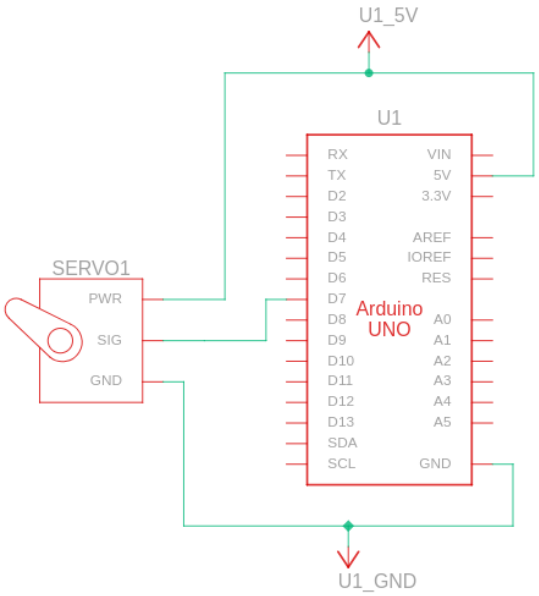
1. Servo Motor

```
#include <Servo.h>
// Define the pin for the servo
#define SERVO_PIN 7
// Create a Servo object
Servo servoMotor;

 void setup() {
// Attach the servo to the pin
servoMotor.attach(SERVO_PIN);
}

 void loop() {
// Move the servo to 0 degrees
servoMotor.write(0);
delay(1000); // Wait for 1 second
// Move the servo to 90 degrees
servoMotor.write(90);
delay(1000); // Wait for 1 second
}
```



2. IR Sensor

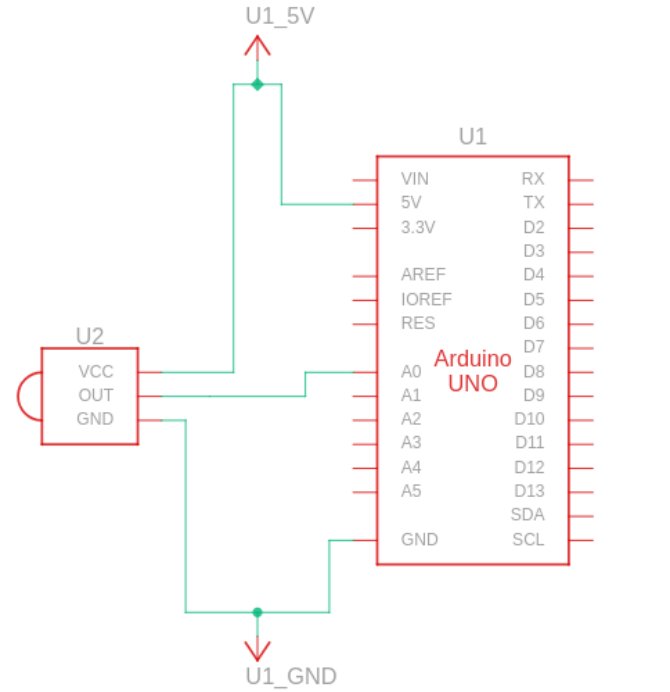
```
#define IR_SENSOR_PIN A0

void setup() {
Serial.begin(9600);
// Initialize serial communication
}

void loop() {
// Read the value from the IR sensor
int sensorValue = analogRead(IR_SENSOR_PIN);
// Print the sensor value to the Serial Monitor
Serial.print("IR Sensor Value: ");
Serial.println(sensorValue);
delay(500); // Delay for stability
}
```

Serial Monitor:

```
IR Sensor Value: 1023
IR Sensor Value: 1023
IR Sensor Value: 1023
IR Sensor Value: 1023
IR Sensor Value: 1023
IR Sensor Value: 1023
```

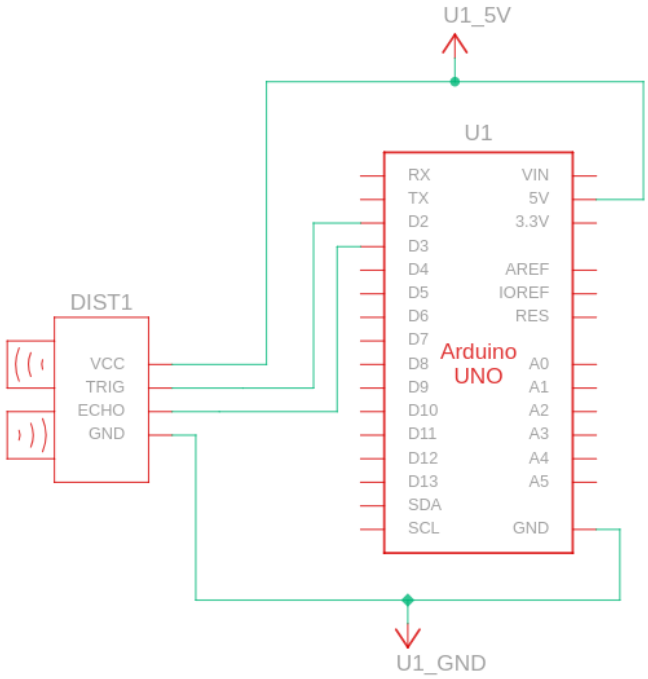


3. Ultrasonic Sensor Distance Calculation

```
#define TRIGGER_PIN 2
#define ECHO_PIN 3

 void setup() {
Serial.begin(9600); // Initialize serial communication
pinMode(TRIGGER_PIN, OUTPUT);
pinMode(ECHO_PIN, INPUT);
}

 void loop() {
long duration, distance;
// Clear the trigger pin
digitalWrite(TRIGGER_PIN, LOW);
delayMicroseconds(2);
// Send a 10 microsecond pulse to trigger the sensor
digitalWrite(TRIGGER_PIN, HIGH);
delayMicroseconds(10);
digitalWrite(TRIGGER_PIN, LOW);
// Read the duration of the echo pulse
duration = pulseIn(ECHO_PIN, HIGH);
// Calculate distance in centimeters
distance = duration * 0.034 / 2;
// Print the distance to the Serial Monitor
Serial.print("Distance: ");
Serial.print(distance);
Serial.println(" cm");
delay(1000); // Wait for stability
}
```



Serial Monitor:

Distance: 112 cm

- Connect the trigger pin of the ultrasonic sensor to digital pin 2 (TRIGGER_PIN).
- Connect the echo pin of the ultrasonic sensor to digital pin 3 (ECHO_PIN).
- The Arduino sends a 10 microsecond pulse to the trigger pin to initiate the sensor.
- The duration of the echo pulse is measured using the pulseIn() function.
- The distance is calculated using the formula: distance = duration * 0.034 / 2 (where 0.034 is the speed of sound in centimeters per microsecond and we divide by 2 to get the one-way distance).
- The distance value is printed to the Serial Monitor.
- Adjust the delay as necessary for your specific application. This delay ensures stability in the readings and prevents flooding the Serial Monitor with too much data.