



Vyšší odborná škola  
a Střední průmyslová škola elektrotechnická  
Plzeň, Koterovská 85

# ROČNÍKOVÁ PRÁCE

Téma: Programovatelné auto

Autor práce:	Jan Ocelík
Obor studia:	78-42-M/01 Technické lyceum
Třída:	3. L
Předmět:	Kybernetika
Zadávací učitel:	Jiří Švihla
Dne:	28. 4. 2023

Hodnocení:



**Vyšší odborná škola a  
Střední průmyslová škola elektrotechnická  
Plzeň. Koterovská 85**

<b>ZADÁNÍ ROČNÍKOVÉ PRÁCE</b>	
Školní rok	2022/ 2023
Studijní obor	78-42-M/01 Technické lyceum
Jméno a příjmení	Jan Ocelík
Třída	3.L
Předmět	Kybernetika
Hodnoceno v předmětu	Kybernetika
Téma	Programovatelné auto
Obsah práce	<ol style="list-style-type: none"><li>1. Sehnání součástek</li><li>2. Design a tisk modelu (šasi auta)</li><li>3. Sestavení prototypu</li><li>4. Vytvoření Framework/SDK/API</li><li>5. Naprogramování</li></ol>
Zadávací učitel Příjmení, jméno	Švihla Jiří
Podpis zadávajícího učitele	
Termín odevzdání	28. dubna 2023

## Anotace

Cílem této ročníkové práce je za pomoci opensource programů navrhnout a z opensource komponent poskládat robotické autíčko pro začátečníky i pokročilé s možností jednoduchého sestavení. Dalším úkolem je vyvinout firmware a API pro jednoduché skriptové programování i komplexnější programování s využitím obrazového vstupu. Dalším úkolem je připravit ukázkové příklady kódu k předvedení jednotlivých funkcí robota. Posledním úkolem je zpříjemnit práci s robotem, zhodnotit přínosy a možnosti využití projektu ve vzdělávání a vypracovat potřebnou dokumentaci.

„Prohlašuji, že jsem tuto práci vypracoval samostatně a použil literárních pramenů a informací, které cituji a uvádím v seznamu použité literatury a zdrojů informací.“

„Souhlasím s využitím mé práce učteli VOŠ a SPŠE Plzeň k výuce.“

Plzni dne: ..... Podpis: .....

## Poděkování

Tímto bych chtěl poděkovat vedoucímu práce Jiřímu Švihlovi za pomoc s výběrem komponent a obsahem dokumentace a rodině a přátelům za psychickou podporu. Také děkuji všem dohromady za to, že mě k tomu včas dokopali.

# Obsah

<b>Anotace</b>	<b>2</b>
<b>Poděkování</b>	<b>3</b>
<b>Obsah</b>	<b>4</b>
<b>Úvod</b>	<b>5</b>
<b>1 Cíle a požadavky</b>	<b>6</b>
1.1 Komponenty (Moduly) . . . . .	6
1.2 Šasi (tělo) . . . . .	6
1.3 API . . . . .	6
<b>2 Návrhový software</b>	<b>7</b>
2.1 Onshape . . . . .	7
2.2 Code OSS/VS Code . . . . .	9

# Úvod

V dnešní době se stále více hovoří o automatizaci a digitalizaci a tyto trendy mají velký vliv na společnost. Robotika a autonomní systémy jsou jednou z klíčových oblastí, které se rozvíjejí v rámci těchto trendů a mají potenciál změnit mnoho aspektů našeho života.

Vzdělávání a výuka v této oblasti se také stávají stále důležitějšími, protože mnoho pracovních pozic, které budou v budoucnosti vyžadovat znalosti robotiky a programování, ještě neexistuje. Výuka v této oblasti tak může být klíčová pro přípravu studentů na pracovní trh budoucnosti.

Proto jsem se rozhodl věnovat svou ročníkovou práci právě problematice robotiky a vytvořit autonomní auto, které bude sloužit jako výuková pomůcka. Můj záměr je ukázat, jak moderní technologie mohou být využity k tomu, aby byly studenti lépe připraveni na budoucí výzvy v oblasti robotiky a informatiky.

Při tvorbě prototypu se budu snažit využít nejnovější poznatky v oblasti robotiky a programování, a to jak z teoretického hlediska, tak z praktického testování. Cílem bude vytvořit zařízení, které bude snadno ovladatelné a srozumitelné pro studenty různých věkových kategorií, a zároveň bude dostatečně funkční a výkonné pro splnění zadaných úkolů.

Věřím, že tato ročníková práce přispěje k popularizaci robotiky a podpoří zájem studentů o tuto oblast. Tím může mít pozitivní vliv na jejich budoucí kariéru a na rozvoj robotiky v České republice.

# 1 Cíle a požadavky

## 1.1 Komponenty (moduly)

Hlavním cílem této ročníkové práce je návrh a výroba robota ze všem dostupných materiálů, v tomto případě ze snadno sehnatelných modulů.

Moduly by měly být zároveň opensource, aby si software pro ně mohl programovat každý, který se k robotovi dostane, nebo si ho sám vyrobí.

Důležitý je také výběr napájení, v případě robotického auta tedy baterie, tak, aby udrželo robota v provozu po přiměřeně dlouhou dobu a aby zároveň dokázalo vykrýt proudové špičky.

## 1.2 Šasi (tělo)

Tělo robota by mělo být vyrobeno tak, aby bylo plně uzavíratelné, tedy bezpečné pro jakýkoli přenos i při nešetrném zacházení, ale aby se dal robot také provozovat s plně přístupnými veškerými propojkami a bylo tedy snadné ladit jeho hardware za provozu.

## 1.3 API

Jedním z dalších požadavků je co nejvíce zpříjemnit uživateli práci s robotickým autem, tedy vytvořit nějaké API nebo knihovnu.

Nedílnou součástí je také dokumentace k danému API/knihovně a to proto, aby si i nově příchozí uživatel zvládl bez jakékoli pomoci robota pomocí daného API/knihovny naprogramovat.

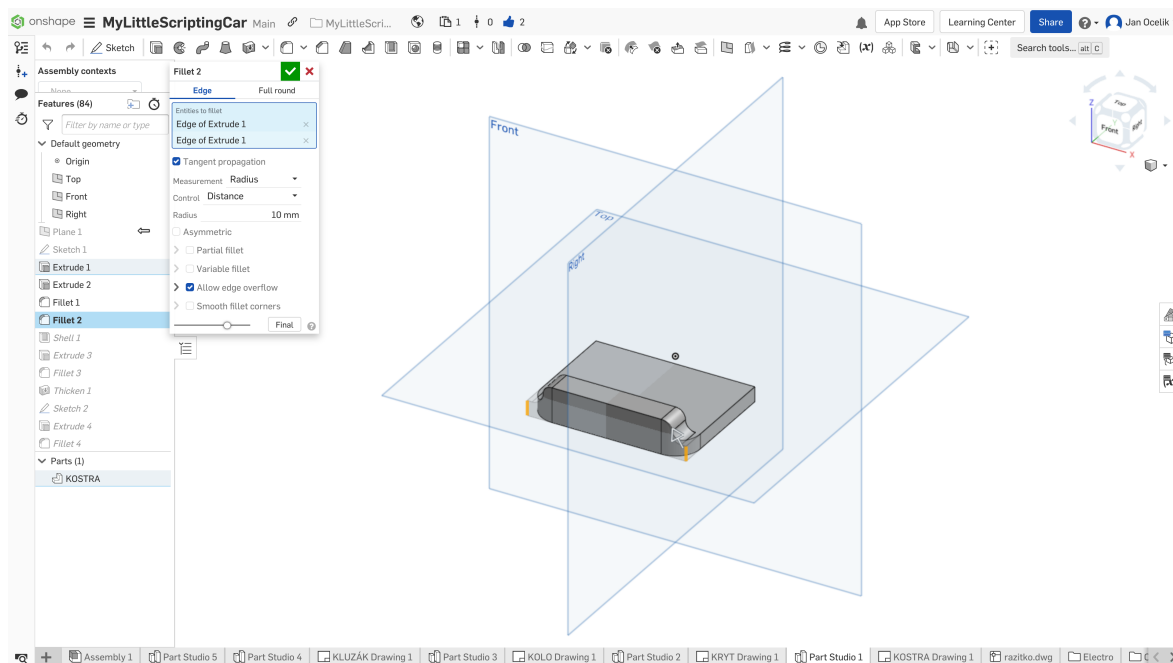
## 2 Návrhový software

### 2.1 Onshape

Onshape je online software pro návrh 3D součástí. V programu se vytváří takzvaný dokument, ve kterém se nadále dají zakládat „studia“. Onshape nabízí mnoho zajímavých možností, typického uživatele však zajímají tato tři hlavní studia - **Part Studio**, **Assembly** a **Drawing**.

#### Part Studio

Part Studio slouží, jak už název napovídá, k vytváření jednotlivých dílů. Díly se vytvářejí a upravují pomocí příkazů, jako například **Sketch**, sloužící k sestrojení 2D náčrtu, **Extrude**, k „vytáhnutí“ náčrtu do třetího rozměru, nebo **Fillet**, k zaoblení hran. Tyto příkazy lze vyvolat kliknutím na jejich ikonu v horní liště programu, přepsáním jejich názvu do **Search tools**, nebo, u některých více používaných, klávesovými zkratkami.

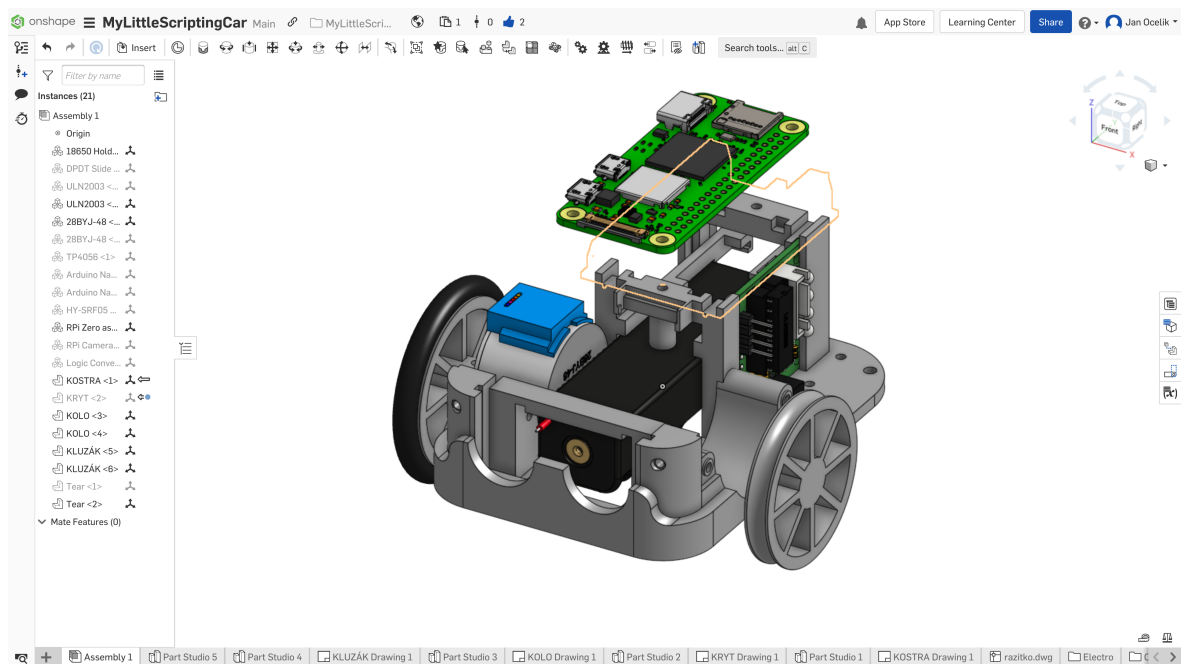


Obrázek 1: Part Studio



## Assembly

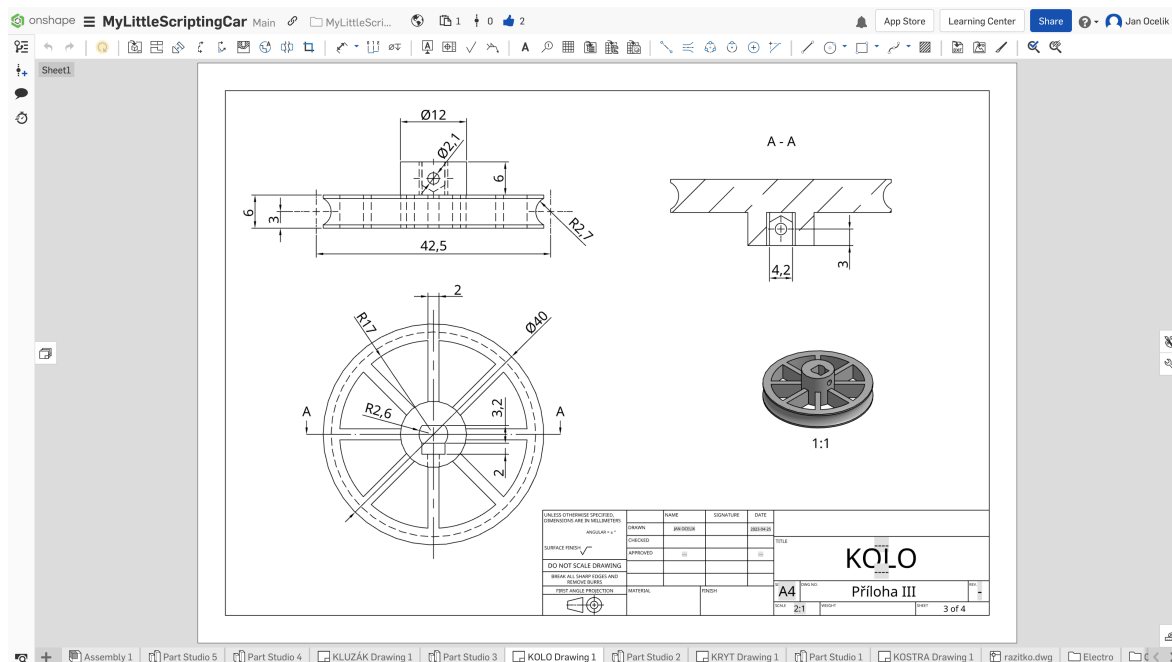
Další studio v programu Onshape, Assembly, slouží ke spojování (nebo alespoň aranžování) jednotlivých dílů. K tomuto účelu slouží příkazy, které se, opět, nechají vyvolat několika způsoby, včetně klepnutím na ikonu v horní liště studia. Další výhodnou funkcí tohoto studia je **Create Part Studio in context**, což dělá přesně to, co se v názvu píše; vytvoří studio dílu v kontextu se sestavou. Tato funkce může hodně usnadnit vytváření některých dílů.



Obrázek 2: Assembly

## Drawing

Drawing se dá vyvolat jak samostatně, tak rovnou v kontextu s daným dílem a to jak z Part Studio, tak z Assembly. Uživateli stačí, kdy v seznamu dílů vybere danou položku a přes kontextové menu vybere možnost **Create Drawing of...** Výkres se zde upravuje také pomocí příkazů, stejně jako v předešlých studiích. **Drawing** studio umožňuje také globální nastavení kót, písma, čar atd.



Obrázek 3: Drawing

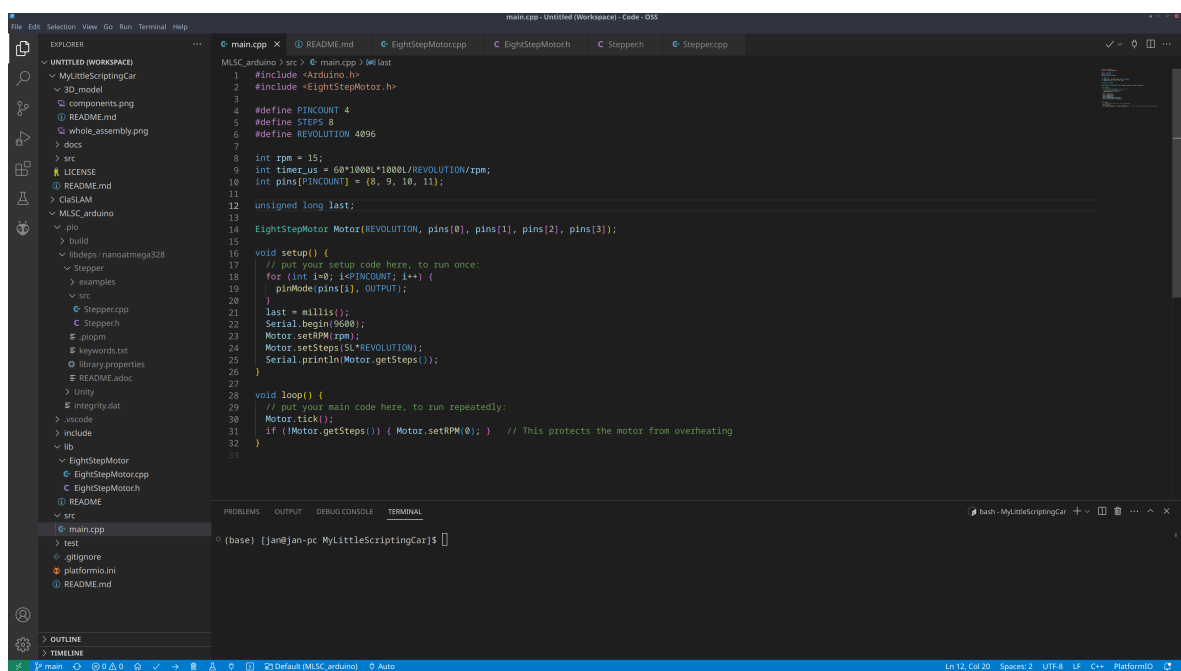
## 2.2 Code OSS/VS Code

Code OSS je opensource IDE vyvíjené firmou Microsoft. Vzniklo jako konkurent programu Atom, který po nějaké době rozválcovával. Stal se tak oblíbeným hlavně díky podpoře obrovského množství programovacích i značkových jazyků (např. python, c++, java, php, xml, html, markdown). Má také velice intuitivní ovládání a šikovně zvolenou paletu barev.

Code OSS a Visual Studio Code jsou téměř identické programy, jediným pozorovatelným rozdílem je absence některých rozšíření v „marketplace“ v Code OSS (i to se dá ale snadno opravit).

## PlatformIO

Platformio je rozšíření do VS Code umožňující překlad C++ kódu a následný upload do předdefinovaných jednočipů (např. atmega328). Je to šikovný kus softwaru s několika užitečnými funkcemi navíc, jako např. automatická detekce zařízení.



Obrázek 4: Code OSS