

# Platforma pro výuku programování na střední škole

Bc. Josef Můck

---

Diplomová práce  
2023



Univerzita Tomáše Bati ve Zlíně  
Fakulta aplikované informatiky

---

Univerzita Tomáše Bati ve Zlíně  
Fakulta aplikované informatiky  
Akademický rok: 2022/2023

Studijní program: Inženýrská informatika  
Forma studia: Prezenční  
Obor/kombinace: Učitelství informatiky pro střední  
školy (pnUI)

## Podklad pro zadání DIPLOMOVÉ práce studenta

Jméno a příjmení: **Bc. Josef Mück**

Osobní číslo: **A21134**

Téma práce: **Platforma pro výuku programování v prostředí Arduino na střední škole**

Téma práce anglicky: **An Arduino-based Tuition Platform for Teaching Programming in a Secondary School**

Jazyk práce: **Čeština**

Vedoucí práce: **Ing. Jan Dolinay, Ph.D.**

Ústav automatizace a řídicí techniky

Zásady pro vypracování:

1. Vypracujte literární rešerši na dané téma.
2. Navrhněte výukovou platformu se zvolenými komponentami.
3. Uvedený návrh hardwarově realizujte.
4. Vytvořte ukázkové programy pro použité komponenty.
5. Zpracujte na základě ukázkových programů sadu řešených úloh.
6. Vypracujte výukovou prezentaci o vytvořené platformě.

Seznam doporučené literatury:

1. CATSOULIS, John. Designing embedded hardware. 2nd ed. Sebastopol, CA: O'Reilly, 2005, xvi, 377 p. ISBN 0596007558.
2. IADMAN, Josef. Elektronické konstrukce pro začátečníky. Praha: BEN – technická literatura, 2001. ISBN 80-730-0015-6.
3. MARGOLIS, Michael, Brian JEPSON a Nicholas Robert WELDIN. Arduino cookbook: recipes to begin, expand, and enhance your projects. Third edition. Sebastopol: O'Reilly Media, 2020. ISBN 149190352x.
4. MCCONNELL, Steve. Dokonalý kód: umění programování a techniky tvorby software. Vyd. 1. Brno: Computer Press, 2005, 894 s. ISBN 802510849x.
5. PINKER, Jiří. Mikroprocesory a mikropočítače. 1. vyd. Praha: BEN – technická literatura, 2004, 159 s. ISBN 80-7300-110-1.
6. WHITE, Elicia. Making embedded systems. Sebastopol: O'Reilly, c2012. ISBN 9781449302146.

Podpis studenta:

Datum:

Podpis vedoucího práce:

Datum:

### **Prohlašuji, že**

- beru na vědomí, že odevzdáním diplomové práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně;
- byl/a jsem seznámen/a s tím, že na moji diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování diplomové práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

### **Prohlašuji,**

- že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne 25.5.2023

Josef Můck, v.r.  
podpis studenta

## **ABSTRAKT**

Cílem práce bylo vytvořit kompaktní platformu pro výuky a seznámení se s vývojovou deskou Arduino. Platforma je především jako podpora učitelů do výuky a také jako možnost žákům si vyzkoušet jednotlivé komponenty. Práce je koncipovaná tak, aby žáci měli možnost se rychle a efektivně seznámit s platformou Arduino a pochopit základní princip a využití jednotlivých komponentů.

V teoretické části se práce zabývá popisem jednotlivých typů desek Arduino se kterými se mohou žáci v průběhu studia setkat. Dále jsou popsány jednotlivé dostupné komponenty, které lze použít při tvorbě nejrůznějších projektů.

V praktické části byla vytvořena vlastní platforma, která byla osazena vývojovou deskou Arduino Mega a k ní jsou připojeny vybrané komponenty, jako je displej nebo klávesnice. Pro možnost rozšíření platformy je také osazena nepájivým polem, které lze využít pro prototypování. Pro platformu byla vytvořena výuková prezentace, a několik praktických úkolů pro žáky.

Klíčová slova: Arduino, mikropočítač, programování, výuka, platforma

## **ABSTRACT**

The aim of the work was to create a compact platform for learning and familiarization with the Arduino development board. The platform is mainly as a support for the teacher in the classroom and also as an opportunity for students to try out the different components. The work is designed in such a way that the pupils have the opportunity to quickly and efficiently get acquainted with the Arduino platform and understand the basic principle and use of each component.

The theoretical part of the thesis deals with the description of the different types of Arduino boards that pupils may encounter during their studies. Furthermore, the different available components that can be used in the creation of various projects are described.

In the practical part, a custom platform was created, which was equipped with an Arduino Mega development board and selected components such as a display or a keyboard are connected to it. For the possibility of extending the platform, it is also fitted with a non-soldering array that can be used for prototyping. An educational presentation was created for the platform, and several practical tasks for the students.

Keywords: Arduino, microcomputer, programming, teaching, platform

Chtěl bych poděkovat vedoucímu diplomové práce panu Ing. Janu Dolinayovi, Ph.D. za odborné vedení, za pomoc a rady při zpracování této práce.

Prohlašuji, že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

# OBSAH

<b>ÚVOD .....</b>	<b>8</b>
<b>I TEORETICKÁ ČÁST .....</b>	<b>9</b>
<b>1 INFORMATIKA A JEJÍ VÝUKA .....</b>	<b>10</b>
1.1 CÍLOVÉ ZAMĚŘENÍ VZDĚLÁVACÍ OBLASTI.....	10
1.2 ZMĚNA VE VÝUCE INFORMATIKY .....	11
1.3 KLÍČOVÉ KOMPETENCE.....	12
<b>2 PLATFORMY PRO VÝUKU PROGRAMOVÁNÍ.....</b>	<b>14</b>
2.1 ARDUINO .....	14
2.2 HISTORIE .....	14
2.3 TYPY DESEK.....	14
2.3.1 Arduino Mini.....	14
2.3.2 Arduino Nano .....	15
2.3.3 Arduino Uno.....	16
2.3.4 Aduino Leonardo.....	17
2.3.5 Arduino Mega .....	17
2.3.6 Arduino Robot.....	18
2.4 KLONY.....	19
2.4.1 Seeduino V4.2 .....	19
2.4.2 Rainbowduino v3.0.....	20
2.5 SHIELD.....	21
2.5.1 Arduino 4 Relé Schield.....	22
2.5.2 Schield šroubového terminálu na Arduino Nano .....	22
2.5.3 Schield pro připojení motorů a senzorů.....	23
2.6 TVORBA PROGRAMU NA ARDUINO PLATFORMU .....	24
2.6.1 Arduino IDE.....	24
2.6.2 Scratch pro Arduino.....	25
2.6.3 mBlock.....	26
2.6.4 AUTODESK Tinkercad.....	26
2.6.5 Fritzing.....	27
2.7 LEGO MINDSTORMS .....	28
2.7.1 Lego Mindstorms RCX.....	28
2.7.2 Lego Mindstorms NXT.....	29
2.7.3 Lego Mindstorms Robotí vynálezce.....	30
<b>3 VÝVOJOVÉ SESTAVY ARDUINO .....</b>	<b>32</b>
3.1 UNO R3 ULTIMATE STARTER KIT .....	32
3.2 STAVEBNICE CHYTRÉHO DOMU .....	33
3.3 TOTEM.....	34
3.3.1 Totem Mini Lab.....	34
3.3.2 Sada pro rozvoj dovedností STEM.....	37
3.3.2.1 Sada pro mladého inženýra: Pes.....	38
3.3.2.2 GripperBbot .....	39
3.3.2.3 Sada podvozku auta řízená přes telefon pomocí Bluetooth .....	40
<b>II PRAKTICKÁ ČÁST .....</b>	<b>41</b>

<b>4</b>	<b>VÝVOJ PLATFORMY.....</b>	<b>42</b>
4.1	POUŽITÉ KOMPONENTY NA PLATFORMĚ.....	42
4.1.1	LCD displej 4x16 .....	42
4.1.2	Klávesnice .....	44
4.1.3	Relé modul .....	46
4.1.4	Modul s tlačítky.....	48
4.1.5	Nepájivé pole.....	49
4.1.6	H můstek .....	51
4.1.7	Driver krokových motorů .....	53
4.1.8	Motor na stejnosměrné napětí .....	54
4.1.9	Krokový motor .....	56
4.1.10	RGB LED kruh.....	58
4.1.11	Ultrazvukový senzor.....	60
4.1.12	Modul pro připojení periférií bez pinů .....	61
4.1.13	Kompletní platforma.....	61
4.2	KONSTRUKCE.....	62
4.3	PODKLAD PLATFORMY.....	62
4.4	TIŠTĚNÉ DRŽÁKY NA KOMPONENTY .....	62
<b>5</b>	<b>POUŽITÝ SOFTWARE PRO PLATFORMU .....</b>	<b>64</b>
5.1	ARDUINO IDE 2 .....	64
5.1.1	Automatické dokončování .....	65
5.1.2	Nástroj pro ladění .....	65
5.2	WIRING.....	66
5.2.1	Deklarace proměnných .....	66
5.2.2	Datové typy .....	67
5.2.2.1	Datový typ Byte.....	67
5.2.2.2	Datový typ Integer.....	67
5.2.2.3	Datový typ Long.....	67
5.2.2.4	Datový typ Float.....	67
5.2.2.5	Datový typ Double .....	67
5.2.2.6	Datový typ Char .....	68
5.3	UKÁZKOVÉ PROGRAMY JEDNOTLIVÝCH PERIFERÍ.....	68
5.3.1	Klávesnice .....	68
5.3.2	LCD Display .....	69
5.3.3	Krokový motor .....	69
5.3.4	Ultrazvukový senzor.....	70
5.3.5	Modul relé .....	71
5.3.6	Modul s RGB LED .....	72
<b>6</b>	<b>SADA ŘEŠENÝCH ÚLOH.....</b>	<b>74</b>
6.1	ÚLOHA 1- VÝPIS ZNAKŮ KLÁVESNICE NA DISPLEJ.....	74
6.1.1	Cíle výuky:.....	74
6.1.2	Zadání: .....	74
6.1.3	Potřebné vybavení: .....	74
6.1.4	Předpokládaná časová dotace:.....	75
6.1.5	Schéma zapojení jednotlivých komponent .....	75
6.1.6	Vytvoření programu .....	76
6.1.7	Bonusový úkol.....	76

6.2	ÚLOHA 2 - VÝPIS VZDÁLENOSTI NA LCD DISPLEJ .....	76
6.2.1	Cíle výuky: .....	76
6.2.2	Zadání: .....	77
6.2.3	Potřebné vybavení: .....	77
6.2.4	Předpokládaná časová dotace: .....	77
6.2.5	Schéma zapojení .....	77
6.2.6	Vytvoření programu .....	77
6.2.7	Bonusový úkol.....	78
6.3	ÚLOHA 3 – OVLÁDÁNÍ RELÉ MODULU POMOCÍ KLÁVESNICE .....	78
6.3.1	Cíle výuky: .....	78
6.3.2	Zadání: .....	79
6.3.3	Potřebné vybavení: .....	79
6.3.4	Předpokládaná časová dotace: .....	79
6.3.5	Schéma zapojení .....	80
6.3.6	Vytvořený program.....	80
6.3.7	Bonusový úkol.....	83
6.4	ÚLOHA 4 – OVLÁDÁNÍ RGB LED MODULU .....	84
6.4.1	Cíle výuky: .....	84
6.4.2	Zadání: .....	84
6.4.3	Potřebné vybavení: .....	84
6.4.4	Předpokládaná časová dotace: .....	84
6.4.5	Schéma zapojení .....	85
6.4.6	Vytvořený program.....	85
6.4.7	Bonusový úkol.....	86
6.5	ÚLOHA 5– AKTIVACE KROKOVÉHO MOTORU POMOCÍ BEZDOTYKOVÉHO SNÍMAČE .....	86
6.5.1	Cíle výuky: .....	86
6.5.2	Zadání: .....	86
6.5.3	Potřebné vybavení: .....	86
6.5.4	Předpokládaná časová dotace: .....	86
6.5.5	Schéma zapojení .....	87
6.5.6	Vytvořený program.....	87
<b>7</b>	<b>PREZENTACE .....</b>	<b>89</b>
	<b>ZÁVĚR .....</b>	<b>90</b>
	<b>SEZNAM POUŽITÉ LITERATURY .....</b>	<b>91</b>
	<b>SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....</b>	<b>95</b>
	<b>SEZNAM OBRÁZKŮ .....</b>	<b>96</b>
	<b>SEZNAM TABULEK .....</b>	<b>99</b>
	<b>SEZNAM PŘÍLOH .....</b>	<b>100</b>



## ÚVOD

Platforma Arduino je vytvořena za účelem zkvalitnění a zefektivnění výuky za pomoci vývojové desky Arduino, kterou můžeme stále častěji nalézt na školách. Platforma Arduino má sloužit jako pomůcka při výuce na základních a středních školách, jelikož je velice rozšířena již existuje několik verzí, modifikací, a stavebnic, které se snaží rozvíjet zájem o programování Arduino. Mezi hlavní nedostatky již prodávaných stavebnic je nepřizpůsobení konstrukce do výuky na školách. Jelikož na školách je ve většině případů požadováno, aby byla možnost pokračovat na již rozpracovaném projektu to může být a běžně prodávaných stavebnic problém. Stavující stavebnice Arduino jsou sice vybaveny velkým množstvím komponentů, ale nemají možnost pevného spojení a jednoduchého přemísťování a uložení pro pozdější práci. Dostupné učební materiály jsou velmi kvalitní, ale neberou v úvahu hardwarové uložení a rozpoložení komponentů.

Cílem této práce je vytvořit jednoduchou a funkční platformu sloužící pro přípravu žáků do výuky programování, a seznámení se s platformou Arduino a využíváním různých komponentů, se kterými se mohou setkávat v běžném životě. Žáci tak budou mít možnost seznámit s těmito komponenty více do hloubky a mít možnost je ovládat a nasimulovat tak nejrůznější situace a mít možnost je použít ve vlastním projektu. Dále je platforma koncipovaná tak, aby byla možnost rozšiřovat a obměňovat různé komponenty, a tak mít nepřehledné množství úkolů a možných situací.

V teoretické části jsem se věnoval vývojem informatiky na jednotlivých stupních základních a středních škol. Dále je pak rozebrán samotný vývoj vývojových desek Arduino. Představuji různé vývojové desky, dále pak zmiňuji i klony, které jsou dostupné a dají se nahradit za platformu Arduino. Popisuji, zde také různé alternativy k platformě Arduino, které je možné využít pro výuku na školách jak základních, tak i středních.

V praktické části se zabývám samotným návrhem, stavbou a realizací výukové platformy, která se osazena vybranými komponenty, tak aby si žáci osvojili práci a funkci na vybraných komponentech. K jednotlivým komponentům je popsán ukázkový program tak aby měli základní povědomí o principů programování. V této práci se také nachází zpracované podklady pro výuku s touto platformou a prezentace sloužící jako podpora výuky pro tuto platformu.

## **I. TEORETICKÁ ČÁST**

## 1 INFORMATIKA A JEJÍ VÝUKA

Informatika je vzdělávací oblast, ve které se především rozvíjí inforatické myšlení a porozumění základním informacím a principům digitálních technologií. Výuka informatiky je založena především na aktivních činnostech žáků, kteří využívají především inforatické pojmy a postupy. Informatika je také o hledání a aplikování vhodného řešení daného problému. Výuka informatiky má také za úkol naučit žáky chápat, jak digitální technologie fungují, jak s nimi pracovat a umět se rozhodnout, kdy je mnohem efektivnější nechat za nás pracovat počítače a jaké vznikání rizika při využívání digitálních technologiích.

Na prvním stupni základního vzdělání můžeme výuku informatiky vést ve formě her a experimentů. Práce s daty je zda omezená, ale i přesto na základě dat jsou rozvíjeny schopnosti analyzovat a popsat možné řešení problému. Jsou také vedeni k osvojením návyků, které jsou spojené s rizikovým chováním a využíváním digitálních technologiích.

Na druhém stupni základního vzdělávání žáci tvoří, experimentují a aktivně hledají řešení vzniklých problémů. Učí se základním inforatickým konceptům, souvislostem a fungování digitálních technologií. Díky poznávání toho, jak a proč digitální technologie fungují, žáci chápou základní principy kódování a modelování. Začínají se také seznamovat s různými zařízeními jako jsou právě robotické stavebnice nebo vývojová deska Arduino. [1]

Na středních školách je výuka informatiky a inforatického myšlení velmi různorodá a odlišná. Na technických středních školách se inforatice věnují mnohem více, a proto jsou jejich dovednosti v digitálních technologiích mnohem komplexnější než na střeních školách, které nekladou takový důraz na inforatické a digitální dovednosti.

### 1.1 Cílové zaměření vzdělávací oblasti

Vzdělávání v dané vzdělávací oblasti směřuje k utváření a rozvíjení klíčových kompetencí tím, že vede žáka k:

- systémovému přístupu při analýze situací a jevů světa kolem něj
- nacházení různých řešení a výběru toho nejvhodnějšího pro danou situaci
- ke zkušenosti, že týmová práce umocněná technologiemi může vést k lepším výsledkům než samostatná práce
- porozumění různým přístupům ke kódování informací i různým způsobům jejich organizace

- rozhodování na základě relevantních dat a jejich korektní interpretace, jeho obhajování pomocí věcných argumentů
- komunikaci pomocí formálních jazyků, kterým porozumí i stroje
- standardizování pracovních postupů v situacích, kdy to usnadní práci
- posuzování technických řešení z pohledu druhých lidí a jejich vyhodnocování v osobních, etických, bezpečnostních, právních, sociálních, ekonomických, environmentálních a kulturních souvislostech
- nezdolnosti při řešení těžkých problémů, zvládání nejednoznačnosti a nejistoty a vy pořádání se s problémy s otevřeným koncem
- otevřenosti novým cestám, nástrojům, snaze postupně se zlepšovat [1]

## 1.2 Změna ve výuce informatiky

Podle průzkumu, které MŠMT provádělo spolu s několika pedagogickými fakultami a desítkami škol v letech 2018 a 2020 provedlo ověřování nové informatiky a digitální gramotnosti. Na základě tohoto průzkumu pak ministerstvo revidovalo rámcový vzdělávací program a souběžně vytvořilo nové učebnice, metodické materiály a vzdělávací programy pro vyučující.

Nově by tedy vyučující měli v hodinách kromě základů uživatelských dovedností pro práci se zařízeními a aplikacemi klást důraz také na informatické myšlení. Žáci by se měli učit pracovat s informacemi, měli by umět analyzovat problém a hledat funkční řešení. Vedle porozumění principům digitálních technologií bude podstatné také jejich bezpečné a etické využívání a týmová práce.

Novým způsobem se informatika začne vyučovat na všech základních školách nejpozději od 1. září 2023, a to ve všech ročnících prvního stupně, a nejpozději od 1. září 2024 ve všech ročnících druhého stupně. Hodinová dotace se změní z jedné hodiny na prvním a na druhém stupni na dvě a čtyři hodiny. Díky náběhovému období pro zavedení změn mají školy dostatek času na přípravu učitelů, vybavení a úpravu svého školního vzdělávacího programu. Součástí podpory jsou modelové školní vzdělávací programy. [2]

### 1.3 Klíčové kompetence

Klíčové kompetence představují souhrn vědomostí, dovedností, schopností, postojů a hodnot důležitých pro osobní rozvoj a uplatnění každého žáka. Žák si rozvíjí veškeré kompetence v průběhu studia a jelikož je osvojení těchto kompetencí dlouhý a složitý proces.

Mezi základní kompetence, které jsou:

#### **Kompetence k učení**

- vybírá a využívá pro efektivní učení vhodné způsoby, metody a strategie, plánuje, organizuje a řídí vlastní učení, projevuje ochotu věnovat se dalšímu studiu a celoživotnímu učení

#### **Kompetence k řešení problémů**

- vnímá nejrůznější problémové situace ve škole i mimo ni, rozpozná a pochopí problém, přemýšlí o nesrovnalostech a jejich příčinách, promyslí a naplánuje způsob řešení problémů a využívá k tomu vlastního úsudku a zkušeností

#### **Kompetence komunikativní**

- formuluje a vyjadřuje své myšlenky a názory v logickém sledu, vyjadřuje se výstižně, souvisle a kultivovaně v písemném i ústním projevu

#### **Kompetence sociální a personální**

- účinně spolupracuje ve skupině, podílí se společně s pedagogy na vytváření pravidel práce v týmu, na základě poznání nebo přijetí nové role v pracovní činnosti pozitivně ovlivňuje kvalitu společné práce

#### **Kompetence občanské**

- respektuje přesvědčení druhých lidí, váží si jejich vnitřních hodnot, je schopen vcítit se do situací ostatních lidí, odmítá útlak a hrubé zacházení, uvědomuje si povinnost postavit se proti fyzickému i psychickému násilí

#### **Kompetence pracovní**

- používá bezpečně a účinně materiály, nástroje a vybavení, dodržuje vymezená pravidla, plní povinnosti a závazky, adaptuje se na změněné nebo nové pracovní podmínky

## Kompetence digitální

Dnes již jedna z důležitých kompetencí každého žáka, jelikož se hojně obklopují digitální technologie všude kolem nás. Proto by na konci vzdělávání měli mít žáci dostatečný přehled o fungování a používání digitálních technologií a to především:

- ovládá běžně používaná digitální zařízení, aplikace a služby, využívá je při učení i při zapojení do života školy a do společnosti; samostatně rozhoduje, které technologie, pro jakou činnost či řešený problém použít
- získává, vyhledává, kriticky posuzuje, spravuje a sdílí data, informace a digitální obsah, k tomu volí postupy, způsoby a prostředky, které odpovídají konkrétní situaci a účelu
- vytváří a upravuje digitální obsah, kombinuje různé formáty, vyjadřuje se za pomoci digitálních prostředků
- využívá digitální technologie, aby si usnadnil práci, zautomatizoval rutinní činnosti, zefektivnil či zjednodušil své pracovní postupy a zkvalitnil výsledky své práce
- chápe význam digitálních technologií pro lidskou společnost, seznamuje se s novými technologiemi, kriticky hodnotí jejich přínosy a reflektuje rizika jejich využívání
- předchází situacím ohrožujícím bezpečnost zařízení i dat, situacím s negativním dopadem na jeho tělesné a duševní zdraví i zdraví ostatních, při spolupráci, komunikaci a sdílení informací v digitálním prostředí jedná eticky [3]

## 2 PLATFORMY PRO VÝUKU PROGRAMOVÁNÍ

Platforma pro výuku programování má mnoho podob a možné modifikace. Jako platformu si můžeme představit desku Arduino a k tomu připojené nepájivé pole, nebo sadu pro stavbu robota, kterého následně naprogramujeme.

### 2.1 Arduino

Arduino je jedna z nejrozšířenějších platforem která se využívá pro výuku na základních a středních školách. Jedná se o otevřenou elektronickou platformu, založenou na jednoduché počítačové desce (hardware) a vývojovém prostředí, které slouží k tvorbě software. Arduino je jednou z nejrozšířenějších a nejpoužívanějších platforem u nás. Díky její jednoduchosti a univerzálnosti se rychle rozšířila a stala se velice oblíbenou, což představuje důvod, proč se s ní můžeme potkávat i na některých školách u nás. Vyznačuje se především velkým množstvím rozšiřujících komponentů a shieldů, které ve spojení s různými typy pinů dělá z této platformy vhodný nástroj jak pro výuku, tak i pro sestavování projektů jak těch jednoduchých, tak i těch komplexních. [4]

### 2.2 Historie

Vývoj prvního Arduina započal v roce 2005, když se členové italského Interaction Design Institute ve městě Ivrea rozhodli vytvořit jednoduchý a levný vývojový set pro studenty, kteří si nechtěli pořizovat, v té době rozšířené a drahé desky BASIC Stamp. Mezi studenty se Arduino uchytilo, a tak se tvůrci rozhodli poskytnout ho celému světu. A to nejenom prodejem vlastních desek, ale i sdílením všech schémat a návodů. Jedná se tedy o Open Source projet. [1]

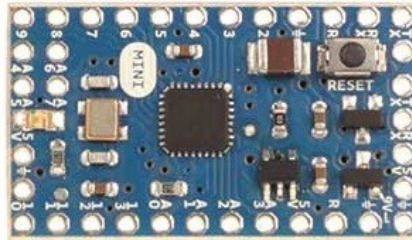
### 2.3 Typy desek

Platforma arduino má více různých vývojových desek, které jsou díky rozlišným konstrukcím vhodné do odlišných projektů, což napomáhá také k tomu, aby uživatel měl široké možnosti pro vytváření svého projektu. Základem každé desky je procesor od firmy Atmel.

#### 2.3.1 Arduino Mini

Arduino Mini je asi nejmenší oficiální verze Arduina, navržená pro úsporu místa. Daní za malé rozměry je však absence USB portu. K programování je tedy nutné použít externí převodník. Běží na procesoru ATmega328 (dříve ATmega168) s taktem 16 MHz. Pro své malé

rozměry je vhodný k použití například v chytrých vypínačích, či dálkových ovladačích a všude tam, kde jsou zapotřebí malé rozměry desky. [4]



Obrázek 1- Arduino Mini [4]

### 2.3.2 Arduino Nano

Arduino je výbavou a výkoností velmi podobný Arduino Mini. Hlavním rozdílem je již zabudovaný USB port přímo na desce a čímž je zajištěn větší komfort při programování desky, odpadá použití externího převodníku.



Obrázek 2 – Arduino Nano [4]

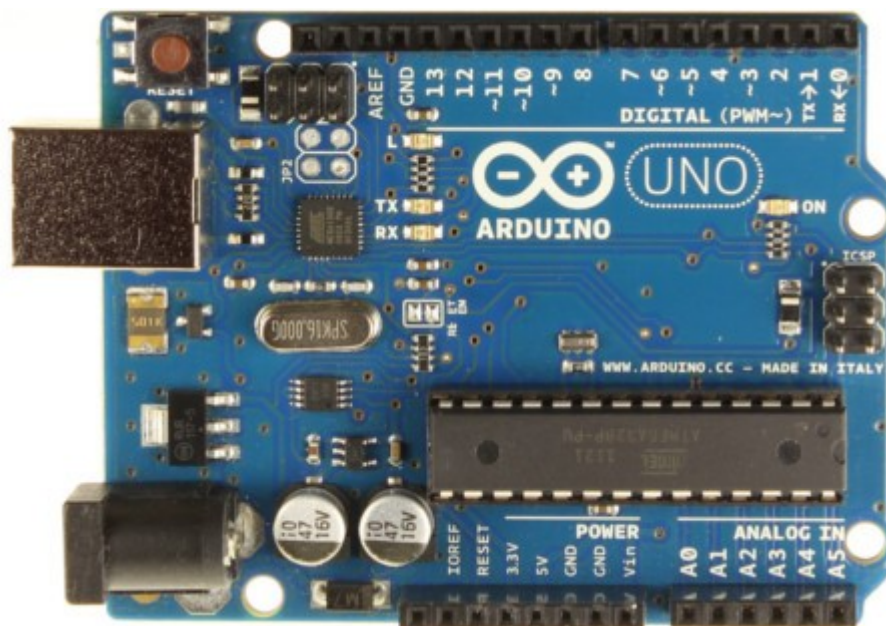
Tabulka 1 Technické specifikace Arduino Nano [4]

Mikrokontrolér	ATmega328	Stejnoseměrný proud na I/O piny	40 mA (I/O kolíky)
Architektura	AVR	Vstupní napětí	7-12V
Provozní napětí	5 V	Digitální I/O piny	22 (z toho 6 PWM)
Flash paměť	32 KB, z toho 2 KB využívá bootloader	PWM výstup	6
SRAM	2 kB	Spotřeba energie	19 mA
Rychlost hodin	16 MHz	Velikost PCB	18 x 45 mm
Analogové piny IN	8	EEPROM	1 kB



### 2.3.3 Arduino Uno

Arduino Uno je v současné době asi nejčastěji používaný typ desky. Je přímým pokračovatelem hlavní vývojové linie, která započala prvním Arduinem se sériovým portem místo USB, pokračující přes Arduino Extreme, NG, Diecimila a Duemilanove až k dnešnímu Uno. Na desce najdeme procesor ATmega328 a již klasické USB. [4]



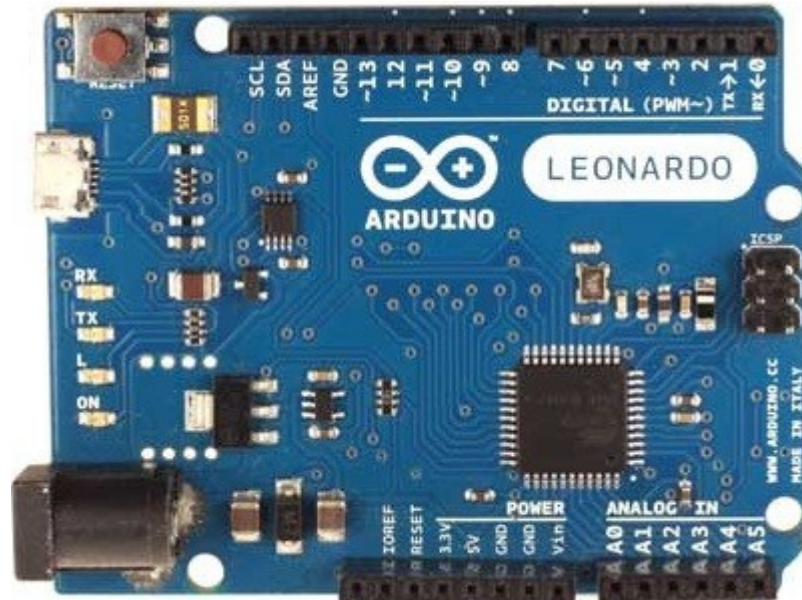
Obrázek 3 – Arduino Uno [4]

Tabulka 2 Technické specifikace Arduino Uno [4]

Mikrokontrolér	ATmega328P	Stejnoseměrný proud pro pin 3,3V	50 mA
Provozní napětí	5V	Flash paměť	32 KB
Vstupní napětí (doporučeno)	7-12V	SRAM	2 KB
Vstupní napětí (limit)	6-20V	EEPROM	1 kB
Digitální I/O piny	14 (z toho 6 poskytuje PWM výstup)	Rychlost hodin	16 MHz
Digitální I/O piny PWM	6	LED_BUILTIN	13
Analogové vstupní piny	6	Délka	68,6 mm

### 2.3.4 Arduino Leonardo

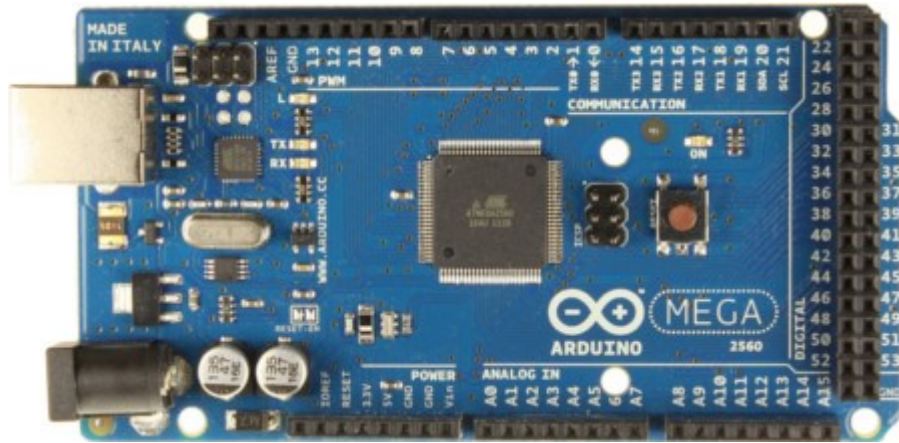
Arduino Leonardo designově navazuje na Arduino Uno. Liší se však použitým čipem. Tím je ATmega32u4, stejný jako je použit v Arduinu Micro. Konstrukce a množství pinů je stejný jako u Arduina Uno. [4]



Obrázek 4 - Arduino Leonardo [4]

### 2.3.5 Arduino Mega

S Arduino Mega2560 je skupina desek, jejichž vzhled vznikl prodloužením designu Arduina Uno. Zvětšení rozměrů přináší prostor pro větší a výkonnější čipy a také více pinů. Předchozí verzí bylo Arduino Mega1280. Hodí se tam, kde je zapotřebí právě většího počtu připojených komponentů [4]



Obrázek 5- Arduino Mega [4]

Tabulka 3 Technické údaje Arduino Mega [4]

Mikrokontrolér	ATmega2560	Flash paměť	256 KB
Provozní napětí	5V	SRAM	8 kB
Vstupní napětí (doporučeno)	7-12V	EEPROM	4 kB
Vstupní napětí (limit)	6-20V	Rychlost hodin	16 MHz
Digitální I/O piny	54 (z toho 15 poskytuje PWM výstup)	LED_BUILTIN	13
Analogové vstupní piny	16	Délka	101,52 mm
Stejnsměrný proud na I/O pin	20 mA	Šířka	53,3 mm
Stejnsměrný proud pro pin 3,3V	50 mA	Hmotnost	37 g

### 2.3.6 Arduino Robot

Arduino Robot je set se kterým je možné si jednoduše vytvořit svého vlastního chytrého robota. Na desce se nachází nejenom mikroprocesor ATmega, ale i mikro USB port a elektronický kompas. [4]



Obrázek 6 – Arduino Robot [4]

## 2.4 Klony

Díky tomu že Arduino je založeno na maximální otevřenosti, jsou dostupné zdrojové kódy a podklady pro vytvoření kompletní vývojové desky. Díky tomu můžeme nalézt mnoho obdobných vývojových desek, které jsou postaveny na základu od Arduina a mohou být kompatibilní s originálními produkty Arduino.

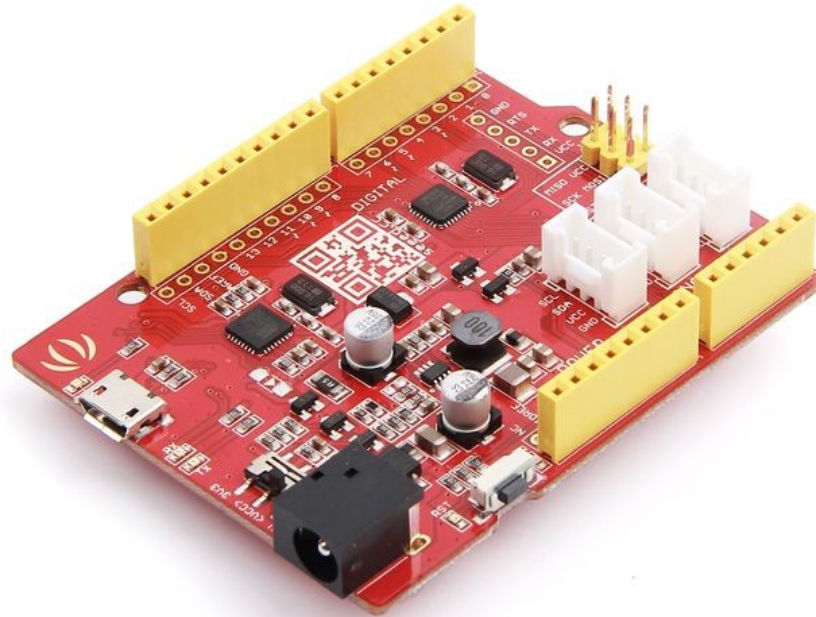
Arduino klony by se daly rozdělit podle toho, jak moc se liší od oficiálních Arduino desek. Můžeme totiž najít klony, které jsou prakticky totožné s originálem, liší se například jenom v barvě. Na druhé straně tohoto dělení leží „exotičtější“ desky, jako je například již zmíněné Rainbowduino, nebo Touch-Board. Ty se tvarem původním deskám vůbec nepodobají a mohou být vybavené i různým specializovaným hardware pro konkrétní aplikace – řízení RGB maticového displeje u Rainbowduino, nebo kapacitní senzory u Touch-Board. [5]

### 2.4.1 Seeduino V4.2

Seeduino V4.2 je vývojová deska kompatibilní s Arduinem UNO, na bázi mikrokontroléru ATmega328p. Jedná se o jeden z nejlepších klonů Arduina UNO, oproti ostatním klonům i originálu nabízí mnoho užitečných funkcí navíc.

Seeduino má převodník UART na USB ATmega16u2 a mikrokontrolér ATmega328p jako originální desky Arduino. Desku lze programovat přes micro-USB kabel, který je využíván

u mobilních telefonů. Na desce je umístěn přepínač pro přepínání napětí logiky. Tímto způsobem si můžete vybrat mezi 3.3 V a 5 V podle periférií, které chcete používat. Seeeduino má také tři konektory Grove pro připojení periférií od Seed Studia. Seeeduino V4.2 je kompatibilní se shiely pro UNO R3. [6]



Obrázek 7 Seeeduino V4.2 [6]

#### 2.4.2 Rainbowduino v3.0

Deska Rainbowduino je řadič kompatibilní s Arduino s profesionálním multiplexovaným LED ovladačem. Může řídit 8x8 RGB LED Matrix nebo 4x4x4 RGB LED Krychle v režimu společné anody. Rainbowduino v3.0 používá dva čipy MY9221, což je 12kanálový (R/G/B x 4) konstantní proud Adaptive Pulse Density Modulation (APDM). Rainbowduino v3.0 má ustanovení pro kaskádování více takových desek s rozhraním I2C.

Rainbowduino v3.0 je vybaveno zavaděčem Arduino, což usnadňuje programování náčrtů pomocí Arduino IDE. Na rozdíl od jiných ovladačů LED je tento dodáván s vestavěným rozhraním USB na UART (FT232RL) pro programování náčrtů. [7]



Obrázek 8 Rainbowduino v3.0[7]

Tabulka 4 Technické údaje Rainbowduino [7]

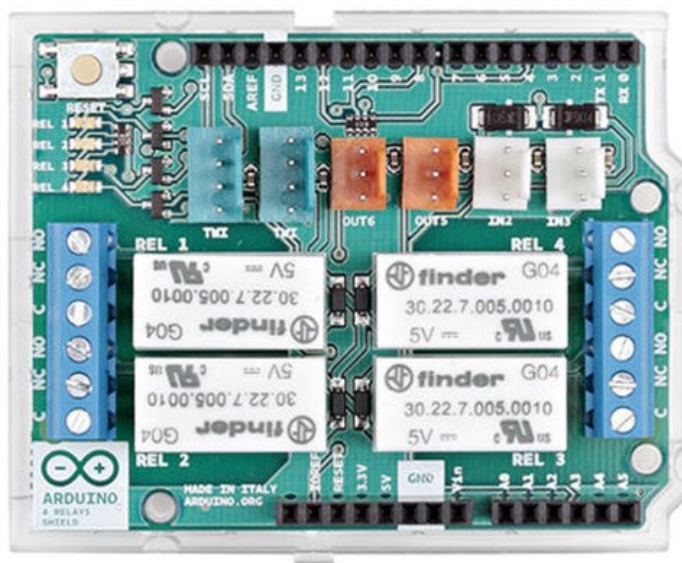
Konstantní proudový výstup	20,8 mA
Špendlíkové hlavičky	2 x 16 pro připojení multiplexních LED
Maximální jízdní schopnost LED	192 (tj. 8x8x3)
Podporované LED displeje	4x4x4 RGB kostka a 8x8 RGB LED Matrix
USB na UART	FT232RL
Napájení DC	Napájení přes USB nebo externí 6V~12V DC
Vestavěný regulátor napětí	5V / 1A

## 2.5 Shield

Základní desky arduino nemusí být vhodné pro všechny aplikace, a proto se specifické použití arduino nabízí rozšiřující desky v podobě shieldů. Takovýmto typem rozšiřující desky uděláme jednoduše zařízení pro specifické ovládání periférií, jakým mohou být například servomotory, krokové motory, řada senzorů nebo dokonce i výkonové spínané zařízení. Tyto desky jsou však vhodné jen na velmi specifické operace.

### 2.5.1 Arduino 4 Relé Shield

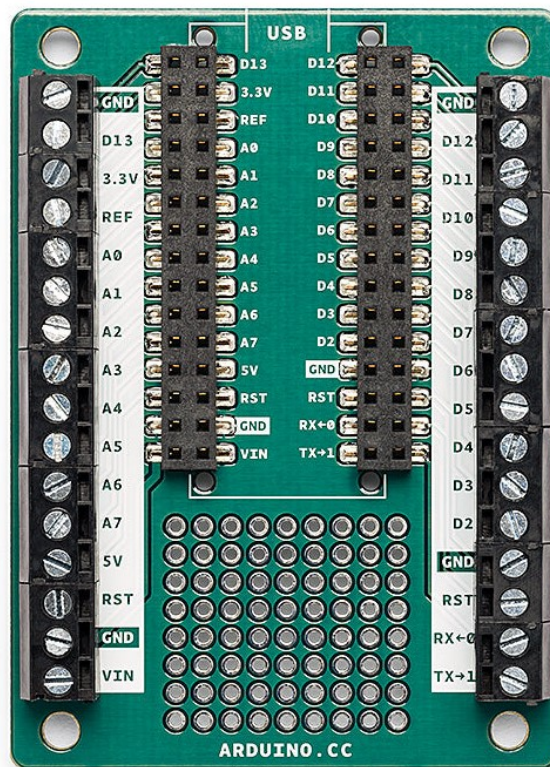
Arduino 4 Relays Shield je řešení pro řízení zátěží s vysokým výkonem, které nelze ovládat digitálními IO Arduino kvůli proudovým a napěťovým limitům ovladače. Shield obsahuje čtyři relé, každé relé poskytuje 2 pólové přepínací kontakty za účelem zvýšení proudového omezení každého výstupu byly 2 přepínací kontakty zapojeny paralelně. Čtyři LED indikují stav zapnutí/vypnutí každého relé. [8]



Obrázek 9 Arduino 4 Relé Shield [8]

### 2.5.2 Shield šroubového terminálu na Arduino Nano

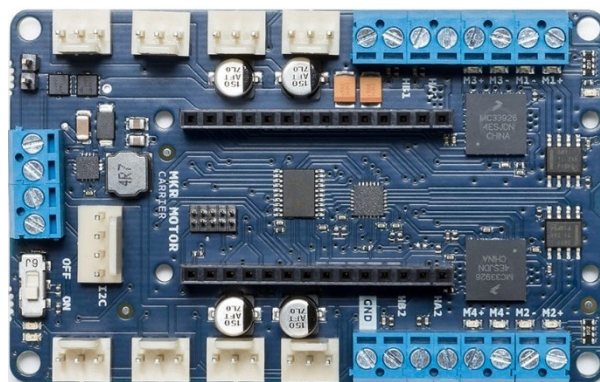
Jedná se o shield pro vytváření robustních projektů, kde je kladen důraz na pevné spolehlivé spojení komponentů, bez nutnosti pájení a případné jednoduché výměny či záměně jednotlivých komponentů. Vytvářejte projekty založené na desce Arduino Nano s bezpečným připojením pomocí šroubových svorek a v případě, že potřebujete přidat nějaké další komponenty, vhodné pro použití v oblasti prototypování.



Obrázek 10 Shield šroubového terminálu na Arduino Nano [9]

### 2.5.3 Schield pro připojení motorů a senzorů

Tento schield je přídatní deska pro připojení a ovládání různých typů motorů, jedná se o servomotory, stejnosměrné motory a krokové motory. K schieldu je také možné připojit senzory nebo jiné akční členy, pomocí 3 pinových konektorů.



Obrázek 11 Schield pro připojení motorů a senzorů [9]



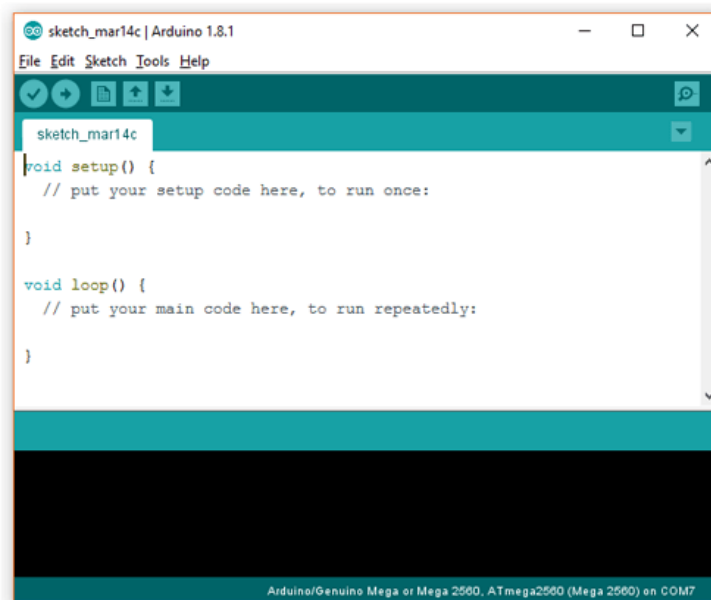
## 2.6 Tvorba programu na Arduino platformu

Dnes již máme několik možností, jak Arduino programovat. Nyní můžeme programovat jak ve standartním programovacím jazyce jako například jazyk C/C++ tak i pomocí sestavování bloků, které je pro začínající žáky mnohdy jednodušší než psaní samotného kódu.

### 2.6.1 Arduino IDE

Arduino IDE (Integrated Development Environment) je oficiální vývojové integrované prostředí, které je přímo vydáno pro platformu Arduino. Vývojové prostředí Arduino IDE obsahuje textový editor pro psaní kódu, textovou konzoli, panel nástrojů a spoustu dalších užitečných funkcí, které nám napomáhají při vývoji programu.[10]

Programy napsané pomocí softwaru Arduino (IDE) se nazývají skici. Tyto skici jsou napsány v textovém editoru a jsou uloženy s příponou .ino. Editor má funkce pro vyjímání/vkládání a pro vyhledávání/nahrazování textu. Oblast zpráv poskytuje zpětnou vazbu při ukládání a exportu a také zobrazuje chyby. Konzole zobrazuje textový výstup softwaru Arduino (IDE), včetně kompletních chybových zpráv a dalších informací. V pravém dolním rohu okna se zobrazuje nakonfigurovaná deska a sériový port. Tlačítka panelu nástrojů umožňují ověřit a nahrát programy, vytvořit, otevřít a uložit náčrty a otevřít sériový monitor. [11]



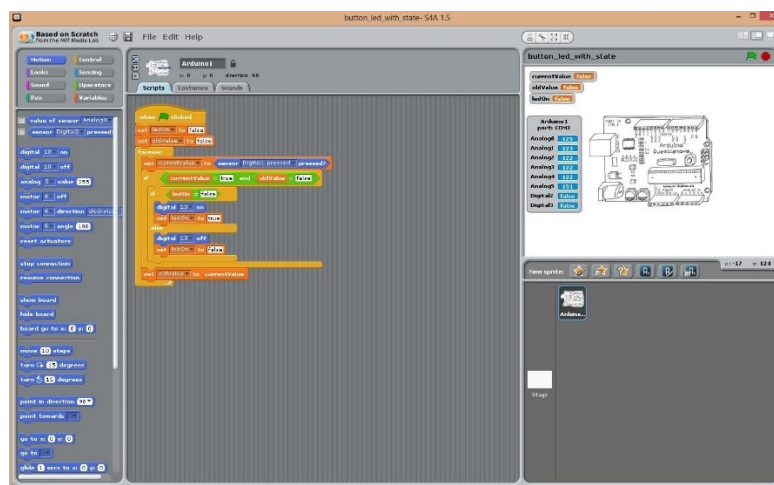
Obrázek 12 Vývojové prostředí Arduino IDE [12]

### 2.6.2 Scratch pro Arduino

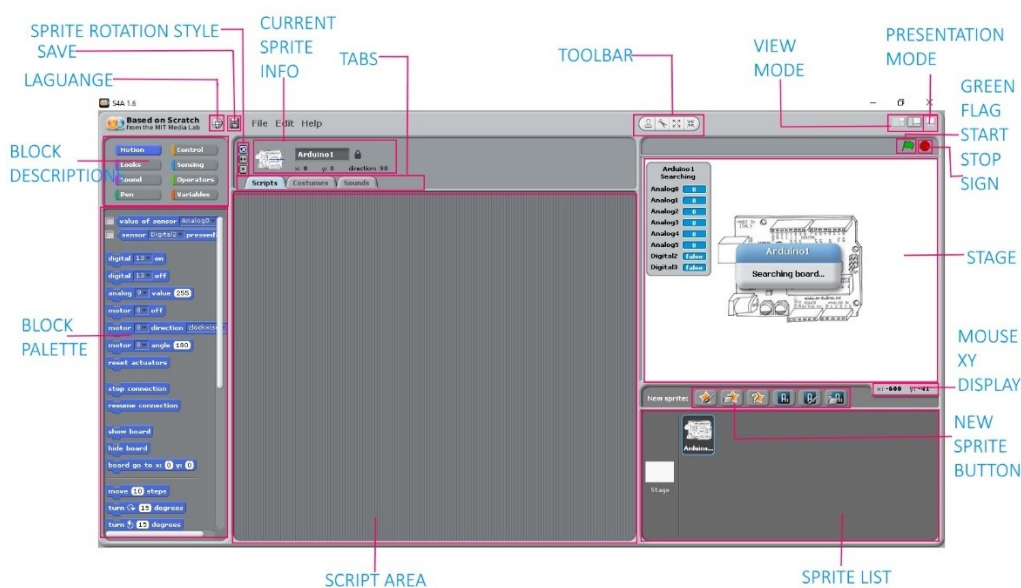
Scratch pro Arduino je open source projekt, který vznikl za účelem přilákat více lidí do světa programování a zjednodušit užívání platformy Arduino.

Scratch pro Arduino je modifikace známého programu Scratch, který má již velké zastoupení v našem školství a hojně se využívá pro učení žáků na našich školách. To je velké výhoda pro žáky, kteří by začínali používat platformu Arduino ve výuce.

Mezi nevýhody je potřeba zmínit fakt, že Scratch pro Arduino je program třetích stran, a tak nemusí správně komunikovat se všemi deskami Arduino, program je testován na Arduino Diecimila, Duemilanove a Uno. [13]



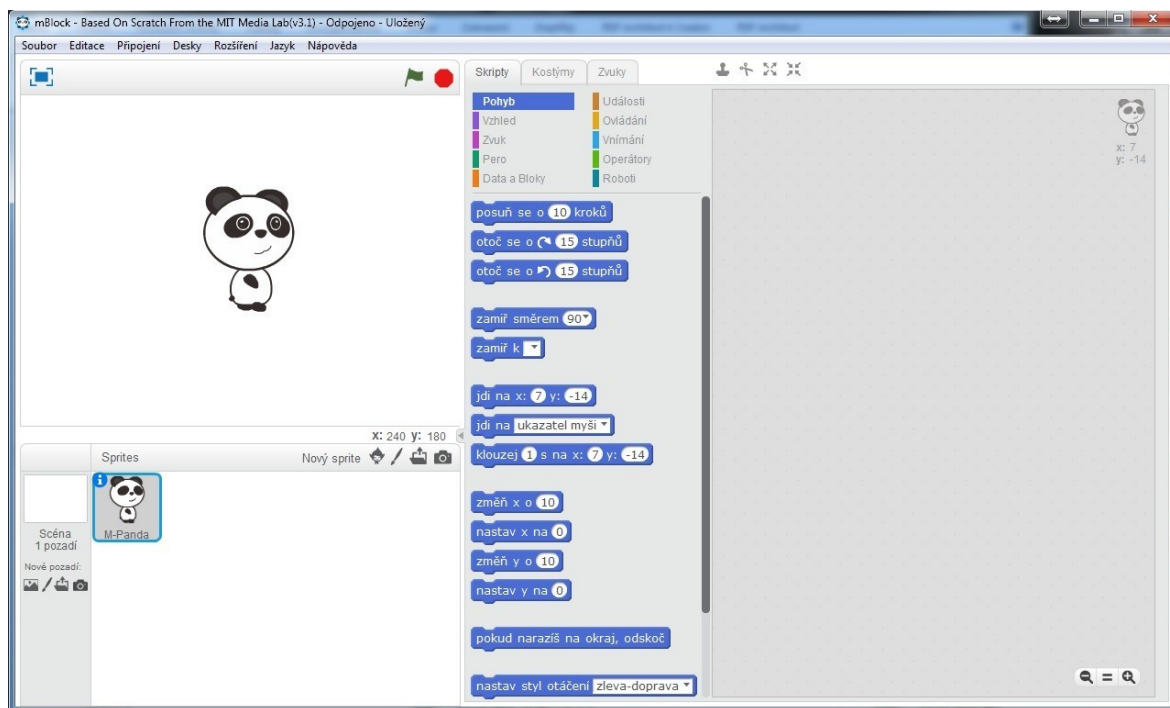
Obrázek 13 Scratch pro Arduino [13]



Obrázek 14 Rozhraní Scratch pro Arduino [13]

### 2.6.3 mBlock

Program mBlock a opět program založený na známém programu Scratch, který stejně jako program předešlý nabízí přívětivé grafické programování, které umožňuje jednoduché programování s Arduino deskami. Rozdíl oproti Scratch pro Arduino je jeho české lokalizace, což může být pro mnohé mnohem přívětivější než verze v cizím jazyce. [14]

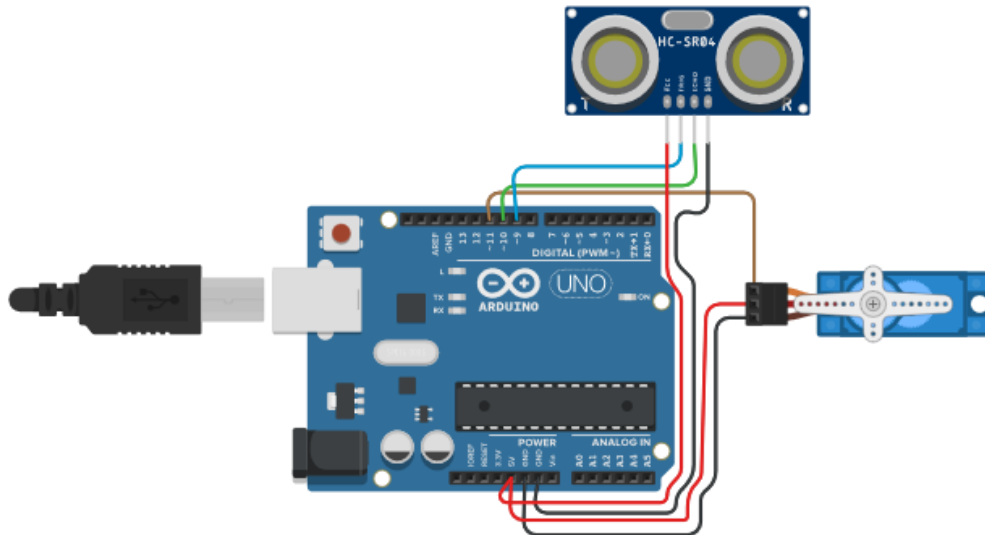


Obrázek 15 Vývojové prostředí mBlock [14]

### 2.6.4 AUTODESK Tinkercad

Tinkercad je online platforma, která uživatelům umožňuje navrhovat a simulovat 3D modely. Je to bezplatný a snadno použitelný nástroj, který je určen pro začátečníky i odborníky. S Tinkercad mohou uživatelé vytvářet 3D modely objektů, budov a další a poté je exportovat pro 3D tisk nebo použít v jiných projektech. Tinkercad má také řadu funkcí, díky kterým je ideální pro vzdělávací účely, jako jsou plány lekcí, návody a nápady na projekty.

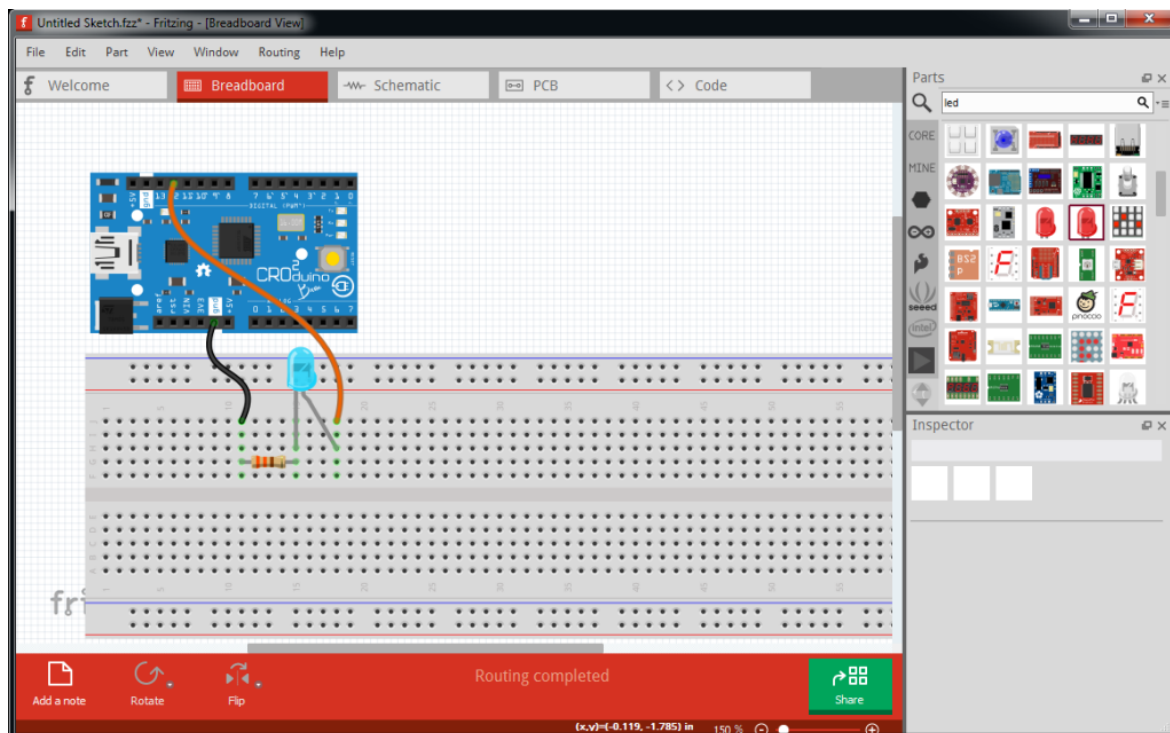
Jednou z nejzajímavějších věcí na Arduino a Tinkercadu je, že je lze používat společně. Tinkercad má řadu elektronických součástek, které lze použít k sestavení obvodů a ovládání zařízení, a má také vestavěný simulátor Arduino, který uživatelům umožňuje programovat a testovat své obvody ve virtuálním prostředí. Díky tomu je snadné prototypování a testování nových nápadů před jejich vytvořením, to vše je možné vytvářet v reálném čase. [15]



Obrázek 16 AUTODESK Tinkercad [16]

### 2.6.5 Fritzing

Fritzing je open-source program, který vyvinula univerzita v Postupimi, stejně jako předešlé programy frizing nabízí mnoho užitečných funkcí, které pomohou při vývoji různých programů. Frizing je komplexní program kde si lze sestavit obvod na nepájivém poli, sestavit schéma, nebo vytvořit přímo originální desku. [17]



Obrázek 17 Fritzing [17]

## 2.7 Lego Mindstorms

Lego Mindstorms je edukační stavebnice, kterou vyvinula společnost Lego. Stavebnice se zaměřena na stavbu robotů a následně jejich programování, které je díky propracovanému softwaru velmi jednoduché a pro žáky zábavné. Proto se tyto stavebnice hojně používají na školách, právě pro základy výuky na programátorských a algoritmizačních technik. Lego Mindstorms se postupně vyvíjelo do několika generací.

### 2.7.1 Lego Mindstorms RCX

Lego Mindstorms RCX je první generace Lego, která využívala 8bitový jednočipový počítač Renesas H8/300 microcontroller jako svoji CPU a RAM s kapacitou 32Kb. Kostka je naprogramována stažením programu z počítače s OS Windows nebo Mac do RAM přes speciální infračervené rozhraní. Po spuštění programu umožňuje RCX kostka samostatné fungování Mindstorms, který reaguje na vnitřní a vnější podněty podle naprogramovaných instrukcí. RCX kostky mohou ve větších počtech spolu vzájemně komunikovat přes infračervené rozhraní. Kromě infračerveného portu jsou k dispozici tři senzorové vstupní porty a tři motorové výstupní porty. Vestavěný LCD může zobrazovat úroveň baterie, stav vstupních a výstupních portů nebo program, který je aktuálně spuštěn.[18]

RCX kostka verze 1.0 disponuje konektorem napájecího adaptéru, který na rozdíl od baterií umožňuje nepřetržitý chod, který byl odstraněn ve verzi 2.0.



Obrázek 18 Lego Mindstorms RCX [18]

### 2.7.2 Lego Mindstorms NXT

LEGO Mindstorms NXT je druhou generací a byla vydána ve dvou verzích. V roce 2006 byla vydána první verze (NXT 1.0), druhá verze (NXT 2.0) byla vydána v roce 2009, obě generace nabízí 32-bit microprocessor, velký matrix displej, 4 vstupní a 3 výstupní porty, Bluetooth a USB komunikační port. [19]



Obrázek 19 Lego Mindstorms NXT [19]



Obrázek 20 Lego Mindstorms NXT periferie [19]

### 2.7.3 Lego Mindstorms Robotí vynálezce

LEGO Mindstorms Robotí vynálezce je poslední verze od společnosti Lego. S pomocí bezplatné aplikace, kterou vydává samotné Lego je možná postavit až 5 unikátních motorizovaných robotů, které lze jednoduše sestavit pomocí návodu. Všechny výtvoři pak mohou přivést k životu pomocí intuitivního programovacího prostředí založeného na jazyku Scratch a pustit se do plnění zábavných aktivit a výzev. Jedná se o zcela nový koncept programovatelných robotů od LEGO MINDSTORMS, kteří jsou vybaveni inteligentním hubem s 5x5 LED obrazovkou, 6osým gyroskopem, reproduktorem, výkonnými motory a přesnými

senzory, pomůžou mladým vynálezčům rozvíjet kreativitu a osvojit si dovednosti STEM, které jsou důležité pro 21. století. [20]



Obrázek 21 Lego Mindstorms Robotí vynálezce [21]



### 3 VÝVOJOVÉ SESTAVY ARDUINO

Vývojové sestavy Arduino jsou základními sestavami, které si lze pořídit. Jsou vhodné pro prvotní seznámení s platformou Arduino a s funkcí jednotlivých komponent.

#### 3.1 UNO R3 Ultimate Starter Kit

Jedná se o základní startovací sadu od společnosti Elegoo. Sada je vhodná nejen pro začátečníky, kteří se snaží poznat svět elektroniky a elektronických součástek a také pro děti. V sadě se nachází spousta potřebných komponentů a dílů pro stavbu jak jednoduchých, tak i složitějších projektů.

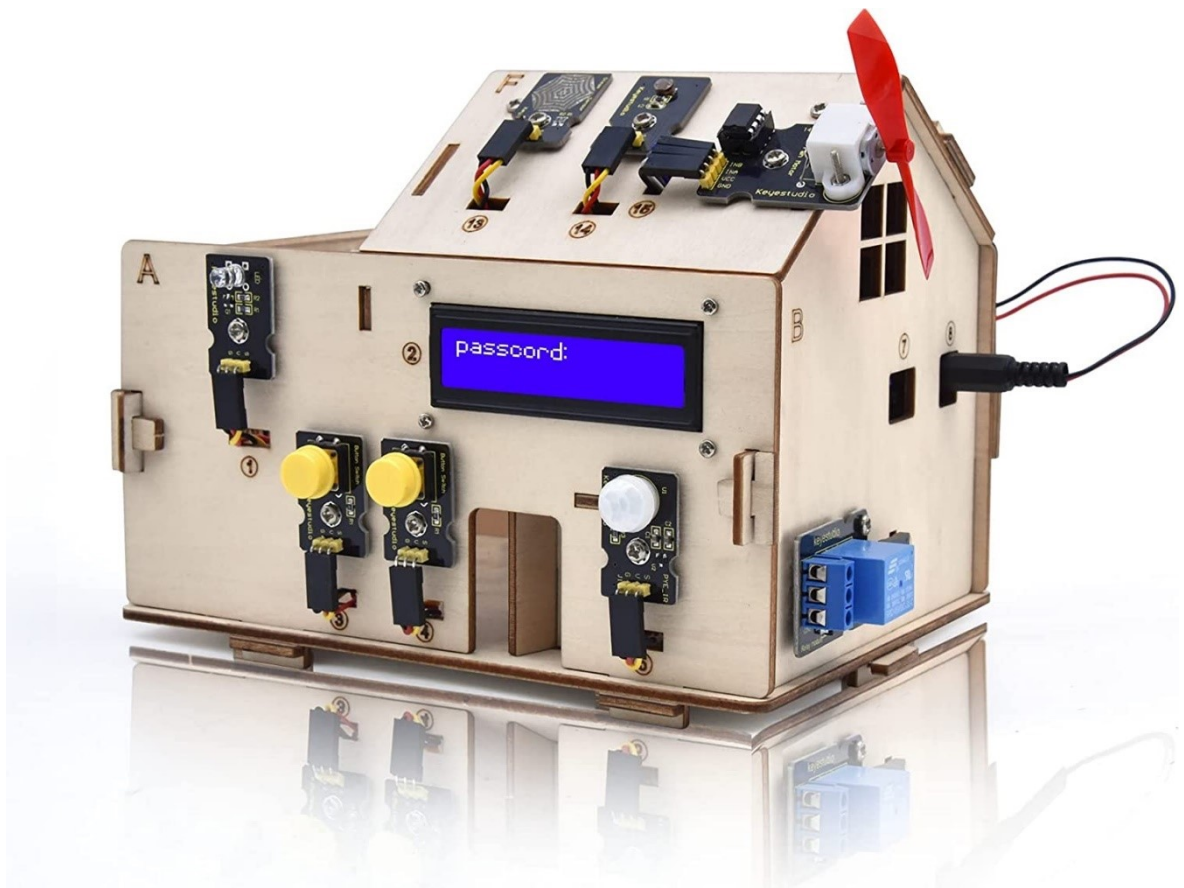
Základ tvoří deska UNO R3. Modul s čipem 74HC595. Modul displeje LCD 1602, který je kompaktní modul LCD1602 s bílým a modrým LED podsvícením má displej s 2 x 16 znaků pro zobrazení textu a symbolů. [22]



Obrázek 22 UNO R3 Ultimate Starter Kit [22]

### 3.2 Stavebnice chytrého domu

Stavebnice chytrého domu je dřevěná stavebnice, která je navržena pro kreativní a hravou výuku za použití platformy Arduino. Stavebnice je navržena tak, aby hravou formou zaujala pozornost a její použití byli velmi realistické. Takže při správném naprogramování všech senzorů a modulů se dům stane inteligentním domem, tak jak je známe nynější moderní domy. Sada obsahuje senzory jako je fotosenzor, infračervený senzor, senzor vlhkosti, senzory na detekci plynu, dále obsahuje tlačítka, relé modul pro spínání výkonových prvků a Bluetooth modul pro bezdrátovou komunikace. Celé je to řízeno Arduinem Uno R3. [23]



Obrázek 23 Stavebnice chytrého domu [23]

### 3.3 Totem

Společnost Totem vytváří konstrukční systém přídavnými díly pro robotiku a prototypování elektroniky. Systém je navržen pro tvůrce všech úrovní. Společnost se tímto snaží, aby svět inženýrství a robotiky byl zábavný a srozumitelný pro každého. Společnost Totem vyvíjí i své vlastní klony Arduino a také vyvíjí své vlastní komponenty a platformy. [24]

#### 3.3.1 Totem Mini Lab

Jedná se o malou, ale výkonnou platformu, která může sloužit jako domácí nebo jako třídní laboratoř. Sada obsahuje různé měřicí přístroje, připravené k použití a různé bloky pro experimentování. Platforma Totem Mini Lab je plně kompatibilní s programovacím prostředím arduino. Hlavním cílem Mini Lab je poskytnout platformu, která by studentům a komunitám mladých tvůrců poskytla snadný a přístupný způsob, jak na ní stavět, a využít ji jako odrazový můstek do světa elektroniky a programování. [25]

Jako hlavní řídicí deska je zde použito TotemDuino.

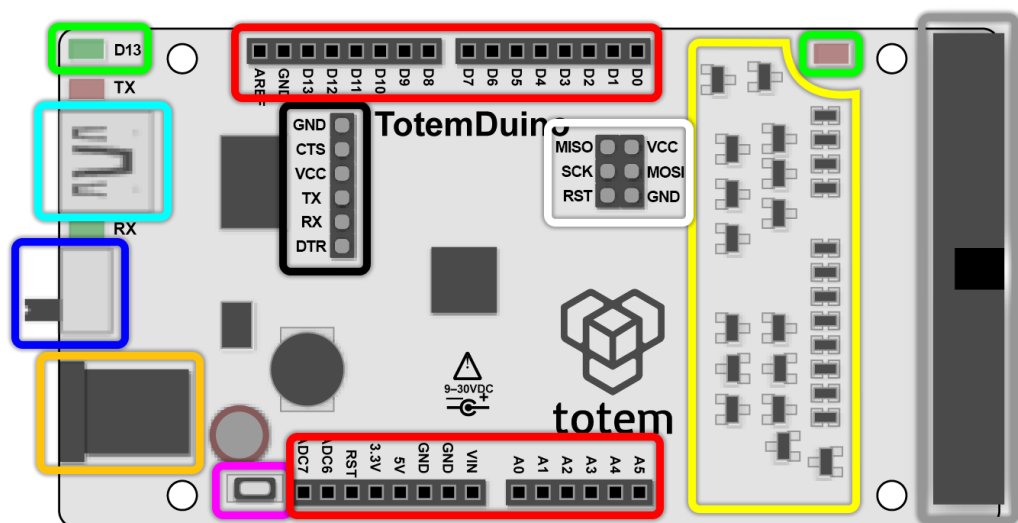
TotemDuino rozšiřuje potenciál platformy Arduino UNO. I když je plně zpětně kompatibilní s Arduinem, obsahuje také mnoho dalších funkcí, jako například:

- Ochrana výstupu – dodává se se všemi výstupními kolíky, které jdou do LabBoardu, chráněny proti přepětí nebo zkratu.
- Rozšiřující port – k LabBoardu se připojuje 34kolíkový plochý kabel pro snadný přístup ke kolíkům.
- Výkonný 5V regulátor – bude méně pravděpodobné, že vám dojde energie při experimentování s vyššími výkonovými zátěžemi, jako jsou motory.
- Volitelné napětí logiky mikrokontroléru - jak svět postupuje od 5 V k logickému napětí 3,3 V, TotemDuino dokáže pracovat s oběma pouhým přepnutím přepínače, aniž by bylo potřeba dalších adaptérů nebo převodníků.

Funkce:

- Mikrokontrolér ATmega328P
- 100% kompatibilní s Arduino Uno, funguje se štíty Arduino
- Pracuje v Arduino IDE
- Rozsah napájení 9-30 V DC nebo 5 V USB

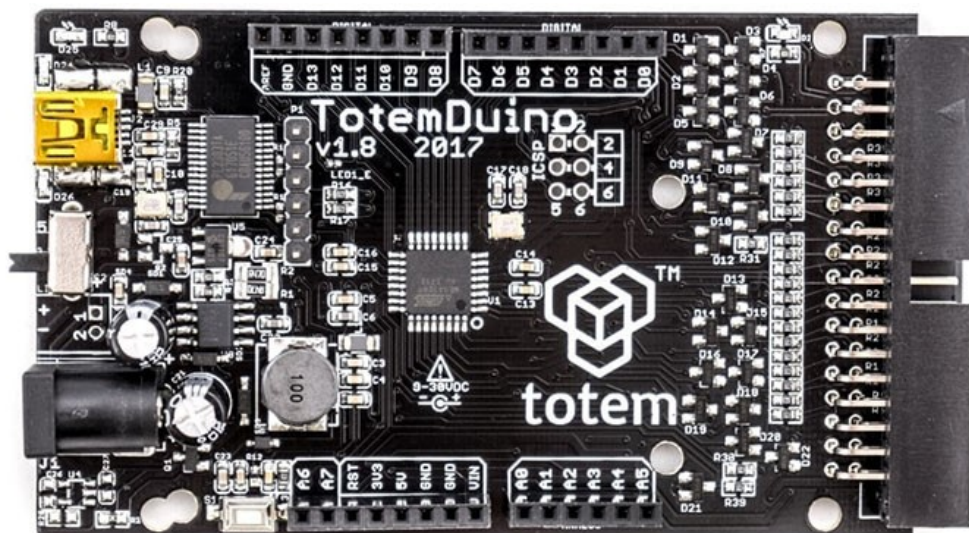
- Při napájení ze stejnosměrného napájecího adaptéru může dodávat proud až 1,5A při 5V.
- Volitelné logické napětí 5V nebo 3,3V
- Konstrukce odolná proti hluku
- Dodatečné filtrování pro přesné analogové měření napětí
- Integrovaný programátor – ke spuštění kódování nejsou potřeba žádné další díly.
- Montážní otvory kompatibilní s Arduino a Totem
- 34kolíkový konektor plochého kabelu pro integraci s Totem Mini Lab [25]



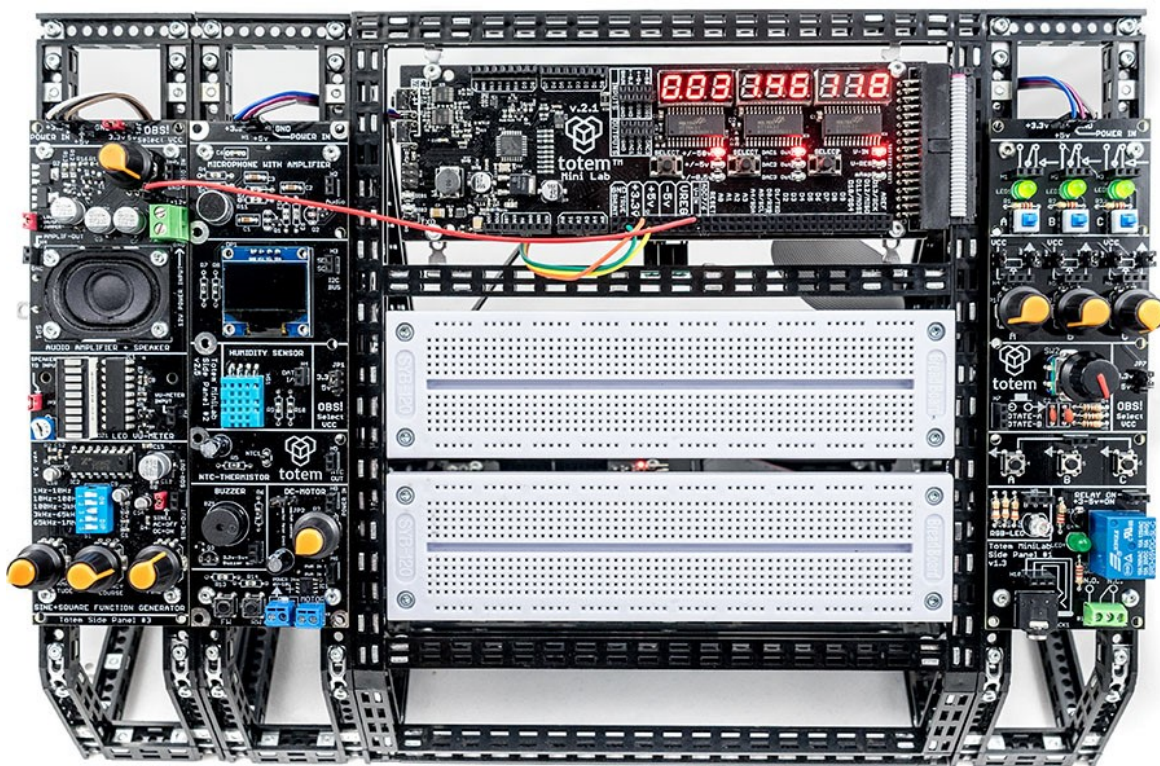
Obrázek 24 Rozložení pinů TotemDuino [25]

- **Arduino piny** – programovatelné piny procesoru ATmega328P. D – digitální, A – analogové, ostatní – napájení.
- **LED** – LED na pravé straně indikuje, zda je TotemDuino zapnuté.  
LED D13 zobrazuje stav pinu D13. Lze použít pro klasické blikání diody.
- **Přepínač napětí** – Přepínač pro výběr mezi logickou úrovní 3,3 V (dolů) nebo 5 V (nahoru). Procesor bude fungovat na zvolené napětí. Standardní desky Arduino (ATmega328P) vždy pracují na logické úrovni 5 V. To je výhodné, pokud potřebujete připojit senzory, které jsou pouze 3,3V (při použití s 5 V by se mohly poškodit).
- **Tlačítko Reset** – reset procesoru. Restartuje běžící skicu. Stejně jako připojení RST k GND
- **Ochranný obvod** – ochranný obvod konektoru pro možné přepětí nebo zkratu.

- **USB vstup** – umožňuje připojit TotemDuino k PC. Používá se pro napájení (pokud není připojen DC), programování (nahrání programu), tisk zpráv (Serial Monitor).
- **DC vstup** – poskytuje napájení pro TotemDuino (a Mini Lab). Může akceptovat napětí mezi 9-30VDC. Přímě směřováno na pin VIN. Slouží k vytvoření regulovaného 3,3V a 5 V.
- **Plochý port na kabel** – port pro připojení TotemDuino s LabBoardem. Směřuje napájení a všechny konektory Arduino na LabBoard.
- **Seriový výstup** – piny výstupu převodníku USB – standardně se nepoužívá.
- **ICSP piny** – standardní piny pro sériové programování v obvodech. [25]



Obrázek 25 TotemDuino [25]



Obrázek 26 Totem Mini Lab [25]

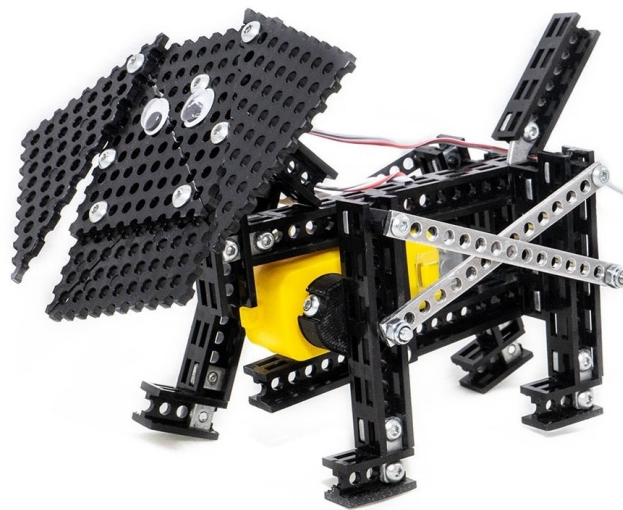
### 3.3.2 Sada pro rozvoj dovedností STEM

Společnost Totem nevyvíjí pouze platformu Totem Mini Lab, ale se i výrobou stavebnic pro rozvoj dovedností STEM (přírodní vědy, technika, technologie a matematika). Na oficiálních stránkách lze nalézt 10 takovýchto sad, které jsou svou složitostí, jak na ovládaní, tak i na sestavení přizpůsobeny pro různé zdatné žáky. Můžeme zde nalézt jak jednoduché modely zaměřené na základy mechaniky, tak i složité, komplexní roboty. Výhodou těchto modelů je dobře zpracovaná dokumentace pro samotné sestavení modelů.

### 3.3.2.1 Sada pro mladého inženýra: Pes

Tato sada je vhodná pro učení se základům inženýrství. Sestavením hračky se dítě může naučit mnoha dovednostem jako jsou například:

- Základy mechaniky v robotice.
- Jednoduchá inženýrská řešení.
- Použití různých mechanických dílů a nástrojů.
- Použití stejných mechanických dílů v různých modelech/situacích.
- Vlastnosti různých materiálů. [26]



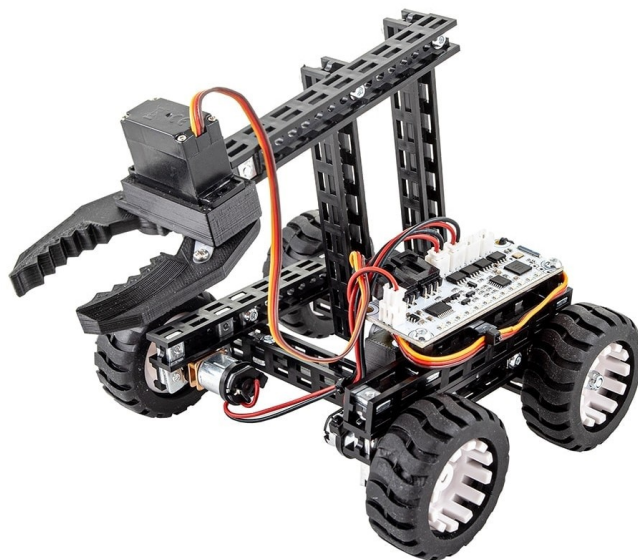
Obrázek 27 Sada pro mladého inženýra: Pes [26]

### 3.3.2.2 *GripperBbot*

Gripper Bot je vyroben z pevných a odolných dílů, které se používají v robotice vyšší úrovně. Naučte se používat nástroje robotiky a objevujte skutečná technická řešení.

Tento malý, ale výkonný konstruktér od Totemu má být zábavou a snadno se naučit mechaniku a základy robotiky.

- Toto malé auto s chapadlem se ovládá aplikací.
- Chapadlo je mini verze robotické paže, která se snadno ovládá a uchopuje věci.
- Spousta zábavy při hraní o samotě nebo soutěžení s přáteli.
- Gripper Bot je ovládán Totem Mini Control Board, což rozšiřuje vaše možnosti sestavení a hraní. [27]



Obrázek 28 Gripper Bot [27]



### 3.3.2.3 Sada podvozku auta řízená přes telefon pomocí Bluetooth

Sestavením tohoto podvozku získáte čtyřkolové vozidlo použitelné jako základ k novému projektu. Sada podvozku nabízí:

- Vysoce kvalitní materiály.
- Použijte svůj smartphone nebo tablet k ovládání vlastního vyrobeného podvozku.
- Možnost přidání dalších prvků [28]



Obrázek 29 Sada podvozku auta řízená přes telefon pomocí Bluetooth [28]

## **II. PRAKTICKÁ ČÁST**

## 4 VÝVOJ PLATFORMY

Vývojem této platformy je vytvořit návrh možného řešení, které lze využít při výuce školách. Platforma je sestavena z běžně dostupných komponentů, tak aby byla snadná její opakovatelnost a jednoduché použití na školách.

### 4.1 Použité komponenty na platformě

Komponenty jsem zvolil s ohledem na možné použití v mnoha různých projektech.

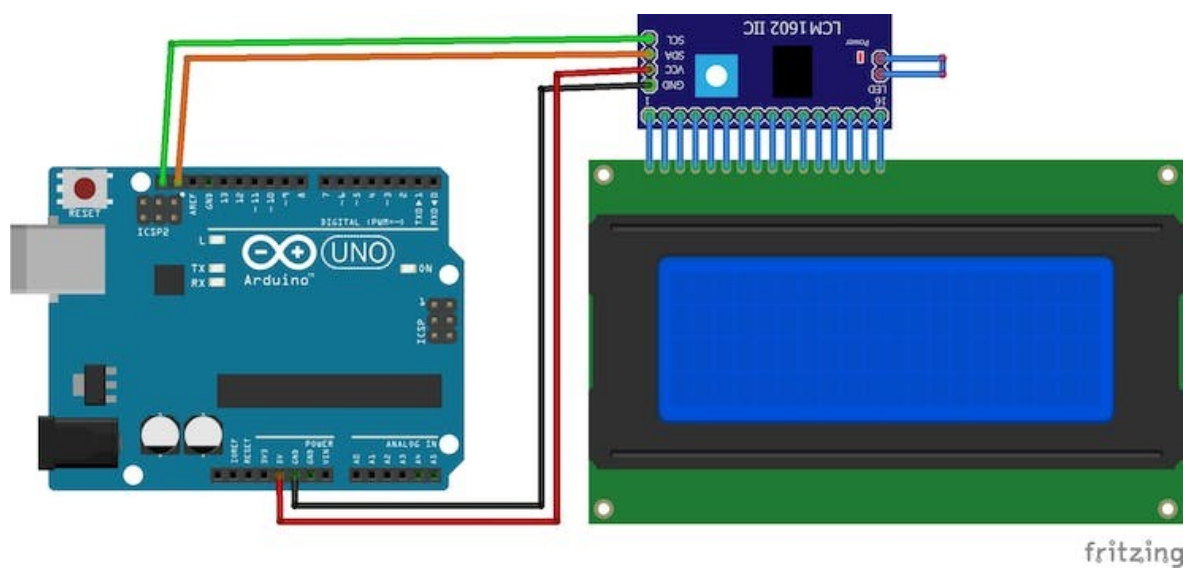
#### 4.1.1 LCD displej 4x16

LCD (Liquid-crystal display) u Arduina slouží jako zobrazovací jednotka. Jeho napájení je 5 V a díky I2C sběrnici má pouze čtyři piny, 2 piny jsou napájecí a 2 komunikační na spojení s deskou Arduino.

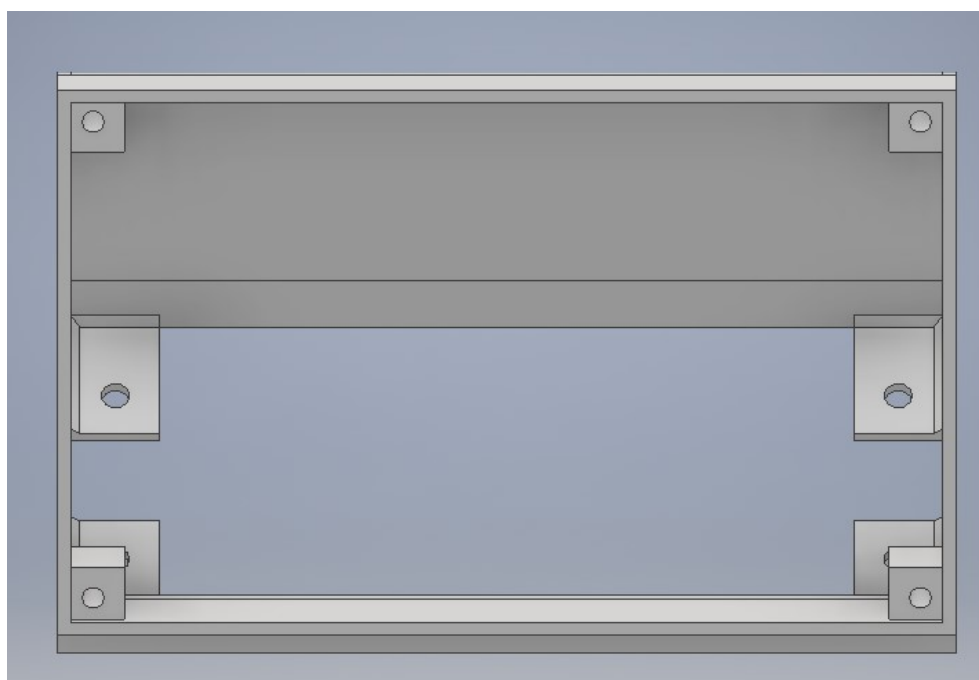


Obrázek 30 LCD display [29]

Základní zapojení displeje pomocí I2c sběrnice. Je právě díky použité sběrnici velmi jednoduché. 2 piny se použijí na samotné napájení displeje a další 2 na přenos dat, které se mají zobrazit na displeji.



Obrázek 31 Zapojení displeje



Obrázek 32 Model držáku displeje



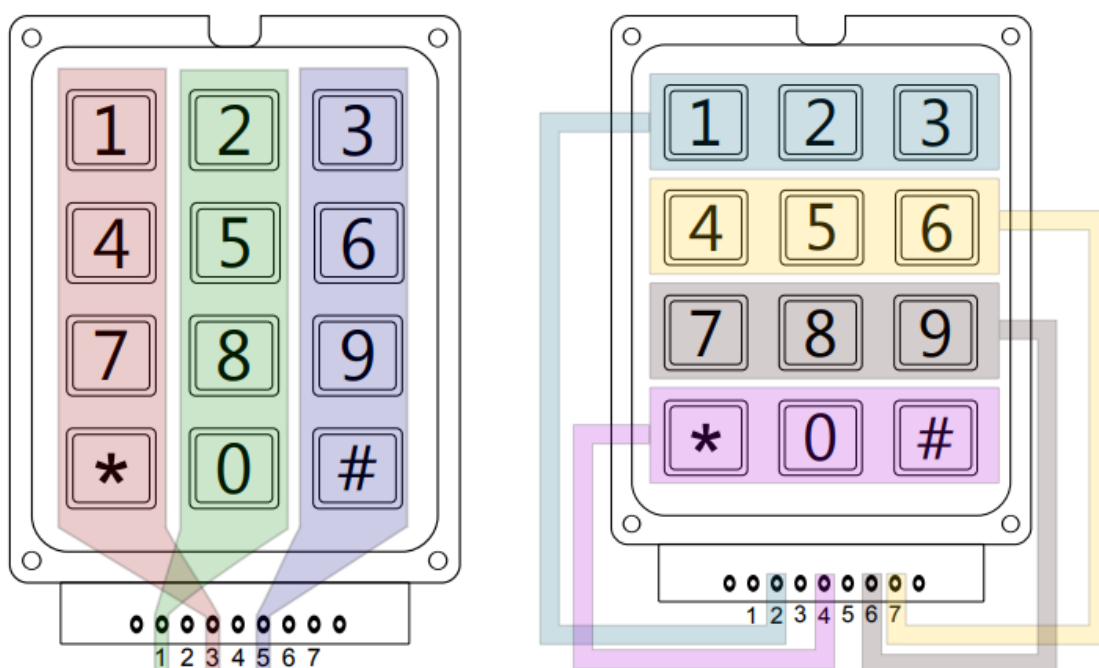
Obrázek 33 Držák osazená LCD displejem

#### 4.1.2 Klávesnice

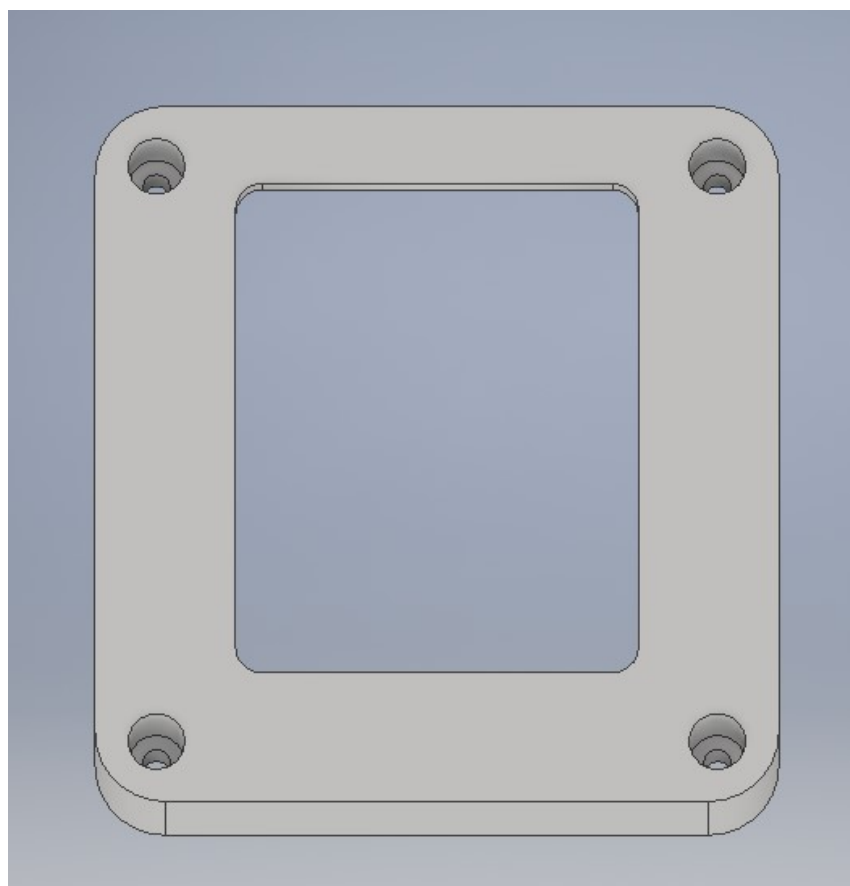
Klávesnice je použita maticová s 12 klávesami, které mají standartní zapojení pomocí 7 pinů. 3 na každý sloupec a 4 na jednotlivé řádky.



Obrázek 34 Klávesnice



Obrázek 35 Vnitřní zapojení klávesnice [30]



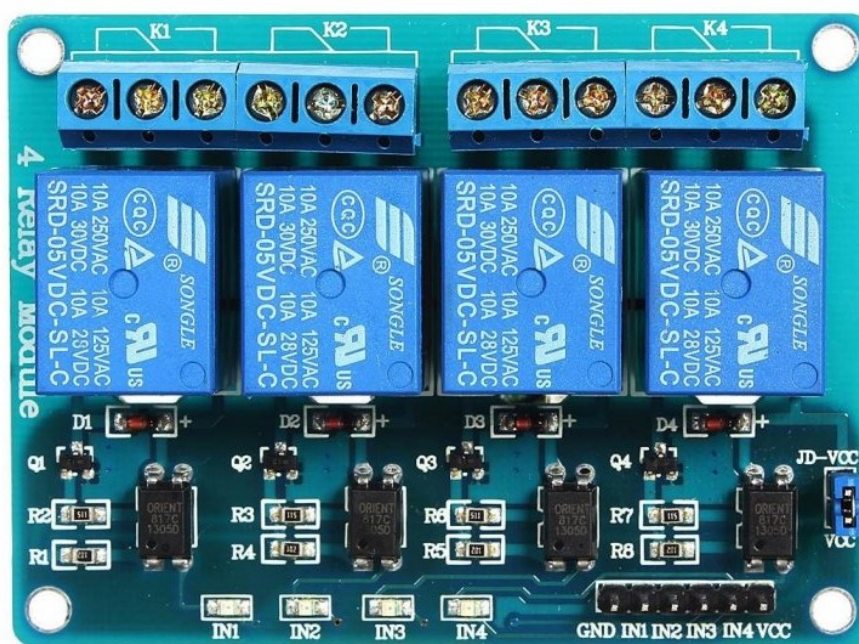
Obrázek 36 Model držáku klávesnice



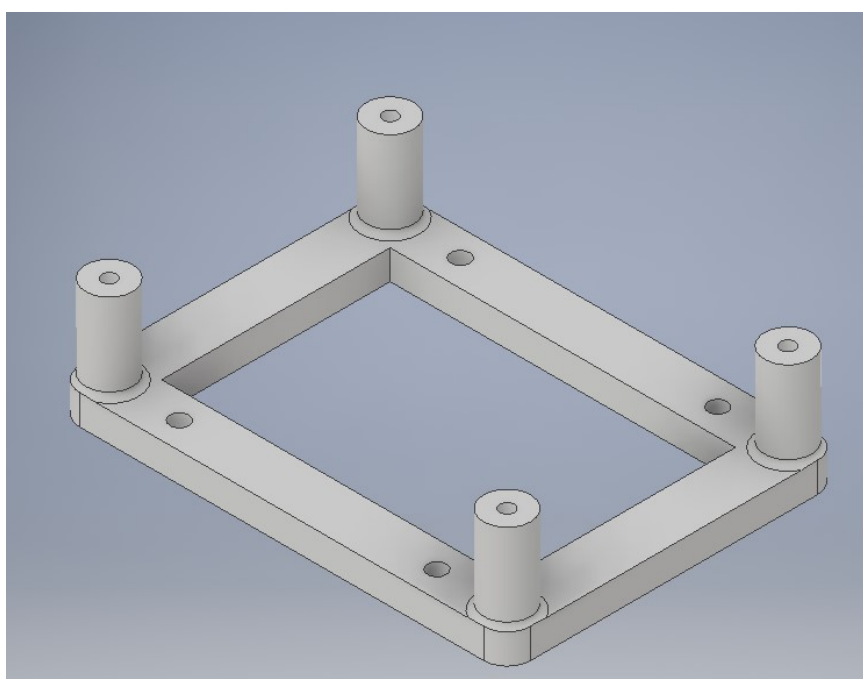
Obrázek 37 Klávesnice uložená v držáku

### 4.1.3 Relé modul

Relé modul obsahuje 4x relé napájený 5V. To je deska, pomocí které jste schopni ovládat různé spotřebiče a další zařízení s velkými proudy. Je možné ovládat přímo mikrokontrolerem nebo čidlem s 5V výstupem. 5V 4-kanálové rozhraní, potřebuje 15-20mA. Vybaven silnoproudým relé, AC250V 10A/DC30V 10A. Standardní rozhraní, které lze ovládat přímo mikrokontrolerem. Zapojení je velmi jednoduché a díky integrovaným stavovým diodám lze snadno ověřit v jakém stavu se jednotlivá relé nachází, zda jsou sepnutá či nikoliv.

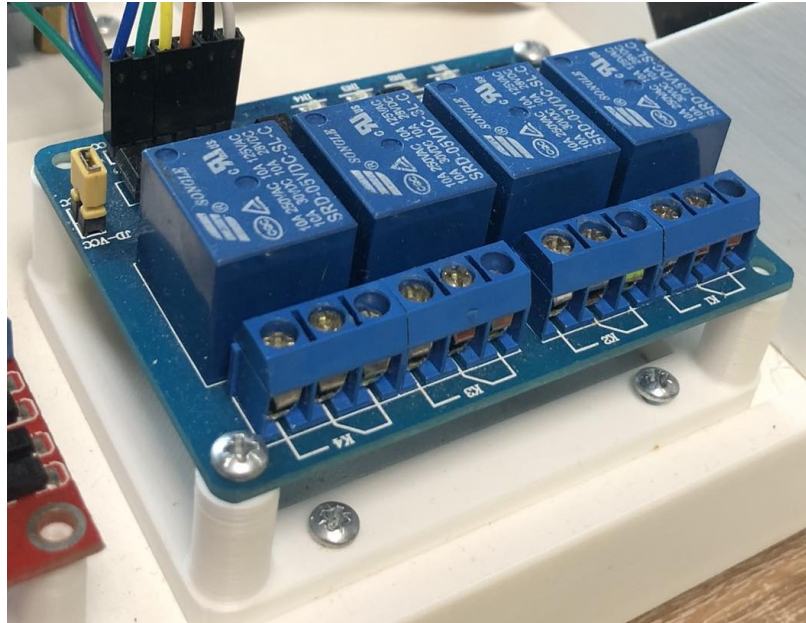


Obrázek 38 Modul 4kanalového relé



Obrázek 39 Model držáku modulu 4kanalového relé

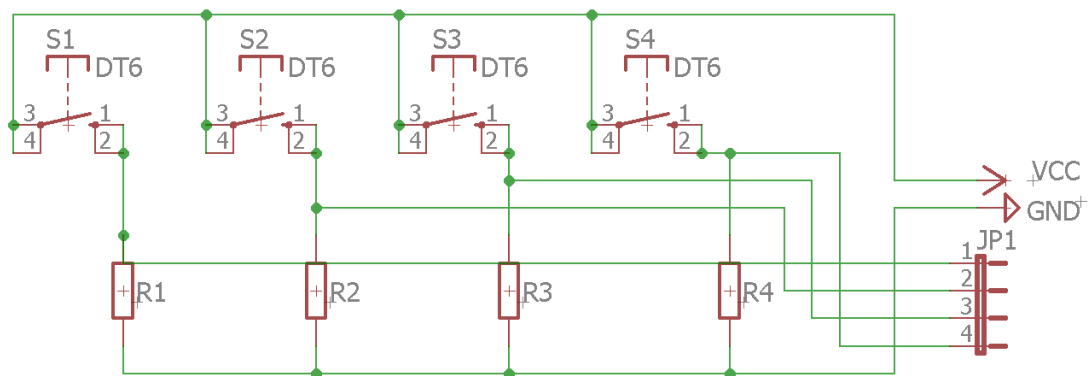




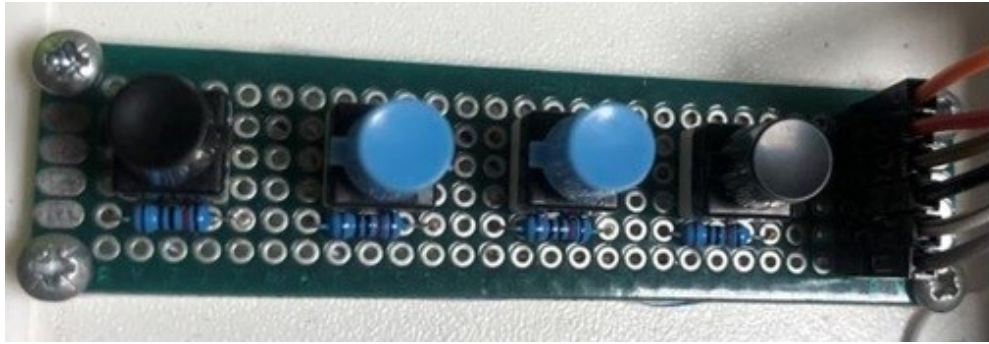
Obrázek 40 Relé modul připevněn na platformě pomocí držáku

#### 4.1.4 Modul s tlačítky

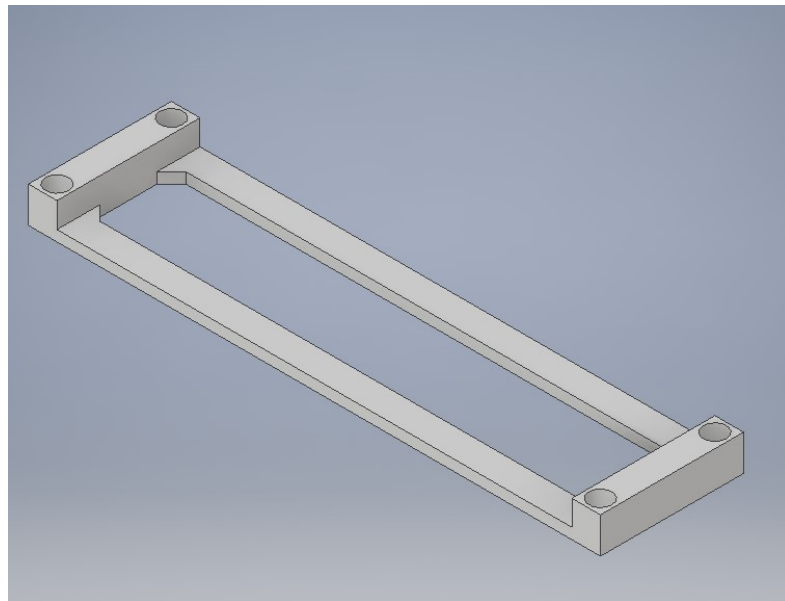
Tento modul je vytvořen za 4 tlačítek, které jsou na připájeny na universální desce plošných spojů.



Obrázek 41 Schéma zapojení modulu s tlačítky



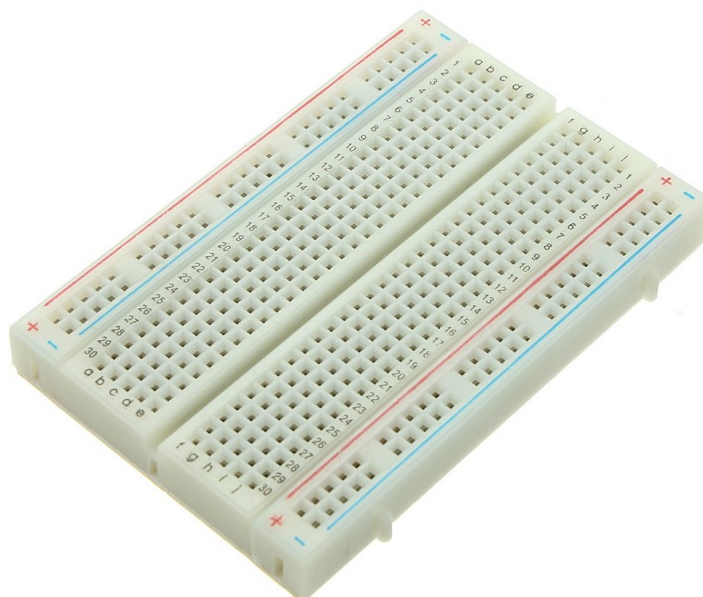
Obrázek 42 Modul s tlačítky



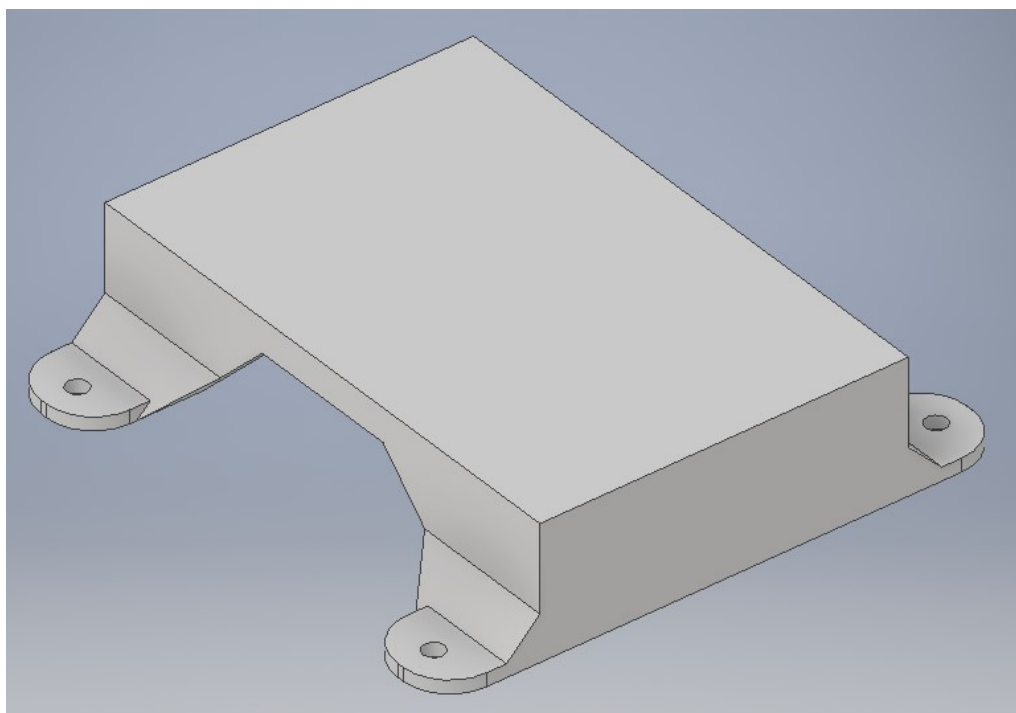
Obrázek 43 Model držáku pro modul s tlačítky

#### 4.1.5 Nepájivé pole

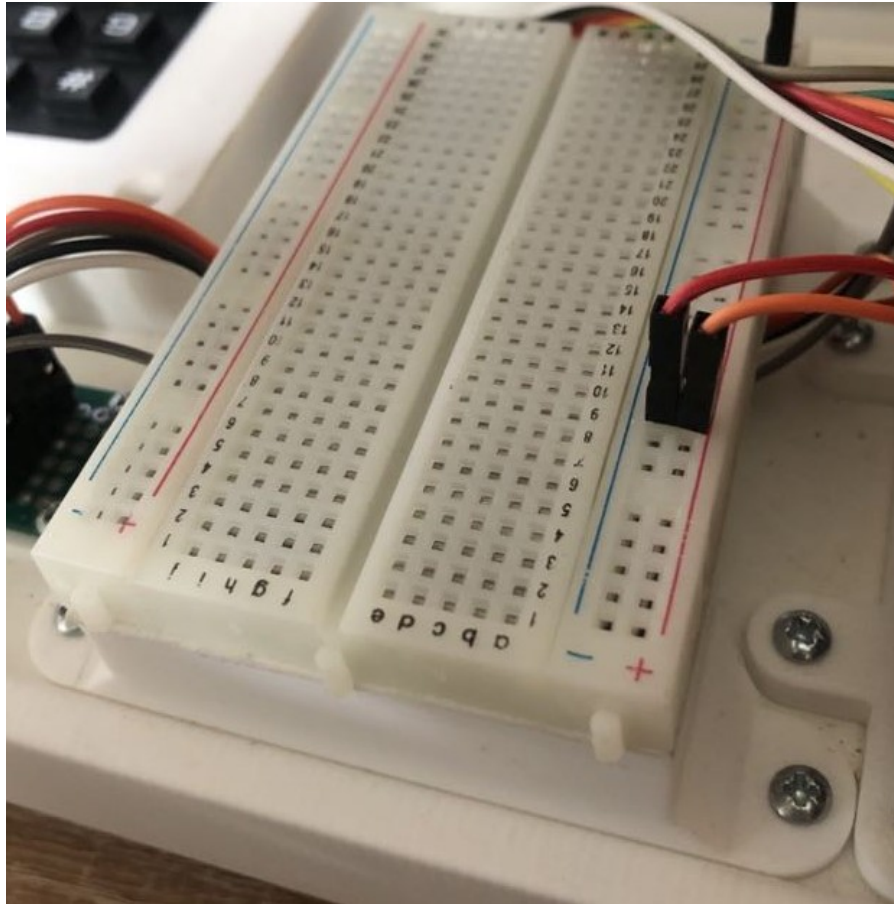
Nepájivé pole jsem umístěno na platformu, jelikož je nepájivé pole je opakovaně použitelná pomůcka k navrhování prototypů elektrických obvodů a pro experimentování s obvodů bez nutnosti pájení součástek.



Obrázek 44 Nepájivé pole



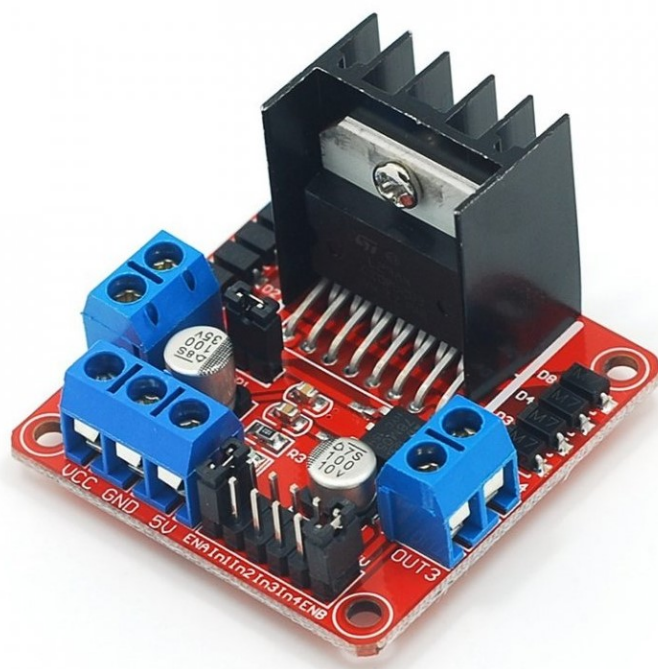
Obrázek 45 Model držáku nepájivého pole



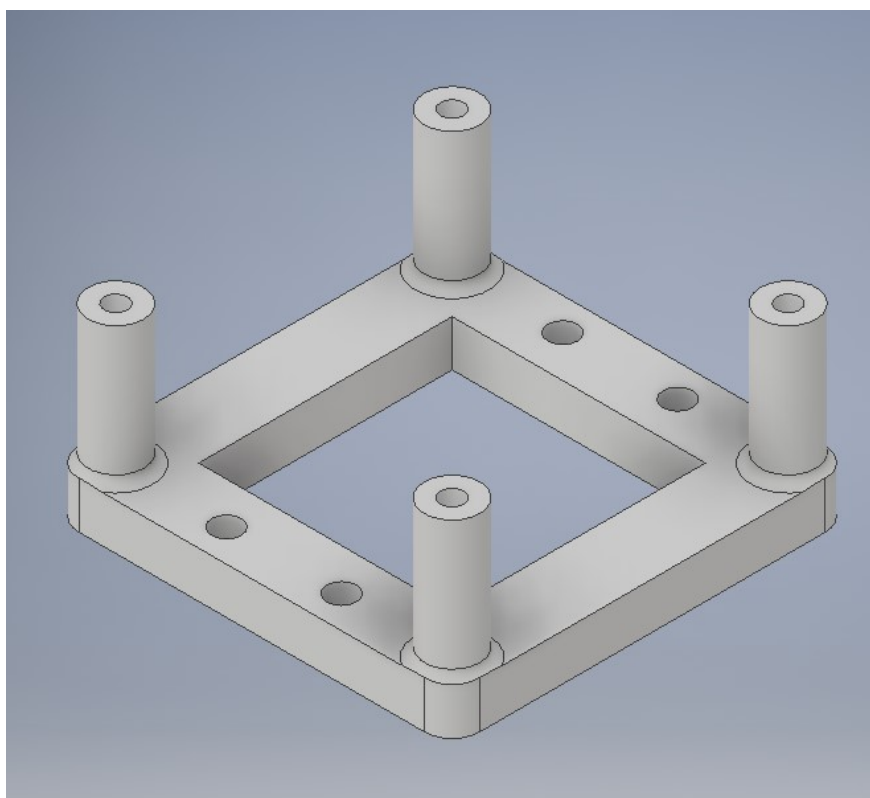
Obrázek 46 Nepájivé pole umístěné na platformě pomocí držáku

#### 4.1.6 H můstek

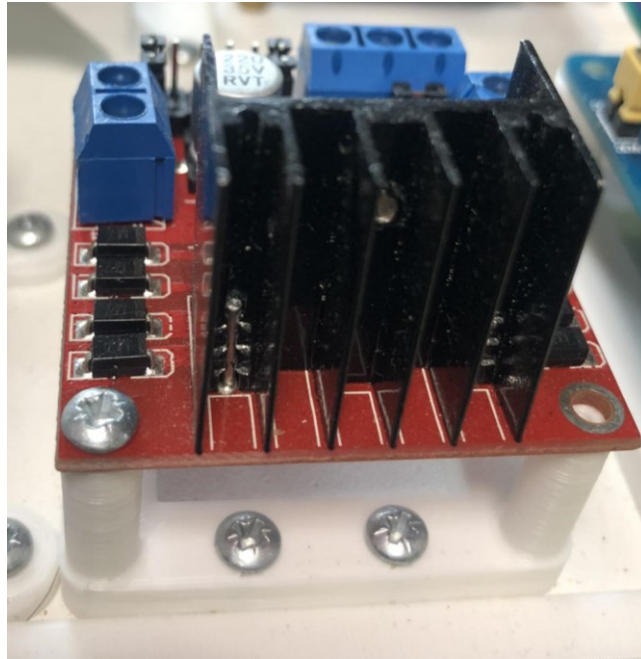
H můstek je velmi běžný modul pro řízení stejnosměrných motorů, proto je tento modul zařazen do platformy. Má široké použití, a to zejména ve stavbě například robotů, kde je řízen pohyb. Tento modul používá ST L298N čip, který má dobrou schopnost řízení a nízkou teplotu při vyšším zatížení. Napájení až do 46V. Proud může dosáhnout maximálně 3A, dlouhodoběji je možno používat při proudovém zatížení 2A. Obvod je schopen řídit jeden 2-fázový krokový motor, jeden 4-fázový krokový motor nebo dva motory na stejnosměrné napětí. Modul má vestavěný 5v stabilizátor, doporučené napájení můstku je 7-35V.



Obrázek 47 H můstek



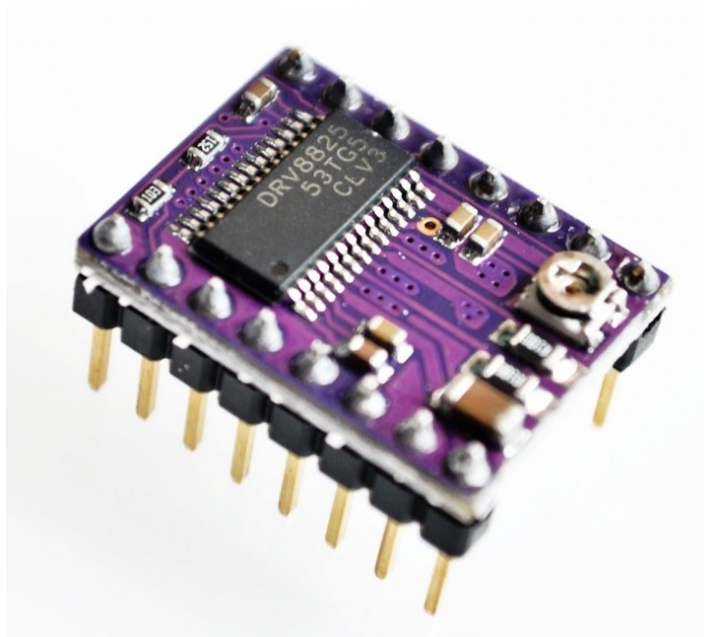
Obrázek 48 Modle držáku H můstku



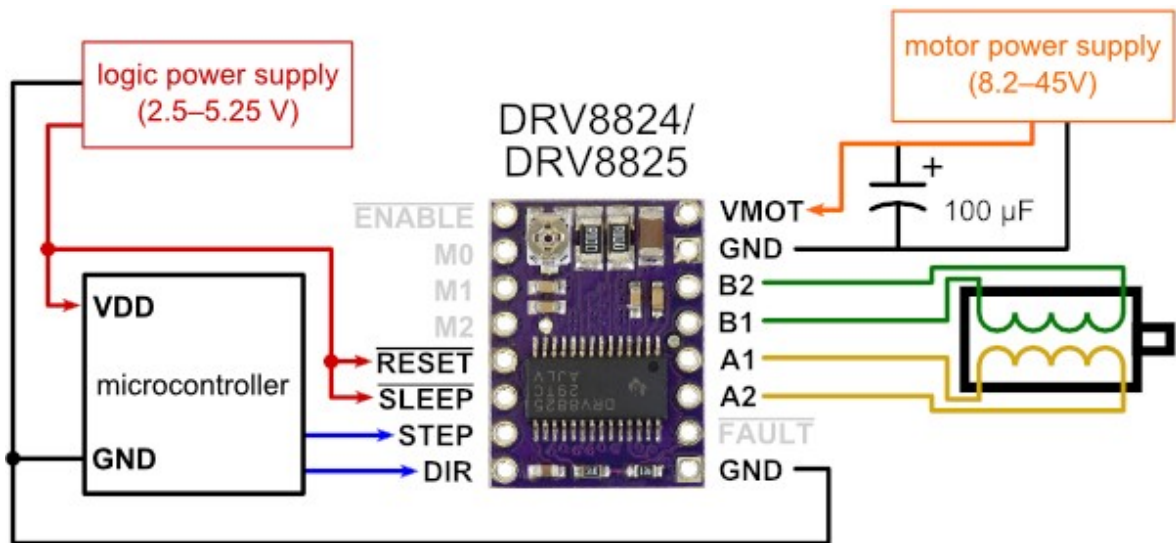
Obrázek 49 H můstek umístěn na platformě

#### 4.1.7 Driver krokových motorů

Driver je komponenta, která má za úkol ovládat krokový motor a to tím, že v přesně generuje pulzy pro jednotlivých vinutí krokového motoru a tím ho uvádí do pohybu. Pulzy se na základě informací z ovládací jednotky například z některé verze Arduino, mění a tím je motor ovládán.



Obrázek 50 Driver DRV8825 pro krokové motory [31]



Obrázek 51 Zapojení driveru pro krokové motory [31]

#### 4.1.8 Motor na stejnosměrné napětí

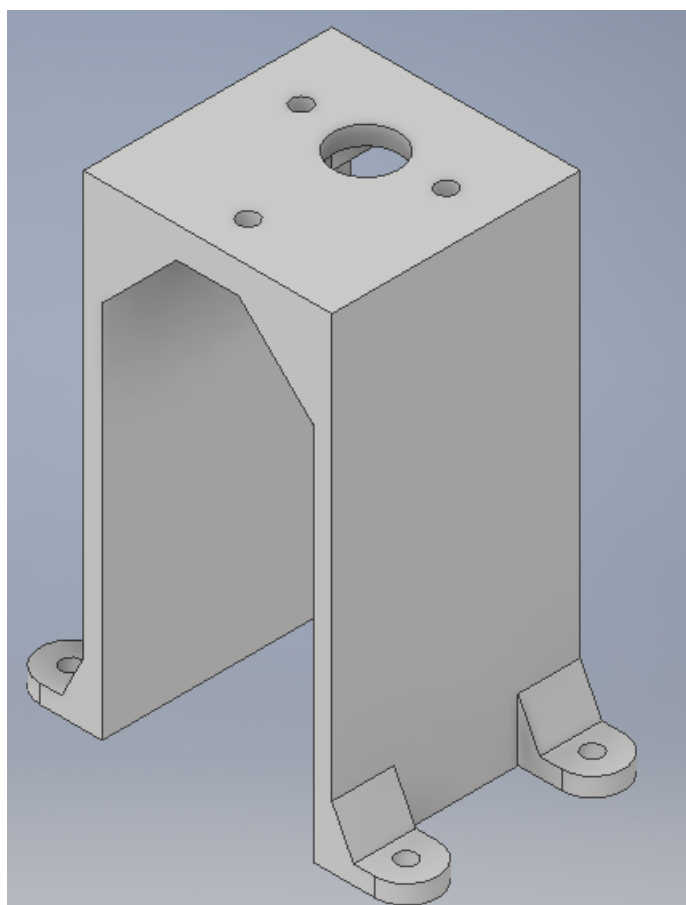
Stejnosměrný motor je jeden z mnoha aktivních součástí, které lze použít na jednotlivé aplikace a ve spojení s H – můstkem můžeme motor efektivně ovládat a využít ho tak pohyb nebo posun na různých projektech. Motor je zařazen do platformy, právě proto aby žáci měli možnost si funkci a ovládání motoru vyzkoušet a mohli tak aplikovat na vlastních projektech.

Pro platformu jsem použil motor s převodovkou, který právě díky převodovce má vyšší točivý moment, ale nižší otáčky než samotný motor.

Zapojení motoru je velmi jednoduché a stačí přivést dané napětí na kontakty motoru.

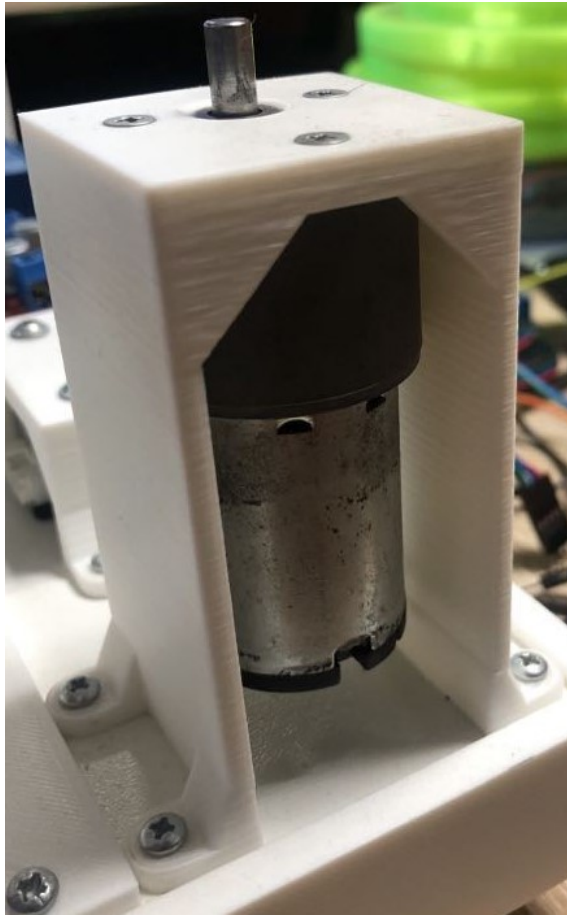


Obrázek 52 Motor s převodovkou na stejnosměrné napětí



Obrázek 53 Model držáku stejnosměrného motoru

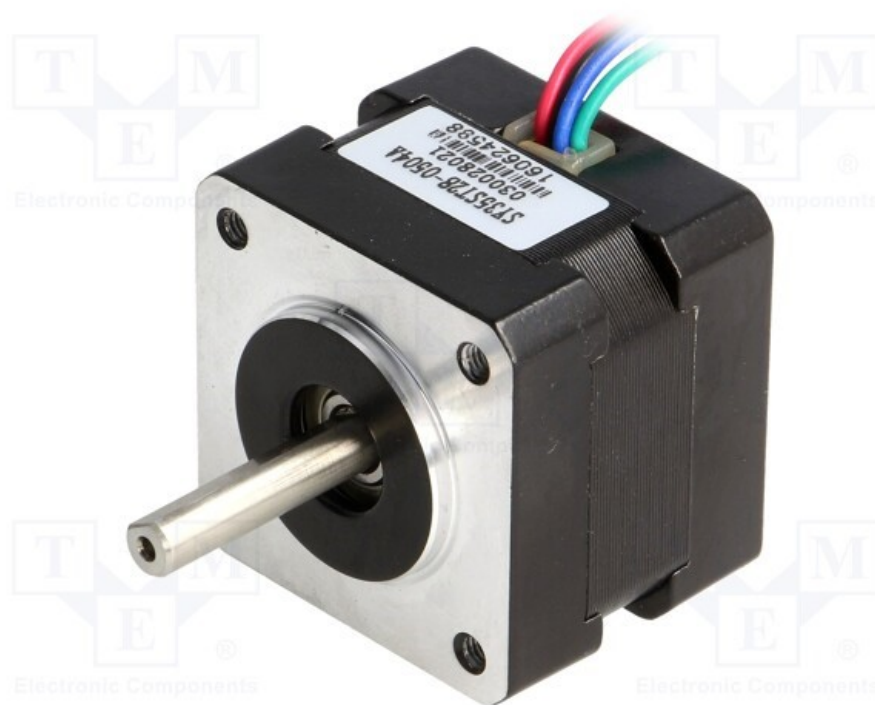




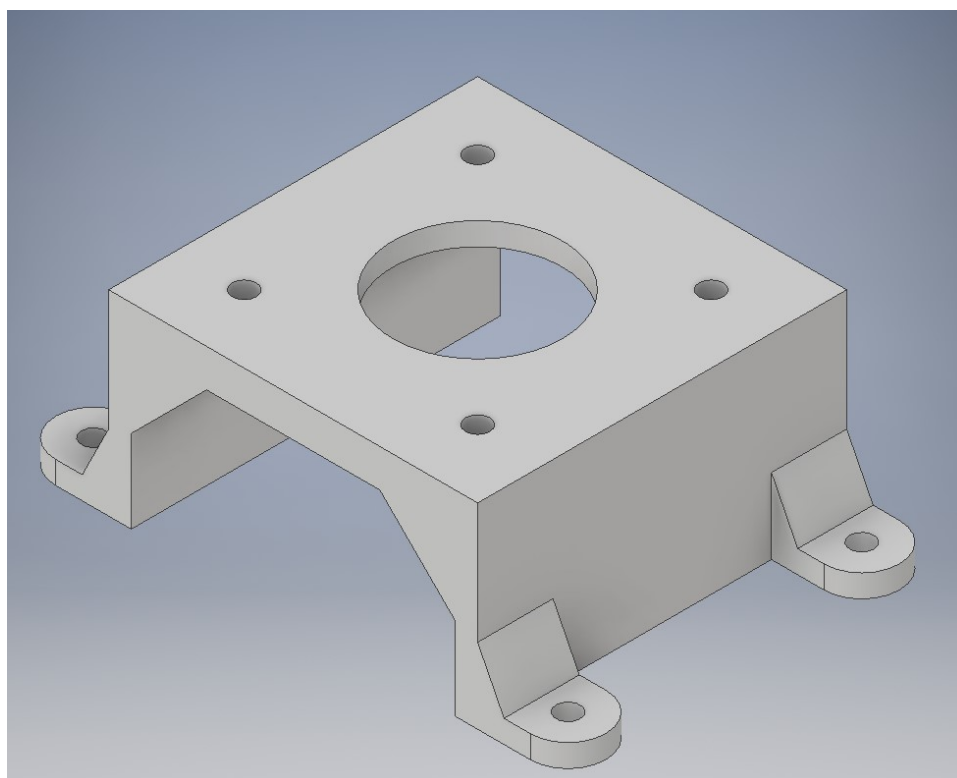
Obrázek 54 Umístění motoru na platformě pomocí držáku

#### 4.1.9 Krokový motor

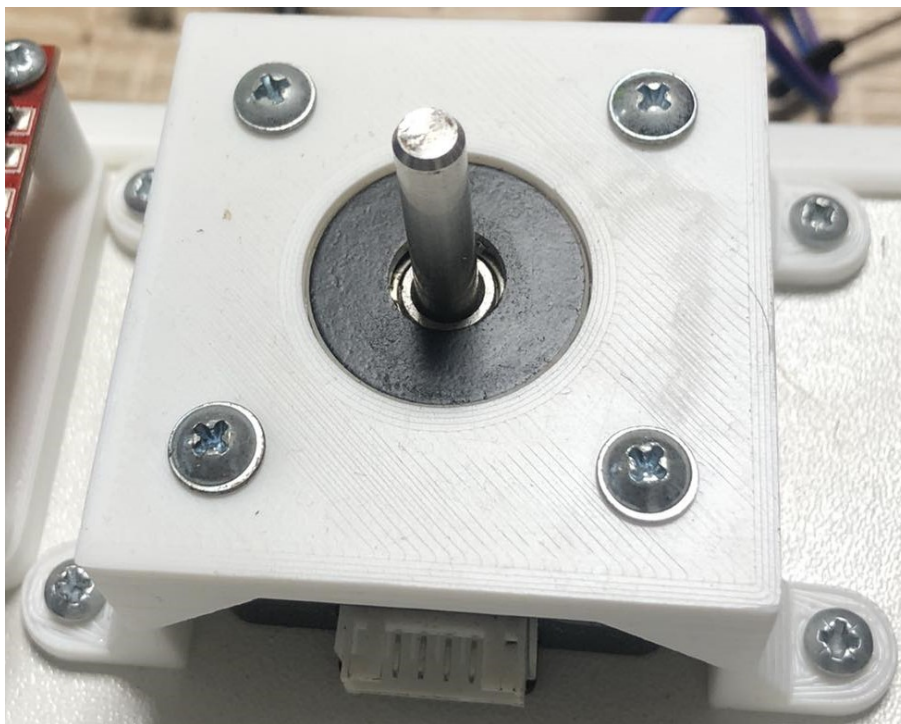
Krokový motor je synchronní zařízení, většinou napájený impulsy stejnosměrného proudu. Pro jeho ovládání je zapotřebí použít driver pro krokové motory, který dává přesné impulsy krokovému motoru. Krokový motor je součástka, která se ve velké míře používá v průmyslu a v automatizovaných provozech. Dnes se s krokovými motory často setkáváme u stále více populárních 3D tiskárnách, proto je tento typ motoru zařazen do platformy, tak aby žáci měli povědomí o tom, jak funguje jedna z důležitých komponent tiskáren.



Obrázek 55 Krokový motor



Obrázek 56 Model držáku krokového motoru



Obrázek 57 Krokový motor umístěný na platformě pomocí držáku

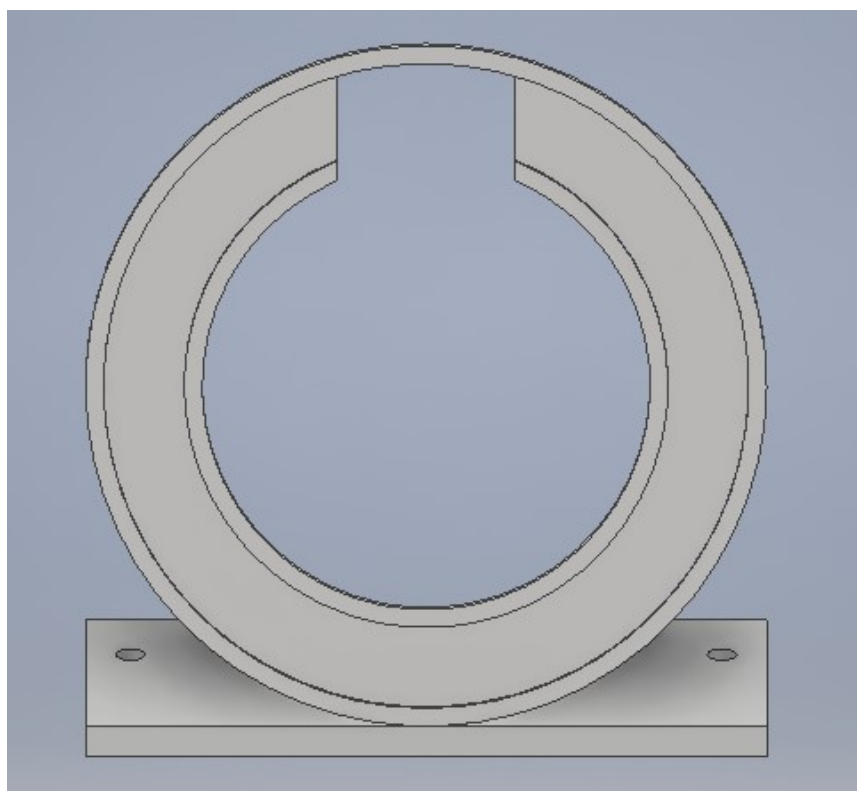
#### 4.1.10 RGB LED kruh

Tento LED Kruh je tvořen LED diodami WS2812B zapojenými do série. Tyto LED diody mají vestavěný čip s pamětí 3 bajty. Těmi se nastavuje intenzita svitu každé ze tří barev. To umožňuje komunikaci pomocí jednokanálového rozhraní. Znamená to tedy, že lze řídit spoustu LED diod pomocí jediného digitálního pinu Arduino.

K naprogramování se používá knihovna Adafruit NeoPixel, která umožní jednoduché ovládání jednotlivých pinů.



Obrázek 58 RGB LED kruh



Obrázek 59 Držák RGB LED kruhu



Obrázek 60 Držák RGB LED kruhu osazený modulem RGB

#### 4.1.11 Ultrazvukový senzor

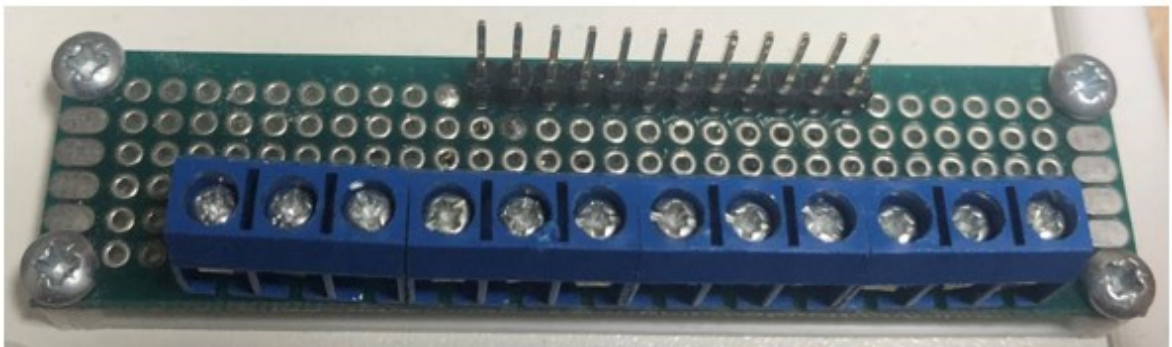
Ultrazvukový senzor se používá pro měření vzdálenosti a je jeden z nejoblíbenějších senzorů při stavbě robotů. Jedná se o snadno použitelný senzor, který má rozsah 2 až 400 cm. Senzor se skládá ze dvou ultrazvukových měničů. Jeden je vysílač, který vydává ultrazvukové zvukové impulsy a druhý je přijímač, který poslouchá odražené vlny. Jeho funkci si můžeme představit jako sonar v ponorkách.



Obrázek 61 Ultrazvukový senzor

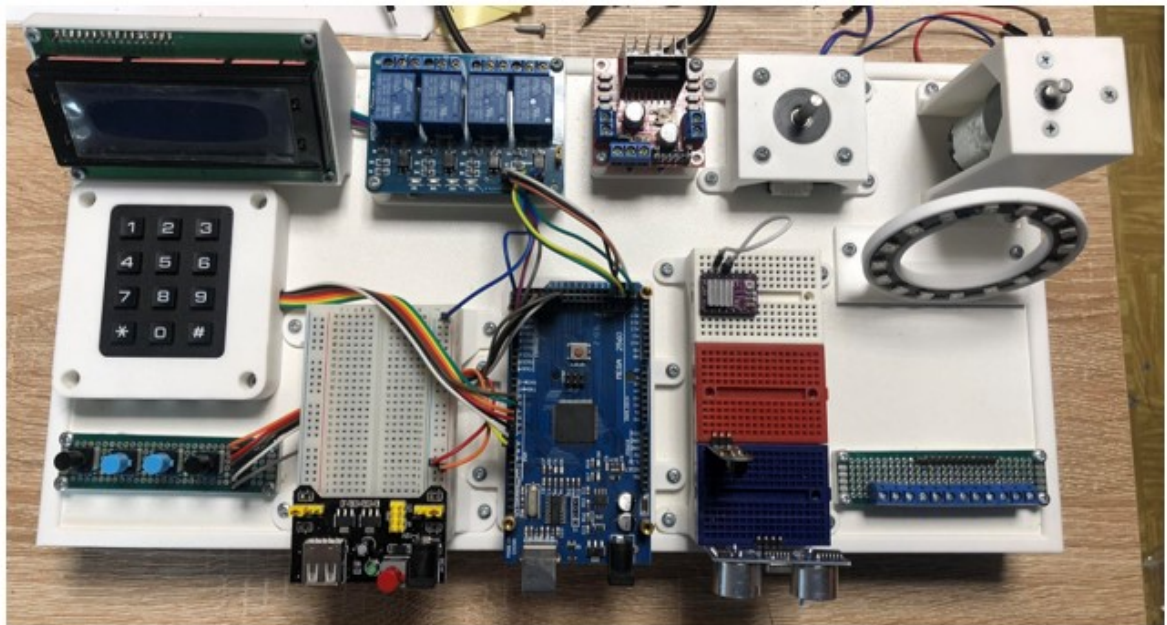
#### 4.1.12 Modul pro připojení periférií bez pinů

Tento modul slouží pro připojení periférií, které nemají standartně připíjené piny pro přímé připojení k desce Arduino, nebo pro použití na nepájivém poli. Jedná se například o motory, senzory s vyšší odolností nebo zařízení, které není primárně určeno pro použití s Arduinem. Jedná se o propojení standartních přípojných pinů a šroubové svorkovnice, tak aby zaručeno jednoduché a efektivní připojení různých periférií.



Obrázek 62 Modul pro připojení periférií bez pinů

#### 4.1.13 Kompletní platforma



Obrázek 63 Kompletní platforma

## 4.2 Konstrukce

Konstrukce platformy byla zvolena z dostupných materiálů, které jsou běžně k dostání a není nijak zásadní problém se k těmto dílům a použitým materiálům dostat.

## 4.3 Podklad platformy

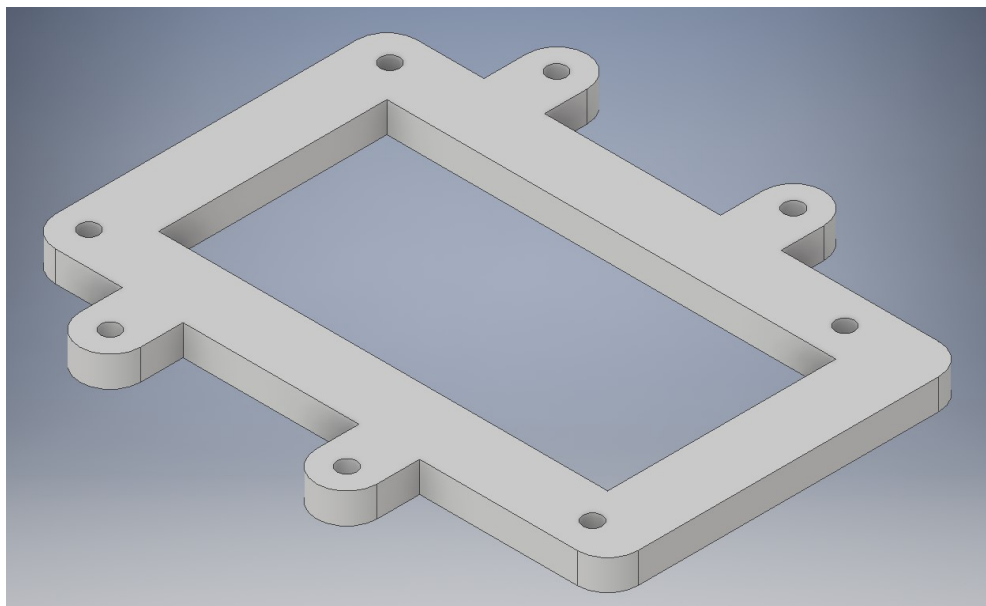
Proto aby platforma byla stabilní a mohly se na ni pevně uložit komponenty zvolil jsem desku z lamina.



Obrázek 64 Deska lamino

## 4.4 Tištěné držáky na komponenty

Proto aby byly periferie pevně uloženy na podkladu bylo potřeba vytvořit několik držáků. Pro tvorbu těchto držáků jsem zvolil 3D tisk, který byl pro tyto držáky vyhovující. Jako materiál pro tisk jsem zvolil materiál PLA, který je pro tuto aplikaci dostatečný jak po pevnostní stránce, tak i z hlediska tepelné odolnosti, jelikož se nepředpokládá nadměrné tepelné zatížení. Jednotlivé modely držáků jsou přiloženy k jednotlivým komponentám.



Obrázek 65 Model držáku Arduino Mega



Obrázek 66 Arduino Mega připevněné na platformě pomocí tištěného držáku



## 5 POUŽITÝ SOFTWARE PRO PLATFORMU

Arduino podporuje více softwaru pro naprogramování, protože je možné ji naprogramovat v jazyce C nebo C++. Nejjednodušší je však používat knihovnu Wiring. Ta je v současné době pro programování Arduina velmi rozšířená. Kvůli její komplexnosti se o ní občas mluví jako o samostatném programovacím jazyku.

### 5.1 Arduino IDE 2

Arduino IDE 2 je vylepšením klasického IDE se zvýšeným výkonem, vylepšeným uživatelským rozhraním a mnoha novými funkcemi, jako je automatické dokončování a vestavěný debugger.



Obrázek 67 Arduino IDE 2 [32]

- **Verify / Upload** (Otevřít/Nahrát) -zkompilujte a nahrajte svůj kód na vaši desku Arduino.
- **Select Board & Port** (vybrat desku a port) – zde se automaticky zobrazí detekované desky Arduino spolu s číslem portu.
- **Sketchbook** (Projekty)– zde najdete všechny vaše skici lokálně uložené ve vašem počítači. Kromě toho se můžete synchronizovat s Arduino Cloud a také získat své náčrty z online prostředí.

- **Boards Manager** (Manažer desek) – procházejte Arduino a balíčky třetích stran, které lze nainstalovat.
- **Library manager** (Správce knihoven) – procházejte tisíce knihoven Arduino vytvořených společnostmi Arduino a její komunitou.
- **Debugger** (Ladit) - testování a ladění programů v reálném čase.
- **Search** (Hledat) – hledání klíčových slov ve vašem kódu.
- **Open Serial Monitor** (Sériový monitor) – otevře nástroj Serial Monitor jako novou kartu v konzole. [32]

### 5.1.1 Automatické dokončování

Automatické dokončování je skvělý nástroj, který nejen ušetří čas, ale také přispívá k menší chybovosti při psaní programu a zvyšuje se tak uživatelský komfort. Ať už píšete kód v C++, JavaScriptu, HTML, CSS nebo jiných běžných jazycích, pochopení syntaxe a způsobu psaní funkcí je absolutní nutností. Je také nedílnou součástí rozvoje vašich dovedností a čím více toho o jazyce víte, tím lepším programátorem se stanete. [33]

### 5.1.2 Nástroj pro ladění

Nově představenou, ale méně známou funkcí Arduino IDE 2 je Debugger.

Ladicí program je softwarový nástroj, který se používá k testování a ladění programů. V zásadě prochází programem kontrolovaným způsobem s pomocí hardwarového rozhraní, které může pomoci procházet prováděním programu. To může pomoci lépe porozumět programu a také pomoci odhalit potenciální nedostatky a chyby v kódu.

Bohužel tato funkce nelze využít u všech typů desek, ale pouze u těch novějších. Podporované typy desek jsou jen ty, které využívají architekturu SAMD. [34]

Jedná se o tyto modely desek.

- MKR Zero
- MKR WiFi 1010
- MKR FOX 1200
- MKR WAN 1300
- MKR WAN 1310
- MKR GSM 1400
- MKR NB 1500

- MKR VIDOR 4000
- Nano 33 IoT
- Zero

## 5.2 Wiring

Wiring je programovací jazyk vytvořený pro programování mikrokontroléru bez specifických znalostí hardwaru. V současné době je nejznámější jako součást open-source platformy Arduino, kde má podobu frameworku v jazyce C++. Wiring vznikl pro vývojový kit podobný Arduino a vychází z open-source projektu. Pro programování v jazyce Wiring se nejčastěji používá integrované vývojové prostředí Arduino IDE, k dispozici jsou ale i další vývojová prostředí jako Arduino Eclipse. Wiring vyžaduje mikrokontrolér se zaváděcím programem, typicky desku Arduino osazenou čipy ATmega.

Program v jazyce Wiring má obvykle dvě hlavní části.

- `setup()` – funkce, která se spustí jednou na začátku programu a zpravidla obsahuje počáteční nastavení
- `loop()` – automaticky opakovaně volaná funkce, když je deska Arduino připojena k napájení

### 5.2.1 Deklarace proměnných

Je potřeba aby všechny proměnné byly ještě před prvním použitím deklarovány. Deklarace je potřebná k tomu, aby se předem definoval její datový typ, většinou je při deklaraci proměnných přiřazovány počáteční hodnota. Deklarovat proměnnou budete v programu jen jednou, hodnotu proměnné můžete v programu měnit podle potřeby.

```
int rele1 = 51;  
int rele2 = 49;  
int rele3 = 50;  
int rele4 = 48;
```

## 5.2.2 Datové typy

Jak už bylo zmíněno, deklarace datového typu je zapotřebí provést hned na začátku před jejím samotným použitím.

### 5.2.2.1 Datový typ *Byte*

Datový typ byte ukládá hodnoty jako osmibitové číselné hodnoty bez desetinných míst. Ukládá hodnotu v rozsahu 0 až 255.

```
byte cisloX = 200;
```

### 5.2.2.2 Datový typ *Integer*

Datový typ integer je jeden z nejpoužívanějších datových typů, zároveň je jeho využití velmi vysoké. Tento datový typ volíme tak, kde je zapotřebí ukládat celá čísla. Ukládaná hodnota může být v rozsahu -32 768 až 32767, ukládá se jako 16bitová hodnota.

```
int promena = 8;
```

### 5.2.2.3 Datový typ *Long*

Datový typ long je obdobný datovému typu integer, stejně tak jako integer se používá pro ukládání celých čísel. Čísla jsou uložena jako 32bitová s rozsahem  $-3,4028235 \times 10^{38}$  až  $3,4028235 \times 10^{38}$ .

```
long velkeCislo = 100000;
```

### 5.2.2.4 Datový typ *Float*

Datový typ float

Datový typ float je určen pro operace s čísly s plovoucí desetinnou čárkou. Čísla s plovoucí desetinnou čárkou mají větší rozlišení než celá čísla. Ukládají se jako 32bitová s rozsahem  $-3,4028235 \times 10^{38}$  až  $3,4028235 \times 10^{38}$ .

```
float mojeCislo = 5.8;
```

### 5.2.2.5 Datový typ *Double*

Datový typ double je stejně jako float určen pro operace s čísly s plovoucí desetinnou čárkou. Datový typ double je některých typech Arduina stejný jako datový typ float, ale na některých

se může jeho využitý prostor pro uložení tohoto typu zdvojnásobit. Je proto důležité si před použitím na konkrétním Arduinu zkontrolovat jeho implementaci na konkrétním typu.

```
double desetinneCislo = 12.5;
```

### 5.2.2.6 Datový typ Char

Datový typ char složí pro ukládání jednoho písmene. Program s uloženým číslem pracuje jako s číslem, jelikož si v paměti uloží jeho kód.

```
char pismeno = 'X';
```

## 5.3 Ukázkové programy jednotlivých periférií

### 5.3.1 Klávesnice

Ukázkový program pro použití klávesnice.

Po načtení knihovny je zapotřebí definovat počet řádků a počet sloupců, následně si definujeme znaky na klávesnici. Definujeme zapojené piny pro řádky a sloupce. Nastavíme komunikaci pro sériový monitor, tak abychom viděli výpis stisknutých kláves na sériovém monitoru v prostředí Arduino IDE.

```
#include <Keypad.h> // načtení knihovny keypad.h

const byte ROWS = 4; // definice počtu řádků
const byte COLS = 3; // definice počtu sloupců

char keys[ROWS][COLS] = { // definice znaků klávesnice pomocí tabulky
  {'1', '2', '3'},
  {'4', '5', '6'},
  {'7', '8', '9'},
  {'*', '0', '#'}
};

byte rowPins[ROWS] = {8, 7, 6, 5}; //připojení řádků pomocí pinů
byte colPins[COLS] = {4, 3, 2}; //připojení sloupců pomocí pinů

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS ); //inicializace instance třídy z knihovny Keypad.h

void setup() {
  Serial.begin(9600); // nastavení sériového monitor pro výpis znaků
}

void loop() {
  char key = keypad.getKey(); // čtení znaků

  if (key) { // Když je libovolné tlačítko stisknuto
    Serial.print(key); // Výpis znaku do sériové komunikace
  }
}
```

Obrázek 68 Program pro klávesnici

### 5.3.2 LCD Display

Ukázkový program pro využití LCD displeje s I2C sběrnici.

Na začátku je zapotřebí načíst knihovny Wire a LiquidCrystal\_I2C tak abychom mohli pracovat s LCD displejem a využít tak sběrnici I2C, která nám umožní připojení pouze pomocí dvou datových pinů. Nejdříve si nastavíme adresu LCD a zadáme počet znaků a počet řádků na displeji. Poté inicializujeme LCD displej a zapneme podsvícení. Poté už máme vypsané údaje, které chceme, aby se nám zobrazili na displeji.

```
#include <Wire.h> // načtení knihovny wire
#include <LiquidCrystal_I2C.h> // načtení knihovny pro LCD komunikaci

LiquidCrystal_I2C lcd(0x27, 16, 4); // nastavení adresy LCD 0x27 a počet znaků na řádek s počtem řádků

void setup()
{
  lcd.init(); // inicializace LCD displeje
  lcd.backlight(); // zapnutí podsvícení LCD displeje
}

void loop()
{
  lcd.setCursor(4, 0); // nastavení kurzoru na 3 znak na nultém (prvním) řádku
  lcd.print("Ahoj svete"); // výpis textu na LCD Ahoj svete
  lcd.setCursor(4, 1); // nastavení kurzoru na 2 znak na druhém řádku
  lcd.print("LCD displej"); // výpis textu na LCD LCD displej
  lcd.setCursor(6, 2); // nastavení kurzoru na 0 znak na třetím řádku
  lcd.print("Arduino"); // výpis textu na LCD Arduino
  lcd.setCursor(6, 3); // nastavení kurzoru na 0 znak na třetím řádku
  lcd.print("-----"); // výpis textu na LCD -----
}
```

Obrázek 69 Program pro LCD display

Po nahrání programu se na LCD display zobrazí text na jednotlivých řádcích.

Ahoj svete

LCD displej

Arduino

-----

### 5.3.3 Krokový motor

Program pro ovládání krokového motoru. Nejprve je zapotřebí definovat použité piny, kterými je ovládán driver a následně krokový motor. Definovat piny stepPin a dirPin jako

výstupní piny a je nastaven dirPin na hodnotu jedna. Poté jsou vypsané dva cykly for, které otáčí motorem na jednu a poté na druhou stranu.

```

const int dirPin = 40; //definice pinu pro rotaci motoru na danou stranu
const int stepPin = 42; //definice pinu pro signal na otočení o jeden krok motoru
const int stepsPerRevolution = 2000; //definice proměnné pro daný počet kroků pro opakování

void setup()
{
  pinMode(stepPin, OUTPUT); // nastavení pinu stepPin jako výstup
  pinMode(dirPin, OUTPUT); // nastavení pinu dirPin jako výstup
}
void loop()
{
  digitalWrite(dirPin, HIGH); //nastavení pinu na logickou hodnotu 1 pro opačnou rotaci motoru

  for(int x = 0; x < stepsPerRevolution; x++) //cyklus for pro počet otočení motoru na danou stranu
  {
    digitalWrite(stepPin, HIGH); //nastavení pinu na logickou hodnotu 1
    delayMicroseconds(800); //zpoždění 400ms
    digitalWrite(stepPin, LOW); //nastavení pinu na logickou hodnotu 0
    delayMicroseconds(800); //zpoždění 400ms
  }
  delay(1000); //zpoždění 1s mezi otáčením motoru

  digitalWrite(dirPin, LOW); //nastavení pinu na logickou hodnotu 0 pro opačnou rotaci motoru

  for(int x = 0; x < stepsPerRevolution; x++) //cyklus for pro počet otočení motoru na danou stranu
  {
    digitalWrite(stepPin, HIGH); //nastavení pinu na logickou hodnotu 1
    delayMicroseconds(800); //zpoždění 400ms
    digitalWrite(stepPin, LOW); //nastavení pinu na logickou hodnotu 0
    delayMicroseconds(800); //zpoždění 400ms
  }
  delay(1000); //zpoždění 1s mezi otáčením motoru
}

```

Obrázek 70 Program pro krokový motor

Po nahrání programu se krokový motor začne otáčet na jednu stranu po dobu jedné sekundy a na stranu druhou po dobu také jedné sekundy.

### 5.3.4 Ultrazvukový senzor

Ukázkový program ultrazvukového senzoru.

Na začátku jsou definované proměnné, poté je nastavena sériová komunikace a nastaven pin trigPin jako výstupní a echoPin je nastaven jako vstupní. Dále už je napsán samotný program pro to aby senzor detekovat vzdálenost a naměřené hodnoty vypisovat na sériového monitoru v prostředí Arduimo IDE.

```
// definice potřebných proměnných
const int trigPin = 9;
const int echoPin = 10;
long duration;
int distanceCm, distanceInch;

void setup() {
  Serial.begin(9600);
  pinMode(trigPin, OUTPUT);           // nastavení pinu trigPin jako výstup
  pinMode(echoPin, INPUT);           // nastavení pinu echoPin jako vstup
}

void loop() {
  digitalWrite(trigPin, LOW);         // zápis logické hodnoty 0 na pin 9
  delayMicroseconds(2);               // zpoždění 2 mikrosekundy
  digitalWrite(trigPin, HIGH);        // zápis logické hodnoty 1 na pin 9
  delayMicroseconds(10);              // zpoždění 10 mikrosekund
  digitalWrite(trigPin, LOW);         // zápis logické hodnoty 0 na pin 9
  duration = pulseIn(echoPin, HIGH);  // čtení pulsů logické hodnoty 1
  distanceCm = duration * 0.034 / 2;  // pro výpočet v jednotkách cm je důležité provést převod informace
  distanceInch = duration * 0.0133 / 2; // pro výpočet v jednotkách palců je důležité provést převod informace
  Serial.print("Distance: ");        // Prints string "Distance" on the Serial
  Serial.print(distanceCm);           // výpis vzdálenosti na Serial displej v jednotkách cm
  Serial.print(" cm\n");              // výpis textu cm na Serial displej a \n pro skok na nový řádek
  delay(10);                          // zpoždění 10ms
  Serial.print("Distance: ");         // výpis textu Distance
  Serial.print(distanceInch);         // výpis hodnoty proměnné na Serial displej
  Serial.print(" inch\n");           // výpis textu inch na Serial displej a \n pro skok na nový řádek
  delay(10);                          // zpoždění 10ms
}
```

Obrázek 71 Program pro ultrazvukový senzor

Po nahrání programu program začne vypisovat vzdálenost na kterou senzor detekuje překážku.

### 5.3.5 Modul relé

Ukázkový program pro spínání relé modulu.

V programu se nejdříve definují piny, kterými se jednotlivé relé ovládá. Poté se nastaví jednotlivým pinům, že se jedná a piny výstupní. Nastavíme na jednotlivé piny hodnotu jedna, aby relé zůstaly vypnuty, relé obvod se spíná hodnotou nula. Poté už je jen napsán program, který postupně spíná a rozpíná jednotlivé relé, kde je každé relé sepnuto na jednu sekundu.



```
int rele1 = 51;           //definice pinu rele1 na pin 51
int rele2 = 49;           //definice pinu rele2 na pin 49
int rele3 = 50;           //definice pinu rele3 na pin 50
int rele4 = 48;           //definice pinu rele4 na pin 48

void setup()
{
  pinMode(rele1,OUTPUT);  //nastavení pinu rele1 jako výstup
  pinMode(rele2,OUTPUT);  //nastavení pinu rele2 jako výstup
  pinMode(rele3,OUTPUT);  //nastavení pinu rele3 jako výstup
  pinMode(rele4,OUTPUT);  //nastavení pinu rele4 jako výstup
  digitalWrite(rele1,HIGH); //nastavení pinu rele1 na hodnotu 1
  digitalWrite(rele2,HIGH); //nastavení pinu rele2 na hodnotu 1
  digitalWrite(rele3,HIGH); //nastavení pinu rele3 na hodnotu 1
  digitalWrite(rele4,HIGH); //nastavení pinu rele4 na hodnotu 1
}

void loop()
{
  digitalWrite(rele1,LOW); //nastavení pinu na hodnotu 0 - rele1 sepnuto
  delay(1000);             //zpoždění 1s
  digitalWrite(rele1,HIGH); //nastavení pinu na hodnotu 1 - rele1 rozepnuto

  digitalWrite(rele2,LOW); //nastavení pinu na hodnotu 0 - rele2 sepnuto
  delay(1000);             //zpoždění 1s
  digitalWrite(rele2,HIGH); //nastavení pinu na hodnotu 1 - rele2 rozepnuto

  digitalWrite(rele3,LOW); //nastavení pinu na hodnotu 0 - rele3 sepnuto
  delay(1000);             //zpoždění 1s
  digitalWrite(rele3,HIGH); //nastavení pinu na hodnotu 1 - rele3 rozepnuto

  digitalWrite(rele4,LOW); //nastavení pinu na hodnotu 0 - rele4 sepnuto
  delay(1000);             //zpoždění 1s
  digitalWrite(rele4,HIGH); //nastavení pinu na hodnotu 1 - rele4 rozepnuto
}
```

Obrázek 72 Program pro relé modul

### 5.3.6 Modul s RGB LED

Ukázkový program na ovládání RGB LED modulu. Na začátku je nutné si načíst knihovnu Adafruit\_Neopixel, která umí pracovat s programovatelnými LED moduly, která se ovládají jen přes jeden pin. Poté se definován ovládací pin a počet RGB LED diod v daném modulu a našem případě se jedná o 16. Následně nastavíme barvu, kterou zadáváme pomocí barevného modelu RGB. Dále inicializujeme všechny pixely a nastavíme jakou intenzitou má LED diody svítit, zde nastavujeme v rozsahu 0-255. Poté máme definovaný cyklus for ve kterém se nám postupně rozsvítí a zhasnou jednotlivé LED Diody.

```
#include <Adafruit_NeoPixel.h> //načtení knihovny

#define PIN      11 //definice pinu pro data
#define NUM_LEDS 16 //definice počtu LED diod

// Parameter 1 = number of pixels in strip
// Parameter 2 = pin number (most are valid)
// Parameter 3 = pixel type:
//   NEO_GRB Pixels are wired for GRB bitstream (most NeoPixel products)
//   NEO_RGB Pixels are wired for RGB bitstream (v1 FLORA pixels, not v2)
Adafruit_NeoPixel strip = Adafruit_NeoPixel(NUM_LEDS, PIN, NEO_GRB);

uint32_t color = strip.Color(75, 250, 100); // nastavení barvy pomocí RGB složky

void setup() {
  strip.begin();
  strip.show(); // inicializace všech pixelů
  strip.setBrightness(40/255); // 40/255 brightness (about 15%)
}

void loop() {
  for (int i = 0; i < NUM_LEDS; i++) {
    strip.setPixelColor(i, color); //nastavení barvy dané LED diody
    strip.show(); //zobrazení aktivace LED kruhu-pásku
    delay(1000); //zpoždění 1s

    strip.setPixelColor(i, 0); //nastavení barvy dané LED diody - zhasnutí - 0
    strip.show(); //zobrazení aktivace LED kruhu-pásku
    delay(1000); //zpoždění 1s
  }
}
```

Obrázek 73 Program pro RGB LED modul

Po nahrání programu se postupně rozsvítí a zhasnou všechny LED diody.

## 6 SADA ŘEŠENÝCH ÚLOH

Následující sada úloh je zaměřena komponenty, které jsou součástí platformy a je možné je využít při výuce informatiky.

Na jednotlivých úlohách si žáci rozvíjí spoustu kompetencí jako je:

- Kompetence k učení
- Kompetence k řešení problému
- Kompetence pracovní
- Kompetence komunikativní
- Kompetence digitální

### 6.1 Úloha 1- Výpis znaků klávesnice na displej

#### 6.1.1 Cíle výuky:

- Žák se seznámí s komponenty jako je klávesnice a LCD displej.
- Žák dokáže vytvořit program pro tyto komponenty.
- Žák dokáže modifikovat ukázkový program.
- Žák chápe připojení jednotlivých komponentů
- Žák umí pracovat s potřebnými knihovnamí.
- Žák umí dohledávat potřebné informace.

#### 6.1.2 Zadání:

- Ověřte zapojení jednotlivých komponent.
- Vytvořte program, který bude vypisovat znaky klávesnice na LCD displej.
- Vytvořený program nahrajte do Arduina.
- Ověřte funkčnost programu.

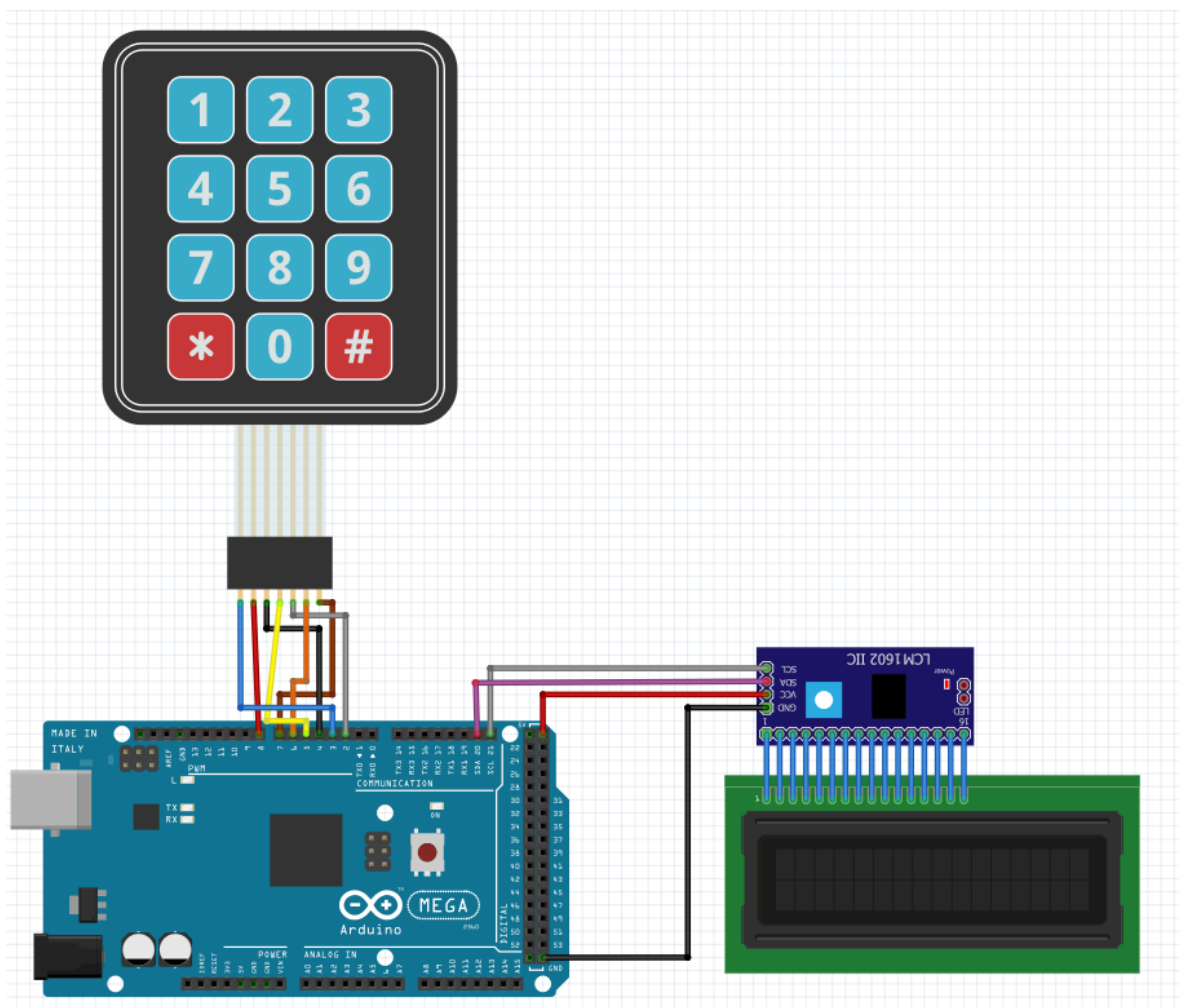
#### 6.1.3 Potřebné vybavení:

- Výuková platforma se zapojeným LCD displejem a klávesnicí.
- Propojovací kabel.
- Počítač s nainstalovaným softwarem Arduino IDE.

#### 6.1.4 Předpokládaná časová dotace:

- Jedna vyučovací hodina – 45minut.

#### 6.1.5 Schéma zapojení jednotlivých komponent



Obrázek 74 Schéma zapojení pro úkol 1

## 6.1.6 Vytvoření programu

```

#include <Wire.h> // načtení knihovny wire
#include <LiquidCrystal_I2C.h> // načtení knihovny pro LCD komunikaci

LiquidCrystal_I2C lcd(0x27, 16, 4); // nastavení adresy LCD 0x27 a počet znaků na řádek s počtem řádků

#include <Keypad.h> // načtení knihovny keypad.h

const byte ROWS = 4; // definice počtu řádků
const byte COLS = 3; // definice počtu sloupců
char keys[ROWS][COLS] = { // definice znaků klávesnice pomocí tabulky
  {'1', '2', '3'},
  {'4', '5', '6'},
  {'7', '8', '9'},
  {'*', '0', '#'}
};
byte rowPins[ROWS] = {8, 7, 6, 5}; //připojení řádků pomocí pinů
byte colPins[COLS] = {4, 3, 2}; //připojení sloupců pomocí pinů

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS ); //inicializace instance třídy z knihovny keypad.h

void setup() {
  Serial.begin(9600); // sériový monitor pro výpis znaků
  lcd.init(); // inicializace LCD displeje
  lcd.backlight(); // zapnutí podsvícení LCD displeje
}

void loop() {
  char key = keypad.getKey(); // čtení znaků

  if (key) { // Když je libovolné tlačítko stisknuto
    lcd.print(key); // výpis znaku key na LCD displej
  }
}

```

Obrázek 75 Výpis znaků klávesnice na displej

Po úspěšném nahrání se rozsvítí displej a po stisknutí jakékoliv klávesy se daný symbol zobrazí na displeji. Jednotlivé kroky jsou popsány přímo v programu.

## 6.1.7 Bonusový úkol

Upravte stávající program tak, abyste nahradili čísla klávesnice za prvních 10 znaků v abecedě.

## 6.2 Úloha 2 - Výpis vzdálenosti na LCD displej

### 6.2.1 Cíle výuky:

- Žák se seznámí s komponenty jako je LCD displej a ultrazvukový senzor.
- Žák dokáže vytvořit program pro ultrazvuková senzor
- Žák dokáže pomocí senzoru měřit vzdálenost
- Žák dokáže vypsát naměřenou vzdálenost na LCD displej

### 6.2.2 Zadání:

- Ověřte, případně proveďte zapojení jednotlivých komponent dle schématu.
- Vytvořte program, který bude vypisovat vzdálenost na LCD displej.
- Vytvořený program nahrajte do Arduina.
- Ověřte funkčnost programu.

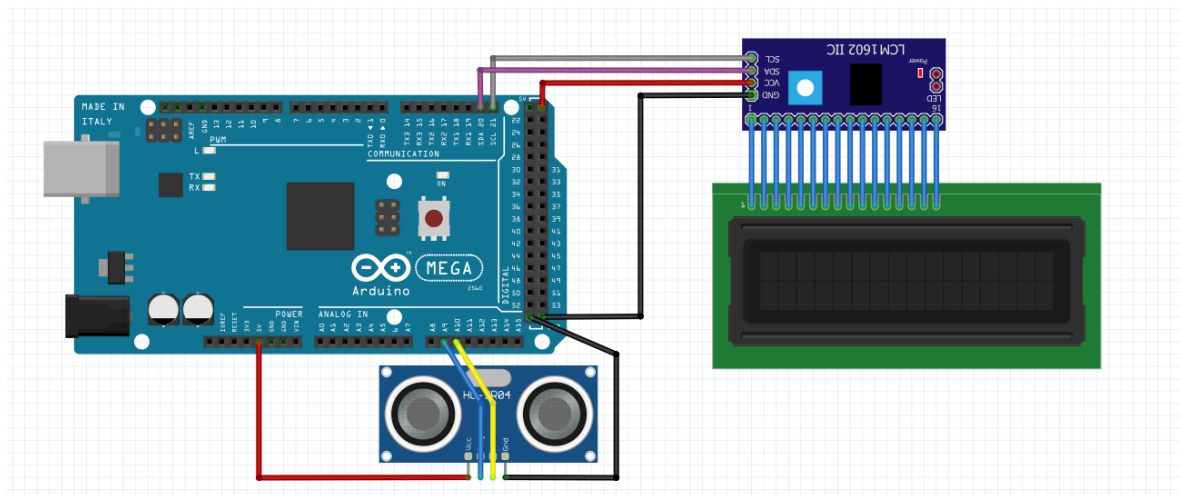
### 6.2.3 Potřebné vybavení:

- Výuková platforma se zapojeným LCD displejem a ultrazvukovým senzorem.
- Propojovací kabel.
- Počítač s nainstalovaným softwarem Arduino IDE.

### 6.2.4 Předpokládaná časová dotace:

- Jedna vyučovací hodina – 45minut.

### 6.2.5 Schéma zapojení



Obrázek 76 Schéma zapojení pro úkol 2

### 6.2.6 Vytvoření programu

```

#include <LiquidCrystal_I2C.h>           // načtení knihovny pro LCD komunikaci

LiquidCrystal_I2C lcd(0x27, 16, 4);     // nastavení adresy LCD 0x27 a počet znaků na řádek s počtem řádků

// definice potřebných proměnných
const int trigPin = 9;
const int echoPin = 10;
long duration;
int distanceCm, distanceInch;

void setup() {
  lcd.init();                          // inicializace LCD displeje
  lcd.backlight();                      // nastavení podsvícení LCD displeje
  pinMode(trigPin, OUTPUT);             // nastavení pinu trigPin jako výstup
  pinMode(echoPin, INPUT);             // nastavení pinu echoPin jako vstup
}

void loop() {
  digitalWrite(trigPin, LOW);           // zápis logické hodnoty 0 na pin 9
  delayMicroseconds(2);                // zpoždění 2 mikrosekundy
  digitalWrite(trigPin, HIGH);         // zápis logické hodnoty 1 na pin 9
  delayMicroseconds(10);               // zpoždění 10 mikrosekund
  digitalWrite(trigPin, LOW);          // zápis logické hodnoty 0 na pin 9
  duration = pulseIn(echoPin, HIGH);   // čtení pulsů logické hodnoty 1
  distanceCm = duration * 0.034 / 2;   // pro výpočet v jednotkách cm
  distanceInch = duration * 0.0133 / 2; // pro výpočet v jednotkách palců
  lcd.setCursor(0, 0);                 // nastavení kurzoru na pozici nultý znak na nultém řádku
  lcd.print("Distance: ");             // Prints string "Distance" on the LCD
  lcd.print(distanceCm);               // výpis vzdálenosti na LCD displej v jednotkách cm
  lcd.print(" cm");                    // výpis textu cm na LCD displej
  delay(10);                            // zpoždění 10ms
  lcd.setCursor(0, 1);                 // nastavení kurzoru na nultý znak na prvním řádku
  lcd.print("Distance: ");             // výpis textu Distance
  lcd.print(distanceInch);             // výpis hodnoty proměnné na LCD displej
  lcd.print(" inch");                  // výpis textu inch na LCD displej
  delay(10);                            // zpoždění 10ms
}

```

Obrázek 77 Výpis vzdálenosti na LCD displej

Po úspěšném nahrání programu se na LCD displeji bude zobrazovat vzdálenost v reálném čase, a to ve dvou jednotkách c centimetrech a v palcích. Jednotlivé kroky jsou popsány přímo v programu.

### 6.2.7 Bonusový úkol

Upravte stávající program tak, aby se na displeji vypisovala vzdálenost místo centimetrů metry a místo palců stopy.

## 6.3 Úloha 3 – Ovládání relé modulu pomocí klávesnice

### 6.3.1 Cíle výuky:

- Žák se seznámí s komponenty jako je LCD displej, klávesnice a relé modul.
- Žák dokáže vytvořit program pro ovládání relé modulu s využitím klávesnice.
- Žák dokáže program ukázkový program modifikovat.

- Žák dokáže propojit více komponentů v jednom programu

### 6.3.2 Zadání:

- Ověřte zapojení, případně zapojte jednotlivé komponenty dle schématu.
- Vytvořte program, jehož funkce bude ovládat jednotlivá relé pomocí čísel 1-4 na klávesnici a jednotlivé stavy relé zobrazit na LCD displeji
- Vytvořený program nahrajte do Arduina.
- Ověřte funkčnost programu.

### 6.3.3 Potřebné vybavení:

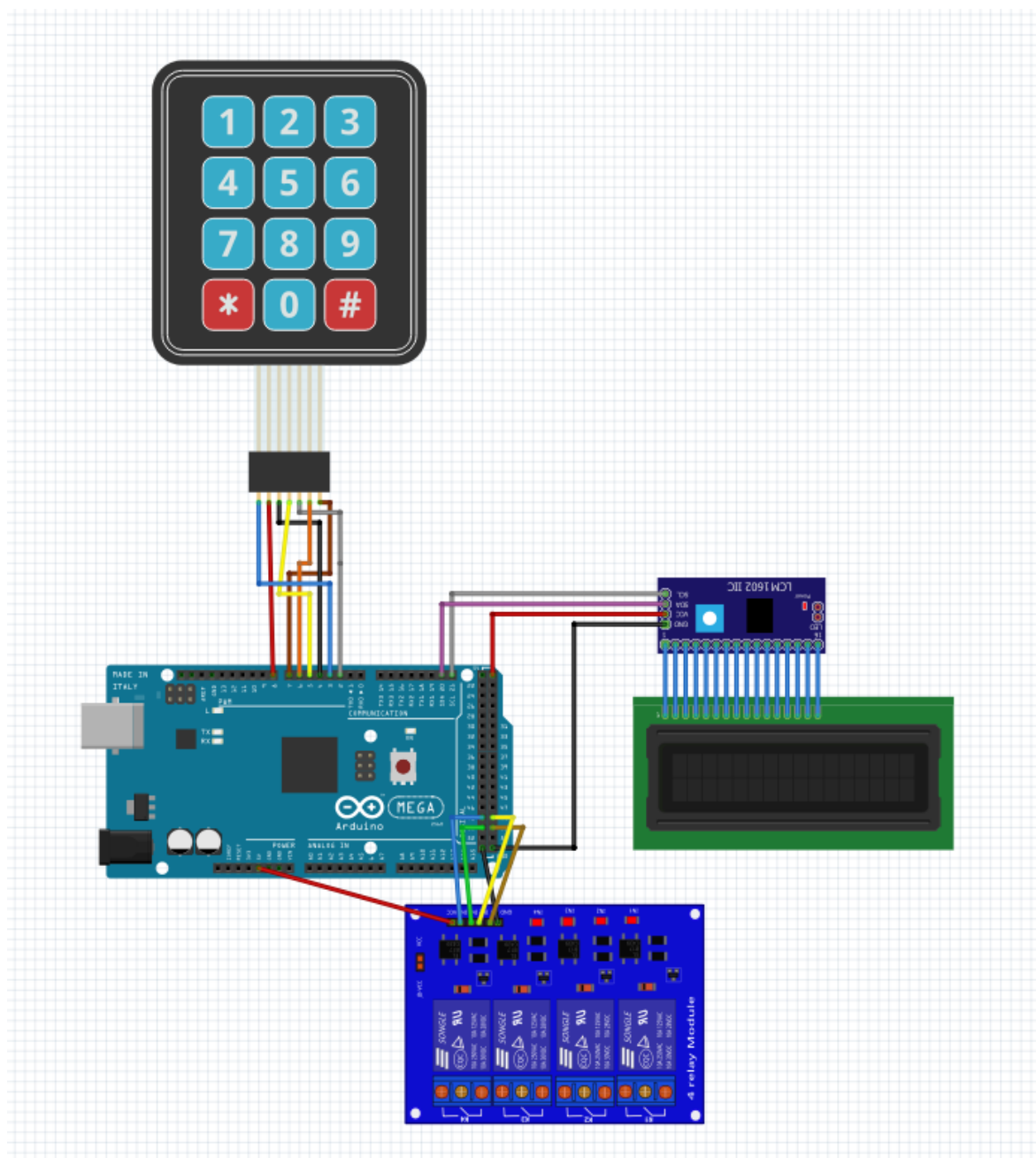
- Výuková platforma se zapojeným LCD displejem, klávesnicí a relé modul.
- Propojovací kabel.
- Počítač s nainstalovaným softwarem Arduino IDE.

### 6.3.4 Předpokládaná časová dotace:

- Dva dvouhodinové úseky – 2x90 minut



### 6.3.5 Schéma zapojení



### 6.3.6 Vytvořený program

```
#include <Keypad.h>
#include <LiquidCrystal_I2C.h> //pro LCD DISPLEJ

// inicialize knihovny pro LCD displej se sbernici I2C
LiquidCrystal_I2C lcd(0x27, 16, 4); //ADRESA,POCETZNAKU,POCETRADKU

const int ROW_NUM = 4; //4 řádky
const int COLUMN_NUM = 3; //3 sloupce

#define rele1 51
#define rele2 49
#define rele3 50
#define rele4 48

//definice pomocných proměnných
int prom = 0;
int prom1 = 0;
int prom2 = 0;
int prom3 = 0;

char keys[ROW_NUM][COLUMN_NUM] = {
  {'1', '2', '3'},
  {'4', '5', '6'},
  {'7', '8', '9'},
  {'*', '0', '#'}
};

byte pin_rows[ROW_NUM] = {8, 7, 6, 5}; //definice číslo pinů klávesnice od řádků
byte pin_column[COLUMN_NUM] = {4, 3, 2}; //definice číslo pinů klávesnice od sloupců

Keypad keypad = Keypad( makeKeymap(keys), pin_rows, pin_column, ROW_NUM, COLUMN_NUM );

void setup() {

  //INICIALIZACE DISPLEJE
  lcd.init(); //inicializace lcd
  lcd.backlight(); //kontrast displeje

  //nastavení pinů jako výstup
  pinMode(rele1, OUTPUT);
  pinMode(rele2, OUTPUT);
  pinMode(rele3, OUTPUT);
  pinMode(rele4, OUTPUT);
```

Obrázek 78 Ovládání relé pomocí klávesnice 1/3

```
//nastavení pinů relátek na rozepnutý stav
digitalWrite(rele1, HIGH);
digitalWrite(rele2, HIGH);
digitalWrite(rele3, HIGH);
digitalWrite(rele4, HIGH);

lcd.setCursor(5, 1); //nastavení kurzoru na 5 znak na prvním radku
lcd.print(" 1 2 3 4 ");
lcd.setCursor(5, 2); //nastavení kurzoru na 5 znak na druhém radku
lcd.print(" 0 0 0 0 ");
}

void loop() {
  char key = keypad.getKey(); // čtení hodnoty znaku

  if (key == '1') { //pokud bylo stisknuto tlačítko 1
    if (prom == 0) //pokud je proměnná prom = 0 provede se kód v podmínce
    {
      digitalWrite(rele1, LOW); //nastavení pinu rele1 na logickou hodnotu 0 - relé sepnuto
      lcd.setCursor(7, 2); //nastavení kurzoru na 6 znak na druhem radku
      lcd.print("I"); //výpis textu I - informace, že relé je sepnuto
      prom++; //přičtení čísla 1 do proměnné prom
    }
    else
    {
      digitalWrite(rele1, HIGH); //nastavení pinu rele1 na logickou hodnotu 1 - relé rozepnuto
      lcd.setCursor(7, 2); //nastavení kurzoru na 6 znak na druhem radku
      lcd.print("0"); //výpis textu 0 - informace, že relé je rozepnuto
      prom--; //odečtení čísla 1 do proměnné prom
    }
  }

  if (key == '2') { //pokud bylo stisknuto tlačítko 2
    if (prom1 == 0) //pokud je proměnná prom1 = 0 provede se kód v podmínce
    {
      digitalWrite(rele2, LOW); //nastavení pinu rele2 na logickou hodnotu 0 - relé sepnuto
      lcd.setCursor(9, 2); //nastavení kurzoru na 9 znak na druhem radku
      lcd.print("I"); //výpis textu I - informace, že relé je sepnuto
      prom1++; //přičtení čísla 1 do proměnné prom1
    }
  }
}
```

Obrázek 79 Ovládání relé pomocí klávesnice 2/3

```
else
{
    digitalWrite(rele2, HIGH);           //nastavení pinu rele2 na logickou hodnotu 1 - relé rozepnuto
    lcd.setCursor(9, 2);                 //nastavení kurzoru na 9 znak na druhem radku
    lcd.print("0");                      //výpis textu 0 - informace, že relé je rozepnuto
    prom1--;                             //odečtení čísla 1 do proměnné prom1
}
}

if (key == '3') {                       //pokud bylo stiksnuo tlačítko 3
    if (prom2 == 0)                     //pokud je proměnná prom2 = 0 provede se kód v podmínce
    {
        digitalWrite(rele3, LOW);      //nastavení pinu rele3 na logickou hodnotu 0 - relé sepnuto
        lcd.setCursor(11, 2);          //nastavení kurzoru na 11 znak na druhem radku
        lcd.print("I");                //výpis textu I - informace, že relé je sepnuto
        prom2++;                       //přičtení čísla 1 do proměnné prom2
    }
    else
    {
        digitalWrite(rele3, HIGH);     //nastavení pinu rele3 na logickou hodnotu 1 - relé rozepnuto
        lcd.setCursor(11, 2);          //nastavení kurzoru na 11 znak na druhem radku
        lcd.print("0");                //výpis textu 0 - informace, že relé je rozepnuto
        prom2--;                       //odečtení čísla 1 do proměnné prom2
    }
}

if (key == '4') {                       //pokud bylo stiksnuo tlačítko 4
    if (prom3 == 0)                     //pokud je proměnná prom3 = 0 provede se kód v podmínce
    {
        digitalWrite(rele4, LOW);      //nastavení pinu rele4 na logickou hodnotu 0 - relé sepnuto
        lcd.setCursor(13, 2);          //nastavení kurzoru na 13 znak na druhem radku
        lcd.print("I");                //výpis textu I - informace, že relé je sepnuto
        prom3++;                       //přičtení čísla 1 do proměnné prom3
    }
    else
    {
        digitalWrite(rele4, HIGH);     //nastavení pinu rele4 na logickou hodnotu 0 - relé rozepnuto
        lcd.setCursor(13, 2);          //nastavení kurzoru na 13 znak na druhem radku
        lcd.print("0");                //výpis textu 0 - informace, že relé je rozepnuto
        prom3--;                       //odečtení čísla 1 do proměnné prom3
    }
}
}
```

Obrázek 80 Ovládání relé pomocí klávesnice 3/3

Po úspěšném nahrání programu se na displeji zobrazí čísla 1,2,3,4, které reprezentují jednotlivá relé a pod každým číslem je 0, která reprezentuje stav, v jakém se nachází jednotlivé relé. 0 značí, že dané relé není sepnuto a 1 značí že dané relé je sepnuto. Jednotlivé relé se dají přepínat do těchto stavů pomocí číslic 1-4 na klávesnici.

### 6.3.7 Bonusový úkol

Upravte stávající program tak, změnilo ovládání jednotlivých relé místo kláves 1-4 na 5-8.

## 6.4 Úloha 4 – Ovládání RGB LED modulu

### 6.4.1 Cíle výuky:

- Žák se seznámí s komponenty jako je RGB LED modul.
- Žák dokáže vytvořit program pro tuto komponentu.
- Žák si uvědomí zapojení dané komponenty, které není obvyklé

### 6.4.2 Zadání:

- Ověřte zapojení, případně zapojte podle schématu RGB LED modul
- Vytvořte program, který bude mít za úkol spínat a vypínat každou druhou RGB LED diodu.
- Barvu nastavte na světle modrou.
- Vytvořený program nahrajte do Arduina.
- Ověřte funkčnost programu.

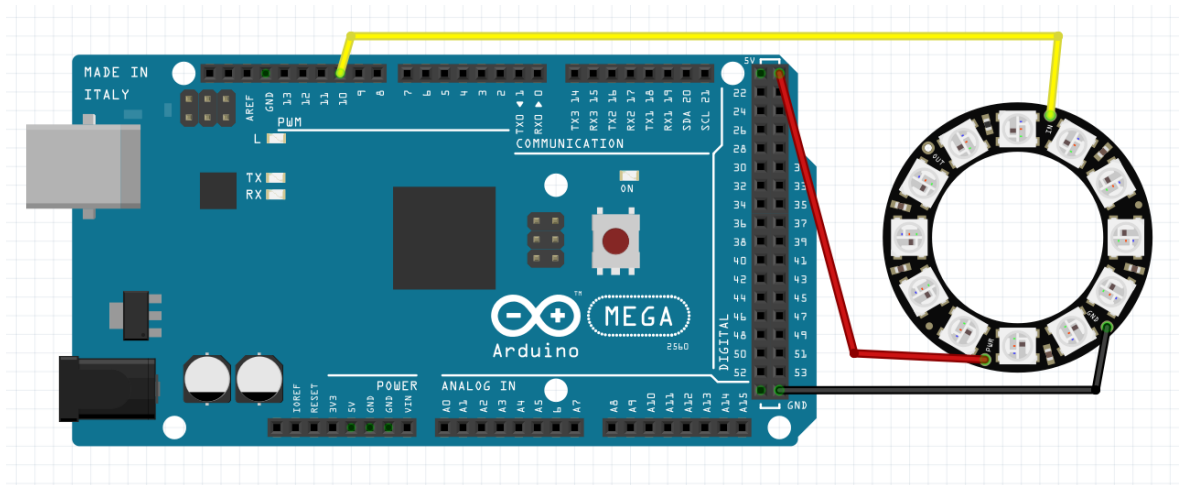
### 6.4.3 Potřebné vybavení:

- Výuková platforma se zapojeným RGB LED modulem.
- Propojovací kabel.
- Počítač s nainstalovaným softwarem Arduino IDE.

### 6.4.4 Předpokládaná časová dotace:

- Jedna vyučovací hodina – 45minut.

### 6.4.5 Schéma zapojení



Obrázek 81 Schéma zapojení úkolu 4

### 6.4.6 Vytvořený program

```
#include <Adafruit_NeoPixel.h> //načtení knihovny

#define PIN      11  //definice pinu pro data
#define NUM_LEDS 16 //definice počtu LED diod

// Parametr 1 = počet pixelů - LED diod
// Parametr 2 = číslo pinu
// Parametr 3 = pixel typ
Adafruit_NeoPixel strip = Adafruit_NeoPixel(NUM_LEDS, PIN, NEO_GRB);

uint32_t color = strip.Color(30, 144, 255); // nastavení modré barvy pomocí RGB složky

void setup() {
  strip.begin();
  strip.show(); // inicializace všech pixelů
  strip.setBrightness(40/255); // 40/255 brightness (about 15%)
}

void loop() {
  for (int i = 0; i < NUM_LEDS; i+=2) { //každá druhá dioda bude svítit
    strip.setPixelColor(i, color); //nastavení barvy dané LED diody
    strip.show(); //zobrazení aktivace LED kruhu-pásku
    delay(1000); //zpoždění 1s

    strip.setPixelColor(i, 0); //nastavení barvy dané LED diody - zhasnutí - 0
    strip.show(); //zobrazení aktivace LED kruhu-pásku
    delay(1000); //zpoždění 1s
  }
}
```

Obrázek 82 Ovládání RGB LED modulu

Po nahrání programu se začne postupně rozsvěcovat každá druhá RGB LED dioda v barvě světle modré.

### 6.4.7 Bonusový úkol

Upravte program tak, aby namísto světle modré barvy byla barva červená a svítila jen každá čtvrtá RGB LED dioda.

## 6.5 Úloha 5– Aktivace krokového motoru pomocí bezdotykového snímače

### 6.5.1 Cíle výuky:

- Žák se seznámí s komponenty jako je krokový motor, driver pro krokové motory a bezdotykový snímač
- Žák dokáže vytvořit program pro tyto komponenty.
- Žák dokáže vytvořit praktický program

### 6.5.2 Zadání:

- Ověřte zapojení, případně zapojte komponenty podle schématu.
- Vytvořte program, který ovládat krokový motor s základě senzoru přiblížení.
- Pokud se přiblížíte k senzoru a senzor detekuje překážku, začne se motor otáčet, v opačném případě motor se nebude otáčet.
- Vytvořený program nahrajte do Arduina.
- Ověřte funkčnost programu.

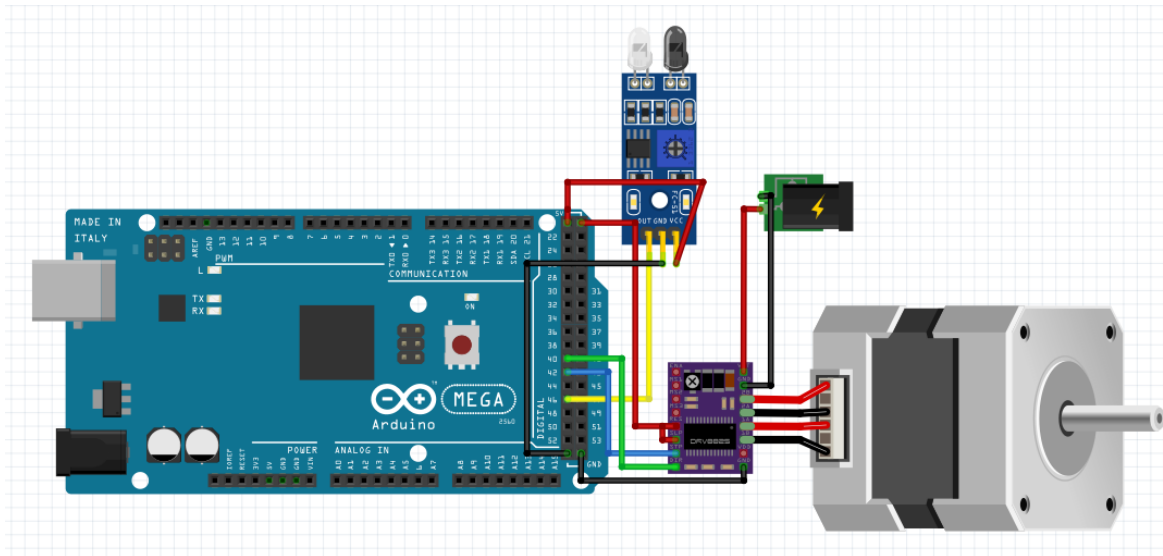
### 6.5.3 Potřebné vybavení:

- Výuková platforma se zapojeným krokovým motorem, driverem pro krokové motory a bezdotykový snímač.
- Propojovací kabel.
- Počítač s nainstalovaným softwarem Arduino IDE.

### 6.5.4 Předpokládaná časová dotace:

- Dvě vyučovací hodiny – 2x 45minut.

### 6.5.5 Schéma zapojení



Obrázek 83 Schéma zapojení úlohy 5

### 6.5.6 Vytvořený program

```

const int dirPin = 40; //definice pinu pro rotaci motoru na danou stranu
const int stepPin = 42; //definice pinu pro signal na otočení o jeden krok motoru
const int pocetKroku = 2000; //definice proměnné pro daný počet kroků pro opakování

#define senzor 46 //definice pinu senzoru

void setup()
{
  Serial.begin(9600);
  pinMode(senzor, INPUT); //nastavení pinu senzoru jako vstup pro čtení dat

  pinMode(stepPin, OUTPUT); // nastavení pinu stepPin jako výstup
  pinMode(dirPin, OUTPUT); // nastavení pinu dirPin jako výstup
}

void loop()
{
  digitalWrite(dirPin, HIGH); //nastavení pinu na logickou hodnotu 1 pro opačnou rotaci motoru

  bool value = digitalRead(senzor); //čtení hodnoty senzory 0 nebo 1
  if (value == 0) //podmínka jestli není před senzorem předmět
  {
    Serial.println(value);
  }

  else //podmínka jestli je před senzorem nějaký předmět
  {
    for (int x = 0; x < pocetKroku; x++) //cyklus for pro počet otočení motoru na danou stranu
    {
      digitalWrite(stepPin, HIGH); //nastavení pinu na logickou hodnotu 1
      delayMicroseconds(400); //zpoždění 400ms
      digitalWrite(stepPin, LOW); //nastavení pinu na logickou hodnotu 0
      delayMicroseconds(400); //zpoždění 400ms
    }
  }
}

```

Obrázek 84 Aktivace krokového motoru pomocí bezdotykového snímače



Po nahrání programu by se motor neměl otáčet a zůstat stát, otáčet by se měl až ve chvíli, kdy se přiblížíte k senzoru a senzor detekuje překážku nebo předmět v blízkosti.

## 7 PREZENTACE

Jako podpora výuky pro učitele i žáky byla vytvořena prezentace. V prezentaci je představena a popsána samotná platforma, uvedeny jednotlivé komponenty a schémata jednotlivých úkolů, které jsou programovány na vývojové desce Arduino Mega.



Obrázek 85 Úvodní snímek prezentace

## ZÁVĚR

Cílem této práce bylo vytvořit funkční platformu, které bude sloužit nejenom učitelům a žákům, ale i široké veřejnosti při výuce informatiky a robotiky. Platforma zajišťuje ideální prostředek pro začínající žáky s platformou Arduino a je tak vhodným základem pro tvorbu jednoduchých, základních projektů. Pro vytváření složitějších projektů je možnost rozšířit platformu a další komponenty nebo přímo o další programovatelnou desku Arduino. Dále je zde demonstrováno několik možností, jak jednotlivé komponenty zprovoznit. Podpůrné materiály pro učitele a žáky jsou ve formě dílčích úkolů, které jsou zpracovány. Při návrhu a realizaci byl kladen důraz na možnou opakovatelnost výroby části platformy a náročnosti pořízení jednotlivých dílů a komponentů. Z tohoto důvodu je záměrně sestavena z dílů, které jsou snáze k dostání.

Jednotlivé konstrukční díly jsou vytištěny na 3D tiskárně, takže jejich případná náhrada nebo úprava je jednoduchá a nenáročná.

Tato práce neměla za cíl pokrýt veškeré možnosti Arduino platformy, ale má za úkol posloužit jako podpůrný materiál pro základy výuky a seznámení s touto platformou, pomoci při testování různých komponentů, ověřování funkčnosti a tvorbě ukázkových příkladů.

Vzhledem k stále se zvyšujícím nárokům na začlenění informatiky a rozvíjení informatického myšlení mezi žáky bude vytvořená platforma přínosem pro mnoho začínajících učitelů a žáků, kteří nemají s platformou Arduino rozsáhlé zkušenosti a mají zájem na tom, aby to změnili.

Díky rozsáhlé komunitě platformy Arduino a velkému množství podpůrného a dostupného materiálu věřím, že vytvořená platforma bude dobrým základem, a to nejenom pro výuku informatiky, ale také jako vhodný doplněk do informatických kroužků anebo jako pomůcka nadšencům do platformy Arduino.

**SEZNAM POUŽITÉ LITERATURY**

- [1] Informatika [online]. Copyright © [cit. 18.05.2023]. Dostupné z: <https://revize.edu.cz/files/informatika-2021.pdf>
- [2] Koncept STEM, Národní pedagogický institut České republiky (dříve Národní ústav pro vzdělávání). Národní pedagogický institut České republiky (dříve Národní ústav pro vzdělávání) [online]. Copyright © 2001 [cit. 18.05.2023]. Dostupné z: <https://archiv-nuv.npi.cz/p-kap/koncept-stem.html>
- [3] Klíčové kompetence [online]. Copyright © [cit. 18.05.2023]. Dostupné z: <https://revize.edu.cz/files/k4-rvp-zv-se-zmenami.pdf>
- [4] Seznámení s Arduinem. Úvodní stránka - Bastlírna HWKITCHEN [online]. Dostupné z: [https://bastlirna.hwkitchen.cz/seznameni-s-arduinem/#O\\_Arduinu](https://bastlirna.hwkitchen.cz/seznameni-s-arduinem/#O_Arduinu)
- [5] Arduino klony - Co to je? Koupit či nekoupit? 1 - Bastlírna HWKITCHEN. Úvodní stránka - Bastlírna HWKITCHEN [online]. Dostupné z: <https://bastlirna.hwkitchen.cz/arduino-klony-co-to-je-koupit-ci-nekoupit/>
- [6] Seeeduino V4.2, Arduino UNO klon | Blomberg.cz. Kvalitní elektronické stavebnice za nízké ceny | Blomberg.cz [online]. Copyright © Copyright 2020 [cit. 17.05.2023]. Dostupné z: <https://www.blomberg.cz/p/seeeduino-v4-2-arduino-uno-klon-00512329/>
- [7] Rainbowduino v3.0 | Seeed Studio Wiki. Seeed Studio Wiki [online]. Copyright © 2023 Seeed Studio, Inc. Built with Docusaurus. [cit. 17.05.2023]. Dostupné z: [https://wiki.seeedstudio.com/Rainbowduino\\_v3.0/](https://wiki.seeedstudio.com/Rainbowduino_v3.0/)
- [8] Access denied. Access denied [online]. Dostupné z: <https://store.arduino.cc/collections/shields/products/arduino-4-relays-shield>
- [9] Access denied. Access denied [online]. Dostupné z: <https://store.arduino.cc/collections/shields/products/nano-screw-terminal>
- [10] MCCONNELL, Steve. Dokonalý kód: umění programování a techniky tvorby software. Vyd. 1. Brno: Computer Press, 2005, 894 s. ISBN 802510849x.
- [11] Arduino Integrated Development Environment (IDE) v1 | Arduino Documentation. Arduino Docs | Arduino Documentation [online]. Dostupné z: <https://docs.arduino.cc/software/ide-v1/tutorials/arduino-ide-v1-basics>

- [12] 403 Forbidden. 403 Forbidden [online]. Dostupné z: <https://www.circuitbasics.com/wp-content/uploads/2020/05/Arduino-IDE-1-1024x878.png>
- [13] Scratch for Arduino (S4A). Buy open-source electronic robotic components at best price in India [online]. Copyright © 2021, Component Services, All Rights Reserved [cit. 17.05.2023]. Dostupné z: <https://www.component7.com/blog/s4a-scratch-for-arduino>
- [14] Česká lokalizace software mBlock!. Úvodní stránka - Bastlírna HWKITCHEN [online]. Dostupné z: <https://bastlirna.hwkitchen.cz/ceska-lokalizace-software-mblock/>
- [15] Circuit design Arduino simulator AND | Tinkercad. Tinkercad | From mind to design in minutes [online]. Dostupné z: <https://www.tinkercad.com/things/flHpO-FbJ1CN-arduino-simulator-and>
- [16] Learn how to use Tinkercad | Tinkercad. Tinkercad | From mind to design in minutes [online]. Dostupné z: <https://www.tinkercad.com/learn/circuits>
- [17] WHAT IS FRITZING, HOW DOES IT WORK AND HOW TO USE IT? - Soldered Electronics. Homepage - Soldered Electronics [online]. Dostupné z: <https://soldered.com/learn/what-is-fritzing-how-does-it-work-and-how-to-use-it/>
- [18] Making Mindstorms RCX Work Again : 7 Steps (with Pictures) - Instructables. [online]. Copyright © 2023 Autodesk, Inc. [cit. 18.05.2023]. Dostupné z: <https://www.instructables.com/Making-Mindstorms-RCX-Work-Again/>
- [19] LEGO MINDSTORMS NXT - LEGO MINDSTORMS - Robotické vzdělávání. [online]. Copyright © 2010 [cit. 18.05.2023]. Dostupné z: <https://lego.zcu.cz/web/moduly-systemu-lego-nxt>
- [20] Robotí vynálezce 51515 | MINDSTORMS® | Oficiální obchod LEGO® CZ . [online]. Copyright ©2022 The LEGO Group. Všechna práva vyhrazena. Použitím těchto stránek vyjadřujete svůj souhlas s podmínkami použití. [cit. 18.05.2023]. Dostupné z: <https://www.lego.com/cs-cz/product/robot-inventor-51515>
- [21] Stavebnice a hry LEGO® | KostičkyLega.cz [online]. Copyright © [cit. 18.05.2023]. Dostupné z: <https://www.kostickylega.cz/www/upload/products/images/20200902084417427.png>
- [22] Elegoo UNO R3 Ultimate Starter Kit, 1 sada - 3DJake Česká republika . Přední internetový obchod pro 3D tisk - 3DJake Česká republika [online]. Copyright © 2010 [cit. 18.05.2023]. Dostupné z: <https://www.3djake.cz/elegoo/uno-r3-ultimate->

starter-kit?sai=14034&gclid=EAIaIQobChMI6Iak-  
478\_gIVAgILCh05kQkvEAQYAIABEgLeFvD\_BwE

- [23] Amazon.com. [online]. Copyright © 1996 [cit. 18.05.2023]. Dostupné z: <https://www.amazon.com/KEYESTUDIO-Starter-Electronics-Automation-Education/dp/B08CZ778DJ>
- [24] Overview - Totem Documentation. Totem Documentation [online]. Dostupné z: <https://docs.totemaker.net/mini-lab/>
- [25] Totem Mini Lab - Mini Electronics Laboratory | Totemaker.net. Unique Construction System for Makers | Totemaker.net [online]. Copyright © 2023 Totemaker.net. [cit. 18.05.2023]. Dostupné z: <https://totemaker.net/product/totem-mini-lab/?fbclid=IwAR3OfxXcRfGDo5luNO2NUd48B2c1rORwRhPoV5hNSJ-VaVdluTklXzKmYaKA>
- [26] Young Engineer Kit: Dog | Totemaker.net. Unique Construction System for Makers | Totemaker.net [online]. Copyright © 2023 Totemaker.net. [cit. 18.05.2023]. Dostupné z: <https://totemaker.net/product/young-engineer-kit-dog/>
- [27] Gripper Bot - Smartphone App Controlled Car | Totemaker.net. Unique Construction System for Makers | Totemaker.net [online]. Copyright © 2023 Totemaker.net. [cit. 18.05.2023]. Dostupné z: <https://totemaker.net/product/gripper-bot-smartphone-app-controlled-car/>
- [28] DIY Smartphone Controlled 4WD Car Kit | Totemaker.net. Unique Construction System for Makers | Totemaker.net [online]. Copyright © 2023 Totemaker.net. [cit. 18.05.2023]. Dostupné z: <https://totemaker.net/product/diy-smartphone-controlled-4wd-motor-wheel-kit/>
- [29] [online]. Copyright © [cit. 18.05.2023]. Dostupné z: [https://ae01.alicdn.com/kf/S5bcd279da2af47ccaa02c713d482de38A/LCD-16x4-1604-Character-LCD-Display-Module-LCM-Blue-Blacklight-5V-for-Arduino.jpg\\_Q90.jpg\\_.webp](https://ae01.alicdn.com/kf/S5bcd279da2af47ccaa02c713d482de38A/LCD-16x4-1604-Character-LCD-Display-Module-LCM-Blue-Blacklight-5V-for-Arduino.jpg_Q90.jpg_.webp)
- [30] Maticová klávesnice 4x3 tlačítková - HWKITCHEN. Váš parťák ve světě tvoření | HWKitchen.cz [online]. Copyright © HWKITCHEN, všechna práva vyhrazena [cit. 18.05.2023]. Dostupné z: <https://www.hwkitchen.cz/maticova-klavesnice-4x3-tlacitkova/>

- [31] DRV8825 driver pro krokové motory | LaskaKit. LaskaKit.cz | by Makers for Makers [online]. Dostupné z: <https://www.laskakit.cz/drv8825-driver-pro-krokovye-motory/>
- [32] Arduino Docs | Arduino Documentation [online]. Dostupné z: <https://docs.arduino.cc/static/e534053def4bc0eb97a3b6ba9cf31853/29114/ide-2-overview.png>
- [33] The Autocomplete Feature | Arduino Documentation. Arduino Docs | Arduino Documentation [online]. Dostupné z: <https://docs.arduino.cc/software/ide-v2/tutorials/ide-v2-autocomplete-feature>
- [34] Debugging with the Arduino IDE 2 | Arduino Documentation. Arduino Docs | Arduino Documentation [online]. Dostupné z: <https://docs.arduino.cc/software/ide-v2/tutorials/ide-v2-debugger>

**SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK**

USB	Universální sériová sběrnice
MHz	Jednotka frekvence
PWM	Pulzně šířková modulace
UART	Počítačová sběrnice sloužící k asynchronnímu sériovému přenosu dat
LED	Elektroluminiscenční dioda
RGB	Barevný model
IO	Vstupně výstupní (například pin)
DC	Stejnoseměrné napětí
AC	Střídavé napětí
V	Jednotka napětí
A	Jednotka proudu
IDE	Vývojové prostředí
CPU	Centrální procesorová jednotka
RAM	Operační paměť
OS	Operační systém
DIL	Pouzdro integrovaných obvodů
LCD	Displej z tekutých krystalů
STEM	Přírodní vědy, technika, technologie a matematika
MŠMT	Ministerstvo školství, mládeže a tělovýchovy



**SEZNAM OBRÁZKŮ**

Obrázek 1- Arduino Mini [4].....	15
Obrázek 2 – Arduino Nano [4] .....	15
Obrázek 3 – Arduino Uno [4].....	16
Obrázek 4 - Arduino Leonardo [4].....	17
Obrázek 5- Arduino Mega [4].....	18
Obrázek 6 – Arduino Robot [4] .....	19
Obrázek 7 Seeduino V4.2 [6].....	20
Obrázek 8 Rainbowduino v3.0[7].....	21
Obrázek 9 Arduino 4 Rel0 Schield [8].....	22
Obrázek 10 Shield šroubového terminálu na Arduino Nano [9] .....	23
Obrázek 11 Schield pro připojení motorů a senzorů [9] .....	23
Obrázek 12 Vývojové prostředí Arduino IDE [12].....	24
Obrázek 13 Scratch pro Arduino [13] .....	25
Obrázek 14 Rozhraní Scratch pro Arduino [13] .....	25
Obrázek 15 Vývojové prostředí mBlock [14].....	26
Obrázek 16 AUTODESK Tinkercad [16] .....	27
Obrázek 17 Fritzing [17] .....	28
Obrázek 18 Lego Mindstorms RCX [18] .....	29
Obrázek 19 Lego Mindstorms NXT [19] .....	30
Obrázek 20 Lego Mindstorms NXT periferie [19] .....	30
Obrázek 21 Lego Mindstorms Robotí vynálezce [21] .....	31
Obrázek 22 UNO R3 Ultimate Starter Kit [22] .....	32
Obrázek 23 Stavebnice chytrého domu [23].....	33
Obrázek 24 Rozložení pinů TotemDuino [25].....	35
Obrázek 25 TotemDuino [25].....	36
Obrázek 26 Totem Mini Lab [25].....	37
Obrázek 27 Sada pro mladého inženýra: Pes [26] .....	38
Obrázek 28 Gripper Bot [27].....	39
Obrázek 29 Sada podvozku auta řízená přes telefon pomocí Bluetooth [28].....	40
Obrázek 30 LCD display [29].....	42
Obrázek 31 Zapojení displeje .....	43
Obrázek 32 Model držáku displeje.....	43

Obrázek 33 Držák osazená LCD displejem.....	44
Obrázek 34 Klávesnice.....	44
Obrázek 35 Vnitřní zapojení klávesnice [30] .....	45
Obrázek 36 Model držáku klávesnice .....	45
Obrázek 37 Klávesnice uložená v držáku.....	46
Obrázek 38 Modul 4kanalového relé .....	47
Obrázek 39 Model držáku modulu 4kanalového relé .....	47
Obrázek 40 Relé modul připevněn na platformě pomocí držáku .....	48
Obrázek 41 Schéma zapojení modulu s tlačítky .....	48
Obrázek 42 Modul s tlačítky.....	49
Obrázek 43 Model držáku pro modul s tlačítky.....	49
Obrázek 44 Nepájivé pole .....	50
Obrázek 45 Model držáku nepájivého pole .....	50
Obrázek 46 Nepájivé pole umístěné na platformě pomocí držáku .....	51
Obrázek 47 H můstek .....	52
Obrázek 48 Model držáku H můstku .....	52
Obrázek 49 H můstek umístěn na platformě .....	53
Obrázek 50 Driver DRV8825 pro krokové motory [31] .....	53
Obrázek 51 Zapojení driveru pro krokové motory [31] .....	54
Obrázek 52 Motor s převodovkou na stejnosměrné napětí .....	55
Obrázek 53 Model držáku stejnosměrného motoru .....	55
Obrázek 54 Umístění motoru na platformě pomocí držáku .....	56
Obrázek 55 Krokový motor .....	57
Obrázek 56 Model držáku krokového motoru.....	57
Obrázek 57 Krokový motor umístěný na platformě pomocí držáku.....	58
Obrázek 58 RGB LED kruh.....	59
Obrázek 59 Držák RGB LED kruhu .....	59
Obrázek 60 Držák RGB LED kruhu osazený modulem RGB.....	60
Obrázek 61 Ultrazvukový senzor.....	60
Obrázek 62 Modul pro připojení periférií bez pinů .....	61
Obrázek 63 Kompletní platforma.....	61
Obrázek 64 Deska lamino.....	62
Obrázek 65 Model držáku Arduino Mega.....	63

Obrázek 66 Arduino Mega připevněné na platformě pomocí tištěného držáku .....	63
Obrázek 67 Arduino IDE 2 [32] .....	64
Obrázek 68 Program pro klávesnici .....	68
Obrázek 69 Program pro LCD display .....	69
Obrázek 70 Program pro krokový motor.....	70
Obrázek 71 Program pro ultrazvukový senzor .....	71
Obrázek 72 Program pro relé modul .....	72
Obrázek 73 Program pro RGB LED modul .....	73
Obrázek 74 Schéma zapojení pro úkol 1 .....	75
Obrázek 75 Výpis znaků klávesnice na displej.....	76
Obrázek 76 Schéma zapojení pro úkol 2 .....	77
Obrázek 77 Výpis vzdálenosti na LCD displej.....	78
Obrázek 78 Ovládání relé pomocí klávesnice 1/3.....	81
Obrázek 79 Ovládání relé pomocí klávesnice 2/3.....	82
Obrázek 80 Ovládání relé pomocí klávesnice 3/3.....	83
Obrázek 81 Schéma zapojení úkolu 4 .....	85
Obrázek 82 Ovládání RGB LED modulu.....	85
Obrázek 83 Schéma zapojení úlohy 5 .....	87
Obrázek 84 Aktivace krokového motoru pomocí bezdotykového snímače .....	87
Obrázek 85 Úvodní snímek prezentace .....	89

**SEZNAM TABULEK**

Tabulka 1 Technické specifikace Arduino Nano [1] .....	15
Tabulka 2 Technické specifikace Arduino Uno [1] .....	16
Tabulka 3 Technické údaje Arduino Mega [1] .....	18
Tabulka 4 Technické údaje Rainbowduino [4] .....	21

## SEZNAM PŘÍLOH

Příloha P I: Výuková prezentace o vytvořené platformě

Příloha P II: CD se zdrojovými kódy

