



ACEBOTT

Cvičení o čtyřnohém biomechanickém pavoukovi

Předmluva

Naše společnost

ACEBOTT STEM Education Tech Co.,Ltd

Historie společnosti: Byla založena v roce 2013 a sídlí v čínském Silicon Valley - Šen-čenu. Vytvořili jsme tým složený z 150 členů, zahrnující výzkum a vývoj, výrobu, prodej a logistiku. Naším cílem je poskytovat zákazníkům vynikající produkty a služby v oblasti STEM vzdělávání. Spolupracujeme s odborníky a obchodními partnery z celého světa v oblasti STEM vzdělávání, abychom zákazníkům poskytli vynikající sady pro STEM vzdělávání. Zároveň poskytujeme zákazníkům služby OEM, včetně balení produktů a služeb na zakázku pro logo na desce plošných spojů (PCB).

Návod

Tento kurz a sada pro učení se čtyřnohému robotovi je určena pro děti a mládež ve věku 8 let a starší, s cílem hlouběji porozumět vývojové desce ESP8266, znalostem čtyřnohých robotů a elektronickému hardwaru. Pokud chcete získat znalosti o čtyřnohých robotech, tato sada vám může poskytnout znalosti a postup při sestavování vašeho vlastního čtyřnohého robota.

Pomocí tohoto sady můžete:

- 1.Naučit se efektivně používat vývojovou desku ESP8266, včetně stahování kódu, porozumění jejím funkcím a programování v Arduino IDE.
- 2.Postavit pevný základ v jazyce C, protože ESP8266 využívá zjednodušený programovací jazyk C/C++ pro ovládání obvodů a senzorů.
- 3.Prozkoumat principy fungování modulů servomotorů a porozumět spolupráci více servomotorů v projektu čtyřnohého robota.
- 4.Postavit své vlastní čtyřnohého robota pomocí sady ACEBOTT podle krok za krokem návodů a zdokonalit své dovednosti tvůrce.
- 5.Implementovat základní funkce v projektu čtyřnohého robota, jako je řízení pohybu dopředu, dozadu, doleva a doprava, řízení otáčení, řízení kombinovaných akcí a řízení přes aplikaci.
- 6.Zlepšit své komplexní porozumění konceptům čtyřnohých robotů a připravit se na budoucí pokročilé učení.

Celkově lze říci, že čtyřnohý robot ACEBOTT je speciálně navržený vzdělávací sada založená na ESP8266 pro začátečníky. S použitím této sady mohou uživatelé získat komplexní porozumění funkcím řídicí desky a servomotorů v čtyřnohých robotech. Díky návodům poskytnutým v sadě mohou studenti různých věkových kategorií získat cenné znalosti o čtyřnohých robotech a úspěšně postavit své vlastní projekty čtyřnohých robotů.

Po-prodejní servis

ACEBOTT je dynamická a rychle se rozvíjející technologická společnost zaměřená na STEM vzdělávání, která si klade za cíl poskytovat vynikající produkty a prvotřídní služby, aby uspokojila vaše očekávání. Vaše zpětná vazba je pro nás důležitá, a proto vás povzbuzujeme, abyste nám posílali své názory a návrhy na adrese support@cebott.com.

Náš zkušený tým inženýrů je zde, aby rychle vyřešil jakékoli problémy nebo dotazy, se kterými se setkáváte při používání našich produktů. Během pracovních dnů vám garantujeme odpověď do 24 hodin.

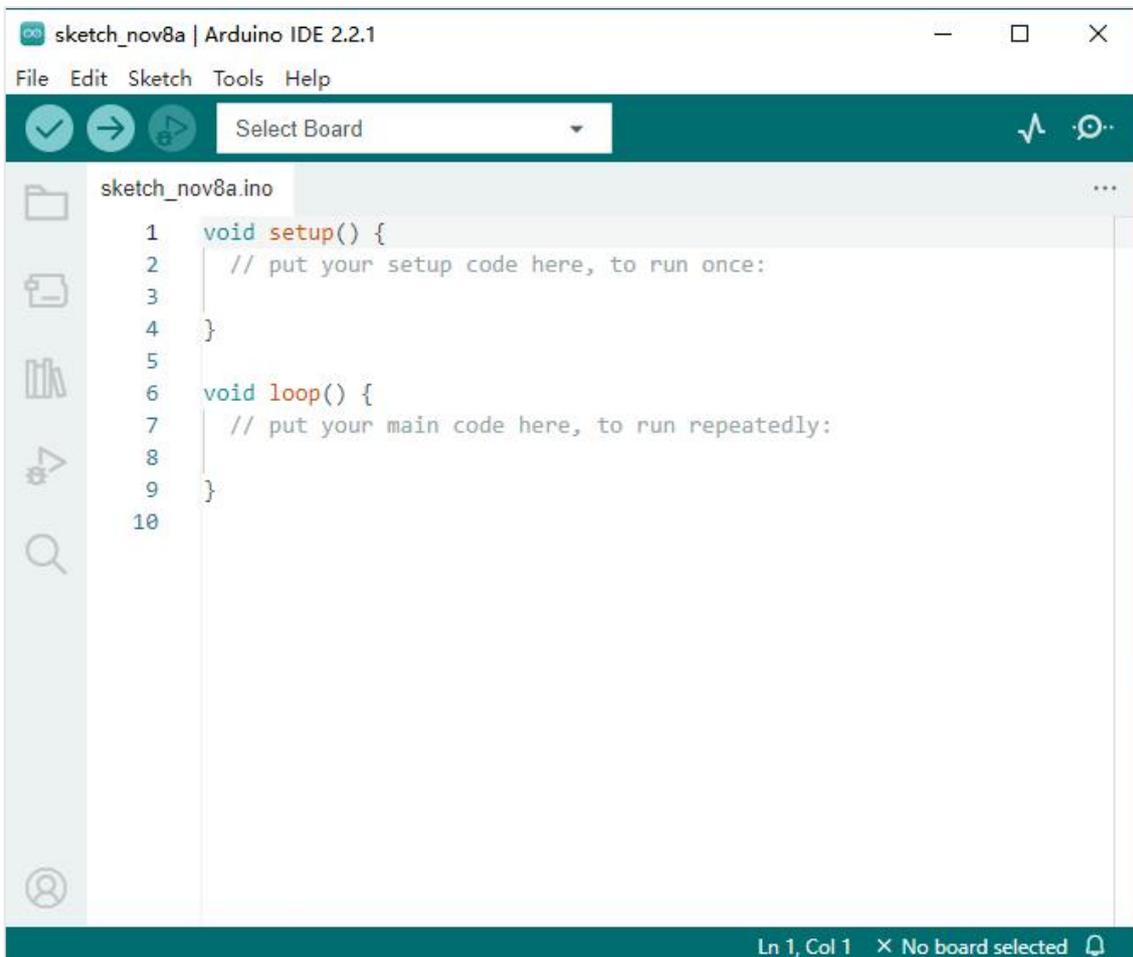
Obsah

Lekce 1: Instalace softwaru a seznámení s hardwarem	1
I .Instalace softwaru	1
II .Seznámení s hardwarem.....	17
III.Řízení servomotorů	20
Lekce 2: Sestavení robota	24
I .Seznam příslušenství	24
II .Seznam konstrukčních dílů	24
III.Montážní postup	25
Lekce 3: Inicializace polohy	33
I .Nulová poloha	33
II .nastavení programu na nulovou pozici	34
Lekce 4: Pohyb robotů vpřed a vzad	38
I .Program pro režim čekání	38
II .Program předního pohybu	40
III.Programy, které se pohybují pozpátku	42
IV.Rozšířené úkoly	44
Lekce 5: Otočení robota doleva a doprava	45
I .Program pro otočení doleva	45
II .Program pro otočení doprava	47
III.rozšířený úkol	49
Lekce 6: Pohyb robota doleva a doprava	51
I .Jedna, program pro pohyb doleva	51
II .Program pro pohyb doprava	52
III.rozšířený úkol	54
Lekce 7:Program tance robotů v	57
I .Základní taneční krok	57
II .Středně pokročilé taneční kroky	59
III.pokročilé tance	61
IV.rozšířený úkol	63
Lekce 8: Ovládání robota přes WiFi	65
I .APP stahování	65
II .Stahování programu	66

Lekce 1: Instalace softwaru a seznámení s hardwarem

I .Instalace softwaru

čtyřnohém projektu s biomimetickým pavoukem budeme hlavně používat Arduino IDE jako programovací software. Jedná se o open-source programovací software, který je kompatibilní s různými deskami, jako jsou Arduino, ESP32, ESP8266, STM32 a další. S použitím Arduino IDE stačí napsat programový kód v IDE a poté ho nahrát na desku. Program řekne desce, co má provádět.



```
sketch_nov8a.ino
1 void setup() {
2   // put your setup code here, to run once:
3
4 }
5
6 void loop() {
7   // put your main code here, to run repeatedly:
8
9 }
10
```

Ln 1, Col 1 × No board selected

1.Instalace Arduino IDE

Nejprve otevřete oficiální webovou stránku ke stažení Arduino IDE:

<https://www.arduino.cc/en/Main/Software>

Podle operačního systému vašeho počítače vyberte příslušný způsob instalace.

(1) Instalace na systému Windows

① Klepněte na místo zobrazené na obrázku.



Arduino IDE 2.2.1

The new major release of the Arduino IDE is faster and even more powerful! In addition to a more modern editor and a more responsive interface it features autocompletion, code navigation, and even a live debugger.

For more details, please refer to the [Arduino IDE 2.0 documentation](#).

Nightly builds with the latest bugfixes are available through the section below.

SOURCE CODE

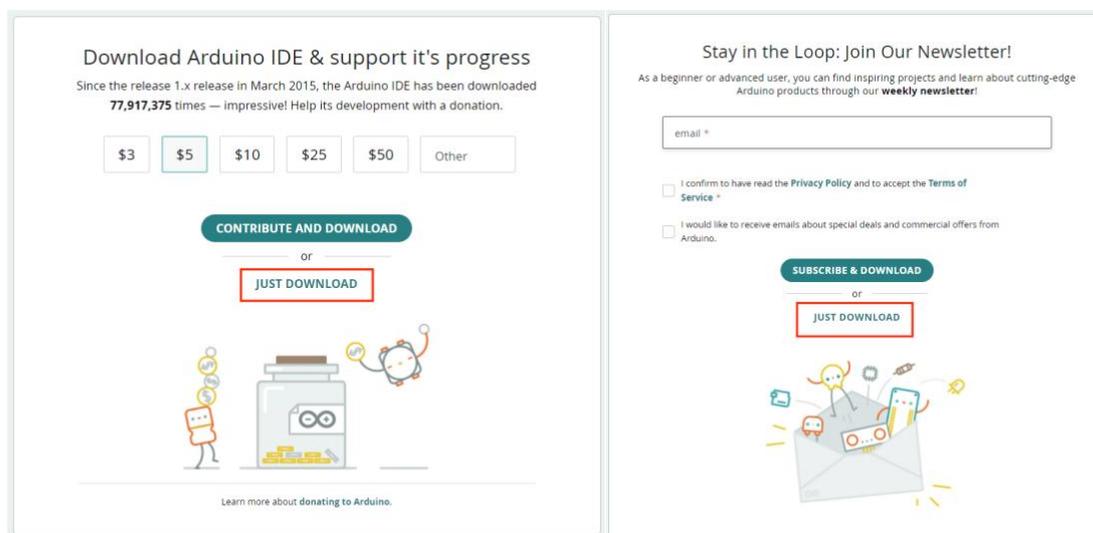
The Arduino IDE 2.0 is open source and its source code is hosted on [GitHub](#).

DOWNLOAD OPTIONS

- Windows** Win 10 and newer, 64 bits
- Windows** MSI installer
- Windows** ZIP file
- Linux** ApplImage 64 bits (X86-64)
- Linux** ZIP file 64 bits (X86-64)
- macOS** Intel, 10.14: "Mojave" or newer, 64 bits
- macOS** Apple Silicon, 11: "Big Sur" or newer, 64 bits

[Release Notes](#)

② Vyberte možnost JUSTDOWNLOAD.



Download Arduino IDE & support it's progress

Since the release 1.x release in March 2015, the Arduino IDE has been downloaded **77,917,375** times — impressive! Help its development with a donation.

\$3 \$5 \$10 \$25 \$50 Other

CONTRIBUTE AND DOWNLOAD

or

JUST DOWNLOAD

Learn more about donating to Arduino.

Stay in the Loop: Join Our Newsletter!

As a beginner or advanced user, you can find inspiring projects and learn about cutting-edge Arduino products through our **weekly newsletter!**

email *

I confirm to have read the [Privacy Policy](#) and to accept the [Terms of Service](#) *

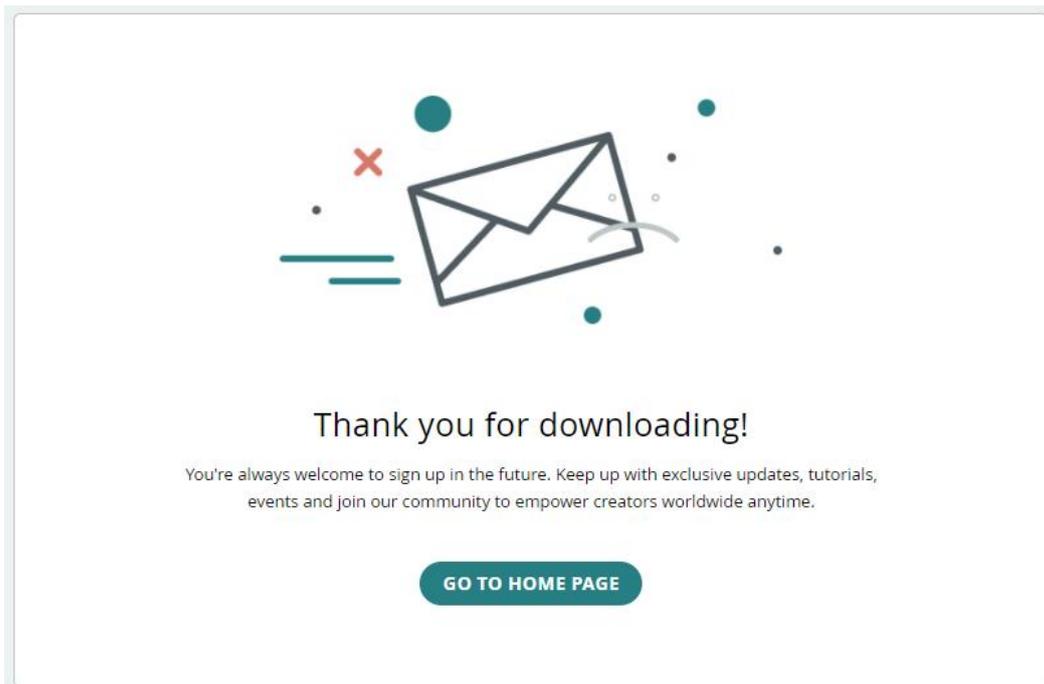
I would like to receive emails about special deals and commercial offers from Arduino.

SUBSCRIBE & DOWNLOAD

or

JUST DOWNLOAD

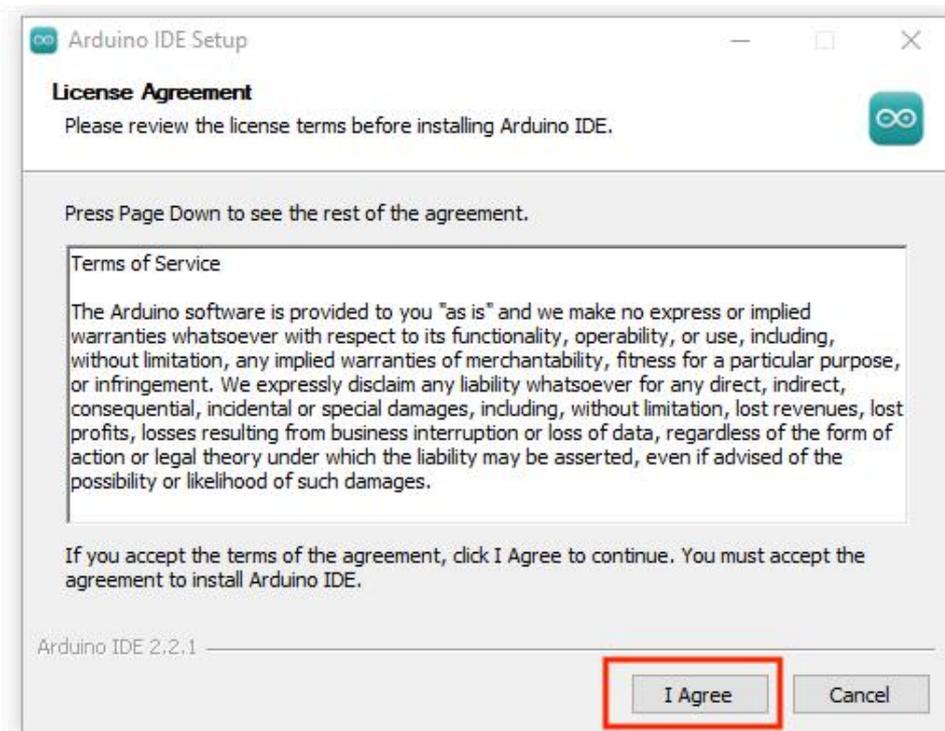
③Když se zobrazí následující rozhraní, znamená to, že se Arduino IDE stahuje.



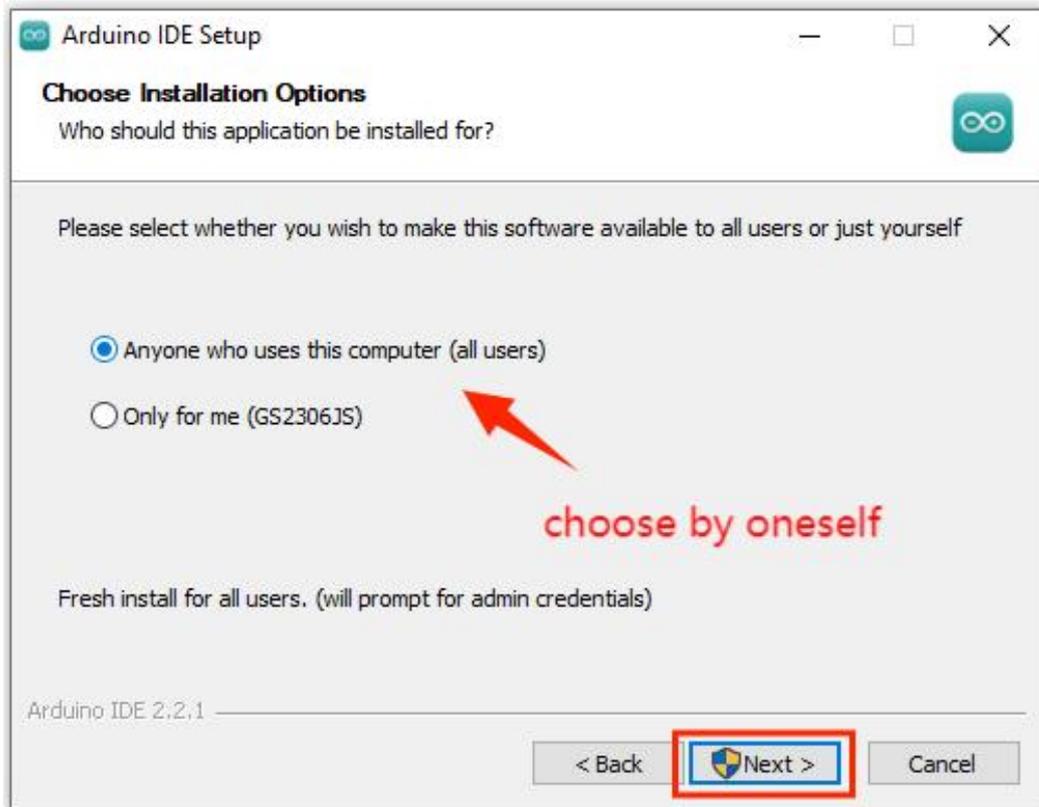
④Po dokončení stahování se zobrazí ikonový soubor. Klepněte na něj a nainstalujte software.



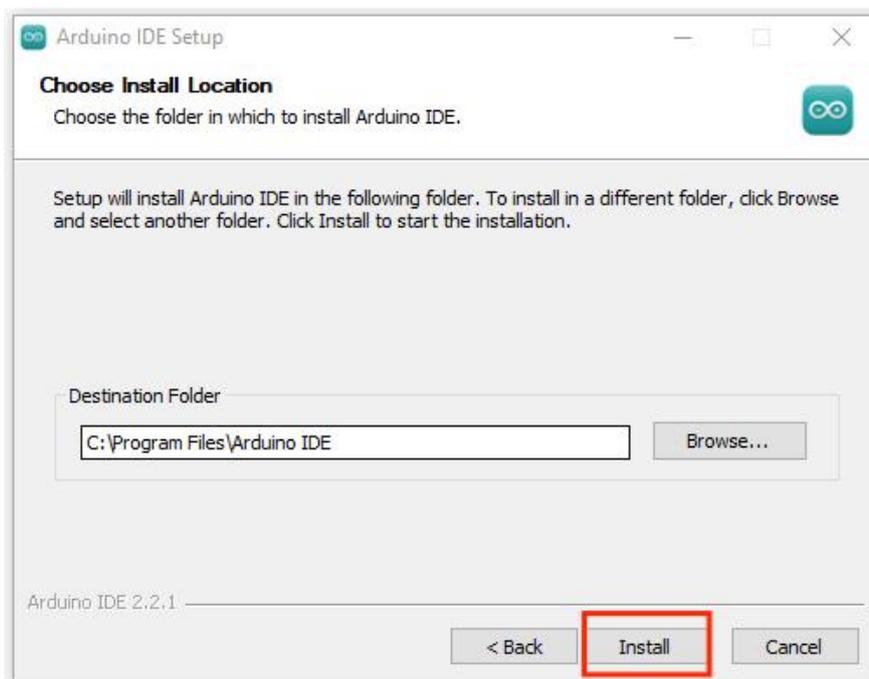
⑤Po instalaci se zobrazí následující rozhraní. Vyberte možnost "I Agree".



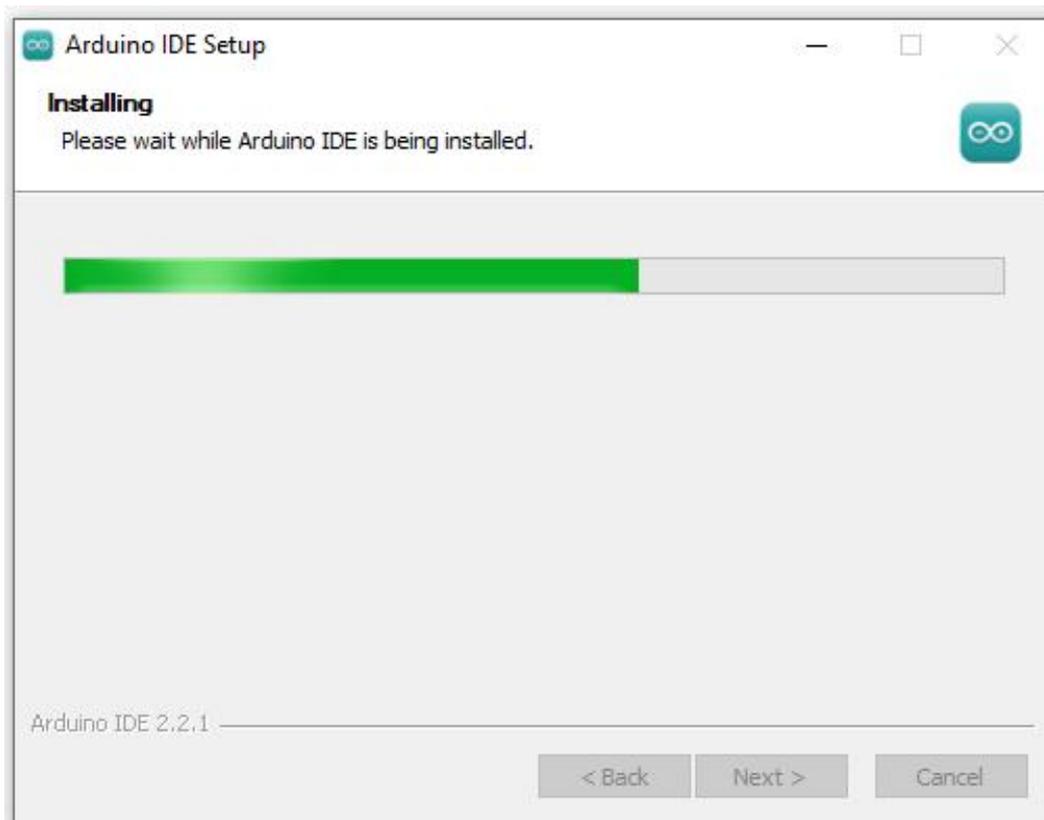
⑥ Po výběru možnosti "I Agree" se zobrazí následující rozhraní. Vyberte možnost "Next".



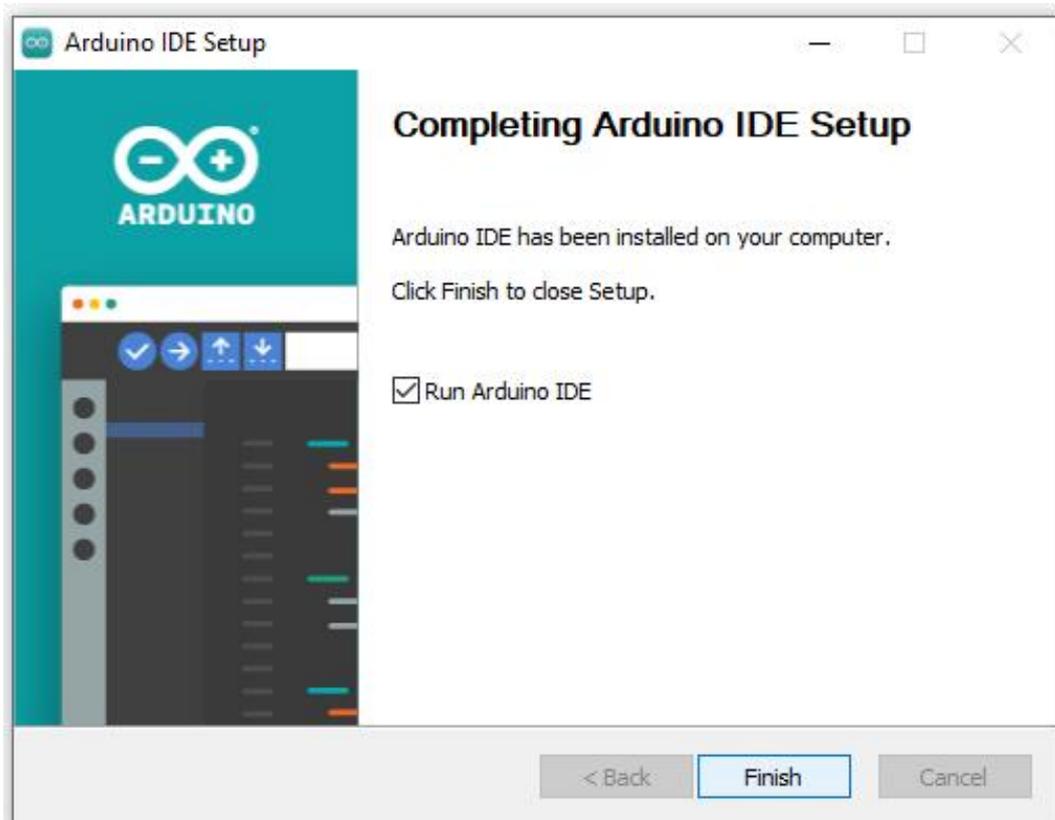
⑦ Po výběru možnosti "Next" se zobrazí následující rozhraní. Vyberte možnost "Install".



⑧ Probíhá instalace softwaru Arduino IDE.



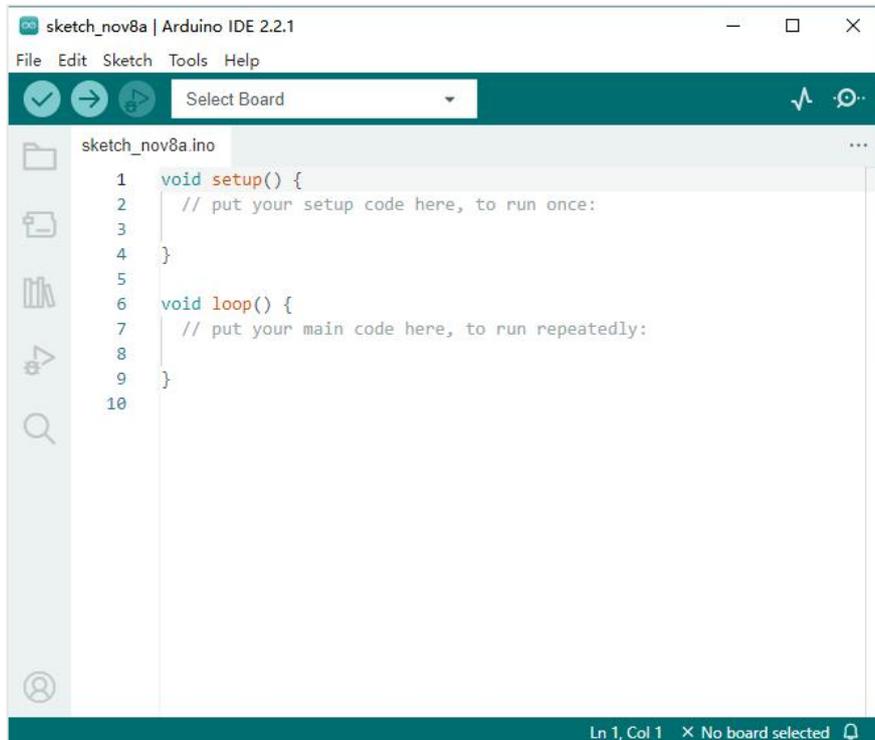
⑨ Po dokončení instalace klepněte na tlačítko Finish.



⑩ Po dokončení instalace se na ploše vašeho počítače objeví zástupce pro Arduino IDE.



① Po otevření se zobrazí následující rozhraní softwaru.



(2) Instalace na systému

① Klepněte na místo zobrazené na obrázku.

Arduino IDE 2.2.1

The new major release of the Arduino IDE is faster and even more powerful! In addition to a more modern editor and a more responsive interface it features autocompletion, code navigation, and even a live debugger.

For more details, please refer to the [Arduino IDE 2.0 documentation](#).

Nightly builds with the latest bugfixes are available through the section below.

SOURCE CODE

The Arduino IDE 2.0 is open source and its source code is hosted on [GitHub](#).

DOWNLOAD OPTIONS

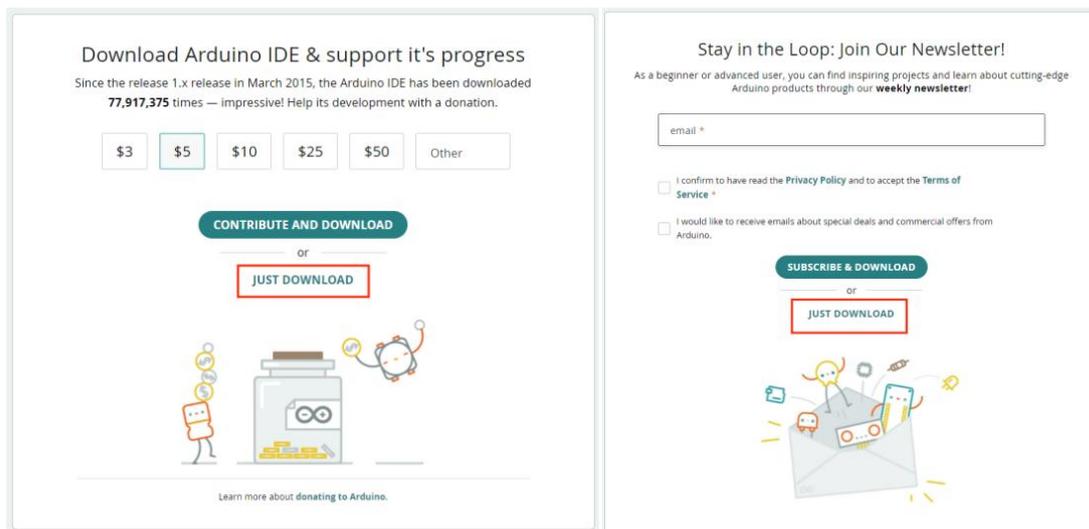
Windows Win 10 and newer, 64 bits
Windows MSI installer
Windows ZIP file

Linux ApplImage 64 bits (X86-64)
Linux ZIP file 64 bits (X86-64)

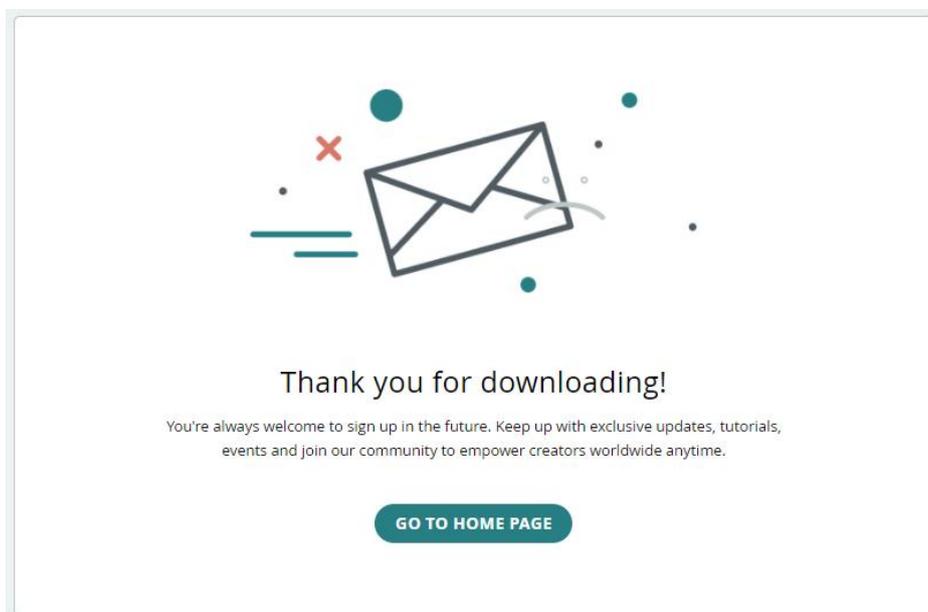
macOS Intel, 10.14: "Mojave" or newer, 64 bits
macOS Apple Silicon, 11: "Big Sur" or newer, 64 bits

[Release Notes](#)

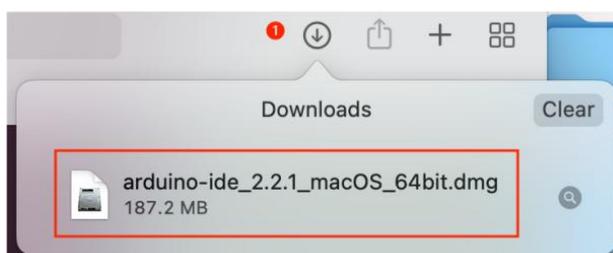
② Vyberte možnost JUSTDOWNLOAD.



③Když se zobrazí následující rozhraní, znamená to, že se Arduino IDE stahuje.



④Po dokončení stahování klepněte na ikonu stahování v prohlížeči a najdete instalační balíček Arduino IDE.



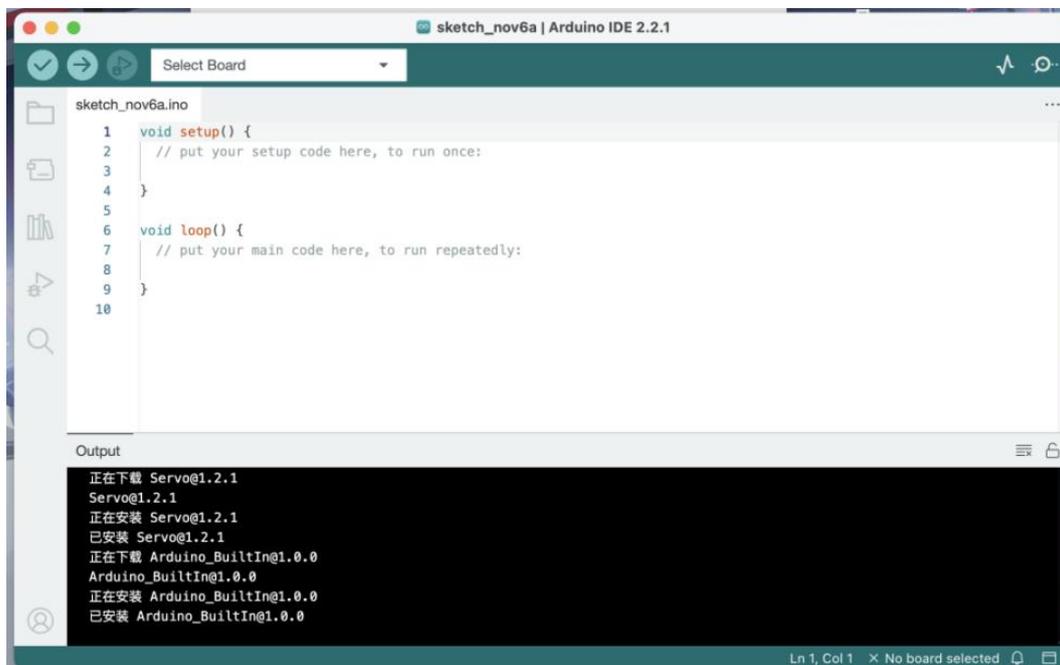
⑤ Klepnutím na instalační balíček se zobrazí instalační rozhraní. Pouze vyberte ikonu Arduino IDE a přesuňte ji do složky Applications, aby se program nainstaloval.



⑥ Na pracovní ploše najdete Arduino IDE a otevřete ji.



⑦ Po otevření uvidíte následující rozhraní softwaru.



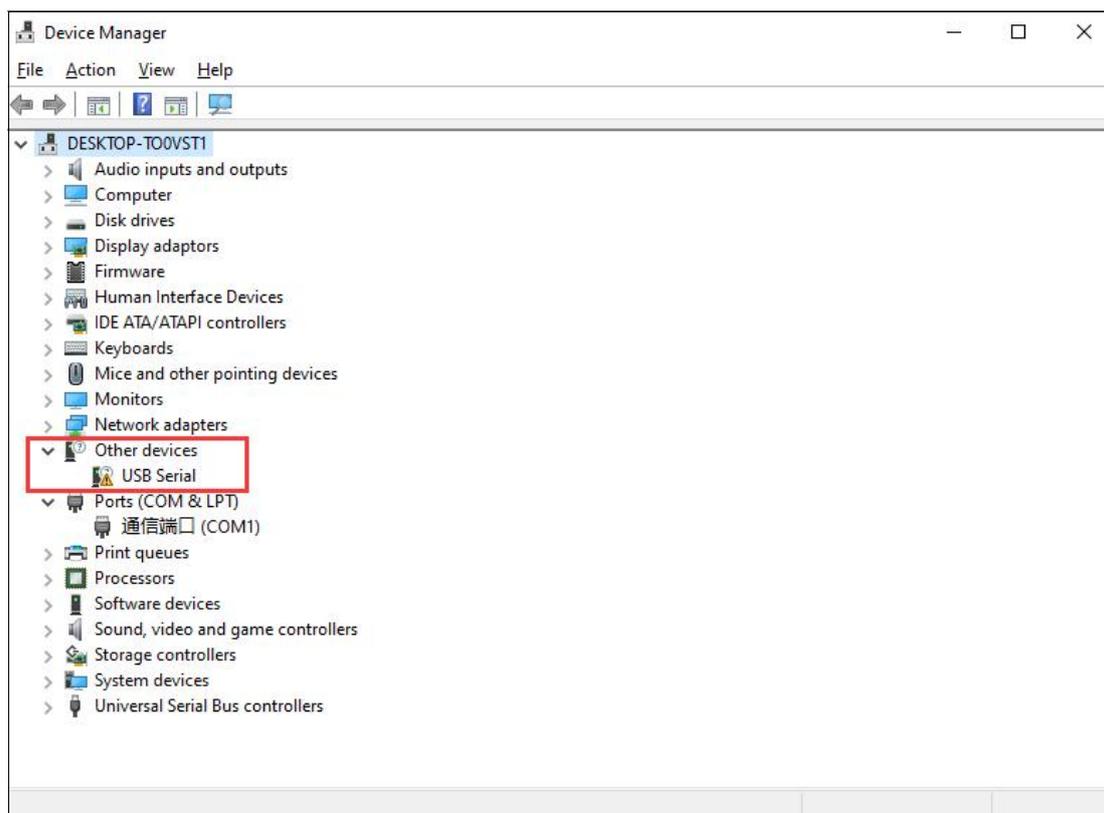
(3) Instalace ovladače sériového portu (pokud již není nainstalován, přeskočte tento krok)

Sériový port je typ komunikačního rozhraní počítače, které se obvykle používá k přenosu dat mezi počítačem a dalšími zařízeními, jako jsou modemy, senzory, tiskárny, mikrokontroléry atd. USB sériový port je nejběžnějším komunikačním rozhraním.

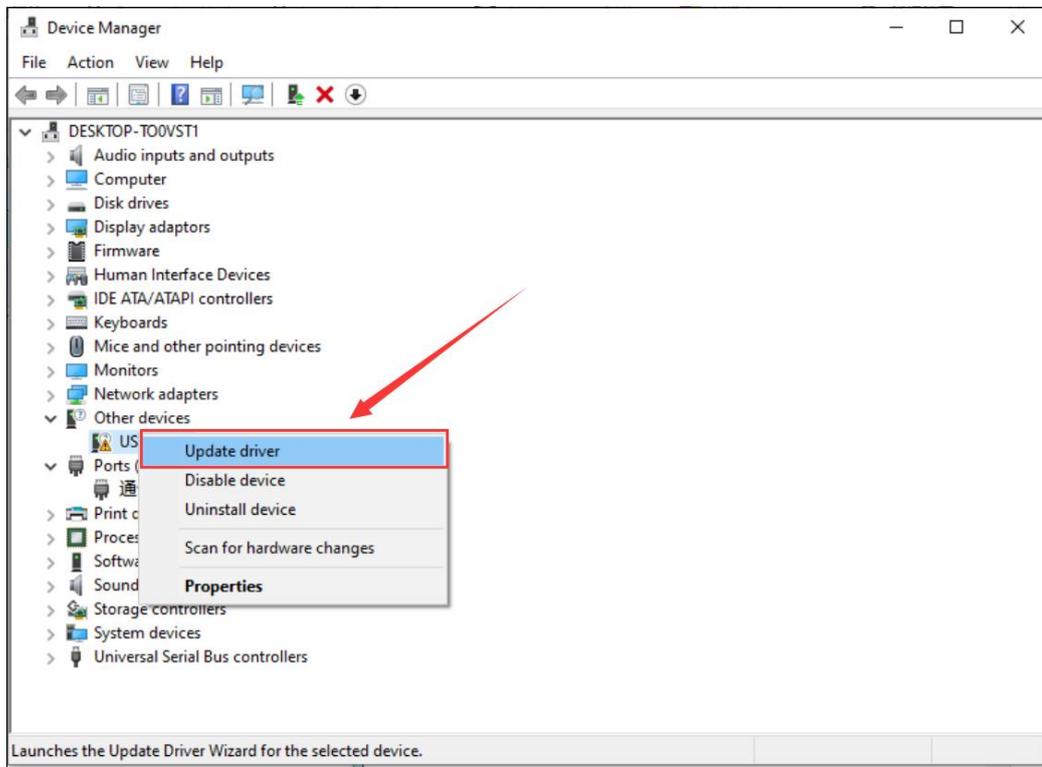
Obvykle je potřeba nainstalovat ovladače sériového portu po první instalaci softwaru Arduino IDE, aby bylo možné nahrát upravený program do desky. USB na sériový port na desce řízení ESP8266 používá čip CH340C, takže je také nutné nainstalovat ovladač tohoto čipu. Ovladač můžete buď stáhnout z internetu, nebo jej najdete ve zdrojovém balíčku souborů, který poskytujeme, ve složce nazvané "CH341SER-Driver". Poté postupujte podle následujících kroků:

① Připojte jednu stranu USB kabelu k desce řízení ESP8266 a druhou stranu do USB portu počítače.

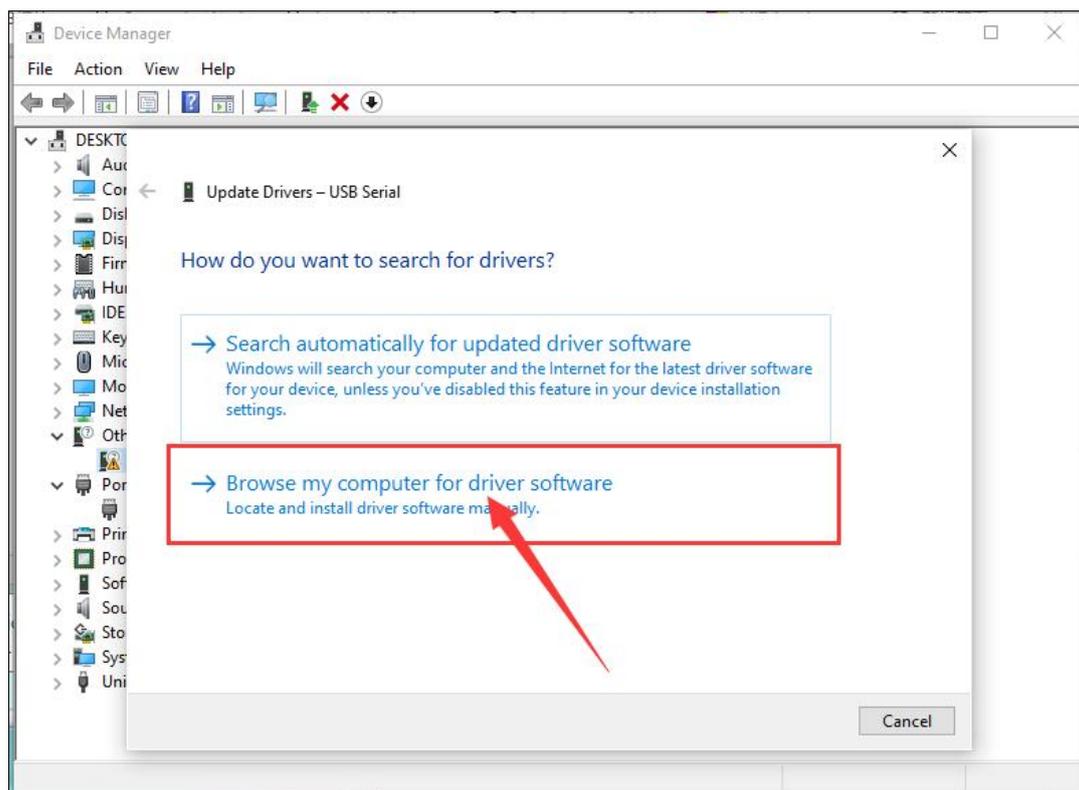
② Při prvním připojení desky ESP8266 k počítači klikněte pravým tlačítkem myši na "Můj počítač" -> "Vlastnosti" -> Klepněte na "Správce zařízení" a pod "Ostatní zařízení" uvidíte "USB-Serial" nebo "Neznámé zařízení".



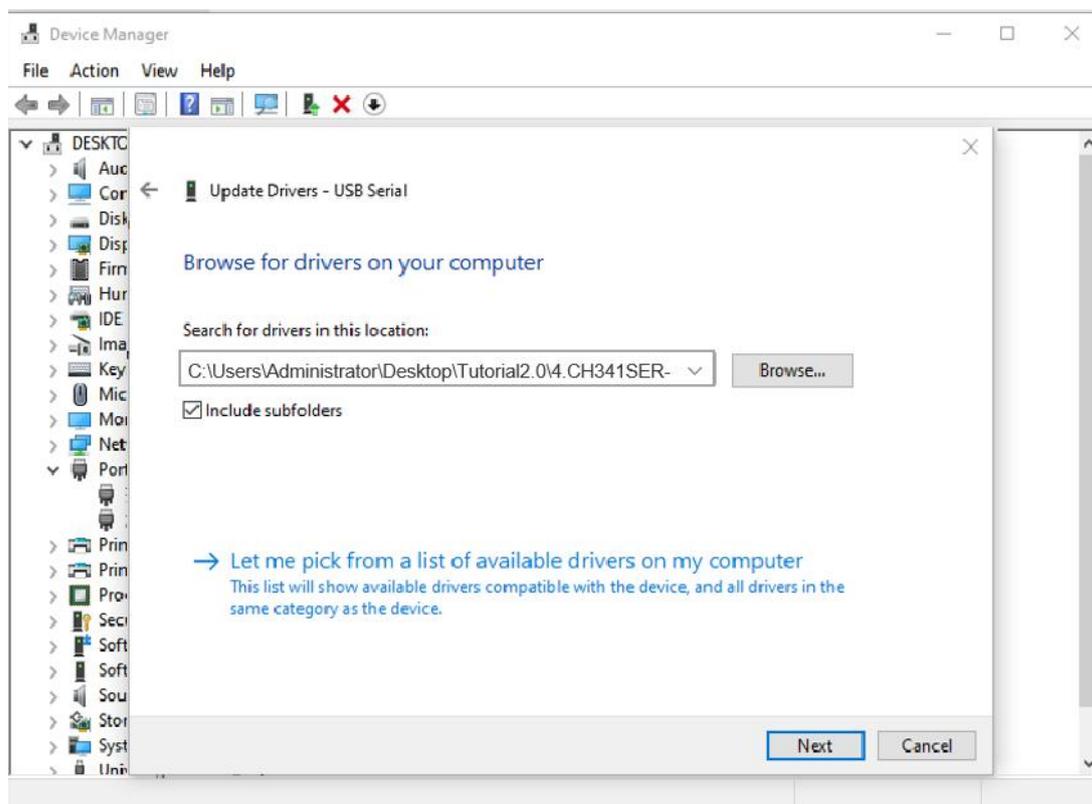
③ Nejprve klikněte pravým tlačítkem myši na zařízení a vyberte možnost v horním menu (Aktualizovat ovladač softwaru), jak je znázorněno na následujícím obrázku.



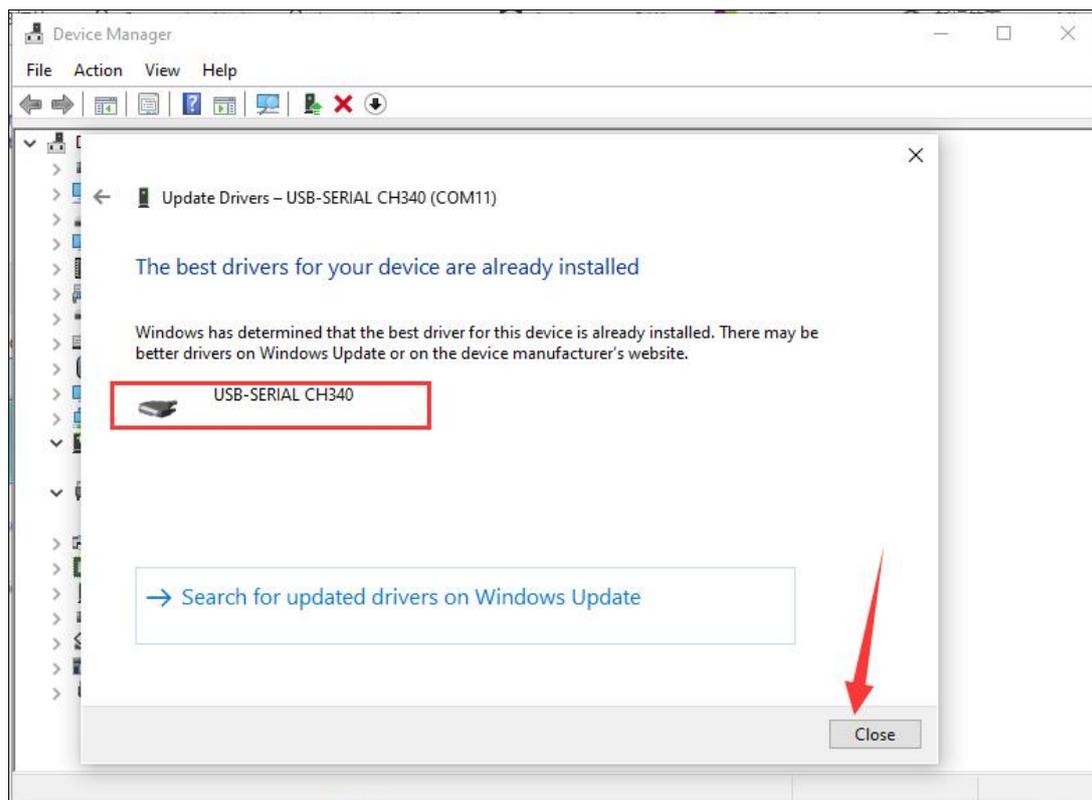
④ Poté se zobrazí možnosti "Automaticky vyhledat aktualizovaný ovladač softwaru" nebo "Procházet můj počítač pro ovladače softwaru", jak je znázorněno na následujícím obrázku. Na této stránce vyberte možnost "Procházet můj počítač pro ovladače softwaru".



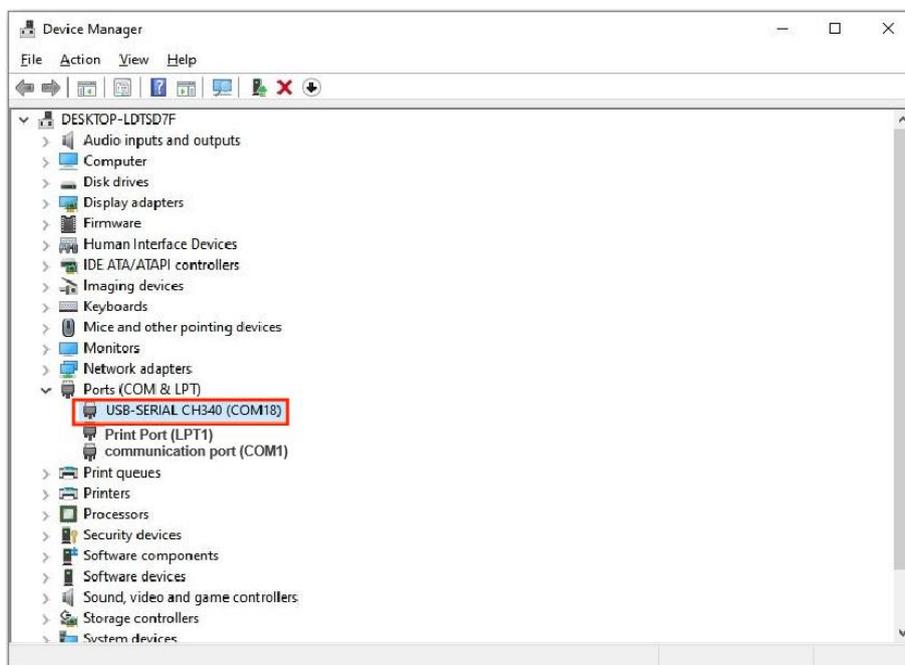
⑤ Poté přidejte cestu k souboru ovladače.



⑥ Po dokončení instalace softwaru obdržíte potvrzovací zprávu. Po dokončení instalace klikněte na "Zavřít".

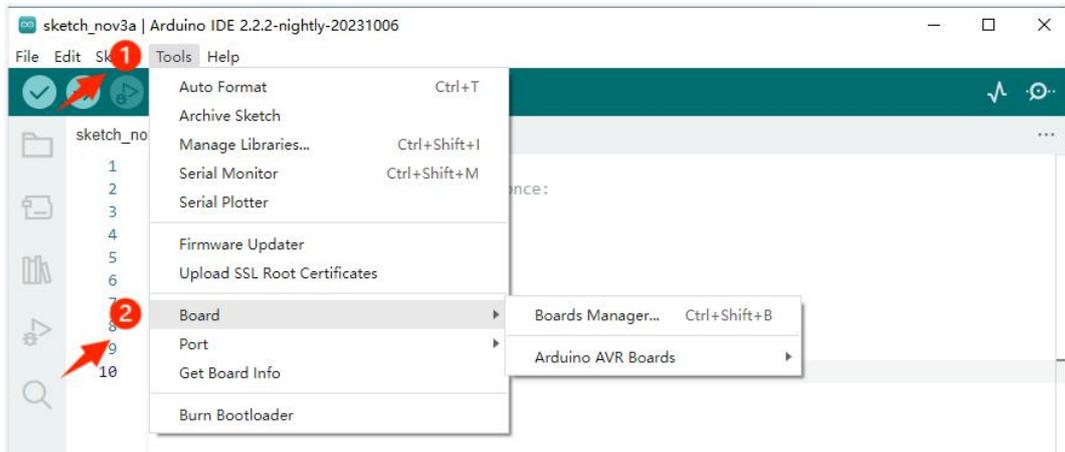


⑦ Pro potvrzení úspěšné instalace připojte jednu stranu USB kabelu k desce řízení ESP8266 a druhou stranu do USB portu počítače. Poté klikněte pravým tlačítkem myši na "Můj počítač" -> "Vlastnosti" -> Klepněte na "Správce zařízení". Pokud byla instalace úspěšná, uvidíte něco podobného následujícímu obrázku.



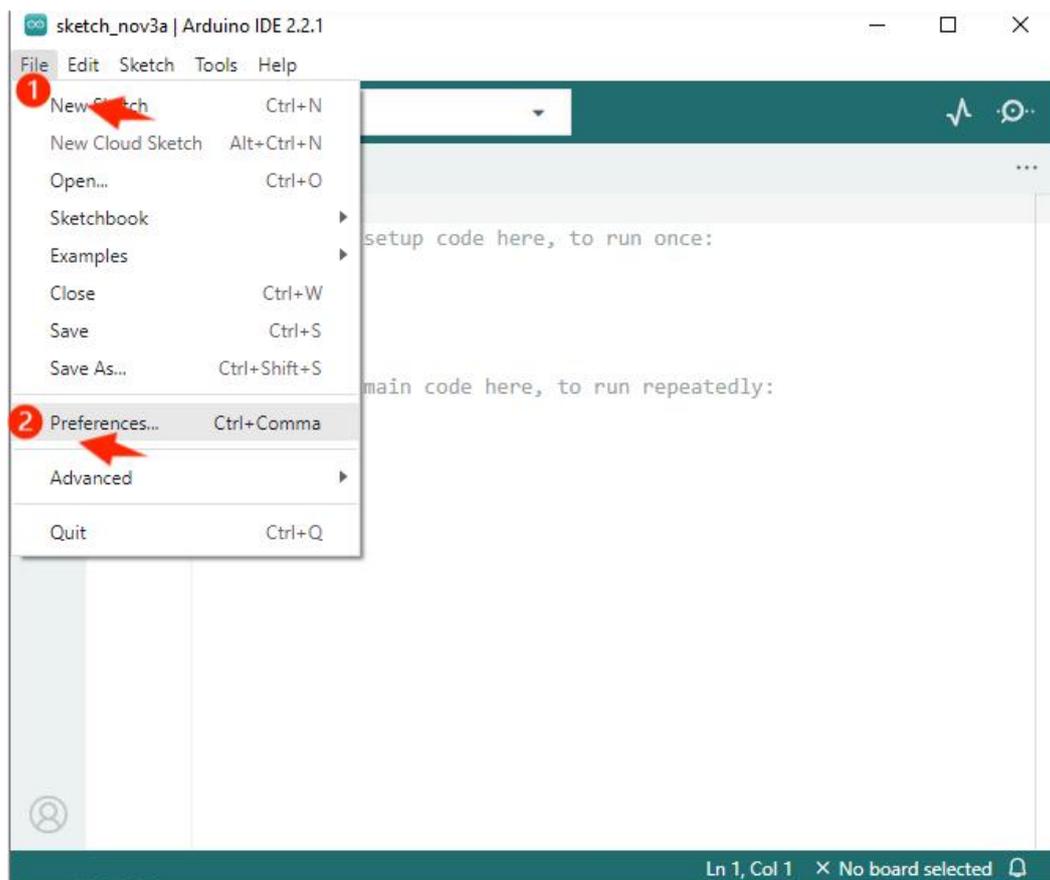
2.Instalace knihovny pro ESP8266

Protože robot využívá desku ESP8266, je nezbytné do Arduino IDE přidat knihovny pro ESP8266, aby bylo možné programovat desku ESP8266. Když otevřete Arduino IDE a přejdete na Nástroje > Deska, zjistíte, že v Arduino IDE jsou k dispozici pouze desky Arduino AVR Boards a ne ESP8266.

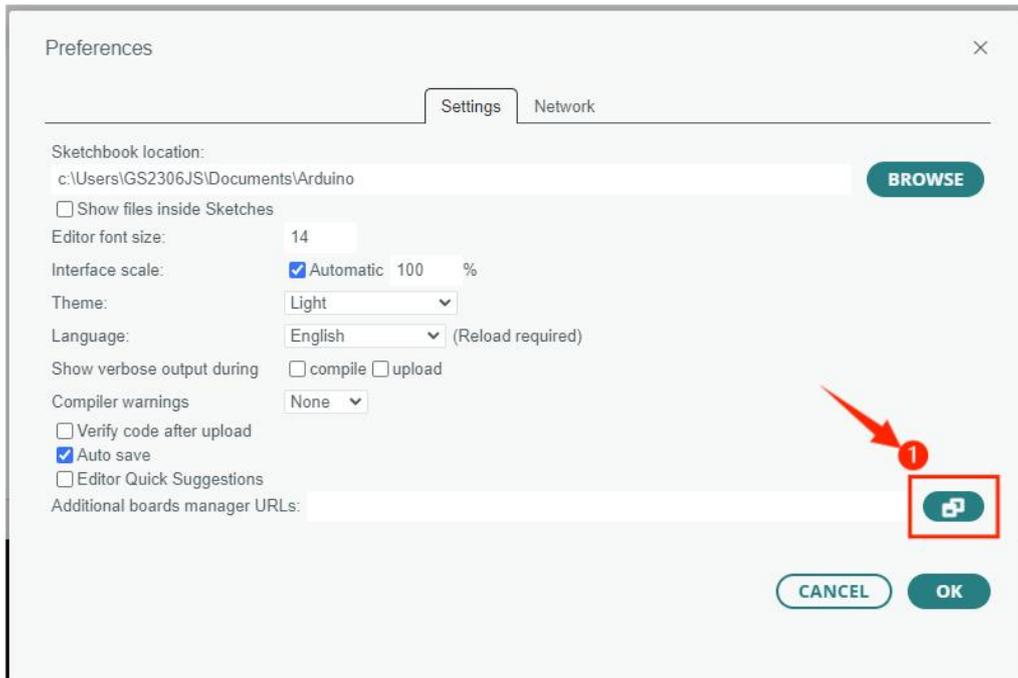


Dalším krokem je instalace knihoven pro ESP8266. Dodržte tyto kroky:

① Otevřete File>Preferences.

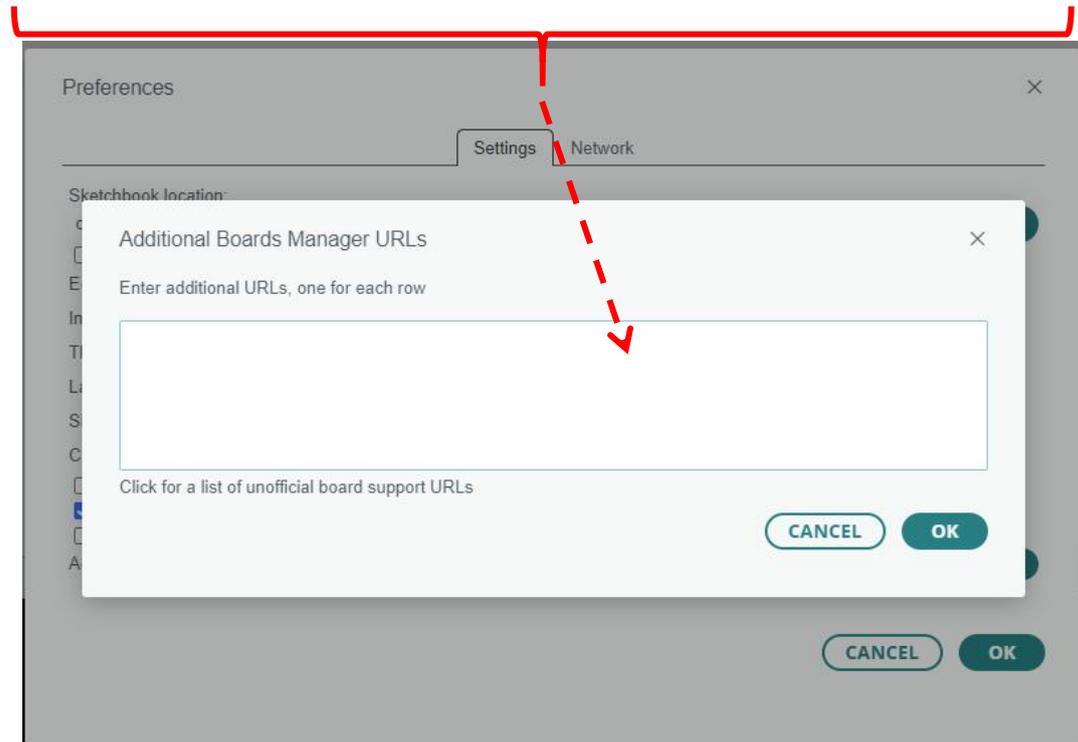


② Přidejte adresu URL pro správu desek.

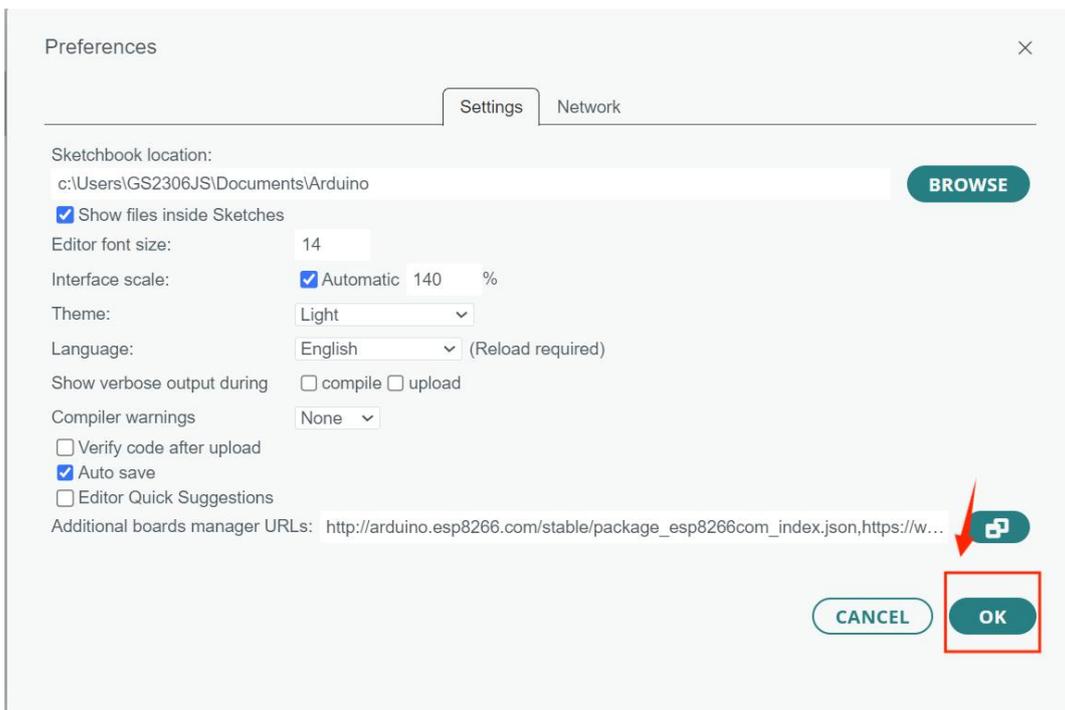
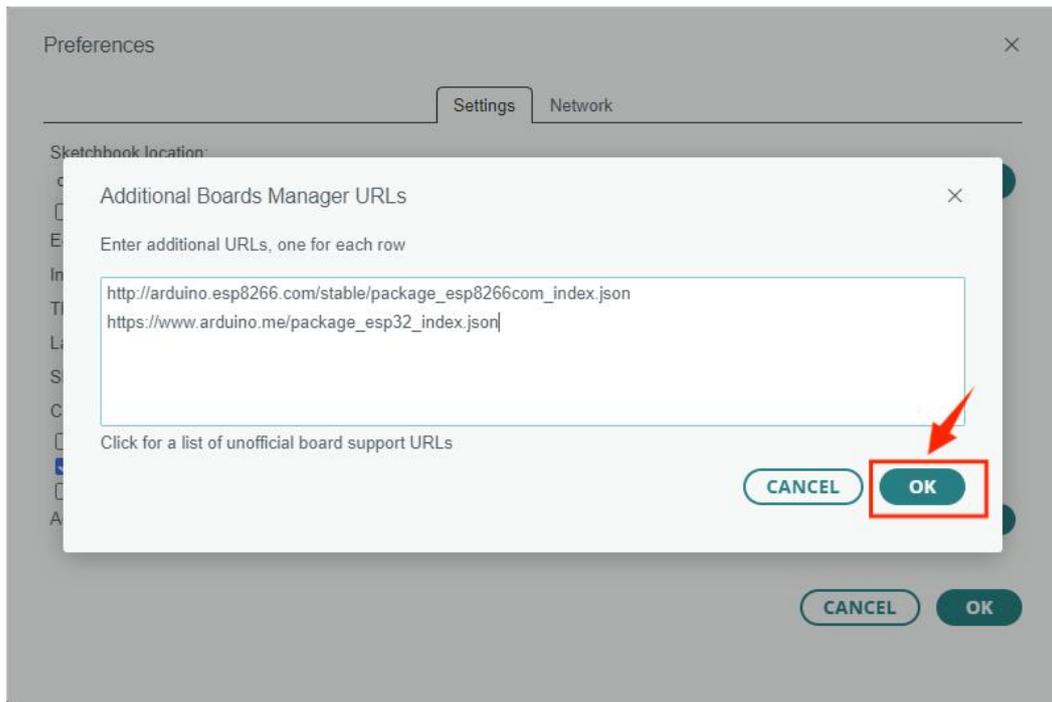


③ Zkopírujte URL adresu z následujícího textového pole a vložte ji do pole "Additional Boards Manager URLs".

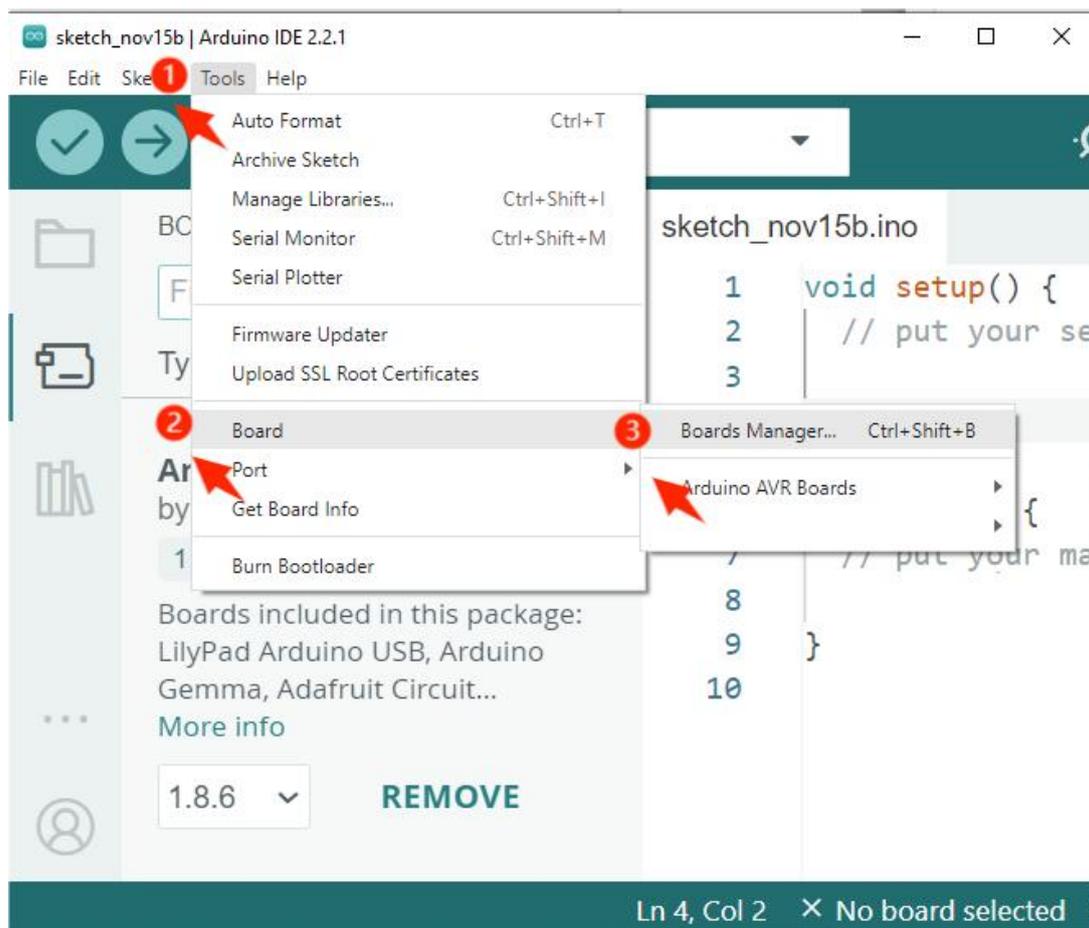
```
http://arduino.esp8266.com/stable/package_esp8266com_index.json  
https://www.arduino.me/package_ESP8266_index.json
```



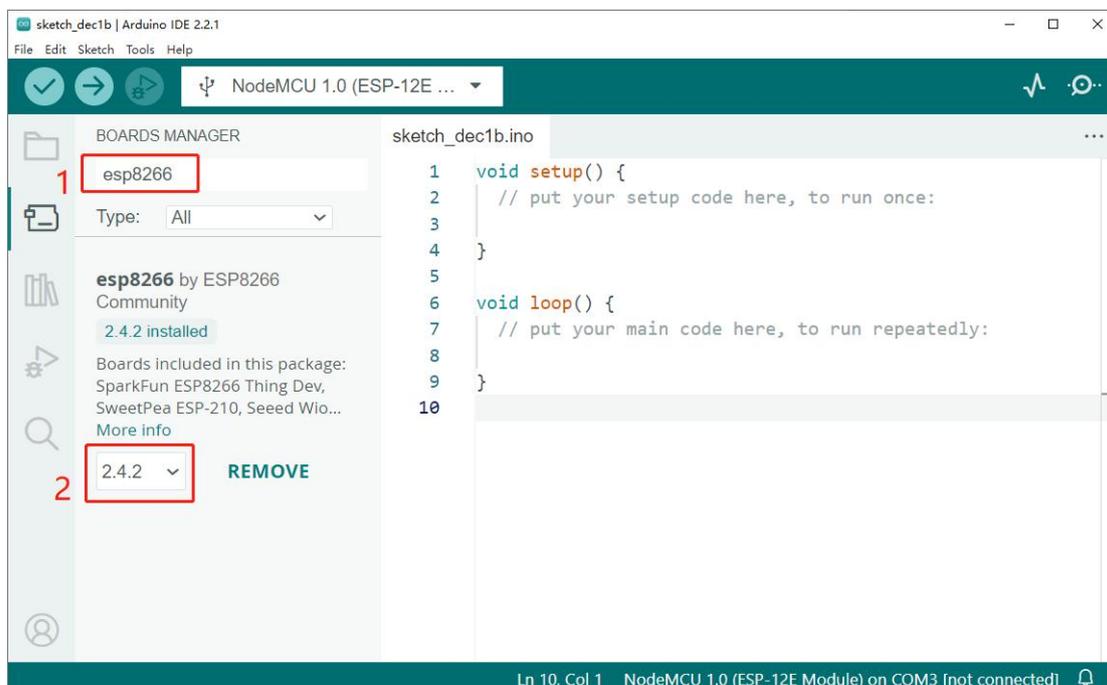
④ Po přidání URL klikněte na tlačítko "OK".



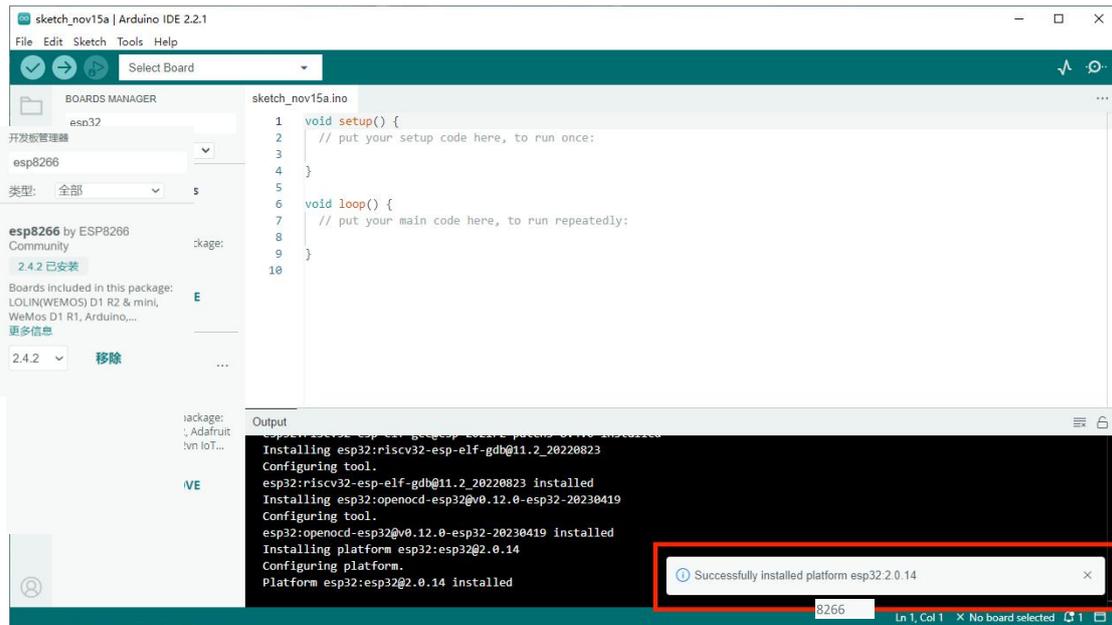
⑤ Klepněte Tools>Board>Boards Manager....



⑥V poli pro vyhledávání v Správci desek zadejte "esp8266" a vyberte verzi 2.4.2 k instalaci.

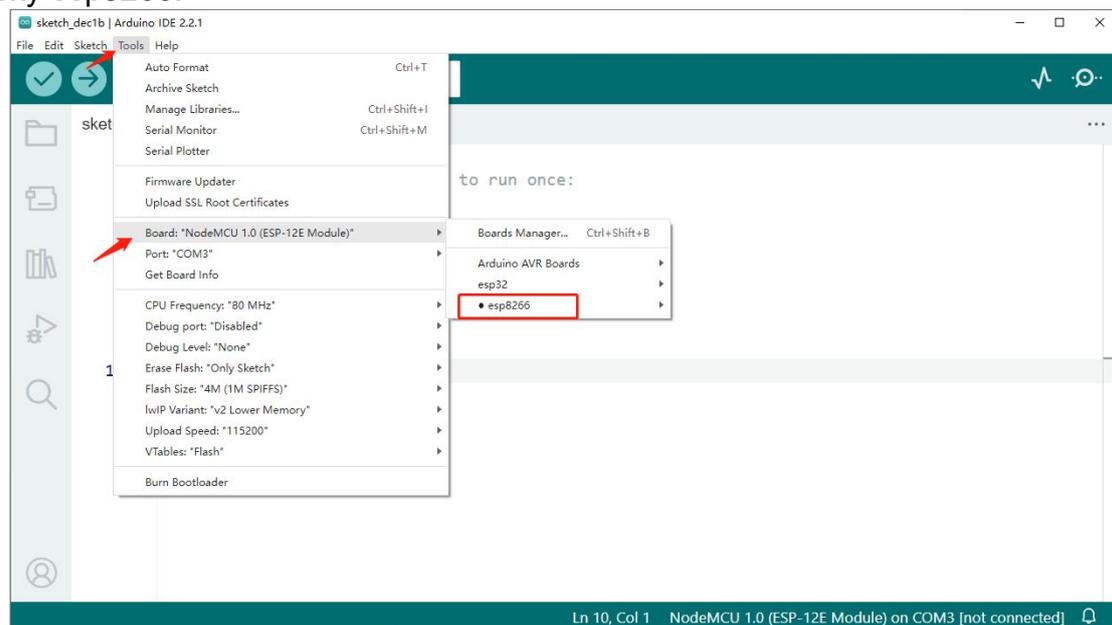


⑦Po zobrazení tohoto rozhraní vyčkejte, dokud není instalace dokončena, poté zavřete Arduino IDE.



Poznámka: Instalační balíček je zveřejněn na GitHubu, takže instalace může být ovlivněna rychlostí internetu. Pokud se instalace nezdaří, zkuste to prosím několikrát.

⑧ Znovu otevřete Arduino IDE, poté přejděte Tools>Board a uvidíte možnost desky esp8266.



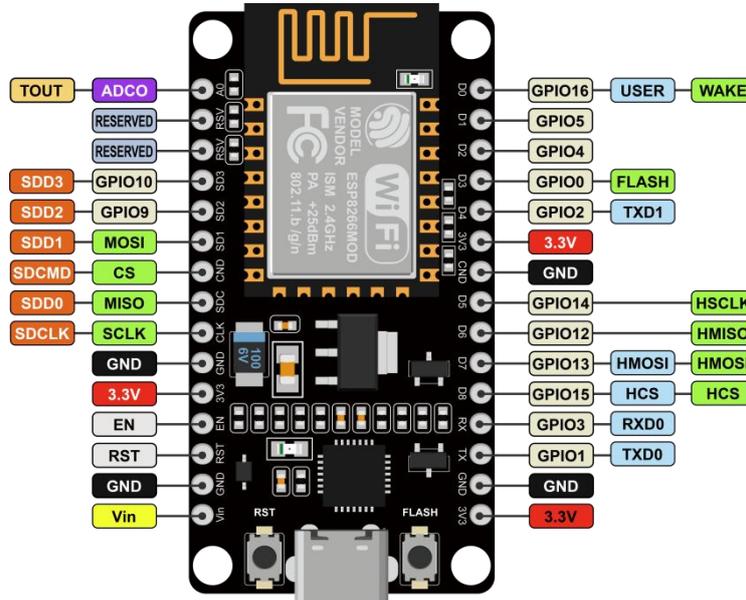
II .Seznámení s hardwarem

1.Seznámení s ESP8266

ESP8266 je WiFi modul s nízkou spotřebou a vysokým výkonem. Integruje funkce WiFi a TCP/IP protokolový stack, umožňující komunikaci s hlavním řadičem

přes sériovou linku. Je vhodný pro mobilní zařízení, nositelnou elektroniku a aplikace Internetu věcí.

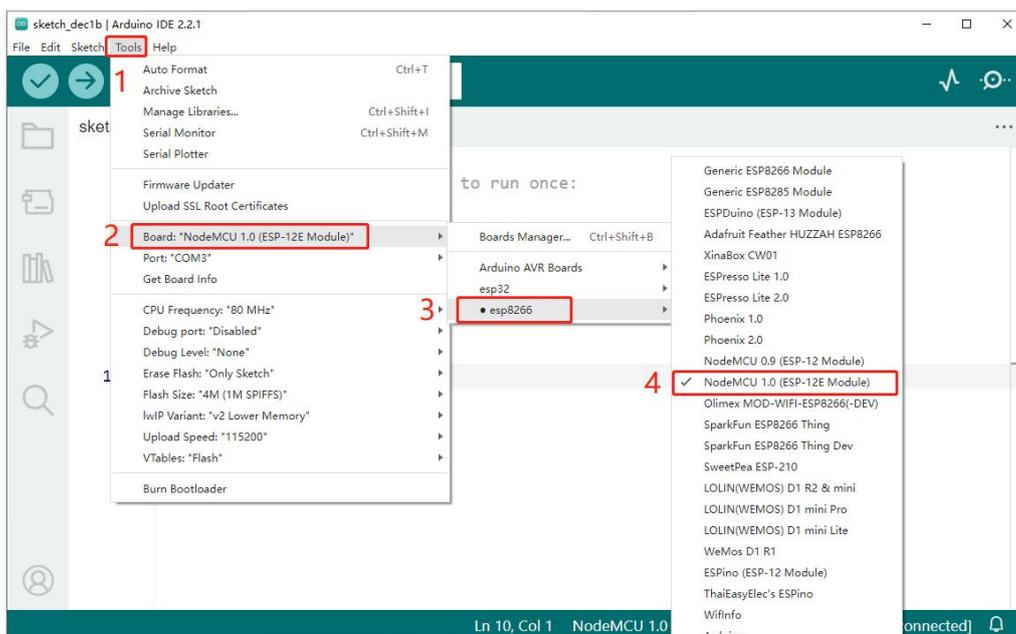
Vývojové desky s ESP8266 zahrnují: ESP-01, ESP-12E, NodeMCU a další.



2. Testovací vývojové prostředí

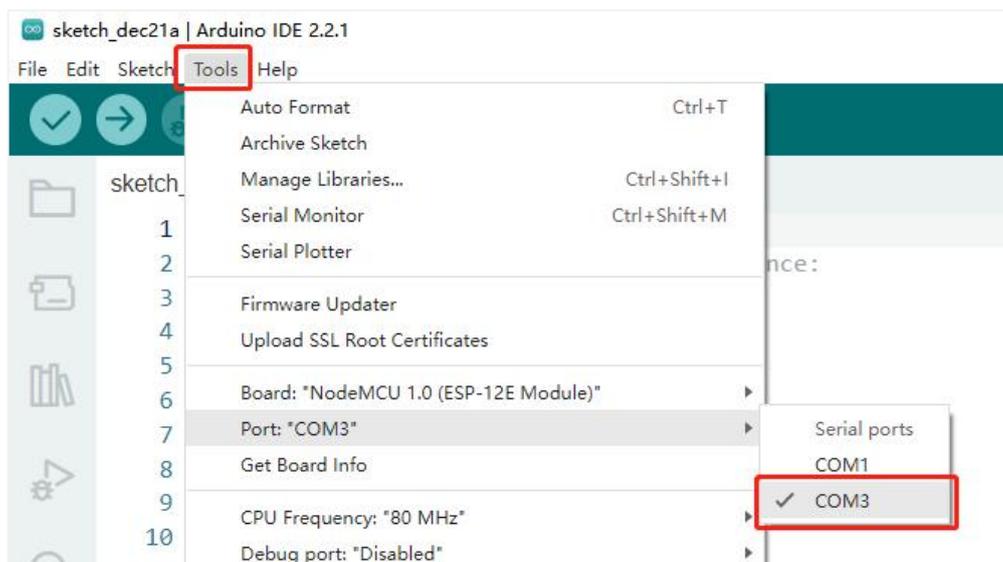
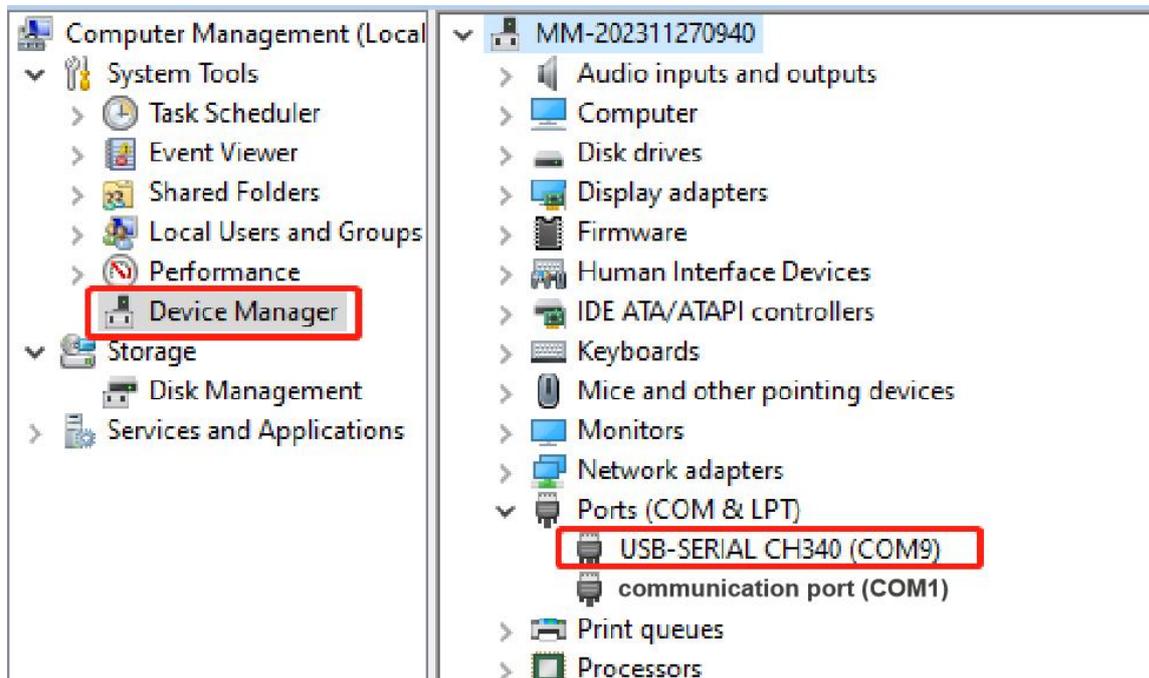
Po nainstalování Arduino IDE a rozšíření knihoven pro ESP8266 můžeme ověřit úspěšnou instalaci prostředí pomocí jednoduchého programu. Postupujte podle následujících kroků:

① Připojte desku k počítači. Otevřete Arduino IDE a klepněte na "Nástroje", poté vyberte "ESP8266" a zvolte "NodeMCU 1.0 (ESP-12E Model)".

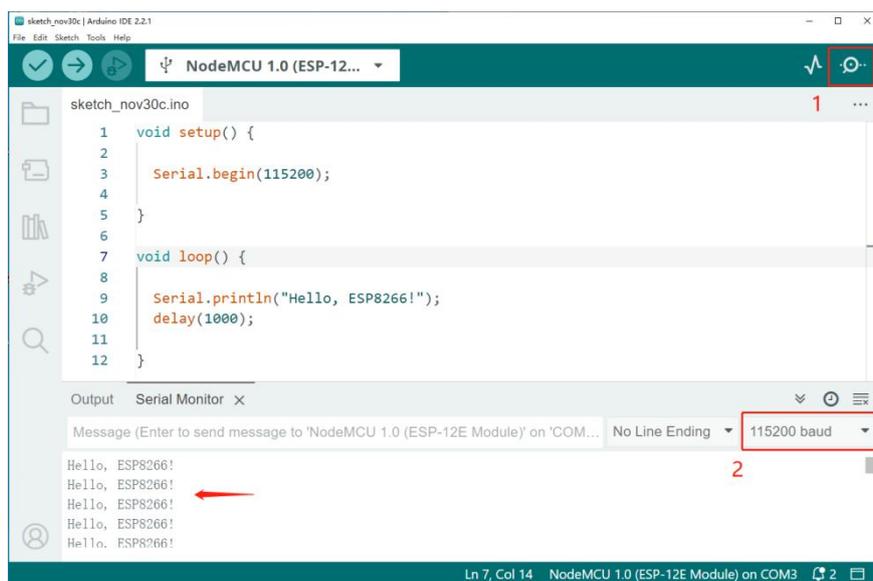
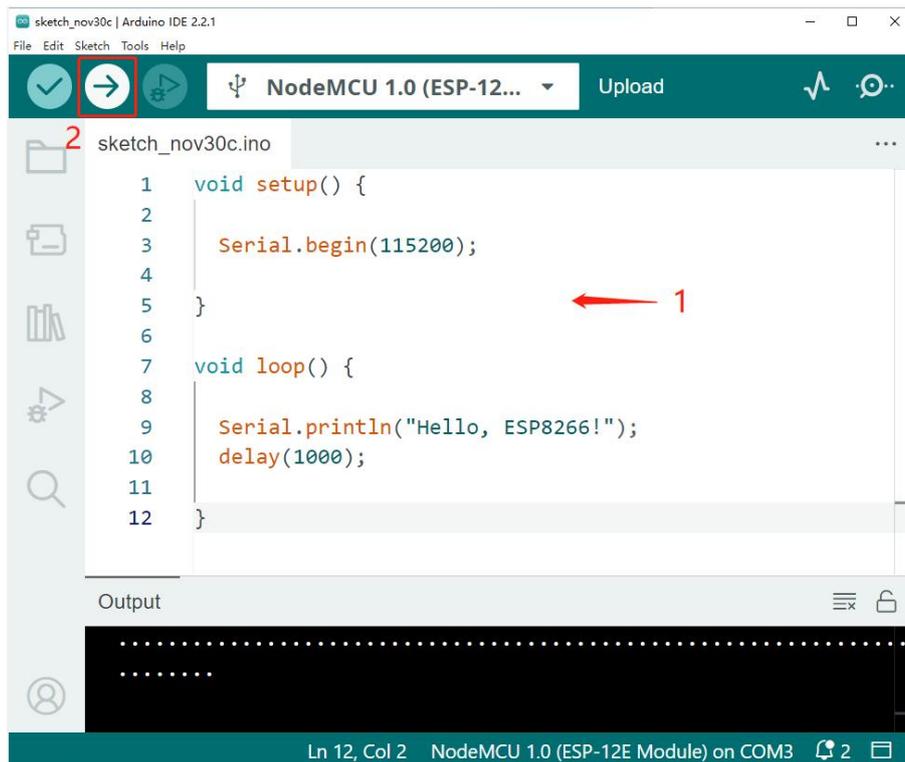


② Vyberte sériový port (můžete zkontrolovat číslo sériového portu v Správci zařízení na vašem počítači a poté zkontrolujte, zda se tento port zobrazuje v nabídce

portů). Každá deska může mít odlišné číslo COM, takže vyberte ten, který odpovídá vašemu aktuálnímu portu.



③ Otevřete soubor "1.1hello_esp32.ino" ve složce "Tutorial2.0\2.Arduino program\Lesson 1\1.1hello_esp32". Připojte ESP8266 vývojovou desku k počítači pomocí USB kabelu a vyberte správnou desku a port v Arduino IDE. Poté nahrajte kód do desky ESP8266. Baudová rychlost by měla být nastavena na 115200. Po nahrání kódu byste měli vidět v sériovém monitoru opakovaný výstup "Hello, ESP8266!".

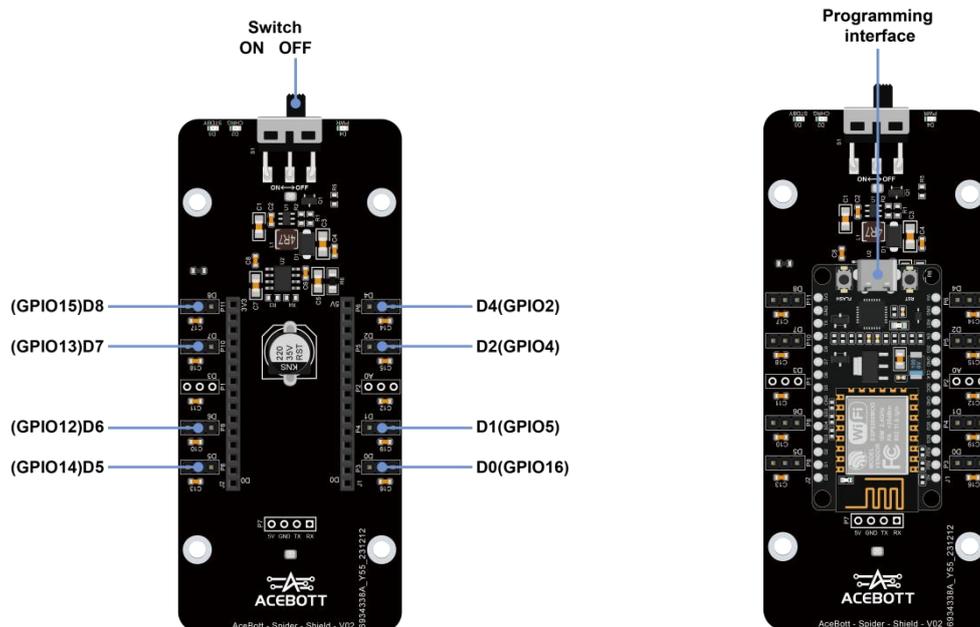


III.Řízení servomotorů

1.Seznámení s rozšiřující deskou pro servomotory

Deskový rozšiřovač pro servomotory slouží hlavně k připojení osmi servomotorů a pod kontrolou hlavní desky ESP8266 řídí pohyby servomotorů. Při použití rozšiřovače pro servomotory dbáte na následující body:

- Připojení k hlavní desce: Zašroubujte piny ESP8266 do zásuvky na rozšiřovači pro servomotory a dejte pozor, aby USB port hlavní desky byl na stejné straně jako přepínač na rozšiřovači.
- Režim nabíjení: Při připojení baterie a přepnutí spínače na POLOHU OFF připojte nabíjecí USB kabel k rozšiřovači pro servomotory.
- Režim používání: Přepněte spínač na POLOHU ON.



2.Program pro řízení servomotorů

(1) Popis pinů na desce rozšiřujícího servomotoru

Číslo	Pinové číslo rozšiřující desky	Číslo pinu GPIO na ESP8266
1	D0	GPIO16
2	D1	GPIO5
3	D2	GPIO4
4	D4	GPIO2
5	D5	GPIO14
6	D6	GPIO12
7	D7	GPIO13
8	D8	GPIO15

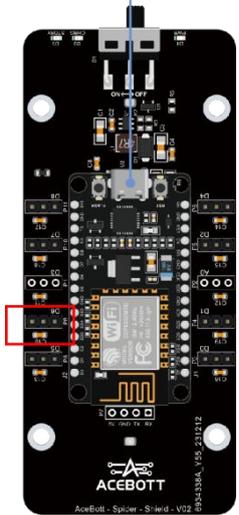
(2) Definice pinů servomotoru

- ① Běžně mají servomotory 3 řídicí dráty: napájecí, zemní a signální.



② Definice pinů servomotoru: Hnědý drát - GND, Červený drát - 5V, Oranžový drát - signál.

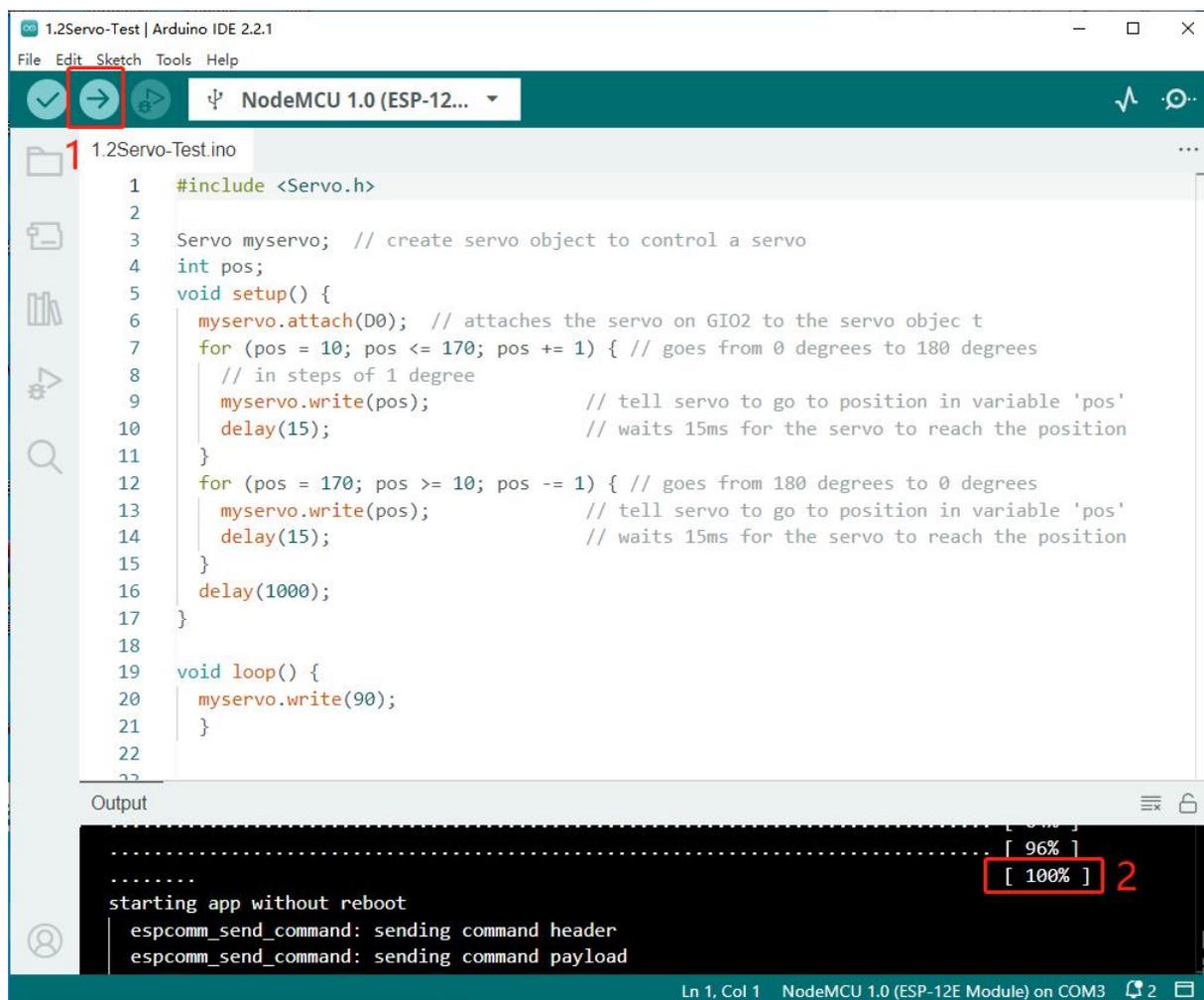
③ Připojení servomotoru k rozšiřovací desce je následující:

Servomotor	Rozšiřující deska	Ilustrace
Hnědý drát	-	
Červený drát	+	
Oranžový drát	D6(GPIO12)	

(3) Kroky pro řízení servomotoru

① Otevřete soubor "1.2Servo-Test.ino" v adresáři "Tutorial2.0\2.Arduino program\Lesson 1\1.2Servo-Test". Pomocí USB kabelu připojte ESP8266 vývojovou desku k počítači a vyberte správnou desku, procesor a port. Nahrajte kód do ESP8266 vývojové desky.

Před nahráním kódu přepněte přepínač servo rozšiřovací desky do polohy OFF; po nahrání kódu přepněte přepínač servo rozšiřovací desky do polohy ON.



② Pokud je servomotor v pořádku, měl by se otáčet od 0 stupňů do 180 stupňů, poté zpět od 180 stupňů do 0 stupňů a nakonec by se měl zastavit ve 90° pozici.

③ Rozbalte dodaných 8 servomotorů z balení a proveďte stejný postup testování s každým z nich, abyste zajistili, že každý servomotor je v pořádku.

Lekce 2: Sestavení robota

I .Seznam příslušenství



II .Seznam konstrukčních dílů



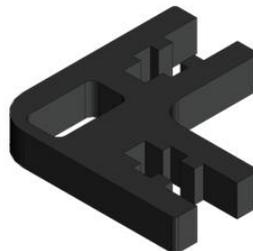
Body x1



Body -Top x1



Coxa x8



Femur x4



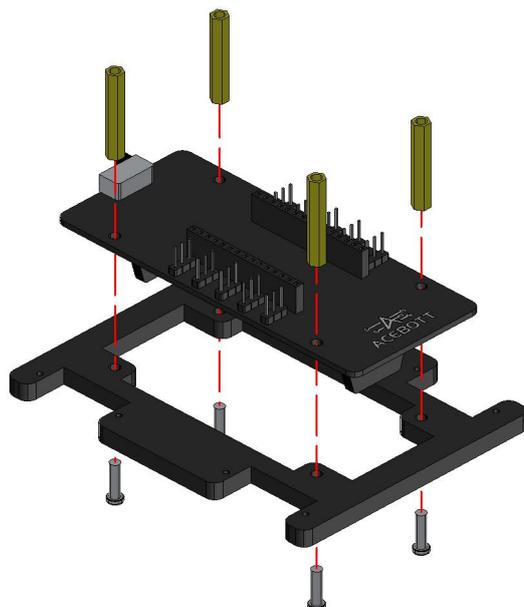
Tibia x4

III. Montážní postup

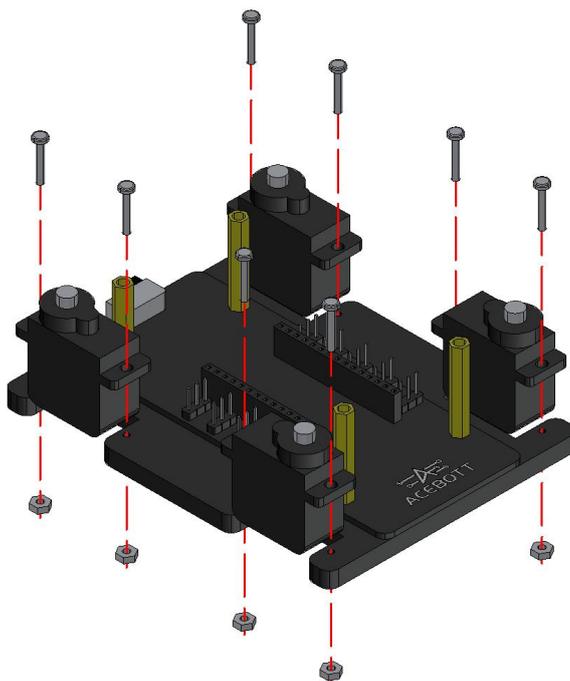
1. Sundání ochranné fólie z akrylátových konstrukčních dílů

2. Instalace rozšiřovací desky pro servomotory na tělo

Použijte 4 ks šroubů M3*10 s kulatou hlavou a 4 ks dvoutřmenových mosazných sloupků M3*25 k upevnění rozšiřovací desky na tělo, jak je znázorněno na následujícím obrázku.

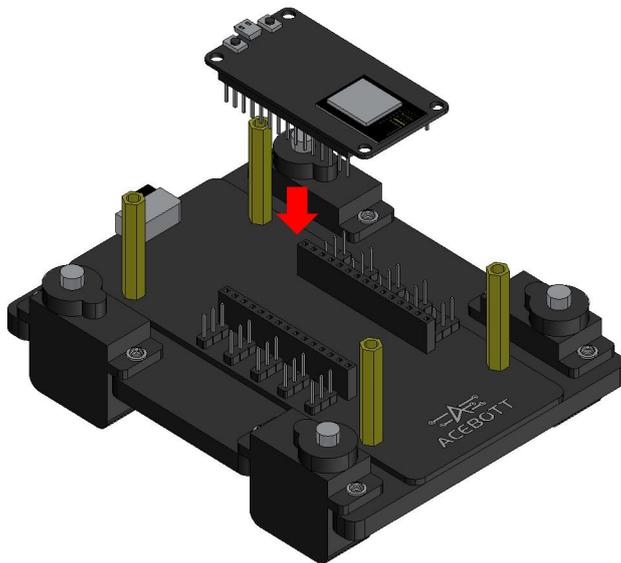


3. Instalace servomotorů na tělo



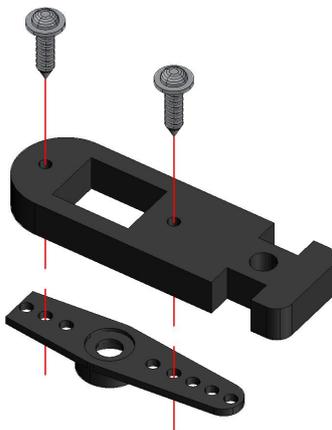
4. Instalace základní desky ESP8266

Instalace základní desky ESP8266 na rozšiřovací desku pro servomotory. USB port základní desky ESP8266 a spínač na rozšiřovací desce pro servomotory by měly být na stejné straně, jak je znázorněno na následujícím obrázku.



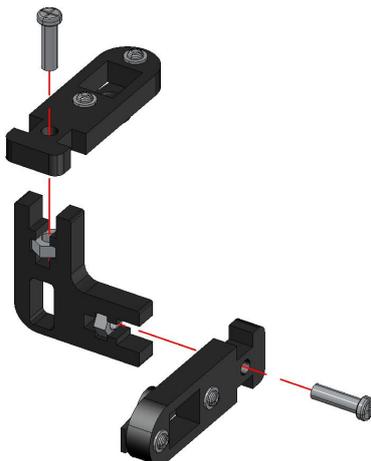
5. Instalace páčky servomotoru na Coxu

Použijte 2 ks samonavrtávacích šroubů s kulatou hlavou M2*8 k upevnění páčky servomotoru a konstrukčního dílu Coxu dohromady, jak je znázorněno na následujícím obrázku.



6. Instalace Coxu a Femuru dohromady

(1) Použijte 2 ks šroubů M3*10 s kulatou hlavou a 2 ks matice M3 k upevnění konstrukčního dílu Coxu a konstrukčního dílu Femuru dohromady, jak je znázorněno na následujícím obrázku.



(2) Instalovaný výsledek by měl vypadat takto.

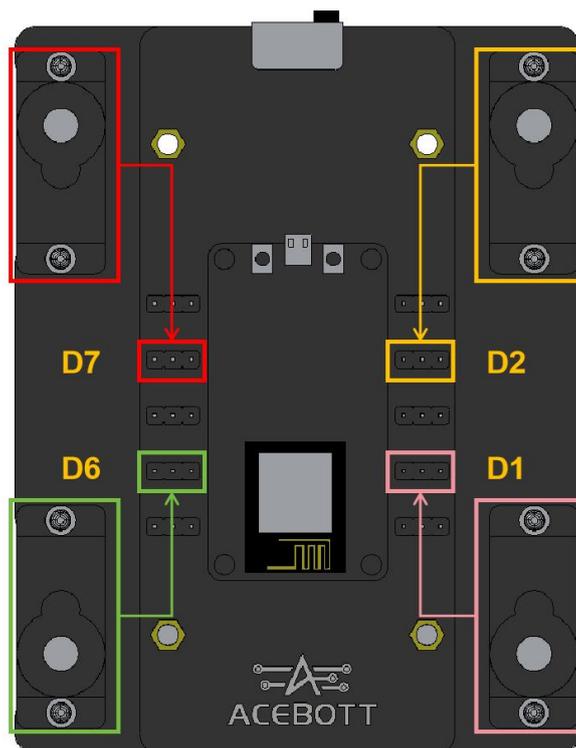


(3) Podobně, instalujte zbývající tři, takže výsledek vypadá jako na následujícím obrázku, pamatujte na správný směr - dva směrem doleva a dva směrem doprava.



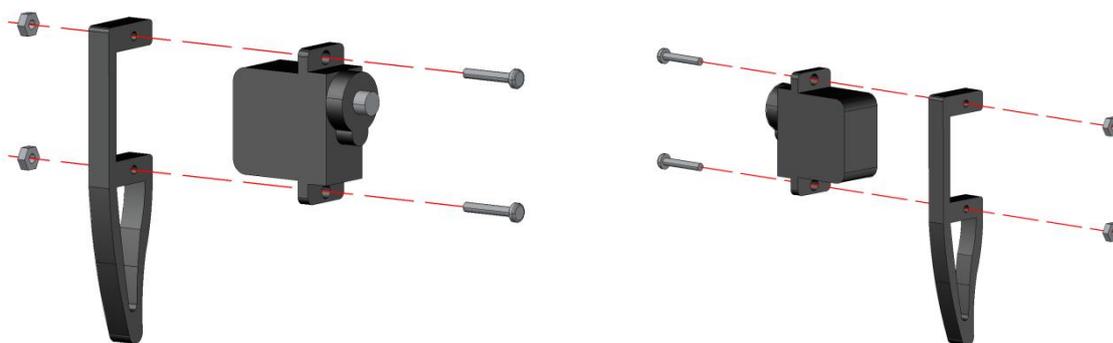
7. Připojte čtyři servomotory na rozšiřovací desku pro servomotory na těle

Vložte kabely servomotoru do řídicí desky podle následujícího obrázku (věnujte pozornost pořadí zapojení servomotoru - hnědá barva v kabelu servomotoru je GND, červená je VCC, žlutá je signálový vodič, nezapomeňte, že nemůžete zapojit kabely servomotoru obráceně).



8. Instalujte čtyři nohy

(1) Použijte 2 ks šroubů M2*14 s kulatou hlavou a 2 ks matic M2 k upevnění servomotoru a konstrukčního dílu Tibie dohromady, jak je znázorněno na následujícím obrázku.



Poznámka: Dva Tibia jsou instalovány podle obrázku vlevo, zatímco další dva jsou instalovány podle obrázku vpravo.

(2) Po instalaci bude výsledek vypadat následovně.

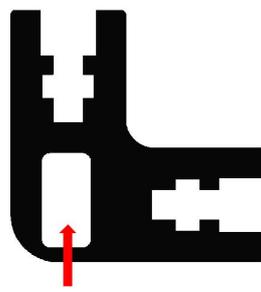


(3) Po instalaci všech čtyř drápků je připevněte k nohám a snažte se udržovat je co nejvíce vodorovně.



Před upevněním čtyř servomotorů na otáčivém kotouči projděte kabely servomotorů otvory pro průchod kabelů, jak je znázorněno na následujícím obrázku.

Poznámka: ① Neukotvujte otáčivý kotouč v této fázi.
② Během instalace nedotýkejte se otáčivého kotouče.



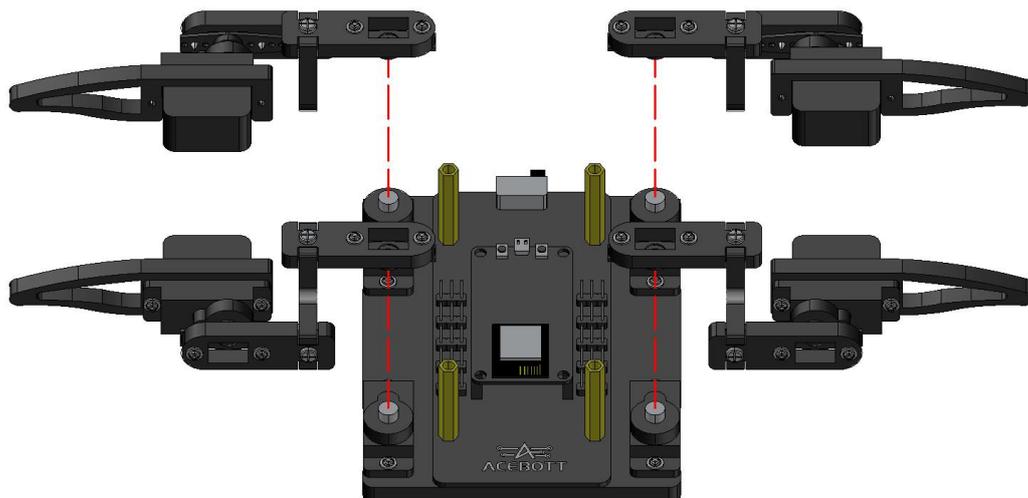
Prolévací otvor

9. Připojte čtyři stehna

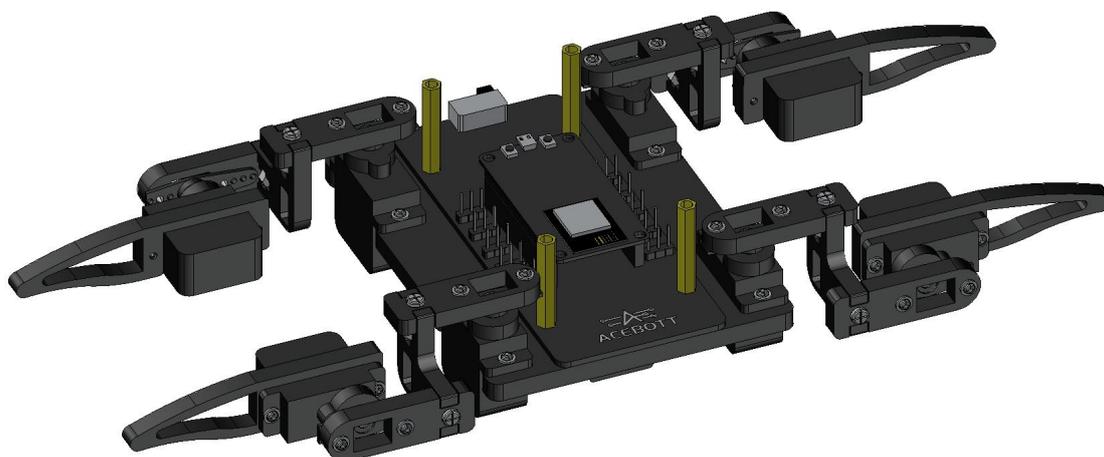
(1) Podle úhlu znázorněného na následujícím obrázku nainstalujte otáčivý kotouč, snažte se udržet co nejvíce svisle.

Poznámka:

- ① Zde zatím nezajišťujte otáčivý kotouč.
- ② Během instalace neotáčejte servomotorem.

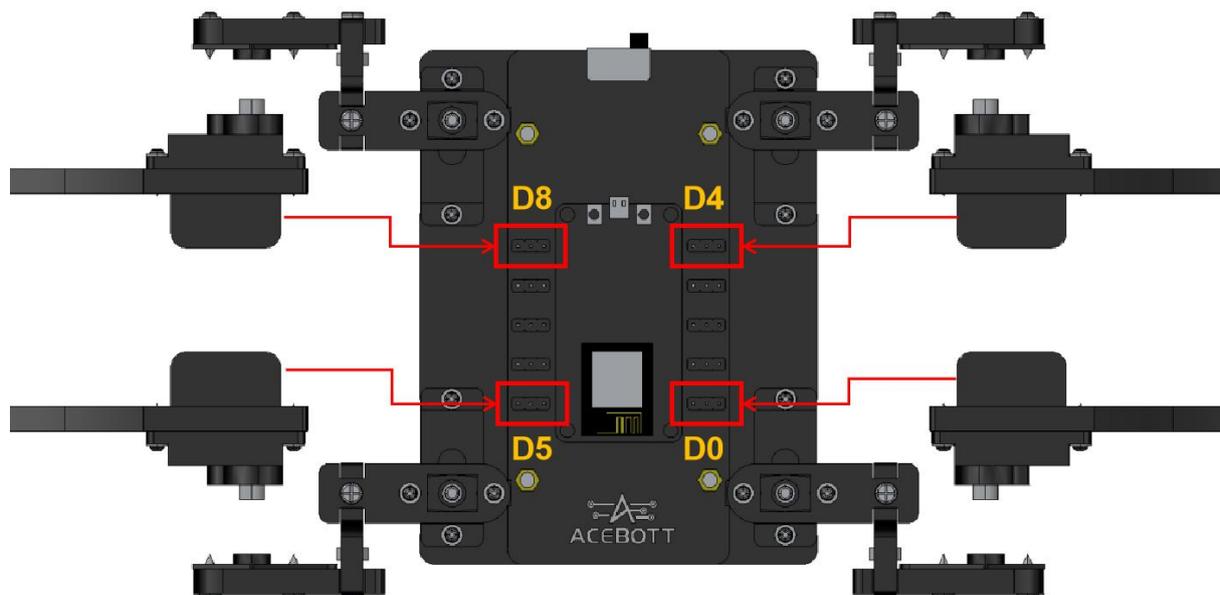


(2) Výsledek po instalaci by měl vypadat jako na následujícím obrázku.



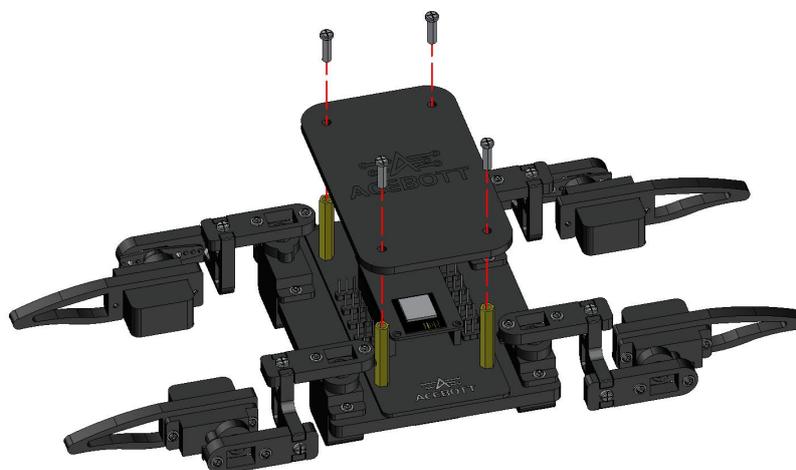
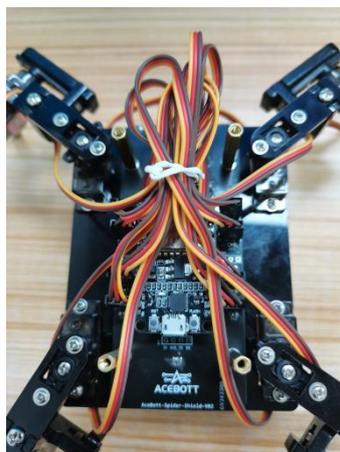
10. Připojení kabelů servomotorů v dolní části nohy

Vložte kabely servomotoru do desky řízení, jak je znázorněno na následujícím obrázku (dávejte pozor na pořadí kabelů servomotoru, hnědý kabel je GND, červený je VCC, žlutý je signálový kabel, důležité je, aby nebyly připojeny naopak).



11.Instalace horní části těla

Po uspořádání kabelů je zafixujte zipovým páskem a upevněte díl Body-Top čtyřmi šrouby M3*10 s kulatou hlavou, jak je znázorněno na obrázku níže.



12. Po dokončení instalace vypadá konečný výsledek takto

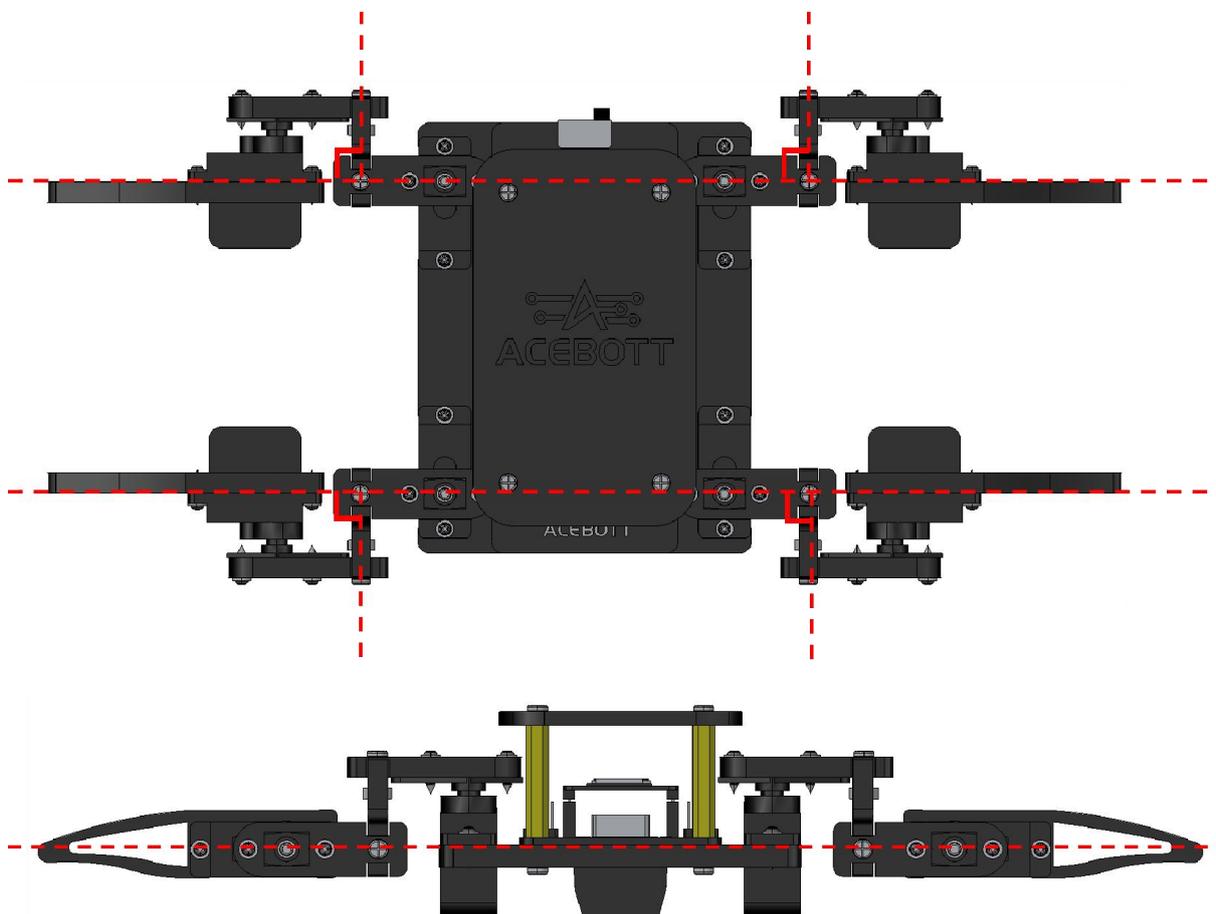


Zde jsme! Čtyřnohý biomechanický pavouk je nyní kompletně sestaven!

Lekce 3: Inicializace polohy

I .Nulová poloha

1.Poloha nula je zobrazena na následujícím obrázku



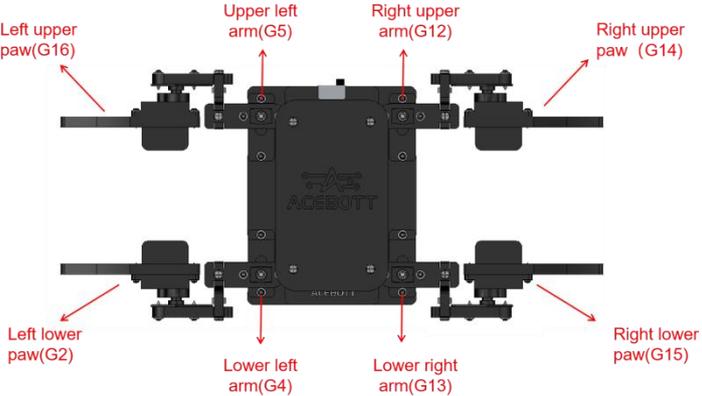
V průběhu montáže jsme robota umístili do počáteční polohy podle návrhu. Tato poloha je počáteční poloha robota, což je jeho nulová poloha. Nicméně, jaká je úhel každého serva v této pozici? Na začátku jsme nezadefinovali úhel každého serva, takže i když má váš sestavený robot nulovou pozici, neznáte úhel každého serva. Po nastavení programu pro nulovou polohu je nutné serva sundat a znovu je namontovat do této nulové polohy.

2.Poloha pinu serva odpovídající pozici

Po instalaci robota každé servo přebírá různé funkce robota. Pokud chceme pomocí programu ovládat úhly každého serva a dosáhnout různých funkcí, musíme znát čísla pinů každého serva.

Robot používá celkem 8 serv, z nichž 4 jsou na vnitřní straně a 4 na vnější straně. Pro snadnější zapamatování nazýváme serva na vnitřní straně robotovými rameny a ta na vnější straně robotovými drápy. Zároveň jsou serva rozdělena podle různých pozic vzhůru, dolů, vpravo a vlevo. Proto jsme 8 poloh serv rozdělili do kategorií: pravý horní dráp, pravý horní rameno, pravý dolní dráp, pravý dolní rameno, levý horní dráp, levý horní rameno, levý dolní dráp, levý dolní rameno.

Jeho odpovídající pinové číslo je uvedeno v následující tabulce:

Číslo	Číslo pinu	Pozice serva	Schéma
1	GPIO14	Right upper paw	
2	GPIO12	Right upper arm	
3	GPIO13	Lower right arm	
4	GPIO15	Right lower paw	
5	GPIO16	Left upper paw	
6	GPIO5	Upper left arm	
7	GPIO4	Lower left arm	
8	GPIO2	Left lower paw	

II .nastavení programu na nulovou pozici

1.Program na nulovou pozici

Otevřete 'Tutorial2.0\2.Arduino program\Lesson 3\3.1zero_0', připojte ESP8266 vývojovou desku k počítači pomocí USB kabelu a vyberte správnou vývojovou desku a port. Nahrajte kód do ESP8266 vývojové desky.

Před nahráním kódu přepněte spínač servo rozšiřující desky do polohy OFF a po nahrání kódu jej přepněte zpět do polohy ON.

Ukázka kódu:

```
#include <Servo.h> //Import servo library
//Create eight servos variables
Servo servo_14;//Upper right [paw]
Servo servo_12;//Upper right [arm]
Servo servo_13;//Lower right [arm]
Servo servo_15;//Lower right [paw]
Servo servo_16;//Upper left [paw]
Servo servo_5;//Upper left [arm]
Servo servo_4;//Lower left [arm]
Servo servo_2;//Lower left [paw]

void setup(){ //Initialize the GPIO pin number of each servo
    servo_14.attach(14);
    servo_12.attach(12);
    servo_13.attach(13);
    servo_15.attach(15);
    servo_16.attach(16);
    servo_5.attach(5);
    servo_4.attach(4);
    servo_2.attach(2);
}

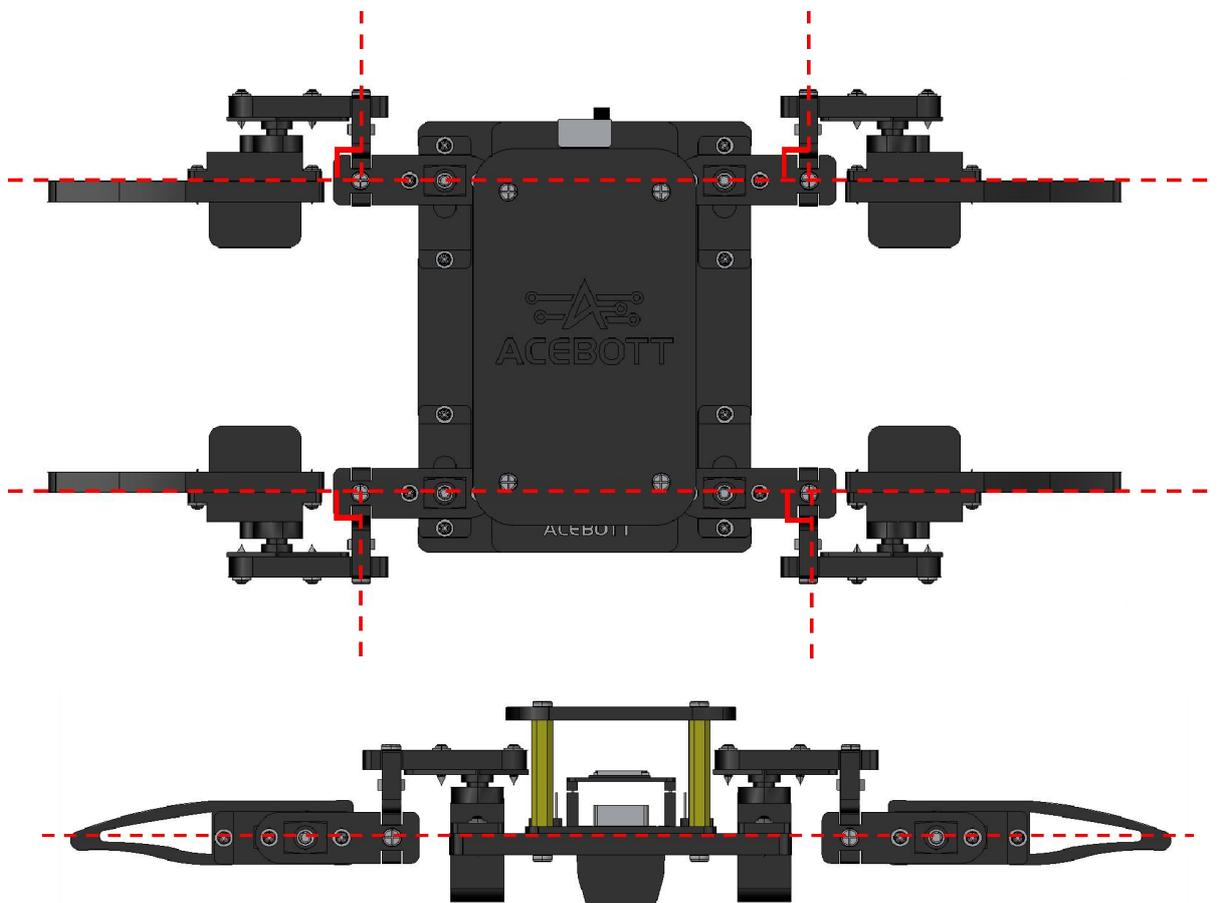
void zero() { //Define the zeroing function
    servo_14.write(135);//Set right upper [paw]135°
    servo_12.write(45);//Set upper right [arm]45°
    servo_13.write(135);//Set lower right [arm]135°
    servo_15.write(45);//Set right lower [paw]45°
    servo_16.write(45);//Set upper left [paw]45°
    servo_5.write(135);//Set upper left [arm]135°
    servo_4.write(45);//Set lower left [arm]45°
    servo_2.write(135);//Set left lower [paw]135°
}

void loop() {
    zero();//Call the zeroing function
}
```

2. Nastavení úhlu servomotorů

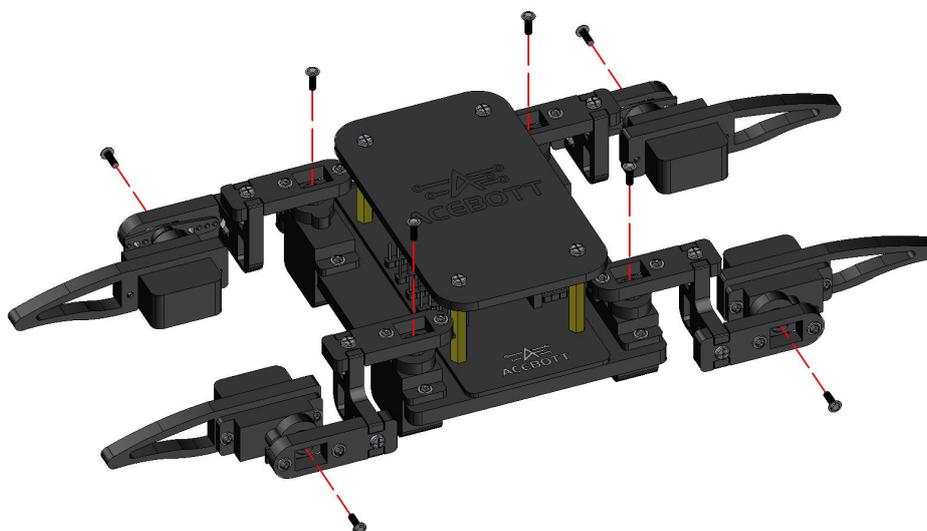
V tuto chvíli vidíme, že servomotory robota neodpovídají nulové poloze (pokud jsou úplně shodné, přeskočte tento krok), takže podle již nahrána programu znovu nainstalujeme servomotory do nulové polohy. Poté se nulový úhel a poloha robota shodují.

Poznámka: Kvůli mezeře ve zubovém kole servomotorů nemusí být možné je přesně nainstalovat do nejlepšího stavu, takže může docházet k mírnému odchýlení. Tento jev je normální.

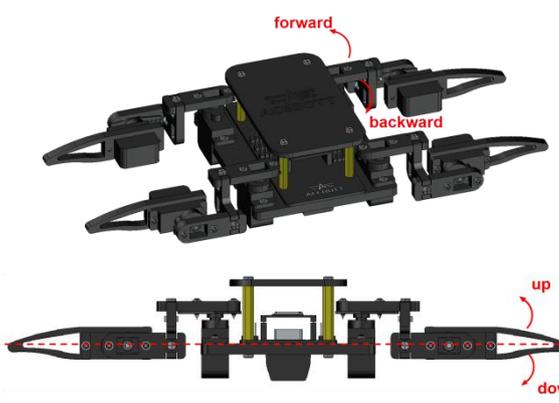


3.Nainstalujte šrouby servohornu

Po finálním nastavení upevněte osm servohornů pomocí osmi kroužkových šroubů M2.5*4.



Pravidla pohybu každého serva na jednotlivých pozicích jsou popsána v následující tabulce:

Číslo	Číslo pinu	Pozice serva	Pohybový režim	Schéma
1	GPIO14	Right upper paw	Úhel je větší, servomotor se pohybuje výše	
2	GPIO12	Right upper arm	Úhel je větší, servomotor se pohybuje dopředu	
3	GPIO13	Lower right arm	Úhel je větší, servomotor se pohybuje dopředu	
4	GPIO15	Right lower paw	Úhel je větší, servomotor se pohybuje dolů	
5	GPIO16	Left upper paw	Úhel je větší, servomotor se pohybuje dolů	
6	GPIO5	Upper left arm	Úhel je větší, servomotor se pohybuje vzad	
7	GPIO4	Lower left arm	Úhel je větší, servomotor se pohybuje vzad	
8	GPIO2	Left lower paw	Úhel je větší, servomotor se pohybuje nahoru	

Lekce 4: Pohyb robotů vpřed a vzad

Základní ovládací pohyby robota zahrnují pohyb vpřed, vzad, doleva a doprava, stejně jako otáčení. Jakmile zvládnete tyto základní pohyby, můžete je kombinovat a vytvářet další složitější pohyby.

I .Program pro režim čekání

Před přechodem na jiné stavy robot potřebuje připravit sebe a své systémy do určitého stavu, který umožní rychlý přechod na další akce.

Otevřete soubor "4.1standby.ino" ve složce Tutorial2.0\2.Arduino program\Lesson 4\4.1standby, připojte ESP8266 vývojovou desku k počítači pomocí USB kabelu a vyberte správnou desku a port. Poté nahrajte kód na ESP8266 vývojovou desku.

Před nahráním kódu přepněte přepínač servo rozšiřující desky do polohy OFF; po nahrání kódu přepněte přepínač do polohy ON.

```
#include <Servo.h>
Servo servo_14;//Upper right [paw]
Servo servo_12;//Upper right [arm]
Servo servo_13;//Lower right [arm]
Servo servo_15;//Lower right [paw]
Servo servo_16;//Upper left [paw]
Servo servo_5;//Upper left [arm]
Servo servo_4;//Lower left [arm]
Servo servo_2;//Lower left [paw]

void setup(){
  servo_14.attach(14);
  servo_12.attach(12);
  servo_13.attach(13);
  servo_15.attach(15);
  servo_16.attach(16);
  servo_5.attach(5);
  servo_4.attach(4);
  servo_2.attach(2);
}
```

```
//Create the standby array
int array1[2][9] = {
//G14 G12 G13 G15 G16 G5 G4 G2 MS
  {90, 90, 90, 90, 90, 90, 90, 90, 500},//Center point position
  {70, 90, 90, 110,110, 90, 90, 70, 500},//Standby mode
};
//A function that sets the standby state
void standby() {
  for (int i = 0; i <= 1; i = i + 1) { //There are two sets of arrays that need to be
traversed twice
    servo_14.write(array1[i][0]); //The servo executes row i and column 1
    servo_12.write(array1[i][1]); //The servo executes row i and column 2
    servo_13.write(array1[i][2]); //The servo executes row i and column 3
    servo_15.write(array1[i][3]); //The servo executes row i and column 4
    servo_16.write(array1[i][4]); //The servo executes row i and column 5
    servo_5.write(array1[i][5]); //The servo executes row i and column 6
    servo_4.write(array1[i][6]); //The servo executes row i and column 7
    servo_2.write(array1[i][7]); //The servo executes row i and column 8
    delay(array1[i][8]); //Row i, column 9, execution delay time
  }
}
//Run the standby function
void loop() {
  standby();
  delay(1000);
}
```

Po nahrání programu zjistíte, že čtyři paže robota jsou uspořádány ve tvaru "X" a čtyři drápy jsou natočeny dovnitř, což je pozice připravená k akci. Tato pozice se nazývá režim přípravy.



II .Program předního pohybu

Otevřete soubor „4.2forward.ino“ v adresáři „Tutorial2.0\2.Arduino program\Lesson 4\4.2forward“. Připojte desku ESP8266 k počítači pomocí USB kabelu a vyberte správnou desku a port v Arduino IDE. Nahrání kódu na desku ESP8266.

Před nahráním přepněte přepínač desky serva na polohu „OFF“, a po nahrání přepněte přepínač zpět na polohu „ON“.

```

/Creates an array of action decompositions for moving forward
int array2[11][9] = { //There are 11 arrays with 9 columns
//G14 G12 G13 G15 G16 G5 G4 G2 MS (The latter is uniformly
sorted according to this pin)
{70, 90, 90, 110, 110, 90, 90, 70, 200},//Ready for standby
{90, 90, 90, 110, 110, 90, 90, 90, 200},//Right upper paw and left lower paw are
raised
{90, 120, 90, 110, 110, 90, 60, 90, 200},//Right upper arm and left lower arm
forward
{70, 120, 90, 110, 110, 90, 60, 70, 200},//Right upper paw and left lower paw drop
{70, 120, 90, 90, 90, 90, 60, 70, 200},//The left upper paw and right lower paw are
raised
{70, 90, 90, 90, 90, 90, 90, 70, 200},//Right upper arm and left lower arm back
{70, 90, 120,90, 90, 60, 90, 70, 200},//Left upper arm and right lower arm forward
{70, 90, 120,110, 110, 60, 90, 70, 200},//Left upper paw and right lower paw drop
{90, 90, 120,110, 110, 60, 90, 90, 200},//Right upper paw and left lower paw are
raised
{90, 90, 90, 110, 110, 90, 90, 90, 200},//The upper left arm and lower right arm
back
{70, 90, 90, 110, 110, 90, 90, 70, 200},//Right upper paw and left lower paw drop
};
//Sets the forward function
void forward() {
    for (int i = 0; i <= 10; i = i + 1) {
        servo_14.write(array2[i][0]);
        servo_12.write(array2[i][1]);
        servo_13.write(array2[i][2]);
        servo_15.write(array2[i][3]);
        servo_16.write(array2[i][4]);
        servo_5.write(array2[i][5]);
        servo_4.write(array2[i][6]);
        servo_2.write(array2[i][7]);
        delay(array2[i][8]);
    }
}
// Run the forward function (to pause, put the swing switch on the servo extension
board into OFF gear)
void loop() {
    forward();// The robot may not necessarily walk in a straight line due to errors
}

```

Po nahrání programu si všimnete, že ramena a klepeta na jedné diagonále robota provedou stejný pohyb, zatímco ramena a klepeta na protilehlé diagonále se pohybují střídavě, což umožňuje pohyb vpřed.



III. Programy, které se pohybují pozpátku

Otevřete soubor „4.3back.ino“ ve složce Tutorial2.0\2.Arduino program\Lesson 4\4.3back. Připojte ESP8266 vývojovou desku k počítači pomocí USB kabelu a vyberte správnou desku a port v Arduino IDE. Nahrávejte kód na ESP8266 vývojovou desku.

Před nahráním kódu nastavte přepínač rozšiřovací desky servomotorů na pozici OFF. Po nahrání kódu přepněte přepínač rozšiřovací desky servomotorů zpět na pozici ON.

```

//Creates a backward-moving action decomposition array
int array3[11][9] = {
  {70, 90, 90, 110, 110, 90, 90, 70, 200},//Ready for standby
  {90, 90, 90, 110, 110, 90, 90, 90, 200},//Right upper paw and left lower paw are
  raised
  {90, 60, 90, 110, 110, 90, 120,90, 200},//Right upper arm and left lower arm back
  {70, 60, 90, 110, 110, 90, 120,70, 200},//Right upper paw and left lower paw drop
  {70, 60, 90, 90, 90, 90, 120,70, 200},//The left upper paw and right lower paw are
  raised
  {70, 90, 90, 90, 90, 90, 90, 70, 200},//Right upper arm and left lower arm forward
  {70, 90, 60, 90, 90, 120,90, 70, 200},//The upper left arm and lower right arm back
  {70, 90, 60,110, 110, 120,90, 70, 200},//Left upper paw and right lower paw drop
  {90, 90, 60,110, 110, 120,90, 90, 200},//Right upper paw and left lower paw are
  raised
  {90, 90, 90, 110, 110, 90, 90, 90, 200},//Left upper arm and right lower arm forward
  {70, 90, 90, 110, 110, 90, 90, 70, 200},//Right upper paw and left lower paw drop
};
//Sets the backoff function
void back() {
  for (int i = 0; i <= 10; i = i + 1) {
    servo_14.write(array3[i][0]);
    servo_12.write(array3[i][1]);
    servo_13.write(array3[i][2]);
    servo_15.write(array3[i][3]);
    servo_16.write(array3[i][4]);
    servo_5.write(array3[i][5]);
    servo_4.write(array3[i][6]);
    servo_2.write(array3[i][7]);
    delay(array3[i][8]);
  }
}
//Running the backoff program
void loop() {
  back();//Calling the backoff function
}

```

Po nahrání programu zjistíte, že končetiny a kleště na diagonále robota provádějí stejný pohyb, a obě dvojice končetin se střídavě pohybují dozadu.



IV.Rozšířené úkoly

Výše uvedené návody již ukázaly, jak se robot pohybuje vpřed a vzad, takže následujícím rozšířeným úkolem ověřte své znalosti. Doporučujeme si nejprve vyzkoušet postup sám a poté si přečíst návod.

(1)popis úkolu

Cyklus pro pohyb robotu vpřed třikrát, poté cyklus pro pohyb vzad dvakrát a nakonec přechod do režimu čekání.

(2)Referenční program

Otevřete soubor '4.4expand.ino' v adresáři Tutorial2.0\2.Arduino program\Lesson 4\4.4.expand, připojte ESP8266 vývojovou desku k počítači pomocí USB kabelu a vyberte správnou desku a port v Arduino IDE. Poté nahrajte kód do desky ESP8266.

Před nahráním kódu přepněte přepínač rozšiřující desky servomotorů do polohy OFF a po nahrání kódu přepněte přepínač do polohy ON.

Upozornění: Inicializace a definice funkcí jsou stejné jako v předchozích částech, zde je zobrazena pouze část kódu pro spuštění programu.

```
void loop() {
  for (int i = 0; i <= 2; i = i + 1){
    forward();//Move forward three times
  }
  for (int i = 0; i <= 1; i = i + 1){
    back();//Move backwards twice
  }
  standby();//Standby mode
}
}
```

Lekce 5: Otočení robota doleva a doprava

V předchozích lekcích jsme se už naučili programy pro pohyb robota vpřed a vzad, ale v této lekci budeme pokračovat a naučíme se otočení robota doleva a doprava. Tato funkce je pro pohyb robota velmi důležitá, protože umožňuje robotovi dostat se na jakékoliv místo na rovině.

Pokud chcete změnit úhel otočení, musíte upravit úhly G12, G4, G13 a G5. Kvůli chybám v reálném otočení a tření jsou úhly otočení obvykle jen polovinou úhlu otáčení servo motorků v pozici "paže". Například od 90° do 135°, tj. otočení o úhel 45°, ale skutečný úhel otočení bude pouze 22,5°. Pokud změníte úhel z 90° na 150°, bude úhel otočení 60 stupňů, takže skutečný úhel otočení bude 30°, a tak dále. Avšak kvůli konstrukci jsou úhly otočení servo motorků robota obvykle omezeny, proto se nedoporučuje překračovat úhly otočení přes 60°.

I .Program pro otočení doleva

Otevřete "5.1turn_left.ino" v Tutorial2.0\2.Arduino program\Lesson 5, připojte ESP8266 vývojovou desku pomocí USB kabelu k počítači a vyberte správnou vývojovou desku a port. Nahrajte kód na ESP8266 vývojovou desku.

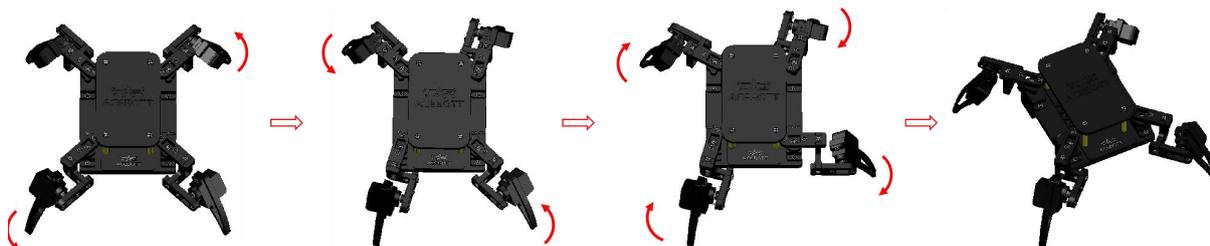
Před nahráním nastavte spínač rozšiřovací desky serv a po nahrání kódu ho vraťte do polohy ON.

```

//Create an array of turns to the left. By default, the robot rotates 22.5 degrees
int array6[8][9] = {
  //G14 G12 G13 G15 G16 G5 G4 G2 MS
  {70, 90, 90, 110, 110, 90, 90, 70, 200},//Ready for standby
  {90, 90, 90, 110, 110, 90, 90, 90, 200},//Right upper paw and left lower paw are
  raised
  {90, 135, 90, 110, 110, 90, 135, 90, 200},//Right upper arm forward, left lower
  arm back
  {70, 135, 90, 110, 110, 90, 135, 70, 200},//Right upper paw and left lower paw
  drop
  {70, 135, 90, 90, 90, 90, 135, 70, 200},//The left upper paw and right lower paw
  are raised
  {70, 135, 135, 90, 90, 135, 135, 70, 200},//Put your upper left arm back and your
  lower right arm forward
  {70, 135, 135, 110, 110, 135, 135, 70, 200},//Left upper paw and right lower paw
  drop
  {70, 90, 90, 110, 110, 90, 90, 70, 200},//Four arms rotation
};
//A function that sets the left turn
void turnleft() {
  for (int i = 0; i <= 7; i = i + 1) {
    servo_14.write(array6[i][0]);
    servo_12.write(array6[i][1]);
    servo_13.write(array6[i][2]);
    servo_15.write(array6[i][3]);
    servo_16.write(array6[i][4]);
    servo_5.write(array6[i][5]);
    servo_4.write(array6[i][6]);
    servo_2.write(array6[i][7]);
    delay(array6[i][8]);
  }
}
// Run the left turn (initialization and function definition are the same as before)
void loop() {
  turnleft();
}

```

Nahráním programu zjistíte, že robot nejprve provede rotaci čtyř paží a nakonec provede rotaci těla doleva.



II .Program pro otočení doprava

5.2turn_right.ino" soubor v Tutorial2.0\2.Arduino program\Lesson 5\5.2turn_right otevřete a pomocí USB kabelu připojte desku ESP8266 k počítači. Vyberte správnou desku a port a nahrát kód do desky ESP8266.

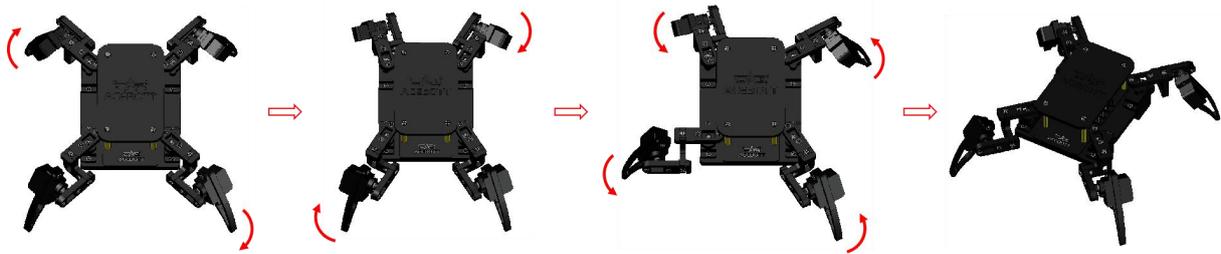
Před nahráním programu přepněte přepínač rozšiřovací desky serv do polohy OFF; po nahrání přepněte přepínač zpět do polohy ON.

```

//Create an array of actions for turning right (for setting the rotation Angle, see
instructions for turning left arrays)
int array7[8][9] = {
    {70, 90, 90, 110, 110, 90, 90, 70, 200},//Ready for standby
    {70, 90, 90, 90, 90, 90, 90, 70, 200},//The left upper paw and right lower paw
are raised
    {70, 90, 45, 90, 90, 45, 90, 70, 200},//The upper left arm goes forward and the
lower right arm goes back
    {70, 90, 45, 110, 110, 45, 90, 70, 200},//Left upper paw and right lower paw
drop
    {90, 90, 45, 110, 110, 45, 90, 90, 200},//Right upper paw and left lower paw are
raised
    {90, 45, 45, 110, 110, 45, 45, 90, 200},//Right upper arm back, left lower arm
forward
    {70, 45, 45, 110, 110, 45, 45, 70, 200},//Right upper paw and left lower paw
drop
    {70, 90, 90, 110, 110, 90, 90, 70, 200},//Four arms rotation
};
//A function that sets the right turn
void turnright() {
    for (int i = 0; i <= 7; i = i + 1) {
        servo_14.write(array7[i][0]);
        servo_12.write(array7[i][1]);
        servo_13.write(array7[i][2]);
        servo_15.write(array7[i][3]);
        servo_16.write(array7[i][4]);
        servo_5.write(array7[i][5]);
        servo_4.write(array7[i][6]);
        servo_2.write(array7[i][7]);
        delay(array7[i][8]);
    }
}
//Run the right turn
void loop() {
    turnright();
}

```

Program nahrán, můžete si všimnout, že robot nejprve provede rotaci paží a poté provede poslední rotaci těla směrem doprava.

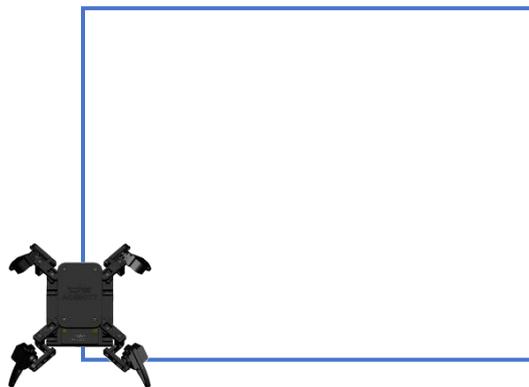


III.rozšířený úkol

Výše uvedený výukový program už ukázal, jak robot může provádět pohyb doleva a doprava. Proto nyní přichází na řadu rozšiřující úkol, který vám umožní otestovat vaše znalosti. Doporučujeme nejprve vyzkoušet úkol sami a poté se podívat na výukový program.

(1)popis úkolu

Kombinujte obsah předchozích lekcí a vytvořte program pro robota, který se pohybuje podobně jako čtverec.



(2)Referenční program

Otevřete „Tutorial2.0\2.Arduino program\Lesson 5\5.3expand2.ino“, připojte ESP8266 vývojovou desku k počítači pomocí USB kabelu a vyberte správnou vývojovou desku a port. Poté nahrajte kód do ESP8266 vývojové desky.

Před nahráním kódu nastavte přepínač servo rozšiřovací desky do polohy OFF a po nahrání kódu přepněte přepínač servo rozšiřovací desky do polohy ON.

Pozor: Parametry programu je třeba upravit podle skutečného výsledku provozu.

```
//Four edges need to repeat forward and rotation actions four times, preferentially  
using the traversal loop.
```

```
void loop() {  
  for (int i = 0; i <= 3; i = i + 1){  
    for (int i = 0; i <= 2; i = i + 1){  
      forward(); //Three times forward  
    }  
    for (int i = 0; i <= 3; i = i + 1){  
      turnright(); //One rotate it four times to the right, which is about 90 degrees  
    }  
  }  
}
```

Lekce 6: Pohyb robota doleva a doprava

V minulých lekcích jsme se již naučili, jak robot pohybovat vpřed a vzad, a také jak ho otáčet doleva a doprava. Nicméně, robot má i další zajímavé způsoby pohybu, jako je například chůze na stranu. Takže nyní se společně podíváme na to, jak robot pohybovat do stran.

I .Jedna, program pro pohyb doleva

Otevřete soubor 6.1left_move.ino ve složce Tutorial2.0\2.Arduino program\Lesson 6\6.1left_move. Připojte ESP8266 vývojovou desku k počítači pomocí USB kabelu a vyberte správnou vývojovou desku a port. Nahrajte kód do ESP8266 vývojové desky.

Před nahráním kódu přesuňte přepínač rozšiřovací desky serv do polohy OFF; po nahrání kódu přesuňte přepínač zpět do polohy ON.

```
//Create an array of moves to the left
int array4[11][9] = {
  {70, 90, 90, 110, 110, 90, 90, 70, 200},//Ready for standby
  {70, 90, 90, 90, 90, 90, 90, 70, 200},//The left upper paw and right lower paw
are raised
  {70, 90, 60, 90, 90, 120,90, 70, 200},//The upper left arm and lower right arm
back
  {70, 90, 60, 110, 110, 120,90, 70, 200},//Left upper paw and right lower paw
drop
  {90, 90, 60, 110, 110, 120,90, 90, 200},//Right upper paw and left lower paw are
raised
  {90, 90, 90, 110, 110, 90, 90, 90, 200},//Left upper arm and right lower arm
forward
  {90, 120,90, 110, 110, 90, 60, 90, 200},//Right upper arm and left lower arm
forward
  {70, 120,90, 110, 110, 90, 60, 70, 200},//Right upper paw and left lower paw
drop
  {70, 120,90, 90, 90, 90, 60, 70, 200},//The left upper paw and right lower paw
are raised
  {70, 90, 90, 90, 90, 90, 90, 70, 200},//Right upper arm and left lower arm back
  {70, 90, 90, 110, 110, 90, 90, 70, 200},//Left upper paw and right lower paw
drop
};
```

```
//Sets the function to move to the left
void leftmove() {
  for (int i = 0; i <= 10; i = i + 1) {
    servo_14.write(array4[i][0]);
    servo_12.write(array4[i][1]);
    servo_13.write(array4[i][2]);
    servo_15.write(array4[i][3]);
    servo_16.write(array4[i][4]);
    servo_5.write(array4[i][5]);
    servo_4.write(array4[i][6]);
    servo_2.write(array4[i][7]);
    delay(array4[i][8]);
  }
}
// Run the program moved to the left
void loop() {
  leftmove();
}
```

Po nahrání programu si všimněte, že pohyb robotu doleva je velmi podobný pohybu vpřed a vzad, jen směr je opačný.



II .Program pro pohyb doprava

Otevřete „6.2right_move.ino“ v Tutorial2.0\2.Arduino program\Lesson 6\6.2right_move, připojte ESP8266 vývojovou desku k počítači pomocí USB kabelu, vyberte správnou vývojovou desku a port a nahrajte kód do ESP8266 vývojové desky.

Před nahráním kódu přepněte přepínač rozšiřující desky serv motorů do polohy OFF a po nahrání kódu přepněte přepínač do polohy ON.

```

//Create an array of actions to move to the right
int array5[11][9] = {
    {70, 90, 90, 110, 110, 90, 90, 70, 200},//Ready for standby
    {90, 90, 90, 110, 110, 90, 90, 90, 200},//Right upper paw and left lower paw are
raised
    {90, 60, 90, 110, 110, 90, 120,90, 200},//Right upper arm and left lower arm
back
    {70, 60, 90, 110, 110, 90, 120,70, 200},//Right upper paw and left lower paw
drop
    {70, 60, 90, 90, 90, 90, 120,70, 200},//The left upper paw and right lower paw
are raised
    {70, 90, 90, 90, 90, 90, 90, 70, 200},//Right upper arm and left lower arm
forward
    {70, 90, 120,90, 90, 60, 90, 70, 200},//Left upper arm and right lower arm
forward
    {70, 90, 120,110, 110, 60, 90, 70, 200},//Left upper paw and right lower paw
drop
    {90, 90, 120,110, 110, 60, 90, 90, 200},//Right upper paw and left lower paw are
raised
    {90, 90, 90, 110, 110, 90, 90, 90, 200},//The upper left arm and lower right arm
back
    {70, 90, 90, 110, 110, 90, 90, 70, 200},//Right upper paw and left lower paw
drop
};
//Sets the function to move to the right
void rightmove() {
    for (int i = 0; i <= 10; i = i + 1) {
        servo_14.write(array5[i][0]);
        servo_12.write(array5[i][1]);
        servo_13.write(array5[i][2]);
        servo_15.write(array5[i][3]);
        servo_16.write(array5[i][4]);
        servo_5.write(array5[i][5]);
        servo_4.write(array5[i][6]);
        servo_2.write(array5[i][7]);
        delay(array5[i][8]);
    }
}
// Run the program moved to the right
void loop() {
    rightmove();
}

```



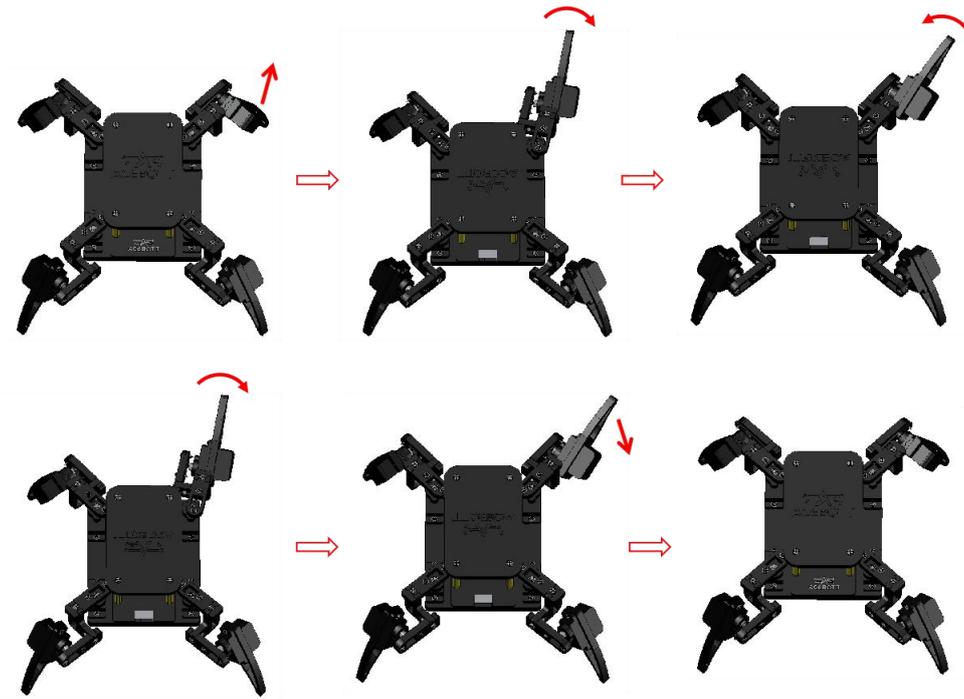
III. rozšířený úkol

Na základě relativně snadných pohybů doleva a doprava by bylo skvělé, kdyby se robot po pohybu mohl pozdravit s diváky, že?

Otevřete soubor "6.3say_hello.ino" ve složce "Tutorial2.0\2.Arduino program\Lesson 6\6.3say_hello". Poté připojte vývojovou desku ESP8266 k počítači pomocí USB kabelu, vyberte správnou vývojovou desku a port a nahrajte kód do vývojové desky ESP8266.

Před nahráním kódu se ujistěte, že přepínač rozšiřující desky serv je v poloze OFF; po nahrání kódu přepněte přepínač rozšiřující desky serv do polohy ON.

```
//Create an array of hello actions
int array8[7][9] = {
    {70, 90, 135, 90, 90, 90, 90, 90, 400},//Left upper paw, right lower paw, left
lower paw down
    {170, 90, 135, 90, 90, 90, 90, 90, 400},//Right upper paw raised
    {170, 130, 135, 90, 90, 90, 90, 90, 400},//Swing your right upper arm forward
    {170, 50, 135, 90, 90, 90, 90, 90, 400},//Swing your right upper arm back
    {170, 130, 135, 90, 90, 90, 90, 90, 400},//Swing your right upper arm forward
    {170, 90, 135, 90, 90, 90, 90, 90, 400},//Swing your right upper arm back
    {70, 90, 135, 90, 90, 90, 90, 90, 400},//Right upper arm down
};
//Set the hello function
void hello() {
    for (int i = 0; i <= 6; i = i + 1) {
        servo_14.write(array8[i][0]);
        servo_12.write(array8[i][1]);
        servo_13.write(array8[i][2]);
        servo_15.write(array8[i][3]);
        servo_16.write(array8[i][4]);
        servo_5.write(array8[i][5]);
        servo_4.write(array8[i][6]);
        servo_2.write(array8[i][7]);
        delay(array8[i][8]);
    }
}
// Run the hello program
void loop() {
    hello();
}
```



Lekce 7: Program tance robotů v

Na konci jsme konečně dokončili učení základních pohybů, rotací a translací robota. Věřím, že už máte nějaké nápady, které byste chtěli co nejdříve realizovat na robotovi. V této lekci se naučíte, jak realizovat několik tanečních pohybů robota, doufám, že vám to otevře nové možnosti a inspiruje vás k vytvoření ještě zábavnějších efektů.

I .Základní taneční krok

Otevřete soubor '7.1dance1.ino' v adresáři Tutorial2.0\2.Arduino program\Lesson 7\7.1dance1". Připojte ESP8266 vývojovou desku k počítači pomocí USB kabelu a vyberte správnou desku a port pro nahrání kódu na ESP8266 v Arduino IDE.

Před nahráním programu přepněte přepínač rozšiřovací desky serv u na OFF a po nahrání programu přepněte přepínač na ON.

```

///Create an action array for the elementary step
int array9[10][9] = {
{90, 90, 90, 90, 90, 90, 90, 90, 400},//Lift all four paws
{50, 90, 90, 90, 90, 90, 90, 90, 400},//Right upper paw down
{90, 90, 90, 130, 90, 90, 90, 90, 400},//Right upper paw up, right lower paw down
{90, 90, 90, 90, 90, 90, 90, 50, 400},//Right lower paw up , left lower paw down
{90, 90, 90, 90, 130, 90, 90, 90, 400},//Left upper paw down, left lower paw up
{50, 90, 90, 90, 90, 90, 90, 90, 400},//Left upper paw up, right upper paw down
{90, 90, 90, 130, 90, 90, 90, 90, 400},//Right upper paw up, right lower paw down
{90, 90, 90, 90, 90, 90, 90, 50, 400},//Right lower paw up, left lower paw down
{90, 90, 90, 90, 130, 90, 90, 90, 400},//Left upper paw down, left lower paw up
{90, 90, 90, 90, 90, 90, 90, 90, 400},//Left upper paw up
};

//A function to set the elementary steps
void dance1() {
  for (int i = 0; i <= 9; i = i + 1) {
    servo_14.write(array9[i][0]);
    servo_12.write(array9[i][1]);
    servo_13.write(array9[i][2]);
    servo_15.write(array9[i][3]);
    servo_16.write(array9[i][4]);
    servo_5.write(array9[i][5]);
    servo_4.write(array9[i][6]);
    servo_2.write(array9[i][7]);
    delay(array9[i][8]);
  }
}

//Run the program for the elementary step
void loop() {
  dance1();
  delay(1000);
}

```

Výše uvedené základní tance ukazují, že každá drápka robota provádí pohyb nahoru a dolů v pořadí. Nyní můžeme přidat trochu složitější pohyby na tuto základní techniku.



II .Středně pokročilé taneční kroky

Otevřete soubor 7.2dance2.ino v adresáři Tutorial2.0\2.Arduino program\Lesson 7\7.2dance2 a pomocí USB kabelu připojte vývojovou desku ESP8266 k počítači. Vyberte správnou desku a port a nahrajte kód na vývojovou desku ESP8266.

Před nahráním přepněte spínač rozšiřovací desky serv na pozici OFF a po nahrání zpět na pozici ON.

```
//Create an action array for intermediate steps
int array10[9][9] = {
    {70, 45, 135, 110, 110, 135, 45, 70, 400},//The four arms are left and right
    together
    {115, 45, 135, 65, 110, 135, 45, 70, 400},//Right upper paw and right lower paw
    are raised
    {70, 45, 135, 110, 65, 135, 45, 115, 400},//Right upper and lower paws down,left
    upper and lower paws up
    {115, 45, 135, 65, 110, 135, 45, 70, 400},//Left upper and lower paws down,right
    upper and lower paws up
    {70, 45, 135, 110, 65, 135, 45, 115, 400},//Right upper and lower paws down,left
    upper and lower paws up
    {115, 45, 135, 65, 110, 135, 45, 70, 400},//Left upper and lower paws down,right
    upper and lower paws up
    {70, 45, 135, 110, 65, 135, 45, 115, 400},//Right upper and lower paws down,left
    upper and lower paws up
    {115, 45, 135, 65, 110, 135, 45, 70, 400},//Left upper and lower paws down,right
    upper and lower paws up
    {75, 45, 135, 105, 110, 135, 45, 70, 400},//Right upper paw and right lower paw
    down
};

//A function to set intermediate steps
void dance2() {
    for (int i = 0; i <= 8; i = i + 1) {
        servo_14.write(array10[i][0]);
        servo_12.write(array10[i][1]);
        servo_13.write(array10[i][2]);
        servo_15.write(array10[i][3]);
        servo_16.write(array10[i][4]);
        servo_5.write(array10[i][5]);
        servo_4.write(array10[i][6]);
        servo_2.write(array10[i][7]);
        delay(array10[i][8]);
    }
}

//Run the intermediate step program
void loop() {
    dance2();
    delay(1000);
}
```

Středně těžké tance vypadají dobře? Můžete vidět plynulé pohyby robota doprava a doleva, což je již o mnoho bohatší, následuje pokročilé tance.



III. pokročilé tance

Opened "7.3dance3.ino" in Tutorial2.0\2.Arduino program\Lesson 7\7.3dance3. Connected ESP8266 development board to the computer using a USB cable, selected the correct development board and port, and burned the code to the ESP8266 development board.

Before uploading, switched the toggle switch of the servo expansion board to the OFF position; after uploading, switched the toggle switch of the servo expansion board to the ON position.

```
//Create an action array for advanced steps
int array11[10][9] = {
  {70, 45, 45, 110, 110, 135, 135, 70, 400},//All four arms back
  {90, 45, 45, 60, 90, 135, 135, 70, 400},//Right upper paw, left upper paw,
right lower paw up
  {70, 45, 45, 110, 110, 135, 135, 70, 400},//Right upper paw, left upper paw
and right lower paw down
  {90, 45, 45, 110, 90, 135, 135, 120, 400},//Right upper paw, left upper paw,
left lower paw up
  {70, 45, 45, 110, 110, 135, 135, 70, 400},//Right upper paw, left upper paw
and left lower paw down
  {90, 45, 45, 60, 90, 135, 135, 70, 400},//Right upper paw, left upper paw,
right lower paw up
  {70, 45, 45, 110, 110, 135, 135, 70, 400},//Right upper paw, left upper paw
and right lower paw down
  {90, 45, 45, 110, 90, 135, 135, 120, 400},//Right upper paw, left upper paw,
left lower paw up
  {70, 45, 45, 110, 110, 135, 135, 70, 400},//Right upper paw, left upper paw
and left lower paw down
  {70, 90, 90, 110, 110, 90, 90, 70, 400},//Standby mode
};

//A function to set advanced steps
void dance3() {
  for (int i = 0; i <= 9; i = i + 1) {
    servo_14.write(array11[i][0]);
    servo_12.write(array11[i][1]);
    servo_13.write(array11[i][2]);
    servo_15.write(array11[i][3]);
    servo_16.write(array11[i][4]);
    servo_5.write(array11[i][5]);
    servo_4.write(array11[i][6]);
    servo_2.write(array11[i][7]);
    delay(array11[i][8]);
  }
}

//Run the advanced step program
void loop() {
  dance3();
  delay(1000);
}
```

Pokročilý tanec ukazuje, jak robot provádí cviky známé jako „kliky“ - levá dolní paže a pravá dolní paže střídavě klesají a stoupají, což vytváří dojem, že robot podpírá tělo pomocí „obou paží a jedné nohy“. Je to opravdu úžasné.



IV.rozšířený úkol

Naučili jste se výše uvedené tři druhy tanců, nezjistili jste, že robot je celkem zábavný? S pomocí různých sestav tanců může robot předvést různé pohyby. Věřím, že už máte chuť to vyzkoušet. Pokračujeme rozšířením úkolu k dalšímu zlepšení vašich dovedností.

(1)popis úkolu

Aplikujte sériovou komunikaci pro ovládání tance robota. Například: po zadání "1" přes sériovou linku, robot zahraje začáteční taneční krok; po zadání "2" zahraje středně pokročilý taneční krok; po zadání "3" zahraje pokročilý taneční krok.

(2)Referenční program

Otevřete soubor '7.4expand.ino' ve složce Tutorial2.0\2.Arduino program\Lesson 7, připojte vývojovou desku ESP8266 k počítači pomocí USB kabelu a vyberte správnou vývojovou desku a sériový port. Poté nahrajte kód do vývojové desky ESP8266.

Před nahráním kódu nastavte přepínač rozšiřovací desky servomechanismů na pozici OFF; po nahrání kódu přepínač nastavte na pozici ON.

Nastavte baudovou rychlost.

```
Serial.begin(115200); //The baud rate of esp8266 serial communication is
```

Vytvořte novou proměnnou pro přijetí dat ze sériového portu.

```
String item; //Because serial data is in string format, the variables are also in string format
item = ""; //Initialize the variable as an empty string
```

Programová logika.

```
//Determine whether there is data in the serial port
if (Serial.available()) {
  item = Serial.readString(); //The variable is assigned to the data read by the
  serial port
  Serial.println(item); //The serial port prints out the input data

//According to the input data, the condition is judged and the corresponding
function is executed
if (item == "1") { //When 1 is input to the serial port, dance1 is executed
  dance1();
}
if (item == "2") { //When 2 is input to the serial port, dance2 is executed
  dance2();
}
if (item == "3") { //When 3 is input to the serial port, dance3 is executed
  dance3();
}
```

Ujistěte se, že v sériovém monitoru není nastaven žádný konec řádku a že rychlost přenosu je nastavena na 115200.



Lekce 8: Ovládání robota přes WiFi

I .APP stahování

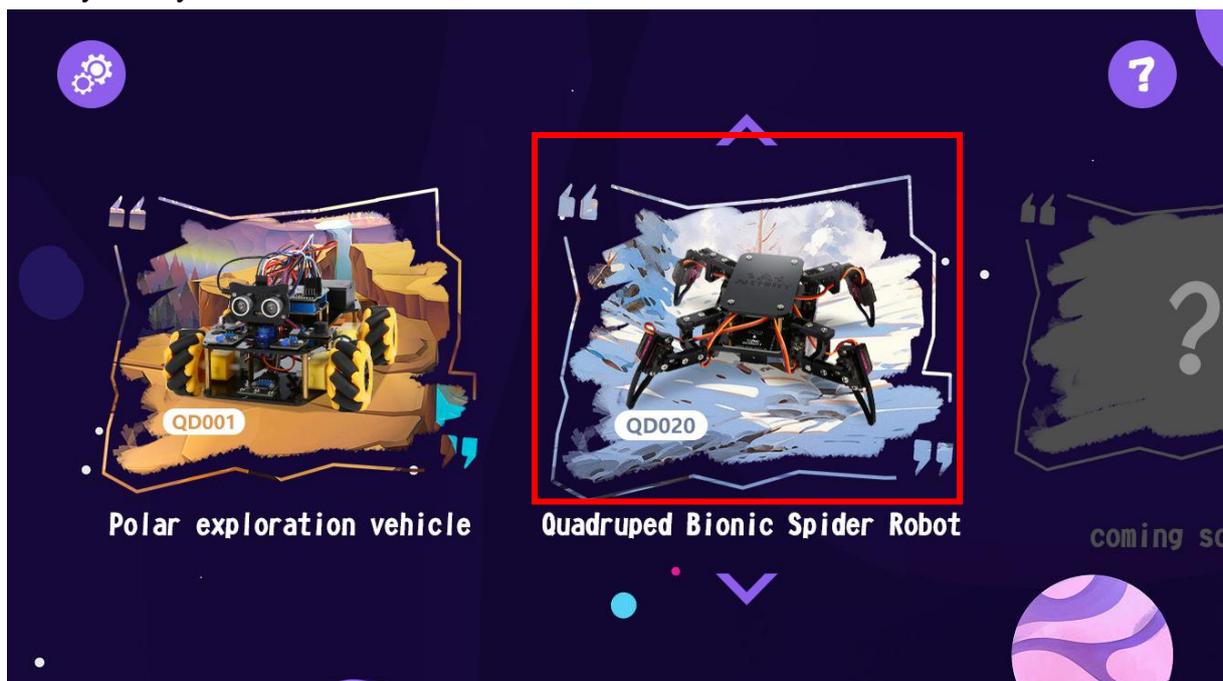
(1) Pokud používáte zařízení s iOS, vyhledejte klíčové slovo "ACEBOTT" v App Store a proveďte stahování. Pokud používáte zařízení s Androidem, vyhledejte klíčové slovo "ACEBOTT" v obchodě Google Play a proveďte stahování. Ikona aplikace je zobrazena na následujícím obrázku.



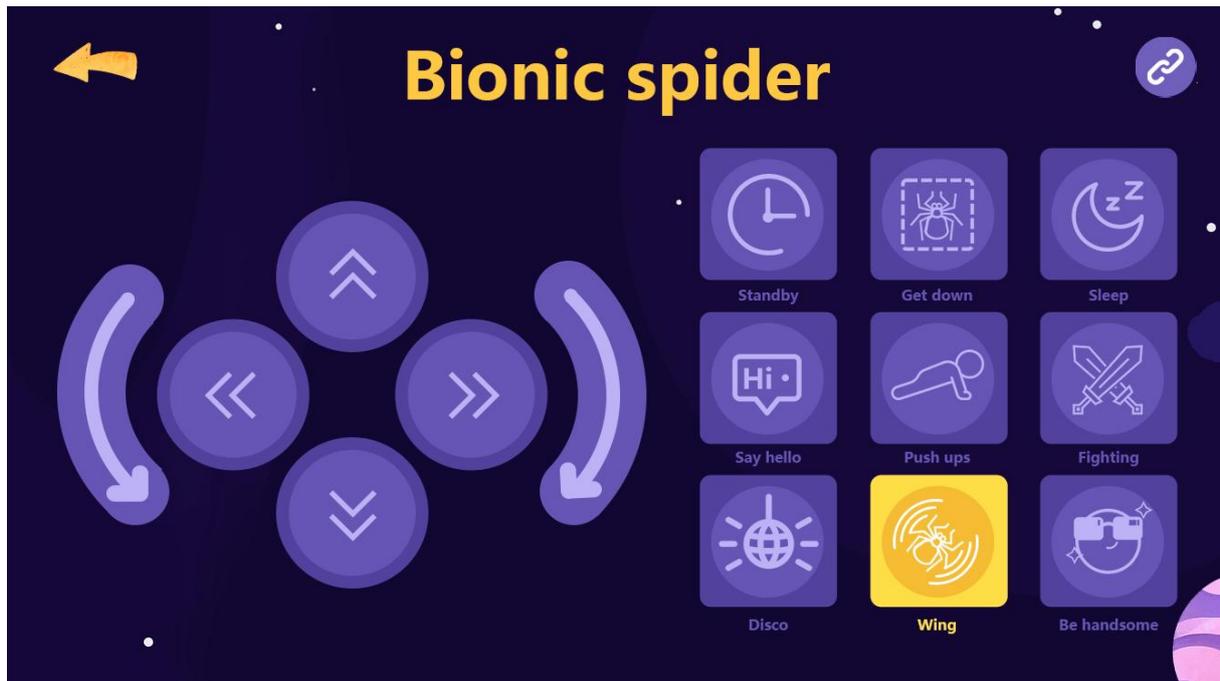
(2) Po kliknutí na aplikaci se dostanete na úvodní obrazovku.



(3) Po kliknutí na aplikaci se dostanete na výběrovou obrazovku, kde vyberete robota na čtyřech nohách.



(4) Vstupte do ovládacího rozhraní robota. (Nyní nelze přímo ovládat, je nutné nahrát program.)



II .Stahování programu

1.Program pro ovládání robota přes WiFi

Momentálně není možné ovládat robota pomocí aplikace. Nejprve je třeba nahrát kontrolní program WiFi do robota.

Otevřete soubor „8.1app_control.ino“ v adresáři Tutorial2.0\2.Arduino program\Lesson 8\8.1app_control. Připojte ESP8266 vývojovou desku k počítači pomocí USB kabelu a vyberte správnou vývojovou desku a port. Nahrajte kód do ESP8266 vývojové desky.

Před nahráním kódu přepněte přepínač rozšíření serva na desce na polohu OFF. Po nahrání kódu přepněte přepínač rozšíření serva na desce na polohu ON.

2.Připojení k WiFi

Mobilní telefon vyhledejte bezdrátové síť WiFi (vypněte sdílení sítě GPRS a další síť a zajistěte, že WiFi je jedinou používanou sítí) (konkrétní postup naleznete v menu telefonu v sekci „Nastavení “ → „WLAN “), připojte se k síti s názvem „ QuadBot-E “ a heslo je 12345678, jak je znázorněno na následujícím obrázku.

← WLAN

WLAN 
[View help](#)

Network acceleration On >

More settings >

ACEBOTT2.4 
Saved, encrypted (no Internet access)

QuadBot-E 
Saved, encrypted (no Internet access)

ChinaNet-QM4V 
Encrypted

MiFi-QIBAO 
Encrypted

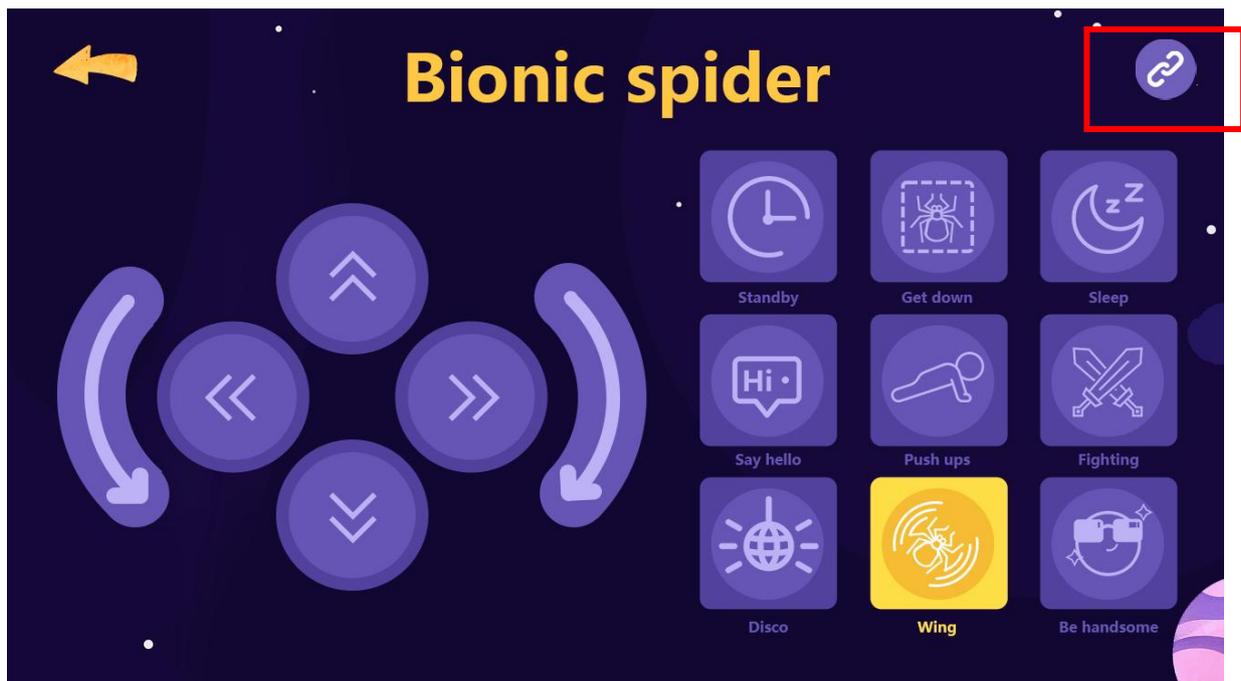
QY2021 
Encrypted

Pozor: Název a heslo hotspotu jsou již definovány v programu, ale uživatel může upravit podle svých potřeb. Pokud máme více čtyřnohých robotů, můžeme každému robotovi přiřadit odlišné názvy WiFi pro identifikaci.

```
const char* ssid = "QuadBot-E";//WiFi name  
const char* password = "12345678";//WiFi password
```

3.Použití aplikace k ovládní

Po připojení k WiFi klikněte na ikonu připojení v pravém horním rohu aplikace k dokončení spojení.



Po provedení výše uvedených kroků se vraťte zpět do rozhraní, které vypadá jako níže uvedeno, a poté můžete podle pokynů na tlačítkách ovládat robota. Ovládané akce zahrnují: stav v pohotovosti, stav ležení na zemi, stav spánku, pozdrav, pozici do klik, bojový stav, taneční stav, kývavý stav, pózování a další.

