|  |  |  |
| --- | --- | --- |
| Logo__SSPU_2016_Barva | | |
| **Závěrečná studijní práce**  **dokumentace** | | |
| **Arduino třídička barev** | | |
| Roman Stanjura | | |
|  | | |
|  | |  |
| **Obor:** | 18-20-M/01 INFORMAČNÍ TECHNOLOGIE  se zaměřením na počítačové sítě a programování | |
| **Třída:**  **Školní rok:** | IT4  2019/2020 | |

**Poděkování**

*Chtěl bych poděkovat panu Ing. Petru Grussmannovi za veškerou pomoc při tvorbě projektu, panu Mgr. Marcelovi Godovskému za poskytnutí potřebných součástek, panu Mgr. Markovi Lučnému za cenné rady k prezentaci a v neposlední řadě paní Ing. Haně Rohanové za pomoc s kódem.*

Prohlašuji, že jsem závěrečnou práci vypracoval samostatně a uvedl veškeré použité   
informační zdroje.

Souhlasím, aby tato studijní práce byla použita k výukovým účelům na Střední průmyslové   
a umělecké škole v Opavě, Praskova 399/8.

V Opavě 31. 12. 2019

*podpis autora práce*

**ANOTACE**

Sestavený projekt třídí barvy pomocí senzoru TCS230, který pomocí zakomponovaných LED diod osvětlí daný objekt a barevné světlo se vrátí zpět do mikrokontroléru, čímž rozpoznává barvy, dvou servo motorů, díky kterým daný objekt automaticky projede celou třídičkou a jednodeskovým počítačem Arduino Nano, na které jsou všechny součástky napojeny. Program tohoto projektu byl vyvinut ve vývojovém prostředí VSCode s rozšířením Platformio, se kterým je právě tato mikropočítačová deska kompatibilní. Použitý programovací jazyk je C++.

**Klíčová slova**

třídička barev, senzor, TCS230, Servo motor, Arduino, Platformio, VSCode

**ANOTATION**

The assembled project sorts colors using the TCS230 sensor, which uses embedded LEDs to illuminate the object and return the color light back to the microcontroller, recognizing the colors of the two servo motors that automatically pass the object through the sorter and Arduino Nano single board computer. that all the components are connected. The program of this project was developed in the VSCode development environment with Platformio extension, with which this microcomputer board is compatible. The programming language used is C ++.

**Keywords**

Color sorter, senzor, TCS230, Servo motor, Arduino, Platformio, VSCode

OBSAH

[Úvod 4](#_Toc29753986)

[1 výroba PROJEKTU 5](#_Toc29753987)

[2 Využité technologie 6](#_Toc29753988)

[2.1 Hardware 6](#_Toc29753989)

[2.1.1 Seznam součástek 6](#_Toc29753990)

[2.1.2 Arduino Nano ATmega328P 7](#_Toc29753991)

[2.1.3 Barevný senzor TCS320 8](#_Toc29753992)

[2.1.4 Servo motor SG90 180° 8](#_Toc29753993)

[2.1.5 Propojovací kabely 9](#_Toc29753994)

[2.2 Software 10](#_Toc29753995)

[2.2.1 Visual Studio Code 10](#_Toc29753996)

[2.2.2 Arduino IDE 10](#_Toc29753997)

[2.2.3 Rozšíření 11](#_Toc29753998)

[3 Způsoby řešení a použité postupy 12](#_Toc29753999)

[3.1 Schéma 12](#_Toc29754000)

[3.2 Použité knihovny 12](#_Toc29754001)

[3.2.1 Arduino.h 12](#_Toc29754002)

[3.2.2 Servo.h 13](#_Toc29754003)

[3.3 Postup kódu 13](#_Toc29754004)

[Závěr 14](#_Toc29754005)

[Seznam použitýCH INFORMAČNÍCH ZDROJů 15](#_Toc29754006)

Úvod

Tento projekt jsem si vybral, protože mi přišel obecně zajímavý jednodeskový mikropočítač Arduino, díky kterému si můžete vytvořit spoustu zajímavých projektů, a to za pohodlí domova a za přijatelnou cenu. Konkrétně na problematiku ohledně třídění barev mě navedl můj malý bratr, který je tak trochu puntičkář, a tak jsem se rozhodl mu nějak pomoci. Jak takovou třídičku udělat jsem našel na internetu. Cílem mého projektu bylo za pomocí již výše zmíněných senzorů a pár kabelů vytvořit jednoduchou třídičku barev, která bude umět třídit malé objekty podle barvy. Všechny součástky potřebné k tomuto projektu jsem koupil na internetu, přesněji z LaskaArduino, což je internetový obchod s elektronikou.

Ze začátku dokumentace se budu věnovat využitým součástkám a softwaru. Poté se Vám budu snažit přiblížit postup jak kódu, tak zapojení a v neposlední řadě se podíváme na řešení vyskytnutých problémů.

# výroba PROJEKTU

Mým prvním krokem bylo vyrobení papírové krabičky (viz úvodní obrázek), na kterou jsem přilepil nepájivé pole. Nejdříve jsem na něm vyzkoušel jednotlivé součástky, jestli fungují tak, jak mají. Poté jsem připojil Arduino Nano a k němu pomocí kabelů dva servo motory. První servo motor má za úkol posunout objekt pod připojený barevný senzor TCS230, který následovně pomocí světel ze zakomponovaných LED diod zjistí barvu objektu. Po vyhodnocení barvy se první servo motor posune do polohy, kde objekt spadne dírou do vyrobené papírové “skluzavky”, která je nalepená na druhý servo motor. Ten má po vyhodnocení barvy za úkol otočit se tak, aby byl ve správné poloze závisle na barvě objektu a následně objekt mohl po „skluzavce“ sjet na správné místo (na stejnobarevnou značku).

# Využité technologie

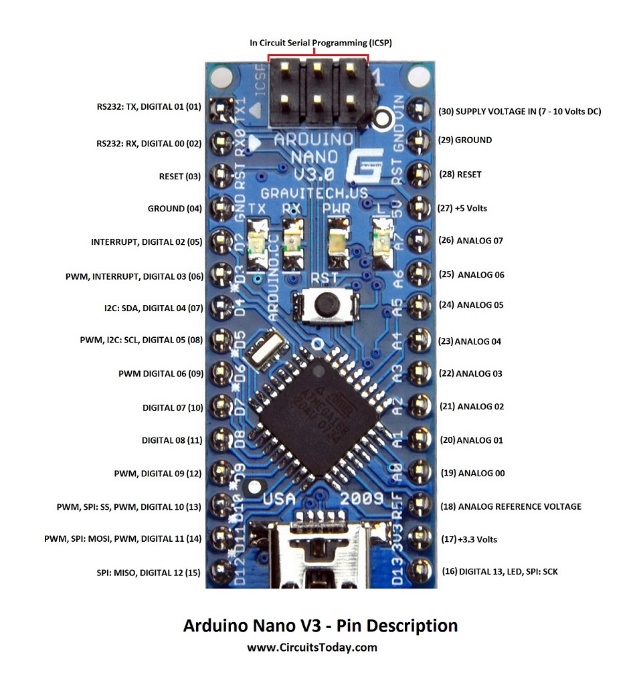
## Hardware

### Seznam součástek

* Jednodeskový mikropočítač Arduino UNO R3 ATmega328P
* Barevný senzor TCS230
* Servo motor SG90 180°
* Propojovací kabely

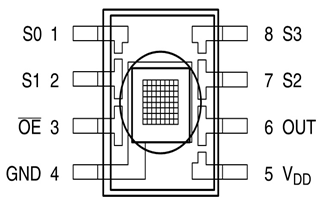
### Arduino Nano ATmega328P

Základem mého projektu je mikropočítačová Arduino Nano ATmega328P. Tuto desku jsem si vybral z toho důvodu, že je kompatibilní s Platformiem, které jsem přidal jako rozšíření k VSCodu, ve kterém jsem pracoval. Má vlastní USB port a převodník. Je to obdoba jednodeskového mikropočítače Arduino UNO, ovšem ten je větší, což mi nevyhovovalo, a proto jsem upřednostnil tento typ. Deska samotná má flash paměť 32 KB, obsahuje 22 vstupních/výstupních pinů, z toho 14 digitálních a 6 analogových. Jako téměř u všech desek tohoto typu zde můžeme najít připojení přes USB a resetovací tlačítko. Deska pracuje s napětím o velikosti 5V.



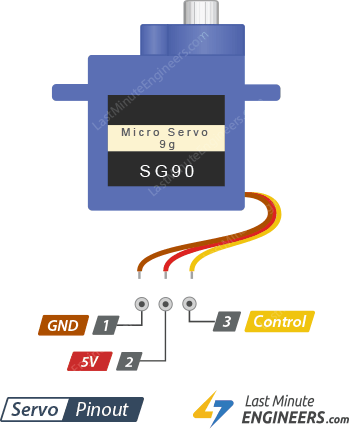
### Barevný senzor TCS320

Barevný senzor TCS230 je určen pro zjištění barvy. Funguje pomocí mikrokontroléru a čtyř LED diod, které posvítí na objekt, který chceme zkoumat. Barevné světlo se odrazí zpět do senzoru, který poté zjistí frekvenci složek RGB, a díky pevně daným parametrům v kódu zjistí barvu objektu. Jeho jedinou nevýhodou je vzdálenost, na kterou se podaří barvu detekovat (1cm). Důsledkem tohoto problému bylo zkomplikování tvorby krabičky, ve které je celý projekt zapojen. Napájecí napětí tohoto modulu je 3-5V a nachází se v něm 8 pinů.



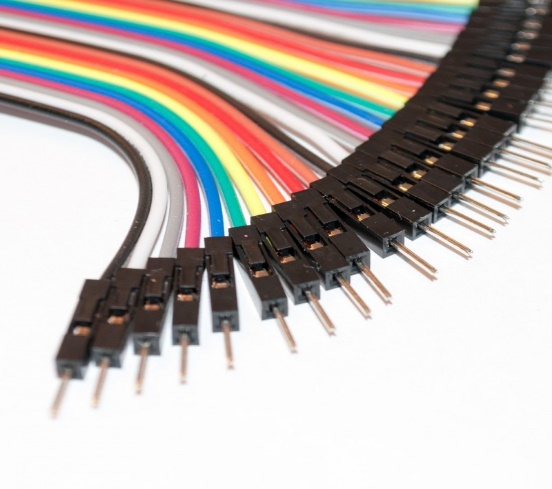
### Servo motor SG90 180°

Servo motor je motor pro pohony, u kterého lze nastavit přesnou polohu, kam se má otočit. Existuje mnoho typů servo motorů, ale k mému projektu stačil ten nejjednodušší, který se otáčí o 180°. Provozní napětí tohoto serva je 3-7V a jeho rychlost otáčení je 60° za necelou půl sekundy. V mém případě jsem jej použil rovnou dvakrát. První servo motor je určen k tomu, aby se objekt, který se má vyhodnotit, posunul pod LED diody senzoru TCS230, který určí jeho barvu. Na druhém servo motoru je přilepená papírová „skluzavka“. Slouží k tomu, aby se po určení barvy a následném opuštění prostoru prvního servo motoru vyhodnocený objekt dostal na správné místo (stejnobarevná značka).



### Propojovací kabely

K propojení všech součástek jsem použil propojovací kabely, které jsou zapojeny do nepájivého pole. Zapojení mi znepříjemnil barevný modul TCS230, který nemohl být zapojený v nepájivém poli. Pro správný chod projektu jsem jej musel umístit do papírové krabičky. Z toho důvodu jsem musel doobjednat jiný typ propojovacích kabelů, konkrétně typ F-F (samice-samice), protože bych modul neměl pomocí klasických kabelů M-M (samec-samec) jak zapojit.



## Software

### Visual Studio Code

K psaní celého kódu jsem použil Visual Studio Code, což je vývojové prostředí od společnosti Microsoft s podporou mnoha programovacích jazyků a pomocnou funkcí našeptávání. Využil jsem jej proto, že celý projekt je založen na Platformiu, což je rozšíření VSCodu, které se používá pro programování jednodeskových počítačů (Arduino, ESP) a je opravdu klíčové, pokud si chcete práci mnohonásobně zjednodušit.



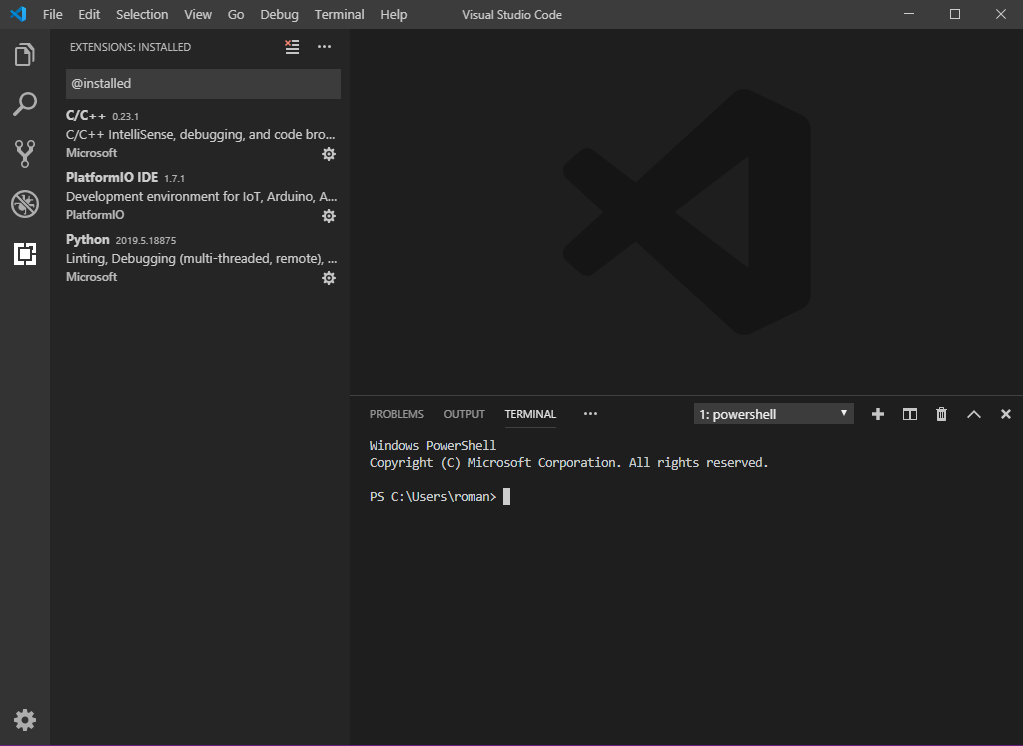
### Arduino IDE

Prvotním vývojovým prostředím, které jsem použil, bylo Arduino IDE. Využil jsem jej z počátku tvorby projektu a sloužilo k tomu, abych vyzkoušel správný chod všech jednotlivých součástek. Při další fázi tvorby projektu jsem musel od Arduina IDE upustit, protože nenabízí tolik uživatelských funkcí jako Visual Studio Code a hlavně nepodporuje rozšíření Platformio.



### Rozšíření

Pro správný chod kódu bylo nezbytné přidat rozšíření (extensions) ve Visual Studio Code. Přidal jsem zde již zmíněné Platformio, což je integrované vývojové prostředí, které jsem použil pro programování všech mých modulů. Je určeno pro programování v jazyce C/C++. Dále jsem přidal Python, bez kterého bych se neobešel při psaní příkazů, které vedly ke spuštění celého programu. No a na závěr jsem přidal rozšíření C/C++, což je programovací jazyk, ve kterém je celý kód napsaný.



# Způsoby řešení a použité postupy

## Schéma



Pro tvorbu schématu zapojení jsem použil open source program Fritzing. Vybral jsem si jej, protože práce v něm je jednoduchá na pochopení, má kvalitní grafické vyobrazení součástek a je velmi přehledné.

## Použité knihovny

### Arduino.h

Knihovna Arduino.h je potřebná ve všech kódech, které pracují s jednodeskovým mikropočítačem Arduino. Knihovy Arduino se skládají ze dvou souborů a těmi jsou záhlaví (s příponou .h) a zdrojového souboru (s příponou .pp). Tento konkrétní soubor Arduino.h obsahuje definice knihovny, což je seznam všeho, co je uvnitř, včetně příkazů (funkcí) a potřebných proměnných.

### Servo.h

Tato knihovna umožňuje desce Arduino ovládat servo motory. Knihovna samotná umožňuje až 12 motorů Servo najednou, takže by pro projekty s mikropočítači měla bohatě stačit všem.

## Postup kódu

Nejprve musíme připojit knihovnu „Servo.h“ a „Arduino.h“, bez kterých by kód nefungoval, definovat piny, ke kterým bude připojen barevný senzor, vytvořit servo objekty a deklarovat proměnné potřebné pro program. V sekci nastavení musíme definovat piny jako výstupy a vstupy, nastavit frekvenční škálování barevného senzoru, definovat servo piny a zahájit sériovou komunikaci pro čtení výsledků barvy načtené na sériovém monitoru. V sekci LOOP začíná program přesunutím horního servomotoru do polohy, kde se nachází trubička, která bude naplněna různě barevnými objekty. Následně se pomocí FORU otáčí a uvede objekt do polohy barevného senzoru. Používáme FOR, abychom mohli regulovat rychlost rotace změnou doby zpoždění v LOOPU. Po půlsekundovém zpoždění pomocí funkce "cteniBarvy()" přečteme barvu objektu. Pomocí čtyř pinů a pinů s frekvenčním výstupem barevného senzoru čteme barvu objektu. Senzor čte 3 různé hodnoty pro každý objekt: červenou, zelenou, modrou (RGB), a podle těchto hodnot zjistíme, jaká je barva objektu. Dále pomocí SWITCHE otočíme spodní servo tak, aby objekt spadl do "skluzavky", která objekt zavede do stejnobarevné nádoby. Poté se spodní servo vrátí na předchozí pozici, aby se proces mohl opakovat.

# **Závěr**

Cílem tohoto projektu bylo sestavit jednoduchou třídičku barev. Za pomocí čtyř poměrně levných součástek se mi úkol podařilo splnit. Tento projekt pravděpodobně bohužel nemá žádné jiné využití než čistě vzdělávací, ale co je pro mě důležité je to, že mě tvorba mého projektu bavila a hlavně jsem pochopil všeobecnou práci s jednodeskovými počítači, a to jak zapojování, tak programování. Celý projekt je zdokumentován na mém githubu: <https://github.com/romanstanjura/projekt---arduino-color-sorter>

Seznam použitýCH INFORMAČNÍCH ZDROJů

[1] Programování Arduino [online] [cit. 2019-12-21].

<<https://arduino.cz/vyvojove-prostredi-platformio-ide/>>

[2] Platformio tutoriál [online] [cit. 2019-12-21]

<<https://docs.platformio.org/en/latest/quickstart.html>>

[3] Servo motor [online] [cit. 2019-12-21].

<<https://www.instructables.com/id/Arduino-Servo-Motors/>>

[4] TCS 230 [online] [cit. 2019-12-21].

<<https://howtomechatronics.com/tutorials/arduino/arduino-color-sensing-tutorial-tcs230-tcs3200-color-sensor/>>

[5] Fritzing Tutorial - A Beginners Guide to Making Circuit & Wiring Diagrams [online] [cit. 2019-12-21].

<<https://www.youtube.com/watch?v=-saXw1EipX0>>