

ZÁPADOČESKÁ UNIVERZITA V PLZNI
FAKULTA ELEKTROTECHNICKÁ

Diplomová práce

INSPEKČNÍ ROBOT PRO HDPE TRUBKY

Filip VESELÝ

Červen 2021

ANOTACE A KLÍČOVÁ SLOVA

Předkládaná diplomová práce je zaměřena na vytvoření malého inspekčního robota, který bude schopný autonomního pohybu v úzkých trubkách. Jeho úkolem bude dojet k místu závady, pořídit fotku a vrátit se zpět. V práci je řešena konstrukce těla robota na 3D tiskárně, pohon stejnosměrnými motory, step-up měnič, komunikace ultrazvukem, senzor otáček, paměťové úložiště, kamerový modul a další elektronika v čele s řídicím procesorem. Součástí je i návrh desek plošných spojů a jejich realizace. Softwarová část se zabývá řídicím programem pro mikroprocesor v jazyce C.

Klíčová slova

Inspekční robot, PAL kamera, ultrazvuk, step-up měnič, jazyk C, STM32 mikroprocesor.

ANOTATION AND KEYWORDS

The presented diploma thesis is focused on the creation of a small inspection robot that will be able to move autonomously in narrow pipes. His task will be to get to the place of the defect, take a photo and return. The work deals with the construction of the robot body on a 3D printer, DC motor drive, step-up converter, ultrasonic communication, speed sensor, memory storage, camera module and other electronics headed by a control processor. It also includes the design of printed circuit boards and their implementation. The software part deals with the control program for the microprocessor in the C language.

Keywords

Inspection robot, PAL camera, ultrasound, step-up converter, STM32 microprocessor, C language.

PODĚKOVÁNÍ

Tímto bych chtěl poděkovat vedoucímu diplomové práce Ing. Petru Weissarovi, Ph.D. za přínosné nápady a otázky, kterými mě dokázal vyvést ze slepých uliček. Především děkuji své rodině za podporu, díky které jsem se mohl plně věnovat studiu a této diplomové práci.

PROHLÁŠENÍ

Předkládám tímto k posouzení diplomovou práci, zpracovanou během mého studia na Fakultě elektrotechnické Západočeské univerzity v Plzni.

Prohlašuji, že jsem tuto práci vypracoval samostatně s použitím uvedené odborné literatury a pramenů a že veškerý software, použitý při jejím řešení a zpracování, byl využit s respektováním všech jeho licenčních podmínek.

V Plzni, dne 28. 5. 2021

Filip VESELÝ

OBSAH

1	Pokládání optických kabelů	2
1.1	Kontrola průchodnosti trubky	2
1.2	Zafouknutí optického kabelu	4
1.3	Rádiová sonda	5
2	Piercův krystalový oscilátor	6
2.1	Zpětnovazební rezistor RF	7
2.2	Zatěžovací kapacita CL	7
2.3	Rezistor Rext	8
2.4	Návrh DPS	8
2.5	Pokud krystal neosciluje	8
3	DCMI rozhraní	10
3.1	Princip časování a přenosu dat	10
3.2	Zabudovaná synchronizace	11
3.3	Externí synchronizace	12
3.4	Režimy snímání	13
3.5	Ořez obrázku	13
4	Analogové video	14
4.1	Kompozitní video	14
4.2	S – video	15
4.3	NTSC kódování	15
4.4	PAL kódování	16
5	Formáty digitalizovaného videa	17
5.1	ITU-R BT.656	17
5.2	8-Bit 4:2:2	17
6	Motivace k vytvoření robota	18
7	Použitá technologie a software	19
8	Tělo robota	21
8.1	Materiál	21
8.2	Tělo s pásy	22
8.3	Tělo s plováky	22
8.4	Tělo s přitlačnou plastovou fólií	23
9	Univerzální vagónek	24

10 Bateriový vagónek	26
10.1 Li-ion baterie	26
10.2 Hallův senzor otáček	26
10.2.1 Hardware	26
10.2.2 Software MCU	27
11 Komunikační vagónek	28
11.1 DC/DC měnič	28
11.2 Lineární stabilizátory	31
11.3 Stejnoseměrný motor	31
11.4 H-můstek	32
11.4.1 Hardware	32
11.4.2 Software MCU	32
11.5 Ultrazvukové čidlo	33
11.5.1 Hardware	33
11.5.2 Software MCU	36
12 Inspekční vagónek	39
12.1 Kamera	39
12.2 Video dekodér	40
12.2.1 Výběr dekodéru	40
12.2.2 Zapojení dekodéru	40
12.2.3 Software	41
12.3 Mikro SD karta	44
12.3.1 Hardware	44
12.3.2 Software MCU	44
12.4 Mikroprocesor	45
12.4.1 Volba mikroprocesoru	45
12.4.2 Napájení	45
12.4.3 Návrh krystalového oscilátoru	45
12.4.4 Volba bootování	45
12.4.5 Programování přes SWD	46
12.4.6 Software MCU	47
13 Závěr	48

SEZNAM SYMBOLŮ A ZKRATEK

ITU	International Telecommunication Union
DCMI	Digital Camera Memory Interface
DPS	Deska plošných spojů
DMA	Direct Memory Access
MCU	Micro Controller Unit
PCB	Printed Circuit Board
FAT	File Allocation Table
FPV	First Person View
LDO	Low Drop Out
TVL	TV Lines

ÚVOD

CÍLEM této práce je navrhnout a zkonstruovat prototyp robota, který se dokáže protáhnout HDPE trubkou o vnitřním průměru 3 cm, dojde k případné poruše a vrátí se zpět s informacemi, které pomohou technikům zvolit správný postup opravy. Robot za sebou nepotáhne napájecí kabel, jak bývá běžné, ale poveze si energii ve formě Li-Ion baterie. Přestože jeho pohyb bude autonomní, půjde mu na dálku posílat příkazy prostřednictvím ultrazvukového signálu a ovlivnit tak jeho chování. Tělo robota bude vyrobeno pomocí 3D tisku a jeho pohon zajistí dvojice stejnosměrných motorů s převodovkou. V části práce věnující se elektronice budu řešit například step-up měnič, ultrazvukový přijímač, řídicí desku s procesorem, snímač otáček nebo záznam obrazu.

Nápad na vytvoření inspekčního robota přišel na letní brigádě ve společnosti zabývající se pokládáním optických kabelů do země. Jde o celkem složitý proces (popsáno dále), při kterém je mimo jiné nutné zjistit průchodnost podzemních HDPE trubek. Toho se dosahuje speciální technikou, jejíž cena se pohybuje ve stovkách tisíc korun. Proto jsem začal přemýšlet o robotovi, který by zvládl to samé a za zlomek ceny. Ve světě jsem zaregistroval podobné pokusy o inspekční roboty do trubek. Vždy ale šlo o trubky mnohem větších průměrů, případně kanalizace. Vytvoření robota, který by se dokázal vejít do trubky průměru 3 cm je proto zajímavá výzva.

V teoretické části této práce popisují způsob pokládky optických kabelů a k tomu potřebné technické vybavení. Dále je zde postup, jak správně navrhnout Piercův krystalový oscilátor, včetně jeho diagnostiky a případné opravy, pokud nechce oscilovat. S návrhem inspekčního robota souvisí i DDMI rozhraní pro příjem digitálního video signálu, kterým disponují procesory od firmy STM. Závěrem teoretické část rozeberu problematiku přenosu analogového a digitálního videa.

Praktická část popisuje konstrukční, elektronickou i programovou stránku robota. Ten se skládá ze tří vagónků. Každý z nich plní svůj specifický úkol a jejich propojením vzniká robot. V první části popisují mechanickou konstrukci vagónku z ABS plastu. Dále se dopodrobna zabývám jednotlivými vagónky a jejich elektronikou. Pokud nějaký modul vagónku potřebuje pro svou činnost řízení mikroprocesorem, je přiložen i popis programu. Schémata, návrhy plošných spojů a další technické dokumenty se nacházejí na konci této práce.

POKLÁDÁNÍ OPTICKÝCH KABELŮ

POLOŽIT dlouhý kabel, vedoucí např. mezi městy je náročné, nejen technicky, ale i právně. Nutnost zajistit povolení všech osob, kterým povede plánovaná trasa přes pozemek, je často zdoluhavé a mnohdy i nemožné. Proto se často využívá historických HDPE trubek, které byly položeny v době, kdy se lidé báli protestovat. Nemusí se tak provádět výkopové práce, což položení kabelu velmi zlevní. Nevýhodou je nutnost kontroly průchodnosti trubky a oprav v místech, kde došlo k jejímu promáčknutí nebo prasknutí. K tomu může dojít např. zatížením zeminy nad trubkou zemědělským strojem. Nejčastěji trubku naruší výkopovými pracemi lidé, kteří předem neprovedou trasování podzemních vedení.

1.1 Kontrola průchodnosti trubky

Nejprve si technik pomocí speciálních map vytvoří povědomí o tom, kudy HDPE trubka vede. Tuto trasu projde nebo projede autem, přičemž se snaží odhalit, kde mohly probíhat nebo právě probíhají výkopové práce. S největší pravděpodobností bude chyba zde. Společnosti, které trubku poškodí, totiž málokdy svoji chybu nahlásí. Pokud je tento postup neúspěšný, je nutné přistoupit k lokalizaci poruchy pomocí sondy často zvané „mlok“.

K ústí trubky se přiveze kompresor. Ten musí mít dostatečný výkon pro natlakování až několikakilometrové trubky. Bývá dokonce tak velký, že je nutné ho na místo dotáhnout malým nákladním autem. Do trubky se nejprve vsune molitanový míček na líný tenis potřený lubrikantem. Na trubku se nasadí spojka, ke které se připojí kompresor. Začne se zvyšovat tlak a míček se dá do pohybu. Pokud po chvíli vyletí z druhé strany trubky, lze usuzovat na alespoň částečnou průchodnost. Tím si technik zajistil, že nedojde ke ztrátě nebo zničení drahé sondy.

Nakonec se do trubky vsune sonda (detailní popis sondy dále) a provede se natlakování. Pokud je trubka průchozí, nabere sonda obrovskou rychlost. Na vyústění trubky je proto nutné nainstalovat lapač, aby sonda při neopatrné manipulaci někoho nezastřelila. V případě, že se po cestě vyskytuje nějaká porucha, sonda do ní narazí a zastaví se. To se projeví mimo jiné vzrůstem tlaku.



Obr. 1.1: Zafukování optického kabelu kompresorem

Aby bylo možné určit, kde k zastavení došlo, obsahuje sonda obvod, který začne pod zemí vysílat signál, často na frekvenci 33 kHz. Nyní přichází na řadu technik. Ten se podle mapy a GPS lokátoru snaží jít nad trubicí. V ruce drží hledáčku (nízkofrekvenční přijímač) a pohybem ze strany na stranu se snaží zachytit signál ze sondy. Když k zachycení signálu dojde, technik místo označí a provede výkop. Následně trubku odkryje, speciálními kleštěmi rozstříhne a sondu vytáhne. Kolem místa poruchy se trubka zkrátí a do vzniklé mezery se vloží, a spojkami připevní, nový kus trubky.



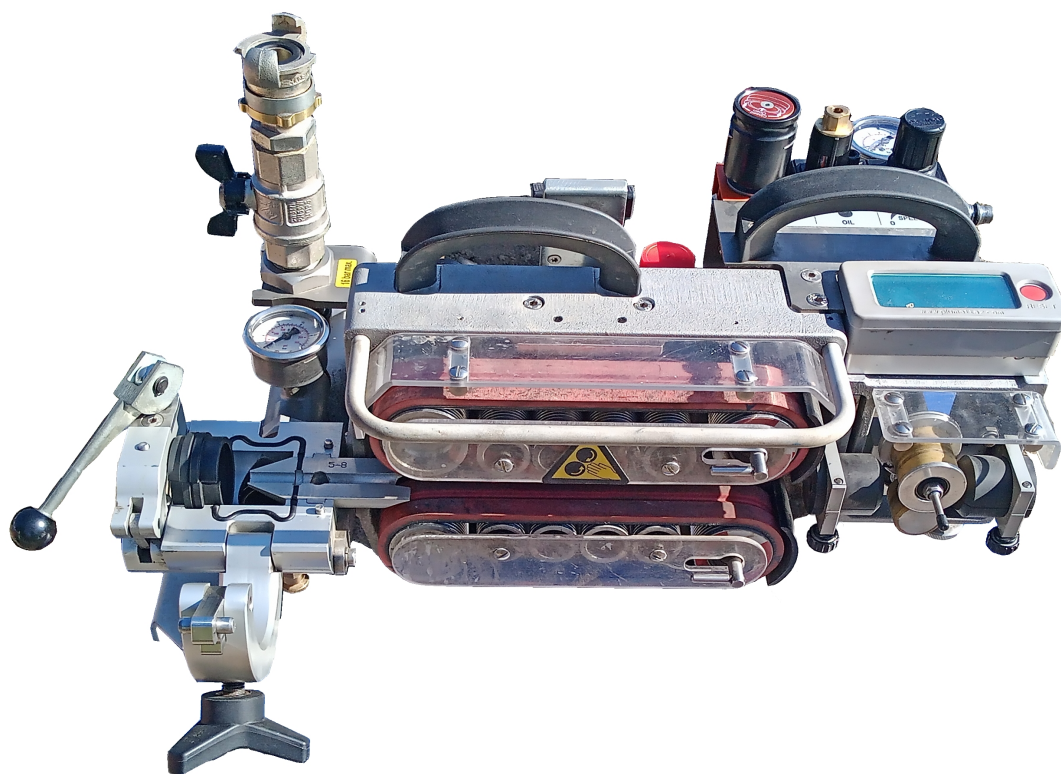
Obr. 1.2: Hledáčka [1]

1.2 Zafouknutí optického kabelu

Po kontrole průchodnosti trubky přijde na řadu naplnění trubky mikro trubičkami. Princip, jakým se mikro trubičky zasunou do trubky, je podobný zafukování optického kabelu, který je popsán dále. Díky mikro trubičkám dojde k navýšení kapacity trubky a snížení budoucích nákladů na úpravu optické trasy. Je to tím, že požadavek na navýšení kapacity nepovede k vytahování dříve položeného optického kabelu a jeho náhradě za silnější, ale v případě použití mikro trubiček bude stačit zafouknout další optický kabel do zatím neobozazené mikro trubičky.

Jak název napovídá, při takzvaném „zafouknutí“ optického kabelu se znovu využije tlak kompresoru. Část kabelu se odmotá z bubnu a zasune pod přitlačné řemeny speciálního pneumatického pohonu. Odtud je kabel veden do těsnění. To zajišťuje, že tlakový vzduch z kompresoru proudí pouze do trubky a kabel je tak unášen na vzduchovém polštáři. Na konci optického kabelu se vytvoří pomocí izolepy bambule (zlepší průchodnost, zvětší odpor vzduchu) a zasune se do trubky. Spustí se kompresor a tím i pneumatický pohon, který začne tlačít kabel do trubky. Jde ale pouze o přídavnou sílu. Hlavní tlačnou práci odvádí tlakový vzduch, který nese kabel trubkou až k jejímu ústí. Pneumatický pohon obsahuje i měřič délky kabelu. Při znalosti délky trubky je možné před dosažením konce zpomalit. Další trik, který se využívá při velmi dlouhých vzdálenostech, je zafukování kabelu ze středu trasy na obě strany.

Závěrem se provádí měření útlumu vlákna pomocí optického reflektometru (OTDR). Jde o drahé zařízení, které kromě celkového útlumu trasy dokáže odhadnout v jaké vzdálenosti se nachází případná porucha kabelu.



Obr. 1.3: Pneumatický pohon [2]

1.3 Rádiová sonda

Sonda se díky tlakovému vzduchu dostane až do místa, kde je trubka zúžená nebo jinak poničená. V tomto místě se zastaví a začne vysílat na povrch, kde může být zachycena hledačkou. Napájení často zajišťuje AAA baterie [3]. Sondy jsou voděodolné. Pro vysílání se používají nízké frekvence, které snáze pronikají skrz horniny. Nejčastěji sondy vysílají na frekvencích 614/512 Hz, 8 kHz nebo 33 kHz. Např. na frekvenci 33kHz lze sondu v hlíně zaměřit až do hloubky 5 m.

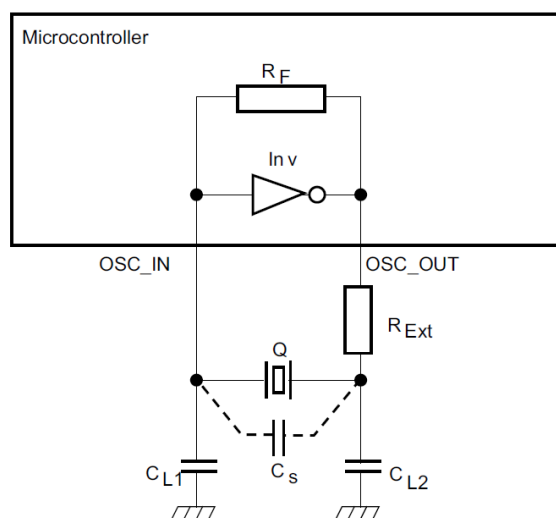


Obr. 1.4: Rádiová sonda

PIERCŮV KRYSTALOVÝ OSCILÁTOR

TENTO typ oscilátoru se velmi často používá jako přesný zdroj externích hodin pro mikroprocesory, analogově digitální převodníky a další. Vyznačuje se velkou stabilitou kmitočtu a nízkými energetickými ztrátami (vysoký činitel jakosti).

Na následujícím obrázku je vidět základní zapojení Piercova oscilátoru. Při paralelní rezonanci se krystal chová jako indukčnost, která ve smyčce spolu se sítí kondenzátorů přidá fázový posun rovný 180° . Invertor uvnitř MCU signál zesílí a přidá další fázový posun o 180° . Tím jsou podmínky pro vznik oscilací splněny.

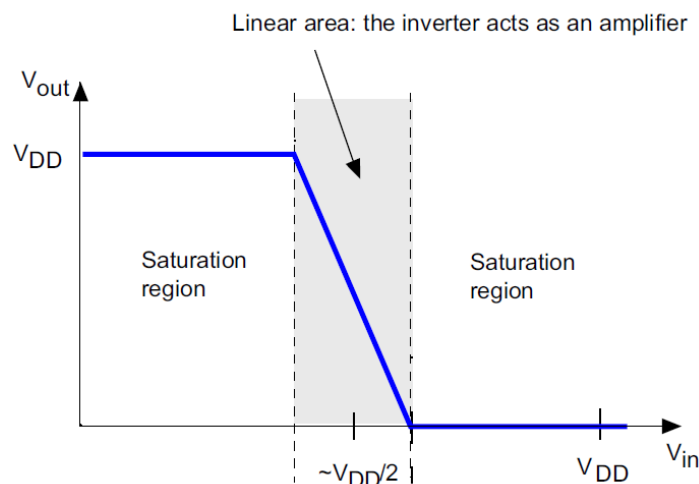


Obr. 2.1: Piercův oscilátor [4]

- Inv** – invertor, který pracuje jako zesilovač
- Q** – krystalický křemen nebo keramický rezonátor
- RF** – interní zpětnovazební rezistor
- RExt** – externí odpor k omezení proudu oscilátorem
- CL1 a CL2** – dva externí zatěžovací kondenzátory
- Cs** – parazitní kapacita pinů MCU (OSC_IN a OSC_OUT) a DPS

2.1 Zpětnovazební rezistor RF

CMOS invertor je díky zpětnovazebnímu rezistoru RF nastaven do středu své převodní charakteristiky. Pracovní bod se zde pohybuje po zhruba lineárním úseku a invertor tak funguje jako zesilovač – malá změna V_{in} vyvolá velkou změnu V_{out} . Čím větší rezistor bude, tím větší bude i zesílení. Nastavení pracovního bodu pomocí rezistoru nebo jiného obvodu už za nás často vyřeší výrobce čipu. Někdy se ale může přidat i vlastní externí rezistor. Díky tomu se z výstupu přivede silnější signál zpět na vstup invertoru, což stimuluje ke spuštění oscilací.



Obr. 2.2: Převodní charakteristika [4]

2.2 Zatěžovací kapacita CL

Zatěžovací kapacita je součtem kapacit připojených k oscilátoru. Patří sem CL_1 , CL_2 , ale i parazitní kapacita mezi plošnými spoji (C_S). Kapacitu CL určuje výrobce krystalu. Pokud ji připojením CL_1 a CL_2 zajistíme ve správné velikosti a dostatečně stabilní, dosáhneme přesné rezonanční frekvence. Kondenzátory by měly být kvalitní keramické, určené pro vysokofrekvenční použití a nejlépe s dielektrikem NP0 nebo C0G. Kapacita C_S se počítá těžko a často se při správném návrhu plošného spoje odhaduje na 3 až 8 pF. Následuje ukázkový výpočet zatěžovacích kapacit CL_1 , CL_2 (volí se stejné), při známé parazitní kapacitě C_S a kapacitě určené výrobcem CL .

$$C_S = 5\text{pF}$$

$$CL = 15\text{pF}$$

$$CL = \frac{1}{2}CL_{1,2} + C_S$$

$$CL_{1,2} = 20\text{pF}$$

2.3 Rezistor Rext

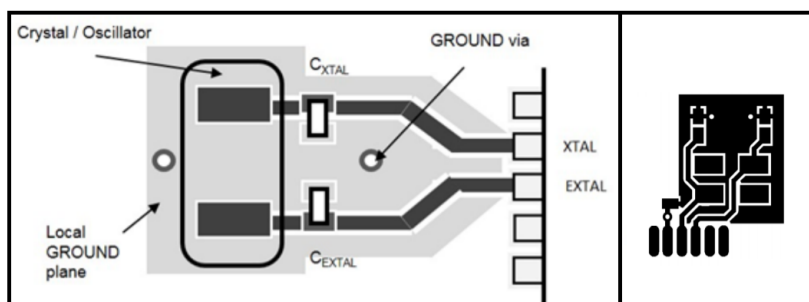
Maximální výkon rozptýlený v krystalu nesmí překročit určitou hodnotu, jinak se může krystal poškodit kvůli nadměrným vibracím. Tento výkon (Drive level – DL) udává výrobce obvykle v mW. Ve většině návrhů tento rezistor není třeba.

2.4 Návrh DPS

Při návrhu oscilátorů se osvědčily tyto techniky:

- Co nejkratší spoje od oscilátoru k MCU i mezi součástkami.
- Spoje by měly mít stejnou délku.
- Kolem vodičů rozlít měď (kterou ovšem neteče proud), aby byly co nejvíce stíněné a tu připojit ke společné zemní rovině.
- Prokovy dělat jen na zem. Pokud je nutné překonat nějakou překážku je lepší použít nulový rezistor.
- Spoje by neměly být kolmé, aby nedocházelo k odrazům signálu.
- Pod oscilátorem na opačné straně desky by neměly vést žádné spoje.

Ukázka správně navrženého plošného spoje s THT a SMD krystalem:



Obr. 2.3: Správně navržené plošné spoje [5] [6]

2.5 Pokud krystal neosciluje

Zde je pár triků, které mohou pomoci, pokud krystal nechce oscilovat.

- Je MCU přepnut na externí hodiny?
- Je MCU správně napájen?
- Nejsou vývody oscilátoru zkratované na zem?
- Měřte se sondou 1:10, ta méně ovlivňuje slabé signály.
- Sondu je dobré připojovat na výstup invertoru. Připojením na vstup invertoru může dojít k rozladění oscilátoru a jeho zastavení.

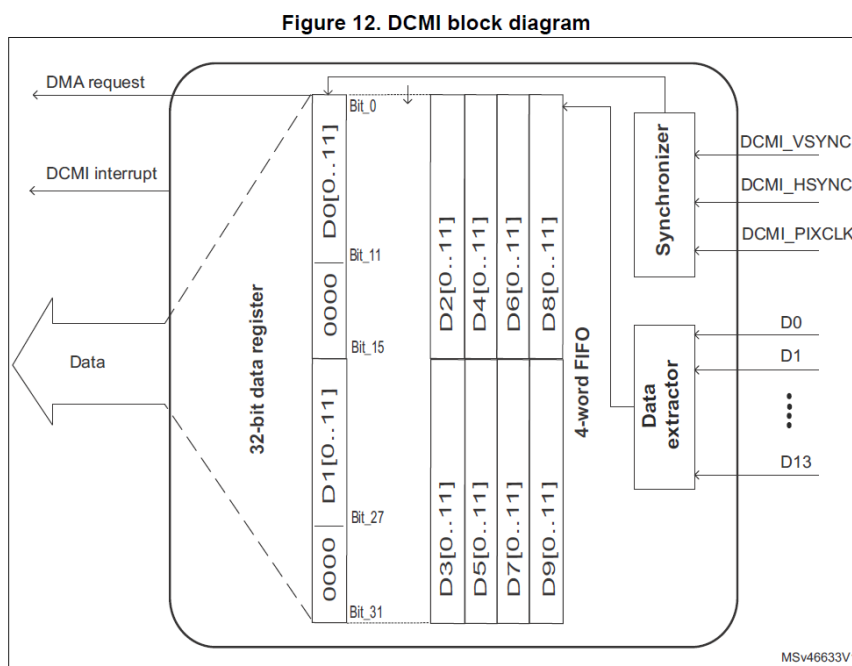
- Zkuste odstranit zatěžovací kondenzátory. Přívody ke krystalu samy o sobě mohou mít dostatečnou kapacitu.
- Zkuste připojit kapacitní trimry a ladit kapacity, až se obvod rozkmitá.
- Zkuste změnit hodnotu externího rezistoru RF.
- Připojením osciloskopické sondy může dojít k zavlečení další kapacity do rezonančního obvodu a ten přestane kmitat. Tedy měřím signál, který tam doopravdy vůbec není. Zkuste měřit aktivní sondou, nebo klasickou jen přiblížte vývodu krystalu. Mělo by se do ní naindukovat malé napětí. Jeho tvar nebude odpovídat tomu co se reálně nachází na krystalu, frekvence by ale mohla být vypovídající.
- Nevhodné tavidlo může způsobit zkraty na zem. Očistěte DPS.
- Místo krystalu použijte oscilátor. Tedy součástku se čtyřmi vývody. Dva jsou pro napájení a dva pro výstupní oscilující signál, kde správný návrh už zajistil výrobce.

DCMI ROZHRANÍ

U MOŽŇUJE přijímat digitální video signál a přes DMA data zapisovat do paměti bez účasti procesoru. Podporuje externí i zabudovanou synchronizaci snímků a řádek. Dále popíše princip a funkce DCMI rozhraní.

3.1 Princip časování a přenosu dat

DCMI pro svou funkci vyžaduje vodiče datové, jeden hodinový (PIXCLK) a v případě externí synchronizace (viz. dále) vodiče VSYNC a HSYCH. Pro oba typy synchronizací ale platí následující: při každém tiku na vývodu PIXCLK je zachyceno zvolené množství video dat a přesunuto do FIFO bufferu o velikosti 4 x 32 bitů. (Pokud je AHB sběrnice zrovna vytížená, nedojde díky vyrovnávacímu bufferu k okamžitému přepisu nevyzvednutých dat.) Odtud se data posouvají do 32bitového datového registru. Když je naplněn, vyšle se požadavek na DMA přenos a data jsou přenesena do cílové paměti, kde mohou být zpracována.



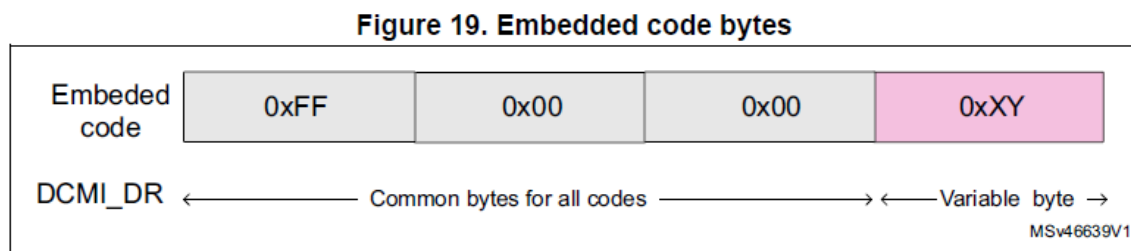
Obr. 3.1: Blokový diagram DCMI [7]

Tento cyklus se v Normal módu opakuje, dokud NDTR čítač DMA přenosů nedosáhne nuly nebo periférie nezahlásí konec přenosu. V Normal módu DMA lze přenést až 65535 32bitových slov. Je-li obrázek větší, jde použít Double buffer mód, který ukládá data do prvního bufferu, zatímco z druhého lze data číst a naopak. Při jeho použití je automaticky zapnut kruhový režim DMA (circular mód). Ten zajišťuje, že po dokončení přesunu dat se provede opětovné naplnění NDTR registru a přenos začíná znovu. Takto lze zpracovat dvakrát větší obrázek než v Normal módu.

Pokud DMA nestihne odebírat data, dojde k přetečení FIFO bufferu. Ten se vyresetuje a čeká na nový příjem. Předejít přetečení se dá zrychlením taktování mikroprocesoru, kdy poměr AHB/ DCMI_PIXCLK musí být roven minimálně 2.5.

3.2 Zabudovaná synchronizace

Do MCU je zavedeno 8 datových vodičů a jeden taktovací signál PIXCLK (pixel clock). Časování a přenos dat probíhá tak, jak bylo popsáno v kapitole 3.1. Aby se poznalo, jaká část obrázku je právě přenášena, jsou do datového toku vloženy synchronizační značky, které indikují začátek (SAV) a konec (EAV) řádku a obrázku. Protokol ITU-R BT.656 definuje takovouto formu značky (32bitové slovo):



Obr. 3.2: SAV a EAV kódy [7]

XY bajt upřesňuje značku a obsahuje tři proměnné a čtyři ochranné bity.

1	F	V	H	P3	P2	P1	P0
---	---	---	---	----	----	----	----

F ... u prokládaného video signálu se přenáší sudé, nebo liché řádky

1 = sudý řádek

0 = lichý řádek

V... vertikální zatemnění.

1 = vertikální zatemňovací interval aktivní

0 = vertikální zatemňovací interval neaktivní

H ... horizontální zatemnění .

1 = horizontální zatemňovací interval aktivní

0 = začíná řádka s obrazovými daty

P3 ... ochranný bit 3

$$P3 = V \text{ xor } H$$

P2 ... ochranný bit 2

$$P2 = F \text{ xor } H$$

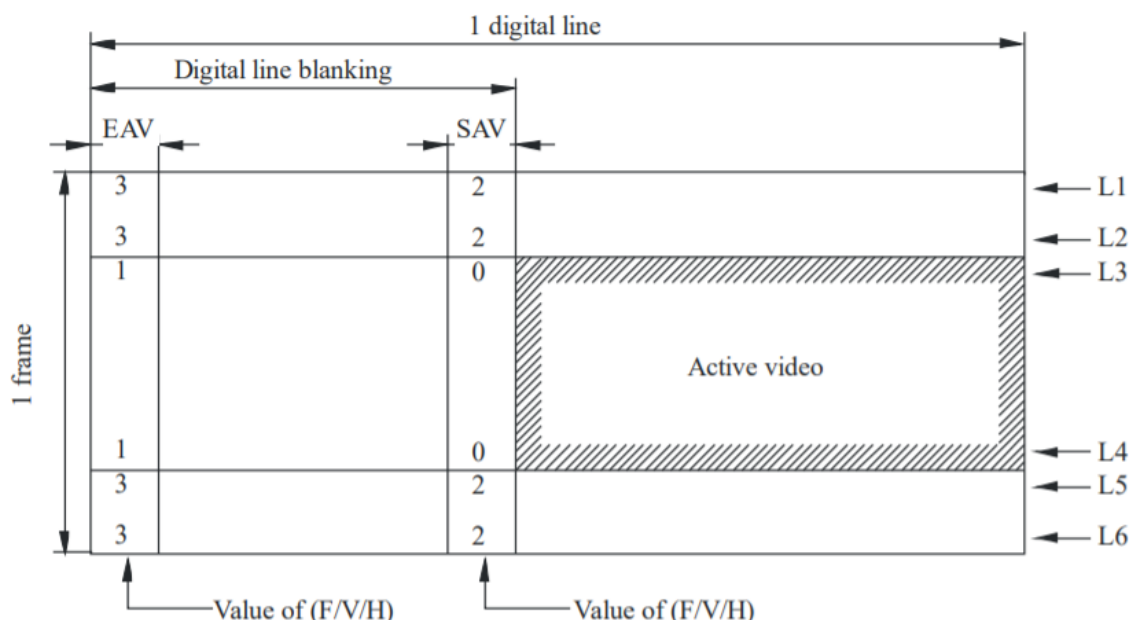
P1 ... ochranný bit 1

$$P1 = F \text{ xor } V$$

P0 ... ochranný bit 0

$$P0 = F \text{ xor } V \text{ xor } H$$

Takto podle normy vypadá přenos obrazových dat doplněných o SAV a EAV značky. Například začátek řádku s obrazovými daty má bity FVH = 000, což spolu s ochrannými bity dá bajt XY = 0x80. Při konfiguraci DDMI rozhraní je nutné správně zapsat SAV a EAV kódy do registrů mikroprocesoru.

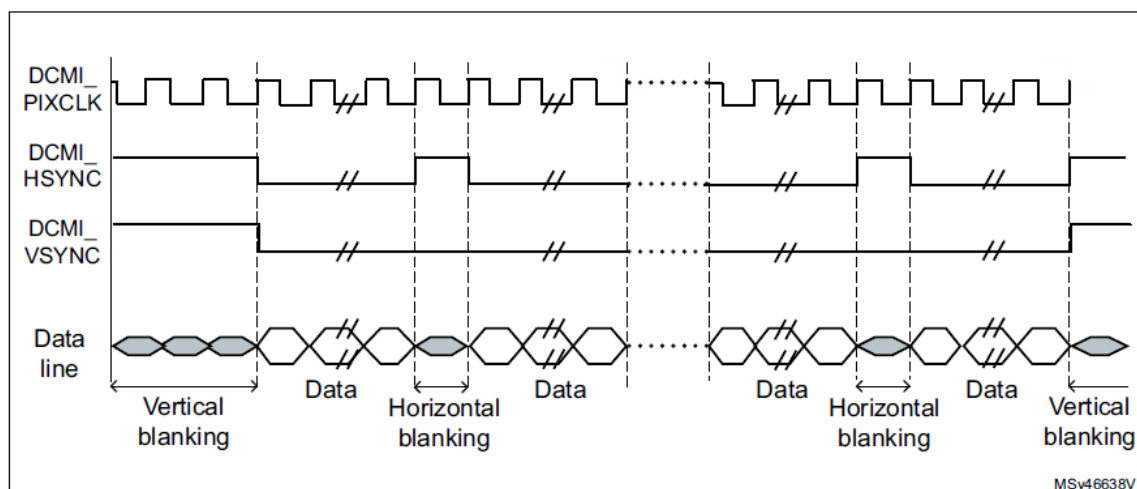


Obr. 3.3: Přenos obrázku se zabudovanou synchronizací [8]

Tento způsob zabudované synchronizace ušetří dva vodiče a návrh desky je jednodušší. Naopak konfigurace DDMI rozhraní je poté složitější.

3.3 Externí synchronizace

Do MCU je zavedeno 8 až 12 datových vodičů, jeden taktovací signál PIXCLK a další dva synchronizační vodiče HSYNC (horizontální synchronizace) a VSYNC (vertikální synchronizace). Princip časování a přenosu dat je stejný jako u zabudované synchronizace. Informaci o tom, k jaké části obrázku přichází data patří, ovšem dávají VSYNC a HSYNC signály. Ve chvíli jejich aktivace, se data neberou jako platná a jsou ignorována a naopak. Obrázek níže zobrazuje průběh synchronizačních pulsů v čase. Data, která se zahazují, jsou vyplněna šedivou barvou.



Obr. 3.4: Přenos obrázku s externí synchronizací [7]

Externí synchronizace je náročnější na počet signálů. Piny MCU určené pro VSYNC a HSYNC se navíc nacházejí na opačných stranách pouzdra a návrh plošného spoje je tak složitější. Softwarovou stránku ale externí synchronizace velmi zjednoduší.

3.4 Režimy snímání

DCMI podporuje dva typy snímání: jeden snímek (Snapshot) a nepřetržitě probíhající snímání (Continuous grab). V závislosti na výběru typu snímání a velikosti obrázku je potřeba správně nastavit DMA přenos (viz. kapitola 3.1).

- **Snapshot mód** – po nahození bitu CAPTURE se čeká na příchod začátku snímku, poté je zachycen jen jeden obrázek a CAPTURE bit je automaticky shozen.
- **Continuous grab mód** – po nahození bitu CAPTURE se čeká na příchod začátku snímku a poté se začnou pořizovat snímky, dokud není softwarově shozen CAPTURE bit. Dokud je mód zapnutý, generují se požadavky na DMA přenosy. Pokud nedojde zavčas k jejich obsluze, DCMI vyhodí chybu.

3.5 Ořez obrázku

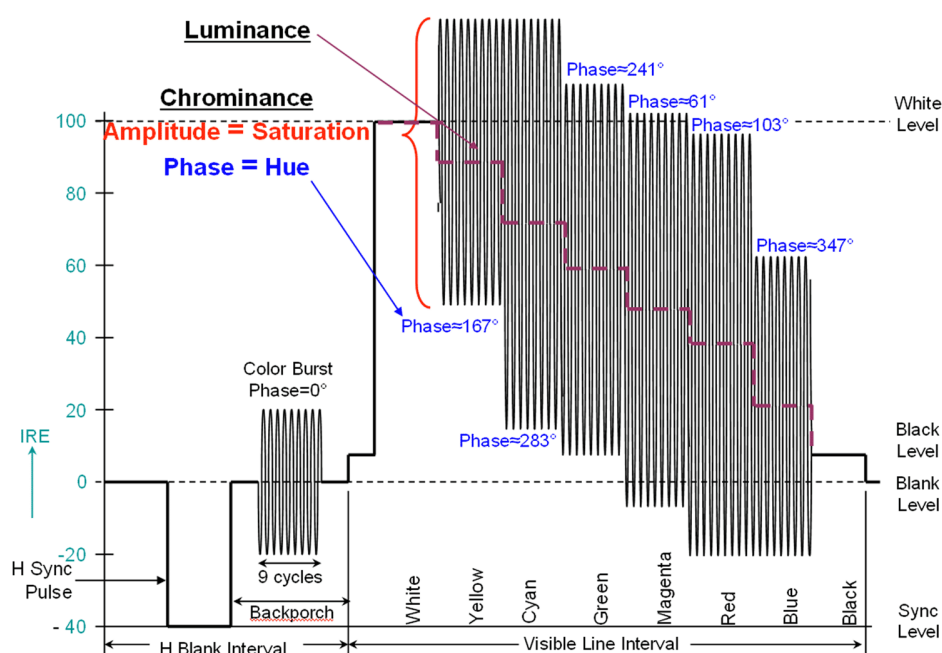
Cropping je další zajímavou funkcí DCMI rozhraní. Umožňuje oříznout přichodící obrázek. Jinak řečeno, část přichodících obrazových dat je zahozena, což sníží nároky na kapacitu úložné paměti. Registry CWSTRTDCMI, CWSIZE pracují na opačné logice. Nastavuje se zde, od jakého řádku a sloupce se začnou obrazová data ukládat a jakou šířku a výšku bude mít vyříznutý obrázek.

ANALOGOVÉ VIDEO

SROSTOUCÍ popularitou dronů začalo přibývat FPV kamer. Mají velmi malé rozměry, a proto se hodí pro inspekční účely ve stísněných prostorech. Kamery jsou založeny na CCD nebo CMOS snímačích. CCD snímače převádějí světlo na elektrický náboj, který se odesílá mimo čip jako analogový signál. Kódování a způsob přenosu má několik variant. Některé z nich uvedu v následujícím textu.

4.1 Kompozitní video

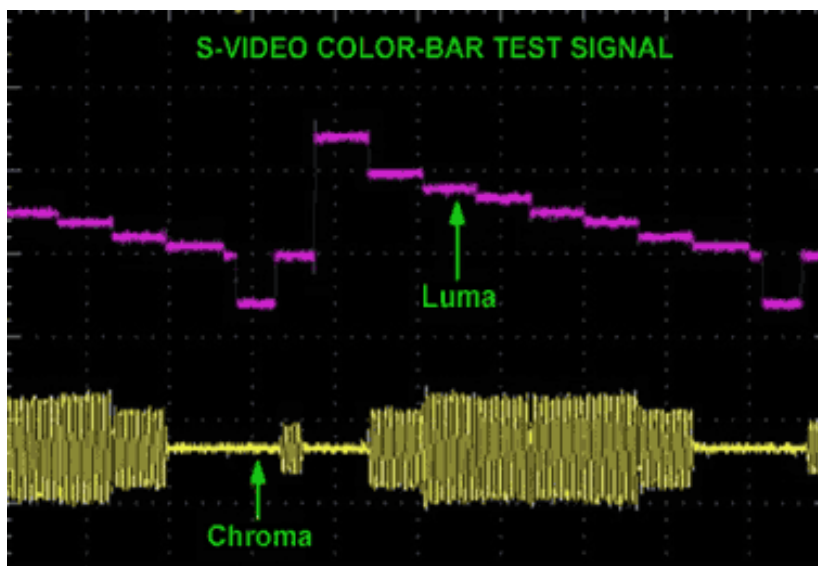
Jde o analogový přenos po dvou vodičích – video a zem. Obraz se z kamery přenáší postupně po řádcích. Signál reprezentující obraz vznikne kombinací amplitudy jasové složky (luminance) a fázově modulované barvosné složky (chrominance). Jde tedy o kvadraturní amplitudovou modulaci. Průběh kódování barvosné složky se liší v závislosti na standardu (PAL, NTSC). Do analogového signálu jsou navíc přidány synchronizační intervaly, kdy dochází buď k přesunu na další řádek (horizontální synchronizace), nebo k přesunu na další snímek (vertikální synchronizace). Vzorový průběh analogového signálu je vidět na následujícím obrázku.



Obr. 4.1: Kompozitní video signál [9]

4.2 S – video

Stejně jako u kompozitního videa se informace o obraze přenáší pomocí luminance a chrominance. Zachovány jsou i synchronizační signály v luminanci. Rozdílný je jen přenos luminance a chrominance po samostatných vodičích, z nichž každý má ještě vlastní zemní vodič. Na přijímací straně se nemusí složky oddělovat a přenášený obraz má větší kvalitu.



Obr. 4.2: S-video [10]

4.3 NTSC kódování

Jde o způsob zakódování barev do chrominance. Z RGB hodnoty pixelu lze přepočtem vypočítat tři složky:

Y – luminanci, stejné kódování pro PAL i NTSC.

U – první diferenci, luminance odečtená od modré složky.

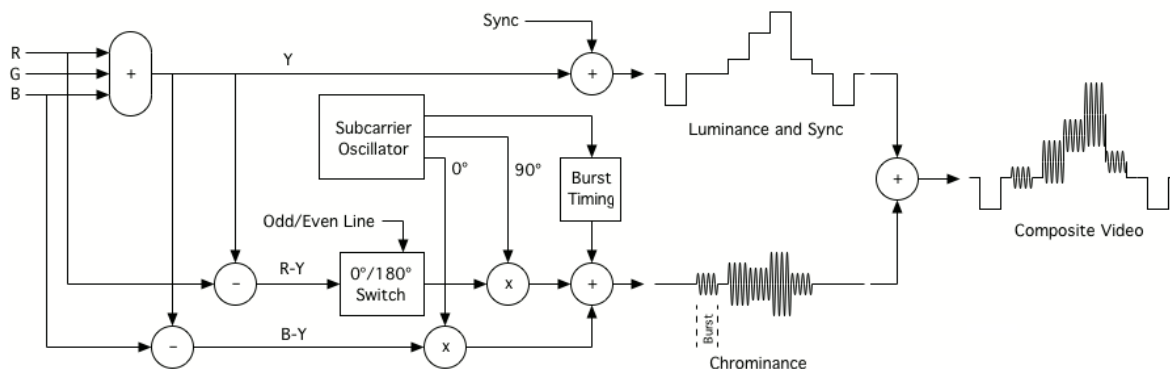
V – druhou diferenci, luminance odečtená od červené složky.

Diference U i V amplitudově modulují své nosné frekvence, které jsou vůči sobě zpožděné o 90° . Výstupní modulované signály se nakonec sečtou, čímž dojde mimo jiné k potlačení nosné frekvence, a vznikne tak signál chrominance. Amplituda chrominance odpovídá přibližně sytosti barvy a fáze odstínu barvy.

Před každým přenosem řádku se vyšle „Color burst“ referenční signál, který umožní přijímači synchronizovat se na nulovou fázi. Průběh kompozitního videa s NTSC modulací chrominance je na Obr. 4.1. Nevýhodou tohoto kódování je, že lehké narušení fáze způsobí změnu v podání barev.

4.4 PAL kódování

Kóduje se téměř stejným principem jako při NTSC, ale navíc je tu vyřešen problém s podáním barev. Aby nedocházelo ke zkreslení fáze (způsobují ho odrazy), vysílá se jedna z barvosných složek střídavě zpožděná o 180° . Nejčastěji se tak děje ob řádek. Stejně fázové zkreslení se pravděpodobně přidá k přímému i zpožděnému signálu. Přijímač poté zpětně vyrovná zpoždění signálů a porovná je proti sobě. Díky chytrému zpoždění se jednomu signálu fáze zvýšila a druhému snížila. Průměrováním signálů dostane přijímač původní správnou fázi signálu. Princip kódování PAL je vidět na následujícím obrázku.



Obr. 4.3: Kódování PAL [11]

FORMÁTY DIGITALIZOVANÉHO VIDEA

VÝSTUPEM analogově digitálních dekodérů bývají data v jednom z těchto formátů:

5.1 ITU-R BT.656

Jde o video protokol se zabudovanými synchronizačními značkami SAV a EAV v proudu 8bitových datových slov přenášených rychlostí 27 MHz. Jde tedy o zabudovanou (embedded) synchronizaci (viz. kapitola 3.2).

5.2 8-Bit 4:2:2

Protokol s diskretní synchronizací. Místo synchronizace v proudu dat se zde využívá přídatných signálů pro horizontální a vertikální synchronizaci. Jde tedy o externí synchronizaci.

U obou protokolů jsou jednotlivé pixely v řádku kódovány ve formátu YCbCr 4:2:2, což je jen jiné označení pro YUV 4:2:2 formát. Data vystupují v pořadí: 8 bitů Cb (diference U), poté 8 bitů Y0 (luma), následovaných 8 bity Cr (diference V) a následně 8 bity Y1. Ze čtyř bajtů lze díky pod vzorkování chromy (4:2:2 formát) získat dva pixely. Spoléhá se zde na to, že lidské oko není tak citlivé na barvu, jako na jas. Barevná složka je proto sdílena dvěma pixely, kdežto jasová je pro každý pixel jedinečná. Ukázka získané datové sekvence:

Cb0,1	Y0	Cr0,1	Y1	Cb2,3	Y2	Cr2,3	Y3
--------------	-----------	--------------	-----------	-------	----	-------	----

Z těchto osmi bajtů lze získat čtyři pixely:

Cb0	Y0	Cr0	Cb1	Y1	Cr1	Cb2	Y2	Cr2	Cb3	Y3	Cr3
------------	-----------	------------	-----	----	-----	------------	-----------	------------	-----	----	-----

MOTIVACE K VYTVOŘENÍ ROBOTY

MĚL jsem to štěstí, že jsem jako brigádník mohl několikrát pracovat u společnosti, která se zabývá telekomunikační infrastrukturou. A to nejen u počítače, ale dostal jsem se i do terénu. Pomáhal jsem s výstavbou technologie DSLAM a také na opravách a pokládkách metalických i optických kabelů. Viděl jsem, na jak vysoké úrovni musejí být používané přístroje, aby bylo například možné svařit dva optické kabely, nebo lokalizovat poruchu na optické trase. Dále mě zaujalo, jak obtížně a zdlouhavě se i v dnešní době kontroluje průchodnost trubek.

Na místo je vždy nutné přivést obrovský kompresor, i když navazující zafukování kabelu není na programu dne. Ve chvíli, kdy se sonda zafoukne do trubky, není možné zjistit v jaké vzdálenosti se tak stalo. Technik proto musí pomalu projít s hledačkou i velmi dlouhou trasu a hledat signál sondy. Snaží se přitom jít podle GPS přímo nad trubkou. Nepřesnost navigace může způsobit, že se od trasy odchýlí. Pokud navíc nad sondou leží v zemi třeba kámen, signál se zeslabí a sonda se tak snadno mine.

Představoval jsem si, jak pohodlné by bylo, přinést v kufříku robota a pustit ho na průzkum trubky. Nejdříve jsem hledal, zda již něco podobného neexistuje. Našel jsem roboty, kterými lze provádět inspekci kanalizací, nebo plynových potrubí, ale nic v tak malém měřítku, aby se to vešlo do trubky průměru 3 cm. Uvědomoval jsem si, že díky velmi malým rozměrům všech konstrukčních prvků i elektronických obvodů se i jednoduché problémy při stavbě takového robota značně ztíží. Byla to pro mě zajímavá výzva, která kombinuje konstrukci s elektronikou, a proto jsem si tvorbu robota zvolil jako svou diplomovou práci.

POUŽITÁ TECHNOLOGIE A SOFTWARE

Pro návrh všech konstrukčních částí robota jsem využil CAD systém **SolidWorks**. Vzhledem k tomu, že jsem počítal s tiskem dílů na 3D tiskárně, mohl jsem si dovolit nakreslit i složité tvary. Jediný díl, který není vytisknut na 3D tiskárně jsou kolečka, která byla vyrobena na CNC soustruhu. Některé díly bylo nutné koupit, což byl často největší problém. Pro takto malé rozměry je sortiment na trhu velmi omezený a ceny značně vysoké. V případě modelu pro prezentaci diplomové práce by náklady na zakázkově vyráběné strojní díly a pohon vzrostly na tisíce korun, což není reálné. Proto jsem někdy i improvizoval s tím, že kdyby se robot vyráběl sériově, použily by se miniaturizované a sofistikované díly.



Obr. 7.1: Trubot

Desky plošných spojů jsem navrhoval v programu **Altium Designer**. Tento program umožňuje udržet si přehled i ve velkých projektech s mnoha schématy. Dalším velkým plusem je vzájemné sledování změn mezi schématem a plošným spojem. Uživatel si poté může zvolit, jaké změny chce použít. Oboustranné, prokovené desky plošných spojů vyrobila firma **Pragobord**. Výrobu jednodušších desek zajistila ve velmi krátkém čase firma **Plošňáky.cz**. Desku pro komunikační vagónek jsem vyráběl sám metodou nažehlování.

Mikroprocesor jsem programoval v **STM32CubeIDE**. Jde o software, které vznikl sloučením vývojového prostředí Atollic TrueSTUDIO a generátoru inicializačních kódů pro STM32 mikroprocesory - STM32Cube. Inicializační kód je generován pomocí funkcí vrstvy HAL (Hardware Abstraction Layer). Díky tomu není tak závislý na použitém hardwaru, což jsem ocenil při časté migraci softwaru mezi vývojovými deskami Nucleo.

TĚLO ROBOTA

Hlavními požadavky, které omezily tvar těla robota byly:

Průměr trubky – prodává se mnoho typů tlustostěnných HDPE trubek. Nejčastější jsou ty s vnitřními průměry 20, 27 a 33 mm. V praxi jsem se ve většině případů setkal s 33 mm průměrem, a právě pro něj je robot určen.

Ohyb trubky – optické trasy se z mnoha důvodů plánují co nejrovnější. Někdy se ale lehkému ohybu trubky přece jen nevyhneme (např. při výstupu trubky na povrch). Z tohoto důvodu jsem počítal s nejhorším možným případem, tedy maximálním ohybem, který povoluje výrobce pro HDPE trubku 40/33 mm. Jelikož záleží i na teplotě země, znovu jsem zvolil nejhorší možný případ, tedy teplotu země 20°C. S těmito parametry vyšel maximální povolený poloměr ohybu 0,8 m.

Teplota	20°C	10°C	0°C
Poloměr ohybu [m]	0,8	1,4	2

Tab. 8.1: Ohyb trubky v závislosti na teplotě

Stoupání trubky – jsou případy, kdy trubka vede s kopce, nebo vystupuje na povrch pod značným sklonem. Stanovil jsem si, že robot by měl zvládnout stoupání pod úhlem 30°. Pro větší stoupání je řešení pohonu velmi obtížné, ale takovou situaci by šlo řešit vypuštěním robota z vyvýšeného místa směrem dolů. Zároveň by za sebou mohl táhnout splétanou rybářskou šňůru (dacron) s vysokou pevností a minimální tvarovou pamětí, kterou by šlo robotu pomoci zpět, pokud by uvízl.

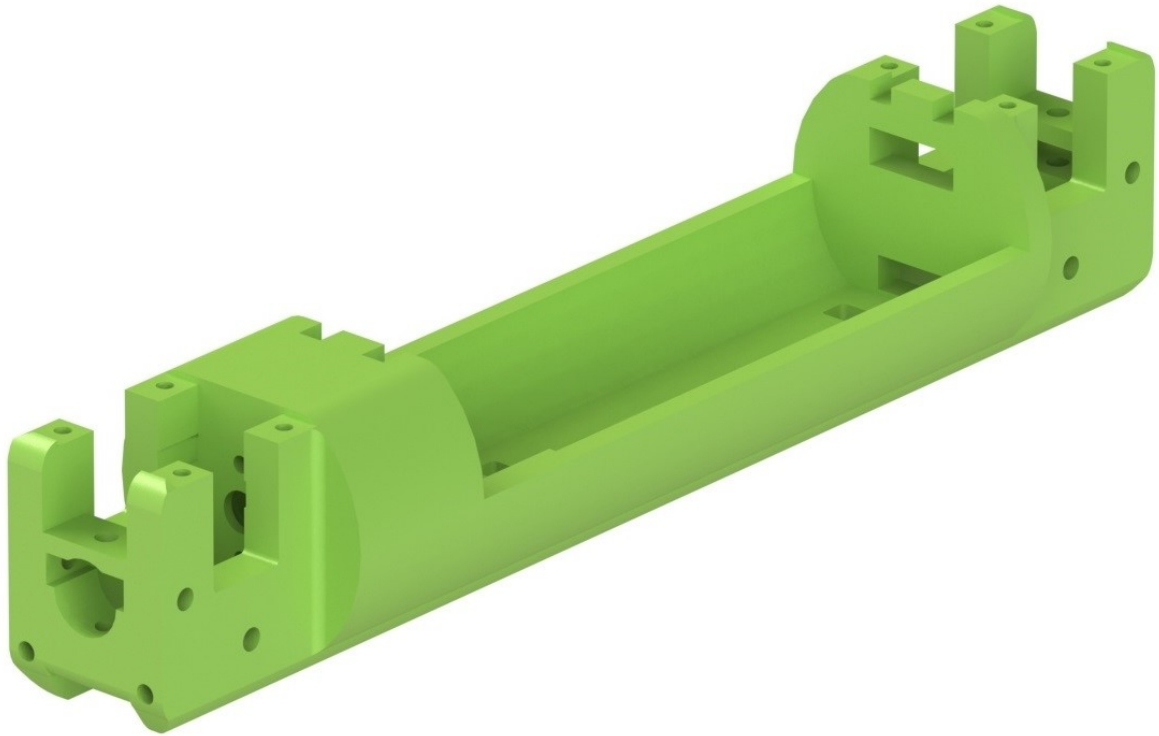
8.1 Materiál

Pro realizaci konstrukce jsem zvolil dnes velmi moderní 3D tisk. Dále bylo nutné vybrat vhodný materiál. Postupně jsem nechal vytisknout mnoho vzorků z různých materiálů a zjistil tyto vlastnosti:

PLA – přesný výtisk, snadno obrobitelný, tenké sloupky praskaly.

Karbon – velmi přesný výtisk, křehký.

ABS – velmi houževnatý, výtisk náchylnější na kroucení. Nejdůležitější je odolnost vagónku, pro finální konstrukci jsem proto vybral ABS filament.



Obr. 8.1: Střední tělo z ABS

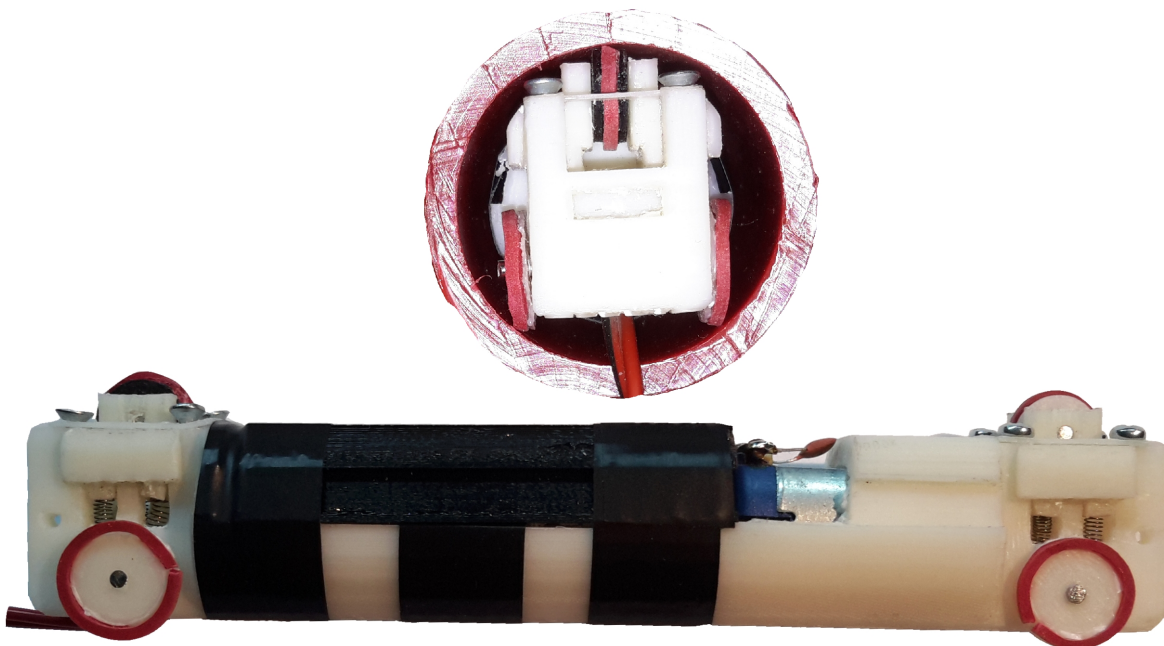
8.2 Tělo s pásy

Hned zpočátku jsem tělo robota navrhoval jako dutý válec, abych co nejvíce využil dostupný prostor uvnitř trubky. Dále jsem vymýšlel pohon robota. Nejlepší by byly dva pásy tlačené proti stěnám trubky a točící se proti sobě. Jeden navrchu robota a jeden vespod. To by ovšem zabralo velkou část dostupného prostoru a nebylo by kam dát elektroniku. Obtížné je i umístění hned dvou pohonů ve stísněném prostoru, a proto jsem tento návrh nerealizoval.

8.3 Tělo s plováky

Druhým nápadem bylo použití hnaných koleček na spodku robota a dalších přítlačných na vrchu. Tento návrh jsem fyzicky vyrobil. Přítlak jsem zajistil pomocí malých pružinek, které tlačily na plovák. Ten se pohyboval mezi vodícími sloupky. Plovákem procházela hřídel s kolečkem. Tato konstrukce s plováky měla tyto nedostatky:

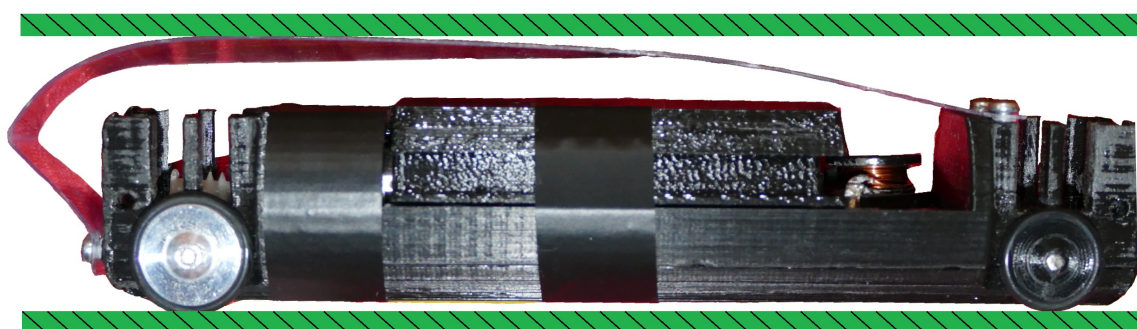
- Robot se velmi obtížně vyráběl.
- Vodící sloupky se ulamovaly.
- Složitě se nastavovala správná velikost přítlaku kolečka.
- Plovák se často ve vodících sloupkách vzpříčil, přestal pružit a robot se zasekl.



Obr. 8.2: Tělo s plováky

8.4 Tělo s přitlačnou plastovou fólií

Jde o poslední a zatím nejlepší konstrukci. Vyniká hlavně svou jednoduchostí, snadnou montáží a houževnatostí. Z rovné plastové fólie jsem vystříhl obdélník delší, než je tělo robota. Fólii jsem zdeformoval a jeden konec přišrouboval do otvorů, které zbyly po plovácích. Druhý konec jsem připevnil na přední nárazník. Jak je vidět z obrázku, fólie se v trubce zploští a tlačí proti její stěně. To způsobuje, že kolečka jsou tlačena proti druhé stěně. Dělal jsem pokusy, jaký rozměr fólie bude mít vhodný přitlak. Jako nejlepší ze všech vyzkoušených fólií se jeví ta s rozměry 20 x 155 mm a tloušťkou 0,45 mm.



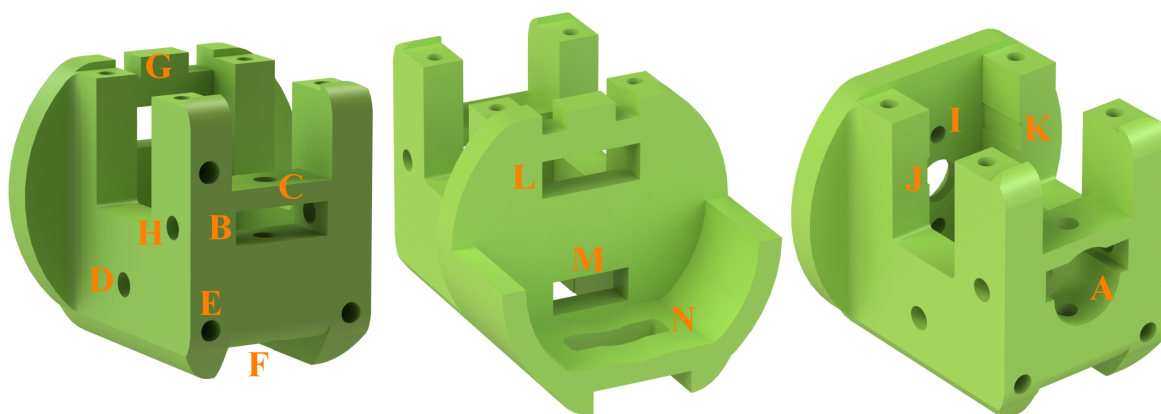
Obr. 8.3: Tělo s přitlačnou fólií

Jak pokračoval vývoj, zjistil jsem, že vychýlení fólie před vláček není vhodné. Poslední verze přitlačné fólie se proto nachází jen na vrchu těla a hledání optimálního tvaru stále pokračuje.

UNIVERZÁLNÍ VAGÓNEK

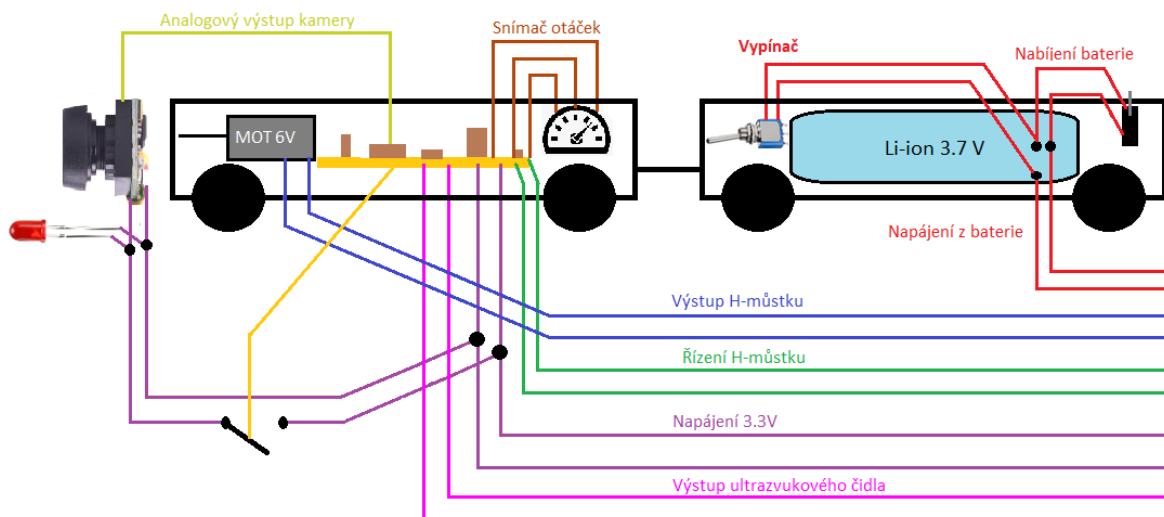
N_A těle s přitlačnou fólií jsem dále prováděl konstrukční úpravy, až vznikl univerzální segment robota, který jsem nazval vagónek. Následuje popis konstrukčních prvků.

- (A) Otvor pro nasunutí šnekového ozubeného kolečka na hřídel motoru.
- (B) Otvor pro spojku jednotlivých vláček.
- (C) Díra pro šroub držící spojku.
- (D) Otvor pro osu koleček.
- (E) Díry pro šroubky držící ultrazvukový modul nebo plastovou fólii.
- (F) Žlábek pro kabeláž mezi vláčky.
- (G) Zámek horního krytu.
- (H) Přídavná díra.
- (I) Díry pro přišroubování motoru.
- (J) Otvor pro osu motoru.
- (K) Zahloubení pro konektor typu Pin Header.
- (L) Otvor pro případnou kabeláž.
- (M) Otvor pro zahnutý Micro-USB kabel (v případě použití L-Nuclea).

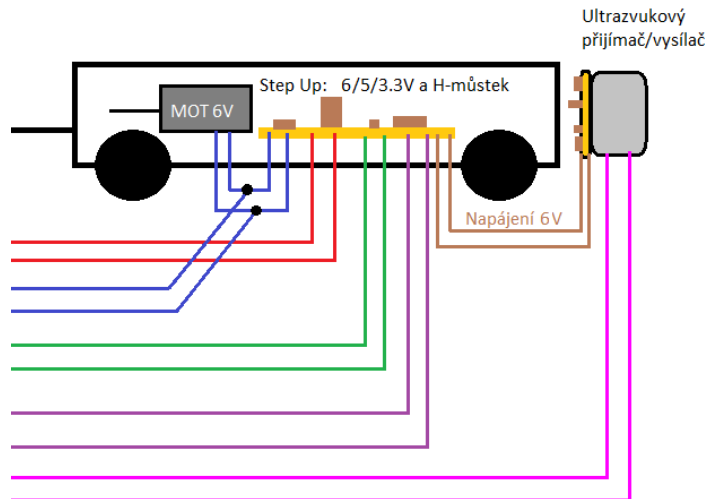


Obr. 9.1: Konstrukce těla robota

Vagónky plní různé funkce a jejich spojením teprve vzniká robot (vláček). Aby podle funkcí nebylo nutné vyrábět mnoho různých typů vagónků, zkombinoval jsem všechny konstrukční požadavky dohromady a vagónky jsou tak univerzální. Lze je použít pro převoz baterie, elektroniky nebo jako pohonnou jednotku. Jednotlivé vagónky jsou pojmenovány podle hlavní funkce, nebo podle toho, jak se projevují navenek. Blokový návrh robota je vidět na následujícím obrázku.



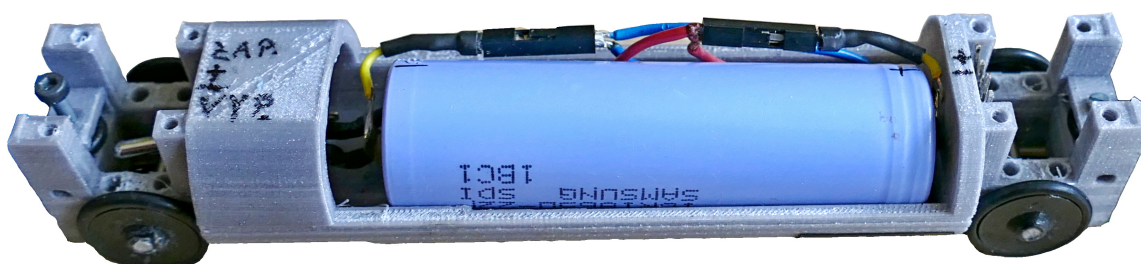
Obr. 9.2: Rozvod kabelů mezi vláčky 1



Obr. 9.3: Rozvod kabelů mezi vláčky 2

Pro „pneumatiky“ na duralových kolečkách robota jsem použil standardní těsnící „O“ kroužky. Vyzkoušel jsem několik materiálů různých tvrdostí a nakonec se rozhodl pro NBR 70 Sh, který má velmi dobrý součinitel tření mezi „pneumatikou“ a trubkou.

BATERIOVÝ VAGÓNEK



Obr. 10.1: Bateriový vagóněk

10.1 Li-ion baterie

Bateriový vagóněk veze sériově vyráběnou Li-ion baterii Samsung ICR18650-22P, která dokonale vyplňuje vnitřní dutinu těla. Parametry baterie jsou:

Napětí: 3.7 V

Průměr: 18.4 mm

Kapacita: 2200 mAh

Max. proud: 10 A

Délka: 65 mm

Min. napětí 2.75 V

Z baterie jsou vyvedeny vodiče do komunikačního vagónku a na nabíjecí konektor přístupný obsluze. Nabíjení lze provádět konstantním napětím 4.2 ± 0.05 V nebo konstantním proudem (pomale 1075 mA, rychle 2150 mA). Také je zde zabudován vypínač, kterým lze robota kompletně odpojit od zdroje energie.

10.2 Hallův senzor otáček

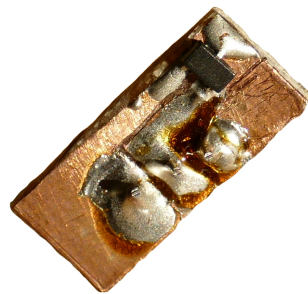
10.2.1 Hardware

Robot má do jednoho hliníkového kolečka zalisovaný silný neodymový magnet. Poblíž jedné strany kolečka je upevněn omnipolární Hallův senzor, který nerozlišuje mezi jižním a severním pólem. Kdykoliv je vystaven magnetickému poli generuje napětí. Při pohybu



Obr. 10.2: Li-ion baterie 18650

vagónku dochází k protínání senzoru magnetickým polem a ten díky přídatným vnitřním obvodům generuje logický napěťový výstup.



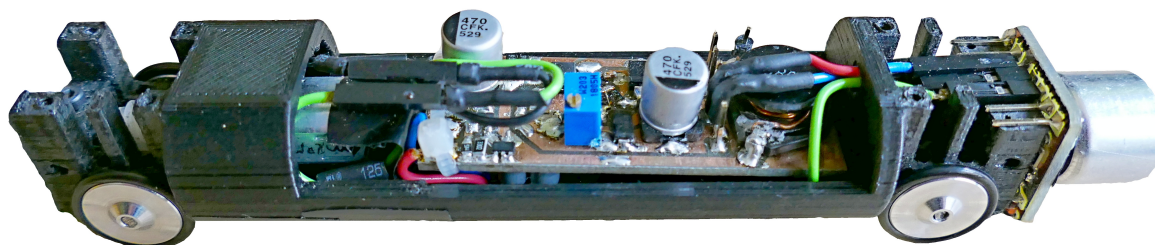
Obr. 10.3: Hallův senzor

10.2.2 Software MCU

Senzor se nachází na vagónku bez pohonu. Ve chvíli, kdy se robot zasekne, případně narazí na překážku zabráňující pohybu vpřed, dojde k protáčení hnacích koleček. Program tak má k dispozici informaci, že motory jsou spuštěné, ale Hallův senzor nedává impulsy. Na to po určité době reaguje vypnutím motorů a indikací zaseknutí.

KOMUNIKAČNÍ VAGÓNEK

TENTO vagónek v sobě kombinuje hned čtyři funkce – změnu napěťových úrovní, řízení motorů, pohon robota a komunikaci ultrazvukem. Popis jednotlivých modulů následuje.



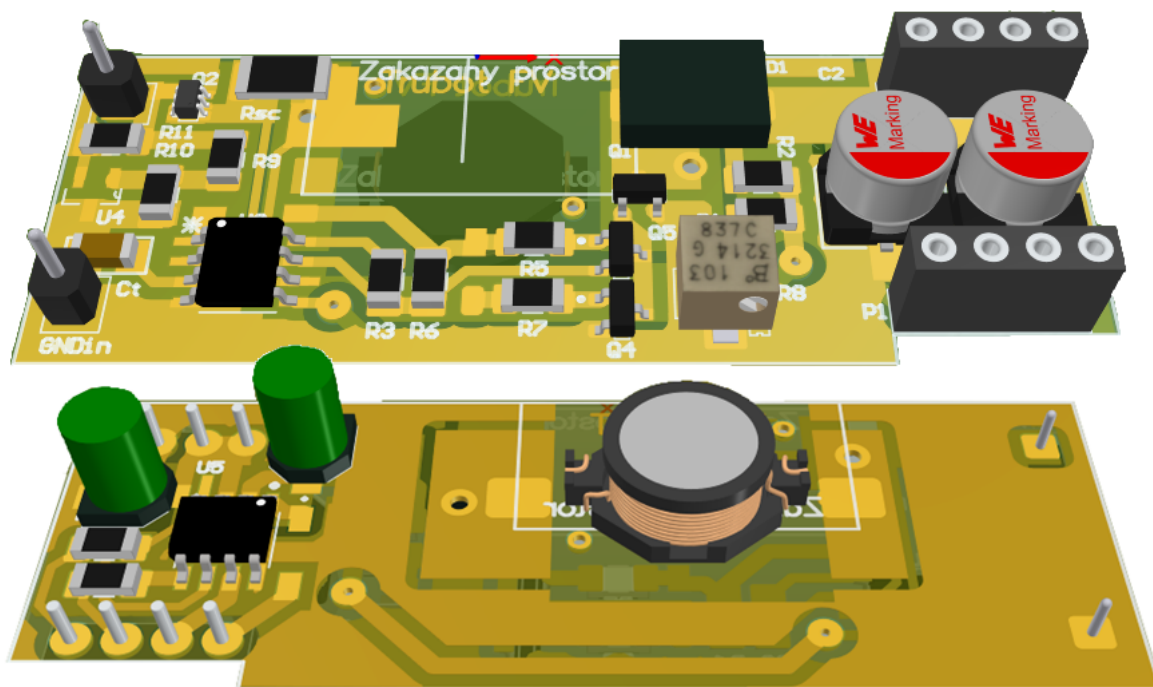
Obr. 11.1: Komunikační vagónek

11.1 DC/DC měnič

Protože Li-ion baterie poskytuje pouze 3.7 V, bylo třeba navrhnout Step-Up měnič na 12 V (po zkušenostech změněno na 6V, viz kapitola Lineární stabilizátory 11.5.2). Maximální proud, který by měl být měnič schopen poskytnout, jsem vzhledem k odběru motorů při zátěži (200 mA) stanovil na 500 mA. První volbou byl integrovaný obvod MC34063. K němu jsem našel doporučené zapojení, ze kterého jsem použil hodnotu cívky a filtračního kondenzátoru. Aby bylo možné odebírat větší proud, doplnil jsem obvod o výkonový MOSFET tranzistor a jeho spínání zajistil dalšími dvěma bipolárními tranzistory. Schéma v příloze A.

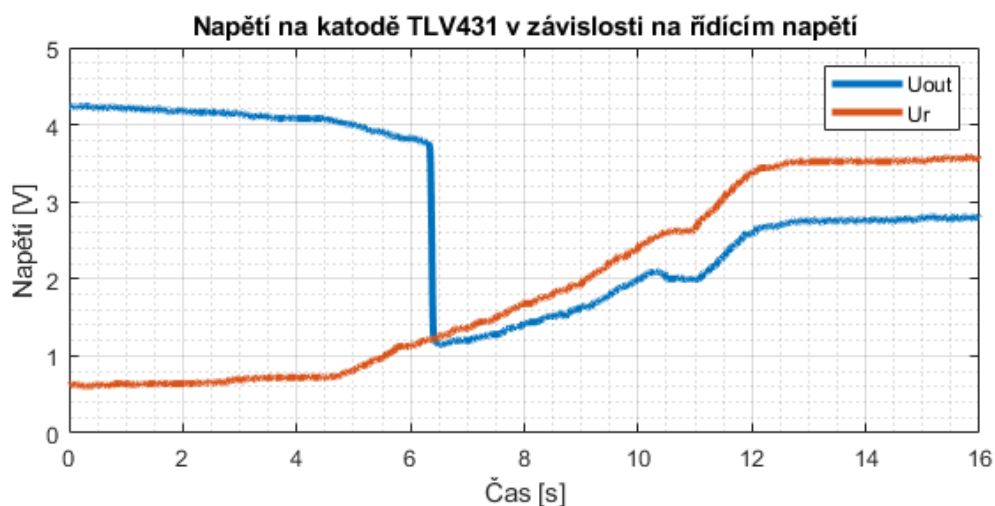
Toto zapojení jsem dále na vstupu doplnil o obvod s nastavitelnou Zenerovo diodou TLV431, který měl v případě pod vybití odpojit baterii od zátěže. Princip by měl být následující. Pokud napětí na vstupu poklesne pod nebezpečnou hodnotu, TLV431 se uzavře a na katodě se objeví kladné napětí, což rozeptne tranzistor Q2 a zamezí napájení DC/DC měniče. Tranzistor jsem volil s co nejmenším $R_{ds(on)}$ a malým potřebným napětím pro sepnutí (AO6409).

Zapojení jsem věřil a použil ho i ve vylepšeném měniči s PAM2423. Jednou se mi při testování řídicí desky baterie vybila na kritickou hodnotu a obvod nezareagoval. Jak se baterie dále vybíjela, napětí za měničem se zvlnilo a procesor na to reagoval náhodnými restarty. Na následujícím grafu je charakteristika obvodu TLV431 při konstantním napájení 5V. Je zde vidět prudký pokles napětí na katodě, když napětí na řídicím vstupu (U_{ref}) dosáhne hodnoty 1.25 V. Tohoto skoku jsem hodlal využít pro rozeptnutí P-MOS tranzistoru.



Obr. 11.2: Deska s měničem MC34063

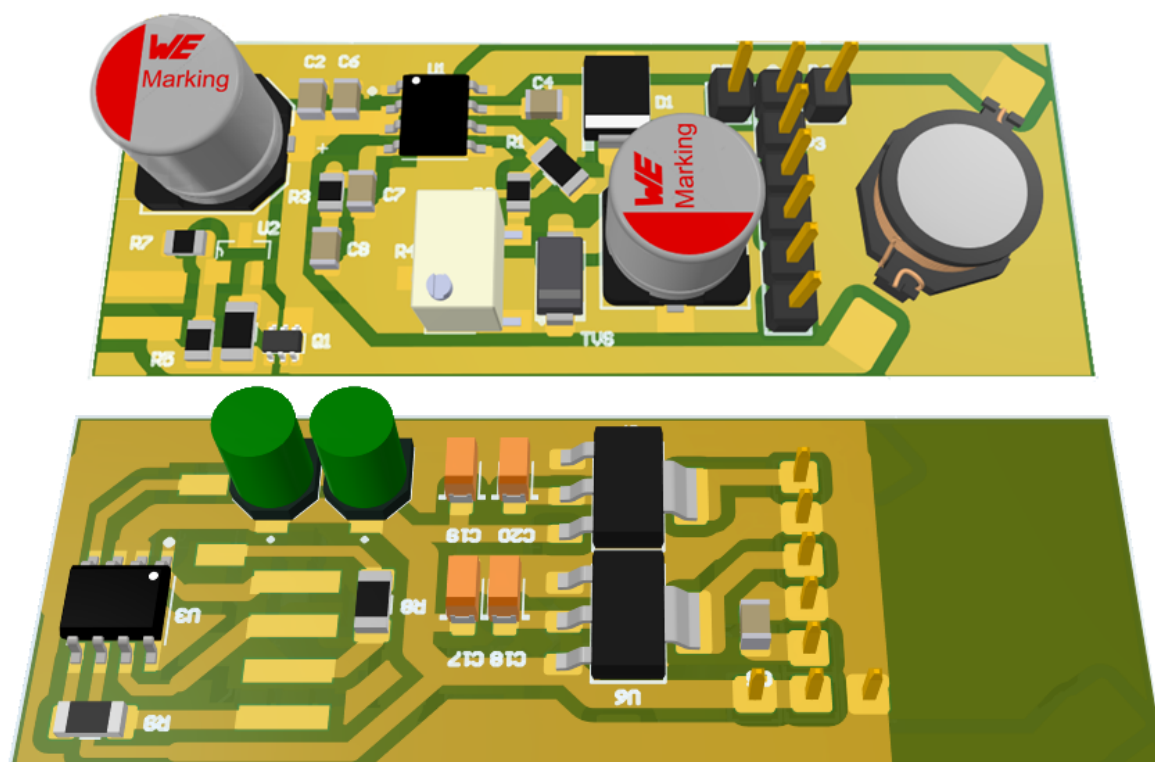
Při proměnném napájecím napětí se ale průběh skoku zploštuje, což není pro logické řízení tranzistoru vhodné. Dokonce se může stát, že při plně nabitě baterii bude na katodě dostatečné napětí, aby tranzistor vypnul. Pro budoucí návrhy bych proto doporučoval použít klasický komparátor s operačními zesilovači.



Obr. 11.3: Charakteristika TLV431

Samotný měnič jsem simuloval v LTspicu při odběru 500 mA a nebyl jsem s ním spokojený. Obvod poskytoval vysoce zvlněné výstupní napětí, které se pohybovalo v mezích <11,7 až 12,4 V>. Problémem byla i nízká účinnost, která se i díky úbytku napětí na MOSFET tranzistoru pohybovala kolem 67%. Obvod jsem přesto fyzicky vyrobil, abych teorii ověřil. Simulace se potvrdila a bylo tak třeba hledat jiné řešení.

Druhým a úspěšným pokusem byl DC/DC měnič založený na obvodu PAM2423, který obsahuje interní MOSFET tranzistor (viz. schéma v příloze B). Obvod pracuje na pevně dané frekvenci 520 kHz a podporuje soft-start. Z předchozího měniče jsem bohužel znovu využil ochranný obvod proti podvybití baterie s TLV431. Jinak je ale měnič mnohem účinnější, což dokazuje tabulka 11.1.



Obr. 11.4: Deska s měničem PAM2423

Zátěž [ohm]	U _{bat} [V]	I _{bat} [A]	U _{out} [V]	P _{out} [W]	P _{in} [W]	Účinnost [%]
2,4	3,55	0,86	2,69	3,02	3,05	98,76
4,7	3,63	0,761	3,33	2,36	2,76	85,41
5,7	3,65	0,763	3,6	2,27	2,78	81,64
47,4	3,75	0,641	9,06	1,73	2,40	72,04
82	3,76	0,584	12,13	1,79	2,20	81,72
118,2	3,85	0,386	12,13	1,24	1,49	83,76

Tab. 11.1: Účinnost měniče s obvodem PAM2423

11.2 Lineární stabilizátory

Napětí 12 V (změněno na 6 V) je určené pro motory a ultrazvukové čidlo. Mikroprocesor potřebuje nižší napětí, konkrétně 3.3 V. Video dekodér TVP5150 pracuje s napětím 1.8V. Do budoucna bylo navíc počítáno s možným připojením 5 V logiky. Pro dosažení těchto napěťových úrovní jsou na desku osazeny dva lineární stabilizátory:

AZ1117H-3.3TRE1 – LDO; lineární; nenastavitelný; 3,3V; 1 A

NCP1117ST50T3G – LDO; lineární; nenastavitelný; 5 V; 1 A

Pro stabilizaci jsou na vstup i výstup stabilizátorů připojeny tantalové kondenzátory. Ty oproti elektrolytickým nabízejí velkou kapacitu už od malých velikostí. Také jsou odolnější vyšším teplotám a mají dobrou frekvenční odezvu. Zapojení stabilizátorů si lze prohlédnout ve spodní části schématu přílohy B. Součástky se nacházejí v pravé spodní části plošného spoje viz. obrázek 11.4.

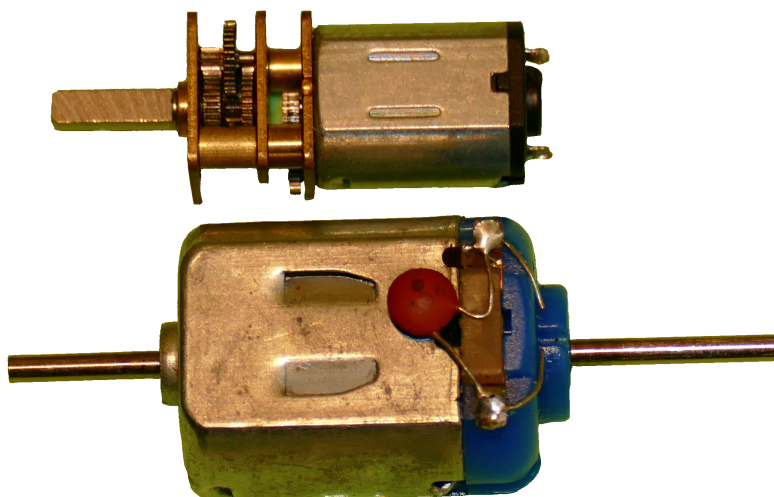
Snižovat napětí z 12V na 3.3V lineárním stabilizátorem je možné jen při odběru nízkých proudů. Ve chvíli, kdy jsem začal řešit kamerový modul, bylo nutné zvýšit velmi výrazně frekvenci procesoru, abych dokázal přijmout rychlý datový tok z video dekodéru. Tím se zvedl proudový odběr procesoru ze 77 mA na 180 mA. K tomu se ještě přidalo 65 mA do kamery a lineární stabilizátor shořel. Řešení:

- Změnou rezistoru ve zpětnovazebním děliči kleslo napětí za DC/DC měničem na 6V. Vše ostatní mohlo zůstat zachované, jen jsem vyměnil používané motory.
- V běžném provozu jsou hodiny procesoru taktované na 16 MHz. Jen při focení a ukládání na SD kartu dojde k chvilkovému přetaktování na vysokou frekvenci 216 MHz.
- Do napájecího vodiče kamery jsem vřadil N-MOS tranzistor, který kameru zapne jen pokud je to třeba. K ovládání tranzistoru používám jeden z pinů procesoru, který jsem při návrhu desky vyvedl na konektor pro možné budoucí použití.

11.3 Stejnoseměrný motor

Nachází se v dutině, kde je šrouby připevněn k plastovému vagónku. Nejdříve jsem používal typizovaný malý stejnosměrný motorek. Ten se prodává s krátkou hřídelí, kterou bylo nutné vyklepnout a nahradit novou, delší hřídelí. Motorek měl velmi vysoké otáčky. Zpomalení a zvětšení kroutícího momentu jsem dosáhl ozubeným převodem – šnek připevněný na hřídeli motoru dosedá na ozubené kolečko připevněné k ose kol. I s touto úpravou se kolečka otáčela příliš rychle a při větším zatížení se robot zastavil.

Koupit malá ozubená kolečka v přesně požadovaných velikostech je velmi obtížné. Proto jsem se vydal cestou snížení otáček motoru pomocí zabudované kovové převodovky. Vyzkoušel jsem mnoho motorů s různými výstupními otáčkami. Pro zvolenou konfiguraci šnekového převodu má nejlepší poměr mezi silou a rychlostí 6 V motor GA12-N20 s 1000 otáčkami za minutu. Robot se s tímto motorem pohybuje trubkou rychlostí přibližně 3.5 m/min. Z předchozích údajů je vidět, že jsem narazil na fyzikální hranici. Robot by potřeboval výkonnější motor, aby mohl jet rychleji, ale takto výkonný motor v rozumné cenové hladině, se do trubky nevejde.



Obr. 11.5: Použitelné stejnosměrné motory

11.4 H-můstek

11.4.1 Hardware

Posledním samostatným obvodem na desce s měničem je H-můstek ZXBM5210 pracující s napětími od 3 V do 18 V. K můstku lze připojit hned dva stejnosměrné motory. Uvnitř čipu je integrovaná přepěťová, tepelná i nadproudová ochrana. Pomocí logických hodnot na pinech FWD a REV lze zapínat módy:

Forward – rotace motoru vpřed

Reverse – rotace motoru vzad

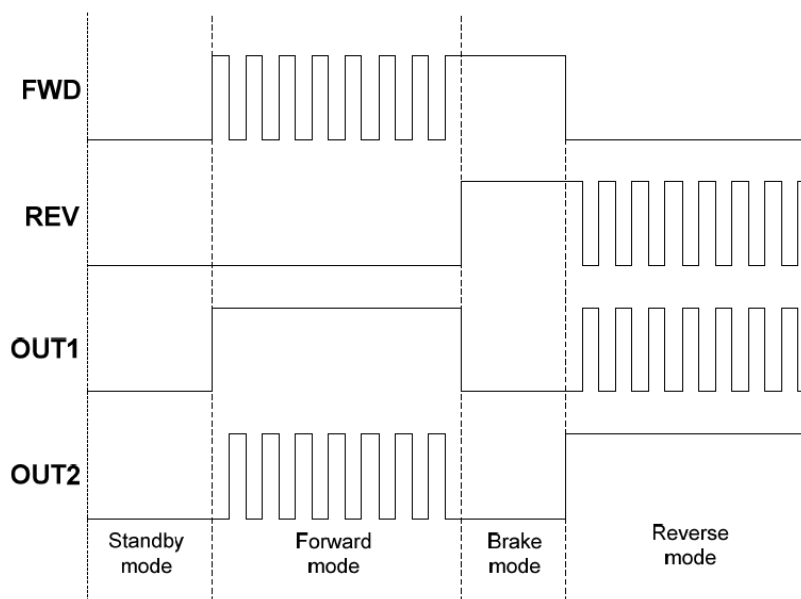
Brake – rychlé zastavení motoru

Standby – vypnutí H-můstku, úspora energie

Rychlost otáček motoru lze řídit buď změnami napětí na pinu Uref, nebo pomocí PWM modulace (8 kHz až 100 kHz) na jednom z pinů FWD, REV. Vzhledem k tomu, že mikroprocesory často obsahují podporu pro PWM řízení, zvolil jsem druhý způsob. Zapojení lze vidět na schématu v příloze C. Fyzická podoba je v levé spodní části obrázku 11.4.

11.4.2 Software MCU

Na řídicí piny H-můstku je připojen první a druhý kanál pokročilého 16bitového vzestupného čítače TIM1. Oba kanály jsou nastaveny na PWM mód 1. To znamená, že přiřazený výstupní pin je aktivní, dokud se nedosáhne hodnoty v porovnávacím registru. Jinak je neaktivní. Dále je aktivnímu stavu přiřazena vysoká a neaktivnímu stavu nízká logická hodnota. Časovač i porovnávací registr má zapnutou funkci „auto-reload preload“ Lze tak měnit hodnoty registrů za běhu čítače, aniž by to mělo okamžitý vliv na výstup. Hodnoty budou zapsány do pracovních registrů až po přetečení a zachová se tak synchronizace s novým cyklem. Hodnoty registrů jsou zvoleny tak, aby čítač generoval obdélníkový signál, jehož duty cyklus lze nastavit změnou hodnoty v porovnávacím registru.



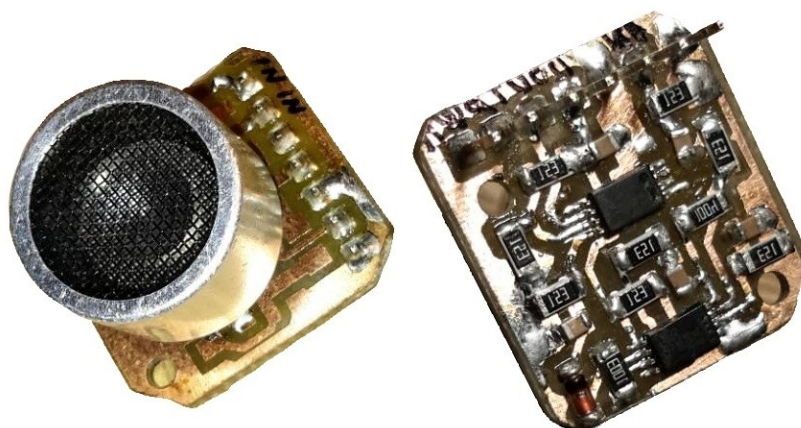
Obr. 11.6: Řízení H můstku [12]

11.5 Ultrazvukové čidlo

11.5.1 Hardware

Robot by měl zajet do trubky, nalézt poruchu a s informacemi se vrátit zpět. To vše bez zásahu operátora. Pokud by ale bylo nutné náhle přerušit průzkum a opustit stanoviště, muselo by se čekat na dokončení inspekční cesty. Abych tomu zabránil, rozhodl jsem se vybavit robota modulem pro příjem povelů. Inspiroval jsem se netopýry, kteří pro lokalizaci kořisti a komunikaci využívají ultrazvuk. Tedy zvuk o frekvenci vyšší než 20 kHz. Při využití ultrazvuku se musí počítat s tím, že jak se zvuk šíří vzduchem, dochází ke tření molekul vzduchu a akustický tlak se postupně snižuje. Tato vlastnost se ale zmírní, pokud se bude zvuk šířit jedním směrem v uzavřeném prostoru. Dále platí, že čím kratší je vlnová délka (vyšší frekvence), tím je větší pohltivost [13].

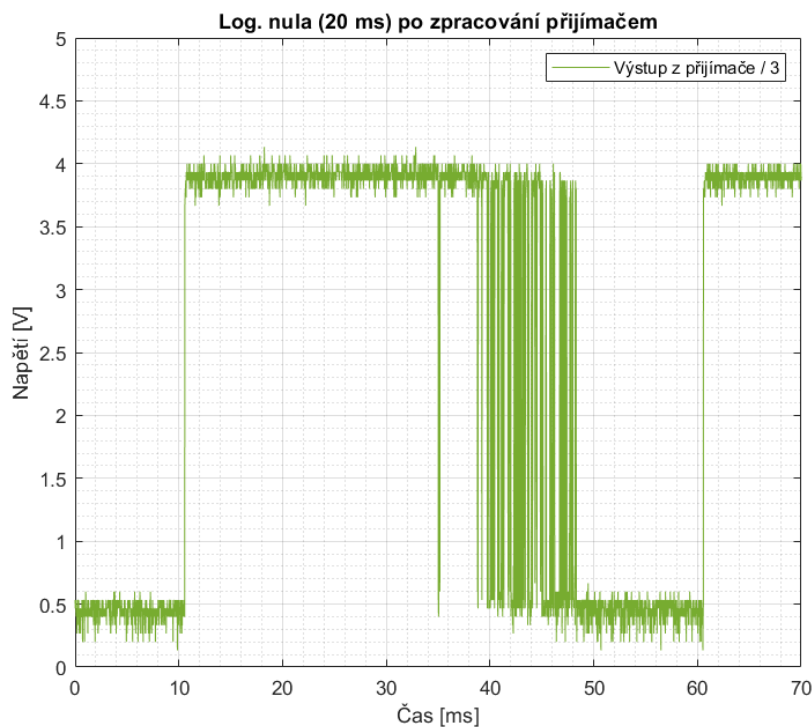
Ultrazvuková čidla dostupná v obchodech pracují často na frekvencích 25 nebo 40 kHz. Z výše zmíněných důvodů by bylo pro komunikaci lepší použít nižší frekvenci. Pro přijímací modul jsem ale vybral 40 kHz čidlo, jelikož jsem právě několik takových vlastnil.



Obr. 11.7: Ultrazvukové čidlo

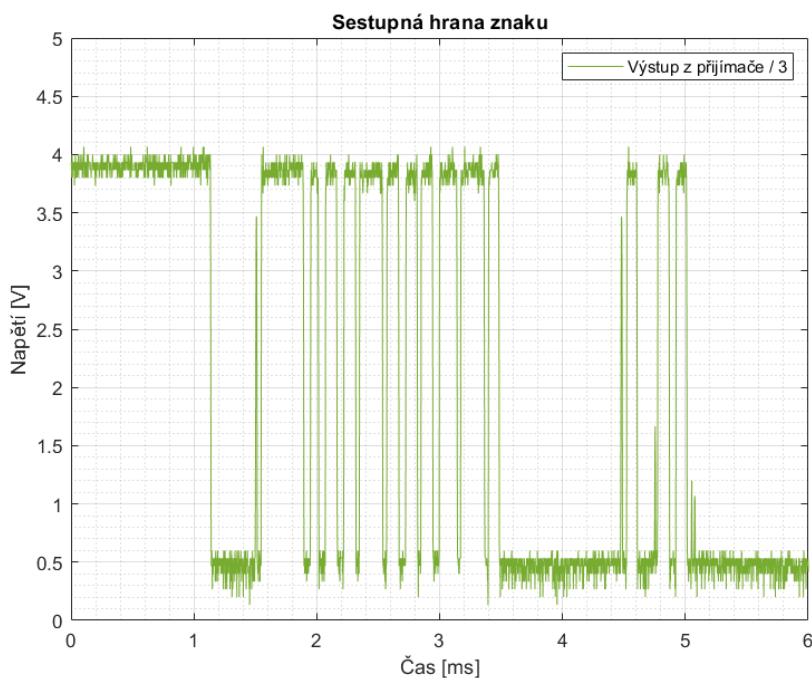
Příjem: Obvodové řešení přijímače je na schématu v příloze E. Vyšel jsem z návrhu pro měřič vzdálenosti s ultrazvukovým čidlem [14] a přizpůsobil ho své potřebě. Přes vazební kondenzátory prochází střídavý ultrazvukový signál na vstup invertujícího zesilovače. Aby jednostranně napájený zesilovač zesílil obě polarity vstupního napětí, je na vstupech zesilovače nastavena stejnosměrná složka 3 V pomocí rezistorů R16 a R19. Signál se zde zesílí cca. 67 x a pokračuje do druhého zesilovače. Zde znovu dojde k cca sedminásobnému zesílení. Kaskádní řazení OZ a postupné zesílení umožňuje zesilovat širší pásmo signálů, a navíc nedojde k inverzi vstupu. Schottkyho dioda D2 spolu s kondenzátorem C14 tvoří kvazi špičkový detektor. Napětí, které z něj vystupuje, je obálkou vysokofrekvenčního vstupního signálu. Následuje OZ ve funkci komparátoru. Jeho rozhodovací úroveň je nastavena na 3 V. Výstupem celého zapojení je v ideálním případě dvouhodnotový obdélníkový signál. Jeho hodnota odpovídá tomu, zda na vstup právě přichází ultrazvukový signál (log 1), nebo je před čidlem ticho (log 0).

Reálně změřené hodnoty se od ideálního průběhu trochu liší. Po dobu příjmu ultrazvukového signálu se občas objevují falešné propady do log. nuly, přestože vysílač generuje signál. Trvají zanedbatelný čas, ale je třeba s nimi při vyhodnocení počítat (viz. Software ultrazvuku 11.5.2). Méně často se objeví falešný výskok do log. jedničky, přestože nepřichází žádný ultrazvukový signál. Naopak téměř vždy jsou v signálu výskoky do log. jedničky, které následují ihned po ukončení vysílání znaku. Tyto jevy si vysvětlují odrazy vln a jejich následným skládáním při průletu trubkou nebo rušivými zvuky okolí. Na obrázku níže je průběh signálů z osciloskopu. Komunikaci jsem testoval v šestimetrové zahnuté trubce s jednou spojkou uprostřed. Přestože byl vyslán jen 20 ms ultrazvukový balík signálu, přijímači chvíli trvá (9 ms), než překlopí zpět do klidového stavu. Z něho je ale ještě několikrát aktivován díky ozvěně, což protáhne znak o dalších cca 10 ms.



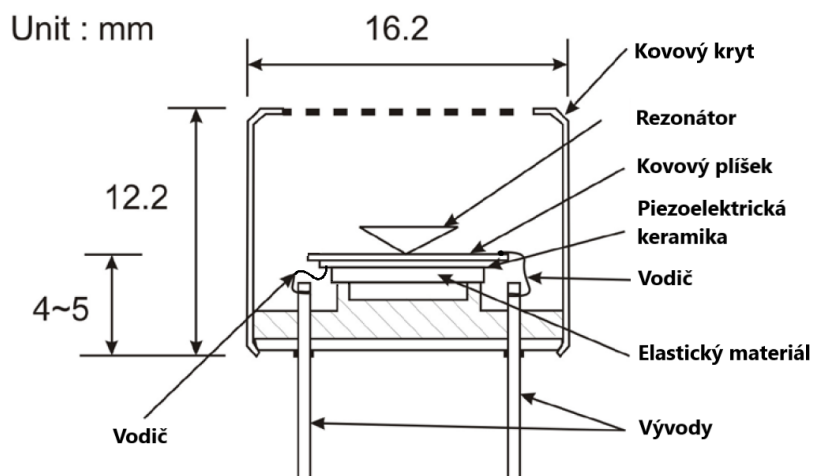
Obr. 11.8: Ozvěna ultrazvuku

Na obrázku 11.9 je zvětšená oblast, kde dochází k registraci ozvěn. Delší časová konstanta RC článku by ozvěny pohltila, ale zase by trvalo déle, než by přijímač přešel do klidového stavu. Proto jsem volil kompromis a další zpracování výstupního signálu nechávám na softwaru.



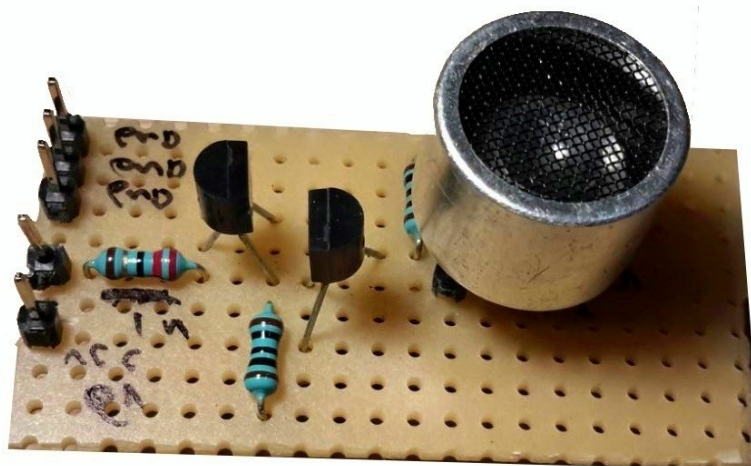
Obr. 11.9: Sestupná hrana znaku

Vysílání: Uvnitř ultrazvukového měniče se nachází piezoelektrická keramika. Přivedením napětí na její povrch dojde k polarizování vnitřních elektrických dipólů do jednoho směru. Ty se navíc podle polarity přivedeného napětí roztáhnou nebo smrští. Z venku to registrujeme jako změnu tloušťky keramiky, případně zvuk. Pokud frekvence budícího napětí odpovídá rezonanční frekvenci keramiky, vznikne mechanická rezonance a přenos energie je neúčinnější. Pro vybraný ultrazvukový měnič je to frekvence 40 KHz [15].



Obr. 11.10: Konstrukce ultrazvukového senzoru [15]

Vysílač je tvořen dvěma MOSFET tranzistory provozovanými ve spínacím režimu. Vstupní tranzistor Q2 je N-MOSFET s nízkým spínacím napětím. Díky tomu jde sepnout přímo výstupním pinem mikroprocesoru. Aby nedošlo k obrácení spínací logiky, je fázově obrácený signál na drainu veden do gate druhého P-MOSFET tranzistoru, který spíná napětí do ultrazvukového vysílače. Rezistory mají nižší hodnoty, aby nabíjení kapacit gate bylo pro frekvenci 40 kHz dostatečně rychlé. Zapojení jsem realizoval na univerzálním plošném spoji s využitím THT součástek. Spolu s vývojovou destičkou Nucleo tvoří vysílačku, kterou může operátor vzdáleně udělovat pokyny robotovi. Schéma zapojení je v příloze D.



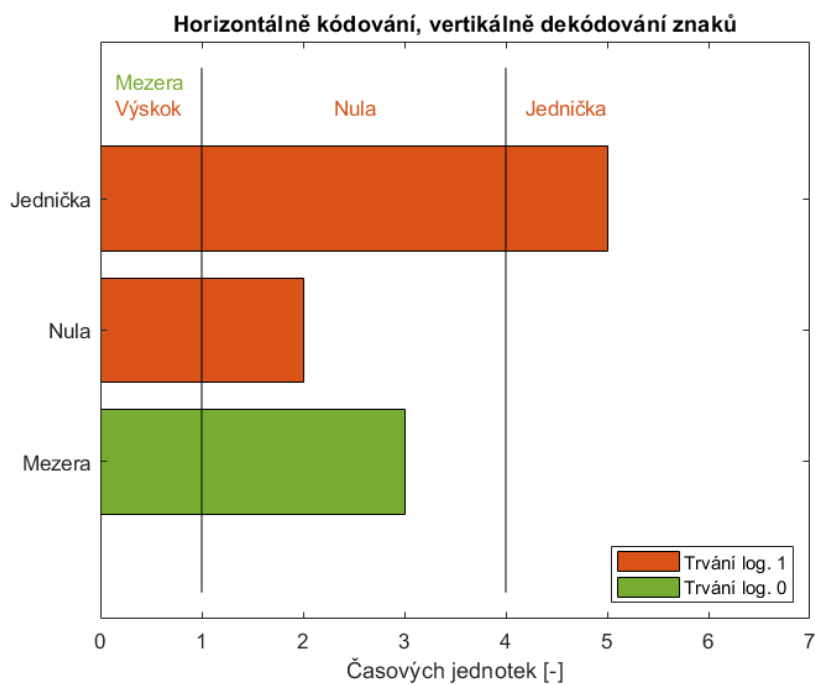
Obr. 11.11: Ultrazvukový vysílač

11.5.2 Software MCU

Zpočátku bylo zřejmé, že kvůli odrazům a dozvukům ultrazvukového signálu v trubce bude nutné oddělit odesílané bity delšími pauzami, jinak splynou. Dále jsem chtěl, aby se přijímač synchronizoval s každým poslaným bitem, tedy i tehdy, když se přenáší logická nula. Na základě těchto požadavků jsem znaky zakódoval. Logickou jedničku i nulu přenáším jako shluk ultrazvukového signálu, ale různé délky trvání. Mezi znaky je vždy vložena mezera, kdy je ultrazvuk ztišen.

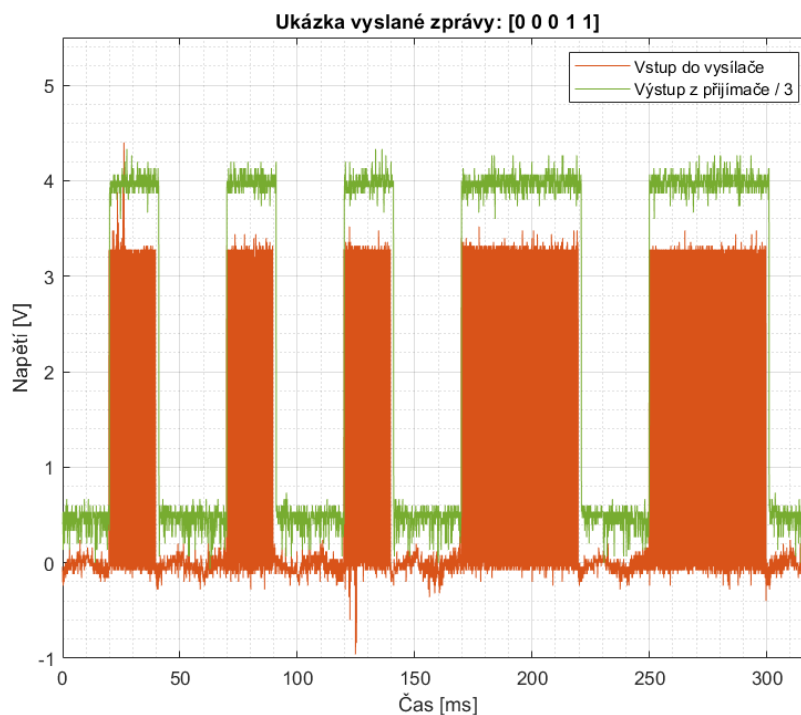
Vysvětlení grafu pro znak nula: Nulu vysílám jako balík ultrazvukového signálu po dobu 2 časových jednotek. Následuje příjem. Pokud po dobu jedné, až dvou časových jednotek přijímám ultrazvukový signál, přišel znak nula.

Vysílání: V grafu si lze všimnout časových rezerv mezi jednotlivými znaky při vysílání. Díky nim je toto kódování odolné vůči rozšíření balíku ultrazvukového signálu, způsobeného odrazem při putování trubkou. Konkrétně je vysílání implementováno takto – metodou *ultrazvuk_priprav_vysilani* podle zprávy vyplním pole *easy_zmen_vysilani*. Pole tak obsahuje časy, za které má dojít ke změně vysílaného znaku. V mainu tyto vysílací časy kontroly proti systémovému časovači a povolují či zakazují generování ultrazvukového signálu pomocí PWM modulace. Jde o neblokující způsob vysílání.



Obr. 11.12: Graf kódování

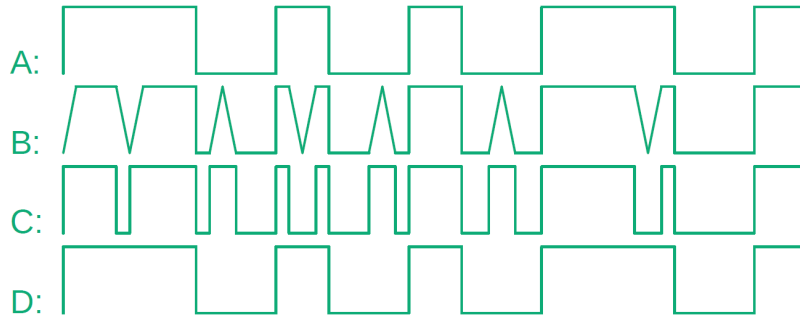
Pokud v průběhu vysílání přijde přerušení, dojde tím k malé časové chybě ve vysílání aktuálního znaku. Protože je nutné vysílat každý znak po velmi dlouhou dobu (desítky milisekund), tato časová výchylka je zanedbatelná. Obrázek níže zobrazuje, jak může vypadat ultrazvukem vyslaná zpráva.



Obr. 11.13: Ukázka vyslané zprávy

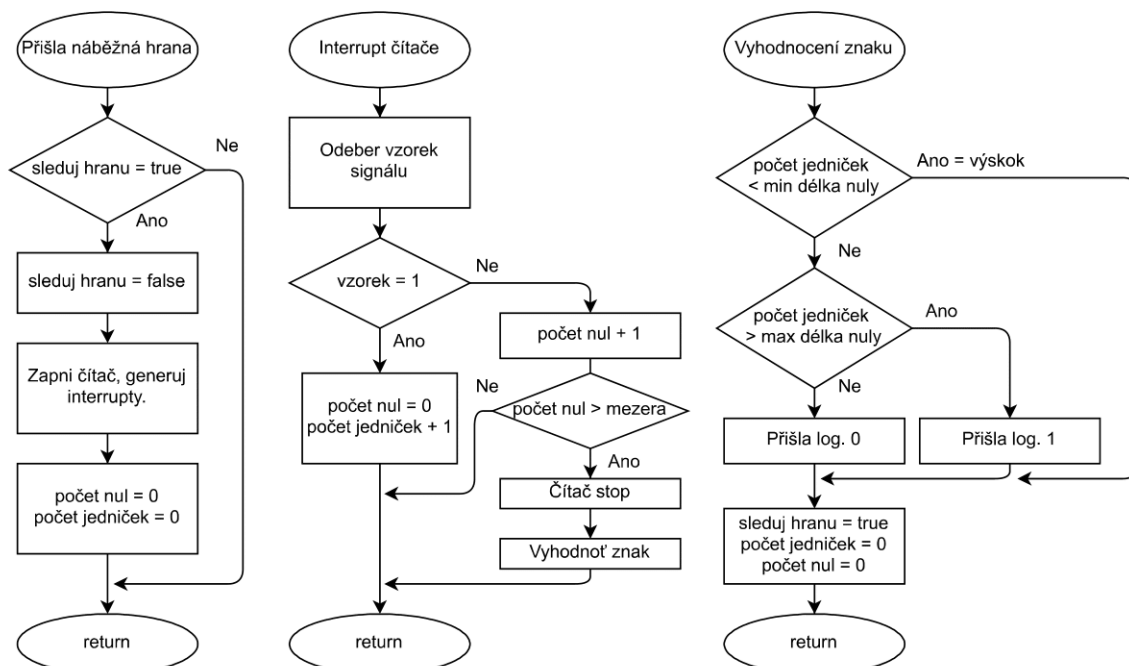
Příjem: Jde o složitější úlohu, protože je nutné provést softwarovou filtraci vstupního signálu. Na obrázku 11.14 je vidět obálka ultrazvukového signálu z vysílače (A). Po cestě trubkou dojde k poškození signálu (B). Obvod s operačními zesilovači vstupní signál zesílí a diskretizuje (C). Po SW zpracování bych ideálně chtěl znovu dostat signál (D), který je kopií signálu A, pouze časově zpožděný (v obrázku nenaznačeno).

Příjem ultrazvukového signálu



Obr. 11.14: Úprava přijatého signálu

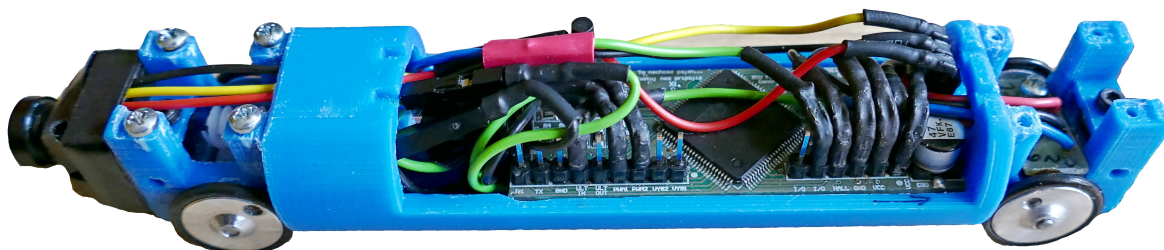
Mikroprocesor proto čeká na náběžnou hranu a ve chvíli, kdy přijde, vyvolá se obsluha přerušování (*HAL_GPIO_EXTI_Callback*). V té se spustí čítač, který generuje přerušování. Při každém přerušování od čítače se pořídí vzorek vstupního signálu. Počítá se, kolik vzorků bylo rovno log. jedničce, dokud se nenavzorkuje xkrát log. nula. To značí příchod mezery a mikroprocesor přejde na vyhodnocení znaku. Rozhoduje se přitom na základě počtu vzorků rovných log. jedničce (viz obrázek 11.12). Po vyhodnocení se znovu čeká na příchod náběžné hrany a cyklus se opakuje. Následuje vývojový diagram této SW filtrace.



Obr. 11.15: Vývojový diagram

INSPEKČNÍ VAGÓNEK

VŠECHNY předcházející části robota jsou podporou pro tento inspekční vagónek. Stejně jako zadní komunikační vagónek disponuje stejnosměrným motorem. Nachází se tu výkonný řídicí mikroprocesor STM32F767VIT6, video dekodér TVP5150, konektor na mikro SD kartu, stabilizátor napětí a další součástky. V čelní části má robot umístěnou malou analogovou kameru. Všechny výše zmíněné součástky a z nich vytvořené obvody (kromě kamery) se musí vejít na destičku o velikosti zhruba 6 x 2 centimetry.



Obr. 12.1: Inspekční vagónek

12.1 Kamera

Existuje mnoho hotových kamerových modulů, které mají vyřešený návrh plošného spoje a často i vzory řídicího programu. Výstupem je rovnou digitální signál, který lze snadno zpracovat DCMÍ rozhraním. Jejich velikost bohužel neumožňuje nasazení ve stísněných prostorech. Proto jsem přemýšlel o použití špionážní kamery (např. v propisce, hrací kostce), která by se na robota pouze připevnila a po průzkumu trubky by se z ní stáhla data. Nepodařilo se mi však najít tak malou kameru, aby se i se svým příslušenstvím (baterií, řídicím obvodem) do trubky vešla. Zbývala tedy jediná možnost. Koupit minikameru s analogovým výstupem a analogově digitální zpracování si udělat sám na řídicí desce.

Pro svůj projekt jsem vybral minikameru s těmito parametry:

Rozlišení - 600 TVL (počet rozlišitelných linek/řádek obrazu).

Minimální osvětlení - 0.5 lux (intenzita osvětlení za jasné noci s úplňkem).

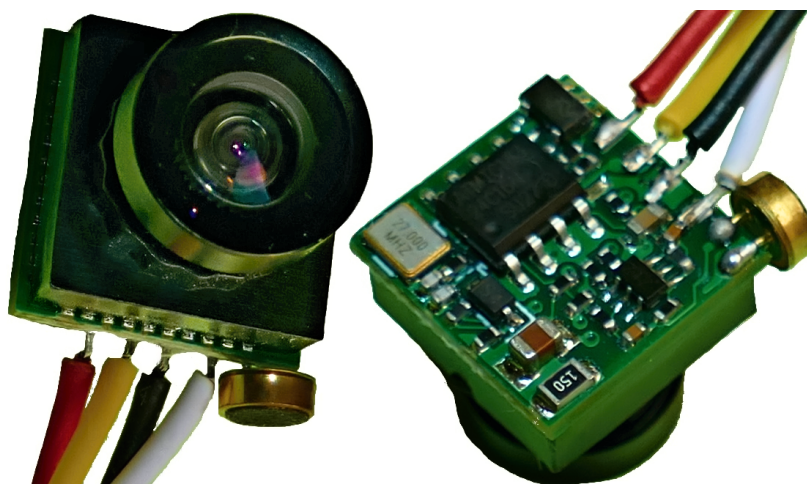
Výstupní formát - PAL, kompozitní (1.0V_{pp} 75Ω)

Objektiv - širokoúhlý (170 stupňů).

Obrazový snímač - CMOS 1/3 palce

Rozměry - 20 x 20 x 13 mm

Napájení - 3.3 V až 5 V, 70 mA



Obr. 12.2: Kamera

12.2 Video dekodér

12.2.1 Výběr dekodéru

Pro digitalizaci analogového signálu z kamery jsem vybral obvod TVP5150AM1PBSR, který obsahuje 9bitový A/D převodník běžící na frekvenci 27 MHz. Do dekodéru mohou být zapojeny dvě kamery s kompozitním výstupem nebo jedna s výstupem typu S-video. Dále čip podporuje kódování PAL a NTSC. Výstupem jsou digitální data ve formátu ITU-R BT.656 nebo 8-Bit 4:2:2.

12.2.2 Zapojení dekodéru

Použil jsem vzorové zapojení od firmy Texas Instruments a upravil ho pro příjem kompozitního videa [16]. Dekodér pro svou funkci potřebuje 14.31818 MHz krystal. Dále je nutné zajistit napájecí napětí 3.3 V a 1.8 V. Tento úkol plní lineární stabilizátory U1(1.8 V) na řídicí desce a U6 na desce s DC/DC měničem. Na vstupu do A/D převodníku je anti-aliasingový filtr tvořený kondenzátory C20, C21 a cívkou L2. Za ním následuje napěťový dělič z rezistorů R10 a R11, který sníží napětí z kamery (až 1 V) pod úroveň 0.75 V. Rezistor R15 spolu s kondenzátorem C7 zajišťují zpožděné zapnutí obvodu po náběhu napájení. K MCU jsem vyvedl signál pro uvedení dekodéru do standby módu (PDN) a signál INTREQ/GPCL/VBLK, který může poskytovat údaje o vnitřním stavu dekodéru. Obvod se konfiguruje posíláním příkazů po I2C sběrnici (viz. kapitola o programování 12.2.3). Adresa pro I2C komunikaci se nastavuje napětím na výstupním pinu YOUT7. Pull-down rezistor

R16 určil adresu 0xB8. Digitální data vystupují na pinech YOUT0 až YOUT7. Pro první verzi řídicí desky jsem volil zabudovanou synchronizaci bez signálů VSYNC a HSYNC, abych ušetřil dva vodiče. Po dlouho trvajících problémech s příjmem jsem u druhé verze desky přešel, i přes složitější návrh desky plošného spoje, na externí synchronizaci viz. příloha H.

12.2.3 Software

Dekodér obsahuje konfigurační registry, do kterých se zapisuje pomocí I2C sběrnice. Z výroby je obvod přednastaven, a proto jsem pracoval jen s následujícími registry (s ukázkou nastavení pro externí synchronizaci):

Device ID Register - Slouží pro diagnostiku. Vyčtením tohoto registru bych měl získat hodnotu 0x51 hex.

Outputs and data rates select - Zápisem hodnoty 0x40 zvolím 8bitový 4:2:2 výstupní formát s externí synchronizací.

Miscellaneous Control Register - Zápisem hodnoty 0x0F dojde k povolení hodin PIXCLK, zapnutí vertikálního a horizontálního synchronizačního signálu a povolení YOUT výstupů.

Status Register 1 - Nachází se zde informace o úspěšnosti zachycení dekodéru na synchronizační pulsy, frekvenci snímků a další.

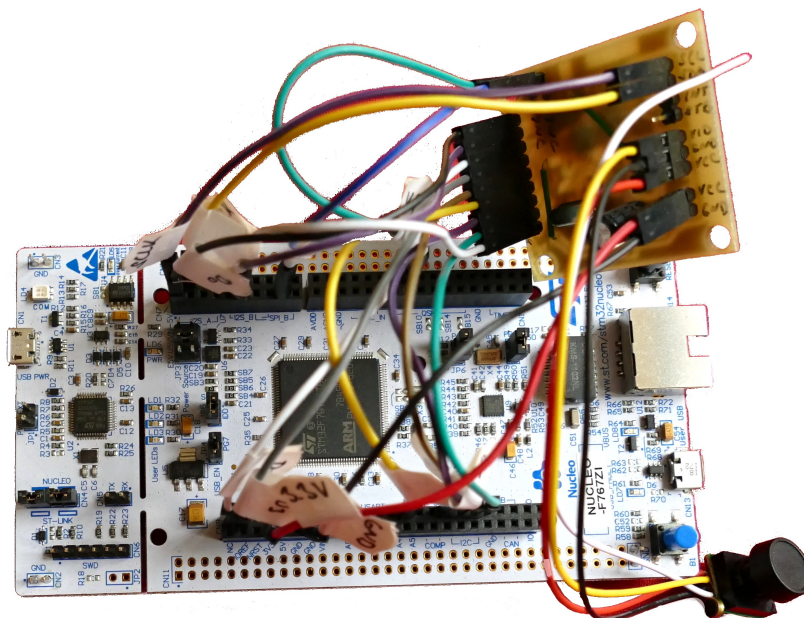
Status Register 2 - Poskytuje informaci o síle vstupního signálu a další.

Nejprve jsem se snažil přijmout obraz se zabudovanou synchronizací a jelikož na přijmutí celého obrázku nemá MCU dostatečnou vnitřní paměť, zvažoval jsem použití funkce ořezu obrázku (Cropp). Jelikož jsem ale v průběhu řešil problémy uvedené níže, k ořezu obrázku jsem se ani nedostal.

1. DCMI nepřijímalo žádná data. Řešení: úprava kondenzátorů u krystalu, aby se rozkmital.
2. DCMI přijímá divná data. Řešení: Opravit nastavení aktivních hrany u synchronizací.
3. Podle NDTR čítače DMA přenosů bylo přeneseno požadované množství dat z kamery, ale pokud si je pošlu do počítače přes sériovou linku, dojde jich vždy náhodné množství. Řešení: PC má přijímací buffer, který data vypíše do konzole až poté, co se naplní. Za data jsem dal výplňové znaky.
4. DCMI přijímá i oblasti, kde nejsou přenášena obrazová data. Obraz je často rozpůlený. Nedokáže se tedy orientovat v synchronizačních značkách.

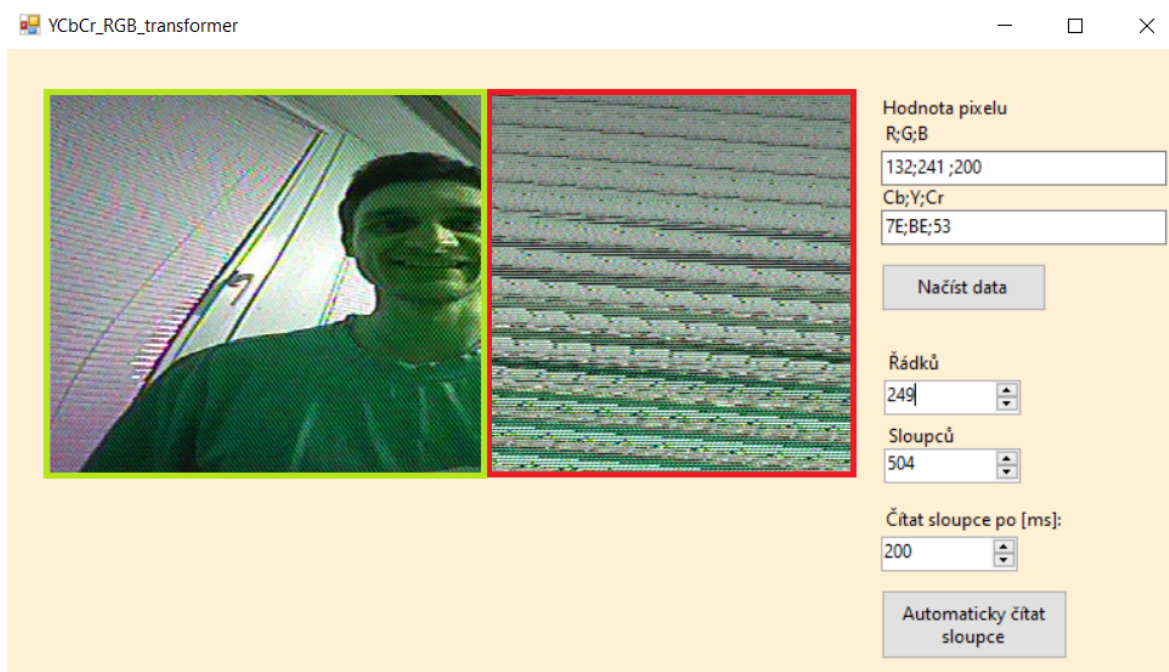
Poslední problém se mi nepodařilo vyřešit. Nastavení synchronizačních značek pro konce a začátky obrázků a řádek jsem mnohokrát kontroloval. Zkusil jsem dokonce všechny jejich kombinace. Přijatý obrázek byl ale vždy rozpůlený, nebo se přijaly i oblasti, ve kterých není obrazová informace. Nepomohla ani velmi stručná dokumentace k zabudované synchronizaci dekodéru a úplně chybějící dokumentace ke kameře. Z těchto důvodů jsem opustil myšlenku zabudované synchronizace a začal se věnovat externí synchronizaci, která velmi zjednodušuje programovou obsluhu.

Na existující řídicí desku nebylo možné přidělat rozvod synchronizačních signálů. Proto jsem navrhl a vyrobil pokusnou desku jen s dekodérem a součástkami pro jeho provoz (obrázek 12.3). Tuto desku jsem propojil s vývojovou deskou Nucleo F767ZI, která disponuje DCMI rozhraním a pokusil se o přenos obrázku s externí synchronizací.



Obr. 12.3: Pokusná deska s dekodérem

Hned prvním pokusem jsem získal velké množství dat ve formátu YCbCr a odeslal je do počítače přes sériový port. Hexadecimální data bylo třeba převést na RGB hodnoty a jednotlivé pixely vykreslit. V C[#] jsem za tímto účelem vytvořil program YCbCr_RGB_transformer, který načte data z textového souboru a převede je na bitmapový obrázek.



Obr. 12.4: Program pro transformaci dat

Počet řádků a sloupců vykreslované bitmapy zadá uživatel. Pokud ale rozměry obrázku neví a zadá je špatně, ukáže se nesrozumitelný obraz. Proto jsem přidal i možnost automatického zvyšování počtu sloupců v určeném časovém intervalu. Program tak stále překresluje obrázek a uživatel zastaví čítání sloupců ve chvíli, kdy je obrázek správně vykreslený. Další užitečnou funkcí programu je, že pokud se klikne na určité místo obrázku, program zobrazí YCbCr a RGB hodnotu označeného pixelu. Ukázka programu i vykreslení obrázku při správně nastaveném počtu sloupců (zelený obdélník) a špatně nastaveném počtu sloupců (červený obdélník) je na obrázku 12.4.

Použil jsem DCMI funkci ořezu dat, abych přijal zhruba střed obrázku a ten se vešel do vnitřní RAM paměti. Kamera má objektiv typu „rybí oko“. Díky tomu má pořízená fotka široký úhel záběru, i když je oříznutá.

Na obrázku níže je několik menších snímků, pořízených se zhruba minutovým rozestupem pomocí zkušební desky. Úplně vpravo je snímek vyfocený mobilním telefonem. Ostatní snímky jsou z kamery. Je vidět silná změna podání barev, která se v čase mění. Šlo o špatnou stabilitu hodinového signálu, která způsobila chybné vyhodnocení fáze vstupního signálu do dekodéru. Řešením byla úprava velikosti zatěžovacích kapacit krystalu. Kdyby se to nepovedlo, byl jsem připravený přejít na černobílé fotky.



Obr. 12.5: Problém s barvami

Po výrobě vylepšené řídicí desky a změně součástek problém s barvami zmizel, ale objevil se jiný. Využitím knihoven FatFs pro přístup k SD kartě znovu vzrostly nároky na vnitřní paměť procesoru a tak bylo nutné přijímat ještě menší fotku. Konkrétně 250x250 pixelů. Naopak pozitivní vliv na rozlišení fotky mělo správné pochopení přijímaných dat. Dříve jsem ze čtveřice přijatých bajtů použil dvě barvonosné složky, jednu jasovou a vypočítal z nich RGB hodnotu jednoho pixelu. Čtvrtý jasový bajt jsem zahazoval, což byla chyba. Čtveřice bajtů je totiž určena hned dvěma pixelům, kde barvonosná složka je pro oba stejná, ale jasové složky se liší. Stačilo tedy využít pro získávání pixelů i čtvrtý bajt a rázem jsem ze stejného množství přijatých dat získal dvojnásobné rozlišení.



Obr. 12.6: Foto z trubky ucpané kamenem

12.3 Mikro SD karta

12.3.1 Hardware

Existuje několik typů konektorů na mikro SD karty:

Push Push - složitý mechanismus s pružinou. Uživatel kartu zasune, tím napne pružinu a karta se zachytí v konektoru. Při opětovném zatlačení na kartu se aktivuje pružina a kartu vysune.

Push Pull - jednoduchá konstrukce. Uživatel kartu zasouvá i vysouvá bez pomoci.

Výklopné - konektor má otočný kloub. Horní kryt konektoru se posune dopředu, tím dojde k jeho rozevření a lze vložit či vyndat kartu.

V prvním návrhu desky jsem použil výklopný typ. Po úpravách konstrukce vláčku bylo nutné řídicí desku otočit a k mikro SD kartě už nebyl dobrý přístup. Proto jsem navrhl novou řídicí desku s konektorem typu Push Push, kde bude karta přístupná otvorem v boku robota. K mikro SD kartě se procesor připojuje přes následující piny:

VSS - zem

VDD - napájení 3.3 V

CLK - hodinový signál

CMD - obousměrná výměna informací a příkazů

DAT 0 až 4 - obousměrná výměna dat

SHIELD - stínění připojené na zem

SWA, SWB - spínač detekující vložení karty

Aby datové piny nezačaly při nepřítomnosti karty plavat, jsou vytaženy pull up rezistory k napájecímu napětí. Schéma s periferními moduly si lze prohlédnout v příloze G.

12.3.2 Software MCU

Mikro SD karta je k MCU připojena přes SDMMC rozhraní, které podporuje komunikaci s Multi Media kartami (MMC), SD kartami (Secure Digital) a SDIO zařízeními (SD I/O). Na příkaz generuje hodinový signál a řídí tok dat.

Pro organizaci souborů na paměťovém médiu používám volně dostupnou open source knihovnu FatFs. Ta dokáže s paměťovým médiem pracovat jako s adresářem. Pomocí FAT tabulky drží povědomí o umístění souborů na disku, ale i další informace, jako kdy byl soubor vytvořen, jeho velikost, zda je jen pro čtení atd. Použitím knihovny mám k dispozici různé funkce jako připojení k paměťovému médiu, čtení a zápis do souborů apod. FatFs knihovna nepřistupuje přímo k hardwaru, ale využívá funkce driveru pro SD karty. Provázání knihovny a driveru už zajistila firma STM32 a programátor se o něj nemusí starat. Všechny funkce, které jsou implementací knihovny k dispozici, se nacházejí v souboru ff.c. V projektu využívám následující funkce.

f_mount - na SD kartě definuje pracovní oblast a provede připojení nebo odpojení

f_getfree - poskytne informace o velikosti dostupné paměti

f_open - otevře nebo vytvoří soubor

f_lseek - přesune pointer o x bajtů, nebo na konec souboru

f_printf - zapíše data do souboru ve zvoleném formátu

f_read - přečte data ze souboru

f_close - zavře soubor

12.4 Mikroprocesor

12.4.1 Volba mikroprocesoru

Mikroprocesor STM32F767VIT6 jsem zvolil s ohledem na jeho podporu DCMI rozhraní a možnost JPEG konverze. Dalšími požadavky byla dostatečná velikost vnitřní RAM paměti (512 kB) na uložení obrázku a vysoká rychlost hodin (216 MHz). Je vyráběn v pouzdře LQFP 100, které zabere mnoho místa na malé desce, ale menší alternativu se stejnými schopnostmi výrobce STM32 v současné době nenabízí.

12.4.2 Napájení

MCU je napájen 3.3 V z lineárního stabilizátoru. Co nejbližší každému z napájecích pinů jsou připájeny blokovací kondenzátory C4 až C8. Ty zabrání šíření proudových špiček od napájecích pinů směrem ke zdroji, tím pádem i poklesům napájecího napětí. Pro stabilizaci regulační smyčky zdroje napětí je přidán ještě keramický kondenzátor C9.

12.4.3 Návrh krystalového oscilátoru

Časování MCU zajišťuje 8 MHz krystal X1. Aby se krystal rozkmital, uvádí výrobce potřebné zatížení kapacitou CL 18 pF. Parazitní zatěžovací kapacitu Cs jsem odhadl na 5 pF a vypočítal dvojici zatěžovacích kondenzátorů (C2, C3) podle vzorečku:

$$2 * (CL - CS) = 2 * (18 - 5) = 26pF$$

Použil jsem nejbližší dostupnou hodnotu kondenzátoru 22 pF. Krystal tak bude kmitat na mírně vyšší frekvenci.

12.4.4 Volba bootování

Napětím na pinech BOOT0 a BOOT1 lze nastavit, jaká adresa se po resetu MCU načte do programového čítače. Lze tak spustit kód z Flash, ROM nebo SRAM paměti. Bootování z ROM se používá např. pro upgrade firmwaru. Pro testování programu je zase dobré využít SRAM paměť. Ta se po malé změně programu nemusí mazat a přepisovat celá jako FLASH paměť a testování probíhá rychleji. Poslední možností je bootování z FLASH paměti. Tu jsem využil a nastavil pomocí rezistoru R8. Následující tabulka udává, jaké napětí na pinech BOOT0 a BOOT1 odpovídá jaké paměti.

BOOT1	BOOT0	Paměť
-	0	Flash
0	1	ROM
1	1	SRAM

Tab. 12.1: Volba bootování

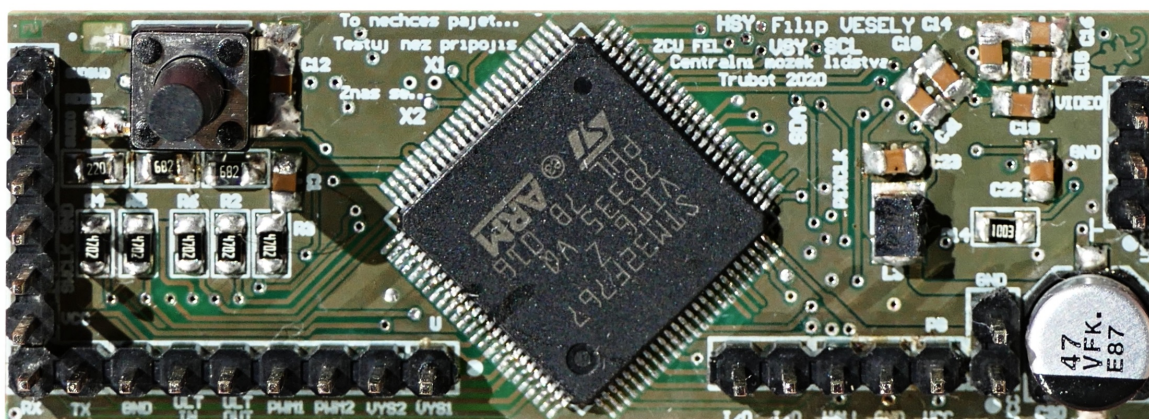
12.4.5 Programování přes SWD

Testování a nahrávání programu je možné díky protokolu SWD od společnosti ARM. Tento protokol definuje pouze dva signály SWCLK a SWDIO. Ty jsou přivedené na konektor P1 a odtud zavedené na příslušné piny do MCU, společně s dalšími čtyřmi volitelnými signály. Pro programování využívám ST-Link programátor zabudovaný do vývojové destičky Nucleo. Signály a jejich funkce jsou popsány v následující tabulce.

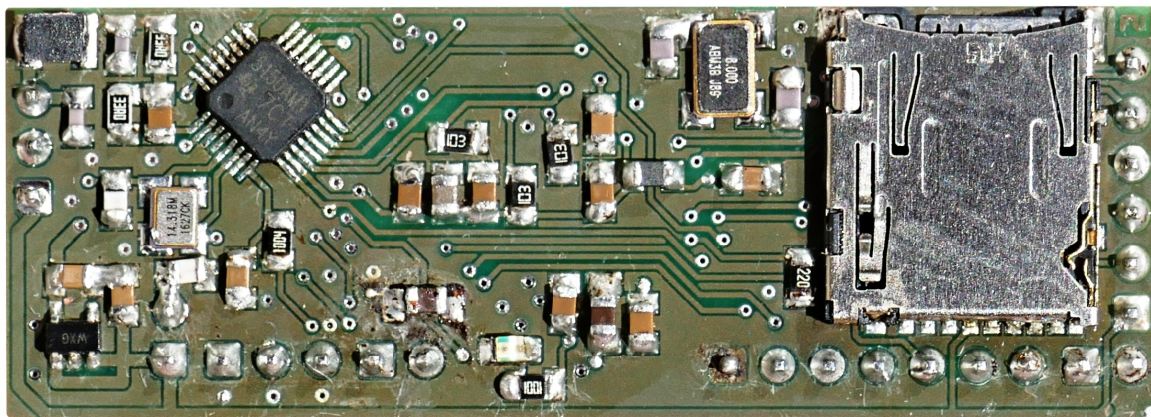
PIN ST-LINK/CN4	POPIS	PIN PROCESORU
1. VDD_TARGET	Měří napětí	Vhodný je analogový 3.3 V pin (VDDA)
2. SWCLK (TCK)	Hodiny	TCK pin (PA14)
3. GND	Zem	Jakýkoliv GND pin
4. SWDIO (TMS)	Data	TMS pin (PA13)
5. NRST	Reset	NRST pin (RESET)
6. SWO	Rezervovaný	SWO pin (PB3)

Tab. 12.2: Programování přes SWD

Pro uživatelsky přívětivější testování programu jsou na konektor P7 přivedeny signály RX a TX sériové sběrnice USART. Díky tomu je možné si mezi MCU a počítačem posílat zprávy. Schéma řídicí desky si lze prohlédnout v příloze F. Reálný vzhled řídicí desky s rozměry 59 x 21 mm je přiložen níže.



Obr. 12.7: Řídicí deska s mikroprocesorem TOP



Obr. 12.8: Řídící deska s mikroprocesorem BOTTOM

12.4.6 Software MCU

Procesoru jsem zapnul ART Acceleration (uleví procesoru od grafických operací), Instruction prefetch (načítá instrukce dopředu a snižuje počet wait stavů), ICache (vloží další instrukční Cache paměti před procesor), DCache (vloží další datové cache před procesor).

Pokud se zrovna nefotí, běží hodiny procesoru na 16 MHz. Při focení a ukládání dat na SD kartu je potřeba dosáhnout vysokých přenosových rychlostí. Dojde proto k přetaktování systémových hodin pomocí fázového závěsu na 216 MHz. Je tak dodržen vzorkovací teorém pro rozhraní DCMI. Přepsáním metody HAL_InitTick, jsem nastavil časovač SYSTICK na 1 ms. Konfiguraci periférií vygeneroval CubeIde automaticky, podle nastavení v grafickém editoru.

Program ve while cyklu je koncipován jako stavový automat. Stisknutí tlačítka vyhodnotí robot jako žádost o inspekční cestu. Zkontroluje zda je zasunuta SD karta a pokud ano, přejde do stavu **inicializace**. Po krátkém čase, kdy je možné robota zasunout do trubky zapne motory a vydá se na cestu - stav **jízda vpřed**. Při pohybu periodicky kontroluje otáčení koleček. Ve chvíli kdy se začnou protáčet na místě, je detekována překážka a informace o ujeté vzdálenosti se zapíše na SD kartu. Hned potom robot přejde do stavu **focení**. Popojede lehce dozadu, přetaktuje procesor a pořídí několik fotek. Po uložení dat na SD kartu takt znovu sníží a naváže stavem **jízda vzad**. Až obsluha znovu zmáčkne tlačítko, uvede se robot do stavu **stand-by** a vypne motory.

Z dostupných schopností mikroprocesoru jsem využil několik čítačů v různých režimech, rozhraní SDMMC (připojení k micro SD kartě), USART3 (komunikace s PC), I2C (konfigurace TVP5150), DCMI (příjem obrazových dat), DMA (přesun obrazových dat z paměti na SD kartu) a knihovnu FatFs pro zápis na SD kartu.

Části kódu které se týkají např. konfigurace video dekodéru, řízení H-můstku a dalších, jsou popsány v kapitolách nadepsaných "Software MCU"u příslušných periférií.

ZÁVĚR

CÍLEM práce bylo vyrobení prototypu robota pro inspekci úzkých trubek. V jejím průběhu jsem se setkal s nechtěnými oscilacemi step-up měniče a krystalu, aliasingem při měření osciloskopem, vyrovnávacími proudy mezi komunikujícími procesory nebo třeba rušením videa a ultrazvuku. Naučil jsem se hodně o formátech a zpracování analogového videa, jeho převodu na digitální data, uložení a zobrazení. Zajímavé zkušenosti jsem získal při řešení přeměn energie v robotovi, spotřebě procesoru při vyšších frekvencích, výběru součástek (omnipolární Hallův senzor, TLV431, procesor, měnič, konektor pro SD kartu) a návrhu oboustranných desek plošných spojů.

Robot, který vznikl, se s rezervou vejde do HDPE trubky. Dokáže jí projíždět, i když pomalu, a nedělá mu potíže ani jízda do kopce nebo na zádech. Po nárazu do překážky pořídí několik fotek a spolu s informací o ujeté vzdálenosti je uloží na SD kartu. Pro rychlejší návrat, lze robotovi pomoci navíjením vlákna, které má připevněné k zadnímu nárazníku. Ultrazvukovou komunikaci jsem úspěšně odzkoušel ve volném prostoru a kratších trubkách. I přes problémy s místem uvnitř vagónků jsem zachoval všechny konektory potřebné k programování a komunikaci s počítačem přes sériový port. Do budoucna vidím tyto cesty, jak robota vylepšit.

3D model - Po zkušenostech se skládáním robota bych přidal více prostupů pro kabeláž. Také bych použil tenčí kabely a pokusil se sehnat specializované konektory, abych nemusel ohýbat dutinky do pravého úhlu. Snadnější montáži by pomohlo i lehké prodloužení celého těla.

Ultrazvuk - V obvodu přijímače by bylo dobré lépe naladit RC filtr, aby nevznikalo tolik falešných impulsů, které zatěžují SW filtr v procesoru. Dále by šlo obvod vylepšit o vysílací část, aby robot mohl na příkazy odpovídat. Zajímavým SW vylepšením by mohla být kalibrace časových konstant, sloužících pro správné dekódování přijatých znaků. První zpráva odeslaná robotovi, by měla přesně daný obsah a robot by si podle ní přizpůsobil dekodér. Operátorský ultrazvukový vysílač jsem v práci také vyřešil, ale dal by se upravit do kompaktnějších rozměrů spolu s volbou vysílaných příkazů.

Kamera - Určitě bych si dal pozor při koupi kamery a vybral tu s neprokládaným výstupním videem. Na prokládané video totiž není DCMI rozhraní procesoru stavěné. Tím by se pravděpodobně vyřešil problém se synchronizací, který občas způsobí příjmem deformované, nebo jinak poškozené fotky. Kvalitě fotek by určitě pomohlo i pořízení procesoru s větší pamětí, do které se vejde celá fotka. K ukládaným bajtům fotky by šlo programově přidat hlavičková data pro bitmapový obrázek a k jejímu zobrazení

v PC by v tu chvíli už nebyl potřeba můj program YCbCr_RGB_transformer. Při pokusech jsem si všiml, jak velký odběr má kamera a přisvicovací LED dioda. Proto jsem vložil do napájecího kabelu MOSFET tranzistor, který spínám pomocí nevyužitého pinu procesoru. V novém návrhu bych tento tranzistor zabudoval na DPS. Dalším doplňkem kamery by měl být nárazník, který zamezí kontaktu kamery s překážkou a zároveň nebude moc vadit při focení.

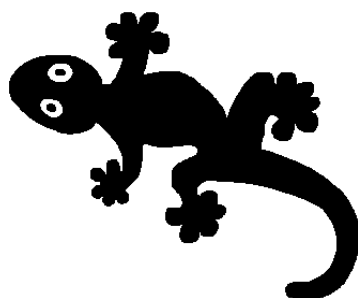
Pohon - V těle vláčku je stále dostatek místa pro výkonnější a rychlejší motor. Podstatného vylepšení by šlo dosáhnout nalezením optimálního tvaru přitlačných plastových fólií.

Napájení - Nefungující zapojení hlídající podvybití baterie se součástkou TLV431 bych nahradil zapojením s operačními zesilovači. LED signalizující přítomnost 3.3V napětí na řídicí desce bych nepřipojil rovnou k napájecímu vodiči, ale k pinu procesoru. To by poskytovalo mnohem lepší diagnostické možnosti a uspořilo energii, pokud nebude signalizace potřeba.

SD karta - Od SD karty jsem nevyvedl signál oznamující její přítomnost v konektoru. FatFs knihovna s tímto signálem ale počítá a pokud dojde k odpojení SD karty za běhu programu, umí se tomu přizpůsobit.

Podzemní vysílač - Robot měl původně disponovat i zemním vysílačem pro lokalizaci robota vyhledávacím zařízením. Od toho mě odradil nedostatek volného prostoru. Možná by se ale dalo využít samotných motorů. Jejich vinutí by mohly sloužit zároveň jako cívky vysílající elektromagnetické vlny do okolí. Stačilo by přidat ochranné diody proti napěťovým špičkám a nastavit PWM modulaci řídicí H-můstek na 33kHz.

Myslím si, že robot je i bez uvedených vylepšení prakticky použitelný pro inspekci kratších úseků kontrolovaných trubek nebo jako výuková pomůcka, na které si lze vyzkoušet mnoho programátorských technik.



LITERATURA

- [1] Radiodetection PDL2BA1-4514 [online], 2020. [cit. Březen 3, 2021]. Dostupné z: <https://accusrc.com/product-Radiodetection-PDL2BA1-4514>.
- [2] Plumett MINIJET [online], 2021. [cit. Březen 3, 2021]. Dostupné z: <https://www.plumettaz.com/en/product/minijet/P02>.
- [3] C.SCOPE. *Accessory Range for Pipe & Cable Location Equipment*. Part no. B1080 Issue 3.
- [4] STMicroelectronics. *Oscillator design guide for STM8AF/AL/S, STM32 MCUs and MPUs*, prosinec 2020. AN2867 Rev. 13.
- [5] NXP Semiconductors. *S12VR Hardware Design Guidelines*, říjen 2016. AN4643 Rev. 3.
- [6] Atmel. *AVR Hardware Design Considerations*, září 2016. AVR042.
- [7] STMicroelectronics. *Digital camera interface (DCMI) for STM32 MCUs*, srpen 2017. AN5020 Rev. 1.
- [8] ITU-R. *Digital interfaces for HDTV studio signals*, leden 2012. BT.1120-8.
- [9] Cliff Win. *Measuring composite-video signal performance requires understanding differential gain and phase, Part 1 of 2* [online], 2007. [cit. Březen 3, 2021]. Dostupné z: <https://www.edn.com/measuring-composite-video-signal-performance-requires-understanding-differential-gain-and-phase-part-1-of-2/>.
- [10] Arpit Mehta. *How to level shift video signals for dc-coupled video amplifiers/filters* [online], 2007. [cit. Březen 3, 2021]. Dostupné z: <https://www.maximintegrated.com/en/design/technical-documents/app-notes/4/4028.html>.
- [11] Gregory C. Ewing. *How the PAL Colour Video System Works* [online], 2019. [cit. Březen 3, 2021]. Dostupné z: https://www.cosc.canterbury.ac.nz/greg.ewing/c64_pal_decoder/PAL/PAL-Video-Encoding.html.
- [12] Diodes Incorporated. *PAM2423 3A, 4.5A, 5.5A PWM STEP-UP DC-DC CONVERTER*, leden 2016. DS36383 Rev. 2 - 2.

- [13] *Mechanika a termika, 4. Akustika*. [cit. Březen 3, 2021]. Dostupné z: http://ufmi.ft.utb.cz/texty/mech_term/F2_04.pdf.
- [14] Seiichi Inoue. Ultrasonic range meter [online], 2001. [cit. Březen 3, 2021]. Dostupné z: http://www.piclist.com/images/www/hobby_elec/e_pic6_6.htm.
- [15] S Senthilkumar and R Vinothraj. Design and study of ultrasound-based automatic patient movement monitoring device for quantifying the intrafraction motion during teletherapy treatment. *Journal of applied clinical medical physics*, 13(6):82–90, 2012.
- [16] Texas Instruments. *TVP5150A Quick Start Guide*, říjen 2003. SLEU044.

SEZNAM OBRÁZKŮ

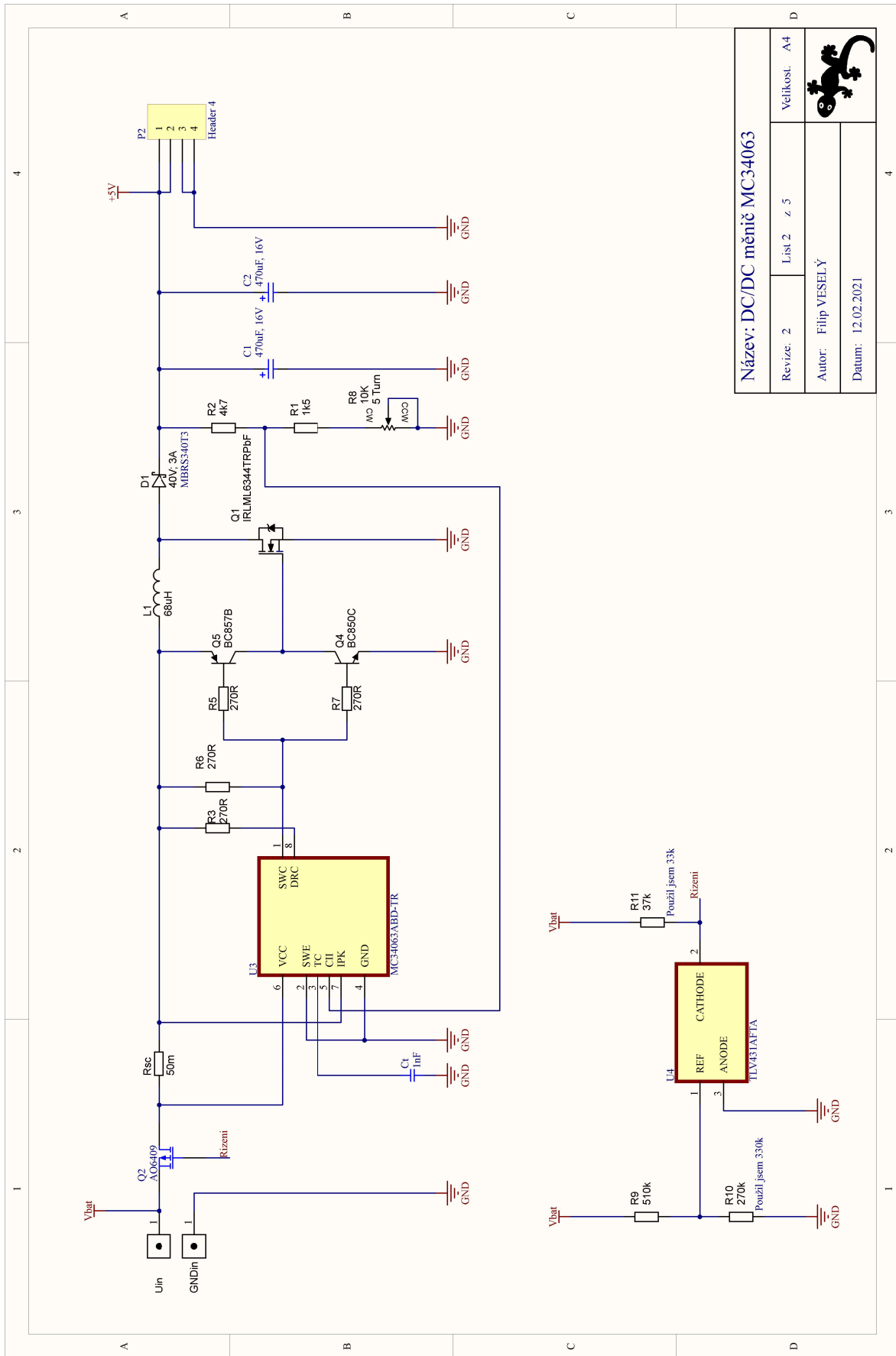
1.1	Zafukování optického kabelu kompresorem	3
1.2	Hledačka [1]	3
1.3	Pneumatický pohon [2]	4
1.4	Rádiová sonda	5
2.1	Piercův oscilátor [4]	6
2.2	Převodní charakteristika [4]	7
2.3	Správně navržené plošné spoje [5] [6]	8
3.1	Blokový diagram DDMI [7]	10
3.2	SAV a EAV kódy [7]	11
3.3	Přenos obrázku se zabudovanou synchronizací [8]	12
3.4	Přenos obrázku s externí synchronizací [7]	13
4.1	Kompozitní video signál [9]	14
4.2	S-video [10]	15
4.3	Kódování PAL [11]	16
7.1	Trubot	19
8.1	Střední tělo z ABS	22
8.2	Tělo s plováky	23
8.3	Tělo s přitlačnou fólií	23
9.1	Konstrukce těla robota	24
9.2	Rozvod kabelů mezi vláčky 1	25
9.3	Rozvod kabelů mezi vláčky 2	25
10.1	Bateriový vagónek	26
10.2	Li-ion baterie 18650	27
10.3	Hallův senzor	27
11.1	Komunikační vagónek	28
11.2	Deska s měničem MC34063	29
11.3	Charakteristika TLV431	29
11.4	Deska s měničem PAM2423	30
11.5	Použitelné stejnosměrné motory	32
11.6	Řízení H můstku [12]	33

11.7	Ultrazvukové čidlo	33
11.8	Ozvěna ultrazvuku	34
11.9	Sestupná hrana znaku	35
11.10	Konstrukce ultrazvukového senzoru [15]	35
11.11	Ultrazvukový vysílač	36
11.12	Graf kódování	37
11.13	Ukázka vyslané zprávy	37
11.14	Úprava přijatého signálu	38
11.15	Vývojový diagram	38
12.1	Inspekční vagónek	39
12.2	Kamera	40
12.3	Pokusná deska s dekodérem	42
12.4	Program pro transformaci dat	42
12.5	Problém s barvami	43
12.6	Foto z trubky ucpané kamenem	43
12.7	Řídící deska s mikroprocesorem TOP	46
12.8	Řídící deska s mikroprocesorem BOTTOM	47


SEZNAM TABULEK

8.1	Ohyb trubky v závislosti na teplotě	21
11.1	Účinnost měniče s obvodem PAM2423	30
12.1	Volba bootování	46
12.2	Programování přes SWD	46

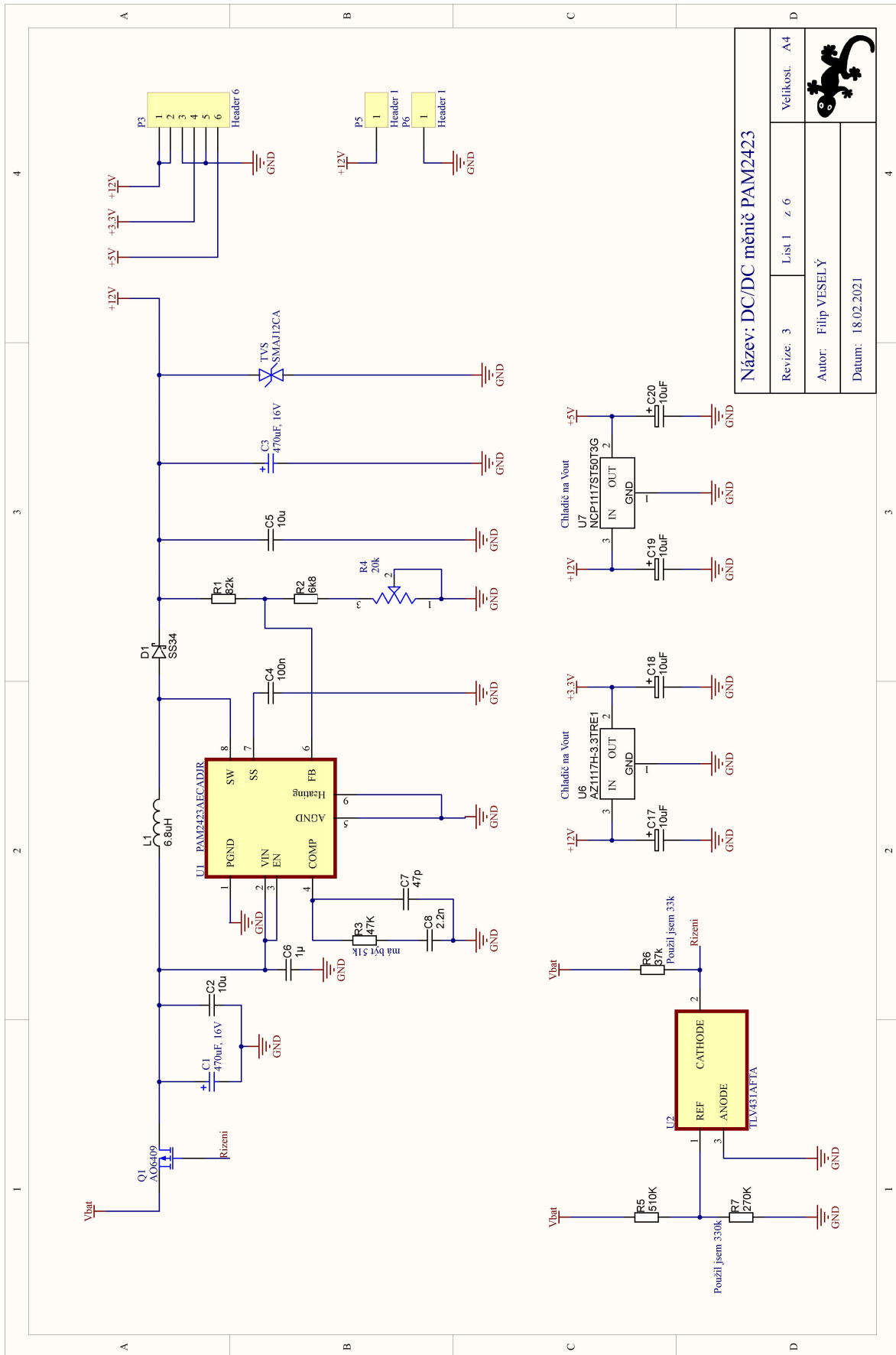
Příloha: A




Název: DC/DC měnič MC34063

Revize: 2	List 2 z 5	Velikost: A4
Autor: Filip VESELÝ		
Datum: 12.02.2021		

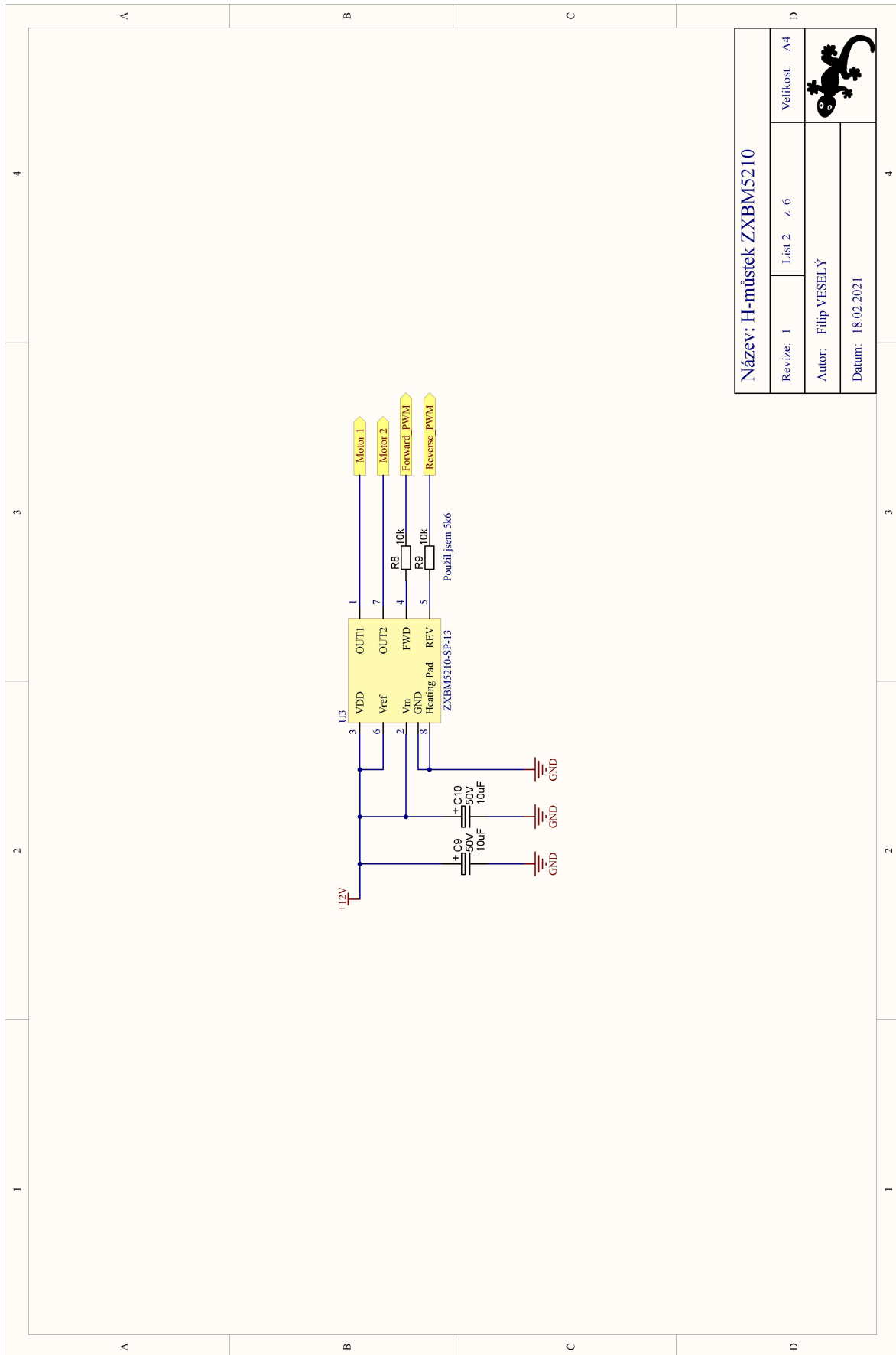
Příloha: B




Název: DC/DC měnič PAM2423

Revize: 3	List 1 z 6	Velikost: A4
Autor: Filip VESELÝ		
Datum: 18.02.2021		

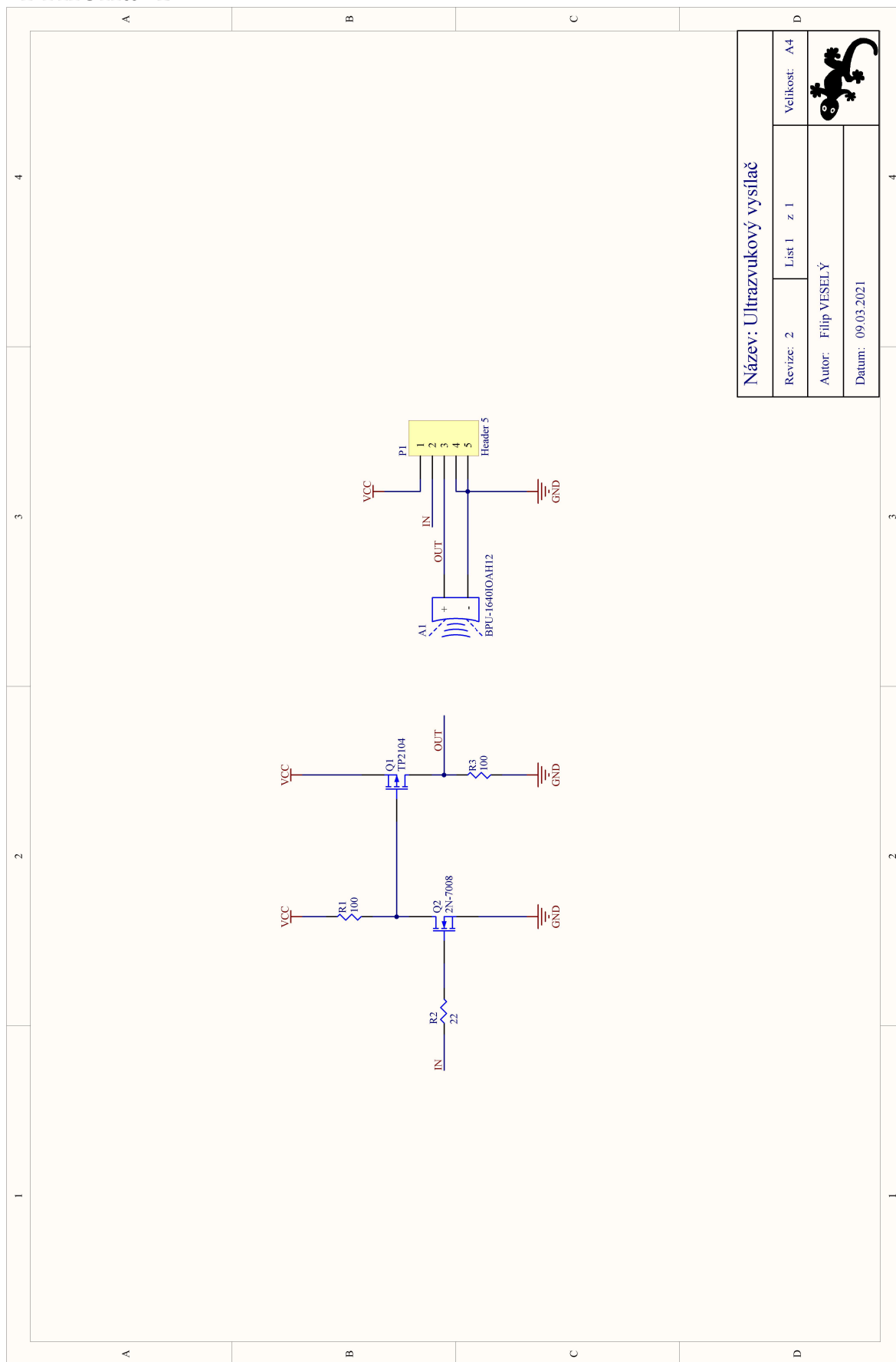
Příloha: C




Název: H-můstek ZXBM5210

Revize: 1	List 2 z 6	Velikost: A4
Autor: Filip VESELÝ		
Datum: 18.02.2021		

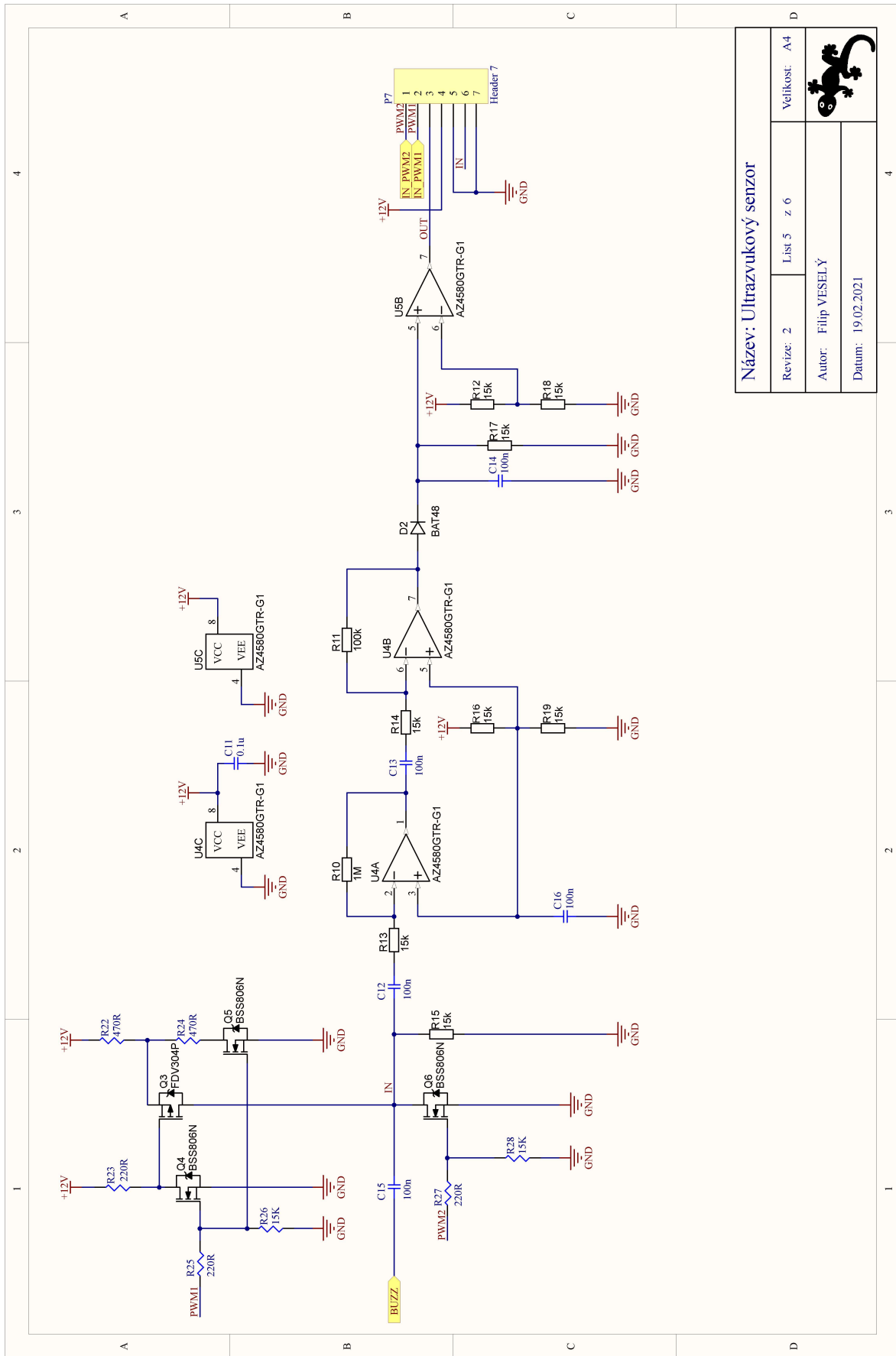
Příloha: D



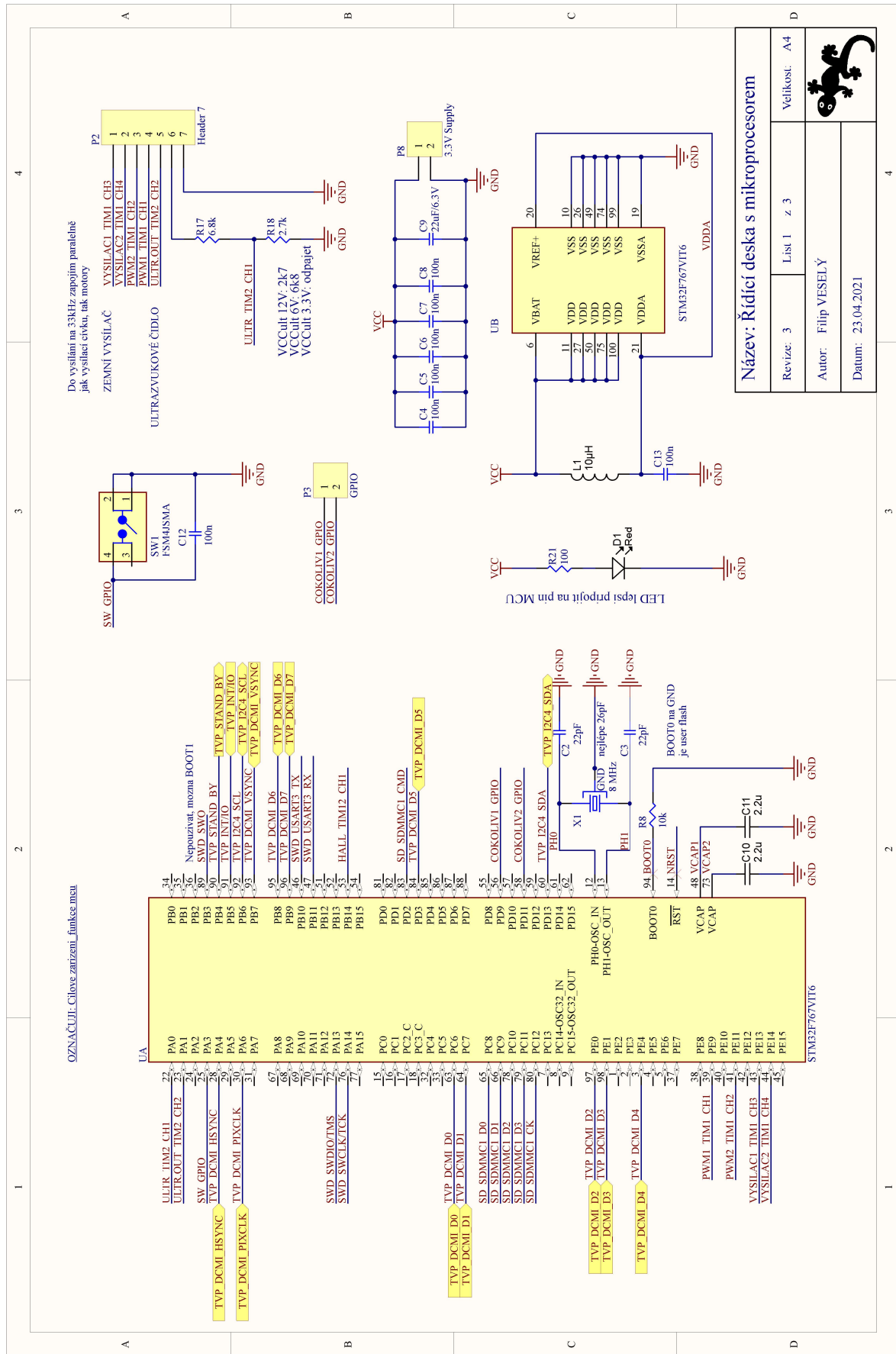
Název: Ultrazvukový vysílač

Revize: 2	List 1 z 1	Velikost: A4
Autor: Filip VESELÝ		
Datum: 09.03.2021		

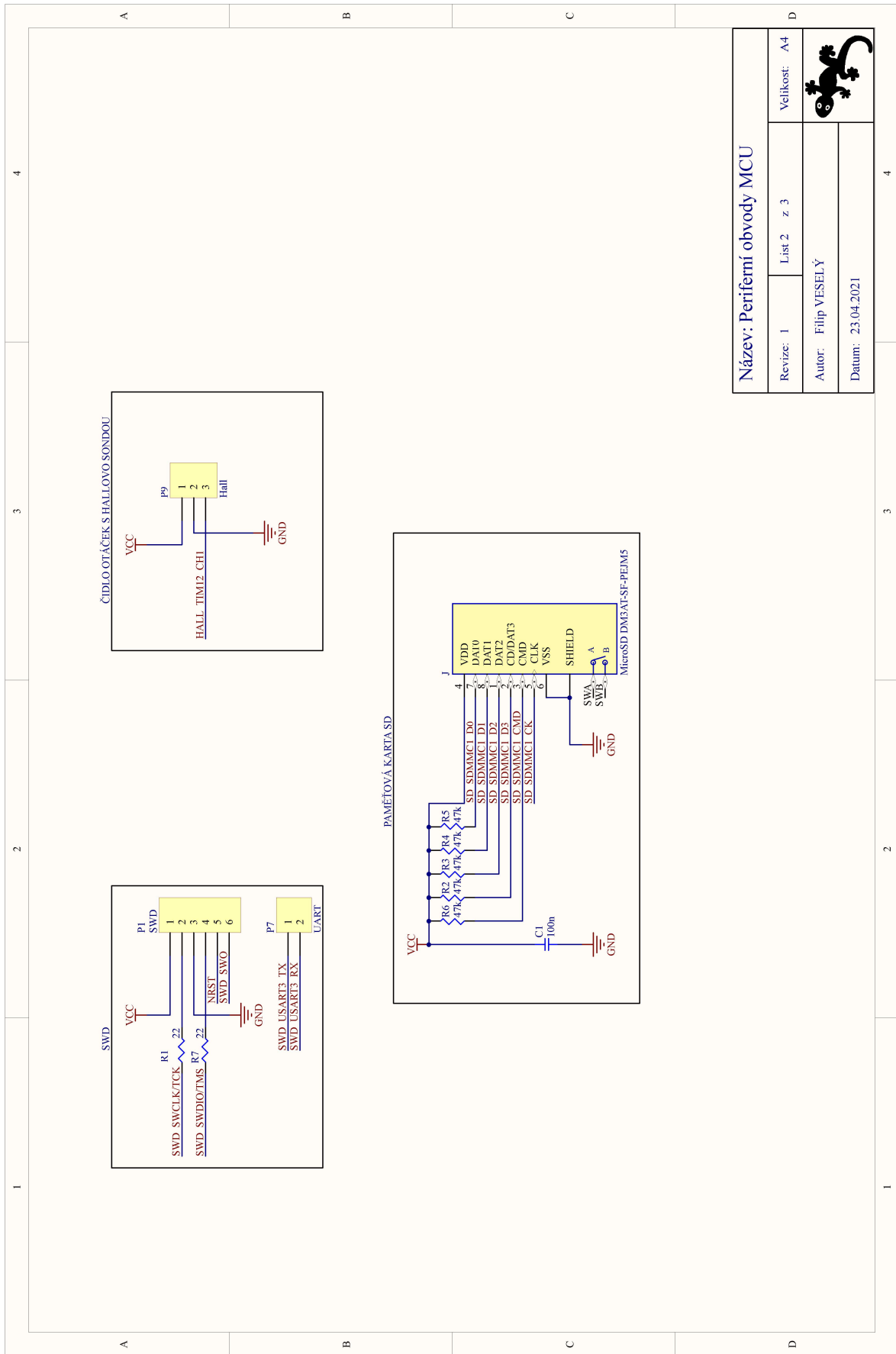
Příloha: E




Příloha: F



Příloha: G



Název: Periferní obvody MCU

Revize: 1	List 2 z 3	Velikost: A4
Autor: Filip VESELÝ		
Datum: 23.04.2021		

Příloha: H

