STŘEDNÍ PRŮMYSLOVÁ ŠKOLA  
MLADÁ BOLESLAV

**ROČNÍKOVÁ PRÁCE**

Adam Bartoš

Mladá Boleslav 2024

STŘEDNÍ PRŮMYSLOVÁ ŠKOLA  
MLADÁ BOLESLAV

**ROČNÍKOVÁ PRÁCE**

**Autor: Adam Bartoš  
Studijní obor: 18-20-M/01 Informační technologie  
Vedoucí práce: Lukáš Mázl**

Mladá Boleslav 2024

# Obsah

[Obsah 3](#_Toc164951471)

[1 Úvod 6](#_Toc164951472)

[2 Použité technologie 7](#_Toc164951473)

[2.1 Elektronické součásti pavouka 7](#_Toc164951474)

[2.2 Materiály tisku 7](#_Toc164951475)

[2.3 ESP32 8](#_Toc164951476)

[2.4 Python a MicroPython 8](#_Toc164951477)

[2.5 I2C modulový driver 8](#_Toc164951478)

[2.6 Servo motor 9](#_Toc164951479)

[3 Schéma zapojení 10](#_Toc164951480)

[4 Sestavení pavouka 10](#_Toc164951481)

[4.1 Tělo pavouka 10](#_Toc164951482)

[4.2 Konstrukce nohou, držáků a konektorů k nim 14](#_Toc164951483)

[5 Programování 19](#_Toc164951484)

[5.1 FLASH MicroPythonu 19](#_Toc164951485)

[5.2 Vysvětlení funkcí jednotlivých souborů 19](#_Toc164951486)

[6 Řešení problematiky 21](#_Toc164951487)

[6.1 Chyba tisku 21](#_Toc164951488)

[6.2 Udržení elektroniky 21](#_Toc164951489)

[7 Závěr 22](#_Toc164951490)

[8 Přílohy 23](#_Toc164951491)

[8.1 Seznam obrázků 23](#_Toc164951492)

[8.2 Zdroje 24](#_Toc164951493)

**Prohlášení**

Prohlašuji, že jsem svou ročníkovou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v přiloženém seznamu.

Nemám závažný důvod proti zpřístupňování této ročníkové práce v souladu se zákonem č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) v platném znění.

V Mladé Boleslavi dne podpis:

# Úvod

Smyslem této ročníkové práce bylo sestavit a naprogramovat pavouka, který zahýbe jednou nohou a zamává pomocí vytvořené funkce. V projektu byla použita technologie programovacího jazyka MicroPython, jednotky ESP32 a servo motorů MG9952. Celý pavouk vznikl pomocí propojení servo motorů a 3D tisku. [[1]](#footnote-1)Funkce byly testovány v programu PuTTY[[2]](#footnote-2). Následně po otestování bylo potřeba kód nahrát do editoru zdrojového kódu Visual Studio Code [[3]](#footnote-3)a vytvořit pomocí programovacího jazyka Python internetové rozhraní pomocí WIFI propojení, aby pavouk mohl zvednout nohu při jednoduchém kliknutí tlačítka v prohlížeči telefonu.

# Použité technologie

## Elektronické součásti pavouka

Ve středu pavouka je umístěna jednotka ESP32 společně s I2C modulovým driverem pro připojení servo motorů, regulátor napětí, [[4]](#footnote-4)napájecí kabely a baterie 9 V. Regulátor napětí byl použit z důvodu, aby nebyla spálena jednota ESP32, jelikož vložené baterie mají napětí 9 V.

## Materiály tisku

Na díly pro pavouka byla využita 3D tiskárna. Pro větší efektivitu a lepší výdrž i bytelnost byl použit materiál PLA a PET-G. Ze začátku byl používán materiál PLA, ale tento materiál po roce používání ztrácí sílu a je biologicky rozložitelný. Bylo tedy lepší použít materiál tisku PET-G a díly nahradit.

#### PLA

První nejčastěji používaný materiál pro 3d tisk je PLA. Nejlepší volba pro hobby tisknutí. Materiál se vyrábí z kukuřičného, bramborového škrobu či cukrové třtiny. Je rozpustný různými kyselinami, jako je například alkohol. PLA materiál má také lepší a optimálnější teplotu na rozehřívání, šetří energii a je pevný.

#### PET-G

Druhý nejčastěji používaný materiál pro 3D tisk je PET-G, který má vyšší teplotní odolnost než PLA, je pružnější a odolnější. Je také odolný vůči kyselinám a rozpouštědlům, na rozdíl od materiálu PLA.

## ESP32

ESP32 je Soc mikrokontroler[[5]](#footnote-5) vhodný pro širokou škálu aplikací, od nízkonapěťových až po náročnější úkony. Obsahuje čip ESP32-DOWDQ. Integrovaný čip je navržen tím způsobem, aby se dal rozšiřovat a přizpůsobit. Frekvence je nastavitelná od 80Mhz po 240Mhz. ESP integruje bohatou sadu periferií[[6]](#footnote-6), jako je například snímač pohybu, vzdálenosti, dotyku a podobně. Jednotka ESP32 také obsahuje Wi-Fi a Bluetooth, které bylo pro vytvoření rozhraní důležité a bez její součásti by nebylo možné rozhraní vytvořit.

## Python a MicroPython

Python a MicroPython jsou programovací jazyky. Rozdíl mezi programovacím jazykem Python a MicroPython je takový, že Python je používán pro psaní kódu, který běží na procesoru, zatímco MicroPython je určen pro to, aby byl kompatibilní s jednotkami jako je ESP32. MicroPython byl použit z důvodu toho, že je optimalizovaný pro mikrokontrolery.

## I2C modulový driver

Pro zapojení všech servo motorů byl použit I2C modulový driver servo motoru PCA9685, který umožňuje připojit 16 servo motorů či jiného zařízení. Rozsah napětí je 2.3 V až 5.5 V. Tento modul využívá komunikaci přes I2C sběrnici[[7]](#footnote-7), což znamená, že může být připojen k mikrokontroleru (jako je Arduino[[8]](#footnote-8)) pouze pomocí dvou pinů (SDA [[9]](#footnote-9)a SCL[[10]](#footnote-10)).

## Servo motor

Servomotor je druh rotačního nebo lineárního aktuátoru[[11]](#footnote-11), který umožňuje přesné řízení polohy, rychlosti a zrychlení v mechanickém systému. Servomotory jsou součástí servomechanismu[[12]](#footnote-12) a skládají se z vhodného motoru spojeného se senzorem pro zpětnou vazbu polohy a řadiče[[13]](#footnote-13). Servomotor je uzavřený servomechanismus, který používá zpětnou vazbu polohy (buď lineární nebo rotační) k řízení svého pohybu a konečné polohy. Vstupem do jeho řízení je signál (buď analogový nebo digitální), který reprezentuje požadovanou polohu výstupní hřídele.

# Schéma zapojení

Pavouk je zapojený pomocí ESP32, modulového driveru a servo motorů, které vedou do modulových driverů. Na poslední testování byl přidán jeden modulový driver navíc a to proto, aby pavoukovi stálo všech šest nohou, na které potřeba 18 servo motorů, kde jeden driver má pouze 16 míst. Z modulových driverů vedou dva kabely na připojení elektřiny z baterie, vždy je plus a mínus na jeden modulový driver. Dále je do PCA driveru zapojen servo motor, a to devětkrát na jeden modulový driver, pro levou a pravou stranu pavouka, který má tři nohy na každé straně a tři servo motory na jedné noze. ESP32 je z druhé strany také napájeno kabelem 3V3, aby bylo zapnuté. Z modulových driverů jde pomocí komunikace I2C a kabely DuPont vše do ESP32 desky, která vytvoří Wi-Fi, na kterou je možné se připojit mobilním telefonem.

Obsah obrázku text, skica, kresba, Dětské kresby

Popis byl vytvořen automaticky

Obrázek 1: Schéma zapojení

# Sestavení pavouka

Na sestavení pavouka je potřeba zručnost, trpělivost a dostatek času. Je potřeba brát ohled na to, že díly na pavouka jsou všechny, krom servo motorů tištěné na 3D tiskárně a nejsou nějak zvláště odolné. Dále je potřeba všechny díly postupně tisknout, kde například jedna noha se tiskne 6 až 8 hodin, zde ale záleží na typu 3D tiskárny a tloušťce trysky. Čím menší a přesnější tryska, tím více času zabere tisknutí dílu. Menší díly samozřejmě trvají menší čas. Proto byl ale zapotřebí čas. Jen samotné vytváření dílů na pavouka trvalo měsíc. Začalo se velkými částmi těla, jako byla například spodní část těla, která byla výhradně určena pro držení všech elektrických součástek, také horní část, pro zakrytí součástek, aby pavouk měl lepší vzhled a byl kryty i shora. Dále také nohy. Celkem je na pavoukovi 6 nohou, které se ale tiskly vícekrát, protože bylo potřeba upravovat konstrukci nohy podle váhy celého pavouka poté, co do těla byla umístěna všechna potřebná technika. Jako další následovaly díly, které držely celé tělo pohromadě a jako poslední menší dílky pro spojení servo motorů s držákem nohy a tělem.

## Tělo pavouka

Po vytištění všech dílu je možné přistoupit na stavění. K sestavení je také potřeba 18\* servo motor, protože díly, které drží nohu jsou modelované tak, aby v každém držáku byl umístěn servo motor, který hýbe s nohou a zároveň ji drží. Dále jsou potřeba šrouby M2,5 a závitové vložky do plastu na pozdější upevnění na šrouby do dílů. Pokud je připravené vše, přesuňme se na samostatné sestavení, které začíná tělem pavouka. Jako první vezmeme spodní díl pavouka a sešroubujeme ho s druhým, tím stejným dílem, kde vznikne první spodní díl pavouka.

Obsah obrázku skica, trojúhelník, design

Popis byl vytvořen automaticky

Obrázek 2: Spodní tělo pavouka

Následně je potřeba sestavit samotné tělo pavouka, které je tvořeno čtyřmi částmi, levá strana těla, pravá strana těla, zadní a přední strana. Díly na tělo pavouka se vkládají na spodní tělo pavouka a následně přišroubují. Všechny mají v horní části připravenou díru pro umístění držáku nohy a servo motorů. Dále mají viditelný vyhrazený prostor pro vedení kabelů dovnitř pavouka. Na obrázku „Tělo pavouka první část“ vidíme první část těla pavouka, kde lze vidět i díry pro piny, které uchytí držák na nohu a jeden vyhrazený prostor pro servo motor. Další díl, který je velmi podobný předchozímu, ale je otočený, a proto bude umístěn přímo naproti předchozímu dílu na druhou stranu. Vidíme na obrázku „Tělo pavouka druhá část“. Třetí část těla pavouka se skládá z dvou míst pro servo motory a zadního otvoru pro případné opravy po sestavení pavouka a místa na kabely. Čtvrtá zadní a také poslední část se skládá také z dvou míst pro servo motory a zadního otvoru pro případné opravy či připevnění jiných dílů v budoucnu. Hlavní části těla se k sobě připevňují pomocí vytištěných přípojek, na které má každá část místo na boku. Vtlačí se do části hlavního těla a poté spojí k sobě. Takto se to udělá u každé části hlavní konstrukce.

Obsah obrázku design

Popis byl vytvořen automaticky se střední mírou spolehlivosti

Obrázek 3: Pin pro upevnění těla

Obsah obrázku design, snímek obrazovky, Obdélník, potisk

Popis byl vytvořen automaticky

Obrázek 4: Tělo pavouka první část

Obsah obrázku černobílá, design

Popis byl vytvořen automaticky

Obrázek 5: Tělo pavouka druhá část

Obsah obrázku černobílá, skica, design

Popis byl vytvořen automaticky

Obrázek 6: Tělo pavouka třetí část

Obsah obrázku design, černobílá

Popis byl vytvořen automaticky se střední mírou spolehlivosti

Obrázek 7: Tělo pavouka čtvrtá část

## Konstrukce nohou, držáků a konektorů k nim

Po spojení všech dílů středu těla umístíme do každého prostoru vyhrazeného pro servo motory servo motor a přesuneme se na konstrukci nohou. Nohy jsou poslední část sestavení pavouka, kde je jako první potřeba připojit držáky nohou na již sestavené tělo pavouka. Držáky jsou rozděleny na levou a pravou stranu, která se liší pozicí vyčnívajícího pinu pro připojení do těla. Na obrázku „Pravý držák nohy“ vidíme, jak vypadá tento držák určený pro pravou stranu těla. Celkem potřebujeme tři díly levého i pravého držáku, na každou nohu jeden, celkem tedy 6 nohou a 6 držáků.



Obrázek 8: Pravý držák nohy

Na druhou stranu umístíme držák určený pro levou stranu, který lze vidět na obrázku „Levý držák nohy“.

Obsah obrázku černobílá, design, záchod, koupelna

Popis byl vytvořen automaticky

Obrázek 9: Levý držák nohy

Pokud máme připravené držáky nohou, vložíme do každého z nich servo motor, který přišroubujeme šrouby v balení společně s dílem pro připojení nohy, který je na obrázku „Spojení nohy a držáku nohy“. Tento díl, který spojuje držák nohy a nohu bude potřeba 12\*. Servo motor slouží nejen pro pohyb, ale také pro spojení držáku nohy a samostatné nohy. Do každého držáku umístíme servo motor, celkem tedy bude 6 servo motorů v držáku. Na vyčnívající pin na straně držáku umístíme díl „Spojení nohy a držáku nohy“ a doprostřed tohoto dílu umístíme díl pro zpevnění konstrukce u nohou, nazván „Spojení mezi držáky“, aby se spojení nerozpadalo a působilo bytelněji. Po tomto kroku můžeme pokračovat na samostatné umístění nohy. V samotné noze je místo vyhrazené pro servo motor, kam ho umístíme. Opět přiděláme servo motor šrouby v balení společně s „Spojení nohou a držáku nohy“ a vznikne nám první noha. Noha má na straně také vyhrazený vyčnívající pin pro umístění „Spojení nohy a držáku nohy“, stejně jako držák nohy levý i pravý. Vyčnívající pin můžeme vidět na obrázku „Levá noha druhý pohled“ a „Pravá noha druhý pohled“. A to proto servo motor k držákům šroubujeme vždy na jedné straně a na druhé straně připevníme tento držák vyhrazeným vyčnívajícím pinem. Postup u každého držáku umístěného v těle takto opakujeme a umisťujeme postupně nohy. Po umístění všech nohou se dostaneme k výsledku a vidíme sestaveného pavouka, který obsahuje 18 servo motorů. Pavouk nemá umístěnou horní část na zakrytí a lepší vzhled a to proto, aby se lépe manipulovalo s elektronikou umístěnou uvnitř a po každém předělávání či zapojování nebo odpojování servo motorů nemusela znovu šroubovat. Na konci se umístí tato část, která lze vidět na obrázku „Zakrytí pavouka“.

Obsah obrázku snímek obrazovky, design

Popis byl vytvořen automaticky

Obrázek 10: Spojení nohy a držáku nohou

Obsah obrázku snímek obrazovky, Obdélník, text, design

Popis byl vytvořen automaticky

Obrázek 11: Spojení mezi držáky

Obsah obrázku oranžová, design

Popis byl vytvořen automaticky

Obrázek 12: Levá noha druhý pohled

Obsah obrázku oranžová

Popis byl vytvořen automaticky s nízkou mírou spolehlivosti

Obrázek 13: Pravá noha druhý pohled

Obsah obrázku černobílá, design

Popis byl vytvořen automaticky se střední mírou spolehlivosti

Obrázek 14: Levá noha

Obsah obrázku černobílá, černobílý

Popis byl vytvořen automaticky

Obrázek 15: Pravá noha

# Programování

## FLASH MicroPythonu

Jako první část je potřeba na začátek stáhnout firmware, který obsahuje binární soustavu společně s MicroPythonem. Vše najdeme na oficiální stránce Micropython.org, kde je v návodech napsáno, jak postupovat, aby bylo vytvořené rozhraní a bylo možné ovládat přes REPL[[14]](#footnote-14), v našem případě PuTTY. Dále je ale potřeba dostat složky z Visual Studio Codu na samotné ESP, a to probíhalo pomocí flashnutí souboru. Jeden takový příklad vypadal takto: “ampy --port COM03 put“ a za put se dodá cesta pro soubor, který chceme nahrát na ESP, například config.py. Pokud chceme zjistit, jaké soubory se nachází v ESP, použijeme příkaz: “ampy --port COM03 ls“ a vypíšou se všechny složky, které jsou vložené na desce. COM03 značí v tomto případě místo připojení do počítače.

## Vysvětlení funkcí jednotlivých souborů

Jako první se v programu nachází soubor boot.py, který automaticky vytvoří Access Point[[15]](#footnote-15), spustí se jako první a nemusí se volat funkcí a udává cesty ke knihovnám. Dále se zde nachází soubor main.py, který má dva módy kdy má být server zapnutý a kdy vypnutý. Jako další také soubor controller.py, který vytvoří síťový soket[[16]](#footnote-16), který když přijde jakýkoliv signál, vyvolá funkci wave, která zamává. Hlavní část zdrojového kódu tvoří soubor wave.py, kde probíhá inicializace I2C sběrnic, které jsou napojené na 2piny (SDA, SCL). Po inicializaci celkové se inicializují také I2C modulové drivery, které předají referenci sběrnicím. Dále je nastavená frekvence, a vytvořené proměnné jako instance servo motoru. Se servo motory se pracuje tak, že jim je nastavený přesný úhel, o který se má pohnout. V tomto případě je zde list wave, kde každé servo má naměřený úhel tak, aby pavouk stál. Dále se zde nachází reset, který provádí inicializaci pavouka do polohy stání. Jako poslední je zde samotná funkce wave, která má nastavené hodnoty tak, aby jedna noha zamávala a na konci se vrátila do původní pozice díky zavolání funkce reset.

# Řešení problematiky

## Chyba tisku

Ačkoliv projekt zní jednoduše, následný problém nebyl jednoduchý vyřešit. První problém nastal již při sestavování pavouka, kdy se 3D tisk pozastavil z důvodu špatného formátování tisku a v trysce se zasekl a vychladl celý materiál na daný tisknutý díl. To znamená, že se nedal odstranit. Prvně se tedy řešilo, jak se díly opraví, jestli je vůbec možné opravit díl anebo je potřeba koupit celé nové tělo. Bylo rozhodnuto vyřešit problém bez nových dílů. Proto bylo potřeba celkové čistění tiskárny, kde bylo zapotřebí odřezat větší části zaseknutého materiálu a následně vyjmout trysku společně s motorem a postupně opatrně nahřívat a odendávat části, které jsou horké natolik, aby byly možné bez poškození odejmout z těla tiskárny. Po postupném sundávání materiálu zůstalo čisté tělo. Zkusilo se tedy nainstalovat zpět do tiskárny a dát testovací tisk. Tisk proběhl úspěšně a problém byl opraven, mohlo se tedy pokračovat dále.

## Udržení elektroniky

Po úspěšném tisku všech dílů bylo zjištěno, že tělo neudrží elektroniku, protože model pro tisk nebyl upravený podle představ. A proto bylo zapotřebí upravit některé díly, jako jsou spodní část těla, kde v původním modelu byly díry, nyní vyplněny. Dále úprava nohy, opět stejný problém, při sestavování bylo zjištěno, že model neodpovídá realitě, a proto bylo potřeba upravit model pro lepší stabilitu pavouka a udržení celkové váhy. Nyní je noha upravena zpevněním a vloženou elektroniku a všechny součásti udrží.

# Závěr

Při konstruování pavouka bylo potřeba vyřešit několik nemalých problémů, které byly vždy inovativně vyřešeny a při dalším výskytu chyb opraveny. Podařilo se zkonstruovat pavouka, který má internetové rozhraní, udrží se na nohách a zamává jednou nohou. Sestavování a celková práce proběhly podle plánu, až na několik menších problémů, které se řešily vždy co nejdříve.

# Přílohy

## Seznam obrázků

[Obrázek 2: Spodní tělo pavouka 11](#_Toc164951494)

[Obrázek 3: Pin pro upevnění těla 12](#_Toc164951495)

[Obrázek 4: Tělo pavouka první část 12](#_Toc164951496)

[Obrázek 5: Tělo pavouka druhá část 13](#_Toc164951497)

[Obrázek 6: Tělo pavouka třetí část 13](#_Toc164951498)

[Obrázek 7: Tělo pavouka čtvrtá část 14](#_Toc164951499)

[Obrázek 8: Pravý držák nohy 14](#_Toc164951500)

[Obrázek 9: Levý držák nohy 15](#_Toc164951501)

[Obrázek 10:Spojení nohy a držáku nohou 16](#_Toc164951502)

[Obrázek 11: Spojení mezi držáky 16](#_Toc164951503)

[Obrázek 12: Levá noha druhý pohled 17](#_Toc164951504)

[Obrázek 13: Pravá noha druhý pohled 17](#_Toc164951505)

[Obrázek 14: Levá noha 18](#_Toc164951506)

[Obrázek 15: Pravá noha 18](#_Toc164951507)

## Zdroje

*Printables* [online]. [cit. 2024-03-21]. Dostupné z: <https://www.printables.com/cs/model/105046-3d-printed-hexapod/files>

*MicroPython* [online]. [cit. 2024-03-31]. Dostupné z: <https://micropython.org/>

*Dratek.cz* [online]. [cit. 2024-04-14]. Dostupné z: [https://dratek.cz/arduino/1686-iic-i2c-modulovy-driver-servo-motoru-pro-arduino-pca9685-16-kanalu-12-bit-pwm.html?gad\_source=1&gclid=CjwKCAjw\_e2wBhAEEiwAyFFFoygEIfKrP-nG7Ta88rvHweWh-nEjy3xmNGRgZ99bSsUYgC1\_kUexdRoCysYQAvD  
\_BwE](https://dratek.cz/arduino/1686-iic-i2c-modulovy-driver-servo-motoru-pro-arduino-pca9685-16-kanalu-12-bit-pwm.html?gad_source=1&gclid=CjwKCAjw_e2wBhAEEiwAyFFFoygEIfKrP-nG7Ta88rvHweWh-nEjy3xmNGRgZ99bSsUYgC1_kUexdRoCysYQAvD_BwE)

*PLA filament*. Online. Dostupné z: <https://www.materialpro3d.cz/materialovy-slovnik/pla/>. [cit. 2024-04-14].

*PET-G filament* [online]. [cit. 2024-04-14]. Dostupné z: <https://www.materialpro3d.cz/petg-filamenty/>

*Servo motor* [online]. [cit. 2024-04-14]. Dostupné z: <https://www.sew-eurodrive.cz/vyrobky/motory/servomotory/servomotoren_3.html>

*ESP32* [online]. [cit. 2024-04-14]. Dostupné z: <https://dratek.cz/arduino/1581-esp-32s-esp32-esp8266-development-board-2.4ghz-dual-mode-wifi-bluetooth-antenna-module.html?gad_source=1&gclid=CjwKCAjw_e2wBhAEEiwAyFFFo_Yl77PuOenvOFUfxFlhH19ndb6-PatzIMp1eutXGdseRIeW3-p4sxoC3RYQAvD_BwE>

*MicroPython na esp32* [online]. [cit. 2024-04-24]. Dostupné z: <https://docs.micropython.org/en/latest/esp32/tutorial/intro.html>

*Baterie obrázek*. Online. Dostupné z: <https://botland.cz/prislusenstvi-k-bateriim/5502-9v-baterie-6f22-s-dratem-5904422362751.html>. [cit. 2024-04-25].

*Modulový driver servo motoru I2C*. Online. Dostupné z: <https://dratek.cz/arduino/1686-iic-i2c-modulovy-driver-servo-motoru-pro-arduino-pca9685-16-kanalu-12-bit-pwm.html?utm_source=ehub&utm_medium=affiliate&ehub=1383fd1aad78422d95b1944e5622382e>. [cit. 2024-04-25].

1. 3D tisk = technologie výroby [↑](#footnote-ref-1)
2. PuTTY = multiplatformní program, který sloužil pro testovaní pohybu pavouka [↑](#footnote-ref-2)
3. Visual Studio Code = editor zdrojového kódu [↑](#footnote-ref-3)
4. Regulátor napětí = reguluje elektrické napětí z baterií pavouka [↑](#footnote-ref-4)
5. Soc mikrokontroler = jednočipový počítač [↑](#footnote-ref-5)
6. Periferie = zařízení, které se připojuje k jednotce a rozšíří možnosti [↑](#footnote-ref-6)
7. I2C sběrnice = Sériová sběrnice, která využívá ke komunikaci dva PINY (SDA, SCL) [↑](#footnote-ref-7)
8. Arduino = jednodeskový počítač, který je založen na mikrokontrolerech [↑](#footnote-ref-8)
9. SDA = PIN, který je zodpovědný za přenos informací [↑](#footnote-ref-9)
10. SCL = PIN, který slouží k přenosu hodinového signálu, určuje „rytmus“ [↑](#footnote-ref-10)
11. Aktuátor = pohon, který přesměruje rotační pohyb na lineární [↑](#footnote-ref-11)
12. Servomechanismus = vnitřní části servo motoru [↑](#footnote-ref-12)
13. Řadič = řídí činnost servomotoru [↑](#footnote-ref-13)
14. REPL = konzolové okno (prostředí programovacího jazyka) [↑](#footnote-ref-14)
15. Access Point = přístupový bod [↑](#footnote-ref-15)
16. Síťový soket = koncový bod [↑](#footnote-ref-16)