Project Name: Automatic Bottle filling Machine.

Introduction:

Our project is to build an automatic bottle filling machine. In this project we have used many components such as Arduino UNO, L293D, IC2 bus, LCD screen (16*2), Stepper motor, Hexadecimal Keypad, LED, Resistors. The main function of this automatic bottle filling machine is that the stepper will rotate and the bottles will be automatically filled up. Here we have set up Arduino code. By using the hexadecimal keypad, we have set that how many bottle will be filled up.

Components List and Price:

Components No.	Components Name	Price(Taka)
01	Arduino Uno	420
02	Stepper Motor	1050
03	LCD	170
04	I2C	150
05	A4988 Module	250
06	Water Pump	100
07	Hexadecimal Keypad	120
08	L293D	80
09	Bread board	100
10	Jumper cable	80
11	Resistors	20
12	Button	15
13	PVC Board	200
14	Glue gum and gum sticks	300
15	Soldering iron	120

Circuit Diagram:

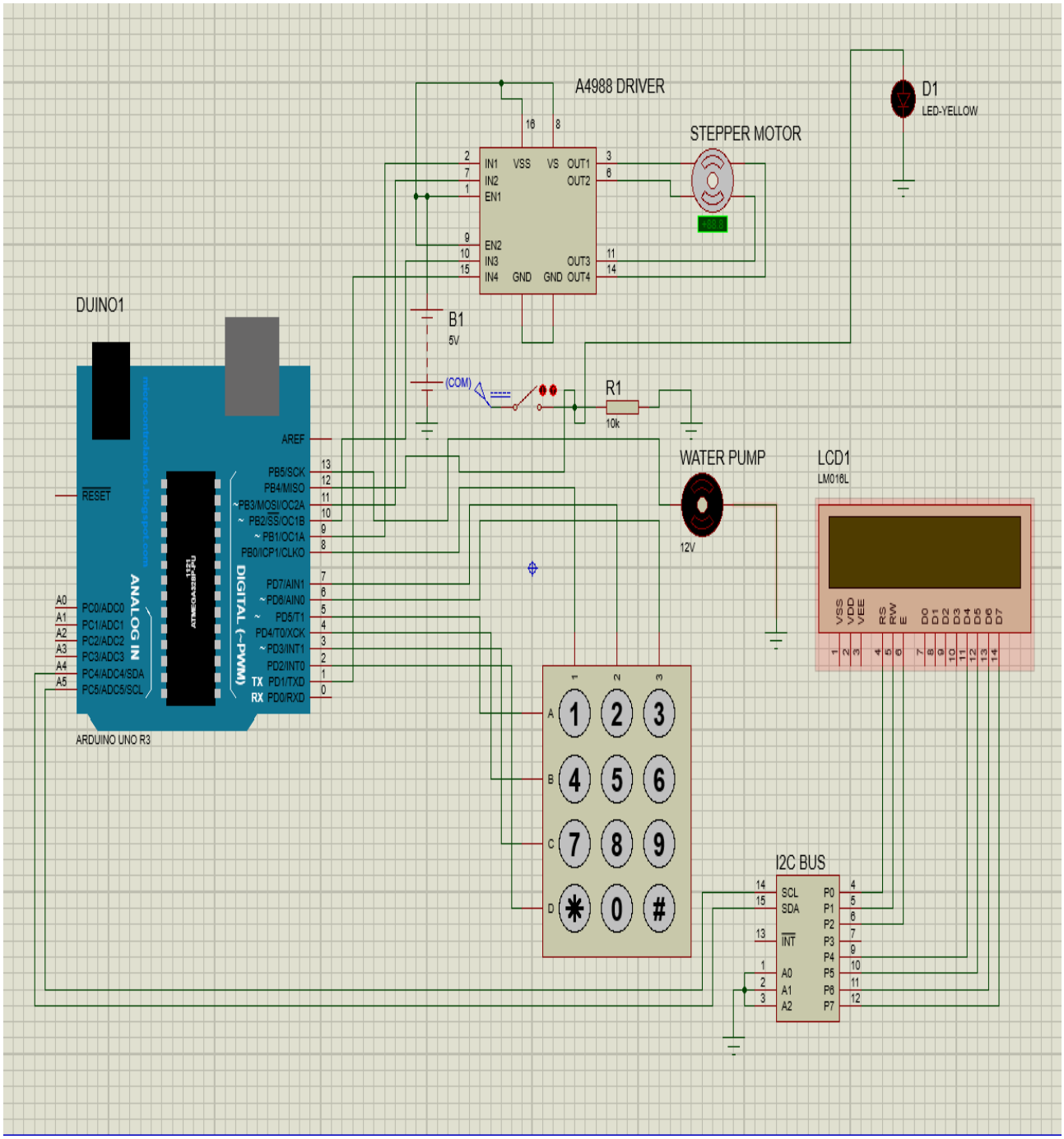

Figure 1 Circuit diagram of automatic bottle filling machine

Arduino Code:

```
#include <Keypad.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
int pulse=650;
//#include <Stepper.h>
const int buttonPin = 12;     // the number of the pushbutton pin
const int ledPin =  10;      // the number of the LED pin
const byte ROWS = 4; //four rows
const byte COLS = 3; //three columns
/*void stepper(bool direction , int steps)
{
  for(int i=0; i<steps ; i++)
  {
  digitalWrite(11,direction);
  digitalWrite(9,1);
  delayMicroseconds(pulse);
  digitalWrite(9,0);
  delayMicroseconds(pulse);
  }
}*/
char keys[ROWS][COLS] = {
  {'1','2','3'},
  {'4','5','6'},
  {'7','8','9'},
  {'*','0','#'}
};
```

```
byte rowPins[ROWS] = {2,3,4,5}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {6,7,8}; //connect to the column pinouts of the keypad

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );
//const int stepsPerRevolution = 200;
//Stepper myStepper(stepsPerRevolution, 8,9, 10, 11);
// variables will change:
int buttonState = 0;         // variable for reading the pushbutton status

void setup() {
  // initialize the LED pin as an output:
  lcd.begin();
  lcd.backlight();
  lcd.setCursor(4,0);
  lcd.print("Hello!");
  delay(2000);
  lcd.clear();
  lcd.print("Enter an integar");
  //lcd.print("initialozing...");
  delay(2000);
  Serial.begin(9600);
  //myStepper.setSpeed(100);
  pinMode(ledPin, OUTPUT);
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin, INPUT);
  pinMode(13,OUTPUT);
  pinMode(9,OUTPUT);
  pinMode(11,OUTPUT);
```

```
  Serial.begin(9600);
}
int flag=0;
int p=0;
int m=0;
int n;
int i=0;
void loop() {
//  lcd.clear();
  lcd.setCursor(1,1);
  lcd.print("no.of bottle ");
  char key = keypad.getKey();

  if (key){
    n=int(key)-48;
    lcd.clear();
    lcd.print("bottle ");
    lcd.print(n);
    //Serial.println(key);
    //Serial.print("\n");
    //Serial.print(n);
    p++;
  }
  if(n<5){
  if (p>=1){
  /*myStepper.step(-stepsPerRevolution);
  delay(5000);*/
  lcd.clear();
```

```
lcd.print("Initializing....");
buttonState = digitalRead(buttonPin);
if(buttonState==LOW)
{
      digitalWrite(11,LOW);
      int U;
      for(U=0;U<800;U++)
      {
         digitalWrite(9,HIGH);
         delayMicroseconds(pulse);
         digitalWrite(9,LOW);
         delayMicroseconds(pulse);
      }
      /*stepper(0,800);//// stepper(direction 1 or 0, steps number)
      delay(2000);*/
}
if (buttonState == HIGH) {
  //Serial.print("LED IS ON");
  digitalWrite(ledPin, HIGH);
  delay(1000);
  while(i<n)
  {
    //lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("no.of bottle ");
    lcd.print(i+1);
    m=m+1;
    if(m>0){
```

```
    digitalWrite(13,HIGH);
    Serial.print(m);
    delay(1500);
    digitalWrite(13,LOW);
    m=0;
    digitalWrite(11,HIGH);
    float x;
    //int A=3;
    //int b=50;
    for(x=0;x<800;x++){
    digitalWrite(9,HIGH);
    delayMicroseconds(pulse);
    digitalWrite(9,LOW);
    delayMicroseconds(pulse);
    /*stepper(1,800);//// stepper(direction 1 or 0, steps number)
    delay(2000);*/
    }
    delay(1000);
 }
 //Serial.print("no.of bottle ");
 //Serial.print(i);
 i++;
// delay(4000);
 if(i== n)
 {
   // goto End;
    lcd.clear();
    lcd.print("task complete");
```

```
        lcd.setCursor(1,1);
        lcd.print("thank you");
        delay(20000);
      }

   }
  }
  else {
                    // turn LED off:
   //Serial.print("LED IS OFF");
   digitalWrite(ledPin, LOW);
   delay(1000);
  }
  }
  }
  else {
   lcd.clear();
   lcd.print("INVALID");
  }
}
 /*End:
   lcd.clear();
   lcd.print("task complete");
   lcd.setCursor(1,1);
   lcd.print("thank you");
   delay(20000);*/
```

Description of construction:

I. Hexadecimal keypad pins are connected to the pin of Arduino 2-8.
II. Arduino Pin 9 and 11 are connected to Reset and step pin of A4988.
III. LCD is connected to the I2C bus bar. VCC of I2C is connected to voltage source and the rest is connected to ground.
IV. The stepper is connected to A4988 (1B, 1A, 2A, 2B).
V. Analog pin A3 and A4 of Arduino is connected with I2C bus bar.
VI. Here L293D pin 1 is connected with voltage source, pin 2 is connected to Arduino pin 13, pin 3 and 4 goes to the pump, pin 5 is connected with ground, pin 8, 16 are connected with voltage source.
VII. A capacitor is connected to A4988 VMOT pin and ground.
VIII. A4988 set and reset pins are shorted.

Working Principle:

I. The stepper motor is used to need speed controlling. The operation of this motor works on the principle that unlike poles attract each other and like pole repel each other. When the stator windings are excited with dc supply, it produces magnetic flux establishes the north and south pole. We have used the motor driver A4988 to control the stepper.
II. I2C is a serial protocol of two wire interface to connect the low speed device Arduino. The SCL line in I2C bus is the clock signal with synchronize the data transfer between the devices on the I2C bus. And it is controlled by master device. The two lines are open drained which means that pull up resistors needs to be attached to them. So that the lines are high because the devices on the I2c bus are active low.
III. L293 D is a motor driver IC which allows the dc motor to drive any direction. It works on the basic principle of H bridge.
IV. LCD is used to show the display how many bottles are filled up. LCD uses a liquid crystal to produce a visible image.
V. An Arduino UNO is used to control the whole set up.

Application:

The automatic bottle filling machine project is mainly industrialized based. Many industries use this project to fill the bottles with water or other liquid.

Reference:

https://github.com/Anurag1702002/Ovi