i PWM (Pulse Width Modulation), vstupních analogových pinů (10 bit A/D převodník, 0-5 V), I2C (Inter-Integrated Circuit) sběrnici, UART (Universal Asynchronous Receiver/Transmitter) sběrnici, ICSP (In Circuit Serial Programming) rozhraní, USB (Universal Serial Bus) rozhraní [8].

2.1.2 Arduino Uno

Arduino Uno je v současné době asi nejčastěji používaný typ desky. Arduino Uno je vývojová jednoprocesorová deska s mikroprocesorem ATMega328. Od roku 2011 je nástupcem Arduina Duemilanove. Změny oproti předchůdci jsou pouze v použitém mikrokontroléru, došlo k zdvojnásobení velikosti paměti na 32 KB flash, 2 KB SRAM, 1 KB EEPROM [9].

2.1.3 Arduino Leonardo

Arduino Leonardo designově navazuje na Arduino Uno, liší se pouze v použitém čipu ATmega32u4 [10].



(a) Arduino Duemilanove

(b) Arduino Uno

(c) Arduino Leonardo

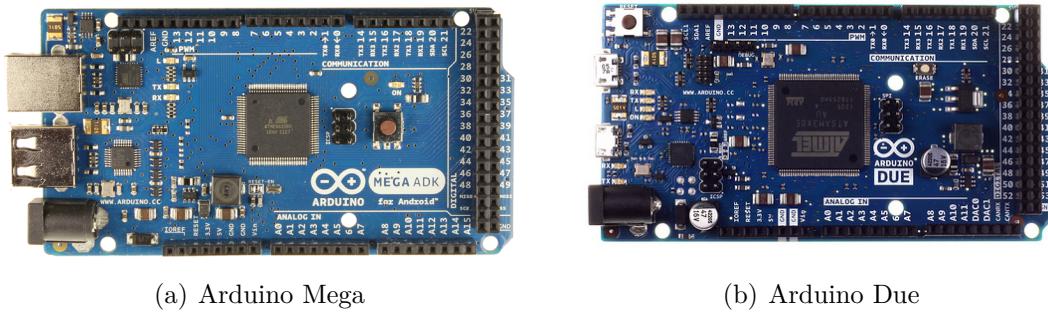
Obr. 2.1: Arduino Duemilanove, Uno a Leonardo

2.1.4 Arduino Mega

Arduino Mega je deska pro náročnější projekty. Oproti klasickému Arduinu má Arduino Mega rychlejší procesor (16 MHz) a také více vstupních a výstupních pinů. K dispozici je 54 digitálních pinů, 14 PWM výstupů, 16 analogových vstupů a 4 hardwareové sériové porty. Dále má 256 kB flash paměti, 8 kB RAM paměti a 4 kB EEPROM paměti [11].

2.1.5 Arduino Due

Arduino Due je nástupcem Arduino Mega a je to první karta Arduino, na níž je umístěn 32-bitový řadič (32-bitový ARM procesor Atmel SAM3X8E). Vysoká taktovací rychlosť 84 MHz ve spojení s celkem 54 I/O piny umožňuje realizaci značně rozsáhlých projektů. K 54 pinům mimo jiné patří 12 PWM výstupů a 12 analogových vstupů, 4 UARTy, 2 I2C a dvojitý digitálně-analogový měnič. Vlastní USB Host poskytuje kartě vedle standardů jako JTAG (Joint Test Action Group), SPI (Serial Peripheral Interface) a Micro USB širší možnosti konektivity [12].



(a) Arduino Mega

(b) Arduino Due

Obr. 2.2: Arduino Mega a Due

2.1.6 Arduino Mini

Arduino Mini je asi nejmenší oficiální verze Arduina, navržená pro úsporu místa. Daní za malé rozměry je však absence USB portu. K programování je tedy nutné použít externí USB2Serial převodník. Jeho výkon však nijak nezaostává za většími deskami. Běží na procesoru ATmega328 s taktem 16 MHz. Pro své malé rozměry je vhodný k použití například v chytrých vypínačích, či dálkových ovladačích [13].

2.1.7 Arduino Micro

Arduino Micro je jedna z desek, která má čip obsahující předprogramovaný převodník ATmega32u4 [14].

2.1.8 Arduino Nano

Arduino Nano navíc obsahuje ještě USB port a převodník [15].



(a) Arduino Mini

(b) Arduino Micro

(c) Arduino Nano

Obr. 2.3: Arduino Mini, Micro a Nano

2.1.9 Arduino Fio

Arduino Fio je přizpůsobená k připojení různých bezdrátových modulů (například ZigBee nebo XBee moduly). Základem je procesor ATmega328P, který běží na frekvenci 8 MHz. Napětí je zde kvůli kompatibilitě s moduly sníženo oproti většině ostatních desek z 5 V na 3,3 V [16].

2.1.10 Arduino MKR1000

Arduino MKR1000 je postavené na čipu ATSAMW25 od Atmelu, který v sobě spojuje ARMové jádro SAMD21 Cortex-M, Wi-Fi čip a šifrovací a autentizační čip ECC508. Tento čip nabízí ECDH (Diffie-Hellman s využitím eliptických křivek) a ECDSA (Elliptic Curve Digital Signature Algorithm). Dále pak generátor náhodných čísel, unikátní 72bitové sériové číslo nebo SHA-256 s volitelným HMAC.



(a) Arduino Fio

(b) Arduino MKR1000

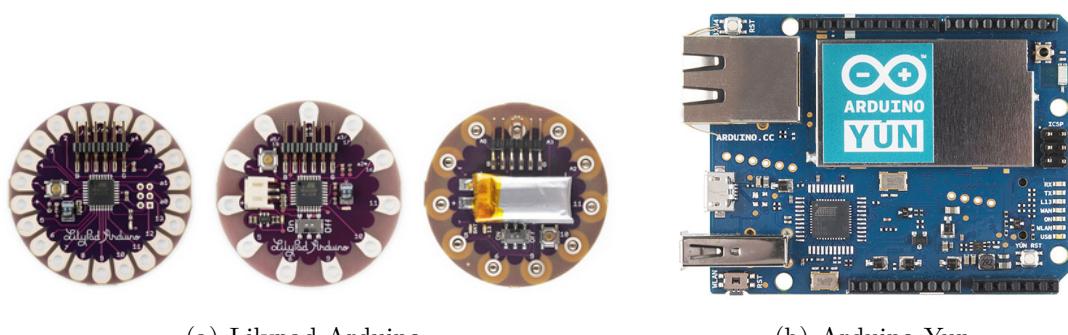
Obr. 2.4: Arduino Fio a MKR1000

2.1.11 Lilypad Arduino

Lilypad Arduino je postaveno na ATmega168V (energeticky úsporná verze ATmega168) nebo ATmega328V. Je určeno pro wearables projekty, zejména pro implementaci do textilií, kdy jsou spoje tvořeny vodivou nití. Není však pratelná. Existuje více variant této desky [17].

2.1.12 Arduino Yun

Arduino Yun je deska založená na ATmega32u4 a Atheros AR9331, který je schopný běhu odlehčeného linuxu Linino. Ve výbavě je softwarový bridge (prostředník, most), který zajišťuje komunikaci mezi oběma čipy. Procesor Atheros podporuje linuxové distribuce založené na OpenWrt s názvem OpenWrt-Yun. Deska má vestavěný Ethernet a WiFi modul, USB-A port, slot pro MicroSD kartu. Dále disponuje 20 digitálními I/O piny, z toho 7 mohou být použito jako výstupy PWM a 12 jako analogové vstupy [19].



Obr. 2.5: Lilypad Arduino a Arduino Yun

2.2 Arduino klony

Jelikož je projekt Arduino open-source, vzniklo množství klonů od dalších firem i jednotlivců. Klony jsou s původním Arduinem kompatibilní, ve většině případů konfigurací odpovídají některému z Arduino modelu, většinou Arduino UNO. Kity, které nemají shodné rozložení pinů neumožňují připojení Arduino shieldů. V této podkapitole je uveden krátký přehled těch nejznámějších. Rozsáhlý přehled kompatibilních klonů lze nalézt na oficiálních stránkách Arduina [20].

2.2.1 Freeduino

Freeduino je klon Arduina, vycházející z Arduino Duemilanove.

2.2.2 LABduino

LABduino je český klon Arduina vytvořený z otevřené elektronické stavebnice MLAB.

2.2.3 Arduelo Libero

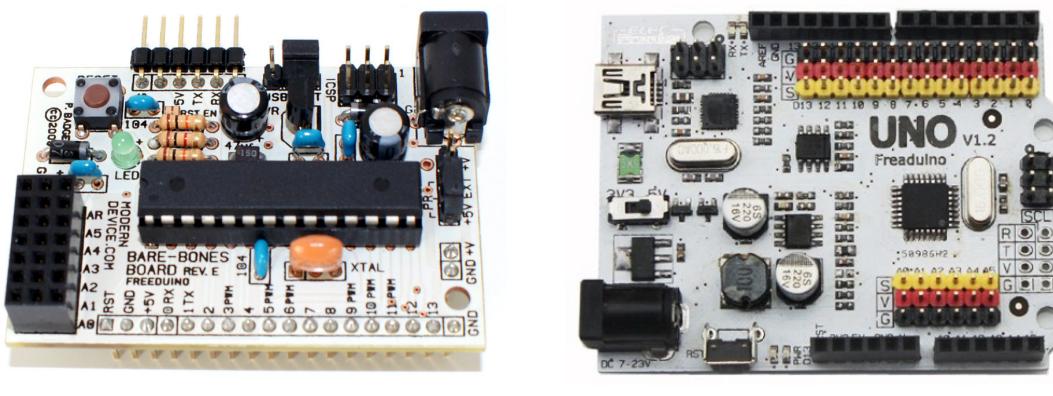
Arduelo Libero je mírně vylepšený český free klon Arduino Diecimila.

2.2.4 Bare Bones Board

Bare Bones Board je kompatibilní deska, tvarově nepřipomínající žádný Arduino produkt. Kvůli rozložení pinů nepodporující shieldy. Vyráběná a prodávaná jako kit firmou Modern Device Company.

2.2.5 Freaduino

Freaduino je kompatibilní deska, tvarově shodná s Arduino UNO, vyráběná a prodávaná firmou ElecFreak jako kit The Freaduino Uno. Podporuje 3,3 V logiku a napájení. Má piny na připojení modulů (XBee). Napájecí piny zvládají zátěž až 2 A.



(a) Bare Bones Board

(b) Freaduino

Obr. 2.6: Bare Bones Board a Freaduino

2.2.6 Runtime

Runtime je kompatibilní deska, tvarově nepřipomínající žádný Arduino produkt. Kvůli rozložení pinů nepodporující shieldy. Vyráběná a prodávaná jako kit firmou NKC Electronics.

2.2.7 Nanode

Nanode je kompatibilní deska, tvarově nepřipomínající žádný Arduino produkt. Tvarově připomíná Arduino UNO, rozložení pinů je kompatibilní.

2.2.8 Seeeduino

Seeeduino je kompatibilní deska, vzhledem připomínající Arduino UNO, parametry shodná s Arduino Mega.

2.2.9 Teensy

Teensy je kompletní vývojový mikrokontrolérový systém na velmi malé desce bez osazených pinů, který je schopen realizovat mnoho typů projektů. Softwarově je kompatibilní s Arduinem, programuje se však pomocí doplňku do Arduino IDE nebo pomocí WinAVR [21].

2.2.10 Diavolino

Diavolino je free klon Arduina, vzhledově i parametricky podobný Arduino UNO, bez vyvedených konektorů. Vyráběná a prodávaná jako kit firmou Evil Mad Scientist.

2.2.11 Boarduino

Boarduino je levnější klon Arduina Diecimila s piny pro zapojení rovnou do nepájivého pole.



(a) Diavolino

(b) Boarduino

Obr. 2.7: Diavolino a Boarduino

2.3 RaspberryPi

RaspberryPi reprezentuje jednodeskový počítač o velikosti zhruba platební karty. Byl vyvinut v roce 2012 s cílem podpořit výuku informatiky a seznámit studenty s řízením různých zařízení přes počítač [22].

Primárním operačním systémem je Linux, k dispozici je několik jeho distribucí, případně lze použít Windows 10 IoT Core. Na rozdíl do Arduina obsahuje RaspberryPi plnohodnotný operační systém, ARM mikrokontrolér, USB pro připojení myši a klávesnice, Ethernet konektor pro připojení sítě, grafický výstup HDMI (High-Definition Multimedia Interface) a kompozitní video, DSI (Display Serial Interface) pro připojení displeje, CSI (Camera Serial Interface) pro připojení kamery

a čtečku paměťových karet, tedy působí spíše jako menší počítač, než vývojová platforma.

Všechny další rozšiřující sběrnice (UART, I2C, SPI, PWM, digitální vstup a výstup, analogový vstup) jsou vyvedeny do 26-40 pinového GPIO konektoru. Na rozdíl od Arduina je možné RaspberryPi pomocí GPIO kontaktů použít nejen k ovládání různých zařízení, ale i k samotnému vývoji příslušných aplikací. Lze ho také použít jako multimediální přehrávač videa nebo hudby nebo i jen pro přístup k Internetu.

RaspberryPi stejně jako Arduino podporuje připojení rozšiřujících karet:

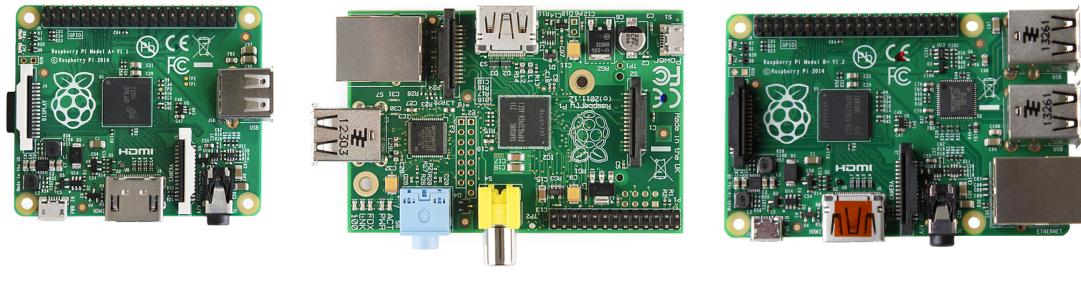
- **Pi T-Cobbler** je pasivní elektronický přípravek, který se k RaspberryPi připojuje pomocí 40 žilového plochého kabelu a slouží k vyvedení pinů do vývojové desky breadboard. Zde na konektorové desce jsou již jednotlivé piny popsány.
- **Gertboard** je rozšiřující deska autora Gerta Van Loo, který rozšiřuje I/O možnosti RaspberryPi. K ní se připojuje pomocí 40 žilového plochého kabelu a rozšiřuje možnosti o 8/10/12-bitový dvoukanálový D/A převodník, 10-bitový dvoukanálový A/D převodník, obvody pro řízení motoru, předprogramovaný Atmel AVR ATmega 328P, 6 výstupů s otevřeným kolektorem a dalších 12 IO pinů [23].
- **UniPi** je rozšiřující deska která rozšiřuje I/O možnosti RaspberryPi. K ní se připojuje pomocí 26 žilového plochého kabelu a dle typu připojeného UniPi zařízení poskytuje I/O funkce navíc. Rozšiřujícími moduly UniPi se bude blíže zabývat následující kapitola (Kap. 3).
- **RaspberryPi to Arduino Shield** je rozšiřující deska, která umožňuje propojení RaspberryPi a vybraných modelů Arduino.

Samozřejmě lze k Arduinu připojit i samotné moduly nebo senzory, přímým připojením na dané piny GPIO konektoru. Je však třeba dbát na to, že RaspberryPi pracuje s 3,3 V logikou, zatímco například Arduino pracuje s 5 V logikou. Popis GPIO konektoru včetně možností připojení je součástí následující kapitoly (Kap. 3.4.6).

2.3.1 RaspberryPi

Původní model RaspberryPi [24] byl zveřejněn v únoru roku 2012. Obsahuje jednojádrový procesor o frekvenci 700 MHz. U této verze existovaly tři modely:

- **Model A+** je odlehčená levná verze modelu B. Nemá žádný paměťový slot. Disponuje 256 MB RAM. Neobsahuje USB port. Má 40 GPIO pinů.
- **Model B** byl původní RaspberryPi. Má slot na SD kartu. Disponuje 512 MB RAM. Obsahuje 1 USB port. Má 26 GPIO pinů. Má samostatný výstup kompozitního videa.
- **Model B+** obsahuje slot na MicroSD kartu. Disponuje 512 MB RAM. Obsahuje 2 USB porty. Má 40 GPIO pinů.



(a) Model A+

(b) Model B

(c) Model B+

Obr. 2.8: RaspberryPi prvních verzí

2.3.2 RaspberryPi 2

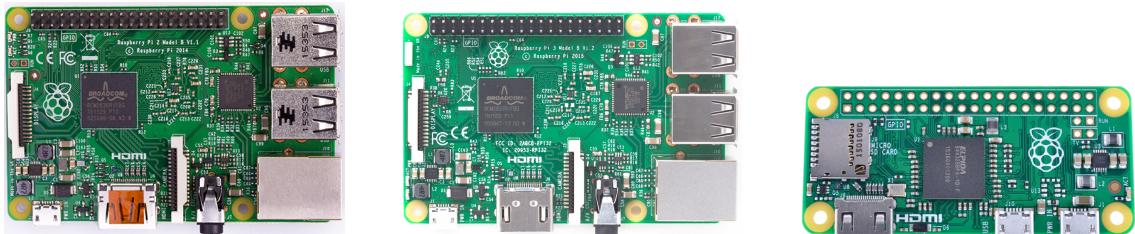
RaspberryPi 2 je pokračováním RaspberryPi, které přináší zejména vyšší výkon. Díky čtyřjádrovému procesoru BCM2836 o taktu 900 MHz by měl být 3–6× rychlejší než jeho předchůdce. Tento model disponuje 1 GB paměti a má 4 USB porty [25].

2.3.3 RaspberryPi 3

RaspberryPi 3, dostupný od roku 2016 je vybaven čtyřjádrovým 64bitovým procesorem ARM Cortex-A53 o taktu 1,2 GHz. Oproti předchozímu modelu přináší integraci WiFi a Bluetooth modulů přímo na desce a měl by být 2x rychlejší [26].

2.3.4 RaspberryPi Zero

RaspberryPi Zero je nejúspornější varianta RaspberryPi, ideální pro použití v IoT. Vychází z modelu A+, ve srovnání s ním nabízí procesor s frekvencí 1 GHz a 512 MB paměti. Má přibližně poloviční velikost, nemá však vyvedené piny GPIO konektoru, USB konektory má ve verzi micro a HDMI ve verzi mini [27].



(a) RaspberryPi 2

(b) RaspberryPi 3

(c) RaspberryPi Zero

Obr. 2.9: RaspberryPi následujících verzí

2.4 RaspberryPi klony

Vzrůstající popularita RaspberryPi dala stejně jak u Arduina vzniknout celé řadě klonů. Tyto klony odvozují ze základního sestavení RaspberryPi a určitým způsobem

ho rozříšují. Jelikož označení „RaspberryPi“ je registrovanou ochrannou známkou, proto mají podobně navžené počítače odvozené názvy, jako BananaPi a OrangePi. Zmíněné klony patří k nejznámějším a každý z nich již existuje v několika verzích, v této podkapitole budou představeny ty nejznámější s uvedením jejich hlavních odchylek od RaspberryPi.

2.4.1 BananaPi

Původní BananaPi, ze kterého vychází řada dalších modelů, je malý jednodeskový počítač, který se na první pohled podobá RaspberryPi. Obsahuje dvoujádrový procesor a 512 MB RAM. Na rozdíl od RaspberryPi obsahuje BananaPi také SATA řadič, mikrofon, který je připájen přímo na desce, gigabitový Ethernet, USB 2.0 OTG, IR přijímač, tlačítko reset a power. Počítač podporuje SATA disky až do velikosti 2 TB. GPIO konektor je vždy kompatibilní s některou verzi RaspberryPi. Za výrobou všech počítačů BananaPi stojí čínská firma SinoVoip CO., Limited [28].

BananaPi BPI-M2 je klon RaspberryPi 2, obsahuje taktéž čtyřjádrový procesor běžící na 1 GHz, má již integrovanou WiFi (Wireless Fidelity), ale neobsahuje SATA (Serial Advanced Technology Attachment) port.

BananaPi BPI-M3 obsahuje osmijádrový procesor 1,8 GHz s 2 GB RAM, dále zahrnuje Wifi b/g/n a integrované Bluetooth 4. Obsahuje SATA port.

BananaPi BPI-M64 obsahuje čtyřjádrový 64 bitový SoC procesor Allwinner A64.

2.4.2 OrangePi

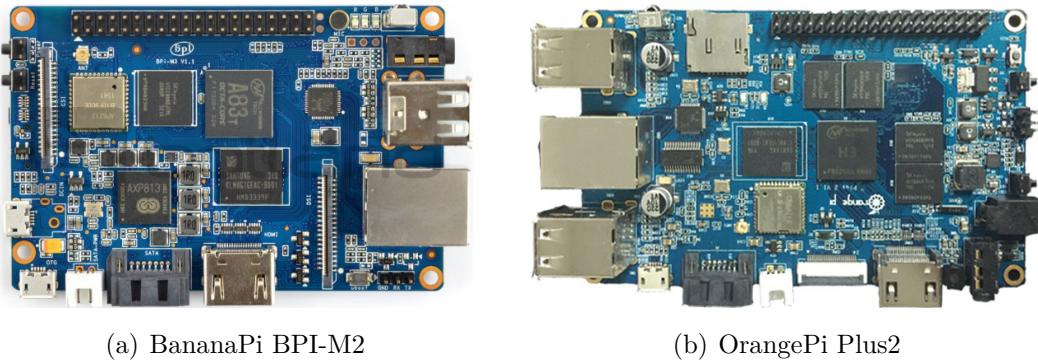
OrangePi je alternativa pro RaspberryPi vznikající v posledních dvou letech. Všechny modely jsou založeny na architektuře ARM Cortex-A7 s SoC Allwinner H3 s čtyřjádrovým CPU, výjimkou jsou OrangePi a OrangePi Mini, které mají SoC Allwinner A20 s dvojádrovým CPU. Grafickým čipem je u všech modelů ARM Mali-400 MP2. Všechny modely podporují HDMI CEC [29].

OrangePi je základní model z rodiny OrangePi, obsahuje čtyřjádrový procesor Allwinner A20 na 1 GHz a 1 GB RAM. Oproti RaspberryPi má navíc pouze mikrofon, IR (Infrared Radiation) port, USB OTG, ale nemá DSI rozhraní.

OrangePi Plus má procesory běžící na 1,6 GHz, 1 GB RAM a 8 GB EMMC Flash. Oproti RaspberryPi má gigabitový Ethernet, integrovaný mikrofon, USB-OTG konektor, integrovaný WiFi modul, IR přijímač. Obsahuje SATA port, který je připojený přes USB převodník.

OrangePi Plus2 oproti předchozí verzi došlo k navýšení pamětí na 2 GB RAM a 16 GB EMMC Flash a doplnění CSI konektoru.

OrangePi One vznikla jako reakce na odlehčenou verzi RaspberryPi Zero. Jedná se o čtyřjádrový procesor na frekvenci 1,2 GHz postavený na čipu ARM Cortex-A7 s grafickým čipem Mali400 MP2. Operační paměť je 512 MB. K dispozici je pouze 10/100 Mbps Ethernet a jeden port USB 2.0.



Obr. 2.10: BananaPi BPI-M2 a OrangePi Plus2

2.4.3 CubieBoard

CubieBoard je alternativou k RaspberryPi z roku 2012. Ačkoliv jsou vzhledově i parametricky velmi podobné, není Cubieboard s RaspberryPi kompatibilní. Jsou postaveny na AllWinner A10 SoC čipu. Výrobce poskytuje vlastní sadu modulů a rozšiřujících desek. Cubieboardy poskytují pinové rozhraní, obsahující základní sběrnice (I2C, SPI, UART) ale i rozšiřující jako LVDS. Desky obsahují navíc SATA konektor [30].

CubieBoard1 je výkonná nízkonákladová deska s ARM A8 o taktu 1 GHz s 1 GB RAM, 4 GB NAND flash a Mali400 GPU. Obsahuje LAN port a dvojici USB portů. Deska má 96 pinů, které zahrnují sběrnice GPIO, I2C, UART, LVDS (Low Resolution Analog to Digital Converter), PWM, SPI, CSI, VGA a jiné. Dále obsahuje 100Mbps Ethernet a dva USB HOST porty, mini USB OTG, čtečku micro SD, HDMI, IR, line in, line out a SATA port.

CubieBoard2 představuje nástupce CubieBoardu1, je s ním zpětně kompatibilní a od předchozí verze se liší pouze dvoujádrovým provedením CPU a GPU.

CubieBoard3 oproti předchozím verzím přináší vylepšení jako 2 GB RAM, 8 GB NAND flash, VGA konektor přímo na desce, gigabitový Ethernet, WiFi a Bluetooth integrované přímo na desce. Pinové rozhranní je zde redukováno na 54 pinů obsahující I2S (Inter-Integrated Sound), I2C, SPI, CVBS (Color Video Blanc Sync), UART, PWM a GPIO.

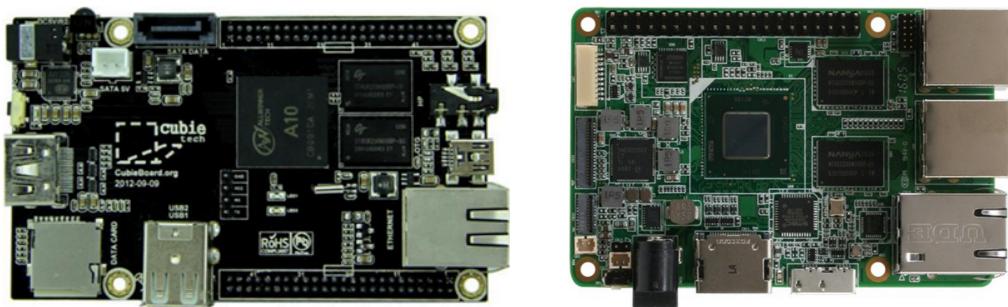
CubieBoard4 je nástupce CubieBoardu 3, je zpětně kompatibilní a oproti předchůdci přináší čtyřjádrový CPU ARM A15x a GPU PowerVR G6230. Dále má microUSB 3.0 OTG a audio konektory umístěné přímo na desce.

CubieBoard5 nabízí osmijádrový procesor Allwinner H8, který doplňují 2 GB RAM. Navíc oproti předchozím verzím má kromě HDMI i DP (Display Port), přináší také konektor pro připojení externí baterie. Došlo k navýšení GPIO pinů na 70, které navíc přináší LRADC (Low Resolution Analog to Digital Converter) a PS2 (Personal System/2). SATA konektor pomocí speciální desky podporuje připojení dvou SATA disků s podporou RAIDu.

CubieBoardy již poskytují dostatečný výkon pro embeeded zařízení, přináší oproti RaspberryPi mnoho rozšiřujících sběrnic, avšak pro nedostatečnou podporu či zařazení v evropských zemí a velmi častou nedostupnost webové výrobce, včetně dostupnosti anglické dokumentace pro programování jednotlivých rozhraní, není moc vhodná pro IoT. Hodí se spíše pro aplikace jako multimediální centrum či nízkokládový počítač.

2.4.4 UpBoard

UpBoard představuje miniaturní jednodeskový počítač na platformě Intel s čtyřjádrovým procesorem Intel Atom. Vzhledově je velice podobný RaspberryPi 3. Tento počítač obsahuje čtyřjádrový procesor Intel Atom x5-Z8300 na frekvenci 1,84 GHz s TDP 2 W. Obsahuje 1 GB RAM a 16 GB flash eMMC (Embedded MultiMedia Card). 40 pinové rozhraní je totožné jako u RaspberryPi2 s níž je částečně kompatibilní. Navíc obsahuje gigabitový Ethernet port, 5 USB 2.0 a jedno USB 3.0. Čip má hardwarovou podporu šifrování AES (Advanced Encryption Standard), je tedy vhodný pro IoT projekty s vyšším zabezpečením. Podporuje Android 5.0, Linux či Windows 10. Dokumentace pro programování GPIO v současnosti neexistuje, dokumentaci tvoří pouze popis GPIO konektoru [31].



(a) CubieBoard1

(b) UpBoard1

Obr. 2.11: CubieBoard1 a UpBoard1

2.4.5 PINE64

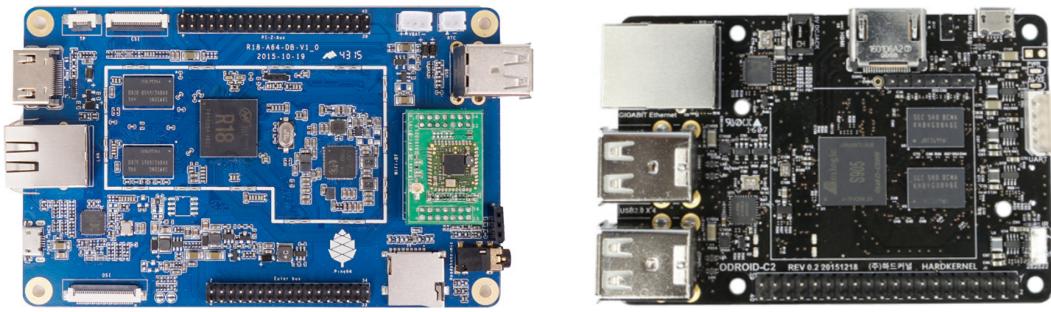
Pine64 je rodina tří jednodeskových počítačů společnosti PINE64. Tyto počítače byly navrženy tak, aby konkurovaly RaspberryPi ve výkonu a ceně. Všechny verze

obsahuje 64bitový čtyřjádrový procesor 1,152 GHz Cortex-A53 a liší se pouze velikostí operační paměti a použitelným operačním systémem. Oproti RaspberryPi obsahuje gigabitový Ethernet, WiFi, Bluetooth a port pro připojení dotykového panelu. Mají GPIO konektor shodný s danou verzí Raspberry, jsou s ní tedy do jisté míry kompatibilní. Zvláštností těchto desek je Eulerova sběrnice, která navýšuje počty sběrnic SPI, UART, GPIO [32].

PINE A64 512MB má 512 MB paměti a podporuje pouze Arch Linux a Debian Linux.

PINE A64+ 1GB má 1 GB paměti a podporuje i Android, Remix OS, Ubuntu a Windows IoT.

PINE A64+ 2GB má oproti předchozí verzi 2 GB operační paměti.



(a) PINE A64+ 2GB

(b) HardKernel Odroid-C2

Obr. 2.12: PINE A64+ 2GB a HardKernel Odroid-C2

2.4.6 HardKernel Odroid

ODROID je řada jednodeskových počítačů od společnosti Hardkernel. Název je odvozen z **Open Android**, ale podporovány jsou i linuxové distribuce. Desky disponují 40 pinovým GPIO kompatibilním s RaspberryPi, ale open-source již nejsou. Desky jsou postaveny na SoC platformě Samsung Exynos. Zvláštností desek je sériové rozhraní s 1,8 V [33].

ODROID-C1 je reakcí na RaspberryPi 1. Nabízí čtyřjádrový SoC Cortex A5 s frekvencí 1,5 GHz a 1 GB RAM. Dále má gigabitový Ethernet a připojení flash úložiště typu eMMC.

ODROID-C2 je reakcí na RaspberryPi 3. Obsahuje čtyřjádrový 64bitový procesor ARMv8 taktovaný na 2 GHz, 2 GB paměti a gigabitový Ethernet. Má podporu sběrnice I2S. Hlavní změnou je podpora HDMI 2.0 a schopnost přehrávat 4K video ve formátu H.265. Podporuje Ubuntu 16.04 nebo Android 5.1.

ODROID-XU4 je výkonejší řada desek, obsahuje čtyřjádrový procesor Samsung Exynos5 ARM Cortex-A15 na frekvenci 2 GHz a čtyřjádrový procesor Cortex-A7

Quad 1,3 GHz, bohužel vzhledem k výkonu je zde již potřeba aktivní chlazení. Deska disponuje grafickým čipem Mali-T628 MP6 a 2 GB RAM paměti.

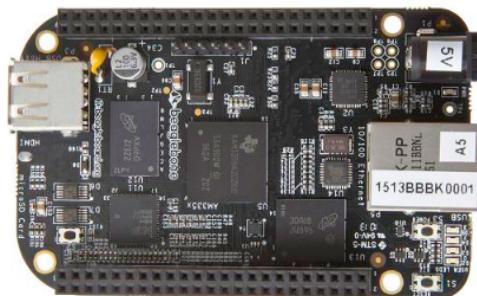
2.4.7 BeagleBoard

BeagleBoard je skupina jednodeskových počítačů produkovaných společností Texas Instruments, navržených na čipu Texas Instrument's OMAP3530 SoC, ten obsahuje ARM Cortex-A8 CPU, který může provozovat Linuxové distribuce, BSD nebo Android. Desky obsahují dva 46pinové GPIO konektory, oproti ostatním přináší podporu CAN (Controller Area Network) sběrnice. Výrobce poskytuje vlastní řadu kompatibilních rozšiřujících desek, nazýva je „capes“ a současně lze připojit až 4 takovéto desky. Výhodou desek je jejich nízká spotřeba, využívají maxminálně 2 W elektrické energie a mohou být napájeny i ze samostatného napájení. Vzhledem k nízké spotřebě energie, nejsou nutné žádné přídavné chladiče [34].

BeagleBoard obsahuje procesor Sitara ARM Cortex-A8 na frekvenci 720 MHz a disponuje dle revize 128 nebo 256 MB RAM. Obsahuje 256 MB NAND paměti.

BeagleBone obsahuje procesor Sitara ARM Cortex-A8 na frekvenci 720 MHz a disponuje 256 MB RAM.

BeagleBoard-X15 je založen na procesoru Sitara AM5728 s dvěma jádry ARM Cortex-A15 na frekvenci 1,5 GHz a dvěma jádry ARM Cortex-M4 na frekvenci 212 MHz and dvěma jádry TI C66x DSP na frekvenci 700 MHz. Disponuje 2 GB RAM. Použitý procesor přináší podporu HDMI 2.1, gigabit Ethernetu a grafického dvoujádrového čipu SGX544 na frekvenci 532 MHz.



Obr. 2.13: BeagleBone Black [34]

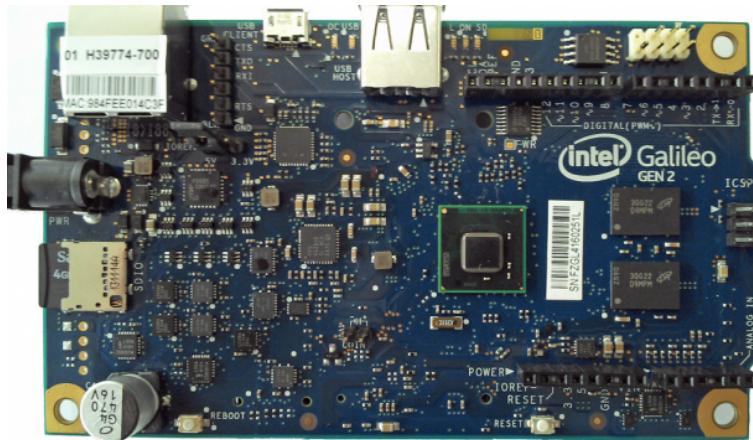
BeagleBone Black má oproti předchůdci zvýšenou paměť na 512 MB, frekvenci procesoru na 1 GHz, bylo přidáno HDMI a 2 GB eMMC flash paměti.

2.5 Intel

Společnosti Intel přináší dva jednodeskové počítače založené na platformě mikroprocesoru x86. Jsou navrženy jak pro vývojáře tak k výuce výpočetní techniky.

2.5.1 Intel Galileo

Intel Galileo je jednočipový počítač, vyvinutý společností Intel, postavený na architektuře x86. Obsahuje procesor Intel Quark x86 na frekvenci 400 MHz. Má 256 MB RAM. Byl navržen pro výuku výpočetní techniky. Jedná se o první zařízení od Intelu, které je hardwarem i softwarově kompatibilní s Arduinem. Lze k němu připojovat Arduino shieldy i moduly a využívat vývojové prostředí Arduina, včetně jeho knihoven. Tento počítač obsahuje 14 digitálních I/O pinů, z toho 6 z nich lze využít jako PWM výstupy. Dále obsahuje 6 analogových vstupů, UART sběrnici, I2C sběrnici, SPI sběrnici, Ethernet konektor, slot na MicroSD kartu. Dále obsahuje 2 USB konektory, jeden USB-host, druhý USB-klient. Druhá generace této desky pak přináší podporu PoE (Power over Ethernet) a další drobné změny [35, 36].



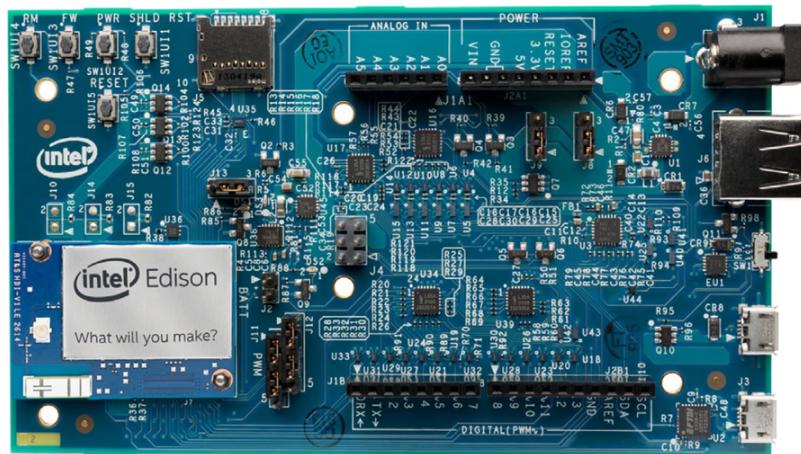
Obr. 2.14: Intel Galileo [35]

2.5.2 Intel Edison

Intel Edison je druhý jednočipový počítač architektury x86 vyvinutý společností Intel. Má velikost SD karty a je určený pro nositelnou elektroniku. Obsahuje dvoujádrový procesor Intel Quark x86 na frekvenci 400 MHz. Dále obsahuje 1 GB RAM a 4 GB flash paměti. Konektivita je zajištěna pomocí 70 pinového Hirose DF40 konektoru, který v sobě sdružuje veškerá dostupná rozhraní (USB, GPIO, SPI, I2C a PWM). Jsou k dispozici dvě rozšiřující desky [37]:

- Arduino board - Arduino board je plně kompatibilní s Arduinem, včetně podpory Arduino shieldů a modulů. Dále tato deska zpřístupňuje 20 digitálních

I/O pinů, z toho 4 z nich lze využít jako PWM výstupy. Dále obsahuje 6 analogových vstupů, UART sběrnici, I2C sběrnici, SPI sběrnici. Dále obsahuje 2 USB konektory, jeden pro napájení, druhý připojený k UART sběrnici a slot na SD kartu.



Obr. 2.15: Arduino board pro Intel Edison

- Intel breakout board - Tato deska je díky svým malým rozměrům vhodná pro prototypování nositelné elektroniky či pro Internet věcí. Obsahuje pájitevnou mřížku pro zpřístupnění všeckých dostupných rozhraní. Na desku jsou vyvedeny pouze dva USB konektory, jeden pro napájení a druhý připojený k UART sběrnici.



Obr. 2.16: Intel breakout board

Desky Intel se hodí spíše pro větší typy projektů, kdy již vývojové prostředí Arduina nestačí a je potřeba využít plného potencálu operačního systému.

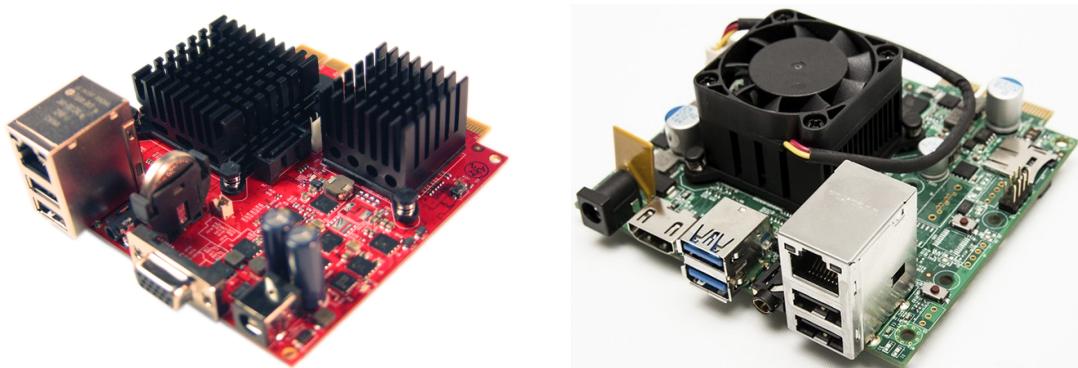
2.6 AMD Gizmo

Gizmo Board a Gizmo Board 2 od firmy AMD jsou alternativou k počítačům RaspberryPi, nabízející však platformu IBM PC a plně 64bitovou architekturu. Umožňuje tedy běh klasických operačních systémů, včetně Windows.

Gizmo Board 1 beží na dvoujádrovém APU G-T40E od firmy AMD na frekvenci 1GHz při příkonu 10 W. Součástí procesoru je grafický čip Radeon HD 6250.

K dispozici je 1 GB RAM. Deska dále obsahuje dvojici USB, VGA, audio výstup, SATA a Ethernet konektor. Další sběrnice jako GPIO, SPI, I2C, UART a PWM jsou dostupné po připojení rozšiřující karty přes LowSpeed [38].

Gizmo Board 2 je vybaven APU AMD GX210HA na frekvenci 1 GHz, s integrovaným GPU AMD Radeon HD 8210E s frekvencí 300 MHz. Příkon je 9 W. Tento model má také 1 GB RAM. Tato verze disponuje 4 USB, HDMI výstupem, MicroSD slotem a Ethernetovým portem. Mezi další rozhraní patří PCI Express (Peripheral Component Interconnect Express), GPIO, SPI, I2C, UART, DAC/ADC nebo PWM [39].



(a) Gizmo Board 1

(b) Gizmo Board 2

Obr. 2.17: AMD Gizmo

Oba počítače již poskytují dostatečný výkon pro embeedeed zařízení, avšak druhá verze zařízení již využívá aktivní chlazení a je hlučnější. Obě zařízení jsou větších rozměrů a nemají dostatečnou dokumentaci k přístupu a programování jednotlivých rozhraní. Hodí se spíše pro aplikace jako multimediální centrum či jednodušší počítač, než pro IoT nebo průmyslovou automatizaci. Komunita okolo AMD Gizmo prakticky neexistuje.

3 ROZŠIŘUJÍCÍ DESKA UNIPI

UniPi je česká firma, nyní dceřiná společnost Faster.cz, původně její oddělení měření a regulace, které se zaměřuje na inteligentní stavební řešení, domácí automatizaci a Internet věcí. Dále provozuje výzkum a vývoj rozšiřujících desek UniPi, včetně jejich softwarového vybavení [40].

UniPi je taktéž pojmenování pro přídavné rozšiřující desky pro RaspberryPi, se kterou je plně kompatibilní ve všech verzích. Je vybavena řadou komponent, jako jsou například digitální galvanicky oddělené vstupy s LED signalizací, 0 - 10 V analogové vstupy, 0 - 10 V analogové výstupy, spínací relé, jednokanálová 1Wire sběrnice, I2C sběrnice, UART sběrnice, SPI sběrnice a RS-485 sběrnice.

UniPi je název, odvozený od slov „RaspberryPi“ a „univerzální“, protože jednoduchost a univerzálnost jsou základní charakteristiky této desky. Deska původně vznikla pro potřeby řízení energetických hodnot vlastního datacentra Zelená Data [41], ale škála odvětví, kde je možné UniPi nasadit je rozsáhlá, pro představu výrobce uvádí několik příkladů [40]:

- Docházkové a přístupové systémy.
- Bezpečnostní systémy.
- Topné, chladící prvky i řízení.
- Větrání, rekuperace.
- Řídící systémy, které nejsou kompletní.
- Dohledové systémy.
- Ovládání světelných prvků.
- Datové vypínače.
- Řízení pivovarnických technologií.
- Zavlažovací systémy.
- Wellness systémy – vířivé vany, bazény, sauny.
- Solární systémy.

V současnosti existují dvě verze rozšiřující desky UniPi:

- UniPi (verze 1)¹.
- UniPi Neuron (verze 2)².

Desky se liší svými vstupně-výstupními možnostmi, rozměry a jsou dostupné v několika variantách.

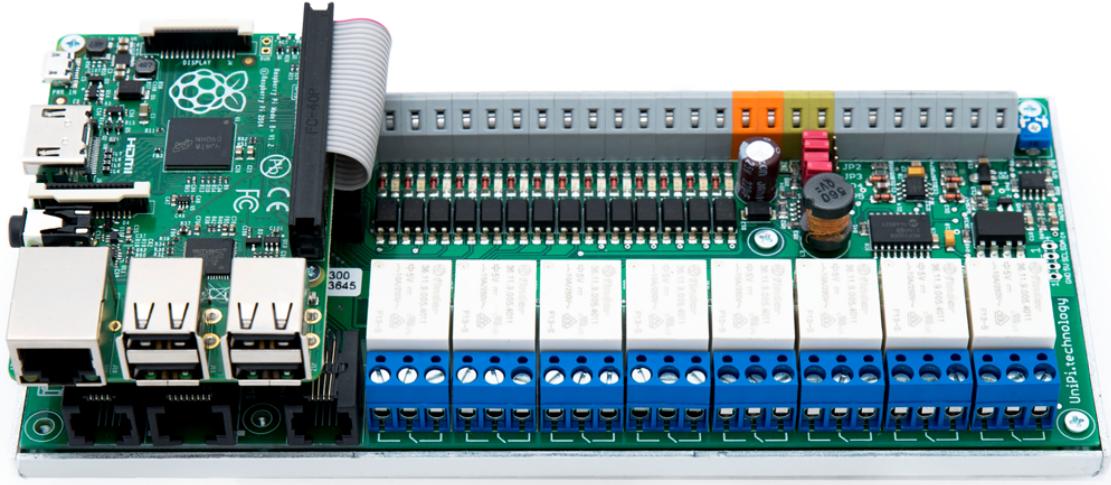
¹Dostupné z: <http://unipi.technology/product/unipi/>

²Dostupné z: <http://unipi.technology/product/unipi-neuron-s103/>

3.1 UniPi v1

Deska UniPi je prezentována jako nejlevnější a nejjednodušší řešení pro inteligentní budovy a IoT. Je navržena pro maximální kompatibilitu s embedded zařízením RaspberryPi. Zařízení bylo vyvinuto primárně jako rozhraní pro příjem vstupních signálů, jejich vyhodnocení a realizaci výstupní reakce na základě naprogramovaných algoritmů [42].

Disponuje (viz Obr. 3.1) osmi relé pro střídavý proud, čtrnácti digitálními vstupy, jedním jednokanálovým 1Wire rozhraním, dvěma 0 - 10 V analogovými vstupy a jedním 0 - 10 V analogovým výstupem. Zajímavou součástí desky je také modul reálného času. Druhý I2C port na RaspberryPi v sobě navíc ukrývá 5 V měnič napětí a ochranu ESD (ElectroStatic Discharge), umožňující tak připojení dalších zařízení. Pro jednoduché připojení jednotlivých sběrnic jsou na desce umístěny konektory.



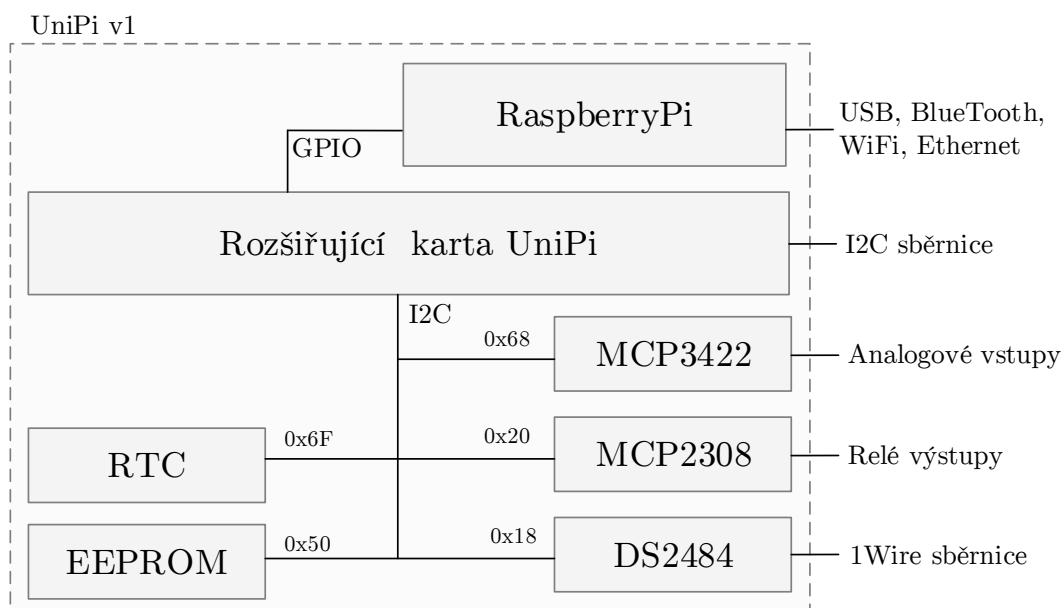
Obr. 3.1: UniPi v1 [42]

Popis desky

- 14 digitálních vstupů 5 – 24 V.
- 1Wire sběrnice pro měření teploty a vlhkosti.
- 8 přepínacích relé 250 V/5 A AC nebo 24 V/5 A DC.
- 1 Analogový výstup 0 – 10 V.
- 2 Analogové vstupy 0 – 10 V.
- Modul reálného času.
- I2C sběrnice.
- EEPROM paměť.
- UART sběrnice.
- Notifikační diody pro zobrazení stavu jednotlivých portů.

Velkou výhodou řídicí jednotky UniPi je zabudovaný čip pro obsluhu teplotních čidel na sběrnici 1Wire. Digitální teploměry mají svou adresu, není tedy nutné je jakkoliv kalibrovat či nastavovat, stačí zapojit.

S RaspberryPi je deska UniPi propojena 26 žilovým kabelem přes GPIO konektor. Jak bylo popsáno v kapitole 3.4.6 o konektoru GPIO, toto propojení je z důvodu kompatibility shodné pro všechny verze RaspberryPi. Vnitřní uspořádání desky je řešeno pomocí funkčních celků (znázorněno na Obr. 3.2), které jsou propojeny pomocí I2C sběrnice. Výstupy jednotlivých celků jsou poté vyvedeny na konektory desky.



Obr. 3.2: Blokové schéma UniPi v1

Napájení desky je řízeno jumperem JP1 a může být řešeno dvěma způsoby:

- Adaptérem 5 V/2 A do UniPi, s distribucí 5 V/750 mA do RaspberryPi.
- Samostatným napájením obou desek.

Pro účely testování a implementace byla zapůjčena deska UniPi s počítačem RaspberryPi 2. Vývoj této desky byl již ukončen a nahrazen druhou verzí, označovanou jako UniPi NEURON.

3.2 UniPi v2 - Neuron

UniPi Neuron představuje modulární PLC (Programmable Logic Controller) pro chytrou domácnost a inteligentní systémy budov, řízení a průmyslovou automatizaci. Díky modulární a kompaktní konstrukci nabízí jedinečnou variabilitu funkcí. UniPi Neuron je univerzální řídící jednotka. Neuron lze použít k řízení chytrého domu nebo

jako domácí server. Je vhodný pro monitorování, sběr a ukládání dat na vzdálený server, nebo jako výkonná a plně vybavená brána pro ostatní zařízení [43].



Obr. 3.3: UniPi rozšiřující deska

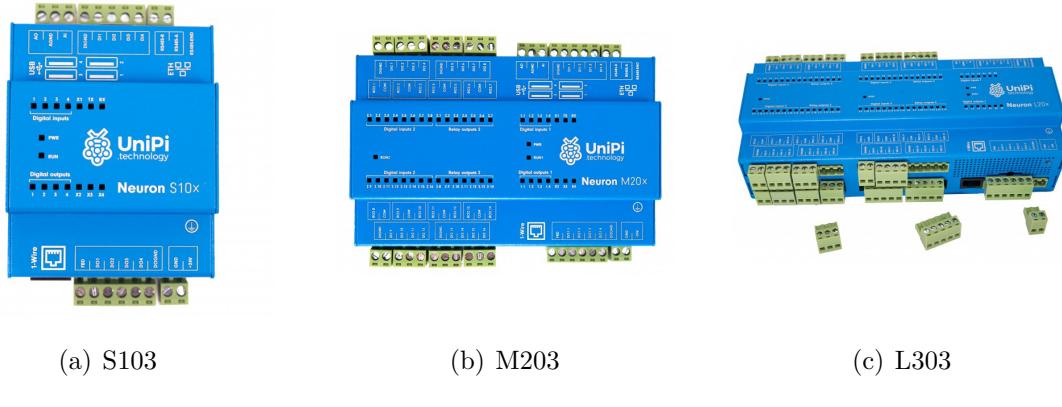
UniPi Neuron je na rozdíl od první verze, kdy se rozšiřující deska RaspberryPi distribuovala zvlášt, již hotové řešení, které se skládá z RaspberryPi, rozšiřující desky UniPi verze 2 (viz Obr. 3.3), propojovací desky pro komunikační moduly (viz Obr. 3.5(b)) a diodového panelu. To vše propojené a uzavřené v modrém plechovém pouzdro s možností montáže na DIN lištu. K dostání je tedy pouze jako hotový výrobek.

Popis desky

- Digitalní vstup 4 - 24 V (počet závislý na konkrétním modelu).
- Tranzistorový výstup 50V/750 mA (počet závislý na konkrétním modelu).
- Analogový výstup 0 - 10 V.
- Analogový vstup 0 - 10 V.
- 1Wire sběrnice.
- RS-485 .
- Modul reálného času.
- Notifikační diody pro zobrazení stavu jednotlivých portů.

UniPi Neuron existuje v několika verzích (viz Obr. 3.4), rozlišených počtem digitálních vstupů a výstupů, parametry procesoru a velikosti paměti RAM. Do budoucna by měly být také k dostání verze s jedním konkrétním modulem (Wireless M-Bus, ZigBee, GPRS, ...) uvnitř.

Standardní modely NEURON mají proměnlivé množství digitálních vstupů a reléových výstupů. Jejich počet je uveden v Tab. 3.1.



Obr. 3.4: Unipi Neuron [43]

Tab. 3.1: Porovnání modelů UniPi NEURON dle I/O [43]

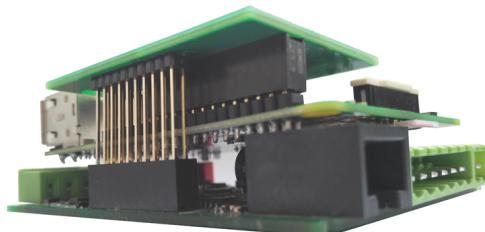
| Model | Počet digitálních vstupů | Počet digitálních výstupů | Velikost na DIN liště |
|-------|--------------------------|---------------------------|-----------------------|
| S10x | 4 | 0 | 4 moduly |
| M10x | 12 | 8 | 8 modulů |
| M20x | 20 | 14 | 8 modulů |
| M30x | 34 | 0 | 8 modulů |
| M40x | 4 | 28 | 8 modulů |
| L20x | 36 | 28 | 12 modulů |
| L30x | 64 | 0 | 12 modulů |
| L40x | 4 | 56 | 12 modulů |

Písmeno x v Tab. 3.2 bývá nahrazeno číslem 1-3 dle osazeného typu RaspberryPi:

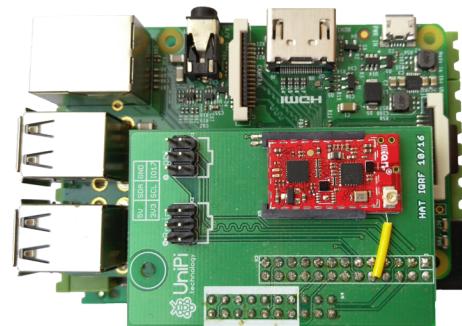
Tab. 3.2: Varianty modelů UniPi NEURON dle CPU a RAM [43]

| x | Osazená deska | CPU | RAM | Další vlastnosti |
|----------|----------------|--------------|--------|----------------------|
| 1 | RaspberryPi B+ | 700 MHz | 512 MB | |
| 2 | RaspberryPi 2 | 4 x 900 MHz | 1 GB | |
| 3 | RaspberryPi 3 | 4 x 1200 MHz | 1 GB | BT 4.1, WiFi 802.11n |

S RaspberryPi je deska UniPi propojena, obdobně jako u první verze, pomocí 26 pinové desky propoující GPIO port na RaspberryPi s konektorem na rozšiřující desce UniPi. Na samotné propoující desce (viz Obr. 3.5(a)) je vyvedena UART a I2C sběrnice.



(a) Propojení desek



(b) UniPi deska osazená WM-Bus modulem

Obr. 3.5: Detaily UNiPi desky

Na I2C sběrnici je dále připojen panel (viz Obr. 3.6(b)) se signalizačními diodami. UART sběrnice je zde připravena pro připojení dalších modulů. Tyto desky jsou k dostání v několika verzích, přizpůsobené konektorem pro konkrétní komunikační modul. Deska na obrázku 3.5(b) je osazena WM-Bus modulem.

Celá sestava desek je poté uložena v kovové krabičce s označením vstupů a výstupů (viz Obr. 3.6(a)).



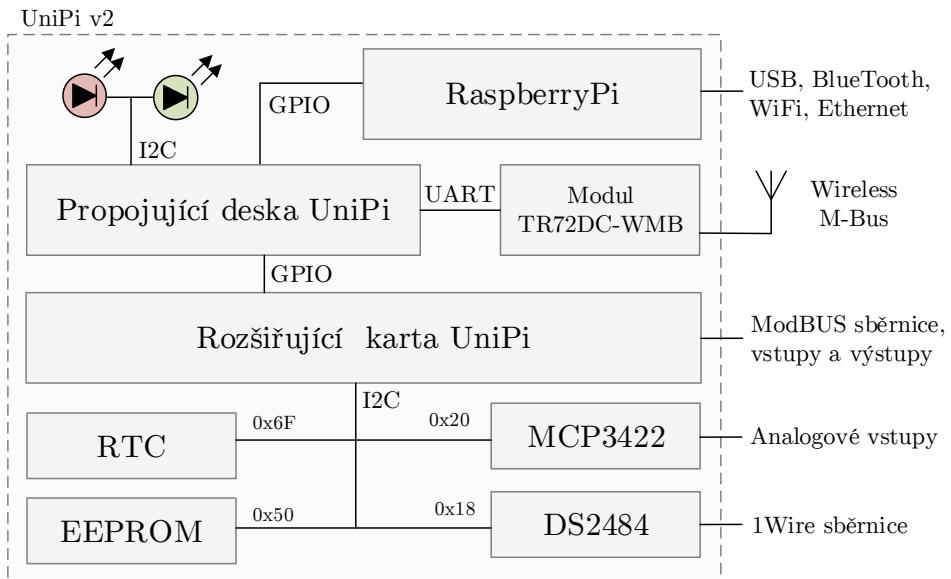
(a) Uložení v krabičce



(b) Připojení diodového panelu

Obr. 3.6: Detaily vnitřního uspořádání UniPi Neuronu S103

Vnitřní uspořádání desky je řešeno pomocí funkčních celků (viz Obr. 3.7), které jsou propojeny pomocí I2C sběrnice. Výstupy jednotlivých celků jsou poté vyvedeny na konektory desky.



Obr. 3.7: Blokové schéma UniPi v2

Napájení desky je pomocí 24 V/1,5 A adaptéru přímo na rozšiřující desku UniPi. Pro účely testování a implementace byla zapojěna deska UniPi Neuron S103 vybavená počítačem RaspberryPi 3.

3.3 Srovnání obou verzí

Jak bylo popsáno v předchozím textu (Kap. 3.1 a 3.2), obě verze UniPi se liší svými parametry a využitím. I když vývoj UniPi byl nahrazen vývojem UniPi NEURONu, stále se najdou aplikace vhodné pouze pro původní desku:

- Deska UniPi má reléově spínané výstupy, lze pomocí ní spínat i silové výstupy do 250 V. Zatímco UniPi NEURON má spínané tranzistorové výstupy pouze do 50 V, pro spínání vyšších napětí je nutné připojit reléový modul.
- Deska UniPi má zpřístupněnou I₂C a UART sběrnici, zatímco na UniPi Neuronu je I₂C využita pouze pro adresování vnitřních bloků a UART sběrnice je alokována pro rozšiřující komunikační moduly.
- Software EVOK a software postavené na něm jsou v tomto okamžiku plně funkční pouze na desce první verze.

Na některé aplikace však již tato deska vhodná není a je lepší využít UniPi NEURON:

- I když UniPi první verze má sběrnici UART a teoretecky do ní lze připojit stejné rozšiřující komunikační moduly jako do UniPi Neuronu, součástí vývoje budou jen rozšiřující desky pro UniPi NEURON, jejichž nabídka má obsahovat spoustu dostupných technologií.

- Vzhledem k rozsáhlé nabídce modelů UniPi NEURON lze zvolit řešení na míru, včetně další konektivity.
- UniPi NEURON disponuje sběrnici RS-485 s protokolem ModBUS.
- UniPi NEURON má na kontaktech vysouvací svorky a celý modul zabírá méně místa.

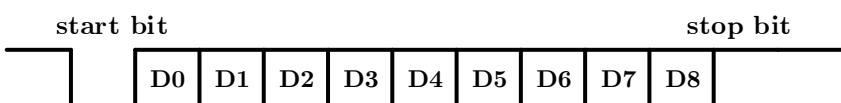
Vzhledem k tomu, že UniPi NEURON je v době vypracování práce jediná vývojem podporovaná verze, bude implementaci WM-Bus protokolu provedena na této verzi, avšak lze bez jakýchkoliv softwarových modifikací a pouze s jednou hardwarovou modifikací implementovat WM-Bus protokol i na UniPi první verze. Stačí pouze propojit příslušné piny IQRF modulu s piny modulárního konektoru UART sběrnice.

3.4 Sběrnice na UniPi

Jak je patrné z předchozích kapitol, jednodeskové počítače i rozšiřující moduly disponují množstvím komunikačních sběrnic. V této kapitole budou stručně představeny všechny zmíněné a pozornost bude zaměřena na sběrnici UART, která bude sloužit pro komunikaci mezi RaspberryPi a WM-Bus modulem.

3.4.1 UART

UART je synchronní a asynchronní sériové rozhraní pro přenos dat mezi zařízeními v obou směrech. Používá se pro komunikaci mezi mikrokontroléry, počítači a dalšími zařízeními podporující tento standard. Využívá dvouvodičovou sběrnici, vysílá data na pinu označovaném obvykle jako TX, přijímá na pinu RX.



Obr. 3.8: UART rámec

Pro přenos se používají rámce, které mohou mít 5 až 9 bitů a jsou od sebe odděleny jedním start bitem a jedním nebo dvěma stop bity. Každý rámec může obsahovat ještě paritní bit pro kontrolu rámce.

Dále je možné nastavit rychlosť přenosu dat od 1 200 bps až do 250 kbps. Lze nastavit buď pro asynchronní režim, označovaný jako SCI, například pro RS-232 či RS-485, anebo pro synchronní režim, běžně označovaný jako SPI. Tato sběrnice

je ve verzi 1 vyvedená do modulárního konektoru na desce, ve verzi 2 již není vyvedená na kontakty, ale je součástí desky plošného spoje, na kterém se přímo nachází slot pro komunikační modul.

3.4.2 SPI

SPI je sériové periferní rozhraní. Používá se pro komunikaci mezi řídícími mikroprocesory a ostatními integrovanými obvody. Jednotlivé obvody jsou propojeny čtyřmi vodiči:

- Datový výstup MOSI (Master Out, Slave In) obvodu Master je připojen na vstupy MOSI všech obvodů Slave.
- Datový vstup MISO (Master In, Slave Out) obvodu Master je propojen s výstupy MISO všech obvodů Slave.
- Výstup hodinového signálu SCK je připojen na vstupy SCK všech obvodů Slave.
- Každý obvod Slave má vstup SS (Slave Select) pro výběr obvodu.

Komunikace je realizována pomocí společné sběrnice, je typu master-slave. Adresace se provádí pomocí zvláštních vodičů, které při logické nule aktivují příjem a vysílání zvoleného zařízení. Tato sběrnice se ani v jedné z desek UniPi nepoužívá ani není vyvedena ven.

3.4.3 RS-485

RS-485 se používá především v průmyslovém prostředí. Vyznačuje se dvouvodičovým propojením jednotek. Tyto vodiče se označují písmeny A a B. Přenos je poloduplexní, a proto se vyžaduje řízení přenosu dat. Pomocí dvouvodičové linky je možné připojit až 32 zařízení. Tato sběrnice není součástí první verze desky, v druhé verzi je vyvedena na kontakty.

3.4.4 I2C

I2C je interní datová sběrnice sloužící pro komunikaci a přenos dat mezi jednotlivými integrovanými obvody většinou v rámci jednoho zařízení. Sběrnice je duplexní a dvoudrátová. Na jednu sběrnici může být připojeno více obvodů, v základní sedmibitové verzi až 128 obvodů. Vodiče jsou označeny jako serial data (SDA) a serial clock (SCL). Sběrnice je typu master-slave. Master při přenosu generuje hodinový signál na vodiči SCL. Když jeden obvod vysílá, všechny ostatní poslouchají a pouze podle adresy určují, zda jsou data určena jim. Obvod, který chce vyslat/přijmout data musí nejprve definovat adresu čipu, s kterým chce komunikovat a zda půjde

o příjem nebo vysílání - tedy o čtení nebo zápis. To určuje bit, který je součástí adresy.

Tato sběrnice je součástí obou verzí desky, využívá se pro propojení vnitřních funkčních bloků (EEPROM, RTC modul, AD převodník, 1Wire master, ...), v první verzi je také vyvedená do modulárního konektoru na desce, v druhé verzi již ne.

3.4.5 1Wire

Sběrnice 1Wire, navržená firmou Dallas Semiconductor, umožňuje připojit několik zařízení k řídící jednotce prostřednictvím pouhých dvou vodičů: data a zem. Sběrnice má jeden řídící obvod (master) a jeden či více ovládaných zařízení (slave). Všechny obvody jsou zapojeny jednak na společnou zem, a jednak paralelně na společný datový vodič. Tato sběrnice je součástí obou verzí desky, slouží pro připojení externích čidel (nejčastěji teplotní čidla) a u obou verzí desky je vyvedena do modulárního konektoru.

3.4.6 GPIO

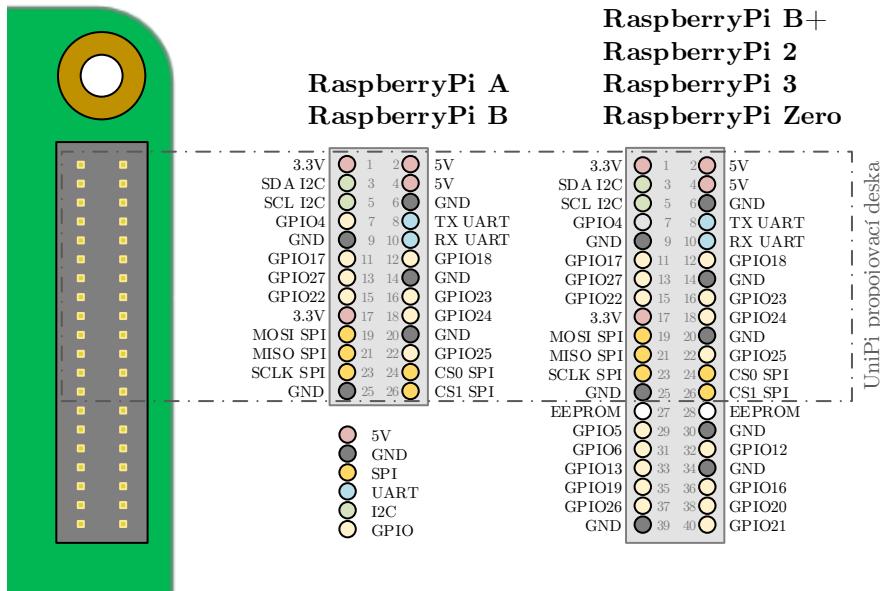
GPIO jsou piny, které lze programovat pomocí softwaru. Do těchto pinů lze posílat elektrický signál nebo jej z nich naopak přijímat. Na RaspberryPi 1 je takových vývodů celkem 26, na RaspberryPi 2 a RaspberryPi 3 je vývodů 40. GPIO vývodů je zde standardně 8, krom nich se zde nachází i dva piny pro UART, 2 pro I2C a 6 pro SPI, ty však jdou také přenastavit pro GPIO využití. Nelze opomenout ani dva výstupy s napětím (3,3 V a 5 V) a zem. Obrázek 3.9 demonstriuje rozložení GPIO konektoru napříč verzemi RaspberryPi.

Jak již bylo popsáno v předchozích kapitolách, GPIO konektor není ve všech verzích RaspberryPi shodný. Model RaspberryPi B má 26 pinový konektor, zatímco verze B+, 2 a 3 mají konektor 40 pinový. Rozdíl je v tom, že u 40 pinového konektoru je prvních 26 pinů shodných a konektor je na zbývajících 14 pinech rozšířen o další vstupy a výstupy. Je tedy zpětně kompatibilní.

3.5 Software pro UniPi

Hlavní výhodou otevřené platformy RaspberryPi je možnost použít zákazníkem zvolený libovolný software. Neexistují omezení ze strany výrobce, proto si může každý svoje řešení postavit na míru.

Výrobce poskytuje vlastní software EVOKE, který se stará o komunikaci desky UniPi přes virtuální server či API (Application Programming Interface) s uživatelem. Většina dalších open-source programů využívá toto API pro svůj provoz. Výrobcem



Obr. 3.9: Zpětná kompatibilita GPIO konektoru

je taktéž podporován software Mervis [47], který z UniPi dělá plnohodnotné PLC (Programmable Logic Controller). Dále je k dispozici několik open-source programů:

- EVOKE - oficiální Python API s websocket a REST podporou.
- PiDome - platforma pro domácí automatizaci.
- pimmaic - platforma pro domácí automatizaci založená na node.js.
- Node-RED - platforma založená na node.js s integrací do společnosti IBM Cloud Bluemix.
- Wyliodrin - programování automatizace na bázi prohlížeče.
- FHEM.de - domácí automatizační projekt napsaný v jazyce Perl.
- JEEDOM - automatizační projekt napsaný v jazyce PHP.

A tři komerční:

- Mervis - profesionální domácí automatizační řešení s on-line SCADA.
- REX - profesionální PLC s podporou mnoha průmyslových protokolů.
- HomeSeer - odlehčená platforma pro automatizaci domácnosti.

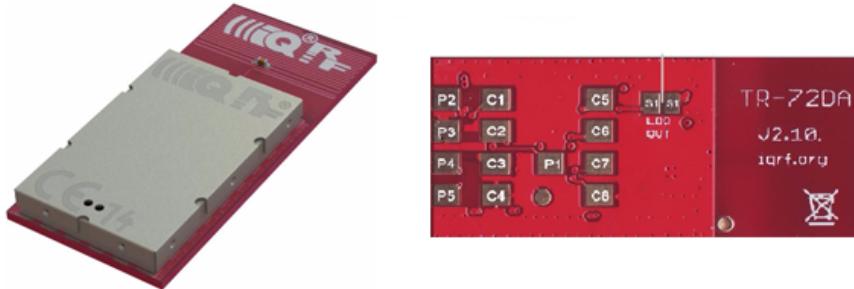
V době psaní této práce byl EVOKE k dispozici i pro druhou verzi desky, avšak bez podpory komunikačních modulů. **Implementace protokolu tedy bude prováděna jako samostatná aplikace s vlastní formou vizualizace naměřených dat.**

4 KOMUNIKAČNÍ MODUL WIRELESS M-BUS

Spolu se zařízením UniPi Neuron S103 byl zapůjčen i modul IQRF TR-72DC-WMB, který do budoucna bude součástí tohoto produktu a bude rozšiřovat konektivitu zařízení o protokol Wireless M-Bus.

4.1 Obecný popis modulu TR-72D-WMB

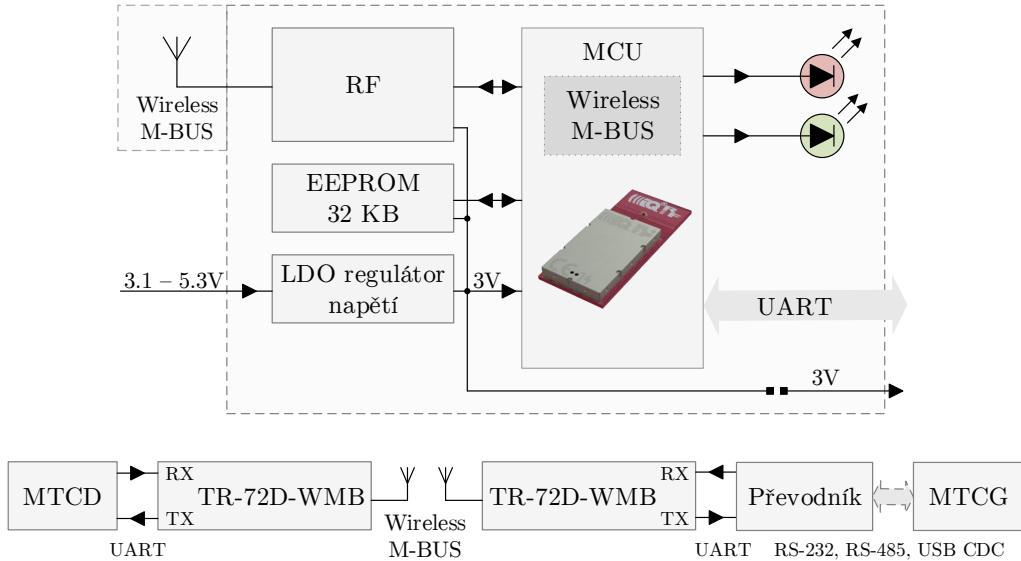
Modul IQRF TR-72DA-WMB je bezdrátový komunikační modul velikosti SIM karty z výroby české firmy MICRORISC s.r.o.. Vychází z řady produktů technologie IQRF, s tím rozdílem, že místo IQRF softwaru má přímo implementovaný Wireless M-Bus protokol [46].



Obr. 4.1: Modul IQRF TR-72DA-WMB [46]

Na malém prostoru se nachází vše potřebné pro uskutečnění bezdrátového přenosu: mikrokontrolér, externí EEPROM, teplotní senzor, kontrolní LED, 6 pinů a anténa dle typu komunikačního modulu (Obr. 4.2).

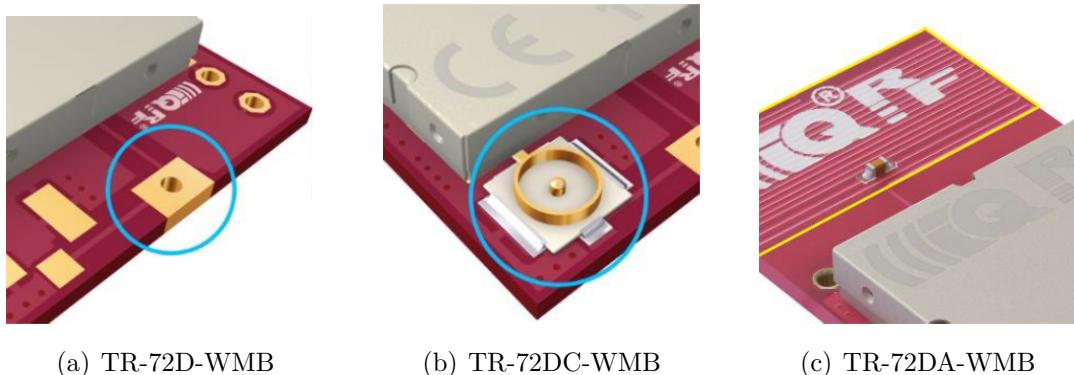
Modul podporuje módy přenosu S1, S2, T1 a T2. Napájecí napětí modulu je v rozsahu 3,1 až 5,3 V se spotřebou $1\mu\text{A}$ v režimu spánku a 8-22 mA ve vysílačním režimu, dle nastavení výstupního výkonu, jehož maximální hodnota je 12,5 W. V České republice je využíván pro přenos v bezlicenčním pásmu 868 MHz, případně 433 MHz nebo 169 MHz.



Obr. 4.2: Blokové schéma modulu TR-72D-WMB [46]

Modul je vyráběn ve třech verzích (viz Obr. 4.3) dle připojení antény:

- TR-72D-WMB má zdírku pro připájení antény.
- TR-72DC-WMB má vyveden koaxiální anténí konektor U.FL.
- TR-72DA-WMB má integrovanou anténu přímo na desce modulu. Dosah signálu toto typu je až 320 m v módu T a 365 m v módu S.



Obr. 4.3: Přehled typu modulu dle antény [46]

4.2 Komunikační módy

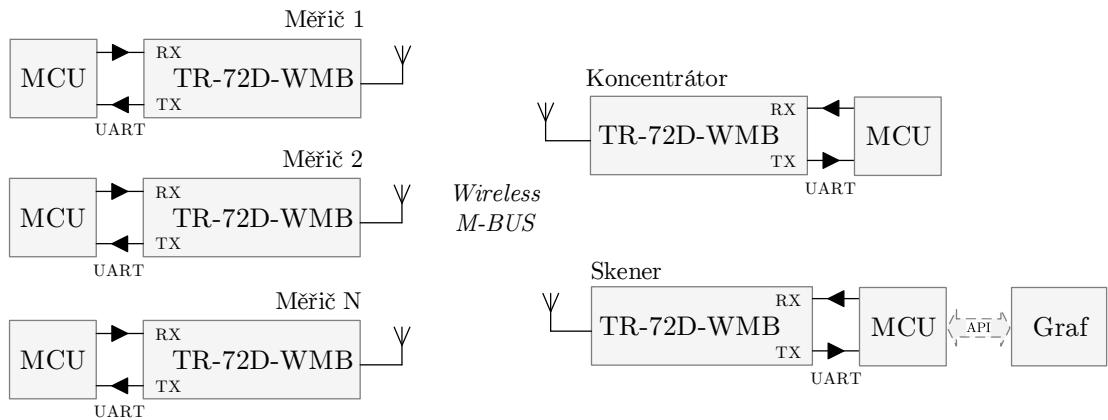
Modul může být v závislosti na použité topologii nastaven do jednoho ze tří provozních módů: měřič, koncentrátor, skener [46].

V módu měřiče může být modul přes UART sběrnici zapojen k mikrokontroléru, který zajistí zpracování dat od senzorů. Může tedy sloužit k sestavení vlastních měřicích zařízení postavených na protokolu Wireless M-Bus.

V módu koncentrátoru slouží modul jako komunikační zařízení pro sběr dat z meřičů. Aktuální firmware podporuje pouze obousměrnou komunikaci s měřiči v režimu S a T a je zatím ve fázi vývoje a do produkce nasazen jako experimentální. Z tohoto důvodu bude při implementaci samotné aplikace modul nasazen v režimu skeneru.

V módu skeneru modul zachytává veškerou dostupnou komunikaci daného módu přenosu. Díky vnitřní implementaci Wireless M-Bus protokolu je modul schopen zachytávat a dešifrovat šifrovanou komunikaci, je však nutné počítat s tím, že současný firmware není stavěný na vyžití modulu pro příjem šifrované komunikace v módu skeneru od více zařízení zároveň. Modul totiž automaticky veškeré zachycené šifrované telegramy automaticky rozšifruje pomocí jediného interního AES klíče. Jedná se však o klíč daného modulu, nikoliv vyčítaného zařízení. Při vyčítání šifrovaných dat je tedy nutné postupovat složitěji a provést nejdříve zpětné zašifrování daných dat tímto klíčem a až poté provést rozšifrování dat dle normy.

Praktické využití jednotlivých módu zobrazuje Obr. 4.4.



Obr. 4.4: Různé módy dle použité topologie [46]

4.3 Komunikační protokol

S řídícím mikrokontrolérem modul komunikuje pomocí rozhraní UART, jehož parametry jsou 19200 Bd, 8 bitů, žádná parita a 1 stop bit.

Modul podporuje jednoduchý formát příkazů sloužící k nastavení konfiguračních parametrů modulu i k samotné komunikaci s modulem. Každý příkaz začíná znakem **>**. Každá zpráva odpovědi začíná znakem **<**. Každému příkazu musí předcházet buďší znak NULL (0x00) následovaný 2 ms pauzou a každý paket příkazů je ukončen znakem CR (0x0D).

Obecná struktura [46] paketu příkazu je >[CC] [RW] [DATA] [CR], kde CC je jednobajtový kód daného příkazu, RW je jednobajtový příznak zápisu (:) či čtení dat (?) dat, DATA jsou zapisovaná data, pokud se jedná o zápis a CR je znak ukončení.

Obecná skruktura odpovědi je <[DATA] [CR], kde DATA obsahuje přenášená data či návratové kódy (OK pro správné dokončení příkazu, ERR1 pro chybu syntaxe a ERR2 pro neplatnou vstupní hodnotu). Některé bajty jsou kódovány v šestnáckové soustavě či využívají uložení BigEndian.

Například dotaz a odpověď pro aktuální AES klíč je:

```
>03?[CR]  
<010203040506070809a0b0c0d0e0f[CR]
```

a pro případnou změnu AES klíče:

```
>03:112233445566778899aabbccddeeff[CR]  
>OK[CR]
```

Ukázku jednoduché komunikace s modulem v jazyce Python obsahuje Kód 4.1.

```
import serial  
  
# Setup a serial port  
ser = serial.Serial(  
    port='/dev/ttyAMA0',  
    baudrate=19200,  
    parity=serial.PARITY_NONE,  
    stopbits=serial.STOPBITS_ONE,  
    bytesize=serial.EIGHTBITS,  
    timeout=1  
)  
output("SCR", "Device is ready: " + str(ser.isOpen()))  
  
# Set default AES key  
ser.write("\x00\x00>03:" + AES_IQRF_DEFAULT + "\x0D")  
y = ser.readline()  
output("SCR", "Default AES key set: " + y[1:])
```

Kód 4.1: Komunikace s modulem přes sériový port

5 WIRELESS M-BUS PROTOKOL

Wireless M-Bus je v Evropě perspektivní otevřený standard pro automatické měření, který pracuje v sub-gigahertzovém bezlicenčním pásmu v okolí 868 MHz. Wireless M-Bus se primárně zaměřuje na použití v SRD (Short Range Device) zařízeních pro bezdrátovou komunikaci s měřiči energií, jako jsou: voda, plyn, teplo, elektřina, atd. Měřiče energií, vybavené bezdrátovým rozhraním Wireless M-Bus jsou schopny komunikovat jak se stacionárními, tak i s mobilními čtecími zařízeními. Předpokládá se, že rádiová část měřiče je napájena z baterie a je schopna provozu po dobu 10-15 let bez výměny baterie. Na čtecích zařízeních, ať už stacionárních nebo mobilních, není takový požadavek na dobu provozu na baterie a čtecí zařízení mohou být napájena i z externího zdroje.

Wireless M-Bus má svůj původ v rámci norem Meter-Bus. Wireless Meter Bus je bezdrátovou variantou drátového Meter-Bus. To je standard zaměřený na aplikace pro sběr dat měřiče plynu, elektřiny a vody. Sběrnice je specifikována v evropské normě EN 13757 [49]. Tato specifikace je rozdělena do pěti částí (viz Tab. 5.1), z nichž jedna se zaměřuje na Wireless M-Bus.

Tab. 5.1: Popis standardu EN-13757 [48]

| Standard | Podrobnosti |
|------------|---|
| EN 13757-1 | Část 1 standardu definuje výměnu dat, která podrobně popisuje základní komunikaci mezi vodoměry a centrálním sběračem dat. Poskytuje přehled komunikačního systému. |
| EN 13757-2 | Tato část normy Meter Bus řeší fyzickou a spojovou vrstvu pro fyzický přenos dat pomocí kabelových spojů. Také popisuje protokol používaný pro přenos dat. |
| EN 13757-3 | Část 3 se týká speciální aplikační vrstvy. Ta popisuje standardní aplikační protokol používaný k tomu, aby se zachovala kompatibilita výrobců, což umožňuje zařízení od několika různých dodavatelů působit v jednom systému. |
| EN 13757-4 | Oddíl 4 popisuje bezdrátový systém. Jedná se o radiové odečet pro provoz v pásmu 868 MHz až 870 MHz. Tato část normy se zabývá fyzickou a linkovou vrstvou pro bezdrátová zařízení. |
| EN 13757-5 | Tato část definuje adresy předávání. To zahrnuje celou řadu návrhů na předávání datových rámců jako prostředek komunikace mezi měřičem a koncentrátem. |

5.1 Princip komunikace

Bezdrátová komunikace Wireless M-Bus fyzicky probíhá ve 12 kanálech v bezlicenčním vysílacím pásmu ISM (industrial, scientific and medical) okolo frekvence 868 MHz (2 kanály 868,3 a 868,95 MHz jsou využívány režimem S a T, 10 uživatelem volitelných kanálů 868,03 + n x 0,06 MHz v režimu R2), přičemž každý z výše

uvedených režimů vyžaduje různé požadavky. Těmi například jsou specifikovaný kanál, přesnost frekvence, toleranci přenosové rychlosti atd. Velmi dobrá je stabilita frekvence až 27 let (dle údaje výrobce). V případě použití čtvrtvlné antény (délky 8,2 cm), tak na přímou viditelnost vysílacího a přijímacího modulu je komunikační dosah 500 až 600 m.

Komunikace má hvězdicovitou strukturu, kdy několik měřících jednotek/snímačů přenáší svá naměřená data jedné centrální jednotce, obvykle tvořené koncentrátem. Ten tedy obvykle slouží pro příjem a shromažďování dat z několika měřících míst, z dále uvedených důvodů nikdy neinicializuje (nezahajuje) vzájemnou komunikaci. Pracuje tedy jako server (Master), tzn. že stále naslouchá a čeká na navázání komunikace měřící jednotkou a jí inicializovaný přenos dat. Ta tedy pracuje jako klient (Slave). V případě nastavené obousměrné komunikace přechází měřič/snímač do přijímacího režimu pouze po krátký čas jím navázané komunikace. Pouze v tomto momentu může koncentrátor vyslat nějaké jednotce řídící data. Časování je rozdílné pro různé režimy a je přesně specifikováno ve standardu.

Adresování ve Wireless M-Bus sběrnici je převzato z klasické drátové verze MBUSu. Zde však pouze klientské jednotky (měřiče/snímače) mají přidělenou adresu a využívají ji jak při příjmu, tak při vysílání. Každý koncentrátor by měl obsahovat tabulku adres, se kterými může komunikovat, resp. od kterých má přijímat data. Tato tabulka se obvykle vytváří automaticky během instalace/registrování nové jednotky do sítě. Samozřejmě je možné se obejít i bez ní, ale pak lze přijímat všechny snímače či měřiče v dosahu. Toho se dá využít jen v malých sítích.

5.2 Režimy přenosu

Nejdůležitější vlastností technologie WM-Bus je možnost bateriového napájení měřicích zařízení. V případě bezdrátové komunikace je výhodné například měřiče tepla nebo vodoměry napájet jen bateriově a tím eliminovat jakoukoliv nutnost pokládání kabelů. To ale znamená velmi omezenou spotřebu elektrické energie, aby baterie vydržely co nejdéle, alespoň několik let. V současné době v případě napájení modulu lze dosáhnout životnosti na jednu baterii až 12 let [58, 59]. Aby to však bylo možné, řízení přenosu dat musí co nejčastěji přecházet do nízkopříkonového stavu (sleep mode) a vysílat data jen v nutných případech v co nejkratších časových slotech. Proto také centrální zařízení (koncentrátor), který obvykle slouží pro příjem a shromažďování dat z několika měřících míst, nikdy nesmí inicializovat vzájemnou komunikaci.

Protokol podporuje několik režimů přenosu, lišících se dle požadavků na konkrétní aplikaci. Je definováno několik režimů označených jako S, T a R představu-

jící 3 různé různé přenosové rychlosti, které se dále dělí na režim 1 a 2, což značí jednosměrný či obousměrný přenos dat. U některých zařízení mohou být doplněny o režimy N, C a F. Tyto režimy jsou shrnuty v Tab. 5.2.

Tab. 5.2: Režimy přenosu WM-Bus protokolu [48]

| Mód | Mód přenosu | Směr | Frekvence | Kódování | Rychlos |
|-----|-----------------------------------|------------------------------|-----------|-----------------------|-------------|
| S | Stacionární | Jednosměrný, i obousměrný | 868 MHz | Manchester | 32,768 kbps |
| T | Častý vysílací | Jednosměrný, i obousměrný | 868 MHz | Manchester a 3 z 6 | 100 kbps |
| R | Častý přijímací | Jednosměrný, i obousměrný | 868 MHz | Manchester | 4,8 kbps |
| N | Úzkopásmový | Jednosměrný, i obousměrný | 169 MHz | NRZ | |
| C | Kompaktní | Jednosměrný, i obousměrný | 868 MHz | Manchester | 50 kbps |
| F | Častý vysílací i přijímací mód | Obousměrný | 433 MHz | NRZ | |

Mód S je určen pro jednosměrnou nebo obousměrnou komunikaci mezi pevnými nebo mobilními zařízeními. Centrální frekvence tohoto módu je 868,3 MHz s dobou provozu 0,02 % za hodinu. Přenosová rychlos je pro tento mód 32,768 kbps. Pro operační mód S jsou definovány tři submódy: S1, S1-m a S2. Submód S1 lze použít pro jednosměrnou komunikaci nevyžadující potvrzení o přijetí rámce a je určen pro aplikace, kdy se vysílá několikrát za den ke statickému přijímači. Pro kódování používají všechny submódy módu S kódování Manchester. Submód S1-m je modifikací submódu S1 pro komunikaci mezi čidlem a koncentrátorem, zasílaný rámec obsahuje zkrácenou hlavičku. Submód S2M podporuje oboustranou komunikaci v kontinuálních cyklech bez nutnosti probouzet zařízení.

V módu T měřič samostatně odesílá data, buď periodicky nebo aperiodicky (když jsou k dispozici). Pro přenos rámce z měřiče k dalším zařízením je použita přenosová rychlos 100 kbps s kódováním 3 z 6, zatímco komunikace v opačném směru má přenosovou rychlos 32,768 kbps a kódování je použito Manchester. Submód T1 je definován jako jednosměrná komunikace, při které měřič nevyžaduje potvrzení od příjemce o přijatém rámci. Měřič odešle data a přepne se do úsporného režimu. Zatímco submód T2 je definován jako obousměrná komunikace. Měřič po odeslání rámce krátkou dobu vyčkává na potvrzení od příjemce. Pokud měřič neobdrží odpověď přepne se do úsporného režimu. Pokud ve stanoveném čase příjemce odpoví,

naváže se obousměrná komunikace mezi měřičem a koncentrátorem.

V módu R měřič samostatně neodesílá změřená data, ale vyčkává na výzvu od koncentrátoru. Měřič je v úsporném režimu a v pravidelných úsecích se periodicky probouzí do režimu příjmu a očekává rámec. Když není přijat žádný validní wake-up rámec, měřič se přepne zpět do úsporného režimu. V opačném případě se naváže obousměrná komunikace mezi měřičem a koncentrátorem.

V režimech S, T a R je každý bajt vysílán s nejvíce důležitým bitem (MSB - Most Significant Bit) na prvním místě. Implementace MSB v jazyce Python je zobrazena v Kódu 5.1.

```
def MSB( bytes ) :  
    new = ""  
    size = len( bytes )  
    while ( size >0 ):  
        new = new + bytes [ size -2:size ]  
        size=size -2  
    return new
```

1
2
3
4
5
6
7

Kód 5.1: Implementace vyčítání uložení MSB

5.3 Struktura zasílaných dat

Komunikace probíhá následovně: nadřazené aplikace realizující aplikační vrstvu standardu M-Bus vyšlou svá data do RF modemu v podobě datové jednotky, která je zobrazena v Tab. 5.3:

| 1 Bajt | 1 Bajt | n Bajtů |
|--------|--------|----------|
| Length | CI | AppLayer |

Tab. 5.3: Formát datové jednotky [51]

Komunikační modul pracující jako modem dle požadavků standardu Wireless M-Bus automaticky přidá následující pole:

- Řídicí pole.
- Označení výrobce dle [50].
- Unikátní komunikační adresy založené na parametrech uložených v paměti modulu.
- Případně se ještě na závěr přidá informace o síle přijímaného signálu RSSI (Received Signal Strength Indication).

Takovýto paket se pak zašifruje (obvykle algoritmem AES-128) a přenáší se vzduchem. V případě, že se realizuje jen bezdrátové tunelování přenosu mezi dvěma

| | | | | | | |
|--------|--------|---------|---------|--------|----------|--------|
| 1 Bajt | 1 Bajt | 2 Bajty | 6 Bajtů | 1 Bajt | n Bajtů | 1 Bajt |
| Length | C | ManID | Address | CI | AppLayer | RSSI |

Tab. 5.4: Formát datové jednotky protokolu Wireless M-Bus [51]

Wireless M-Bus modemy, je povolen i režim bez zasílání adresy a jí přidružených informacích o měřící jednotce. Rámec se pak výrazně zjednoduší a jeho struktura je zobrazena v Tab. 5.5.

| | | | |
|--------|--------|----------|--------|
| 1 Bajt | 1 Bajt | n Bajtů | 1 Bajt |
| Length | CI | AppLayer | RSSI |

Tab. 5.5: Zkrácený formát datové jednotky [51]

Obsah pole AppLayer je již dán aplikační vrstvou definovanou ve standardu M-Bus, které se používá jako mechanizmus komunikace z linkové vrstvy do vyšších protokolových vrstev, a je tedy shodný s obsahem pro klasický drátový M-Bus přenos. Data následující za polem CI jsou již závislá na aplikační vrstvě M-Bus. Komunikace mezi měřící jednotkou a RF modemem či mezi koncentrátorem a RF modem obvykle probíhá prostřednictvím sériového přenosu UART, například s využitím RS-232, RS-485 či USB.

Při přenosu datové jednotky uvedené v Tab. 5.5 IQRF modulem dochází k jejímu rozšíření o položky uvedené v Tab. 5.6.

| | | | | | | |
|--------|--------|--------------|--------|--------|--------|--------|
| 1 Bajt | 1 Bajt | 12 - n Bajtu | 1 Bajt | 1 Bajt | 1 Bajt | 1 Bajt |
| Length | Status | ... | CRC | RSSI | CR | 0A |

Tab. 5.6: Formát datové jednotky po přijetí modulem IQRF

5.4 Popis jednotlivých vrstev

Norma EN 13757-4 specifikuje fyzickou a linkovou vrstvu. Na ně následně navazuje aplikační vrstva, která je shodná s původním M-Bus protokolem.

5.4.1 Fyzická vrstva Wireless M-Bus

Fyzická vrstva definuje jak mají být bity kódovány a vysílány, tedy radiofrekvenční charakteristiky a radiofrekvenční parametry. Fyzická vrstva je realizována hardwarem, případně v kombinaci s firmwarem daného hardware.

Wireless M-Bus dle normy ČSN EN 13757-4 [52] využívá tři pásmo pro tři různé módy komunikace: 868,3 MHz pro módy Sx, 868,95 MHz pro módy Tx a 868,33 MHz pro mód R2 jsou definovány tři různé operační módy komunikace. Všechny tři módy

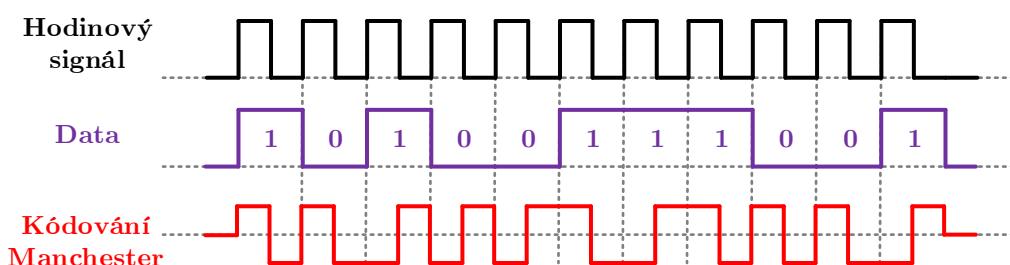
používají modulaci 2-FSK, tedy dvoustavovou frekvenční modulaci. Pro některé módy jsou některé parametry fyzické vrstvy stejné, proto je fyzické zařízení schopné s nezměněným hardwarem komunikovat v různých operačních módech.

Kódování používaná ve Wireless M-Bus

Wireless M-Bus definuje dvojí možné kódování:

- kódování Manchester,
- kódování 3 ze 6.

Kódování Manchester (viz Obr. 5.1) slučuje datový a hodinový signál do jediného signálu. Toto kódování se krom bezdrátových přenosů používá i v sítích LAN, konkrétně v síti Ethernet. Výhodou kódu Manchester je konstantní střední hodnota takového signálu, která je 50 % z maximální hodnoty. Náběžné hrany ohraničují jeden bit dat a sestupné hrany určují kód Manchester. Logická jednička je reprezentována náběžnou hranou a logická nula hranou sestupnou.



Obr. 5.1: Princip kódování Manchester

Pokud nejsou vysílána žádná data, výstup kódování Manchester je hodinový signál. Nevýhodou použití Manchester kódování je to, že na přenos jednoho bitu informace je potřeba dvou hodinových taktů.

Princip kódování 3 ze 6 (viz Tab. 5.7) spočívá v tom, že každé 4 bity (nibble) jsou zakódovány jako 6ti bitová data, přičemž zakódované slovo obsahuje stejné množství nul a jedniček. Zároveň v kódu musí být alespoň dvě změny, tzn. není možné použít „000111“ nebo „111000“. Takto zakódovaná data jsou přenášené s nejvýznamnějším bitem jako prvním. Toto kódování by mělo být aplikováno při použití módu častého vysílání (módy T1 a T2) a při komunikaci měřiče s koncentrátorem. Koncentrátor může odpovědět měřiči zprávou kódovanou kódováním Manchester.

5.4.2 Linková vrstva Wireless M-Bus

Linková vrstva poskytuje rozhraní mezi fyzickou a aplikační vrstvou. Její hlavní funkce jsou:

- Poskytování služeb převádějících data mezi fyzickou a aplikační vrstvou.

Tab. 5.7: Tabulka kódování 3 ze 6 [54]

| NRZ kód | Desítkově | 3 ze 6 | Desítkově | Počet změn v kódu |
|---------|-----------|--------|-----------|-------------------|
| 0 | 0 | 10110 | 22 | 4 |
| 1 | 1 | 1101 | 13 | 3 |
| 10 | 2 | 1110 | 14 | 2 |
| 11 | 3 | 1011 | 11 | 3 |
| 100 | 4 | 11100 | 28 | 2 |
| 101 | 5 | 11001 | 25 | 3 |
| 110 | 6 | 11010 | 26 | 4 |
| 111 | 7 | 10011 | 19 | 3 |
| 1000 | 8 | 101100 | 44 | 3 |
| 1001 | 9 | 100101 | 37 | 4 |
| 1010 | 10 | 100110 | 38 | 3 |
| 1011 | 11 | 100011 | 35 | 2 |
| 1100 | 12 | 110100 | 52 | 3 |
| 1101 | 13 | 110001 | 49 | 2 |
| 1110 | 14 | 110010 | 50 | 3 |
| 1111 | 15 | 101001 | 41 | 4 |

- Generování CRC pro odchozí zprávy.
- Detekování CRC chyb v příchozích zprávách.
- Poskytování adresování fyzické vrstvy.
- Kontrola ACK u obousměrných přenosů.
- Vytváření rámců.
- Kontrola chyb rámců v příchozích zprávách.

Rámec linkové vrstvy se skládá z bloků dat. Každý blok dat obsahuje 16bitové CRC pole. První blok má pevnou délku 12 bajtů a obsahuje L, C, M a A pole.

L-Pole

- Určuje velikost přenášených dat, ale bez samotného L-pole a kontrolního součtu.

C-Pole

- Identifikuje typ rámce (SEND, CONFIRM, REQUEST, RESPONSE).
- Používá se pro zasílání základních příkazů.

M-Pole

- Obsahuje identifikaci výrobce zařízení.
- Je kódováno jako třípísmenný kód, který se získává následovně:

$$\begin{aligned} \text{Manufacturer ID} = & [\text{ASCII}(Znak1) - 64] + 32 + 32 \\ & + [\text{ASCII}(Znak2) - 64] + 32 \\ & + [\text{ASCII}(Znak3) - 64] \end{aligned}$$

A-Pole

- Obsahuje 6 bajtů určující adresu zařízení.
- U rámců SEND a REQUEST je zde adresa vysílajícího zařízení.
- U rámců CONFIRM a RESPONSE je zde adresa zařízení, které je paket určen.
- Je tvořen následovně:
 - 4 bajty (identifikační číslo) kódované jako 8 BCD znaků. Jedná se o unikátní identifikaci stanovenou výrobcem.
 - 2 bajty (verze zařízení) určující generaci daného zařízení ve výrobním procesu výrobce.
 - 2 bajty (typ zařízení), kódované dle Tab. 5.8.

Tab. 5.8: Identifikace typu zařízení

| Hodnota | Bit 16 | Bit 15 | Bit 8 | Bit 7 | Médium |
|---------|--------|--------|-------|-------|--------------------|
| 0 | 0 | 0 | 0 | 0 | Ostatní |
| 1 | 0 | 0 | 0 | 1 | Olej |
| 2 | 0 | 0 | 1 | 0 | Elektřina |
| 3 | 0 | 0 | 1 | 1 | Benzín |
| 4 | 0 | 1 | 0 | 0 | Vytápění |
| 5 | 0 | 1 | 0 | 1 | Pára |
| 6 | 0 | 1 | 1 | 0 | Horká voda |
| 7 | 0 | 1 | 1 | 1 | Voda |
| 8 | 1 | 0 | 0 | 0 | Tepelné čerpadlo |
| 9 | 1 | 0 | 0 | 1 | Rezervováno |
| A | 1 | 0 | 1 | 0 | Benzín 2 |
| B | 1 | 0 | 1 | 1 | Vytápění 2 |
| C | 1 | 1 | 0 | 0 | Horká voda 2 |
| D | 1 | 1 | 0 | 1 | Voda 2 |
| E | 1 | 1 | 1 | 0 | Tepelné čerpadlo 2 |
| F | 1 | 1 | 1 | 1 | Rezervováno |

CI-Pole

- Určuje typ přenášených dat.
- Nejčastější typy jsou uvedeny v Tab. 5.9.

Tab. 5.9: Kódování CI-Pole

| Hodnota | Direction | Protokol |
|---------|-----------------------------|-------------|
| 50h | Výběr aplikace zařízení | pouze M-Bus |
| 51h | Požadavek na zařízení | pouze M-Bus |
| 52h | Výběr zařízení | pouze M-Bus |
| 5Ah | Požadavek na zařízení | (W)M-Bus |
| 5Bh | Požadavek na zařízení | M-Bus |
| 60h | Požadavek na zařízení | DLMS |
| 61h | Požadavek na zařízení | DLMS |
| 64h | Požadavek na zařízení | SML |
| 65h | Požadavek na zařízení | SML a |
| 6Ch | Synchronizace času zařízení | všechny OMS |
| 6Dh | Synchronizace času zařízení | všechny OMS |
| 6Eh | Chyba zařízení | všechny OMS |
| 6Fh | Chyba zařízení | všechny OMS |
| 70h | Chyba zařízení | pouze M-Bus |
| 71h | Alarm zařízení | pouze M-Bus |
| 72h | Odpověď zařízení | (W)M-Bus |
| 74h | Alarm zařízení | všechny OMS |
| 75h | Alarm zařízení | všechny OMS |
| 78h | Odpověď zařízení | (W)M-Bus |
| 7Ah | Odpověď zařízení | (W)M-Bus |
| 7Ch | Odpověď zařízení | DLMS |
| 7Dh | Odpověď zařízení | DLMS |
| 7Eh | Odpověď zařízení | SML |
| 7Fh | Odpověď zařízení | SML |

CRC

- CRC obsahuje kontrolní součet pro kontrolu správnosti přenosu.
- Jako kontrolní polynom se dle specifikace používá $x^{16} + x^{13} + x^{12} + x^{11} + x^{10} + x^8 + x^6 + x^5 + x^2 + 1$.

RSSI

- Received Signal Strength Indication.
- Určuje sílu signálu při přijetí paketu.
- Pro převod je využita lineární konverze:

$$\text{RSSI } [\text{dBm}] = \text{RSSI_LEVEL}/2 - 130$$

5.4.3 Aplikační vrstva Wireless M-Bus

V souladu se specifikací OMS (Open Metering Standard) 3.0.1 [62], která vychází z normy EN 13757-4 [52] pro bezdrátový protokol WM-Bus, jsou některé položky aplikační vstvy shodné pro většinu zařízení protokolu WM-Bus.

Access Number

- Binárně kódované pořadí přístupu.
- Při každém odeslání paketu je jeho hodnota zvýšena o jedničku.
- Po dosažení hodnoty 254 se začíná odznova.

Status

- Obsahuje chyby vysílajícího zařízení.
- Může nastat i několik chyb zároveň.
- Definované chyby jsou uvedeny v Tab. 5.10.

Tab. 5.10: Hodnoty Status pole

| Bit | Hex hodnota | Význam |
|-----|-------------|------------------------|
| 0 | 00h | Žádná chyba |
| | 01h | Aplikace zaneprázdněna |
| 1 | 02h | Obecná chyba aplikace |
| | 03h | Neočekávaný stav |
| 2 | 04h | Vybitá baterie |
| 3 | 08h | Trvalá chyba |
| 4 | 10h | Dočasná chyba |
| 5 | 20h | Specifický kód výrobce |
| 6 | 40h | Specifický kód výrobce |
| 7 | 80h | Specifický kód výrobce |

Struktura zbytku aplikační vrstvy je dána opakováním určité sekvence (viz Tab. 5.11) bajtů, určující typ a hodnotu přenášených dat.

Tab. 5.11: Struktura dat aplikační vrstvy

| Byte 1 | Byte 2 | Byte 3 | Byte 4 | Byte 5-n |
|------------------------------|---|-------------------------------|---|-------------|
| Data Information Block (DIB) | | Value Information Block (VIB) | | Data Values |
| Data Information Field (DIF) | Data Information Field Extension (DIFE) | Value Information Field (VIF) | Data Information Field Extension (VIFE) | |

Data Information Block (DIB)

DIB definuje typ přenášených dat a skládá se z DIF a z nepovinného DIFE.

Data information Field (DIF)

DIF definuje datový typ přenášených dat a má strukturu dle Tab. 5.12.

Tab. 5.12: Kódování DIF Pole

| Bit 1 | Bit 2 | Bit 3 | Bit 4 | Bit 5 | Bit 6 | Bit 7 | Bit 8 |
|-----------------|-------------|-----------------|-------|-------|-------|-------|-------|
| Rozšiřující Bit | LSB uložení | Funkční Položka | | | | | Data |

Rozšiřující bit pole určuje jaký blok bajtů následuje po DIF. Možnosti jsou shrnutý v Tab. 5.13.

Tab. 5.13: Kódování rozšiřujícího bitu DIF pole

| Bit | Další informace je obsažena v |
|-----|-------------------------------|
| 0 | DIF |
| 1 | DIFE |

Funkční pole definuje typ přenášené hodnoty z hlediska její aktuálnosti či limitnosti. Možnosti jsou shrnutý v Tab. 5.14.

Data pole určuje datový typ přenášené hodnoty. Možnosti jsou shrnutý v Tab. 5.15.

Data Information Field Extension (DIFE)

DIFE obsahuje upřesnění veličiny či informace o tarifu dle struktury zobrazené v Tab. 5.16.

Tab. 5.14: Kódování funkčního pole DIF pole

| Hodnota | Význam |
|---------|----------------------------|
| 00b | Okamžitá hodnota |
| 01b | Minimální hodnota |
| 10b | Maximální hodnota |
| 11b | Hodnota při chybovém stavu |

Tab. 5.15: Kódování Data pole DIF pole

| Délka hodnoty [b] | Kód | Význam | Kód | Význam |
|-------------------|------|----------------|------|-------------------|
| 0 | 0000 | Žádná data | 1000 | Volba pro hodnotu |
| 8 | 0001 | 8-bit Integer | 1001 | 2 cifry BCD |
| 16 | 0010 | 16-bit Integer | 1010 | 4 cifry BCD |
| 24 | 0011 | 24-bit Integer | 1011 | 6 cifer BCD |
| 32 | 0100 | 32-bit Integer | 1100 | 8 cifer BCD |
| 32 | 0101 | 32-bit Real | 1101 | Proměnlivá délka |

Tab. 5.16: Kódování DIFE Pole

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-----------------|----------|-------|-------|-------|-------|---------|-------|
| Rozšiřující bit | Jednotka | Tarif | | | | Hodnota | |

Value Information Block (VIB)

VIB definuje typ přenášené hodnoty a skládá se z VIF a z nepovinného VIFE.

Value Information Field (VIF)

VIF definuje veličinu přenášených dat a má strukturu dle Tab. 5.17.

Tab. 5.17: Kódování VIF Pole

| Bit 1 | Bit2 | Bit 3 | Bit 4 | Bit 5 | Bit 6 | Bit 7 | Bit 8 |
|-----------------|------|-------|-------|-------|-------|-------|-------|
| Rozšiřující bit | | | | | Data | | |

Rozšiřující bit pole určuje jaký blok bajtů následuje po VIF. Možnosti jsou shrnutý v Tab. 5.18.

Data pole určuje datový typ přenášené hodnoty. Možnosti jsou shrnuty v Tab. 5.19.

Tab. 5.18: Kódování rozšiřujícího bitu VIF pole

| Bit | Další informace je obsažena v |
|-----|-------------------------------|
| 0 | VIF |
| 1 | VIFE |

Value Information Field Extension (VIFE)

VIFE obsahují upřesnění, doplňující informaci či přenos chybového stavu dané položky. Jejich kompletní přehled je uveden ve specifikaci [55].

Data Value

Pole Data Value již obsahuje přenášenou hodnotu, definovanou dle DIB a VIB.

Datové typy F a G

V protokolu Wireless M-Bus je datum kódováno ve formátu G, viz Tab. 5.21 a datum i čas ve formátu F, viz Tab. 5.22.

Ukázky implementace obou formátů v jazyce Python uvádí Kód 5.2.

```
# Get date in G format
1
def get_date(date_bytes):
2
    date = str(bin(int(date_bytes[0:2], 16))[2:]) . zfill(8) +
3
        str(bin(int(date_bytes[2:4], 16))[2:]) . zfill(8)
4
    year = str(int(date[0:4] + date[8:11], 2))
5
    month = str(int(date[4:8], 2))
6
    day = str(int(date[11:16], 2))
7
    vysledek = day + " ." + month + ".20" + year
8
    return vysledek
9

# Get time from F format
10
def get_time(time_bytes):
11
    time = str(bin(int(time_bytes[0:2], 16))[2:]) . zfill(8) +
12
        str(bin(int(time_bytes[2:4], 16))[2:]) . zfill(8)
13
    hour = str(int(time[3:8], 2))
14
    minute = str(int(time[10:16], 2)) . zfill(2)
15
    vysledek = hour + ":" + minute
16
    return vysledek
```

Kód 5.2: Implementace F a G formátu

Tab. 5.19: Kódování Data pole VIF pole

| Bity | Veličina | Jednotka | Rozsah |
|-----------|------------------------|------------------------------------|---------------------------|
| E000 0nnn | Energie | $10^{(nnn-3)}$ Wh | 0.001 Wh - 10000 Wh |
| E000 1nnn | Energie | $10^{(nnn)}$ J | 0.001 kJ - 10000 kJ |
| E001 0nnn | Objem | $10^{(nnn-6)}$ m ³ | 0.001 l - 10000 l |
| E001 1nnn | Hmotnost | $10^{(nnn-3)}$ kg | 0.001 kg - 10000 kg |
| E010 00nn | Provozní čas | nn = 00 sekundy nn = 01 minuty | |
| E010 01nn | Operační čas | nn = 10 hodiny nn = 11 dny | |
| E010 1nnn | Výkon | $10^{(nnn-3)}$ W | 0.001 W - 10000 W |
| E011 0nnn | Výkon | $10^{(nnn)}$ J/h | 0.001 kJ/h - 10000 kJ/h |
| E011 1nnn | Průtok | $10^{(nnn-6)}$ m ³ /h | 0.001 l/h - 10000 l/h |
| E100 0nnn | Průtok | $10^{(nnn-7)}$ m ³ /min | 0.0001 l/min - 1000 l/min |
| E100 1nnn | Průtok | $10^{(nnn-9)}$ m ³ /s | 0.001 ml/s - 10000 ml/s |
| E101 0nnn | Protok (hmotnosti) | $10^{(nnn-3)}$ kg/h | 0.001 kg/h - 10000 kg/h |
| E101 10nn | Teplota (průtoku) | $10^{(nn-3)}$ °C | 0.001 °C - 1 °C |
| E101 11nn | Teplota (návratová) | $10^{(nn-3)}$ °C | 0.001 °C - 1 °C |
| E110 00nn | Teplota (rozdíl) | $10^{(nn-3)}$ K | 1 mK - 1000 mK |
| E110 01nn | Temperature (externí) | $10^{(nn-3)}$ °C | 0.001 °C - 1 °C |
| E110 10nn | Tlak | $10^{(nn-3)}$ bar | 1 mbar - 1000 mbar |
| E110 110n | Datum a čas | n = 0 datum n = 1 datum a čas | Datový typ F a G |
| E110 1110 | Tepelná výměna | | bezrozměrné |
| E110 1111 | Rezervováno | | |
| E111 00nn | Průměrné trvání | nn = 00 sekundy nn = 01 minuty | |
| E111 01nn | Aktuální trvání | nn = 10 hodiny nn = 11 dny | |
| E111 1000 | Výrobní číslo | | |
| E111 1001 | Rozšířená identifikace | | Datový typ C |
| E111 1010 | Adresa sběrnice | | Datový typ C |

Tab. 5.20: Kódování VIFE Pole

| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-----------------|----------|-------|-------|-------|---------|-------|-------|
| Rozšiřující bit | Jednotka | Tarif | | | Hodnota | | |

Tab. 5.21: Kódování data ve formátu G

| | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-----------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Bajt 1 | | | | | | | | | | | | Bajt 2 | | | | | | | | | | | |
| Rok (1/2) | | | | | | Den | | | | | | Rok (2/2) | | | | | | Měsíc | | | | | |

Tab. 5.22: Kódování data a času ve formátu F

| | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------|-------|--------|-------|-------|-------|-------|-------|-------|-------|-------|--------|--------|-------|-------|-------|-------|---------------|-------|-------|-------|-------|-------|-------|--------|--------|
| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | | |
| Bajt 1 | | | | | | | | | | | | Bajt 2 | | | | | | | | | | | | Bajt 3 | Bajt 4 |
| 0 | 0 | Hodina | | | | | | 0 | 0 | 0 | Minuta | | | | | | Dle formátu G | | | | | | | | |

5.5 Šifrování dat

Pro šifrování přenášených dat se v protokolu Wireless M-Bus používají tři šifrovací algoritmy:

- DES (Data Encryption Standard) bez inicializačního vektoru,
- DES s inicializačním vektorem a
- AES (Advanced Encryption Standard) s inicializačním vektorem.

Šifrování DES dnes již není moc využívané, je již nedostačující a zastaralé. Drtivá většina dnešních zařízení umožňujících šifrování přenos využívá šifrování AES, konkrétně verzi AES128 CBC.

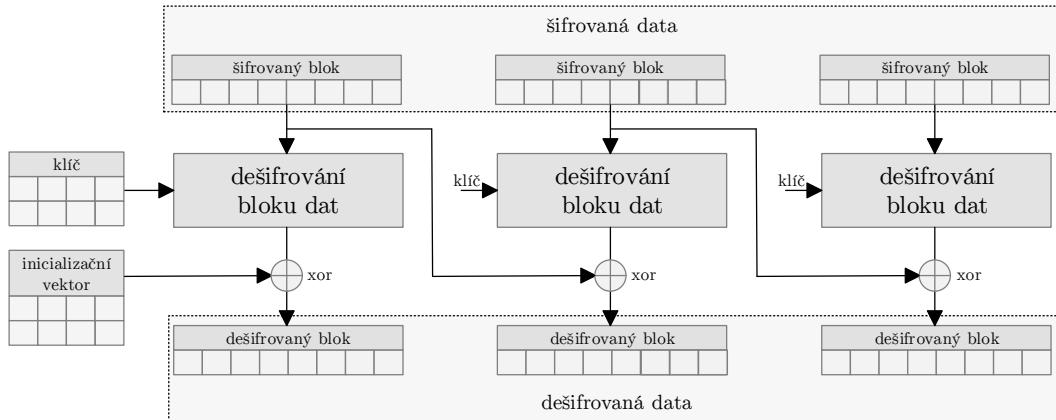
5.5.1 Šifrovací algoritmus DES

Data Encryption Standard je v kryptografii symetrická šifra vyvinutá v 70. letech. V roce 1977 byla zvolena za standard FIPS 46 [56]. V současnosti je tato šifra považována za nespolehlivou, protože používá klíč pouze o délce 64 bitů, z toho 8 je kontrolních a 56 efektivních. Navíc algoritmus obsahuje slabiny, které dále snižují bezpečnost šifry. Díky tomu je možné šifru prolomit útokem hrubou silou za méně než 24 hodin.

5.5.2 Šifrovací algoritmus AES

Advanced Encryption Standard je symetrická bloková šifra (pro šifrování i dešifrování využívá stejný klíč na data s pevně danou délkou bloku), která nahradila

dříve užívanou šifru DES [57]. AES šifra je rychlá v softwaru i hardwaru a na rozdíl od svého předchůdce DES nepoužívá Feistelovu síť. AES má pevně danou velikost bloku na 128 bitů a velikost klíče na 128, 192 nebo 256 bitů. Pokud jsou šifrovaná data delší, zpracovávají se po jednotlivých blocích.



Obr. 5.2: Princip algoritmu AES v módu CBC

Pro šifrovaný přenos dat v protokolu WM-Bus se využívá AES, konkrétně mód s inicializačním vektorem (CBC - Cipher Block Chaining). Ten funguje (viz Obr. 5.2) tak, že před zašifrováním se odpovídající blok otevřeného textu xoruje předcházejícím blokem zašifrovaného textu. To znamená, že jednotlivé bloky jsou na sobě závislé, aby došlo k dešifrování konkrétního bloku, je nutné dešifrovat i všechny předchozí. Je tedy nutné mít nějaký nulový blok dat pro zašifrování prvního bloku dat. K tomu se využívá inicializační vektor. Tomuto bloku se pak říká inicializační vektor (IV). Tento blok se použije k dešifrování prvního bloku a pak zahodí.

5.5.3 Inicializační vektor

Inicializační vektor má délku 16 bajtů (128 bitů, odtud označení AES-128) a v případě protokolu WM-Bus je tvořený dynamicky z nešifrovaných bajtů polí paketu, způsobem popsaným v Tab. 5.23 a implementovaných dle Kódu 5.3.

```

# Build Initialization Vector from incoming packet data
device = parsedstring[8:24].upper()
access = str(parsedstring[26:28])
AES_IV = binascii.unhexlify(device + access * 8)

```

Kód 5.3: Sestavení inicializačního vektoru

První 2 bajty obsahují přidělené identifikační údaje výrobce, další čtyři obsahují sériové číslo daného zařízení, následující dva obsahují verzi zařízení a zbylých osm Bajtů je tvořeno opakováním se přístupového čísla. Vzhledem k faktu, že přístupové

Tab. 5.23: Formát inicializačního vektoru

| Bit | Obsah | Význam |
|-----|---------------|-----------------------|
| LSB | M-Pole | Identifikace výrobce |
| 1 | | |
| 2 | | |
| 3 | | |
| 4 | | |
| 5 | | |
| 6 | | |
| 7 | | |
| 8 | Access Number | Identifikace zařízení |
| 9 | | |
| 10 | | |
| 11 | | |
| 12 | | |
| 13 | | |
| 14 | | |
| MSB | Access Number | Identifikace paketu |

číslo se s každým vysláním telegramu změní, je nutné inicializační vektor přepočítat pro každý přijatý paket. Tím je zajištěna dynamičnost šifrování danou metodou.

5.5.4 Šifrovací klíč

Šifrovací klíč AES je sekvence bajtů o velikosti 128, 192 nebo 256 bitů. Tento klíč slouží pro šifrování a dešifrování přenášených dat a je unikátní pro každé vyčítané zařízení. Bez znalosti tohoto klíče nelze tedy vycítat zařízení se šifrovaným přenosem dat.

5.5.5 Určení šifrovaných dat

Aplikační vrstva protokolu WM-Bus obsahuje položku ConfigurationWord případně **SignatureField**, která deklaruje typ použitého šifrovací algoritmu, délku šifrované části a způsob datového šifrování. Pole je složeno ze dvou bajtů. První bajt obsahuje NNNNCCHHb a druhý bajt obsahuje BASOMMMMb. Význam jednotlivých položek je shrnut v Tab. 5.24.

Tab. 5.24: Význam bitů pole ConfigurationWord

| Bit | Označení | Význam |
|-----|----------|------------------------|
| MSB | B | Obousměrnost |
| 14 | A | Dostupnost |
| 13 | S | Synchronizace |
| 12 | 0 | Synchronizace |
| 11 | M | Šifrování |
| 10 | M | Šifrování |
| 9 | M | Šifrování |
| 8 | M | Šifrování |
| 7 | N | Počet kódovaných bloků |
| 6 | N | Počet kódovaných bloků |
| 5 | N | Počet kódovaných bloků |
| 4 | N | Počet kódovaných bloků |
| 3 | C | Obsah telegramu |
| 2 | C | Obsah telegramu |
| 1 | H | Počítací skoků |
| LSB | H | Počítací skoků |

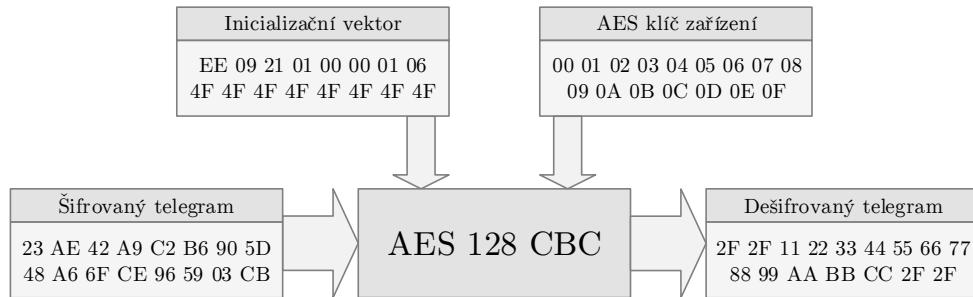
Bity šifrování nabývají těchto hodnot:

- 4 pro AES se statickým inicializačním vektorem,
- 5 pro AES s dynamickým inicializačním vektorem,
- 6 je rezervovaná,
- 7 až 15 jsou pro využití výrobcem.

Pro režim AES s dynamickým inicializačním vektorem bity jsou 0101 a vyjadřují hodnotu 5.

5.5.6 Princip dešifrování

Pro dešifrování přijatých dat je nutná znalost šifrovacího algoritmu, šifrovacího klíče a sestavení inicializačního vektoru. Potom lze aplikací dešifrovacího algoritmu získat přenášená data. Obecné schéma dešifrování AES-128 CBC je znázorněno na Obr. 5.3 a implementováno v Kódu 5.4.



Obr. 5.3: Obecné schéma dešifrování AES-128 CBC

```

1 from Crypto.Cipher import AES
2
3 encryptor = AES.new(AES_KEY_IQRF, AES.MODE_CBC, IV=AES_IV)
4 OUTPUT_DECRYPTED = encryptor.decrypt(INPUT_ENCRYPTED)
  
```

Kód 5.4: Implementace AES

5.5.7 Kontrola rozšifrování dat

Ke kontrole správnosti dešifrovaných dat slouží definovaná počáteční sekvence dat. U algoritmu DES začínají dešifrovaná data dvěma bajty obsahujícími datum a čas. Pro algoritmus AES jsou první dva bajty šestnáckové a oba obsahují znak 2Fh, jak znázorňuje Kód 5.5.

```

1 # Verify control sequence after decrypt
2 aes_control = binascii.hexlify(TELEGRAM_ORIGINAL[0:2]).upper()
3 if (aes_control == b'2F2F'):
4     binascii.hexlify(TELEGRAM_ORIGINAL).upper().decode('ascii')
  
```

Kód 5.5: Ověření kontrolní sekvence AES

6 WIRELESS M-BUS ZAŘÍZENÍ

Pro účely testování komunikace bylo využito několik typů dostupných zařízení:

- pokojové čidlo teploty a vlhkosti Weptech OMST-868A [58],
- modul pro vodoměry Bonega [59],
- ultrazvukový měřič tepla a chladu Kamstrup Multical 402 [60],
- třífázový elektroměr ZPA ZE302 [61].

Všechna tyto zařízení poskytují formát dat dle platné specifikace OMS (Open Metering Standard) 3.0.1 [62], která vychází z normy EN 13757-4 [52] pro bezdrátový protokol WM-Bus.

Pro základní komunikaci bylo zvoleno čidlo teploty a vlhkosti Weptech OMST-868A, z důvodu volné dostupnosti kompletní dokumentace a možností nastavení parametrů vysílání včetně volitelného šifrování přenášených dat. Jako jediné z výše jmenovaných čidel nevyžaduje ke své činnosti žádná doplňující média či přístroje.

6.1 Weptech OMST-868A

Weptech OMST-868A je teplotní a vlhkostní čidlo podporující protokol Wireless M-Bus. Je určeno pro vnitřní využití a proto je dodáváno v pouzdře určeném pro montáž na zed.



(a) Zapouzdřené čidlo

(b) Deska čidla

Obr. 6.1: Čidlo Weptech OMST-868A [58]

Parametry čidla

- Rozsah měření vlhkosti: 20 až 80 %.
- Přesnost měření vlhkosti: $\pm 2\%$.
- Rozsah měření teploty: -10 °C až 55 °C.
- Přesnost měření teploty: $\pm 0,3\text{ }^{\circ}\text{C}$.
- Teplotní hystereze: 0,1 °C.
- Mód přenosu: S nebo T.
- Interval přenosu: konfigurovatelný v rozsahu 5 sekund až 24 hodin.

- Šifrování přenosu: volitelný AES-128 mód 5.
- Napájení: 2 x AA baterie.
- Výdrž baterie: dle módu a intervalu přenosu až 10 let.

Formát telegramu

Telegram má specifickou základní strukturu popsanou v Tab. 6.1 [58]:

Tab. 6.1: Telegram ze zařízení Weptech 868A [58]

| Pole | Popis pole | Hodnota |
|--------------------|--|---------|
| L-Field | Délka telegramu | 1Eh |
| C-Field | Typ telegramu | 44h |
| M-Field | Výrobce zařízení | B0h |
| M-Field | Výrobce zařízení | 5Ch |
| A-Field | Sériové číslo | 11h |
| A-Field | Sériové číslo | 47h |
| A-Field | Sériové číslo | 15h |
| A-Field | Sériové číslo | 08h |
| A-Field | Verze zařízení | 01h |
| A-Field | Typ zařízení | 1Bh |
| Ci-pole | Odpověď od zařízení | 7Ah |
| Access Number | Číslo přístupu | 41h |
| Status | Status zařízení | 00h |
| Configuration word | Konfigurační řetězec AES | 00h |
| Configuration word | Konfigurační řetězec AES | 00h |
| AES verification | Ověření AES | 2Fh |
| AES verification | Ověření AES | 2Fh |
| DR1 | DIF: 4 cifry BCD | 0Ah |
| DR1 | VIF: teplota ve stupních Celsia na mínus první | 66h |
| DR1 | Hodnota teploty | 99h |
| DR1 | Hodnota teploty | 01h |
| DR2 | DIF: 4 cifry BCD | 0Ah |
| DR2 | VIF: První rozšiřovací tabulka | FBh |
| DR2 | VIFE: vlhkost procentech na mínus první | 1Ah |
| DR2 | Hodnota relativní vlhkosti | 93h |
| DR2 | Hodnota relativní vlhkosti | 02h |
| DR3 | DIF: 16bit integer/binary | 02h |
| DR3 | VIF: Druhá rozšiřovací tabulka | FDh |
| DR3 | VIFE0: Chybové stavy | 97h |
| DR3 | VIFE1: Norma | 1Dh |
| DR3 | Příznak sabotáže | 00h |
| DR3 | Příznak vybité baterie | 00h |
| Fill | Výplňové byty (13x) | 2Fh |

Některé z položek je potřeba blíže vysvětlit:

- Příznak sabotáže čidla - Pokud čidlo pomocí integrovaného spínače detekuje uvolnění krytu z montážní desky, pošle výstrahu přes rádio do přijímače, tedy změní pro nejbližší a všechny následující vysílání tamper bit v telegramu. Tento bit slouží jako ochrana před neoprávněnou manipulací s čidlem a může být vymazán pouze restartem zařízení. Tedy vyjmutím starých baterií, ponecháním zařízení několik minut bez napájení, aby došlo k vybití všech kondenzátorů a následným vložením baterií.
- Příznak vybité baterie - Pokud elektronika v čidle vyhodnotí úroveň nabité baterie jako nedostatečnou, nastaví bit vybití baterie do sekce chyb v telegramu. Tento bit ošetruje stavy, kdy nedostatečně nabité baterie způsobí příliš velký rozptyl naměřených hodnot, v krajních případech i mimo měřící rozsah čidla. Tento bit může být vymazán také pouze restartem zařízení, jako v předchozím případě.
- Položky hodnota teploty, hodnota vlhkosti, výrobce zařízení a sériové číslo jsou uloženy v kódování big-endian, tedy na paměťové místo s nejnižší adresou se uloží nejvíce významný bajt a za něj se ukládají ostatní bajty až po nejméně významný bajt na konci. Uživatelská hodnota se tedy vyčítá pozpátku po jednotlivých bajtech.
- Telegram je ukončen 13 výplňovými bajty, které nenesou žádnou informaci.

Nastavení čidla

Čidlo má k dispozici několik nastavení. Některé z nich lze nastavit pomocí čtyř přepínačů DIP (Dual Inline Package) na desce plošných spojů. První přepínač zapíná AES-128 šifrování, druhý přepínač přepíná mezí módem vysílání S (poloha ON) a módem T (poloha OFF), třetí a čtvrtý přepínač určují interval zasílání telegramu, jejich nastavení shrnuje Tab. 6.2.

Tab. 6.2: Konfigurace intervalu zasílání pomocí DIP přepínače [58]

| Interval zasílání | přepínač DIP 3 | přepínač DIP 4 |
|-------------------|----------------|----------------|
| 1 minuta | ON | ON |
| 5 minut | OFF | ON |
| 10 minut | ON | OFF |
| 15 minut | OFF | OFF |

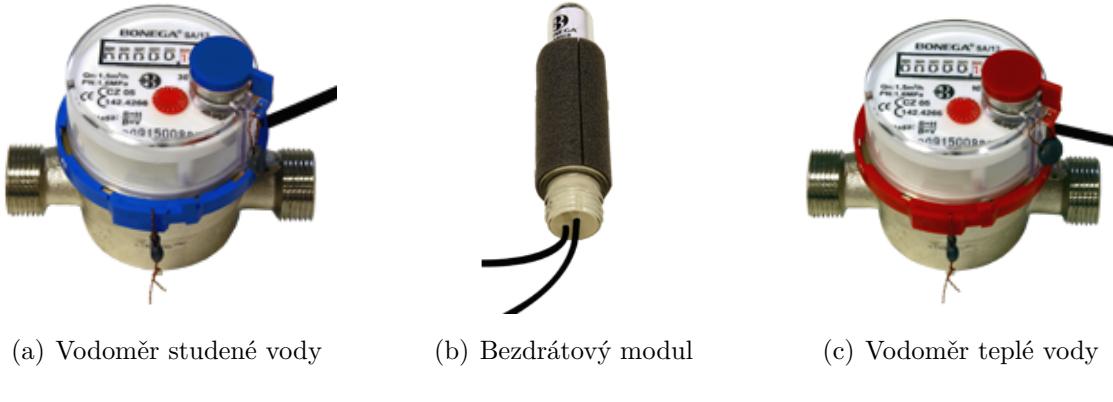
Jiné mohou být nastaveny pouze během výroby daného setu, a to do příslušných továrních nastavení, či uživatelsky vyžádaných nastavení, viz Tab. 6.3.

Tab. 6.3: Přehled nastavení čidla [58]

| Parametr | Popis | DIP přepínač |
|-----------------------|---|--------------|
| AES enable | Možnost zapnutí či vypnutí šifrování přenášených dat. | 1 |
| wM-Bus mode | Implementovány jsou módy S1-m a T1. Ostatní módy lze nastavit pouze při tovární výrobě. | 2 |
| Transmission interval | Interval je výrobcem konfigurovatelný v intervalu 2 až 65534 sekund. Předvolby (60s, 300s, 600s, 900s) jsou uživatelsky nastavitelné pomocí DIP přepínače. | 3 a 4 |

6.2 Bonega

Modul Bonega je bezdrátové čidlo podporující protokol Wireless M-Bus. Jedná se o samostatné zařízení, které je určené pro montáž na vodoměry Bonega. Na řadu vodoměrů podporujících tento modul je možná i dodatečná montáž. Elektronická část modulu slouží současně pro vyčítání dvou vodoměrů, na teplou i studenou vodu.



Obr. 6.2: Sada Bonega [59]

Parametry modulu

- Rozsah měření: 0 až 65536 m².
- Přesnost měření: ± 1 litr.
- Maximální detekovatelný průtok: 6 m³/hod.
- Mód přenosu: T1.
- Stupeň krytí: IP64.
- Interval přenosu: 20-24 sekund v odpočtovém období (od 1.11. do 15.1.)

- Interval přenosu: 4 minuty mimo odpočtové období.
- Šifrování přenosu: AES-128 mód 5.
- Napájení: integrovaná baterie.
- Výdrž baterie: až 12 let.

Formát telegramu

Zařízení vysílá postupně dva telegramy, s rozlišením šestým bajtem adresy zařízení, jeden pro vodoměr teplé vody a druhý pro vodoměr studené vody. Telegram má specifickou základní strukturu popsanou v Tab. 6.4 [59]:

Tab. 6.4: Telegram z modulu Bonega [59]

| Pole | Popis pole | Hodnota |
|-----------------|---|---------|
| L-Field | Délka telegramu | 1Eh |
| C-Field | Typ telegramu | 44h |
| M-Field | Výrobce zařízení | B0h |
| M-Field | Výrobce zařízení | 5Ch |
| A-Field | Sériové číslo | 11h |
| A-Field | Sériové číslo | 47h |
| A-Field | Sériové číslo | 15h |
| A-Field | Sériové číslo | 08h |
| A-Field | Verze zařízení | 01h |
| A-Field | Typ zařízení | 1Bh |
| Ci-pole | Odpověď od zařízení | 7Ah |
| Access Number | Číslo přístupu | 41h |
| Status | Status zařízení | 00h |
| Signature field | Konfigurační řetězec AES | 00h |
| Signature field | Konfigurační řetězec AES | 00h |
| Data | Ověření AES | 2Fh |
| Data | Ověření AES | 2Fh |
| Data | DIF: 2 cifry BCD (určení formátu průtoku) | 04h |
| Data | VIF: Objemový průtok v m krychlových na mínuš první | 13h |
| Data | Hodnota průtoku | 99h |
| Data | DIF: 2 cifry BCD (určení formátu času) | 6Dh |
| Data | DIF: 2 cifry BCD | 6Dh |
| Data | Čas odeslání měření | 99h |
| Data | Čas odeslání měření | 99h |
| Data | Čas odeslání měření | 99h |
| Data | Ověření AES | 2Fh |
| Data | Ověření AES | 2Fh |

Modul Bonega pracuje pouze v režimu šifrování přenášenných dat pomocí AES128 mód 5. Při přenosu je tedy celá sekce data šifrována, telegram popsaný v Tab. 6.4 je popisován po dešifrování.

Některé z položek je potřeba blíže vysvětlit:

- Hodnota průtoku - aktuální hodnota průtoku je zde vyjádřena čtyřmi hexadecimálními bajty v LSB pořadí.
- Čas odeslání měření - Datum a čas provedení posledního měření. Nejedná se o čas posledního odečtu či odeslání posledního telegramu.

6.3 Kamstrup

Kamstrup Multical 402 je kompatkní ultrazvukový měřič tepla a chladu, tedy kombinace kalorimetru a ultrazvukového průtokoměru. Je určen k měření tepla, chladu a kombinovanému měření tepla a chladu ve všech systémech na bázi vody s rozmetím teplot 2 °C az 130 °C. Skládá se z kalkulátoru, průtokoměru a dvou teplotních snímačů.



Obr. 6.3: Kamstrup Multical 402 [60]

Parametry zařízení

- Rozsah měření průtoku: 0,6 až 15 m³/hod.
- Rozsah měření teploty vody: 2 až 180 °C.
- Rozsah teploty vody kalkulátorem: 2 až 130 °C.
- Mód přenosu: T1 nebo C1.
- Interval přenosu: 15 minut

- Šifrování přenosu: AES-128 mód 5.
- Napájení: integrovaná baterie.
- Výdrž baterie: až 16 let.

Měřič pracuje v režimech T1 a C1 povinným šifrováním přenášených dat pomocí AES128 v módu CBC. Při každém přenosu poskytuje 9 aktuálních hodnot a 2 souhrnné hodnoty za poslední rok:

- Tepelná energie (přívodní nebo vratné potrubí).
- Energie chladu (přívodní nebo vratné potrubí).
- Energie vratného potrubí.
- Energie přívodního potrubí.
- Měření průtoku.
- Měření výkonu.
- Minimální a maximální průtok a výkon.
- Měření teploty.

Souhrnné hodnoty jsou ve výchozím stavu zasílány jako roční přehled, ale lze je překonfigurováním WM-Bus modulu v měřiči změnit na měsíční interval.

Formát telegramu

Telegram je ve výchozím stavu vysílán každých 15 minut pro aktuální hodnoty a má specifickou základní strukturu popsanou v Tab. 6.5 [60], kde sekce DR1 až DR12 postupně odpovídají měřeným hodnotám:

1. Tepelná energie (přívodní nebo vratné potrubí).
2. Energie chladu (přívodní nebo vratné potrubí).
3. Energie vratného potrubí.
4. Energie přívodního potrubí.
5. Průtok.
6. Výkon.
7. Teplota.
8. Minimální průtok a výkon.
9. Minimální výkon.
10. Maximální průtok.
11. Maximální výkon.
12. Čas provedení odečtu.

Tab. 6.5: Telegram ze zařízení Kamstrup Multical 402 [60]

| Pole | Popis pole | Hodnota |
|--------------------|------------------------------|---------|
| A-Field | Sériové číslo | 11h |
| A-Field | Sériové číslo | 37h |
| A-Field | Sériové číslo | 35h |
| A-Field | Verze zařízení | 11h |
| A-Field | Typ zařízení | 04h |
| Ci-pole | Odpověd od zařízení | 72h |
| AccNo | Číslo přístupu | CAh |
| Status | Status zařízení | 10h |
| Configuration word | Nastavení šifrování | 50h |
| Configuration word | Nastavení šifrování | 05h |
| AES encryption | Kontrola dešifrování | 2Fh |
| AES encryption | Kontrola dešifrování | 2Fh |
| DR1 | DIF: 4 binární bajty | 04h |
| DR1 | VIF: energie v kWh | 0Fh |
| DR1 | hodnota energie | 99h |
| ... | ... | .. |
| DR12 | DIF: 4 binární bajty | 04h |
| DR12 | VIF: průtok v m ³ | 14h |
| DR12 | hodnota průtoku | 99h |
| Fill | Výplňové bajty | 2Fh |

6.4 ZPA

ZPA ZE310 je třífázový elektronický dvoutarifní elektroměr pro měření činné energie, určený pro přímé i nepřímé připojení. Daný model vysílá v módu T2 s intervalem vysílání jedna minuta, vysílány jsou hodnoty spotřeby v obou tarifech.



Obr. 6.4: ZPA ZE.310 [61]

Parametry zařízení

- Počet měřených fází: 3 (daný subtyp).
- Počet tarifů: 2 (daný subtyp).
- Režim krytí: IP54.
- Mód přenosu: T1.
- Interval přenosu: 1 minuta.

Formát telegramu

Telegram je ve výchozím stavu vysílán každou minutu a poskytuje výrobcem doplněnou základní strukturu popsanou v Tab. 6.6 [61].

Tab. 6.6: Telegram ze zařízení ZPA ZE.302 [61]

| Pole | Popis pole | Hodnota |
|-----------------------|---------------------------|---------|
| L-Field | Délka telegramu | 2Ah |
| C-Field | Typ telegramu | 44h |
| M-Field | Výrobce zařízení | 01h |
| M-Field | Výrobce zařízení | 6Ah |
| A-Field | Sériové číslo | 44h |
| A-Field | Sériové číslo | 93h |
| A-Field | Sériové číslo | 67h |
| A-Field | Sériové číslo | 12h |
| A-Field | Verze zařízení | 01h |
| A-Field | Typ zařízení | 02h |
| CRC | Kontrolní součet | 22h |
| CRC | Kontrolní součet | 80h |
| Ci-pole | Odpověď od zařízení | 72h |
| Identification Number | Sériové číslo | 44h |
| Identification Number | Sériové číslo | 93h |
| Identification Number | Sériové číslo | 67h |
| Identification Number | Sériové číslo | 12h |
| Manufacturer ID | Výrobce zařízení | 01h |
| Manufacturer ID | Výrobce zařízení | 6Ah |
| Version | Verze zařízení | 01h |
| Device type | Typ zařízení | 02h |
| AccNo | Číslo přístupu | CAh |
| Status | Status zařízení | 10h |
| Configuration word | Položky šifrovaní AES | 50h |
| Configuration word | Položky šifrovaní AES | 05h |
| DIF | DIF | 86h |
| DIFE | DIFE | 20h |
| VIF | VIF | 83h |
| CRC | Kontrolní součet | C8h |
| CRC | Kontrolní součet | 97h |
| VIFE | VIFE | 00h |
| DATA1 | Hodnota spotřeby tafifu 1 | 76h |
| DATA1 | Hodnota spotřeby tafifu 1 | 23h |

| Pole | Popis pole | Hodnota |
|-------------|---------------------------|----------------|
| DATA1 | Hodnota spotřeby tafifu 1 | 85h |
| DATA1 | Hodnota spotřeby tafifu 1 | 01h |
| DATA1 | Hodnota spotřeby tafifu 1 | 00h |
| DATA1 | Hodnota spotřeby tafifu 1 | 00h |
| DIF | DIF | 86h |
| DIFE | DIFE | 20h |
| VIF | VIF | 83h |
| VIFE | VIFE | 00h |
| DATA2 | Hodnota spotřeby tafifu 2 | 97h |
| DATA2 | Hodnota spotřeby tafifu 2 | 31h |
| DATA2 | Hodnota spotřeby tafifu 2 | 92h |
| DATA2 | Hodnota spotřeby tafifu 2 | 00h |
| DATA2 | Hodnota spotřeby tafifu 2 | 00h |
| CRC | Kontrolní součet | C8h |
| CRC | Kontrolní součet | 97h |
| Data | Kontrolní data | 00h |
| CRC | Kontrolní součet | C8h |
| CRC | Kontrolní součet | 97h |

Položky DATA1 a DATA2 obsahují hodnoty obou tarifů uložené v LSB a vyjádřené v KWh.

7 NÁVRH IMPLEMENTACE

Jak již bylo zmíněno na začátku práce, samotná implementace je rozdělena do dvou částí:

1. Komunikace RaspberryPi přes rozšiřující desku UniPi s bezdrátovým modulem a pomocí něj s poskytnutými WM-Bus zařízeními.
2. Zachytávání šifrované i nešifrované komunikace s WM-Bus zařízeními. Uložení, analýza a následná vizualizace zachycených dat.

Jelikož žádný z dostupných softwarů pro UniPi nepodporuje daný bezdrátový modul, ani UART zařízení obecně, je nutné tuto komunikaci implementovat již na úrovni operačního systému.

7.1 Výběr OS

Jako operační systém je využita aktuální verze Raspbianu Jessie s datem vydání 2017-03-02. UART rozhraní se na RaspberryPi verze 1 a 2 nachází v `/dev/ttyAMA0`. To se ale v případě RaspberryPi 3 odkazuje na integrovaný BT modul a původní sériový port je zde v `/dev/ttys0`. Samotné UART rozhraní je ale ve výchozím nastavení Raspbianu zakázáno.

Pro zpřístupnění UART rozhraní je nutné provést drobné úpravy jeho konfigurace:

1. Nejdříve je nutné provést kompletní aktualizaci Raspbianu, tedy v konzoli spustit posloupnost příkazů:

```
sudo apt-get update  
sudo apt-get upgrade  
sudo apt-get dist-upgrade  
sudo apt-get rpi-upgrade
```

2. Poté je potřeba v `/boot/config.txt` změnit položku `ENABLE_UART` na hodnotu 1. Tím dojde k zpřístupnení sběrnice UART. Tato položka může být v budoucnu při aktualizaci Raspbianu přepsána, proto při prvním náznaku nefunkčnosti, je potřeba tuto položku zkontolovat jako první.
3. V souboru `/boot/cmdline.txt` je potřeba odebrat text `console=ttyAMA0, 115200`, aby při startování systému nedocházelo k výpisu do sériové linky.
4. V případě, že se jedná o RaspberryPi verze 3, je potřeba do `/boot/config.txt` dopsat položku `dtoverlay=pi3-minuart-bt`, která zakáže BT na mini-UART a provede přemapování zpět na `/dev/ttyAMA0`. Tento krok je takto řešený z důvodu kompatibility, kdy je sériová komunikace směrována přes `/dev/ttyAMA0`.

nezávisle na použité verzi RaspberryPi.

Po každém z těchto kroků je doporučován restart zařízení. Kroky byly otestovány pouze na výše zmíněné verzi Raspbianu a v jiných distribucích se monou mírně lišit. Úspěšnost provedení těchto kroků lze zkontrolovat pomocí zadání příkazu konzole `sudo dmesg | grep tty` jehož výstup by měl být následující:

```
[0.000974] console [tty1] enabled
[0.130442] 20201000.uart: ttyAMA0 at MMIO 0x20201000
(irq = 81, base_baud = 0) is a PL011 rev2
```

7.2 Výběr programovacího jazyka

Jelikož primárním jazykem využívaným na platformě RaspberryPi je Python, který již obsahuje knihovny pro sériovou komunikaci, je současný kód napsán v programovacím jazyce Python 3.

7.3 Nastavení komunikačního modulu a čidla

Před samotným vyčítáním dat bylo potřeba zjistit či nastavit přenosové parametry všech použitých zařízení:

- Komunikační modul IQRF nastaven do módu T1 ve funkci skeneru.
- Čidlo Weptech je nastaveno do módu T1 s intervalem zasílání 1 minuta.
- Modul Bonega je nastaven do módu T1 se zapnutým šifrováním AES128 v módu 5 a s intervalem zasílání 20-24 sekund v odpočtovém období a intervalem 4 minuty mimo odpočtové období.
- Elektroměr ZPA je nastaven do módu T2 s intervalem vysílání 1 minuta.
- Měřič Kamstrup je nastaven do módu T1 s intervalem vysílání 15 minut.

7.4 Zajištění dedikovaného běhu

Pro zajištění běhu aplikace nezávisle na typu provozu RaspberryPi bude daný program spouštěn ihned po startu operačního systému pomocí příkazu screen. Je tedy nutné ho doinstalovat:

```
sudo apt-get update
sudo install screen
```

7.5 Zajištění podpory šifrování

Některá ze zařízení používají pro přenos dat šifrování. Pro zajištění podpory šifrování byla zvolena knihovna PyCrypto, která podporuje jak šifrování DES tak i AES. Umožňuje pohodlnou implementaci AES128 pomocí jazyku Python3. Na rozdíl od ostatních knihoven není závislá na balíčku OpenSSL a je součástí repozitářů Raspbianu.

Je nutné doinstalovat nezbytné balíčky:

```
sudo apt-get update  
sudo install python-crypto  
sudo install python-dev
```

7.6 Zpracování dat

7.6.1 Nešifrovaný přenos

Jednoduchým spuštěním komunikačního modulu v módu skeneru byl zachycen telegram

```
32002E44B05C10000000021B7A620800002F2F0A6699010AFB1  
A930202FD971D01002F2F2F2F2F2F2F2F2F2F2F2F879e0D0A
```

který byl pomocí datasheetu použitého komunikačního modulu [46] a čidla [58] analyzován, a přehledně zobrazen do Tab. 7.1.

7.6.2 Šifrovaný přenos

V okamžiku, kdy bylo zařízení přepnuto do šifrovaného módu dle Tab. 6.3, byl zachycen šifrovaný telegram

```
32002e44b05c10000000021b7ac40820053ed44a38a9c3c86f5  
8210f9b979353c39dc1d5e0c873eb81994d28c099ef1d55b008
```

který byl pomocí datasheetů použitého komunikačního modulu [46] a normy [49, 57] analyzován a byly vyparovány položky nezbytné pro dešifrování dat:

- 30-33 pro informaci použitém šifrování,
- 8-25 pro sestavení inicializačního vektoru a
- 38-93 pro šifrovanou část dat.

Poté byla daná data v souladu s normou [57] dešifrována dle Kap. 7.9.4 a byl získán dešifrovaný telegram:

| |
|---|
| 32002e44b05c10000000021b7ac40820052F2F0A6699010AFB1 |
| A930202FD971D01002F2F2F2F2F2F2F2F2F2F879e0D0A |

který je až na číslo přístupu a CRC shodný s předchozím nešifrovaným telegramem. Nyní tedy lze dešifrovaná data vyparovat jako při nešifrovaném přenosu popsaném v předchozí kapitole.

Tab. 7.1: Rozklíčovaný zachycený paket

| Pozice | Bajty | Pole | Popis | Hodnota | Vyjádření | Význam pro uživatele |
|--------|-------|--------------|-----------------------------|---------|-----------|----------------------------|
| 4 | 2E | L-Pole | Délka telegramu | 1Eh | 46 | Paket má 46 bytů |
| 6 | 44 | C-Pole | Typ telegramu | 44h | 44 | Paket je typu SND-NR |
| 8 | B0 | M-Pole | Výrobce zařízení | B0h | 5CB0 | Výrobcem je WEPtech |
| 10 | 5C | M-Pole | Výrobce zařízení | 5Ch | | |
| 12 | 10 | A-Pole | Sériové číslo | 11h | 10 | SN je 00000010 |
| 14 | 00 | A-Pole | Sériové číslo | 47h | | |
| 16 | 00 | A-Pole | Sériové číslo | 15h | | |
| 18 | 00 | A-Pole | Sériové číslo | 08h | | |
| 20 | 02 | A-Pole | Verze zařízení | 01h | 2 | Druhá verze zařízení |
| 22 | 1B | A-Pole | Typ zařízení | 1Bh | 1B | Zařízení je pokojové čidlo |
| 24 | 7A | Ci-Pole | Odpověď od zařízení | 7Ah | 7A | jedná se o M-Bus |
| 26 | 62 | AccNo | Číslo přístupu | 41h | 214 | 214. přístup |
| 28 | 08 | Status | Status zařízení | 00h | 8 | |
| 30 | 0000 | Config. word | Šifrování AES | 0000h | | Telegram není šifrován |
| 34 | 2F2F | AES encr. | Ověření AES | 2F2Fh | | |
| 40 | 66 | DR1 | VIF: první měřená veličina | 66h | 66 | Teplota v °C ⁻¹ |
| 42 | 99 | DR1 | hodnota teploty | 99h | 0199 | Teplota je 19.9°C |
| 44 | 01 | DR1 | hodnota teploty | 01h | | |
| 50 | 1A | DR2 | VIFE: druhá měřená veličina | 1Ah | 1A | Relativní vlhkost v %-1 |
| 52 | 93 | DR2 | hodnota vlhkosti | 93h | 0293 | Vlhkost je 29.3 % |
| 54 | 02 | DR2 | hodnota vlhkosti | 02h | | |
| 62 | 1D | DR3 | VIFE1: Norma | 1Dh | | |
| 64 | 01 | DR3 | Příznak sabotáže | 00h | 1 | Čidlo bylo otevřeno |
| 66 | 00 | DR3 | Příznak vybité baterie | 00h | 0 | Baterie je nabítá |
| 94 | 87 | CRC | Kontrolní součet | 87h | | |
| 96 | 9e | RSSI | Síla přijímaného signálu | 9Eh | 158 | Síla signálu je -51dBm |

Z tabulky je patrné, že je nutné vyparovat položky na následujících pozicích:

- 8-23 pro informace o daném čidlu,
- 24-25 pro určení pořadí telegramu,
- 42-45 pro hodnotu naměřené teploty,
- 52-55 pro hodnotu naměřené vlhkosti,
- 64-67 pro kontrolu stavu čidla a
- 96 pro úroveň signálu.

Jejich následnou správnou interpretací dle specifikace (zohlednění uložení LSB, převod hexadecimálních hodnot na dekadické, ...) předat k dalšímu zpracování či uložení do databáze.

7.7 Zajištění uložení dat

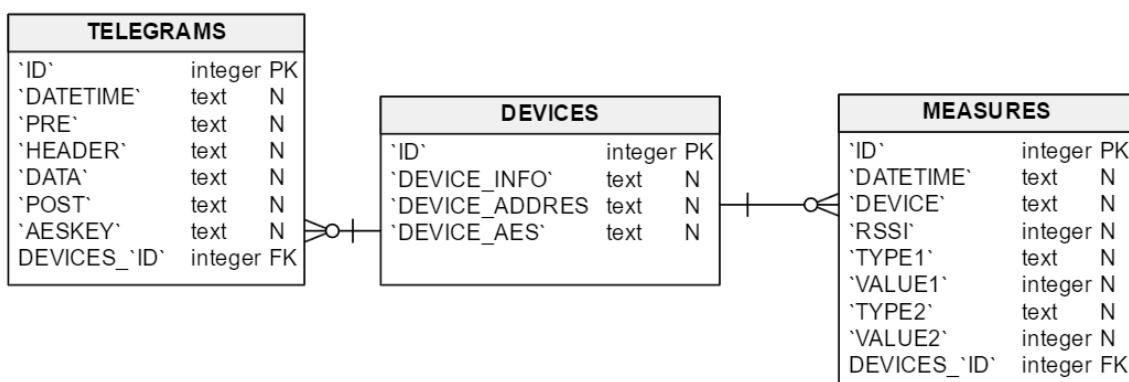
Zachycená a naměřená data se ukládají do databáze k pozdějšímu zpracování či vizualizaci. Zvolena byla databáze SQLite3 pro svoji jednoduchost, nenáročnost na systémové prostředky a možnosti instalace z repozitáře Raspbianu:

```
sudo apt-get update
sudo apt-get install sqlite3
```

Byla zvolena jedna databáze se třemi tabulkami:

- DEVICES - evidence známých zařízení a jejich AES klíčů.
- VALUES - uložení naměřených hodnot.
- TELEGRAMS - uložení zachycených dat a AES klíče modulu.

Strukturu tabulek, definici sloupců a vazby mezi nimi popisuje schéma na Obr. 7.1.



Obr. 7.1: Model zvolené SQLite 3 databáze

Prohlížení obsahu databáze je součástí vizualizační aplikace pod záložkou **Database explorer**.

7.8 Zajištění vizualizace dat

Zachycená a uložená data lze vykreslovat do grafů. Zvoleno bylo Google Charts API [3] běžící na webovém serveru Apache2 a generovaném pomocí PHP. Pro tyto potřeby je nutné doinstalovat následující balíčky:

```
sudo apt-get update  
sudo apt-get install apache2  
sudo apt-get install php7.0
```

Dále je nezbytné do umístění `\var\www\html\` nahradit zdrojové soubory vizualizační aplikace.

Prohlížení vizualizovaných dat se ve visualizační aplikaci nachází pod záložkou **Graphs Explorer**.

7.9 Struktura aplikace

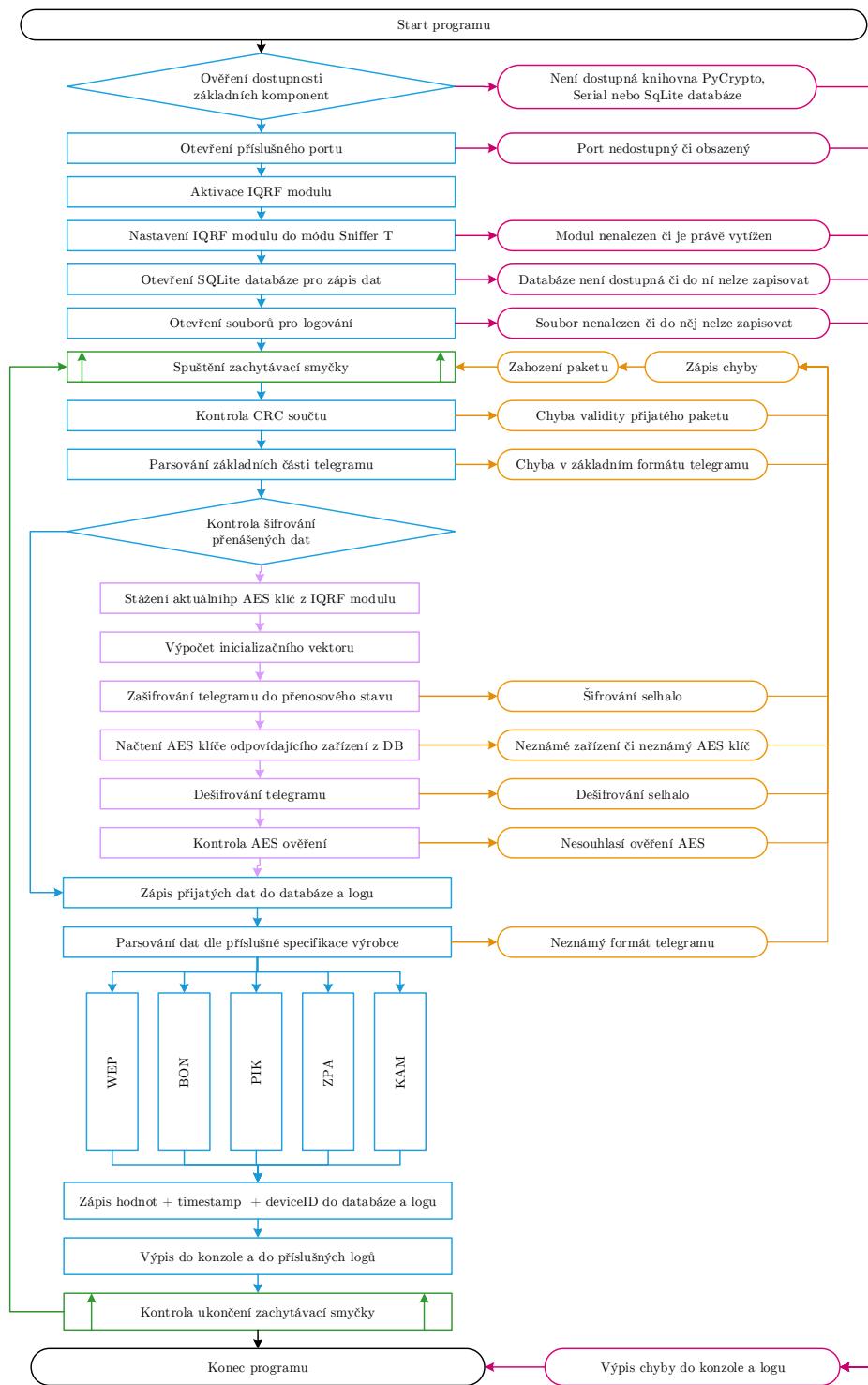
Vzhledem k výše uvedeným požadavkům a technologiím byla zvolena struktura aplikace znázorněná na Obr. 7.2. Diagram je pro přehlednost odlišen barevnými bloky:

- modrou barvou je znázorněna kostra programu,
- zelenou barvou je nekonečná smyčka naslouchání dat,
- růžovou barvou je případné dešifrování přenášených dat,
- červenou barvou jsou chyby znemožňující běh programu,
- oranžovou barvou jsou chyby znemožňující platnou analýzu či dešifrování daného telegramu a
- černou barvu je řízení samotného programu.

V následujících podkapitolách budou jednotlivé bloky aplikace představeny podrobněji.

7.9.1 Start programu v rámci operačního systému

Program je nyní spouštěn automaticky po startu operačního systému interpretem jazyka Python v příkazu screen. Tímto dojde k oddělení běhu programu od ostatních aplikací, možností vzdáleného připojení ke konzolovým výstupům aplikace a nezávislosti na případných restartech zařízení. Ukončení programu nastává pouze násilným ukončením aplikace, restartem zařízení nebo závažnou chybou při startu programu.



Obr. 7.2: Vývojový diagram aplikace pro vyčítání dat

7.9.2 Start programu z pohledu aplikace

Program při startu kontroluje, zdali má k dispozici všechny potřebné komponenty pro svůj běh. Program je závislý na knihovně PyCrypto, Serial nebo SQLite databázi. Dále program kontroluje přítomnost a možnost otevření sériového portu. V případě úspěšného otevření portu je na něj zaslán příznak pro probuzení komunikačního modulu z úsporného režimu. Po probuzení následuje příkaz, který nastaví modul do režimu skeneru v komunikačním módu T1. Pokud některá z operací selže, je zaznamenán chybový stav a dojde k ukončení programu.

7.9.3 Základní kontrola a cyklus příjmu dat

Následně dojde ke spuštění smyčky naslouchání příchozích telegramů. Každý příchozí telegram je podroben sérii kontrol, pro ověření správnosti příjmu. Jako první je telegram podroben kontrole délky odpovídající sudým násobkům. Následně je telegram podroben základní analýze položek aplikační a linkové vrstvy. Dochází ke kontrole délky telegramu a jeho CRC součtu. Je provedena kontrola obsahu CI-pole, Status pole a ConfigurationWord pole. Pokud některá z položek neodpovídá, telegram je zaevidován a aplikace se vrátí do stavu čekání na příchod dalšího telegramu. Na základě obsahu položky ConfigurationWord (viz Kap. 5.5.5) se program nadále větví na zpracování šifrovaného (Kap. 7.9.4) a nešifrovaného (Kap. 7.9.5) telegramu.

7.9.4 Dešifrování dat

Pokud je analýzou položky ConfigurationWord zjištěno šifrování dat přenášených dat algoritmem AES128 CBC, je zahájen proces dešifrování dat.

Díky vnitřní implementaci IQRF modulu jsou všechny zachycené šifrované pakety automaticky rozšifrovány pomocí AES klíče nahraného v paměti modulu. Jedná se však o klíč daného modulu, nikoliv vyčítaného zařízení. Tato dešifrovaná data jsou tedy nevalidní. Celý postup dešifrování (viz Kód 7.1) se tak komplikuje:

1. Dojde ke stažení aktuálního AES klíče (viz Kap. 4.3) nastaveného v IQRF modulu, s pomocí něj a sestaveného inicializačního vektoru (viz Kap. 5.5.3) jsou přijatá data zašifrována zpět do šifrovaného stavu jaký mají při přenosu, tedy data vyslaná daným měřícím zařízením, zašifrovaná AES klíčem daného měřicího zařízení.
2. Pokud existuje, tak je z databáze načten odpovídající klíč příslušného měřícího zařízení a s pomocí již sestaveného inicializačního vektoru jsou data rozšifrována.
3. Správnost rozšifrování se ověřuje pomocí kontrolních bajtů, v případě AES šifrování mají první dva bajty rozšifrovaných dat hodnotu 2Fh.

```

from Crypto.Cipher import AES
1
2
3
4
5
6
7
8
9

# Encrypt telegram with AES key from IQRF module
# Encrypt telegram with AES key from IQRF module
encrytor_back = AES.new(AES_KEY_IQRF, AES.MODE_CBC, IV=AES_IV)
TELEGRAM_CRYPTED = encrytor_back.encrypt(TELEGRAM_DECRYPTED)

# Decrypt telegram with AES key of used device
# Decrypt telegram with AES key of used device
encrytor_new = AES.new(AES_KEY_DEVICE, AES.MODE_CBC, IV=AES_IV)
TELEGRAM_ORIGINAL = encrytor_new.decrypt(TELEGRAM_CRYPTED)

```

Kód 7.1: Implementace AES dešifrování

Poté program s telegramem zachází jako s nešifrovaným, popsaným v Kap. 7.9.5. Pokud není nalezen AES klíč odpovídajícího zařízení, pole ConfigurationWord obsahuje neimplementovaný algoritmus dešifrování či se proces dešifrování nezdaří, program vyvolá výjimku, telegram je zaevidován a pokračuje se zpracováváním dalšího telegramu.

7.9.5 Parsování dat

Každý telegram je před svým zpracováním uložen do databáze, aby v případě potřeby (špatné dešifrování, neadekvátní dešifrovací klíč, chyba aplikace, neznámý senzor, ...) mohlo být provedeno nové zpracování daného telegramu. Poté je analyzováno M-Pole a v případě že se jedná o výrobce, jehož parsovací schéma je v této aplikaci implementováno, dochází k vyparování (viz Kód 7.2) přenášených hodnot. Následně v souladu s parsovacím schématem dochází k formátování a odpovídající interpretaci získaných dat (viz Kap. 7.1). Poté jsou získaná data uložena do databáze a zobrazena uživateli na výstup konzole.

```

# Parse values from Weptech
if (sensor_manu == "WEP"):
    if parsedstring[66:68] == "01": errors = "Battery dead"
        temperature = str(int(LSB(parsedstring[42:46])) / 10)
        humidity = str(int(LSB(parsedstring[52:56])) / 10)
# Parse values from Bonega
elif (sensor_manu == "BON"):
    Spotreba = str(int(LSB(parsedstring[42:46]), 16))
    Cascteni = get_date(LSB(parsedstring[58:62])) + " " +
                get_time(LSB(parsedstring[54:58]))

```

Kód 7.2: Ukázka parsování dat

Pokud parsování dat selže, dojde k výjimce, daný telegram je zaevidován, v jeho zpracování se již nepokračuje a aplikace se vrátí do stavu čekání na příchod dalšího telegramu.

7.9.6 Uložení dat

Správně interpretované hodnoty doplněné o časovou značku a informaci o měřícím zařízení jsou zapsány do databáze. Do této databáze se ukládají všechny příchozí validně zpracované telegramy, určené k pozdějšímu zpracování. Obohacení o časovou značku a údaje o zařízení je možné později tyto data efektivně třídit či v nich vyhledávat. Ukázka implementace je v Kódu 7.3.

```
db = sqlite3.connect(SQLITE_DB_NAME)
1
def sql(query):
2     db.execute(query)
3     db.commit()
4     output('SQL', query)
5     return
6
7
8
...
sql("INSERT INTO TELEGRAMS VALUES ('"+time.strftime("%Y-%m-%d
%H:%M")+"', '"+parsedstring[0:4]+"', '"+parsedstring[4:34]+",
'+parsedstring[34:-4]+',
'+parsedstring[-4:]+', '+AES_IQRF_DEFAULT+')")
```

Kód 7.3: Ukázka ukládání dat

Prohlížení obsahu databáze je součástí vizualizační aplikace pod záložkou **Database explorer**.

7.9.7 Ošetření výjimek

Aplikace má ošetřeny dva chybové stavy. Hrubé chyby vedoucí k násilnému ukončení programu a lehké chyby znemožňující analýzu konkrétního telegramu. Hrubé chyby mohou vyvolat chybějící komponenty (databáze, komunikační modul, sériový port, ...), zatímco lehké chyby nastanou v případě, že daný telegram nemůže být zpracován (nevalidní příjem telegramu, neznámá struktura telegramu, neznámé zařízení, neznámý výrobce zařízení, neplatný šifrovací klíč, neplatné parsovací schéma, ...), program vyvolá danou výjimku, zanechá zpracovávání aktuálního telegramu a následně přejde do stavu čekání na příchod dalšího telegramu.

Všechny tyto chyby jsou ukládány do příslušné tabulky v databázi, do chybového logu a taktéž sdělovány uživateli do konzole. Prohlížení obsahu chybových logů je součástí vizualizační aplikace pod záložkou **Logs explorer**.

7.10 Export dat

V případě potřeby je vizualizační aplikace (viz Kód 7.4) schopna dávkově vyexportovat uložená data určitého senzoru za daný časový úsek do Google Spreadsheets.

```
from oauth2client.service_account import ServiceAccountCredentials  
1  
  
scope = [ 'https://spreadsheets.google.com/feeds' ,  
2 'https://www.googleapis.com/auth/drive' ]  
3  
creds =  
4     ServiceAccountCredentials.from_json_keyfile_name('Account.json' ,  
      scope)  
client = gspread.authorize(creds)  
5  
...  
6  
DOKUMENT = client.create(dokument_name)  
7  
LIST = DOKUMENT.add_worksheet(list_name , 1 , 3)  
8  
...  
9  
for value in values:  
10    LIST.append_row([value[ "DATETIME" ] , value[ "VAL1" ] , value[ "VAL2" ]])  
11
```

Kód 7.4: Ukázka exportu dat

Vzhledem k časové náročnosti (dáno odezvou Google Charts API [3]) při daném počtu naměřených hodnot (pro čidlo vysílající každou minutu je za den nasbíráno 1440 hodnot, jejichž export zabere cca 3 hodiny) je tato funkctionalita implementována spíše jako experimentální pro jednorázové exporty. V praxi se při automatických exportech velkých objemů naměřených hodnot projevila jako náročná na výpočetní čas.

7.11 Vizualizace dat

Nad těmito daty se potom provádí vizualizace pomocí Google Charts API (viz. Kód 7.5).

```
//Load GoogleAPI and corechart package  
1  
google.charts.load('current' , { 'packages' :[ 'corechart' ]});  
2  
google.charts.setOnLoadCallback(drawChart);  
3  
  
// Create and draw our chart  
4  
var chart = new  
5     google.visualization.AreaChart(document.getElementById('chart_div'));  
6  
chart.draw(data , options);  
7
```

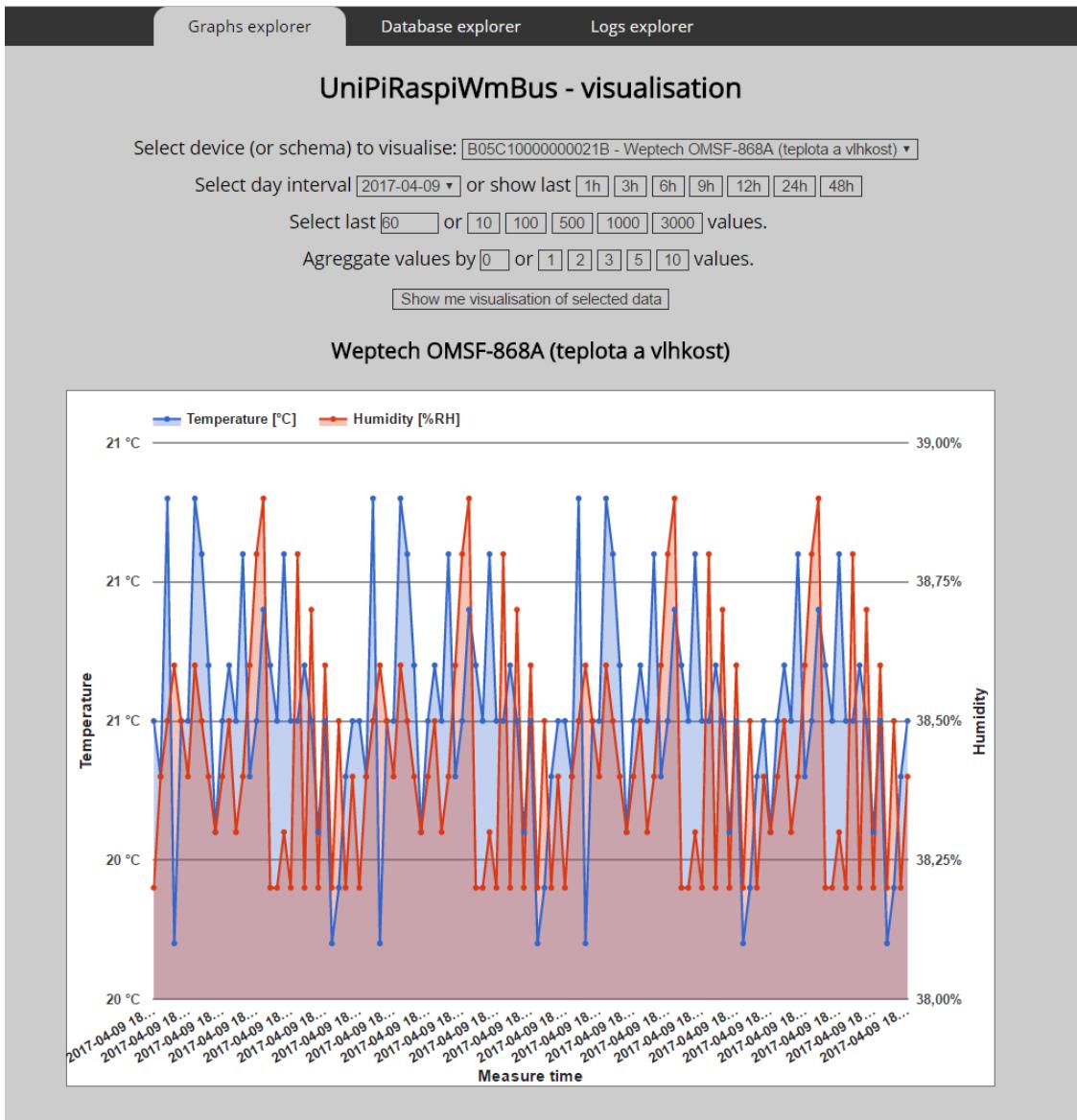
Kód 7.5: Ukázka vizualizace dat

To umožňuje interaktivní vykreslení a vyčítání uložených hodnot zachycených daným senzorem za vybraný časový úsek. Parametry vykreslení definuje funkce pro vykreslení grafu, zobrazená v Kódu 7.6.

```
// Define values and options of the graph
function drawChart() {
    var data = google.visualization.arrayToDataTable([
        ['Timestamp', 'Temperature (C)', 'Humidity (%RH)'],
        ['2017-04-09 18:10', 20.6, 35.8],
        ['2017-04-09 18:11', 20.5, 36.0],
        ['2017-04-09 18:12', 20.8, 35.9],
        ['2017-04-09 18:13', 20.9, 36.1],
        ['2017-04-09 18:14', 20.6, 36.0],
        ['2017-04-09 18:15', 20.7, 36.2],
        ['2017-04-09 18:16', 20.6, 35.9],
        ['2017-04-09 18:17', 20.6, 36.0],
        ['2017-04-09 18:18', 20.7, 36.2],
        ['2017-04-09 18:19', 20.6, 35.9]]);
    var options = {
        enableInteractivity: true,
        legend: { position: 'top', textStyle: { fontSize: 12, bold: true } },
        hAxis: { title: 'Measure time', format: 'mm-dd HH:MM',
            textStyle: { fontSize: 12, bold: true }, titleTextStyle: { fontSize: 14, bold: true, italic: false } },
        vAxes: {
            0: { title: 'Temperature', format: '###C', textStyle: { fontSize: 12, bold: true },
                titleTextStyle: { fontSize: 14, bold: true, italic: false }, gridlines: { color: 'gray' } },
            1: { title: 'Humidity', format: '###%', textStyle: { fontSize: 12, bold: true },
                titleTextStyle: { fontSize: 14, bold: true, italic: false }, gridlines: { color: 'gray' } }
        },
        series: {0: {pointShape: 'circle', targetAxisIndex:0}, 1: {pointShape: 'circle', targetAxisIndex:1}, 2: {pointShape: 'circle', targetAxisIndex:0}}
    };
}
```

Kód 7.6: Ukázka parametrizace vizualizovaných dat

Snímek obrazovky vizualizační aplikace je uveden na Obr. 7.3.



Obr. 7.3: Snímek obrazovky vizualizační aplikace

Záhlaví stránky obsahuje rozcestník:

- Graphs explorer - Samotná vizualizační aplikace, popsaná v Kap. 7.8 a 7.11.
- Database explorer - Prohlížeč obsahu databáze, popsaný v Kap. 7.7 a 7.9.6.
- Logs explorer - Prohlížeč průběžných i chybových logů, zmíněný v Kap. 7.9.7.

Stránka je rozdělena do dvou částí. V první části stránky se nachází formulář s výběrem čidla, intervalu pro zobrazení a dalších možných parametrů, pomocí kterých lze filtrovat data k vizualizaci. Pod formulářem se nachází již vykreslený graf.

Ukázky vizualizací zachycených dat jednotlivých zařízení jsou uvedeny v Příloze C.

8 ZÁVĚR

V této diplomové práci byla popsána problematika M2M (Machine-to-Machine) komunikace pomocí protokolu Wireless M-bus a její implementace do produktu UniPi NEURON.

V první části práce (Kap. 1) byla popsána M2M komunikace z pohledu spotřebitelského a průmyslového Internetu věcí.

Druhá část (Kap. 2) se zabývá embedded zařízeními pro IoT (Internet of Things), přináší přehled nejznámějších z nich, popisuje jejich možnosti, uvádí možnosti připojení senzorů a zmiňuje nedostatky zařízení. Zařízení RaspberryPi je následně použité k samotné implementaci v praktické části. Jsou zde popsány předchozí verze, důvod výběru konkrétního modelu, design i kroky potřebné k implementaci.

Třetí část (Kap. 3) obsahuje popis rozšiřující desky UniPi a zařízení UniPi NEURON. Popisuje blíže parametry obou zařízení, možnosti jejich konektivity a softwarového vybavení. Zařízení bylo vyvinuto primárně jako rozhraní pro příjem vstupních signálů, jejich využití a realizaci výstupní reakce na základě naprogramovaných algoritmů. Je vhodné pro monitorování, sběr a ukládání dat na vzdálený server, nebo jako výkonná a plně vybavená brána pro ostatní zařízení.

Čtvrtá část (Kap. 4) se zabývá Wireless M-Bus modulem TR-72D-WMB výrobce IQRF, komunikující přes sběrnici UART, a popisuje strukturu příkazů a formát dat pro komunikaci s tímto modulem.

Pátá část (Kap. 5) se zaměřila na protokol Wireless M-Bus, konkrétně na princip komunikace, režimy přenosu a jednotlivé vrstvy. Díky nutnosti znalosti fyzické a linkové vrstvy pro pozdější analýzu zachytávaných dat byly tyto vrstvy rozebrány podrobněji.

V šesté části (Kap. 6) byly popsány vyčítaná zařízení a struktura dat jejich telegramů.

Závěrečná (Kap. 7) část obsahuje návrh a samotnou implementaci vzorové aplikace pro vyčítání dat. Jsou popsány jednotlivé kroky nutné ke zprovoznění komunikace mezi RaspberryPi a vyčítaným senzorem, provedeno zachycení vzorového telegramu, jeho analýza a následné předání zvolených informací. Z výstupu aplikace je patrné, že pakety obsahují příslušná data, komunikace mezi modulem a zařízením funguje, data ze senzoru se přenášejí, následně vyčítají a přehledně zobrazují v implementované vizualizaci pomocí Google Charts API.

Na zakladě této realizace je vytvořená aplikace schopna analyzovat jen předem známá zařízení. Pokud by řešení mělo být plně využitelné pro Internet věcí, bylo by nutné implementovat velké množství parsovacích schémat, nebo vytvořit obecný algoritmus pro parsování datových jednotek všech výrobců dodržující platnou specifikaci.

LITERATURA

- [1] Národní iniciativa Průmysl 4.0 *Ministerstvo průmyslu a obchodu*[online]. 2015 [cit. 2016-10-15]. Dostupné z: <http://www.spcr.cz/images/priloha001-2.pdf>
- [2] KORBEL, Petr. *Průmyslová revoluce 4.0: Za 10 let se továrny budou řídit samy a produktivita vzroste o třetinu*[online]. 2016 [cit. 2016-10-15]. Dostupné z: <http://byznys.ihned.cz/c1-64009970-prumyslova-revoluce-4-0-za-10-let-se-tovarny-budou-ridit-samy-a-produktivita-vzroste-o-tretinu>
- [3] Google Developers *Google Chart API* [online]. 2015 [cit. 2017-03-29]. Dostupné z: <https://developers.google.com/chart/>
- [4] VITEK, Jan. *Internet of Things: propojená budoucnost*[online]. 2016 [cit. 2016-10-15]. Dostupné z: <http://www.svethardware.cz/internet-of-things-propojena-budoucnost/39560>
- [5] POHANKA, Pavel. *Internet věcí*[online]. 2016 [cit. 2016-10-15]. Dostupné z: <http://i2ot.eu/internet-of-things/>
- [6] BARRAGAN, Hernando *About Wiring* [online]. 2016 [cit. 2016-10-15]. Dostupné z: <http://wiring.org.co/about.html>
- [7] Processing. *Arduino.cz* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://arduino.cz/processing/>
- [8] ArduinoBoard Duemilanove. *Arduino* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardDuemilanove>
- [9] ArduinoBoard Uno. *Arduino* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardUno>
- [10] ArduinoBoard Leonardo. *Arduino* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardLeonardo>
- [11] ArduinoBoard Mega. *Arduino* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardMega>
- [12] ArduinoBoard Due. *Arduino* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardDue>
- [13] ArduinoBoard Mini. *Arduino* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardMini>

- [14] ArduinoBoard Micro. *Arduino* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardMicro>
- [15] ArduinoBoard Nano. *Arduino* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardNano>
- [16] ArduinoBoard Fio. *Arduino* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardFio>
- [17] ArduinoBoard Lilypad. *Arduino* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardLilypad>
- [18] ArduinoBoard MKR1000. *Arduino* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoMKR1000>
- [19] ArduinoBoard Yun. *Arduino* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardYun>
- [20] Arduino and Arduino-Compatible Hardware. *Arduino Playground* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://playground.arduino.cc/main/similarBoards>
- [21] Teensy USB Development Board. *PJRC* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://www.pjrc.com/teensy/>
- [22] RaspberryPi products. *Raspberry Pi* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.raspberrypi.org/products/>
- [23] Gertboard for Raspberry Pi *ELEMENT14 Community* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.element14.com/community/docs/DOC-69381/1/gertboard-for-raspberry-pi>
- [24] RaspberryPi Model A+. *Raspberry Pi* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.raspberrypi.org/products/model-a-plus/>
- [25] RaspberryPi 2 model B. *Raspberry Pi* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>
- [26] RaspberryPi 3 Model B. *Raspberry Pi* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>
- [27] RaspberryPi Zero. *Raspberry Pi* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.raspberrypi.org/products/pi-zero/>

- [28] Open Source Hardware Products *Banana Pi Products* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://www.banana-pi.org/product.html>
- [29] What's Orange Pi Plus? *OrangePi Community* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://www.orangepi.org/>
- [30] A series of open source hardware *CubieBoard* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://cubieboard.org/model/>
- [31] UpBoard - Specifications *UP - bidge the gap* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://www.up-board.org/up/specifications/>
- [32] SoC and Memory Specification *PINE64* [online]. 2016 [cit. 2016-12-14]. Dostupné z: http://wiki.pine64.org/index.php/Main_Page#SoC_and_Memory_Specification
- [33] ODROID Platforms *HardKernel - Products* [online]. 2016 [cit. 2016-12-14]. Dostupné z: http://www.hardkernel.com/main/products/prdt_info.php
- [34] BeagleBone Black *BeagleBoard.org Foundation* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://beagleboard.org/black>
- [35] Intel Galileo. *IoT Hardware Share* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://www.intel.com/content/www/us/en/embedded/products/galileo/galileo-overview.html>
- [36] Arduino Galileo. *Arduino* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.arduino.cc/en/ArduinoCertified/IntelGalileo>
- [37] Intel Edison. *IoT Hardware Share* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://www.intel.com/content/www/us/en/do-it-yourself/edison.html>
- [38] GIZMO 1 *GizmoSphere - Development unleashed* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://www.gizmosphere.org/products/gizmo-explorer-kit/>
- [39] GIZMO 2 *GizmoSphere - Development unleashed* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://www.gizmosphere.org/products/gizmo-2/>
- [40] UniPi.technology *UniPi.technology* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://unipi.technology/>
- [41] ZELENÁ DATA - Datacentrum s inteligentní energií společnosti Faster CZ. *ZELENÁ DATA - DATACENTRUM FASTER* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://zelenadata.cz/cs/>

- [42] UniPi board *UniPi.technology* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://unipi.technology/product/unipi/>
- [43] UniPi Neuron S103 *UniPi.technology* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://unipi.technology/product/unipi-neuron-s103/>
- [44] Sběrnice USART *Wikipedie - otevřená encyklopédie* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://cs.wikipedia.org/wiki/USART>
- [45] PiBrella compatibility *ModMyPi* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://forum.modmypi.com/technical-support/pibrella-compatibility-t181.html>
- [46] TR-72D-WMB series *IQRF - Technology for Wireless* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://www.iqrf.org/products/transceivers/tr-72d-wmb>
- [47] MERVIS *UniPi.technology* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://unipi.cz/software/mervis/>
- [48] Wireless Meter Bus, WM-Bus Technology *Radio-Electronics.com - Adrio Communications Ltd* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <http://www.radio-electronics.com/info/wireless/wireless-mbus/basics-tutorial.php>
- [49] EN 13757-1. *Communication system for and remote reading of meters - Part 1: Data exchange*. Wien: Austrian Standards Institute. [online] 2016 [cit. 2016-12-14] Dostupné z: <https://shop.austrian-standards.at/Preview.action;jsessionid=4B46107107AC62A5CB24E33F6A51A5E4?preview=&dokkey=467673&selectedLocale=en>
- [50] FLAG - Registered Manufacturers Identification Characters *FLAG Association Limited* [online]. 2008 [cit. 2016-12-14]. Dostupné z: <http://www.dlms.com/flag/INDEX.HTM>
- [51] Sběrnice Wireless M-BUS - jde to i bezdrátově... *Automatizace.hw.cz* [online]. 2010 [cit. 2014-10-28]. Dostupné z: <http://automatizace.hw.cz/sbernice-wireless-mbus-jde-i-bezdratove>
- [52] EN 13757-4. *Communication systems for meters and remote reading of meters - Part 4: Wireless meter readout (Radio Meter reading for operation in the 868-870 MHz SRD band)*. [online] 2016 [cit. 2016-12-14] Brusel: EUROPEAN COMMITTEE FOR STANDARDIZATION, 2003. Dostupné z: <http://oldfjarrvarme.unc.se/download/1309/fj>

- [53] Fairhurst, Garry. *Manchester Encoding* [online]. 2007 [cit. 2017-03-29]. Dostupné z: <http://www.erg.abdn.ac.uk/users/gorry/course/phy-pages/man.html>
- [54] SILICON LABS. WIRELESS M-BUS SOFTWARE IMPLEMENTATION. 2010, 14 s. Dostupné z: <https://www.silabs.com/Support%20Documents/TechnicalDocs/AN451.pdf>
- [55] M-Bus Usergroup *The M-Bus: A Documentation* [online]. 1998 [cit. 2017-03-29]. Dostupné z: <http://www.m-bus.com/mbusdoc/mb8.php>
- [56] National Institute of Standards and Technology *DATA ENCRYPTION STANDARD (DES)* [online]. 2001 [cit. 2017-03-29]. Dostupné z: <http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf>
- [57] National Institute of Standards and Technology *ADVANCED ENCRYPTION STANDARD (AES)* [online]. 2001 [cit. 2017-03-29]. Dostupné z: <http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf>
- [58] Wireless M-Bus / OMS Humidity and temperature sensor WEP-OMSF-868A *WEPTECH elektronik GmbH* [online]. 2016 [cit. 2016-12-14]. Dostupné z: <https://www.weptech.de/products/oms-humidity-and-temperature-sensor-wep-omsf-868a.html>
- [59] Ultra-antimagnetické bytové vodoměry s bezdrátovým přenosem dat *Vodoměry BONEGA* [online]. 201 [cit. 2017-2-20]. Dostupné z: <http://www.bonega.cz/vodomery/index.htm>
- [60] Nejflexibilnejší meric na trhu MULTICAL 403 *Kamstrup ČR* [online]. 2016 [cit. 2017-03-11]. Dostupné z: <https://www.kamstrup.com/cs-cz/products-and-solutions/thermal-energy-meters/multical-403>
- [61] Trifazovy elektromer ZE310 *ZPA Smart Energy a.s.* [online]. 2016 [cit. 2017-03-11]. Dostupné z: <https://www.zpa.cz/produkty-a-reseni/elektromery:c1/ze-310:p4.htm>
- [62] The Open Metering System specification *OMS-Group* [online]. 2016 [cit. 2017-03-11]. Dostupné z: <http://oms-group.org/en/oms-group/about-oms-group/>

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

| | |
|------|-----------------------------------|
| ACK | Acknowledgement |
| AES | Advanced Encryption Standard |
| AMR | Automated Meter Reading |
| API | Application Programming Interface |
| APU | Accelerated Processing Unit |
| ARM | Advanced (Acorn) RISC Machine |
| AVR | Alf & Vegard Risc procesor |
| bd | baud |
| B | Byte |
| CAN | Controller Area Network |
| CBC | Cipher Block Chaining |
| cIoT | Customer IoT |
| CPU | Central Processing Unit |
| CR | Carriage Return |
| CRC | Cyclic Redundancy Check |
| CSI | Camera Serial Interface |
| CVBS | Color Video Blank Sync |
| dBm | decibel on milliwatt |
| DES | Data Encryption Standard |
| DIB | Data Information Block |
| DIF | Data Information Field |
| DIFE | Data Information Field Extended |
| DIP | Dual Inline Package |
| DPA | Direct Peripheral Access |

| | |
|--------|---|
| DSI | Display Serial Interface |
| EEPROM | Electrically Erasable Programmable Read-Only Memory |
| eMMC | embedded MultiMedia Card |
| ESD | ElectroStatic Discharge |
| FPU | Floating-Point Unit |
| GPIO | General Purpose Input Output |
| GB | GigaByte |
| GHz | GigaHertz |
| GPU | Graphical Processing Unit |
| HDMI | High-Definition Multimedia Interface |
| HCA | Heat Cost Allocator |
| Hz | Hertz |
| H2H | Human to Human |
| ICSP | In Circuit Serial Programming |
| IDE | Integrated Development Environment |
| iIoT | Industry IoT |
| IoP | Internet of People |
| IoS | Internet of Services |
| IoT | Internet of Things |
| ISM | Industrial, Scientific and Medical |
| IV | Initialization Vector |
| I/O | Input / Output |
| I2C | Inter-Integrated Circuit |
| I2S | Inter-Integrated Sound |
| JTAG | Joint Test Action Group |

| | |
|---------|--|
| KB | KiloByte |
| KHz | KiloHertz |
| LRADC | Low Resolution Analog to Digital Converter |
| LSB | Least Significant Bit |
| LVDS | Low-Voltage Differential Signaling |
| MB | MegaByte |
| M-Bus | Meter Bus |
| MCU | Micro-Controller Unit |
| MHz | MegaHertz |
| MicroSD | Micro Secure Digital |
| MISO | Master Input Slave Output |
| MOSI | Master Output Slave Input |
| MSB | Most Significant Bit |
| MTC | Machine Type Communication |
| MTCD | Machine Type Communication Device |
| MTCG | Machine Type Communication Gateway |
| M2M | Machine to Machine |
| NRZ | Non Return to Zero |
| OMS | Open Metering Standard |
| PCI | Peripheral Component Interconnect |
| PLC | Programmable Logic Controller |
| PoE | Power over Ethernet |
| PS2 | Personal System/2 |
| PWM | Pulse Width Modulation |
| QoE | Quality of Experience |

| | |
|--------|---|
| REQ | Request |
| RSSI | Received Signal Strength Indication |
| RTC | Real Time Clock |
| RF | Radio Frequency |
| RX | Receive |
| RZ | Return to Zero |
| SATA | Serial Advanced Technology Attachment |
| SCLK | Serial Clock |
| SD | Secure Digital |
| SDIO | Serial Data Input Output |
| SoC | System on Chip |
| SRAM | Static Random Access Memory |
| SRD | Short Range Device |
| SS | Slave Select |
| TX | Transmit |
| UART | Universal Asynchronous Receiver/Transmitter |
| USB | Universal Serial Bus |
| VIB | Value Information Block |
| VIF | Value Information Field |
| VIFE | Value Information Field Extended |
| Wi-Fi | Wireless Fidelity |
| WM-Bus | Wireless Meter Bus |

SEZNAM PŘÍLOH

| | | |
|----------|---|------------|
| A | Přehled parametrů jednotlivých jednodeskových počítačů | 104 |
| B | Ukázka zachycených dat | 105 |
| C | Ukázka vizualizace dat | 106 |
| D | Obsah přiloženého DVD | 109 |

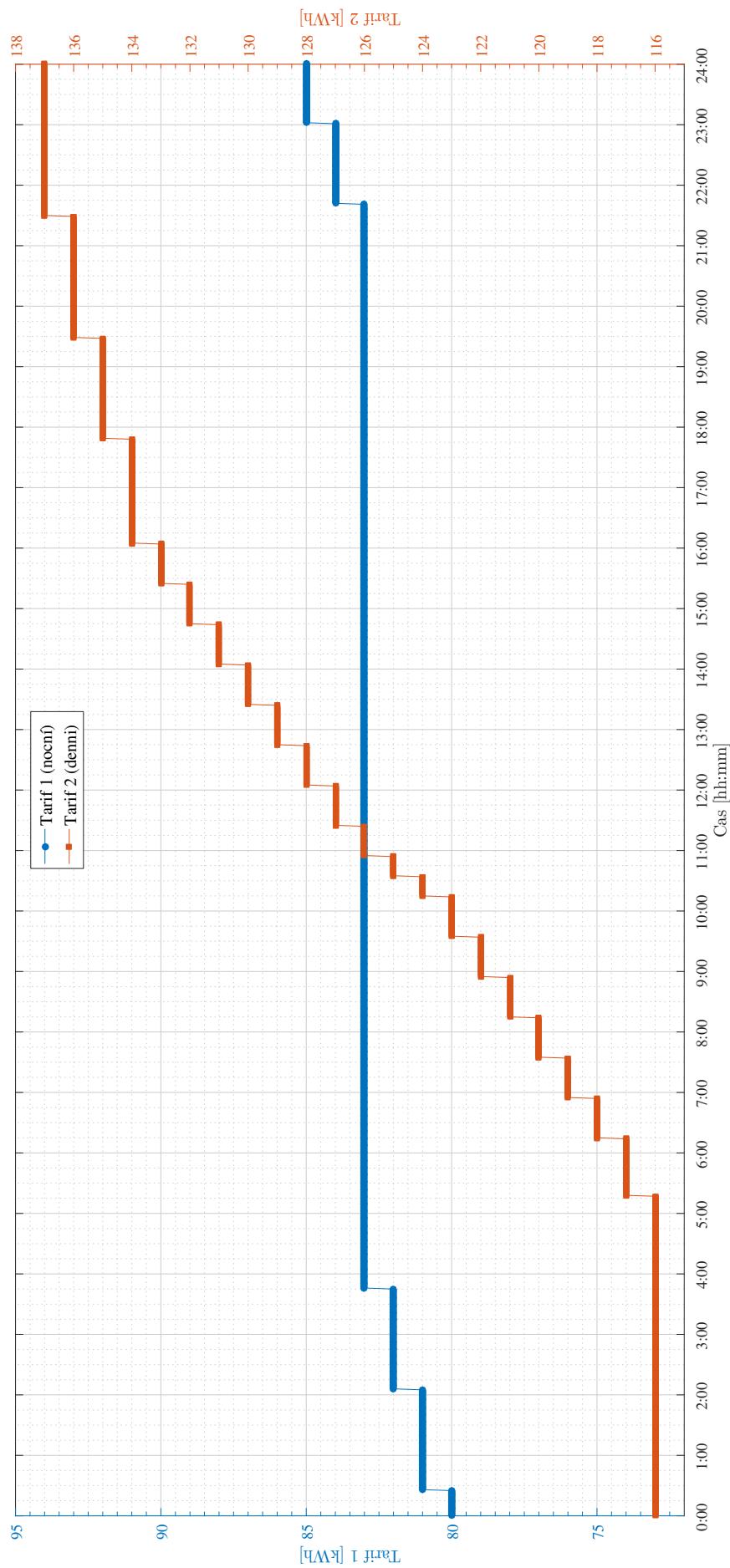
A PŘEHLED PARAMETRŮ JEDNOTLIVÝCH JEDNODĚSKOVÝCH PO-ČÍTAČŮ

| Výrobce desky | | Označení modelu | | Mikrokontroler | | Platforemá | | EEPROM [KiB] | | RAM [KiB] | | Flash [KiB] | | Překvapení | | MicroSD | | Mini PCI-e | | SATA | | PCIe | | Bluetooth | | Grafický výstup | | HDMI | | Podpora sběrnice | | USB rozhraní | | Rozmery | |
|---------------------|--------------------------|------------------|----------|----------------|-----|------------|--------|--------------|------|-----------|------|-------------|------|------------|------|---------|------|------------|------|------|------|------|------|-----------|------|-----------------|-----------------|------------------|----------------|------------------|--|--------------|--|---------|--|
| Dicimila | ATmega168 | ARM | 16 MHz | 16 | 0.5 | 1 | 14 | 6 | 6 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 68.6 x 53.3 mm | | | | | | |
| Duemilanove (v2) | ATmega328P | ARM | 16 MHz | 32 | 1 | 2 | 14 | 6 | 6 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 68.6 x 53.3 mm | | | | | | | |
| Uno (R3) | ATmega328P | ARM | 16 MHz | 32 | 1 | 2 | 14 | 6 | 6 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 68.6 x 53.3 mm | | | | | | | |
| Due | ATMEVL SAM3U | ARM | 84 MHz | 512 | 1 | 96 | 54 | 12 | 16 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | Prog + Native | 101.5 x 53.3 mm | | | | | | | | |
| Mega (2560) | ATmega2560 | ARM | 16 MHz | 256 | 4 | 8 | 54 | 14 | 16 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 101.5 x 53.3 mm | | | | | | | | |
| Leonardo | ATmega32u4 | ARM | 16 MHz | 32 | 1 | 2 | 14 | 6 | 12 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 68.6 x 53.3 mm | | | | | | | | |
| Fio | ATmega328P | ARM | 8 MHz | 32 | 1 | 2 | 14 | 6 | 8 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | UART | 40.6 x 27.9 mm | | | | | | | | |
| Mini | ATmega328P | ARM | 16 MHz | 32 | 1 | 2 | 14 | 6 | 8 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | uart | 30.5 x 18.0 mm | | | | | | | | |
| Micro | ATmega32u4 | ARM | 16 MHz | 32 | 1 | 2.5 | 20 | 7 | 12 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | uart | 50.0 x 13.0 mm | | | | | | | | |
| Nano v2 | ATmega328 | ARM | 16 MHz | 32 | 1 | 2 | 14 | 6 | 8 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 43.0 x 18.0 mm | | | | | | | | |
| LilyPad (v2) | ATmega328V | ARM | 8 MHz | 16 | 0.5 | 1 | 14 | 6 | 6 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | UART | ø 50mm | | | | | | | | |
| Yún | Atheros AR9331 | x86 | 400 MHz | 16 MB | - | 64 MB | - | 1 | 1 | 34 | 21 | 1 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | Atmega32u4 | 68.6 x 53.3 mm | | | | | | | |
| Teensy (v 3.2) | MK20DX256 | ARM | 72 MHz | 256 | 64 | 2 | 14 | 6 | 6 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 30.5 x 18.0 mm | | | | | | | | |
| Freeduino | ATmega168 | ARM | 16 MHz | 16 | 0.5 | 1 | 14 | 6 | 6 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 68.6 x 53.3 mm | | | | | | | | |
| Labduino | ATmega328P | ARM | 16 MHz | 32 | 1 | 2 | 14 | 6 | 6 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 51.0 x 51.0 mm | | | | | | | | |
| Arduino Libero | ATmega168 | ARM | 16 MHz | 16 | 0.5 | 1 | 14 | 6 | 6 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 68.6 x 53.3 mm | | | | | | | | |
| Bare Bones Board | ATmega328P | ARM | 16 MHz | 32 | 1 | 2 | 20 | 6 | 6 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 50.7 x 40.6 mm | | | | | | | | |
| Nanode | ATmega328P | ARM | 16 MHz | 32 | 1 | 2 | 14 | 6 | 6 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 68.6 x 53.3 mm | | | | | | | | |
| Freeduino | ATmega328P | ARM | 16 MHz | 32 | 1 | 2 | 14 | 6 | 6 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 68.6 x 53.3 mm | | | | | | | | |
| Seeduino | ATmega1280 | ARM | 16 MHz | 128 | 4 | 8 | 54 | 14 | 16 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 68.6 x 53.3 mm | | | | | | | | |
| Diavolino | ATmega328P | ARM | 16 MHz | 32 | 1 | 2 | 14 | 6 | 6 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 68.6 x 53.3 mm | | | | | | | | |
| Boarduino | ATmega328P | ARM | 16 MHz | 16 | 0.5 | 1 | 14 | 6 | 6 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 75.0 x 20.0 mm | | | | | | | | |
| Galleo | Intel Quark X1000 | x86 | 400 MHz | 8 MB | 8 | 256 MB | 14 | 6 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | 2 x USB | 123.8 x 72.0 mm | | | | | | | | |
| Edison | Intel Atom + Intel Quark | x86 | 500 MHz | 4 GB | 8 | 1 GB | 20 | 4 | 6 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | idle desktop | 35.5 x 25.0 mm | | | | | | | |
| AMD | Gizmo 1 | AMD GX210H A | amd64 | 1 GHz | ne | - | 1 GB | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 101.6 x 101.6 mm | | | | | | | |
| Raspberry Pi 2 | Raspberry Pi Zero | Broadcom BCM2835 | ARM | 1.1 GHz | 1 | 1 GB | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | 2x USB | 101.6 x 101.6 mm | | | | | | | |
| Raspberry Pi 3 | Raspberry Pi 2 | Broadcom BCM2837 | ARM | 1.2 GHz | ne | - | 1 GB | 13 | 1 | ne | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 87.5 x 56.5 mm | | | | | | | |
| Raspberry Pi 3 | Raspberry Pi 3 | Broadcom BCM2835 | ARM | 1.1 GHz | ne | - | 512 MB | 13 | 1 | ne | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | FTDI | 65.5 x 30.5 mm | | | | | | | |
| Banana Pi (M3) | Allwinner A83T | ARM | 1.8 GHz | 8 GB | 1 | 2 GB | 13 | 1 | ne | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | 2x USB SoC | 92.0 x 60.0 mm | | | | | | | | |
| OrangePi (+2) | Allwinner H3 | ARM | 1.6 GHz | 16 GB | - | 2 GB | 13 | 1 | ne | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | 4x USB SoC | 85.6 x 53.98 mm | | | | | | | | |
| CubieBoard (v5) | Allwinner H8 | ARM | 2 GHz | 8 GB | - | 2 GB | auto | ne | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | 3x USB SoC | 108 x 67.0 mm | | | | | | | | |
| UpBoard (v1) | Intel X5-Z8350 | ARM | 1.92 GHz | 64 GB | - | 4 GB | auto | auto | ne | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | 4x USB SoC | 110 x 80 mm | | | | | | | | |
| Raspberry klonov | HardKernel Odroid (C2) | Cortex A53 | ARM | 1.2 GHz | ne | - | 2 GB | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | 4x USB SoC | 85.6 x 56.5 mm | | | | | | | | |
| PINE64 (+2) | Amlogic S905 | ARM | 1.5 GHz | 2 GB | - | 2 GB | auto | ne | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | 4x USB SoC | 127 x 79 mm | | | | | | | | |
| BeagleBoard (Black) | Shara A3335S/9 | ARM | 1.1 GHz | 2 GB | - | 512 MB | auto | 8 | auto | 4 | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | auto | 2x USB SoC | 86.4 x 53.3 mm | | | | | | | | |

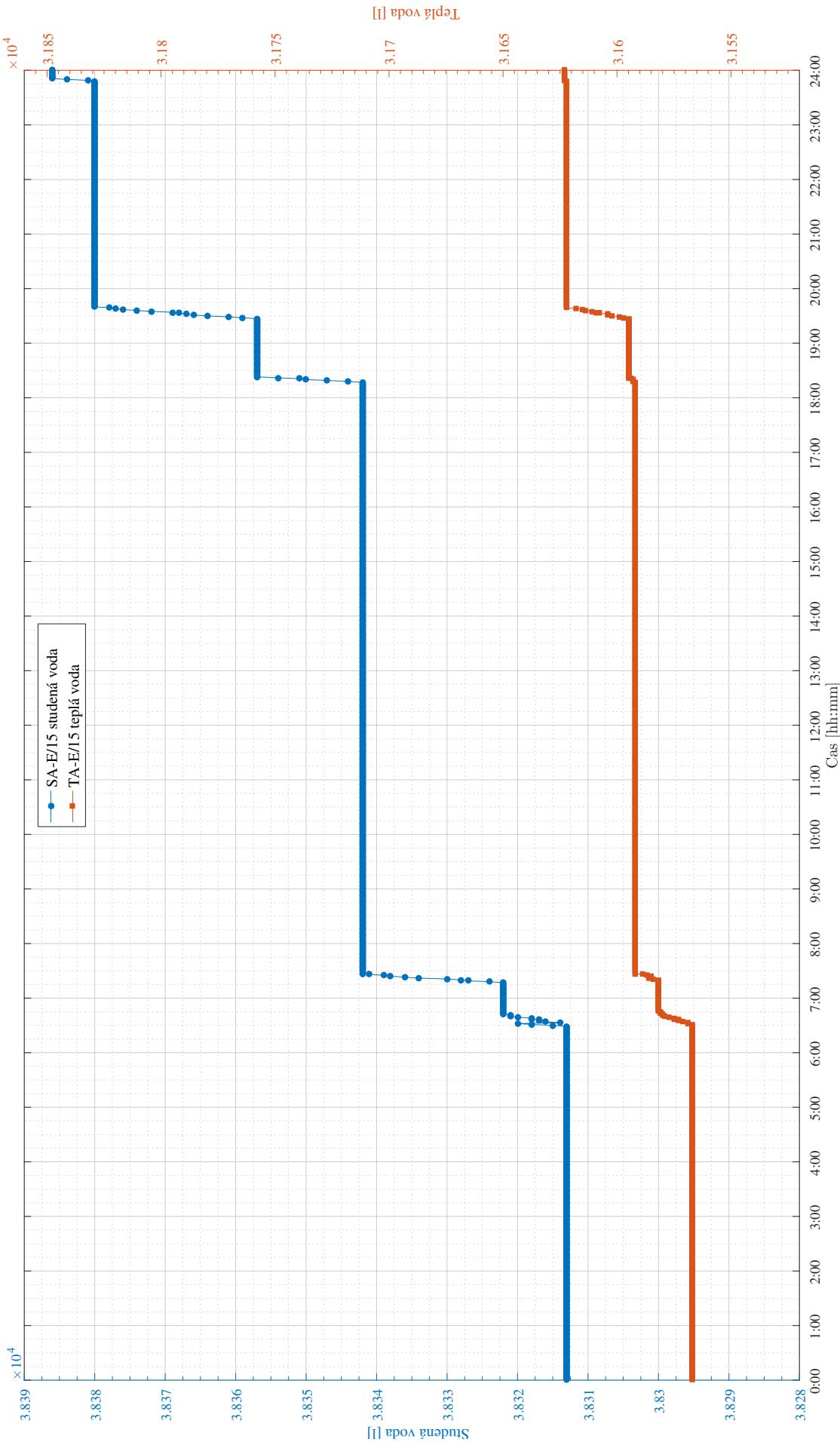
B UKÁZKA ZACHYCENÝCH DAT

| | |
|---------------------|--|
| 19/04/2017 20:47:48 | Running in : Real sniffer mode. |
| 19/04/2017 20:47:48 | Device is on AMA0: True |
| 19/04/2017 20:47:49 | Device is waked up: OK |
| 19/04/2017 20:47:50 | Device is set as Sniffer T: OK |
| 19/04/2017 20:47:51 | Default AES key set : OK |
| 19/04/2017 20:47:51 | Sniffing now: |
| 19/04/2017 20:48:14 | AccNo: 181 Device: BON.06.00000121.01 RSSI: -49.5dB AES: True Volume: 315671 |
| 19/04/2017 20:48:46 | AccNo: 182 Device: BON.07.00000121.01 RSSI: -49.5dB AES: True Volume: 286781 |
| 19/04/2017 20:48:47 | AccNo: 203 Device: WEP.1b.00000010.02 RSSI: -40.0dB AES: True Temperature: 20.4C Humidity: 35.1% |
| 19/04/2017 20:49:12 | AccNo: 71 Device: ZPA.02.01754247.01 RSSI: -74.5dB AES: False Tariff1: 3.0kWh Tariff2: 11.0kWh |
| 19/04/2017 20:49:19 | AccNo: 182 Device: BON.06.0000121.01 RSSI: -49.5dB AES: True Volume: 315671 |
| 19/04/2017 20:49:48 | AccNo: 204 Device: WEP.1b.00000010.02 RSSI: -40.0dB AES: True Temperature: 20.4C Humidity: 35.0% |
| 19/04/2017 20:49:49 | AccNo: 183 Device: BON.07.00000121.01 RSSI: -49.0dB AES: True Volume: 286781 |
| 19/04/2017 20:50:12 | AccNo: 71 Device: ZPA.02.01754247.01 RSSI: -74.0dB AES: False Tariff1: 3.0kWh Tariff2: 11.0kWh |
| 19/04/2017 20:50:18 | AccNo: 183 Device: BON.06.0000121.01 RSSI: -49.0dB AES: True Volume: 315671 |
| 19/04/2017 20:50:48 | AccNo: 205 Device: WEP.1b.00000010.02 RSSI: -40.0dB AES: True Temperature: 20.4C Humidity: 35.2% |
| 19/04/2017 20:50:50 | AccNo: 184 Device: BON.07.00000121.01 RSSI: -49.0dB AES: True Volume: 286781 |
| 19/04/2017 20:51:12 | AccNo: 71 Device: ZPA.02.01754247.01 RSSI: -74.0dB AES: False Tariff1: 3.0kWh Tariff2: 11.0kWh |
| 19/04/2017 20:51:22 | AccNo: 184 Device: BON.06.0000121.01 RSSI: -49.0dB AES: True Volume: 315671 |
| 19/04/2017 20:51:48 | AccNo: 206 Device: WEP.1b.00000010.02 RSSI: -40.0dB AES: True Temperature: 20.4C Humidity: 35.0% |
| 19/04/2017 20:51:52 | AccNo: 185 Device: BON.07.00000121.01 RSSI: -49.0dB AES: True Volume: 286781 |
| 19/04/2017 20:52:12 | AccNo: 71 Device: ZPA.02.01754247.01 RSSI: -74.0dB AES: False Tariff1: 3.0kWh Tariff2: 11.0kWh |
| 19/04/2017 20:52:24 | AccNo: 185 Device: BON.06.0000121.01 RSSI: -49.0dB AES: True Volume: 315671 |
| 19/04/2017 20:52:49 | AccNo: 207 Device: WEP.1b.00000010.02 RSSI: -40.0dB AES: True Temperature: 20.4C Humidity: 34.9% |
| 19/04/2017 20:52:52 | AccNo: 186 Device: BON.07.00000121.01 RSSI: -49.0dB AES: True Volume: 286781 |
| 19/04/2017 20:53:12 | AccNo: 71 Device: ZPA.02.01754247.01 RSSI: -74.0dB AES: False Tariff1: 3.0kWh Tariff2: 11.0kWh |

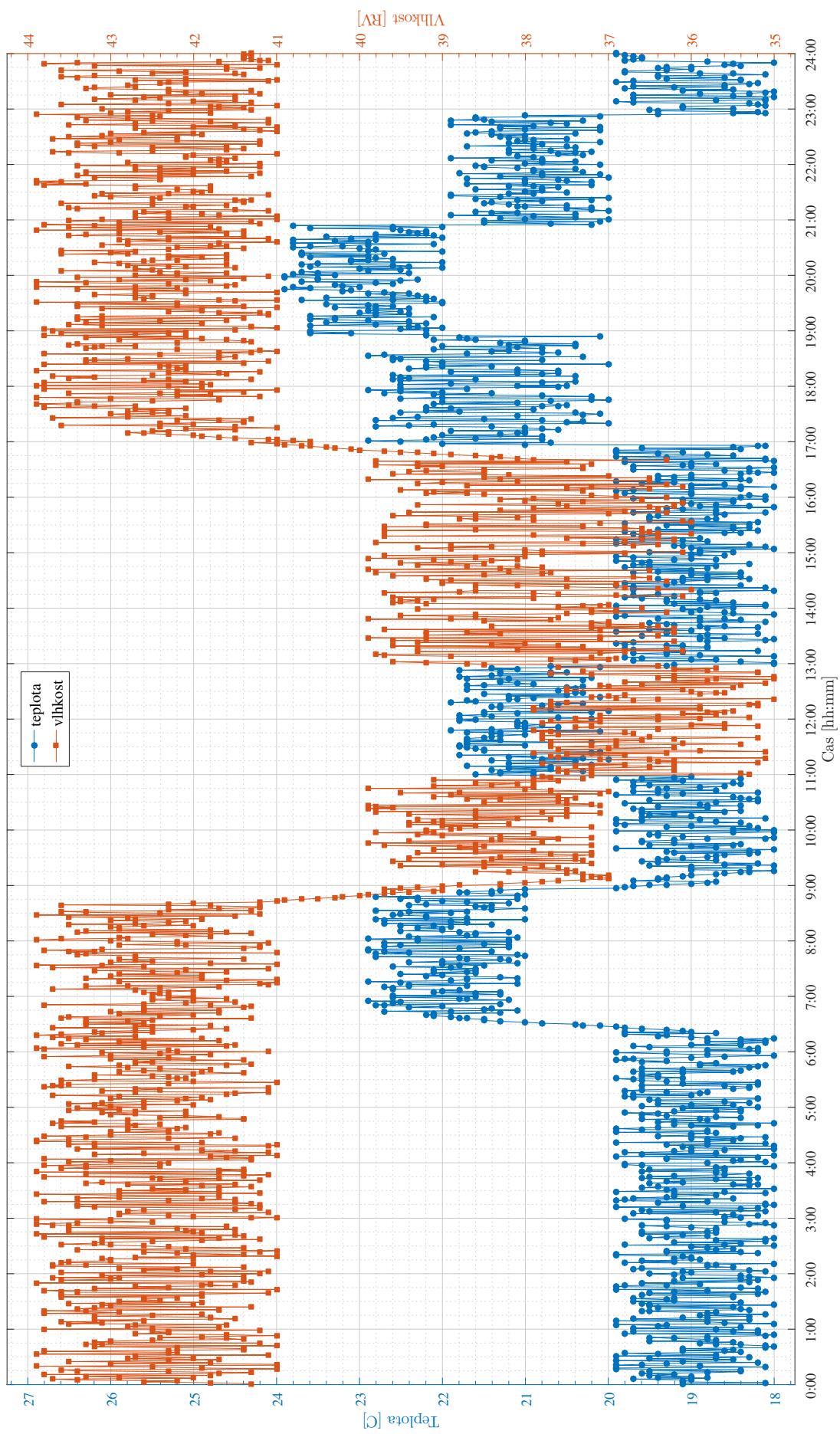
C UKÁZKA VIZUALIZACE DAT



Obr. C.1: Vizualizace měření elektroměrem ZPA (interval 24 hodin)



Obr. C.2: Vizualizace měření vodoměry Bonega (interval 24 hodin)

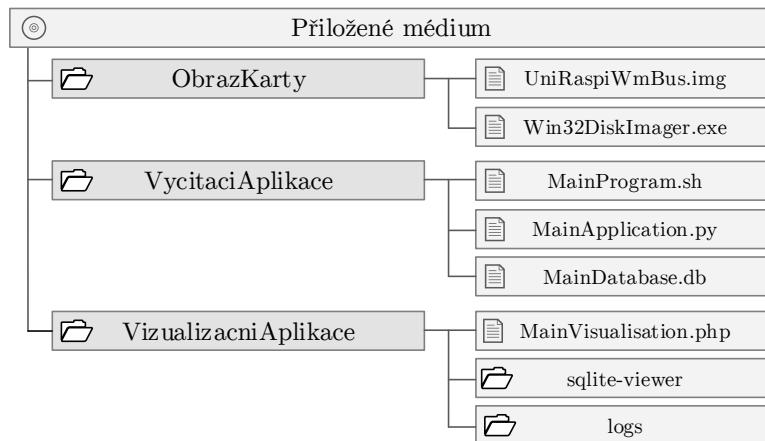


Obr. C.3: Vizualizace měření čidlem Weptech (interval 24 hodin)

D OBSAH PŘILOŽENÉHO DVD

K diplomové práci je přiloženo CD, obsahující bitový obraz MicroSD karty se systémem Raspbian, ve kterém je nainstalováno a nastaveno vše potřebné ke spuštění vzorové aplikace a zahájení komunikace s vyčítanými Wireless M-Bus zařízeními. Taktéž jsou zde uloženy zdrojové kódy vyčítací i vizualizační aplikace.

Médium obsahuje následující strukturu:



Návod pro spuštění aplikace:

1. Pomocí aplikace Win32DiskImager z \ObrazKarty\Win32diskimager.exe zařaďte obraz \ImageKarty\UniPiRaspiWmBus.img na pamětovou kartu typu MicroSD minimální velikosti 4GB.
2. Kartu zasuňte do jednotky UniPi Neuron a zapněte tuto jednotku. Po startu jednotky dojde k aktivaci aplikace a zachytávání WM-Bus komunikace.
3. Jednotka očekává přidělení IP adresy z DHCP serveru. Po přidělení IP adresy lze provádět vizualizaci zachytávaných dat pomocí aplikace na adrese <http://ip-adresa-jednotky/>
4. Případně po ssh přihlášení [unipiraspibus\wmbusunipirasi] a zadání příkazu screen -r lze sledovat přímo výstup aplikace v konzoli.

Vyčítací aplikace může být spuštěna samostatně, bez přítomnosti RaspberryPi, rozšiřující desky UniPi či IQRF komunikačního modulu. Je implementován demonstrační režim s předpřipravenou sadou dříve zachycených telegramů:

- Režim příjmu zašifrovaných telegramů modulem IQRF:
`python MainApplication.py aes_iqrf`
- Režim příjmu zašifrovaných obecných telegramů:
`python MainApplication.py aes_clean`
- Režim příjmu nešifrovaných telegramů:
`python MainApplication.py clean`