

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

BEZDRÁTOVÁ IP KAMERA S RASPBERRY PI

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

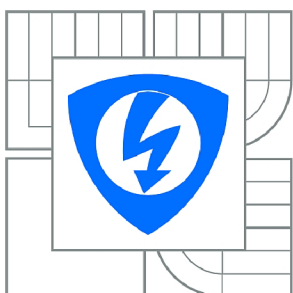
AUTOR PRÁCE
AUTHOR

PETER SIDOR

BRNO 2015



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ**
ÚSTAV TELEKOMUNIKACÍ

**FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS**

BEZDRÁTOVÁ IP KAMERA S RASPBERRY PI

WIRELESS IP CAMERA WITH A RASPBERRY PI

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

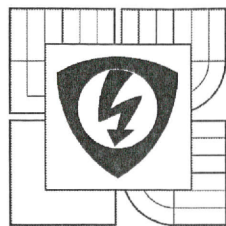
AUTOR PRÁCE
AUTHOR

PETER SIDOR

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. VLADIMÍR ČERVENKA, Ph.D.

BRNO 2015



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ
Fakulta elektrotechniky
a komunikačních technologií
Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student: Peter Sidor
Ročník: 3

ID: 154868
Akademický rok: 2014/15

NÁZEV TÉMATU:

Bezdrátová IP kamera s Raspberry Pi

POKYNY PRO VYPRACOVÁNÍ:

Cílem práce je realizovat dohledovou IP kameru s pomocí Raspberry Pi, která by při detekci pohybu ukládala zachycené obrázky a ty dále zpřístupnila pomocí webového portálu. Kamera musí fungovat i při výpadku síťového napájení, tedy s pomocí akumulátoru. Součástí práce je návrh a realizace autonomní rozšiřující desky s MCU, senzory, správou napájení (stabilizace, nabíjení baterie), vytvoření jednoduché notifikační aplikace pro platformu Android, zobrazující nově pořízené fotografie a dále úprava operačního systému pro rychlý start. Kvalita návrhu řešení bude posuzována na základě výsledné energetické efektivity a rychlosti probuzení celého systému.

DOPORUČENÁ LITERATURA:

- [1] UPTON, Eben. Raspberry Pi: uživatelská příručka. 1. vyd. Brno: Computer Press, 2013, 232 s. ISBN 978-80-251-4116-8.
[2] BARSUKOV, Yevgen a Jinrong QIAN. Battery power management for portable devices. Boston: Artech House, c2013, xvi, 241 p. Artech House power engineering series. ISBN 16-080-7491-9.

Termín zadání: 9. 2. 2015

Termín odevzdání: 2.6.2015

Vedoucí práce: Ing. Vladimír Červenka, Ph.D.

Konzultanti bakalářské práce:



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 a výrobou prídavného modulu pre Raspberry Pi a optimalizáciou efektivity napájania celého zariadenia. Taktiež sa zaoberá zrýchlením operačného systému, ktorý bude využívať celé zariadenie. V prvej časti popisuje princíp činnosti senzorov a DC/DC meničov. Zdôvodňuje výber vhodných súčiastok pre tento modul. Pre meranie efektivity boli vytvorené v programe Eagle dosky plošných spojov vybratých DC/DC meničov. Ďalej bol vytvorený program pre riadenie vypínania a zapínania celého zariadenia. V ďalšej časti je popis možností zrýchlenia operačného systému pre Raspberry Pi. V poslednej časti sa práca venuje návrhu finálneho operačného systému.

KLÚČOVÉ SLOVÁ

IP, kamera, Arduino, Raspberry Pi, LDO, senzor, napájanie, efektivita, Raspbian, DC/DC, Linux

ABSTRACT

This thesis deals with the developing of an additional module for Raspberry Pi and with optimization of the efficiency of powering the whole device. The thesis also deals with the acceleration of the Operating System which is used by the entire device. The first part of this work describes the principles of functioning of the sensors and DC/DC converters. The work focuses on the selection of proper components for this module. The Printed Circuit Boards of the selected DC/DC converters were created in the program Eagle for efficiency measuring. As for the next part, there was created a program for turning on and off the entire device. This part presents a description of the possibilities of accelerating the operating system for Raspberry Pi. The last part of the thesis focuses on design of the final operating system.

KEYWORDS

IP, camera, Arduino, Raspberry Pi, LDO, sensor, power management, efeciency, Raspbian, DC/DC, Linux

SIDOR, Peter *Bezdrôtová kamera s Raspberry Pi*: bakalárska práca. VUT v Brne: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2014. 63 s. Vedúci práce bol Ing. Vladimír Červenka,

PREHLÁSENIE

Prehlasujem, že som svoju bakalársku prácu na tému „Bezdrôtová kamera s Raspberry Pi“ 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 prehlasujem, 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/nebo majetkových a som si plne vedomý následkov porušenia ustanovenia § 11 a nasledujúcich autorského zákona č. 121/2000 Sb., o právu autorskom, o právach súvisejúcich s právom autorským a o zmene niektorých zákonov (autorský zákon), vo znení neskorších predpisov, vrátane možných trestnoprávných dôsledkov vyplývajúcich z ustanovenia časti druhej, hlavy VI. diel 4 Trestného zákoníka č. 40/2009 Sb.

VUT v Brne

.....

(podpis autora)

POĎAKOVANIE

Rád by som poďakoval vedúcemu semestrálne práce pánovi Ing. Vladimírovi Červenkovi za odborné vedenie, konzultácie, trpezlivosť a podnetné návrhy k práci.

VUT v Brne

.....

(podpis autora)

OBSAH

Úvod	11
1 Bezdrôtová kamera s Raspberry Pi	12
2 Raspberry Pi	14
2.1 Parametre (Model B)	14
2.2 Operačné systémy	14
2.3 Napájanie	15
3 DC/DC meniče	16
3.1 Lineárne regulované meniče	16
3.2 Spínané meniče	16
4 Senzory	18
4.1 PIR senzor	18
4.2 Ultrazvukový senzor	19
5 Návrh rozširujúceho modulu	20
5.1 Výber PIR senzora	20
5.2 Výber ultrazvukového senzora	21
5.3 Porovnanie spínaných meničov	22
5.3.1 Návrh DPS meničov	23
5.3.2 Meranie efektívnosti meničov	24
5.4 Výber mikrokontroléra	26
5.5 Návrh programu pre Arduino	27
5.6 Battery management	28
5.6.1 Akumulátory	29
5.6.2 Výber nabíjača	30
5.7 Návrh DPS rozširujúceho modulu	31
6 Operačný systém	33
6.1 Embedded systém	34
6.1.1 Vytvorenie embedded systému	34
6.2 Operačný systém Raspbian	35
6.3 Konečný návrh OS	37
6.3.1 Meranie spotreby finálneho OS	39
7 Záver	41

Literatúra	42
Zoznam symbolov, veličín a skratiek	44
Zoznam príloh	45
A Obsah priloženého CD	46
A.1 Schémy a návrhy DPS	46
A.2 Program pre Arduino	46
A.3 Operačný systém	46
A.4 Webová stránka	46
B Návrhy DC/DC meničov	47
C Rozširujúci modul	54
D Zdrojový kód programu	55
E Skripty	60
E.1 Skript fotka	60
E.2 Skript poweroff.py	60
E.3 Skript reboot.py	60
E.4 Skript script-copy	61
E.5 Skript mail	61
E.6 Skript script-find	61
E.7 Skript ping	62
E.8 Skript S41mount	62
F Webová stránka	63

ZOZNAM OBRÁZKOV

1.1	Bloková schéma riešenia.	12
2.1	Raspberry Pi(Model B).	14
2.2	Schema zapojenia LDO v Raspberry Pi [11].	15
4.1	Príklad PIR senzora.	18
4.2	Príklad ultrazvukového senzora [12].	19
5.1	Príklad návrhu DPS	23
5.2	TPS54395 - Graf závislosti efektivity na výstupnom prúde	24
5.3	TPS63060 - Graf závislosti efektivity na výstupnom prúde	25
5.4	LTC3111 - Graf závislosti efektivity na výstupnom prúde	25
5.5	Schéma zapojenia Arduina s prídavnými komponentami	26
5.6	Vývojový diagram	27
5.7	Arduino Nano	28
5.8	Rozdelenie akumulátorov [13].	29
5.9	Schéma zapojenia nabíjačky	30
5.10	Schéma zapojenia rozširujúceho modulu	32
6.1	Konfiguračné okno Buildroot[14]	34
6.2	Menu pretaktovania	35
6.3	Súbor cmdline.txt	36
6.4	Inicializačný súbor inittab	36
6.5	Skripty spúšťané pri štarte	38
6.6	Testovací prípravok	40
B.1	Schéma zapojenia DC/DC meniča LTC3111	47
B.2	Návrh DPS meniča LTC3111 - vrchná strana	47
B.3	Návrh DPS meniča LTC3111 - spodná strana	48
B.4	Schéma zapojenia DC/DC meniča MAX710	48
B.5	Návrh DPS meniča MAX710 - vrchná strana	49
B.6	Návrh DPS meniča MAX710 - spodná strana	49
B.7	Schéma zapojenia DC/DC meniča TPS62290	50
B.8	Návrh DPS meniča TPS62290 - vrchná strana	50
B.9	Schéma zapojenia DC/DC meniča TPS62160	51
B.10	Návrh DPS meniča TPS62160 - vrchná strana	51
B.11	Schéma zapojenia DC/DC meniča TPS63060	51
B.12	Návrh DPS meniča TPS63060 - vrchná strana	52
B.13	Návrh DPS meniča TPS63060 - spodná strana	52
B.14	Schéma zapojenia DC/DC meniča TPS54395	53
B.15	Návrh DPS meniča TPS54395 - vrchná strana	53
B.16	Návrh DPS meniča TPS54395 - spodná strana	53

C.1	Návrh DPS rozširujúceho modulu - vrchná strana	54
C.2	Návrh DPS rozširujúceho modulu - spodná strana	54

ZOZNAM TABULIEK

5.1	Porovnanie PIR senzorov podľa odoberaného kludového prúdu	21
5.2	Porovnanie ultrazvukových senzorov podľa odoberaného prúdu	22
5.3	Porovnanie spínaných meničov podľa efektívnosti a odoberaného prúdu	22
6.1	Namerané časy spustenia a odbery OS	33
6.2	Nameraný prúdový odber v čase	39
6.3	Nameraný prúdový odber finálneho OS	39

ÚVOD

V dnešnej dobe sa čím ďalej tým viac stretávame so zabezpečovacími systémami, ktoré sú vybavené rôznymi špeciálnymi technológiami, ktoré sa snažia čo najlepšie a najefektívnejšie zabezpečiť daný chránený objekt. S pokrokom vedy a techniky sa začali využívať rôzne moderné technológie, ako bezdrôtové zabezpečovacie systémy alebo prvky s využitím komunikácie cez Internet. Či už ide o zabezpečenie firmy, obytného domu alebo pozemku, sú kladené väčšie a väčšie nároky na cenu a efektívnosť napájania zabezpečovacieho systému.

Tieto moderné prvky sú tvorené špeciálnymi senzormi, kamerami, mikroprocesorovým riadením a zložitou elektronikou pre správu napájania. A preto sa táto práca zaoberá realizáciou týchto zariadení. Odkúšané bolo viacero možných riešení a senzorov, ale pre praktické riešenie boli vybraté iba tie najvhodnejšie.

Pre ukážku použitia týchto technológií bola vytvorená bezdrôtová IP kamera s Raspberry Pi, s rozširujúcim hardvérom pre správu napájania, nabíjania a pre pripojenie ďalších senzorov, ktoré budú riadené mikroprocesorom. Pri napájaní, obzvlášť napájaní z batérie, je kladený dôraz na efektívnosť, preto je nutné zvoliť vhodné efektívne zapojenie pre napájanie celého modulu.

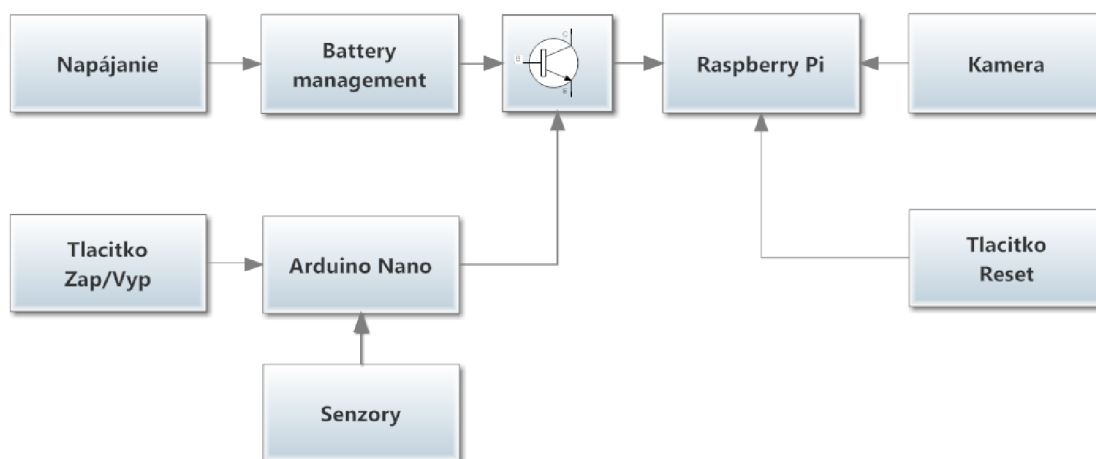
Práca rieši problém efektívnosti napájania a rýchlosť štartu operačného systému pre počítač Raspberry Pi a prídavnú kameru.

1 BEZDRÔTOVÁ KAMERA S RASPBERRY PI

Ako už bolo povedané v úvode, trendom zabezpečovacích systémov je bezdrôtový prenos a prenos pomocou IP protokolu.

Táto práca sa snaží vyriešiť a ujasniť viaceré problémy. Dvoma hlavnými problémami je efektívnosť napájania pri napájaní z prídavnej batérie a zníženie odoberaného prúdu Raspberry Pi pri nečinnosti. Keďže Raspberry Pi v režime spánku má veľký prúdový odber (až 100 mA), riešenie obsahuje mikrokontrolér, ktorý za pomoci senzorov obsluhuje vypínanie a zapínanie napájania pre Raspberry Pi.

Pre správnu a efektívnu funkciu obsahuje toto riešenie viacero modulov, ktoré sú zobrazené na obr. 1.1.



Obr. 1.1: Bloková schéma riešenia.

Každá časť blokovej schémy bude popísaná v ďalších kapitolách a podkapitolách tejto práce. Najdôležitejšou časťou je rozširujúci modul, ktorý pozostáva z battery managementu, mikroprocesora, prídavných senzorov, tlačidiel a potrebných obvodov ktoré ovládajú napájanie počítača.

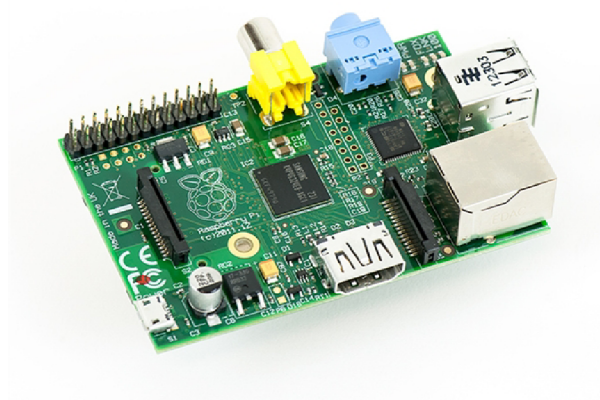
Modul Raspberry Pi má za úlohu pomocou kamery a softvéru Motion detection urobiť fotografiu pri zaznamenaní pohybu a poslať súbor cez wifi rozhranie na webový server. Zároveň musí byť schopný fungovať aj pri výpadku sieťového napájania.

Ďalej zapojenie obsahuje battery management, ktorý sa stará o správne nabíjanie batérie. Ak nastane výpadok napájacieho napätia, Raspberry Pi začne byť napájané z batérie. Vtedy preberajú úlohu stráženia objektu senzory a mikropočítač zapne šetriaci mód. Následne riadiacim signálom korektne vypne Raspberry Pi a odpojí napájací prúd z batérie, pretože aj pri vypnutom operačnom systéme počítač odoberá veľký prúd.

Pre napájanie Raspberry Pi a prídavných senzorov z batérie je potrebné obmedziť odoberaný prúd na čo najnižšiu hodnotu a zefektívniť celé napájanie. Pre zefektívnenie napájania a zmenšenie odoberaného prúdu je potrebné LDO (Linearny regulátor – Low-Drop Out) obvody zameniť za vhodné spínané DC/DC meniče. Spínané DC/DC meniče sú známe svojou vysokou efektivitou. Vlastnosti, výhody a nevýhody týchto meničov sú riešené v kapitole **DC/DC meniče 3**.

2 RASPBERRY PI

Ide o malý jedno doskový počítač vid' obr. 2.1 s rôznymi vstupno-výstupnými konektormi o rozmeroch 85,6 mm x 54 mm, vyvinutý organizáciou Raspberry Pi Foundation. Raspberry Pi má niekoľko výhod. Jedna z najväčších je cena. Tento počítač je určený hlavne pre vzdelávanie v oblasti programovania, ale samozrejme má využitie aj v iných oblastiach, ako napríklad FTP server, NAS server, ovládanie vstupno-výstupných zariadení ako senzory, motory a podobne. Je dostupný v troch verziách, model A, model B a model B+.



Obr. 2.1: Raspberry Pi(Model B).

2.1 Parametre (Model B)

- Procesor ARM11 taktovaný na frekvenciu 700 MHz
- 512 MB SDRAM
- Grafický čip integrujúci OpenGL ES 2.0 standard
- Kompozitný a HDMI video výstup
- Dvoj-portový USB 2.0 hub a ethernetový radič
- SD/MMC/SDIO slot (namiesto harddisku)
- Zvukový výstup cez 3,5 mm konektor
- 12×GPIO, UART, I²C, zbernicu SPI

2.2 Operačné systémy

Samotný výrobca ponúka k počítaču operačné systémy ARM verzie linuxových distribúcií Debian a Arch. Výrobca tiež ohlásil prácu na systéme Razdroid pre Rasp-

berry Pi na báze Android 4.0[1]. Na internete sa dá nájsť mnoho ďalších neoficiálnych operačných systémov.

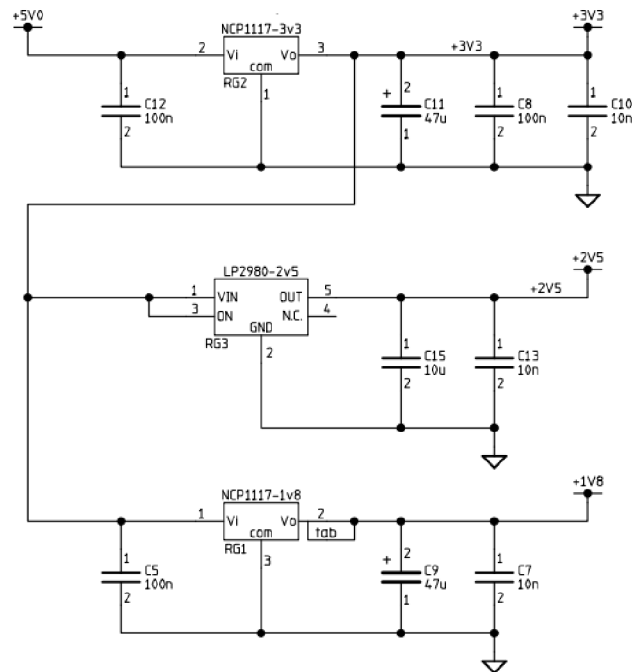
Typy OS:

Oficiálne voľne stiahnutelné operačné systémy ponúkané výrobcom:

- Raspbian
- Pidora
- Openelec
- Raspbmc
- Risc OS

2.3 Napájanie

Raspberry Pi je napájané prostredníctvom štandardného mikroUSB konektoru s napätím o veľkosti 5 V. Pri privedení napájania sa o stabilizáciu starajú integrované obvody LDO (Lineárny regulátor – Low-Drop Out), konkrétne pre 3,3 V a 1,8 V typ NCP1117 a pre 2,5 V typ LP2980 obr. 2.2, z ktorých je napájané celé zariadenie. V porovnaní so spínanými DC/DC meničmi majú LDO obvody malú účinnosť 3.



Obr. 2.2: Schema zapojenia LDO v Raspberry Pi [11].

3 DC/DC MENIČE

Využitelnosť jednosmerných meničov v dnešnej dobe je veľmi veľká. Nachádzajú sa takmer vo všetkých elektrických zariadeniach. Menia vstupné jednosmerné napätie na výstupné napätie inej veľkosti. Sú schopné výstupné napätie zvýšiť alebo znížiť, to záleží od typu meniča. Podľa konštrukcie môžeme DC/DC meniče rozdeliť do dvoch základných skupín:

- Lineárne regulované meniče
- Spínané (impulzné) meniče

3.1 Lineárne regulované meniče

Sú to skôr stabilizátory ako meniče, pretože sú schopné napätie len znížiť. Ich využitie nájdeme hlavne tam, kde ide o malé rozdiely vstupného a výstupného napätia, pretože so stúpajúcim rozdielom rapídne klesá účinnosť daného meniča.

Pracujú na princípe napäťového deliča, ktorý je zastúpený regulovaným tranzistorom, na ktorom vzniká konštantný úbytok napätia bez ohľadu na zaťažovací prúd.

Výhody:

- Jednoduchosť
- Nízka cena
- Takmer žiadne rušenie
- Dobrá stabilizácia výstupného napätia

Nevýhody:

- Malá účinnosť
- Väčšia hmotnosť
- Potreba chladenia
- Nelineárne súčiastky
- Neumožňujú galvanické oddelenie

3.2 Spínané meniče

Meniče, využívajúce spínané prvky, sú v poslednej dobe na vzostupe. Spínané zdroje doslova ovládli trh. Je to spôsobené hlavne ich vysokou účinnosťou, malými rozmermi a pomerne ľahkou a presnou reguláciou. Sú taktiež výhodné pri veľkom rozdielom vstupného a výstupného napätia.

V porovnaní so spojitými regulovanými meničmi vyhrávajú hlavne vďaka vysokej účinnosti, ktorá sa pohybuje okolo 90 %. Táto vlastnosť je cenená všade tam, kde máme obmedzenú zásobu energie. Napríklad pri napájaní z batérií, akumulátorov a podobne.

Spínané meniče pracujú na princípe akumulácie energie v indukčnostiach alebo kapacitoroch. Oproti klasickým stabilizátorom sú schopné vytvoriť aj napätie inverzné voči zdroju. Výkonný stupeň, v podobe tranzistora alebo tyristora, je prevádzkovaný impulzne. Môžeme teda využívať pulzného režimu, v ktorom súčiastky vydržia krátkodobo oveľa väčší výkon[2]. Ich praktický návrh však prináša celú radu nevýhod.

Nevýhody:

- Zvlnené výstupné napätie
- Sú zdrojom frekvenčného rušenia

Rozdelenie:

- Buck (Step-down) meniče
- Boost (Step-up) meniče
- Buck-Boost meniče

Buck meniče nazývané aj step-down meniče sú meniče, ktoré vstupné napätie vedia iba znížiť na určitú nastavenú hodnotu. Opačnú funkciu, zvýšenie vstupného napätia, majú boost meniče nazývané taktiež step-up meniče. Buck-boost meniče umožňujú udržať výstupné napätie aj pri kolísaní vstupného napätia pod aj nad nastavenú hodnotu výstupného napätia.

4 SENZORY

Senzor je fyzikálne zariadenie, ktoré reaguje na zmenu meranej veličiny. Najpoužívanějšíe senzory sú elektrické. Sú schopné reagovať na meranú veličinu a následne ju transformovať na výstupný elektrický signál. Z pohľadu vstupných meraných veličín môžeme senzory rozdeliť na snímače mechanické, akustické, magnetické, tepelné, chemické a pod. V súčasnosti majú senzory veľmi širokú škálu využiteľnosti, napríklad v automatizácii, medicíne, automobiloch, priemysle a zabezpečovacích systémoch.

Snímače sa rozdeľujú podľa premeny neelektrickej veličiny na elektrický signál, na aktívne a pasívne.

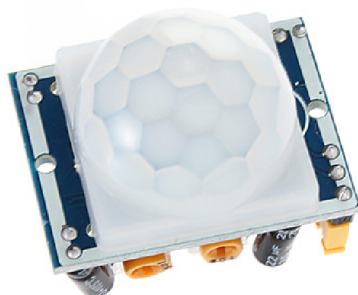
Aktívne snímače premieňajú priamo mechanickú, tepelnú alebo chemickú energiu na energiu elektrickú. Sú priamo zdrojom elektrickej energie (napr. piezoelektrický menič).

Pasívne snímače menia svoje elektrické vlastnosti (indukčnosť, odpor), pôsobením napríklad tlaku alebo teploty. Aby mohol pasívny snímač fungovať potrebuje byť napájaný elektrickou energiou.

Rozširujúci modul umožňuje pripojiť mnoho druhov sensorov. Vzhľadom na cenu a dostupnosť, pre ukážku funkčnosti zariadenia bol vybraný ultrazvukový senzor a PIR (Pasívny infračervený senzor – Passive infrared senzor).

4.1 PIR senzor

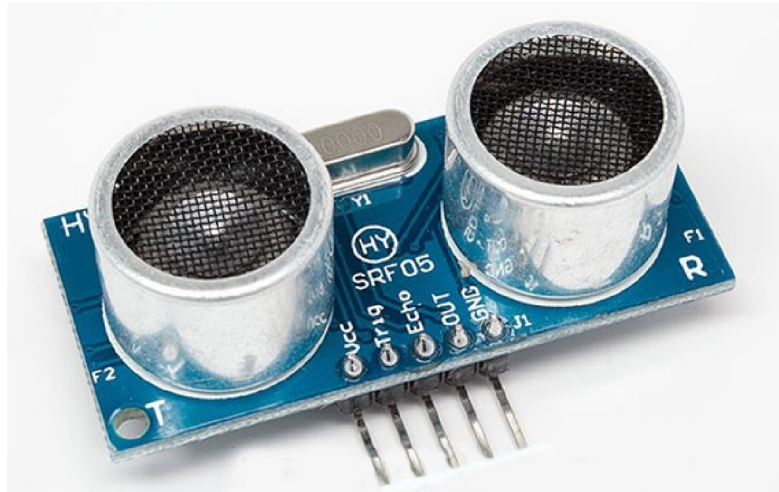
Sú to najčastejšie využívané senzory v zabezpečovacích systémoch. Pracujú na princípe merania teplotnej zmeny objektu oproti teplotnému pozadiu snímanej oblasti. Pri týchto senzoroch sa ráta s tým, že každé živé telo je zdrojom infračerveného žiarenia. Tieto senzory nevyžarujú žiadnu energiu čiže patria do skupiny pasívnych snímačov.



Obr. 4.1: Príklad PIR senzora.

4.2 Ultrazvukový senzor

Sú založené na princípe merania času šírenia zvukovej vlny. Snímač najskôr vyvolá vysokofrekvenčný zvukový signál a následne čaká na odrazený signál od prekážky. Potom elektronika senzora pomocou nameraného času, dopočíta výslednú vzdialenosť. Snímač má veľkú nevýhodu. Ak je meraná prekážka rôznych tvarov a materiálov, vysielaná vlna sa môže chybné odraziť a narušiť správnosť výpočtu.



Obr. 4.2: Príklad ultrazvukového senzora [12].

5 NÁVRH ROZŠIRUJÚCEHO MODULU

V tejto kapitole budú uvedené a podrobne popísané všetky časti navrhovaného prídavného modulu pre IP kameru s Raspberry Pi. Je tu popísaný výber daných súčiastok a zdôvodnené ich použitie. Celý návrh spočíva vo výbere najlepších súčiastok pre navrhovanú kameru a je postavený na myšlienke dosiahnuť najlepšiu efektívitu napájania. Pre zaručenie nízkej spotreby zariadenia je potrebné spraviť meranie a porovnanie tých najlepších dostupných komponentov.

V nasledujúcich podkapitolách budú detailne rozobraté časti celého zariadenia. Ku každej časti budú uvedené porovnávané súčiastky, schéma zapojenia a popřípade potrebné meranie, z ktorého sa vychádzalo pri výbere vhodných komponentov.

Návrh všetkých častí prídavného modulu kamery, t.j. schémy zapojenia a návrh DPS (Doska plošných spojov) bol uskutočnený pomocou počítačového programu EAGLE.

5.1 Výber PIR senzora

Na trhu sa vyskytuje veľa rozličných senzorov od rôznych výrobcov. Každý senzor má svoje parametre, podľa ktorých sa dajú rozlíšiť. Táto práca sa zaoberá hlavne efektívnosťou napájania, čiže pri výbere senzorov je dôležité pozeráť hlavne na odobieraný kludový prúd.

Nieje veľa kvalitných a osvedčených výrobcov senzorov na trhu, naopak čoraz častejšie sa objavujú lacné menej kvalitné kópie. Ich veľká výhoda je cena a dostupnosť. Avšak majú aj nevýhody. Hlavnou nevýhodou je ťažko dostupná dokumentácia a nepravdivé údaje v nej, čo sťažuje výber a porovnanie senzorov. Väčšina týchto senzorov má uvedené hodnoty odobieraného prúdu okolo $50\ \mu\text{A}$. Tieto hodnoty nie sú nereálne, avšak po nahliadnutí do firemnej dokumentácie alebo po uskutočnení merania zistíme, že hodnoty neodpovedajú skutočnosti. Vzhľadom na to, že na internete je veľa dostupných článkov a meraní, ktoré sa zaoberajú práve týmito senzormi, nebolo nutné robiť merania pre výber. Viacero obstojných výrobcov ponúka senzory s hodnotami ešte nižšími, avšak tu je veľkým problémom cena.

V tabuľke 5.1 je možné vidieť porovnanie dostupných senzorov, ktoré je možné využiť pre toto zariadenie. Ak sa nehladá na cenu, vyhovuje viacero senzorov od osvedčených výrobcov.

Táto práca sa zaoberá aj finančnou efektívnosťou, preto musel byť spravený prieskum lacnejších senzorov a vybratý kompromis medzi cenou, dostupnosťou a odobieraným prúdom.

Ako už bolo spomenuté vyššie, pri lacných senzoch je problémom najst dôverhodné údaje o nich. Preto bolo potrebné pozrieť do dokumentácie používaných integrovaných obvodov a zistiť ich odoberaný prúd. Tieto lacné senzory väčšinou neobsahujú žiadne označenie, pretože zväčša ide o okopírovaný produkt iného výrobcu s použitím lacnejších súčiastok. To zapríčinilo obtiažne hľadanie parametrov týchto senzorov. Keďže senzory nemáme k dispozícii, bolo nutné najst fotku každého senzora a vyhľadať na nej aké súčiastky používa. Veľkou komplikáciou je, že mnoho senzorov používa súčiastky bez označenia. Kvôli nedostatočnému označeniu lacnejších senzorov bolo nemožné urobiť ich prehľad a porovnanie týchto senzorov.

Tab. 5.1: Porovnanie PIR senzorov podľa odoberaného kludového prúdu

Katalog. č.	Výrobca	I_q [μ A]	U_{vst} [V]	Vzdialenosť [m]	Cena [€]
EKMB1203111	Panasonic	2	2,3 - 4,0	12	16,8
EKMC1603111	Panasonic	170	3,0 - 6,0	12	8,9
555-28027	Parallax	130	3,0 - 6,0	9	8,9
MS-320	IR-TEC	800	1,5 - 5,0	12	7
MS-320LP	IR-TEC	10	0,5 - 3,0	12	11
EKMC1601112	Panasonic	170	3,0 - 6,0	5	12
KC7783R	Trusens	300	4,7 - 12,0	5	8

Z porovnávaných senzorov je vhodný pre toto riešenie typ EKMB1203111, pretože spĺňa požadované parametre. Jeho kludový prúd I_q je najnižší z porovnávaných senzorov. Nevýhodou je ale cena a preto nebol vybraný pre toto riešenie. Kvôli nízkemu finančnému rozpočtu a veľkej dostupnosti bol vybraný senzor, ktorý je zobrazený na obrázku 4.1.

5.2 Výber ultrazvukového senzora

Taktiež môžeme najst na trhu širokú škálu týchto senzorov, či už ide menej kvalitné kópie senzorov alebo o kvalitnejšie a drahšie senzory. Väčšina čínskych senzorov neobsahuje katalógové čísla a ani žiadne označenie. Pri týchto senzoch bol výber urobený na podobnom princípe ako u PIR senzorov, čiže podľa fotografií na internete zistené dané súčiastky a následne zistený odoberaný prúd. Toto porovnanie je menej komplikované, pretože na fotkách sú ľahšie rozpoznateľné použité súčiastky. Pre výber senzora bol urobený kompromis medzi cenou, dostupnosťou a veľkosťou odoberaného prúdu. Z porovnávaných senzorov je pre toto riešenie vhodný LV-PROXSONAR-EZ0, pretože má najnižší kludový prúd z porovnávaných senzorov ale cenovo nevyhovuje tomuto zariadeniu. Kvôli nízkemu finančnému rozpočtu

a veľkej dostupnosti bol vybraný HY-SRFO5, ktorého cena je veľmi nízka.

Tab. 5.2: Porovnanie ultrazvukových senzorov podľa odoberaného prúdu

Katalog. č.	Výrobca	I_q [mA]	U_{vst} [V]	Výstup	Cena [€]
Ping))) 28015	Parallax	30	5	Digital	25
LV-proxsonar-EZ0	MaxBotix	2	2,5 - 5,5	Digital	24
HRLV-MaxSonar-EZ0	MaxBotix	3,1	2,5 - 5,5	Digital	32
HY-SRFO5		5	5	Digital	2

5.3 Porovnanie spínaných meničov

V tejto kapitole sú porovnané dostupné spínané meniče. Meniče boli vybrané a porovnávané od známych výrobcov t.j. Linear, Texas Instrument a Maxim Integrated. Títo výrobcovia ponúkajú širokú škálu spínaných meničov či už sa jedná o meniče pracujúce s nízkymi hodnotami napätí a prúdov alebo vysokými.

Od týchto výrobcov boli vybrané súčiastky, ktoré spĺňali požadované parametre (vstupné a výstupné napätia, vysoká efektivita, kludový prúd). Porovnávané boli podľa najhlavnejších parametrov, efektívnosti a kludového prúdu I_q . Porovnanie podľa parametrov udávaných výrobcom môžeme vidieť v tabuľke 5.3. Hodnoty efektivity boli odčítané pri výstupnom prúde 0,4 A. Tento prúd je očakávaný pri optimalizácii napájania celého zariadenia.

Tab. 5.3: Porovnanie spínaných meničov podľa efektívnosti a odoberaného prúdu

Katalog. č.	Výrobca	Typ	U_{vst} [V]	U_{vys} [V]	I_q [mA]	Účinnosť
TPS62040	TI	Buck	2,5-6	3,3	0,018	94,8
TPS62290	TI	Buck	2,3-6	3,3	0,015	94
TPS62250	TI	Buck	2-6	3,3	0,018	92
TPS62410-Q1	TI	Buck	2,5-6	3,3/1,8	0,023	94
LM26400Y	TI	Buck	3-20	5/3,3	1,8	90
TPS54395	TI	Buck	4,5-18	5/3,3	1,2	91
TPS63060	TI	Buck-Boost	2,5-12	5	0,03	92
TPS62160	TI	Buck	3,17	5	0,017	96
TPIC74101-Q1	TI	Buck-Boost	1,5-40	5	0,1	80
LTC3111	Linear	Buck-Boost	2,5-25	5	0,055	95
MAX710	Maxim	Buck	1,8-11	5/3,3	0,1	87
LTM4609	Linear	Buck-Boost	4,5-36	5	2,8	98

Vzhľadom k tomu že výrobcovia ponúkajú možnosť poslať bezplatne vzorky meničov takzvané sample, boli urobené schémy a návrhy vybratých meničov. Následne bolo spravené meranie ich efektívnosti.

Medzi porovnávané meniče boli vybraté step-down meniče, ktoré znižujú napätie, buck-boost meniče, ktoré dokážu zvýšiť alebo znížiť vstupné napätie a duálne step-down meniče, ktoré obsahujú dva výstupy s rôznymi hodnotami napätia. Najväčším problémom bolo zistenie, že výrobcovia neposielaajú vzorky ku každej vybratej súčiastke a daná súčiastka sa dá kúpiť iba vo veľkom množstve. Pre meranie boli vybrané meniče ktoré spĺňali parametre (dostupnosť, efektívnosť).

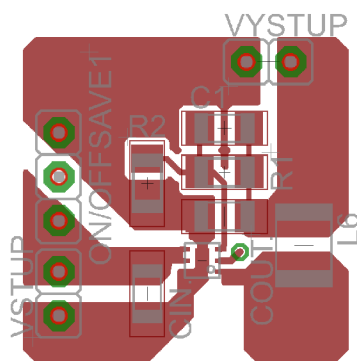
5.3.1 Návrh DPS meničov

Pre každý vybraný menič bola spravená schéma a následne návrh DPS. Všetky tieto návrhy je možné nájsť v prílohách B.

Vybrané meniče:

- MAX710
- TPS62290
- TPS54395
- LTC3111
- TPS63060
- TPS62160

Pri návrhu DPS je dôležité dávať pozor na umiestnenie danej súčiastky. Je to dané tým že meniče pracujú na vysokých frekvenciách a ak by neboli dodržané isté pravidlá, menič by bol nefunkčný. Väčšina výrobcov udáva v dokumentácii súčiastok pravidlá, ktoré musia byť dodržané a postup akým DPS navrhnuť. Taktiež je v dokumentácii možné nájsť odporúčané súčiastky (kondenzátory, cievky). V niektorých dokumentáciách výrobca udáva aj základný návrh DPS.



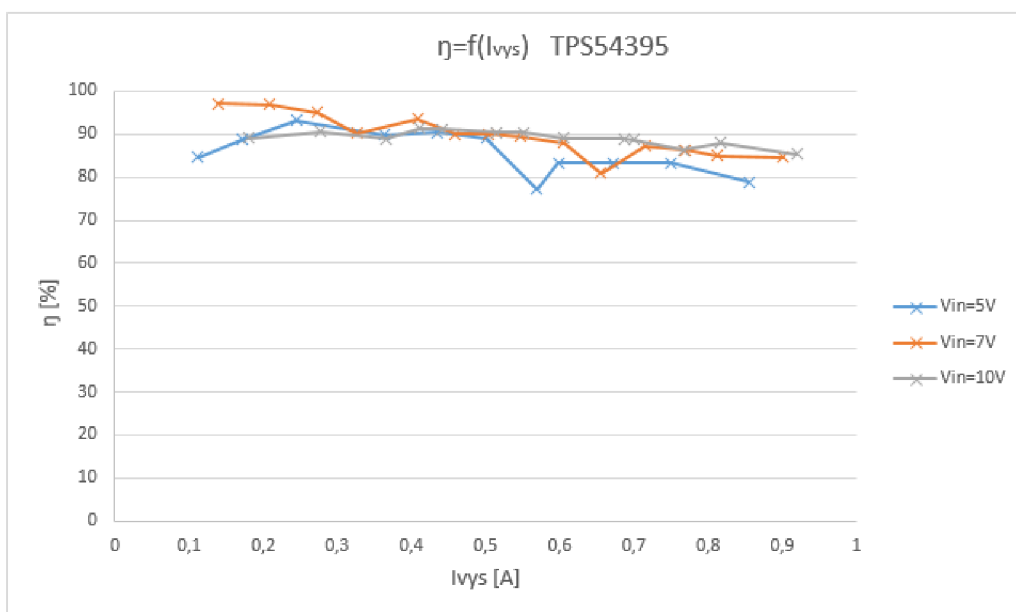
Obr. 5.1: Príklad návrhu DPS

5.3.2 Meranie efektívnosti meničov

Po výrobe a kompletizácii vybratých meničov bolo urobené meranie efektívnosti. Výber meničov bol zameraný hlavne na meniče s výstupným napätím 5 V, čo je napájacie napätie zariadenia. Taktiež boli vybrané meniče s výstupom 3,3 V pre nahradenie LDO meniča v Raspberry Pi.

Pri meraní bolo zistené, že niektoré meniče nepracovali korektne čo znamená nekompletnosť merania. Nefunkčnosť meničov môže byť spôsobená zlým návrhom DPS, pretože ako už bolo spomínané, návrh DPS pre DC/DC menič je veľmi obťažný. Taktiež môže spôsobiť nefunkčnosť neodborná kompletizácia meniča v nie ideálnych laboratórnych podmienkach.

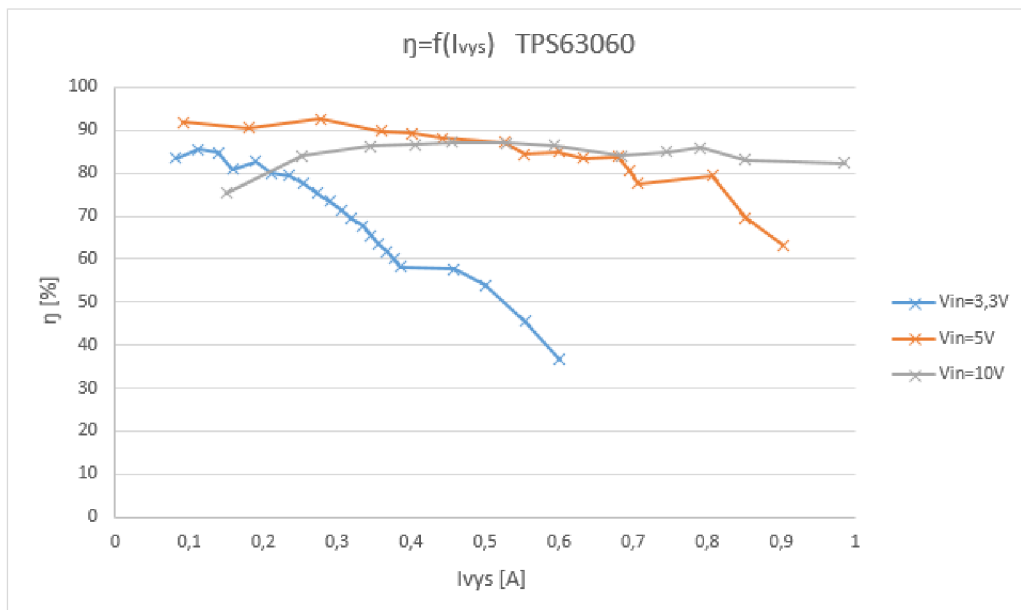
Pre funkčné meniče bolo urobené meranie efektivity, v rozsahu výstupného odoberaného prúdu 0-1 A. Meranie bolo vykonané pre viacero vstupných napätí. Nameňované hodnoty v tabulkách je možné nájsť v prílohách tejto práce, ktoré boli následne graficky znázornené vid. obrázky (pre TPS63060 obr.5.3, pre TPS54395 obr.5.2 a pre LTC3111 obr.5.4).



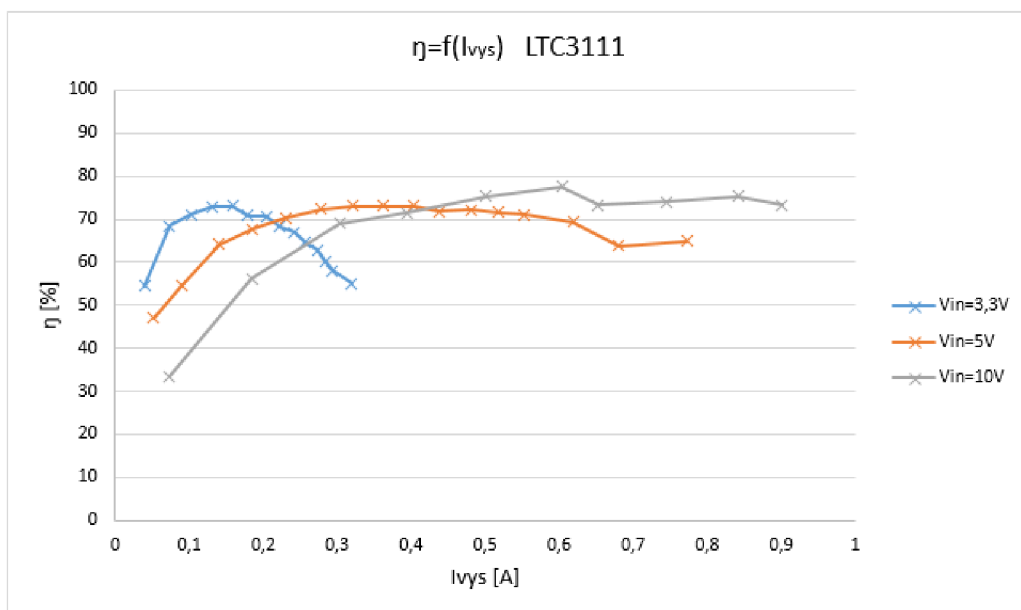
Obr. 5.2: TPS54395 - Graf závislosti efektivity na výstupnom prúde

Raspberry Pi pracuje v rozsahu odoberaného prúdu 400 mA až 700 mA, záleží od vyťaženia procesora a prídavných periférií. Porovnávané boli hlavne hodnoty efektivity v tomto rozsahu.

Vybratý bol menič TPS54395 od výrobcu Texas Instruments, ktorý mal najvyššiu nameranú efektivitu z funkčných meničov, vo veľkom rozsahu vstupného napätia a odoberaného prúdu.



Obr. 5.3: TPS63060 - Graf závislosti efektivity na výstupnom prúde

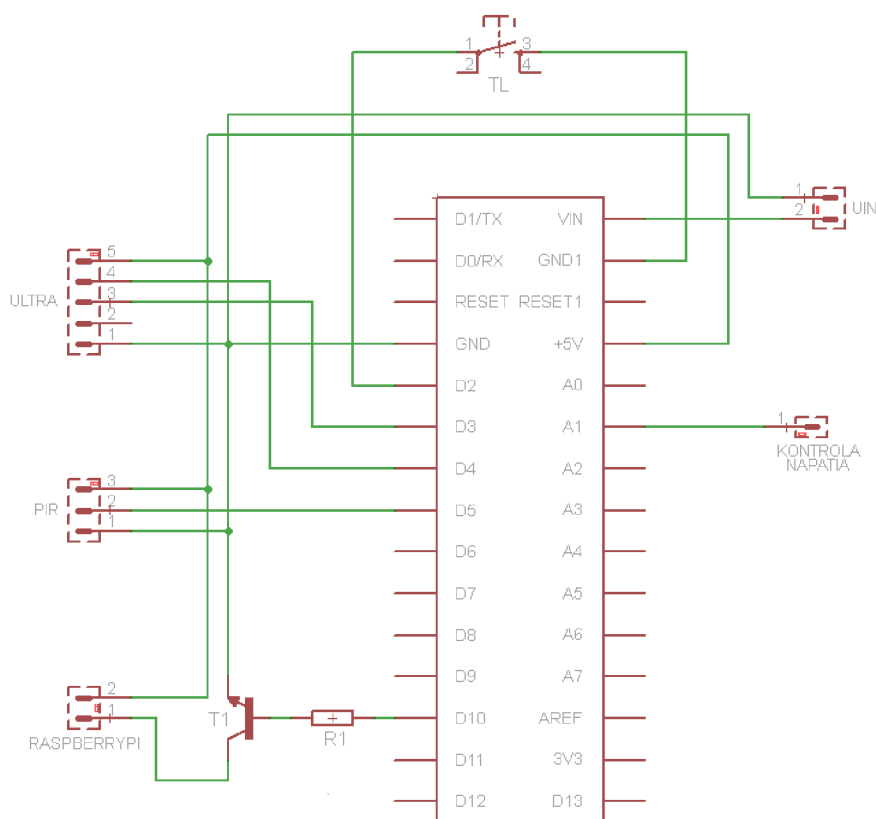


Obr. 5.4: LTC3111 - Graf závislosti efektivity na výstupnom prúde

5.4 Výber mikrokontroléra

Pri napájaní z batérie má mikrokontrolér za úlohu riadiť vstupné napájanie pre Raspberry Pi, čo je riešené pomocou tranzistora. Mikrokontrolér sleduje pomocou prídavných senzorov pohyb v zabezpečovanej oblasti a pri zaznamenaní pohybu zapne Raspberry Pi pomocou tranzistora. Celé zapojenie môžeme vidieť na obr. 5.5.

Ako mikrokontrolér bol vybratý model z rady Arduino, konkrétne Arduino Nano. Výber bol urobený na základe už získaných skúseností s týmto zariadením a aj preto, že daný model je veľmi jednoduchý a ponúka mnoho možností. Hlavnou výhodou je, že obsahuje v sebe zabudovaný programátor, ktorý nám umožňuje ľahké programovanie procesora, len za pomoci USB portu. Taktiež výrobca poskytuje vlastný softvér pre programovanie svojich zariadení, obsahujúci už vopred naprogramované príklady na ovládanie základných zariadení ako napríklad tlačítko alebo displej.



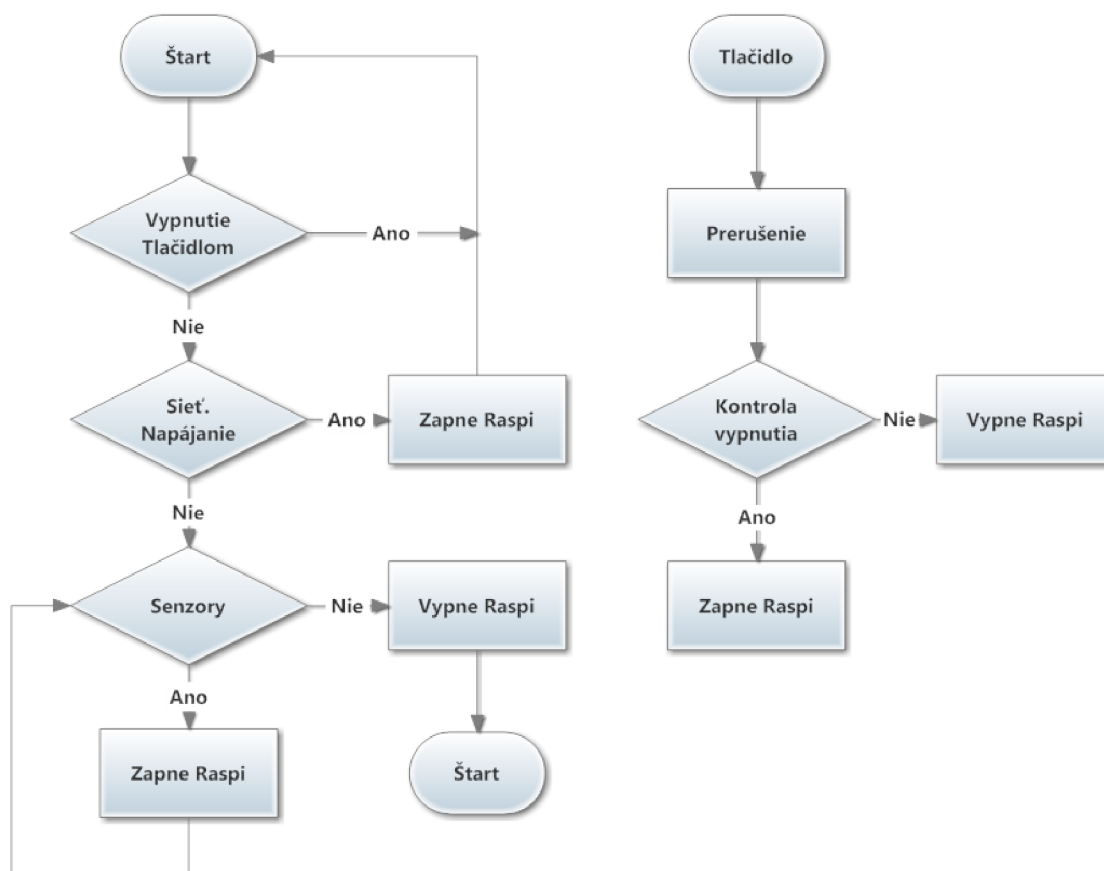
Obr. 5.5: Schéma zapojenia Arduina s prídavnými komponentami

Pre toto zapojenie bolo urobené meranie prúdového odberu pri napájaní z 5 V zdroja. Celé zapojenie vrátane prídavných senzorov má odber 49 mA.

5.5 Návrh programu pre Arduino

Táto podkapitola sa zaoberá návrhom obslužného programu pre senzory, ktorý bude riadiť celé zariadenie. Je tu popísaný celý postup ako sa program vytvára, od počítačného návrhu až po samostatné programovanie.

Prvý a najdôležitejší krok je zostaviť vývojový diagram. Ten hovorí o tom ako celý program bude fungovať a čo bude program ovládať. Požiadavkou tejto práce je aby mikroprocesor ovládal prídavné senzory, tlačítko pre prípadné vypnutie a zapnutie zariadenia a tranzistor, ktorým bude ovládané napájacie napätie pre Raspberry Pi.



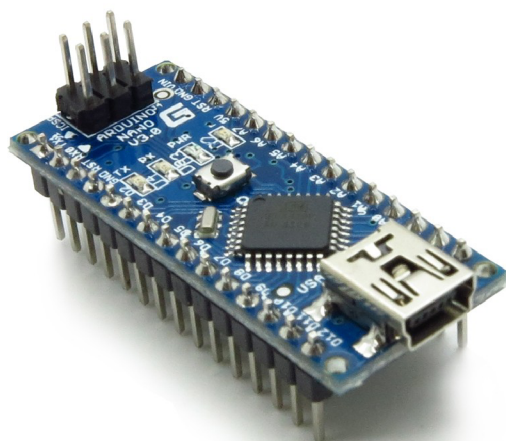
Obr. 5.6: Vývojový diagram

Program je navrhnutý tak aby nepretržite kontroloval vstupné napájacie napätie, pomocou vstupného analógového pinu Arduino a pri výpadku napájacieho napätia zo siete začal kontrolovať prídavné senzory.

Detekcia pohybu pomocou ultrazvuku je založená na meraní vzdialenosti. Keď sa v danom objekte nachádza pohyblivý predmet, senzor detekuje zmenu meranej vzdialenosti a tým program zareaguje a zapne Raspberry Pi. Ak dané senzory nedeckujú žiadny pohyb, Arduino vyšle riadiaci pulz do Raspberry Pi, ktorý prikazuje

počítaču aby sa korektne vypol. Následne počká na vypnutie a odpojí napájacie napätie. Naopak ak je počítač vypnutý a senzory zaznamenajú pohyb, Arduino zopne tranzistor a počká na riadiaci pulz od Raspberry Pi, ktorý informuje o zapnutí a spravení fotografie. Počas celej tejto akcie mikroprocesor musí kontrolovať ďalší pohyb. Pri detekovaní pohybu Arduino stále čaká a drží počítač zapnutý.

Počas celého programu je nepriamo kontrolované tlačidlo. Toto umožňuje funkcia interrupt-prerušenie, ktorá pri stlačení tlačidla preruší celý program a vykoná danú úlohu. Pri programovaní prerušenia treba brať do úvahy, že podprogram prerušenia nepodporuje časové slučky, čo je využívané pri ošetrovaní tlačítka proti zákmitom. V tomto prípade prerušenie zabezpečuje vypnutie a zapnutie počítača. Ak je počítač zapnutý, pri stlačení tlačidla sa korektne vypne, ako už bolo popísané vyššie a zostane vypnutý dokým sa opätovným stlačením nezapne. Pri zapnutí program kontroluje prídavné senzory a pri nedetegovaní pohybu opätovne počítač vypne.



Obr. 5.7: Arduino Nano

5.6 Battery management

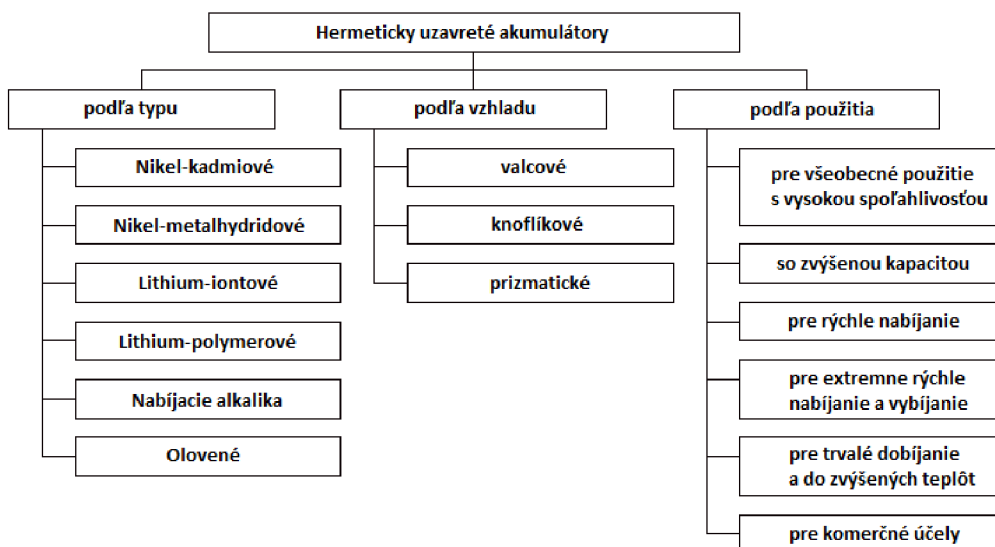
Battery management má za úlohu v tomto zariadení správne nabíjať záložnú batériu a pri výpadku napájacieho napätia musí napájanie automaticky prepnúť na záložnú batériu. Opätovne pri obnovení napájacieho napätia zo siete musí zaobstarat nabíjanie batérie.

Výrobcovia vyrábajú integrované obvody, ktoré vyššie uvedené požiadavky spĺňajú. Integrované súčiastky sa líšia rôznymi parametrami. Hlavnými parametrami sú, aký druh batérií a koľko článkov naraz dokážu nabíjať. Väčšinou jeden obvod dokáže spolu nabíjať Lithium-Ionové a Lithium-Polymerové, Ni-Cd a Ni-MH alebo zvlášť olovené batérie.

5.6.1 Akumulátory

Elektrické akumulátory sú chemické zdroje, ktoré pri nabíjaní dokážu prijímať elektrickú energiu a ukladať ju ako chemickú a naopak pri vybíjaní meniť chemickú uloženú energiu na elektrickú.

V dnešnej dobe poznáme viacero druhov batérií. Rozdelenie hermeticky-uzavretých batérií, s ktorými sa bežne stretávame, je zobrazené na obr. 5.8.



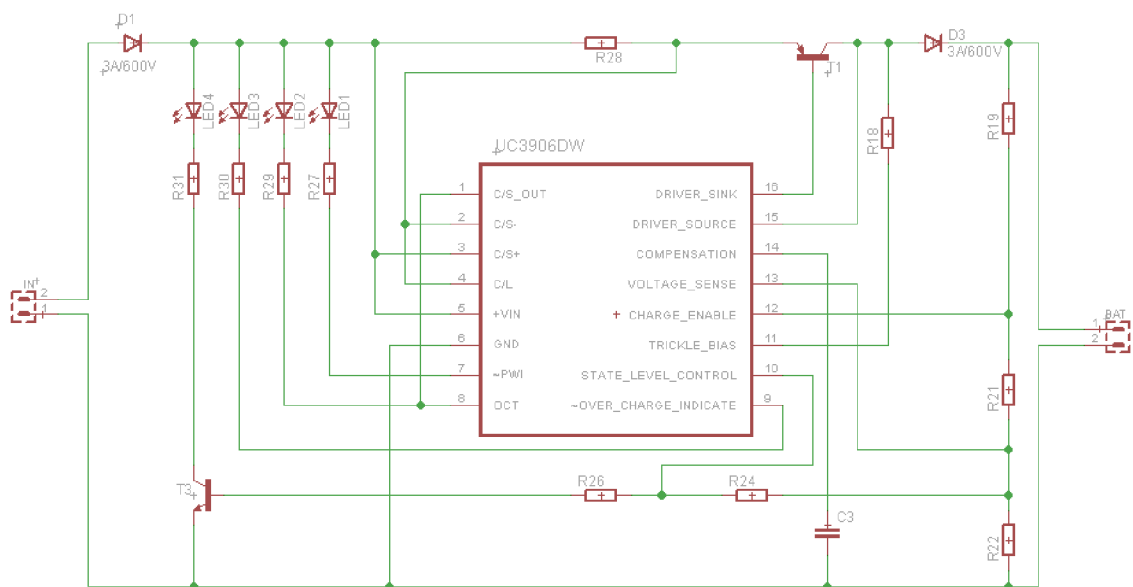
Obr. 5.8: Rozdelenie akumulátorov [13].

V zabezpečovacích zariadeniach sa využívajú väčšinou olovené akumulátory, pretože majú veľkú kapacitu, nízku cenu a dlhú životnosť. Vzhľadom na to, že celé zariadenie potrebuje 5 V vstupné napätie a pri nabenutí systému odoberá prúd okolo 0,6 A, bol vybratý 12 V olovený akumulátor s kapacitou 1,3 Ah. Každopádne kapacita akumulátora by mala byť vyššia, ale pre zníženie nákladov bol vybraný kompromis.

5.6.2 Výber nabíjača

Aby nedochádzalo k prebíjaniu nabíjaného akumulátora musí jadro nabíjačky tvoriť integrovaný obvod ktorý riadi celé nabíjanie na základe merania veličín batérie (napätie, prúd, teplota...).

Výber bol urobený na základe jednoduchosti, spoľahlivosti a hlavne dostupnosti použitých súčiastok. Z dostupných zdrojov na internete bol vybraný nabíjač s integrovaným obvodom s označením UC3906 od Texas Instruments (viď obr.5.9), čo znížilo výdavky, pretože ako už bolo spomínané firma ponúka rýchle a bezplatné poslanie vzorkou[10].



Obr. 5.9: Schéma zapojenia nabíjačky

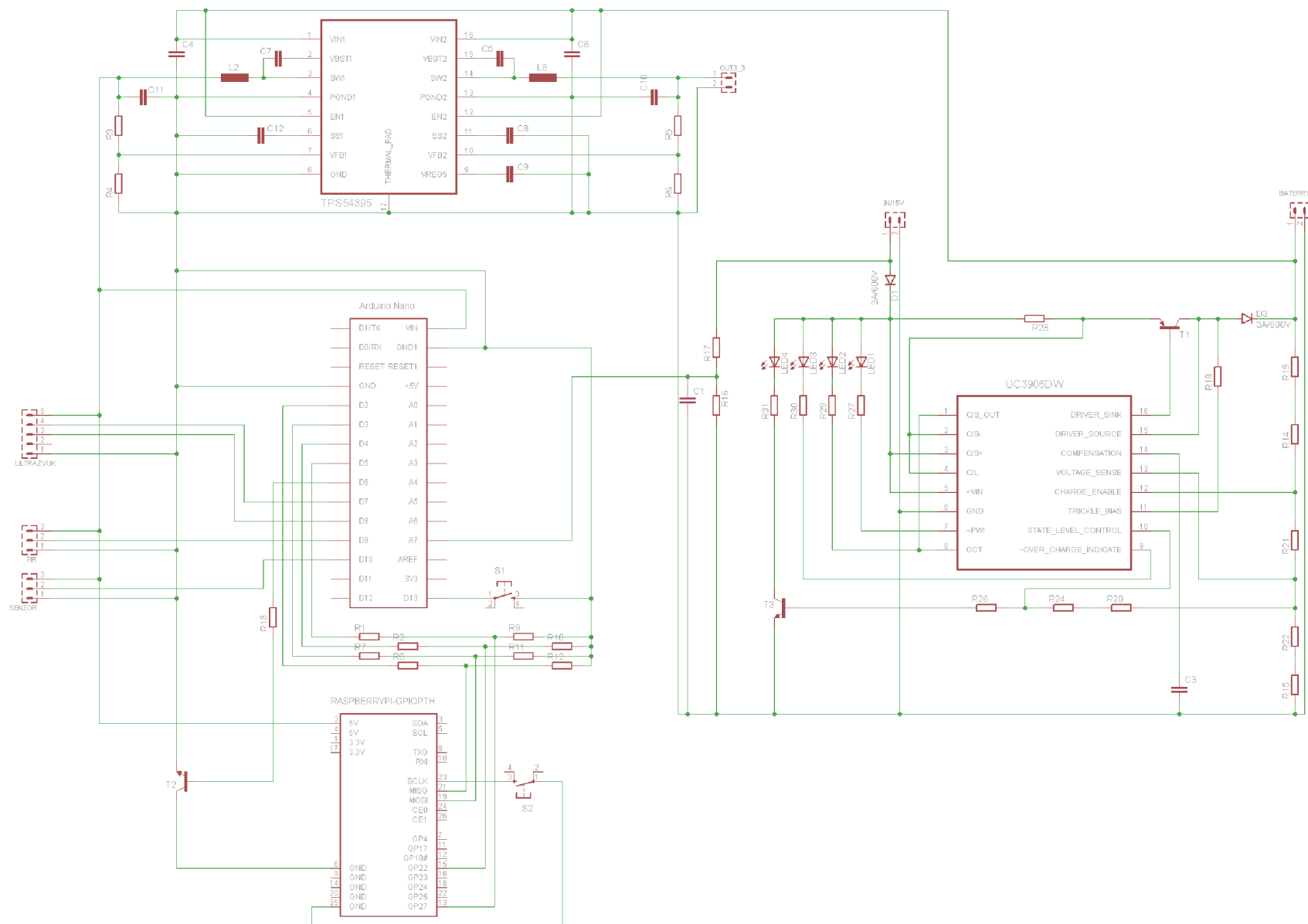
Obvod obsahuje signalizačné LED diódy, ktoré signalizujú proces nabíjania, následne výkonový rezistor R28, ktorý slúži ako snímač prúdu pre integrovaný obvod, ktorý pomocou výkonového prvku T1 reguluje napätie a prúd. Rezistory R19 až R22 slúžia ako snímač napätia jednotlivých fáz nabíjania akumulátora. Maximálny nabíjací prúd, ktorý je závislý na kapacite akumulátora je nastavený pomocou výkonového rezistora R28.

5.7 Návrh DPS rozširujúceho modulu

V tejto podkapitole je popísaný celý konečný návrh DPS rozširujúceho modulu s vybratými súčiastkami. Komponenty, ktoré obsahuje doska už boli popísané v predchádzajúcich kapitolách.

Celá doska, viď. schéma obr.5.10, obsahuje vstupné svorky pre pripojenie vstupného napájania a záložnej batérie, svorky pre pripojenie prídavných senzorov, nabíjač olovených akumulátorov, DC/DC menič, tlačítka pre zapnutie, vypnutie a reštart systému a dutinkové lišty pre pripojenia Arduina a Raspberry Pi.

Zo vstupných svoriek je privedené napájacie napätie na nabíjačku s integrovaným obvodom UC3906, ktorý pri zapojení napájacieho zdroja nabíja batériu a taktiež napája celé zariadenie. Ak je zdroj odpojený, napájanie preberá batéria. O stabilizáciu a zmenu napätia na požadovaných 5 V sa stará DC/DC menič TPS54395, ktorý napája mikropočítač s prídavnými senzormi. Mikropočítač Arduino Nano pomocou tranzistora T1 ovláda vstupné napätie pre Raspberry Pi. Arduino Nano cez analógový vstup A7 deteguje pomocou napätového deliča na vstupe napájania (R17, R16), pripojené napájacie napätie. Tlačítko S1, pripojené k Arduinu na vstupný digitálny pin D13, ovláda zapnutie a vypnutie Raspberry Pi. Komunikácia medzi Arduinom a Raspberry Pi je zabezpečená pomocou štyroch digitálnych portov. Cez jeden port Arduino posiela signál pre korektné vypnutie Raspberry Pi, ostatné porty sú pre budúcu doplňujúcu komunikáciu medzi týmito modulmi. Vzhľadom na to, že vstupné digitálne porty Raspberry Pi dovoľujú priviesť maximálne napätie 3,3 V, musia byť prepojené na Arduino cez napätový delič. Je to z dôvodu, že Arduino pracuje na 5 V logike. Tlačítkom S2, ktoré je pripojené na Raspberry Pi je ovládaný reštart operačného systému. Návrh DPS je umiestnený v prílohách tejto práce C.



Obr. 5.10: Schéma zapojenia rozširujúceho modulu

6 OPERAČNÝ SYSTÉM

Ako už bolo spomenuté, operačné systémy pre Raspberry Pi majú veľký čas spustenia, čo je pre naše riešenie nežiadúce. Pri napájaní z batérie sa bude Raspberry Pi vypínať a pri detegovaní pohybu senzormi opätovne zapínať. To znamená, že je nutné aby sa počítač zapol a urobil fotografiu za čo najkratší čas. V tabuľke 6.1, je možné vidieť časy spustenia a priemerné prúdové odbery po zapnutí oficiálnych operačných systémov ponúkaných výrobcom. Časy sú merané bez pripojenia prídavných periférií k Raspberry Pi a namerané prúdové odbery sú merané s pripojeným wifi adaptérom a kamerou.

Tab. 6.1: Namerané časy spustenia a odbery OS

OS	čas spustenia [s]	priemerný odber [mA]	odber po vypnutí [mA]
Raspbian	35,2	610	110
Pidora	35,5	470	300
Openelec	37,3	550	146
Raspbmc	45,7	430	205
Risc OS	39	510	130

Pri pripojení prídavných periférií (kamera, klávesnica, wifi adapter) sa časy spustenia systému menia minimálne (max 1 s). Pri používaní wifi adaptéru a požiadavke prístupu na internet sa môže čas spustenia navýšiť približne o 10 sekúnd. Záleží na type operačného systému a type internetového pripojenia.

Periférie ovládané operačným systémom:

- Raspberry Pi
 - model B
 - rev. 2 (2011.12)
- USB wifi adaptér
 - Chipset RTL8191S
 - 802.11b/g/n
 - USB 2.0
- Kamera Raspberry Pi 1.3
 - 5 Mpx
 - Rozlíšenie 2592 x 1944
 - Podpora videa 1080 p a 30 fps, 720 p a 60 fps
 - 15-pin MIPI rozhranie

Existujú dve možnosti ako urýchliť operačný systém pre Raspberry Pi. Vypnutím nepotrebných služieb, ktoré sa spúšťajú pri štarte systému alebo vytvorením vlastného embedded operačného systému, ktorý bude podporovať iba potrebné služby.

6.1 Embedded systém

Embedded systém je jednoúčelový systém, v ktorom je riadiaci počítač zabudovaný do ovládaného zariadenia. Na rozdiel od univerzálnych počítačov sú zabudované počítače väčšinou jednoúčelové, určené na predom definované činnosti.

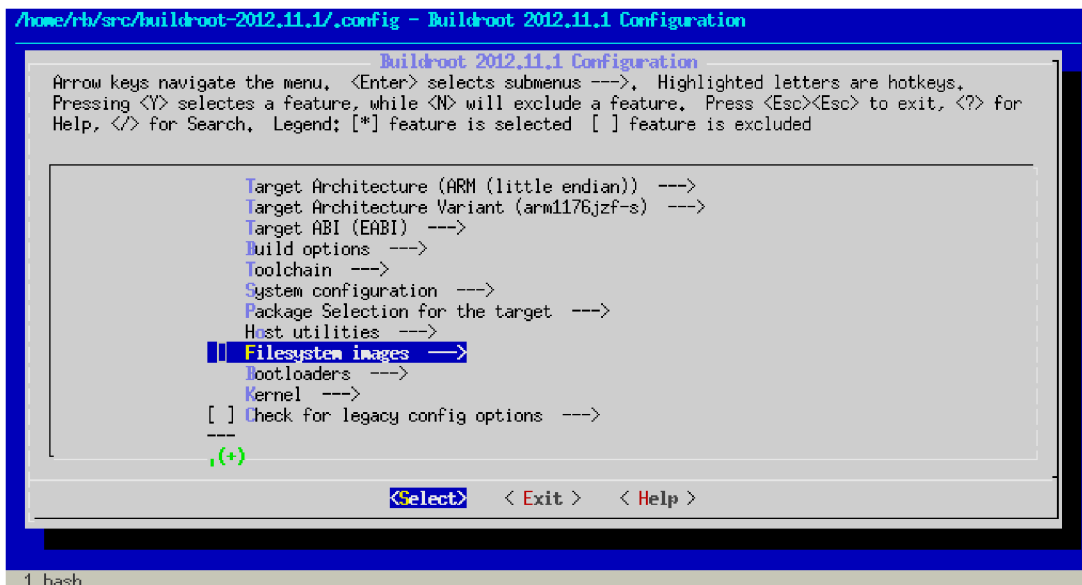
Embedded systémy založené na linuxovom jadre sa používajú v spotrebnej elektronike (set-top box, TV), v sieťových zariadeniach (smerovače), v priemyselnej automatizácii apod. [8].

6.1.1 Vytvorenie embedded systému

Pre vytvorenie vlastného embedded operačného systému na linuxovom jadre existuje nástroj Buildroot.

Buildroot je súbor, ktorý zjednodušuje a automatizuje proces vytvorenia kompletného linuxového systému. Umožňuje automaticky vytvoriť koreňový súborový systém, zostaviť obraz linuxového jadra a generovať zavádzač pre cieľový embedded systém. Je primárne určený pre použitie v embedded systémoch založených na rôznych počítačových architektúrach a inštrukčných sadách (x86, ARM).

Pre prácu s Buildrootom a vytvorenie vlastného operačného systému je potrebné pracovať v Linuxovom operačnom systéme. Ako prvý krok pre vytvorenie systému je potrebné stiahnuť z oficiálnych stránok balík Buildroot. Po rozbalení a presunutí sa do adresára buildroot, pri zadaní príkazu *make manconfig* je následne otvorené konfiguračné okno 6.1.



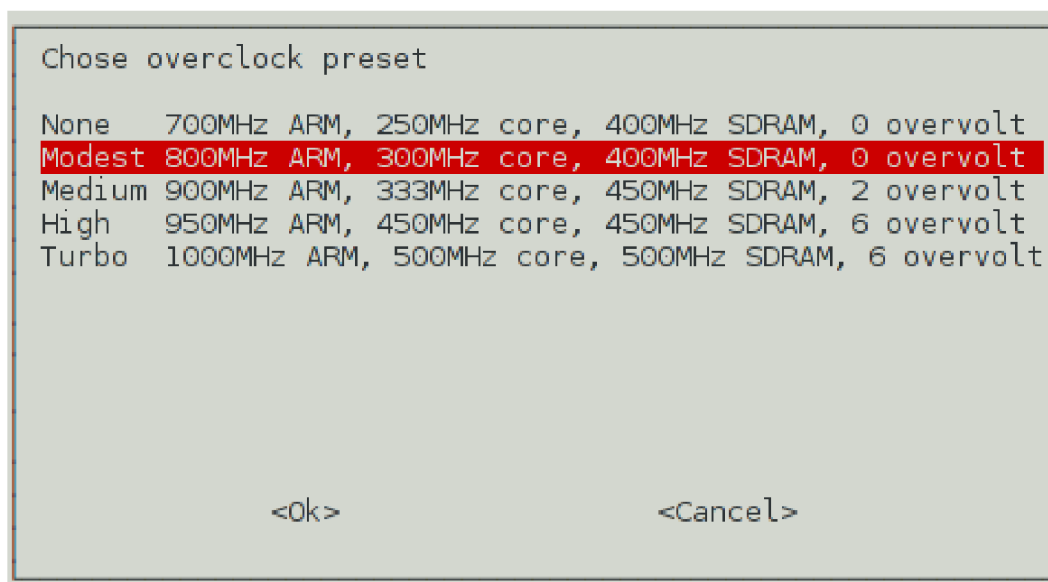
Obr. 6.1: Konfiguračné okno Buildroot[14]

Ako vidieť na obrázku 6.1, v konfiguračnom okne je možné nastaviť všetky parametre vlastného operačného systému. Je možné zvoliť architektúru cieľového počítača, kompilátor, druh súborového systému, verziu linuxového jadra apod. Taktiež je možné v konfiguračnom okne zvoliť základné programy a knižnice, ktoré budú súčasťou systému. Na oficiálnych stránkach Buildrootu je možné nájsť obsiahly manuál k nastaveniam celého systému[7]. Po nastavení požadovaných parametrov je potrebné pomocou príkazu *make all* spustiť kompiláciu systému.

6.2 Operačný systém Raspbian

Raspbian je voľný operačný systém založený na Debiane, optimalizovaný pre Raspberry Pi. Raspbian poskytuje viac ako operačný systém. Je dodávaný s viac ako 35000 prídavnými balíkmi[16]. V komunite Raspberry Pi je najobľúbenejším a najvyužívanejším systémom.

Zníženie času spustenia operačného systému je možné dosiahnuť pretaktovaním procesora na vyššie pracovné frekvencie. Zmenu frekvencií je možné urobiť pri prvom spustení operačného systému alebo následne zadaním príkazu *raspi-config*, po ktorom sa otvorí konfiguračné okno, kde je možné robiť rôzne nastavenia ako napríklad pretaktovanie, zmena užívateľského hesla, povolenie alebo zakázanie rôznych služieb atď. Pri otvorení ponuky *Overclock* sa otvorí okno s možnosťami pretaktovania vid. obr.6.2



Obr. 6.2: Menu pretaktovania

Taktiež zníženie času spustenia je možné dosiahnuť úpravou textového súboru `cmdline.txt` v adresári `/boot` kde zakážeme výpis systémových správ pri zapínaní systému. Úprava súboru je vidieť na obrázku 6.3

```
/boot/cmdline.txt
dwc_otg.lpm_enable=0 root=/dev/mmcblk0p2 rootfstype=ext4 rootflags=commit=120,data=writeback elevator=deadline
noatime nodiratime data=writeback rootwait quiet
```

Obr. 6.3: Súbor `cmdline.txt`

Ďalšia úprava systému pre zníženie času je úprava tzv. runlevelov v súbore `inittab`, v adresári `/etc`, kde sa nachádzajú konfiguračné súbory a inicializačné skripty. Skrátenú verziu súboru `inittab` a runlevely je vidieť na obr.6.4.

```
# Level to run in
id:2:initdefault:

# Boot-time system configuration/initialization script.
si::sysinit:/etc/rc.sysinit

# What to do in single-user mode.
~:S:wait:/sbin/sulogin

# /etc/init.d executes the S and K scripts upon change
# of runlevel.
#
# Runlevel 0 is halt.
# Runlevel 1 is single-user.
# Runlevels 2-5 are multi-user.
# Runlevel 6 is reboot.

l0:0:wait:/etc/rc 0
l1:1:wait:/etc/rc 1
l2:2:wait:/etc/rc 2
l3:3:wait:/etc/rc 3
l4:4:wait:/etc/rc 4
l5:5:wait:/etc/rc 5
l6:6:wait:/etc/rc 6
```

Obr. 6.4: Inicializačný súbor `inittab`

`Inittab` opisuje, ktoré procesy sú spustené pri štarte systému a pri bežnej prevádzke. Raspberry Pi využíva 6 runlevelov. Nultý a šiesty runlevel slúži na vypnutie alebo reštart systému, ktorý odkazuje na skripty umiestnené v adresároch `/etc/rc0` a `/etc/rc6`. Runlevely 2-5 sú pre užívateľov. Štandardne je pri spustení ako prvý runlevel nastavený runlevel 2, ktorý odkazuje na adresár `/etc/rc2`, kde sú umiestnené skripty pre spustenie daných služieb. Pre zníženie času je možné v súbore `inittab` vymazať runlevely 3-5 a znemožniť systému prístup do ďalších užívateľských runlevelov. Následne je nutné upraviť adresár `/etc/rc2` tak, aby obsahoval skripty programov ktoré sú potrebné pre naše riešenie a chod systému. Pri testovaní nebola táto úprava

vykonaná pretože je veľmi obtiažna a vyžaduje znalosť princípu všetkých spúšťaných služieb operačného systému Raspbian.

Najväčšie zníženie času štartu dosiahneme nainštalovaním a nastavením programu Systemd. Systemd je sada spravujúca démonov, knižnice a utility určená pre konfiguráciu a správu linuxových operačných systémov. Je to náhrada init systému linux - prvý proces vykonávaný v užívateľskom priestore počas spustenia systému. Umožňuje paralelné spúšťanie démonov užívateľských programov, čo zaručí rýchly štart systému[6].

Vykonaním týchto úprav dosiahneme zníženie času štartu operačného systému Raspbian na 13 s čo pre toto riešenie nieje postačujúce.

6.3 Konečný návrh OS

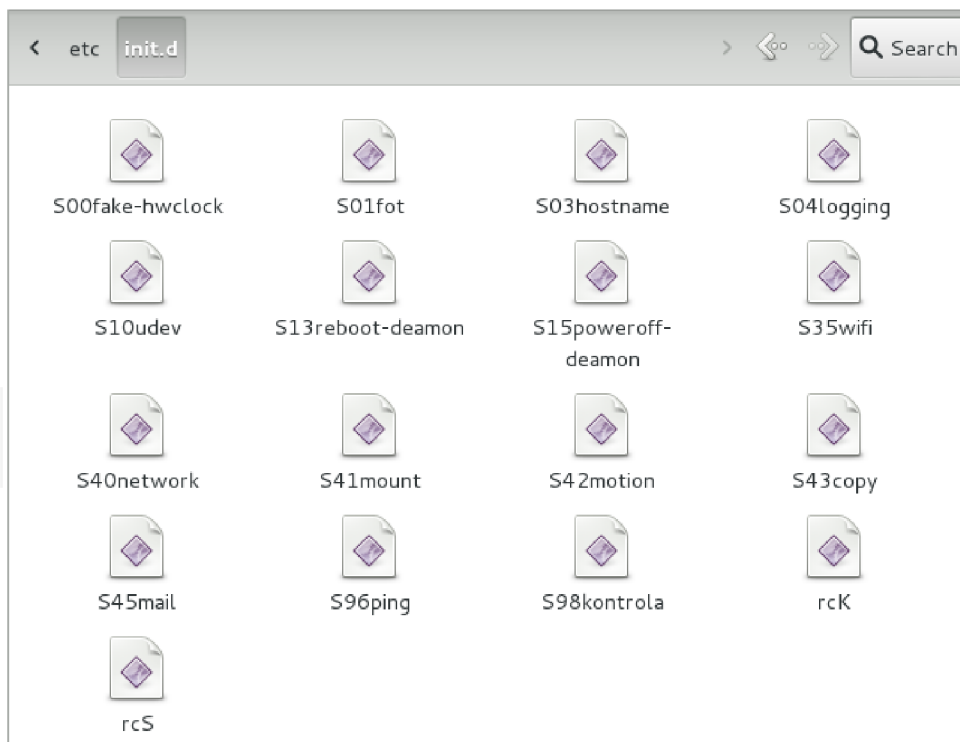
V tejto podkapitole bude opísaný konečný návrh operačného systému pre toto riešenie, s optimalizovaným časom štartu a so všetkými ovládacími skriptami.

Na internete je možné nájsť mnoho neoficiálnych operačných systémov pre Raspberry Pi. Jedným z najzaujímavejších pre toto riešenie je Motion Pie. Tento systém je voľne sťahovateľný, vytvorený pre obsluhu Raspberry Pi a jeho prídavnú kameru. Bol vytvorený pomocou nástroja Buildroot a jadro obsahuje všetky ovládače pre kameru a wifi usb kartu. Taktiež obsahuje nainštalovaný softvér Motion detection, ktorý pri detegovaní pohybu zaznamená fotku.

Konečný návrh obsahuje jadro operačného systému Motion Pie, program Motion detection a prídavné skripty ktoré riadia celý systém. Systém je upravený tak, aby bola pri spustení Raspberry Pi čo najrýchlejšie zaznamenaná fotka. Systémy vytvorené pomocou nástroja Buildroot nepodporujú spúšťanie systému v runleveloch. Pri spúšťaní systému sú postupne spúšťané skripty umiestnené v adresári */etc/init.d*. Skripty umiestnené v tomto adresári musia byť označené písmenom *S* a poradovým číslom v akom poradí budú spúšťané^{6.5}.

Po privedení napájacieho napätia do Raspberry Pi, sa do 5 s začne každých 500 ms zaznamenávať fotografia. Zaznamenávanie fotografie trvá až pokiaľ sa ne načíta wifi zariadenie a nepripojí k sieti. To trvá 22 s od zapnutia napájania. Po pripojení na sieť sa spúšťa Motion detection, ktorý pri zaznamenaní pohybu urobí fotografiu. Následne sa Raspberry Pi pripojí na vzdialený webový server kam odošle všetky zaznamenané fotky. Po 27 s od štartu sú presúvané fotografie do dátového úložiska. V našom prípade pre simuláciu dátového úložiska je použitý notebook, na ktorom je spustený webový server. Následne je odoslaný e-mail klientovi s upozornením, že bol zaznamenaný pohyb. Do 30 s od štartu je umožnené majiteľovi prehliadať zaznamenané fotografie na webovej stránke.

Pri testovaní zariadenia bolo zistené, že pri dlhšom nepretržitom chode Raspberry Pi a dlhším nepoužívaním wifi rozhrania, zlyhá pripojenie k internetu. Pre zamedzenie tohto zlyhania je každých päť minút urobený ping na známu ip adresu. V našom prípade je to Google.



Obr. 6.5: Skripty spúšťané pri štarte

- Skript S00fake-hwclock je použitý z operačného systému Raspbian a slúži na nastavenie hodín, podľa ktorých sú pomenované neskôr zaznamenané fotografie.
- Skript S01fot spúšťa démona skriptu s názvom fotka, ktorý je umiestnený v adresári `/home/pi/scripty_fotka`. Tento skript slúži pre spustenie programu raspistill, ktorý každých 500 ms zaznamenáva fotografiu.
- Skript S13reboot-deamon spúšťa démona skriptu reboot umiestneného v adresári `/home/pi/scripty_reboot`, ktorý obsluhuje pomocou GPIO portu prídavné tlačítko na reštart systému.
- S15poweroff-deamon má podobnú funkciu. Ak pošle Arduino na zvolený pin GPIO portu pulz, tak dá príkaz Raspberry Pi aby sa korektné vypol.
- O nastavenie wifi a pripojenie k sieti sa starajú skripty S35wifi a S40network, ktoré sú použité zo systému Motion Pie.
- Skript S41mount slúži na pripojenie k dátovému úložisku, v našom prípade zdieľaný priečinok na sieti.

- Pre zapnutie programu motion detection slúži skript S42motion.
- S43copy je skript pre spustenie démona skriptu s názvom skript-copy, ktorý je umiestnený v adresári */home/pi/scripty_copy*. Tento skript slúži na presunutie doteraz zaznamenaných fotografií do dátového úložiska. Ak je zariadenie pripojené k internetu, všetky fotografie sa presunú do dátového úložiska. Ak by nastalo zlyhanie pripojenia na internet, fotografie zostávajú uložené na SD karte v Raspberry Pi.
- Skript S98kontrola spúšťa démona programu s názvom script-find, ktorý je umiestnený v adresári */home/pi/scripty_mail*. Tento skript nepretržite kontroluje priečinok kam sú ukladané fotografie a pri navýšení priečinka o jednu fotografiu upozorní majiteľa poslaním mailovej správy.

Všetky skripty je možné nájsť v prílohách tejto práce E.

Realizácia aplikácie pre operačný systém Android bola vynechaná, pretože pre zobrazenie zaznamenaných fotografií je vytvorená galéria ako webová stránka, ktorú je možné zobrazit na akomkoľvek zariadení, ktoré má webový prehliadač a prístup k internetu. Zdrojový kód webovej stránky je možné vidieť v prílohách F.

6.3.1 Meranie spotreby finálneho OS

Pre finálny operačný systém bolo urobené meranie prúdového odberu. Meraný bol odber v čase spúšťania Raspberry Pi s pripojenou kamerou a wifi adaptérom.

Tab. 6.2: Nameraný prúdový odber v čase

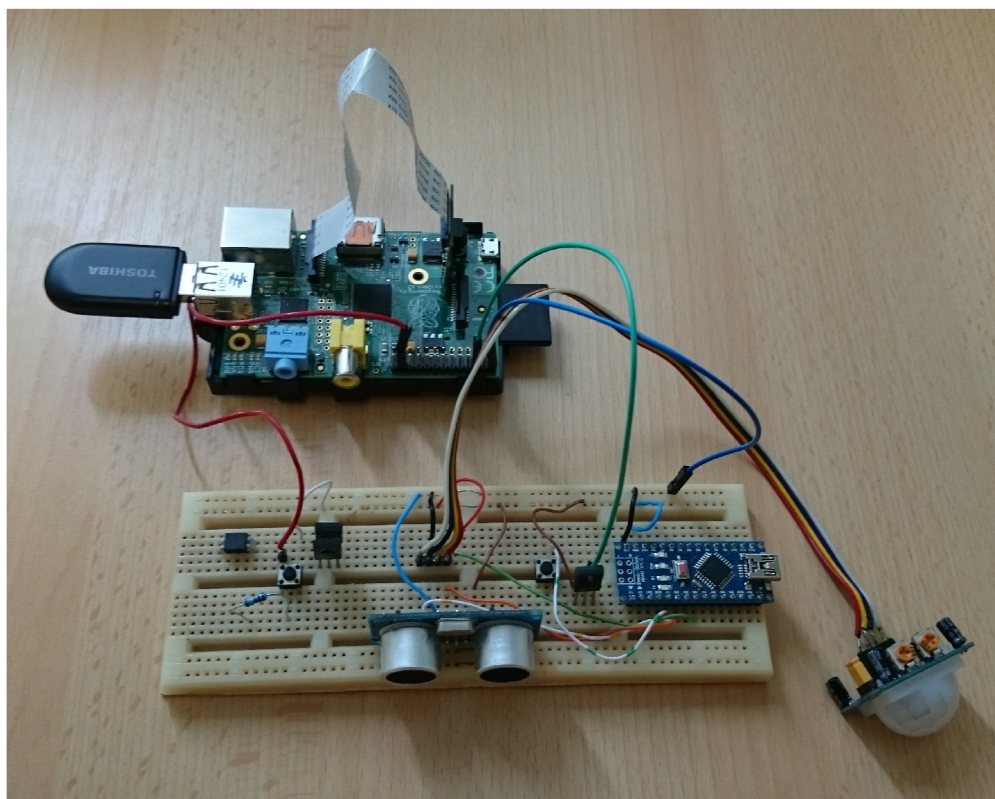
t[s]	2	4	6	8	10	12	14	16
I[mA]	126	375	377	399	565	666	572	689
t[s]	18	20	22	24	26	28	30	32
I[mA]	510	648	709	480	553	472	483	502

Tieto hodnoty odpovedajú vstupnému napájaciemu napätiu 5 V, pri ktorom bolo meranie urobené. Následne bolo urobené meranie prúdového odberu v čase spúšťania celého zariadenia s prídavnými senzormi, Arduinom a DC/DC meničom pripojeným na záložnú 12 V batériu vid. tab.6.3..

Tab. 6.3: Nameraný prúdový odber finálneho OS

t[s]	2	4	6	8	10	12	14	16
I[mA]	90	167	172	193	254	332	291	337
t[s]	18	20	22	24	26	28	30	32
I[mA]	241	325	351	244	265	257	260	264

Všetky merania a testy boli uskutočnené na vytvorenom testovacom prípravku zloženého z nepájivého pola a komponentov, ktoré boli vybraté touto prácou.



Obr. 6.6: Testovací přípravok

7 ZÁVER

Predmetom tejto bakalárskej práce bolo realizovať Bezdrôtovú IP kameru s pomocou Raspberry Pi, ktorá pri detekcii pohybu zaznamenáva fotografiu. Súčasťou kamery je rozširujúci modul, ktorý obsahuje prídavné senzory, správu napájania, stabilizáciu a nabíjanie záložnej batérie.

V prvej časti práce sú popísané všetky časti, ktoré obsahuje rozširujúci modul. V ďalšej časti je popísaný návrh rozširujúceho modulu a výber vhodných súčiastok. Keďže práca sa zaoberá energetickou efektívnosťou boli porovnávané všetky súčiastky podľa odoberaného prúdu a následne boli vybraté najvhodnejšie. Bolo urobené meranie efektivity DC/DC meničov a následne bol vybratý menič TPS54395, ktorého efektívnosť v prúdovom rozsahu 400 mA až 700 mA je v okolo 90 %. Taktiež je v tejto kapitole popísaný mikrokontrolér Arduino Nano a navrhnutý program pre správu senzorov a riadenie napájania pre Raspberry Pi.

V ďalšej časti práce sú popísané možnosti skrátenia štartu operačného systému, ktorý bude Raspberry Pi využívať. Následne bol navrhnutý konečný model operačného systému, ktorý dokáže do 5 s zaznamenať fotografiu, poslať ju do dátového úložiska a upozorniť majiteľa o zaznamenaní pohybu.

Pri vypnutom Raspberry Pi, keď úlohu snímania pohybu preberajú senzory, bol odber prúdu dosiahnutý na 30 mA. Dosiahnutý celkový odber prúdu pri zapnutom zariadení, napájanom z 12V batérie je v priemere 250 mA.

LITERATÚRA

- [1] Prispěvatelé Wikipedie, *Raspberry Pi* [online], Wikipedie: Otevřená encyklopedie, c2014, poslední aktualizácia 4. 11. 2014 [cit. 8. 11. 2014]. Dostupné z URL: <http://cs.wikipedia.org/w/index.php?title=Raspberry_Pi&oldid=11984718>>.
- [2] BLAŽEK, J. *Elektromagnetická kompatibilita DC/DC měničů* [online]. 2012, [cit. 2014-11-19]. Bakalárska práca. ZÁPADOČESKÁ UNIVERZITA V PLZNI, Fakulta elektrotechnická. Vedúci bakalárskej práce Zdeněk Kubík. Dostupné z URL: <<http://theses.cz/id/gzasy8/>>.
- [3] UPTON, Eben. *Raspberry Pi: uživatelská příručka. 1. vyd.* Brno: Computer Press, 2013, [cit. 2014-11-19]. 232 s. ISBN 978-80-251-4116-8.
- [4] BARSUKOV, Yevgen a Jinrong QIAN. *Battery power management for portable devices*. Boston: Artech House, c2013, xvi, 241 p. [cit. 2014-11-20]. Artech House power engineering series. ISBN 16-080-7491-9
- [5] VESELY, A. *Systemd* [online], Debian, c2013, poslední aktualizácia 28. 4. 2015 [cit. 17. 5. 2015]. Dostupné z URL: <<https://wiki.debian.org/systemd>>.
- [6] VYSKOČIL, M. *Systemd-vlastnosti a schopnosti* [online], Abclinux, c2011, poslední aktualizácia 23. 3. 2011 [cit. 17. 5. 2015]. Dostupné z URL: <<http://www.abclinuxu.cz/clanky/systemd-vlastnosti-a-schopnosti>>.
- [7] Buildroot developers, *Buildroot manual* [online], Buildroot, c2015, poslední aktualizácia 1. 3. 2015 [cit. 16. 5. 2015]. Dostupné z URL: <<http://buildroot.uclibc.org/downloads/manual/manual.html>>.
- [8] Prispěvatelé Wikipedie, *Buildroot* [online], Wikipedie: Otevřená encyklopedie, c2015, poslední aktualizácia 27. 3. 2015 [cit. 16. 5. 2015]. Dostupné z URL: <<http://en.wikipedia.org/w/index.php?title=Buildroot&oldid=653750742>>.
- [9] CALIN, Crisan. *MotionPie* [online], MotionPie, c2014, poslední aktualizácia 15. 3. 2015 [cit. 18. 5. 2015]. Dostupné z URL: <<https://github.com/ccrisan/motionPie/wiki>>.
- [10] SLÁNSKÝ, M. *Inteligentní nabíječka Pb akumulátorů - konstrukce* [online]. 2006 [cit. 12. 5. 2015]. Dostupné z URL: <<http://www.hw.cz/teorie-a-praxe/konstrukce/inteligentni-nabijecka-pb-akumulatoru-konstrukce.html>>.

- [11] *Raspberrypi.org* [online]. 2012 [cit. 12. 12. 2014]. Dostupné z URL: <<http://www.raspberrypi.org/wp-content/uploads/2012/04/Raspberry-Pi-Schematics-R1.0.pdf>>.
- [12] *Smart-prototyping.com* [online]. 2013 [cit. 13. 12. 2014]. Dostupné z URL: <<http://smart-prototyping.com/Ultrasonic-Module-HY-SRF05-Distance-Sensor.html>>.
- [13] *Battex.info* [online]. 2009 [cit. 13. 12. 2014]. Dostupné z URL: <<http://www.battex.info/hermeticke-akumulatory/rozdeleni-hermetickych-akumulatoru>>.
- [14] *cellux.github.io* [online]. 2013 [cit. 16. 4. 2015]. Dostupné z URL: <<http://cellux.github.io/articles/diy-linux-with-buildroot-part-1>>.
- [15] <http://www.ucsimply.cz> [online]. 2014 [cit. 16. 4. 2015]. Dostupné z URL: <<http://www.ucsimply.cz/elnx/sestaveni-linuxove-distribuce>>.
- [16] <http://www.raspbian.org> [online]. 2012 [cit. 16. 4. 2015]. Dostupné z URL: <<http://www.raspbian.org/FrontPage>>.

ZOZNAM SYMBOLOV, VELIČÍN A SKRATIEK

LDO	Linearny regulátor – Low-Drop Out
PIR	Pasívny infračervený senzor – Passive infrared sensor
DPS	Doska plošných spojov
Ni-Cd	Nikel-kadmiové batérie
Ni-MH	Nikel-metalhydridové batérie
U_{vst}	Vstupné napätie
U_{vys}	Výstupné napätie
I_q	Kľudový prúd
OS	Operačný systém
GPIO	Univerzálny vstup/výstup – General Purpose Input/Output
UART	Univerzálny asynchrónny prijímač/vysielač – Universal Asynchronous Receiver Transmitter
ARM	Advanced RISC Machines
I ² C	Sériové komunikačné rozhranie – Inter-Integrated Circuit
SPI	Sériové periférne rozhranie – Serial Peripheral Interface

ZOZNAM PRÍLOH

A	Obsah priloženého CD	46
A.1	Schémy a návrhy DPS	46
A.2	Program pre Arduino	46
A.3	Operačný systém	46
A.4	Webová stránka	46
B	Návrhy DC/DC meničov	47
C	Rozširujúci modul	54
D	Zdrojový kód programu	55
E	Skripty	60
E.1	Skript fotka	60
E.2	Skript poweroff.py	60
E.3	Skript reboot.py	60
E.4	Skript script-copy	61
E.5	Skript mail	61
E.6	Skript script-find	61
E.7	Skript ping	62
E.8	Skript S41mount	62
F	Webová stránka	63

A OBSAH PRILOŽENÉHO CD

A.1 Schémy a návrhy DPS

V priloženom CD sa nachádzajú všetky schémy a návrhy DPS vytvorené pre túto prácu. Schémy a návrhy DPS pre DC/DC meniče sú uložené v priečinkoch s názvami „DCDC“ a „DCDC1“. Konečný návrh rozširujúceho modulu je umiestnený v priečinku „finalHW“.

Všetky návrhy boli vytvorené pomocou programu EAGLE verzii 7.2.0.

A.2 Program pre Arduino

V priečinku s názvom „Ardprg“ sa nachádza celý program pre Arduino Nano. Program bol vytvorený vo vývojovom prostredí Arduino verzie 1.6.1.

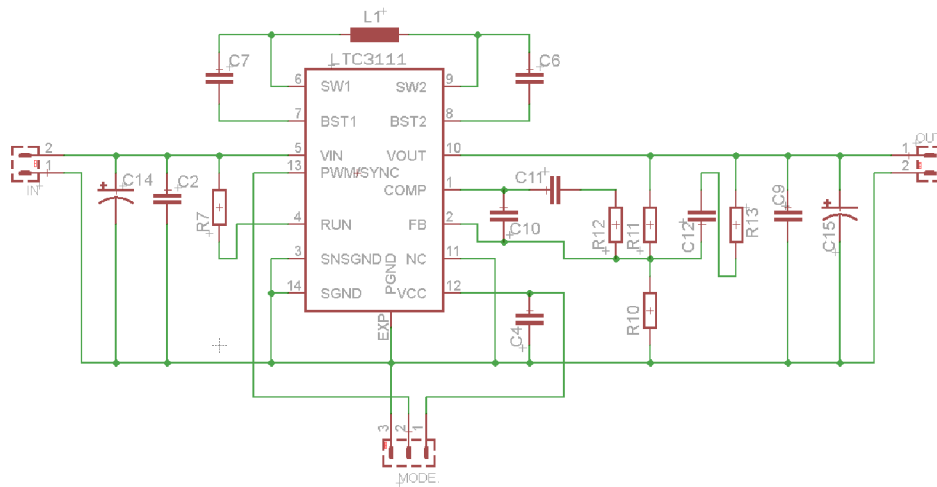
A.3 Operačný systém

Taktiež sa na CD nachádza celý operačný systém, vrátane skriptov popísaných v tejto práci. Systém je uložený v priečinku s názvom „OS“. Pre spustenie systému je potrebné prekopírovať všetky časti systému na zvolené médium (SD karta, USB kľúč), ktoré bude Raspberry Pi využívať ako pevný disk.

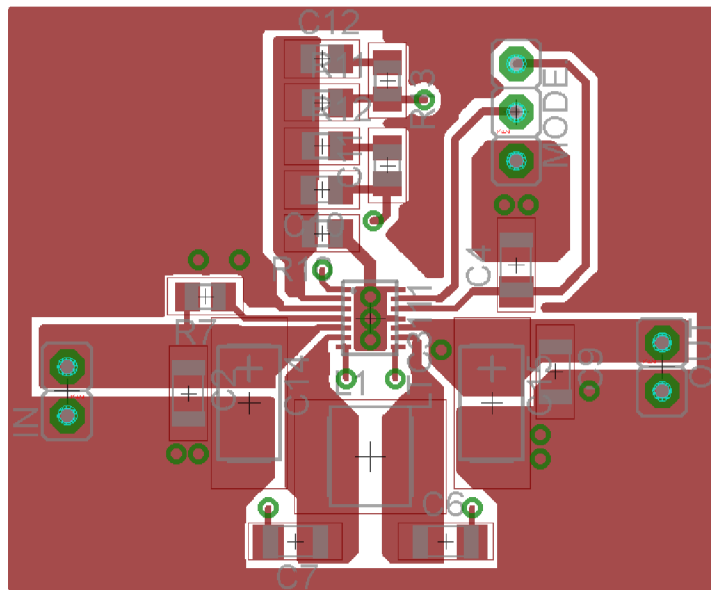
A.4 Webová stránka

V priečinku s názvom „Php“ sa nachádza zdrojový kód vytvorenej webovej galérie a priečinkov s názvom „img“, kde je príklad zachytených obrázkov kamerou.

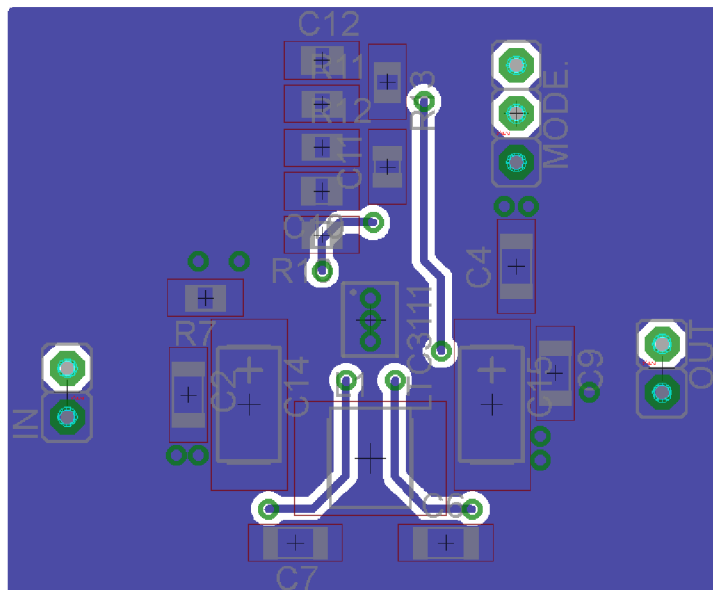
B NÁVRHY DC/DC MENIČOV



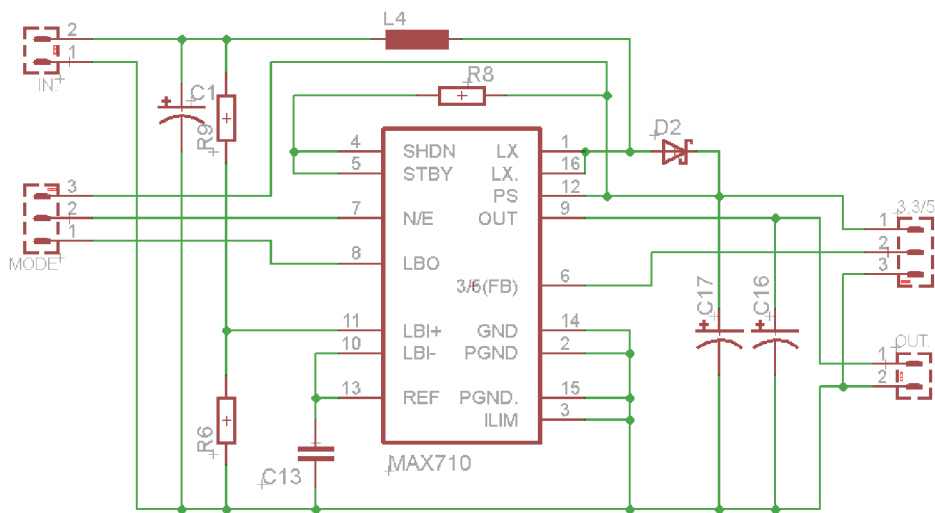
Obr. B.1: Schéma zapojenia DC/DC meniča LTC3111



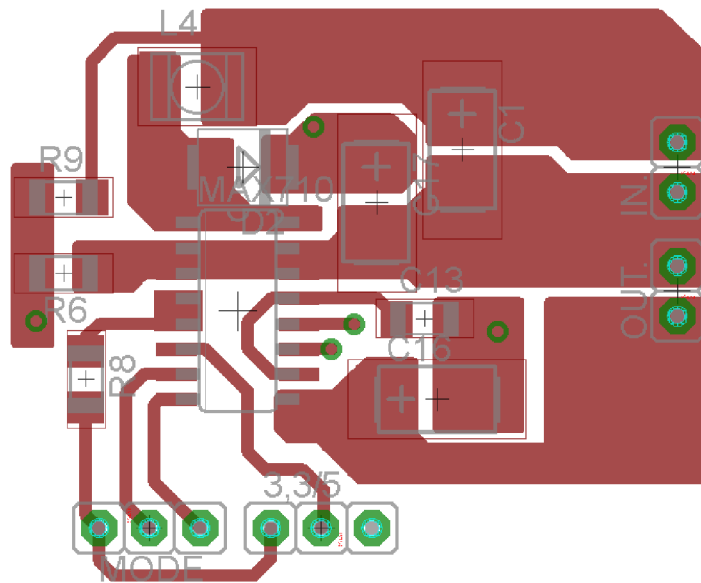
Obr. B.2: Návrh DPS meniča LTC3111 - vrchná strana



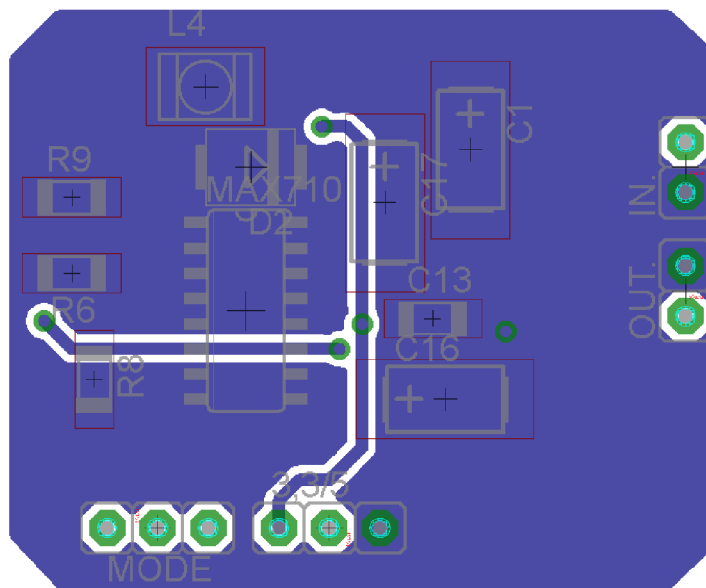
Obr. B.3: Návrh DPS meniča LTC3111 - spodná strana



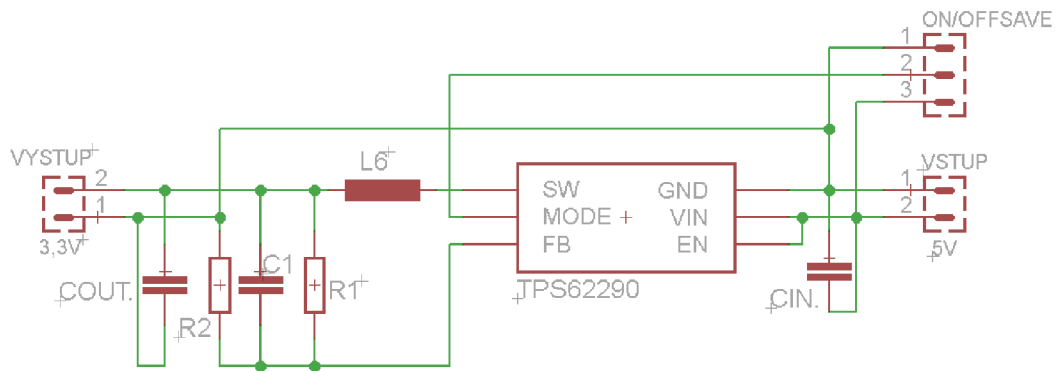
Obr. B.4: Schéma zapojenia DC/DC meniča MAX710



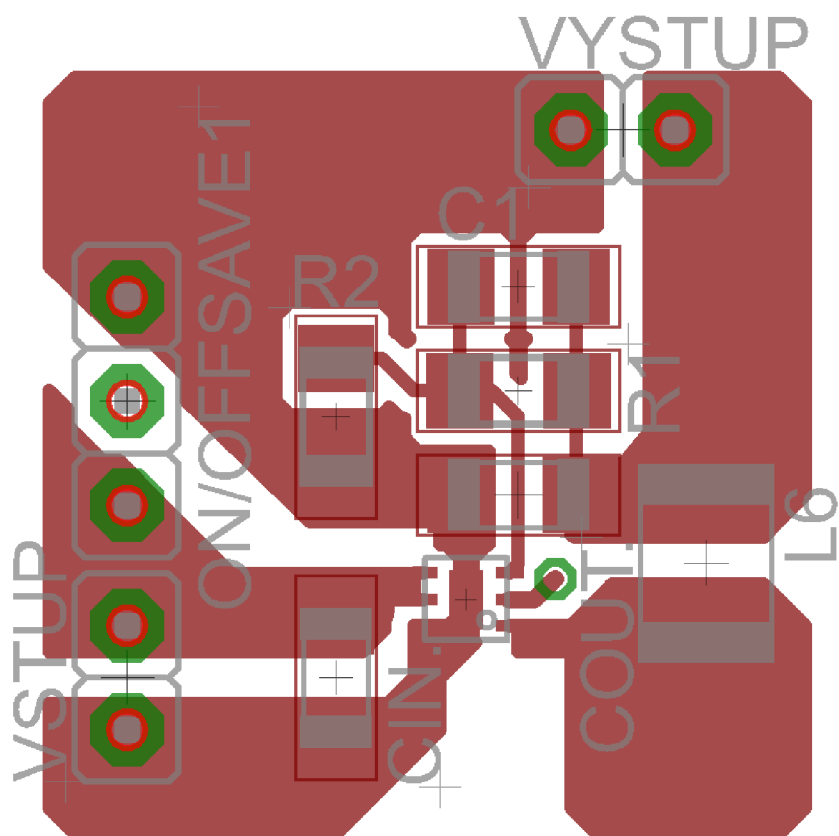
Obr. B.5: Návrh DPS meniča MAX710 - vrchná strana



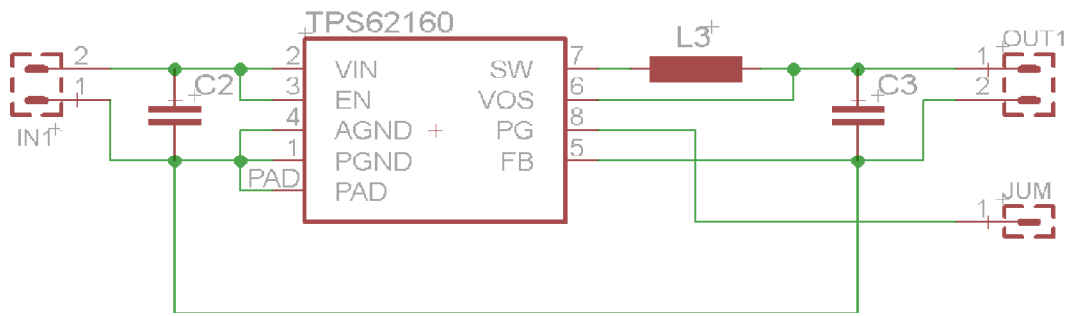
Obr. B.6: Návrh DPS meniča MAX710 - spodná strana



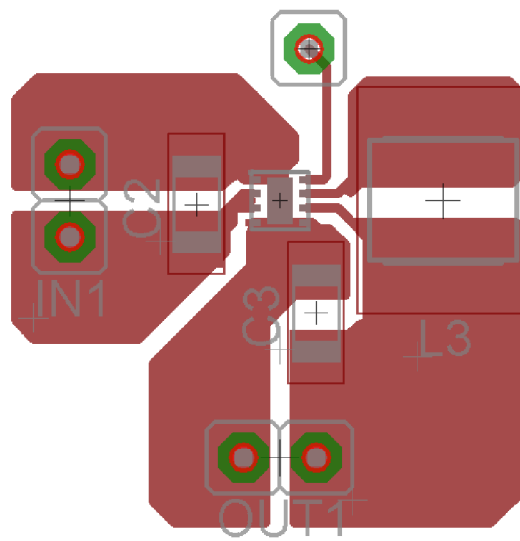
Obr. B.7: Schéma zapojenia DC/DC meniča TPS62290



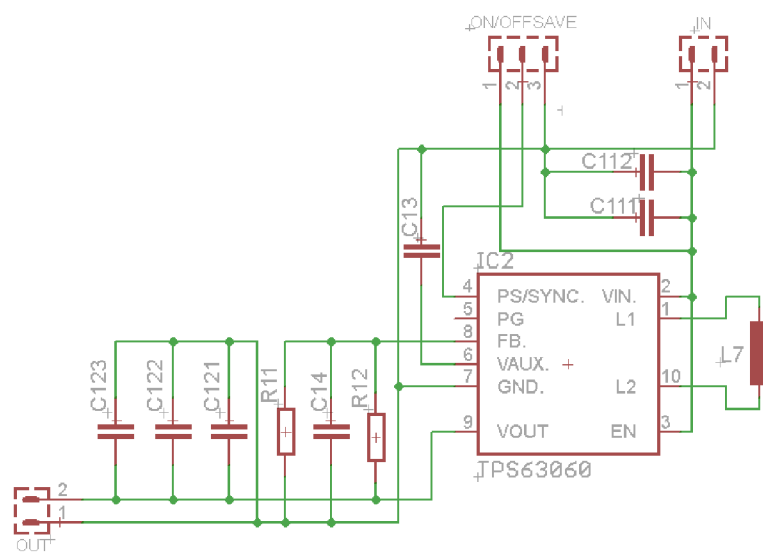
Obr. B.8: Návrh DPS meniča TPS62290 - vrchná strana



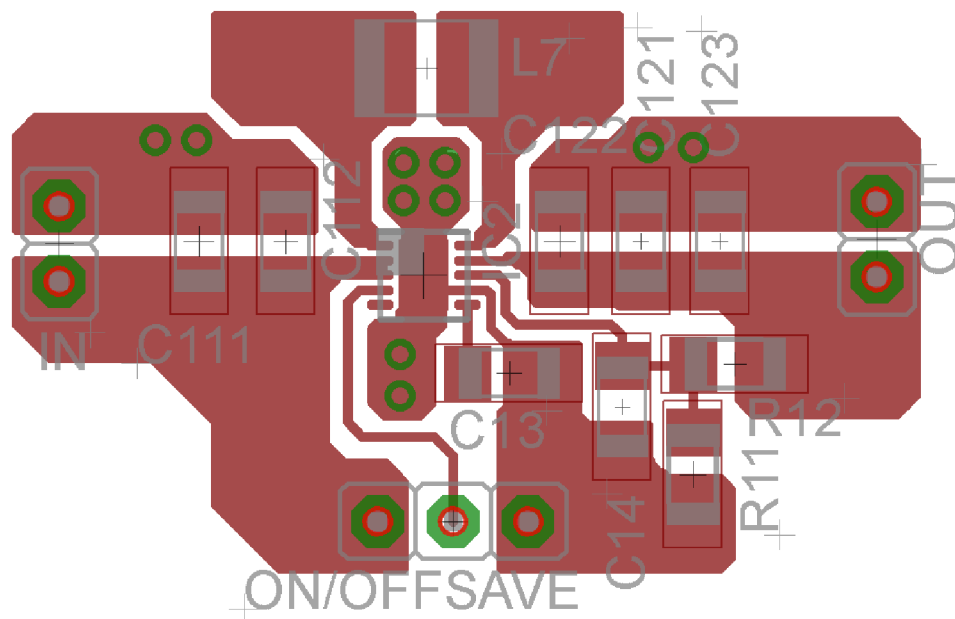
Obr. B.9: Schéma zapojenia DC/DC meniča TPS62160



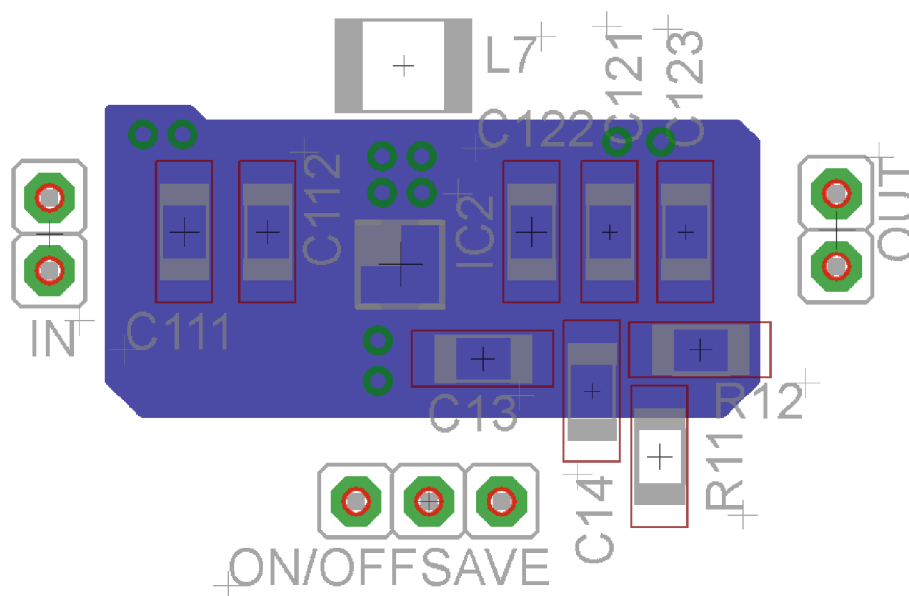
Obr. B.10: Návrh DPS meniča TPS62160 - vrchná strana



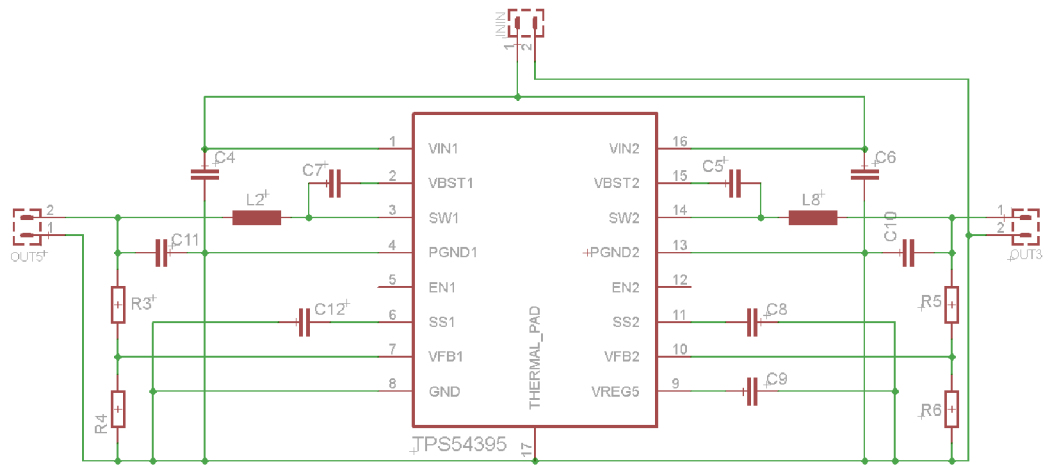
Obr. B.11: Schéma zapojenia DC/DC meniča TPS63060



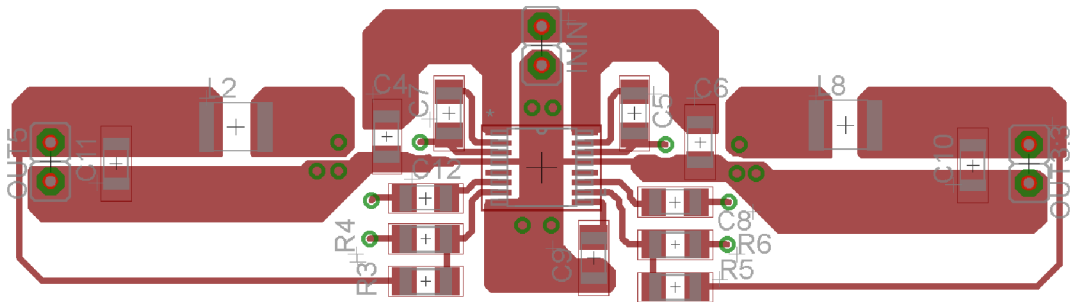
Obr. B.12: Návrh DPS meniča TPS63060 - vrchná strana



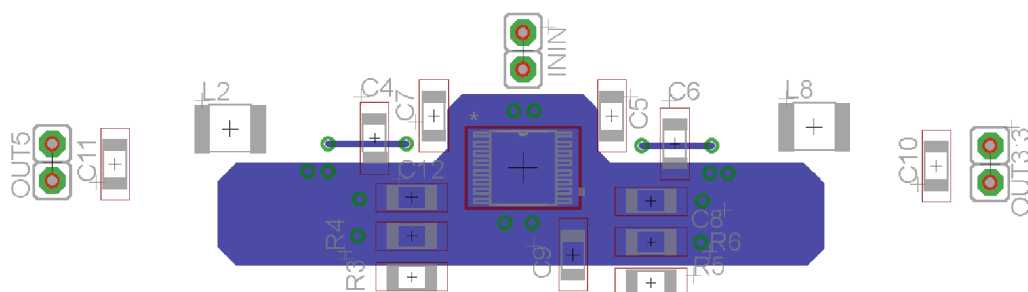
Obr. B.13: Návrh DPS meniča TPS63060 - spodná strana



Obr. B.14: Schéma zapojenia DC/DC meniča TPS54395

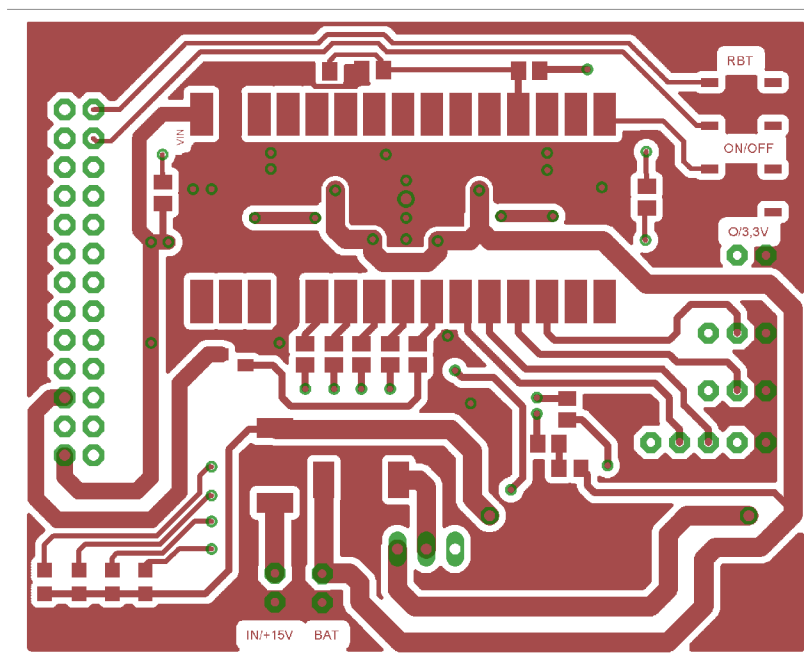


Obr. B.15: Návrh DPS meniča TPS54395 - vrchná strana

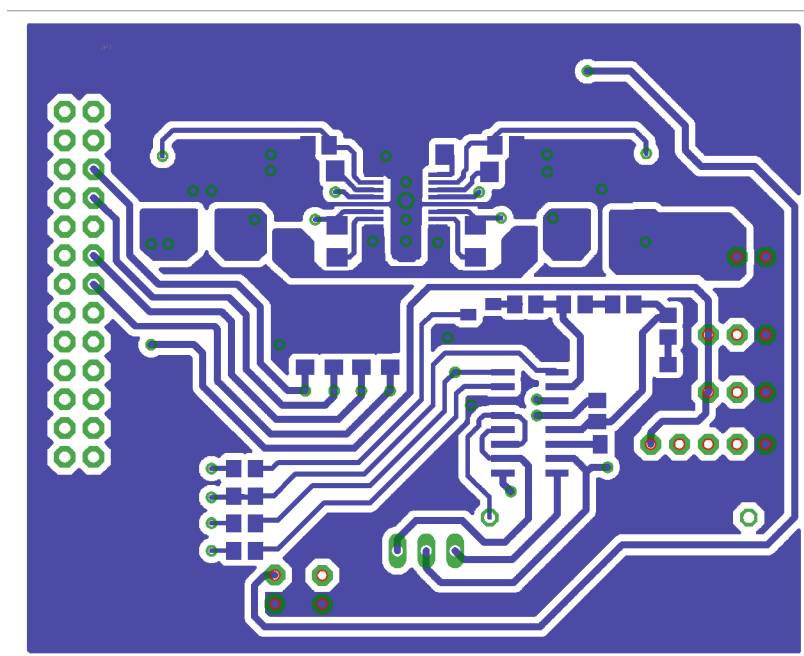


Obr. B.16: Návrh DPS meniča TPS54395 - spodná strana

C ROZŠIRUJÚCI MODUL



Obr. C.1: Návrh DPS rozširujúceho modulu - vrchná strana



Obr. C.2: Návrh DPS rozširujúceho modulu - spodná strana

D ZDROJOVÝ KÓD PROGRAMU

```
// definovanie pinov procesora
const int ledPin = 10;
const int pirPin = 5;
const int napajanie = A1;
const int TRIG_PIN = 4;
const int ECHO_PIN = 3;
const int button = 2;

// definovanie premenných
int vyp = 1;
int v = 1;
int u, p, nap;
signed long duration,distance, distance1, distance2,
    distance3, vysdistance;

// nastavenie použitých pinov
void setup()
{
    pinMode(button, INPUT);
    digitalWrite(button, HIGH);
    pinMode(TRIG_PIN,OUTPUT);
    pinMode(ECHO_PIN,INPUT);
    pinMode(ledPin, OUTPUT);
    digitalWrite(ledPin, HIGH);
    pinMode(pirPin, INPUT);
    Serial.begin(9600);
    attachInterrupt(0,prerusenie, FALLING); // prerušenie na pine D1

// meranie počiatočenej vzdialenosti ultrazvukom
    digitalWrite(TRIG_PIN, LOW);
    delayMicroseconds(2);
    digitalWrite(TRIG_PIN, HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIG_PIN, LOW);
    duration = pulseIn(ECHO_PIN,HIGH);
    distance1 = duration / 29.1 / 2 ;
}
```

```

//hlavný program
void loop()
{
  nap = analogRead(napajanie);

  if(vyp == 0){
    loop();
  }
  else{

    if(nap < 700)//kontrola napajania
    {
      Serial.println("kontrola nabijania");
      kontrolaSenz();
    }
    else {
      digitalWrite(ledPin, HIGH);
      v = 1;
    }
  }
}

void kontrolaSenz() //podprogram pre kontrolu senzorov
{
  Serial.println("kontrola senzorov");
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN, LOW);
  duration = pulseIn(ECHO_PIN,HIGH);

  distance2 = duration / 29.1 / 2 ;
  vysdistance = distance1 - distance2;

  if(((vysdistance > 6)|| (vysdistance < -6)) ||
(digitalRead(pirPin) == HIGH))

```

```

{
    zapnutie();
}
else {
    vypnutie();
}
}

void vypnutie() //podprogram pre vypnutie
{
    if(v == 1)
    {
        Serial.println("posle pulz k vypnutiu a pocka");
        delay(1000);
        Serial.println("vypne raspi");
        digitalWrite(ledPin, LOW);
        delay(1000);
        v = 0;
    }
}

void zapnutie() //podprogram pre zapnutie
{
    if(v == 0)
    {
        digitalWrite(ledPin, HIGH);
        Serial.println("zapne napajanie s baterky");
        delay(40000);
        v = 1;
        Serial.println("pocka na fotku");
        delay(1000);
        cakaj();
    }
}

void cakaj() //podprogram pre ďalší pohyb
{
    for(int i=0; i < 50; i++)

```

```

{
    delay(50);
    digitalWrite(TRIG_PIN, LOW);
    delayMicroseconds(2);
    digitalWrite(TRIG_PIN, HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIG_PIN, LOW);
    duration = pulseIn(ECHO_PIN,HIGH);

    distance3 = duration / 29.1 / 2 ;
    distance = distance1 - distance3;

    if(((distance > 6)|| (distance < -6))||
(digitalRead(pirPin) == HIGH))
{
    u++;
    Serial.println("zaznamenanie dalsieho pohybu");
}
}
if(u > p)
{
    p = u;
    cakaj();
}
}

void prerusenie() //podprogram prerušenia pre tlačidlo
{

    static unsigned long last_millis = 0;
    unsigned long m = millis();
    int state;
    int buttonTime;

    if (m - last_millis < 500)
{
}
else
{

```

```
state = !state;
buttonTime=millis();
last_millis = m;

    if (v == 0)
{
    digitalWrite(ledPin, HIGH);
    v = 1;
    vyp = 1;
}

    else
{
    digitalWrite(ledPin, LOW);
    v = 0;
    vyp = 0;
}
}
```


E SKRIPTY

E.1 Skript fotka

```
#!/bin/bash

for i in {2..14}; do
DATE=$(date +"%Y-%m-%d-%H-%M-%S")
raspistill -n -t 500 -o /home/fotky/$DATE.jpg

done
```

E.2 Skript poweroff.py

```
#!/usr/bin/python

import RPi.GPIO as GPIO
import os

GPIO.setmode(GPIO.BOARD)
GPIO.setup(24, GPIO.IN, pull_up_down=GPIO.PUD_UP)

while True:
    input_state = GPIO.input(24)
    if input_state == False:
        os.system('poweroff')
```

E.3 Skript reboot.py

```
import RPi.GPIO as GPIO
import os

GPIO.setmode(GPIO.BOARD)
GPIO.setup(26, GPIO.IN, pull_up_down=GPIO.PUD_UP)

while True:
```

```
input_state = GPIO.input(26)
if input_state == False:
    os.system('reboot')
```

E.4 Skript script-copy

```
#!/bin/bash
mv /home/fotky/* /camdata/
```

E.5 Skript mail

```
#!/bin/bash
DATE=$(date +"%Y-%m-%d-%H-%M-%S")
mail -s "POZOR!! Zaznamenana fotka! s casom:
$DATE" ptr.sidor@gmail.com < /dev/null
```

E.6 Skript script-find

```
#!/bin/bash
sleep 10
pocet=0
akt=0
conf=3
DATE=$(date +"%Y-%m-%d-%H-%M-%S")
cd /camdata
pocet=$(ls -l | wc -l)

while [ 1 ]; do

cd /camdata
akt=$(ls -l | wc -l)

if [ "$akt" -gt "$pocet" ]; then
mail -s "POZOR!! navysenie priecinka s casom:
$DATE" ptr.sidor@gmail.com < /dev/null
sleep 10
pocet=$akt$
```

```
else
sleep 1
fi
done
```

E.7 Skript ping

```
#!/bin/sh
while [ 1 ]; do
ping -c 1 google.com
    sleep 300
done
```

E.8 Skript S41mount

```
#!/bin/bash
mount -t cifs //192.168.156.1/img -o username=Sidi,
password=baran /camdata
```

F WEBOVÁ STRÁNKA

```
<!DOCTYPE html>
<html lang="cs-cz" dir="ltr">
  <head>
    <meta charset="UTF-8">

    <title>Obrázky</title>
    <style>

      a {margin: 10px;}

      a img {border: 2px solid #000; width: 100%; max-width: 300px;}

      a:hover img {border-color: #a0a0a0;}

    </style>
  </head>
  <body>

    <?php
    $dir    = 'img/';
    $files = glob($dir.'*.{jpg,gif,png,JPG,PNG,GIF}', GLOB_BRACE);

    usort($files, function($a, $b) {
      return filemtime($a) < filemtime($b);
    });

    foreach ($files as $soubory){
      echo '<a href="' . $soubory . '><img src="' . $soubory . '></a>';
    }
    ?>

  </body>
</html>
```