



## **Středoškolská technika 2023**

**Setkání a prezentace prací středoškolských studentů na ČVUT**

### **Realizace stavebnice robotického vozidla CONNÍK**

**Dalibor Soboslai**

VOŠ SPŠ a JŠ s právem státní jazykové zkoušky  
Masarykova 197, 284 11  
Kutná Hora

## **Anotace**

Tato práce se zabývá konstrukcí kompaktního modulárního robota, který umožní nést několik druhů senzorů. Cílem je uživateli výrazně usnadnit konstrukci robota, aby se plně mohl věnovat programování.

## **Klíčová slova**

Robot, senzory, Arduino, 3D tisk, bludiště, Fusion 360

## **Annotation**

This paper talks about construction of compact and modular robot which carries multiple sensors. Aim was to significantly reduce construction of robot so user can fully

## **Keywords**

Robot, sensors, Arduino, 3D print, maze, Fusion 360

# Obsah

1	Úvod.....	5
2	Popis výrobku .....	6
3	Návrh plošného spoje.....	6
3.1	Řešení tvaru plošného spoje.....	6
4	Součástky .....	9
4.1	Arduino UNO.....	9
4.2	Tlačítka.....	9
4.3	Driver Motoru .....	11
4.3.1	Motory .....	12
4.4	OLED .....	13
4.5	Registr 74HC595 a LED .....	13
4.6	Ultrasonické senzory .....	14
4.6.1	Laserové senzory .....	15
4.7	Baterie .....	15
4.8	Hallova sonda.....	16
4.8.1	Kola.....	17
5	3D tisk.....	17
5.1	FDM .....	17
5.2	LCD SLA .....	19
6	Software.....	20
6.1	Vyhodnocování tlačítek a menu.....	20
6.1.1	Zobrazování menu na OLED .....	21
6.2	Pohyb.....	22
6.3	Ovládání posuvného registru.....	24
6.4	Měření vzdálenosti z ultrasonických senzorů .....	24
7	Závěr .....	25
8	Použitá literatura .....	26
9	Seznam obrázků.....	27
10	Příloha.....	28
10.1	Robot jako učební pomůcka .....	28
10.2	Možnosti programování.....	28

# 1 ÚVOD

Ve třetím ročníku jsme s týmem zápasili s konstrukcí robota v soutěži zaměřenou na programování. To mi vnučovalo nápad sestavit robota, kterého bychom si vybavili před soutěží potřebnými sensory a mohli se plně věnovat programování. Postupem času vznikali v sešitech mezi učivem nákresy základních křivek robota a různé návrhy rozložení komponentů. Protože mi zanedlouho nestačil papír a tužka, tak jsem začal přenášet myšlenku do Fusion 360. Poté jsem náhodně narazil na sociálních sítích na soutěž s roboty na řešení bludiště, s dvěma koly a sensory a kontrolérem. Ty se staly hlavní inspirací projektu. Konstrukčně jsem chtěl dosáhnout podobné jednoduchosti, a zároveň mít k dispozici velkou nabídkou konfigurace. Stejně jako u vzorových robotu, plošný spoj slouží i jako konstrukční prvek a nese veškerou váhu všech komponentů. S plošnými spoji jsem neměl žádné zkušenosti, ale díky volně dostupným materiálům online jsem navrhl první prototyp. Ten prošel několika změnami a byl doplněn díly z 3D tiskárny.

## 2 POPIS VÝROBKU

Robot by se dal rozdělit na několik vrstev. Ta nejvrchnější by mohla být pojmenována vrstvou senzorovou, protože se zde nachází všechny senzory, displej a baterie. Nejnápadnější jsou senzory ultrasonické, jsou kolmo připevněny na desku a fungují i vypadají, jako oči robota. Dalším výrazným prvkem v této vrstvě je určitě baterie. Skládá se z černého pouzdra s bílým límcem a jsou v něm vloženy dvě světle zelené baterie. Tyto baterie jsou zdrojem energie pro celého robota.

Střední vrstvou je plošný spoj, bílá obdélníková destička s tloušťkou několik milimetrů. Je to základní prvek robota, propojuje všechny prvky robota. Shora i ze spodu je osázen konektory, ve kterých jsou zapojeny ostatní prvky. Svou funkci plní i jako konstrukční prvek, protože je k němu přidělán držák na motory, takže nese veškerou váhu.

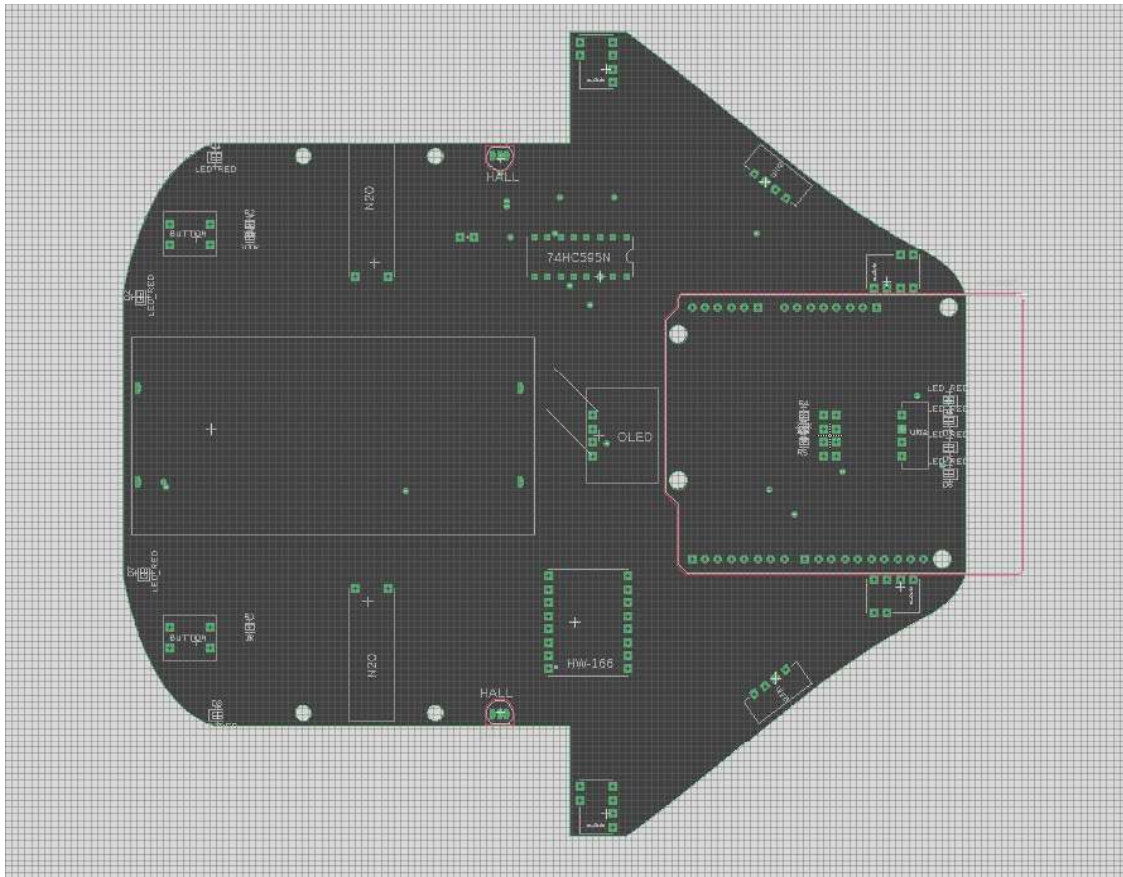
Vrstvou nejnižší je podvozek. Zde se nachází samotný mozek robota, modré Arduino, z kterého jsou vidět hlavně vstupní konektory, zbytek je schován. Najdeme tady i motory, oproti robotovi jsou poměrně malých rozměrů. K nim jsou připevněny černá plastová kola, ty sahají od nejnižšího bodu skoro po ten nejvyšší. Spolu s koly se dotýká země ještě přední noha. Je umístěna pod Arduinem a má průhledně červenou barvu.

## 3 NÁVRH PLOŠNÉHO SPOJE

Navrhovat jsem začal na papíře. U robota mi šlo hlavně o tvar, takže jsem hlavně řešil umístění senzorů tak, aby směřovaly do správných směrů. Využil jsem součástek, které jsem měl doma a různě jsem kombinoval sestavy dokud jsem s rozmístěním nebyl spokojen.

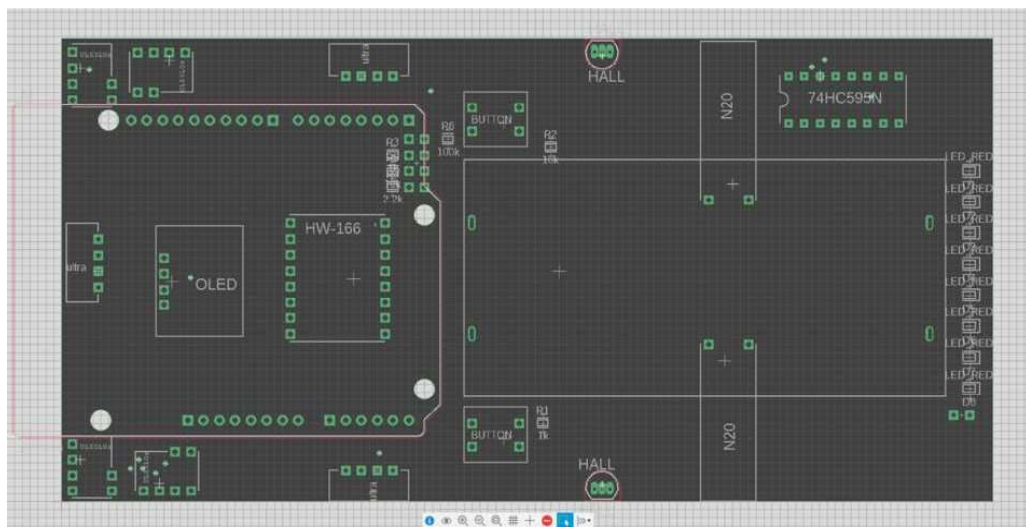
### 3.1 Řešení tvaru plošného spoje

První návrh měl boční senzory po úhlem tak, že robot by vlastně mohl vědět o překážce ještě předtím, než k ní přijede. V dalších prototypch jsem od tohoto upustil, protože se výroba výrazně prodražovala. Tento prototyp byl taky zbytečně velký a rozestupy mezi komponenty nevhodné. Už na tomto modelu jsem vyřešil připojení Arduina jako shield, kdy samotný robot je připojen shora na Arduinu.



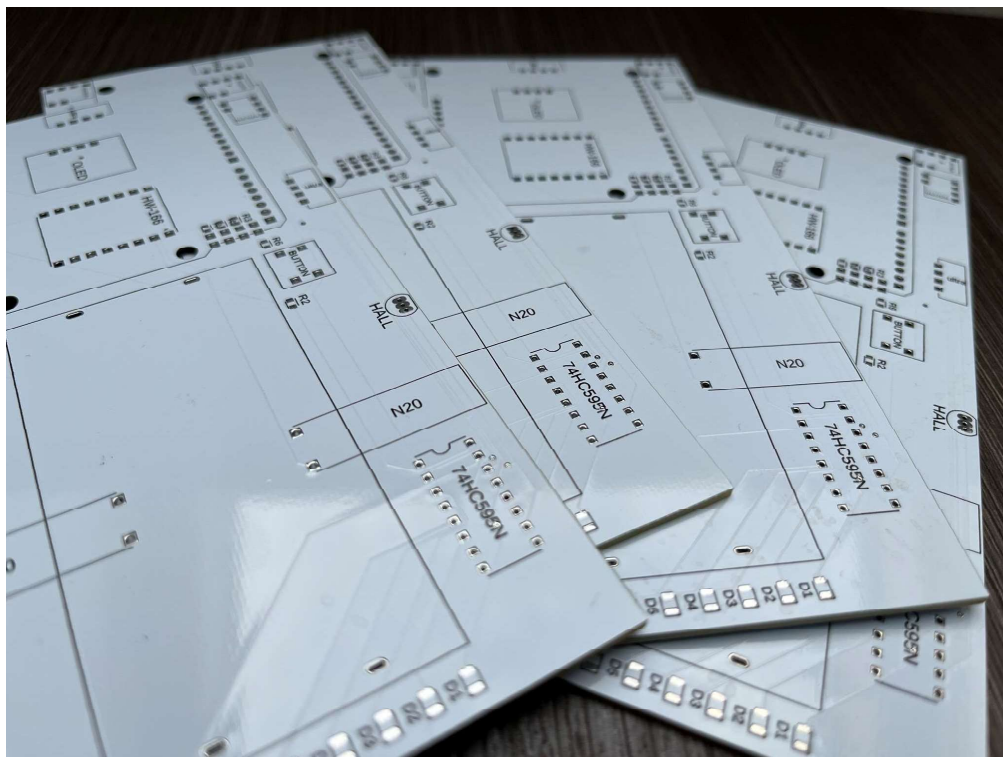
Obrázek 1: První prototyp plošného spoje

U druhého návrhu jsem již upustil od zbytečně složitých tvarů. Postranní senzory jsou již kolmé na sensor přední. Několik komponentů taktéž změnilo pozici, například můstek pro ovládání motorů jsem přesunul pod OLED. Můstek je taky jediná součástka, která by mohla způsobit komplikace při výměně, protože není v konektoru, ale je přímo napájena na desku.



Obrázek 2: Druhý prototyp plošného spoje

Do výroby jsem odeslal plošný spoj s druhou verzí rozmístění. Několikrát jsem ještě předělával cesty a občas jsem i změnil orientaci součástek. Bohužel kvůli nedostatku zkušeností jsou cesty dost chaotické, ale většinou funkční. Po doručení zásilky jsem začal pájením konektorů a postupně jsem přidával i součástky.



Obrázek 3: Hotové plošné spoje

## 4 SOUČÁSTKY

Většinu součástí jsem volil dle zkušeností z minulých projektů. Jediné dvě součástky, které jsem neměl vyzkoušené, byli laserové moduly a můstek pro ovládání motorů HW-166. Od laserových modulů jsem si sliboval vyšší přesnosti než u podobných senzorů. Můstek jsem volil podle velikosti a výkonu. Vzhledem k velikosti robota jsem zvolil menší motory, takže i samotný můstek mohl být menší a méně výkonný.

### 4.1 Arduino UNO

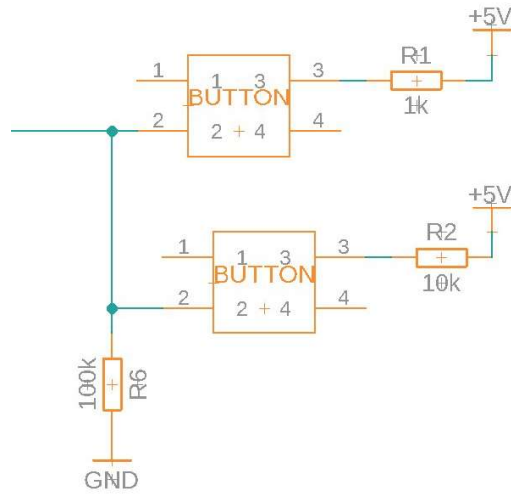
Jako mozek robota jsem zvolil Arduino UNO, a to i přes jeho větší rozměry. Největší výhodou je nepochybně robustní konektor. Kdyby nebylo potřeba pevného konektoru, mohlo by se použít například Arduino mini nebo mikro.

Deska Arduino UNO využívá čipu ATmega328P od firmy Atmel. Má 14 digitálních vstupů a 6 analogových pinů. Podporuje I2C sběrnici a je osazena regulátorem napětí, který napájí většinu součástí na robotovi. Vstupní neregulované napětí může dosáhnout až dvaceti voltů. Jsou dostupné i jeho klony, takže celková cena projektu může být výrazně sražena.

### 4.2 Tlačítka

Použil jsem dvě tlačítka jako uživatelský vstup. Pro základní nastavování hodnot a výběr možností je počet dostačující. V zapojení používám dva pull-up rezistory. Vzhledem k tomu, že bylo potřeba uvolnit jeden pin tak jsem připojil výstup obou tlačítek na jeden analogový vstup. Arduino potom detekuje dvě napěťové úrovně, a to díky úbytkům napětí na dvou různých rezistorech. Rezistor R1 je zvolen tak, aby byla stále rozeznatelná logická úroveň napětí, a zároveň se dostatečně lišila od stisku druhého tlačítka. Rezistor R2 by mohl být úplně vynechán, přidal jsem ho hlavně kvůli symetrii.

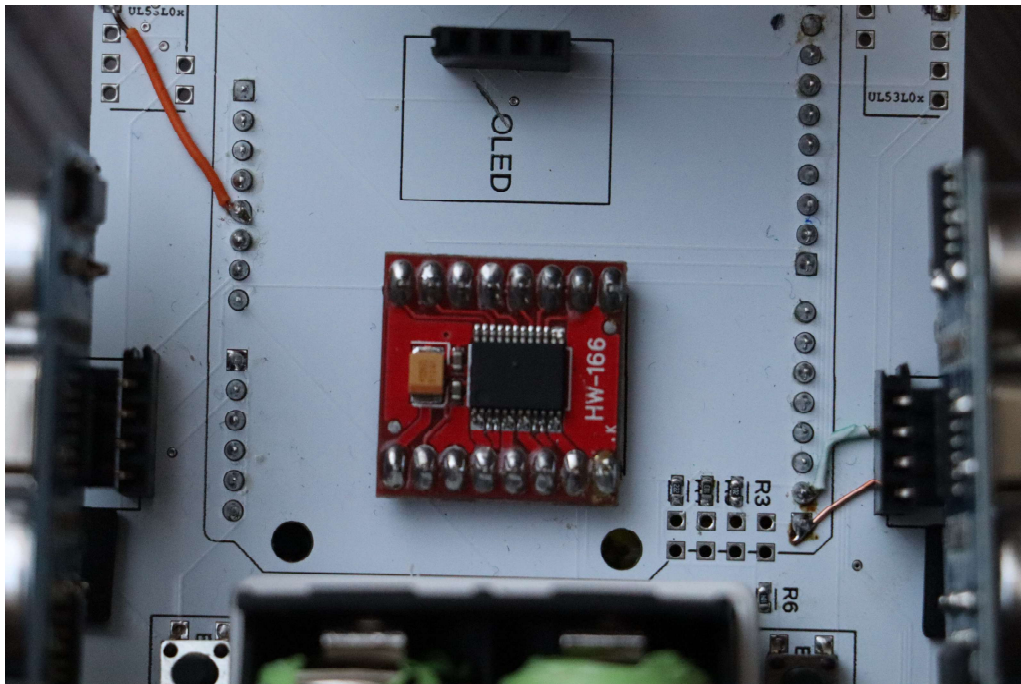




Obrázek 4: Schéma zapojení tlačítek, nepřipojený vodič vede do vstupu Arduina

### 4.3 Driver Motoru

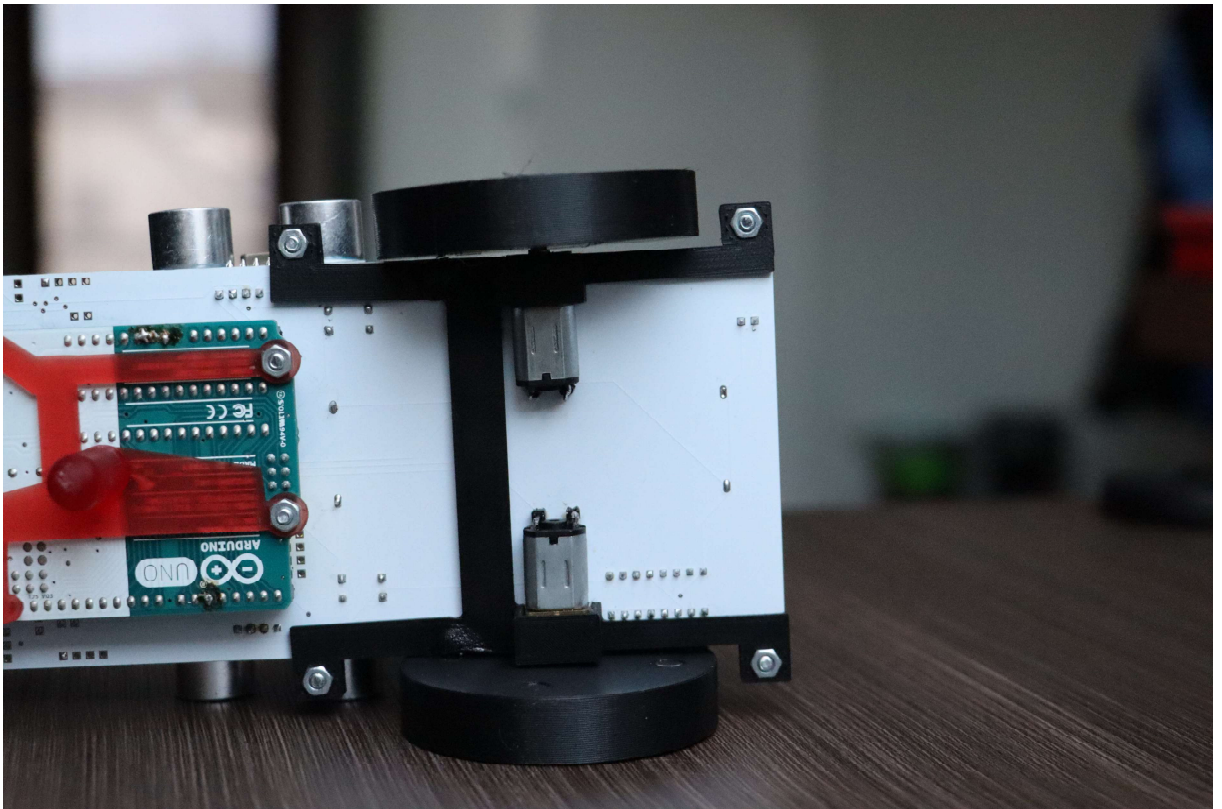
HW-166 je H-můstek se šesti digitálními vstupy. Čtyři z nich mají na starost pomocí jednoduché logiky ovládat směr točení kol, a i jejich brždění. Pro ovládaní rychlosti používá dva vstupy pro PWM signál. Tento můstek je důležitý, protože samotné Arduino není schopné dodávat dostatečné napětí pro motory a také by se při dotáčení kol mohlo poškodit. Samotný čip snese až 15 voltů, ale k této hodnotě se robot nedostane. Dále má samotná deska, na které je čip osazen, pin STBY, což je zkratka pro standby. Jedná se o mód, při kterém se uvede můstek do spánku a spotřebovává méně energie. Problémem bylo, že daný pin musí být připojen k zemi, aby byl deaktivován, a to jsem bohužel neudělal, takže je spoj veden drátem.



Obrázek 5: Detail můstku HW-166

### 4.3.1 Motory

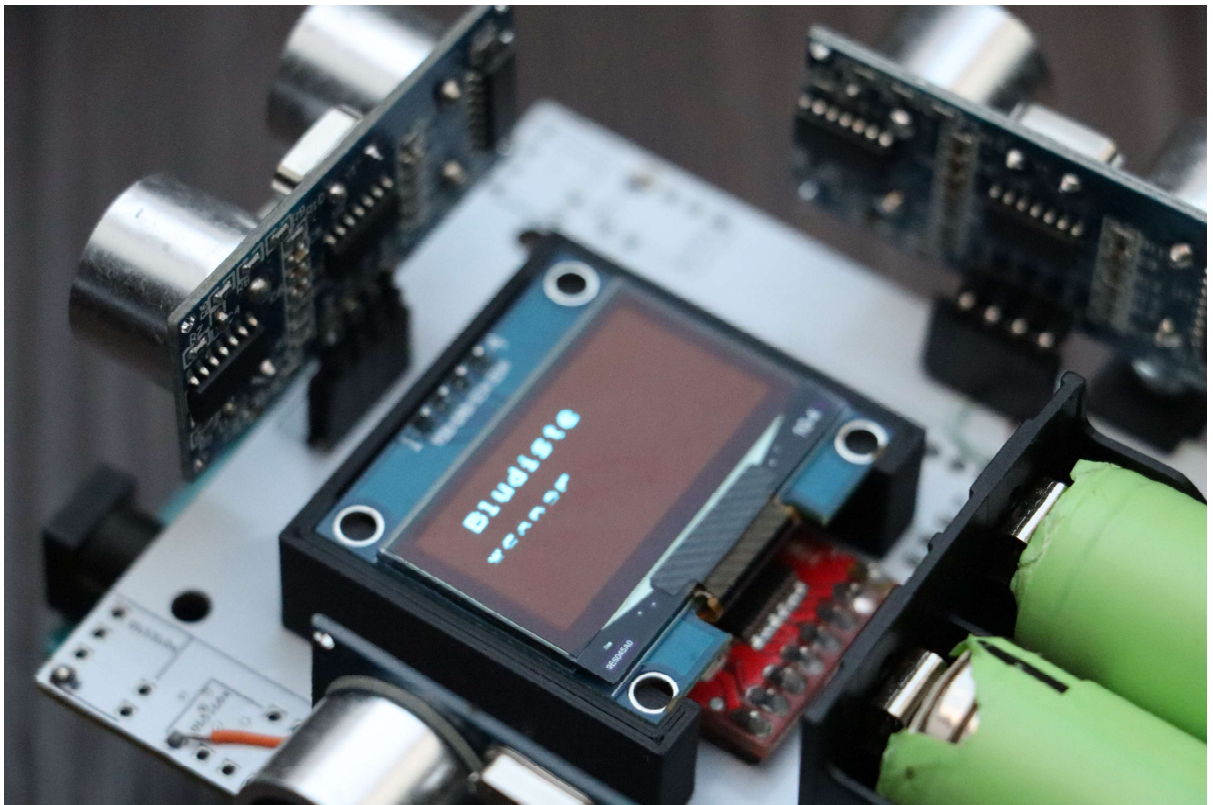
Motory jsem volil dle velikosti a otáček za minutu. Tyto motory jsou typu N20, napájeny šesti až dvanácti volty. Díky kovovým ozubeným převodům jsou nabízeny různé otáčky, já jsem zvolil 60 otáček za minutu. S motory jsem počítal již při návrhu plošného spoje, takže jsou připájeny přímo k desce. Objevil se, ale problém s nadměrným pohybem samotných motorů, Rozhodl jsem se je připevnit pomocí lepidla. To bylo neúspěšné, takže po několika pokusech jsem se rozhodl díl vytisknout na 3D tiskárně.



Obrázek 6: Motory z pohledu ze spodu připevněny černým dílem z 3D tiskárny

## 4.4 OLED

Pro zobrazení menu výběru režimu robota jsem použil OLED displej. Výhodou je nízká cena, dobrá čitelnost a ovládání jednotlivých pixelů. Další výhodou je sběrnice I2C, stejně jako u laserových sensorů. Displej má svojí vlastní adresu, s kterou komunikuje s Arduinem. Při rozlišení 128 x 64 pixelů nebyl problém s přehledností a splýváním znaků. Pro ovládání jsem zvolil knihovnu U8g2, která spolu s sebou nainstaluje knihovnu U8x8, která je pouze textová a posílá pouze znaky. Pro moji aplikaci to bylo vyhovující, protože potřeboval jsem zobrazit tři možnosti a vybírat mezi nimi. Knihovna je sama o sobě velmi lehká a na rozdíl od konkurenčních knihoven opravdu velmi jednoduchá.

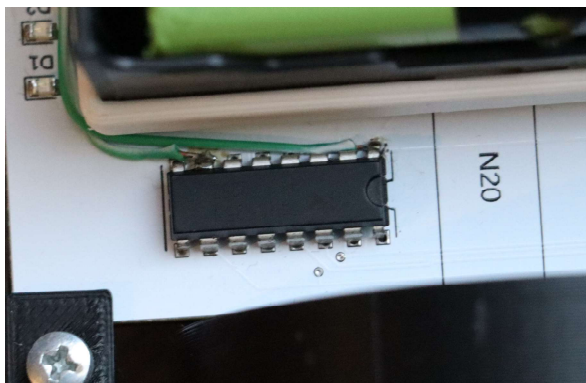


Obrázek 7: Detail OLED displeje v rámečku z 3D tiskárny

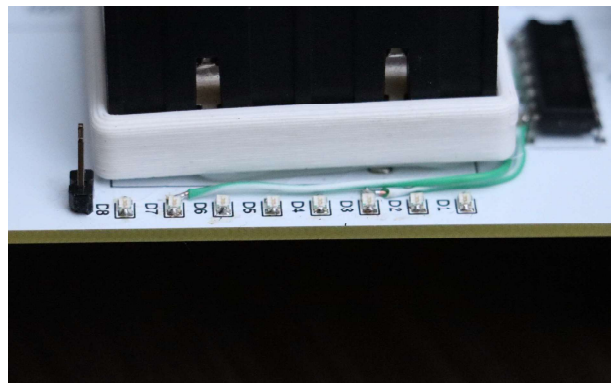
## 4.5 Registr 74HC595 a LED

Posuvný registr (Obrázek 8) je čip s několika klopnými obvody, které posouvají bity pomocí hodinového signálu. Umožňuje tedy zapisovat jedničky a nuly do jeho osmi výstupů pouze ze tří výstupů Arduina. Jeden pin určuje hodnotu, další posouvá bity a poslední je vyšle na výstupy.

V projektu jsem ho použil pro ovládání osmi LED (Obrázek 9) a zároveň jsem ho připojil i k vypínacím vstupům [laserových modulů](#). LED jsou ještě připojeny přes konektor, který je spojuje se zemí, takže umožňuje jejich vyřazení, aby nesvítili, když nejsou potřeba.



Obrázek 8: Detail posuvného registru 74HC595

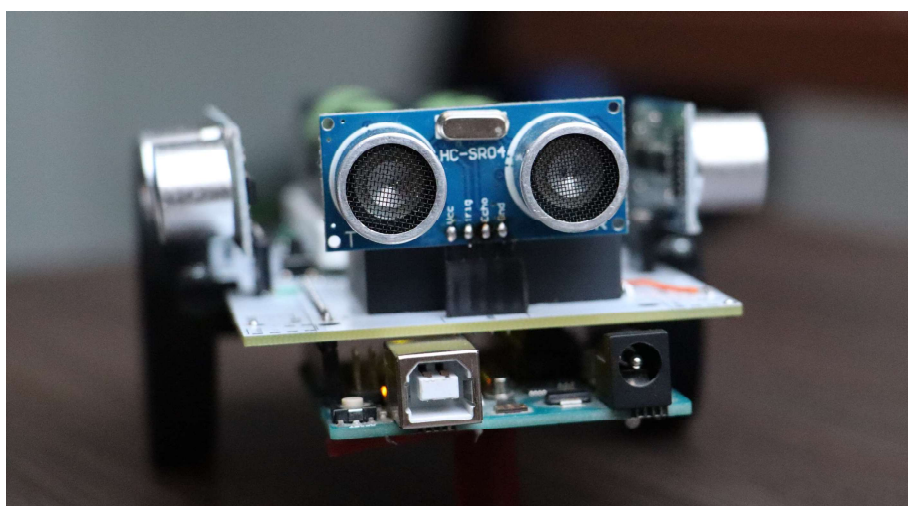


Obrázek 9: Fotografie řady LED a opravy pomocí drátu

## 4.6 Ultrasonické senzory

Sensory HC-SR04 jsou velmi oblíbené a jednoduše se s nimi pracuje. Fungují na principu měření času za jak dlouho odrazí vyslaný zvuk o frekvenci 40kHz. Vyšle se krátký puls do pinu označeného „trig“ a poté se změří čas, dokud se impuls neodrazí od překážky a nevrátí se do sensoru. Tento čas se vynásobí rychlostí zvuku a vydělí se dvěma, abychom dostali vzdálenost k snímanému objektu.

Robot má konektory na tři tyto sensory, jeden vpředu a dva z každé strany. To umožňuje detekci překážek a volných cest při jízdě vpřed. Toto uskupení sensorů je ideální na řešení bludišť a orientaci v prostoru s překážkami.

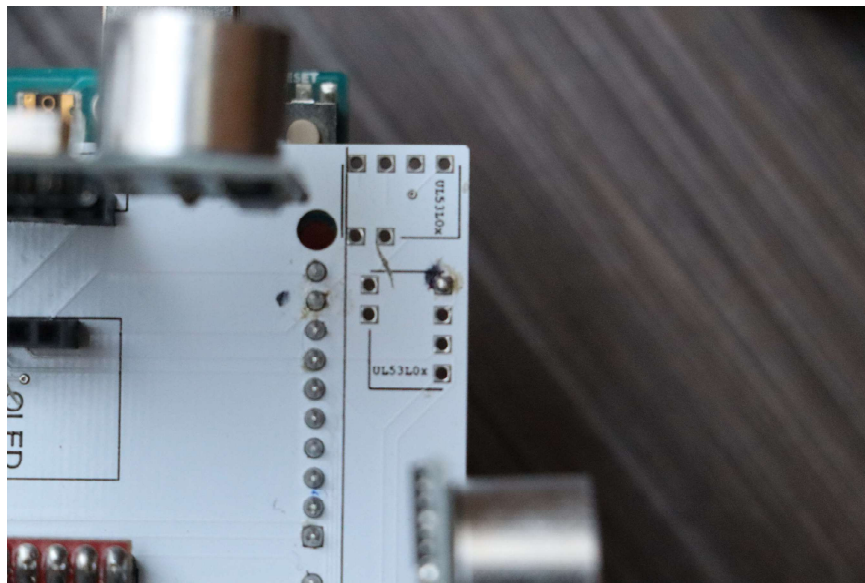


Obrázek 10: Pohled na přední ultrasonický senzor

### 4.6.1 Laserové senzory

Tyto senzory stejně jako OLED displej používají I2C sběrnici. Vzhledem k tomu, že každý senzor má stejnou adresu, jsou senzory vybaveny vypínacím pinem. Tento pin jsem propojil s registrem 74HC595.

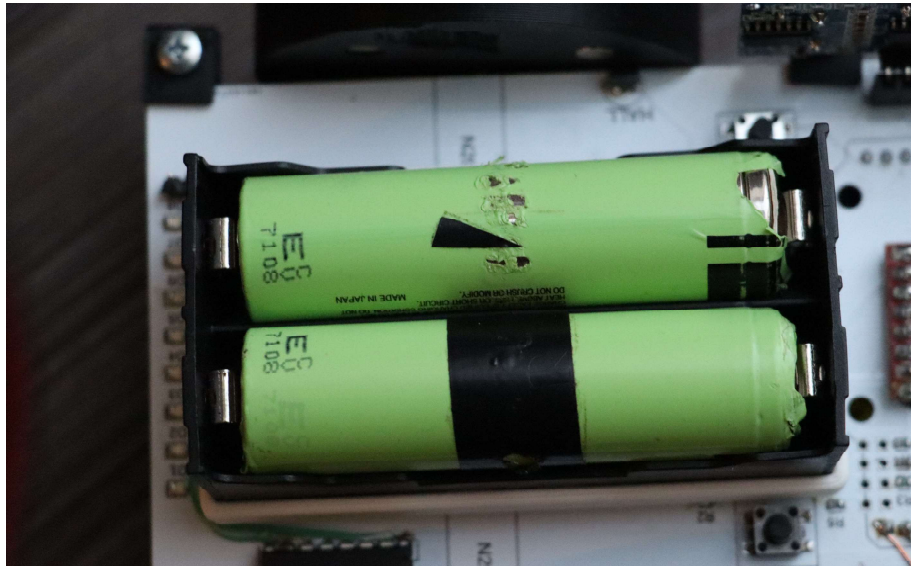
Stejně jako ultrazvukové senzory, měří i tyto senzory čas od vyslání signálu do jeho přijmutí. Nepoužívají k měření vzdálenosti zvuk, ale infračervený laser. Chtěl jsem je použít, protože mohou mít menší úhel detekce, takže neznamenávají překážky, které nejsou hned před robotem. To jsem chtěl využít k ověření, jestli se robot vejde do uličky. Po několika testech se, ale zdáli nespolehlivé, takže jsem je odstranil.



Obrázek 11: Připravený konektor na laserový senzor

## 4.7 Baterie

Zdroj pro robota poskytují dvě vysokokapacitní baterie NCR18650B, každá baterie má 3,35 Ah a nominální napětí 3,7V. Baterie jsou připojeny sériově a poskytují napětí až 8.4V. Na tohoto robota jsou tyto baterie nadměru dostačující. Jsou typu NiMH a mezi vlastnosti patří krom velké kapacity i vyšší hmotnost a oproti LiPO menší vybíjecí proud. Díky jejich dostupnosti a spolehlivosti jsou pro tento projekt vyhovující.



Obrázek 12: Baterie shora

## 4.8 Hallova sonda

K měření otáček kol jsem využil snímače magnetického pole. Jedná se o polovodičovou součástku, která využívá Hallova jevu. Samotný snímač při detekci magnetického pole přivede výstup do nuly. Mezi datovým pinem a napájecím pinem je pull-up rezistor, který naopak v stavu bez magnetického pole přivádí pin do logické jedničky.

Jednou z úprav zapojení této sondy byl právě pull-up rezistor, neboť senzor odmítal pracovat s integrovaným rezistorem Arduina.



Obrázek 13: Hallova sonda vedle kola s magnety

### 4.8.1 Kola

Kola na obrázku 14 ještě nejsou vybaveny pneumatikou, ale na finálním výrobku jsou osazeny pneumatikou z duše z bicyklu. Bez ní vytištěné kolo nemá dostatečnou trakci a jednoduše se zasekává.

V kolech jsou připraveny i díry na magnety, které jsou použity pro snímání otáček. Každé kolo má čtyři magnety, takže pohyb robota může být krokován po čtvrt otáčkách nebo přibližně po 16,5 milimetrech.



Obrázek 14: Kolo s magnety z pravé strany (bez pneumatiky)

## 5 3D TISK

3D tisk je druh výroby při které je materiál přidáván a ne ubírán, jak je tomu u tradičních metod výroby, jako je například CNC obrábění. Projekt používá dva způsoby 3D tisku FDM a LCD.

FDM používá tiskovou hlavu, ve které se taví filament, který je z pravidla z termoplastu. Hlava je různými kombinacemi os naváděna na souřadnice, kde precizně umísťuje na pracovní plochu materiál. Jedná se o nejrozšířenější a nejlevnější tiskárny.

LCD SLA používá displej podsvícený UV světlem k utvrzení pryskyřice. Samotná pracovní plocha je ponořována do této pryskyřice. Při první vrstvě se plocha přitiskne k displeji a po nasvícení každé vrstvy se posune od displeje o nastavené milimetry. Oproti FDM je tento způsob přesnější, ale dražší a vyžaduje čištění a vytvrzení před použitím.

### 5.1 FDM

Materiál je v tomto procesu umísťován po vrstvách. To může být pro některé aplikace nevhodné, protože je poté výrobek slabší v tahu v ose, která je kolmá k vrstvám. Jednou z

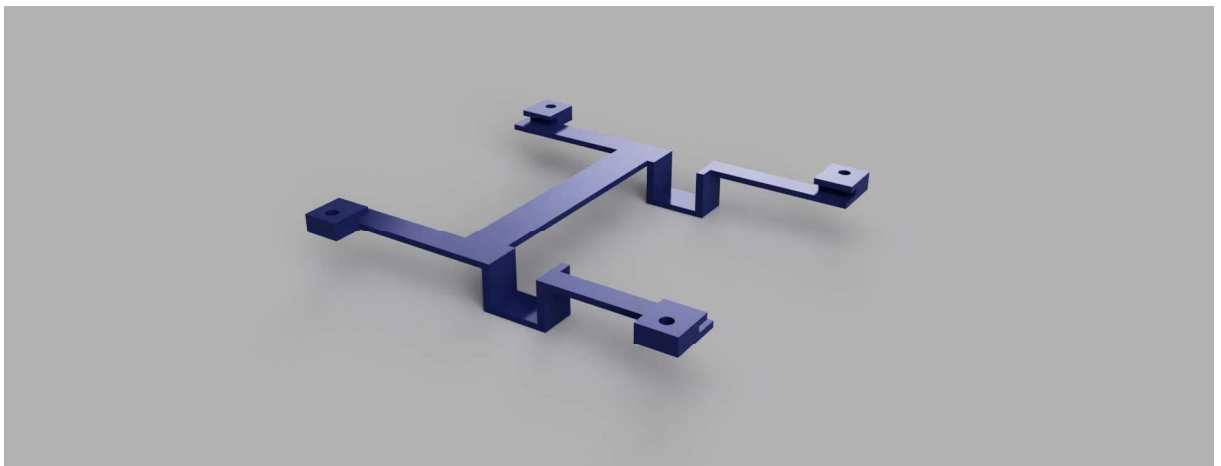


možností, jak tento efekt minimalizovat je zvýšení teploty, tak, aby vrstvy mezi sebou lépe splynuly. Vyhnout se negativním vlastnostem nelze nikdy úplně, takže je s nimi nutné pracovat.

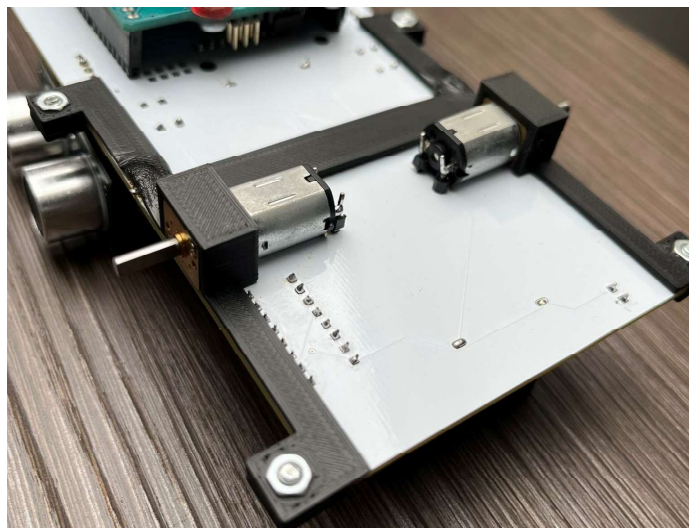
V projektu jsem tímto způsobem vyrobil tři součástky. První byla kola, ty byla na tisk nejméně náročná, protože neobsahovala žádné převisy ani speciální požadavky na pevnost. Druhým objektem byl díl pro uchycení motoru. Zde byl tisk orientován tak, aby díl dosáhl vyšší pevnosti, a to i přes větší množství nutných podpěr. Posledním dílem byl držák na OLED, ten zajišťuje, aby se displej neviklal v konektoru.

Kola jsou na motorech nasazena bez jakéhokoliv přídržného mechanismu a drží pouze pomocí tření. Magnety jsou do plastu natlačeny po lehkém nahřátí horkovzdušnou pistolí.

Uchycení motorů je připevněno pomocí šroubů, které svírají plošný spoj plastovými podložkami. Celý díl se skládá z pěti kusů, čtyři z toho jsou právě podložky. Hlavní kus má v sobě díry na částečné zapuštění matek.



Obrázek 15: Počítačově vygenerovaná vizualizace vytištěného držáku

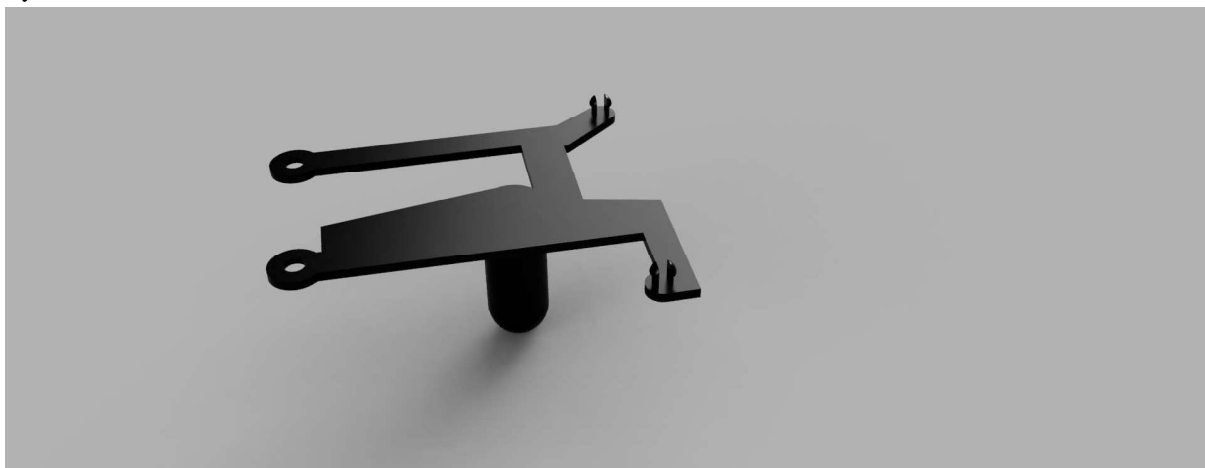


Obrázek 16: Vytisknutý držák motorů

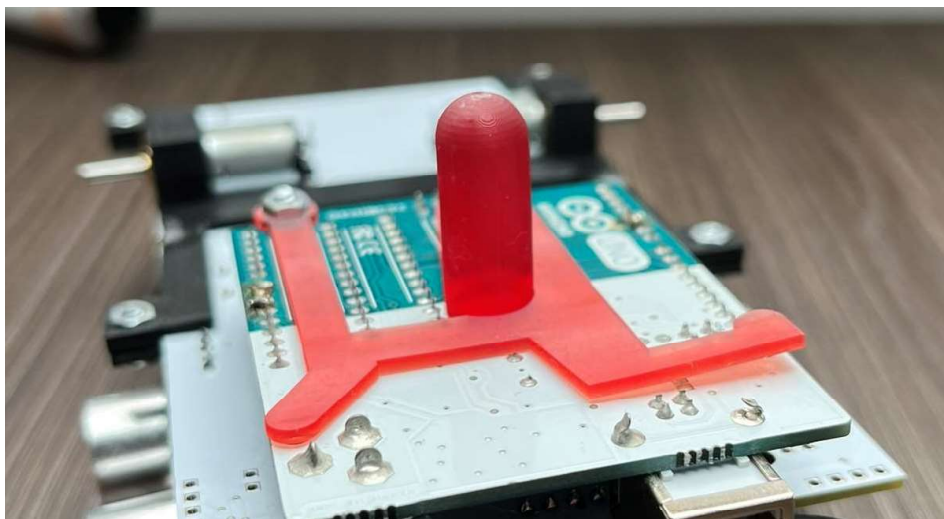
## 5.2 LCD SLA

Pomocí LCD tiskárny je na modelu vytvořena součástka, která funguje jako přední kluzák robota. Je připevněna pomocí dvou šroubků a dvou zacvakávacích pinů. Piny kvůli menší pevnosti pryskyřice mají tendenci se lámat. V pozdějších verzích je problém vyřešen přichycením kolem konektoru.

Na rozdíl od FDM tisku bylo nutné s dílem nakládat v chirurgických rukavicích a před použitím ho omýt v technickém lihu. Díl jsem ještě před instalací nechal pár dní na slunci, aby řádně vytvrdl.



Obrázek 17: Počítačově vygenerovaná vizualizace vytisknutého předního kluzáku



Obrázek 18: Vytištěný kluzák připevněný na Arduino

## 6 SOFTWARE

Celý program je psán ve Visual studiu code. Samotná aplikace neumí pracovat s Arduinem, ale po instalaci pluginu PlatformIO podporuje i seriový monitor. V aplikaci je funkce doplňování příkazů, takže je práce svižná a díky podbarvování i přehledná.

Začal jsem psaním jednotlivých funkcí pro čtení ze senzorů a pro ovládání periférií. Ověřil jsem si tak postupně funkčnost jednotlivých součástek. Kód vlastně slouží spíše jako demonstrace funkčnosti než jako celek se složitým algoritmem pro řešení bludiště.

### 6.1 Vyhodnocování tlačítek a menu

Pro vyhodnocování obou tlačítek jsem použil jeden analogový vstup. V klidu je hodnota ButtVal na nule. Když je stisknuto tlačítko jedna, tak se hodnota změní a bude mezi 900 a 950. Pro tyto hodnoty nabývá proměnná ButR logické jedničky. Pro hodnoty přes 1000 je jednička připisována ButL

```

void Button(){
  ButtVal = analogRead(ButtPin);
  if (ButtVal >= 900 && ButtVal <= 950){
    ButR = 1;
  }
  else{
    ButR = 0;
  }
  if (ButtVal >= 1000){
    ButL = 1;
  }
  else{
    ButL = 0;
  }
}

```

Obrázek 19: Kód pro vyhodnocování tlačítek

Další částí programu jsou logické funkce pro vyhodnocování náběžných hran pro ButL a ButR. Hrana ButR přičítá k proměnné menu jedničku a ButL je použita pro potvrzování.

```

if(prevButR != ButR ){
  if(ButR == 1){
    u8x8.clearDisplay();
    menu++;
  }
}
prevButR = ButR;

if(prevButL != ButL ){
  if(ButL == 1){
    confirm = 1;
  }
}
prevButL = ButL;

```

Obrázek 20: Kód pro detekci náběžné hrany

### 6.1.1 Zobrazování menu na OLED

Tato část programu zajišťuje zobrazování menu na OLED. Podle proměnné s názvem menu se zobrazuje na displeji určitá informace. Vždy se zobrazuje hvězdička u možnosti, která je zvolena. Opakovaným stiskem pravého tlačítka se uživatel dostane na začátek. Samotné potvrzování levým tlačítkem se nachází až ve funkci loop.

```

void Menu() {
  if(menu == 4)menu = 1;
  switch (menu)
  {
  case 1:
    u8x8.setFont(u8x8_font_chroma48medium8_r);
    u8x8.drawString(1,1,"*Bludiste");
    u8x8.drawString(2,3,"Sonar");
    u8x8.drawString(2,5,"Demo");
    break;
  case 2:
    u8x8.setFont(u8x8_font_chroma48medium8_r);
    u8x8.drawString(2,1,"Bludiste");
    u8x8.drawString(1,3,"*Sonar");
    u8x8.drawString(2,5,"Demo");
    break;
  case 3:
    u8x8.setFont(u8x8_font_chroma48medium8_r);
    u8x8.drawString(2,1,"Bludiste");
    u8x8.drawString(2,3,"Sonar");
    u8x8.drawString(1,5,"*Demo");
    break;
  }
}

```

Obrázek 21: Kód pro zobrazování textu na OLED displeji

## 6.2 Pohyb

Ihned po vstupu do této funkce se zapínají oba motory ve směru vpřed. Program postupuje do příkazu while, v něm se počítají náběžné hrany ze hallových sond. Poté co senzor dosáhne požadovaných hodnot, tak se příslušné kolo uvede do klidového stavu. Funkce while je naplněna po dosažení cílových hodnot obou kol.

Pohyb vzad je řešen obdobně, pouze s rozdílem počátečního otáčení vzad.

Zatáčení je řešeno jednoduše, a to úpravou pohybu vpřed. V programu je vynecháno zapínání jednoho kola a while čeká na splnění hodnot jen u jednoho kola.

```

void moveF(int numPulses) {
    leftPulses = 0;
    rightPulses = 0;
    bool prevHal;
    bool Hal;
    bool prevHaR;
    bool HaR;

    digitalWrite(AIN1,HIGH);
    digitalWrite(AIN2,LOW);
    digitalWrite(BIN1,HIGH);
    digitalWrite(BIN2,LOW);

    analogWrite(PWMA,127);
    analogWrite(PWMB,127);

    while (leftPulses < numPulses || rightPulses < numPulses) {
        Hal = digitalRead(HallL);
        if(prevHal != Hal ){
            if(Hal == 1){
                leftPulses++;
            }
        }
        HaR = digitalRead(HallR);
        if(prevHaR != HaR ){
            if(HaR == 1){
                rightPulses++;
            }
        }
    }
    prevHaR = HaR;
    prevHal = Hal;

    if (leftPulses >= numPulses) {
        digitalWrite(AIN1, HIGH);
        digitalWrite(AIN2, HIGH);
    }

    if (rightPulses >= numPulses) {
        digitalWrite(BIN1, HIGH);
        digitalWrite(BIN2, HIGH);
    }
}
}

```

Obrázek 22: Kód pro pohyb vpřed se synchronizací otáček

## 6.3 Ovládání posuvného registru

Pro posuvný registr jsem vytvořil pole dataArray. Jednotlivé LED neodpovídají posloupnosti výstupů registru, ale díky poli mohou vždy rozsvítit LED v pořadí odpovídajícímu rozložení na plošném spoji. Takže když chci rozsvítit LED číslo jedna, tak se opravdu rozsvítí LED označená jedničkou.

Nachází se zde dvě funkce for pro postupné zapínání a vypínání LED. Vždy na začátku se přivede RCLK na nulu, zapíší se hodnoty a opětovným přivedením RCLK na jedničku se tyto hodnoty zobrazí na výstupu registru. Ke konci se ještě přivedou všechny výstupy na nulu.

```
void shiftreg(){
  for (int num = 0; num < 8; num++){
    digitalWrite(RCLK, LOW);
    shiftOut(SER, SRCLK, MSBFIRST, dataArray[num]);
    digitalWrite(RCLK, HIGH);
    delay(100);
  }
  for(int num = 8; num > 0; num--){
    digitalWrite(RCLK, LOW);
    shiftOut(SER, SRCLK, MSBFIRST, dataArray[num]);
    digitalWrite(RCLK, HIGH);
    delay(100);
  }
  digitalWrite(RCLK, LOW);
  shiftOut(SER, SRCLK, MSBFIRST, dataArray[0]);
  digitalWrite(RCLK, HIGH);
}
```

Obrázek 23: Kód pro postupné rozsvěcení a zhasínání LED

## 6.4 Měření vzdálenosti z ultrasonických senzorů

Měření vychází z rychlosti zvuku ve vzduchu. Na začátku je vyslán puls o délce deseti mikrosekund a poté se počítá za jaký čas přijde odražený signál. Výsledný čas se vynásobí 0,034 a vydělí dvěma, protože kdybychom tento krok opomenuli, tak budeme mít cestu zvuku k objektu i od něj.

```
digitalWrite(LTrig, LOW);
delayMicroseconds(2);
digitalWrite(LTrig, HIGH);
delayMicroseconds(10);
digitalWrite(LTrig, LOW);
Lcas = pulseIn(LEcho, HIGH);
Lvzdal = Lcas * 0.034 / 2;
```

Obrázek 24: Kód pro měření vzdálenosti pro jeden senzor

## 7 ZÁVĚR

Projekt splnil mé očekávání, a to hlavně ve spolehlivosti. Na návrhu jsem strávil několik víkendů a na programování obdobně. Krom časové investice se nejednalo ani o finančně náročný projekt. Částka odhadem nepřesáhla 2000 korun. Výsledný výrobek i přes to nabízí mnoho funkcí s kompaktními rozměry. Vizuálně nepůsobí robot nijak nedodělaně. Nabízí jednoduché výměny poškozených dílů, což bych určitě taky přidal k funkcím. Nabídka senzorů je postačující na jednoduché řešení bludišť nebo na podobné úlohy závislé na měření vzdálenosti. Displej přidává dojmu hotového výrobku a díky tlačítkům se i příjemně ovládá. Displej poměrně usnadňuje pochopení algoritmů a hledání chyb v programu. Sám jsem toho při ožívování několikrát využil.

Tento projekt odráží poznatky a zkušenosti ze soutěží, kterých jsem se zúčastnil. Sloužit by určitě mohl jako nástroj na některé podobné soutěže nebo by také by mohl pomoci někomu, kdo s roboty začíná. Cesta od myšlenky k robustnímu hotovému výrobku je náročná, ale výsledek stojí za to.



## 8 POUŽITÁ LITERATURA

- [1] <https://navody.drateg.cz/>
- [2] <https://www.hobbytronics.co.uk/>
- [3] <https://forum.arduino.cc/>
- [4] <https://howtomechatronics.com/>
- [5] <https://www.cirexx.com/>
- [6] <https://www.circuitbasics.com/>
- [7] <https://all3dp.com/>
- [8] <https://www.hubs.com/>
- [9] <https://en.wikipedia.org/>

## 9 SEZNAM OBRÁZKŮ

Obrázek 1: První prototyp plošného spoje.....	7
Obrázek 2: Druhý prototyp plošného spoje .....	7
Obrázek 3: Hotové plošné spoje .....	8
Obrázek 4: Schéma zapojení tlačítek, nepřípojený vodič vede do vstupu Arduina .....	10
Obrázek 5: Detail můstku HW-166 .....	11
Obrázek 6: Motory z pohledu ze spodu připevněny černým dílem z 3D tiskárny .....	12
Obrázek 7: Detail OLED displeje v rámečku z 3D tiskárny.....	13
Obrázek 8: Detail posuvného registru 74HC595 .....	14
Obrázek 9: Fotografie řady LED a opravy pomocí drátu .....	14
Obrázek 10: Pohled na přední ultrasonický senzor .....	14
Obrázek 11: Připravený konektor na laserový senzor .....	15
Obrázek 12: Baterie shora.....	16
Obrázek 13: Hallova sonda vedle kola s magnety .....	16
Obrázek 14: Kolo s magnety z pravé strany (bez pneumatiky).....	17
Obrázek 15: Počítačově vygenerovaná vizualizace vytištěného držáku .....	18
Obrázek 16: Vytištěný držák motorů.....	19
Obrázek 17: Počítačově vygenerovaná vizualizace vytištěného předního kluzáku .....	19
Obrázek 18: Vytištěný kluzák připevněný na Arduino .....	20
Obrázek 19: Kód pro vyhodnocování tlačítek .....	21
Obrázek 20: Kód pro detekci náběžné hrany .....	21
Obrázek 21: Kód pro zobrazování textu na OLED displeji.....	22
Obrázek 22: Kód pro pohyb vpřed se synchronizací otáček .....	23
Obrázek 23: Kód pro postupné rozsvícení a zhasínání LED .....	24
Obrázek 24: Kód pro měření vzdálenosti pro jeden senzor .....	24
Obrázek 25: Ukázka kódu.....	28
Obrázek 26: Obdobu kódu v jazyku Scratch .....	29
Obrázek 27: Ukázka pohybu robota .....	29

## 10 PŘÍLOHA

Na podnět mého konzultanta jsem se rozhodl dodatečně změnit obor práce. Hlavním důvodem byla modulární konstrukce robota a způsob jakým byl kód napsán.

### 10.1 Robot jako učební pomůcka

Myšlenkou je využít robota na základních a středních školách. Na základních školách by robot mohl být programován jazykem Scratch. Vzhledem k tomu, že jsou funkce pro senzory a pohyb již napsané, tak by byla aplikace jednoduše realizovatelná. Pro střední školy by stačilo schéma se zapojením a základní popis funkcí a limitů.

Všechny senzory jsou odnímatelné a je jednoduché je vyměnit nebo odebrat. Robot bude plně fungovat i bez nich. V základní sestavě stačí zapojit Arduino a přidat baterie, zbytek je napevno připojen.

### 10.2 Možnosti programování

Programovací jazyk Scratch je programovacím jazykem grafickým. Samotné příkazy by byly napsané, jako například pohybové příkazy. Žáci by tak nemuseli řešit, jak se ovládá můstek pro motory, ale pouze by dali příkaz, aby robot jel kupředu a počet kroků.

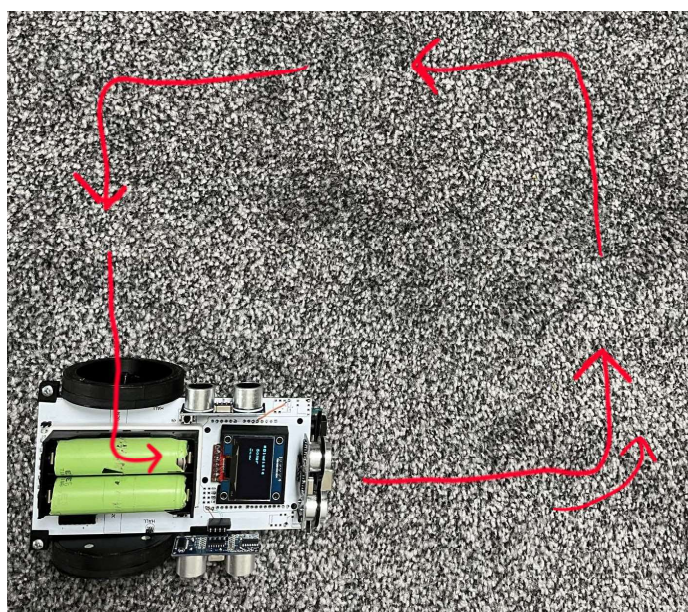
Jako příklad jsem si připravil jednoduchou ukázkou kódu, kde robot objede čtverec. Program je napsán v textové podobě, ale pro představu to stačí. Dále následuje obdoba tohoto kódu v jazyku Scratch.

```
void loop() {  
  moveF(5); //pohyb kupředu 5 kroků  
  moveL(2); //otočení pravého kola o 2 kroky tzn. otočení doleva  
  moveF(5);  
  moveL(2);  
  moveF(5);  
  moveL(2);  
  moveF(5);  
}
```

Obrázek 25: Ukázka kódu



Obrázek 26: Obdoba kódu v jazyku Scratch



Obrázek 27: Ukázka pohybu robota

Obdobným způsobem je možno operovat i senzory, pouhým zavoláním příkazu pro měření vzdálenosti se do proměnných zapíší hodnoty.

Pro střední školy by stačilo blokové schéma se zapojením periférii. Studenti by připojili požadované senzory a našli si kam jsou připojeny a můžou začít programovat. Na schématu by bylo blokové schéma s perifériemi na sběrnici a senzory a jejich zapojení do Arduina.

Jako návod k použití by sloužila tato dokumentace, je v ní vše potřebné pro orientaci v projektu, se schématy a vysvětlením, jak fungují jednotlivé senzory. Robot by mohl být distribuován jako stavebnice nebo jako hotový výrobek. Přiloženo by bylo zapojení a návod k použití. Projekt by díky tomu mohl mít bohaté využití, jak v rukou studentů, tak v ruce profesorů.