

```
#include <Servo.h>
const int IRPin = 2; // Replace with your IR receiver pin
const unsigned long startSignalMinDuration = 4000; // Minimum duration of start
signal in microseconds
const unsigned long startSignalMaxDuration = 5000; // Maximum duration of start
signal in microseconds
const unsigned long continuousSignalInterval = 115000; // Interval between
continuous signals in microseconds
const unsigned long continuousSignalMinDuration = 2000; // Minimum duration of
continuous signal in microseconds
const unsigned long continuousSignalMaxDuration = 3000; // Maximum duration of
continuous signal in microseconds
unsigned long lastSignalTime = 0; // Time of the last detected signal
int lastButtonPressed = 0; // ID of the last button pressed
int servoSelected = 0; // ID of servo selected

// Define servo objects for 6 servos
Servo servo1;
Servo servo2;
Servo servo3;
Servo servo4;
Servo servo5;
Servo servo6;

void setup() {
  pinMode(IRPin, INPUT);
  Serial.begin(9600);

  // Attach the servo objects to the corresponding pins
  servo1.attach(8);
  servo2.attach(9);
  servo3.attach(10);
  servo4.attach(11);
  servo5.attach(12);
  servo6.attach(13);

  // Initialize the servos to their initial positions
  servo1.write(60); //60
  servo2.write(175); //175
  servo3.write(145); //145
  servo4.write(5); //5
  servo5.write(45); //45
  servo6.write(65); //65
}
```

```
void loop() {
  if (detectStartSignal()) {
    lastSignalTime = micros();
    String bitPattern = readBitPattern();

    if (bitPattern == "00000000111111111010001001011101") {
      Serial.println("Button 1 Pressed");
      servoSelected = 1;
      lastButtonPressed = 1; //1 claw
    } else if (bitPattern == "000000001111111110110001010011101") {
      Serial.println("Button 2 Pressed");
      servoSelected = 2;
      lastButtonPressed = 2; //2 wrist up
    } else if (bitPattern == "0000000011111111110001000011101") {
      Serial.println("Button 3 Pressed");
      servoSelected = 3;
      lastButtonPressed = 3; //3 wrist
    } else if (bitPattern == "000000001111111110010001011011101") {
      Serial.println("Button 4 Pressed");
      servoSelected = 4;
      lastButtonPressed = 4; //4 elbow
    } else if (bitPattern == "000000001111111110000001011111101") {
      Serial.println("Button 5 Pressed");
      servoSelected = 5;
      lastButtonPressed = 5; //5 shoulder
    } else if (bitPattern == "00000000111111111100001000111101") {
      Serial.println("Button 6 Pressed");
      servoSelected = 6;
      lastButtonPressed = 6; //6 base
    } else if (bitPattern == "000000001111111111000000011111") {
      Serial.println("Button 7 Pressed");
      lastButtonPressed = 7; //7
    } else if (bitPattern == "00000000111111111010100001010111") {
      Serial.println("Button 8 Pressed");
      lastButtonPressed = 8; //8
    } else if (bitPattern == "00000000111111111001000001101111") {
      Serial.println("Button 9 Pressed");
      lastButtonPressed = 9; //9
    } else if (bitPattern == "000000001111111110110100010010111") {
      Serial.println("Button 10 Pressed");
    }
  }
}
```

```

    lastButtonPressed = 10;                               /*
} else if (bitPattern == "0000000011111111001100001100111") {
    Serial.println("Button 0 Pressed");
    lastButtonPressed = 0;                               //0
} else if (bitPattern == "0000000011111111011000001001111") {
    Serial.println("Button 11 Pressed");
    lastButtonPressed = 11;                             //#
} else if (bitPattern == "00000000111111110001100011100111") {
    Serial.println("Button 12 Pressed");
    lastButtonPressed = 12;                             //up control
forward/up/left
} else if (bitPattern == "00000000111111110001000011101111") {
    Serial.println("Button 13 Pressed");
    lastButtonPressed = 13;                             //left
} else if (bitPattern == "00000000111111110011100011000111") {
    Serial.println("Button 14 Pressed");
    lastButtonPressed = 14;                             //ok
} else if (bitPattern == "00000000111111110101101010100101") {
    Serial.println("Button 15 Pressed");
    lastButtonPressed = 15;                             //right
} else if (bitPattern == "00000000111111110100101010110101") {
    Serial.println("Button 16 Pressed");
    lastButtonPressed = 16;                             //down
control back/down/right
}

// Check for continuous signal
while (micros() - lastSignalTime < continuousSignalInterval) {
    if (detectContinuousSignal()) {
        Serial.print("Button ");
        Serial.print(lastButtonPressed);
        Serial.println(" still pressed");
        lastSignalTime = micros(); // Update last signal time
    }

// Change Position
if (servoSelected == 1) {
    // Perform actions based on button 1
    switch (lastButtonPressed) {
        case 12:
            // Increase servo1 position
            if (lastButtonPressed == 12) {
                // You can add logic to control the servo position here
                int newPosition = servo1.read() + 1; // Example: Increase by 1
degree

```

```
        servo1.write(newPosition);
    }
    break;
    case 16:
        // Reduce servo1 position
        if (lastButtonPressed == 16) {
            // You can add logic to control the servo position here
            int newPosition = servo1.read() - 1; // Example: Decrease by 1
degree
            servo1.write(newPosition);
        }
        break;
    }
}
if (servoSelected == 2) {
    // Perform actions based on button 2
    switch (lastButtonPressed) {
        case 12:
            // Increase servo2 position
            if (lastButtonPressed == 12) {
                // You can add logic to control the servo position here
                int newPosition = servo2.read() + 1; // Example: Increase by 1
degree
                servo2.write(newPosition);
            }
            break;
        case 16:
            // Reduce servo2 position
            if (lastButtonPressed == 16) {
                // You can add logic to control the servo position here
                int newPosition = servo2.read() - 1; // Example: Decrease by 1
degree
                servo2.write(newPosition);
            }
            break;
        }
    }
}
if (servoSelected == 3) {
    // Perform actions based on button 3
    switch (lastButtonPressed) {
        case 12:
            // Increase servo3 position
            if (lastButtonPressed == 12) {
                // You can add logic to control the servo position here
```

```
        int newPosition = servo3.read() + 1; // Example: Increase by 1
degree
        servo3.write(newPosition);
    }
    break;
    case 16:
        // Reduce servo3 position
        if (lastButtonPressed == 16) {
            // You can add logic to control the servo position here
            int newPosition = servo3.read() - 1; // Example: Decrease by 1
degree
            servo3.write(newPosition);
        }
        break;
    }
}
if (servoSelected == 4) {
    // Perform actions based on button 4
    switch (lastButtonPressed) {
        case 12:
            // Increase servo4 position
            if (lastButtonPressed == 12) {
                // You can add logic to control the servo position here
                int newPosition = servo4.read() + 1; // Example: Increase by 1
degree
                servo4.write(newPosition);
            }
            break;
        case 16:
            // Reduce servo4 position
            if (lastButtonPressed == 16) {
                // You can add logic to control the servo position here
                int newPosition = servo4.read() - 1; // Example: Decrease by 1
degree
                servo4.write(newPosition);
            }
            break;
        }
    }
}
if (servoSelected == 5) {
    // Perform actions based on button 1
    switch (lastButtonPressed) {
        case 12:
            // Increase servo5 position
            if (lastButtonPressed == 12) {
```

```
    // You can add logic to control the servo position here
    int newPosition = servo5.read() + 1; // Example: Increase by 1
degree
    servo5.write(newPosition);
}
break;
case 16:
    // Reduce servo5 position
    if (lastButtonPressed == 16) {
        // You can add logic to control the servo position here
        int newPosition = servo5.read() - 1; // Example: Decrease by 1
degree
        servo5.write(newPosition);
    }
    break;
}
}
if (servoSelected == 6) {
    // Perform actions based on button 6
    switch (lastButtonPressed) {
        case 12:
            // Increase servo6 position
            if (lastButtonPressed == 12) {
                // You can add logic to control the servo position here
                int newPosition = servo6.read() + 1; // Example: Increase by 1
degree
                servo6.write(newPosition);
            }
            break;
        case 16:
            // Reduce servo6 position
            if (lastButtonPressed == 16) {
                // You can add logic to control the servo position here
                int newPosition = servo6.read() - 1; // Example: Decrease by 1
degree
                servo6.write(newPosition);
            }
            break;
        }
    }
}
}
```

```
bool detectStartSignal() {
```

```
    unsigned long duration = measureHighDuration();
    return (duration >= startSignalMinDuration && duration <=
startSignalMaxDuration);
}

String readBitPattern() {
    String pattern = "";
    for (int i = 0; i < 32; i++) {
        unsigned long duration = measureHighDuration();
        if (duration >= 400 && duration <= 1000) {
            pattern += '0';
        } else if (duration >= 1000 && duration <= 2000) {
            pattern += '1';
        }
    }
    return pattern;
}

unsigned long measureHighDuration() {
    while (digitalRead(IRPin) == LOW) {}
    unsigned long startTime = micros();
    while (digitalRead(IRPin) == HIGH) {}
    return micros() - startTime;
}

bool detectContinuousSignal() {
    unsigned long duration = measureHighDuration();
    return (duration >= continuousSignalMinDuration && duration <=
continuousSignalMaxDuration);
}
```