

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

BAKALÁŘSKÁ PRÁCE



# 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

## ZVÝŠENÍ ODOLNOSTI KOMUNIKAČNÍ ARCHITEKTURY PRO DÁLKOVOU SPEKTROSKOPII

INCREASE OF ROBUSTNESS OF COMMUNICATION ARCHITECTURE FOR REMOTE SPECTROSCOPY

### BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

### AUTOR PRÁCE

AUTHOR

Patrik Cebo

### VEDOUCÍ PRÁCE

SUPERVISOR

prof. Ing. Jozef Kaiser, Ph.D.

BRNO 2018



# Bakalářská práce

bakalářský studijní obor **Teleinformatika**  
Ústav telekomunikací

**Student:** Patrik Cebo

**ID:** 186040

**Ročník:** 3

**Akademický rok:** 2017/18

## NÁZEV TÉMATU:

### Zvýšení odolnosti komunikační architektury pro dálkovou spektroskopii

#### POKYNY PRO VYPRACOVÁNÍ:

Popište aktuální stav architektury pro dálkovou spektroskopii a teoreticky pojednejte o kritických bodech s ohledem na možnou chybovost celého systému. Provedte teoretickou rešerši na zvýšení robustnosti architektury. Na základě výsledků rešerše vyberte vhodné řešení a navrhnete možnou implementaci do stávajícího systému.

Navrhnete a zrealizujete prototypové zařízení se zvýšenou robustností architektury do aktuálního systému. Vytvořte firmware, který bude obsluhovat komunikaci mezi jednotlivými moduly. V konečném kroku popište Vaše kompletní řešení funkčního vzorku.

#### DOPORUČENÁ LITERATURA:

- [1] B. Dobkin and J. Hamburger, Analog Circuit Design Volume Three: Design Note Collection. Newnes, 2014.
- [2] T. Martin, The Designer's Guide to the Cortex-M Processor Family, Second edition. Newnes, 2016.
- [3] J. Smith, "Basics of industrial ethernet", Plant Engineering (Barrington, Illinois), vol. 58, no. 8, pp. 65-71, 0408AD.

**Termín zadání:** 5.2.2018

**Termín odevzdání:** 29.5.2018

**Vedoucí práce:** prof. Ing. Jozef Kaiser, Ph.D.

**Konzultant:** doc. Ing. Petr Číka, Ph.D.

**prof. Ing. Jiří Mišurec, CSc.**  
*předseda oborové rady*

#### UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské 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**

Táto práca sa zaoberá návrhom novej komunikačnej architektúry zariadenia pre diaľkovú laserovú spektroskopiu rLIBS. Konceptný návrh vychádza z analýzy súčasného stavu poznania v oblasti diaľkovej laserovej spektroskopie. V práci je pojednané o výhodách a nevýhodách bežne dostupných zberníc s ohľadom na ich chybovosť v priemyselnom nasadení. Výsledkom práce je výber a návrh novej komunikačnej architektúry, ktorá je realizovaná štandardom Ethernet. Práca sa ďalej zaoberá problematikou a návrhom riadenia krokových motorov.

## **KLÚČOVÉ SLOVÁ**

LIBS, diaľková laserová spektroskopia, krokový motor, mikrokontrolér, RS485, Ethernet, ARM, LwIP, FreeRTOS, STM, Trinamic

## **ABSTRACT**

This thesis deals with the design of new communication architecture of a device for remote laser spectroscopy. The conceptual model is based on the analysis of the state of the art systems for remote laser spectroscopy. Moreover, the advantages and disadvantages of custom electric buses and its failures in deployment are discussed within this thesis. The result of the work includes the design of new communication architecture supported by Ethernet standards. The next part of the thesis focuses on the customized stepper motor controller composition, which will resolve the theoretical findings arising from the analysis.

## **KEYWORDS**

LIBS, remote laser spectroscopy, stepper motor, microcontroller, RS485, Ethernet, ARM, LwIP, FreeRTOS, STM, Trinamic

CEBO, Patrik. *Zvýšení odolnosti komunikační architektury pro dálkovou spektroskopii*. Brno, 2018, 70 s. Bakalárska práca. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedúci práce: prof. Ing. Jozef Kaiser, Ph.D.

## VYHLÁSENIE

Vyhlasujem, že som svoju bakalársku prácu na tému „Zvýšení odolnosti komunikační architektury pro dálkovou spektroskopii“ vypracoval samostatne pod vedením vedúceho bakalárskej práce, využitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce.

Ako autor uvedenej bakalárskej práce ďalej vyhlasujem, že v súvislosti s vytvorením tejto bakalárskej práce som neporušil autorské práva tretích osôb, najmä som nezasiahol nedovoleným spôsobom do cudzích autorských práv osobnostných a/alebo majetkových a som si plne vedomý následkov porušenia ustanovenia § 11 a nasledujúcich autorského zákona Českej republiky č. 121/2000 Sb., o práve autorskom, o právach súvisiacich s právom autorským a o zmene niektorých zákonov (autorský zákon), v znení neskorších predpisov, vrátane možných trestnoprávných dôsledkov vyplývajúcich z ustanovenia časti druhej, hlavy VI. diel 4 Trestného zákoníka Českej republiky č. 40/2009 Sb.

Brno .....

.....

podpis autora

## POĎAKOVANIE

Touto cestou by som sa chcel rád poďakovať prof. Ing. Jozefovi Kaiserovi, Ph.D. za umožnenie realizácie bakalárskej práce na zariadení rLibs. Ďalej by som chcel poďakovať Ing. Michalovi Bradovi za uvedenie do problematiky zo stavby zariadenia, Ing. Lukášovi Novákovi za pripomienky pri tvorbe prototypových zariadení a celej LIBS skupine z VUT FSI. Za poskytnutie ilustrovaných zobrazení zariadenia by som rád poďakoval Ing. Janovi Novotnému, Ph.D. a za konzultácie k vizuálnej forme práce Ing. Petrovi Číkovi, Ph.D. Najväčšie poďakovanie patrí mojej rodine zato že mi umožnili študovať na vysokej škole.

Brno .....

.....

podpis autora

# OBSAH

Úvod	10
<b>1 Laserová spektroskopia</b>	<b>11</b>
1.1 Mobilná diaľková spektroskopia . . . . .	12
1.2 Súčasné zariadenia pre mobilnú laserovú spektroskopiu . . . . .	13
<b>2 Zariadenie pre mobilnú diaľkovú laserovú spektroskopiu rLIBS</b>	<b>15</b>
2.1 Opticko-mechanický princíp . . . . .	16
2.2 Elektronické zapojenie . . . . .	17
2.2.1 Komunikácia s počítačom . . . . .	17
2.2.2 Primárna zbernica . . . . .	18
2.2.3 Ovládanie motorov . . . . .	18
2.2.4 Bloková schéma . . . . .	19
2.3 Použité zbernice . . . . .	19
2.3.1 Zbernica RS232 . . . . .	20
2.3.2 Zbernica RS485 . . . . .	20
2.3.3 Univerzálna sériová zbernica - USB . . . . .	22
2.4 Chybovosť systému . . . . .	23
2.4.1 Rušenie zberníc . . . . .	23
2.4.2 Komunikačný protokol . . . . .	23
<b>3 Návrh novej architektúry pre zariadenie rLIBS</b>	<b>25</b>
3.1 Motorizácia . . . . .	25
3.1.1 Budič krokového motoru TMC260A-PA . . . . .	26
3.1.2 Kontrolér TMC429-PI24 . . . . .	27
3.1.3 Budič krokového motoru TMC262-LA . . . . .	28
3.2 Uvažované zbernice . . . . .	28
3.2.1 Zbernica CAN . . . . .	29
3.2.2 Zbernica Ethernet . . . . .	30
3.2.3 Porovnanie zberníc . . . . .	32
3.3 Návrh komunikácie cez zbernicu ethernet . . . . .	33
3.3.1 Univerzálne rozhranie MII/RMII . . . . .	33
3.3.2 Ethernetový prevodník DP83848CVV/NOPB . . . . .	35
3.3.3 Mikrokontrolér STM32F407VGT6 . . . . .	36
3.4 Návrh komunikačného protokolu . . . . .	37
3.4.1 Objektový zápis JSON . . . . .	38

3.4.2	Inicializácia . . . . .	41
<b>4</b>	<b>Implementácia navrhutej architektúry pre zariadenie rLIBS</b>	<b>42</b>
4.1	Testovanie štandardu ethernet . . . . .	42
4.2	Vývoj riadiacej elektroniky . . . . .	43
4.3	Programovanie . . . . .	45
4.3.1	Nízkoúrovňová implementácia TCP/IP . . . . .	47
4.3.2	Operačný systém reálneho času RTOS . . . . .	49
4.3.3	Vývoj firmwaru . . . . .	51
<b>5</b>	<b>Testovanie</b>	<b>54</b>
	<b>Záver</b>	<b>58</b>
	<b>Literatúra</b>	<b>59</b>
	<b>Zoznam symbolov, veličín a skratiek</b>	<b>62</b>
	<b>Prílohy</b>	<b>65</b>
	Obsah priloženého CD . . . . .	65
	Príloha č.1 . . . . .	66
	Príloha č.2 . . . . .	67
	Príloha č.3 . . . . .	68
	Príloha č.4 . . . . .	69
	Príloha č.5 . . . . .	70

# ZOZNAM OBRÁZKOV

1.1	Schéma aparatury pre spektroskopiu LIBS . . . . .	11
1.2	Rozdelenie zariadení pre mobilný LIBS . . . . .	12
1.3	Robotické vozidlo Curiosity. . . . .	13
1.4	rLIBS - zariadenie pre mobilnú diaľkovú spektroskopiu. . . . .	14
2.1	rLIBS - zariadenie pre mobilnú diaľkovú spektroskopiu. . . . .	15
2.2	Bloková optická schéma zariadenia rLIBS . . . . .	16
2.3	Drivre Trinamic . . . . .	19
2.4	RS232 - Priebeh komunikácie. . . . .	21
2.5	RS485 - Priebeh komunikácie. . . . .	21
2.6	Prehľad konektorov štandardu USB. . . . .	22
2.7	Štruktúra datagramov protokolu Trinamic. . . . .	24
3.1	Bloková schéma TMC429-PI24 . . . . .	27
3.2	Vnútna štruktúra TMC262-LA a TMC260A-PA . . . . .	28
3.3	Štandard ISO 11898. . . . .	29
3.4	Zbernica CAN - Priebeh komunikácie. . . . .	29
3.5	Model TCP/IP. . . . .	31
3.6	Ethernetový rámec. . . . .	31
3.7	Bloková schéma komunikácie. . . . .	33
3.8	Bloková schéma prevodníka DP83848CVV. . . . .	35
3.9	Vyhľadávanie zariadení v sieti. . . . .	37
4.1	Bloková schéma riadiacej elektroniky náhľadového systému. . . . .	43
4.2	DPS riadiacej elektroniky. . . . .	44
4.3	Proces kompilácie . . . . .	46
4.4	Štruktúra firmwaru. . . . .	51
4.5	Stavový diagram komunikácie. . . . .	52
5.1	Testovacie zariadenie. . . . .	54
5.2	Pridelená IP adresa. . . . .	55
5.3	Test dostupnosti zariadenia. . . . .	55
p.1	Ladenie v programe Qt Creator. . . . .	66
p.2	UDP uvítací datagram. . . . .	67
p.3	TCP inicializačný segment. . . . .	68
p.4	TCP segment - žiadosť o vlastnosti. . . . .	69
p.5	TCP segment - odoslané vlastnosti. . . . .	70

## ZOZNAM TABULIEK

2.1	Štandardy a rýchlosti univerzálnej sériovej zbernice. . . . .	22
2.2	Prehľad statusov odpovede. . . . .	24
2.3	Prehľad najčastejšie používaných príkazov. . . . .	24
3.1	Prehľad používaných štandardov ethernetu. . . . .	30
3.2	Porovnanie zberníc z hľadiska použitia pri návrhu novej architektúry	32
3.3	Popis signálov rozhrania MII. . . . .	34
3.4	Popis signálov rozhrania RMII. . . . .	34
3.5	Základné parametre mikrokontroléru STM32F407VGT6 . . . . .	36

# ÚVOD

Potreba identifikovať chemické prvky obsiahnuté v skúmanom materiáli sa prakticky vyskytuje v každej vedeckej, alebo priemyselnej oblasti. V súčasnosti je použitie rozsiahle a siaha od zdravotníctva, geológie, biológie cez potravinárstvo, stavebníctvo až po moderné odvetvia, ktoré sa zaoberajú nanotechnológiami či polovodičovými štruktúrami [1].

Jedným zo zástupcov analýzy prvkového zloženia materiálov je spektroskopia laserom budenej plazmy (Laser-Induced Breakdown Spectroscopy, skr. LIBS), ktorej história sa spája s objavom laseru (Theodore Maiman, 1960). Technológia LIBS patrí medzi moderné spôsoby materiálovej analýzy a je využívaná k rýchlemu skúmaniu látok vo všetkých skupenstvách umiestnených v atmosfére, vo vákuu či dokonca pod vodou. Výhodou tejto metódy je možnosť analyzovať vzorky na diaľku bez prípravy povrchu a z relatívne malým poškodením (tzv. kvazi-nedeštruktívna metóda).

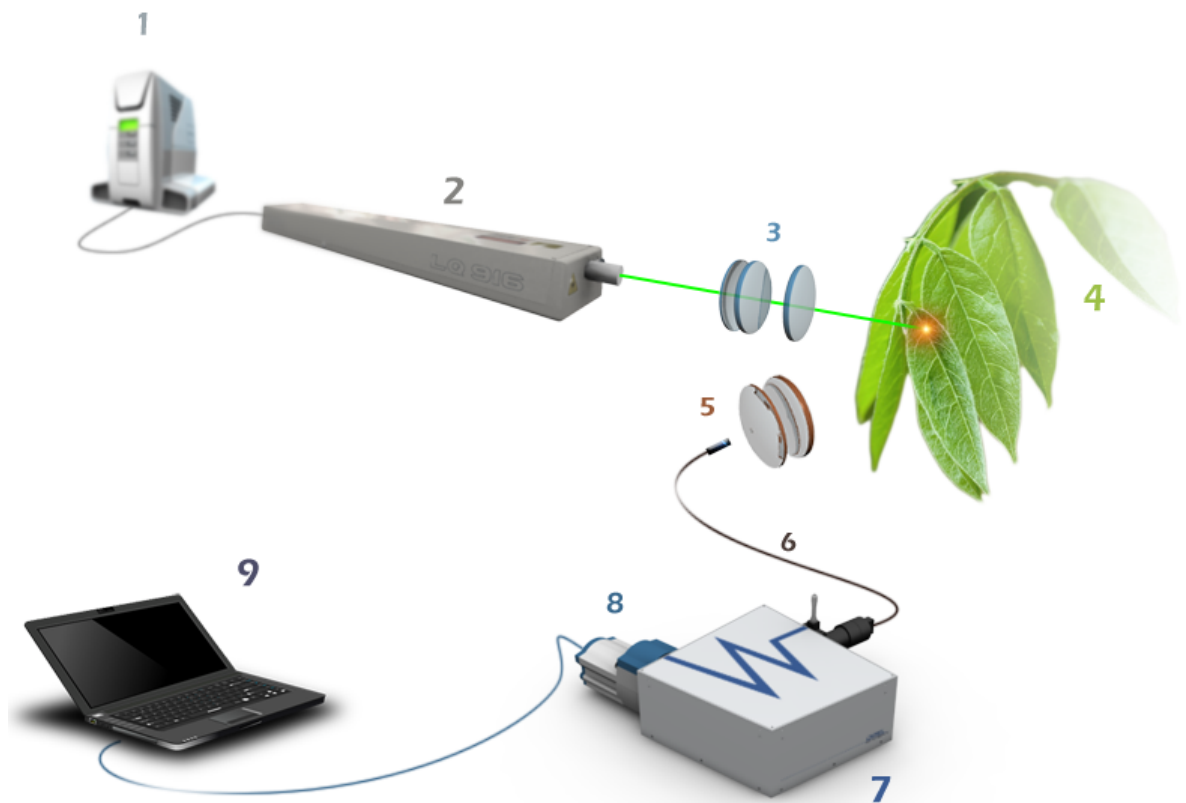
V prípade že vzdialenosť medzi vzorkou a aparátom presahuje metre, hovoríme o tzv. „diaľkovej spektroskopii“. Vzdialenosť detekcie sa môže pohybovať v rozmedzí desiatok až stoviek metrov podľa špecifického uplatnenia metódy LIBS. Technológia LIBS je využívaná v ťažko dostupných miestach, kedy nie je možné skúmanú vzorku preniesť do laboratória. Jedným z príkladov tejto metódy je analýza hornín na planéte Mars, kde robotické vozidlo (robot Curiosity, viac v podkapitole 1.2) zbiera informácie o prvkovom zložení planéty.

Laboratórium laserové spektroskopie na Ústave fyzikálneho inžinierstva (ÚFI) Fakulty strojného inžinierstva VUT v Brne sa od roku 1997 aktívne zaoberá vývojom metód materiálovej analýzy technikami LIBS. Výskumná skupina pod vedením prof. Ing. Jozefa Kaisera, Ph.D. disponuje potrebnou aparatúrou pre realizáciu laboratórnej spektroskopie technikou LIBS a mobilnej diaľkovej spektroskopie rLIBS.

Predmetom tejto práce je popísať aktuálny stav architektúry zariadenia rLIBS a pojednať o jeho kritických bodoch s ohľadom na chybovosť komunikačnej zbernice. Na základe výsledkov je potrebné navrhnúť riešenie na zvýšenie odolnosti komunikačnej architektúry s ohľadom na použitie moderných technológií.

# 1 LASEROVÁ SPEKTROSKOPIA

LIBS pracuje na princípe analýzy spektra žiarenia plazmy. Plazma je vytvorená tesne nad povrchom vzorky odparením časti skúmaného materiálu pomocou pulzného laseru. Laserový zväzok je fokusovaný na malý priestor veľkosti niekoľkých  $\mu\text{m}$ , vďaka čomu dosahuje vysokú plošnú hustotu žiarenia, typicky až  $\text{GW}\cdot\text{cm}^{-2}$ . V mieste dopadu laserového zväzku sa materiál zahreje, odparuje, atomizuje, ionizuje a vzniká horúca plazma. Žiarenie plazmy je zachytené zbernou optikou a ďalej prenesené pomocou optického vlákna do spektrálneho analyzátoru, kde dopadá na CCD detektor kamery. Výsledný signál predstavuje závislosť intenzity žiarenia na vlnovej dĺžke. Celý proces analýzy je schematicky znázornený na obrázku 1.1.



Obr. 1.1: Schéma aparatury pre spektroskopiu LIBS:

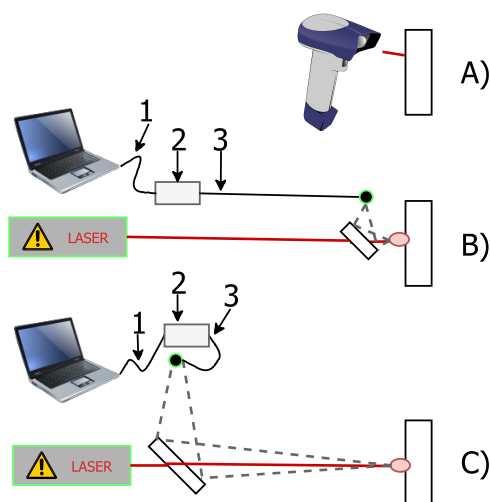
1) zdroj laseru, 2) hlavica laseru, 3) fokusačná optika, 4) vzorka, 5,6) zberná sústava, 7) spektrometer, 8) detektor, 9) notebook zo softvérom na vyhodnotenie spektra.

## 1.1 Mobilná diaľková spektroskopia

Mobilná diaľková laserová spektroskopia umožňuje vykonať rýchlu *in-situ* chemickú analýzu na vzorkách, ktoré nie je možné dopraviť do laboratória. Analyzované vzorky sú buď nebezpečné na prenos (výbušné poprípade rádioaktívne materiály), alebo sa nachádzajú na ťažko dostupných miestach (napr. na planéte Mars) poprípade nie je možné vykonať analýzu v laboratóriu z dôvodu možného oslabenia skúmaného prvku (konštrukcie budov, mostov atď.).

Mobilné zariadenie pre diaľkovú laserovú spektroskopiю je tvorené rovnakými komponentami ako laboratórne zariadenie a odlišuje sa iba usporiadaním jednotlivých komponentov, ich veľkosťou a výkonom z dôvodu celkovej miniaturizácie a mobility zariadenia. Pri vonkajšom použití môže byť zariadenie vystavované extrémnym teplotám, prašnosti, vlhkosti a ďalším nebezpečným vplyvom v ktorých musí byť schopné pracovať. Mobilné zariadenia sa ďalej rozdeľujú do troch skupín na dva druhy [2], podľa obr. 1.2.

- Prenosné LIBS zariadenia (Portable LIBS systems) sú kompaktné prenosné zariadenia slúžiace na rýchlu materiálovú analýzu.
- Zariadenia LIBS pre diaľkové meranie sa ďalej rozdeľuje podľa prenosu žiarenia plazmy na prenos žiarenia optickým vláknom (Remote LIBS systems) a prenos žiarenia vzduchom (Stand-off systems).



Obr. 1.2: Rozdelenie zariadení pre mobilný LIBS:

A) Portable LIBS systems, B) Remote LIBS systems, C) Stand-off LIBS systems, 1. USB kábel, 2. Spektrálny analyzátor, 3. Optické vlákno.

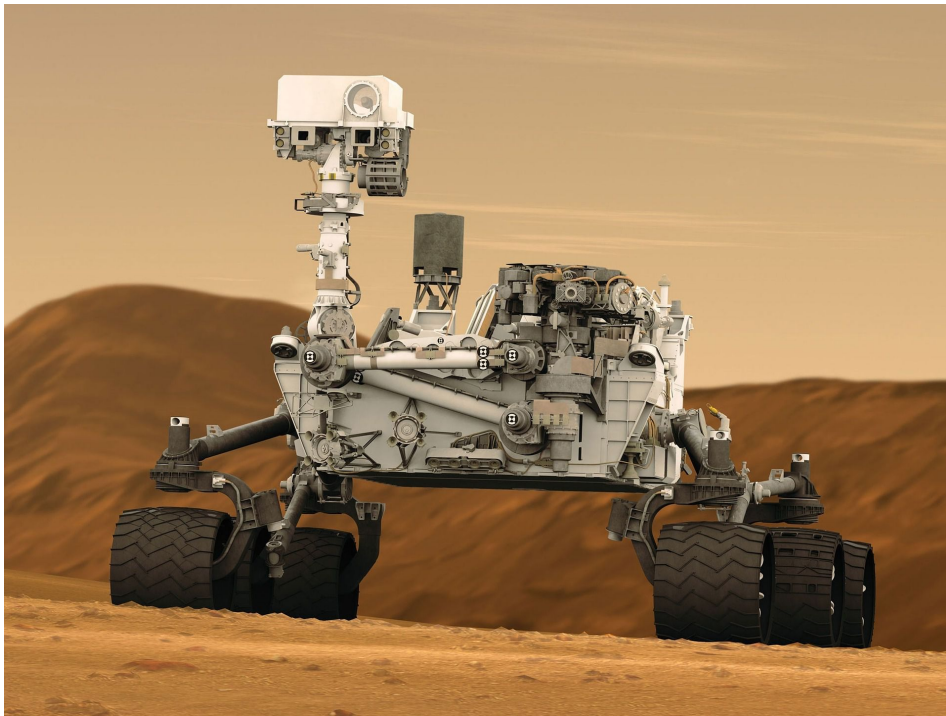
## 1.2 Súčasná zariadenia pre mobilnú laserovú spektroskopiu

- ST-LIBS:

Komerčný potenciál mobilnej diaľkovej laserovej spektroskopie je rozsiahly, čoho dôkazom je projekt ST-LIBS firmy Applied Photonics, ktorá predstavila v októbri roku 2009 už piatu generáciu stand-off zariadenia. Toto zariadenie využíva prevažne americká armáda na detekciu výbušnín a je schopné detektovať potencionálnu hrozbu na vzdialenosť 100 m [8].

- ChemCam:

Zariadenie ChamCam [6] (z angl. Chemistry and Camera) využíva diaľkovú laserovú spektroskopiu (konkrétne Stand-off system) pre *in-situ* chemickú analýzu zloženia hornín na planéte Mars [4]. ChemCam je súčasťou robotického vozidla Curiosity, pozri obr. 1.3 ktoré predstavuje jedno z desiatich vedeckých zariadení, ktoré slúžia na analýzu planéty. ChemCam dokáže analyzovať vzorky na vzdialenosť 2-7 metrov a výsledky analýzy sú prenášané cez optické vlákno do polohovacej hlavy a ďalej do centrálného riadiaceho systému.



Obr. 1.3: Robotické vozidlo Curiosity [5].

- rLIBS:

Zariadenie pre mobilnú diaľkovú laserovú spektroskopiu rLIBS (z angl. Remote Laser-Induced Breakdown Spectroscopy) zobrazené na obrázku 1.4 bolo zostrojené tímom vedcov z CEITEC VUT v roku 2014. V rámci medzinárodného strojárskeho veľtrhu, ktorý sa konal v roku 2014, bolo zariadenie ocenené zlatou medailou za naj-inovatívnejší exponát výstavy [9].



Obr. 1.4: rLIBS - zariadenie pre mobilnú diaľkovú spektroskopiu.

## 2 ZARIADENIE PRE MOBILNÚ DIALKOVÚ LASEROVÚ SPEKTROSKOPIU RLBS

Súčasná konštrukcia zariadenia rLIBS pozostáva z pojazdného vozíka a motorizovaného teleskopu, čo umožňuje rýchlu a mobilnú analýzu v teréne. V prípade statického merania je možné umiestniť motorizovaný teleskop na statív, pričom sú zachované všetky funkcie zariadenia, pozri obrázok 2.1. Celé zariadenie váži približne 180 kg a obsahuje všetko potrebné pre *in-situ* analýzu. Mobilný vozík obsahuje osobný počítač, zdroj na napájanie elektroniky, zdroj na napájanie laseru, spektroskop a priemyselnú dotykovú obrazovku ktorá je súčasťou konštrukcie. V motorizovanom teleskope sa nachádza laser, zberná a laserová optika, náhľadová kamera a elektronika na riadenie systému. Motorizovaný teleskop je prepojený s vozíkom pomocou flexibilnej hadice v ktorej vedú všetky dátové linky, silové rozvody a hadice na chladenie, vďaka čomu je možné umiestniť jednotku na ťažko dostupné miesta do vzdialenosti 5 m od zadnej strany vozíka. Vďaka motorizácii je možné počítačom riadiť eleváciu a rotáciu teleskopu a tým zameriavať na miesto merania. Pre meranie metódou LIBS je navyše potreba riadiť ohniskovú vzdialenosť zbernej a laserovej optiky [7].

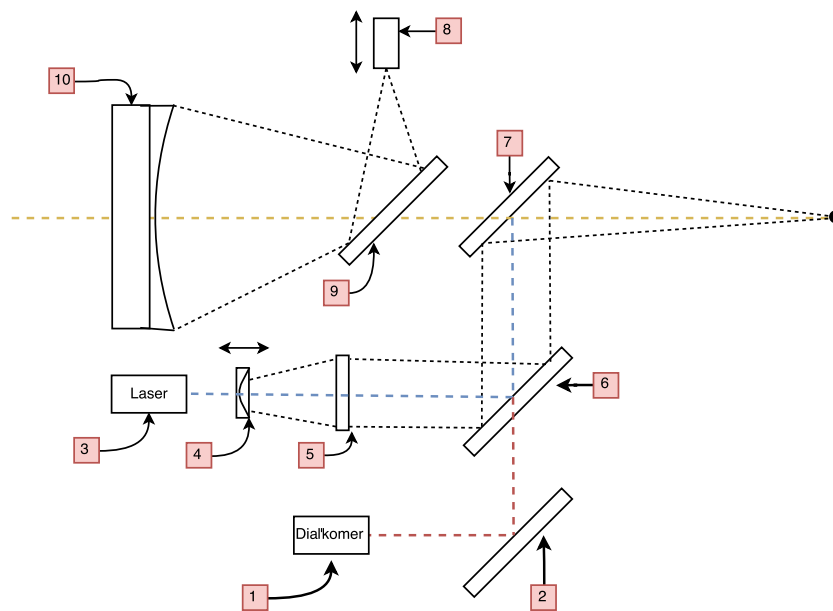


Obr. 2.1: rLIBS - zariadenie pre mobilnú diaľkovú spektroskopiu.

## 2.1 Opticko-mechanický princíp

Motorizovaný teleskop obsahuje dve optické sústavy ktoré sa dajú rozdeliť na dve časti a ich bloková schéma je zobrazená na obrázku 2.2.

- **Laserová optika** zabezpečuje navedenie laserového zväzku do optickej osy teleskopu. Laserový zväzok je roziahnutý konkávnou šošovkou a následne je pomocou konvexnej šošovky opäť zaostrý. Poloha konkávnej šošovky sa môže meniť vďaka motorizovanému posuvu, čím je zabezpečené zaostrenie zväzku na analyzovaný materiál. Laserový zväzok z konvexnej šošovky putuje na periskop, ktorý navedie zväzok do optickej osy. Periskop obsahuje dva zrkadlá, pričom zrkadlo č.6 je polopriepustné čo umožňuje privedenie ďalších laserových zväzkov do optickej osy, ktoré slúžia na zameriavanie a meranie vzdialenosti od skúmaného predmetu [3].
- **Zberná optika** je tvorená newtonovým teleskopom, ktorý sa skladá z primárneho parabolického a rovinného sekundárneho zrkadla, ktoré smeruje žiarenie plazmy do optického vlákna. Optickým vláknom sa šíri zväzok žiarenia ďalej do spektrografu. Vzdialenosť vstupu optického vlákna od sekundárneho zrkadla je nastaviteľná pomocou motorizovaného posuvu ktorý zabezpečuje zaostrenie zberu [3].



Obr. 2.2: Bloková optická schéma zariadenia rLIBS.

1 - Dialkometer, 2,7,9 - Rovinné zrkadlo, 3 - Pulzný laser, 4 - Konkávná šošovka, 5 - Konvexná šošovka, 6 - Polopriepustné zrkadlo, 8 - Motorizované optické vlákno, 10 - Parabolické zrkadlo.

## 2.2 Elektronické zapojenie

Zariadenie rLIBS je napájané zo sieťového napätia 230V/50Hz podľa ČSN 33 0010 ed.2 [10]. Z tohoto dôvodu je potreba elektrickej siete v mieste merania, poprípade nutnosť použitia externého zdroja elektrického napätia, najčastejšie formou elektrocentrály. Použitie batérii ako zdroj elektrickej energie nieje možný nakoľko má prístroj príkon 1600 W a takéto riešenie by predstavovalo nárast rozmerov a hmotnosti celého zariadenia.

Primárna komunikácia riadiaceho počítača s elektronikou je realizovaná 9 bajtovým protokolom pomocou spoločnej sériovej zbernice. Protokol definovala spoločnosť Trinamic, ktorá ho využíva pre komunikáciu medzi radičmi krokových motorov. Protokol je veľmi dobre škálovateľný a obsahuje všetky potrebné inštrukcie pre ovládanie zariadenia. Spoločnosť Trinamic ho v súčasnosti používa vo všetkých svojich výrobkoch a je použitý v mnohých zariadeniach, hlavne však v 3D tlačiarňach.

Okrem spoločnej sériovej zbernice a zbernice USB obsahuje zariadenie aj bezpečnostnú zbernicu ILC (z angl. Inter Lock), ktorá je prepojená so všetkými bezpečnostnými prvkami v zariadení. Bezpečnostné prvky sú 4 a slúžia na odpojenie napájania od siete v prípade poruchy. Bezpečnostná zbernica obsahuje dve stop tlačítka a koncové spínače, ktoré sú umiestnené za referenčnými spínačmi v osách elevácie a rotácie pre prípad ich zlyhania. Vozík je prepojený s motorizovaným teleskopom pomocou flexibilnej hadice v ktorej sú vedené dve USB zbernice, napájanie elektroniky veľkosti 24 V, bezpečnostná zbernica, optické vlákno, napájanie laseru a komunikačná zbernica laseru. Z hľadiska riadenia zariadenia je prístroj rozdelený nasledovne: pohyb v ose rotácie, pohyb v ose elevácie, fokusovanie laserového zväzku, fokusovanie optického zberu, meranie vzdialenosti, náklon náhľadového modulu, zoom a zaostrenie náhľadového modulu.

### 2.2.1 Komunikácia s počítačom

Zariadenie komunikuje pomocou dvoch usb zberníc. Prvou zbernicou je zbernica štandardu USB 2.0 ktorá ovláda riadenie prístroja. Po vstupe do motorizovaného teleskopu je zbernica USB rozvetvená pomocou USB hubu, pričom jedna vetva vedie do prevodníka USB na RS485 a druhá vetva vedie do prevodníka USB na RS232, ktorý je použitý na získanie dát z laserového merania vzdialenosti. Druhou zbernicou je zbernica USB 3.0 ktorá slúži pre prenos dát z kamery náhľadového modulu.

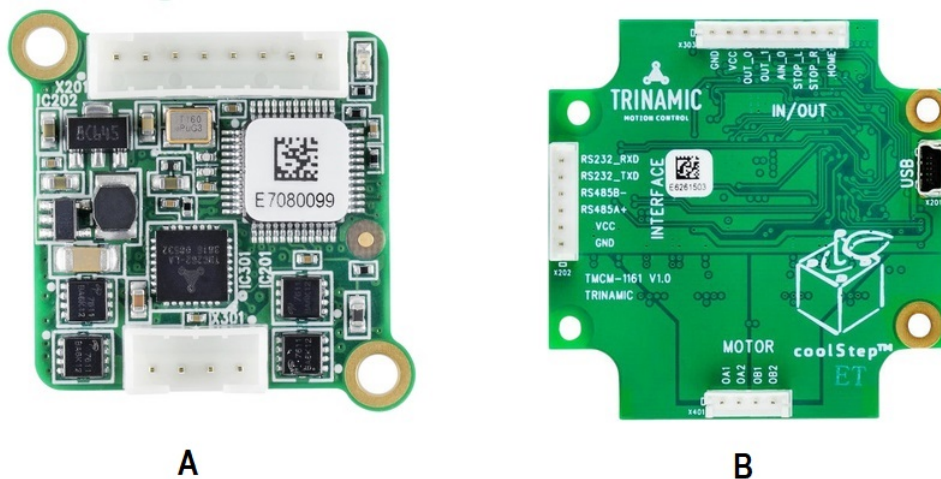
## 2.2.2 Primárna zbernica

Z hľadiska komunikácie medzi jednotlivými systémami elektroniky bol prístroj vybavený zbernicou RS485 ktorá je rozvetvená v celom systéme. Zbernica RS485 je vedená pomocou krútených dvojliniek popri napájaní o veľkosti 24V a pre komunikáciu používa protokol Trinamic. Zbernica je v zariadení niekoľkokrát rozvetvená pomocou svorkovnic umiestnených na plošnom spoji čo znižuje množstvo použitej kabeľáže, ale zároveň znižuje odolnosť voči rušeniu. Podrobný popis protokolu Trinamic je uvedený v sekcii 2.4.2.

## 2.2.3 Ovládanie motorov

Zariadenie obsahuje celkovo 6 krokových motorov viď. príloha s označením V-02391-01.pdf v adresári pdf. Všetky motory sú ovládané pomocou driverov od spoločnosti Trinamic a sú napojené na vnútornú zbernicu RS485, ktorá je rozvetvená v zariadení. Každý driver umožňuje vyčítanie dvoch koncových spínačov na osu, pričom v prípade nárazu na mechanický spínač dôjde k automatickému zastaveniu motora. Drivere od spoločnosti Trinamic umožňujú riadiť niekoľko vstupno-výstupných pínov, čo umožňuje využiť ich výstupy na riadenie elektromagnetických bázd, ktoré sa nachádzajú na osách rotácie a elevácie. V zariadení sú celkovo použité dva typy budičov: TMCM-1021 a TMCM-1161 ktorých fotografia je zobrazená na obrázku 2.3.

- **Zariadenie TMCM-1021** predstavuje malý kompaktný jednoosový driver s výstupom až 1,4A RMS pre cievku s rozhraním RS485. Rozmery driveru sú kompatibilné so štandardom NEMA11(28×28mm) čo umožňuje jednoduchú montáž na zadnú stranu motora [11]. Zariadenie navyše obsahuje 2 vstupné a 2 vstupno-výstupné piny ktoré je možné ovládať pomocou protokolu trinamic.
- **Zariadenie TMCM-1161** predstavuje rozšírenú verziu zariadenia TMCM-1021 ktoré je schopné dodať až 2,8A RMS pre cievku. Okrem rozhrania RS485 je vybavené rozhraním RS232 a USB. Z hľadiska mechanickej konštrukcie je určený pre montáž na motor so štandardom NEMA23(57×57mm) a NEMA24(60×60mm) [12]. Rovnako ako TMCM-1021 obsahuje aj toto zariadenie vstupné a výstupné piny, konkrétne 2 na výstup a 4 na vstup.



Obr. 2.3: Drivre krokových motorov Trinamic: A) TMC201, B) TMC1161.

### 2.2.4 Bloková schéma

V prvom kroku bola vykonaná pasportizácia zariadenia ktorej výsledkom je vyhotovená bloková schéma zariadenia, ktorá obsahuje informáciu o názvoch a prepojení jednotlivých dosiek plošných spojov DPS. Bloková schéma bola vytvorená pomocou programu Eagle s využitím blokov a portov, ktoré sa objavili v programe Eagle od verzie 7.0. Výsledná bloková schéma sa nachádza v prílohe v adresári pdf s označením V-02391-01.pdf. Výsledky pasportizácie slúžia k vyhľadávaniu použitých komponentov a názvov DPS.

## 2.3 Použité zbernice

V zariadení rLIBS sú použité celkovo 3 rôzne typy zberníc. Prvou zbernicou je zbernica USB, ktorou sú prenášané dáta od riadiaceho počítača do motorizovaného teleskopu a naopak. Štandard USB je pomerne rýchly, ale ako sa v priebehu používania ukázalo, je veľmi náchylný na rušenie. Podrobný popis je uvedený v sekcii 2.3.3. Zo štandardu USB sú pomocou prevodníkov získané zbernice RS232 a RS485. Zbernica RS232 je historický prežitok minulej doby a je použitá na zameriavanie a meranie vzdialenosti k analyzovanej vzorke. Zbernica RS485 bola použitá z dôvodu riadenia krokových motorov drivermi od firmy Trinamic ako je uvedené v sekcii 2.2.3. Zbernica RS485 sa ukázala ako stabilný štandard a jej popis je uvedený v sekcii 2.3.2.

### 2.3.1 Zbernica RS232

Sériová linka RS232 (z angl. Recommended Standard 232) je štandard používaný od roku 1960 v osobných počítačoch na komunikáciu medzi 2 zariadeniami. Využíva sériovú komunikáciu a umožňuje plne duplexné prepojenie pomocou liniek RxD a TxD. Štandard RS232 je definovaný na fyzickej vrstve a obsahuje predpis komunikačných rýchlostí a napätových úrovní [13]. Logické úrovne signálu sú definované rozsahom napätia nasledovne : Log 1 (-15V až -3V) Log 0(3V až 15V) a ich priebeh je zobrazený na obr. 2.4. Jednotkou prenášaných dát je baud, ktorý predstavuje počet zmien stavu vodiča za jednu sekundu. Výhodou tohoto štandardu je možnosť použiť synchronný a asynchronný prenos dát.

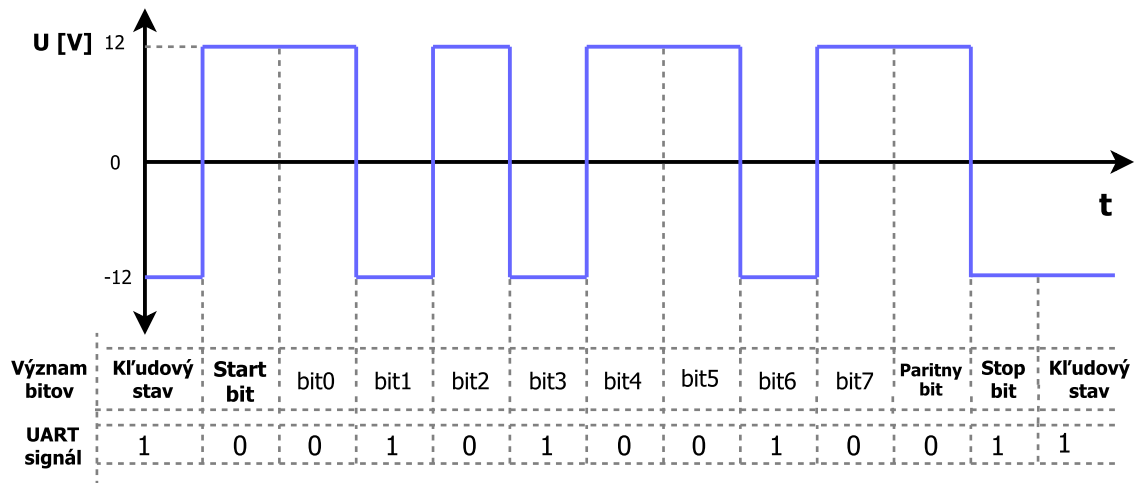
- **Synchronný režim** komunikácie je režim pri ktorom je spolu s dátami prenášaný aj hodinový signál, čím je zaistená synchronizácia príjmu a vysielania. Začiatok a koniec dátového toku je označovaný synchronizačnými značkami. Tento spôsob komunikácie je rýchlejší ako asynchronný režim, ale je náročnejší na programové spracovanie dát.

- **Asynchronný režim** komunikácie je režim pri ktorom musí prijímač generovať svoj vlastný hodinový signál čo spôsobuje odchýlky frekvencie a dochádza k chybivosti pri prenose dát. Začiatok vysielania je označovaný štart bitom po ktorom nasledujú dáta, paritný bit a stop bit.

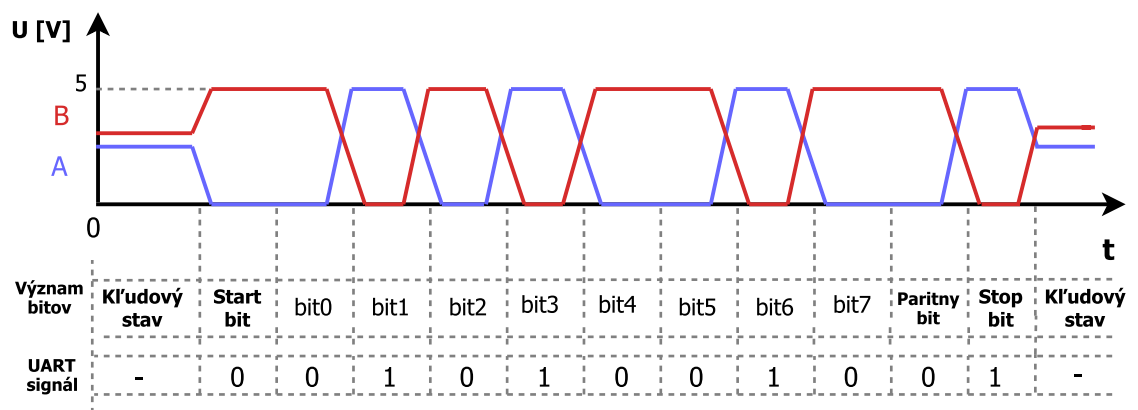
Medzi nevýhody štandardu RS232 patrí hlavne obmedzená dátová rýchlosť, maximálna vzdialenosť zbernice a komunikácia typu bod-bod. Maximálna doporučená dátová rýchlosť pri komunikácii na vzdialenosť 12 m je 20 kbs, čo je pre dnešné požiadavky priemyslu nedostatočné. Výhodu použitia tohoto režimu je prevažne plne duplexný režim komunikácie.

### 2.3.2 Zbernica RS485

Sériová linka RS485 (z angl. Recommended Standard 485) je štandard definovaný v roku 1983 organizáciou EIA. Štandard RS485 je narozdiel od štandardu RS232 navrhnutý tak aby využíval iba dvojvodičové poloduplexné spojenie. Tento štandard je navyše zbernicový a umožňuje pripojiť až 32 zariadení do zbernice na vzdialenosť až 1200 m. Podľa organizácii EIA je definovaný na fyzickej vrstve a využíva dva vodiče označované ako A a B (niekedy tiež + a -) pričom logické úrovne sú definované ako rozdiel napätí medzi vodičom A a vodičom B pozri obr. 2.5.



Obr. 2.4: RS232 - Priebeh komunikácie.

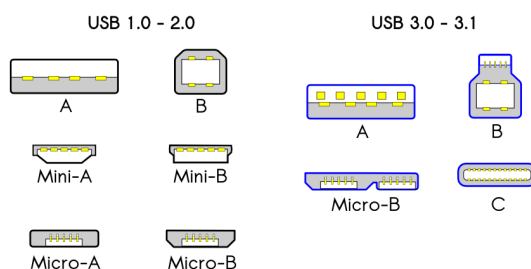


Obr. 2.5: RS485 - Priebeh komunikácie.

Kedže je signál vo vodičoch A a B invertovaný, neprejavuje sa šum a rušenie spôsobené napätovými špičkami, nakoľko sú pripočítané k obom linkám a výsledný rozdiel napätia je rovnaký. Typická impedancia vedenia má 50-60  $\Omega$  a je realizovaná dvomi rezistormi na začiatku a konci zbernice. Medzi výhody tohoto štandardu patrí prevažne komunikačná rýchlosť, ktorá dosahuje až 50Mbit/s [14] a veľká odolnosť voči rušeniu. Medzi nevýhodu patrí poloduplexný režim, ktorý neumožňuje súčasné vysielanie a príjem dát a jednotlivé prenosy je potrebné potvrdzovať.

### 2.3.3 Univerzálna sériová zbernica - USB

Univerzálna sériová zbernica USB (z angl. Universal Serial Bus) vznikla v roku 1994 v spolupráci siedmich firiem, (medzi ktoré patrí napr. Microsoft, IBM, Intel a ďalší) ako možný spôsob náhrady doteraz používaných štandardov na osobných počítačoch. Samotný štandard USB definuje použité káble, konektory a komunikačný protokol [15]. Štandard bol určený prevažne pre počítače a rýchlo nahradil štandard RS232, no ukázalo sa, že je natoľko univerzálny, že sa dnes používa takmer v každom zariadení. Základné typy konektorov sú zobrazené na obr. 2.6. Štandardne používaný konektor typu A obsahuje 4 vodiče, dva slúžia na napájanie o veľkosti 5V a dva vodiče označené ako Data+ (D+) a Data- (D-) sú použité na prenos dát. Dáta sú prenášané pomocou diferenciálneho signálu rovnako ako pri RS485, kedy je jeden signál posielaný s kladnou polaritou a druhý je inevertovaný. Dáta sú kódované pomocou invertovaného signálu bez návratu k nule (NRZI z angl. Non Return to Zero Inverted). Štandard USB zaznamenal po svojom príchode veľký rozmach a v priebehu 6 rokov bola vydaná už verzia USB 2.0 ktorá podporovala rýchlosť až 480 Mbit/s [16], detailný zoznam štandardov a ich rýchlostí je uvedený v tabuľke 2.1.



Obr. 2.6: Prehľad konektorov štandardu USB [18].

Tab. 2.1: Štandardy a rýchlosti univerzálnej sériovej zbernice.

Názov Štandardu	Dátum vydania	Prenosová rýchlosť
USB 1.0	Január 1996	1,5 Mbit/s
USB 1.1	August 1998	12 Mbit/s
USB 2.0	Apríl 2000	480 Mbit/s
USB 3.0	November 2008	5 Gbit/s
USB 3.1	Júl 2013	10 Gbit/s
USB 3.2	September 2017	20 Gbit/s

## 2.4 Chybovosť systému

Nakoľko bol pri tvorbe zariadenia kladený dôraz na mechanickú konštrukciu je elektronická časť značne zastaraná a využíva štandardy používané koncom minulého storočia. Technický pokrok spektroskopov a kamier si vyberá vysokú daň, hlavne v množstve prenášaných dát. Dnešné zariadenia potrebujú prenášať stále viac dát prevažne väčšou rýchlosťou, čo spôsobuje neustálu obnovu existujúcich štandardov. Napríklad pri štandarde USB bola rýchlosť navyšovaná niekoľkokrát, ako je uvedené v sekcii 2.3.3. Nakoľko je komunikácia s riadiacou elektronikou poloduplexná, riadiaci software čaká na odpoveď čo v prípade chyby vedie k veľkému oneskoreniu. Chybovosť systému sa prejavuje hlavne pri prenose dát z náhľadovej kamery, nakoľko sú prenášané veľké dátové toky.

### 2.4.1 Rušenie zberníc

Hlavným zdrojom chybovosti systému je zbernica USB, ktorá nie je dimenzovaná na prenos dát na veľkú vzdialenosť. Motorizovaný teleskop je prepojený s vozíkom káblami dĺžky 7m, ktoré sú vedené spolu so silovými rozvodmi. Hlavným zdrojom rušenia je vysoké napätie, ktoré je privádzané do lasera. V prípade funkcie laseru vznikajú obrovské napätové špičky, ktoré vnášajú rušenie do USB káblov.

### 2.4.2 Komunikačný protokol

Komunikačný protokol od firmy Trinamic má pevne stanovenú štruktúru datagramu veľkosti 9 bajtov, pozri obr. 2.7. Prvý bajt predstavuje adresu zariadenia s ktorým chceme komunikovať, keďže sa jedná o protokol určený na zbernicu môžeme naadresovať až 255 zariadení. Druhý bajt datagramu predstavuje číslo príkazu, ktorý sa má vykonať, popis aktuálne používaných príkazov je v tabuľke 2.3. Nasledujúci bajt predstavuje typ príkazu a ide o definovanú podmnožinu inštrukcií pre daný príkaz. V závislosti na type príkazu predstavuje tretí bajt číslo motoru poprípade pinu. Nasledujúce 4 bajty predstavujú dáta a posledný bajt je kontrolný súčet celého datagramu. Pri každom prijatom datagrame musí byť odoslané potvrdenie formou datagramu odpovede dĺžky 9 bajtov, pozri obr. 2.7, ktorý je odlišný od vysielacieho. Prvý bajt predstavuje návratovú adresu ktorá musí byť rovnaká pre všetky odpovede a odpovedá adrese riadiaceho člena, pričom nasledujúci bajt predstavuje práve adresu zariadenia ktoré odpoveď posiela. Tretí bajt predstavuje status príkazu ako je uvedené v tabuľke 2.2 a nasledujúci bajt informuje o type príkazu. Ďalej je už štruktúra zhodná z datagramom na odoslanie, 4 bajty dát a kontrolný súčet.

Formát vysielania		Formát odpovede	
Počet bajtov	Popis bajtu	Počet bajtov	Popis bajtu
1	Adresa modulu	1	Adresa cieľa
1	Číslo príkazu	1	Adresa modulu
1	Typ príkazu	1	Status
1	Číslo motoru	1	Číslo príkazu
4	Dáta	4	Dáta
1	Kontrolný súčet	1	Kontrolný súčet

Obr. 2.7: Štruktúra datagramov protokolu Trinamic.

Tab. 2.2: Prehľad statusov odpovede.

Kód	Význam statusu
100	Úspech - žiadna chyba
101	Príkaz načítaný do programu TMCL
1	Nesprávny kontrolný súčet
2	Neznámy príkaz
3	Nesprávny typ
4	Nepovolená hodnota
5	Pamäť EEPROM zamknutá
6	Príkaz nieje k dispozícii

Tab. 2.3: Prehľad najčastejšie používaných príkazov.

Typ príkazu	Číslo príkazu	Popis príkazu
MST	3	Stop motor movement - zastavenie pohybu motora
MTP	4	Move to position - presun na pozíciu
SAP	5	Set axis parameter - nastavenie parametrov pre motor
GAP	6	Get axis parameter - vyčítanie parametrov pre motor
SGP	9	Set global parameter - nastavenie globálnych parametrov
GGP	10	Get global parameter - vyčítanie globálnych parametrov
RFS	13	Reference search - nájdenie referencie
SIO	14	Set output - nastavenie výstupu
GIO	15	Get input/output - vyčítanie hodnoty, stavu vstupu/výstupu

## 3 NÁVRH NOVEJ ARCHITEKTÚRY PRE ZARIADENIE RLIBS

Z hľadiska riadenia prístroja je potrebné nahradiť všetky použité štandardy zberníc univerzálnym štandardom, ktorý je schopný prenášať veľké množstvo dát a zároveň je odolný voči rušeniu. Najväčšie dátové toky predstavuje náhľadová kamera a preto by mal byť použitý rovnaký štandard ako štandard zbernice kamery. Po odstránení všetkých existujúcich zberníc je potrebné nahradiť elektroniku riadenia, ktorá komunikuje pomocou štandardu RS485 a takisto drive krokových motorov. Nasledujúca kapitola sa zaoberá problematikou vývoja elektroniky pre budenie krokových motorov, výberom vhodného štandardu a návrhom komunikačného protokolu.

### 3.1 Motorizácia

Spoločnosť Trinamic predáva okrem hotových riešení ovládania motorov taktiež integrované obvody, ktoré používa vo svojich produktoch. Pretože sa v zariadení nachádzajú aj motory s väčším prúdovým odberom, je nutné použiť 2 rôzne výkonné integrované obvody. Pre riadenie prístroja je dôležité zoskupiť určité skupiny motorov z hľadiska ich polohy a výkonnosti.

#### **Riadenie fokusu zberu:**

Pre riadenie fokusácie zbernej optiky a laserového zväzku sú použité menšie motory od spoločnosti Microcon typu SX16-0402N s maximálnym prúdovým odberom 0,4A RMS. Z hľadiska ochrany a správnej funkcie posuvov sú obe zostavy vybavené koncovými spínačmi, ktoré je taktiež potrebné zahrnúť do riadenia.

#### **Riadenie náhľadu:**

Riadenie náhľadu pozostáva z náklonu, zaostrovania a zoomu náhľadovej kamery. Náklon kompenzuje polohu optickej osy náhľadového systému pre rôzne ohniskové vzdialenosti. Uhol náklonu je ovládaný motorom od značky Microcon typu SX16-0301D-1 s prúdovým odberom 0,35 A RMS. Zoom objektívu je riadený cez ozubené súkolie motorom od firmy Servo Drive typu 211-13-02 s max. prúdovým odberom 1,3 A RMS. Zaostrenie je zabudované priamo v objektíve náhľadového systému a pre riadenie je použitý sériový protokol od výrobcu objektívu, ktorý vychádza z protokolu SPI. Rovnako ako pri fokusácii je potrebné zabezpečiť zastavenie motorov pri dosiahnutí maximálnej polohy, čo je zaistené koncovými spínačmi, pozri prílohu v adresári pdf s označením V-02391-01.pdf. Náhľadový modul navyše obsahuje osvetlenie, ktoré musí byť riadené pomocou komunikačného protokolu.

### **Riadenie polohovania:**

Ovládanie rotácie a elevácie je zabezpečené pomocou dvoch výkonných motorov od firmy EC Motion typu SECM266-E2.0 s maximálnym prúdovým odberom 2,8 A RMS. Keďže osy polohovania nie su samosvorné, každý motor obsahuje elektromagnetickú brzdu od firmy EC Motion typu BRK26S, ktorá je z bezpečnostných dôvodov odbrzdená až po privedení napätia na cievku brzdy. Medzi ďalšie bezpečnostné prvky patria koncové spínače umiestnené tesne pri dorazoch jednotlivých os a bezpečnostné spínače, ktoré odpoja napájanie v prípade zlyhania brzd alebo koncových spínačov.

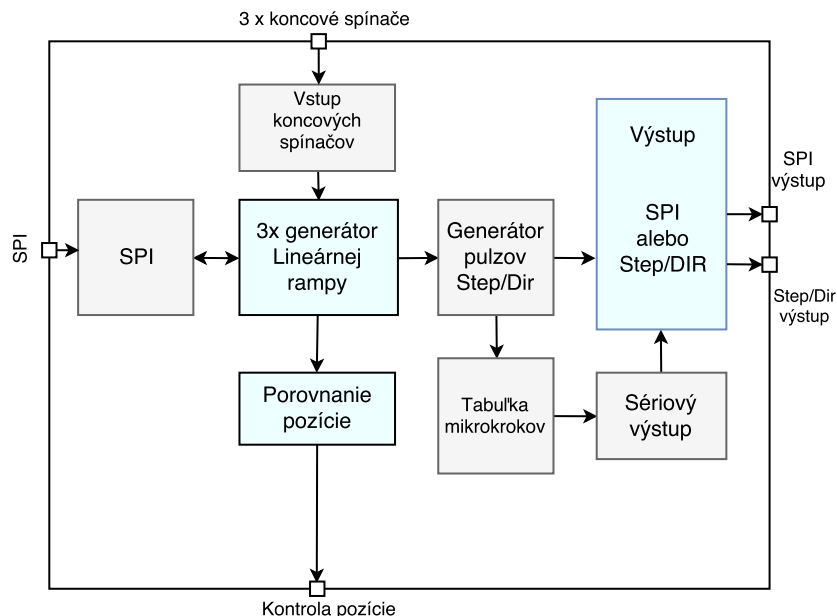
### **3.1.1 Budič krokového motoru TMC260A-PA**

Budič krokových motorov TMC260A-PA je produktom spoločnosti Trinamic, ktorý je bežnou súčasťou ich riadiacich zariadení. Tento integrovaný obvod dokáže napájať krokové motory napätím 0-40 V a pri dostatočnom chladení je schopný dodať maximálny prúd 2 A pre každú cievku. Samotný obvod disponuje senzorom proti prehriatiu ktorý zabezpečuje odpojenie cievky v prípade zvýšenej teploty, ďalej je schopný snímať prúd do cievky a zefektívniť tak spotrebu podľa aktuálnej záťaže motora či detekovať prerušenie, alebo nepripojenú cievku. Základné parametre, ako sú kludový prúd cievky, maximálny prúd cievky, mikrokrokovanie, voľba módu a podobne sa nastavujú pomocou protokolu SPI, ktorý dokáže komunikovať pri použití externého 16 MHz oscilátora rýchlosťou až 8 MHz. Riadenie počtu a smeru otáčok sa vykonáva pomocou signálov Step (počet krokov) a Dir (smer otáčania). Okrem toho je možné tieto signály vynechať a použiť vnútorný sínusový generátor, ktorý je možné konfigurovať pomocou protokolu SPI. Tento spôsob ovládania je však programovo zložitejší a hlavne pomalší. Napájacie napätie je možné voľiť v rozsahu 3,3 - 5 V. Napätová tolerancia vstupných signálov predšavuje 1,2 násobok napájacieho napätia [19]. Vnútorná štruktúra obvodu je schematicky zobrazená na obrázku 3.2 a zapuzdrená v TQFP-44 púzdre o veľkosti 10 × 10 mm čím predstavuje komplexné riešenie pre budenie krokových motorom.

Frekvencia signálov Step a Dir závisí prevažne na použitom mikrokrovaní a maximálnej frekvencii oscilátora, ktorá môže byť až 16 MHz. Pri maximálnom rozlíšení 256 mikrokrokov na krok je maximálna frekvencia signálu Step rovná polovici frekvencie oscilátora, naopak pri minimálnom rozlíšení, kedy jeden signál step odpovedá jednému kroku motora je táto frekvencia  $512 \times$  menšia. Kontrolér obsahuje celkovo 5 20 bit registrov, ktoré sú nastavovateľné pomocou protokolu SPI. Nastavenie registrov je potrebné vykonať pri každom štarte, poprípade reštarte kontroléru.

### 3.1.2 Kontrolér TMC429-PI24

Kontrolér TMC429-PI24 je integrovaný obvod od spoločnosti Trinamic a je určený na riadenie až troch budičov krokových motorov napr. typu TMC260A-PA, pozri 3.1.1. Komunikácia s kontrolérom je realizovaná pomocou protokolu SPI a jeho výstupom je buď signál Step a Dir pre každý budič, alebo zbernica SPI ktorá je prepojená paralelne medzi všetkými použitými regulátormi. Tento integrovaný obvod dokáže zabezpečovať plynulý rozjazd krokových motorov pomocou funkcie lineárnej rampy, ktorá zabezpečuje postupný nárast rýchlosti až do nastavenej hodnoty. Okrem funkcie na pozvoľný rozbeh disponuje obvod generátorom pulzov, ktoré sa využívajú pre signály Step. Do integrovaného obvodu je možné pripojiť koncové spínače, ktoré slúžia na ochranu alebo kalibráciu zariadenia. Integrovaný obvod je napájaný napätím 3-3,6 V a nachádza sa v SMD púzdre veľkosti 15×12 mm typu SOP-24 [21]. Vnútna štruktúra je zobrazená na obrázku 3.1.

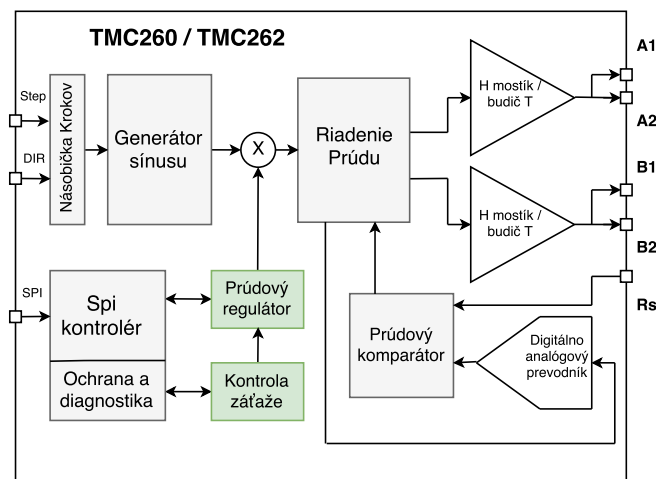


Obr. 3.1: Bloková schéma kontroléru TMC429-PI24.

Komunikácia je vedená cez protokol SPI s pevným datagramom o veľkosti 32 bitov, kde prvých 8 bitov predstavuje číslo motora, typ príkazu a možnosť zápisu poprípade čítania a zvyšných 24 bitov je použitých na prenos dát. Z hľadiska nastavenia registrov rozlišuje obvod globálne parametre a parametre pre jednotlivé motory. Medzi globálne parametre patria napr. nastavenie polarít signálov SPI prípadne Step/Dir, konfigurácia pinov pre koncové spínače. Parametre pre jednotlivé motory predstavujú napr. nastavenie maximálnej rýchlosti motora, maximálnej akcelerácie, aktuálnej pozície a ďalších, pozri [21].

### 3.1.3 Budič krokového motoru TMC262-LA

Budič krokových motorov TMC262-LA je obdoba TMC260A-PA, avšak neobsahuje integrovaný H mostík, pozri obrázok 3.2. Tento integrovaný obvod je určený na použitie z externým H mostíkom a je možné s ním riadiť prúdy cievky napr. veľkosti 10 A, záleží len na použitom type mostíka. Komunikácia a registre sú rovnaké ako pri TMC260A-PA, jediný rozdiel je v SMD púzdre ktoré je veľkosti 5×5 mm a typu QFN32. Napájacie napätie je v tolerancii od 3-5,25 V a obvod je schopný dodať výstupné napätie pre tranzistory v rozsahu 0-10 V [20].



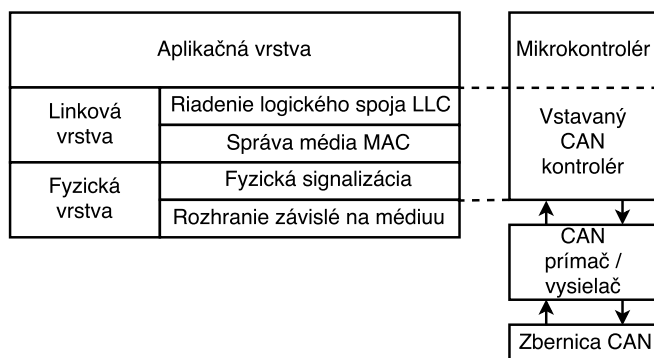
Obr. 3.2: Vnútroštruktúra Regulatorov prúdu krokových motorov TMC262-LA a TMC260A-PA.

## 3.2 Uvažované zbernice

V dnešnej dobe existuje mnoho zberníc, ktoré by mohli byť použité pre riadenie zariadenia rLIBS. Keďže bude systém navrhnutý na riadenie pomocou vstavaných (ďalej len anglicky embedded), systémov je potrebné vybrať vhodné zbernice z pohľadu možnej implementácie do mikrokontroléra. Štandardné 8 bitové mikrokontroléry obsahujú rozhranie univerzálneho asynchrónneho prijímača / vysielača UART (Universal Asynchronous Receiver/Transmitter), vďaka čomu môžeme použiť prevodník na zbernice RS232 a RS485. V súčasnom automobilovom priemysle sa rozvinul štandard CAN, pozri 3.2.1 a tak ho začali podporovať aj výrobcovia mikrokontrolérov. S nástupom architektúry ARM začali výrobcovia mikrokontrolérov vydávať 32 bitové mikrokontroléry, ktoré umožnili použitie rýchlejších štandardov, ako USB 2.0, pozri sekciu 2.3.3 a sekciu Ethernet 3.2.2.

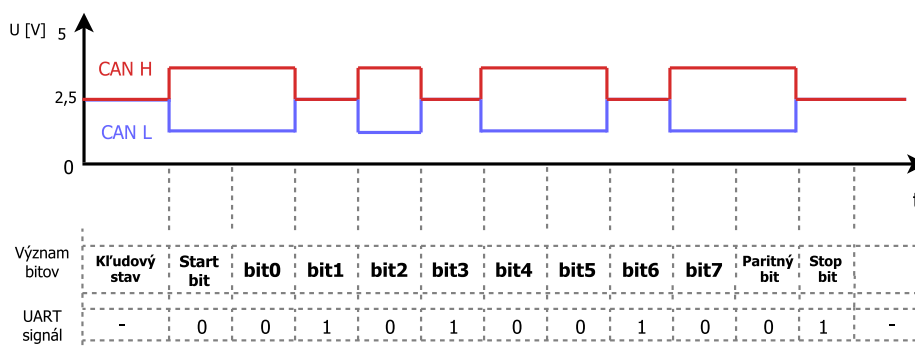
### 3.2.1 Zbernica CAN

Sériová zbernica CAN (z angl. Controller Area Network) je štandardná zbernica používaná v automobilovom priemysle, ktorá bola predstavená firmou Robert Bosch GmbH v roku 1983. Zbernica CAN obsahuje na rozdiel od štandardu RS485 definíciu linkovej vrstvy, čo predstavuje pevne definovaný rámec o veľkosti až 8 bajtov. V roku 1993 predstavila organizácia ISO štandard ISO 11898, ktorý bol rozdelený na dve časti. Prvá časť ISO 11898-1 obsahuje popis komunikácie na linkovej vrstve a druhá časť ISO 11898-2 popisuje fyzickú vrstvu komunikácie [22], pozri obr. 3.3.



Obr. 3.3: Štandard ISO 11898.

Zbernica CAN využíva dvojvodičové vedenie a signály sú označované ako CANH a CANL, pričom impedancia vedenia je rovnaká ako pri štandarde RS485. Logická úroveň signálu je definovaná podobne ako pri štandarde RS485 rozdielom napätí pričom pri log 1 je rozdiel napätí 0 V a pri log 0 je rozdiel napätí 2 V, pozri obr. 3.4. Maximálna komunikačná rýchlosť dosahuje 1 Mb/s pri vzdialenosti 40 m a klesá z narastajúcou vzdialenosťou až na 0.05 Mb/s pri vzdialenosti 1000 m [23].



Obr. 3.4: Zbernica CAN - Priebeh komunikácie.

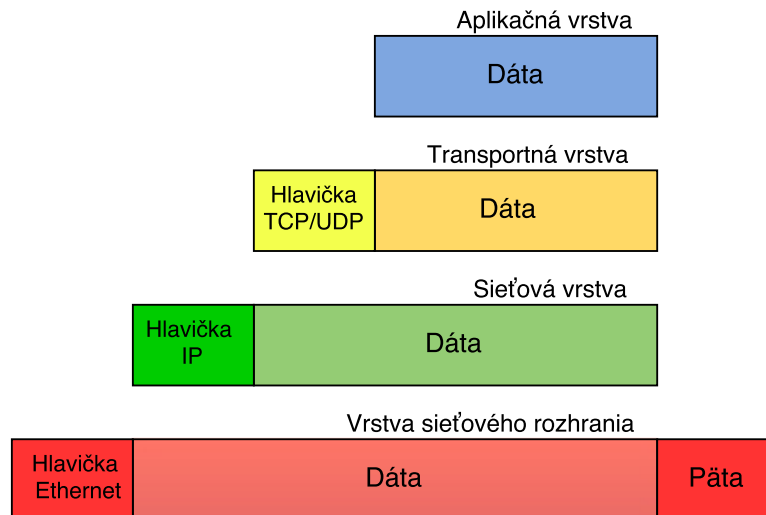
### 3.2.2 Zbernica Ethernet

Zbernica ethernet je v dnešnej dobe najpoužívanejšou technológiou na budovanie priemyselných a domácich počítačových sietí. Vznik ethernetu začal vo výskumnom centre PARC v Kalifornii v roku 1972, a bol inšpirovaný experimentálnou sieťou pre komunikáciu medzi havajskými ostrovmi s názvom ALOHAnet, ktorú navrhol inžinier Bob Metcalf [24]. Prvá verejná publikácia štandardu pre 10 Mbit/s ethernet bola vydaná až v osemdesiatych rokoch minulého storočia. Ethernet zažil veľký rozmach a začal sa nasadzovať na budovanie počítačových sietí. Najčastejšie používané médium bol koaxiálny kábel s predpísanou impedanciou 50 ohm ako predpisoval štandard 10Base5. Neskôr prišla potreba prenášať dáta väčšou rýchlosťou a tak sa začala používať dnes dobre známa krútená dvojlinka. Základný prehľad štandardov a prenosových rýchlostí je uvedený v tabuľke 3.1.

Tab. 3.1: Prehľad používaných štandardov ethernetu.

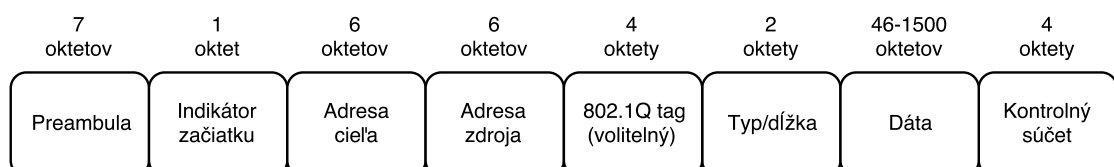
Druh káblu	Názov štandardu	Prenosová rýchlosť	Konektor
Koaxiálny kábel	100Base-2	10 Mbit/s	BNC
Koaxiálny kábel	100Base-5	10 Mbit/s	AUI
Krútená dvojlinka, kat. 3	10Base-T	10 Mbit/s	RJ-45
Krútená dvojlinka, kat. 5	100Base-T	100 Mbit/s	RJ-45
Krútená dvojlinka, kat. 5e	1000Base-T	1 Gbit/s	RJ-45
Krútená dvojlinka, kat. 6	1000Base-TX	1 Gbit/s	RJ-45
Krútená dvojlinka, kat. 7	10GBase-TX	10 Gbit/s	RJ-45
Krútená dvojlinka, kat. 8	40GBase-TX	40 Gbit/s	RJ-45
Optické vlákno	40GBase-ER4	40Gbit/s	SC/LC
Optické vlákno	100GBase-ER4	100 Gbit/s	SC/LC

Zbernica Ethernet potrebuje na rozdiel od ostatných zberníc zložitejší predpis komunikácie a je potrebné aby použitý hardware a software spolu dokonale spolupracovali. Z hľadiska komunikácie sa prenos dát najčastejšie popisuje modelom primárneho prenosového protokolu/protokolu sieťovej vrstvy TCP/IP (z angl. Transmission Control Protocol/Internet Protocol). Model obsahuje 4 vrstvy a zapúzdrenie dát je zobrazené na obrázku 3.5.



Obr. 3.5: Model TCP/IP.

- **Aplikačná vrstva** obsahuje protokoly na prenos dát napr.: HTTP, DNS, FTP ktoré pre svoju činnosť využívajú protokoly transportnej vrstvy.
- **Transportná vrstva** poskytuje spoľahlivý prenos dát s kontrolou spojenia pomocou protokolu TCP alebo nespoľahlivý a nekontrolovaný prenos protokolom UDP.
- **Sieťová vrstva** zaisťuje adresovanie, smerovanie a predávanie datagramov. Vrstva je implementovaná vo všetkých koncových zariadeniach a smerovačoch.
- **Vrstva sieťového rozhrania** je najnižšia vrstva a umožňuje prístup k fyzickému médiu. Z hľadiska implementácie je závislá na použitom type siete napr. Ethernet, Token ring, FDDI a ďalšie. V sieti Ethernet sú dáta reprezentované ethernetovým rámcov ktorého štruktúra je uvedená na obrázku 3.6.



Obr. 3.6: Ethernetový rámeč.

### 3.2.3 Porovnanie zberníc

Pri výbere vhodnej zbernice na návrh novej komunikačnej architektúry je potrebné brať do úvahy množstvo faktorov. Najdôležitejší faktor je maximálna prenosová rýchlosť zariadenia, ako je uvedené v tabuľke 3.2, pričom sú najvhodnejšie zbernice typu USB a ethernet. Ďalej je potrebné aby bola zbernica určená do priemyselných aplikácií a aby bola odolná voči rušeniu. Zbernica USB nie je práve najvhodnejším kandidátom na tvorbu priemyselných aplikácií, nakoľko sa konektorovanie rieši pomerne obtiažne, vhodnejšími kandidátmi sú skôr sériové zbernice CAN a RS485. Medzi najdôležitejšie kritérium však patrí možnosť prenosu dát z náhľadovej kamery bez akejkoľvek kompresie a to dokáže iba štandard ethernet a USB. Z výsledkov porovnania sa ukazuje ethernet ako najlepšia voľba pre návrh novej komunikačnej architektúry.

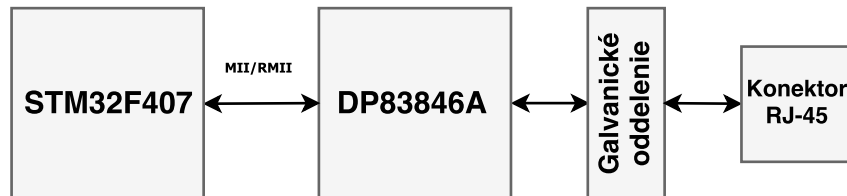
Tab. 3.2: Porovnanie zberníc z hľadiska použitia pri návrhu novej architektúry

Druh Zbernice	Maximálna prenosová rýchlosť	Odolnosť voči rušeniu	Priemyselné použitie	Prenos dát z kamery
RS232	20 kbit/s	Nízka	Nie	Nie
RS485	50 Mbit/s	Vysoká	Áno	Nie
CAN	1 Mbit/s	Vysoká	Áno	Nie
USB	10 Gbit/s	Stredná	Nie	Áno
Ethernet	100 Gbit/s	Vysoká	Áno	Áno

Zbernica ethernet má niekoľko hlavných výhod. Pri použití metalických krútených dvojliniek vhodnej kategórie môžeme dosahovať rýchlosť až 10 Gbit/s, pozri 3.1. Ethernet je dnes už bežnou súčasťou priemyselných aplikácií a existuje mnoho štandardov, ktoré presahujú rozsah použitia v domácnosti. Medzi tieto technológie patrí štandard s vysokou spoľahlivosťou a nízkou redundanciou HSR (z angl. High availability Seamless Redundancy) [25], ktorý je používaný napr. pre riadenie systémov s vysokou prioritou. HSR pracuje na princípe poslania ethernetového rámca dvoma smermi, čo znižuje možnosť straty dát. Každé zariadenie tak disponuje dvomi ethernet konektormi v zapojení do kruhu. Po odvysielaní prechádzajú rámce cez ostatné zariadenia v kruhu a pokiaľ im nie sú určené duplikujú prijatú správu na druhý port, čím sa rámec dostane na cieľový port. Pri bezchybnom stave sú tak na koncové zariadenie doručené dva rovnaké rámce avšak v rozdielnom čase (kvôli veľkosti a počtu zariadení v kruhu). Pokiaľ dorazí na koncové zariadenie iba jeden rámec alebo sa rámce nerovnajú, vyšle sa multikástová správa a nájde sa pôvodca straty či zmeny rámca. Celá správa systému je z pohľadu zariadení implementovaná na druhej - spojovej vrstve modelu OSI.

### 3.3 Návrh komunikácie cez zbernicu ethernet

Pre návrh novej komunikačnej architektúry bude použitý ethernetový prevodník zo spojovej vrstvy (MAC vrstvy) na fyzickú od firmy Texas Instruments typu DP83848C, ktorého detailný popis je uvedený v sekcii 3.3.2. Prevodník komunikuje pomocou rozhrania nezávislého na použítom médiu MII (z angl. Media Independent Interface) alebo redukovaného rozhrania nezávislého na použítom médiu RMII (z angl. Reduced Media Independent Interface). Spôsob komunikácie rozhraním MII/RMII je popísaný v sekcii 3.3.1. Riadenie prevodníka DP83846A bude zabezpečovať 32 bitový mikrokontrolér s architektúrou ARM od firmy STMicroelectronics typu STM32F407VG, ktorého základný popis je uvedený v sekcii 3.3.3. Bloková schéma navrhnutej architektúry je zobrazená na obrázku 3.7.



Obr. 3.7: Bloková schéma komunikácie.

#### 3.3.1 Univerzálne rozhranie MII/RMII

Rozhranie MII bolo definované ako univerzálne rozhranie medzi spojovou a fyzickou vrstvou. Popis rozhrania sa nachádza v štandarde IEEE 802.3u a popisuje možnosť použitia rôznych médií na prenos dát, napr. optické vlákno či krútená dvojlinka. Maximálna prenosová rýchlosť je 100 Mbit/s a dáta sú prenášané frekvenciou 25 MHz so šírkou kanálu 4 bity v oboch smeroch [26]. Rozhranie obsahuje taktiež menežovateľný vstupno výstupný signál MDIO (z angl. Management Data Input/Output), ktorý slúži na nastavovanie základných parametrov rozhrania, ako je napr. adresa a podobne. Celkový prehľad signálov je uvedený v tabuľke 3.3.

Rozhranie RMII vzniklo ako potreba zníženia počtu signálov a vychádza z rozhrania MII. Redukcia nastala pri prenose dát, kedy sa použilo namiesto 4 bitov iba polovica a zvýšila sa tak frekvencia prenosu na 50 MHz [26]. Niektoré signály ako napríklad signál COL boli vynechané a niektoré boli spojené do jedného, ako je uvedené v tabuľke 3.4.

Tab. 3.3: Popis signálov rozhrania MII.

Názov signálu	Popis signálu	Smer prenosu
RX_CLK	Hodinový signál prijímača	fyzická -> spojová
RXD[0-3]	Prímané dátové bity	fyzická -> spojová
RX_DV	Príjem platných dát	fyzická -> spojová
RX_ER	Detekcia chyby pri prenose	fyzická -> spojová
TX_CLK	Hodinový signál vysielača	spojová -> fyzická
TXD[0-3]	Vysielané dátové bity	spojová -> fyzická
TX_EN	Povolenie prenosu	spojová -> fyzická
TX_ER	Detekcia chyby pri vysielaní	spojová -> fyzická
CRS	Indikácia sieťovej aktivity	fyzická -> spojová
COL	Detekcia kolízií	fyzická -> spojová
MDIO	Prenos riadiacich informácií	fyzická -> spojová
MDC	Hodinový signál pre MDIO	spojová <-> fyzická

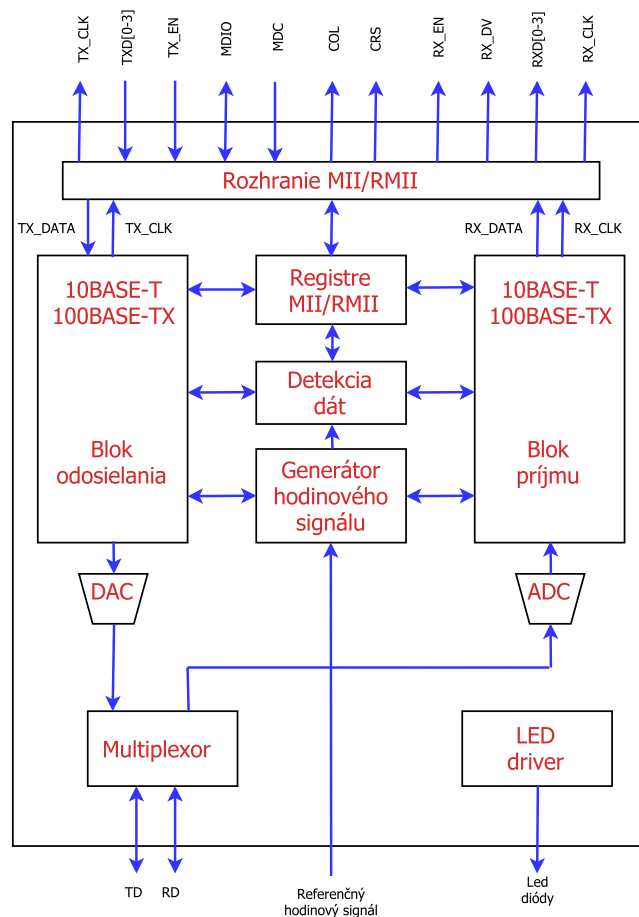
Tab. 3.4: Popis signálov rozhrania RMII.

Názov signálu	Popis signálu	Smer prenosu
REF_CLK	Spoločný hodinový signál	externý príjem
RXD[0-1]	Prímané dátové bity	fyzická -> spojová
RX_ER	Detekcia chyby pri prenose	fyzická -> spojová
TXD[0-1]	Vysielané dátové bity	spojová -> fyzická
TX_EN	Povolenie prenosu	spojová -> fyzická
CRS_DV	Indikácia sieťovej aktivity	fyzická -> spojová
MDIO	Prenos riadiacich informácií	fyzická -> spojová
MDC	Hodinový signál pre MDIO	spojová <-> fyzická

Dnes je už bežne dostupný štandard GMII ktorý slúži na prenos rýchlosťou 1 Gbit/s. Jeho štruktúra je rovnaká ako pri rozhraní MII, odlišný je iba spôsob prenosu dát, ktorý je realizovaný 8 bitmi v každom smere frekvenciou 125 MHz.

### 3.3.2 Ethernetový prevodník DP83848CVV/NOPB

Ethernetový prevodník zo spojovej vrstvy na fyzickú od firmy Texas Instruments typu DP83848C je kompaktný čip veľkosti 7×7 mm v púzdre LQFP-48. Prevodník podporuje štandardy 10BASE-T a 100BASE-TX a komunikuje z mikrokontrolérom pomocou rozhraní MII, alebo rýchlejšieho RMI. Prevodník používa CMOS technológiu s napájacím napätím veľkosti 3,3 V a je určený do extrémnych podmienok, pretože dokáže pracovať v teplotnom rozsahu -40 až 105 °C. Základný princíp je zobrazený na obrázku 3.8 pričom popis signálov je uvedený v tabuľkách 3.3 a 3.4. Prenos je podporovaný v plnom, ale aj poloduplexnom režime pri oboch rýchlostiach. Pri štarte prevodníku je potrebné previesť inicializáciu a nastaviť základné parametre ako napríklad prenosovú rýchlosť či MAC adresu zariadenia. Nastavenie je realizované pomocou rozhrania MDIO alebo SMI ktoré sú definované v IEEE 802.3 [28].



Obr. 3.8: Bloková schéma prevodníka DP83848CVV.

### 3.3.3 Mikrokontrolér STM32F407VGT6

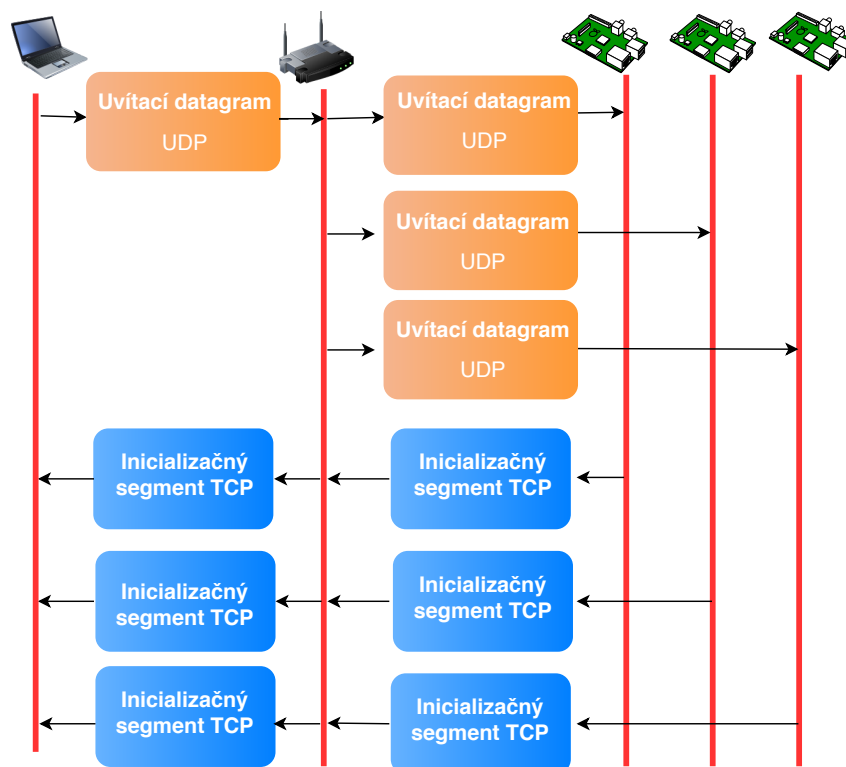
Mikrokontrolér od firmy STMicroelectronics využíva 32 bitovú architektúru ARM s jadrom typu M4. Vďaka maximálnej frekvencii 168 MHz a dostatočnej pamäti je schopný podporovať aj náročnejšie štandardy, ako napríklad USB, ethernet či pripojenie kamery pomocou paralelnej zbernice veľkosti až 14 bitov s rýchlosťou 54 Mbit/s. Vďaka vstavanému generátoru náhodných čísel umožňuje mikrokontrolér používať mnohé šifrovacie funkcie na zabezpečenie a prenos dát, ako napríklad posielanie dát protokolom TLS cez štandard ethernet. Základné parametre mikrokontroléru sa nachádzajú v tabuľke 3.5. Programovanie a ladenie mikrokontroléru prebieha pomocou rozhraní JTAG poprípade SWD [27].

Tab. 3.5: Základné parametre mikrokontroléru STM32F407VGT6

Napájacie napätie	1,8 - 3,6 V
Programová pamäť	1024 kB
Pamäť SRAM	192 kB
Maximálna frekvencia	168 MHz
Počet vstupno-výstupných pinov	82
Počet časovačov	14
A/D prevodník	16 × 12 bit
D/A prevodník	2 × 12 bit
Rozhranie SPI	3 ×
Rozhranie I2C	3 ×
Rozhranie CAN	2 ×
Rozhranie USART/UART	4/2 ×
USB 2.0	2 x
Rozhranie pre paralelnú kameru	áno
Rozhranie pre ethernet	áno
Generátor náhodných čísiel	ano

### 3.4 Návrh komunikačného protokolu

Nakoľko je potrebné navrhnuť univerzálny komunikačný protokol ktorý bude možné implementovať v embeded systémoch, voľba padla na hypertextový protokol HTTP. Protokol HTTP umožňuje prenášať rôzne skriptovacie jazyky, akými sú napríklad javascript či JSON. Návrh komunikácie bol realizovaný práve za pomoci štruktúry JSON. Po štarte zariadení je potrebné preskenovať sieť a zistiť všetky dostupné zariadenia v komunikačnom okruhu. Skenovanie siete je zabezpečené poslaním uvítacieho datagramu na broadcastovú adresu na predom definovaný UDP port. Štruktúra uvítacieho datagramu obsahuje špecifický reťazec s pevne stanovenou dĺžkou a informácie o odosielateľovi datagramu, v ktorom sa nachádza jeho verzia použitej IP adresy, ďalej samotná IP adresa a číslo TCP portu na ktorom čaká odpoveď. Pokiaľ zariadenie prijme na predom definovanom UDP porte datagram, skontroluje jeho špecifický reťazec a v prípade zhody odošle späť informácie o svojej konfigurácii. Tieto informácie obsahujú názov respektíve popis zariadenia, verziu použitého firmwaru, ďalej obsahujú číslo portu, typ IP adresy a samotnú adresu zariadenia, pozri sekciu 3.4.2. Odpoveď zariadenia je zaslaná na TCP port s adresou ktorá bola získaná z uvítacieho datagramu. Štruktúra nadviazania komunikácie je znázornená na obrázku 3.9.



Obr. 3.9: Vyhľadávanie zariadení v sieti.

### 3.4.1 Objektový zápis JSON

Java skriptový objektový zápis JSON (z angl. JavaScript Object Notation) je špeciálna forma textového súboru ktorá definuje spôsob zápisu dát a umožňuje tak prenášať rozne dátové typy [30]. Podobne ako značkovací jazyk XML (z angl. eXtensible Markup Language) môže obsahovať ľubovoľný počet objektov a podobjektov a je tak veľmi ľahko škálovateľný. Prenášané dáta môžu byť vo forme polí alebo objektov a využívajú všetky bežné premenné jazyka C++. Medzi hlavné výhody patrí prevažne možnosť reťazenia jednotlivých funkcií, čo značne znižuje počet odoslaných správ. Každé zariadenie musí implementovať minimálne tieto funkcie:

- **Property** popis každého parametru ktoré zariadenie podporuje.
- **SET** metóda pre nastavenie hodnoty podporovaného parametru.
- **GET** metóda pre vyčítanie hodnoty podporovaného parametru.
- **Subscribe** prihlásenie na odber informácií v prípade zmeny určitého parametru.
- **Notify** informuje odberateľov o zmene sledovaného parametru.

#### Property (Vlastnosti)

Parametre zariadenia nesú informácie o schopnostiach a rozsahu nastavenia dát. Štruktúra parametru obsahuje názov parametru, minimálnu hodnotu, maximálnu hodnotu a aktuálnu hodnotu. Napríklad pokiaľ máme možnosť ovládať PWM striedu na 8 kanáloch embeded zariadenia s rozlíšením 12 bitov, bude štruktúra vyzeráť podobne ako je naznačené vo výpise 3.1. Pokiaľ prebehne prvotná výmena uvítacieho datagramu, užívateľ alebo iný riadiaci člen môže poslať požiadavku na odoslanie parametrov. V prípade tejto požiadavky odošle zariadenie svoje vlastnosti odosielaťovi správy.

```
1 "properties": {
2     "PWM0" : {"min" : 0, "max" : 4095, "value" : 0},
3     "PWM1" : {"min" : 0, "max" : 4095, "value" : 0},
4     .
5     .
6     "PWM7" : {"min" : 0, "max" : 4095, "value" : 200}
7 }
```

Výpis 3.1: Príklad zápisu parametrov.

## SET (Nastavenie)

Príkaz Set umožňuje nastaviť parametre ktoré užívateľ obdržal pomocou property. Napríklad nastavenie PWM striedy na kanále 0 s hodnotou 2048 je zobrazené vo výpise 3.2. Príkaz SET môže obsahovať ľubovoľný počet požiadaviek na nastavenie a umožňuje tak vykonať celú konfiguráciu zariadenia v jednej správe. Rozsah nastavenia parametrov je uvedený vo vlastnostiach zariadenia.

```
1 {
2     "command" : "SET",
3     "values" : [
4         {
5             "property" : "PWM0",
6             "value" : 2048
7         }
8     ]
9 }
```

Výpis 3.2: Príklad nastavenia parametru.

## GET (Získavanie)

Príkaz Get slúži na vyčítanie parametrov. Napríklad pokiaľ chce užívateľ zistiť aktuálny stav PWM striedy odošle príkaz ako je naznačené vo výpise 3.3. Rovnako ako príkaz Set umožňuje aj tento príkaz poslať v rámci jednej správy viac požiadaviek.

```
1 { //požiadavka
2     "command" : "GET",
3     "properties" : [
4         "PWM0"
5     ]
6 }
7 { //odpoveď
8     "command" : "GET",
9     "values" : [
10        {
11            "property" : "PWM0",
12            "value" : 2048
13        }
14    ]
15 }
```

Výpis 3.3: Príklad vyčítania PWM striedy.

## Subscribe / Unsubscribe (Odber)

Pomocou príkazu subscribe sa môže akékoľvek zariadenie prihlásiť na odber hodnoty určitých parametrov. Pomocou tohoto príkazu môžu byť zariadenia združované a synchronizované. Štruktúra príkazu je znázornená vo výpise 3.4. Príkaz unsubscribe predstavuje opak príkazu subscribe a slúži na zrušenie odberania zmien pri sledovanom parametri.

```
1 {
2   "command" : "SUBSCRIBE",
3   "properties" : [
4
5     "PWM0"
6   ]
7 }
8 {
9   "command" : "UNSUBSCRIBE",
10  "properties" : [
11    "PWM0"
12  ]
13 }
```

Výpis 3.4: Príklad prihlasovania a odhlasovania odoberania zmien sledovaných parametrov.

## Notify (Oznámenie)

Pomocou príkazu notify zasiela zariadenie informáciu o zmene parametru všetkým odberateľom sledovaného parametru. Štruktúra je zobrazená vo výpise 3.4.

```
1 {
2   "command" : "NOTIFY",
3   "values" : [
4     {
5       "property" : "PWM0",
6       "value" : 1024
7     }
8   ]
9 }
```

Výpis 3.5: Príklad zaslania oznámenia každému odberateľovi.

### 3.4.2 Inicializácia

Pri odoslaní broadcastového datagramu obdrží každé zariadenie v sieti špecifický uvítací datagram ktorý je popísaný v sekcii 3.4. Odpoveď na tento datagram posiela každé zariadenie v sieti po každom prijatí špecifického datagramu. Týmto spôsobom je zabezpečená možnosť vstupu užívateľa do systému v ľubovolnom čase prípadne možnosť použitia viacerých riadiacich členov. Odpoveď obsahuje základné informácie o zariadení, ako je zobrazené vo výpise 3.6.

```
1 {
2   GET / HTTP/1.1
3   Host: 10.0.0.43:44575
4   Content-Lenght: 207
5   {
6     "command": "BROADCAST_REPLY",
7     "device-info": {
8       "address" : {
9         "ip" : "10.0.1.56",
10        "port": 1058
11      },
12      "class" : "MOT",
13      "firmware" : "F-0083",
14      "locked-by" : {
15        "type" : "address",
16        "ip" : "xx.xx.xx.xx",
17        "port" : 10286
18      }
19    }
20  }
21 }
```

Výpis 3.6: Štruktúra odpovede inicializácie.

Vo výpise 3.6 je uvedený formát odpovede zariadenia formou hypertextového transportného protokolu. V hlavičke sa nachádza veria použitého HTTP protokolu, cieľová IP adresa, číslo TCP portu a dĺžka odosielaného reťazca. Za hlavičkou sú uvedené parametre zariadenia formou JSON zápisu.

## 4 IMPLEMENTÁCIA NAVRHNUTEJ ARCHITEKTÚRY PRE ZARIADENIE RLIBS

Táto kapitola sa v prvej časti venuje testovaniu štandardu ethernet s ohľadom na hardwarovú funkcionálnosť na základe čoho bolo zostrojené finálne zapojenie ktoré je následne podrobne popísané. Ďalšia časť sa venuje programovacím nástrojom na správu a ladenie kódu, ďalej výberom vhodných programových balíčkov na vývoj firmwaru. Posledná časť sa zaoberá popisom firmwaru pre zhotovené testovacie zariadenie.

### 4.1 Testovanie štandardu ethernet

Pri testovaní komunikácie cez štandard ethernet bol použitý vývojový kit od firmy STMicroelectronics typu STM32f4 discovery, ktorý bol prepojený rozhraním RMI s prevodníkom typu DP83848CVV. Pre testovanie bol použitý balík testovacích knižníc od výrobcu STMicroelectronics pre prácu so štandardom Ethernet [29]. Zdrojový kód sa nachádza v prílohe v adresári Firmware\_test a cieľom testovania bolo overiť použitie vybraných komponentov na tvorbu výslednej konfigurácie.

V prvej časti testu bola testovaná aplikačná vrstva pomocou protokolu HTTP, kde mikrokontrolér predstavoval server z IP adresou 192.168.0.10 a bol spojený prostredníctvom routera z notebookom. Pakety boli zachytávané pomocou programu Wireshark kde bol prenos analyzovaný a následne zrekonštruovaný.

V druhej časti testovania bola otestovaná transportná vrstva protokolom TCP. Mikrokontrolér s IP adresou 192.168.0.10 predstavoval server a čakal na spojenie na porte číslo 7. V zariadení klienta bola testovaná dostupnosť portu pomocou aplikácie echotool.exe. Parametre testovania v príkazovom riadku boli nasledovné :

```
echotool 192.168.0.10 /p tcp /r 7 /n 10 /t 2 /d Testovanie.
```

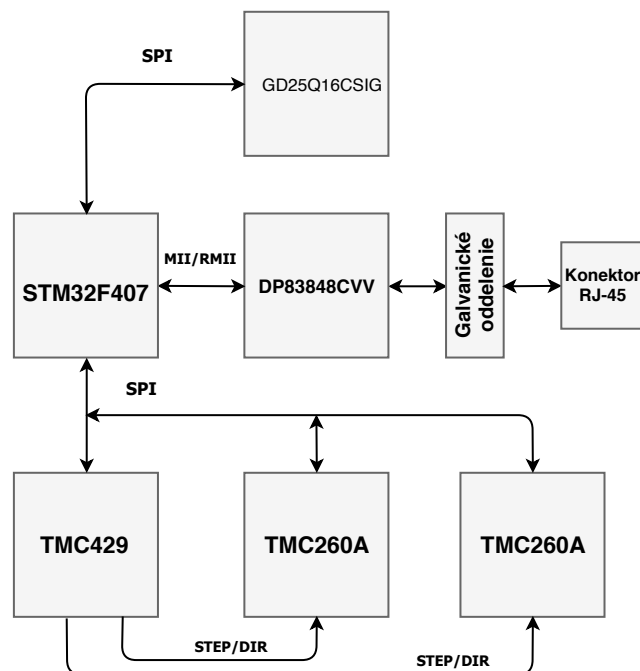
kde : /p predstavuje typ testovaného protokolu, /r predstavuje číslo portu, /n predstavuje počet priebehov testu, /t predstavuje maximálny čas na pripojenie a /d predstavuje správu posielanú serveru.

Server odpovedal na všetkých 10 testov z priemernou dobou trvania odpovede 1 ms. Použitie navrhnutých komponentov sa ukázalo ako funkčné a bola zahájená práca na návrhu robustnej komunikačnej architektúry pre zariadenie rLIBS pomocou štandardu Ethernet.

## 4.2 Vývoj riadiacej elektroniky

Pri vývoji riadiacej elektroniky bol kladený dôraz na spoľahlivosť a odolnosť voči rušeniu pri prevádzke zariadenia. Na otestovanie architektúry bol zvolený náhľadový modul ktorý obsahuje 2 krokové motory, bistabilnú závierku, osvetľovací segment a náhľadovú kameru z objektívom. Požiadavky na návrh DPS teda zahrňovali ovládanie dvoch krokových motorov, spínanie napájania pre osvetľovací segment a výstup signálových pinov na prepojenie riadiacej dosky s doskou riadenia závierky.

Zapojenie bolo navrhnuté v programe Eagle 8 a vychádza z blokovej schémy ktorá je zobrazená na obrázku 4.1. Podrobná schéma sa nachádza v adresári pdf s označením V-02773-01.pdf a podklady k výrobe sa nachádzajú v adresári Board.

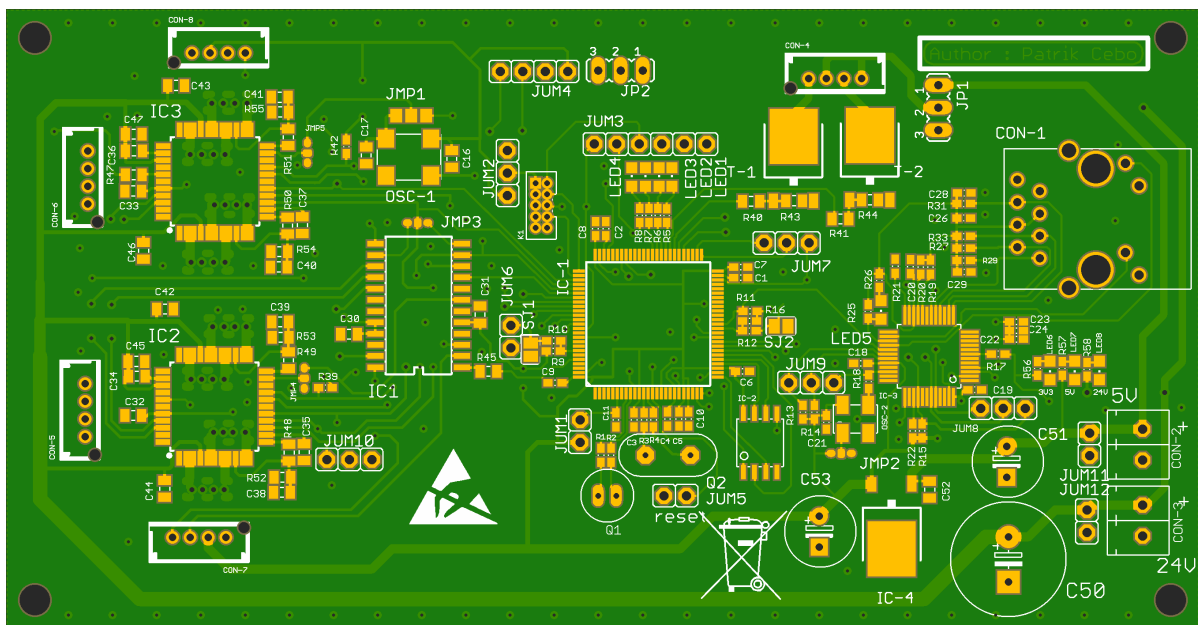


Obr. 4.1: Bloková schéma riadiacej elektroniky náhľadového systému.

Na riadiacej doske sú umiestnené dve svorkovnice ktoré slúžia na prívod napätia. Pomocou prvej svorkovnice je privádzané napätie 4-10 V, ktoré je po vyfiltrovaní privedené na lineárny stabilizátor, ktorého výstup o veľkosti 3,3 V slúži na napájanie všetkých integrovaných obvodov na DPS. Druhý konektor slúži na napájanie cievok krokových motorov, kde bolo zvolené doporučené napätie 24 V. Riadenie celého systému zabezpečuje mikrokontrolér architektúry ARM s jadrom M4 typu

STM32F407VGT6. Zapojenie ethernetového prevodníka vychádza z blokovej schémy ktorá je uvedená na obrázku 3.7. Prevodník DP83848CVV je prepojený z riadiacim mikrokontrolérom pomocou rozhrania RMI. Ovládanie krokových motorov zabezpečuje generátor impulzov a smeru otáčania TMC429 ktorý riadi výkonové budiče TMC260A. Komunikácia z generátorom je realizovaná cez rozhranie SPI ktoré je pripojené k riadiacemu mikrokontroléru. Na spínanie osvetlovacieho segmentu boli použité výkonové tranzistory typu MOSFET s kanálom N ktoré spínajú zem pripojeného segmentu osvetlenia. Riadiaca doska ďalej obsahuje flash pamäť pripojenú na samostatnú zbernicu SPI ktorá slúži na prípadné rozšírenie programovej pamäte. Programovanie mikrokontroléra je možné cez programovací SWD konektor, JTAG konektor poprípade cez USART konektor.

Doska plošného spoja má rozmery 135 x 70 mm a obsahuje diery veľkosti 3.4 mm na pripevnenie k distančným stĺpikom ako je zobrazené na obrázku 4.2.



Obr. 4.2: DPS riadiacej elektroniky.

Výstup z výkonových budičov je privedený na konektory typu JST z rastrom 2 mm, ktoré sú štandardizované pre krokové motory. Konektor na pripojenie referenčných snímačov je takisto typu JST rovnako ako konektor na pripojenie osvetlovacieho segmentu.

## 4.3 Programovanie

Pre vývoj aplikácií platformy ARM existuje veľký počet komerčne dostupných nástrojov na písanie kódu a ladenie. Každý z nástrojov používa inú štruktúru ukladania údajov a výsledné projekty sú často medzi sebou nekompatibilné. Existuje však celá rada OpenSource nástrojov, ktoré umožňujú kompilovať, ladiť a simulovať programový kód embeded zariadenia. V rámci vývoja bola zvolená nezávislosť na platforme operačného systému a boli použité nástroje ktoré sú určené pre všetky platformy.

### Qt Creator

Qt Creator je nástroj na správu a vývoj kódu. Obsahuje možnosti písania aplikácií od operačných systémov až po nízkoúrovňové embeded systémy. Tento nástroj umožňuje nastavenie ľubovlného kompilátoru a ladiaceho nástroju. Pre vývoj bol použitý arm none eabi kompilátor, debugovací nástroj openocd a nástroj stlink na programovanie mikrokontroléru.

### Kompilátor

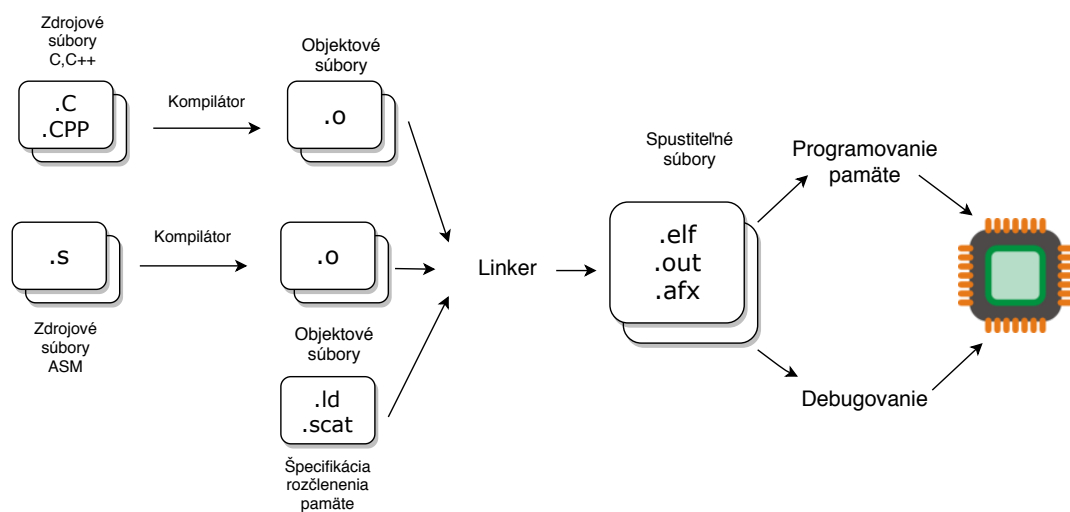
Kompilátor arm none eabi umožňuje kompilovanie projektov pre mikrokontroléri s architektúrou ARM cortex M0-M9. Celý nástroj je opensource pričom je súčasťou platformy GNU a je možné ho ďalej vyvíjať alebo reportovať nájdené chyby. Pre kompiláciu sa využíva Makefile ktorý umožňuje nalinkovanie všetkých potrebných knižníc a scriptov.

### Ladenie

Ladenie písaného kódu je užitočná vlastnosť zvlášť pri ladení zložitejších algoritmov. Qt Creator podporuje štandardný opensourcový ladiaci nástroj GDB ( z angl. GNU Debugger) ktorý spolupracuje so všetkými bežnými programovacími jazykmi ako sú: C, C++, Java, Go, Pascal a mnohé ďalšie, na platormách Windows a Unix [31]. Pri programovaní a ladenie embeded systémov nemáme na výber z množstva nástrojov ako je to pri tvorbe programov pre operačné systémy. Voľba padla na nástroj OpenOCD (z angl. Open On-Chip Debugger) ktorý napísal Dominic Rath vrámci diplomovej práce a naďalej ho vyvíjal. Tento nástroj podporuje nahrávanie, mazanie a ladení firmwaru pre najčastejšie používané integrované obvody. Samotný nástroj funguje iba ako prekladač medzi inštrukciami z GDB a driverom použitého programátora [32]. Pre programovanie a ladenie mikrokontroléru značky STM bolo použité zariadenie ST-Link/v2 od rovnakého výrobcu ako mikrokontrolér. Zariadenie dokáže pracovať zo štandartmi JTAG aj SWD, čo umožňuje rýchly a spoľahlivý zápis do programovej pamäte.

## Makefile

Makefile je nástroj pre vytváranie spustiteľných súborov a je nezávislý na platforme operačného systému. Pokiaľ teda píšeme súbor makefile pod GNU štandardom, máme zaručenú prenositeľnosť medzi operačnými systémami. Makefile pre embedded systémy je obdobou klasických aplikácií avšak vyžaduje si omnoho viac pravidiel pre linkovanie súborov. Proces kompilácie je znázornený na obrázku 4.3. Po preklade zdrojových súborov C, C++ a ASM na objektové súbory vstupujú súbory spolu s linkovacím skriptom do linkéru, ktorý vytvorí výsledný spustiteľný súbor [33]. Linkovací skript obsahuje údaje o verzii a type jadra použitého mikrokontroléru, ďalej obsahuje veľkosť programovej pamäte a adresy pre zápis do pamäte flash.



Obr. 4.3: Proces kompilácie

Pomocou tohoto nástroja bolo realizované napalovanie a ladenie mikrokontroléra prostredníctvom vlastne definovaných makier (úloh). Okrem bežných úloh ako sú all, clean a .PHONY bol súbor rozšírený o úlohy release, debug, flash a debug\_start. Úloha flash zavolá podúlohu ktorá skompiluje program s predom definovanou optimalizáciou a následne zapíše program do flash pamäti mikrokontroléra pomocou nástroja OpenOCD. Úloha debug\_start zavolá opäť podúlohu, tentokrát debug ktorá má nastavenú minimálnu optimalizáciu kvôli debugovaciemu procesu a skompiluje výslednú aplikáciu. Po kompilácii sa program zapíše do programovej pamäte rovnako ako v úlohe flash a nasleduje spustenie debugéra OpenOCD. Všetky spustiteľné úlohy sa ľahko implementujú do programu Qt Creator a uľahčujú tak komunikáciu s mikroprocesorom. V programe Qt Creator je pri ladení potrebné nastaviť pripojenie na spustený ladiaci nástroj, ktorý komunikuje pomocou lokálneho TCP portu.

### 4.3.1 Nízkoúrovňová implementácia TCP/IP

LwIP (z angl. lightweight IP) je nízkoúrovňová implementácia TCP/IP modelu ktorá bola napísaná švédskym programátorom Adamom Dunkelsom. Celý TCP/IP model je napísaný v jazyku C a je prispôsobený na beh s malou programovou pamäťou. Pri implementácii do mikrokontroléra zaberá približne 10 kbit pamäte RAM a 40 kbit programovej pamäte.[34] Model obsahuje implementáciu nasledujúcich protokolov:

#### **IP vrstva:**

- **IP** Internet Protocol obsahuje IPv4 a IPv6
- **ICMP** Internet Control message Protocol
- **IGMP** Internet Group Management Protocol

#### **Sieťová vrstva:**

- **UDP** User Datagram Protocol
- **TCP** Transmission Control Protocol

#### **Aplikačná vrstva:**

- **DNS** Domain Name System
- **SNMP** Simple Network Management Protocol
- **DHCP** Dynamic Host Configuration Protocol
- **HTTP** Hypertext Transfer Protocol
- **MQTT** Message Queuing Telemetry Transport
- **TFTP** Trivial File Transfer Protocol
- **SMTP** Simple Mail Transfer Protocol
- **SNTP** Simple Network Time Protocol

LwIP model obsahuje tri spôsoby prístupu k sieťovému rozhraniu. Najnižšia úroveň tzv. RAW API je úroveň ktorá pracuje na sieťovej vrstve s protokolmi UDP a TCP. Táto úroveň nie je vhodná na prácu z operačným systémom reálneho času nakoľko je nízkoúrovňová. Vyššou úrovňou sú sekvenčné aplikácie tzv. Netconn Api a Socket Api, ktoré sú vhodné na spoluprácu s operačnými systémami. Netconn a Raw Api sú napísané iba pre LwIP model. Pokiaľ chceme napísať aplikáciu ktorá je prenositeľná medzi operačnými systémami sme nútení použiť Socket Api ktorá implementuje štandard BSD (z angl. Berkeley sockets) ktorý umožňuje medziprocesorovú komunikáciu.

**NETCONN API** Pri vývoji firmwaru s operačným systémom FreeRtos bol použitý práve tento spôsob prístupu k dátam. Netconn Api sú navrhnuté na uľahčenie práce s modelom a podporujú prácu s protokolmi TCP, UDP a RAW IP. Na rozdiel od RAW API vyžadujú pre beh spoluprácu s operačným systémom. Celá komunikácia je napísaná v špecifickom vlákne tzv. tcpip thread, ktoré pracuje na sieťovej vrstve modelu TCP/IP. Programátor má pre prácu k dispozícii nasledovné funkcie :

**netconn\_new\_with\_proto\_and\_callback()** ktorá slúži na vytvorenie nového spojenia

**netconn\_delete()** vymaže existujúce spojenie

**netconn\_bind** priradí k spojeniu port

**netconn\_connect** nadviaže spojenie z portom

**netconn\_listen** príma dáta z TCP

**netconn\_accept** akceptuje prichádzajúce spojenie

**netconn\_write** odošle dáta (TCP)

**netconn\_close** zatvorí aktuálne TCP spojenie

**netconn\_send** odošle dáta na pripojený port (UDP)

**netconn\_sendto** odošle dáta na ľubovolný port (UDP)

**netconn\_recv** príma dáta

### 4.3.2 Operačný systém reálneho času RTOS

Operačný systém reálneho času RTOS (z angl. Real-time operating system) je systém, ktorý zabezpečuje použitie v aplikáciách, kde je potrebné vykonávať úlohy v pevne stanovených časových intervaloch. Využitie týchto systémov si nájde uplatnenie najmä v priemysle napríklad v regulačnej a riadiacej technike. Typickým príkladom je riadiaci systém lietadla, kde je potrebné zaručiť odozvu na riadiaci impulz v krátkom a pevne definovanom maximálnom čase. Rozdelenie operačných systémov reálneho času je nasledovné :

- **Hard real time** je systém, kde je bezpodmienečne nutné dodržať časový interval. Nedodržanie časového intervalu môže viesť k nebezpečným a život ohrozujúcim situáciám.
- **Firm real time** je systém ktorý akceptuje občasné nedodržanie časového intervalu.
- **Soft real time** je systém kde nie je nutné dodržať časový interval a oneskorenie nespôsobí žiadny vážny následok.

Základný princíp operačných systémov reálneho času je rovnaký ako princíp činnosti bežných operačných systémov bežiacich napríklad na stolovom počítači. Princíp pozostáva v periodickom striedaní jednotlivých úloh podľa priradenej priority. Úlohy operačného systému môžu nadobudnúť tieto stavy :

- **Running** (alebo bežiaci) je stav systému, kedy je úloha vykonávaná na procesore pričom v jeden časový okamih môže byť vykonávaná práve jedna úloha. Úloha opustí vykonávanie buď sama, alebo nútene pomocou plánovača (z angl. scheduler).
- **Ready** (alebo pripravený) je stav kedy je úloha pripravená na vykonanie a čaká vo fronte spolu s ďalšími na vykonanie svojej činnosti na procesore. Jednotlivé úlohy sú zoradené vo fronte podľa pridelenej priority od najvyššej po najnižšiu.
- **Blocked** (alebo blokovaný) je stav kedy je úloha dočasne v režime blokovania a čaká na vstup dát. V tomto stave môže byť iba istý čas po ktorom uplynutí je automaticky presunutá do stavu pripravený (Ready).
- **Suspend** (alebo pozastavený) je stav ktorý môže nastať iba zavolaním príslušnej funkcie aplikačného rozhrania API.

Synchronizácia použitých úloh je realizovaná pomocou semaforov ktoré sa rozdeľujú na tri druhy. Prvým a najjednoduchším druhom je binárny semafor, ktorý môže nadobúdať vo fronte iba dve hodnoty, 0 a 1 ktoré indikujú dostupnosť úlohy. Binárne semaforey sú často riadené pomocou hlavnej úlohy. Ďalším druhom sú čítacie semaforey ktoré môžu nadobúdať viacero hodnôt a môžu byť viacnásobne prečítané. Posledným druhom sú semaforey so vzájomným vylúčením tzv. mutexy, ktoré sa používajú pri rekurzívnych volaniach. Okrem synchronizácie pomocou semaforov ktoré sú implementované priamo v jadre systému máme možnosť použiť časovače, ktoré sa rozdeľujú na dva druhy. Prvým druhom je jednorázový timer ktorý po uplynutí časového intervalu zavolá požadovanú úlohu práve jedenkrát. Druhým typom je automatický časovač ktorý je po uplynutí času automaticky reštartovaný a môže byť použitý na periodické volanie úloh.

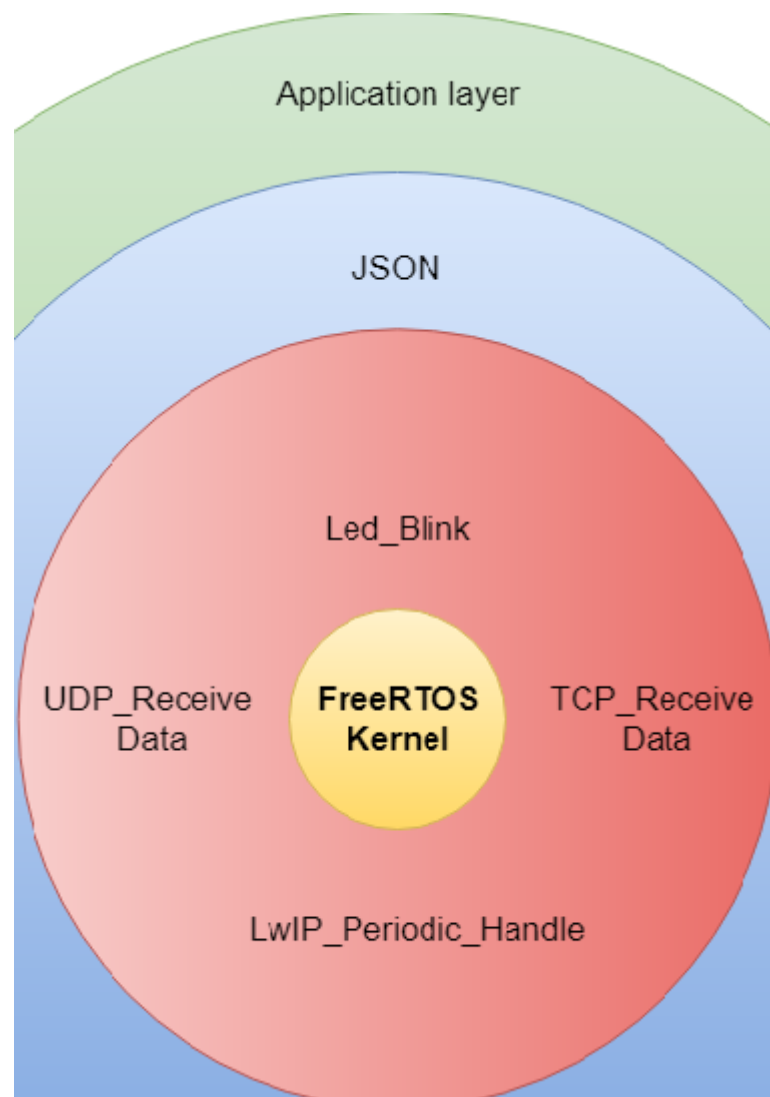
Komunikácia medzi úlohami je realizovaná pomocou zdieľanej pamäte a využívajú sa prevažne fronty. Pokiaľ je potrebná výmena dát medzi jednotlivými úlohami je potrebné alokovať potrebné množstvo v pamäti. Dáta sa medzi úlohami vymieňajú výhradne pomocou správ ktoré sú vkladané do fronty FIFO (z angl. First In First Out). Odosielané môžu byť jednoduché dátové typy ale aj celé štruktúry, záleží iba na veľkosti dostupnej pamäti.

Operačných systémov reálneho času je mnoho a stále sa pracuje na ich rozvoji. Pre použitie s mikrokontrolérom s obmedzenou programovacou pamäťou a nárokmi na open source bol zvolený operačný systém FreeRTOS, ktorý je voľne šíriteľný na komerčné účely pod licenciou MIT. Tento operačný systém je distribuovaný formou zdrojových súborov, ktoré obsahujú veľké množstvo demonštračných aplikácií. Operačný systém umožňuje vytvoriť ľubovoľný počet úloh a je závislý iba na veľkosti pamäte RAM . Systém FreeRTOS navyše podporuje ochranu pamäte MPU (z angl. Memory Protection Unit) vďaka čomu nemôžeme alokovať viac pamäte ako je dostupné a zabráni sa tým pádu aplikácie [35]. Systém bol napísaný v jazyku C no pre naše účely bol začlenený do knižnic C++ aby bolo možné lepšie pracovať s parsovaním dát ktoré požaduje JSON knižnica.

Pri programovaní bol využitý binárny semafor a jednotlivé časti kódu boli rozdelené do úloh. Medzi základne úlohy ktoré sa pravidelne striedajú na procesore patrí príjem dát na sieťovej vrstve TCP/IP štandardu. Pre každý z protokolov na tejto vrstve bola vytvorená samostatná úloha ktorá kontroluje príjem dát na predom definovanom porte. Pokiaľ je prijatý datagram alebo segment, vytvorí sa nová úloha s vyššou prioritou ktorá spracuje vstupné dáta a vykoná potrebné úkony spojené s riadením systému.

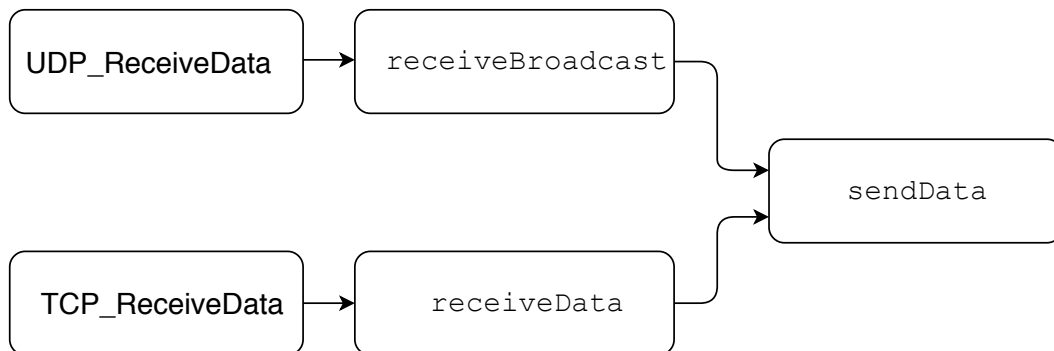
### 4.3.3 Vývoj firmwaru

Pri vývoji firmwaru bola použitá nízkoúrovňová nadstavba knižníc CMSIS (z angl. Cortex Microcontroller Software Interface Standard) od výrobcu mikrokontroléru s názvom HAL (z angl. Hardware Abstraction Layer). Knižnice HAL predstavujú aplikačnú úroveň riadenia mikrokontroléra a nie je potrebné pristupovať k registrom ako je to napr. pri programovaní mikrokontrolérov platformi ARM. Firmware mikrokontroléru bol písaný v jazyku C++, čo vyžadovalo abstrakciu všetkých programových balíkov písaných v jazyku C. Architektúru výsledného firmwaru je možné jednoducho popísať pomocou vrstvomého prístupu, pozri obrázok 4.4.



Obr. 4.4: Štruktúra firmwaru.

Po prvotnej inicializácii mikrokontroléru a skoku do programovej pamäte ktorá odkazuje na výslednú aplikáciu sa spustí jadro operačného systému FreeRTOS. Po štarte jadra sa vytvorí štartovacie vlákno, ktoré inicializuje LwIP stack súčasťou ktorého je inicializácia ethernetového prevodníka DP83848CVV. Pri tejto inicializácii je spustený DHCP klient, ktorý si vyžiada IP adresu zo siete. V štartovacom vlákne sú ďalej vytvorené 3 vlákna ako je uvedené v druhej vrstve na obrázku 4.4. Štvrté vlákno je vlákno LwIP stacku a obsluhuje všetku prichádzajúcu a odchádzajúcu komunikáciu. Vlákno s názvom UDP\_ReceiveData slúži na príjem datagramov na predom definovanom porte, naopak vlákno TCP\_ReceiveData slúži na príjem segmentov na inom porte. Posledné vlákno s názvom Led\_Blink signalizuje beh operačného systému. Príjem dát protokolu UDP je zobrazený vo výpise 4.1. Po prijatí datagramu sa zavolá metóda s názvom ReceiveBroadcast ktorá je súčasťou vyššej vrstvy JSON, pozri obr. 4.4. V tejto metóde sa skontroluje špecifický reťazec a v prípade zhody sa zavolá funkcia sendData, ktorá vytvorí nové vlákno a odošle informácie o svojej konfigurácii. Po odoslaní je toto vlákno automaticky zmazané. Rovnako ako príjem dát protokolom UDP je realizovaný príjem segmentov protokolu TCP. Pri komunikácii pomocou protokolu TCP je na rozdiel od UDP potrebné nadviazanie spojenia a jeho následné ukončenie. Štruktúra implementovaných funkcií je zobrazená na obrázku 4.5. Výsledný firmware sa nachádza v prílohách v adresári Firmware.



Obr. 4.5: Stavový diagram komunikácie.

```

1 void UDP_ReceiveData(const void* argument)
2 {
3     err_t err, recv_err, accept_err;
4     static struct netconn *conn, *newconn;
5     static struct netbuf* buf;
6     ip_addr_t* addr;
7     Identification hostID;
8     uint16_t port;
9     for (;;) {
10        LWIP_UNUSED_ARG(argument);
11        conn = netconn_new(NETCONN_UDP);
12        if (conn != NULL) {
13            err = netconn_bind(conn, IP_ADDR_ANY, 1682);
14            if (err == ERR_OK) {
15                while (1) {
16                    recv_err = netconn_recv(conn, &buf);
17                    if (recv_err == ERR_OK) {
18                        addr = netbuf_fromaddr(buf);
19                        port = netbuf_fromport(buf);
20                        MyArmSDD.init_sdd();
21                        char* data = (char*)buf->ptr->payload;
22                        hostID.ipv4_32 = addr->addr;
23                        hostID.port = port;
24                        hostID.version = IPVer::IPv4;
25                        std::string* text = new std::string(data);
26                        HAL_GPIO_TogglePin(led_4_GPIO_Port, led_4_Pin);
27                        text->erase(std::remove(text->begin(),
28                            text->end(), '\n'), text->end());
29                        MyArmSDD.mySDD.receiveBroadcast(hostID,
30                            MyArmSDD.myId, *text);
31                        netconn_delete(conn);
32                        netbuf_delete(buf);
33                        delete (text);
34                    }
35                }
36            } else netconn_delete(conn);
37        } else printf("can create new UDP netconn");
38        vTaskDelay(4);
39    }
40 }

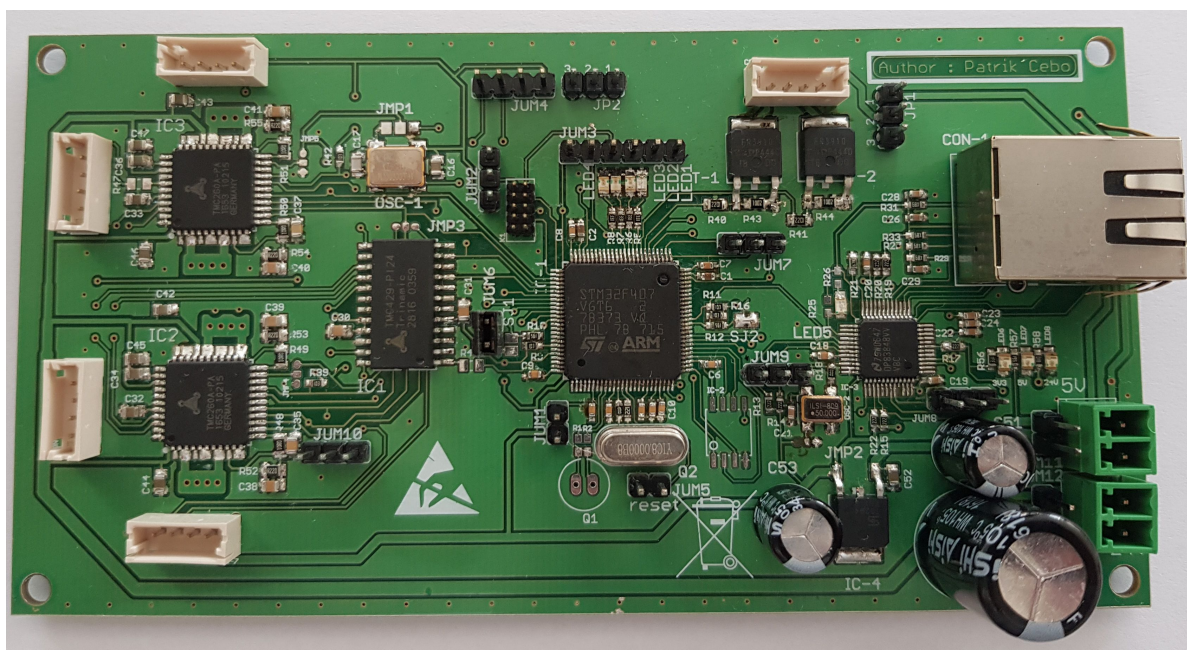
```

Výpis 4.1: Příjem dat protokolu UDP.

## 5 TESTOVANIE

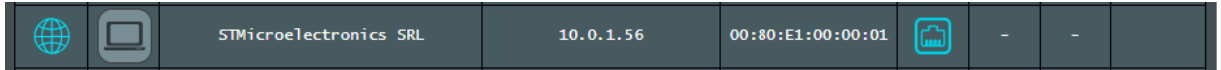
V kapitole testovanie je podrobne popísaný proces oživovania a testovania zhotoveného testovacieho zariadenia. V prvej časti je uvedený proces otestovania funkcie zariadenia, následne je uvedený spôsob testovania dostupnosti prípravku na sieti a v poslednej časti je ukážka zachytávania komunikácie medzi riadiacim členom a testovacím prípravkom.

V prvom kroku po osadení súčiastok na DPS bola vykonaná skúška odberu prúdu, aby sa zistil kludový odber a detekoval sa tak prípadný problém alebo skrat na zariadení. Kludový odber predstavoval 150 mA čo bolo pod hranicou očakávania, pričom prítomnosť napájacieho napätia signalizovali 3 LED diódy umiestnené v pravej časti DPS pod konektorom RJ-45. Lavá dióda značí prítomnosť stabilizovaného napätia veľkosti 3.3 V, stredná prítomnosť napájacieho napätia a pravá dióda prítomnosť napätia pre krokové motory. Následne bol do mikrokontroléra napálený firmware pomocou rozhrania SWD a programátora ST-Link/v2. Testovacie zariadenie je zobrazené na obrázku 5.1.



Obr. 5.1: Testovacie zariadenie.

Testovací firmware s operačním systémem obsahoval v prvej fázy iba výcho- diskové vlákno, ktoré inicializovalo prevodník DP83848CVV, LwIP stack a spustil dhcp server. Následne bola odskúšaná komunikácia z ethernetovým prevodníkom pomocou ladiaceho nástroja v programe Qt Creator. Ukážka ladenia v programe Qt Creator je uvedená v prílohe č.1 na obrázku p.1. Po inicializácii prevodníku bola overená funkcia dhcp servera, pridelená IP adresa je zobrazená na obrázku 5.2



Obr. 5.2: Pridelená IP adresa.

Po pridelení IP adresy nasledoval test dostupnosti pomocou programu Ping s využitím protokolu ICMP. Prvý test prebehol zo 100% úspešnosťou ako je zobrazené na obrázku 5.3. V druhom teste bola testovaná dlhodobá dostupnosť zariadenia kde bol program ping spustený po dobu 1 hodiny a výsledkom tohoto experimentu bola 6% stratovosť paketov.

A screenshot of a terminal window titled "patrik@Patres:~". The terminal shows the output of a ping command: [patrik@Patres ~]\$ ping 10.0.1.56. The output displays 9 successful ping attempts with various response times. At the end, it shows ping statistics: 9 packets transmitted, 9 received, 0% packet loss, time 8010ms, and rtt min/avg/max/mdev = 0.898/1.199/2.712/0.569 ms. The prompt [patrik@Patres ~]\$ is visible at the bottom.

Obr. 5.3: Test dostupnosti zariadenia.

Ďalšie testovanie prebiehalo v spolupráci s demonštračným, programom ktorý simuloval riadiaci člen a s programom Wireshark, ktorý slúži na zachytávanie komunikácie sieťového rozhrania. V tomto teste bol do zariadenia nahraný finálny firmware.

V prvom kroku bola zachytávaná komunikácia nadviazania spojenia ako je popísané v sekcii 3.4. Riadiaci člen poslal na broadcastovú adresu uvítací datagram ktorého štruktúra je uvedená vo výpise 5.1. Zachytený UDP datagram v programe Wireshark je zobrazený v prílohe č.2 na obrázku p.2

```
1 {
2     z(Prh3u*PfQ29HXqTGqs8LL&jJLaa4(vo3hPAJ%5twBAUnch
3     TuQKH$iTGw9jT{"ip":"10.0.1.60","ipver":4,"port":43157}
4 }
```

Výpis 5.1: Uvítací datagram.

V prípade prijatia datagramu nasledovalo zachytenie odpovede, ktorá využíva HTTP protokol. Zariadenie odpovedalo pomocou správy BroadcastReply na TCP port, ktorý je obsiahnutý spolu z IP adresou v uvítacom datagrame. Štruktúra odpovede je zobrazená vo výpise 3.6 v sekcii 3.4.2. Zachytený segment na TCP vrstve je zobrazený v prílohe č.3 na obrázku p.3.

Po úspešnej výmene informácií medzi demonštračným programom a testovacím zariadením nasledovala požiadavka na zaslание špecifických vlastností zariadenia (Property) formou odoslania HELLO príkazu. Štruktúra príkazu je uvedená vo výpise 5.2. Zachytený TCP segment je zobrazený v prílohe č.4 na obrázku p.4.

```
1 {
2     GET / HTTP/1.1
3     Host: 10.0.0.60:46229
4     Content-Lenght: 116
5     {
6     "command": "HELLO",
7     "from": {
8         "ip" : "10.0.1.56",
9         "ipver" : 4,
10        "port": 56698
11    }
12 }
13 }
```

Výpis 5.2: Dotazovanie sa na vlastnosti.

Odpoveď zariadenia na príkaz HELLO pozostáva zo špecifických vlastností ako je uvedené v sekcii 3.4.1 a štruktúra odpovede je zobrazená vo výpise 5.3. Zrekonštruovaný TCP segment je zobrazený v prílohe č.5 na obrázku p.5.

```
1
2 GET / HTTP/1.1
3 Host: 10.0.0.60:46229
4 Content-Lenght: 991
5 {
6   "command": "WELCOME",
7   "from": {
8     "ip" : "10.0.1.56",
9     "ipver" : 4,
10    "port": 56698
11  },
12  "properties": {
13    "PWM0" : {"min" : 0, "max" : 4095, "value" : 0},
14    "PWM1" : {"min" : 0, "max" : 4095, "value" : 0},
15    .
16    .
17    "PWM7" : {"min" : 0, "max" : 4095, "value" : 200}
18  }
19 }
```

Výpis 5.3: Zaslanie špecifických vlastností.

Cieľom testovania bola demonštrácia základných funkcií navrhnutého prototypového zariadenia a komunikačného protokolu. Počas testovania bola overená základná funkcia nadviazania komunikácie a výmeny špecifických vlastností. Posledným testom bolo testovanie riadenia krokových motorov pomocou sledov periodicky sa meniacich metód ako je uvedené v prílohe v adresári Firmware\_MOT. Pri týchto testoch bola overená funkcia budičov krokových motorov TMC260 v spolupráci s riadiacim členom TMC429. Test bol vykonávaný na krokovom motore typu SX16-0402N od výrobcu Microcon.

## ZÁVER

Cielom tejto práce bolo popísať aktuálny stav architektúry zariadenia rLIBS a pojednať o jeho kritických bodoch s ohľadom na chybovosť komunikačnej zbernice. Ďalej bolo potrebné navrhnúť riešenie na zvýšenie odolnosti komunikačnej architektúry s ohľadom na použitie moderných technológií.

V prvej fáze bolo analyzované aktuálne elektrické zapojenie prístroja a bola vytvorená bloková schéma zapojenia celej elektroniky. V blokovej schéme sa nachádza rozpis všetkých použitých elektronických prvkov a informácie o ich vzájomnom prepojení. Následne bola testovaná chybovosť pri prenose dát, s ktorej vyplynulo, že je potrebné vynechať štandard USB a nahradiť ho odolnejším štandardom, akým je štandard ethernet.

V druhej fáze bolo potrebné navrhnúť nový typ architektúry a zjednotiť všetky použité zbernice. Vynechanie aktuálne používanej zbernice RS485 vyžadovalo nahradiť drivre na ovládanie krokových motorov a tak bolo navrhnuté a testované zapojenie, ktoré je súčasťou výsledného prototypového zariadenia.

V ďalšej fáze bol navrhnutý, popísaný a implementovaný komunikačný protokol s ohľadom použitia na zbernici ethernet. Tento protokol je ľahko škálovateľný a je určený na univerzálne použitie v embeded systémoch. Po otestovaní základných funkcií riadenia krokových motorov na zhotovenom prípravku a otestovaní ethernetového prevodníka s vývojovým kitom bola započatá práca na navrhnutí a zostrojení finálneho prototypového zariadenia s ohľadom na odolnosť voči rušeniu. V práci je popísaná základná funkcionálna navrhnutého zariadenia. Počas vývoja zariadenia bol vyvíjaný firmware pre výslednú aplikáciu, kde bol sčasti implementovaný navrhnutý komunikačný protokol.

Posledná časť práce sa zaoberá testovaním zariadenia. Pri testoch bola overená základná funkcionálna zariadenia akou je komunikácia so sieťovým prevodníkom a s riadiacim členom budičov krokových motorov.

Výsledkom práce je funkčné testovacie zariadenie, ktoré rieši problémy zvýšenia komunikačnej architektúry pre prístroj diaľkovej laserovej spektroskopie rLIBS. Toto zariadenie bude ďalej vyvíjané a postupne začleňované do prístroja rLIBS kde nahradí aktuálne použité riešenie elektroniky.

# LITERATÚRA

- [1] DUTOUQUET, Christophe. *LIBS Detection of Nanomaterials for Process Control and in the Workplace* [online]. 2015, 24-33 [cit. 2017-10-5]. Dostupné z: <http://www.spectroscopyonline.com/libs-detection-nanomaterials-process-control-and-workplace>
- [2] FORTES, F.J. a J.J. LASERNA. The development of fieldable laser-induced breakdown spectrometer: No limits on the horizon. *Spectrochimica Acta Part B-Atomic Spectroscopy*. 2010, , 975-990. DOI: 10.1016/j.sab.2010.11.009. ISBN 10.1016/j.sab.2010.11.009. Dostupné z: <http://linkinghub.elsevier.com/retrieve/pii/S058485471000296X>
- [3] NOVOTNÝ, Jan. *Dálkově řízená laserová spektroskopie (LIBS)*. Brno, 2012, 97 s. Dizertační práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství. Vedoucí práce Doc. Ing. Jozef Kaiser, Ph.D.
- [4] NACHON, M., N. MANGOLD, O. FORNI, et al. Chemistry of diagenetic features analyzed by ChemCam at Pahrump Hills, Gale crater, Mars. *Icarus*. 2017, 121-136. DOI: 10.1016/j.icarus.2016.08.026. ISBN 10.1016/j.icarus.2016.08.026. Dostupné z: <http://linkinghub.elsevier.com/retrieve/pii/S0019103516305346>
- [5] SKEEZE. Mars Rover. In: *PIXABAY* [online]. USA [cit. 2017-10-11]. Dostupné z: <https://pixabay.com/sk/mars-rover-zvedavos%C5%A5-kozmic%C3%A9-lety-1241266/>
- [6] CHEMCAM. In: *CNES* [online]. [cit. 2017-10-11]. Dostupné z: [https://msl-curiosity.cnes.fr/en/MSL/GP\\_chemcam.htm](https://msl-curiosity.cnes.fr/en/MSL/GP_chemcam.htm)
- [7] BRADA, Michal. *Polohovací jednotka pro laserovou spektroskopii*. Brno, 2012, 74 s. Diplomová práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství. Vedoucí práce Ing. Daniel Koutný, Ph.D.
- [8] APPLIED PHOTONICS. *Stand-off LIBS for security and protection* [online]. In: . [cit. 2017-10-11]. Dostupné z: [http://www.appliedphotonics.co.uk/products\\_services/standoff\\_libs.htm](http://www.appliedphotonics.co.uk/products_services/standoff_libs.htm)
- [9] MOBILNÍ DÁLKOVÁ LABORATOŘ OCENĚNA ZLATOU MEDAILÍ NA MSV 2014. In: *VUT* [online]. Brno [cit. 2017-11-5]. Dostupné z: <https://www.vutbr.cz/f19527/d93788>

- [10] ČSN 33 0010 ED. 2. *Elektrická zařízení - Rozdělení a pojmy*. Česká republika: Úřad pro technickou normalizaci, metrologii a státní zkušebnictví, 2014.
- [11] TRINAMIC. *TMCM-1021 V1.4 Hardware Manual*. 2014. Dostupné z: <https://www.trinamic.com/products/modules/details/tmcm-1021/>
- [12] TRINAMIC. *TMCM-1161 V1.0 Hardware Manual*. 2013. Dostupné z: <https://www.trinamic.com/>
- [13] AXELSON, Jan. *Serial Port Complete: COM Ports, USB Virtual COM Ports, and Ports for Embedded Systems (Complete Guides series)*. Lakeview Research, 2007. ISBN 9781931448062.
- [14] TEXAS INSTRUMENTS. *Interface Circuits for TIA/EIA-485(RS-485)*,. 2007. Dostupné z: <http://www.ti.com/>
- [15] COMPAQ, INTEL, MICROSOFT a NEC. *Universal Serial Bus Specification: Revision 1.1*. . 1998, 372.
- [16] Compaq, Hewlett-Packard, Intel, Lucent, Microsoft, NEC a Philips. *Universal Serial Bus Specification: Revision 2.0*. 2000, 650.
- [17] Hewlett-Packard Company, Intel, Microsoft, NEC, ST-NXP Wireless a Texas Instruments. *Universal Serial Bus 3.0 Specification: Revision 1.0*. 2008, 482.
- [18] MILOS634. *Comparison of USB connectors* [online]. In: . [cit. 2017-10-12]. Dostupné z: [https://upload.wikimedia.org/wikipedia/commons/c/c2/USB\\_konektory.png](https://upload.wikimedia.org/wikipedia/commons/c/c2/USB_konektory.png)
- [19] TRINAMIC. *TMC260/ TMC260A & TMC261 DATASHEET: Rev. 2.10*. Hamburg, Germany, 2016. Dostupné z: <https://www.trinamic.com/products/integrated-circuits/details/tmc260a-pa/>
- [20] TRINAMIC. *TMC262 DATASHEET: V2.14*. Hamburg, Germany, 2016. Dostupné z: <https://www.trinamic.com/products/integrated-circuits/details/tmc262-1a/>
- [21] TRINAMIC. *TMC429 DATASHEET: Rev. 2.04*. Hamburg, Germany, 2015. Dostupné z: <https://www.trinamic.com/products/integrated-circuits/details/tmc429/>
- [22] TEXAS INSTRUMENTS. *Introduction to the Controller Area Network (CAN): SLOA101B*. 2002. Dostupné z: <http://www.ti.com/apps/docs/litabs/multiplefilelist.tsp?literatureNumber=sloa101b&docCategoryId=1&appId=209>

- [23] TEXAS INSTRUMENTS. *ControllerAreaNetworkPhysicalLayerRequirements: SLLA270*. 2008. Dostupné z: <http://www.ti.com/general/docs/litabsmultiplefilelist.tsp?literatureNumber=slla270>
- [24] ABRAMSON, Norman. *THE ALOHA SYSTEM-Another alternative for computer communications*. Honolulu, Hawaii, 1970. University of Hawaii.
- [25] KIRRMANN, H., K. WEBER, O. KLEINEBERG a H. WEIBEL. HSR: Zero recovery time and low-cost redundancy for Industrial Ethernet (High availability seamless redundancy, IEC 62439-3). *2009 IEEE Conference on Emerging Technologies & Factory Automation*. 2009, , 1-4. ISSN 1946-0740. Dostupné z: <http://ieeexplore.ieee.org/document/5347037/citations?part=1>
- [26] 802.3-2015. *IEEE Standard for Ethernet*. IEEE, 2016, 1-4017. Dostupné z: <http://ieeexplore.ieee.org/document/7428776/>
- [27] STMICROELECTRONICS. *STM32F405xx STM32F407xx: DocID022152 Rev 8*. 2016. Dostupné z: <http://www.st.com/en/microcontrollers/stm32f407vg.html>
- [28] TEXAS INSTRUMENTS. *DP83848C/I/VYB/YB PHYTER QFP Single Port 10/100 Mb/s Ethernet Physical Layer Transceiver: SNLS266E*. REV. E. 2007. Dostupné z: [http://www.ti.com/product/DP83848C/datasheet/application\\_implementation\\_and\\_layout](http://www.ti.com/product/DP83848C/datasheet/application_implementation_and_layout)
- [29] STMICROELECTRONICS. *LwIP TCP/IP stack demonstration for STM32F4x7 microcontrollers: AN3966*. 2013. Dostupné z: <http://www.st.com/en/embedded-software/stsw-stm32070.html>
- [30] JSON. *JSON* [online]. [cit. 2018-04-15]. Dostupné z: <https://www.json.org/>
- [31] GDB: The GNU Project Debugger. *GNU* [online]. [cit. 2018-04-19]. Dostupné z: <https://www.gnu.org/software/gdb/>
- [32] *Open On-Chip Debugging* [online]. [cit. 2018-04-22]. Dostupné z: <http://openocd.org/>
- [33] YIU, Joseph. *The Definitive Guide to ARM® Cortex®-M3 and Cortex®-M4 Processors*. 3rd Edition. Newnes, 2013. ISBN 9780124080829.
- [34] *LWIP* [online]. [cit. 2018-04-22]. Dostupné z: [http://www.nongnu.org/lwip/2\\_0\\_x/index.html](http://www.nongnu.org/lwip/2_0_x/index.html)
- [35] *FreeRTOS* [online]. [cit. 2018-04-23]. Dostupné z: <https://www.freertos.org/>

# ZOZNAM SYMBOLOV, VELIČÍN A SKRATIEK

LIBS	Spektroskopia laserom budenej plazmy - Laser-Induced Breakdown Spectroscopy
rLIBS	Dialková spektroskopia laserom budenej plazmy - Remote Laser-Induced Breakdown Spectroscopy
CCD	Nábojovo viazaná štruktúra - charge-coupled device
ST-LIBS	Produktové označenie výrobku firmy Applied Photonics
ChemCam	Modul pre materiálovú analýzu zo snímokovaním - Chemistry and Camera
3D	Trojrozmerný priestor - Three-dimensional space
ILC	Bezpečnostná zbernica - Interlock
USB	Univerzálna sériová zbernica - Universal Serial Bus
RMS	Kvadratický priemer - root mean square
DPS	Doska plošných spojov
PCB	Doska plošných spojov - Printed circuit board
RxD	Príjem dát - Receive Data
TxD	Vysielanie dát - Transmit Data
LOG	Logická úroveň signálu
NRZI	Invertovaný kód bez návratu k nule - Non Return to Zero Inverted
SPI	Sériové periférne rozhranie - Serial Peripheral Interface
QFN	Štvorcové ploché bez-pinové púzdro - Quad Flat No-leads package
CAN	Zbernica - Controller Area Network
CANH	Vysoký signál zbernice CAN - Controller Area Network High
CANL	Nízky signál zbernice CAN - Controller Area Network Low
ISO	Medzinárodná organizácia pre normalizáciu - International Organization for Standardization
HTTP	Hypertextový prenosový protokol - Hypertext Transfer Protocol
SSI	skriptovací jazyk na strane serveru - Server Side Includes
CGI	protokol pre prepojenie klienta zo serverom - Common Gateway Interface
DNS	Systém názvov domén - Domain name system
FTP	Protokol prenosu súborov - File Transfer Protocol
MII	Rozhranie nezávislé na prenosovom médiu - Media-independent interface
RMII	Redukované rozhranie nezávislé na prenosovom médiu - Reduced Media-independent interface
GMII	Gigabitové rozhranie nezávislé na prenosovom médiu - Gigabit Media-independent interface

MDIO	Sériová zbernica pre RMI - Management Data Input/Output
HSR	Štandard s vysokou spoľahlivosťou a nízkou redundanciou - High availability Seamless Redundancy
ARM	Architektúra procesorov - Advanced RISC Machine
UART	Univerzálne asynchrónne sériové rozhranie - Universal asynchronous receiver-transmitter
LED	dióda emitujúca svetlo - Light-Emitting Diode
SOP	Druh púzdra - Small Outline Package
MOSFET	Tranzistor riadený polom s hradlovou oxidovou vrstvou - Metal Oxide Semiconductor Field Effect Transistor
SMD	Súčiastka pre povrchovú montáž - Surface Mount Device
JTAG	Konektor pro testovanie plošných spojov a programovanie - Joint Test Action Group
Firmware	Zdrojový kód pre mikrokontrolér
Software	Zdrojový kód pre počítač
SWD	Sériové debugovanie - Serial Wire Debug
IP	Základný komunikačný protokol Internetu - Internet Protocol
TCP	Protokol riadenia prenosu - Transmission Control Protocol
UDP	Používateľský datagramový protokol - User Datagram Protocol
HTML	Hypertextový značkový jazyk - HyperText Markup Language
TLS	Protokol na zabezpečenie komunikáciu - Transport Layer Security
DHCP	Protokol na pridelenie a získavanie IP adries - Dynamic Host Configuration Protocol
ICMP	Protokol na overenie dostupnosti - Internet Control Message Protocol
TCP/IP	Balík internetových protokolov - Transmission Control Protocol/Internet Protocol
JSON	JavaScriptový objektový zápis - JavaScript Object Notation
XML	Rozšíriteľný značkovací jazyk - eXtensible Markup Language
LwIP	Software implementujúci TCP/IP - Lightweight Internet Protocol
RTOS	Operačný systém reálneho času - Real-Time Operating System
FreeRTOS	Software pre operačný systém reálneho času - Free Real-Time Operating System
GNU	Slobodný vývoj operačných systémov - GNU's Not Unix
GDB	Ladiaci nástroj GNU licencie - GNU Debugger
OpenOCD	Slobodný nástroj na ladenie - Open On-Chip Debugger
MIT	Slobodná softvérová licencia - Massachusetts Institute of Technology license
FIFO	Algoritmus prístupu k údajom - First In - First Out
RAM	Pamäť s priamym prístupom - Random Access Memory

MPU	Ochrana pamäte - Memory Protection Unit
RJ-45	Štandard konektoru - Registered jack
JST	Typ konektoru - Japan Solderless Terminal
CMSIS	Spolahlivý prístup k perifériam - Cortex Microcontroller Software Interface Standard
HAL	Hardwarová abstraktná vrstva - Hardware Abstraction Layer

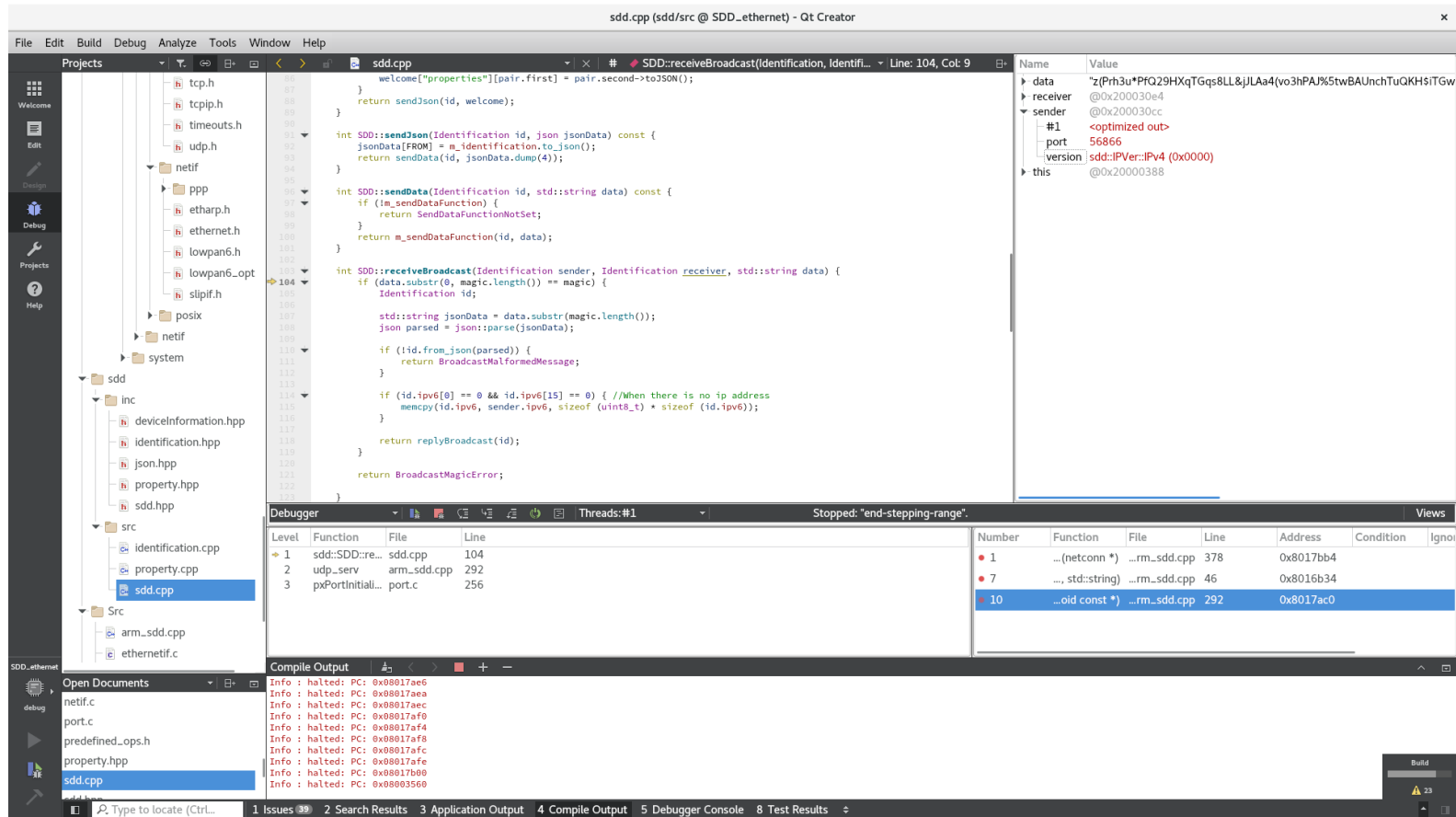
# PRÍLOHY

## Obsah priloženého CD

Prilohy.....	Hlavný adresár
├ Board .....	Podklady k doske plošného spoja
├ Firmware_ETH.....	Firmware pre navrhnutý komunikačný protokol
├ Firmware_MOT .....	Testovací firmware pre riadenie krokových motorov
├ Firmware_test .....	Testovací firmware pre mikrokontrolér STM32f40x
├ pdf .....	Schematické zapojenia a blokové schémy vo formáte pdf
├ └ V-02391-01.pdf .....	Bloková schéma zapojenia prístroja rLIBS
├ └ V-02773-01.pdf .....	Schéma zapojenia prototypového prípravku

# Príloha č.1

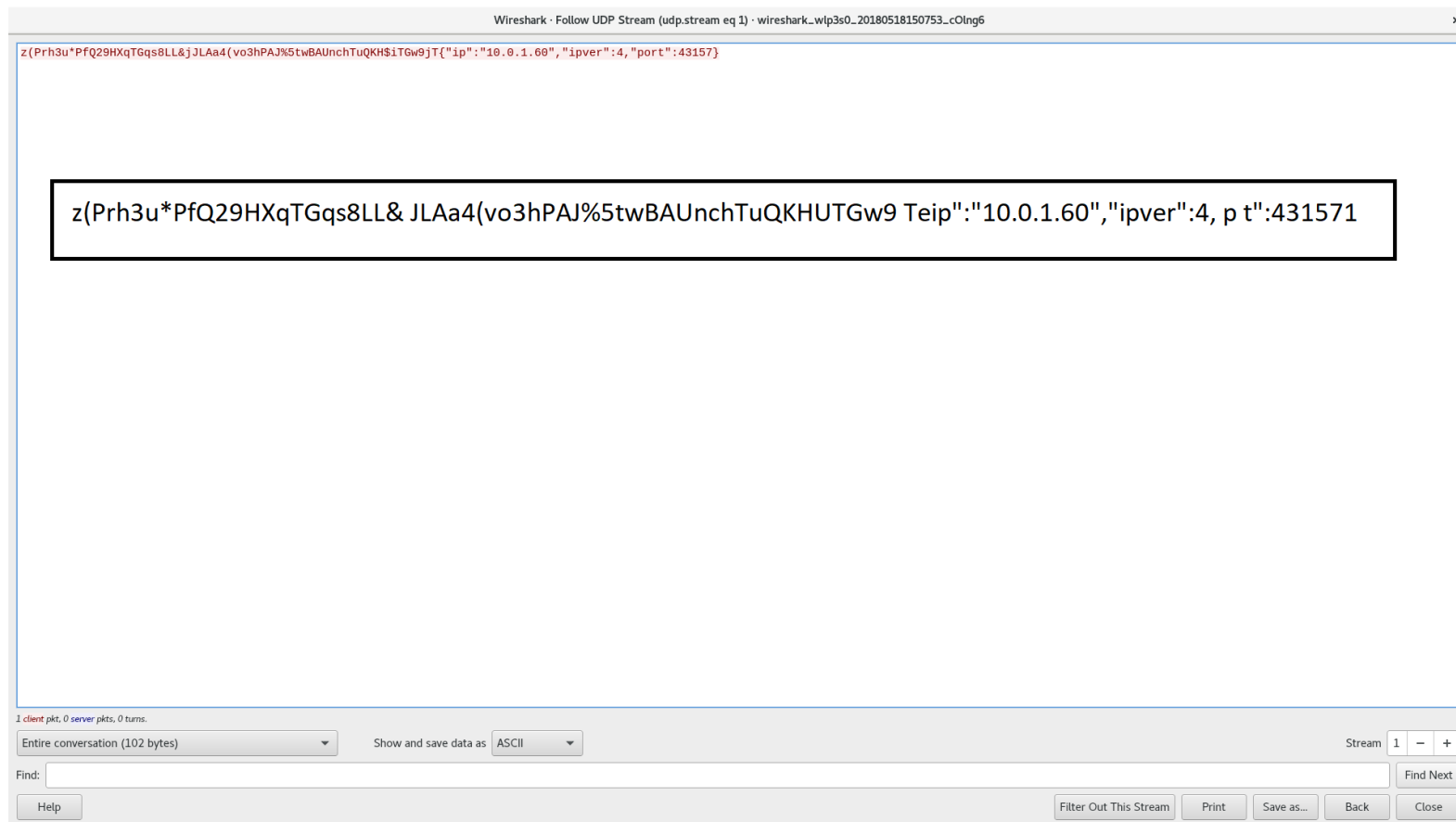
99



Obr. p.1: Ladenie v programe Qt Creator.

## Príloha č.2

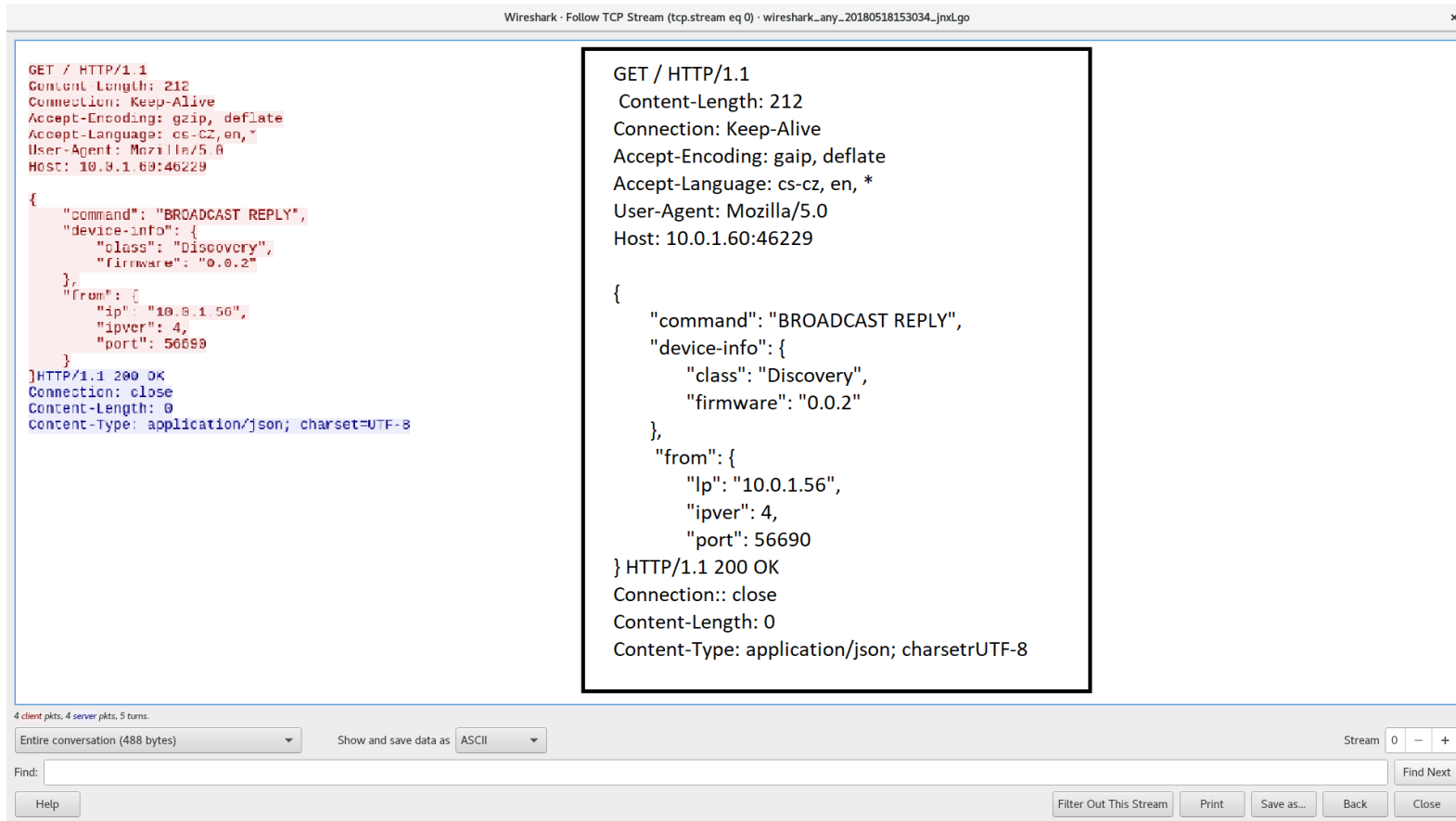
67



Obr. p.2: UDP uvítací datagram.

# Príloha č.3

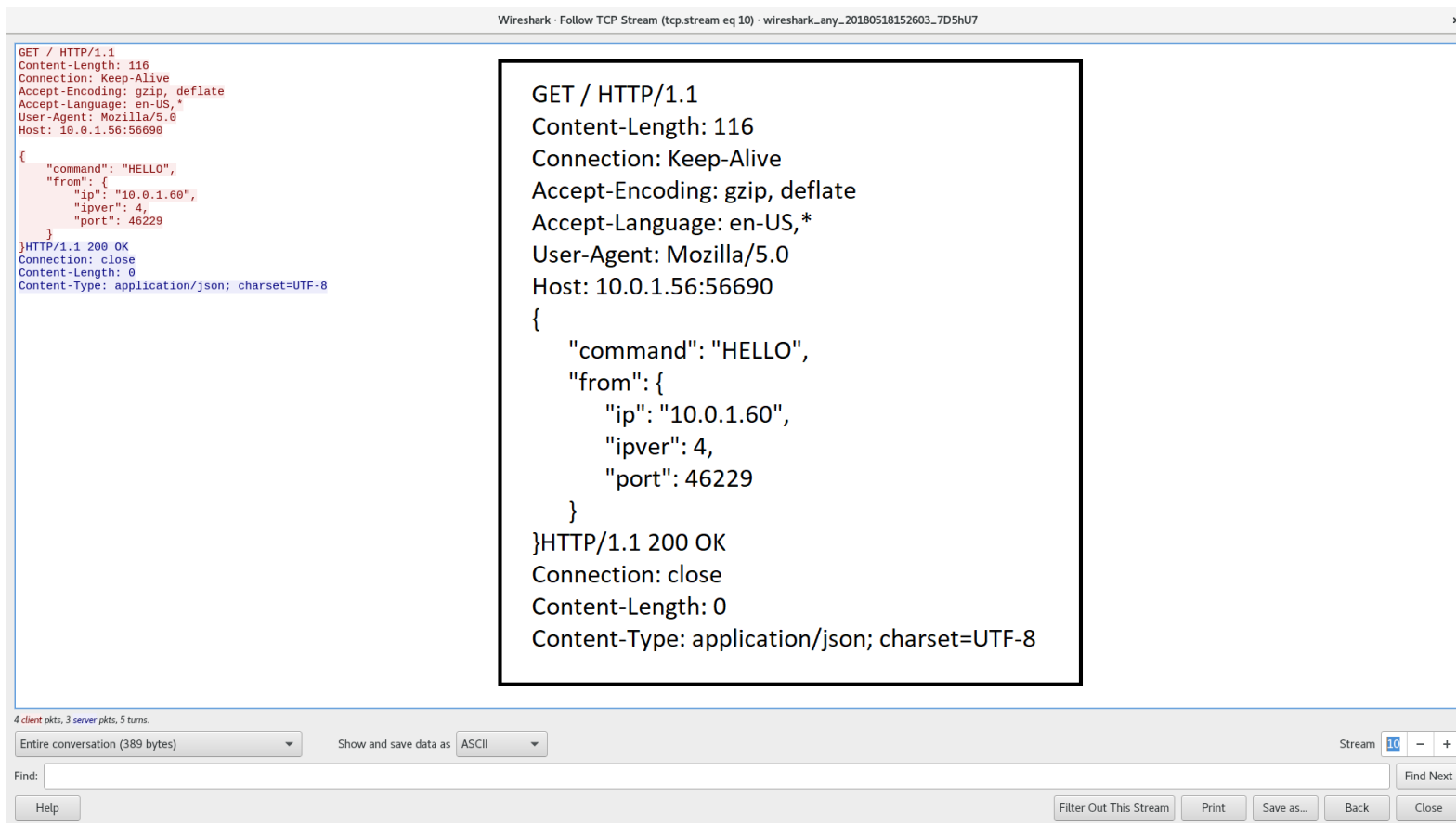
89



Obr. p.3: TCP inicializačný segment.

# Príloha č.4

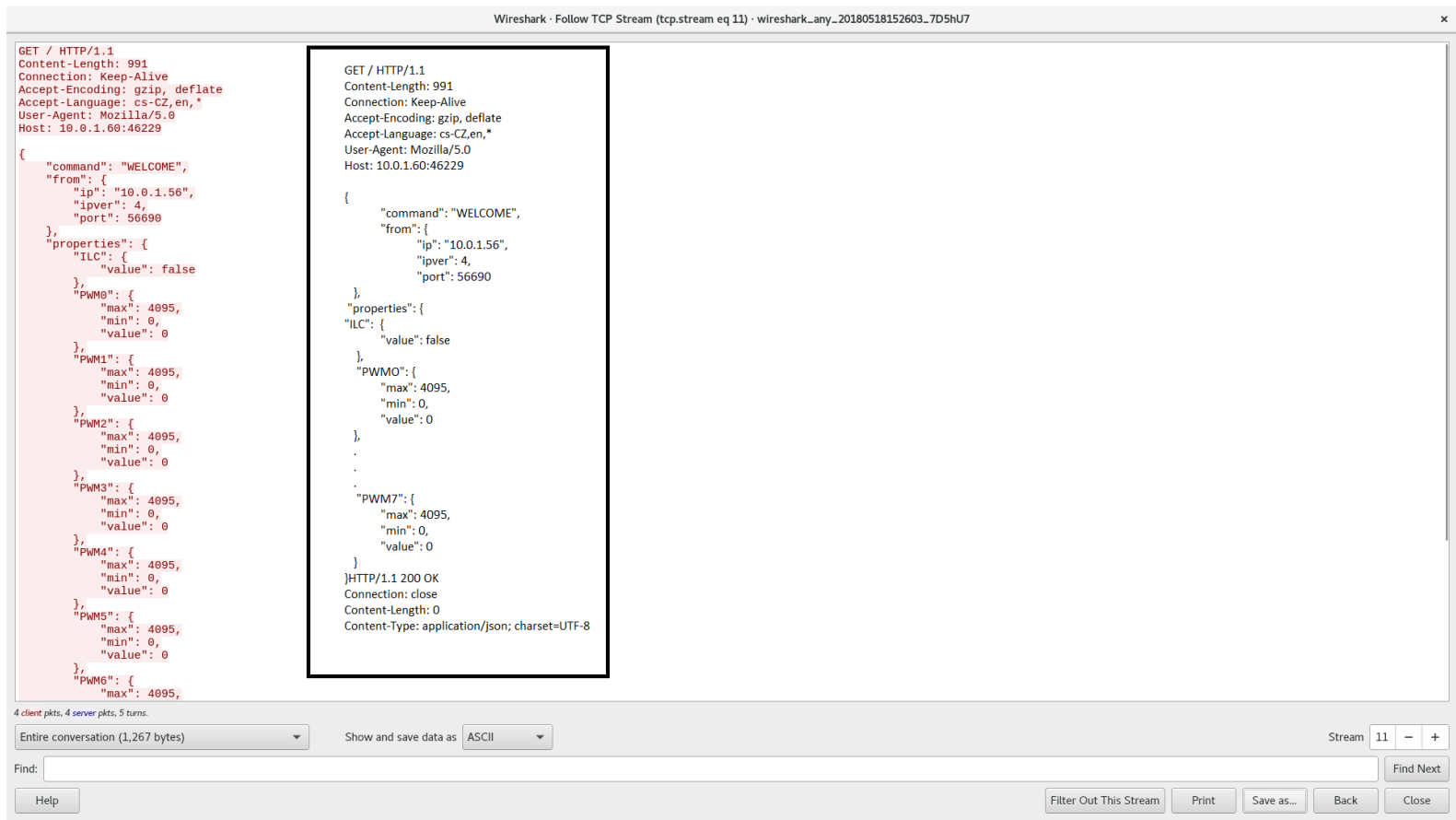
69



Obr. p.4: TCP segment - žiadost o vlastnosti.

# Príloha č.5

70



Ob. p.5: TCP segment - odoslané vlastnosti.