

Embedded systém Arduino ve výuce na střední škole

Bc. Jindřich Baumann

Diplomová práce
2018

 Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2017/2018

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Jindřich Baumann**

Osobní číslo: **A16207**

Studijní program: **N3902 Inženýrská informatika**

Studijní obor: **Učitelství informatiky pro střední školy**

Forma studia: **prezenční**

Téma práce: **Embedded systém Arduino ve výuce na střední škole**

Téma anglicky: **The Arduino Embedded System in Secondary School Education**

Zásady pro vypracování:

1. **Provedte literární rešerši na téma projektové výuky v technických předmětech.**
2. **Vyhodnoťte možnosti využití embedded systému Arduino v tvůrčí práci žáků.**
3. **Navrhněte a s ohledem na rostoucí náročnost zadání připravte sadu úloh pro cílovou skupinu žáků.**
4. **Realizujte zvolené zadání a toto dle možností ověřte v praxi.**
5. **Zhodnoťte výsledky práce a jeho realizace.**

Rozsah diplomové práce:

Rozsah příloh:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **BANZI, Massimo.** Getting started with Arduino. 2nd ed. Farnham: O'Reilly, 2011. ISBN 9781449309879.
2. **BOXALL, John.** Arduino workshop: a hands-on introduction with 65 projects. San Francisco: No Starch Press, 2013. ISBN 978-1-59327-448-1.
3. **DVOŘÁKOVÁ, Markéta.** Projektové vyučování v české škole: vývoj, inspirace, současné problémy. Praha: Karolinum, 2009. ISBN 978-80-246-1620-9.
4. **KRATOCHVÍLOVÁ, Jana.** Teorie a praxe projektové výuky. Brno: Masarykova univerzita, 2006. ISBN 80-210-4142-0.
5. **MARGOLIS, Michael.** Arduino cookbook. 2nd ed. Sebastopol, Calif.: O'Reilly, c2012. ISBN 1449313876.
6. **PINKER, Jiří.** Mikroprocesory a mikropočítače. Praha: BEN - technická literatura, 2004. ISBN .isbn80-730-0110-1.
7. **TOMKOVÁ, Anna, Jitka KAŠOVÁ a Markéta DVOŘÁKOVÁ.** Učíme v projektech. Praha: Portál, 2009. ISBN 978-80-7367-527-1.

Vedoucí diplomové práce:

prof. Mgr. Roman Jašek, Ph.D.

Ústav informatiky a umělé inteligence

Datum zadání diplomové práce:

1. prosince 2017

Termín odevzdání diplomové práce:

16. května 2018

Ve Zlíně dne 11. prosince 2017

doc. Mgr. Milan Adámek, Ph.D.
děkan



prof. Mgr. Roman Jašek, Ph.D.
garant oboru

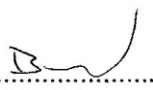
Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové/bakalářské 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/bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou/bakalářskou 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/bakalářskou práci nebo poskytnout licenci k jejímu využití jen přípouští-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/bakalářské 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/bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové/bakalářské 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/bakalářské 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 14. 5. 2018


.....
podpis diplomanta

ABSTRAKT

Cílem diplomové práce je sestavení řady praktických úloh doplněných metodickým popisem demonstrujícím možnosti embedded systémů na platformě Arduino. Teoretická část obsahuje informace o projektové výuce a embedded systému Arduino. Součástí práce jsou návrhy dílčích projektů pro samostatnou tvůrčí práci žáků a studentů, jejichž zájem o technické předměty má být tímto podpořen. Materiály mohou být využity také v technicky zaměřených zájmových kroužcích na 2. stupni základních škol.

Klíčová slova: projekt, projektové vyučování, Arduino, robot, vzdělávání

ABSTRACT

The aim of the diploma thesis is to build a number of practical tasks supplemented by methodical descriptions demonstrating the possibilities of embedded systems on the Arduino platform. The theoretical part contains information of project education and embedded system Arduino. Part of the thesis are partial project proposals for the independent creative work of pupils and students whose interest in technical subjects is to be supported. Materials can also be used in technically-targeted interest groups at second level of primary school.

Keywords: project, project teaching, Arduino, robot, education

Děkuji prof. Mgr. Romanu Jaškovi, Ph.D. za konzultace a možnost si vyzkoušet prezentování části diplomové práce na studentské soutěži.

„Rozvoj vědy a pokrok poznání se stávají stále obtížnější. Na experimentování již nestačí zápalky a sláma.“

-Richard Feynman

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

OBSAH

ÚVOD.....	9
I TEORETICKÁ ČÁST.....	10
1 LITERÁRNÍ REŠERŠE.....	11
2 PROJEKTOVÁ VÝUKA.....	13
2.1 HISTORIE.....	13
2.1.1 Americká pragmatická pedagogika počátkem 20. století.....	13
2.1.2 Projektová výuka za První republiky.....	14
2.1.3 Projektová výuka v současnosti.....	14
2.2 PROJEKT.....	15
2.3 FÁZE PROJEKTU.....	16
2.4 PROJEKTOVÁ METODA.....	16
2.5 PROJEKTOVÉ VYUČOVÁNÍ.....	17
2.6 ZÁSADY PŘI VÝBĚRU UČIVA DLE A. GRÁCE.....	17
2.7 PROJEKTOVÁ VÝUKA VS. TEMATICKÁ VÝUKA.....	17
2.8 PŘEDNOSTI A NEGATIVA PROJEKTOVÉ VÝUKY.....	17
3 EMBEDDED SYSTÉM ARDUINO.....	19
3.1 ARDUINO UNO.....	20
3.2 ZÁKLADNÍ POJMY, VELIČINY A JEDNOTKY.....	21
3.3 POROVNÁNÍ S LEGO MINDSTORMS.....	22
3.4 VIRTUÁLNÍ PROSTŘEDÍ PRO TVORBU PROJEKTŮ TINCECAD.....	25
3.5 MIT APP INVENTOR.....	25
3.6 ARDUINO CLOUD.....	27
4 MOŽNOSTI VYUŽITÍ EMBEDDED SYSTÉMU ARDUINO.....	29
4.1 VÝUKA PROGRAMOVÁNÍ A ALGORITMIZACE.....	29
4.2 PODPORA JINÝCH PŘEDMĚTŮ.....	29
4.3 PROPAGACE TECHNICKÝCH VĚD.....	30
4.4 PRŮMYSLOVÉ POUŽITÍ.....	30
4.5 BUDOUCNOST ARDUINA.....	31
II PRAKTICKÁ ČÁST.....	32
5 PROJEKT.....	33
6 PROGRAMOVACÍ A PROTOTYPOVACÍ PROSTŘEDÍ.....	35
6.1 CÍLE VÝUKY.....	35
6.2 ZADÁNÍ.....	35
6.3 VYPRACOVÁNÍ.....	35
6.3.1 Stažení a instalace.....	35
6.3.2 Připojení knihoven.....	38
6.3.3 TincerCAD.....	38
7 PROGRAMOVACÍ JAZYK WIRING.....	40

7.1	CÍLE VÝUKY	40
7.2	ZADÁNÍ	40
7.3	VYPRACOVÁNÍ	40
8	SEZNÁMENÍ S ARDUINO DESKOU A ZÁKLADNÍMI SOUČÁSTKAMI.....	42
8.1	CÍLE VÝUKY	42
8.2	ZADÁNÍ	42
8.3	VYPRACOVÁNÍ	42
8.4	ÚKOLY K VYŘEŠENÍ.....	44
9	SENZORY.....	45
9.1	CÍLE VÝUKY	45
	ZADÁNÍ 45	
9.2	VYPRACOVÁNÍ	45
9.2.1	Potřebné součástky.....	45
9.2.2	Schéma zapojení.....	46
9.2.3	Program DHT22.....	47
9.2.4	Program HC-SR04	47
9.2.5	Ověření funkčnosti	48
9.3	ÚKOLY K VYŘEŠENÍ	48
10	DC A SERVO MOTOR.....	49
10.1	CÍLE VÝUKY	49
10.2	ZADÁNÍ	49
10.3	VYPRACOVÁNÍ	49
10.3.1	Potřebné součástky.....	49
10.3.2	Schéma zapojení.....	50
10.3.3	Program DC motor.....	50
10.3.4	Program Servo.....	52
10.3.5	Ověření funkčnosti	52
10.4	ÚKOLY K VYŘEŠENÍ	52
11	SPÍNÁNÍ ZÁTĚŽE	53
11.1	CÍLE VÝUKY	53
11.2	ZADÁNÍ	53
11.3	VYPRACOVÁNÍ	53
11.3.1	Schéma zapojení.....	54
11.3.2	Program	54
11.3.3	Ověření funkčnosti	55
11.4	ÚKOLY K VYŘEŠENÍ	55
12	PRÁCE S DISPLEJEM.....	56
12.1	CÍLE VÝUKY	56
12.2	ZADÁNÍ	56
12.3	VYPRACOVÁNÍ	56
12.3.1	Součástky	56
12.3.2	Schéma zapojení.....	56
12.3.3	Program	57
12.3.4	Ověření funkčnosti	57

12.4	ÚKOLY K VYŘEŠENÍ	57
13	RFID.....	59
13.1	CÍLE VÝUKY	59
13.2	ZADÁNÍ	59
13.3	VYPRACOVÁNÍ	59
13.3.1	Součástky	59
13.3.2	Schéma zapojení.....	60
13.3.3	Program	60
13.3.4	Ověření funkčnosti	62
13.4	ÚKOLY K VYŘEŠENÍ	62
14	BEZDRÁTOVÁ KOMUNIKACE POMOCÍ BLUETOOTH	63
14.1	CÍLE VÝUKY	63
14.2	ZADÁNÍ	63
14.3	VYPRACOVÁNÍ	63
14.3.1	Součástky	63
14.3.2	Schéma zapojení.....	64
14.3.3	Program	64
14.3.4	Ověření funkčnosti	65
14.4	ÚKOLY K VYŘEŠENÍ	65
15	NÁVRH A TVORBA APLIKACE PRO OVLÁDÁNÍ ARDUINA POMOCÍ MOBILNÍHO ZAŘÍZENÍ S OS ANDROID	66
15.1	CÍLE VÝUKY	66
15.2	ZADÁNÍ	66
15.3	VYPRACOVÁNÍ	66
15.3.1	Návrh.....	67
15.3.2	Program	67
15.4	ÚKOLY K VYŘEŠENÍ.....	68
16	DÁLKOVĚ OVLÁDANÉ VOZÍTKO (SONDA)	69
16.1	CÍLE VÝUKY	69
16.2	ZADÁNÍ	69
16.3	VYPRACOVÁNÍ	69
16.3.1	Součástky	69
16.3.2	Schéma zapojení.....	71
16.3.3	Program	71
16.4	ÚKOLY K VYŘEŠENÍ	75
	ZÁVĚR	76
	SEZNAM POUŽITÉ LITERATURY	77
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	79
	SEZNAM OBRÁZKŮ.....	80
	SEZNAM TABULEK	82
	SEZNAM PŘÍLOH.....	83

ÚVOD

Trh práce v současné době trápí kromě nedostatku pracovní síly celkově, i dlouhodobý nedostatek technicky vzdělaných lidí. Situaci je třeba řešit tam, kde se nachází příčina problému a to ve školství. Hovoří se o návratu „dílen“ do základních škol, které ale samy o sobě problém nevyřeší. Dle mého názoru je problém především v koncepci výuky, která probíhá formou frontální výuky bez ohledu na zájmy žáků a aktuální problémy společnosti.

Výchovné a vzdělávací cíle je potřeba řešit časově a finančně nenáročnou cestou a tou by mohla být mírně modifikovaná projektová výuka. Výstupem této diplomové práce je sada úloh, které lze vyučovat jak klasicky, tak i formou projektu, i když ne v úplně čisté podobě. Žáci sice budou pracovat samostatně bez zásahu učitele, ale dostanou k dispozici základní informace a materiály tak, aby ušetřili čas při jejich hledání.

K takové výuce lze využít běžné počítače již přítomné ve škole a finančně nenáročné Arduino stavebnice. Projekt se dá vyzkoušet i zcela bezplatně formou online programu dostupného na internetu. Až v případě že se výuka osvědčí, je možné pořídit stavebnice a případně i další příslušenství nad rámec původního projektu.

Projekt lze modifikovat podle požadavků RVP a může do sebe integrovat další témata rezonující v dnešní společnosti. Žáci mohou pracovat ve skupině, zlepšovat komunikační a sociální dovednosti a při tom si osvojit manuální práce, při kterých mohou realizovat zajímavé projekty jako vozítko na solární energii, nebo chytrý skleník. Něco takového je velmi těžké realizovat při běžné výuce.

I. TEORETICKÁ ČÁST

1 LITERÁRNÍ REŠERŠE

Učíme v projektech. Tomková, Anna, Jitka Kašová a Markéta Dvořáková.

Kniha se sice primárně zabývá projektovou výukou na základní škole, ale zmínění poznatky se dají přenést i na starší žáky. Teorii o projektové výuce doplňuje srovnání s tematickou výukou. Velká část knihy je věnována praktickým příkladům projektové výuky, které obsahují všechny důležité části projektu včetně scénáře. Kniha je vhodnější pro čerpání inspirace z praktických ukázek, než ke studiu teorie.

Teorie a praxe projektové výuky. Jana Kratochvílová.

První kapitola je věnována proměnám vzdělávacího systému. Druhá kapitola, se podrobněji zabývá historií projektu ve výuce. Třetí kapitola obsahuje teoretické poznatky o projektu včetně jeho předností a nedostatků. Čtvrtá kapitola, která má pouze dvě stránky, zmiňuje tematickou výuku. Pátá kapitola pojednává o profesní přípravě studentů učitelství, šestá obsahuje příklady projektů a sedmá empirický výzkum.

Projektové vyučování v české škole. Markéta Dvořáková

Zhruba polovina knihy je věnována historii projektové výuky v Americe a u nás. Zbytek se věnuje projektové výuce v současnosti a její integrace do vyučování včetně ukázek.

Arduino cookbook. Michael Margolis.

Kniha obsahuje vše, od instalace prostředí až po pokročilé úpravy knihoven. Jednotlivé kapitoly jsou logicky řazeny a jsou zpracované způsobem „problém-řešení-diskuze“. Úlohy obsahují přehledné schéma i kód. Trochu nelogicky je kapitola o elektronických součástkách umístěna až na konci. Autor zřejmě předpokládá alespoň základní znalost elektroniky.

Arduino workshop: a hands-on introduction with 65 projects. John Boxall.

Kniha se věnuje v podstatě tomu stejnému co kniha předešlá, jen neobsahuje složitější témata a jednotlivé kapitoly obsahují i informace o použitých elektronických součástkách. Je tedy vhodnější pro začátečníky, kteří se chtějí i něco naučit a ne jen zapojovat podle návodu.

Exploring Arduino: tools and techniques for engineering wizardry. Jeremy Blum.

Dle mého názoru je toto jedna z nejlepších knih o Arduinu, ze kterých lze čerpat materiály pro projektovou výuku. Jak sám autor zmiňuje, nejedná se o „kuchařku“, kterou obvykle připomínají jiné knihy, nebo tutoriály na internetu. Úlohy jsou komplexní a je v nich vše srozumitelně vyjasněno. Kniha je vhodná i pro lidi, kteří nemají ponětí o tom, co je to napětí, nebo elektrický odpor. Čtenář se tedy dozví informace z několika oblastí a získá pevný teoretický základ, na kterém může dále stavět.

Hradla, volty, jednočipy. Martin Malý

Kniha seznamuje čtenáře se základy elektrotechniky, popisuje základní součástky, polovodiče principy komunikace a paměti. Dále plynule přechází na projekty s využitím Arduina. Velkou výhodou je, že kniha je zdarma ke stažení na internetu!

Průvodce světem Arduina. Zbyšek Voda

Tato česká kniha je v podstatě soubor projektů ze stránek www.arduino.cz. Vhodná pro ty co nechtějí pracovat s literaturou v angličtině. Kniha je volně ke stažení na internetu.

www.arduino.cc

Oficiální stránky projektu Arduino. Na stránkách lze nakoupit Arduino zařízení, je zde rozsáhlá příručka pro programování a také návody na mnoho projektů. Důležitá je i sekce „Education“ kde jsou veškeré informace vztahující se k výuce s Arduinem.

www.arduino.cz

Česká obdoba oficiálních Arduino stránek. Lze zde najít návody obsažené v knize Průvodce světem Arduina. Autoři také provozují e-shop, kde je možné nakoupit Arduino a příslušenství.

www.zive.cz

Webový seriál „Pojďme programovat elektroniku“ na stránkách zive.cz je skvělou inspirací pro projekty nejen s Arduinem. Vše je jednoduše vysvětleno včetně odkazů, kde koupit použité součástky.

2 PROJEKTOVÁ VÝUKA

2.1 Historie

2.1.1 Americká pragmatická pedagogika počátkem 20. století

Hlavním představitelem americké pragmatické pedagogiky byl John Dewey. Byl kladen důraz na motorické projevy dětí, nonverbální komunikaci, modelování, kreslení, malování, psaní hudbu a řeč. Odmítán byl přehnaný formalismus, striktní disciplína, pasivní učení a tupý dril. Učení bylo chápáno jako proces aktivního získávání zkušeností. Žák získával poznatky jako výsledky své samostatné práce, jejíž náplní bylo samostatné řešení problémů. Tato práce vycházela ze zájmů dítěte, začínala od jednoduchých věcí a postupovala od praxe k teorii. Cílem byla výroba nějakého díla, prostřednictvím kterého dítě získá nové poznatky. Prostředkem dosažení cíle je řešení problémů praktickou činností, která vede k vlastní zkušenosti. [1]

Projektovou metodu jako první vymezil William Heard. Dítě má přirozenou touhu učit se, samo studuje, objevuje a řeší problémy. Projekty jsou smysluplné a spjaté se životem. Mají nějaký praktický cíl. Cílem projektové metody bylo odstranit strach a nenávisť žáků ke škole. Heard definoval čtyři fáze projektu: záměr, plán, provedení a posouzení. [1]

Pojem projekt definovali různí autoři mírně odlišně, ale shodli se na tom, že důležitý je postoj žáka k projektu. Projekt se stává jeho záměrem a žák za něj přebírá zodpovědnost. Projekt je prostředkem k dosažení cíle. Shoda panovala i na tom, že žáci by neměli mít úplnou volnost v provedení a výběru projektu. Neshody panovaly převážně v tom, zda zachovat klasickou výuku a paralelně s ní výuku projektovou, nebo zda klasické vyučování úplně zrušit, protože projekt i tak byl propojením faktů z více učebních předmětů. [1]

Projekty se dělily do dvou skupin, na projekty spontánní- volné projekty a uložené- zadané učitelem. [1]

V projektové výuce se začala měnit i role učitele, který musel změnit svou přípravu i vztah k žákům. Jeho hlavní činností bylo poznávání žáků a jejich zájmů a nabídka nových činností a pomůcek. [1]

2.1.2 Projektová výuka za První republiky

Projektová výuka byla reakcí na kritiku herbartovské školy, které nedovedla probudit v žácích zájem a byla založená na pamětním učení. Škola byla považována za nepřirozenou, bez vztahu ke skutečnému životu. [1]

Na rozdíl od americké progresivní pedagogiky se ta česká snažila být kompromisem mezi klasickou a projektovou výukou. Projektové vyučování nikdy nenahradilo systematické vyučování a učební předměty. [1]

Hlavním představitelem české progresivní pedagogiky byl S. Vrána. Ten vycházel z Američana W. S. Kilpatricka a stavěl na stejném základu, že žák přebírá plnou zodpovědnost za projekt a projekt je tedy „podnik žáka“. Vrána připouštěl, že součástí projektové výuky může být i teoretická činnost a memorování. [1]

Byly definovány jednotlivé etapy projektu:

- První etapa- motivace a volba cíl
- Druhá etapa- plánování a hledání prostředků
- Třetí etapa- provedení plánu
- Čtvrtá etapa- využití a zhodnocení projektu [1]

Prudce se zvýšili nároky na učitele. Ten musel být pohotový a mít rozsáhlé vědomosti napříč všemi oblastmi výuky. [1]

Na rozdíl od Ameriky sloužilo u nás projektové vyučování jako metoda samoučení, ne jako zlepšení sociálních dovedností a výchova k demokracii. [1]

2.1.3 Projektová výuka v současnosti

Projektová výuka je dnes brána jako jeden z účinných prostředků naplnění cílů současné školy. Má za cíl připravit žáka na celoživotní učení a odstranit nedostatky tradičního vyučování jako je izolovanost a roztržitost vědění, či strnulost školní práce. Není cílem úplně nahradit tradiční vyučování, ale být jeho komplementárním doplňkem. Žák nezískává hotové poznatky, ale je nucen si informace aktivně vyhledávat. Projektové vyučování umožňuje propojení poznatků z různých předmětů. [1]

J. Skalková vidí současné projektové vyučování jako prostředek rozvíjení myšlenkových struktur, spojení teorie s praxí a školy a života. [1]

A. Tomková považuje projektové vyučování za metodu, při které je uplatněno několik dílčích metod a forem práce a vede k samostatné práci žáků. [1]

2.2 Projekt

Projekt je charakterizován jako vlastní podnik žáků. Musí mít stanoven nějaký cíl a smysluplný závěr. Žákovi projekt vyhovuje a vnitřně se s ním identifikuje. Přebírá zodpovědnost za realizaci a uspokojivé zakončení, při kterém je projekt zhodnocen a to buď klasicky, nebo sebehodnocením. [2]

Na projekt můžeme nahlížet z více úhlů pohledu:

- Metoda vyučování
- Organizační forma vyučování
- Způsob zpracování obsahu
- Způsob koncentrace učiva
- Typ vzdělávací strategie
- Typ učebního úkolu [1]



Obr. 1- schéma projektu

2.3 Fáze projektu

Plánování

Definuje se smysl projektu, jeho cíl a časový harmonogram. Důležitou fází plánování je motivace žáka k realizaci projektu. [2]

Realizace

Při realizaci žáci postupují dle plánu, případně je učitel usměřňuje, pokud se od plánu odkloní. Při realizaci hraje velkou roli shromažďování a třídění informací. [2]

Prezentace výstupu

Prezentace může mít mnoho podob. Vždy záleží na formě zpracovávaného projektu. Prezentují se jak výstupy diskuzní, či písemné, tak i finální výrobky. Presentovat je možné před třídou, pedagogy, rodiči, případně i na soutěžích, nebo při jiných příležitostech. [2]

Hodnocení

Projekt by měl být hodnocen jako celek a to jak ze strany žáků, tak i učitele. Hodnocení vychází z předem nastavených kritérií, na kterých se ideálně shodli všechny zainteresované strany. Z hodnocení by mělo vyplynout ponaučení do dalších projektů. [2]

2.4 Projektová metoda

System navzájem propojených činností, ve kterých figurují žáci i učitel. Základem je tvořivá práce žáků, založená na praktické práci a experimentech, které by měly odrážet realitu běžného života. Komplexnost úloh vyžaduje zapojení různých druhů metod a způsobů práce. [2]

Charakteristické znaky projektové metody jako systému činností:

1. Organizovaná učební činnost směřující k určitému cíli.
2. Činnost, která nemůže být dopředu zcela jasně krok za krokem naplánována.
3. Činnost vyžadující aktivitu žáka a jeho samostatnost.
4. Činnost tvořivá a reagující na změny v průběhu projektu.
5. Činnost převážně vnitřně řízená- auto-regulovaná.
6. Činnost teoretická i praktická rozvíjející celou osobnost žáka k učení a přispívá k rozvoji jeho sebepojetí. [2]

2.5 Projektové vyučování

Projektové vyučování lze obecně definovat jako specifický model vyučování, kdy žák přebírá zodpovědnost za svůj úkol a jeho realizaci. [1]

2.6 Zásady při výběru učiva dle A. Gráce

Učební látka:

- Se vztahuje k osobnímu, nebo obecnému dobru
- Obsahuje ty zkušenosti, které může žák v životě opravdu užít.
- Každý poznatek má možnost aplikace v osobním, obchodním, hospodářském životě
- Rozvíjí maximálně myšlení žáka, ale je srozumitelná
- Umožňuje získání správných návyků v osvojování nových poznatků a tvoří základ dalšího možného studia
- Využívá individuální schopnosti a zájmy žáka [1]

2.7 Projektová výuka vs. tematická výuka

Projektová výuka, nebo také projektové vyučování, je založena na projektu, za který žáci přebírají plnou zodpovědnost. Součástí projektové výuky může být i tematická výuka, Pomocí které mohou žáci získat potřebné informace.

Tematická výuka se zabývá sběrem informací, jejich zpracováním, propojením a uvědoměním si souvislostí mezi nimi. Na rozdíl od projektu nemá stanovený žádný konkrétní cíl a zabývá se věcmi více ze široka. Velmi často se tematická výuka stává základem pro projekt. [2]

2.8 Přednosti a negativa projektové výuky

Přednosti

- Přirozenost námětu práce
- Učení pro poznání a porozumění
- Spontánní zájem žáka
- Vlastní úsilí žáka
- Osobní odpovědnost
- Vyrvalost v překonávání překážek

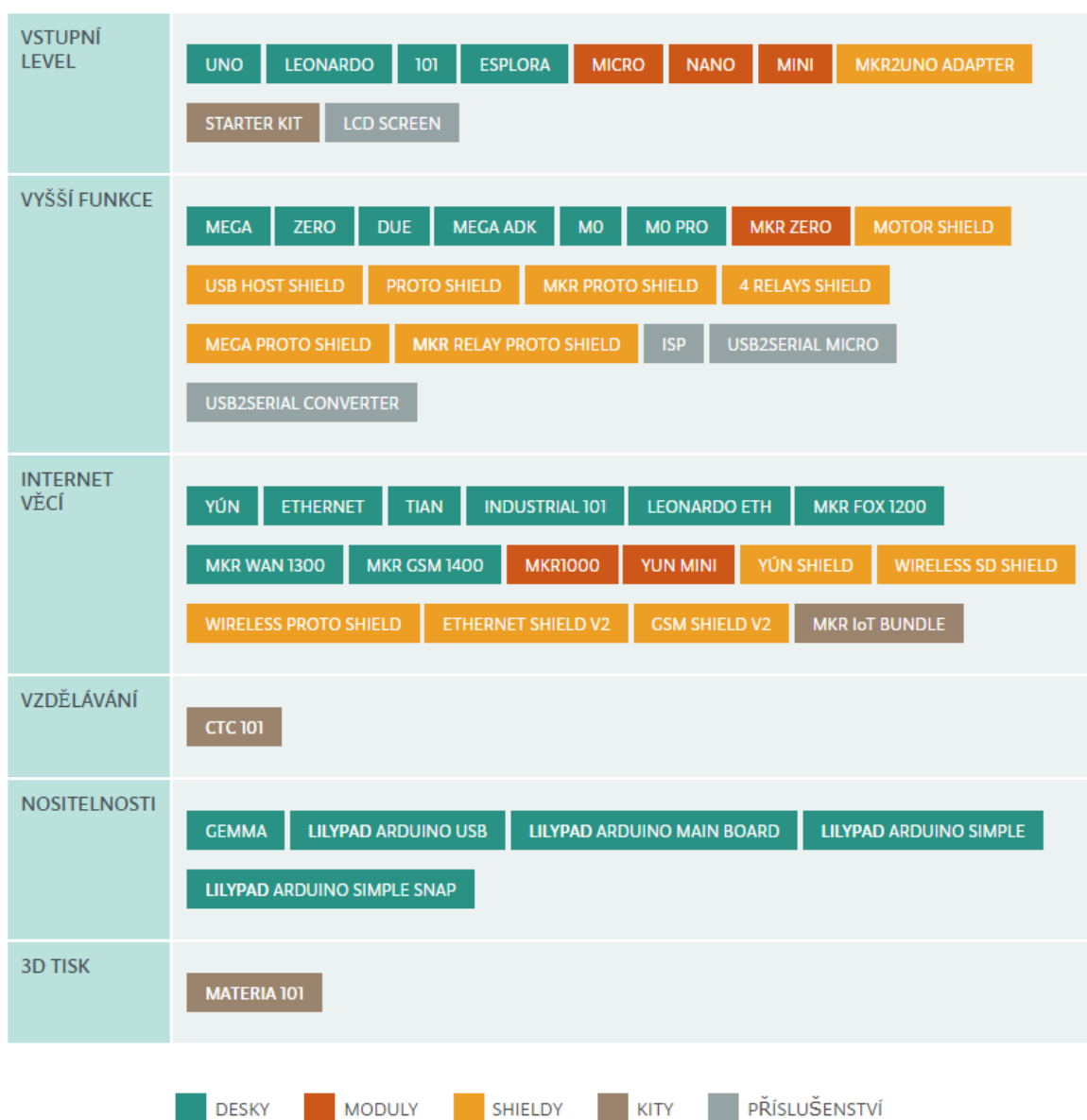
- Snaha splnit cíl
- Uspokojení z vlastní práce
- Využití získaných poznatků
- Získané poznatky nejsou izolované
- Plní vzdělávací cíle i cíle výchovné
- Rozvíjí zodpovědnost, vůli a vnitřní motivaci [1]

Negativa

- Narušení osnov a struktury předmětů
- Nemůže nahradit systematické předmětné vyučování
- Spontánní problémy se vyskytují jen zřídka
- Náročné pro učitele
- Vyhovuje především nadaným žákům
- Nesystematičnost
- Náročné na organizaci
- Neposkytuje dostatečnou teoretický základ [1]

3 EMBEDDED SYSTÉM ARDUINO

Arduino je open source platforma využívající jednoduchý hardware a software přizpůsobený začátečníkům. Arduino vzniklo v Ivrea Interaction Design Institute odkud se rozšířilo dál a vznikla velká komunita lidí, tvořících projekty. Díky tomu, že je hardware i software open source, je možné jej přizpůsobovat a upravovat podle potřeb. Velkou výhodou je velké množství cenově dostupných klonů jak Arduina, tak i příslušenství. Téměř cokoliv lze získat v řádech desítek korun a to i včetně samotné desky. Dle potřeb uživatelů se vyvíjely různé druhy desek viz. obrázek níže. [4]



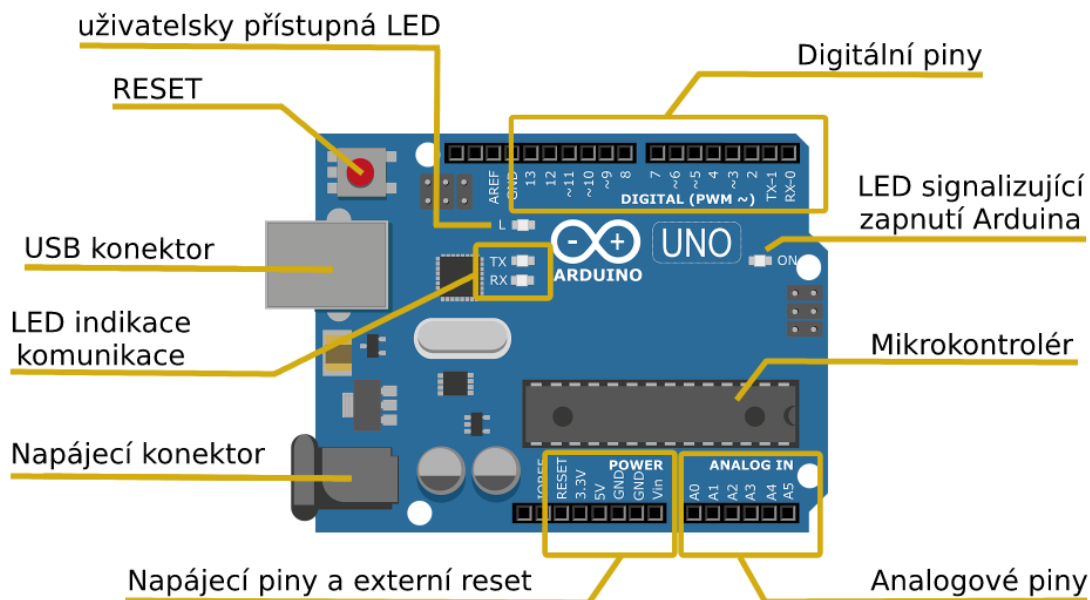
Obr. 2- Arduino produkty [20]

Pro první seznámení a malé, až středně velké projekty, je vhodné Arduino UNO, kterému bude věnován zbytek diplomové práce.

3.1 Arduino UNO

Na čelní straně desky se nachází USB konektor, sloužící pro propojení desky s počítačem. Zároveň může sloužit jako zdroj napájení. Vedle něj se nachází napájecí konektor a tlačítko sloužící pro reset mikrokontroléru, který je mozkiem celého zařízení. Zapnutí Arduina signalizuje LED s nápisem ON, další dvě LED TX a Rx signalizují probíhající komunikaci desky s periferiemi a čtvrtá, poslední dioda je přístupná pro potřeby uživatele a může plnit různé funkce. První blok pinů tvoří napájecí konektory s konektorem pro reset. Vedle se nacházejí analogové piny, sloužící pro měření analogových signálů. Piny A4 a A5 mohou rovněž sloužit pro odesílání dat. Další blok je vyhrazen pro digitální piny. Jak už z názvu vyplývá, slouží pouze pro přijímání, či odesílání digitálních dat. Piny Tx a Rx mohou sloužit i pro sériovou komunikaci. Piny označené „~“ mohou být použity pro PWM.

Různé klony se mohou od originálního Arduina lehce lišit v návrhu desky, rozmístění klíčových prvků je ale vždy stejné.



Obr. 3- Arduino UNO

Tab. 1- Parametry Arduino desky [5]

Mikrokontroler	ATmega328P
Operační napětí	5V
Doporučené vstupní napětí	7-12V
Hraniční vstupní napětí	6-20V
Digitální vstupně-výstupní piny	14
PWM digitální vstupně-výstupní piny	6
Analogové vstupní piny	6
DC proud na pin	20mA
DC proud pro 3.3V pin	50mA
Flash paměť	32KB
SRAM	2KB
EEPROM	1KB
Frekvence	16MHz
Velikost	68,6 x 53,4mm

3.2 Základní pojmy, veličiny a jednotky

Elektrický proud- elektrický proud udává, kolik projde vodičem náboje, za jednotku času. Značí se „I“ a jednotkou je ampér.

„Ampér je stálý elektrický proud, který při průchodu dvěma přímými rovnoběžnými nekonečnědlouhými vodiči zanedbatelného kruhového průřezu umístěnými ve vakuu ve vzájemné vzdálenosti 1 metr vyvolá mezi nimi stálou sílu o velikosti 2×10^{-7} newtonu na 1 metr délky vodiče.“ [3]

Elektrické napětí- elektrické napětí je potenciál mezi dvěma body a představuje práci vynaloženou na přesun jednotkového kladného náboje z místa nižšího potenciálu do místa s vyšším potenciálem. Značí se „U“ a jednotkou je volt. [3]

Elektrický výkon- výkon vyjadřuje množství práce vykonanou za jednotku času. Součinem napětí a proudu získáme elektrický výkon. Ten se značí jako „P“ a jednotkou je watt. [3]

Elektrický odpor- elektrický odpor vyjadřuje schopnost vodiče vést elektrický proud. Protékajícímu proudu vodičem klade odpor krystalová mřížka, na které ztrácejí elektrony svou energii. Odlišné materiály mají různý elektrický odpor. [3]

PWM- pulzně šířková modulace je způsob, jakým lze s digitálního zdroje získat analogový výstup. U digitálního signálu známe pouze stavy zapnuto/ vypnuto. Pomocí změny času mezi stavem „zapnuto“ a „vypnuto“, nebo 1 a 0, dosáhneme požadované velikosti analogového signálu. Stejně jako u notebooku můžeme měnit jas obrazovky pomocí PWM, tak stejně tak můžeme měnit u Arduina jas LED, nebo rychlost motorů. [8]

Analogový signál- analogový signál je spojitý signál, tvořený spojitou funkcí závislou na čase. U Arduina, se s analogových signálem setkáme například při odečítání hodnot z potenciometru. [9]

Digitální signál- digitální signál je realizován pomocí dvou stavů, typicky 1 a 0, nebo ANO a NE, které reprezentují určitou hodnotu napětí. Digitální signál se z analogového získává pomocí A/D převodníku, metodou vzorkování a kvantování. [10][11]

Bit a baud- bit je základní jednotka informace. Značí se „b“, případně „bit“ a nabývá dvou stavů „0“ a „1“. Baud je jednotka modulační rychlosti, značí se „Bd“ a v případě sériového přenosu na Arduinu platí převod $1 \text{ b} = 1 \text{ Bd}$. [12][13]

3.3 Porovnání s LEGO Mindstorms

Stavebnice LEGO Mindstorms už je poměrně etablovaná na půdě škol a universit. Nabízí vyvážený mix vlastností, které se hodí do výuky. Všechny díly jsou unifikované a je možné z nich skládat i poměrně složité struktury. Řídící jednotka je zapouzdřena v obalu s montážními otvory a stává se tedy součástí konstrukce. Jako napájení je použita baterie v řídicí jednotce. K nabíjení složí nabíječka se standardizovaným konektorem. K programování slouží aplikace pro desktop. Dostupná je i mobilní aplikace. Vlastnosti této stavebnice lze považovat jak za výhodu, tak i nevýhodu. Práce se stavebnicí je poměrně jednoduchá, ale nabízí velmi malé možnosti rozšíření. Dokupování dalších konstrukčních prvků je i finančně náročné.

Arduino je téměř přesným opakem LEGO stavebnice. Standardizovaná je pouze samotná deska a její parametry. Připojit se k ní dá téměř cokoliv a jakkoliv. Součástí Arduina nejsou žádné konstrukční prvky, ty je potřeba koupit, nebo vyrobit a to včetně obalu samotného Arduina. Proto se typicky Arduino připojuje už k hotovým výrobkům jako staré autíčka na dálkové ovládání, konstrukce dronů a další elektronice. Jako napájení lze použít téměř cokoliv, co poskytne stejnosměrné napětí o velikosti 6-20V. Arduino je na rozdíl od LEGO stavebnice náchylné na špatné zapojení, tudíž není vhodné pro žáky, kteří nejsou seznámeni alespoň se základy elektroniky a elektrotechniky. LEGO lze dát i žákům prvního stupně základní školy a nestane se, že by něco špatně zapojili.

Ideální situace by nastala, pokud by na základní škole žáci začali pracovat s LEGO Mindstorms a plynule přešli na střední škole na Arduino v jakékoliv podobě. Arduino není pouze jedna specifická deska, ale skupina desek, kde každá má jiné použití. Výuka se tak dá přizpůsobovat pro různé typy škol. V LEGU vidím dobrou možnost jak ztraktivnit mladším studentům výuku, Arduino pak dává větší prostor pro vlastní kreativitu a tvořivost.

Velkou výhodou Arduina je cena viz tabulka a to jak z pohledu školy, tak žáků, kteří si mohou koupit stavebnici domů. Cena je sice stále poměrně vysoká, ale dá se srazit na zlomek původní ceny nákupem klonů z čínských e-shopů, které jsou dnes velmi populární.

Další výhodou může být relativně malá velikost Arduina, které se jako takové společně se základními součástkami vleze do malé krabičky, nebo sáčku. Krabice s LEGO Mindstorms je poměrně velká a patnáct takových přepravek zabere hodně místa, s čímž mlhou mít některé školy problém.

Tab. 2- Porovnání Arduino vs. LEGO Mindstorms

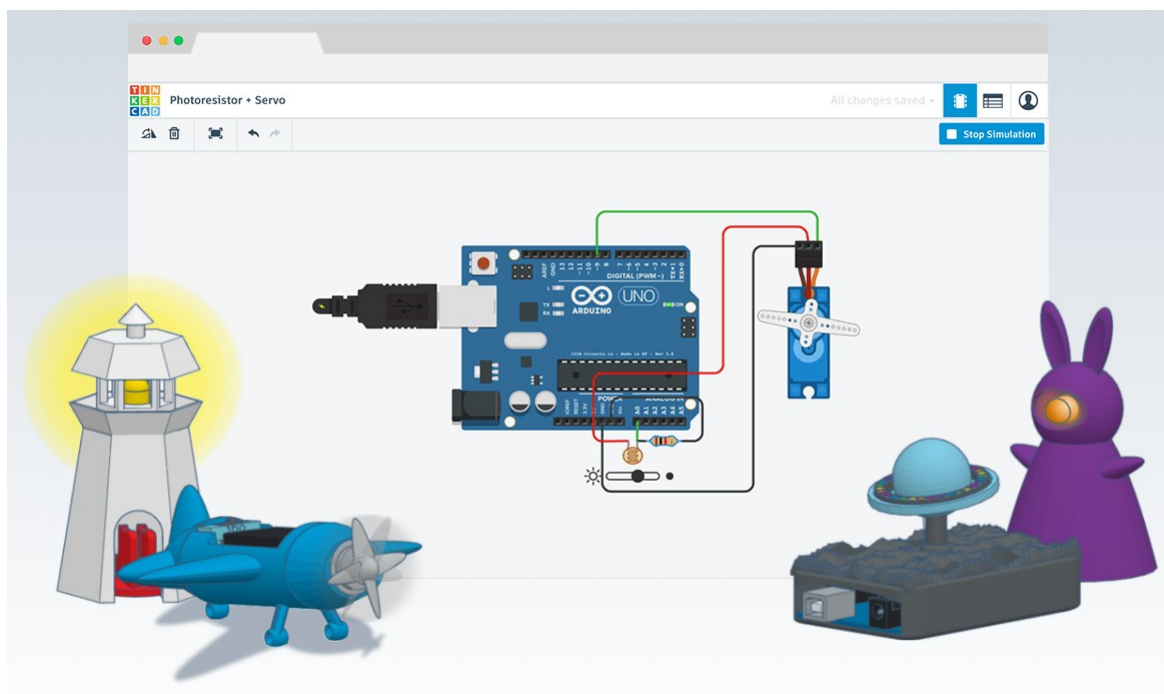
Název	Cena	Rozšiřitelnost	Obtížnost
Arduino starter kit	2600 Kč*	Vysoká	Vysoká
LEGO Mindstorms EV3	10499 Kč	Nízká	Střední

*Z oficiálních stránek je možné stavebnici objednat za 79,9 €, při koupi alespoň 12 balení za cenu 55,99 €/kus

3.4 Virtuální prostředí pro tvorbu projektů Tincercad

Tincercad je online aplikace pro tvorbu 3D objektů a návrh elektronických obvodů. Pro plnohodnotnou funkci není potřeba instalovat žádná programy ani ovladače. Navrhnuté 3D modely je možné vytisknout na 3D tiskárně a potom je oživit pomocí elektroniky. Skvělou funkcí je výuková část aplikace, kde je možné vytvořit vlastní třídu, pozvat do ní své studenty a mít přehled o jejich práci.

Část věnovaná elektronickým obvodům není původní součástí Tincercadu, ale migrovala z projektu circuits.io. Pokud žáci z nějakého důvodu nemohou pracovat se skutečnými součástkami, mohou si své projekty zkusit v tomto virtuálním prostředí.

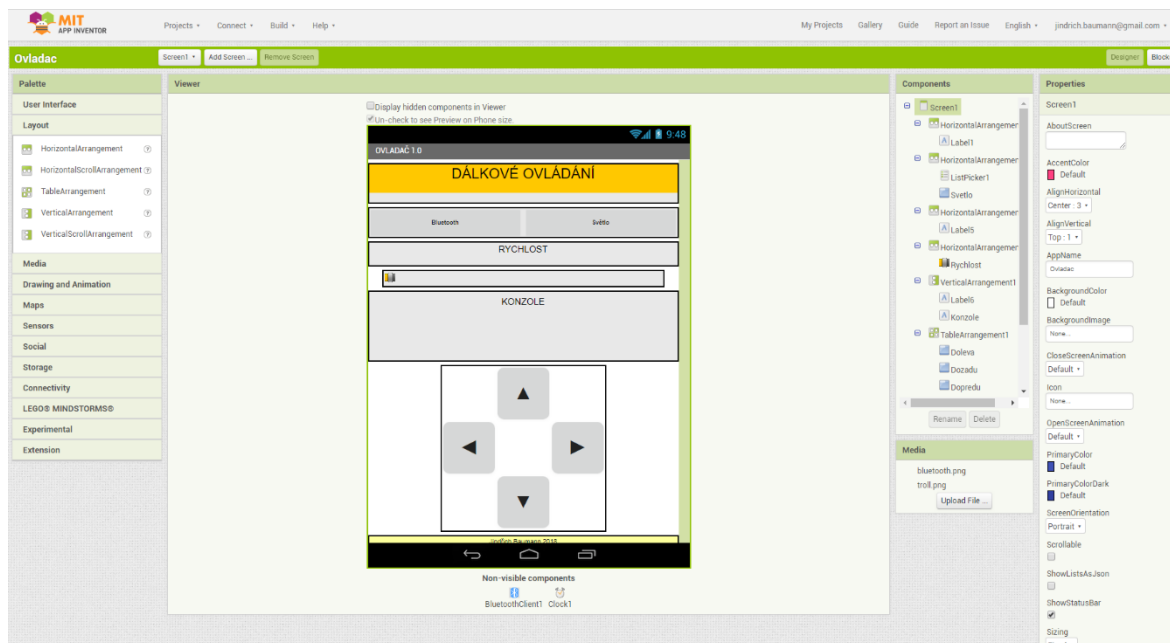


Obr. 6- Tincercad [14]

Kromě základních součástek aplikace obsahuje i Arduino a jeho příslušenství. Obvody se dají programovat i simulovat. Pokud si tedy nejsou žáci jistí zapojením, mohou si správnost otestovat v aplikaci bez následků na fyzickém hardwaru. Pro úplné začátečníky jsou připraveny již hotové projekty, stejně tak se žáci mohou podělit se svými projekty a sdílet je s ostatními.

3.5 MIT App Inventor

MIT App Inventor je programovací prostředí založené na blocích, ve kterém zvládne vytvářet aplikace i člověk bez znalosti programovacích jazyků.

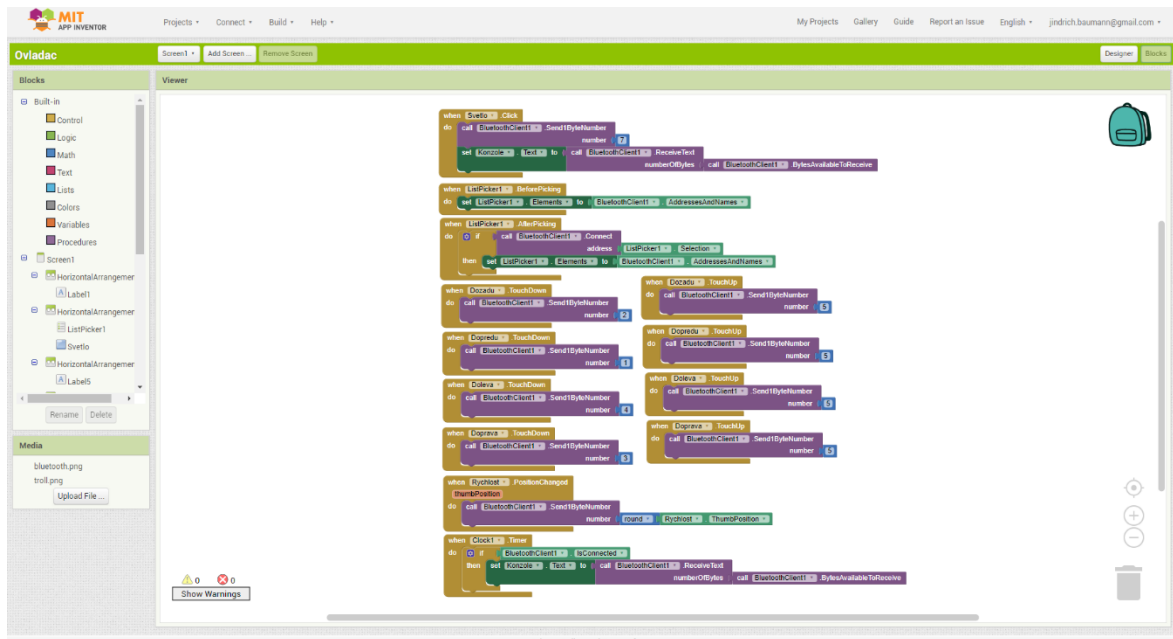


Obr. 7- prostředí MIT App Inventor

Prostředí se skládá ze dvou základních částí- Designer a Blocks. V části designer si žáci mohou navrhnout rozmístění a design jednotlivých částí. Díky real-time propojení s Android zařízeními jsou provedené změny ihned promítnuty do vzhledu aplikace na mobilním zařízení. Část Designer je rozdělena na čtyři části. Úplně nalevo jsou jednotlivé prvky, které lze v aplikaci použít, uprostřed je náhled výsledné aplikace, třetí část zleva zobrazuje použité prvky a poslední část úplně vpravo nabízí jejich nastavení.

Po přepnutí do části Blocks můžou žáci naprogramovat svou aplikaci pomocí předdefinovaných bloků, při čemž různé prvky použité v aplikaci nabízejí různé druhy bloků. Díky mnoha návodům dostupných na internetu není problém vytvořit jednoduchou aplikaci v řádech desítek minut.

Po dokončení je možné aplikaci stáhnout, nebo nainstalovat do mobilního zařízení pomocí QR kódu.



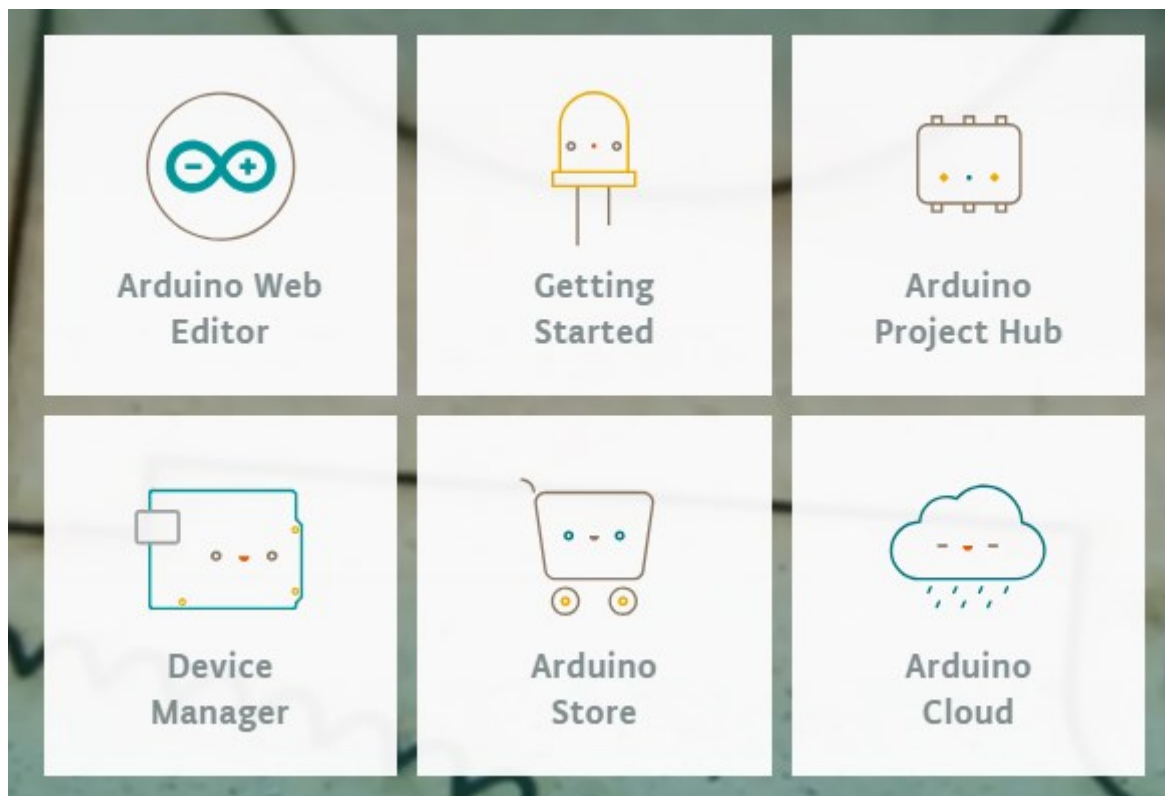
Obr. 8- programovací bloky aplikace MIT App Inventor

3.6 Arduino Cloud

Arduino cloud nabízí řadu nástrojů, které mohou být žákům užitečné. Hlavním nástrojem je Arduino Web Editor, který slouží k programování Arduino softwaru. Ačkoli se jedná o online nástroj běžící v cloudu, není zcela nezávislý na hardwaru. K plnohodnotnému použití je potřeba nainstalovat klienta, který slouží jako most mezi webovou aplikací a Arduinem, fyzicky připojeným k počítači.

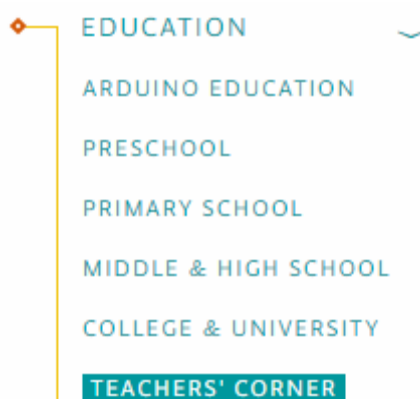
Na Arduino Project Hub je možné najít velké množství projektů, kterými se lze inspirovat, nebo je rovnou použít při výuce. Projekty lze třídit a jedním z kritérií je i obtížnost projektu.

Sekce Getting Started, Device Manager a Arduino cloud složí k přidání a správě zařízení do „cloudu“. Taková zařízení lze pak vzdáleně ovládat a diagnostikovat. V dnešní době chytrých domácností je to vynikající nástroj a inspirace pro žáky.



Obr. 9- Arduino cloud

Arduino Store je jak již název napovídá obchod, kde lze zakoupit Arduino produkty a příslušenství. Z pohledu výuky je zajímavá sekce „Education“, kde lze nalézt produkty rozdělené na základě stupně školy.



Obr. 10- Arduino education

4 MOŽNOSTI VYUŽITÍ EMBEDDED SYSTÉMU ARDUINO

4.1 Výuka programování a algoritmizace

Výuka na školách v současnosti probíhá formou frontální výuky, kdy učitel programovací jazyk vysvětluje na sérii ukázek, které si žáci opíšou a vyzkouší, bez jakéhokoliv kontextu a spojení s praxí. Pokud už výuka probíhá na nějakém hardwaru, bývá to mikrokontroler 8051 a podobně. Sice si žáci prakticky něco vyzkouší, ale možnosti takovýchto projektů jsou značně omezeny. Programovací jazyk Assembler také spíše studenty odradí, než aby výuku programování ztraktivnil.

Populární platforma Arduino je ideálním nástrojem pro výuku. Díky rozsáhlé komunitě a velkému množství atraktivních projektů dokáže zaujmout i znučené studenty. Odpadá prvotní zděšení z náročného programování. Programovací jazyk Wiring je jednoduchý na pochopení i zápis. Lze čerpat z oficiálních materiálů a není potřeba kupovat obsáhlé publikace a příručky. Není potřeba dodržovat pořadí probírané látky, studenti si doplní potřebné informace dle aktuální potřeby. Projekty lze dělit od těch nejjednodušších až po složité a komplexní výrobky, tedy každý žák si může vybrat dle svých schopností a na konci okusí radost ze splnění cíle.

Při tvorbě projektů pochopitelně vznikají problémy, které ale nejsou na škodu, naopak jsou základem projektové výuky. Žáci jsou tak nuceni o problémech přemýšlet a plánovat další postup. Chyby nejsou pouze abstraktní, ale skutečně se projeví a mnohdy není možné je odstraňovat stylem pokus-omyl.

4.2 Podpora jiných předmětů

Z definice projektové výuky vyplývá, že se jedná o propojení poznatků napříč předměty. Některé předměty takto nelze zapojit, ale na druhou stranu projekty na Arduinu dokáží pokrýt jiné aspekty rámcového vzdělávacího plánu, které se jiným způsobem těžko naplní. Žák se sám vzdělává, což má pozitivní vliv napříč celou výukou. Pracuje v týmu, učí se třídit a zpracovávat informace a přebírá zodpovědnost za svou práci.

Na Arduino projekty lze navázat látku z jiných předmětů jako je fyzika, matematika, informatika, případně další. Záleží především na učiteli, jak dokáže integrovat poznatky z jiných předmětů.

4.3 Propagace technických věd

Z hlediska atraktivity je na tom Arduino velmi dobře. Nikdy nedokáže zaujmout 100% všech žáků, ale je na tom jednoznačně líp, než klasická výuka programování formou opísování z tabule, nebo vyrábění budky pro ptáky. Projekty na Arduinu se často objevují na soutěžích, což dokazuje jejich popularitu.

Je potřeba zmínit, že jsou i zajímavější prostředky jak přilákat mladé lidi k technickým oborům, například 3D tiskárny, drony a podobně. Ty jsou ale cenově pro mnoho škol nedostupné a v mnoha případech i velmi náročné na výuku, ať už z pohledu učitele, tak i žáků.



Obr. 11- Braccio robot [16]

Důležitým aspektem je propojení teorie a praxe, kdy studenti vidí možnosti skutečného využití. Výuka není omezená na blikání LED, nebo točení krokového motorku. Hotové projekty jsou skutečně použitelné v praxi a aplikovatelné na nejrůznější životní problémy. Robota s pár senzory lze snadno upravit na měřicí sondu ve skleníku, robota sledujícího lampičku přetvořit na sledování slunce pro elektrovoltaické panely. Stačí pouze žáky dostatečně motivovat a poskytovat jim dostatek podnětů k tvůrčí práci.

4.4 Průmyslové použití

Pro použití v průmyslu jsou vhodné produkty ArduPLC, M-DUINO, nebo například Industruino, které splňují průmyslové standarty. S těmi se žáci při výuce nejspíš nesetkají,

ale jádro těchto produktů je tvořeno stejným základem jako Arduino a proto je i programování stejné.



Obr. 12- ArduPLC [15]

4.5 Budoucnost Arduina

Vzhledem k velikosti komunity má Arduino poměrně jistou budoucnost. Existují sice levnější, menší, nebo výkonnější alternativy, ale Arduino nabízí ideální mix těchto vlastností a hlavně se stále vyvíjí. V současné době se společnost soustředí na produkty pro „internet věcí“, což je sice důležité, ale hlavní je, že se nezapomíná na vzdělání. Výuce je věnována poměrně velká část portfolia značky a dokonce je dostupný výukový program Arduino CTC 101 pro školy. Společnost si tak vychovává mladé zákazníky, kteří se již ve škole setkají s jejími produkty.

Do programu CTC 101 se zapojilo 735 škol, 17700 studentů a 1550 učitelů po celém světě- [17]



Obr. 13- CTC 101 [17]

II. PRAKTICKÁ ČÁST

5 PROJEKT

Praktickou část diplomové práce tvoří dvanáct úloh, které mohou sloužit jako metodické materiály pro učitele, učící na středních školách, nebo v zájmových kroužcích. Úlohy by měly být zpracovávány formou projektové výuky, nikoliv formou frontální výuky. Úlohy jsou až na výjimky snadno rozšiřitelné, tak aby pokrývaly široké spektrum žáků. Každý žák by měl dosáhnout stanovených cílů, při čemž aktivní žáci si mohou stanovit cíle nad rámec projektu. V souboru úloh jsou i úlohy navíc, které nejsou nutné ke splnění finální nejsložitější úlohy. Je možné je v případě potřeby vynechat.

Základní informace o projektu jsou uvedeny v tabulce níže.

Tab. 3- informace o projektu

Forma realizace:	Dlouhodobý projekt
Cílová skupina:	Střední škola, zájmové kroužky
Časový rámec:	Pololetí
Vzdělávací oblast:	Informační a komunikační technologie Jazyk a jazyková komunikace Matematika a její aplikace Člověk a příroda
Průřezová témata:	Rozvoj schopností poznávání Seberegulace a sebeorganizace Kreativita Komunikace Kooperace a kompetice Řešení problémů a rozhodovací dovednosti
Cíle (očekávané výstupy)	Žák zná zásady bezpečnosti práce s elektronikou a elektrickými zařízeními. Žák chápe význam projektu a spolupráce s ostatními

	<p>v teamu.</p> <p>Žák se dokáže orientovat ve schématech zapojení, ve kterých rozezná jednotlivé součástky, ke kterým zvládne přiřadit odpovídající vlastnosti.</p> <p>Žák dokáže podle schématu zapojit funkční model a nahrát do něj příslušný program.</p> <p>Žák zvládne vytvořit jednoduchý program, který v případě potřeby rozšíří podle instrukcí v zadání, za pomoci odborné literatury a internetu.</p> <p>Žák se orientuje v základní struktuře programu a běžně používaných prvcích jazyka Wiring.</p> <p>Žák úspěšně dokončí poslední úlohu v projektu a dokáže ji obhájit.</p>
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

6 PROGRAMOVACÍ A PROTOTYPOVACÍ PROSTŘEDÍ

6.1 Cíle výuky

- Seznámení žáků s prostředím pro psaní zdrojového kódu na PC a návrhem projektů online.
- Žák bude umět vyhledat a nainstalovat program pro požadovaný operační systém.
- Žák se bude orientovat v základní struktuře prostředí programu.
- Žák si osvojí práci v online prostředí Tincercad.

6.2 Zadání

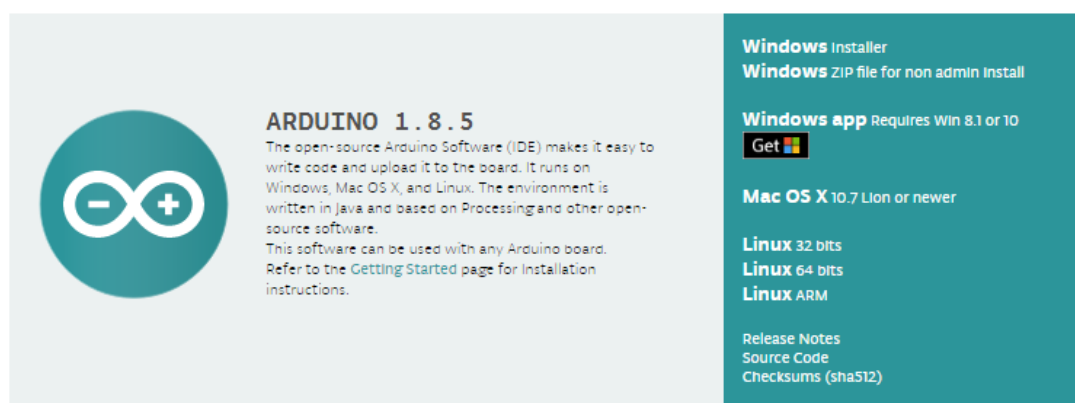
1. Nainstalujte si oficiální Arduino IDE a seznamte se s prostředím programu.
2. Vyzkoušejte si online verzi vývojového prostředí.
3. Seznamte se s prostředím Tincercad a vyzkoušejte si sestrojít jednoduchý obvod.

6.3 Vypracování

6.3.1 Stažení a instalace

Oficiální Arduino IDE lze získat více způsoby. Na webu arduino.cc lze v sekci SOFTWARE stáhnout instalační soubor pro požadovaný operační systém.

Download the Arduino IDE



The image shows a screenshot of the Arduino IDE download page. On the left, there is a circular logo with a minus sign and a plus sign. To the right of the logo, the text reads: **ARDUINO 1.8.5**. Below this, it says: "The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software. This software can be used with any Arduino board. Refer to the [Getting Started](#) page for installation instructions." On the right side of the page, there are several installation options: "Windows Installer", "Windows ZIP file for non admin Install", "Windows app Requires Win 8.1 or 10" with a "Get" button, "Mac OS X 10.7 Lion or newer", "Linux 32 bits", "Linux 64 bits", "Linux ARM", "Release Notes", "Source Code", and "Checksums (sha512)".

Obr. 14- stažení a instalace 1 [19]

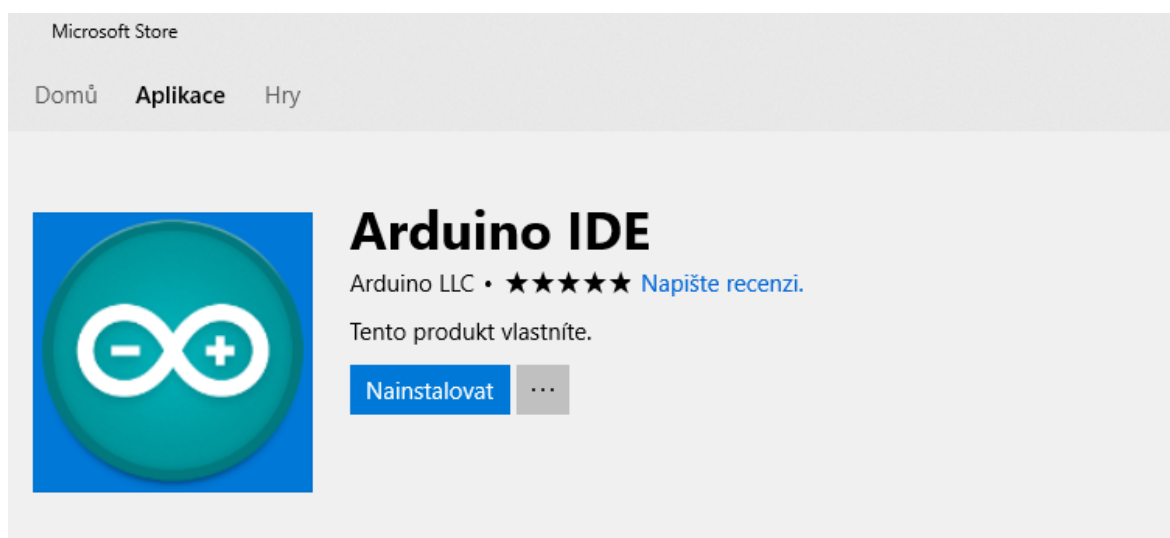
Ve stejné sekci je možné naleznout odkaz na online prostředí, které ale vyžaduje instalaci pluginu Arduino create agent pro komunikaci mezi deskou a prostředím v internetovém prohlížeči, proto z mého pohledu poněkud ztrácí smysl.

Access the Online IDE

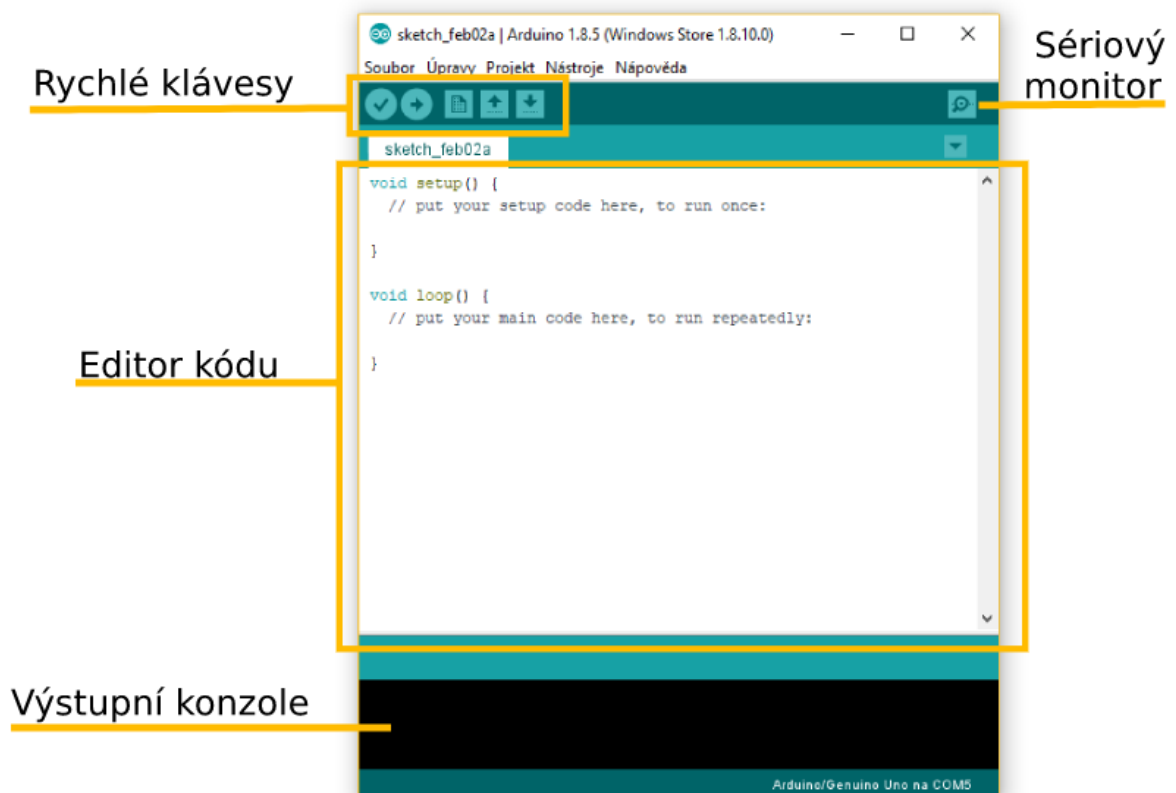


Obr. 15- stažení a instalace 2 [19]

Další možností je na novějších verzích Windows vyhledat aplikaci v Microsoft Store a nainstalovat jedním kliknutím.



Obr. 16- stažení a instalace 3 [19]



Obr. 17- prostředí arduino IDE

Rychlé klávesy reprezentují nejpoužívanější funkce. Zleva doprava se jedná o kompilaci, odeslání programu do Arduina, založení nového projektu, otevření projektu a uložení. Na stejné úrovni úplně napravo se nachází ikona pro otevření sériového monitoru, pomocí něhož lze s Arduinem komunikovat. Editor kódu slouží pro vlastní psaní a úpravu zdrojového kódu. Výstupní konzole je důležitá pro vypsání chybových hlášení při kompilaci.

V záložce „Soubor“ se nacházejí stejné funkce jako u jiného používaného softwaru. Zajímavou položkou je zde podzáložka s příklady. Ty lze například použít jako zdroj inspirace, nebo je používat při testování nového hardwaru.

V záložce „Úpravy“ jsou funkce především pro úpravu textu. Nejdůležitější položkou zde bude pravděpodobně funkce „Hledat“.

Záložka „Projekt“ ukrývá již zmíněné dvě funkce kompilace a nahrání. Podstatnou funkcí jde zde možnost správy knihoven, bez kterých se nedá moc efektivně pracovat.

Záložka „Nástroje“ obsahuje pro začátek dvě podstatné věci. Výběr typu Arduino desky a číslo portu, na kterém je deska připojena. To lze zjistit ve správci zařízení systému Win-

dows, kde je připojené zařízení zobrazeno včetně čísla portu. Další možností vybrat číslo portu metodou pokus/omyl.

Poslední záložka „Nápověda“ již klasicky obsahuje nápovědu a další věci s tím související.

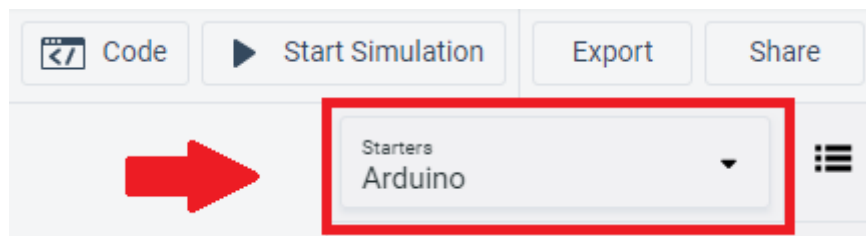
6.3.2 Připojení knihoven

Knihovna se do programu připojuje za pomoci funkce `include` a jména knihovny. Připojení vypadá následovně: `#include <DHT.h>`. Pokud knihovna není přítomna v Arduino IDE, je potřeba ji nejdříve nainportovat. Nejjednodušší způsob je, stáhnou knihovnu ve formátu .zip a v Arduino IDE, v nabídce Projekt->Přidat knihovnu->Přidat .ZIP knihovnu.

6.3.3 TincerCAD

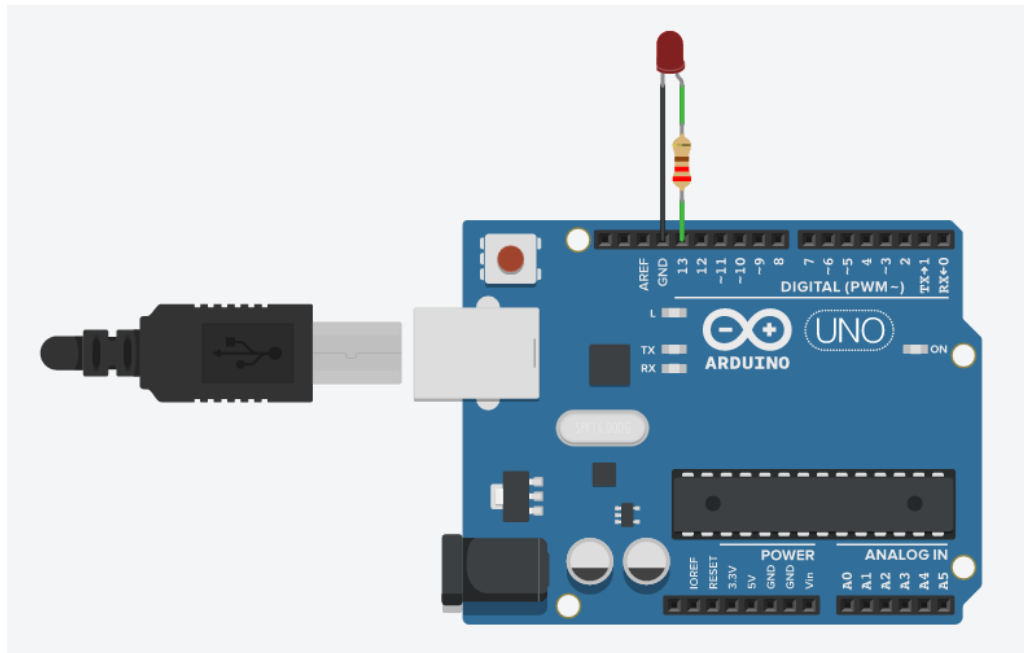
Kromě programovacího prostředí je možné psát kód v online prostředí Tincercad, kde je možné program nahrát do virtuálního hardware a otestovat jej.

Po spuštění prostředí Tinkercad vybereme v nabídce Starters Arduino viz obrázek. V nově zpřístupněné nabídce jsou na výběr základní Arduino zapojení. Po přetažení obvodu na pracovní plochu prostředí s ním můžeme dále pracovat.



Obr. 18- výběr součástek

Po kliknutí na ikonu „Star Simulation“ začne obvod pracovat dle naprogramování.



Obr. 19- schéma zapojení LED

Po kliknutí na ikonu „Code“ se otevře kód patřící k vybranému obvodu. Ten si můžeme zobrazit buď v podobě bloků, kódu, nebo obojího. Kód můžeme dále použít v jiném editoru, nahrát ho do Arduino desky a zapojení si zprovoznit.



Obr. 20- ukázka kódu

Při dalším testování se nemusíme omezovat pouze na předem vytvořená zapojení, ale můžeme si vytvořit i zcela vlastní. Stačí v nabídce místo „Starters“ vybrat „Components“ a najít si součástky dle potřeby. Bohužel nejsou dostupné všechny součástky, se kterými se v reálném světě potkáme. Prostředí Tincercad je ideální pro testování zapojení, pokud si nejsme jistí jejich správností. Prostředí nás například upozorní při překročení maximálního proudu a podobně. Je třeba mít na vědomí, že ne vše se dá v tomto prostředí řádně odsimulovat.

7 PROGRAMOVACÍ JAZYK WIRING

7.1 Cíle výuky

- Žák se seznámí se základní strukturou, funkcemi a dalšími součástmi programu.
- Žák se bude orientovat ve vlastních i cizích programech

7.2 Zadání

1. Seznamte se se základní strukturou programu.
2. Projděte si základní operátory.
3. Nastudujte si základní funkce a vyzkoušejte je v praxi.
4. Projděte si základní datové typy a vyhledejte, ve kterých případech se který z nich používá.
5. Vyzkoušejte si jednotlivé příkazy jazyka Wiring ze sekce „Příkazy jazyka Wiring“

7.3 Vypracování

Základní struktura

Setup()- funkce setup se provede pouze jednou na začátku programu, je vhodná k načtení proměnných, pinmódů atd.

Loop()- smyčka loop se vykonává stále dokola a jsou v ní umístěny příkazy, které chceme vykonávat

Základní operátory

Tab. 4- základní operátory

Aritmetické operátory		Porovnávací operátory		Booleanovské oprátory	
%	Zbytek po dělení	!=	Není rovno	!	Logické ne
*	Násobení	<	Menší než	&&	AND
+	Součet	<=	Menší, nebo rovno		OR
-	Rozdíl	==	Je rovno		
/	Dělení	>	Větší než		
=	Přiřazení	>=	Větší, nebo rovno		

Základní funkce

digitalRead() - přečte hodnotu z digitálního pinu, vrací hodnoty HIGH, nebo LOW

digitalWrite() - zapíše hodnotu HIGH, nebo LOW na výstup digitálního pinu

pinMode() - nastaví pin jako vstup, nebo výstup

analogRead() - přečte hodnotu z analogového pinu v rozmezí 0–1023

analogWrite() - zapíše na analogový pin hodnotu v rozmezí 0–255

delay()- pozastaví program na zadaný čas v milisekundách

delayMicroseconds() - pozastaví program na zadaný čas v mikrosekundách

micros() - vrátí hodnotu času v mikrosekundách od startu běhu programu

millis() - vrátí hodnotu času v milisekundách od startu běhu programu

Základní datové typy

string, array, bool, byte, char, double, float, int, long, short, void

Základní příkazy jazyka Wiring

#define- umožňuje přiřadit jméno konstantní hodnotě

#include- používá se k připojení knihoven k programu

Break- používá se k vyskočení z cyklu for, while, nebo k vyskočení ze switch...case

Continue- používá se k přeskokování částí cyklů

For- cyklus s konečným počtem kroků

Goto- používá se ke skokům v programu, nedoporučuje se používat

If...else- podmínka

Return- ukončí funkci a vrátí hodnotu

Switch...case- podobné podmínce

While- cyklus probíhající, dokud platí podmínka

Do...while- stejné jako while, ale s podmínkou na konci, provede se tedy minimálně jednou

8 SEZNÁMENÍ S ARDUINO DESKOU A ZÁKLADNÍMI SOUČÁSTKAMI

8.1 Cíle výuky

- Žák se seznámí s Arduino deskou, rozložením konektorů a způsobem napájení.
- Žák bude vědět, k čemu slouží základní elektronické součástky a dokáže je rozeznat ve schématu zapojení.
- Žák dokáže rozblikat LED na desce Arduino UNO
- Žák zvládne zapojit jednoduchý obvod a pomocí programu ho oživit.
- Žák dokáže samostatně vyhledat a zodpovědět otázky z poslední kapitoly „Otázky k vyřešení“.

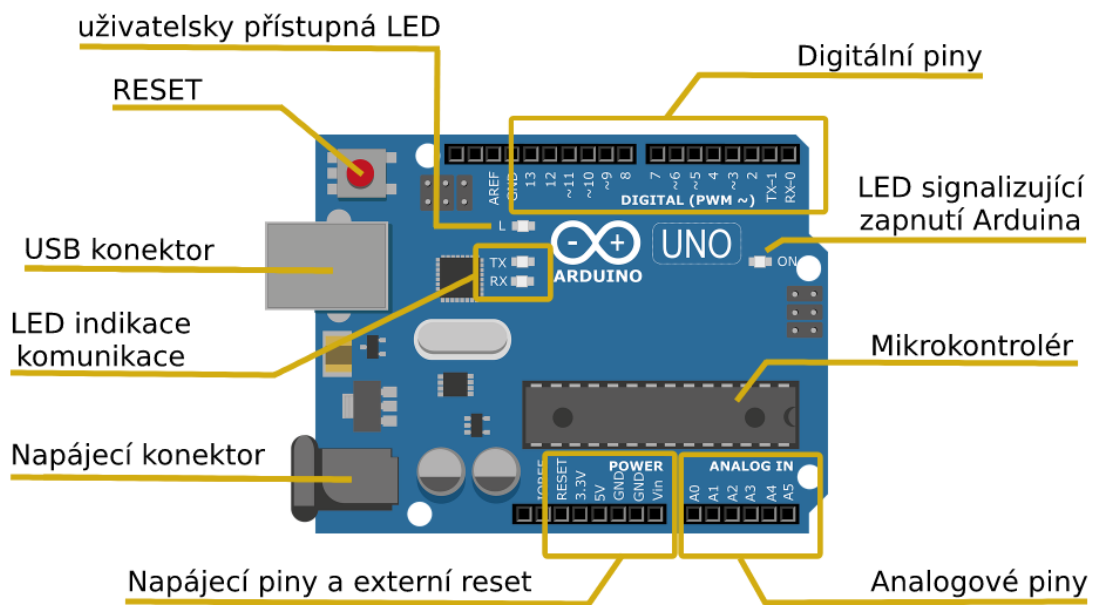
8.2 Zadání

1. Seznamte se z deskou Arduino UNO. Nastudujte si parametry desky a omezení jednotlivých pinů.
2. Vyzkoušejte si připojení desky k PC a nahrání základního programu.
3. Seznamte se s funkcí základních elektronických součástek.
4. Upravte základní program tak, aby jste byli schopní rozsvítit a zhasnout LED na desce Arduina. Vyhledejte si, na kterém pinu je LED připojena.
5. Sestrojte jednoduchý obvod s jednou LED a jedním rezistorem a rozblikajte LED stejně jako v předchozím úkolu.
6. Vypracujte, či zodpovězte úkoly z kapitoly „Úkoly k vyřešení“.

8.3 Vypracování

Arduino UNO

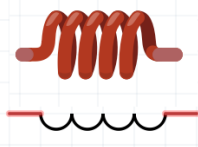
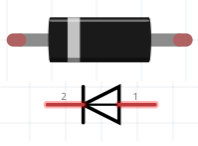
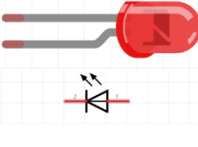
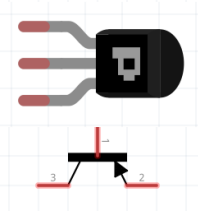
Jednotlivé klony Arduino desek se mohou lehce lišit, ale rozložení pinů, je vždy stejné. U novějších modifikací může být USB konektor nahrazen jeho micro verzí. Potřebný kabel je většinou přiložen v balení. Deska může být napájena buď skrze USB konektor (5V), napájecí konektor (6-20V), nebo přes napájecí piny (5V). Nedodržení předepsaných hodnot může nenávratně poškodit desku, případně i ostatní připojené periferie!



Obr. 21- Popis Arduino desky

Tab. 5- základní elektronické součástky

Základní elektronické součástky		
Název	Obrázek a značka	Popis
Rezistor		<p>Rezistor je diskrétní součástka, která klade odpor procházejícímu elektrickému proudu.</p> <p>Hodnotu odporu odečteme ze značení na rezistoru, nebo změříme k tomu určeným měřidlem. [18]</p>
Kondenzátor		<p>Kondenzátor je součástka, která uchovává elektrický náboj. Množství náboje je dáno velikostí kapacity kondenzátoru.</p> <p>U elektrolytických kondenzátorů je potřeba dodržet polaritu vývodů! [18]</p>

Cívka		Cívka je součástka tvořená navinutým vodičem kolem jádra, jejíž vlastností je impedan- ce. V Arduino projektech se s ní téměř nese- tkáme. [18]
Dioda		Dioda vzniká spojením dvou typů polovodičů typu P a N. Vlastností diody je to, že vede proud pouze jedním směrem. [18]
LED		Zvláštní případ diody je LED, která díky pou- žitým materiálům vyzařuje viditelné světlo. Existují i další druhy diod jako například foto diody, nebo infradiody. [18]
Tranzistor		Tranzistor je polovodičová součástka se třemi vývody a dělí se podle uspořádání polovodičů na typ PNP a NPN. Na rozpoznání typu tran- zistoru existuje mnemotechnická pomůcka „NPN, šipka ven“. Jednou z nejdůležitějších vlastností je proudový zesilovací činitel. Tranzistor se tedy používá jako zesilovač, ale pro nás a číslicovou techniku je zajímavější použití tranzistoru jako spínače. [18]

8.4 Úkoly k vyřešení

1. Proč musíme před LED zapojit rezistor a jak spočítáme jeho velikost?
2. Je možné rozblikat LED více způsoby? Pokud ano, popište jakými.

9 SENZORY

Pomocí různých senzorů Arduino může měřit, případně získávat zpětnou vazbu ze svého okolí. Takových sensorů, či čidel existují desítky. My se v následující úloze seznámíme se dvěma typy sensorů, které se velmi často používají.

9.1 Cíle výuky

- Žák dokáže připojit čidla k Arduino.
- Žák dokáže vytvořit program a odladit ho tak, aby dával uspokojivé hodnoty.
- Žák chápe, jakým způsobem funguje ultrazvukové čidlo a zvládne upravit program pro fungování v různých prostředích (vzduch, voda, atmosféra Marsu).

Zadání

1. Připojte čidlo DHT22 k Arduino dle schématu.
2. Vytvořte program, který bude na seriovém monitoru periodicky vypisovat naměřenou teplotu a vlhkost.
3. Porovnejte naměřené výsledky s referenčními údaji, případně s naměřenými hodnotami ostatních skupin.
4. Připojte čidlo HC-SR04 k Arduino dle schématu.
5. Vytvořte program, který bude na seriovém monitoru periodicky vypisovat naměřenou vzdálenost.
6. Ověřte, zda vzdálenost odpovídá skutečnosti.
7. Vypracujte úkoly, či zodpovězte otázky z kapitoly „Úkoly k vyřešení“.

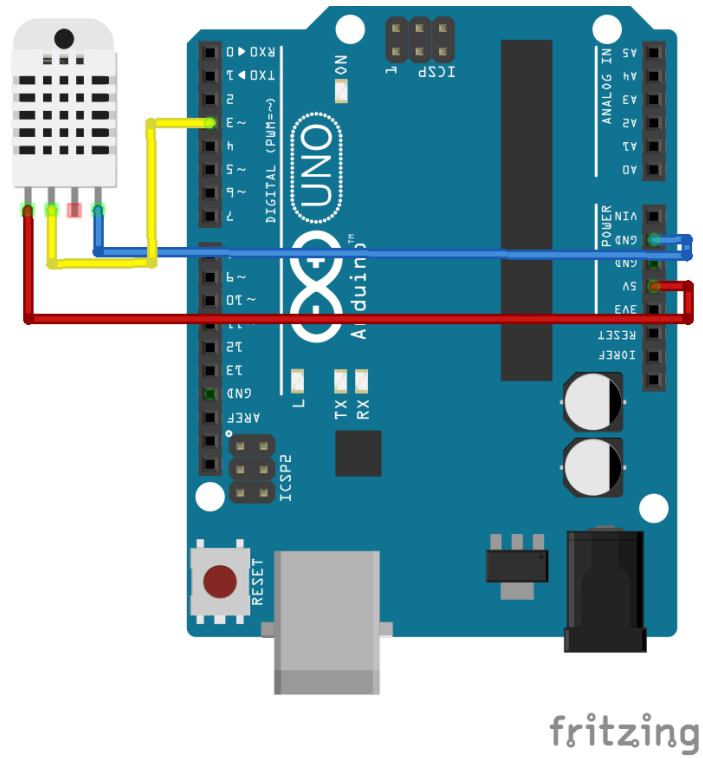
9.2 Vypracování

9.2.1 Potřebné součástky

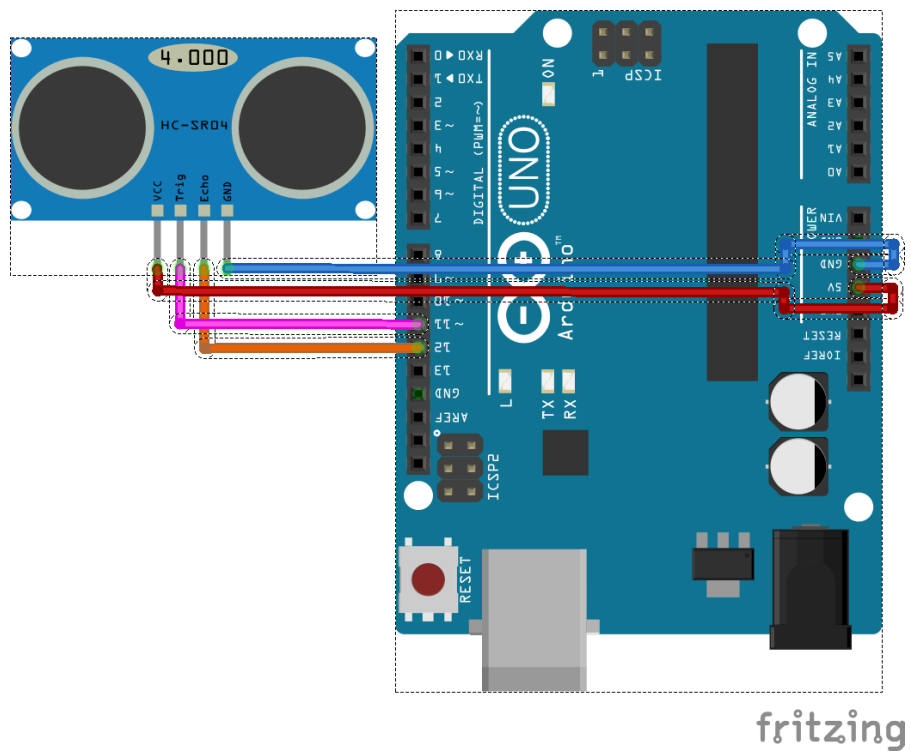
- **Arduino UNO**
- **Digitální senzor teploty a vlhkosti DHT22-** Teplotní rozsah čidla je $-40-80^{\circ}\text{C}$ s přesností $\pm 0,5^{\circ}\text{C}$. Rozsah měření vlhkosti $0-100\%$ s přesností $\pm 2\%$. Napájení $3.3-6\text{V DC}$.
- **Ultrazvukový senzor HC-SR04-** Měří od 2cm do 4m, při čemž při praktickém použití se ukázala jako minimální hodnota 6 cm a maximální 365 cm. Napájecí napětí je 5V DC . Úhel měření je 15° .

- Propojovací vodiče

9.2.2 Schéma zapojení



Obr. 22- schéma zapojení DHT22



Obr. 23- schéma zapojení HC-SR04

9.2.3 Program DHT22

```
#include <DHT.h> //knihovna pro čidlo DHT
#define typDHT22 DHT22 //definování typu DHT čidla
DHT cidlo(3,DHT22); //číslo pinu, ke kterému je připojené DHT čidlo
//deklarace proměnných
float temp;
float hum;
void setup() {
  Serial.begin(9600); //zahájení sériové komunikace rychlostí 9600 baudů
  cidlo.begin(); //zahájení komunikace s cídlem DHT
}
void loop() {
  //načtení teploty a vlhkosti do proměnných
  temp = cidlo.readTemperature();
  hum = cidlo.readHumidity();
  delay(5000); //spoždění 5s
  //výpis proměnných do seriového monitoru
  Serial.print(" Teplota: ");
  Serial.print(temp);
  Serial.print(" C ");
  Serial.print(" Vlhkost: ");
  Serial.print(hum);
  Serial.println("% ");
}
```

9.2.4 Program HC-SR04

```
//deklarace proměnných
long odezva;
long vzdalenost;
void setup() {
  Serial.begin(9600); //zahájení sériové komunikace rychlostí 9600 baudů
  //nastavení pinů ultrazvukového čidla
  pinMode(11,OUTPUT);
  pinMode(12,INPUT);
}
```



```
void loop() {  
  //pomocí funkce digitalWrite vyšleme impulz  
  digitalWrite(11, LOW);  
  delayMicroseconds(2);  
  digitalWrite(11, HIGH);  
  delayMicroseconds(5);  
  digitalWrite(11, LOW);  
  // pomocí funkce pulseIn získáme následně délku pulzu v mikrosekundách (us)  
  odezva = pulseIn(12, HIGH);  
  // přepočítání získaného času na vzdálenost v cm  
  vzdalenost = odezva / 58.31;  
  delay(2000);  
  //vypsání vzdálenosti na seriový monitor  
  Serial.print(" Vzdalenost: ");  
  Serial.print(vzdalenost);  
  Serial.print(" cm ");  
  Serial.println();  
}
```

9.2.5 Ověření funkčnosti

Po připojení Arduina s programem DHT22 k počítači, bude na seriovém monitoru zobrazována teplota a vlhkost v intervalu 5 sekund. Tento interval lze měnit pomocí funkce `delay()`. Správnost údajů můžeme ověřit referenčním teploměrem a vlhkoměrem. V případě nepřesnosti lze údaje doladit vynásobením vhodnou konstantou.

Po připojení Arduina s programem HC-SR04 k počítači, bude na seriovém monitoru zobrazována vzdálenost v intervalu 2 sekund. Interval lze měnit funkcí `delay()`. Vzdálenost lze ověřit vhodným měřidlem a případně v programu doladit pomocí úpravy použité konstanty.

9.3 Úkoly k vyřešení

1. Jak se spočítá konstanta „58.31“ v programu HC-SR04?
2. Bude ultrazvukový senzor dávat stejné výsledky při různé teplotě a v odlišných prostředích? Vyhledejte rychlost zvuku ve vodě a upravte program tak, aby dával správné výsledky.

10 DC A SERVO MOTOR

10.1 Cíle výuky

- Žák dokáže připojit motory k Arduino a prakticky je využít.
- Žák chápe rozdíl mezi servo a krokovým motorem.
- Žák zná několik oblastí pro praktické využití jednotlivých typů motorů.

10.2 Zadání

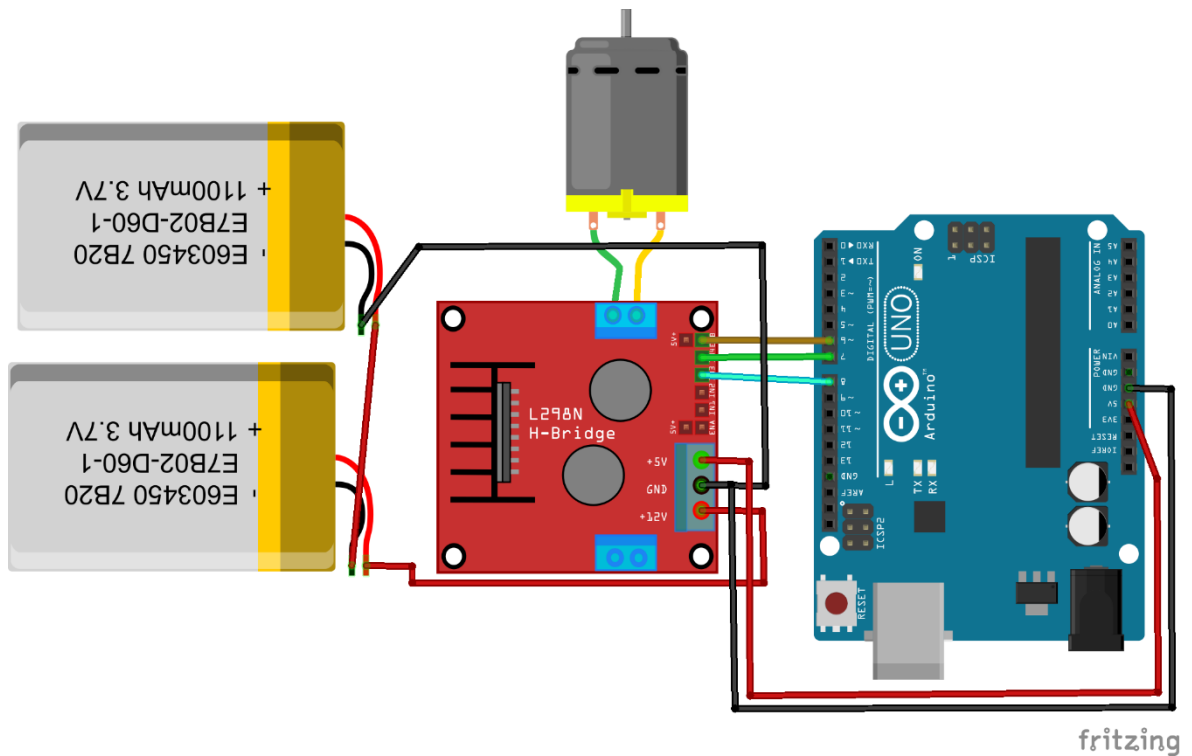
1. Připojte DC motor k Arduino dle schématu pomocí H můstku L298.
2. Vytvořte program pro postupné zvyšování otáček motoru a změnu směru otáčení v periodických intervalech.
3. Připojte servo motor k Arduino dle schématu.
4. Vytvořte program pro ovládání servo motoru
5. Vypracujte úkoly, či zodpovězte otázky z kapitoly „Úkoly k vyřešení“.

10.3 Vypracování

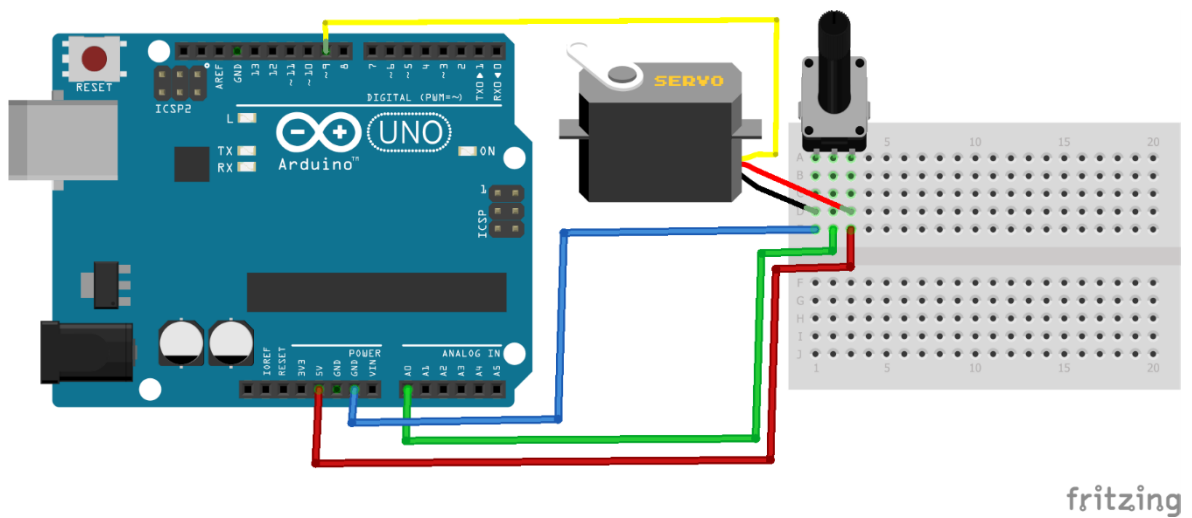
10.3.1 Potřebné součástky

- **Arduino UNO**
- **DC motor s převodovkou- 3-12V**
- **Servo motor SG90-** napětí 4,2–6V, rychlost změny 0.3s/60°
- **H můstek L298-** napětí 5–35V, proud 2A, max. výkon 25W, může řídit, jeden dvoufázový krokový motor, jeden čtyřfázový krokový motor, nebo dva DC motory. Vestavěný 5V stabilizátor
- **Propojovací vodiče**

10.3.2 Schéma zapojení



Obr. 24- schéma zapojení DC motoru



Obr. 25- Schéma zapojení servo motoru

10.3.3 Program DC motor

```
//deklarace proměnných
int rychlost =0;
void setup() {
```

```
Serial.begin(9600); //zahájení sériové komunikace rychlostí 9600 baudů
//nastavení pinů motorů
pinMode(6,OUTPUT);
pinMode(7,OUTPUT);
pinMode(8,OUTPUT);
}
void loop() { //nastavení H můstku
    digitalWrite(8,HIGH);
    digitalWrite(7,LOW);
    //cyklus pro zvyšování otáček motoru
    for (rychlost i=0; i <= 255; i++){
        analogWrite(6,rychlost);
        delay(10);
    }
    //cyklus pro snižování otáček motoru
    for (rychlost i=255; i >= 0; i--){
        analogWrite(6,rychlost);
        delay(10);
    }
    //změna otáčení motoru
    digitalWrite(8,LOW);
    digitalWrite(7,HIGH);
    for (rychlost i=0; i <= 255; i++){
        analogWrite(6,rychlost);
        delay(10);
    }
    for (rychlost i=255; i >= 0; i--){
        analogWrite(6,rychlost);
        delay(10);
    }
    delay(2000); //pauza 2s
}
```

10.3.4 Program Servo

```
#include <Servo.h> //připojení knihovny Servo.h
Servo servo1; //deklarace servomotoru
//deklarace proměnných
int potenciometr = 0;
int val;
void setup()
{
  servo1.attach(9); //číslo pinu připojeného motoru
}
void loop()
{
  //napětí na potenciometru (0 až 1023)
  val = analogRead(potenciometr);
  //převod z 0 až 1023 na 0 až 180
  val = map(val, 0, 1023, 0, 180);
  //nastavení polohy podle potenciometru
  myservo.write(val);
}
```

10.3.5 Ověření funkčnosti

Při spuštění programu bude DC motor zvyšovat plynule své otáčky do maxima a následně je zase snižovat. Při zastavení se obrátí směr otáčení a motor zase zrychlí a zpomalí. Na hřídelku motoru je vhodné přilepit kousek papíru pro lepší určení směru otáčení.

10.4 Úkoly k vyřešení

Jaký je rozdíl mezi servo motorem a krokovým motorem? Jaké jsou jejich výhody a nevýhody?

11 SPÍNÁNÍ ZÁTĚŽE

11.1 Cíle výuky

- Žák dokáže připojit relé k Arduino.
- Žák dokáže vytvořit program, kterým bude relé ovládat.
- Žák si uvědomuje nebezpečí při zacházení se zařízeními pracujícími se síťovým napětím.

11.2 Zadání

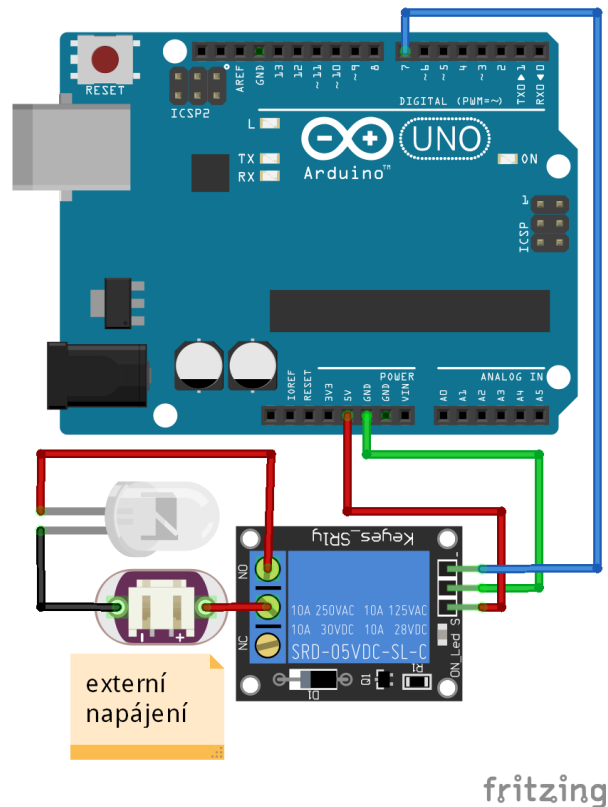
1. Připojte relé k Arduino dle schématu.
2. Spínané zařízení zvolte dle vlastního uvážení.
3. Vytvořte program, který bude zařízení v pěti sekundových intervalech zapínat a vypínat.
4. Vypracujte úkoly, či zodpovězte otázky z kapitoly „Úkoly k vyřešení“.

11.3 Vypracování

Potřebné součástky

- **Arduino UNO**
- **Relé 5V-** spínání 3,5–12V volty, spínání až 220V zátěže
- **Propojovací vodiče**

11.3.1 Schéma zapojení



Obr. 26- schéma zapojení relé

11.3.2 Program

```
void setup() {  
    //číslo portu, ke kterému je připojeno relé a nastavení pinu jako výstup  
    pinMode(7, OUTPUT);  
}  
void loop() {  
    //sepnutí relé  
    digitalWrite(7, HIGH);  
    //pauza 5s  
    delay(5000);  
    //vypnutí relé  
    digitalWrite(7, LOW);  
    delay(5000);  
}
```

11.3.3 Ověření funkčnosti

Při spuštění programu bude relé spínat a rozepínat každých pět sekund a LED na jeho výstupu tedy bude blikat.

11.4 Úkoly k vyřešení

1. Vyhledejte, na jakém principu relé funguje.
2. Nastudujte si zásady bezpečnosti při zacházení se síťovým napětím.

12 PRÁCE S DISPLEJEM

12.1 Cíle výuky

- Žák dokáže připojit displej k Arduino a vypisovat na něj informace.
- Žák zvládne upravit ostatní úlohy pro použití s displejem.
- Žák si dokáže poradit s omezeným počtem znaků na displeji.

12.2 Zadání

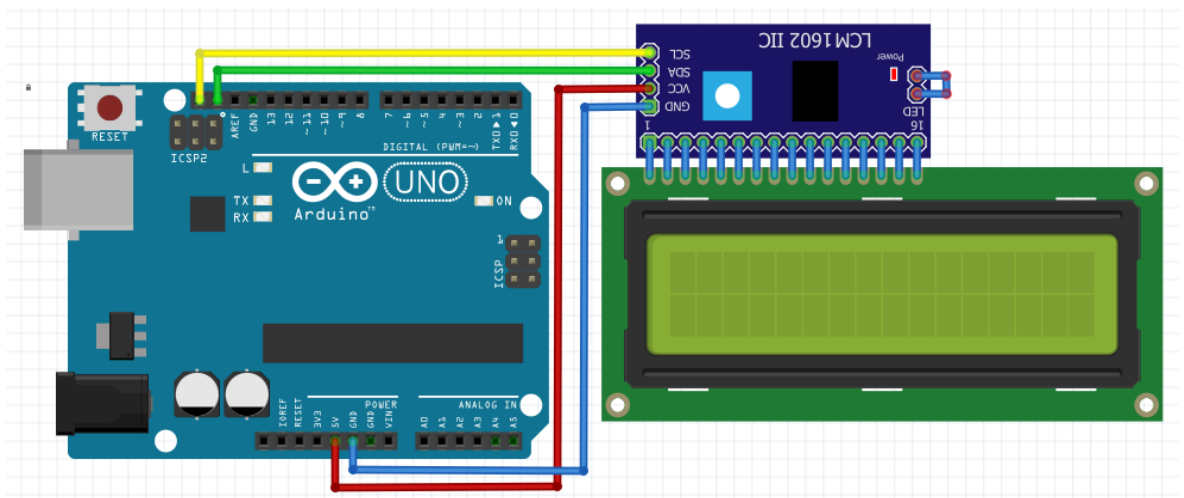
1. Připojte displej k Arduino pomocí I2C modulu.
2. Vytvořte program, pomocí kterého na první řádek displeje vypíšete své jméno a na druhý řádek příjmení.
3. Vypracujte úkoly, či zodpovězte otázky z kapitoly „Úkoly k vyřešení“.

12.3 Vypracování

12.3.1 Součástky

- **Arduino UNO**
- **LCD displej 16x2 5V**
- **I2C modul**
- **Propojovací vodiče**

12.3.2 Schéma zapojení



Obr. 27- Sériové zapojení

12.3.3 Program

```
#include <Wire.h> //připojení knihovny pro I2C sběrnici
#include <LiquidCrystal_I2C.h> //připojení knihovny pro LCD displej
LiquidCrystal_I2C lcd(0x27, 16, 2); //parametry LCD, adresa, sloupce, řádky
void setup() {
  lcd.begin(); //zahájení komunikace s LCD
  lcd.backlight(); //zapnutí podsvícení
  Serial.begin(9600); //zahájení sériové komunikace rychlostí 9600 baudů
}
void loop() {
  //vymazání obrazovky displeje
  lcd.clear();
  //nastavení kurzoru na první znak prvního řádku
  lcd.setCursor(0,0);
  //vypsání jména
  lcd.print("Jindrich");
  //nastavení kurzoru na první znak druhého řádku
  lcd.setCursor(0,1);
  //vypsání příjmení
  lcd.print("Baumann");
  delay(5000);
}
```

12.3.4 Ověření funkčnosti

Po spuštění programu se na displej vypíše jméno a příjmení. Po pěti sekundách se text vymaže a vypíše znovu.

12.4 Úkoly k vyřešení

1. Jde připojit displej k Arduino bez použití I2C modulu a jak by takové zapojení vypadalo?
2. Jaká je obnovovací frekvence zobrazování údajů, při které lidské oko nevnímá blikání?
3. Jakým způsobem je řešeno u displejů mobilních zařízení zvyšování a snižování intenzity podsvícení?

4. Rozšiřte úlohu o zapojení servo motoru a na displeji vypisujte data načtená z potenciometru.

13 RFID

13.1 Cíle výuky

- Žák zvládne připojit RFID čtečku k Arduinu
- Žák se orientuje v programu a dokáže ho upravit i pro jiné, než přiložené RFID tagy
- Žák zná potenciální rizika využití RFID čtečky za účelem zabezpečení.

13.2 Zadání

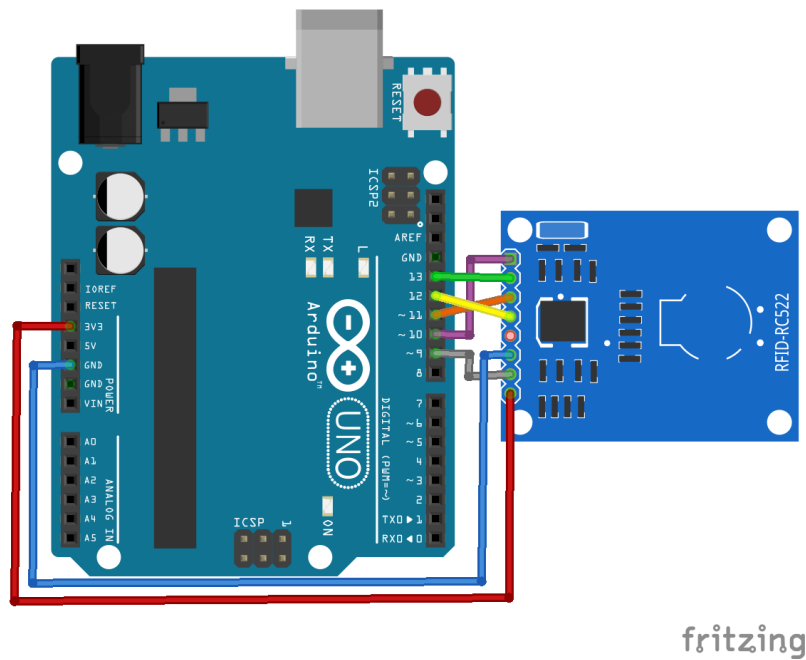
1. Připojte čtečku k Arduinu dle schématu.
2. Vytvořte program, který podle RFID tagu rozpozná uživatele a odemkne, či zamkne „dveře“.
3. Vypracujte úkoly, či zodpovězte otázky z kapitoly „Úkoly k vyřešení“.

13.3 Vypracování

13.3.1 Součástky

- **Arduino UNO**
- **RFID RC-522 13.56 Mhz modul-** pracovní frekvence 13.56 Mhz, napětí 3.3V, čtecí vzdálenost max. 6cm
- **RFID karta a token**
- **Propojovací vodiče**

13.3.2 Schéma zapojení



Obr. 28-schéma zapojení RFID

13.3.3 Program

```
// připojení knihoven
#include <SPI.h>
#include <MFRC522.h>
// definování pinů SDA a RST
#define SDA 10
#define RST 9
// vytvoření instance RFID čtečky z knihovny
MFRC522 rfid(SDA,RST);
//deklarace proměnných
int pom1=0;
int pom2=0;
void setup() {
  // zahájení komunikace přes sériovou linku
  Serial.begin(9600);
  // inicializace komunikace přes SPI
  SPI.begin();
  // inicializace komunikace s RFID čtečkou
  rfid.PCD_Init();
```

```
}  
// podprogram pro výpis adresy RFID tagu v hexa formátu  
void vypisHex(byte *buffer, byte bufferSize) {  
    for (byte i = 0; i < bufferSize; i++) {  
        Serial.print(buffer[i] < 0x10 ? " 0" : " ");  
        Serial.print(buffer[i], HEX);  
    }  
}  
  
void loop() {  
    // kontrola RFID tagů v okolí modulu,  
    // pokud není žádný tag v okolí, volá se loop funkce od začátku,  
    if ( ! rfid.PICC_IsNewCardPresent())  
        return;  
    // kontrola správného přečtení RFID tagu  
    if ( ! rfid.PICC_ReadCardSerial())  
        return;  
    // výpis adresy tagu v hexa formátu  
    Serial.println("Adresa RFID tagu: ");  
    vypisHex(rfid.uid.uidByte, rfid.uid.size);  
    Serial.println();  
    //rozpoznání uživatele  
    if(rfid.uid.uidByte[0] == 0x80 & rfid.uid.uidByte[1] == 0x88 & rfid.uid.uidByte[2] ==  
0x71 & rfid.uid.uidByte[3] == 0xA3)  
        //rozhodnutí, zda odemknout, nebo zamknout  
        {if (pom1==0) {  
            Serial.println("Uzivatel 1: odemceno");  
            pom1=1;  
        }  
        else {  
            Serial.println("Uzivatel 1: zamceno");  
            pom1=0;  
        }  
    }  
    else if(rfid.uid.uidByte[0] == 0x8C & rfid.uid.uidByte[1] == 0xB0 & rfid.uid.uidByte[2]  
== 0xB8 & rfid.uid.uidByte[3] == 0x79)  
    {
```

```
    if (pom2==0) {Serial.println("Uzivatel 2: odemceno");
        pom2=1;
    }
    else {Serial.println("Uzivatel 2: zamceno");
        pom2=0;}
}
else {
    Serial.println("Pristup odepren");
}
Serial.println();
// ukončení komunikace a jejího zabezpečení
rfid.PICC_HaltA();
rfid.PCD_StopCrypto1();
}
```

13.3.4 Ověření funkčnosti

Při přiložení tagu ke čtečce vypíše Arduino na seriový monitor hexa adresu tagu a pokud se shoduje se zadanou hodnotou v programu, odemkne, či zamkne na základě aktuálního stavu. Při prvním použití je potřeba aktualizovat v programu adresu tagu na základě vypsané adresy. Po aktualizaci je možné část programu s výpisem adresy odstranit. Při pokusu o odemčení neznámým tagem Arduino vypíše hlášku „Pristup depren“.

13.4 Úkoly k vyřešení

1. Rozšiřte zapojení o relé, které se bude spínat na základě přiložení RFID tagu.
2. Jaký protokol zabezpečení využívá použitá RFID čtečka? Je bezpečný?

14 BEZDRÁTOVÁ KOMUNIKACE POMOCÍ BLUETOOTH

14.1 Cíle výuky

- Žák se naučí propojit Arduino s Bluetooth modulem tak, aby probíhala komunikace desky a připojeného zařízení po sériové lince.
- Žák bude schopen modifikovat program tak, aby fungoval i pro jiná zařízení připojena na jiných pinech Arduina.

14.2 Zadání

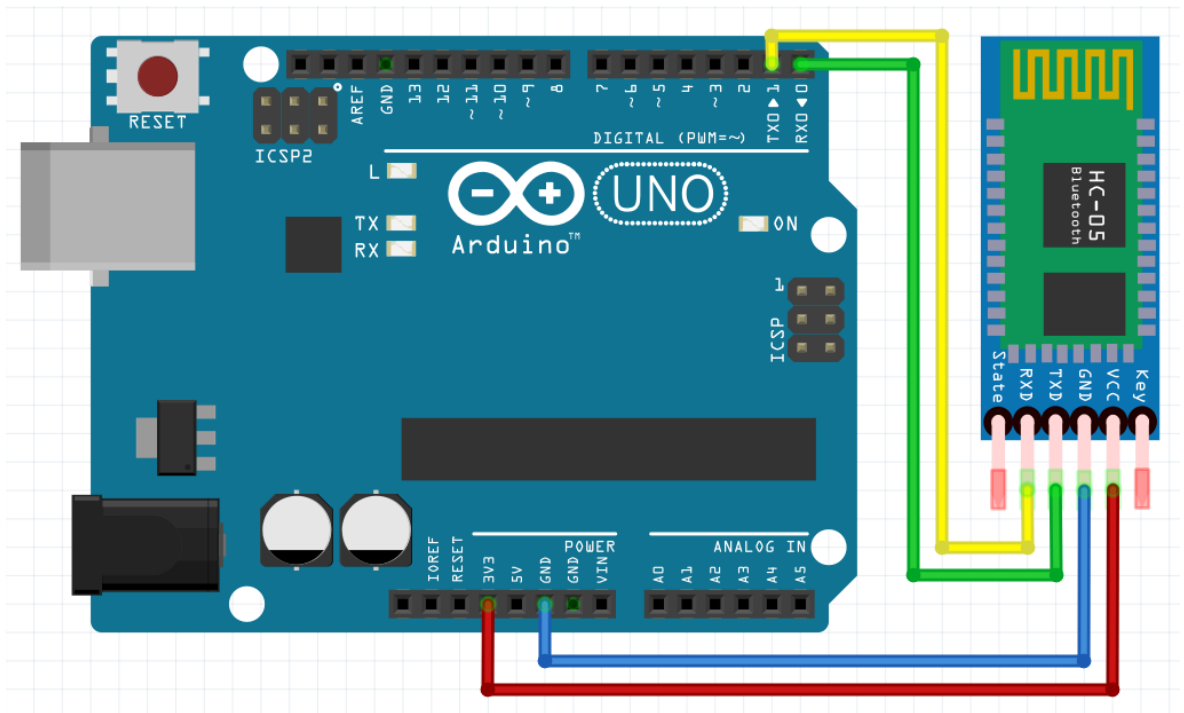
1. Propojte jednotlivé součástky dle schématu, případně ověřte spávnost ve virtuálním prostředí Tincercad.
2. Vytvořte program pro spínání LED diody na desce Arduina, která je připojena na PINU 13.
3. Nahrajte program do Arduina. Před nahráním je nutné odpojit Bluetooth modul!
4. Ověřte funkčnost takto zprovozněného zařízení pomocí počítače, nebo mobilního zařízení.
5. Vypracujte úkoly, či zodpovězte otázky z kapitoly „Úkoly k vyřešení“.

14.3 Vypracování

14.3.1 Součástky

- **Arduino UNO**
- **Bluetooth modul HC-05-** Verze bluetooth 2.0. Napájení 3,3V, vnitřní logika 3,3V, ale zvládne pracovat i na 5V. Podpora master i slave módu. Teoretický dosah 10m.
- **Propojovací vodiče**

14.3.2 Schéma zapojení



Obr. 29- Schéma zapojení bluetooth modulu

14.3.3 Program

```
//Proměná pro uložení přijímaných dat
char data = 0;
void setup()
{
  //Zahájení sériové komunikace rychlostí 9600 baudů
  Serial.begin(9600);
  //Nastavení diody na pinu 13 jako výstup
  pinMode(13, OUTPUT);
}
void loop()
{
  // Podmínka je splněna pokud jsou přijmuta data ze sériové linky
  if(Serial.available() > 0)
  {
    //Přečtení a uložení dat do proměné "data"
    data = Serial.read();
  }
}
```

```
//Vypsání dat na sériovém monitoru
Serial.print(data);
//Odřádkování
Serial.print("\n");
// Pokud je podmínka splněna, je výstup na pinu 13 nastaven na hodnotu "HIGH"
if(data == '1')
    digitalWrite(13, HIGH);
// Pokud je podmínka splněna, je výstup na pinu 13 nastaven na hodnotu "LOW"
else if(data == '0')
    digitalWrite(13, LOW);
}
}
```

14.3.4 Ověření funkčnosti

K ověření funkčnosti je potřeba nainstalovat libovolný sériový monitor pro mobilní zařízení, nebo počítač. Při zadání čísla „1“ se rozsvítí LED na pinu 13, při zadání čísla „0“ zhasne.

14.4 Úkoly k vyřešení

1. Proč je potřeba odpojit Bluetooth modul před nahráním kódu do Arduina?
2. Jaký je rozdíl mezi modulem HC-05 a HC-06?

15 NÁVRH A TVORBA APLIKACE PRO OVLÁDÁNÍ ARDUINA POMOCÍ MOBILNÍHO ZAŘÍZENÍ S OS ANDROID

15.1 Cíle výuky

- Žák se orientuje v aplikaci MIT App Inventor.
- Žák je schopný navrhnout jednoduché grafické prostředí programu.
- Žák zvládne řetězit jednotlivé programové bloky tak, aby aplikace byla funkční.
- Žák bude schopen upravit program na základě změn zapojení hardwaru.

15.2 Zadání

1. Navrhněte jednoduché grafické prostředí tak, aby vyhovovalo požadavku na snadné ovládání a přehlednost.
2. Naprogramujte aplikaci tak, aby ovládací signály korespondovaly s nastavením v aplikaci z úlohy č. 17.
3. Nainstalujte aplikaci na mobilní zařízení a otestujte na úloze č. 17.
4. Vypracujte úkoly, či zodpovězte otázky z kapitoly „Úkoly k vyřešení“.

15.3 Vypracování

Nejdříve je pomocí prvků z nabídky „Layout“ vytvořena základní struktura vzhledu. Dále jsou přidány prvky z nabídky „User interface“. Rozložení prvků je čistě individuální záležitost. Vhodné je velikost prvků nastavovat v procentech. Rovněž jsou přidány prvky, které nejsou v rozložení vidět a to „Bluetooth client“ a „Clock“.

Skutečný vzhled je možné vidět přímo na mobilním zařízení díky použití funkce „Al companion“ z nabídky „Connect“ v horní liště prostředí.

15.3.1 Návrh

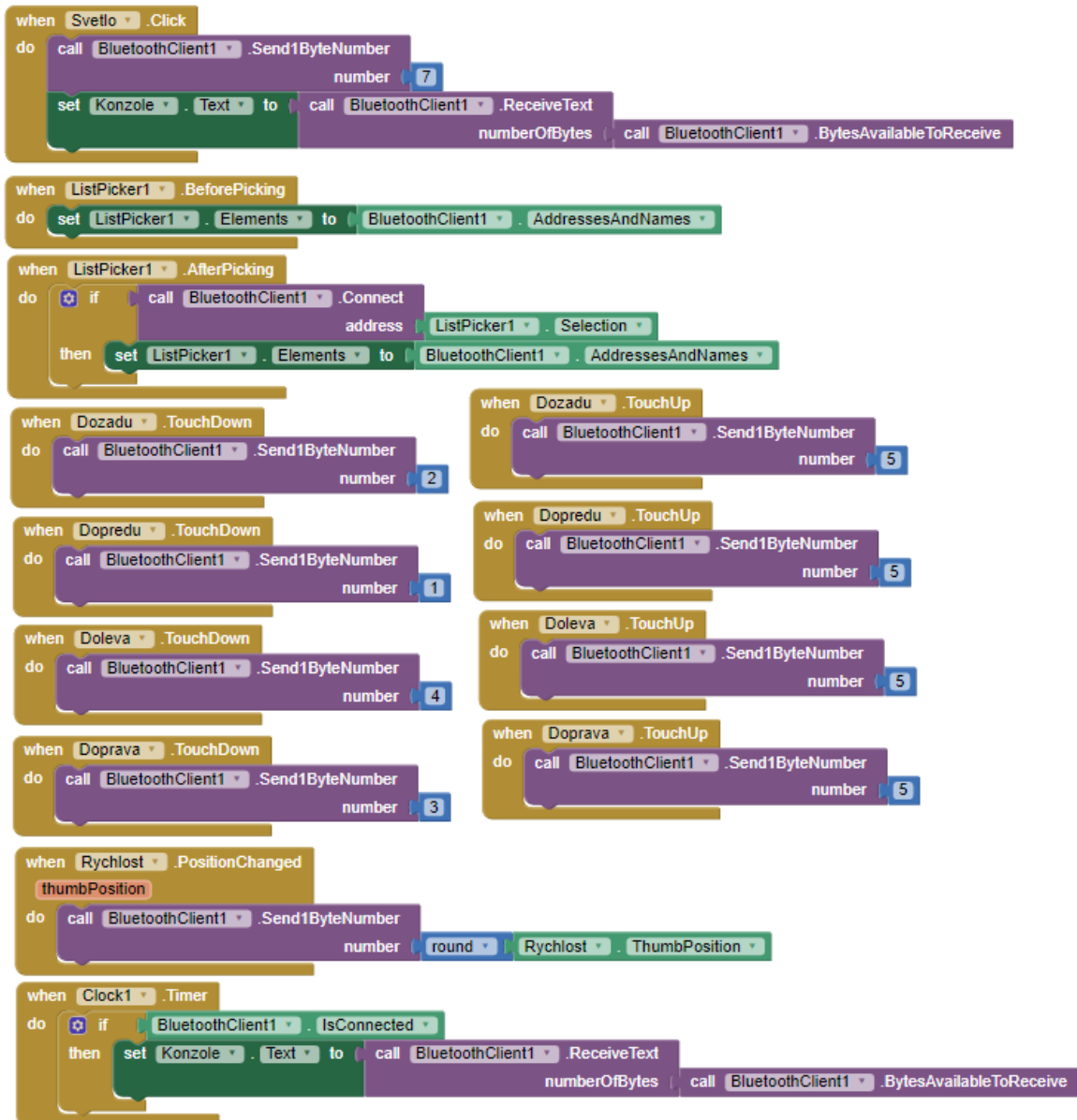


Obr. 30- Mobilní aplikace

15.3.2 Program

Po přepnutí do sekce Blocks je možné aplikaci „naprogramovat“. Některé nabídky se zpřístupní až po přidání prvků v sekci Designer, proto je lepší si aplikaci nejdříve celou navrhnout. První velký blok obstarává zapínání diody. Druhý a třetí blok řeší připojení

k Bluetooth. Prostřední část programu tvoří bloky, které zajišťují ovládání robota. Předposledním blokem je řízení nastavení rychlosti a poslední blok v pravidelném intervalu vypisuje do konzole data příchozí přes Bluetooth.



Obr. 31- Program vytvořený pomocí bloků

15.4 Úkoly k vyřešení

1. Upravte aplikaci tak, aby tlačítko zapínající světlo měnilo barvu podle toho, zda světlo svítí, nebo nesvítí.

16 DÁLKOVĚ OVLÁDANÉ VOZÍTKO (SONDA)

16.1 Cíle výuky

- Žák aplikuje znalosti nabyté v předchozích úkolech a sestaví dálkově ovládané vozítko.
- Žák propojí vozítko s aplikací a dokáže ho vzdáleně ovládat.
- Žák dokáže odstranit případné problémy s konstrukcí a nedokonalosti v měření.

16.2 Zadání

1. Poskládejte podvozek vozidla dle přibaleného návodu, případně si zkonstruujte vlastní.
2. Rozvrhněte umístění součástek na podvozku tak, abyste minimalizovali délku propojovacích vodičů a tak, aby měřicí součástky nebyly ovlivněny vyzařovaným teplem jiných komponent. Dbejte na přístupnost vstupních a výstupních konektorů součástek.
3. Propojte jednotlivé součástky.
4. Vytvořte program a odstraňte případné chyby v kódu.
5. Zkontrolujte zapojení a nahrajte program.
6. Vozítko otestujte a odlaďte.
7. Vypracujte úkoly, či zodpovězte otázky z kapitoly „Úkoly k vyřešení“.

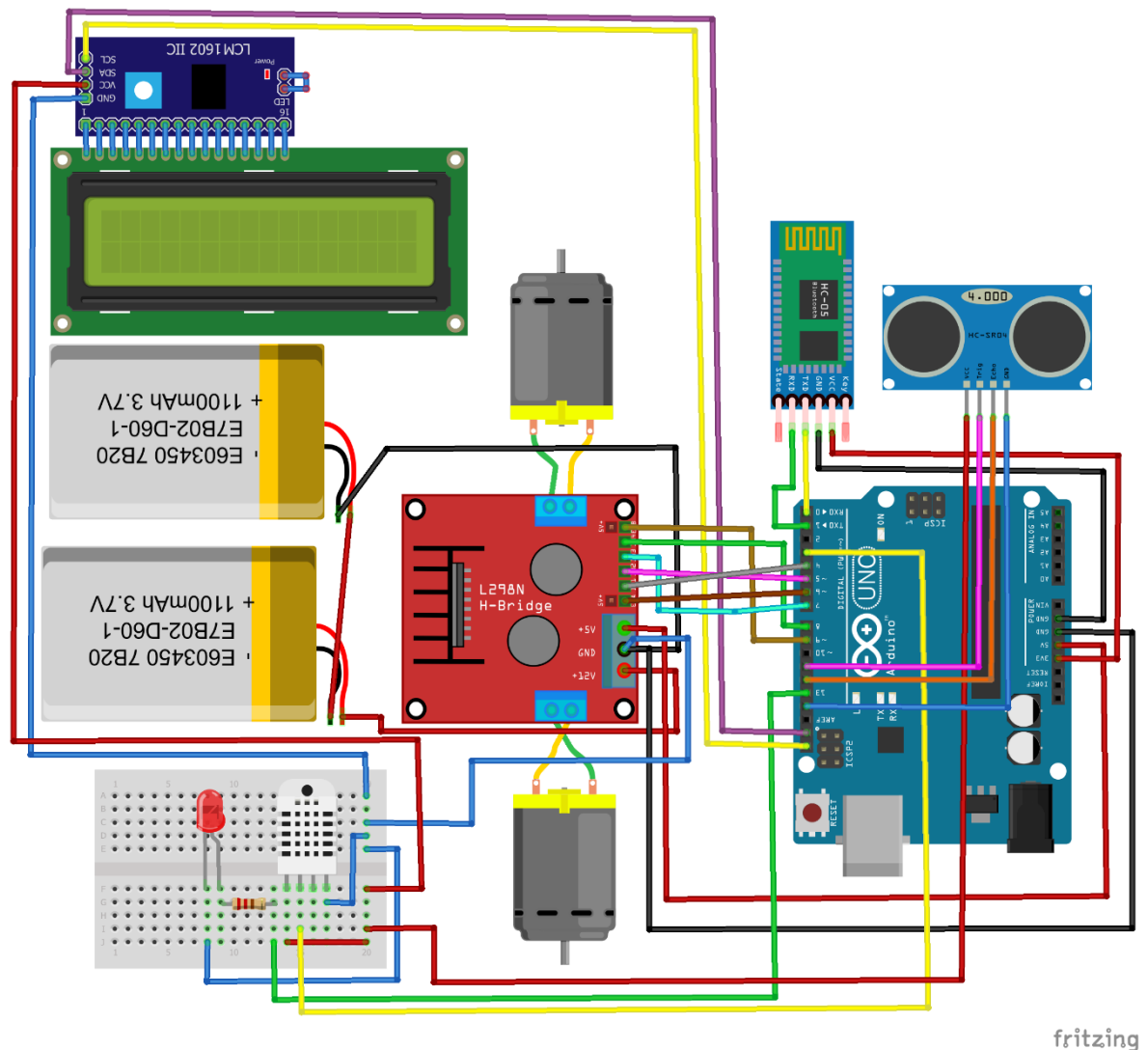
16.3 Vypracování

16.3.1 Součástky

- **Arduino UNO**
- **Podvozek vozítka-** koupený, nebo vlastní výroby
- **2, případně 4 DC motory s převodovkou-** 3–12V
- **H můstek L298-** napětí 5–35V, proud 2A, max. výkon 25W, může řídit, jeden dvoufázový krokový motor, jeden čtyřfázový krokový motor, nebo dva DC motory.
Vestavěný 5V stabilizátor
- **Bluetooth modul HC-05-** Verze bluetooth 2.0. Napájení 3,3V, vnitřní logika 3,3V, ale zvládne pracovat i na 5V. Podpora master i slave módu. Teoretický dosah 10m.
- **LCD displej 16x2 5V+ I2C**

- **Digitální senzor teploty a vlhkosti DHT22-** Teplotní rozsah čidla je $-40-80^{\circ}\text{C}$ s přesností $\pm 0,5^{\circ}\text{C}$. Rozsah měření vlhkosti $0-100\%$ s přesností $\pm 2\%$. Napájení $3.3-6\text{V DC}$.
- **Ultrazvukový senzor HC-SR04-** Měří od 2cm do 4m, při čemž při praktickém použití se ukázala jako minimální hodnota 6 cm a maximální 365 cm. Napájecí napětí je 5V DC . Úhel měření je 15° .
- **Držák ultrazvukového senzoru**
- **LED jakékoliv barvy**
- **Rezistor**
- **Nepájivé pole**
- **Zdroj napájení**
- **Propojovací vodiče**

16.3.2 Schéma zapojení



Obr. 32- Schéma zapojení

16.3.3 Program

```
#include <DHT.h> //knihovna pro čidlo DHT
#include <Wire.h> //knihovna pro I2C sběrnici
#include <LiquidCrystal_I2C.h> //vlození knihovny pro LCD displej
LiquidCrystal_I2C lcd(0x27, 16, 2); //parametry LCD, adresa, sloupce, řádky
#define typDHT22 DHT22 //definování typu DHT čidla
DHT cidlo(3,DHT22); //číslo pinu, ke kterému je připojené DHT čidlo
//deklarace proměnných
char data = 0;
byte rychlost = 100;
```



```
long odezva;
long vzdalenost;
float temp;
float hum;
long cas;
int ledstatus=0;
void setup() {
    lcd.begin(); //zahájení komunikace s LCD
    lcd.backlight(); //zapnutí podsvícení
    Serial.begin(9600); //zahájení sériové komunikace rychlostí 9600 baudů
    cidlo.begin(); //zahájení komunikace s cidlem DHT
    pinMode(13, OUTPUT); //nastavení diody na pinu 13 jako výstup
    //nastavení pinů motorů
    pinMode(4,OUTPUT);    pinMode(5,OUTPUT);
    pinMode(6,OUTPUT);    pinMode(7,OUTPUT);
    pinMode(8,OUTPUT);    pinMode(9,OUTPUT);
    //nastavení pinů ultrazvukového čidla
    pinMode(11,OUTPUT);
    pinMode(12,INPUT);
}
//procedury
void dopredu(){digitalWrite(8,HIGH);    digitalWrite(7,LOW);
               analogWrite(9,rychlost);
               digitalWrite(5,HIGH);    digitalWrite(4,LOW);
               analogWrite(6,rychlost); }
void dozadu(){ digitalWrite(7,HIGH);    digitalWrite(8,LOW);
               analogWrite(9,rychlost);
               digitalWrite(4,HIGH);    digitalWrite(5,LOW);
               analogWrite(6,rychlost); }
void doprava(){digitalWrite(7,HIGH);    digitalWrite(8,LOW);
               analogWrite(9,rychlost);
               digitalWrite(5,HIGH);    digitalWrite(4,LOW);
               analogWrite(6,rychlost); }
```

```
void doleva(){
    digitalWrite(8,HIGH);    digitalWrite(7,LOW);
    analogWrite(9,rychlost);
    digitalWrite(4,HIGH);    digitalWrite(5,LOW);
    analogWrite(6,rychlost);
}

void zastav(){analogWrite(9,0);    analogWrite(6,0);}

void mereni(){digitalWrite(11, LOW);    delayMicroseconds(2);
    digitalWrite(11, HIGH);    delayMicroseconds(5);
    digitalWrite(11, LOW);
    // pomocí funkce pulseIn získáme následně délku pulzu v mikrosekundách
    odezva = pulseIn(12, HIGH);
    // přepočet získaného času na vzdálenost v cm
    vzdalenost = odezva / 58.31;
    temp = cidlo.readTemperature();    hum = cidlo.readHumidity();
    void vypis(){lcd.clear(); lcd.setCursor(0,1); lcd.print("vzdalenost:");
    lcd.setCursor(11,1);
    lcd.print(vzdalenost);

    lcd.setCursor(14,1);
    lcd.print("cm");
}

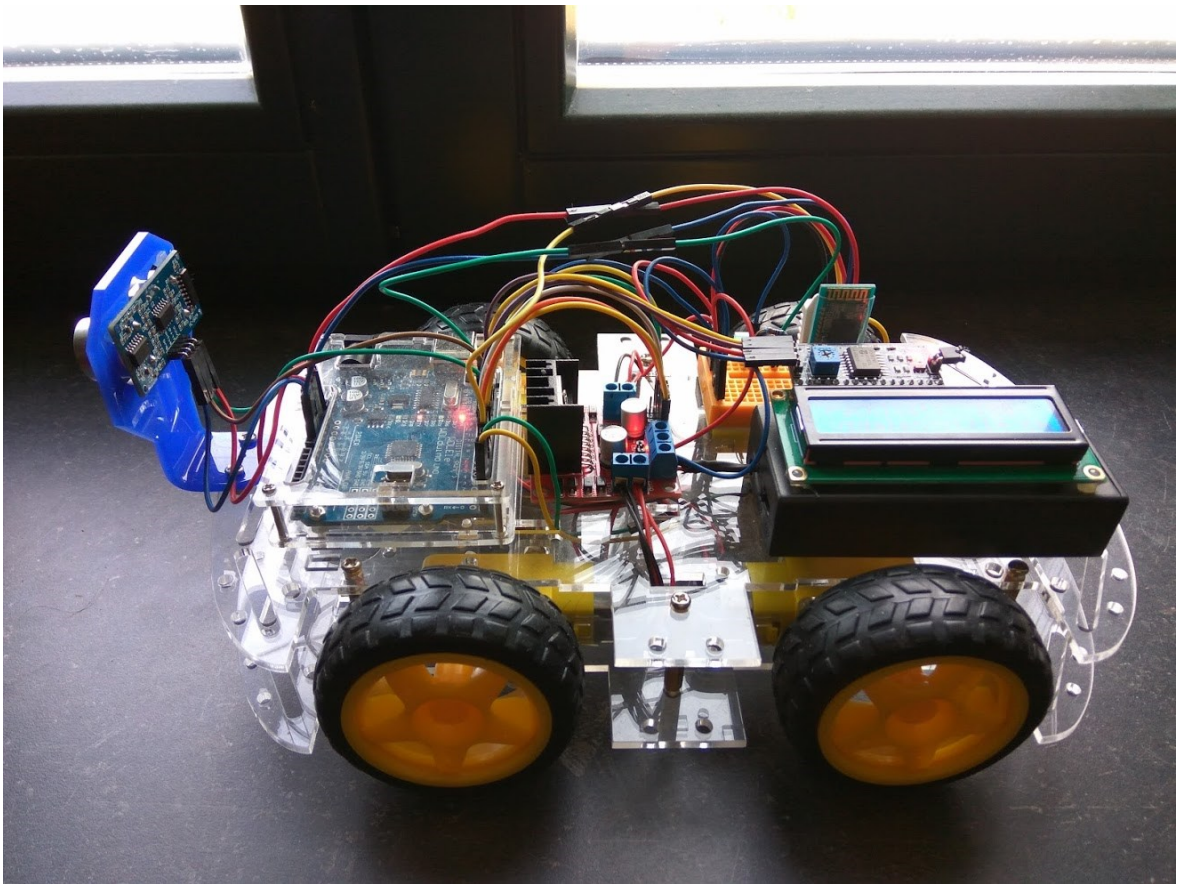
void vypis2(){Serial.print(" Vzdalenost: "); Serial.print(vzdalenost);
    Serial.print(" cm ");    Serial.print(" Teplota: ");
    Serial.print(temp);    Serial.print(" C ");
    Serial.print(" Vlhkost: "); Serial.print(hum);
    Serial.println("% ");    }

void loop()
//periodické vypisování naměřených údajů na LCD a na seriovou linku pomocí funkce millis
if(millis() >= cas+1000){ cas = millis();mereni();vypis();vypis2(); }
//pokud je vzdálenost menší než 20 cm a zároveň robot jede dopředu, robot zastaví a vypíše varovnou hlášku
mereni();
```

```
if((vzdalenost<20)&& (data==1)){zastav(); lcd.clear();
    lcd.print("STOP, COUVEJ");
    Serial.print("STOP, COUVEJ"); delay(1000);}
//ověření zda jsou na seriové lince dostupná data
if(Serial.available() > 0)
{
    data = Serial.read(); //přečtení a uložení dat do proměné "data"
    if(data>79){rychlost=data;} //oddělení dat rychlosti od dat řídicích povelů
    //ovládání LED na pinu 13 jedním tlačítkem pomocí pomocné proměnné
    if(data==7) {if(ledstatus==0){digitalWrite(13, HIGH); ledstatus=1;}
        else {digitalWrite(13, LOW); ledstatus=0;}
    }
    //soustava podmínek pomocí kterých se robot pohybuje
    if((data==1) && (vzdalenost>20)){dopredu();}
    else if(data==2){dozadu();}     else if(data==3){doprava();}
    else if(data==4){doleva();}     else {zastav();}
}
}
```

Ověření funkčnosti

Zapněte robota. Ten by měl téměř okamžitě začít vypisovat na displeji nastavenou rychlost a vzdálenost od překážky. Připojte své mobilní zařízení k vozítku pomocí Bluetooth. Na displeji mobilního zařízení by se měly zobrazovat naměřené veličiny. Pomocí tlačítek ověřte, zda robot jezdí požadovaným směrem. Vyzkoušejte, zda se zapíná a vypíná LED pomocí tlačítka v aplikaci.



Obr. 33- Výsledek finální úlohy

16.4 Úkoly k vyřešení

1. Vozítko při požadavku jízdy vpřed nejede rovně a chyba není způsobena konstrukcí podvozku. Jak chybu diagnostikujete a odstraníte?

ZÁVĚR

Z literární rešerše vyplývá, že je dostatek kvalitní literatury pro učitele, kteří by chtěli své hodiny vést jako projektovou výuku. Online zdroje jsou na tom hůře a většina stránek se zabývá projektovou výukou jen okrajově. Zdrojů ohledně Arduina i příslušenství je díky rozsáhlé komunitě dostatek, jak v češtině, tak i anglicky. Zahraniční knihy jsou poměrně špatně dostupné jak z hlediska dopravy, tak i ceny. České knihy jsou naproti tomu zdarma ke stažení.

V diplomové práci jsou popsány jednotlivé oblasti využití embedded systému Arduino, z toho nejdůležitější je pro nás nasazení ve výuce a především té praktické. Díky projektům žáci pojmu informace z více oblastí zábavnou formou. V teoretické části diplomové práce jsou popsány výhody a nevýhody projektové výuky, při čemž pozitiva převládají nad negativy, které mají dopad především na učitele a to především na náročnost přípravy na výuku.

V praktické části jsou navrženy úlohy, které splňují požadavky na projekt a to především to, že směřují ke konkrétnímu cíli. Jednotlivé úlohy jsem realizoval a otestoval tak, aby bylo jisté, že jsou plně funkční a žáci se nesetkají s problémy při zapojování. Finální projekt jsem realizoval, otestoval a následně prezentoval na soutěži STOČ 2018 v Ostravě. Mnoho dalších soutěžních příspěvků bylo realizováno pomocí platformy Arduino, což ukazuje na fakt, že se jedná o velmi oblíbenou platformu. Studenti, kteří budou realizovat projekty právě na této platformě, nemusí své projekty dělat zbytečně, ale mohou se s nimi zúčastňovat podobných soutěží.

Praktickou část se bohužel nepodařilo ověřit v praxi, kvůli velké časové náročnosti. Jako začínající pedagog bohužel nemám možnost měnit současné ŠVP pro aktuální školní rok a jiná témata není možné do výuky začlenit, natož měnit celou koncepci výuky. I přes to se podařilo v praxi ověřit alespoň krátkodobě praktickou formu výuky, kdy si studenti volili vlastní témata prezentací a byl jim poskytnut jen nezbytně nutný teoretický základ. Úkolem bylo si vše samostatně nastudovat a realizovat. Ukázalo se, že studenti mají o výuku větší zájem a projeví vlastní iniciativu, pokud je jim dána jistá volnost a za výsledek mají vlastní zodpovědnost.

SEZNAM POUŽITÉ LITERATURY

- [1] DVOŘÁKOVÁ, Markéta. Projektové vyučování v české škole: vývoj, inspirace, současné problémy. Praha: Karolinum, 2009. ISBN 9788024616209.
- [2] KRATOCHVÍLOVÁ, Jana. Teorie a praxe projektové výuky. Brno: Masarykova univerzita, 2006. ISBN 978-80-210-4142-4.
- [3] VOBECKÝ, Jan a Vít ZÁHLAVA. Elektronika: součástky a obvody, principy a příklady. 3., rozš. vyd. Praha: Grada, 2005. ISBN 80-247-1241-5.
- [4] What is Arduino? Arduino [online]. 2018 [cit. 2018-02-04]. Dostupné z: <https://www.arduino.cc/en/Guide/Introduction>
- [5] ARDUINO UNO REV3. Arduino [online]. 2018 [cit. 2018-01-29]. Dostupné z: <https://store.arduino.cc/arduino-uno-rev3>
- [6] ARDUINO UNO WITH STARTER KIT DELUXE BY ODDWIRES. Oddwires [online]. [cit. 2018-03-31]. Dostupné z: <http://www.oddwires.com/arduino-starter-kit-deluxe/>
- [7] LEGO MINDSTORMS Education EV3 Core Set. Lego education [online]. [cit. 2018-03-31]. Dostupné z: <https://education.lego.com/en-us/products/lego-mindstorms-education-ev3-core-set-/5003400>
- [8] PWM. Arduino [online]. [cit. 2018-03-31]. Dostupné z: <https://www.arduino.cc/en/Tutorial/PWM>
- [9] Analogový signál Zdroj: https://it-slovník.cz/pojem/analogovy-signal/?utm_source=cp&utm_medium=link&utm_campaign=cp. IT slovník [online]. [cit. 2018-05-02]. Dostupné z: <https://it-slovník.cz/pojem/analogovy-signal>
- [10] Digitální signál. IT slovník [online]. [cit. 2018-05-02]. Dostupné z: <https://it-slovník.cz/pojem/digitalni-signal>
- [11] Digitální signál. Wikipedia [online]. 2017 [cit. 2018-05-02]. Dostupné z: https://cs.wikipedia.org/wiki/A/D_p%C5%99evodn%C3%ADk
- [12] Baud. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2018-05-02]. Dostupné z: <https://cs.wikipedia.org/wiki/Baud>

- [13] Bit. In: Wikipedia: the free encyclopedia [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2018-05-02]. Dostupné z: <https://cs.wikipedia.org/wiki/Bit>
- [14] Circuits has arrived on Tinkercad. TINCERCAD [online]. [cit. 2018-05-03]. Dostupné z: <https://www.tinkercad.com/>
- [15] ARDUPLC V2. IRAY [online]. [cit. 2018-05-04]. Dostupné z: <http://tienda.ray-ie.com/home/95-arduplc-v2.html>
- [16] TINKERKIT BRACCIO ROBOT. ARDUINO [online]. [cit. 2018-05-04]. Dostupné z: <https://store.arduino.cc/tinkerkit-braccio>
- [17] Teaching, Inspiring and Empowering!. ARDUINO [online]. [cit. 2018-05-04]. Dostupné z: <https://www.arduino.cc/en/Main/Education>
- [18] MALÝ, Martin. Hradla, volty, jednočipy: úvod do bastlení. Praha: CZ.NIC, z.s.p.o., 2017. CZ.NIC. ISBN 978-80-88168-23-2.
- [19] Download the Arduino IDE. Arduino [online]. [cit. 2018-05-13]. Dostupné z: <https://www.arduino.cc/en/main/software>
- [20] Arduino Products. Arduino [online]. [cit. 2018-05-13]. Dostupné z: <https://www.arduino.cc/en/Main/Products>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

RVP Rámcový vzdělávací program.

ŠVP Školní vzdělávací program

USB Univerzální sériová sběrnice.

LED Light emitting diode.

PWM Pulzně šířková modulace.

3D Trojrozměrný.

SEZNAM OBRÁZKŮ

Obr. 1- schéma projektu.....	15
Obr. 2- Arduino produkty [20]	19
Obr. 3- Arduino UNO	20
Obr. 4- Stavebnice Arduino [6]	24
Obr. 5- Stavebnice Lego Mindstorms [7]	24
Obr. 6- Tincercad [14]	25
Obr. 7- prostředí MIT App Inventor	26
Obr. 8- programovací bloky aplikace MIT App Inventor	27
Obr. 9- Arduino cloud.....	28
Obr. 10- Arduino education	28
Obr. 11- Braccio robot [16]	30
Obr. 12- ArduPLC [15].....	31
Obr. 13- CTC 101 [17]	31
Obr. 14- stažení a instalace 1 [19]	35
Obr. 15- stažení a instalace 2 [19]	36
Obr. 16- stažení a instalace 3 [19]	36
Obr. 17- prostředí arduino IDE.....	37
Obr. 18- výběr součástek	38
Obr. 19- schéma zapojení LED.....	39
Obr. 20- ukázka kódu.....	39
Obr. 21- Popis Arduino desky	43
Obr. 22- schéma zapojení DHT22	46
Obr. 23- schéma zapojení HC-SR04.....	46
Obr. 24- schéma zapojení DC motoru	50
Obr. 25- Schéma zapojení servo motoru	50
Obr. 26- schéma zapojení relé	54
Obr. 27- Sériové zapojení	56
Obr. 28-schéma zapojení RFID	60
Obr. 29- Schéma zapojení bluetooth modulu	64
Obr. 30- Mobilní aplikace.....	67
Obr. 31- Program vytvořený pomocí bloků.....	68
Obr. 32- Schéma zapojení.....	71

Obr. 33- Výsledek finální úlohy 75

SEZNAM TABULEK

Tab. 1- Parametry Arduino desky [5]	21
Tab. 2- Porovnání Arduino vs. LEGO Mindstorms.....	23
Tab. 3- informace o projektu	33
Tab. 4- základní operátory	40
Tab. 5- základní elektronické součástky.....	43

SEZNAM PŘÍLOH

P I CD obsahující diplomovou práci a složky se schémata a programy