

**Program no**\_\_\_\_13\_\_\_\_

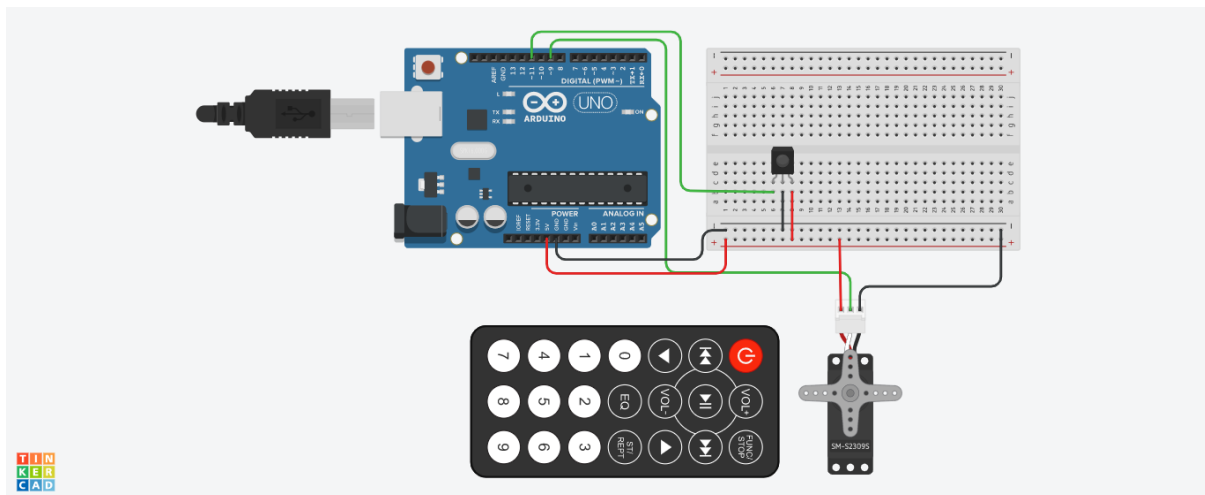
**Program Title**\_\_\_\_\_ **SERVO motor controller**\_\_\_\_\_

**Aim to rotate the SERVO motor clockwise and counter clockwise.**

### Hardware Required

- Arduino Board
- IR sensor
- Micro SERVO motor
- IR Remote
- wires

### Circuit Diagram



### Code

Program  
SERVO motor Controller.

Code:

```
#include < Servo.h>
#include < IRremote.h>
int RECV_PIN = 11;
IRrecv irrecv(RECV_PIN);
decode_results results;

Servo myServo;

void setup()
{
    serial.begin(9600);
    irrecv.enableIRIn();
}

void loop()
{
    if (irrecv.decode(&results))
    {
        switch (results.value)
        {
            case 0xFD00FF:
                myServo.attach(9);
                serial.println("Start");
                break;
            case 0xFD609F:
                myServo.attach(360);
                serial.println("Clockwise");
                break;
            case 0xFD20BF:
```

```
myservo.write(-360);  
Serial.println("Counter clockwise");  
break;  
default:  
Serial.print("unrecognized code received: 0x");  
Serial.println(result.value, HEX);  
break;  
}  
irrecv.resume();  
}  
}
```

## Observation /Output

1BM19CS400 - SERVO motor controller

Simulator time: 00:00:02.018

Code Stop Simulation Export Share

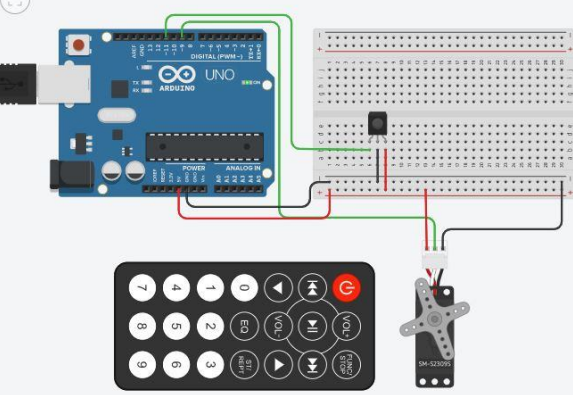
1 (Arduino Uno R3)

```
1 #include <Servo.h>
2 #include <IRremote.h>
3
4 int RECV_PIN = 11;
5 IRrecv irrecv(RECV_PIN);
6 decode_results results;
7
8 Servo myservo;
9
10 void setup() {
11   Serial.begin(9600);
12   irrecv.enableIRIn();
13 }
14
15 void loop() {
16   if (irrecv.decode(&results)) {
17     Serial.println(results.value);
18     switch (results.value) {
19       case 0: myservo.write(0); break;
20       case 1: myservo.write(90); break;
21       case 2: myservo.write(180); break;
22       case 3: myservo.write(270); break;
23       case 4: myservo.write(360); break;
24     }
25   }
26 }
```

Serial Monitor

Start  
Clockwise  
Counter Clockwise  
Clockwise

Send Clear



The screenshot displays a digital circuit simulation in a software environment. On the left, an Arduino Uno R3 is connected to a breadboard. A black IR receiver module is plugged into the breadboard, with its signal pin connected to digital pin 11 of the Arduino. A grey servo motor is also connected to the breadboard, with its control pin connected to digital pin 10. A black keypad with various buttons is connected to the breadboard. The right side of the interface shows the Arduino IDE code editor with a C++ program that uses the Servo and IRremote libraries. The code sets up a serial connection at 9600 baud and listens for IR remote commands. The loop function decodes these commands and sends the corresponding servo position values (0, 90, 180, 270, 360 degrees) to the serial monitor. The serial monitor shows the output: 'Start', 'Clockwise', 'Counter Clockwise', and 'Clockwise'. The top of the window shows the title '1BM19CS400 - SERVO motor controller' and the simulator time '00:00:02.018'. The bottom of the window has buttons for 'Code', 'Stop Simulation', 'Export', and 'Share'.