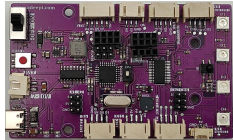




Lesson 2 How to Control the Servo

In this lesson, we will learn how to control the Servo.

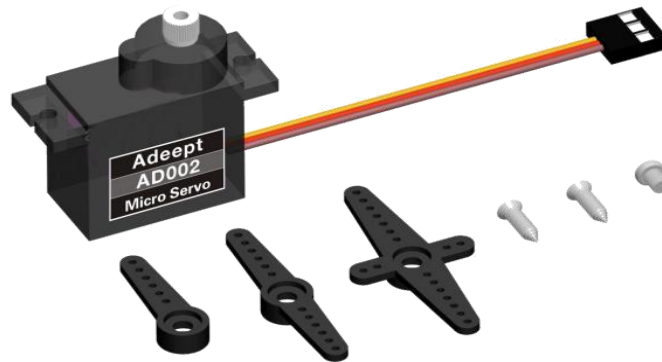
2.1 Components used in this course

Components	Quantity	Picture
Adeept Robot Control Board	1	
Type-C USB Cable	1	
Servo	1	

2.2 The introduction of the Servo

2.2.1 Servo

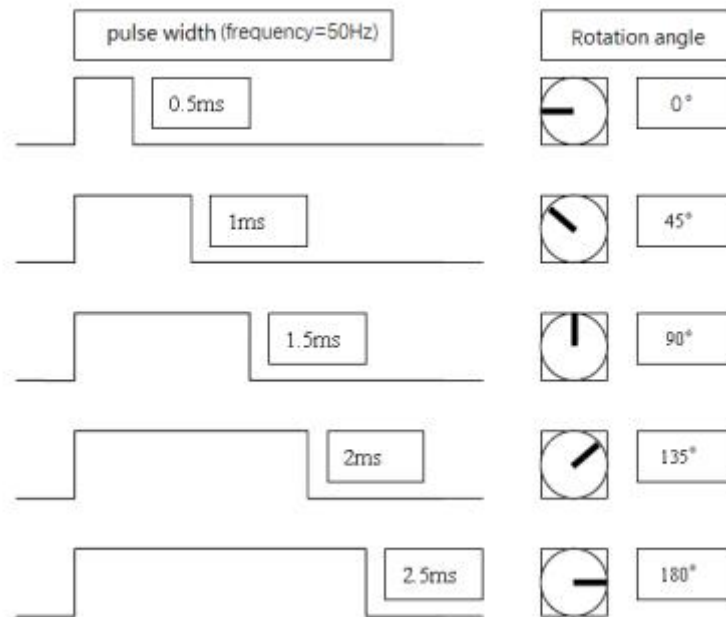
Servo motor refers to the engine that controls mechanical component operation in the servo system. It is a kind of auxiliary motor indirect transmission device. The servo motor is a gear motor that can rotate only 180 degrees. It is controlled by sending pulses from the microcontroller. These pulses tell the server where to move. The servo motor system includes housing, circuit board, non-core motor, gearing and position detection. Servo motor is shown in the figure:



2.2.2 The working principle of the Servo

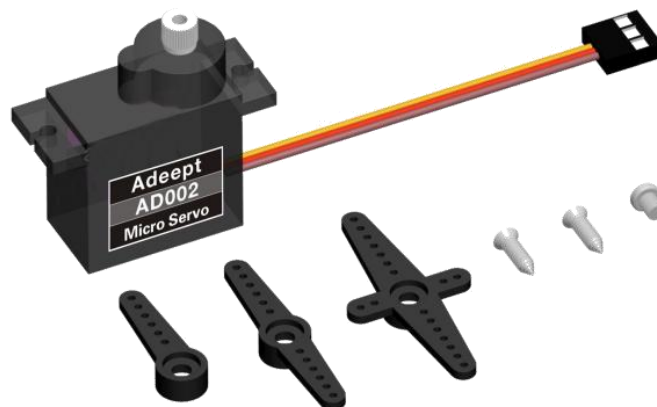
The servo mechanism is an automatic control system that enables the object's position, orientation, state and other output control quantities to follow arbitrary changes in the input target (or given value). The servo mainly depends on Pulsefor location. Basically, it can be understood that the servo motor receives an impulse and rotates the angle corresponding to the impulse to realize displacement. Because the servo motor itself has the function of sending out pulses, the servo motor rotates every time at an angle, and a corresponding number of pulses will be sent out. In this way, the pulses received by the servo motor form a response, or a closed loop. In this way, the system will know how many pulses are sent to the servo motor and how many pulses are received. In this way, it is possible to precisely control the rotation of the motor, thereby achieving precise positioning.

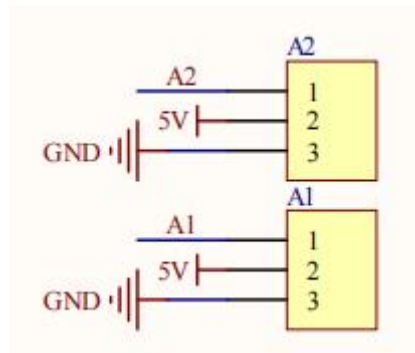
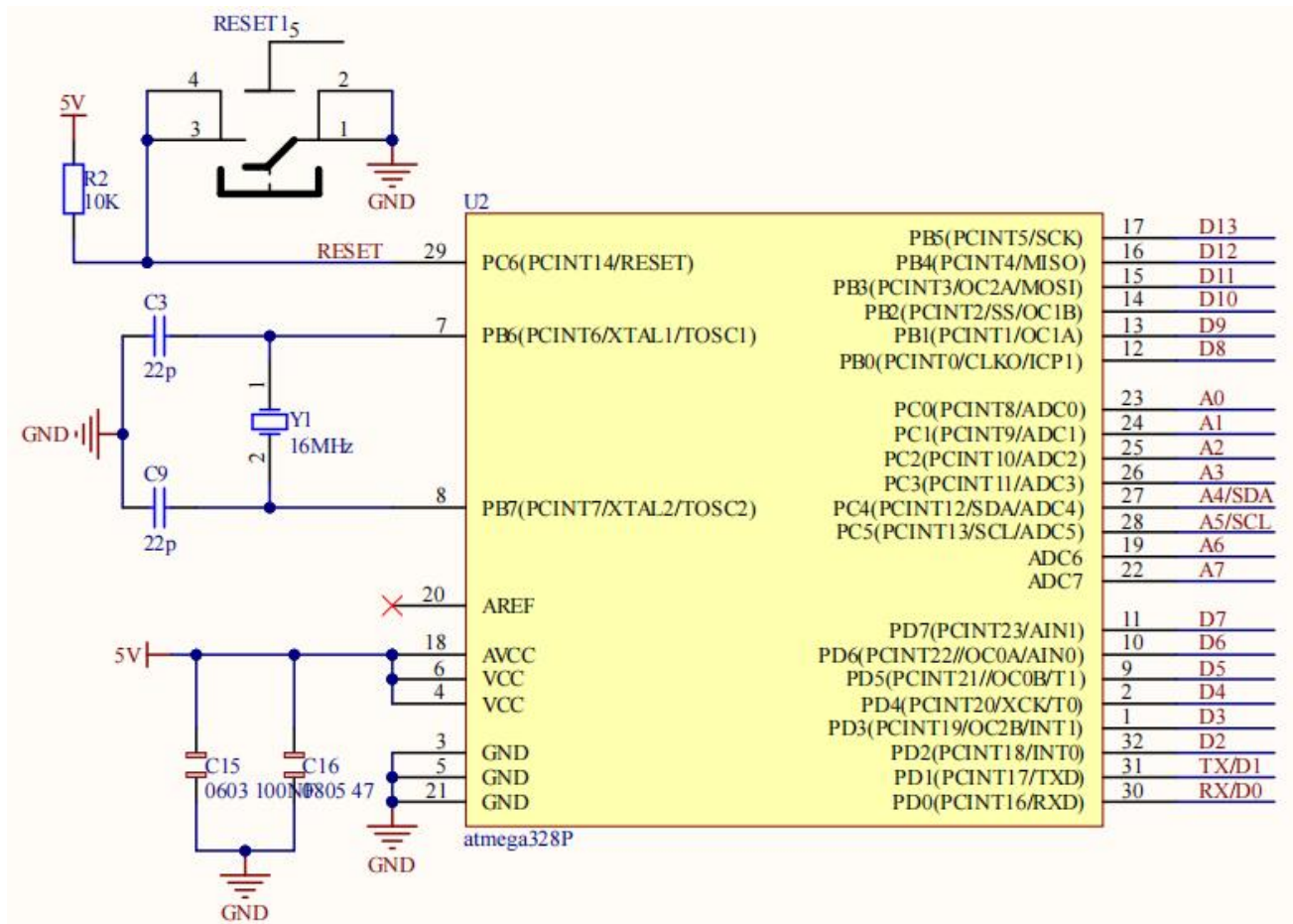
Adeept Robot Control Board sends a PWM signal to a servo motor, which is then processed by an IC on the circuit board to calculate the rotation direction of the drive motor, which is then transmitted through a reduction gear to the swing arm. At the same time, the position detector returns a position signal to determine whether the set position has been reached or not.



2.2.3 The principle of write () function

In the program, we use the write() function to control the rotation of the servo. For standard servos, the write() function will rotate the servo axis to the corresponding angular position. For the continuous rotation type of servo, the write() function can set the rotation speed of the servo (0 indicates that the servo rotates at full speed in one direction, 180 indicates that the servo rotates at full speed in another direction, and 90 indicates that the servo is stationary). The servo which is used this time is a standard servo.

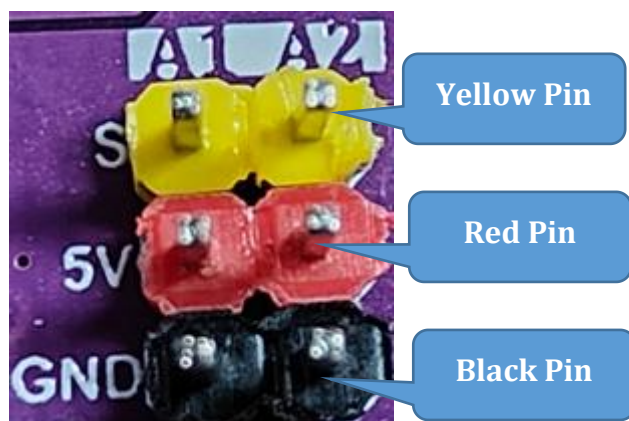
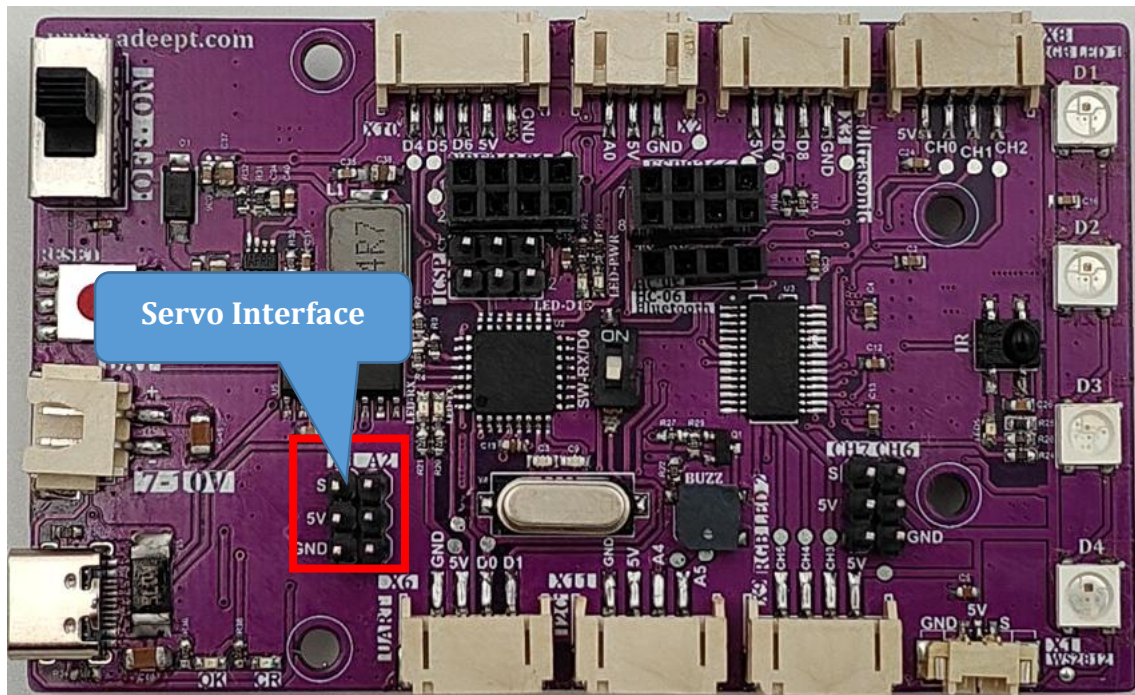




2.3 Wiring diagram

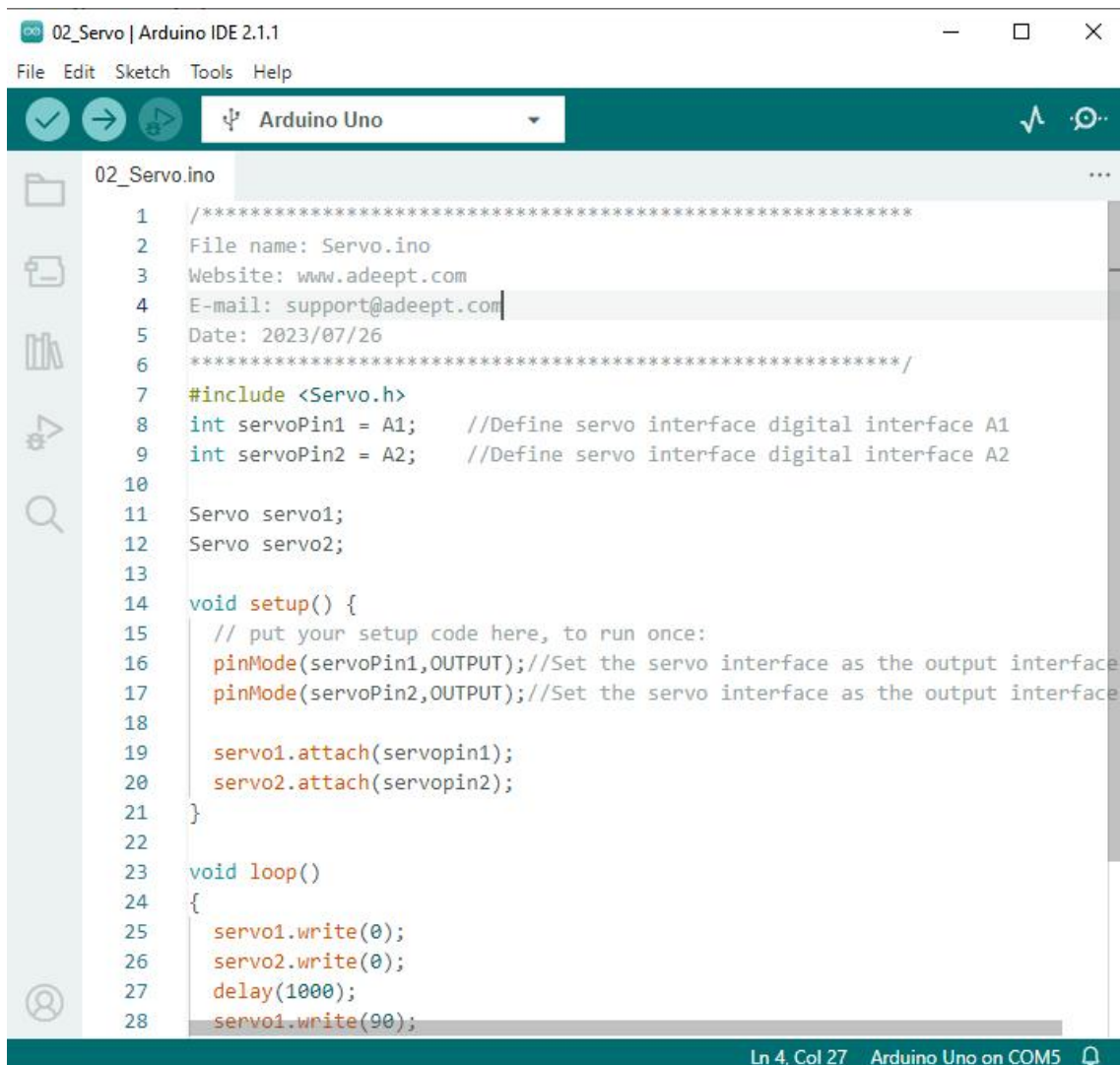
Connect the servo to the A1/A2 port. Note: The yellow servo wire is connected to the yellow pin, the red servo wire is connected to the red pin, and the black servo wire is connected to the black servo pin.

Figure as below:



2.4 How to control Servo

1. Connect your computer and Adeept Robot Control Board with a USB cable.
2. Open "02_Servo" folder in `/Code`, double-click `"02_Servo.ino"`.



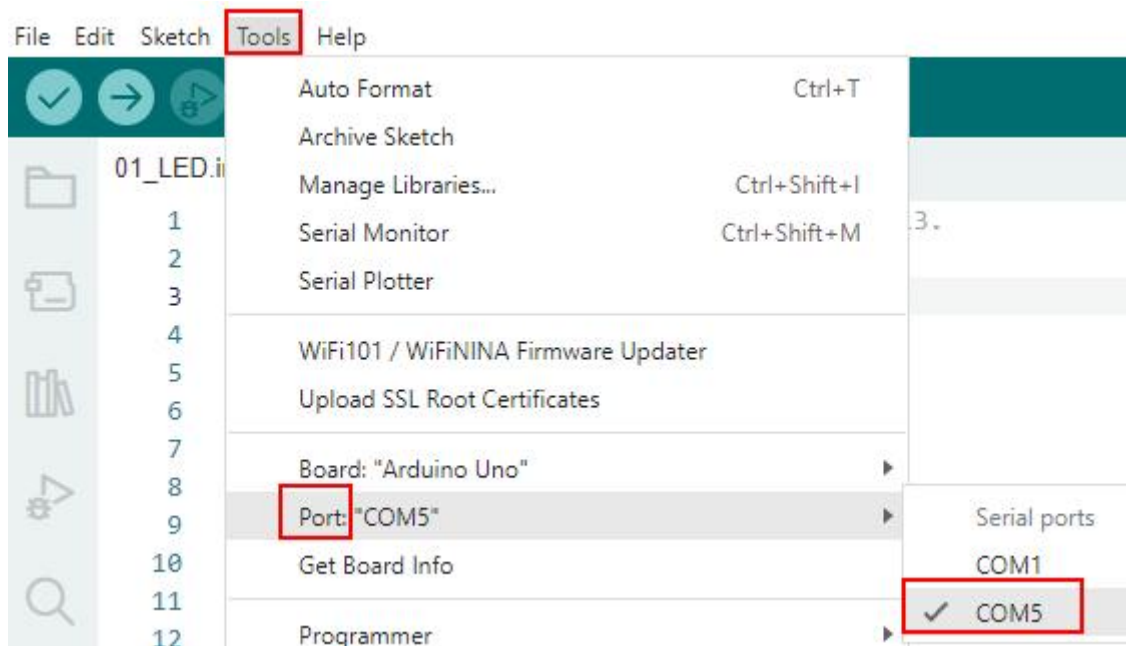
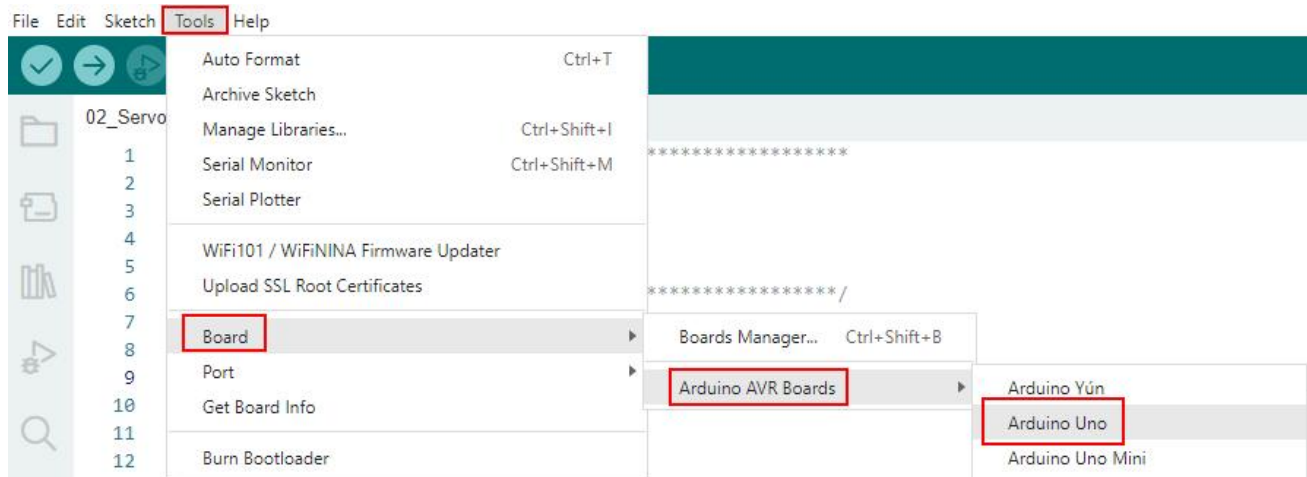
```
1  /*****
2  File name: Servo.ino
3  Website: www.adeept.com
4  E-mail: support@adeept.com
5  Date: 2023/07/26
6  *****/
7  #include <Servo.h>
8  int servoPin1 = A1;    //Define servo interface digital interface A1
9  int servoPin2 = A2;    //Define servo interface digital interface A2
10
11  Servo servo1;
12  Servo servo2;
13
14  void setup() {
15    // put your setup code here, to run once:
16    pinMode(servoPin1,OUTPUT);//Set the servo interface as the output interface
17    pinMode(servoPin2,OUTPUT);//Set the servo interface as the output interface
18
19    servo1.attach(servopin1);
20    servo2.attach(servopin2);
21  }
22
23  void loop()
24  {
25    servo1.write(0);
26    servo2.write(0);
27    delay(1000);
28    servo1.write(90);
```


3. Select development board and serial port.

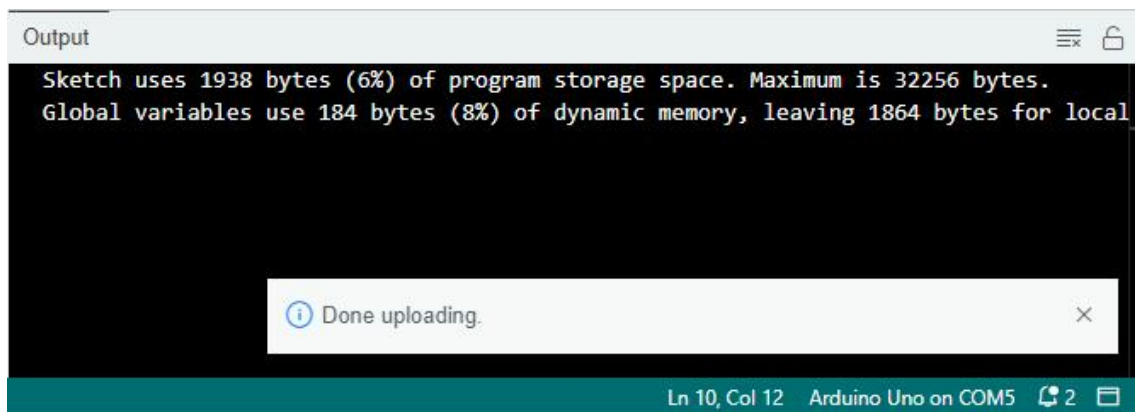
Board: Tools--->Board--->Arduino AVR Boards--->Arduino Uno

Port: Tools --->Port--->COMx

Note: The port number will be different in different computers.



4. After opening, click  to upload the code program to the Arduino. If there is no error warning in the console below, it means that the Upload is successful.



5. After successfully running the program, the servo will turn back and forth.

2.5 Code

```
1. #include <Servo.h>
2. int servoPin1 = A1;    //Define servo interface digital interface A1
3. Servo servo1;
4. void setup() {
5.     // put your setup code here, to run once:
6.     pinMode(servoPin1,OUTPUT); //Set the servo interface as the output interface
7.     servo1.attach(servoPin1);
8. }
9.
10. void loop()
11. {
12.     servo1.write(0); // Set servo 1 to turn to 0 degrees.
13.     delay(1000);     // delay 1s.
14.     servo1.write(90); // 90 degrees.
15.     delay(1000);
16.     servo1.write(180); // 180 degrees.
17.     delay(1000);
18. }
```