Vyšší odborná škola informačních studií a Střední škola elektrotechniky, multimédií a informatiky

Novovysočanská 280/48, Praha 9

Maturitní práce

Animatronické slechy

Miroslav Novan

SE.4.

2018

Čestné prohlášení

Prohlašuji, že svou maturitní práci "Animatronické slechy" jsem vypracoval(a) samostatně pod vedením vedoucího maturitní práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor(ka) uvedené maturitní práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne _____

Animatronické slechy

Abstrakt

Slechy se budou pomocí naprogramovaného Arduina otáčet za zdrojem zvuku. Budou tak znázorňovat podobné vnímání normálních slechů u psovitých šelem. Na konstrukci budou 3 mikrofony, které je nastaví do určité polohy. Mikrofony přijímají hluk z okolí a ten jde přes analogový vstup do Arduina. Tam se určí výška dB a podle toho se pak slechy otočí.

Pokud zvuk přijde z pravé strany, natočí se pravý slech, pokud přijde zvuk z

Levé strany natočí se levý slech, a pokud přijde zvuk zepředu, oba slechy se natočí dopředu. K sestavení použiji servomotory SG90 , 3 mikrofony a Arduino Nano.

Animatronic ears

Abstract

The ears will be rotated by the programmed Arduine as the sound source. They will

To illustrate similar perception of normal ears in canine beasts. There will be 4 microphones on the structure to set them to a certain position. The microphones receive noise from the surroundings and they move through the analog input to the Arduin. The dB height is determined and then rotates accordingly.

If the sound comes from the right side, the right ear will rotate if the sound comes from the left side turns to the left ear, and if the sound comes from the front, both ears are turned to the front. For assembly I will use servo motors SG90, 3 microphones and Arduino Nano.

1 Úvod **7**

2 Teoretická část **7**

2.1 Polohy slechů 7

2.2 Anatomie slechů 8

2.3 Arduino Nano 9

2.4 Zvuk 10

3 Praktická část **11**

3.1 Programování 11

3.2 Servomotory 13

3.3 Schéma 14

3.4 Sestavení výrobku 15

4 Závěr **17**

5 Seznam použitých zdrojů **18**

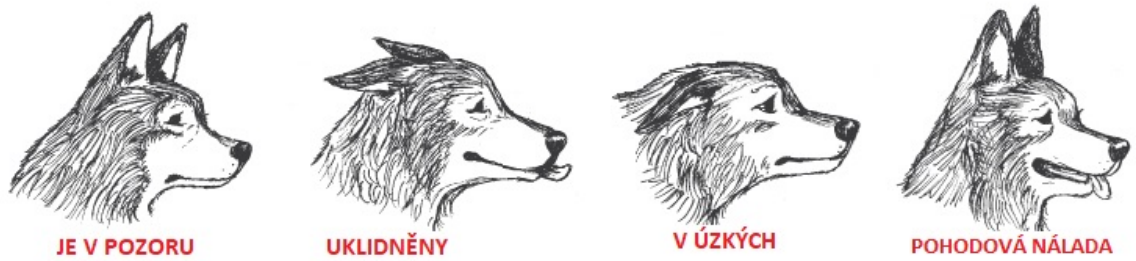
1 Úvod

Psi dokážou slyšet mnohem vetší zvukové frekvence než lidé a to až 60 000 Hz (samozřejmě záleží na plemeni a stáří jedince). Také dokážou vnímat zvuk až k 40 Hz a to lidé už také neslyší. Samozřejmě že to bývá u každého organizmu trochu jinak. Většina psů dokáže svýma ušima podvědomě hýbat skoro všemi směry, podle toho odkud daný zvuk přichází. A proto se pokusím v mé práci o to, abych dané chování psích uší napodobil jak nejlépe to půjde.

2 Teoretická část

2.1 Polohy slechů

Psi také dokážou svými slechy vyjadřovat i náladu. Pokud má pes své slechy sklopené dozadu znamená to, že je v klidu. Pokud má slechy přímo vztyčené, znamená to, že naslouchá zvukům z okolí. A když jsou slechy sklopené dolů tak to znamená, že je zřejmě v úzkých. Avšak psi dokážou své slechy podvědomě částečně otočit pomocí ušních boltců, aby zjistili , odkud přichází zvuk signálu.

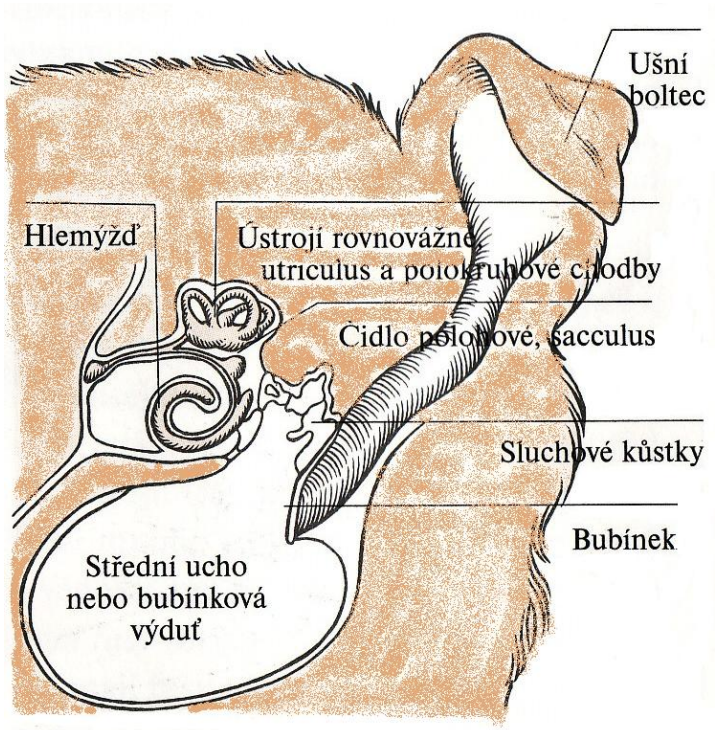


Vysvětlení pozic psích slechů

http://www.tailsfromthelab.com/tag/body-language/

2.2 Anatomie slechů

Lidské a psi uši se anatomickou stavbou až tak moc neliší. Sluchový orgán se dělí na vnější ucho, střední ucho a vnitřní ucho. Vnější ucho se skládá ze zvukovodu a ušního boltce, který je tvořen chrupavkou a je v něm množství svalů a je pokryt kůží. Boltce zachytávají zvuky z okolí a přeposílají je zvukovodem do středního ucha. Jako u člověka rozhraní vnější a střední ucha tvoří bubínek, který se chvěje s příchodem zvuků z vnějšího ucha. Chvění je předáno sluchovým kůstkám (kladívko, kovadlinka, třmínek) a chvění se nese do středního ucha. Nakonec doputuje do kostního hlemýždě, ze kterého se zvuk šíří zvukovým nervem do sluchové oblasti v mozku.



Průřez uchem

https://www.slideserve.com/nixie/r-mci-projektu

2.3 Arduino Nano

Jde o malý jednodeskový počítač z otevřené platformy Arduino (Opensource), to znamená, že má otevřený zdrojový kód. Je založeno na mikrokontrolerech od firmy Atmel.

Zde je pár technických údajů:

Mikroprocesor	ATmega328
Architektura	AVR
Provozní napětí (logická úroveň)	5V
Vstupní napětí (doporučeno)	7-12V
Počet digitálních I/O pinů	22 pinů, z toho 6 s PWM
Počet analogových vstupů	8 pinů
Proudové zatížení na 1 pin	40 mA
Flash paměť	32 KB
SRAM	2 KB
EEPROM	1KB
Rychlost hodin	16 MHz

2.4 Zvuk

Jedná se o mechanické vlnění v prostředí. Frekvence, které člověk vnímá, jsou značně individuální a leží v intervalu přibližně 16 Hz až 20 000 Hz. Mechanické vlnění mimo tento frekvenční rozsah sluchový vjem nevyvolává, přesto se někdy také označuje jako zvuk.

Jak vzniká zvuk?

- úderem
- drnkáním
- smýkáním
- rychlým pohybem
- prouděním vzduchu okolo ostré hrany
- prudkou změnou tlaku

Frekvenční rozsah u lidí

Většina lidí slyší zvuky od 16Hz do 16kHz, avšak s rostoucím věkem se horní hranice snižuje. Proto, aby byla řeč srozumitelná je potřeba 2-4kHz , v těchto rozmezích je lidské ucho nejcitlivější.)

Barva zvuku

Zvuky se mohou lišit odlišným zabarvením. Barva zvuku je určena počtem vyšších harmonických tónů ve složeném tónu. A jejich amplitudami. Podle barvy zvuku rozeznáváme hudební nástroje od hlasů lidí.

Periodické kmity jsou tvořeny složkami, jejichž frekvence jsou celistvé násobky frekvence základního tónu. Má-li harmonická frekvence dvojnásobný počet kmitů proti kmitu základnímu, jde o druhou harmonickou. Obecně platí, že liché násobky kmitočtu zvuk zostřují (např. u žesťových hudebních nástrojů), sudé násobky harmonické frekvence zvuk zjemňují (např. dřevěné dechové nástroje).

Vlastnosti zvuku

Výška zvuku je dána jeho frekvencí, čím je vyšší výška, tím vyšší je frekvence. U jednoduchých tónů s harmonickým průběhem určuje jejich frekvence absolutní výšku tónu. Relativní výška tónu je důležité pro subjektivní hodnocení zvuku, což je podíl frekvence daného tónu vůči frekvenci referenčního tónu. Jako základní tón bylo určeno 440 Hz. Technici používají jako základní referenční tón 1000Hz (což je tedy celý 1kHz).

Záznam zvuku

První zvuková stopa byla zaznamenána na přístroji zvaném fonograf a tato stopa byla zaznamenána známým zesnulým vědcem Thomasem Alvou Edisnem. Fonograf funguje tak že na voskový válec se pomocí jehly zaznamenává stopa, avšak jelikož byly z vosku jejich trvání nebylo věčné. Tyto fonografy byly předchůdci dnešních gramofonů, které se opět vrací do trendů dnešního života. Tak roku 1895 přišel, Emile Berliner s gramofonem poháněným klikou. Později byly gramofony už poháněny elektrickým proudem a dala se u nich zvyšovat i hlasitost.

3 Praktická část

3.1 Programování

K úspěšnému chodu anomatronických slechů je zapotřebí správně napsaný program. V mém případě šlo o to, abych dostal z analogového signálu z mikrofonů, dostal digitální výsledek a tím pohnul servama.

A takhle vypadá program:

```
#include <Servo.h>
        //Definice mikrofonu
#define hlukmax 500
Servo myservo[2];
int pos[] = {0, 0, 0, 0};
int data[5];
int maximum[4];
int hluk[5];
int pins[] = {A0, A1, A2, A3};  //Piny pro mikrofony analog
int pind[] = {9, 10};   //Piny pro serva digital
long casPreruseni;
int air = 0;
```

```
void setup() {
  Serial.begin(9600);
  // načtení první hodnoty jako maxima
  for(int i = 0; i < 4; i++){
     maximum[i] = analogRead(pins[i]);
  }
  for(int i = 0; i < 2; i++){
     myservo[i].attach(pind[i]);
  }
}
void loop() {
    for(int i = 0; i < 4; i++){
        data[i] = analogRead(pins[i]);
       Serial.println(data[i]);                // tento příkaz slouží  k tomu aby se nám ukázaly hodnoty v seriál printu
        if ( data[i] > maximum ) {
           maximum[i] = data[i];
        }                                      // zde se porovnávají hodnoty zvuku, podle toho jaké si maximum určíme
        if( data[i] > hlukmax ) {
           hluk[i] = data[i];
        }else{
           hluk[i] = 0;
        }
    }
    hluk[4] = hluk[0];
    data[4] = data[0];
                int dir = 0, air = 0;
    for(int i = 0; i < 4; i++){
        if(hluk[i] != 0 && hluk[i + 1] != 0){
           float a,b;
           a = hluk[i] / maximum[i];
           a = a * 100;
           b = hluk[i+1] / maximum[i+1];
           b = b * 100;
           //80 / 2 = 45 každý má maximálně 45
           float m1,m2;
           m1 = (a / 100) * 45;
           m2 = (b / 100) * 45;
           float rot = m1 + m2;
           air = dir + rot;
        }else if(hluk[i] != 0){
           air = dir;
        }
        dir+= 80;              // zde potom se podle přijatého zvuku a zjištění z jakého mikrofonu signál přišel, se
                                  servomotory po 45 stupních otočí
    }
    for(int i = 0; i < 2; i++){
       Serial.println(air);
    }
}
```

3.2 Servomotory

Servomotor, zkráceně servo, je motor pro pohony (většinou elektrické), lze u něj nastavit přesnou polohu natočení osy. Ovládají se jím například posuvy u CNC strojů, nastavení čtecí hlavičky u pevného disku. Všechny RC (Radio control) modely používají malá modelářská serva.

V širším slova smyslu je servomotorem jakýkoli motor sloužící k motorickému řízení polohy nahrazující práci člověka. Může to být:

- elektromagnet,
- elektromotor,
- pneumatický či hydraulický píst
- nebo i topný odporový drát na bimetalu.

<u>Řízení servomotorů</u>

O poloze zařízení lze usuzovat bez zpětné vazby, např.:

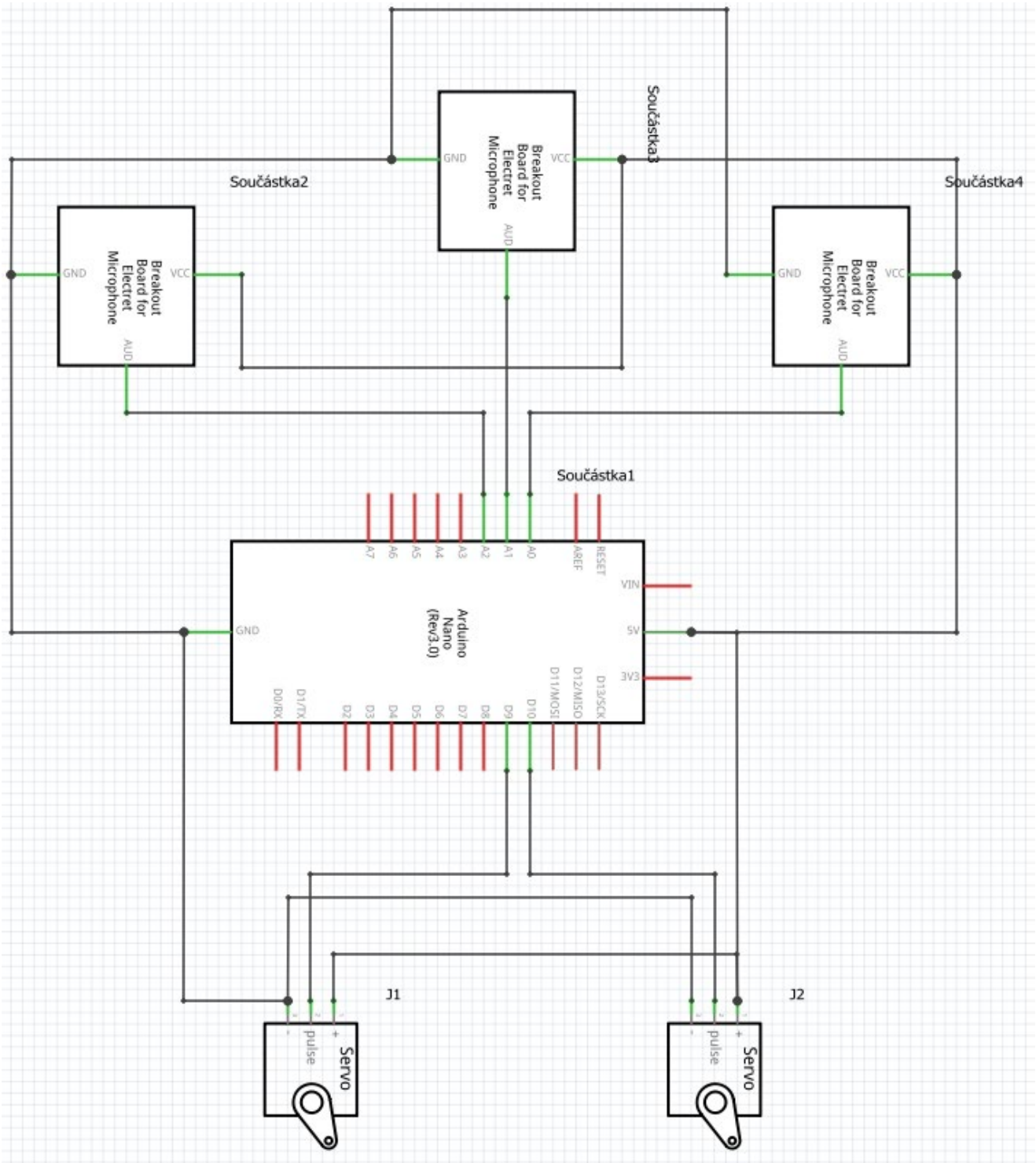
- podle počtu impulzů vyslaných do krokového motorku,
- podle času po který byl pohon zapnutý.

U servomotorů jsou snímače pro nastavení polohy. Signál těchto čidel lze využít k dalšímu řízení pohonu například, když se servomotor vypneme tak se pomalu vrací do své startovací polohy. Zavedením lineární záporné zpětné vazby můžeme servomotorem řídit polohu zařízení v celém rozsahu pracovní dráhy. Je k tomu zapotřebí regulátor.

Poloha hřídele servomotoru bývá zjišťována elektricky pomocí fotoelektrického snímače (encoder) nebo pomocí rozkladače (selsynu). Pro levné aplikace lze použít optické snímání pomocí kódového kotoučku či proužku, gray code. Nedoporučuje se potenciometr.

Signál snímače polohy je přiveden pomocí zpětné vazby na regulátor, který porovnává skutečnou polohu motoru s žádanou polohou. Na základě rozdílu žádané a skutečné polohy regulátor (často velmi složitý) řídí měnič a tak nastavuje motor na žádanou polohu.

3.3 Schéma

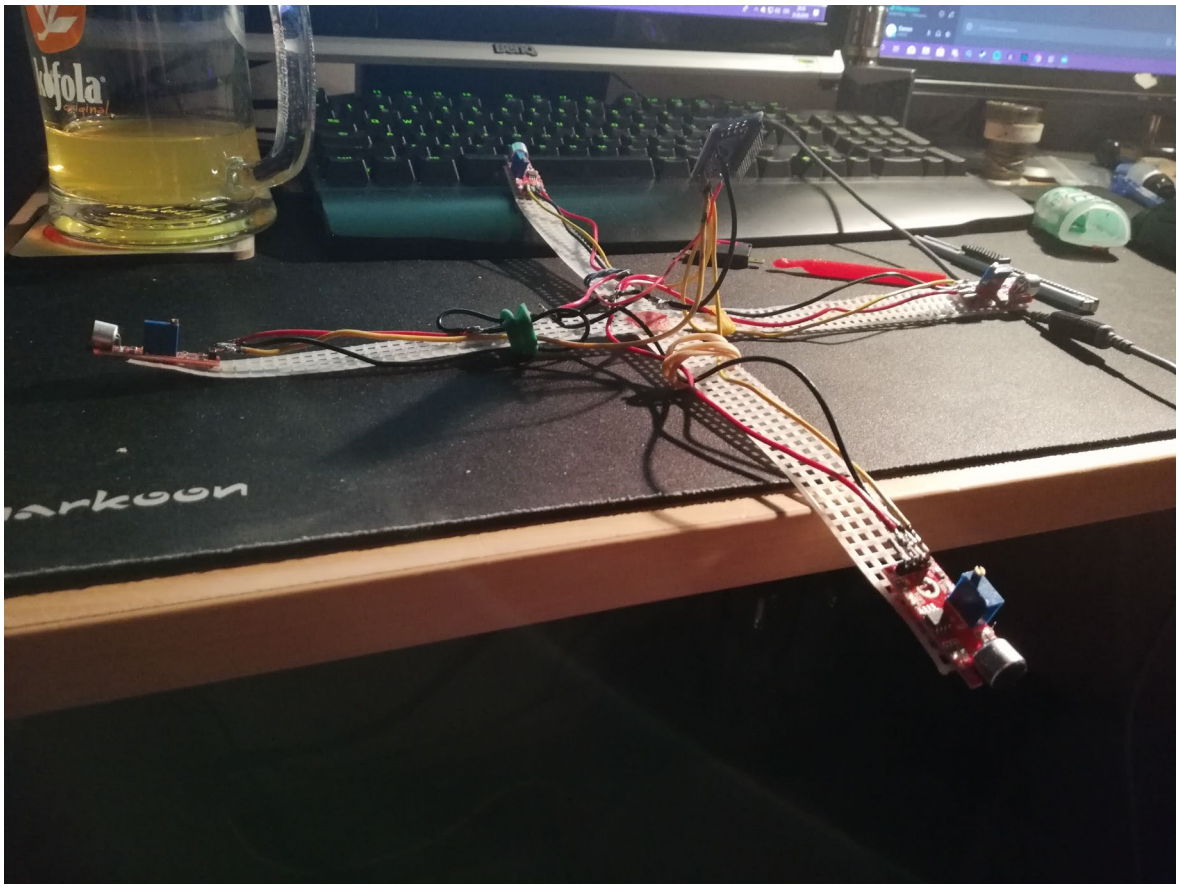

Součástka1
Součástka2
Součástka3
Součástka4
Breakout Board for Electret Microphone
GND
VCC
AUD
Arduino Nano (Rev3.0)
A7
A6
A5
A4
A3
A2
A1
A0
AREF
RESET
VIN
5V
3V3
D0/RX
D1/TX
D2
D3
D4
D5
D6
D7
D8
D9
D10
D11/MOSI
D12/MISO
D13/SCK
J1
J2
Servo
pulse

3.4 Sestavení výrobku

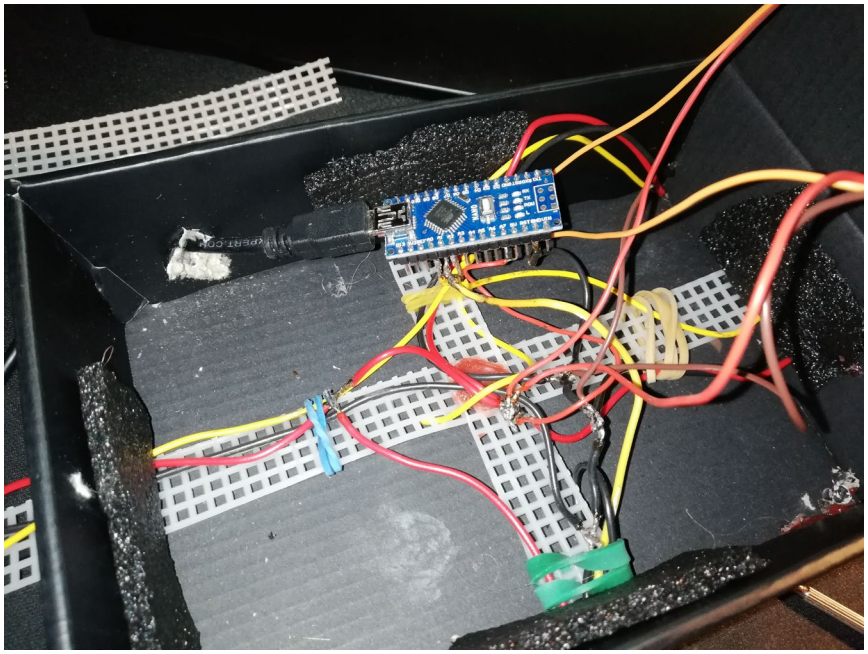
Začal jsem s křížkovou vyšívací mřížkou, která je dobrý materiál při konstrukci podobných věcí. Z něj jsem udělal kostru pro mikrofony, umístil každý na jeden konec a přilepil.

Potom přivedl k mikrofonům napájecí kabel, minusový kabel a kabel s analogovým výstupním signálem (ty se žlutou izolací). Tyto 3 žluté kabely jsem přivedl na analogové piny A0,A1,A2. Napájecí kabely na +5Vpin a minusové na GND pin.

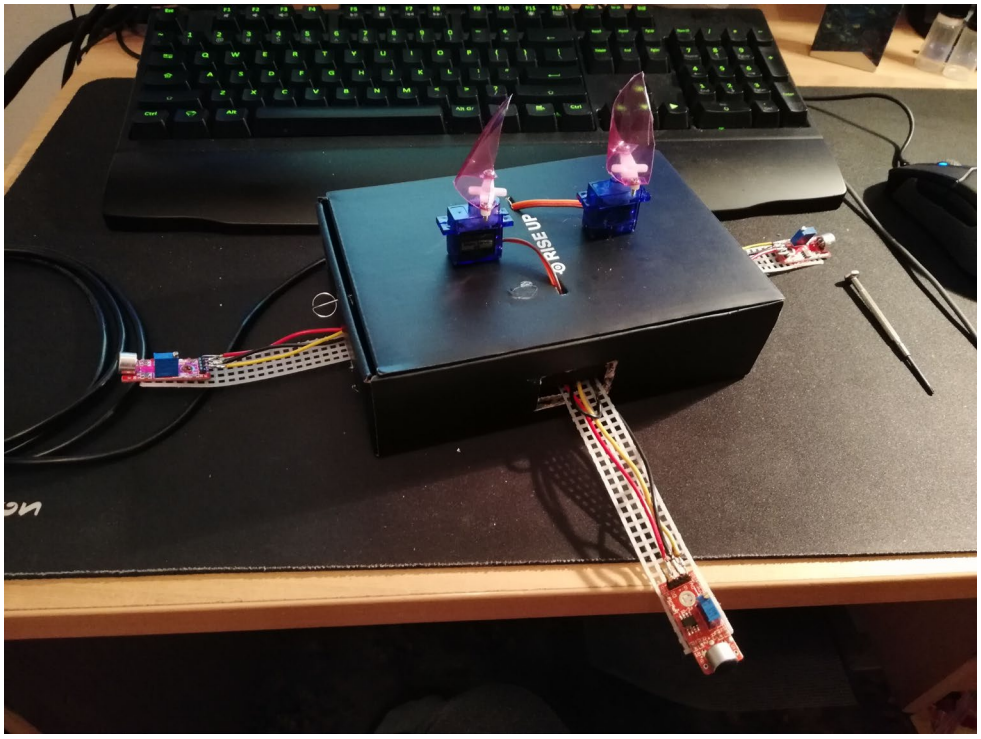
Vše jsem umístil do krabičky a pokusil se o ladný kabel management. Na víko krabičky jsem umístil obě Serva. Levé servo na D9 pin a pravé na D10. Opět jsem spojil napájecí a minusové kabely s piny +5V a GND.



-Toto je fotka prvního návrhu, který měl navíc jeden mikrofon, který byl potom odstraněn, aby byl výrobek více reálný a dal se tak víc přirovnat k psím uším.



Toto je fotka vnitřku výrobku. Je zde hodně kabelů, které jsou připevněné ke kostře gumičkami. Jelikož Arduino má pouze dva GND piny, musel jsem si vymyslet vlastní způsob svedení drátků na GND pin.



-Foto hotového výrobku, kde už jsou vidět i Servo motorky s plastovými uši.

4 Závěr

Při mé práci jsem se setkal s mnoha problémy, hlavně když jsem zjistil, že obyčejné mikrofony mi jsou k ničemu a musím použít mikrofony, které už mají u sebe nějaký odpor a vlastní kondenzátor. Potom co jsem konečně sehnal vše, co potřebuju. Jsem začal testovat mikrofony, abych získal zvuk okolí a po tlesknutí zaznamenat hodnotu. Takže jsem měl zvuk ticha a zvuk tlesknutí. Teď už jen stačilo jen pomocí funkce for a if rozchodit obě Serva tak, aby se tlesknutím otočily na mikrofon, na který přišlo tlesknutí jako první. Bohužel tlesknutí je moc silné, a jelikož zvuk se šíří prostorem jako vlna, tak stejně dochází k tomu že se Serva otočí tam a zpátky. Je tu ještě jeden problém a to ten že pokud by se výrobek testoval někde v hlučném prostředí musela by citlivost mikrofonů změněna. A v programu opět měřit nejvyšší možné změny zvuku. Až nakonec by bylo skoro nemožné naměřit ve velmi hlučném prostředí zaznamenat nejvyšší možnou změnu hluku. Ale zase na druhou stranu by se dalo v tichém prostředí zjistit přímo výborně zjistit hluk, a odkud přichází, avšak stejně by se předtím opět musela změnit citlivost. Proto při ukázce před komisí si budu muset nejdřív seřídit všechny mikrofony, až potom bude zařízení připraveno zachytávat tlesknutí.

5 Seznam použitých zdrojů

Literatura

PROCHÁZKA, Zdeněk. Chov psů. Litomyšl: Paseka, 2005. 314 s. ISBN 80-7185-768-8. Kapitola 3.1. biologie a anatomie psa, s. 41-42. (česky)

KOLEKTIV AUTORŮ. Koldův atlas veterinární anatomie. Praha: Grada Publishing, 1999. 704 s. ISBN 80-7169-352-9. (česky)

ČERNÝ, Hugo. Veterinární anatomie pro studium a praxi. Brno: Noviko a.s., 2004. 528 s. ISBN 80-86542-05-X. (česky)

BUDRAS, Klaus-Dieter; FRICKE, Wolfgang; MCCARTHY, Patrick, a kol. Anatomy of the Dog (Vet): An Illustrated Text. Hannover: Schlutersche, 28.9.2007. 224 s. ISBN 978-3-89993-018-4. (anglicky)