

```

#include <IRremote.h>
// includes the library for the IR Remote

#include <Servo.h>
// includes the library for the Servos

#define Ready 13
#define userWin 12
#define rpsWin 11
// these are the LEDs used for the winners

Servo Index;
Servo Middle;
Servo Ring;
Servo Pinky;
// all servos are identified for each finger

int RECV_PIN = 10;
// the IR sensor is attached to pin 10

IRrecv irrecv(RECV_PIN);

decode_results results;

int indexPos = 0;
int middlePos = 0;
int ringPos = 180;
int pinkyPos = 180;
//these are the starting positions when the hand is turned on (paper)
int command;
int rpsPlay = 1;
// this is the variable that will randomize the robot's play against the user
int userPlay = 0;
// this is the user's play against the robot

void setup(){
  Serial.begin(9600);
  irrecv.enableIRIn();
  // Start the receiver

  Index.attach(3);
  Middle.attach(4);
  Ring.attach(5);
  Pinky.attach(7);

```

```

    // these attach the servos to their respective pins

    pinMode(Ready, OUTPUT);
    digitalWrite(Ready, LOW);
    // LED indicating the game is ready to be played
    pinMode(userWin, OUTPUT);
    digitalWrite(userWin, LOW);
    // LED indicating that the user has won
    pinMode(rpsWin, OUTPUT);
    digitalWrite(rpsWin, LOW);
    // LED indicating that the RPS has won
}

void loop() {
    if (irrecv.decode(&results)) {
        // this decodes the IR input and are used for each command
        //Serial.println(results.value, HEX);
        //displays command codes for each button on the remote
        if (results.value == 0xFF30CF){
            //1 on remote (ROCK)
            userPlay = 1;
            digitalWrite(Ready, HIGH);
        }
        if (results.value == 0xFF18E7){
            //2 on remote (PAPER)
            userPlay = 2;
            digitalWrite(Ready, HIGH);
        }
        if (results.value == 0xFF7A85){
            //3 on remote (SCISSORS)
            userPlay = 3;
            digitalWrite(Ready, HIGH);
        }

        if (results.value == 0xFFC23D){
            //play/pause on remote
            digitalWrite(Ready, LOW);
            indexPos = 180;
            middlePos = 180;
            ringPos = 0;
            pinkyPos = 0;
            Index.write(indexPos);
            Middle.write(middlePos);
            Ring.write(ringPos);
        }
    }
}

```

```
Pinky.write(pinkyPos);  
delay(1000);
```

```
indexPos = 0;  
middlePos = 0;  
ringPos = 180;  
pinkyPos = 180;  
Index.write(indexPos);  
Middle.write(middlePos);  
Ring.write(ringPos);  
Pinky.write(pinkyPos);  
delay(1000);
```

```
indexPos = 0;  
middlePos = 0;  
ringPos = 0;  
pinkyPos = 0;  
Index.write(indexPos);  
Middle.write(middlePos);  
Ring.write(ringPos);  
Pinky.write(pinkyPos);  
delay(1000);  
// this will cycle the hand into rock, then paper, then scissors
```

```
command = random(1,4);  
if (command == 1){ //ROCK  
  indexPos = 180;  
  middlePos = 180;  
  ringPos = 0;  
  pinkyPos = 0;  
  rpsPlay = 1;  
  Index.write(indexPos);  
  Middle.write(middlePos);  
  Ring.write(ringPos);  
  Pinky.write(pinkyPos);  
}  
if (command == 2){ //PAPER  
  indexPos = 0;  
  middlePos = 0;  
  ringPos = 180;  
  pinkyPos = 180;  
  rpsPlay = 2;  
  Index.write(indexPos);  
  Middle.write(middlePos);
```

```

    Ring.write(ringPos);
    Pinky.write(pinkyPos);
}
if (command == 3){ //SCISSORS
    indexPos = 0;
    middlePos = 0;
    ringPos = 0;
    pinkyPos = 0;
    rpsPlay = 3;
    Index.write(indexPos);
    Middle.write(middlePos);
    Ring.write(ringPos);
    Pinky.write(pinkyPos);
    // after the cycle, the hand will throw up one of the three options at random
}
Serial.println(command);
Serial.println(userPlay);
Serial.println(rpsPlay);
    // serial print is helpful for debugging so you can see what moves are being played
}

if (results.value == 0xFFC23D){ //play/pause on remote
    if (((userPlay==1) && (rpsPlay==3)) | ((userPlay==2) && (rpsPlay==1)) | ((userPlay==3) &&
(rpsPlay==2))){
        digitalWrite(userWin, HIGH);
        delay(3000);
        digitalWrite(userWin, LOW);
        // this indicates a user win against the robot with the green LED lighting up
    }
    if (((userPlay==1) && (rpsPlay==2)) | ((userPlay==2) && (rpsPlay==3)) | ((userPlay==3) &&
(rpsPlay==1))){
        digitalWrite(rpsWin, HIGH);
        delay(3000);
        digitalWrite(rpsWin, LOW);
        // this indicates a robot win against the user with the red LED lighting up
    }
    if (userPlay==rpsPlay){
        digitalWrite(userWin, HIGH);
        digitalWrite(rpsWin, HIGH);
        delay(3000);
        digitalWrite(userWin, LOW);
        digitalWrite(rpsWin, LOW);
        // this indicates a draw between the user and the robot with both the green and red LEDs
        lighting up
    }
}

```

```
}  
userPlay = 0;  
    // this resets the user play to 0 so the game is ready to be played again  
}  
  
irrecv.resume();  
    // Receive the next value  
}  
}
```