# robotická ruka pomocí arduina – joystick

Většina industriálních i domácích systémů má nějaké vstupní zařízení – ovladač. V této kapitole se seznámíte s joystickem, který znáte z počítačových her, jako vstupním zařízením pro ovládání robotické ruky.

## CÍle

1. Pochopit princip joysticku a jeho využití jako vstupní zařízení pro konstrukce na bázi Arduina.
2. Sestrojit jednoduchou robotickou ruku za pomoci krokového motorku, servo motorku a dílů vytištěných na 3D tiskárně.
3. Tuto ruku naprogramovat a nastavit tak, aby uměla vymáchat pytlík čaje v připraveném hrníčku s vodou o požadované teplotě.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Servo motor |  | Krokový motor včetně ovladače |  | PS2 Joystick |  | LCD display |  | Potenciometr |
| Použité součástky | | | |  |  |  |  |  |

|  |  |
| --- | --- |
| Čas: 135 min  Úroveň: | Vychází z: 5, 6, 7 |

PrŮvodce hodinou – Joystick I

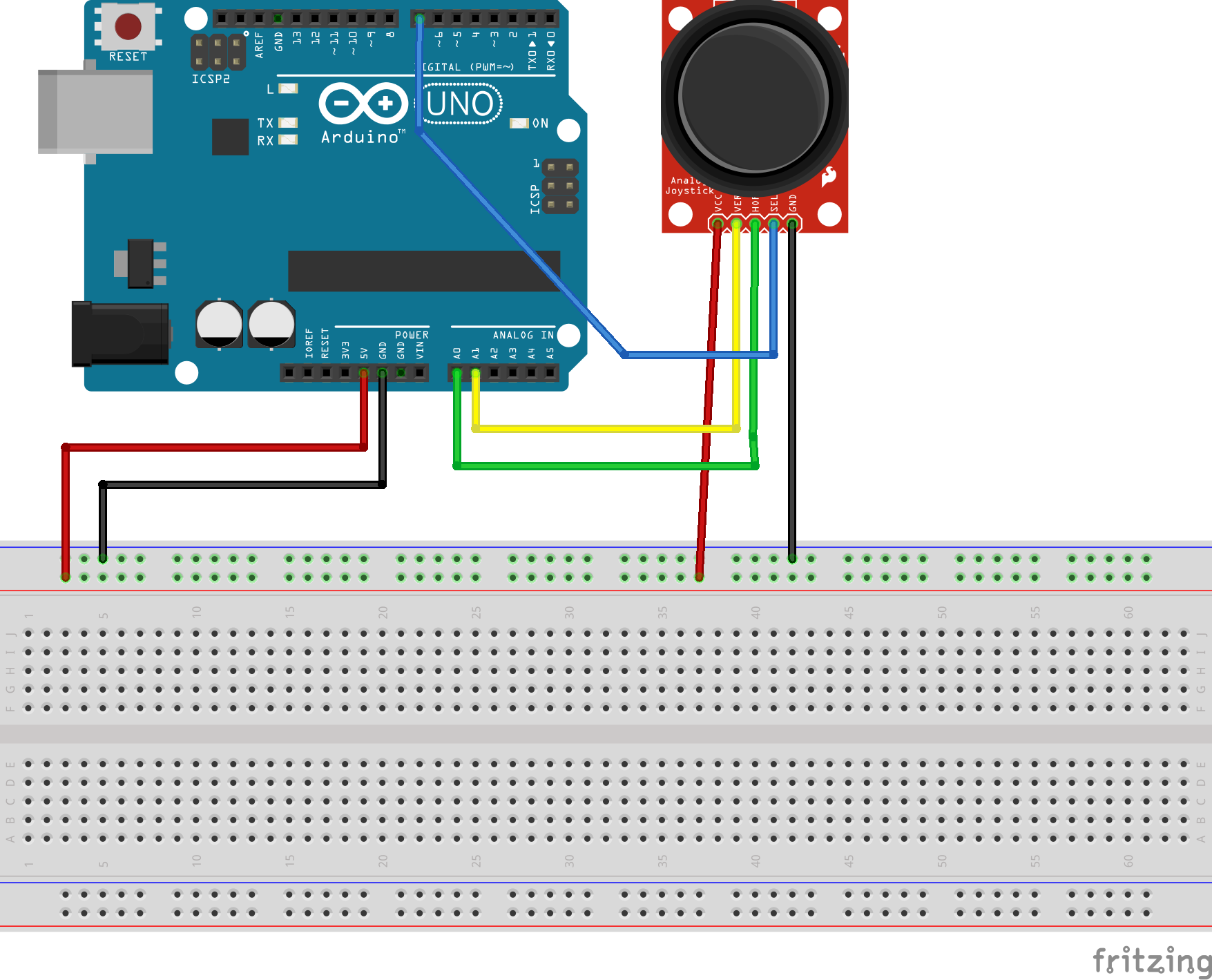
Poznámka – V této kapitole na sebe úlohy přímo navazují, a proto zejména mezi druhou a třetí hodinou, je-li to možné nerozpojujte obvod a ponechte pro další hodinu-

KROK 10 minut

Na úvod rozdejte studentům sady Arduino. Pohovořte o různých dálkového ovládání. Popište si princip joysticku a kde všude se joystick používá.

KROK 5 minut

Studenti si sestaví jednoduchý obvod s joystickem.



KROK 10 minut

Studenti přepíší (nahrají si) následující kód do programu Arduino IDE, přeloží jej a nahrají do Arduina:

int JoyStick\_X = 0; //Xová osa joysticku – analogový pin 0

int JoyStick\_Y = 1; //Yová osa joysticku – analogový pin 1

int JoyStick\_Z = 7; //Tlačítko joysticku – pin 7

int x,y,z;

void setup() {

Serial.begin(9600);

pinMode(JoyStick\_Z, INPUT\_PULLUP); //Nastavení tlačítka joysticku

Serial.println("Test joysticku");

}

void loop() {

x=analogRead(JoyStick\_X);

y=analogRead(JoyStick\_Y);

z=digitalRead(JoyStick\_Z);

Serial.print("X = ");

Serial.print(x);

Serial.print(", Y = ");

Serial.print(y);

Serial.print(", Z = ");

Serial.println(z);

delay(500);

}

KROK 10 minut

Studenti si v Arduino IDE spustí sériový monitor a otestují chování joysticku.

KROK – volitelný

Zbyde-li čas nechte studenty připojit LCD panel (zapojení v úloze 3 této kapitoly) a nechte je vypisovat hodnoty joysticku na tento panel.

Alternativně lze tuto úlohu též řešit namísto úlohy 3, zejména pokud nemáte 3D tiskárnu.

Pracovní lIST – JOYSTICK I

První seznámení s ovládáním arduina pomocí JOYSTICKU. V této části se seznámíte s jeho zapojením a funkčnostÍ.

## Co se naučíte

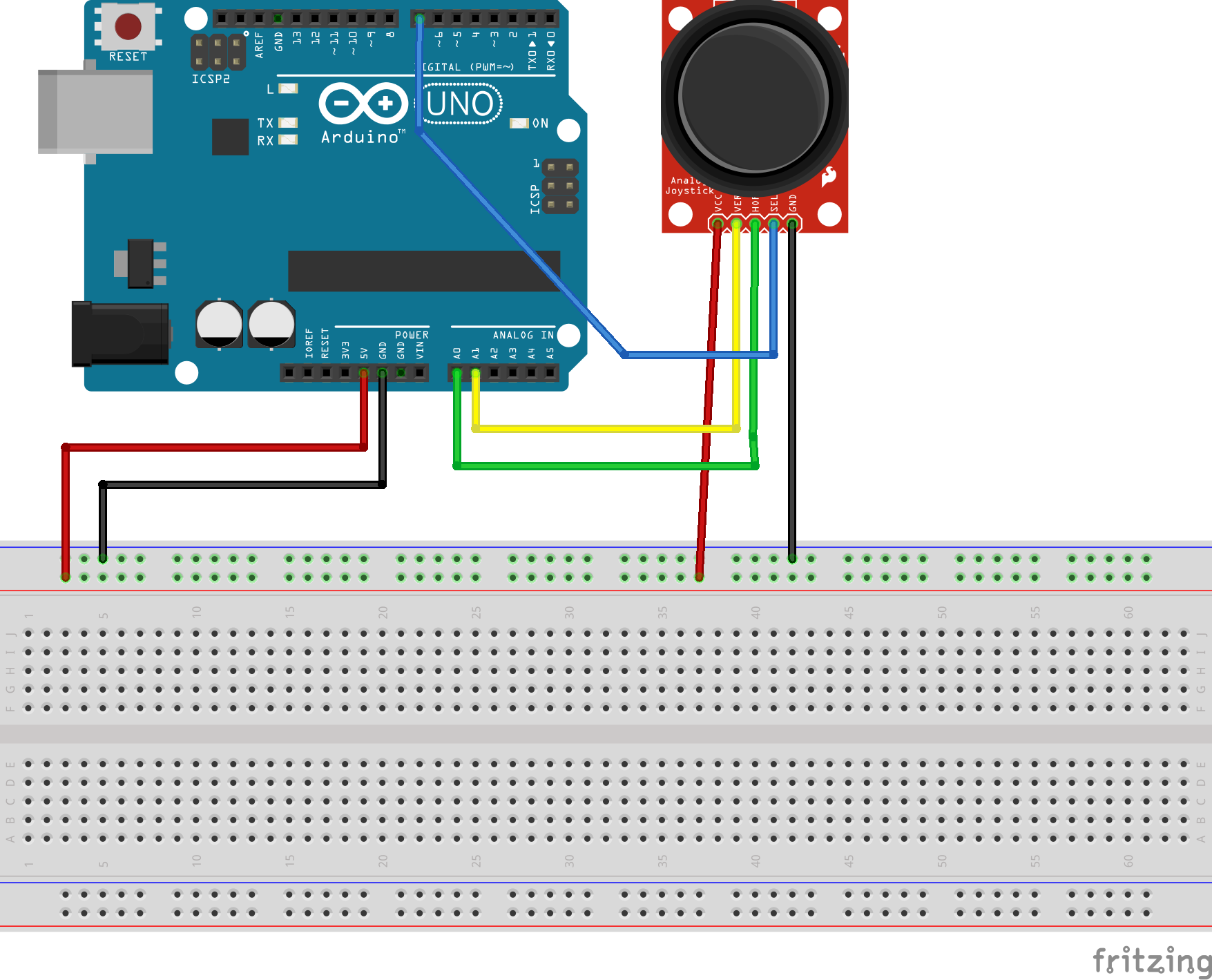
1. Princip ovládání joysticku.
2. Zapojení joysticku.
3. Naprogramování prvního programu pro ovládání pomocí joysticku.

## Co budete potřebovat

1. Joystick.
2. Desku Arduino.
3. Kontaktní pole.
4. Vodiče typu samec-samec.
5. Volitelně LCD panel a potenciometr

## A jděte na to …

Podle schématu zapojte elektronický obvod.



int JoyStick\_X = 0; //Xová osa joysticku – analogový pin 0

int JoyStick\_Y = 1; //Yová osa joysticku – analogový pin 1

int JoyStick\_Z = 7; //Tlačítko joysticku – pin 7

int x,y,z;

void setup() {

Serial.begin(9600);

pinMode(JoyStick\_Z, INPUT\_PULLUP); //Nastavení tlačítka joysticku

Serial.println("Test joysticku");

}

void loop() {

x=analogRead(JoyStick\_X);

y=analogRead(JoyStick\_Y);

z=digitalRead(JoyStick\_Z);

Serial.print("X = ");

Serial.print(x);

Serial.print(", Y = ");

Serial.print(y);

Serial.print(", Z = ");

Serial.println(z);

delay(500);

}

Spusťte program Arduino IDE a napište následující programový kód.

1. Přeložte program a nahrajte jej do Arduina.
2. Otevřete si v Arduino IDE Sériový monitor, kliknutím na ikonu
3. Testujte joystick a sledujte odezvu v sériovém monitoru

|  |  |
| --- | --- |
|  | Nefunguje joystick Zapojení v desce – zkontrolujte, zda jsou vývody zapojeny do odpovídajících pinů desky Arduino.  Zapojení v joysticku – zkontrolujte, zda jsou vývody zapojeny do odpovídajících pinů joysticku. Nejde nahrát kód do desky USB kabel – ujistěte se, že máte desku Arduino připojenou k počítači.  Správný port – ujistěte se, že máte vybraný správný port pro připojení k desce Arduino pomocí USB kabelu. Nefunguje Sériový monitor Nezobrazuje se text – Zkontrolujte zapojení USB kabelu, zkontrolujte nastavení správné rychlosti – 9660 baudů.  Zobrazuje se nesmyslný text – Zkontrolujte nastavení správné rychlosti – 9660 baudů. |

#### Úkol pro vás

* Zkuste zapojit LCD panel a zobrazovat stav joysticku na něm.
* Použijte schéma z úlohy 3 této kapitoly.



Pro tuto hodinu musíte mít díly vytištěné na 3D tiskárně. Pokud nemáte 3D tiskárnu nebo jí nechcete používat, vytvořte místo této a příští úlohy, úlohu se zapojením joysticku a LCD displeje.

PrŮvodce hodinou – JoystIck II

## 

## Příprava

Co bude v této hodině potřeba?

1. Součásti obvodu – deska Arduino s USB kabelem, kontaktní pole, vodiče typu samec-samec, joystick, Servo, obvod L9110H (ovladač motoru) a DC motor.
2. Osobní počítač pro studenty s nainstalovaným Arduino IDE.
3. Pokud je k dispozici, tak dataprojektor.
4. KROK 5 minut

Na úvod rozdejte studentům sady Arduino. Řekněte, že v této hodině naváží na předchozí příklad, který se týkal zapojení a ovládání obvodu s joystickem. Naučí se vzdáleně ovládat servo motor a krokový motor pomocí joysticku.

1. KROK 15 minut

Sestavte obvod dle schématu. Schéma jim promítněte na projektoru a naleznou jej i na pracovních listech.

1. KROK 10 minut

Studenti si zapíší kód do Arduina IDE a nahrají si jej do Arduina. Kód studentům vhodným způsobem připravte, ať jej nemusí opisovat, ale promítněte jej na projektoru a vysvětlete jej.

1. KROK 15 minut

Pokud studenti vše zvládli, mohou nyní testovat robotickou ruku pomocí joysticku.

#### Úkoly pro studenty

* A) Experimentujte s úhly, o které se otáčí servo.
* B) Experimentujte s rychlostí DC motoru.





Pracovní list – joystick II

Pokračování v seznamování se s joystickem a jeho používáním. Tentokrát budeme pomocí joysticku ovládat dva motorky – DC motor a servo.

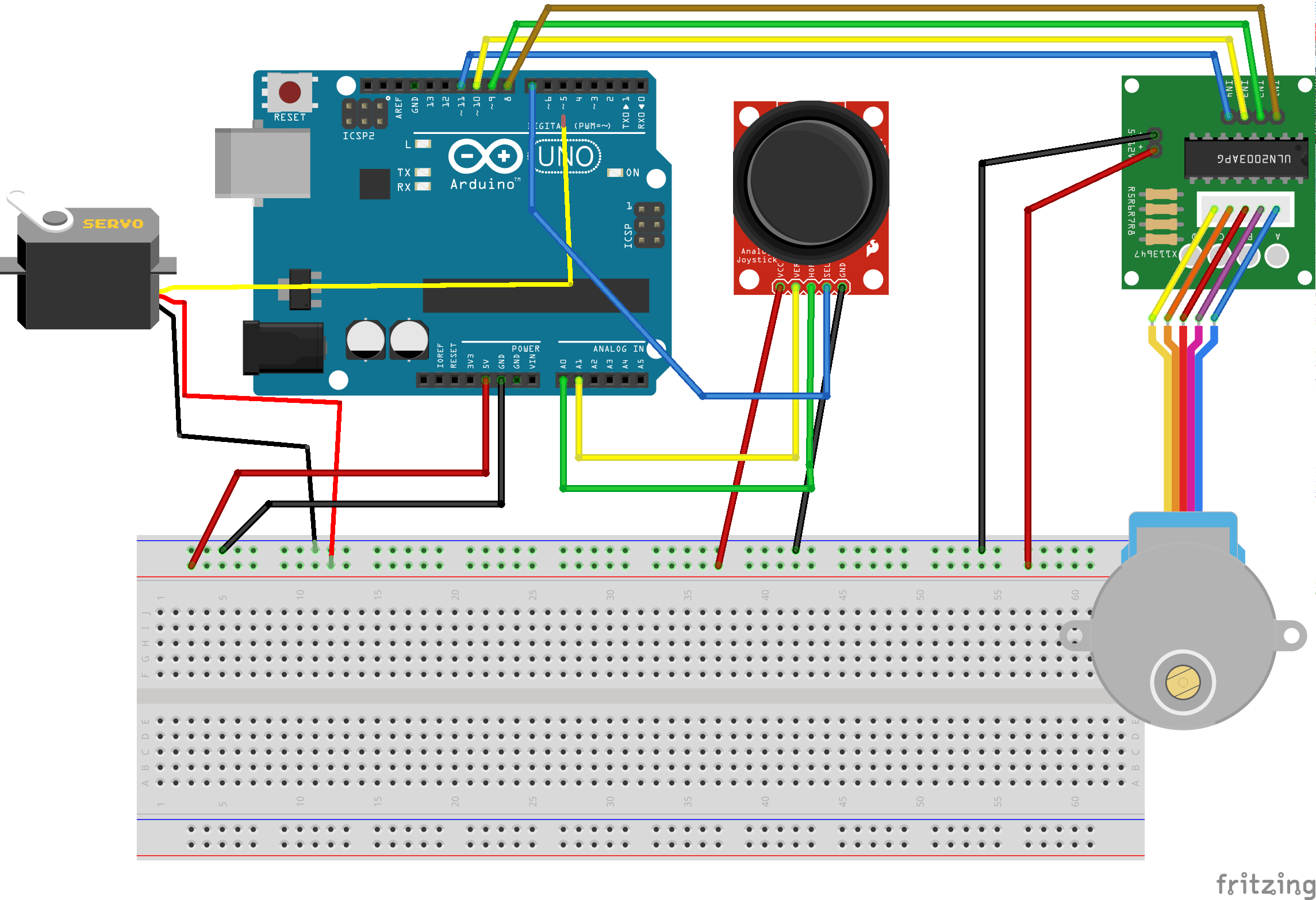
## Co se naučíte

1. Zopakujete si, zapojení joysticku a jeho použití.
2. Zopakujete si zapojení DC motoru a serva.
3. Vytvoření programu pro vzdálené ovládání DC motoru a serva pomocí joysticku.
4. Vytvořené zapojení si otestujete.

## Co budete potřebovat

1. deska Arduino s USB kabelem
2. kontaktní pole
3. vodiče typu samec-samec
4. joystick
5. Servo
6. DC motor.

## A jděte na to …

Podle schématu zapojte elektronický obvod.

Spusťte program Arduino IDE a napište následující programový kód.

#include <Servo.h>

Servo myservo;//Vytvoření objektu pro řízení krokového motoru

int poloha; //Svislá poloha ruky

// Piny pro krokový motor

const int in1 = 8;

const int in2 = 9;

const int in3 = 10;

const int in4 = 11;

// proměnná pro nastavení rychlosti,

// se zvětšujícím se číslem se rychlost zmenšuje

int rychlost = 8;

//Joystick

int JoyStick\_X = 0; //Xová osa joysticku - analogový pin 0

int JoyStick\_Y = 1; //Yová osa joysticku - analogový pin 1

int JoyStick\_Z = 7; //Tlačítko joysticku - pin 7

int x,y,z;

void setup() {

myservo.attach(6);//Servo motor je na pinu 6

myservo.write(0);//Ruka do výchozí polohy

poloha = 0; //Pamatuj si tuto polohu

// inicializace digitálních výstupů pro krokový motor

pinMode(in1, OUTPUT);

pinMode(in2, OUTPUT);

pinMode(in3, OUTPUT);

pinMode(in4, OUTPUT);

//incializace Joysticku

pinMode(JoyStick\_Z, INPUT\_PULLUP); //Nastavení tlačítka joysticku

}

void loop() {

pohyb();

while(1);

}

void rotacePoSmeru(int uhel) {

for(int i=0;i<(uhel\*64/45);i++){

krok(1,0,0,0);

krok(1,1,0,0);

krok(0,1,0,0);

krok(0,1,1,0);

krok(0,0,1,0);

krok(0,0,1,1);

krok(0,0,0,1);

krok(1,0,0,1);}

}

void rotaceProtiSmeru(int uhel) {

for(int i=0;i<(uhel\*64/45);i++){

krok(1,0,0,1);

krok(0,0,0,1);

krok(0,0,1,1);

krok(0,0,1,0);

krok(0,1,1,0);

krok(0,1,0,0);

krok(1,1,0,0);

krok(1,0,0,0);}

}

void krok(int a, int b, int c, int d){

digitalWrite(in1, a);

digitalWrite(in2, b);

digitalWrite(in3, c);

digitalWrite(in4, d);

delay(rychlost);

}

void pohyb(){

int x,y,z;

z=1;

while (z) {

x=analogRead(JoyStick\_X);

y=analogRead(JoyStick\_Y);

z=digitalRead(JoyStick\_Z);

if (x>550) { //doprava

rotacePoSmeru(5);

}

else if (x<480){ //doleva

rotaceProtiSmeru(5);

}

else if (y<480){ //dolu

if (poloha>=5) {

poloha=poloha-5;

myservo.write(poloha);

delay(1000);

}

}

else if (y>550){ //nahoru

if (poloha<=170) {

poloha=poloha+5;

myservo.write(poloha);

delay(1000) ;

}

}

}

delay(100);

}

Program odlaďte a nahrajte do Arduina.

Nyní vezměte váš joystick a vyzkoušejte program.

Pokud vše funguje, tak výborně. Můžete se vrhnout na samostatné úkoly.

Poznámka na závěr: Pokud budete řešit i následující úkol, ponechte si vše zapojené.

#### Úkol PRO vás

* Experimentujte se změnou různých parametrů. S úhly o které se otáčí servo a rychlostí DC motoru.



#### VOLITELNý úkol

* Máte-li vytištěné díly pro stavbu robotické ruky, můžete jí nyní sestrojit a zkusit rozpohybovat.



PrŮvodce hodinou – JoystIck III

## Příprava

Co bude v této hodině navíc proti minule potřeba?

1. LCD displej
2. Potenciometr.
3. Libovolný šálek či hrníček, tak do 8 cm výšky a 8 cm průměru.
4. Pokud chcete opravdu vařit čaj, je třeba rychlovarná konvice a sáčky čaje.
5. KROK 15 minut

Sestavte obvod vče tně robotické ruky.

1. KROK 15 minut

Nahrajte kód do Arduina. Vzhledem k rozsáhlosti tohoto kódu nelze doporučit jeho opisování z tabule ani z pracovních listů, proto kód studentům připravte ke stažení. Prodiskutujte s nimi změny oproti minulému případu.

1. KROK 15 minut

. Vyzkoušejte si ovládání systému nejprve s prázdným hrnkem a pokud vše půjde dobře zkuste si uvařit skutečně čaj.



Pracovní list – joystick III

Pokračování v seznamování se s joystickem a jeho používáním. Tentokrát sestrojíme robotickou ruku pro míchání pytlíku s čajem

## Co se naučíte

1. Zopakujete si, zapojení LCD panelu.
2. Vyzkoušíte si práci s jednoduchou robotickou rukou

## Co budete potřebovat

Oproti minulé hodině budeme navíc potřebovat

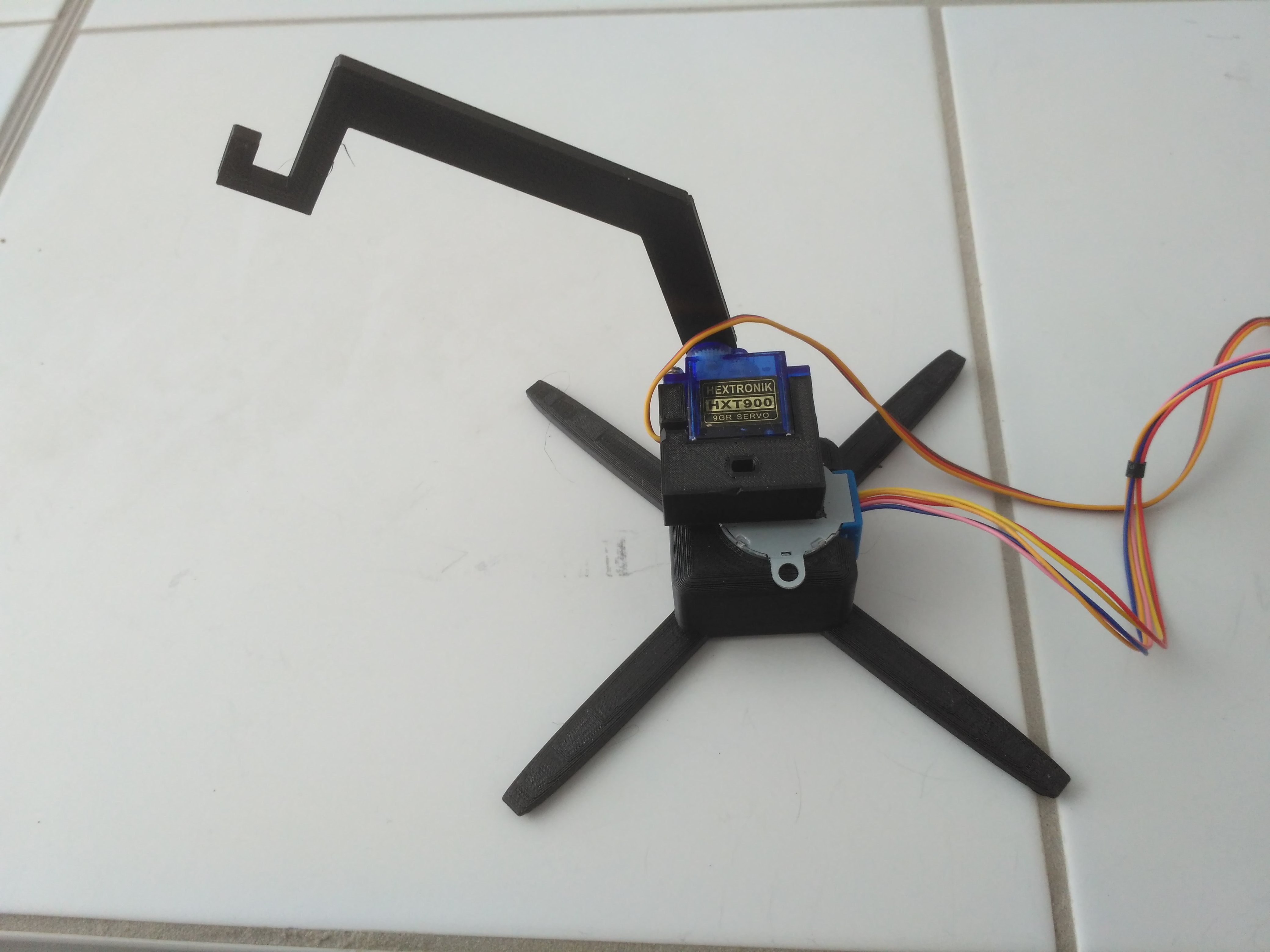
1. LCD displej
2. Potenciometr

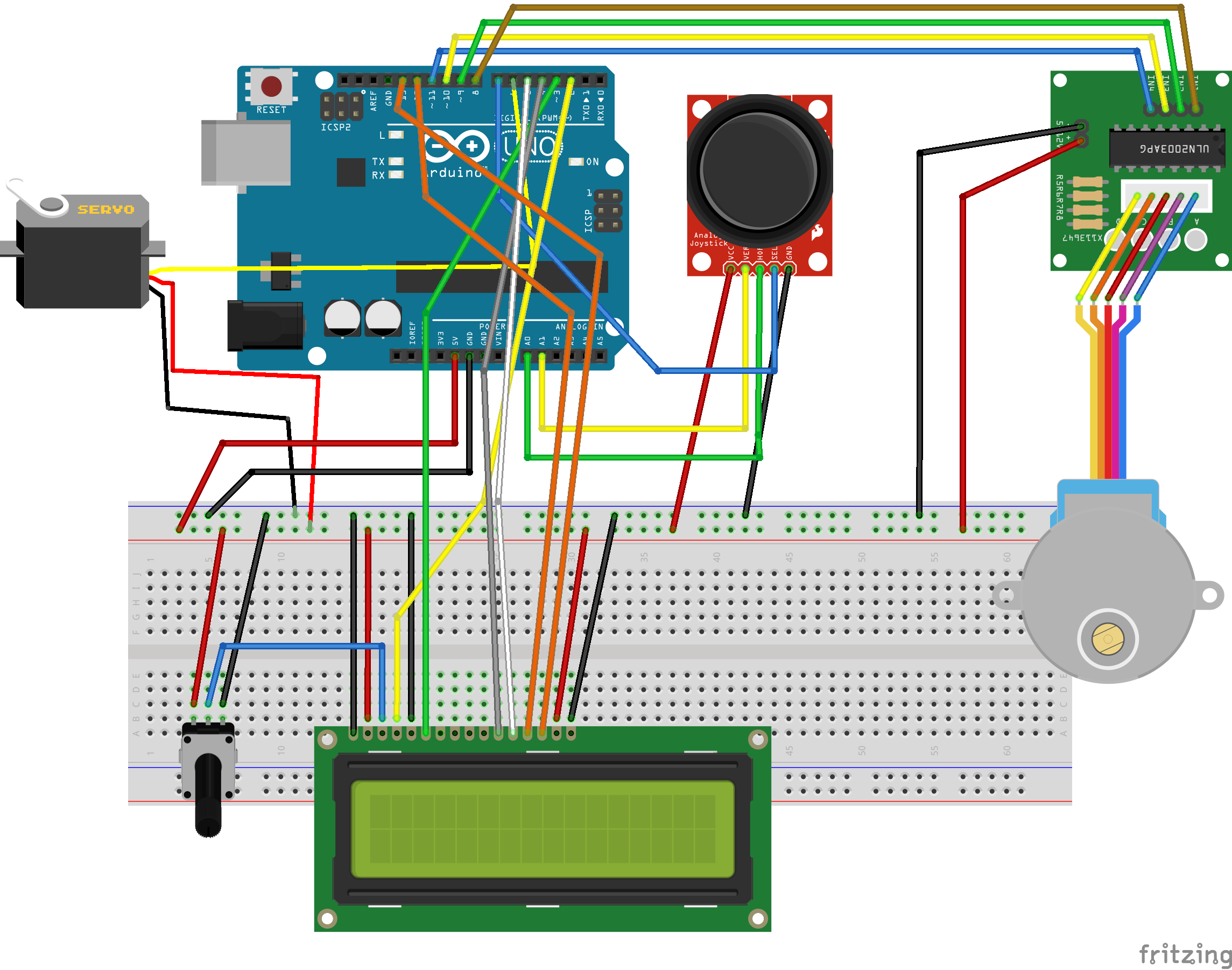
## A jděte na to …

Pokud ještě nemáte, sestavte robotickou ruku, dle fotografie.

Podle schématu zapojte elektronický obvod.

Dle následujícího obrázku sestavte ruku





Spusťte program Arduino IDE a napište následující programový kód.

#include <LiquidCrystal.h>

#include <Servo.h>

Servo myservo; //Vytvoření objektu pro řízení krokového motoru

int poloha; //Svislá poloha ruky

// Piny pro krokový motor

const int in1 = 8;

const int in2 = 9;

const int in3 = 10;

const int in4 = 11;

// Proměnná pro nastavení rychlosti,

// se zvětšujícím se číslem se rychlost zmenšuje

int rychlost = 8;

int uhel1;

int x,y,z;

int i,j;

//Joystick

int JoyStick\_X = 0; //Xová osa joysticku - analogový pin 0

int JoyStick\_Y = 1; //Yová osa joysticku - analogový pin 1

int JoyStick\_Z = 7; //Tlačítko joysticku - pin 7

// Piny pro připojení displeje

LiquidCrystal lcd(2, 3, 4, 5, 12, 13);

int minut; //Počet minut pro máchání čaje

void setup() {

myservo.attach(6); //Servo motor je na pinu 6

myservo.write(0); //Ruka do výchozí polohy

poloha = 0; //Pamatuj si tuto polohu

// inicializace digitálních výstupů pro krokový motor

pinMode(in1, OUTPUT);

pinMode(in2, OUTPUT);

pinMode(in3, OUTPUT);

pinMode(in4, OUTPUT);

//incializace Joysticku

pinMode(JoyStick\_Z, INPUT\_PULLUP); //Nastavení tlačítka joysticku

lcd.begin(16, 2); // Počet sloupců a řádek LCD displeje

}

void loop() {

//main program

lcd.clear();

lcd.print("Pouzij joystick");

lcd.setCursor(0,2);

lcd.print("pro nastaveni");

pocatecni\_nastaveni(); //Nastavení polohy ruky "nad hrnek"

lcd.clear(); //Nastavení polohy pro připevnění pytlíku

myservo.write(poloha+15);

delay(1000);

rotaceProtiSmeru(90);

delay(1000);

lcd.print("Ruka pripravena");

lcd.setCursor(0,2);

lcd.print("pripevni caj a potvrd"); //Potvrdit stiskem joysticku

delay(1000);

z=1;

while (z) {

x=analogRead(JoyStick\_X);

y=analogRead(JoyStick\_Y);

z=digitalRead(JoyStick\_Z);

delay(100);

}

delay(1000);

minut=pocetMinut(); //Nastavení počtu minut pro máchání

delay(1000);

rotacePoSmeru(90); //Najedeme nad čaj

delay(1000);

myservo.write(poloha-5); //Máchání čaje

for (i=minut;i;i--)

{

lcd.clear();

lcd.print("Zbyva:");

lcd.setCursor(0,1);

lcd.print(i);

lcd.print(" minut");

for (j=1;j<10;j++){

myservo.write(poloha-7);

delay(3000);

myservo.write(poloha+7);

delay(3000);

}

}

lcd.clear(); //Konec máchání

lcd.print("Hotovo");

myservo.write(poloha+20);

delay(10000);

rotacePoSmeru(90); //Odjezd doprava

myservo.write(15);

while(1) { } //Nekonečná smyčka

}

// zde následují funkce pro volání jednotlivých

// kroků pro otočení po či proti směru hodinových

// ručiček

void rotacePoSmeru(int uhel) {

for(int i=0;i<(uhel\*64/45);i++){

krok(1,0,0,0);

krok(1,1,0,0);

krok(0,1,0,0);

krok(0,1,1,0);

krok(0,0,1,0);

krok(0,0,1,1);

krok(0,0,0,1);

krok(1,0,0,1);}

}

void rotaceProtiSmeru(int uhel) {

for(int i=0;i<(uhel\*64/45);i++){

krok(1,0,0,1);

krok(0,0,0,1);

krok(0,0,1,1);

krok(0,0,1,0);

krok(0,1,1,0);

krok(0,1,0,0);

krok(1,1,0,0);

krok(1,0,0,0);}

}

// každý krok obsahuje výrobcem dané pořadí

// pro správné spínání motoru a následnou

// pauzu, kterou určujeme rychlost otáčení

void krok(int a, int b, int c, int d){

digitalWrite(in1, a);

digitalWrite(in2, b);

digitalWrite(in3, c);

digitalWrite(in4, d);

delay(rychlost);

}

void pocatecni\_nastaveni(){

int x,y,z;

z=1;

while (z) {

x=analogRead(JoyStick\_X);

y=analogRead(JoyStick\_Y);

z=digitalRead(JoyStick\_Z);

if (x>550) { //doprava

rotacePoSmeru(5);

}

else if (x<480){ //doleva

rotaceProtiSmeru(5);

}

else if (y<480){ //dolu

if (poloha>=5) {

poloha=poloha-5;

myservo.write(poloha);

delay(1000);

}

}

else if (y>550){ //nahoru

if (poloha<=170) {

poloha=poloha+5;

myservo.write(poloha);

delay(1000);

}

}

}

delay(100);

}

int pocetMinut(){

int m=2;

int x, y, z;

z=1;

lcd.setCursor(0,0);

lcd.print("Maximum 9 minut");

lcd.setCursor(0,1);

lcd.print("Louhovat: ");

lcd.print(m);

lcd.print(" min");

while (z) {

x=analogRead(JoyStick\_X);

y=analogRead(JoyStick\_Y);

z=digitalRead(JoyStick\_Z);

if (y>550){ //dolu

if (m) {

m=m-1;}

lcd.setCursor(10,1);

lcd.print(m);

}

if (y<480){ //nahoru

if (m<9) {

m=m+1;}

lcd.setCursor(10,1);

lcd.print(m);

}

delay(300);

}

return m;

}

Program odlaďte a nahrajte do Arduina.

Nyní zkuste nasimulovat vymáchání pytlíku v čaji a až si budete jistí, můžete si opravdu zkusi uvařit čaj. Postupujte dle následujícího návodu.

### Práce s robotickou rukou

1. Po spuštění programu je třeba jako první věc nastavit robotickou ruku nad hrnek. Doporučuji nastavovat na prázdný hrnek. Ruku nastavte trochu napravo od středu hrnku a částečně jí ponořte pod horní okraj (cca. 1 cm). Potvrďte stiskem joysticku.
2. Ruka si nyní najede vlevo od hrnku na pozici, na které můžete připevnit pytlík s čajem. Snažte se při tom nepohybovat s rukou. Pokud se vám to nepodaří, připevněte pytlík, stiskněte reset Arduinu a vraťte se na krok jedna. Alternativně začínejte rovnou s připevněným pytlíkem.
3. Připravte si talířek nebo nějakou podložku, nad kterou ruka donese vylouhovaný pytlík. Umístěte jí na pozici asi 90° napravo od hrnku.
4. Nastavte počet minut, po které se má čaj louhovat, nalejte do hrnku vodu s požadovanou teplotou a potvrďte joystickem.
5. Robotická ruka provede vymáchání čaje a po nastavené době odnese pytlík napravo od hrnku nad připravenou podložku.
6. Tímto celý cyklus končí, pro jeho opakování je nutné stisknout reset na Arduinu.

Podrobný průvodce teorií

## Robotická ruka

Též zvaná manipulátor, patří mezi takzvané sériové roboty neboli řetězce. To jsou druhy robotů, charakteristické pravidelným střídáním pevných částí (zvaných linky nebo ramena) s pohyblivými částmi (zvanými klouby). Tyto klouby mohou být sférické, cylindrické nebo posuvné. V našem případě použijeme dva cylindrické (rotační) klouby sestavené za pomoci krokového motorku a serva známých z předchozích příkladů. Schéma viz následující obrázek.

Rameno 2

Servo

Rameno 1

Krokový motor

Díly pro tuto robotickou ruku jsou vytištěné na 3D tiskárně a je možné si je stáhnout z WWW stránek. Jsou velmi jednoduché a jejich sestavení je patrné z přiložených obrázků. Předpokládám, že 3D tiskárnu již má většina škol k dispozici. V případě nouze, lze ramena sestavit např. z několika spojených kusů lepenkového papíru. S takovouto rukou však řešte pouze první dvě úlohy. Pokud chcete řešit i závěrečnou úlohu této kapitoly – máchání čaje pomocí robotické ruky, pak je zakončení ruky nutné udělat z nějakého vodě (čaji) odolného materiálu anebo ruku testujte pouze „nasucho“ – bez vody.

Počet kloubů u robotické ruky u ní určuje takzvaný stupeň volnosti. Tato ruka má tedy dva stupně volnosti. Pokud byste někdy chtěli sestrojit robotickou ruku, tak aby dokázala to, co umí ruka lidská potřebujete 6 stupňů volnosti. Průmyslově vyráběné a používané robotické ruce – manipulátory mají obvykle 5 až 7 stupňů volnosti.

## Joystick

Zvaný též pákový nebo křížový ovladač. Je vstupní zařízení, které se sestává z páky připevněné na základně, která se dokáže pohybovat ve směru x a y, přičemž je snímán úhel a velikost vychýlení. Obvykle je rovněž vybaven jedním nebo více tlačítky, u kterých snímáno stlačení.

Joystick se používá pro hraní videoher, ale rovněž pro ovládání letadel, automobilů, robotických rukou atd.

Joystick obsažený ve vaší stavebnici má jedno tlačítko na vrcholu „kloboučku“. Pro připojení potřebuje pět vodičů. Dva jsou napájení. Jeden slouží jako indikace stlačení tlačítka a připojuje se na digitální vstup. Je nastaven tak, že implicitně proud prochází a při sepnutí nikoliv. Další dva vodiče se připojují na analogové vstupy a poskytují informaci o pohybu joysticku ve směru os x a y jako číslo v intervalu 0 až 1023. 512 znamená, že joystick je uprostřed a obě hraniční čísla pak znamenají maximální vychýlení. V následujícím příkladu se s joystickem blíže seznámíte.

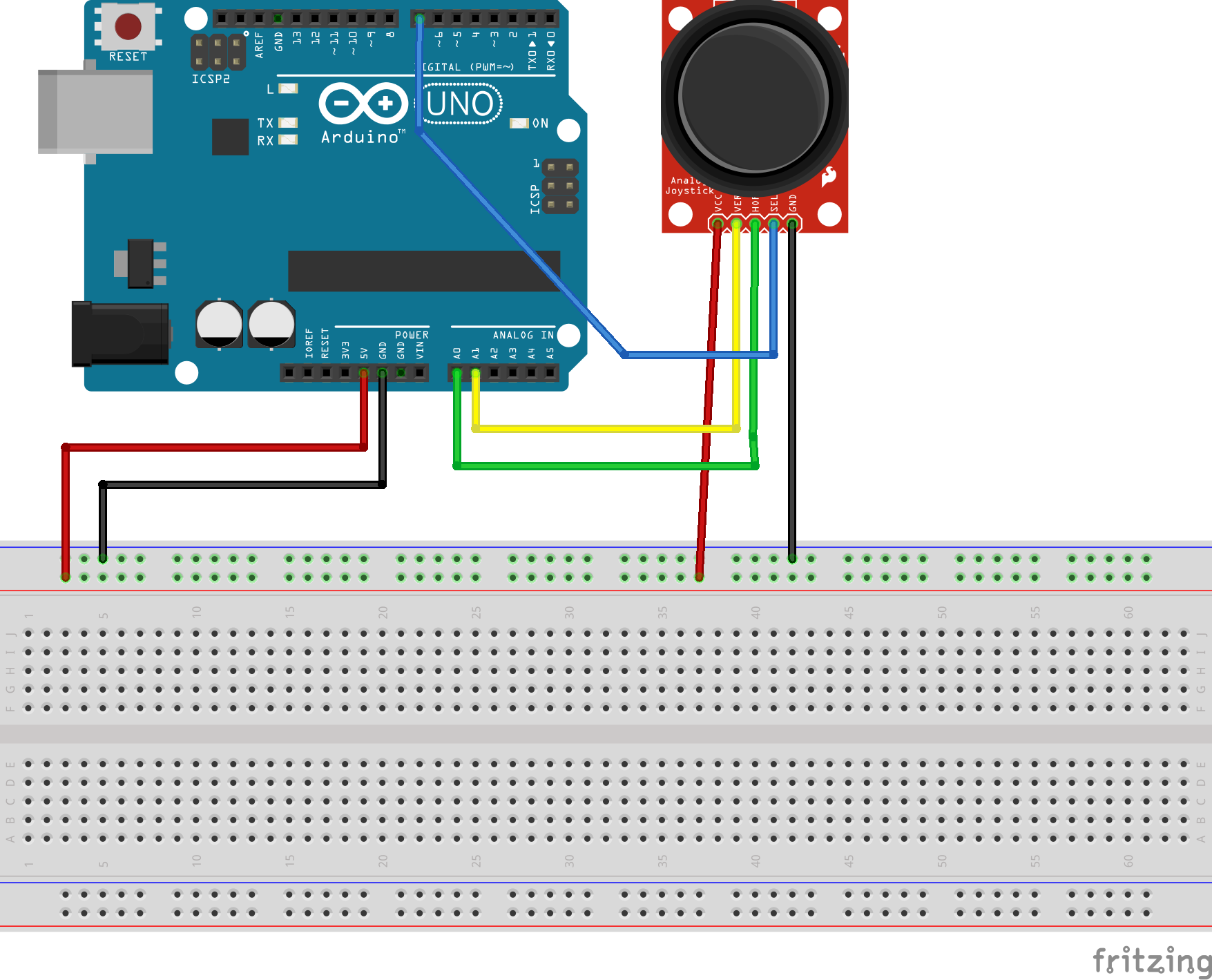
## Poznámka

Následující úkoly na sebe navazují, a proto po vyřešení prvního úkolu není nutné obvod rozpojovat, ale naopak postupně k němu budete přidávat další součástky. Stejně tak i program je koncipován od jednoduššího ke složitějšímu.

## Úkol 1

Zapojte joystick a monitorujte jeho stavy pomocí sériové komunikace.

### Zapojení obvodu

Sestavte obvod dle následujícího obrázku:

### programový kód

c

b

a

|  |  |  |
| --- | --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24 | int JoyStick\_X = 0; //Xová osa joysticku – analogový pin 0  int JoyStick\_Y = 1; //Yová osa joysticku – analogový pin 1  int JoyStick\_Z = 7; //Tlačítko joysticku – pin 7  int x,y,z;  void setup() {  Serial.begin(9600);  pinMode(JoyStick\_Z, INPUT\_PULLUP); //Nastavení tlačítka joysticku  Serial.println("Test joysticku");  }  void loop() {  x=analogRead(JoyStick\_X);  y=analogRead(JoyStick\_Y);  z=digitalRead(JoyStick\_Z);  Serial.print("X = ");  Serial.print(x);  Serial.print(", Y = ");  Serial.print(y);  Serial.print(", Z = ");  Serial.println(z);  delay(500);  } |  |

1. Nastavení připojení joysticku. Vodiče pro pohyb ve směru X a Y jsou připojeny na analogové piny na portech A0 a A1. Vodič pro osu Z (tlačítko joysticku) na pin 7 (digitální). Proměnné x, y a z jsou jejich instance v programu.
2. Úvodní nastavení. Nastavení tlačítka joysticku a příprava sériového portu.
3. Hlavní program s cyklickým opakováním. Načtení hodnot z joysticku a jejich výpis na sériový port. Aby se výpis neopakoval příliš často, čeká se před následujícím načtením půl vteřiny

Program přeložte a nahrajte do Arduina. V programu Arduino si otevřete Sériový monitor. Pohybujte joystickem a sledujte, jak se při tom mění výpis.

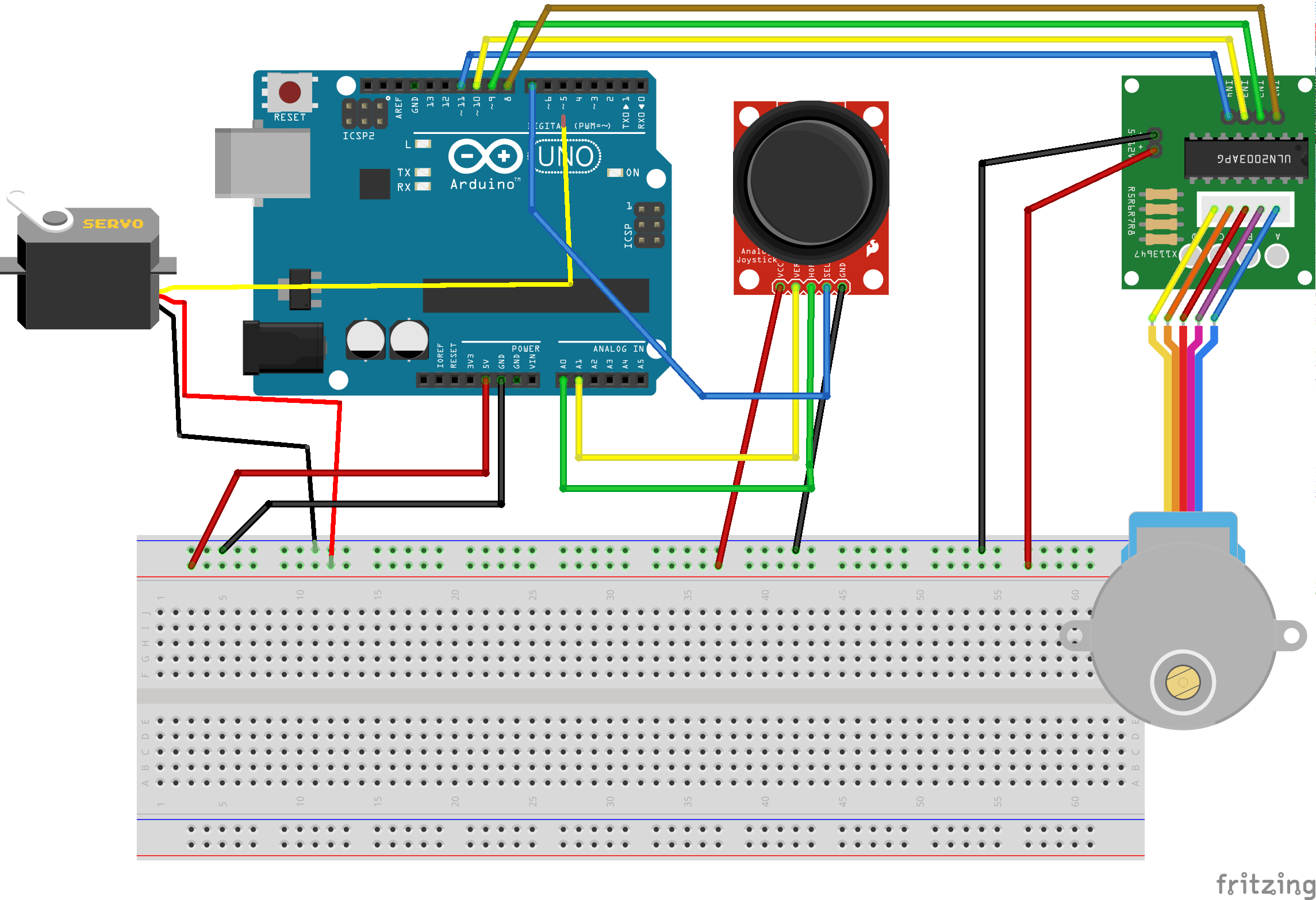
|  |  |
| --- | --- |
|  | Nefunguje joystick Zapojení v desce – zkontrolujte, zda jsou vývody zapojeny do odpovídajících pinů desky Arduino.  Zapojení v joysticku – zkontrolujte, zda jsou vývody zapojeny do odpovídajících pinů joysticku. Nejde nahrát kód do desky USB kabel – ujistěte se, že máte desku Arduino připojenou k počítači.  Správný port – ujistěte se, že máte vybraný správný port pro připojení k desce Arduino pomocí USB kabelu. Nefunguje Sériový monitor Nezobrazuje se text – Zkontrolujte zapojení USB kabelu, zkontrolujte nastavení správné rychlosti – 9660 baudů.  Zobrazuje se nesmyslný text – Zkontrolujte nastavení správné rychlosti – 9660 baudů. |

|  |  |
| --- | --- |
|  | Zkuste úlohu řešit tak, že si místo sériového monitoru zapojíte LCD displej a pohyby joysticku si budete zobrazovat na něm. Můžete využít schéma zapojení a zdrojový kód z Úlohy 3 v této kapitole. |

## Úkol 2

Sestrojte robotickou ruku ovládanou pomocí joysticku. Součástky na výrobu ruky vytiskněte na 3D tiskárně.

### Zapojení obvodu

Sestavte ze součástek robotickou ruku dle fotografie. Obvod zapojte dle následujícího schématu. K joysticku přibydou dva motorky. Krokový motor se zapojuje pomocí ovladače.

### programový kód

e

d

c

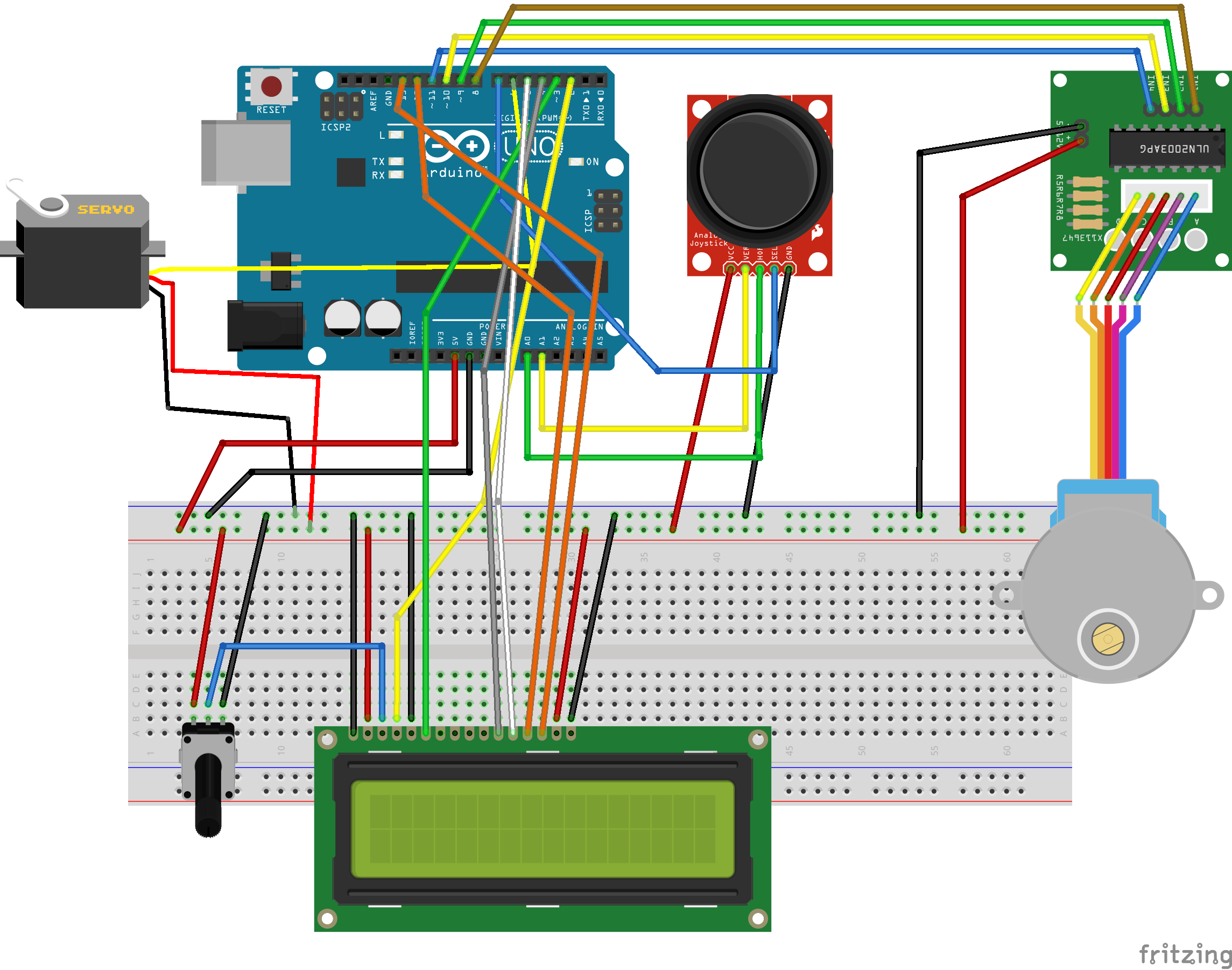
|  |  |  |
| --- | --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107 | #include <Servo.h>  Servo myservo;//Vytvoření objektu pro řízení krokového motoru  int poloha; //Svislá poloha ruky  // Piny pro krokový motor  const int in1 = 8;  const int in2 = 9;  const int in3 = 10;  const int in4 = 11;  b  // proměnná pro nastavení rychlosti,  // se zvětšujícím se číslem se rychlost zmenšuje  int rychlost = 8;  //Joystick  int JoyStick\_X = 0; //Xová osa joysticku - analogový pin 0  int JoyStick\_Y = 1; //Yová osa joysticku - analogový pin 1  int JoyStick\_Z = 7; //Tlačítko joysticku - pin 7  int x,y,z;  void setup() {  myservo.attach(6);//Servo motor je na pinu 6  myservo.write(0);//Ruka do výchozí polohy  poloha = 0; //Pamatuj si tuto polohu  // inicializace digitálních výstupů pro krokový motor  pinMode(in1, OUTPUT);  pinMode(in2, OUTPUT);  pinMode(in3, OUTPUT);  pinMode(in4, OUTPUT);  //incializace Joysticku  pinMode(JoyStick\_Z, INPUT\_PULLUP); //Nastavení tlačítka joysticku  }  void loop() {  pohyb();  while(1);  }  void rotacePoSmeru(int uhel) {  for(int i=0;i<(uhel\*64/45);i++){  krok(1,0,0,0);  krok(1,1,0,0);  krok(0,1,0,0);  krok(0,1,1,0);  krok(0,0,1,0);  krok(0,0,1,1);  krok(0,0,0,1);  krok(1,0,0,1);}  }  void rotaceProtiSmeru(int uhel) {  for(int i=0;i<(uhel\*64/45);i++){  krok(1,0,0,1);  f  f  krok(0,0,0,1);  krok(0,0,1,1);  krok(0,0,1,0);  krok(0,1,1,0);  krok(0,1,0,0);  krok(1,1,0,0);  krok(1,0,0,0);}  }  void krok(int a, int b, int c, int d){  digitalWrite(in1, a);  digitalWrite(in2, b);  digitalWrite(in3, c);  digitalWrite(in4, d);  delay(rychlost);  }  void pohyb(){  int x,y,z;  z=1;  while (z) {  x=analogRead(JoyStick\_X);  y=analogRead(JoyStick\_Y);  z=digitalRead(JoyStick\_Z);  if (x>550) { //doprava  rotacePoSmeru(5);  g  }  else if (x<480){ //doleva  rotaceProtiSmeru(5);  }  else if (y<480){ //dolu  if (poloha>=5) {  poloha=poloha-5;  myservo.write(poloha);  delay(1000);  }  }  else if (y>550){ //nahoru  if (poloha<=170) {  g  poloha=poloha+5;  myservo.write(poloha);  delay(1000) ;  }  }  }  delay(100);  } | a |

1. Nastavení servo motoru (svislý pohyb). Proměnná poloha slouží k zapamatování aktuální svislé polohy ruky.
2. Nastavení krokového motoru (vodorovný pohyb). Proměnná rychlost slouží pro nastavení pauzy mezi jednotlivými pohyby motoru.
3. Nastavení pinů joysticku (viz předchozí příklad).
4. Inicializace potřebných proměnných a knihoven.
5. Hlavní program. Nedělá nic jiného, než spustí proceduru pohyb, která se stará o pohyb ruky, dle pohybu joysticku. Konstrukce while(1) je nekonečná smyčka ve které skončí program po stisku tlačítka joysticku. Pro opětovné spuštění je nutné stisknout tlačítko reset na Arduinu.
6. Funkce řídící pohyb krokového motorku. Tyto funkce jsou opsané z manuálu výrobce motorku (dle manuálu k stavebnici, kterou máte k dispozici). Máte-li jiný motorek, než ten ze stavebnice je možné (ale nepravděpodobné), že bude tyto funkce nutné přepsat dle manuálu vašeho motorku.
7. Hlavní funkce. Načítá pohyb joysticku do proměnných x a y a hlídá stisk tlačítka (proměnná z). Dle těchto hodnot pak řídí pohyb ruky pomocí obou motorků. Funkce končí stiskem tlačítka (v z je po stisku 0).

## Úkol 3

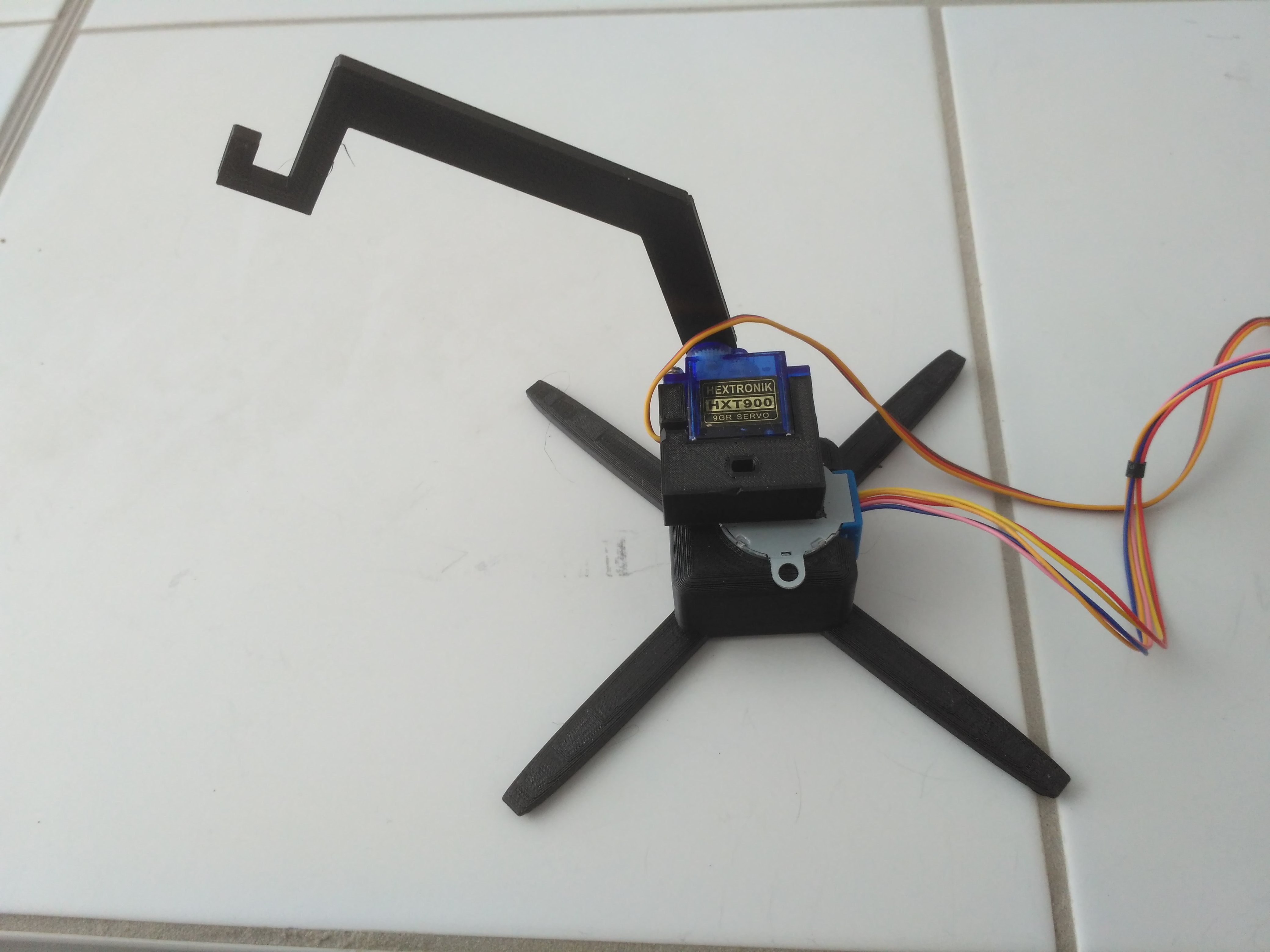
Sestrojte robotickou ruku pro máchání čaje, ovládanou pomocí joysticku. Tato ruka by měla umět připevněný čajový pytlík máchat předem danou dobu v hrnečku a pak jej přesunout mimo hrneček.

### Zapojení obvodu

Upravte zapojení z předchozího příkladu. Doplňte LCD panel, na který se budou zobrazovat informace a stavy ruky. Zkontrolujte opravdu pečlivě jeho zapojení vzhledem k velkému počtu vodičů.

### Fotografie ruky

Včetně dílů z 3D tiskárny



### programový kód

Následující kód vychází z předchozího příkladu a popsány proto budou pouze odlišnosti.

a

a

|  |  |  |
| --- | --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127  128  129  130  131  132  133  134  135  136  137  138  139  140  141  142  143 144  145  146  147  148  149  150  151  152  153  154  155  156  157  158  159  160  161  162  163  164  165  166  167  168  169  170  171  172  173  174  175  176  177  178  179  180  181  182  183  184  185  186  187  188  189  190  191 192  193  194  195  196  197  198  199  200  201 | #include <LiquidCrystal.h>  #include <Servo.h>  Servo myservo; //Vytvoření objektu pro řízení krokového motoru  int poloha; //Svislá poloha ruky  // Piny pro krokový motor  const int in1 = 8;  const int in2 = 9;  const int in3 = 10;  const int in4 = 11;  // Proměnná pro nastavení rychlosti,  // se zvětšujícím se číslem se rychlost zmenšuje  int rychlost = 8;  int uhel1;  int x,y,z;  int i,j;  //Joystick  int JoyStick\_X = 0; //Xová osa joysticku - analogový pin 0  int JoyStick\_Y = 1; //Yová osa joysticku - analogový pin 1  int JoyStick\_Z = 7; //Tlačítko joysticku - pin 7  // Piny pro připojení displeje  LiquidCrystal lcd(2, 3, 4, 5, 12, 13);  int minut; //Počet minut pro máchání čaje  void setup() {  myservo.attach(6); //Servo motor je na pinu 6  myservo.write(0); //Ruka do výchozí polohy  poloha = 0; //Pamatuj si tuto polohu  // inicializace digitálních výstupů pro krokový motor  pinMode(in1, OUTPUT);  pinMode(in2, OUTPUT);  pinMode(in3, OUTPUT);  pinMode(in4, OUTPUT);  //incializace Joysticku  pinMode(JoyStick\_Z, INPUT\_PULLUP); //Nastavení tlačítka joysticku  a  lcd.begin(16, 2); // Počet sloupců a řádek LCD displeje    }  void loop() {  //main program  lcd.clear();  lcd.print("Pouzij joystick");  lcd.setCursor(0,2);  lcd.print("pro nastaveni");  pocatecni\_nastaveni(); //Nastavení polohy ruky "nad hrnek"  lcd.clear(); //Nastavení polohy pro připevnění pytlíku  myservo.write(poloha+15);  delay(1000);  rotaceProtiSmeru(90);  delay(1000);  lcd.print("Ruka pripravena");  lcd.setCursor(0,2);  lcd.print("pripevni caj a potvrd"); //Potvrdit stiskem joysticku  delay(1000);  b  z=1;  while (z) {  x=analogRead(JoyStick\_X);  y=analogRead(JoyStick\_Y);  z=digitalRead(JoyStick\_Z);  delay(100);  }  delay(1000);  minut=pocetMinut(); //Nastavení počtu minut pro máchání  delay(1000);  rotacePoSmeru(90); //Najedeme nad čaj  delay(1000);  myservo.write(poloha-5); //Máchání čaje  for (i=minut;i;i--)  {  lcd.clear();  lcd.print("Zbyva:");  lcd.setCursor(0,1);  lcd.print(i);  lcd.print(" minut");  for (j=1;j<10;j++){  myservo.write(poloha-7);  delay(3000);  myservo.write(poloha+7);  delay(3000);  }  b  }  lcd.clear(); //Konec máchání  lcd.print("Hotovo");  myservo.write(poloha+20);  delay(10000);  rotacePoSmeru(90); //Odjezd doprava  myservo.write(15);  while(1) { } //Nekonečná smyčka  }  // zde následují funkce pro volání jednotlivých  // kroků pro otočení po či proti směru hodinových  // ručiček  void rotacePoSmeru(int uhel) {  for(int i=0;i<(uhel\*64/45);i++){  krok(1,0,0,0);  krok(1,1,0,0);  krok(0,1,0,0);  krok(0,1,1,0);  krok(0,0,1,0);  krok(0,0,1,1);  krok(0,0,0,1);  krok(1,0,0,1);}  }  void rotaceProtiSmeru(int uhel) {  for(int i=0;i<(uhel\*64/45);i++){  krok(1,0,0,1);  krok(0,0,0,1);  krok(0,0,1,1);  krok(0,0,1,0);  krok(0,1,1,0);  krok(0,1,0,0);  krok(1,1,0,0);  krok(1,0,0,0);}  }  // každý krok obsahuje výrobcem dané pořadí  // pro správné spínání motoru a následnou  // pauzu, kterou určujeme rychlost otáčení  void krok(int a, int b, int c, int d){  digitalWrite(in1, a);  digitalWrite(in2, b);  digitalWrite(in3, c);  digitalWrite(in4, d);  delay(rychlost);  }  void pocatecni\_nastaveni(){  int x,y,z;  z=1;  while (z) {  x=analogRead(JoyStick\_X);  y=analogRead(JoyStick\_Y);  z=digitalRead(JoyStick\_Z);  if (x>550) { //doprava  rotacePoSmeru(5);  }  else if (x<480){ //doleva  rotaceProtiSmeru(5);  }  else if (y<480){ //dolu  if (poloha>=5) {  poloha=poloha-5;  myservo.write(poloha);  delay(1000);  }  }  else if (y>550){ //nahoru  if (poloha<=170) {  poloha=poloha+5;  myservo.write(poloha);  delay(1000);  }  }  }  delay(100);  }  int pocetMinut(){  int m=2;  int x, y, z;  z=1;  c  lcd.setCursor(0,0);  lcd.print("Maximum 9 minut");  lcd.setCursor(0,1);  lcd.print("Louhovat: ");  lcd.print(m);  lcd.print(" min");  while (z) {  x=analogRead(JoyStick\_X);  y=analogRead(JoyStick\_Y);  z=digitalRead(JoyStick\_Z);  if (y>550){ //dolu  if (m) {  m=m-1;}  lcd.setCursor(10,1);  lcd.print(m);  }  if (y<480){ //nahoru  if (m<9) {  m=m+1;}  lcd.setCursor(10,1);  lcd.print(m);    }  delay(300);    }  return m;  } |  |

1. Nastavení LCD panelu včetně pinů, na které je zapojen.
2. Vlastní program, tentokrát složitější. Postupně se prochází následující kroky:
   1. Počáteční nastavení polohy ruky – nad hrnek.
   2. Přesun ruky na pozici, kde se připevní pytlík čaje.
   3. Nastavení doby pro máchání pytlíku.
   4. Vlastní máchání pytlíku.
   5. Posun ruky na místo, kde je možné pytlík sejmout.
   6. Nekonečná smyčka na závěr.
3. Funkce pro nastavení počtu minut pro máchání čaje.

### Práce s robotickou rukou

1. Po spuštění programu je třeba jako první věc nastavit robotickou ruku nad hrnek. Doporučuji nastavovat na prázdný hrnek. Ruku nastavte trochu napravo od středu hrnku a částečně jí ponořte pod horní okraj (cca. 1 cm). Potvrďte stiskem joysticku.
2. Ruka si nyní najede vlevo od hrnku na pozici, na které můžete připevnit pytlík s čajem. Snažte se při tom nepohybovat s rukou. Pokud se vám to nepodaří, připevněte pytlík, stiskněte reset Arduinu a vraťte se na krok jedna. Alternativně začínejte rovnou s připevněným pytlíkem.
3. Připravte si talířek nebo nějakou podložku, nad kterou ruka donese vylouhovaný pytlík. Umístěte jí na pozici asi 90° napravo od hrnku.
4. Nastavte počet minut, po které se má čaj louhovat, nalejte do hrnku vodu s požadovanou teplotou a potvrďte joystickem.
5. Robotická ruka provede vymáchání čaje a po nastavené době odnese pytlík napravo od hrnku nad připravenou podložku.
6. Tímto celý cyklus končí, pro jeho opakování je nutné stisknout reset na Arduinu.

### Poznámky

Věnujte opravdovou pečlivost zapojení obvodu. Zejména zapojení LCD panelu je poměrně složité díky velkému počtu vodičů.

Vzhledem k tomu, že se zde pracuje s kapalinou (čajem) dávejte pozor, aby nedošlo k namočení Arduina, motorků nebo jiných elektronických součástek. Pokud k tomuto přece jen dojde, okamžitě odpojte Arduino od zdroje elektrického proudu a nechte vše dobře vyschnout, nejlépe do příštího týdne.

Můžete se setkat s nepřesnostmi v pohybu ruky, které jsou způsobeny tím, že ruka zavadí o nějaký předmět (nejčastěji hrnek). Pokud si tohoto všimnete v počátečních fázích, raději systém resetujte a začněte od začátku.

## Závěr

V tomto příkladu jste se naučili zapojit obvod sestávající z dvou motorků, displeje a joysticku. Dozvěděli jste se něco málo o robotice a vyzkoušeli si ovládání jednoduché robotické ruky. Je třeba si uvědomit, že robotické ruce v praxi mívají nejméně pět stupňů volnosti – pět motorků a jejich ovládání a programování je podstatně složitější.