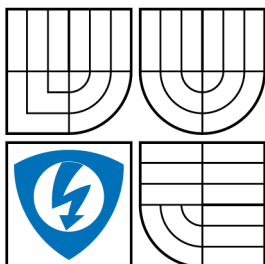


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY



FACULTY OF ELECTRICAL ENGINEERING AND
COMMUNICATION
DEPARTMENT OF CONTROL AND INSTRUMENTATION

DÁLKOVÉ OVLÁDÁNÍ PRO QUADROTOR THE REMOTE CONTROL OF THE QUADROTOR

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

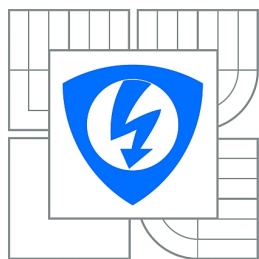
AUTOR PRÁCE
AUTHOR

JIŘÍ RŮŽIČKA

VEDOUcí PRÁCE
SUPERVISOR

ING. PETR GÁBRLÍK

BRNO 2014



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav automatizace a měřicí techniky

Bakalářská práce

bakalářský studijní obor
Automatizační a měřicí technika

Student: Jiří Růžička
Ročník: 3

ID: 146943
Akademický rok: 2013/2014

NÁZEV TÉMATU:

Dálkové ovládání pro quadrotor

POKYNY PRO VYPRACOVÁNÍ:

Cílem práce je vytvořit dálkové ovládání pro quadrotor z předloženého komerčního herního ovladače. Ovládání bude umožňovat dálkové řízení quadrotoru pomocí daného komunikačního protokolu a modulů ZigBee a také vizualizovat provozní data na displeji.

Zadání lze shrnout do následujících bodů:

1. Seznamte se s možnostmi dálkového řízení robotů a fungováním předloženého herního ovladače.
2. Navrhněte schéma nové elektronické části herního ovladače, která navíc umožní připojení ZigBee modulu a displeje.
3. Dle schématu vytvořte DPS, osadte jej a ověřte funkčnost zapojení.
4. Vytvořte program pro mikrokontrolér, který bude číst stavy ovládacích prvků herního ovladače.
5. Dále implementujte požadovaný komunikační protokol, zprovozněte komunikaci pomocí ZigBee modulů a implementujete vizualizaci provozních dat quadrotoru na displeji.
6. Funkčnost dálkového ovládání otestujte dle pokynů vedoucího.

DOPORUČENÁ LITERATURA:

[1] HOLLAND, John. Designing Autonomous Mobile Robots. [s.l.] : Newnes, c2004. 335 s. ISBN 0-7506-7683-3.

Termín zadání: 10.2.2014

Termín odevzdání: 26.5.2014

Vedoucí práce: Ing. Petr Gábrlík
Konzultanti bakalářské práce:

doc. Ing. Václav Jirsík, CSc.
Předseda oborové rady

ABSTRAKT

Tato bakalářská práce se zabývá návrhem dálkového ovládání pro Quadrotor. Cílem projektu je návrh a praktická realizace základní desky, jako náhradu vysloužilého gamepadu, která bude umožňovat bezdrátové ovládání Quadrotoru a zobrazování telemetrie na display.

KLÍČOVÁ SLOVA

Dálkové ovládání, Quadrotor, Návrh DPS, ZigBee, Xmega256A3U

ABSTRACT

This bachelor's thesis deals with the design of the remote control for Quadrotor. The goal of project is design and practical implementation of the motherboard as a replacement of retired gamepad, which will allow wireless control of Quadrotor and show telemetry on display.

KEYWORDS

Remote control, Quadrotor, Design of PCB, ZigBee, Xmega256A3U

RŮŽIČKA J. *Dálkové ovládání pro quadrotror*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2014. 37 s. Vedoucí práce Ing. Petr Gábrlík.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Dálkové ovládání pro quadrotor“ jsem vypracoval samostatně pod vedením vedoucího semestrálního projektu a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením tohoto semestrálního projektu jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení §11 a následujících autorského zákona č. 121/2000Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení §152 trestního zákona č. 140/1961Sb.

V Brně dne

.....

(podpis autora)

PODĚKOVÁNÍ

Děkuji vedoucímu bakalářské práce Ing. Petru Gábrlíkovi za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce.

V Brně dne

.....

(podpis autora)

OBSAH

Úvod	10
1 Výběr komponent	12
1.1 Základní deska	12
1.1.1 MCU	12
1.1.2 Ovládací prvky	13
1.1.3 Displej	13
1.1.4 ZigBee	13
1.2 Napájení	14
1.2.1 TPS76733QD	14
1.2.2 LM2621	15
2 Návrh DPS	18
2.1 Rozložení komponent	18
2.2 Zapojení MCU	18
2.2.1 Port A	19
2.2.2 Port B	19
2.2.3 Port C	19
2.2.4 Port D	20
2.2.5 Port E	20
2.3 DPS verze 1.1	21
2.3.1 Chyby	21
2.3.2 Opravy	22
3 Software	23
3.1 Požadavky	23
3.2 Čtení tlačítek	23
3.2.1 DPS v1.1	24
3.3 Displej	24
3.4 Komunikace	25
3.4.1 UranusLink	26
3.5 A/D převodníky	28
4 Závěr	29
Reference	30
Seznam zkratek	31

Seznam příloh	32
A DPS 1.1	33
A.1 Horní strana desky	33
A.2 Dolní strana desky	33
B Osazovací plány	34
B.1 Horní strana desky	34
B.2 Dolní strana desky	34
C Příloha na CD a schéma zapojení	35
D Fotodokumentace 1	36
E Fotodokumentace 2	37

SEZNAM OBRÁZKŮ

1	Quadrotor Uranus [1]	10
2	Blokové schéma stávající situace	11
3	Blokové schéma nového řešení	11
1.1	Blokové schéma komponent základní desky	12
1.2	Výrobce doporučené zapojení	15
1.3	Zapojení zdroje LM2621	16
1.4	Závislost spínací frekvence na R_{FQ}	17
1.5	Závislost špičkového proudu cívky na proudu do zátěže	17
2.1	Nově vytvořená součástka pro Eagle	19
2.2	Programovací konektor	19
2.3	Schéma zapojení tlačítek	20
2.4	Opravy v zapojení zdroje LM2621	22
3.1	Zobrazení telemetrie	25
3.2	Struktura protokolu	26

SEZNAM TABULEK

1.1	Parametry MCU	13
1.2	Parametry displeju	14
1.3	Parametry XBee-Pro	14
1.4	Parametry TPS76733QD	15
1.5	Parametry diody SS16	17

ÚVOD

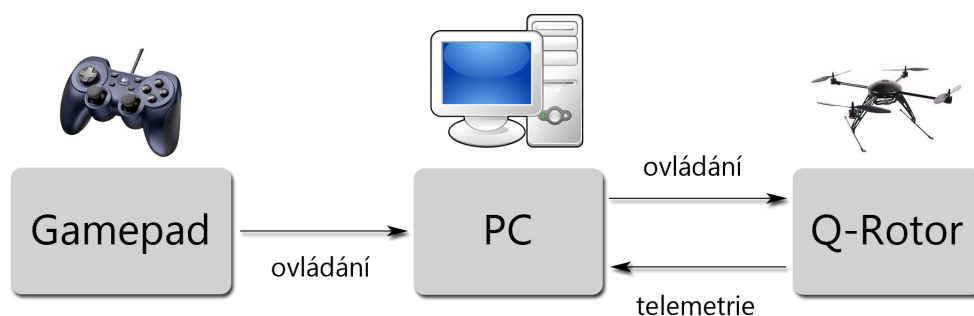
Quadrotor je čtyř vrtulový létající stroj, v některé literatuře nazýván také jako Quadrocopter. V tomto případě je to robot Uranus 1 vyvíjený na Ústavu automatizace a měřicí techniky výzkumnou skupinou Robotiky a umělé inteligence. Konstrukce vychází z jednoduchého řešení znázorněného na obrázku 1, ve střední části je blok s elektronikou a na dvou hliníkových příčkách jsou umístěny motory.



Obrázek 1: Quadrotor Uranus [1]

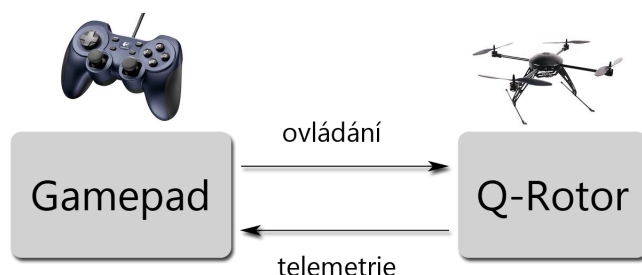
Komunikaci zajišťují dva rádiové moduly ZigBee, jeden na palubě robota a druhý na pozemní stanici. Nyní je robot ovládán na dálku ovladačem v podobě klasického gamepadu připojeného do počítače. Operátor tak posílá příkazy počítači, který je zpracuje a odešle robotu. Celý proces, popsáný blokovým schématem na obrázku 2, vyžaduje počítač se softwarem zpracovávajícím povely z gamepadu a přípravek s komunikačním modulem.

Toto řešení je výhodné pro použití v laboratorních podmínkách, ovladač nepotřebuje baterie a na počítači je přehledně zobrazena telemetrie z Quadrotoru. Robot je ve fázi prototypu, který je stále zdokonalován a je na něm testován nový software i hardware. Pro toto testování, např. GPS modulu, však laboratoř nepostačuje a je třeba provádět letové testy v exteriéru. Zde je použití počítače překážkou, proto je



Obrázek 2: Blokové schéma stávající situace

potřeba zaimplementovat vlastnosti počítače přímo do samotného gamepadu. Tedy ovládání Quadrotoru a zobrazení telemetrie, bez počítače jako zprostředkovatele. Řešení je znázorněno na obrázku 3.



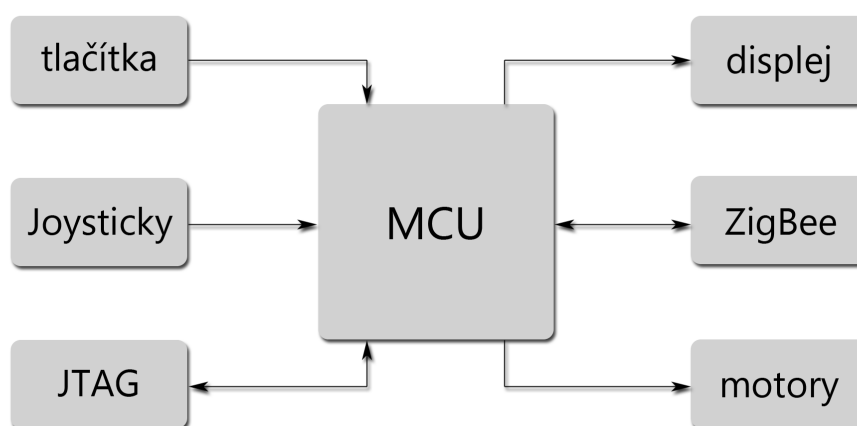
Obrázek 3: Blokové schéma nového řešení

Ovladač by měl poskytovat dostatek ovládacích prvků a vhodný displej. Klasický gamepad poskytuje 16 tlačítek a dva joysticky. Velikost zabudovaného displeje bude mít vliv na čitelnost a množství zobrazovaných informací. Vzhledem k malým rozměrům gamepadu je nutné vynechat několik tlačítek ve střední části ovladače. Konkrétní výběr komponent, hardwarové a softwarové řešení je popsáno v následujících kapitolách.

1 VÝBĚR KOMPONENT

1.1 Základní deska

K realizaci dálkového ovládání je k dispozici vyřazený gamepad, ze kterého bude využito šasi, čtyři přední tlačítka, hmatníky zbylých tlačítek a dva vibrační motory. Původní základní deska je využita k zaměření montážních otvorů a přesných rozměrů pro výrobu nové. Celá elektronika se skládá z několika základních částí: MCU, ovládací prvky, displej, vibrační motory a ZigBee modul (viz. Obr. 1.1). Programování zajišťuje JTAG. Napájecí obvody bude řešit další kapitola.



Obrázek 1.1: Blokové schéma komponent základní desky

1.1.1 MCU

MCU zpracovává povely z ovládacích prvků (tlačítka a joysticky), komunikuje prostřednictvím ZigBee s Quadrotorem a zobrazuje přijímanou telemetrii na displeji. Procesor tedy musí obsahovat komponenty pro zpracování analogových signálů z joysticků, disponovat sběrnici pro komunikaci se ZigBee a displejem. Vzhledem k použití grafického displeje, je potřeba zohlednit paměťovou náročnost grafických dat (např. znaková sada).

V tomto případě byl zvolen Atmel ATxmega256A3U. Poskytuje dostatek paměti pro grafická data, ADC s dostačujícím rozlišením a nízké napájecí napětí od 1,6 V umožňuje použití akumulátorů. Parametry (Tab. 1.1) čerpány z katalogového listu MCU [3].

Tabulka 1.1: Parametry MCU

Napájecí napětí	1,6 - 3,6 V
Pracovní kmitočet	32 MHz
Paměť flash	256 KB + 8 KB
Paměť EEPROM	4 KB
SRAM	16 KB
AD převodníky	12 bit, 2 MSps, 2x 8 kanálů
Programování	JTAG
Programovatelných I/O pinů	50

1.1.2 Ovládací prvky

Základními prvky jsou dva joysticky, každý obsahuje dva potenciometry a jedno tlačítko. Dále čtyři původní tlačítka v přední části gamepadu a osm nových mikrospínačových tlačítek. Od původního řešení chybí čtyři tlačítka ve střední části, kde je nyní displej.

1.1.3 Displej

Vzhledem k velikosti gamepadu je třeba zvolit displej s co nejmenšími rozměry. Alfnumerický displej malých rozměrů, často pouze dvouřádkový, není vhodný kvůli omezeným možnostem zobrazení. Připadá tak v úvahu jedině grafický displej.

Nejvhodnějšími jsou LCD Batron COG-BTG12864-01 a OLED Densitron DD-12864YO-3A. LCD je citlivější na provozní teplotu, maximální hodnota napájecího napětí při pokojové teplotě je 10,9 V, při teplotě -20°C je typická hodnota tohoto napětí 11,3 V. OLED je v tomto ohledu odolnější, napájecí napětí je sice vyšší (12-13 V), ale nezávislé na okolní teplotě. Dalším kritériem výběru je cena, LCD je dvakrát dražší než OLED. Srovnání parametrů z katalogových listů (OLED [4] a LCD [5]) je v tabulce 1.2. Komunikace s MCU lze programově volit po SPI nebo I²C.

1.1.4 ZigBee

Pro komunikaci s Quadrotrem v pásmu 2,4 GHz (podle standardu IEEE 802.15.4) je použit ZigBee modul Digi XBee-PRO s U.FL konektorem pro anténu. Dosah v otevřeném prostoru je omezen maximálním výkonem, který dovoluje ETSI, 10 mW při použití na území Evropy a to na 750 m. Při maximálním výkonu 63 mW je dosah až 1600 m. Informace v Tab. 1.3 jsou převzaty z katalogového listu výrobce [6].

Tabulka 1.2: Parametry displejů

Parametr	OLED	LCD
Rozlišení	128 x 64 px	128 x 64 px
Celkové rozměry	45 x 29 mm	45 x 29 mm
Rozměry zobrazovací plochy	35 x 17,5 mm	37 x 20 mm
Driver	SSD1305	NT7534H-BD
Napájecí napětí log. obvodů	2,4 - 3,5 V	3,14 - 3,47 V
Napájecí napětí displeje	12 - 13 V	10,5 - 10,9 V (25°C)
Maximální odběr	38 mA	40,5 mA

Tabulka 1.3: Parametry XBee-Pro

Dosah v exteriéru	750 m (ETSI)
Vysílací výkon	10 mW (ETSI)
Datový tok	250 kb/s
Citlivost přijímače	-100 dBm
Napájecí napětí	2,8 - 3,4 V
Vysílací proud	150 mA (ETSI)
Přijímací proud	55 mA

1.2 Napájení

Předpokládá se použití jedno-článekového Li-Pol akumulátoru s napětím 3,6 V. Většině obvodů na základní desce postačí napájení 3,3 V, výjimkou je 12 V pro displej.

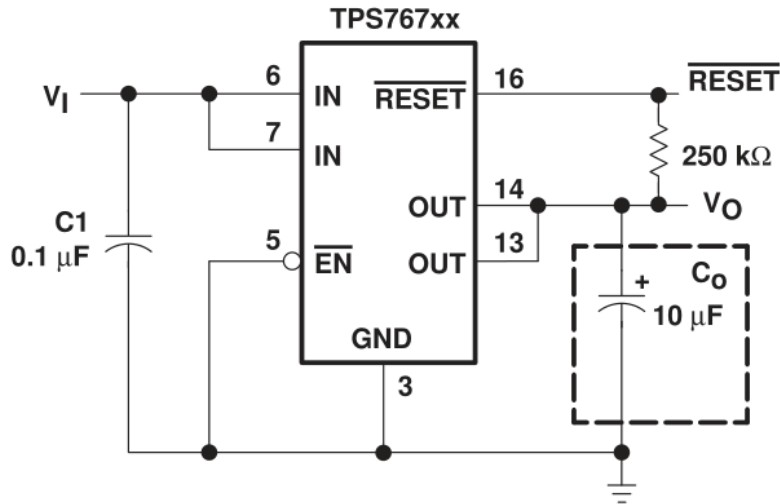
Pro napájecí větev 3,3 V byl zvolen integrovaný DC-DC měnič TI TPS76733QD a pro větev 12V TI LM2621.

1.2.1 TPS76733QD

Jedná se o jednoduchý LDO regulátor s pevným výstupním napětím 3,3 V a maximálním výstupním proudem 1 A, který postačí pro napájení všech obvodů ve větvi 3,3 V na základní desce. Rozsah vstupních napětí 2,7 - 10 V vyhovuje pro napájení z akumulátoru (viz. Tab. 1.4). Filtrační tantalové kondenzátory jsou zvoleny podle doporučení (viz. Obr. 1.2) v katalogovém listu výrobce [7].

Tabulka 1.4: Parametry TPS76733QD

Rozsah vstupních napětí	2,7 - 10 V
Výstupní napětí	3,3 V
Výstupní proud	až 1 A



Obrázek 1.2: Výrobce doporučené zapojení

1.2.2 LM2621

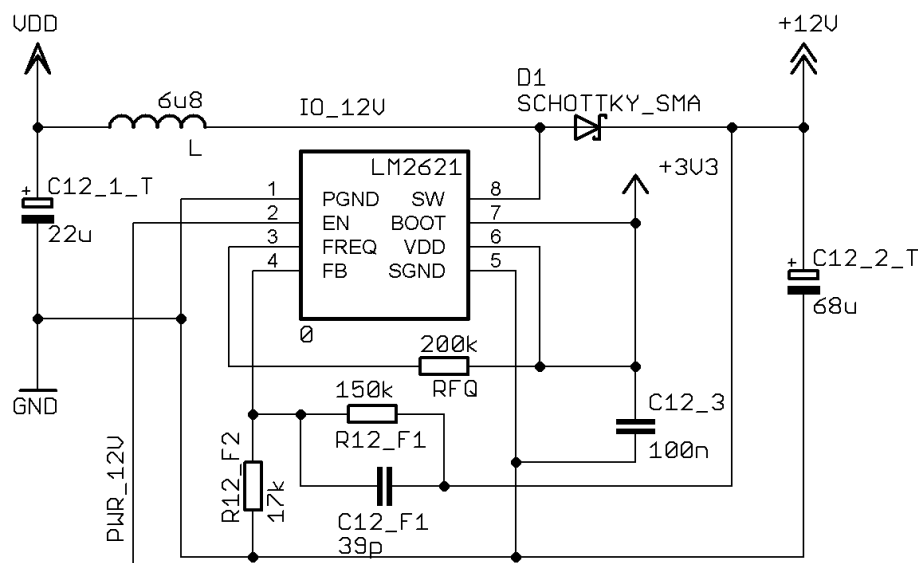
Pro 12 V větev je třeba napětí akumulátoru zvýšit více než na trojnásobek, k tomu je použit Step-Up konvertor od Texas Instruments LM2621 [8]. Jedná se o regulovatelný DC-DC měnič s velkým rozsahem vstupních i výstupních napětí.

Pro správnou funkci vyžaduje integrovaný obvod další podpůrné součástky (viz. Obr. 1.3), cívku pro střádání energie, filtrační kondenzátory a rezistory pro nastavení požadovaných parametrů výstupního napětí.

Nastavení výstupního napětí je provedeno podle rovnice 1.1, hodnota rezistoru $R_{F1} = 150 \text{ k}\Omega$ je výrobcem doporučena. Po dosazení doporučeného R_{F1} a požadovaného výstupního napětí 12 V, je získána hodnota rezistoru $R_{F2} = 17,3 \text{ k}\Omega$ (nejbližší nižší hodnota z řady E24 je 16k).

$$R_{F2} = R_{F1} / [(V_{OUT} / 1,24) - 1] \quad (1.1)$$

Kondenzátor C_{F1} je zvolen podle doporučení 39 pF, pomáhá ve zpětné vazbě redukovat zvlnění výstupního napětí. Rezistorem R_{FQ} je nastavena frekvence spínání. Výrobce udává použitelné rozmezí frekvencí od 300 kHz do 2 MHz, podle grafu (1.4)



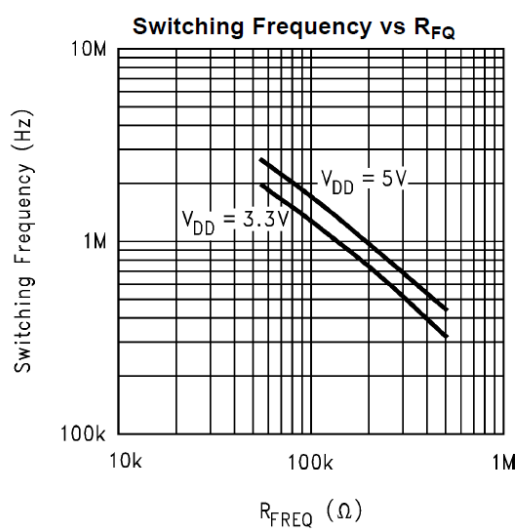
Obrázek 1.3: Zapojení zdroje LM2621

volíme hodnou rezistoru 200 k Ω odpovídající spínací frekvence je pak 700 kHz při napájecím napětí 3,3 V a 1 MHz při 5 V.

Stíněná cívka 6,8 μ H od firmy Bourns (viz. katalogový list [9]) je zvolena s ohledem na požadované vlastnosti. Saturační proud cívky musí být vyšší než špičkový proud tekoucí obvodem. Špičkový proud je odvozen z grafu na obrázku 1.5, protože zdroj napájí jen displej, zatěžovací proud bude menší než 100 mA, špičkový proud by tedy neměl překročit hodnotu 1,5 A. Odpor vinutí musí být menší než 100 m Ω , aby byla zaručena co nejvyšší účinnost zdroje. Saturační proud zvolené cívky je 4,3 A a odpor vinutí je 76,2 m Ω , bude tedy postačující.

Výstupní dioda musí splňovat tyto požadavky: maximální propustný proud větší než proud do zátěže a maximální závěrné napětí větší než výstupní napětí. Kvůli vysoké spínací frekvenci je nutné použití rychlé Schottkyho diody. Zvolená dioda Vishay SS16 splňuje všechny tyto podmínky (viz. tabulka 1.5) [10].

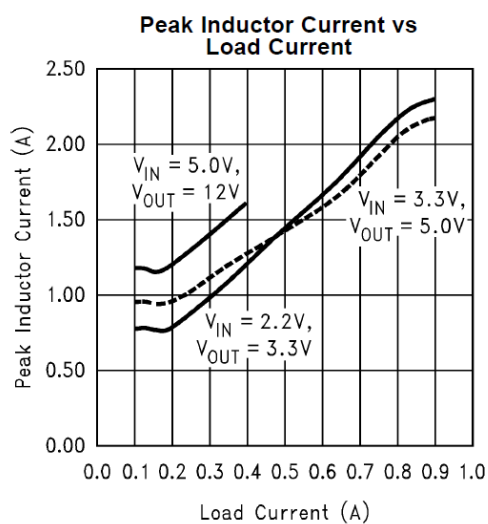
Dále byly zvoleny tantalové kondenzátory vstupní $C_1 = 22 \mu$ F a výstupní $C_2 = 68 \mu$ F, rezistor $R_1 = 510 \Omega$ a kondenzátor $C_3 = 100$ nF podle doporučení výrobce. Vývod EN pro sepnutí/vypnutí zdroje je přiveden na MCU vodičem PWR_12V, napájení displeje tak bude řízeno programově.



Obrázek 1.4: Závislost spínací frekvence na R_{FQ}

Tabulka 1.5: Parametry diody SS16

Maximální špičkové závěrné napětí	60 V
Maximální efektivní závěrné napětí	42 V
Maximální dovolený propustný proud	1 A



Obrázek 1.5: Závislost špičkového proudu cívky na proudu do zátěže

2 NÁVRH DPS

K návrhu základní desky byl použit program Eagle 6.5 s freeware licencí. Tato licence dovoluje umístění součástek pouze na ploše 100x80 mm, požadovaný rozměr desky je 130x60 mm, proto byly některé součástky umístěny až těsně před výrobou na školním počítači s plnou verzí programu.

2.1 Rozložení komponent

Ovládací prvky na desce mají pozici danou původním řešením, tlačítka a joysticky proto musejí být přesně zaměřeny, aby byla možná montáž do šasi gamepadu. Displej je umístěn ve střední části, tomu musí odpovídat umístění jeho konektoru. Uprostřed na horní straně desky je také umístěn konektor pro ZigBee modul.

Kvůli nedostatku místa na horní straně desky, kde nejvíc místa zabírá ZigBee, je MCU na straně spodní spolu s napájecími zdroji. Napájecí zdroj 12 V je umístěn co nejbližší ke konektoru displeje a co nejdále od analogové části joysticků. Filtrační kondenzátory jsou v těsné blízkosti napájecích vývodů integrovaných obvodů.

Programovací konektor je na kraji desky v přední části pro pohodlné připojení i ve složené podobě gamepadu. Je zvolen konektor běžně používaný pro Micro USB 3.0, především kvůli svým kompaktním rozměrům.

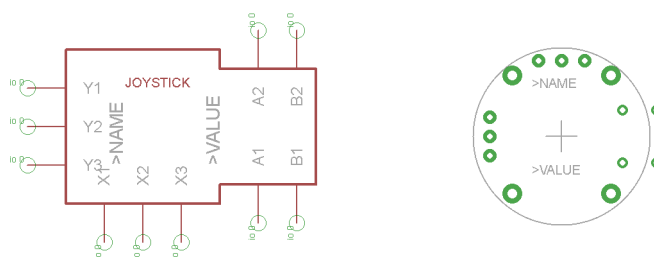
2.2 Zapojení MCU

Základem návrhu je sestavení schématu zapojení, tak aby bylo propojení MCU se všemi periferiemi co nejjednodušší.

- Port A je využit pro analogové signály, je využito pět kanálů ADC, čtyři pro joysticky a jeden pro měření napájecího napětí.
- Port B slouží k programování a ovládání napájení displeje.
- Port C je vyhrazen tlačítkům zapojeným v matici, horní polovina portu jako výstup nastavující aktivní sloupec a dolní polovina je vstup pro čtení řádků.
- Port D pro komunikaci a spínání vibračních motorů.
- Port E je vyhrazen pro výstup na displej.
- Port F je neobsazen.

2.2.1 Port A

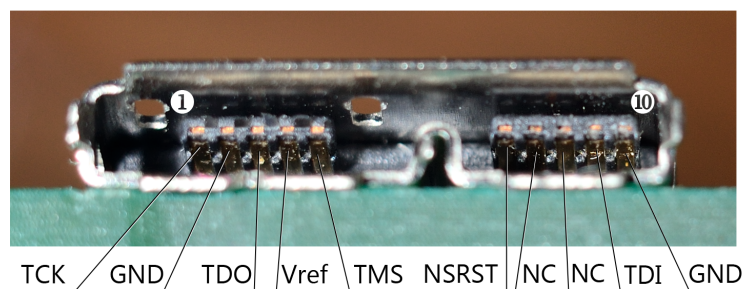
Joysticky tohoto typu žádná standardní knihovna pro Eagle neobsahuje, byla proto potřeba vytvoření vlastní, podle měření původní součástky (viz. Obr. 2.1). Schéma obsahuje dva trojité vývody potenciometrů, vývody pro tlačítko a čtyři montážní otvory.



Obrázek 2.1: Nově vytvořená součástka pro Eagle

2.2.2 Port B

Zde jsou programovací vývody JTAG: TCK, TDO, TDI, TMS. Spolu s referenčním napětím Vref, zemí GND a resetovacím NSRST vývodem MCU jsou vyvedeny na programovací konektor. Dva piny jsou nevyužity a označeny jako NC (viz. Obr. 2.2).



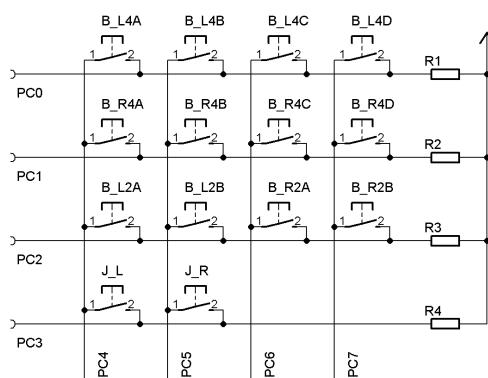
Obrázek 2.2: Programovací konektor

Tento port také obsluhuje spínací vývod napájecího zdroje pro 12 V větev. Pro bezpečný provoz displeje je nutné dodržet postup zapínání a vypínání podle doporučení výrobce. Logické obvody musejí být napájeny dříve než samotný displej.

2.2.3 Port C

Na tomto portu jsou připojena tlačítka zapojená v matici 4x4, jak je znázorněno na obrázku 2.3. Nastavením logické hodnoty nula na vývodech PC4 - PC7 se akti-

vuje sloupec, sepnutím tlačítka v tomto sloupci se objeví logická hodnota nula na příslušném vývodu PC0 - PC3.



Obrázek 2.3: Schéma zapojení tlačítek

2.2.4 Port D

Na vývody sériové sběrnice TX a RX je přiveden ZigBee modul. Dále na pinech PD4 a PD5 jsou připojeny tranzistory pro spínání vibračních motorů, které budou sloužit pro varování obsluhy ovladače před nebezpečnými stavy samotného přístroje nebo Quadrotoru.

2.2.5 Port E

Vývody displeje BS1, CS#, RES#, D/C#, SCLK, SDIN, jsou připojeny k tomuto portu.

- BS1 pro nastavení komunikace přes SPI nebo I²C, na výstupu PE5.
- CS# pro aktivaci komunikace displeje na výstupu PE4.
- RES# vývod pro resetování displeje na PE3.
- D/C# na výstupu PE2 určuje, jaký typ dat se bude přijímat, jestli příkazy driveru nebo data pro zobrazení.
- SCLK je pro hodinový signál z PE1.
- SDIN pro vstup sériových dat z výstupu PE0.

2.3 DPS verze 1.1

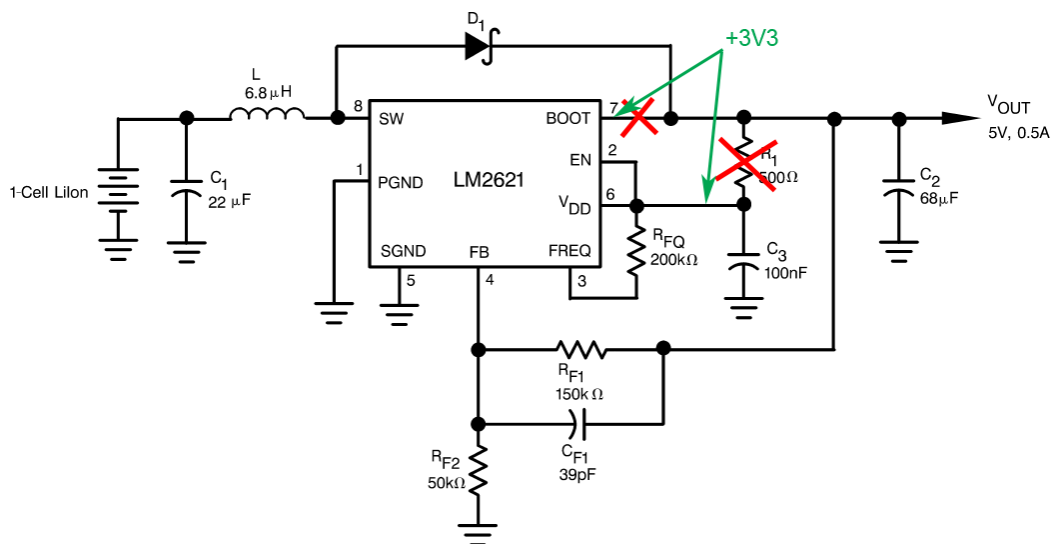
Stávající řešení se ukázalo jako nevyhovující. Při návrhu nastalo několik chyb. Proto byla navržena nová DPS, která tyto problémy řeší. Vstupy a výstupy procesoru jsou přizpůsobeny rozmístění součástek a celý kontrolér je otočen tak, aby byla analogová část co nejblíže joystickům.

2.3.1 Chyby

- Konektor pro displej je zrcadlově otočen, aby mohl být displej umístěn v požadované pozici, musí být připojen přes prodlužovací kabel.
- Napájecí zdroj na 12 V byl zapojen podle doporučení výrobce, které nepočítalo s vysokým výstupním napětím a napájecí zdroj tak nefungoval. Momentálně opraveno drátovou propojkou.
- Joysticky byly připojeny přímo na napájecí napětí 3,3 V, AD převodníky procesoru ovšem dokáží měřit pouze do 2 V. Převodníky tak nebyly schopné měřit polohu joysticků v celém rozsahu.
- Zapojení tlačítek v matici je složité jak z hlediska místa na DPS, tak z hlediska detekce stlačení.
- Programovací konektor se neosvědčil, protikus na kabelu je problematický, špatně chráněný proti namáhání připájených kontaktů, které se při neopatrném zacházení rozpadají.
- ZigBee modul byl s procesorem propojen pouze komunikačními vývody RX a TX. Není tak možné využít funkcí modulu, jako detekce zaplněného bufferu.
- Neoddělená analogová a digitální zem.

2.3.2 Opravy

- Displejový konektor je nyní umístěn na spodní straně se správnou polaritou, nebude tak potřeba prodlužovacího kabelu.
- Zapojení napájecího zdroje 12 V bylo opraveno (viz Obr. 2.4).
- Napětí na potenciometrech joysticků bylo sníženo rezistory a lze je měřit AD převodníky v celém rozsahu.
- Tlačítka byla zapojena zvlášť, tak aby bylo fyzické připojení na desce co nej-jednodušší. Programová obsluha bude tak jednodušší. Pro připojení 14 tlačítek je na procesoru dostatek pinů.
- Programovací konektor umístěn na spodní straně v klasickém provedení a za-pojení podle standardu JTAG.
- Modul je nyní připojen k procesoru všemi vývody a lze tak využít všech jeho funkcí.
- Analogová a digitální zem rozdělena a spojena v jednom bodě nulovým re-zistorem.



Obrázek 2.4: Opravy v zapojení zdroje LM2621

3 SOFTWARE

Řídicí program je psán v jazyce C, ve vývojovém prostředí Atmel Studio 6.2.

3.1 Požadavky

Základním požadavkem je zpracování dat z AD převodníků a odesílání povelů robotu. Komunikace probíhá přes rádiový modul ZigBee prostřednictvím protokolu UranusLink, kterému je zpracování dat v ovladači přizpůsobeno. Procesor bude zpracovávat velké množství dat, je tak potřeba zajistit co největší výkon. Funkce `setClockTo32MHz()` nastaví vnitřní oscilátor z výchozího 2 MHz na 32 MHz.

Zpracování a zobrazování dat probíhá v rámci přerušení od čítačů/časovačů. Je proto třeba nastavit správné parametry dělení kmitočtu a perioda v registrech `TCx0_CTRLA` a `TCx0_CTRLB` (hodnota `x` značí port čítače). Priorita přerušení se nastává v registru `TCx0_INTCTRLA`, všechny přerušení od čítačů/časovačů jsou nastaveny na nízkou úroveň priority, nejvyšší prioritu má přerušení od komunikačního portu. Pro periody 30 ms a 100 ms vychází při 64 násobném dělení frekvence hodnota `TCx0_PER` na 15000 a 50000 podle vzorce 3.1.

$$TCx0_PER = \frac{F_CPU \cdot T}{64} \quad (3.1)$$

Kde:

F_CPU : je pracovní frekvence procesoru.

T : je požadovaná perioda přerušení.

3.2 Čtení tlačítek

Pro vyčtení stlačeného tlačítka v matici slouží funkce `ReadButtons()`. Při zavolání mění pozici aktivního sloupce a čte stavy v řádcích. Pokud detekuje stlačení, uloží do globální proměnné `button` hodnotu tlačítka (1-14) a vrátí hodnotu 0. Není-li detekováno žádné stlačení, proměnná s hodnotou tlačítka se nemění a funkce vrací hodnotu 1.

V hlavním programu je pak každému tlačítku přiřazena nějaká funkce. Při stlačení tlačítka, kdy čtecí funkce vrací hodnotu 0, se nevykonává nic. Jakmile je vrácena hodnota 1 (tlačítko již není stlačeno a jeho hodnota je uložena), `switch(button)` vybere funkci odpovídající stlačenému tlačítku a proměnnou `button` vynuluje. Je tak zajištěna reakce na sestupnou hranu a nehrozí vícenásobná detekce jednoho stlačení.

3.2.1 DPS v1.1

V nové verzi desky, kdy už nejsou tlačítka zapojena do matice, bude detekce probíhat jinak. Budou nastavena přerušení od jednotlivých tlačítek a v obslužných rutinách se bude řešit jejich funkcionality. Přerušení lze přímo nastavit na sestupnou hranu, odpadne tak složité programové řešení.

3.3 Displej

Pro obsluhu displeje byla použita hotová knihovna, která byla upravena pro použití v této aplikaci. Původní knihovna byla připravena pro použití s jiným procesorem a neumožňovala zobrazení číselných typů.

Na začátku programu je displej inicializován funkcí `OLED_init()`, která zajistí výrobcem požadovaný postup spuštění. Driver displeje je napájen okamžitě po spuštění ze zdroje 3,3 V, zobrazovací panel je připojen k napájení 12 V s určitým zpožděním. Jakmile je displej připraven ke komunikaci, je odeslána sekvence příkazů pro nastavení požadovaných funkcí, jako například orientace displeje.

Následují některé z funkcí, které byly přidány do původní knihovny.

```
void Show_uint16(uint16_t A, uint8_t B, uint8_t C);  
void Show_int16_point(int16_t A, uint8_t B, uint8_t C);  
void Show_uint8_point(uint8_t A, uint8_t B, uint8_t C);
```

- Funkce `Show_uint16()` slouží k vypsání 16 bitové neznaménkové proměnné A do řádku B a sloupce C, nevrací žádnou hodnotu. Dále knihovna obsahuje funkce pro znaménkové typy, 8 bitové a 16 bitové.
- Funkce `Show_int16_point()` sloužící pro zobrazení desetinných čísel zobrazuje 16 bitové znaménkové číslo A s jedním desetinným místem, opět na řádek B a sloupec C. Rozsah zobrazovaných čísel je od -3276.7 do 3276.8. Na obrázku (3.1) jsou to hodnoty náklonů R:, P: a Y: ve stupních, nulová hodnota je zobrazena jako .0.
- Funkce `Show_uint8_point()` je obdobou předchozí, je však pro neznaménkové 8 bitové čísla v rozsahu od .0 do 25.5.

Lze volit mezi zobrazením různých dat, přepínáním několika obrazovek. Globální proměnná `displej` určuje, která obrazovka se zrovna bude zobrazovat a pokud se stav nezměnil, obnovují se pouze dynamické data. Dojde-li ke změně obrazovky, stávající se smaže a vypíše se nové data odpovídající této obrazovce.

Na hlavní obrazovce je telemetrie (Obr. 3.1). Mód robota je na prvním řádku a může být ve čtyřech stavech, konfigurační indikuje `config` (0), letový dálkově ovládaný `flightRC` (1), letový neovládaný `flightPH` (2) a pokud není potvrzena změna stavu `waiting!` (?). Na dalším řádku je stav inicializace, GPS a XKF, pokud jsou aktivní, je zobrazeno OK pokud ne tak --. Dále informace o náklonech a natočení ve stupních. Doba provozu od spuštění v minutách a sekundách, stav baterie, odebíraný proud a průměrný výkon motorů. Jako poslední informace je počet chyb v komunikaci, při špatném signálu je tak obsluha informována a může učinit opatření.



Obrázek 3.1: Zobrazení telemetrie

Obnovování dynamických dat na displeji (mód, náklony, doba provozu, atd.) je vyvoláváno přerušením od čítače/časovače přibližně 30 krát za sekundu.

3.4 Komunikace

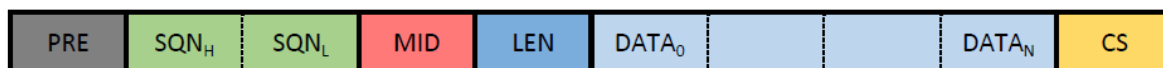
Pro komunikaci byla použita knihovna od výrobce procesoru Atmel. Umožňuje komunikaci na USART sběrnici prostřednictvím přerušení od komunikačního portu. Přenosová rychlost je nastavena na 57 600 b/s. Rychlost komunikace se nastavuje v registrech `BAUDCTRLA` a `BAUDCTRLB`. K výpočtu správných hodnot jsou použity makra `BSEL(f_cpu, baud)` a `BSCALE(f_cpu, baud)` z knihovny `xmega_baud.h`, vstupními parametry jsou pracovní kmitočet procesoru a požadovaná rychlost komunikace. Dále se inicializuje řadič přerušení `USART_InterruptDriver_Initialize()` a nastaví se formát dat (osm bitů dat, bez paritního bitu a jeden stop bit). Přerušení od portu je nastaveno na vysokou prioritu a má tak přednost před ostatními přerušeními od čítačů/časovačů. Jako poslední proběhne povolení odesílání, přijímání a přerušování.

Přijímané data jsou ukládána do softwarového bufferu `usart_data->buffer`, ze kterého jsou následně vyčítána a tříděna podle komunikačního protokolu UranusLink [2].

3.4.1 UranusLink

Jedná se o protokol, prostřednictvím kterého probíhá komunikace mezi robotem a dálkovým ovladačem. Paket obsahuje šest prvků od synchronizační preamble až po kontrolní součet (viz Obr. 3.2).

- PRE (preamble, 1 byte), obsahuje hodnotu 0xFD a slouží k synchronizaci. Přijetí tohoto symbolu naznačuje začátek paketu. Tato hodnota se může vyskytnout také na konci paketu v bytu CS, pak je tato hodnota jako poslední z předchozího paketu a první z následujícího. Pokud jsou přijaty dva byty s touto hodnotou, uvažuje se druhý z nich jako PRE.
- SQN (sekvenční číslo, 2 byty), obsahuje hodnotu od 0 do 64 766, slouží ke kontrole toku dat. Odpověď na příkaz PING nebo změnu módu robota obsahuje právě sekvenční číslo dotazujícího paketu.
- MID (identifikační číslo, 1 byte), určuje typ dat, které paket nese. Hodnoty 0x00 - 0x07 jsou pro data z ovládání k robotu, hodnoty 0x80 - 0x8F pro data od robota k ovládání.
- LEN (počet bytů dat, 1 byte), maximální délka je 252 bytů. Maximální použitá délka je 20 bytů, pro příjem informací o poloze robota.
- DATA (n bytů), samotné data. Počet bytů je dán typem dat a hodnotou LEN.
- CS (kontrolní součet, 1 byte), slouží ke kontrole dat. Pokud tento součet neodpovídá skutečnému součtu, došlo k chybě v bezdrátové komunikaci, data jsou poškozena a nemohou být použita.



Obrázek 3.2: Struktura protokolu

U tohoto protokolu není pevně definovaná délka paketu, proto je potřeba přechíst byte LEN ještě před kontrolou správnosti dat, což může způsobit problém, pokud

by chyba nastala právě v bytu LEN. Po přijetí tohoto bytu se odpočítá požadovaný počet dat a z posledního bytu CS se porovná kontrolní součet. Pokud součet odpovídá, mohou se data uložit podle MID do požadovaných proměnných.

Robot odesílá paket AllDataStream (MID 0x86) s periodou 10 ms. Tento paket nese informace o náklonech (roll, pitch a yaw), výšce nadmořské i nad zemí, výkony všech čtyř motorů a stavový byte. Desetkrát za sekundu posílá informaci o poloze (0x8F), zeměpisnou šířku a délku, výšku, horizontální a vertikální přesnost. S nejmenší frekvencí, jednou za sekundu, odesílá informace o stavu baterie (0x81) a dobu provozu (0x8D).

V opačném směru se odesílají příkazy od joysticků (0x07) a to ve formátu 4 bytů roll, pitch, yaw a trust. Ostatní příkazy se neodesílají periodicky, například příkaz na změnu stavu robota (0x05), nebo příkaz pro kontrolu spojení Ping ve směru k robotu (0x00) i ve směru k ovládání (0x87). Na příkaz pro změnu módu musí dotazovaný odeslat odpovídající Ack se sekvenčním číslem. Při odesílání se uloží SQN číslo do proměnné `Ack_SQN` a nastaví se příznak očekávání odpovědi `WaitForACK` s informací o jaký mód se jednalo. Při přijetí ACK se porovná přijaté číslo s `Ack_SQN` a pokud je to očekávaná odpověď, příznak čekání se vynuluje. V rámci přerušovací rutiny pro měření ADC (perioda 100 ms) se kontroluje také stav proměnné `WaitForACK`. V každé periodě se inkrementuje počítadlo `counter`, pokud během deseti period měření (přibližně 1 s) nepřišla odpověď od robota, odešle se příkaz znovu. Mód, o který se jednalo, je v proměnné zakódován, horní 4 bity jsou nastaveny na hodnotu 0xF a spodní 4 bity podle módu na 0x0, 0x1 nebo 0x2. Pokud se neočekává žádná odpověď, je příznak vynulován. Stav robota zobrazený na displeji se nastaví až v případě, že je robotem potvrzen, do té doby je na displeji předchozí stav.

3.5 A/D převodníky

Pro obsluhu převodníků byla použita knihovna `adc_driver` od výrobce procesoru. Umožňuje nastavení parametrů ADC a čtení dat. Při inicializaci se načtou kalibrační hodnoty, nastaví se 8 bitové rozlišení, referenční napětí na `REFSEL_INTVCC` ($0,625 \cdot U_{CC}$) a prescaler na osminovou frekvenci. Dále se nastaví příslušné kanály multiplexeru převodníku. Každý kanál je nastaven jako singleended s jednotkovým zesílením a je mu přiřazen příslušný pin. Následně je převodník povolen a uloženy hodnoty offsetů. Joysticky musí být při inicializaci ve výchozí poloze, která se bude považovat za nulovou. Polohy joysticků jsou pak dány 8 bitovým číslem od -120 do 120. V blízkosti nuly je vytvořena mrtvá zóna, aby nedocházelo vlivem šumu k nechtěným příkazům.

Data z převodníků jsou vyčítána v přerušovací rutině od čítače/časovače TCD0, ten přeteče každých 100 ms. Převod se zahájí nastavením příznaku `ADC_CH_START` v registru `ADCA.CHn.CTRL`. Konec převodu je indikován v registru `ADC.CHn.INTFLAGS`, jakmile je připravena změřená hodnota je nastaven příznak `ADC_CH_CHIF`. Výsledky měření jsou přispůsobeny na požadovaný formát a uloženy do proměnných `ADC_RX`, `ADC_RY`, `ADC_LX` a `ADC_LY`. Na konci rutiny je přímo odeslán paket joystickovými příkazy `usartD0_TX_joycommand()`.

4 ZÁVĚR

V rámci předchozího semestrálního projektu byla navržena první verze DPS, která během programování a testování ukázala několik nedostatků. Chyba v návrhu zapojení joysticků neumožnila jejich zabudování. Z toho důvodu byl v této práci vypracován návrh nového zapojení a DPS, u které jsou odstraněny chyby a nedostatky. Testování funkčnosti jednotlivých komponent a komunikace bylo prováděno na první verzi desky upravené tak, aby byla provozuschopná. Především tedy zapojení zdroje LM2621 a připojení displeje přes prodlužovací kabel. Vzhledem k fázi laboratorního prototypu, nebylo v této práci řešeno napájení z akumulátoru, předpokládá se použití jednoho článku Li-Pol, pro testování byl použit síťový adaptér se stejnosměrným výstupním napětím 5,3 V.

Program pro mikrokontrolér byl vytvořen za pomoci několika knihoven, převážně od výrobce procesoru. Knihovna pro OLED displej byla upravena pro potřeby této aplikace. Komunikační protokol UranusLink byl implementován a otestován. Při komunikaci s počítačem byl test úspěšný, ověření celistvosti paketu kontrolním součtem, dekodování dat a reakce na příkaz Ping. K testu byl použit převodník USB→USART. Test komunikace s robotem neproběhl úspěšně, ovladač data přijímal i odesílal, ale po určité době se program zasekl. Při měření průběhu komunikace na osciloskopu, byla zjištěna možná příčina, ZigBee modul odesílá data v dávkách, kdy místo jednoho paketu po 10 ms odešle dva pakety přibližně po 20 ms. To je způsobeno polo-duplexní rádiovou komunikací, pokud přijímá, nemůže zároveň odesílat. Je tedy možné, že ovladač nestihne po přijetí jednoho paketu data zpracovat. Při dalším testu komunikace s PC byla snaha napodobit výstup ze ZigBee, ale ani při zřetězení několika paketů a zdvojnásobení přenosové rychlosti se nepodařilo docílit stejného chování, jako při komunikaci s robotem. K zaseknutí programu došlo, až při úmyslném zpoždění zpracování dat o 500 μ s. Během testování se nepodařilo odhalit příčinu tohoto selhávání, bude potřeba ovladač podrobit dalším testům.

Jako pokračování této práce by bylo možné uskutečnit výrobu nové verze DPS, zabudovat joysticky, odhalit a odstranit příčinu zasekávání ovladače.

REFERENCE

- [1] GÁBRLÍK, P. *Univerzální řídicí systém pro quadcopter*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2012. 83s. Vedoucí diplomové práce byl doc. Ing. Luděk Žalud, Ph.D.
- [2] GÁBRLÍK, Petr a Vlastimil KŘÍŽ. *UranusLink protocol specification*. VUT v Brně, 2013.
- [3] Atmel Corporation *ATxmega256A3U* [online]. 2013, poslední aktualizace 02/2013 [cit. 27. 12. 2013]. Dostupné z URL: <http://www.atmel.com/Images/Atmel-8386-8-and-16-bit-AVR-Microcontroller-ATxmega64A3U-128A3U-192A3U-256A3U_datasheet.pdf>.
- [4] Densitron Technologies *DD-12864YO-3A* [online]. 2010, poslední aktualizace 28. 11. 2012 [cit. 27. 12. 2013]. Dostupné z URL: <<http://www.densitron.com/GetPdf.aspx?nDisplayID=665>>.
- [5] Batron *COG-BTG12864-03* [online]. 2009, poslední aktualizace 21. 2. 2008 [cit. 27. 12. 2013]. Dostupné z URL: <http://batron.com/tl_files/dm_20120918/data/specification_BT45282.pdf>.
- [6] Digi International Inc. *XBP24-AUI-001* [online]. 2009, poslední aktualizace 23. 9. 2009 [cit. 27. 12. 2013]. Dostupné z URL: <http://ftp1.digi.com/support/documentation/90000982_B.pdf>.
- [7] Texas Instruments Inc. *TPS76733QD* [online]. 2004, poslední aktualizace 01/2004 [cit. 27. 12. 2013]. Dostupné z URL: <<http://www.ti.com/lit/ds/symlink/tps76733.pdf>>.
- [8] Texas Instruments Inc. *LM2621* [online]. 2005, poslední aktualizace 03/2005 [cit. 27. 12. 2013]. Dostupné z URL: <<http://www.ti.com/lit/ds/snvs033c/snvs033c.pdf>>.
- [9] Bourns *SRP5030T* [online]. 2013, poslední aktualizace 03/2013 [cit. 27. 12. 2013]. <<http://www.bourns.com/data/global/pdfs/SRP5030T.pdf>>.
- [10] Vishay General Semiconductor *SS16* [online]. 2012, poslední aktualizace 10. 2. 2013 [cit. 27. 12. 2013]. <<http://www.vishay.com/docs/88746/ss12.pdf>>.

SEZNAM ZKRATEK

ADC Analog-Digital Converter – analogově digitální převodník

DPS Deska plošných spojů

ETSI European Telecommunications Standards Institute – Evropský ústav pro telekomunikační normy

I²C Inter Integrated Circuit – sériová sběrnice

JTAG Joint Test Action Group – rozhraní pro programování a debugování

LCD Liquid Crystal Display – displej z tekutých krystalů

LDO Low-Dropout – obvod s nízkým úbytkem napětí

MCU Micro Controller Unit – Mikrokontroler

OLED Organic Light Emitting Diode/Display – displej využívající organické elektroluminiscenční diody

SPI Serial Peripheral Interface – sériová sběrnice

USART Universal Synchronous Asynchronous Receiver/Transmitter – univerzální synchronní asynchronní sériové rozhraní

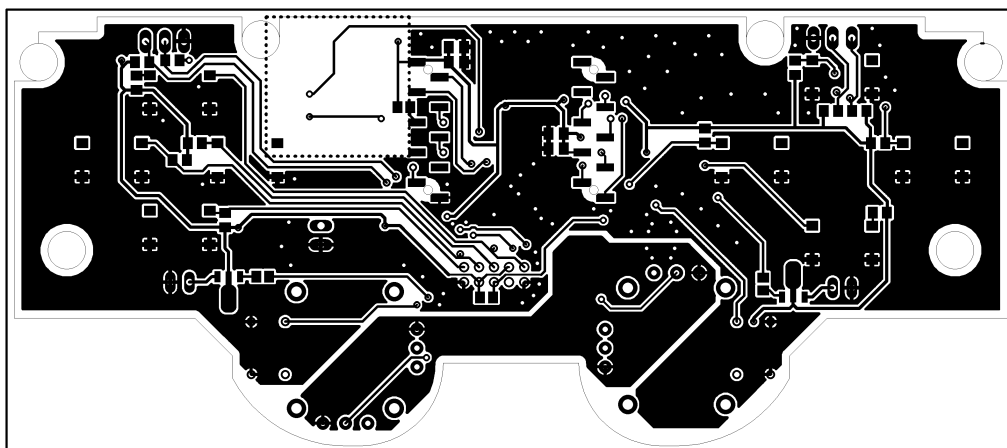
USB Universal Serial Bus – univerzální sériová sběrnice

SEZNAM PŘÍLOH

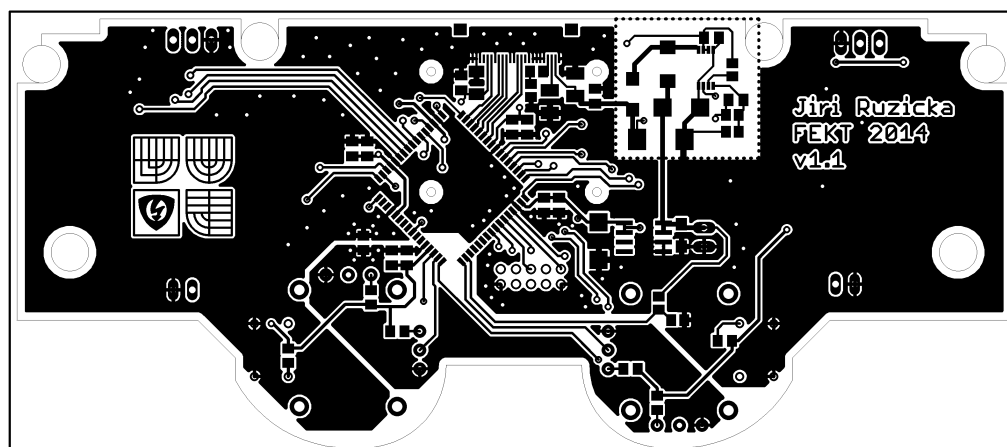
A	DPS 1.1	33
	A.1 Horní strana desky	33
	A.2 Dolní strana desky	33
B	Osazovací plány	34
	B.1 Horní strana desky	34
	B.2 Dolní strana desky	34
C	Příloha na CD a schéma zapojení	35
D	Fotodokumentace 1	36
E	Fotodokumentace 2	37

A DPS 1.1

A.1 Horní strana desky



A.2 Dolní strana desky



C PŘÍLOHA NA CD A SCHÉMA ZAPOJENÍ

Obsah CD:

- Elektronická verze práce
- Elektronická verze schématu zapojení formátu A3
- Zdrojové kódy
- Zdrojové soubory Eagle (*.brd, *.sch) DPS v1.0 a DPS v1.1
- Fotodokumentace v elektronické podobě

Schématem zapojení

Výkres formátu A3 se schématem je přiložen jako volný list.

D FOTODOKUMENTACE 1



E FOTODOKUMENTACE 2

