



## **Středoškolská technika 2018**

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

### **Dopravník s rozlišovačem barev**

Tomáš Vladyka

VOŠ, SPŠ a JŠ Kutná Hora  
Masarykova 197



VYŠŠÍ ODBORNÁ ŠKOLA, STŘEDNÍ PRŮMYSLOVÁ ŠKOLA  
A JAZYKOVÁ ŠKOLA S PRÁVEM STÁTNÍ JAZYKOVÉ ZKOUŠKY

# **PRAKTICKÁ MATURITA**

## **Dopravník s rozlišením barev**



## ZADÁNÍ

PRAKTICKÉ MATURITNÍ ZKOUŠKY Z ODBORNÝCH PŘEDMĚTŮ

studijní obor: Elektrotechnika, zaměření Automatizační technika

Přijetí a jméno žáka: Tomáš Vladyka

Třída:

E4C

Školní rok:

2017/2018

Téma č.:

A1820

Název tématu:

Dopravník s rozlišením barev

Minimální rozsah:

10

Konkrétní úkoly, které jsou v práci řešeny:

Sestavení modelu zařízení schopného třídit předměty podle barev. Řídicí jednotka na bázi Arduina, volba vhodného modulu, vytvoření programové aplikace pro mobilní zařízení.

Termín odevzdání:

3. 4. 2018

Vedoucí práce: Ing. Stanislav Moravec

podpis

.....  
*Moravec*

Oponent: Ing. Bc. Pavel Stejskal

podpis

.....

Zadání schváleno dne: 8. 9. 2017

**Čestné prohlášení:**

Prohlašuji, že jsem praktickou maturitní práci vypracoval samostatně a použil jsem pouze literatury uvedené v soupisu.

Souhlasím s trvalým umístěním mé práce ve školní knihovně, kde bude k dispozici pro potřeby školy.

V Kutné Hoře dne

podpis

## **Anotace**

Cílem mé maturitní práce bylo navrhnout a zprovoznit linku, která by rozpoznala barevné spektrum objektu (v tomto případě barevného míčku) a na základě tohoto údaje správně roztřídila objekty do předem připravených zásobníků. Tato linka se dá vzdáleně ovládat přes aplikaci v mobilu nebo tabletu se systémem Android přes Bluetooth. Tato aplikace také může vygenerovat a poslat tabulku ve formátu .csv s počty rozřazených objektů. Celá linka je realizována pomocí Arduina a kompatibilních periférií.

## **Annotation**

The goal of my graduation work was to design and put into practice a line that would recognize the color spectrum of an object (in this case a color ball) and based on this data correctly sort the objects into prepared containers. This line can be remotely controlled via an app on a mobile or an Android tablet via Bluetooth. This app can also generate and send a .csv table with the numbers of sorted objects. The whole line is realized using Arduino and compatible peripherals.

## Obsah

Čestné prohlášení:.....	3
Anotace.....	4
Annotation.....	4
Obsah.....	5
Úvod.....	6
Detektor barvy TCS230.....	6
Obecná charakteristika	
Arduino detektor barvy je modul, kterým lze detekovat barvy v jeho blízkosti. Na modulu se nachází senzor TCS230, který se skládá ze 64 fotodiod s různými barevnými filtry, a čtyř bílých LED diod pro osvětlení snímaného předmětu. Díky této kombinaci je tedy možné zjistit barvu zkoumaného předmětu, kdy po osvětlení bílými LED diodami se světlo odrazí zpět do senzoru TCS230 a ten dokáže po konfiguraci uživatelem na svém výstupu vygenerovat frekvenci výskytu barevných složek RGB. Pokud tedy například přiložíme červený předmět, odražené světlo bude obsahovat nejvíce červenou složku barevného spektra, ta vygeneruje nejvíce proudu skrze fotodiody s červeným filtrem, který je následně převeden na frekvenci. Čím vyšší výskyt dané barvy, tím nižší je jeho frekvence na výstupu.....	6
Realizace maturitní práce.....	7
První prototyp.....	7
Druhý prototyp.....	9
Finální produkt.....	10
Aplikace pro Android.....	12
Výběr programovacího prostředí.....	12
Program v MIT App Inventoru.....	13
.....	14
Závěr.....	14

## Úvod

V rámci své maturitní práce jsem se zabýval problematikou automatizované detekce barev. Jako řídicí systém jsem si vybral Arduino Uno kvůli jeho jednoduchosti a cenové dostupnosti. Pro rozeznávání barvy jsem použil TCS230, který rozezná barvu snímaného objektu díky rozpoznání jednotlivých složek spektra odražených z povrchu tohoto objektu. Objekt je poté zařazen do jednoho z připravených zásobníků. Arduino zároveň pošle přes Bluetooth do připojeného zařízení se systémem Android informaci o rozřazené barvě. Spojení přes Bluetooth je realizováno modulem HC-05.

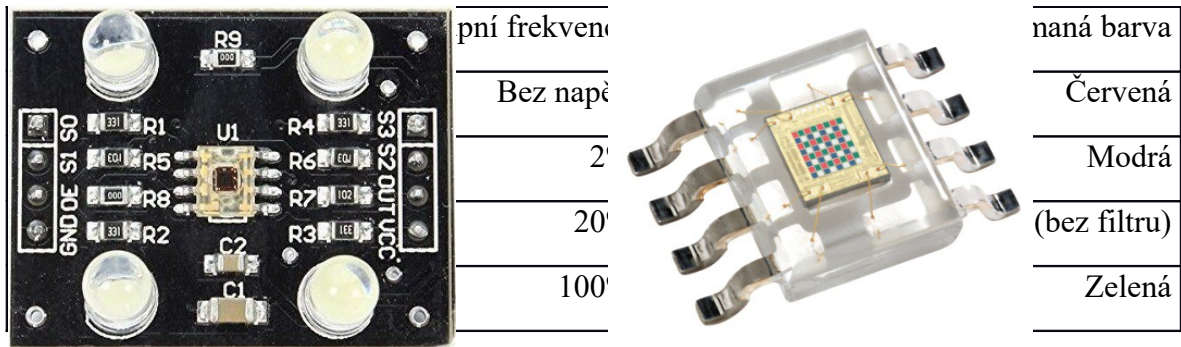
## Detektor barvy TCS230

### Obecná charakteristika

Arduino detektor barvy je modul, kterým lze detekovat barvy v jeho blízkosti. Na modulu se nachází senzor TCS230, který se skládá ze 64 fotodiód s různými barevnými filtry, a čtyř bílých LED diód pro osvětlení snímaného předmětu. Díky této kombinaci je tedy možné zjistit barvu zkoumaného předmětu, kdy po osvětlení bílými LED diodami se světlo odrazí zpět do senzoru TCS230 a ten dokáže po konfiguraci uživatelem na svém výstupu vygenerovat frekvenci výskytu barevných složek RGB. Pokud tedy například přiložíme červený předmět, odražené světlo bude obsahovat nejvíce červenou složku barevného spektra, ta vygeneruje nejvíce proudu skrze fotodiody s červeným filtrem, který je následně převeden na frekvenci. Čím vyšší výskyt dané barvy, tím nižší je jeho frekvence na výstupu.

### Použití

Tento modul má pět vstupů (S0, S1, S2, S3 a  $\bar{O}\bar{E}$ ), jeden výstup (OUT), vstup Vcc pro napětí 5V nebo 3,3V a GND. Vstupy S0 a S1 nastavují výstupní frekvenci, S2 a S3 určují snímanou barvu fotodiody.



Výstupní pin OUT vrací hodnotu výstupní frekvence naměřené barvy jedním typem fotodiod. Tyto informace jsem pak použil ve svém kódu, kdy pomocí funkce digitalWrite nastavím hodnotu pinů S2 a S3 na hodnoty 1/0 pro snímání požadované části barevného spektra. Funkce pulseIn poté počítá čas, kdy je pin OUT v hodnotě 0, a tuto hodnotu uloží do příslušné proměnné. Modul se dá napájet od 2,7V do 5,5V kdy platí, že čím menší vstupní napětí, tím menší svit LED diod.

```
int frekvenceCervena, frekvenceZelena, frekvenceModra; // proměnné pro ukládání hodnot barev

digitalWrite(pinS2, LOW); // nastavení měření červené barvy
digitalWrite(pinS3, LOW); // nastavení měření červené barvy

delay(50); // krátká pauza pro přesné měření

frekvenceCervena = pulseIn(pinOut, LOW); //uložení hodnoty červené barvy, funkce pulseIn čeká,
//až bude pin LOW, začne počítat čas a čeká až bude pin HIGH,
//poté přestává počítat a vrací tuto hodnotu v mikrosekundách

digitalWrite(pinS2, HIGH); // nastavení měření zelené barvy
digitalWrite(pinS3, HIGH);

delay(50);

frekvenceZelena = pulseIn(pinOut, LOW); // uložení hodnoty zelené barvy

digitalWrite(pinS2, LOW); // nastavení měření modré barvy
digitalWrite(pinS3, HIGH);

delay(50);

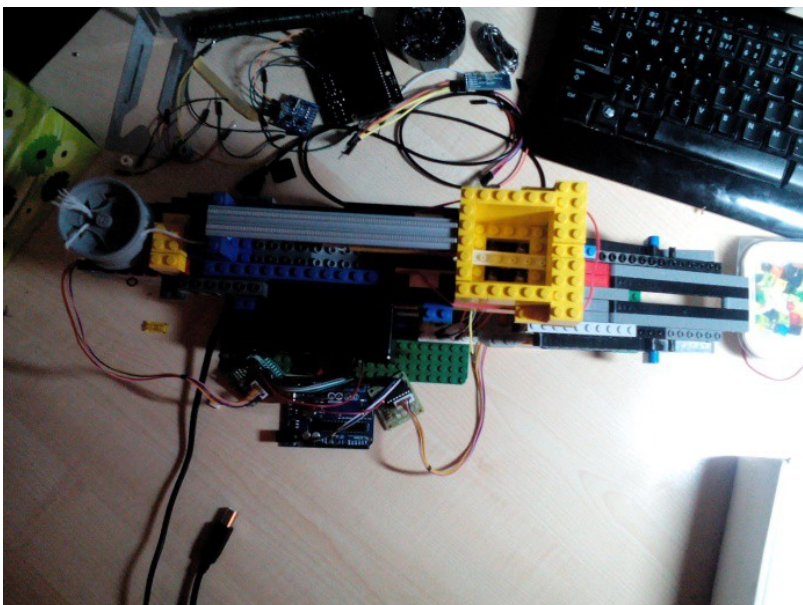
frekvenceModra = pulseIn(pinOut, LOW); // uložení hodnoty modré barvy
```

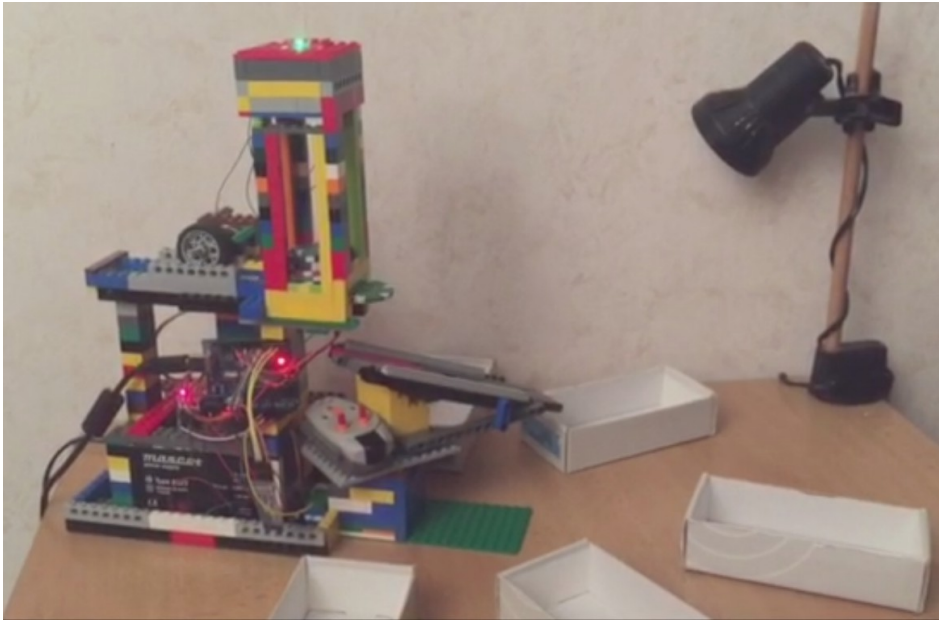
## Realizace maturitní práce

### První prototyp



První prototyp měl za úkol rozřazovat podle barvy pokerové žetony do připravených krabiček. Žetony jsem vybral proto, že byly ideálními objekty pro testování barevného senzoru díky jejich stejné velikosti, která dokonale zakrývala senzor pro rozpoznání barvy, a jednobarevnosti. Konstrukci jsem postavil z Lega. Výhodou Lega byla jeho snadná stavba a úprava, nevýhodou byla opravdu malá stabilita. Modul detektoru barvy byl umístěn uprostřed žluté části této konstrukce. Žetony musely být vyskládány do sloupce nad tímto modulem. Snímač barvy zjistil barvu nejspodnějšího žetonu. Po zjištění barvy se rampa, navazující na tuto část, otočila o příslušný úhel tak, aby žeton mohl spadnout do předem nadefinované krabičky. Dále se na nejvyšší části konstrukce rozsvítila RGB LED dioda na určenou barvu. Následným vysunutím šoupátka došlo k vytlačení žetonu na rampu a dopadu do dané krabičky. Po zasunutí plošky spadly zbývající žetony vlivem gravitační síly na detektor barvy a celý proces se zopakoval. Přítomnost žetonu zjišťoval ultrazvukový měřič vzdálenosti, který byl umístěn v nejvyšší části konstrukce pod RGB LED diodou. Původním záměrem bylo tímto modulem měřit přesný počet zbývajících žetonů v zásobníku, ale žetony byly příliš tenké na spolehlivé určení vzdálenosti (výška žetonu - 3mm, efektivní přesnost měření ultrazvukovým měřičem - 5-6 mm). Manipulace rampy i vysouvacího šoupátka byla realizována krokovými motory. Jejich výhodou bylo snadné otočení o libovolný úhel, velká síla i v klidné poloze a okamžitá odezva při zastavení či spouštění motoru. Velkou nevýhodou byla potřeba externího napájecího zdroje (12V) a složitost naprogramování.





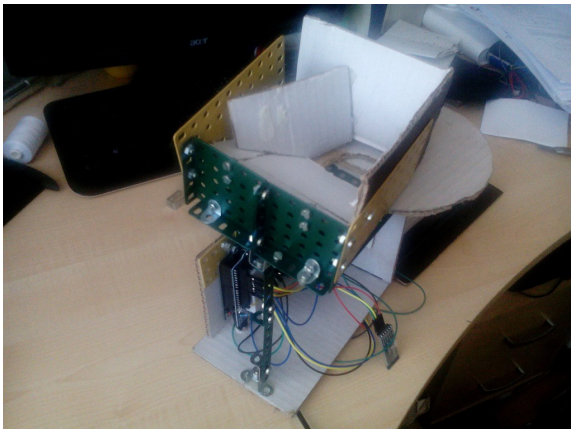
### Druhý prototyp

Úkolem druhého prototypu bylo odstranit chyby předchozího projektu. Největší nedostatky prvního pokusu spočívaly ve velice malé stabilitě a křehkosti konstrukce z Lega a napájení krokových motorů externím zdrojem. Rozhodl jsem se vyzkoušet postavit konstrukci z kartonu a vyztužení nosných částí konstrukce Merkurem. Takto vytvořená konstrukce byla daleko stabilnější než konstrukce předchozí. V předcházejícím pokusu byl problém i s vysouváním žetonů, kdy se čas od času některý žeton zasekl a byl tak nasnímán dvakrát. Tento problém jsem se pokusil odstranit pomocí karuselu- otočného kulatého disku s otvorem pro žeton. Toto řešení se ukázalo být efektivnější než řešení předcházející. Mohl jsem tak nahradit i ultrazvukový měřič vzdálenosti, který na provoz vyžadoval dva piny, za infračervený optický senzor, který využívá jen jeden digitální pin na zjištění přítomnosti žetonu v otvoru karuselu. Nakonec jsem nahradil krokové motory jednoduchými servy. Serva jsou napájena přímo z pinů Arduina a není tak třeba externího zdroje. Program na ovládání motorů se navíc zjednodušil díky knihovně Servo.h prakticky na dva řádky. Žetony se vhodí do zásobníku a jeden žeton propadne do otvoru v karuselu. Servo poté otočí karuselem o 90°, tím se dostane nad modul senzoru barev. Po nasnímání barvy se

rozsvítí RGB LED dioda na příslušnou barvu a rampa se natočí tak, aby žeton dopadl do připravené krabičky. Poté se karusel otočí o dalších 90° a žeton dopadne na rampu sklouzne do krabičky. K tomuto prototypu jsem navíc přidal Bluetooth modul HC-05. Díky tomu lze z mobilu nebo tabletu se systémem Android určit, která barva RGB LED diody se má rozsvítit u jednotlivých barev žetonů a do které krabičky mají spadnout.

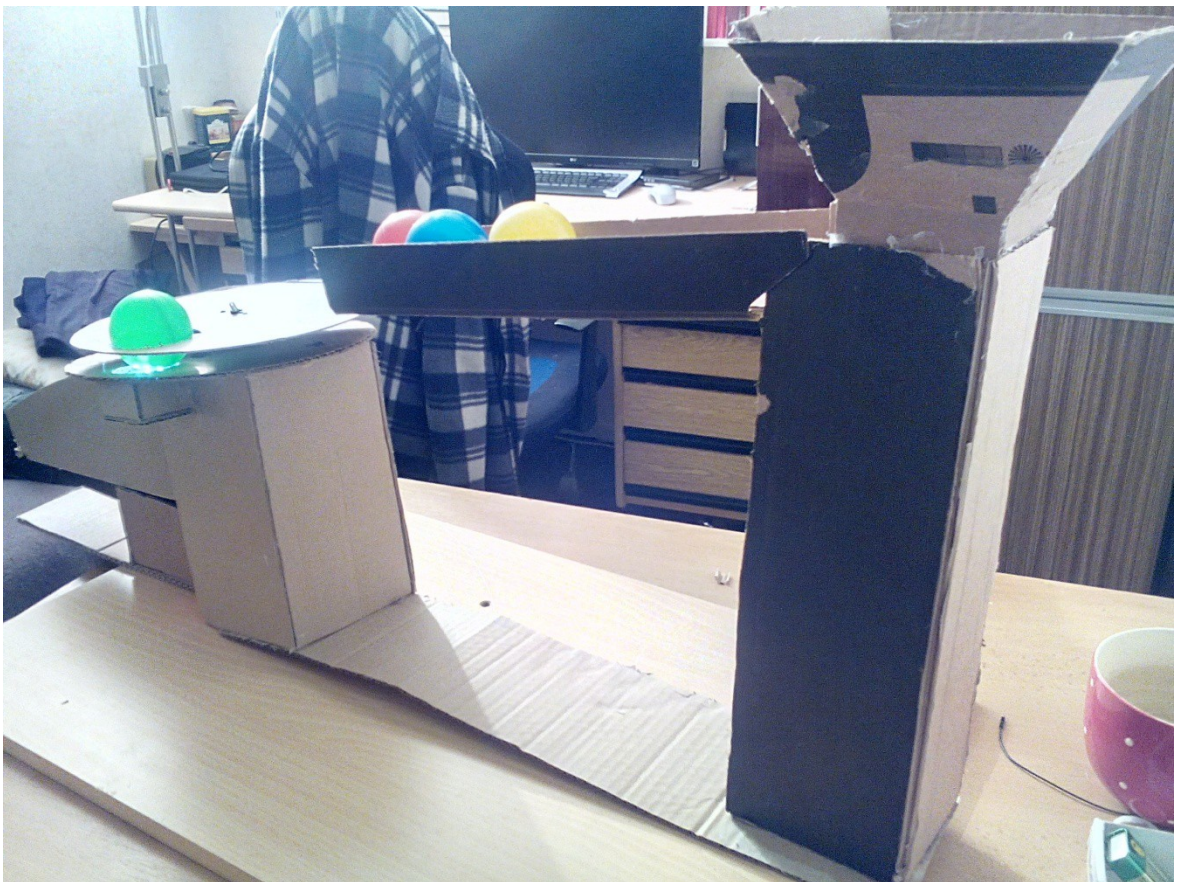
### Finální produkt

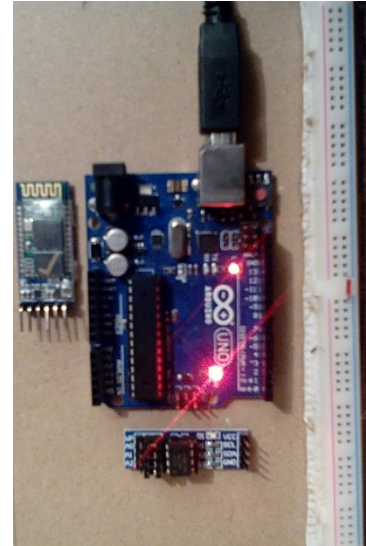
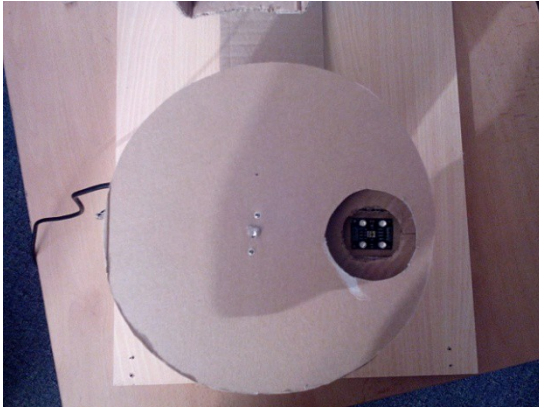
Po odzkoušení druhého prototypu jsem se finální projekt. Za úkol jsem měl přestavět prototyp č.2 tak, aby místo žetonů rozřazoval barevné plastové míčky. Posouvání žetonu pomocí karuselu se mi osvědčil, rozhodl jsem se proto znovu karusel vytvořit, nyní však ve větším měřítku. Ve výsledku tak vypadá tento finální projekt jako zvětšená verze druhého prototypu. Míčky se vhodí do zásobníku a postupně po jednom vypadávají na nakloněnou samospádový dopravník. Na jejím konci je otvor v karuselu, do kterého míček zapadne. Následně se karusel otočí o 90° nad modul senzoru barev. Po vyhodnocení barvy míčku se rozsvítí RGB LED dioda podle barvy zvolené v aplikaci v mobilu či tabletu. Dále se natočí rampa do pozice, která se také navolí v aplikaci. Poté se karusel znovu otočí o 90° a míček vypadne do zvolené krabičky. Arduino zároveň odešle díky modulu HC-05 přes Bluetooth informaci o právě rozřazené barvě. Aplikace tuto informaci uloží do své databáze a na



vyžádání uživatelem může odeslat na mail tabulku ve formátu .csv s časem a počtem jednotlivých rozřazených barev. Nevýhodou prototypu č.2 byla nutnost po interním restartu Arduino v aplikaci znovu nastavovat požadované hodnoty a Arduino muselo znovu kalibrovat všechny rozeznávané barvy. Proto jsem v tomto projektu začal využívat EEPROM Arduino, do které se ukládají krajní hodnoty jednotlivých barev i nastavení z

aplikace. Jelikož má EEPROM jen omezený počet možných zápisů, přidal jsem k Arduino pro jistotu externí paměť EEPROM, která komunikuje pomocí sběrnice I2C.





## Aplikace pro Android

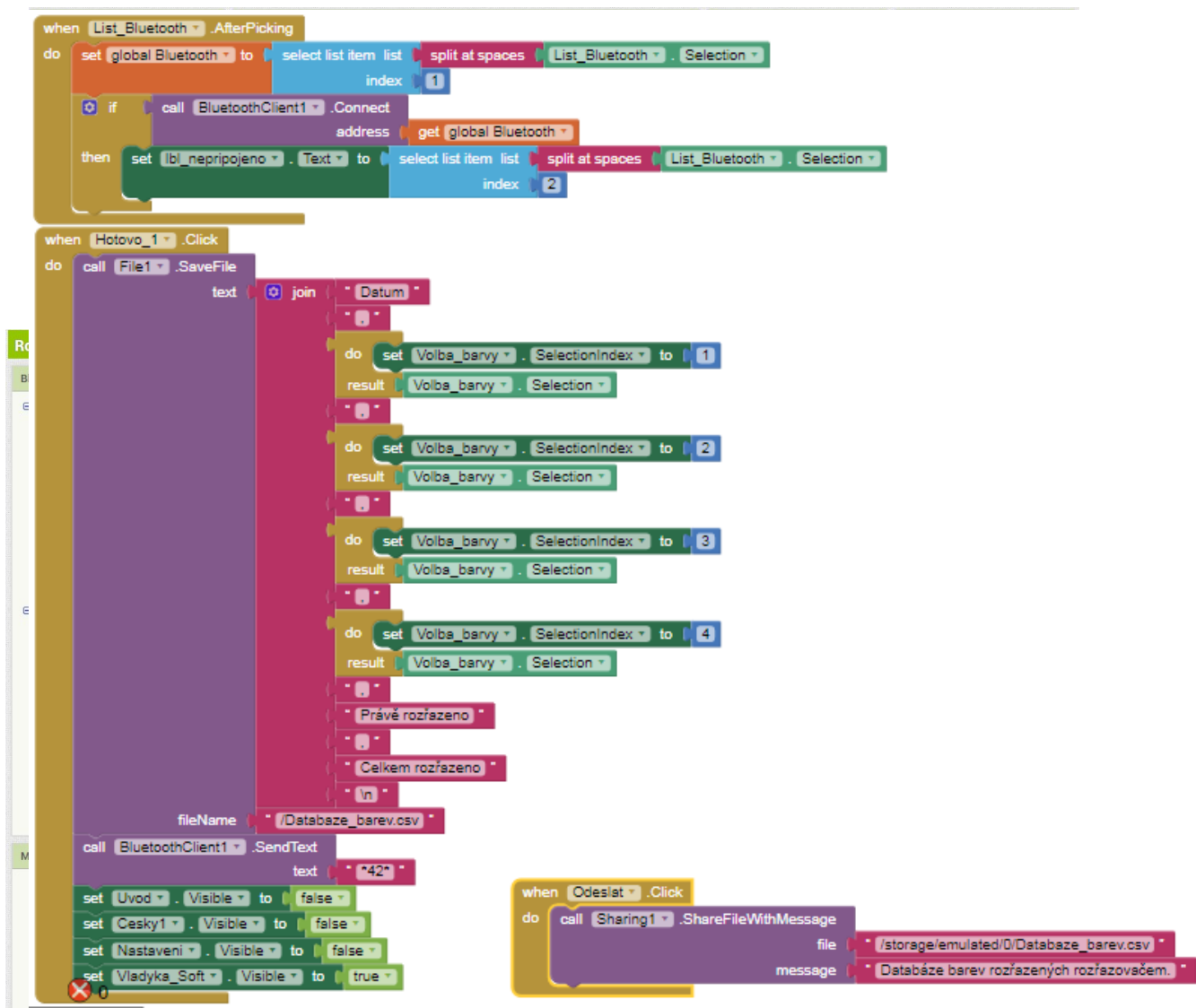
### Výběr programovacího prostředí

Hned po dokončení prvního prototypu jsem se začal zabývat myšlenkou vzdáleného ovládání této linky. Původně jsem zamýšlel linku ovládat pomocí programu v počítači. Po několika zkušebních programech v prostředí Microsoft Visual Studio jsem dokázal vytvořit program, komunikující přes sériovou linku, pro vybrání barvy RGB LED diody, která má při rozřazení jednotlivých barev svítit, a pro natočení rampy do požadované pozice. Těžší už bylo vytvořit v tomto programu zapisování hodnot do databáze, protože USB, přes které bylo Arduino připojeno k počítači, mohlo v jednu chvíli pracovat jen v jednom směru (buď data vysílat do Arduina, nebo data přijímat do počítače). Problémem bylo, že pokaždé, když se změnil směr pro přijímání hodnot z Arduina do počítače, Arduino provedlo reset a program se tak nemohl dostat z nastavení k rozlišování barev. Tento problém jsem vyřešil instalací externího programu, hodnoty jsem ale mohl zapisovat jen do Poznámkového bloku. Navíc k odesílání tohoto souboru na mail bylo zapotřebí stáhnout další program. Další nevýhodou byla potřeba neustálého propojení Arduina s počítačem přes USB. Proto jsem se rozhodl pro bezdrátové řešení pomocí Bluetooth. K Arduinu jsem dokoupil modul HC-05 a začal vytvářet aplikaci pro android. Jako první jsem začal psát aplikaci v programu Android Studio. Programovací jazyk, které Android Studio využívá, má velice blízko k programovacímu jazyku java. V tomto jazyce jsem nikdy dříve neprogramoval, a tak jsem si tuto alternativu nechal jen jako záložní plán. Po chvíli hledání vhodného programovacího prostředí na internetu jsem našel stránku MIT App Inventor. Toto programovací prostředí funguje na principu Scratch, kdy jednotlivé příkazy jsou ve formě "puzzle". Při samotné programování jen vyberete dílek s příkazem a "přípevníte ho" k

jiným dílkům kódu. Ze začátku se může zdát, že toto prostředí je jen pro jednoduché aplikace, ale díky mnoha funkcím (rozeznávání hlasu, text na řeč, přístup k fotoaparátu, gyroskopům v zařízení, a další ...) zde jdou vytvořit i velice komplexní programy.

### **Program v MIT App Inventoru**

Ze začátku jsem vytvořil jednoduchý program s pomocí návodu na internetu. Tento program po zmáčknutí tlačítka poslal přes Bluetooth v mobilu zprávu Arduino, a to obratem poslalo zprávu o přijetí, která se zobrazila v textovém poli. Další program měl za úkol pomocí odeslané zprávy nastavit na Arduino barvu RGB LED diody. Po odzkoušení tohoto programu jsem prakticky stejným způsobem naprogramoval i ovládání natočení rampy. Následně jsem tento program přepsal do funkčního programu dopravníku s rozlišením barev. Arduino na začátku svého programu čeká, až uživatel nastaví barvy RGB LED diody a natočení rampy jednotlivých barev, a poté posílá zpět do mobilu nebo tabletu informaci o právě rozřazené barvě. Program v mobilu si tuto hodnotu uloží do své databáze a připíše k ní čas, kdy tuto hodnotu přijal. Pokud chce uživatel tyto hodnoty poslat na mail, program vytvoří tabulku formátu .csv (comma separated values - hodnoty oddělené čárkami) a odešle ji na zvolenou adresu.



## Závěr

	Soubor	Zobrazit	Upravit	Buňky	Vložit	Data	
	A	B	C	D	E	F	G
1	Datum		Barva2	Barva3	Barva4	Práve rozřazeno	Celkem rozřazeno
2	03.16.18 09:47	1	1	1	1	4	4
3	03.16.18 09:48	0	0	0	0	0	4

Osobně bych řekl, že se mi podařilo úspěšně splnit většinu zadaných úkolů. Během zpracování této maturitní úlohy jsem dokázal prohloubit své znalosti o spoustě modulů k Arduino, se kterými jsme ve škole pracovali jen chvíli, nebo se o nich jen bavili v teoretických hodinách. Modul detekce a rozeznávání barev mě velice zaujal, a proto jsem

byl potěšen, že jsem se tomuto tématu mohl věnovat i ve své maturitní práci. Zároveň jsem se naučil i programovat složitější aplikace na zařízení s Androidem, což byla velice zajímavá zkušenost, která se bude do budoucna určitě hodit. Nakonec bych rád poděkoval panu Ing. Stanislavu Moravcovi za jeho návrhy na vylepšení práce a jeho ochotu poradit s řešením problémů.

## **Zdroje**

Arduino Reference. Arduino - Home [online]. Dostupné z:

<https://www.arduino.cc/reference/en/>

EDU: MIT. [online]. Dostupné z:

<http://edu.vos-kh.cz/vyuka/course/index.php?categoryid=16>



[Detektor barvy TCS3200] - laskarduino.cz. Dostupné z:

<https://laskarduino.cz/vstupni-periferie-cidla/230159-detektor-barvy-tcs3200.html>

Arduino detektor barvy TCS230 | Arduino návody. Dostupné z:

<http://navody.arduino-shop.cz/navody-k-produktum/arduino-detektor-barvy.html>

Krokový motor 28BYJ-48 a driver ULN2003 | Arduino návody. Dostupné z:

<http://navody.arduino-shop.cz/navody-k-produktum/krokovy-motor-a-driver.html>

Arduino Bluetooth modul HC-05 | Arduino návody. Dostupné z:

<http://navody.arduino-shop.cz/navody-k-produktum/arduino-bluetooth-modul-hc-05.html>

Servo motor | Arduino návody. Dostupné z:

<http://navody.arduino-shop.cz/arduino-projekty/servo-motor.html>

Infračervený optický senzor | Arduino návody. Dostupné z:

<http://navody.arduino-shop.cz/navody-k-produktum/infracervený-optický-senzor.html>

MIT App Inventor | AI2 App Inventor. Dostupné z:

<http://ai2.appinventor.mit.edu/?locale=en>

Using an RGB LED to Detect Colours | Instructables - Copyright © 2017 Autodesk, Inc.

Dostupné z: <http://www.instructables.com/id/Using-an-RGB-LED-to-Detect-Colours/>

Lekce 17 - Arduino a EEPROM :: Arduino. Home [online]. Copyright © 2015

Dostupné z: <https://arduino8.webnode.cz/news/lekce-17-arduino-a-eprom/>



VYŠŠÍ ODBORNÁ ŠKOLA, STŘEDNÍ PRŮMYSLOVÁ ŠKOLA  
A JAZYKOVÁ ŠKOLA S PŘÁVEM STÁTNÍ JAZYKOVÉ ZKOUŠKY



**KONZULTAČNÍ LIST**

Přijetí a jméno žáka: Vladyka Tomáš

Třída: E4C

Téma maturitní práce: Dopravník s rozlišením barev

Datum kontroly: 10. 11. 2017

Obsah kontroly:

Hodnocení kontroly: 1 2 3 4 5

Podpis vedoucího práce: *Musarov*

Datum kontroly: 9. 3. 2018

Obsah kontroly:

Hodnocení kontroly: 1 2 3 4 5

Podpis vedoucího práce: *Musarov*

Datum kontroly:

Obsah kontroly:

Podpis vedoucího práce:

\_\_\_\_\_