|  |  |
| --- | --- |
| **Stredná priemyselná škola**  **Novomeského 5/24, 036 36, Martin** | |
| **STREDOŠKOLSKÁ ODBORNÁ ČINNOSŤ**  č. odboru:12  Elektrotechnika a hardware | |
| Automatická miešačka nápojov riadená Arduinom | |
| **2017** | riešitelia  **Jozef Chovanec**  **Ľubomír Csicsai** |
| Martin | ročník štúdia : **štvrtý** |

|  |  |
| --- | --- |
| **Stredná priemyselná škola**  **Novomeského 5/24, 036 36, Martin**  **Zriaďovateľ**  **Žilinský samosprávny kraj** | |
| **STREDOŠKOLSKÁ ODBORNÁ ČINNOSŤ**  č. odboru: 12  Elektrotechnika a hardware | |
| Automatická miešačka nápojov riadená Arduinom | |
| **2017** | riešitelia  **Jozef Chovanec**  **Ľubomír Csicsai** |
| Martin | ročník štúdia : **štvrtý** |
|  | školiteľ  Ing. Eduard Jadroň |

|  |
| --- |
| **Čestné vyhlásenie**  Prehlasujem, že prácu s názvom „Automatická miešačka nápojov“, som vypracoval samostatne s použitím uvedenej literatúry a súčasne, že som túto prácu, neprihlásil a neprezentoval v žiadnej inej súťaži vyhlásenej – MŠVVaŠ SR.  Martin, 20. február 2017 ..........................................  *vlastnoručný podpis* |

|  |
| --- |
| **Čestné vyhlásenie**  Prehlasujem, že prácu s názvom „Automatická miešačka nápojov“, som vypracoval samostatne s použitím uvedenej literatúry a súčasne, že som túto prácu, neprihlásil a neprezentoval v žiadnej inej súťaži vyhlásenej – MŠVVaŠ SR.  Martin, 20. február 2017 ..........................................  *vlastnoručný podpis* |

|  |
| --- |
|  |

|  |
| --- |
| **Poďakovanie**  Chceli by sme sa poďakovať „**Jaroslav Johanides“** za pomoc s konštrukciou ,ktorú nám pomohol pozvárať a „**Ľubomír Csicsai starší“** za asistenciu,  za to im pekne ďakujeme. |

**Obsah**

[Úvod 1](#_Toc474064913)

[1 Problematika a prehľad literatúry 1](#_Toc474064914)

[2 Cieľ práce 1](#_Toc474064916)

[3 Materiál a metodika práce 1](#_Toc474064917)

[4 Výsledky práce 1](#_Toc474064918)

[5 Diskusia 1](#_Toc474064919)

[6 Závery práce 1](#_Toc474064920)

[Zhrnutie 1](#_Toc474064921)

[Zoznam použitej literatúry 1](#_Toc474064922)

[Prílohy 1](#_Toc474064923)

Úvod

Automatická miešačka nápojov slúži na uľahčenie života nejedného barmana. Budeme si môcť s pomocou mikroprocesora namiešať nápoj podľa našej vôle a nebudeme musieť zdvíhať fľaše . Zariadenie sa samo o to postará a namieša nám nápoj podľa našej chuti a výberu.

1. Problematika a prehľad literatúry

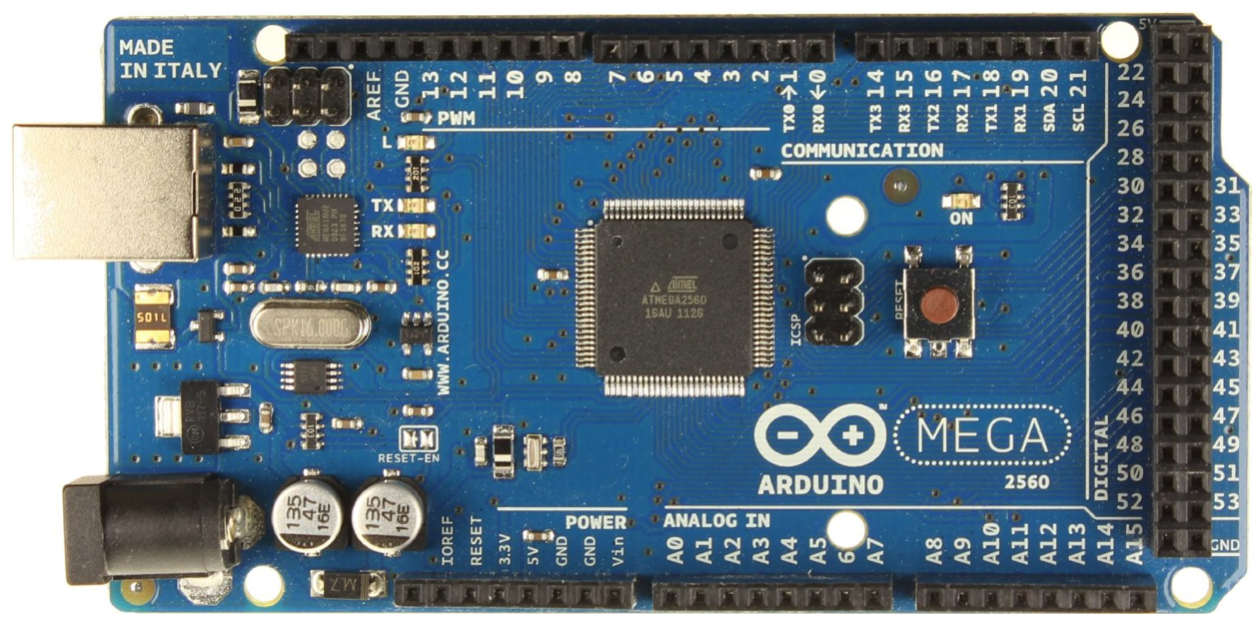
**1.1 O Arduine**

Vývoj prvého Arduina sa píše k roku 2005, keď ľudia z talianského Interaction Design Institute v meste Ivrea sa rozhodli vytvoriť jednoduchý lacný vývojový set pre študentov, ktorý si nechceli kupovať, v tej dobe rozšírené a drahé dosky Basic Stamp. Medzi študentami sa Arduino uchytilo, a tak sa autori rozhodli poskytnúť celému svetu. A to nielen predajom vlastných dosiek ale aj zdieľaním všetkých schém a návodov (týka sa to Open Source projektov). Programová časť Arduina bola založená na Processing, čo je programovací jazyk s vlastným editorom, určený pre výučbu programovania. V dnešnej dobe sa predalo už niekoľko stotisíc dosiek Arduino. Dôkazom, že táto platforma nie je mŕtva, môže byť i to, že Microsoft oznámil spoluprácu s Arduinom. Windows sa stal prvým operačným systémom, oficiálne certifikovaným Arduinom a predstavil svetu Windows Virtuálne shieldy pre Arduino, ktoré veľmi zjednodušujú prepojenie fyzického sveta okolo Arduina a výpočtového výkonu Windows zariadení. Za desať rokov vývoja vzniklo veľa rôzných typov Arduina. Pretože sa jedná o opensource projekt, vznikalo spoločne s hlavnou líniou projektu i mnoho ďalších, neoficiálnych typov. Takzvaných klonov.



Obr. 1 Oficiálne logo Arduina ([www.arduino.cc](http://www.arduino.cc))

**1.2 Arduino Mega 2560 (čínsky klon)**



## Vývojová doska MEGA je mikropočítačová vývojová doska založená na ATmega2560, kompatibilná s Arduino Mega Nová verzia R3 REV3

Má 54 digitálnych vstupno/výstupných pinov (z nich 14 môže byť použité ako PWM analogové výstupy), 16 analogových vstupov, 4x UART (hardwarové sériové porty), 16 MHz kryštalový oscilátor, USB konektor pre pripojenie s PC, napájací jack, ICSP konektor a resetovacie tlačídlo.  
Doska obsahuje všetko potrebné pre beh mikroprocesora. Len ju pripojíte k PC USB kabelom alebo pripojíte napájenie z AC/DC adaptéra alebo batérie. Mega je kompatibilné s väčšinou rozšírení (shieldov) vytvorených pre iné verzie Arduina, ako je napr. Uno.

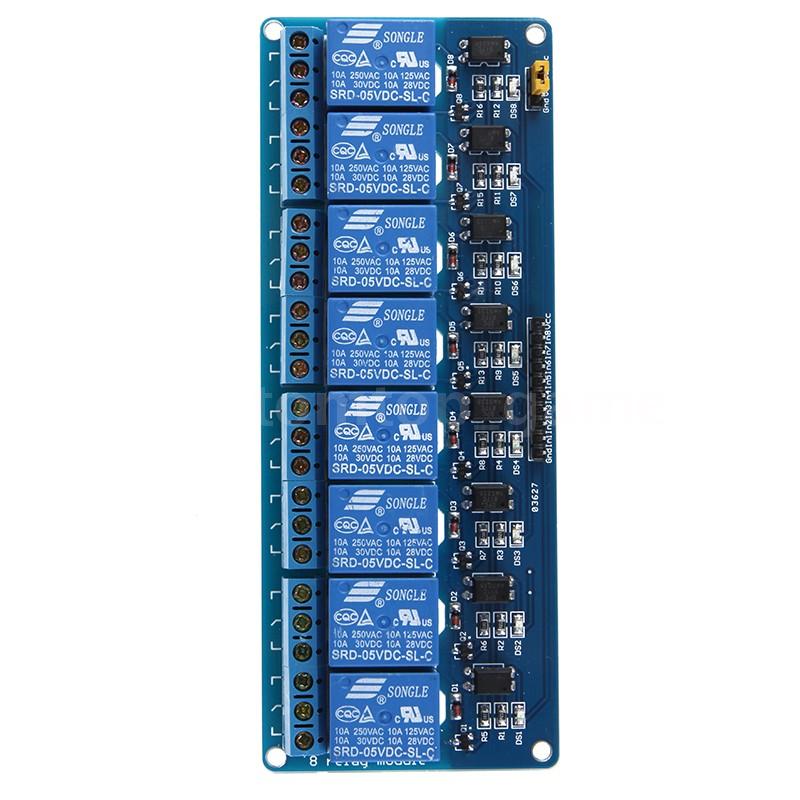
Výhodou tejto verzie oproti Arduino Uno je väčší počet vstupov/výstupov a hlavne 256kB flash pamäte (oproti 32kB u Uno). Do pamäte tejto veľkosti sa vojdú aj tie najzložitejšie programy.  
  
- ATmega2560  
- 54 digitálních vstupov/výstupov, z nich 14 môže byť použitých ako PWM analogové výstupy  
- 16 analogových vstupov  
- Kryštál 16MHz  
- Pripojenie k PC cez USB  
- Napájanie 7-12V  
- Pamäť programu Flash 256 KB, 4 KB použité bootloaderom  
- Pamäť dát SRAM 8 KB  
- Pamäť dát EEPROM 4 KB  
- max. prúd poskytovaný dig. I/O pinami - 40 mA  
- max. prúd poskytovaný na 3.3V pine - 50 mA  
  
Doska poskytuje úžasne možnosti merania, monitorovania a ovládania zariadení, a to pomocou jednoduchého programovacieho prostredia Arduino, vychádzajúceho z jazyka C++. Součástí softwarového balíčku je spousta příkladů a knihovny pro ovládání ďalších rozširujúcich modulov (shieldov).

**1.3 8-kanálový relay modul.**

## Modul 8x 5V relé pre Arduino, PIC, ARM, AVR, DSP.

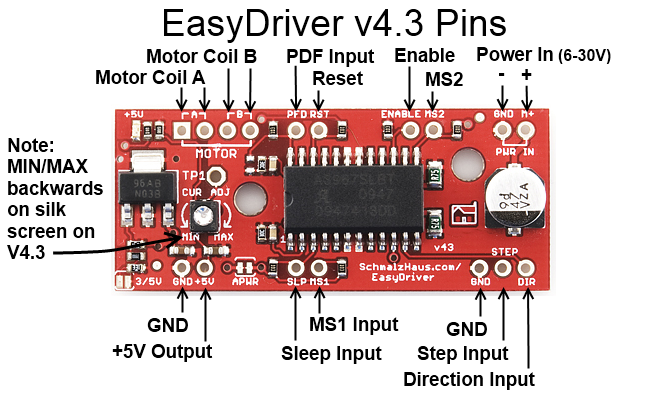
Umožňuje ovládať zariadenia s veľkými prúdovými nárokmi.  
Tento modul môže byť použitý s Arduino Shield V4.0.  
Jednoduché ovládanie veľkým množstvom mikroprocesorov (Arduino , 8051, AVR, PIC, DSP, ARM, MSP430, PLC,TTL logic).  
**Pri odpojení jumpera VCC-VCCJD je možné použiť 5V na cievky a spínanie ovládať zemou z Raspberry GPIO 3,3V. Je nutné pripojiť GND.**

Relé AC250V 10A; DC30V 10A.  
Rozmery: 13.9 x 5.2 x 1.7 cm.



Obr. 3 | 8 Channel DC 5V Relay Module for Arduino (www.ebay.com)

**1.4 Radič na krokový motor „EasyDriver“**



Driver A3967 k riadeniu krokových motorov  
MS1 a MS2 rozlíšenie mikrokrokov – plný krok, polovičný krok, štvrtinový krok a osminový krok  
Kompatibilní s 4, 6 a 8 vývodovými krokovými motory  
Nastaviteľný prúd od 150mA do 750mA na fáze  
Riadiace napájanie : 5V  
Rozsah napajacích napätí od 6 do 30V /čím väčšie napätie tým väčší krútiaci moment/  
Rozmery dosky: 48mm x 20mm**1.5 Krokový motor NEMA17**



Krokový motor s krútiacim momentom 0.36Nm.  
Krokový motor pre menší CNC stroje, ponúkajúci presnosť 200 krokov na otáčku.

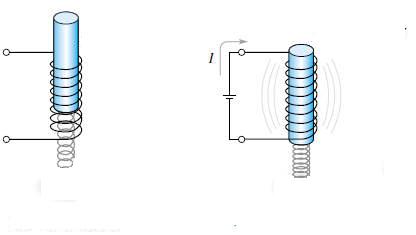
Moment 0.36Nm (3.6 Kg/cm).   
  
Inštalačná príruba: Nema17  
Počet vývodov: 4  
Definovaný prúd: 1,68 A  
Moment: 0,36 Nm  
Dĺžka motora: 38 mm  
Šírka/Výška motoru: 42x42 mm  
Priemer osy: 5 mm  
Osi motora A - Osa na jednu stranu  
Priemer inštalačných otvorov: 3 mm  
Inštalačné rozpätie: 31x31 mm  
Krokov na otáčku: 200 Krokov / 1.8° krok

**1.6 Solenoid 12V**



Je to cievka s hustými závitmi rovnakého kruhového tvaru po celej dĺžke. Dĺžka takej cievky zvyčajne prevyšuje jej priemer, magnetické pole vnútri cievky sa tak zvyčajne považuje za rovnomerné (konštantné).

* Napätie: DC 12V
* Výkon: 8W
* Prúd: 0.6A
* Tlak: 0.02- 0.8Mpa
* Maximálna teplota:100°C
* Operačný mód: normálne zatvorené



1. Cieľ práce

Hlavný cieľ bol zostrojiť automatickú miešačku nápojov, ktorá nám v prvom rade uľahčí život. Nebudem musieť neprakticky nalievať manuálne rukou, ale s pomocou počítača si zvoliť nápoj podľa svojej vôle. Šesť ľubovolných nápojov sa bude dať navzájom skombinovať. Posuvný pás posunie pohár pod ľubovoľnú (podľa programu) nádobu s tekutinou. Spustením solenoidov nám sa otvorí vzduchový otvor vo vrchnej časti nádoby a uvoľní sa nám podtlak ktorí nám držal tekutinu vo fľaši.

Pre dosiahnutie hlavného cieľa bolo potrebné zostrojiť konštrukciu ktorá by dokázala uniesť hmotnosť sklenených nádob na tekutiny a bola dostatočne stabilná. Medzi podmienky patrilo aj to, aby hadičky v solenoidoch dostatočne tesnili. na koši dostatok miesta pre umiestnenie infračerveného senzora, servo motora, a Arduina.

Napísanie programu bola najväčšia výzva, pretože sme nemali žiadne skúsenosti s programovaním v jazyku c.

1. Materiál a metodika práce

Prvá vec ktorou sme sa začali zaoberať bolo dostať nápad a predstaviť si ho v mysli. Pokračovanie boli hrubé návrhy na papiery. Následne sme začali tvoriť zoznam súčiastok a ich objednaní z portálu ebay. Kým boli súčiastky na ceste začali sme sa zaoberať strojníckym návrhom konštrukcie a následne jej tvorbou, striekaním, zváraním s ktorým nám pomohol známi. Keď súčiastky dorazili začali sme sa zoznamovať s arduinom a pomaly sme začali kódovať a zapájať všetko do seba. Postupom času sme zdokonali zapojenie súčiastok a samotný kód ktorý sa nam rapídne rozšíril. Až sme prišli ku súčastnému stavu.

/\*

\* Automatická miešačka nápojov riadená Arduinom

\* Software: Jozef Chovanec

\* Hardware: Ľubomír Csicsai

\*/

//zadefinovanie pinov na solenoidy

#define Solenoid\_1 2

#define Solenoid\_2 3

#define Solenoid\_3 4

#define Solenoid\_4 5

#define Solenoid\_5 6

#define Solenoid\_6 7

#include <Stepper.h> //kniznica na krokovy motor

int prepinac; //switch

int dlzka\_pasu = 15000; //dlzka pasu

int rychlost\_pasu = 100; //rychlost

//piny na motor

int dirPin = 8;

int stepPin = 9;

int cas\_nalievania = 3000; //cas nalievania napoja, definuje aj jeho objem

int cas\_posunu = 5000; //cas presunu pasu ku inemu napoju

void setup() {

Serial.begin(9600); //pripojenie ku seriovemu rozhraniu

Serial.print("Prosim zadajte poradie namiesania vo formate [prvy],[druhy],[treti] ");

//inicializovanie pinov na solenoidoch

pinMode(Solenoid\_1, OUTPUT);

pinMode(Solenoid\_2, OUTPUT);

pinMode(Solenoid\_3, OUTPUT);

pinMode(Solenoid\_4, OUTPUT);

pinMode(Solenoid\_5, OUTPUT);

pinMode(Solenoid\_6, OUTPUT);

//inicializovanie pinov na krokovy motor

pinMode(dirPin, OUTPUT);

pinMode(stepPin, OUTPUT);

digitalWrite(dirPin, LOW);

digitalWrite(stepPin, LOW);

prepinac = 1;

}

void loop() {

//zapnutie do stavu ON

digitalWrite(Solenoid\_1, LOW);

digitalWrite(Solenoid\_2, LOW);

digitalWrite(Solenoid\_3, LOW);

digitalWrite(Solenoid\_4, LOW);

digitalWrite(Solenoid\_5, LOW);

digitalWrite(Solenoid\_6, LOW);

delay(1000);

if (Serial.available() > 0) //cakanie na odpoved

{

for(int i=1; i<=3; i++) {

String prvy = Serial.readStringUntil(','); //precitanie hodnoty prveho napoja

if(prvy != ""){

String druhy = Serial.readStringUntil(','); //precitanie hodnoty druheho napoja

if(druhy != ""){

String treti = Serial.readStringUntil(','); //precitanie hodnoty tretieho napoja

Serial.print("\nNamiesavam napoj v nasledujucom poradi: \n");

Serial.print(prvy);

Serial.print("\n");

Serial.print(druhy);

Serial.print("\n");

Serial.print(treti);

//skonvertiovanie na integer

int prvy\_int = prvy.toInt();

int druhy\_int = druhy.toInt();

int treti\_int = treti.toInt();

//osetrenie pre pripad vpisania chybnej hodnoty

if(prvy\_int > 6 or prvy\_int < 1)

{

Serial.print("\n Error: Iba cislo od 1-6 !\n");

break;

}

if(druhy\_int > 6 or druhy\_int < 1)

{

Serial.print("\n Error: Iba cislo od 1-6 !\n");

break;

}

if(treti\_int > 6 or treti\_int < 1)

{

Serial.print("\n Error: Iba cislo od 1-6 !\n");

break;

}

delay(3000);

if (prepinac < dlzka\_pasu + 1) // Je stale v prvej pozici ?

{

digitalWrite(dirPin, LOW); // vypne pin smerovania

}

else

{

digitalWrite(dirPin, HIGH); // zapne pin smerovania

}

digitalWrite(stepPin, HIGH);

delayMicroseconds(rychlost\_pasu); // vycka urcitu dobu

digitalWrite(stepPin, LOW);

delayMicroseconds(rychlost\_pasu); // vycka urcitu dobu

prepinac += 1; // pocitanie smeru posunu

if (prepinac > dlzka\_pasu \* 2) // ak prejdeme obydva smery

{ prepinac = 1; } // restartujeme prepinac spat

// ziskanie hodnoty prveho napoja

if(prvy\_int == 1)

{

digitalWrite(Solenoid\_1, HIGH); // otvori solenoid

Serial.print("\nNapoj c.1 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_1, LOW); // zavrie solenoid

Serial.print("Napoj c.1 sa nalial. \n");

}

if(prvy\_int == 2)

{

digitalWrite(Solenoid\_2, HIGH);

Serial.print("\nNapoj c.2 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_2, LOW);

Serial.print("Napoj c.2 sa nalial. \n");

}

if(prvy\_int == 3)

{

digitalWrite(Solenoid\_3, HIGH);

Serial.print("\nNapoj c.3 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_3, LOW);

Serial.print("Napoj c.3 sa nalial. \n");

}

if(prvy\_int == 4)

{

digitalWrite(Solenoid\_4, HIGH);

Serial.print("\nNapoj c.4 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_4, LOW);

Serial.print("Napoj c.4 sa nalial. \n");

}

if(prvy\_int == 5)

{

digitalWrite(Solenoid\_5, HIGH);

Serial.print("\nNapoj c.5 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_5, LOW);

Serial.print("Napoj c.5 sa nalial. \n");

}

if(prvy\_int == 6)

{

digitalWrite(Solenoid\_6, HIGH);

Serial.print("\nNapoj c.6 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_6, LOW);

Serial.print("Napoj c.6 sa nalial. \n");

}

if (prepinac < dlzka\_pasu + 1) // Je stale v prvej pozici ?

{

digitalWrite(dirPin, LOW); // vypne pin smerovania

}

else

{

digitalWrite(dirPin, HIGH); // zapne pin smerovania

}

digitalWrite(stepPin, HIGH);

delayMicroseconds(rychlost\_pasu); // vycka urcitu dobu

digitalWrite(stepPin, LOW);

delayMicroseconds(rychlost\_pasu); // vycka urcitu dobu

prepinac += 1; // pocitanie smeru posunu

if (prepinac > dlzka\_pasu \* 2) // ak prejdeme obydva smery

{ prepinac = 1; } // restartujeme prepinac spat

delay(cas\_posunu);

// ziskanie hodnoty druheho napoja

if(druhy\_int == 1)

{

digitalWrite(Solenoid\_1, HIGH);

Serial.print("\nNapoj c.1 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_1, LOW);

Serial.print("Napoj c.1 sa nalial. \n");

}

if(druhy\_int == 2)

{

digitalWrite(Solenoid\_2, HIGH);

Serial.print("\nNapoj c.2 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_2, LOW);

Serial.print("Napoj c.2 sa nalial. \n");

}

if(druhy\_int == 3)

{

digitalWrite(Solenoid\_3, HIGH);

Serial.print("\nNapoj c.3 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_3, LOW);

Serial.print("Napoj c.3 sa nalial. \n");

}

if(druhy\_int == 4)

{

digitalWrite(Solenoid\_4, HIGH);

Serial.print("\nNapoj c.4 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_4, LOW);

Serial.print("Napoj c.4 sa nalial. \n");

}

if(druhy\_int == 5)

{

digitalWrite(Solenoid\_5, HIGH);

Serial.print("\nNapoj c.5 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_5, LOW);

Serial.print("Napoj c.5 sa nalial. \n");

}

if(druhy\_int == 6)

{

digitalWrite(Solenoid\_6, HIGH);

Serial.print("\nNapoj c.6 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_6, LOW);

Serial.print("Napoj c.6 sa nalial. \n");

}

if (prepinac < dlzka\_pasu + 1) // Je stale v prvej pozici ?

{

digitalWrite(dirPin, LOW); // vypne pin smerovania

}

else

{

digitalWrite(dirPin, HIGH); // zapne pin smerovania

}

digitalWrite(stepPin, HIGH);

delayMicroseconds(rychlost\_pasu); // vycka urcitu dobu

digitalWrite(stepPin, LOW);

delayMicroseconds(rychlost\_pasu); // vycka urcitu dobu

prepinac += 1; // pocitanie smeru posunu

if (prepinac > dlzka\_pasu \* 2) // ak prejdeme obydva smery

{ prepinac = 1; } // restartujeme prepinac spat

\*/

delay(cas\_posunu);

// ziskanie hodnoty tretieho napoja

if(treti\_int == 1)

{

digitalWrite(Solenoid\_1, HIGH);

Serial.print("\nNapoj c.1 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_1, LOW);

Serial.print("Napoj c.1 sa nalial. \n");

}

if(treti\_int == 2)

{

digitalWrite(Solenoid\_2, HIGH);

Serial.print("\nNapoj c.2 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_2, LOW);

Serial.print("Napoj c.2 sa nalial. \n");

}

if(treti\_int == 3)

{

digitalWrite(Solenoid\_3, HIGH);

Serial.print("\nNapoj c.3 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_3, LOW);

Serial.print("Napoj c.3 sa nalial. \n");

}

if(treti\_int == 4)

{

digitalWrite(Solenoid\_4, HIGH);

Serial.print("\nNapoj c.4 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_4, LOW);

Serial.print("Napoj c.4 sa nalial. \n");

}

if(treti\_int == 5)

{

digitalWrite(Solenoid\_5, HIGH);

Serial.print("\nNapoj c.5 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_5, LOW);

Serial.print("Napoj c.5 sa nalial. \n");

}

if(treti\_int == 6)

{

digitalWrite(Solenoid\_6, HIGH);

Serial.print("\nNapoj c.6 sa nalieva... \n");

delay(cas\_nalievania);

digitalWrite(Solenoid\_6, LOW);

Serial.print("Napoj c.6 sa nalial. \n");

}

Serial.print("Napoj uspesne namiesany \n");

if (prepinac < dlzka\_pasu + 1) // Je stale v prvej pozici ?

{

digitalWrite(dirPin, LOW); // vypne pin smerovania

}

else

{

digitalWrite(dirPin, HIGH); // zapne pin smerovania

}

digitalWrite(stepPin, HIGH);

delayMicroseconds(rychlost\_pasu); // vycka urcitu dobu

digitalWrite(stepPin, LOW);

delayMicroseconds(rychlost\_pasu); // vycka urcitu dobu

prepinac += 1; // pocitanie smeru posunu

if (prepinac > dlzka\_pasu \* 2) // ak prejdeme obydva smery

{ prepinac = 1; } // restartujeme prepinac spat

}

}

}

}

}

1. Výsledky práce

Máme k dispoziícií miešačku nápojov ktorá podľa vašej vôľe Vám bude namiešavať nápoj. Ma až širokospektrálne využitie, napr. podnikoch, v domácnosti a hlavne v baroch.

Súčastný stav nášho výdobytku nieje finálny plánujeme pracovať na daľších detailoch, funkciach, dizajne a zdokonalovanie.

So súčastným stavom ešte niesme úplne spokojný a budeme tvrdo pracovať na jeho vylepšeniach.

Diskusia

Naše vyhotovenie automatickej miesačky aby bola aplikovatelná v praxi čiže v podnikoch alebo baroch by musel byť optimalizovaná na nátlak zákoznikov aby zládala ich požiadavky na plinulé plnenie pohárov a miešanie nápojov. Nebolo by to nič jednoduché ,ale náklady spojené s vyplacaním barmana alebo zamestnanca ,ktorého by miešačka mohla nazhradzovať by sa zminimalizovali. Náklady spojené zo stavbou takéjto miešačky určenej do podniku odhadujem na cca 200€. Čiže keď si to porovnáme s mesačným vyplacaním barmana je to obrovský rozbiel.

1. Závery práce

Automatická miešačka nápojov ma byť určená do podnykov ako náhrada za platenie mzdy pre barmana a čiže zniženie nákladou a aj čiastočne ako prilákanie zákoznikov na takúto možnosť si namiešať a vybrať nápojov úplne sám pomocou pár klikov. Miešačka ttiež musí ustať nátlak alebo nápor zákoznikov, čiže čas namiešania nápoja musí byť najkratší ako sa dá aby zákozníci nečakali príliš dlho a netvorili sa dlhé rady čakajúcich nespokojných zákazníkov. Takže cieľom je znížiť náklady majiteľa podniku a zjednodušenie výberu a miešania nápojov. Ďalšia možnosť ,ktorá by výber dosť uľahčila je aplikácia pre smartphone čiže uživateľ by mal vždy pri sebe ovládací prvok pre miešačku. Ďalší doplnok by bolo platenie cez kartu alebo PayPal, čiže zakazník by pri sebe nemusel mať hotovosť čo je len ďalšia výhoda.

# Zhrnutie

Cieľom bolo zostaviť automatizovanú miešačku nápojov bez potreby zásahu človeka. Tento cieľ sa nám v hrubom prevedení splniť podarilo ,ale budeme ďalej pokračovať a snažiť sa doťiahnuť tento project do konca a maximálne ho zdokonaliť. Či už dokončiť pás, alebo zdokonaliť tesnenie až po tie najmenšie detaily ,ktoré budú dotvárať plne funkčný projekt.

# Zoznam použitej literatúry

Pre citovanie literárnych prameňov ako aj tvorbubibliografických odkazov sa na Slovensku využíva norma STN ISO 690 (1998)a STN ISO 690-2 (2001).

Napr.:

TIMKO, Ján - SIEKEL, Pavol - TURŇA, Juraj. 2004. *Geneticky modifikovanéorganizmy*. Bratislava: Veda, 2004. 104 s. ISBN 80-224-0834-4.

The Silver-Meal Heuristic Method for Deterministic Time-Varying Demand. 2010.[online]. 2010, [cit. 2010-04-26]. Dostupné na internete:

<http://www.shvoong.com/exact-sciences/499883-silver-meal-heuristic-methoddeterministic/>

Prílohy

**Zoznam príloh:**

1. Dotazník
2. Grafické spracovanie výsledkov dotazníka.
3. Nákresy
4. Fotografie
5. CD médium

Na začiatku sa nachádza zoznam všetkých príloh.

Názvy príloh v zozname musia byt’ zhodné s názvami uvedenými na príslušných prílohách. Každá príloha začína na novej strane a je označená samostatným písmenom (Príloha A, Príloha B, ...).

Pri písaní práce sa všetky netextové časti (tabuľky, grafy, mapy, fotografie, CD, ...) umiestňujú do príloh. Tie sú podľa potreby rozčlenené na jednotlivé časti (poradie nie je fixné, väčšinou sa však tabuľky a grafy umiestňujú do prednej časti a fotodokumentácia na koniec príloh).

Podľa charakteru môžu byť zviazané s textovou časťou, alebo dôsledne uložené vo zvláštnom obale tak, aby pri manipulácii s nimi nedošlo k ich poškodeniu, alebo k strate. Sú očíslované a uvedené na zozname príloh a môžu to byť nákresy (ilustrácie), tabulky, grafy, mapy, fotodokumentácia a iný dokumentačný materiál, diskety alebo CD atď.

**Príloha A: Dotazník**

Priložený dotazník.

