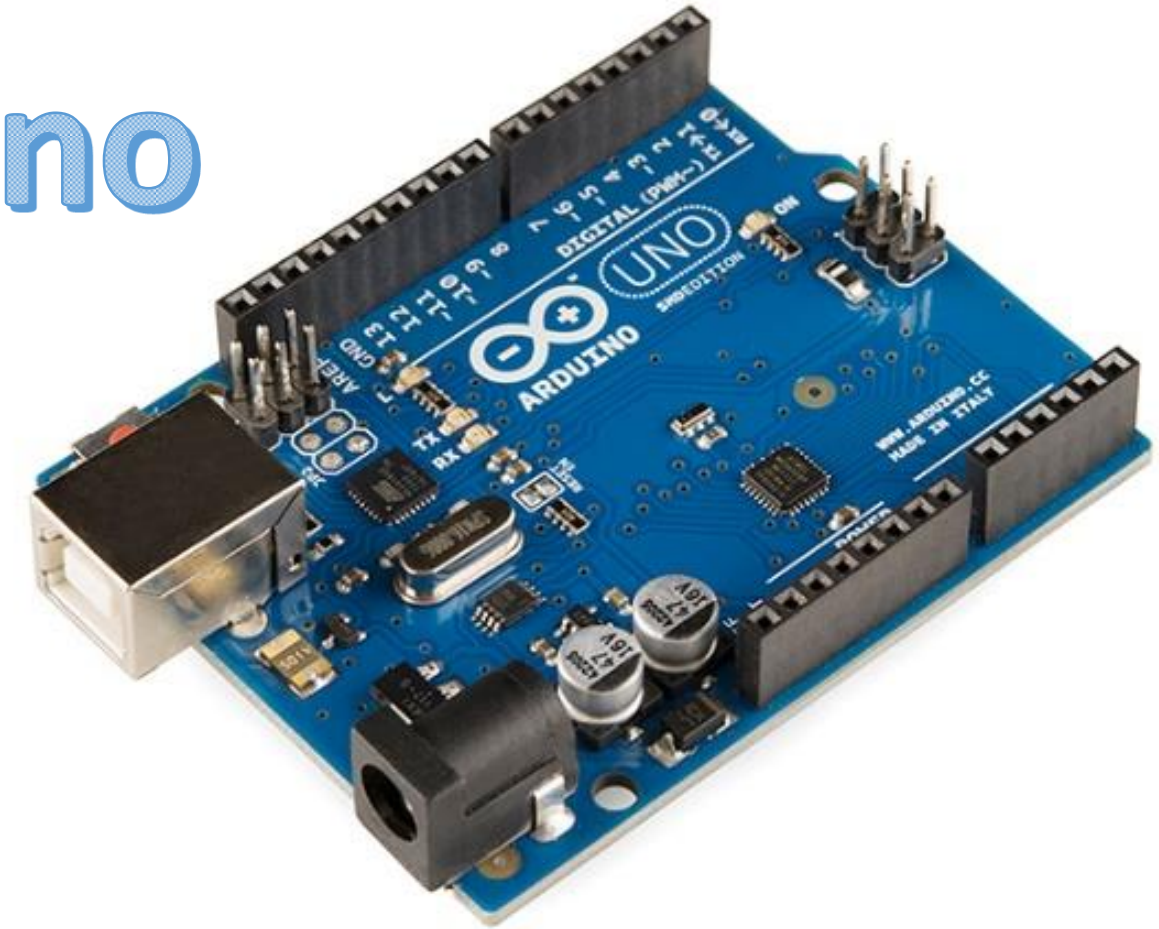
Project Arduino



What is Arduino?

- It's an open source electronics prototyping platform:
  - Open source: resources that can be used, redistributed or rewritten free of charge, often software or hardware.
  - Electronics: technology which makes use of the controlled motion of electrons through different media.
  - Prototyping: an original form that can serve as a basis or standard for other things.
  - Platform: hardware architecture with software framework on which other software can run.

Basic Arduino program- LED

```
int d1 = 1;
int d2 = 2;
int tipkalo = 3;
void setup() {
  // put your setup code here, to run once:
  pinMode(d1, OUTPUT);
  pinMode(d2, OUTPUT);
  pinMode(tipkalo, INPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
 int a = digitalRead(tipkalo);
  if (a == HIGH) {
    stisnuto();
  }
  else {
    otpusteno();
  }
}
```

```
void stisnuto() {
  digitalWrite(d1, HIGH);
  digitalWrite(d2, LOW);
  delay(1000);
  digitalWrite(d2, HIGH);
  digitalWrite(d1, LOW);
  delay(1000);
}
void otpusteno() {
  digitalWrite(d1, HIGH);
  digitalWrite(d2, HIGH);
  delay(50);
  digitalWrite(d2, LOW);
  digitalWrite(d1, LOW);
  delay(50);
}
```

Explanation of the code-declaring pins

```
int d1 = 1;
int d2 = 2;
int tipkalo = 3;
```

- We determine and declare pins which we will be using in our program

Input/Output

```
void setup() {
  // put your setup code here, to run once:
  pinMode(d1, OUTPUT);
  pinMode(d2, OUTPUT);
  pinMode(tipkalo, INPUT);
}
```

- In the void setup we determine our Inputs and Outputs
- We chose for d1 and d2 to be the outputs while we chose “tipkalo” to be the input

Void loop & IF-ELSE loop

```
void loop() {
  // put your main code here, to run repeatedly:
 int a = digitalRead(tipkalo);
  if (a == HIGH) {
    stisnuto();
  }
  else {
    otpusteno();
  }
}
```

- In the void loop we write the program which we want to run forever, in other words infinitely(in loops).
- In the code we are checking if the “tipkalo” is pressed or not.
- After that, we have a IF-ELSE loop with 2 subprograms, one for the case in which “tipkalo” is pressed and one for the case in which “tipkalo” is not pressed.

Subprogram

```
void stisnuto() {
  digitalWrite(d1, HIGH);
  digitalWrite(d2, LOW);
  delay(1000);
  digitalWrite(d2, HIGH);
  digitalWrite(d1, LOW);
  delay(1000);
}
```

- If the “tipkalo” is pressed the LEDs will alternately light up.
- The delay is that which makes the LED blink, it puts the LED in a state of doing nothing and hence the light turns off making the illusion of blinking. We chose the delay to be 1000 microseconds, in other words 1 second.

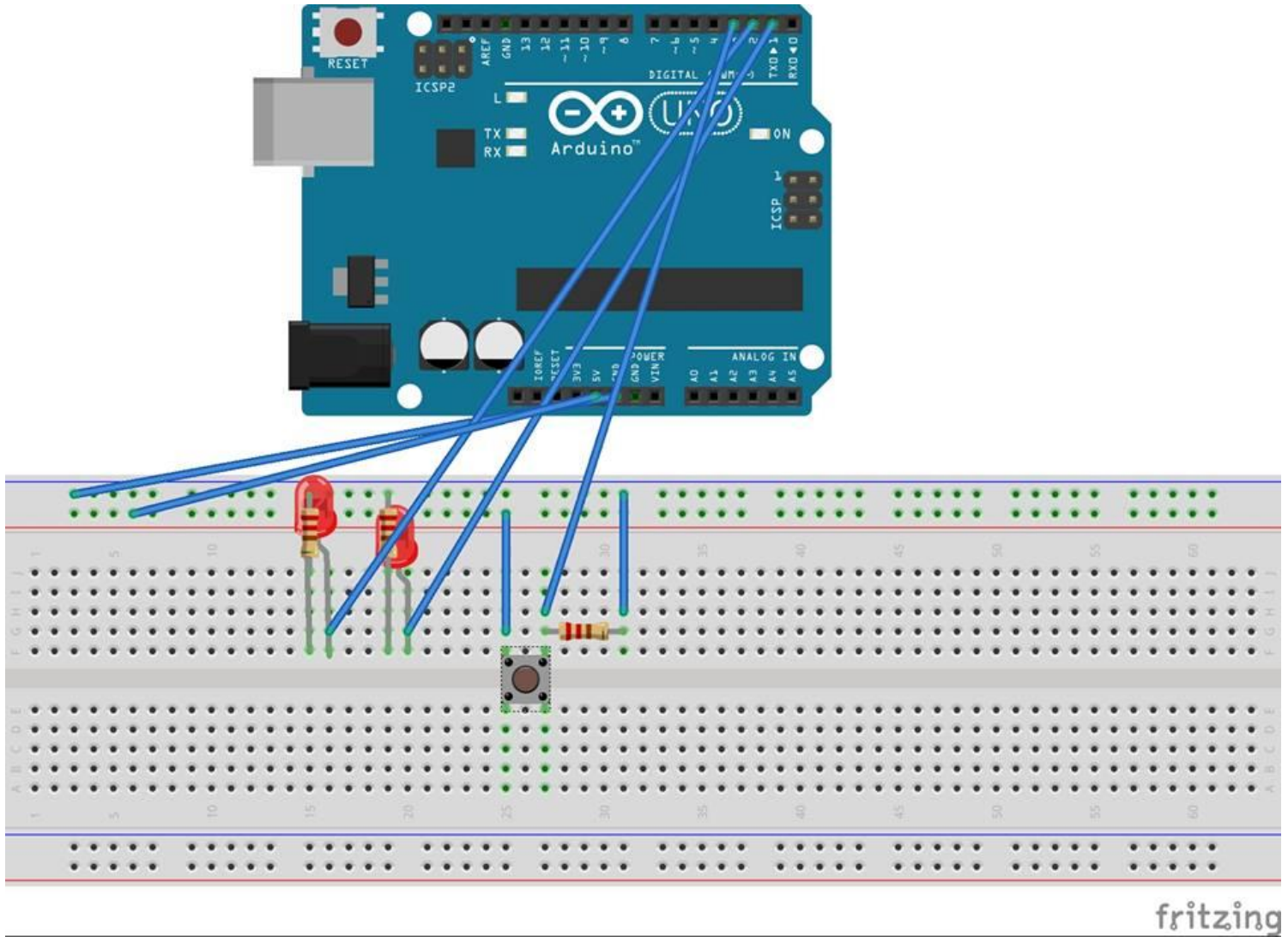
Subprogram

```
void otpusteno()
  digitalWrite(d1, HIGH);
  digitalWrite(d2, HIGH);
  delay(50);
  digitalWrite(d2, LOW);
  digitalWrite(d1, LOW);
  delay(50);
}
```

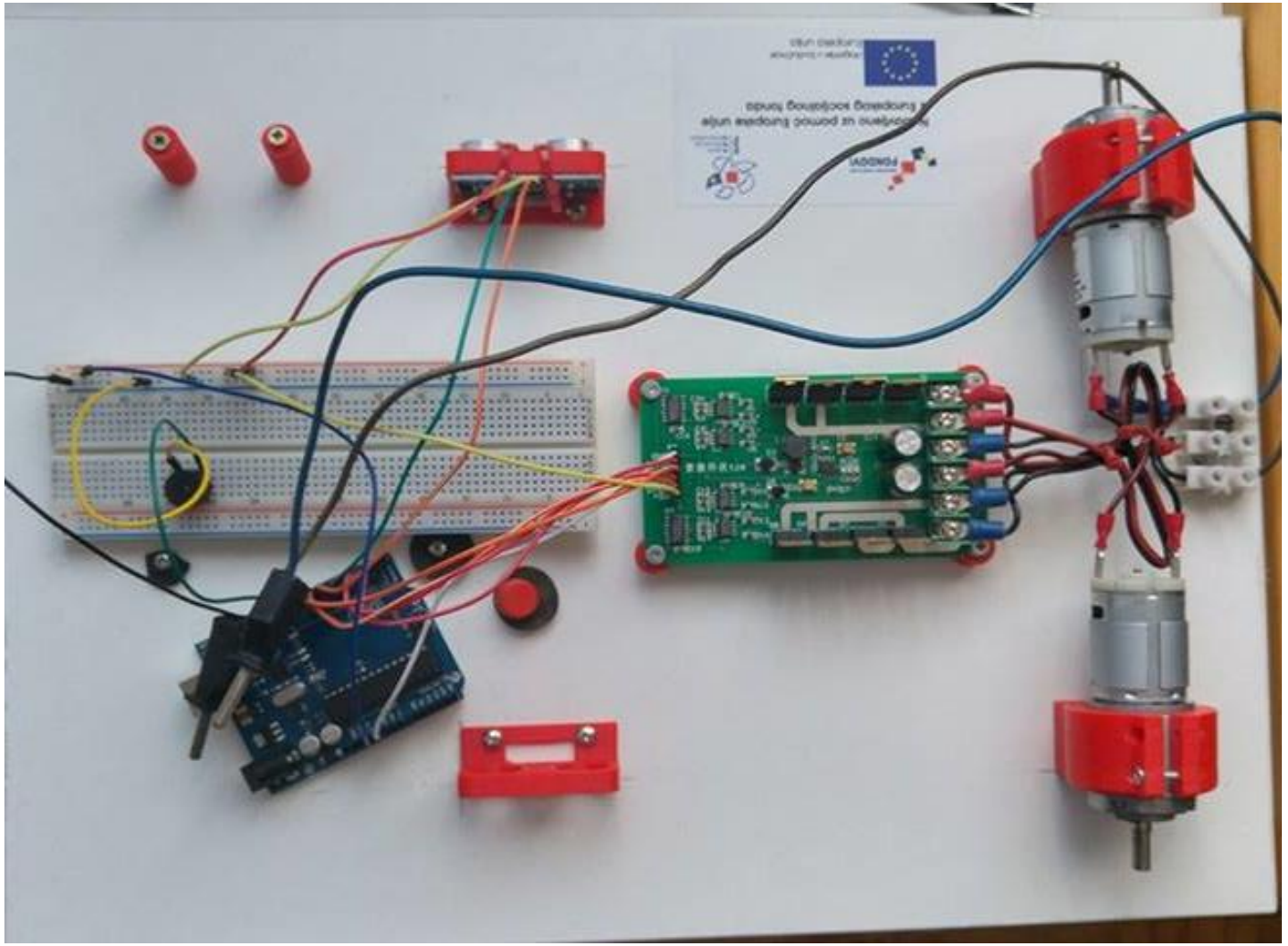
- If the “tipkalo” is not pressed both LEDs will blink at the same time
- They will blink in the span of 50ms.

Connecting example

- In the program “Fritzing”, you can simulate how the physical connecting of the Arduino components will look like.

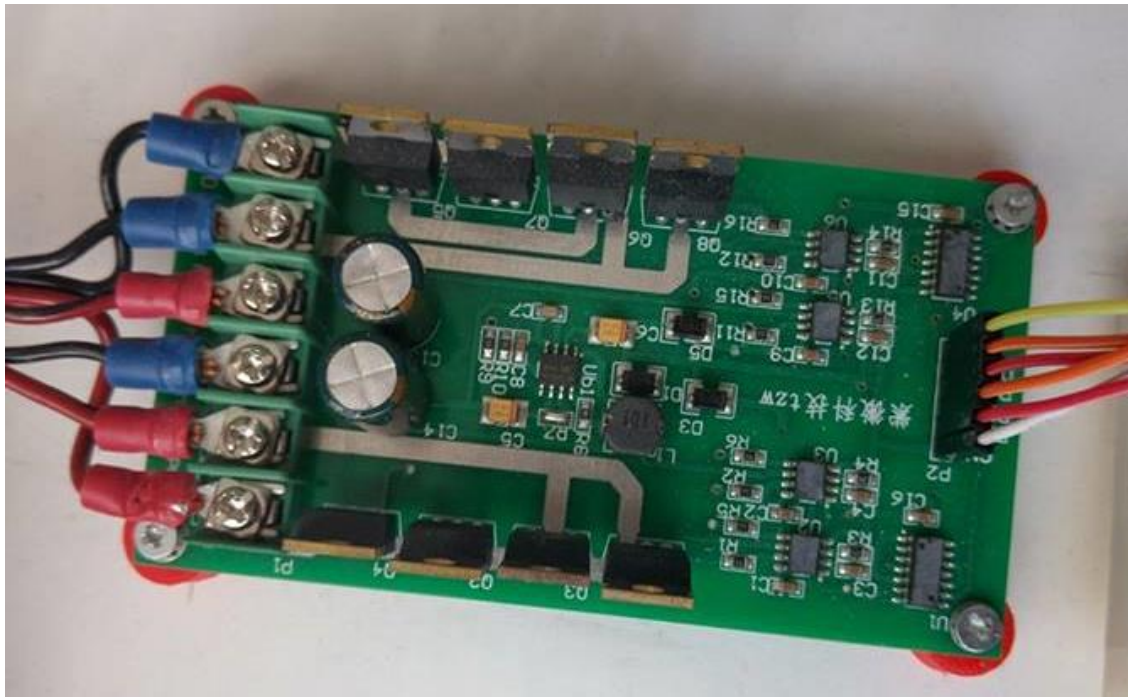


Project X (Motors controlled by sensors)



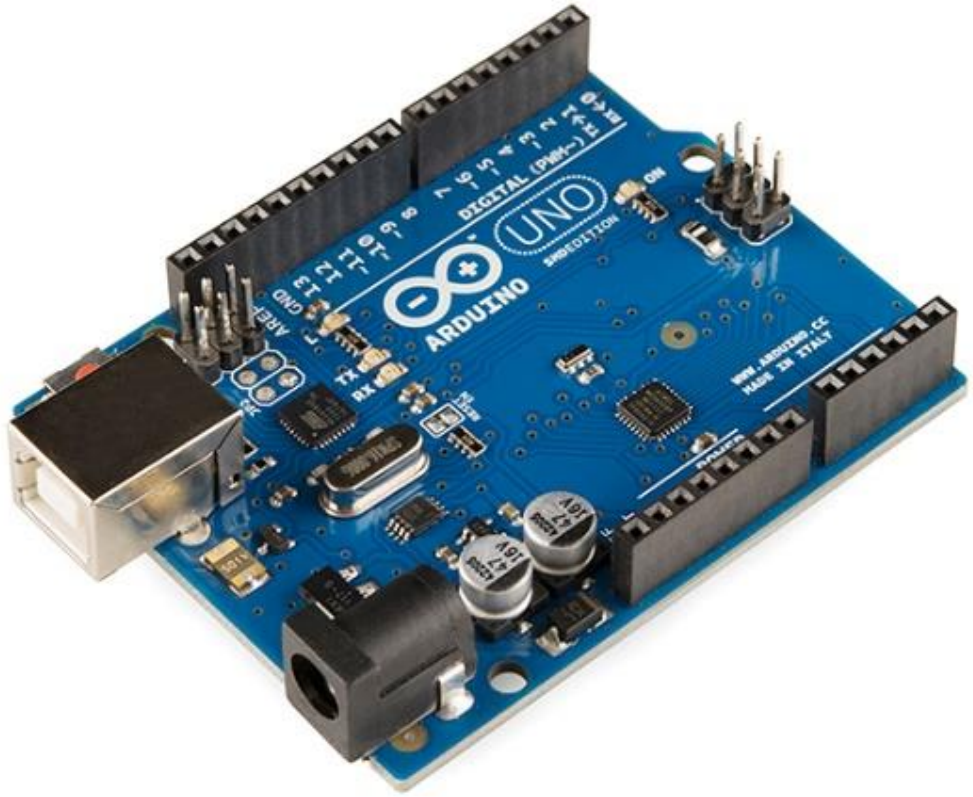
Made of:
1) H-bridge dual motor control
2)Arduino UNO board
3)Ultrasonic sensor
4)Electromotors
5)Breadbord

Parts - H-bridge dual motor control



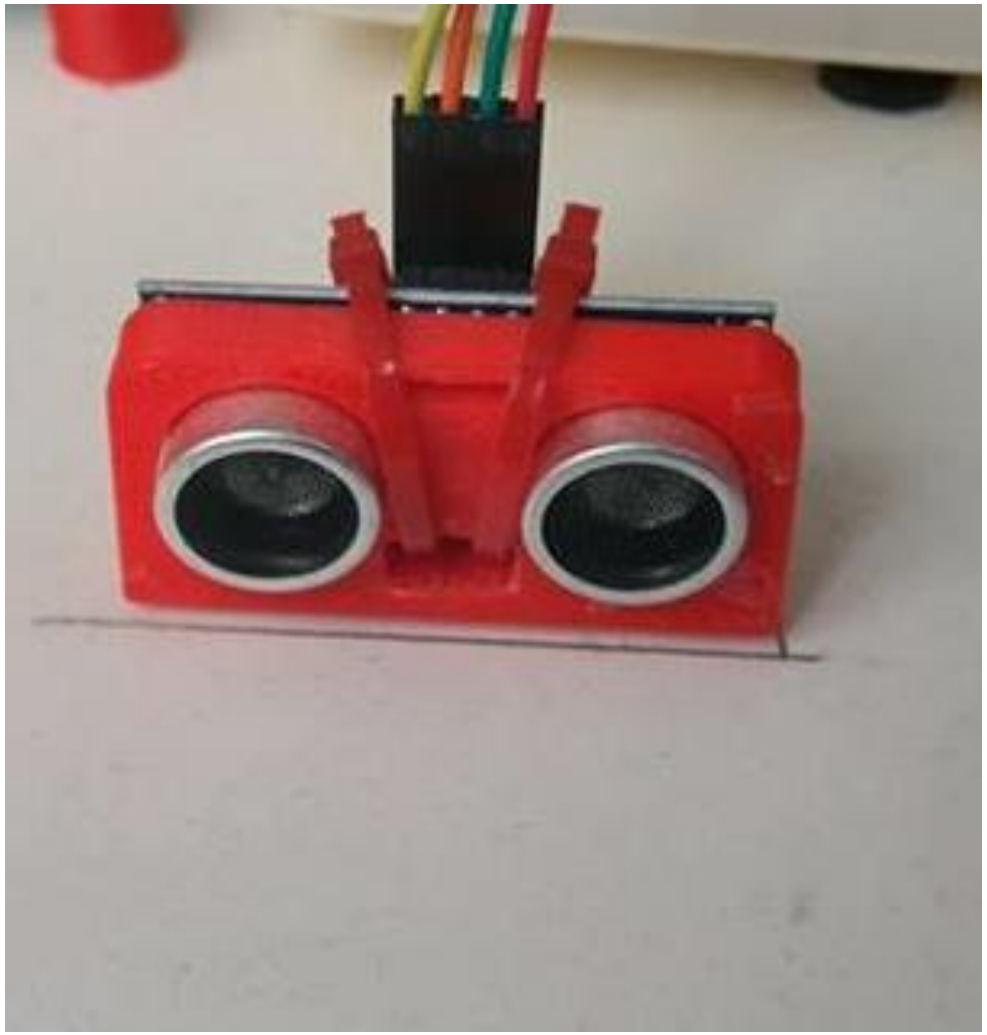
H - Bridge dual motor control serves as an aid in the process of connecting so as not to have to connect transistors , resistors and capacitors . (It replaces a large amout of breadboards)

Parts – Arduino UNO board



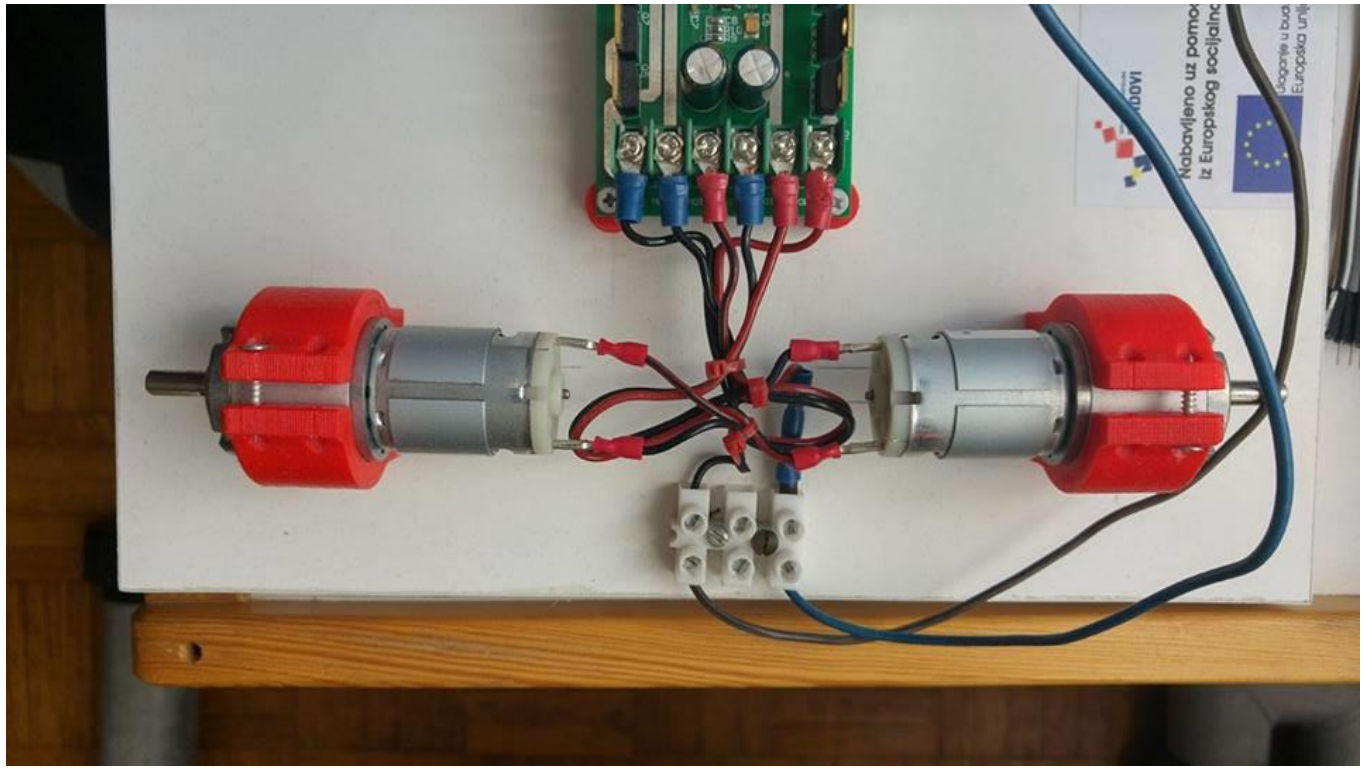
Arduino UNO is a microcontroller which connects a PC to the outside world.

Parts – Ultrasonic sensor



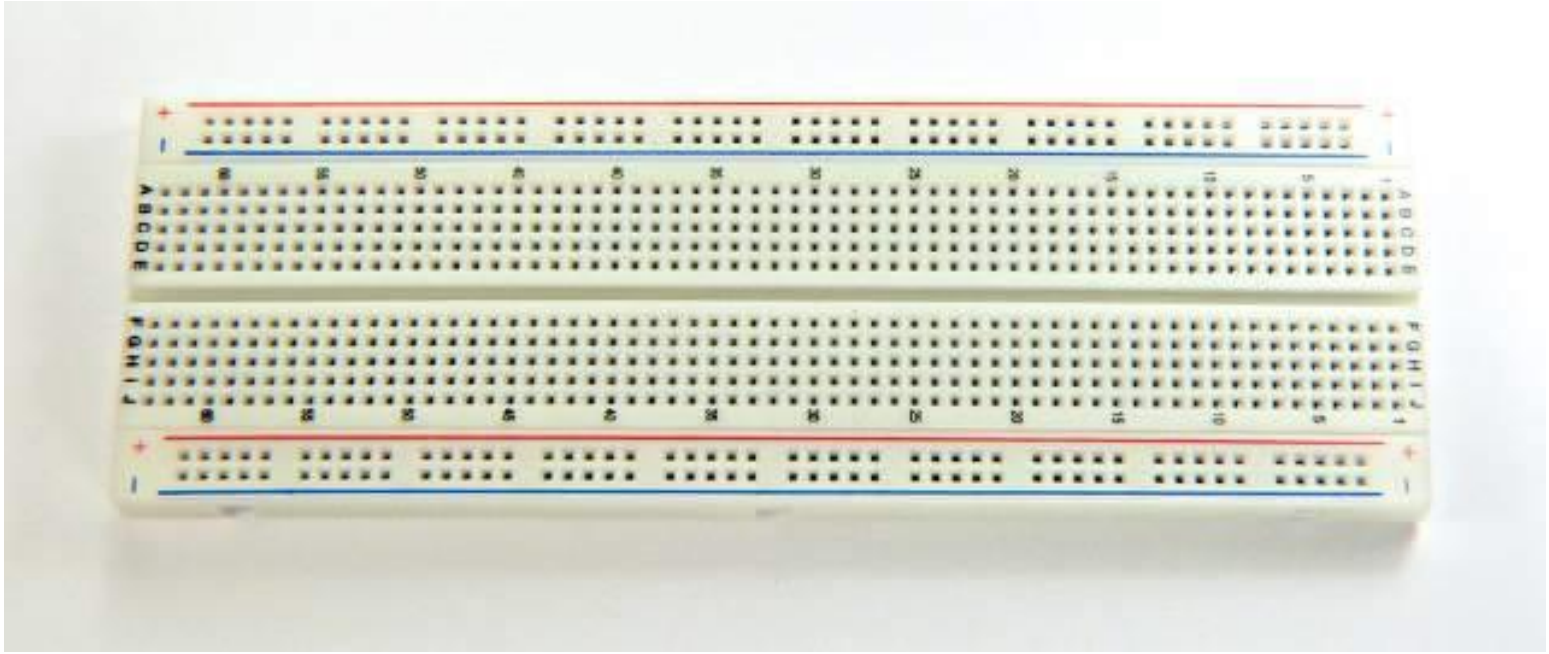
A ultrasonic sensors purpose is to detect obstacles and to control the direction of wheels if a obstacle is detected.

Parts - Electromotor



Electromotors serve for moving and changing the direction of wheels.

Parts - Breadboard



Breadboard is a extension which is used when more pins are needed.

Electromotor controlled by sensors

```
int pwm1 = 2;
int pwm2 = 3;
int dir1 = 8;
int dir2 = 9;
int trigger = 6;
int echo = 7;
int pistanje = 30;
int kocenje = 20;
int buzzer = 11;
void setup() {
  // put your setup code here, to run once:
  pinMode(pwm1, OUTPUT);
  pinMode(pwm2, OUTPUT);
  pinMode(dir1, OUTPUT);
  pinMode(dir2, OUTPUT);
  Serial.begin (9600);
  pinMode(trigger, OUTPUT);
  pinMode(echo, INPUT);
}
```

- We determine and declare the pins that will be used during the program
- Determine inputs and outputs.
- Trigger - ultrasonicsensor
- Echo - Beep

Voidloop

```
void loop() {
  // put your main code here, to run repeatedly:
  digitalWrite(trigger, LOW);
  delayMicroseconds(2);

  digitalWrite(trigger, HIGH);
  delayMicroseconds(10);

  digitalWrite(trigger, LOW);
  int duration = pulseIn(echo, HIGH);
  //Calculate the distance (in cm) based on the speed of sound.
  int distance = duration / 58.82;
  Serial.println(distance);
```

- The main code that will constantly repeat
- It serves to activate ultrasonic sensors and sound at a certain distance from obstacles.

IF ELSE

```
  if(distance>pistanje and distance>kocenje){
    naprijed();
  }
  else if(distance<=pistanje and distance>kocenje){
    ptc();
    naprijed();
  }
  else if(distance<=kocenje){
    stani();
    tisina();
  }
}
```

IF-ELSE loop determines the order of execution of seven sub-programs which are used to:
-Move forward
-Move backward
-Break
-Silence (Sound off)
-Turn left
-Turn right

Subprogram

```
void naprijed() {
  digitalWrite(dir1, HIGH);
  analogWrite(pwm1, 255);
  digitalWrite(dir2, LOW);
  analogWrite(pwm2, 255);
}
```

1.)The first subroutine is used to start the engine in the first (and only) gear

Subprogram

```
void stani() {
  digitalWrite(dir1, LOW);
  analogWrite(pwm1, 0);
  digitalWrite(dir2, HIGH);
  analogWrite(pwm2, 0);
}
```

2. subroutine is used to stop the engine

Subprogram

```
void stani() {
  digitalWrite(dir1, LOW);
  analogWrite(pwm1, 0);
  digitalWrite(dir2, HIGH);
  analogWrite(pwm2, 0);
```

3. Used for stopping the motor

Subprogram

```
void ptc() {
  tone(buzzer, 1000);
  delay(100);
  noTone(buzzer);
  delay(100);
}
void tisina(){
  noTone(buzzer);
}
```

4.) Used for repeated activation of sound

5.) For silence

Subprogram

```
void desno(){
  digitalWrite(dir1,HIGH);
  analogWrite(pwm1,255);
  digitalWrite(dir2,HIGH);
  analogWrite(pwm2,255);
}
void lijevo(){
  digitalWrite(dir1,LOW);
  analogWrite(pwm1,255);
  digitalWrite(dir2,LOW);
  analogWrite(pwm2,255);
}
```

6. & 7 . Subroutine used to turn to the left or to the right