

```
188     if (m<9) {  
189         m=m+1;}  
190     lcd.setCursor(10,1);  
191     lcd.print(m);  
192  
193     }  
194     delay(300);  
195 }  
196 return m;  
197 }
```

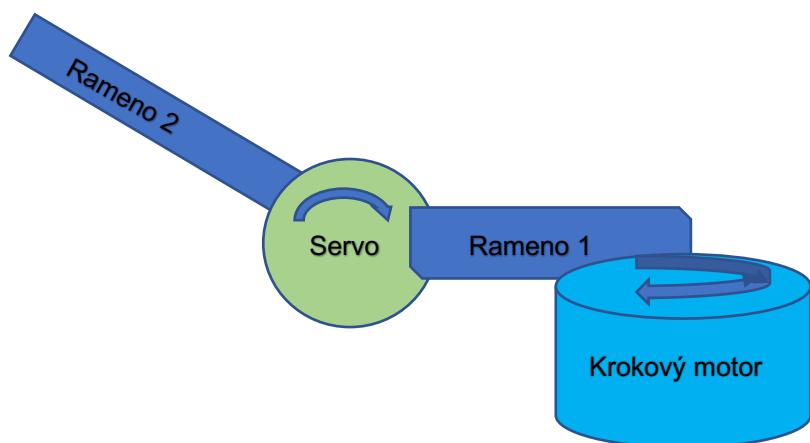
- ④ Program odladěte a nahrajte do Arduina.
- ⑤ Nyní zkuste nasimulovat vymáčkání pytlíku v čaji a až si budete jistí, můžete si opravdu zkusí uvařit čaj. Postupujte dle následujícího návodu.

## 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 vymáčkání 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.



Díly pro tuto robotickou ruku jsou vytisknuty 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 takovou 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 tlačítka, 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 seznámíme.

## POZNÁMKA

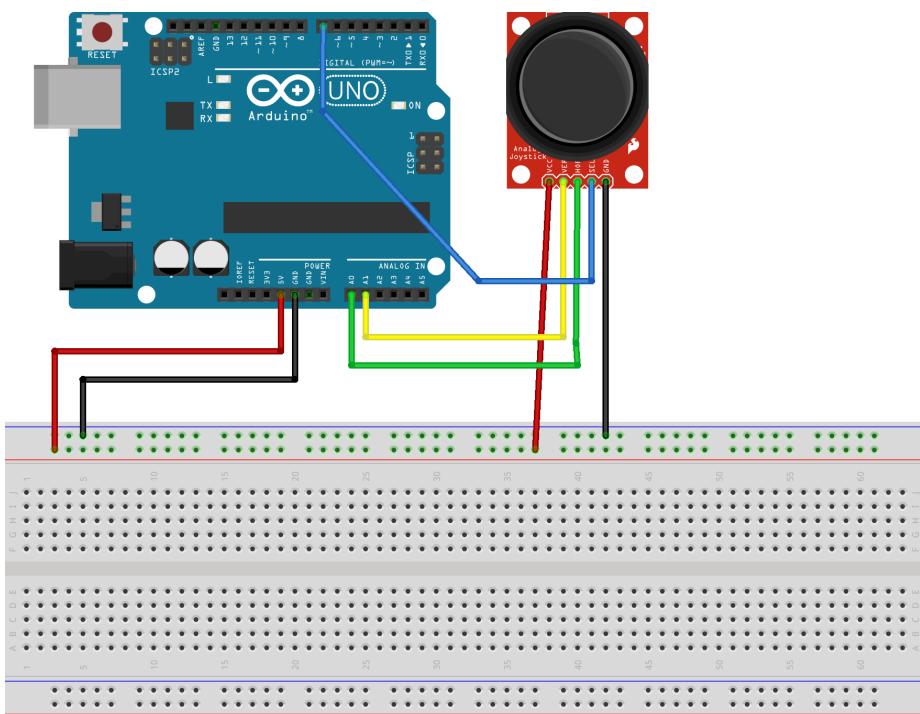
Následující úkoly na sebe navazují, a proto po vyřešení prvního úkolu není nutné obvod rozpojovat, ale naopak postupně k němu budete přidávat další součástky. Stejně tak i program je koncipován od jednoduššího ke složitějšímu.

# ÚKOL 1

Zapojte joystick a monitorujte jeho stavu pomocí sériové komunikace.

## ZAPOJENÍ OBVODU

Sestavte obvod dle následujícího obrázku:



## PROGRAMOVÝ KÓD

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

- 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.



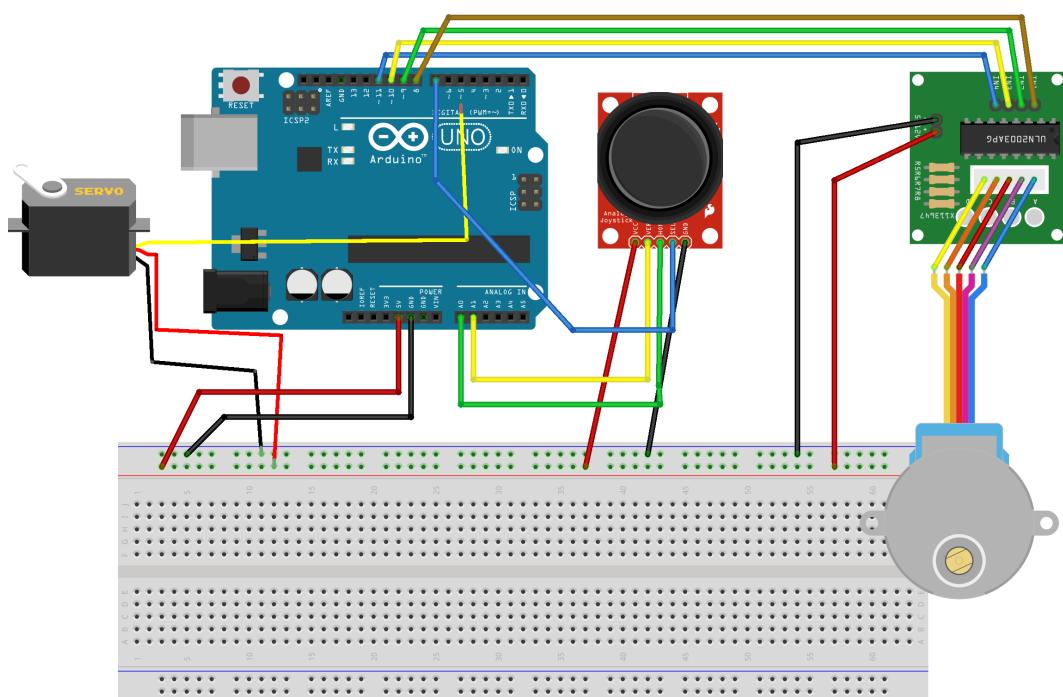
Zkuste úlohu řešit tak, že si místo sériového monitoru zapojíte LCD displej a pohyby joysticku si budete zobrazovat na něm. Můžete využít schéma zapojení a zdrojový kód z Úlohy 3 v této kapitole.

## 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

```
1 #include <Servo.h>
2
3 Servo myservo;//Vytvoření objektu pro řízení krokového
4 motoru
5 int poloha; //Svislá poloha ruky
6
7 // Piny pro krokový motor
8 const int in1 = 8;
9 const int in2 = 9;
10 const int in3 = 10;
11 const int in4 = 11;
12 // proměnná pro nastavení rychlosti,
13 // se zvětšujícím se číslem se rychlosť zmenšuje
14 int rychlosť = 8;
15
16
17 //Joystick
18 int JoyStick_X = 0; //Xová osa - analogový pin 0
19 int JoyStick_Y = 1; //Yová osa - analogový pin 1
20 int JoyStick_Z = 7; //Tlačítko - pin 7
21 int x,y,z;
22
23
24
25 void setup() {
26     myservo.attach(6);//Servo motor je na pinu 6
27     myservo.write(0);//Ruka do výchozí polohy
28     poloha = 0; //Pamatuj si tuto polohu
29     // inicializace digitálních výstupů pro krokový motor
30     pinMode(in1, OUTPUT);
31     pinMode(in2, OUTPUT);
32     pinMode(in3, OUTPUT);
33     pinMode(in4, OUTPUT);
34     //inicjalizace Joysticku
35     pinMode(JoyStick_Z, INPUT_PULLUP); //Nastavení tlačítka
36 joysticku
37 }
38
39 void loop() {
40     pohyb();
41     while(1);
42 }
43
44
45
46
```

The code is annotated with five callout numbers on the right side:

- ① Lines 3-5: Declaration of servo object and hand position variable.
- ② Lines 12-14: Variable declarations for speed and initial speed value.
- ③ Lines 18-21: Variable declarations for joystick axes and button pin.
- ④ Lines 26-35: Setup function block.
- ⑤ Lines 39-42: Loop function block.

```

47 void rotacePoSmeru(int uhel) {
48     for(int i=0;i<(uhel*64/45);i++){
49         krok(1,0,0,0);
50         krok(1,1,0,0);
51         krok(0,1,0,0);
52         krok(0,1,1,0);
53         krok(0,0,1,0);
54         krok(0,0,1,1);
55         krok(0,0,0,1);
56         krok(1,0,0,1);
57     }
58     void rotaceProtismeru(int uhel) {
59         for(int i=0;i<(uhel*64/45);i++){
60             krok(1,0,0,1);
61             krok(0,0,0,1);
62             krok(0,0,1,1);
63             krok(0,0,1,0);
64             krok(0,1,1,0);
65             krok(0,1,0,0);
66             krok(1,1,0,0);
67             krok(1,0,0,0);
68     }
69
70     void krok(int a, int b, int c, int d){
71         digitalWrite(in1, a);
72         digitalWrite(in2, b);
73         digitalWrite(in3, c);
74         digitalWrite(in4, d);
75         delay(rychlost);
76     }
77
78     void pohyb(){
79         int x,y,z;
80         z=1;
81         while (z) {
82             x=analogRead(JoyStick_X);
83             y=analogRead(JoyStick_Y);
84             z=digitalRead(JoyStick_Z);
85             if (x>550) { //doprava
86                 rotacePoSmeru(5);
87             }
88             else if (x<480){ //doleva
89                 rotaceProtismeru(5);
90             }
91             else if (y<480){ //dolu
92                 if (poloha>=5) {
93                     poloha=poloha-5;

```

⑥

⑦

```

94         myservo.write(poloха);
95         delay(1000);
96     }
97 }
98 else if (y>550){ //nahoru
99     if (polоха<=170) {
100         polоха=polоха+5;
101         myservo.write(poloха);
102         delay(1000) ;
103     }
104 }
105 }
106 delay(100);
107 }

```

⑦

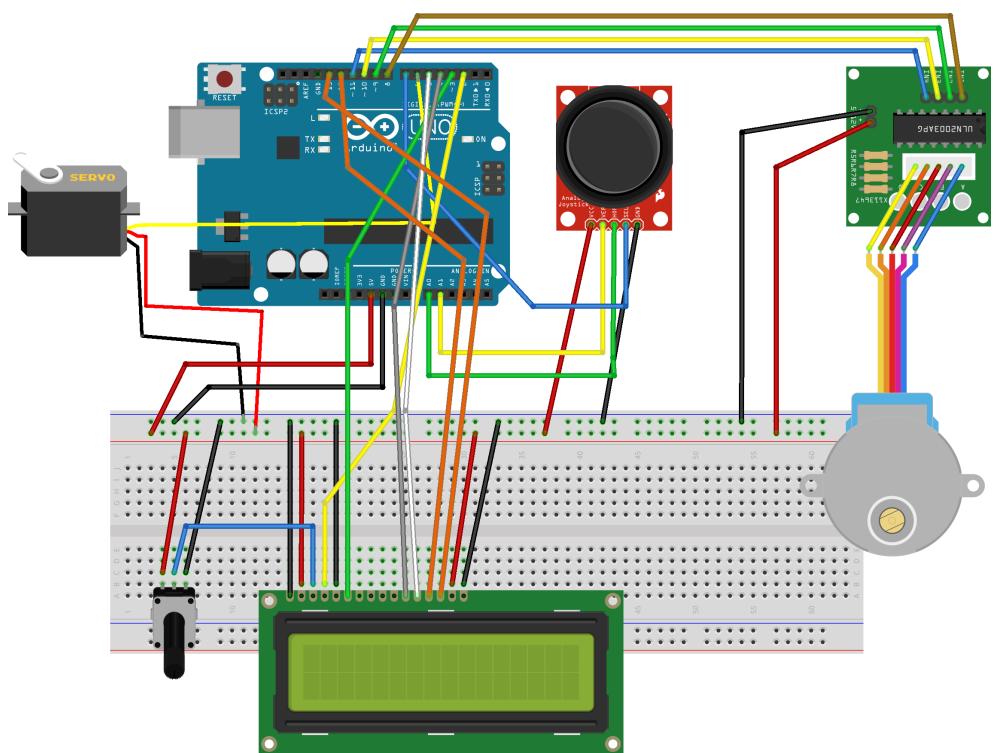
- ④ Nastavení servo motoru (svislý pohyb). Proměnná poloха slouží k zapamatování aktuální svislé polohy ruky.
- ⑤ Nastavení krokového motoru (vodorovný pohyb). Proměnná rychlosť slouží pro nastavení pauzy mezi jednotlivými pohyby motoru.
- ⑥ Nastavení pinů joysticku (viz předchozí příklad).
- ⑦ Inicializace potřebných proměnných a knihoven.
- ⑧ 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.
- ⑨ 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.
- ⑩ 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 dobou 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.

```
1 #include <LiquidCrystal.h> ①
2
3 #include <Servo.h>
4
5 Servo myservo; //Vytvoření objektu pro řízení krokového
6 motoru
7
8 int poloha; //Svislá poloha ruky
9
10 // Piny pro krokový motor
11 const int in1 = 8;
12 const int in2 = 9;
13 const int in3 = 10;
14 const int in4 = 11;
15
16 // Proměnná pro nastavení rychlosti,
17 // se zvětšujícím se číslem se rychlosť zmenšuje
18 int rychlosť = 8;
19 int uhel1;
20 int x,y,z;
21 int i,j;
22
23 //Joystick
24 int JoyStick_X = 0; //Xová osa joysticku - analogový pin 0
25 int JoyStick_Y = 1; //Yová osa joysticku - analogový pin 1
26 int JoyStick_Z = 7; //Tlačítko joysticku - pin 7
27
28 // Piny pro připojení displeje
29 LiquidCrystal lcd(2, 3, 4, 5, 12, 13); ①
30
31 int minut; //Počet minut pro máchání čaje
32
33 void setup() {
34     myservo.attach(6); //Servo motor je na pinu 6
35     myservo.write(0); //Ruka do výchozí polohy
36     poloha = 0; //Pamatuj si tuto polohu
37     // inicializace digitálních výstupů pro krokový motor
38     pinMode(in1, OUTPUT);
39     pinMode(in2, OUTPUT);
40     pinMode(in3, OUTPUT);
41     pinMode(in4, OUTPUT);
42     //inicIALIZACE Joysticku
43 }
```

```
44     pinMode(JoyStick_Z, INPUT_PULLUP); //Nastavení tlačítka  
45     joysticku  
46     lcd.begin(16, 2); // Počet sloupců a řádek LCD displeje  
47  
48 }  
49  
50 void loop() {  
51     //main program  
52     lcd.clear();  
53     lcd.print("Pouzij joystick");  
54     lcd.setCursor(0,2);  
55     lcd.print("pro nastavení");  
56     pocatecni_nastaveni(); //Nastavení polohy ruky "nad  
57     hrnek"  
58     lcd.clear(); //Nastavení polohy pro připevnění pytlíku  
59     myservo.write(poloha+15);  
60     delay(1000);  
61     rotaceProtismeru(90);  
62     delay(1000);  
63     lcd.print("Ruka pripravena");  
64     lcd.setCursor(0,2);  
65     lcd.print("pripevni caj a potvrd"); //Potvrdit stiskem  
66     joysticku  
67     delay(1000);  
68     z=1;  
69     while (z) {  
70         x=analogRead(JoyStick_X);  
71         y=analogRead(JoyStick_Y);  
72         z=digitalRead(JoyStick_Z);  
73         delay(100);  
74         }  
75         delay(1000);  
76         minut=pocetMinut(); //Nastavení počtu minut pro máchání  
77         delay(1000);  
78         rotacePoSmeru(90); //Najedeme nad čaj  
79         delay(1000);  
80         myservo.write(poloha-5); //Máhání čaje  
81         for (i=minut;i;i--)  
82         {  
83             lcd.clear();  
84             lcd.print("Zbyva:");  
85             lcd.setCursor(0,1);  
86             lcd.print(i);  
87             lcd.print(" minut");  
88             for (j=1;j<10;j++){  
89                 myservo.write(poloha-7);  
90                 delay(3000);  
91             }  
92         }  
93     }  
94 }
```

②

```
91         myservo.write(poloha+7);
92         delay(3000);
93     }
94 }
95 lcd.clear(); //Konec máchání
96 lcd.print("Hotovo");
97 myservo.write(poloha+20);
98 delay(10000);
99 rotacePoSmeru(90); //Odjezd doprava
100 myservo.write(15);
101 while(1) {} //Nekonečná smyčka
102 }
103
104 // zde následují funkce pro volání jednotlivých
105 // kroků pro otočení po či proti směru hodinových
106 // ručiček
107 void rotacePoSmeru(int uhel) {
108     for(int i=0;i<(uhel*64/45);i++){
109         krok(1,0,0,0);
110         krok(1,1,0,0);
111         krok(0,1,0,0);
112         krok(0,1,1,0);
113         krok(0,0,1,0);
114         krok(0,0,1,1);
115         krok(0,0,0,1);
116         krok(1,0,0,1);
117     }
118     void rotaceProtiSmeru(int uhel) {
119         for(int i=0;i<(uhel*64/45);i++){
120             krok(1,0,0,1);
121             krok(0,0,0,1);
122             krok(0,0,1,1);
123             krok(0,0,1,0);
124             krok(0,1,1,0);
125             krok(0,1,0,0);
126             krok(1,1,0,0);
127             krok(1,0,0,0);
128     }
129     // každý krok obsahuje výrobcem dané pořadí
130     // pro správné spínání motoru a následnou
131     // pauzu, kterou určujeme rychlosť otáčení
132     void krok(int a, int b, int c, int d){
133         digitalWrite(in1, a);
134         digitalWrite(in2, b);
135         digitalWrite(in3, c);
136         digitalWrite(in4, d);
137         delay(rychlosť);
```

```
138 }
139
140 int pocetMinut(){
141     int m=2;
142     int x, y, z;
143     z=1;
144     lcd.setCursor(0,0);
145     lcd.print("Maximum 9 minut");
146     lcd.setCursor(0,1);
147     lcd.print("Louhovat: ");
148     lcd.print(m);
149     lcd.print(" min");
150     while (z) {
151         x=analogRead(JoyStick_X);
152         y=analogRead(JoyStick_Y);
153         z=digitalRead(JoyStick_Z);
154         if (y>550){ //dolu
155             if (m) {
156                 m=m-1;}
157             lcd.setCursor(10,1);
158             lcd.print(m);
159         }
160         if (y<480){ //nahoru
161             if (m<9) {
162                 m=m+1;}
163             lcd.setCursor(10,1);
164             lcd.print(m);
165         }
166         delay(300);
167     }
168 }
169     return m;
170 }
171
172 void pocatecni_nastaveni(){
173     int x,y,z;
174     z=1;
175     while (z) {
176         x=analogRead(JoyStick_X);
177         y=analogRead(JoyStick_Y);
178         z=digitalRead(JoyStick_Z);
179         if (x>550) { //doprava
180             rotacePoSmeru(5);
181         }
182         else if (x<480){ //doleva
183             rotaceProtiSmeru(5);
```

③

```

185 }
186 else if (y<480){ //dolu
187     if (poloha>=5) {
188         poloha=poloha-5;
189         myservo.write(poloha);
190         delay(1000);
191     }
192 }
193 else if (y>550){ //nahoru
194     if (poloha<=170) {
195         poloha=poloha+5;
196         myservo.write(poloha);
197         delay(1000);
198     }
199 }
200 }
201 delay(100);
}

```

- ① Nastavení LCD panelu včetně pinů, na které je zapojen.
- ② Vlastní program, tentokrát složitější. Postupně se prochází následující kroky:
  - a. Počáteční nastavení polohy ruky – nad hrnek.
  - b. Přesun ruky na pozici, kde se připevní pytlík čaje.
  - c. Nastavení doby pro máchání pytlíku.
  - d. Vlastní máchání pytlíku.
  - e. Posun ruky na místo, kde je možné pytlík sejmout.
  - f. 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 louhat, nalejte do hrnku vodu s požadovanou teplotou a potvrďte joystickem.
- ⑤ Robotická ruka provede vymáhání čaje a po nastavené době odnesou 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.