

**технология.**  
моделирование роботов

90%

# Подключение моторов

Технология.  
Моделирование роботов с использованием механических элементов

Видеоинструкция

Оборудование

- Гусеница с мотором
- Колесо с мотором
- Колесо
- Мобильный манипулятор

Компоненты-датчики

- Датчик температуры и влажности
- Датчик дистанции
- Оптический датчик
- Датчик освещенности
- Акселерометр
- Датчик наклона

00m: 10s: 257ms

# Программный код для базовых движений

Технология.  
Моделирование роботов с использованием механических элементов

Видеоинструкция

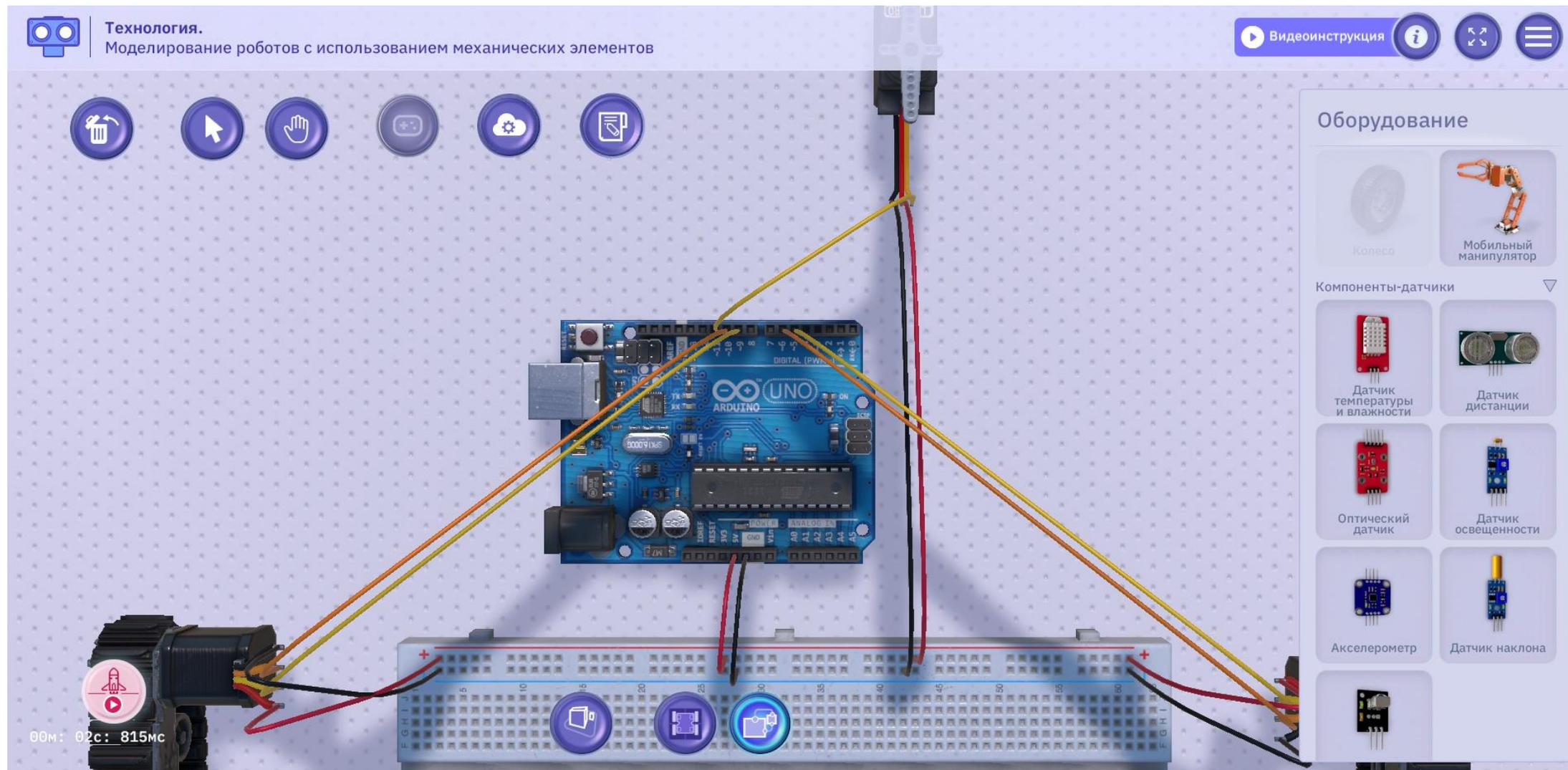
Управляющая программа\*

```
1 void setup()
2 {
3     //правый
4     pinMode(5, OUTPUT);
5     pinMode(6, OUTPUT);
6     //левый
7     pinMode(9, OUTPUT);
8     pinMode(10, OUTPUT);
9 }
10
11 void forward()
12 {
13     //вперед
14     digitalWrite(5, 1);
15     digitalWrite(6, 0);
16     digitalWrite(9, 1);
17     digitalWrite(10, 0);
18     delay(300);
19 }
20
21 void backward()
22 {
23     //назад
24     digitalWrite(5, 0);
25     digitalWrite(6, 1);
26     digitalWrite(9, 0);
27     digitalWrite(10, 1);
28     delay(120);
29 }
30
31 void loop()
32 {
33     forward();
34     backward();
35 }
```

компиляция успешна

00м: 10с: 257мс

# Подключение сервопривода



# Программный код для сервопривода

Технология.  
Моделирование роботов с использованием механических элементов

Видеоинструкция

Управляющая программа\*

```
1 // подключаем библиотеку Servo.h
2 #include <Servo.h>
3
4 // создаем объект myservo
5 Servo myservo;
6
7
8 void setup() {
9   myservo.attach(11); //указываем пин, к которому подключен сервопривод
10 }
11
12 void loop()
13 {
14   for (int i=0; i<=90; i = i + 1)
15   {
16     myservo.write(i);
17     delay(3);
18   }
19   for (int i=90; i>=0; i = i - 1)
20   {
21     myservo.write(i);
22     delay(3);
23   }
24 }
25
```

компиляция успешна

Sketch uses 2038 bytes (6%) of program storage space. Maximum is 32256 bytes.  
Global variables use 50 bytes (2%) of dynamic memory, leaving 1998 bytes for local variables. Maximum is 2048 bytes.

00м: 02с: 815мс

# Подключение УЗ датчика

Технология.  
Моделирование роботов с использованием механических элементов

Видеоинструкция

Оборудование

- Гусеница с мотором
- Колесо с мотором
- Колесо
- Мобильный манипулятор

Компоненты-датчики

- Датчик температуры и влажности
- Датчик дистанции
- Оптический датчик
- Датчик освещенности
- Акселерометр
- Датчик наклона

00m: 03c: 978ms

# Программный код УЗ датчика

Технология.  
Моделирование роботов с использованием механических элементов

Видеоинструкция

Управляющая программа

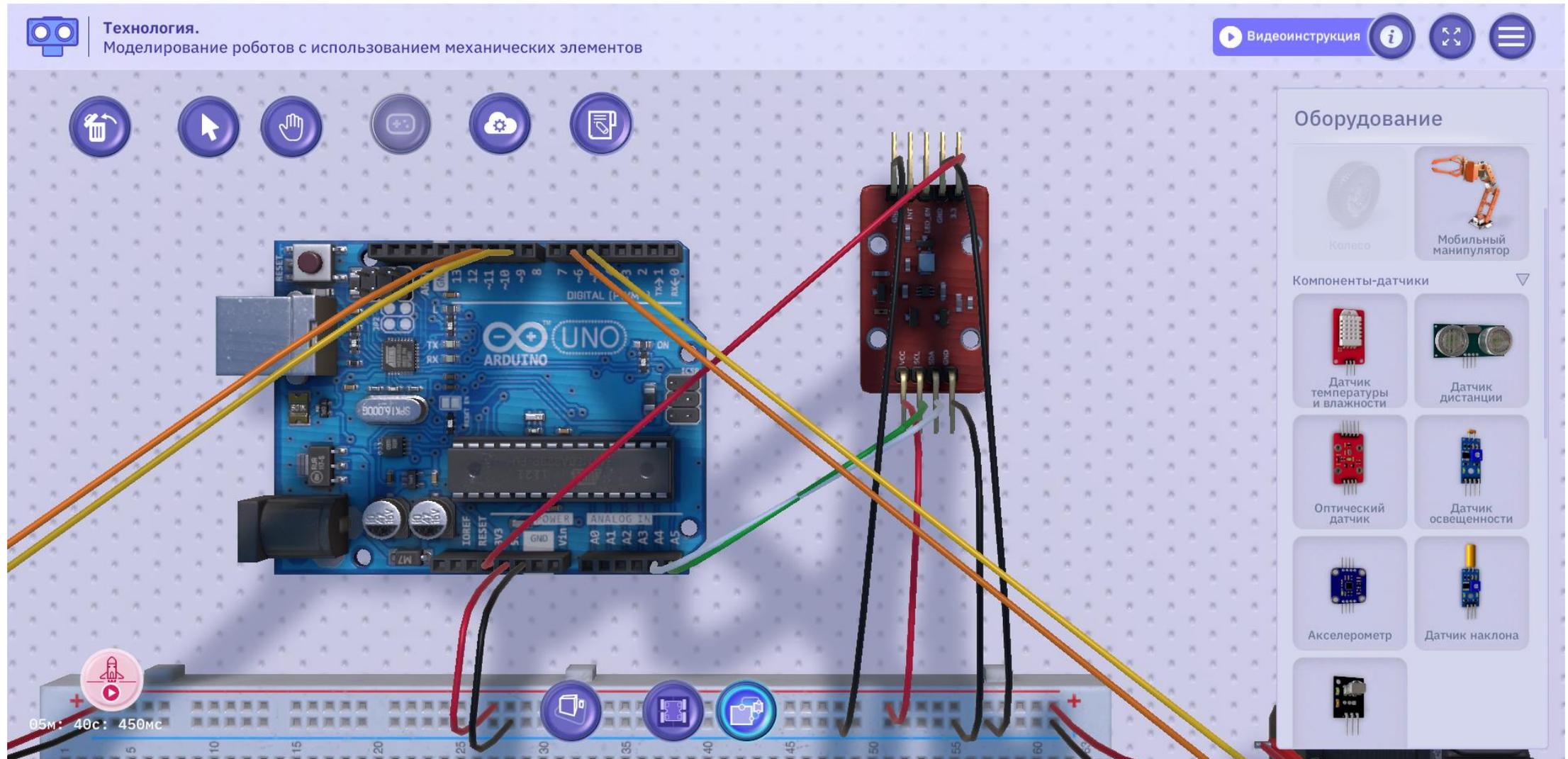
```
1 // Укажем пины для подключения УЗ датчика
2 int trigPin = 12;
3 int echoPin = 11;
4
5 void setup() {
6   Serial.begin (9600);
7   //пины УЗ
8   pinMode(trigPin, OUTPUT);
9   pinMode(echoPin, INPUT);
10  }
11
12 int dist()
13 {
14   int duration, distance;
15   // установить высокий уровень на пине Trig
16   digitalWrite(trigPin, HIGH);
17   // подождать 10 мс
18   delay(0.06);
19   digitalWrite(trigPin, LOW);
20   // узнаем длительность высокого сигнала на пине Echo
21   duration = pulseIn(echoPin, HIGH);
22   // вычисляем расстояние
23   distance = duration * 0.034/2;
24   return (distance);
25 }
26
27 void loop() {
28   Serial.println(dist());
29   delay(30);
30 }
```

компиляция успешна

```
[00:00:00.051] 101
[00:00:00.089] 101
[00:00:00.125] 101
[00:00:00.162] 101
[00:00:00.198] 101
[00:00:00.236] 101
```

00м: 03с: 978мс

# Подключение датчика освещенности (цвета)



# Программный код для датчика освещенности (цвета)

Технология. Моделирование роботов с использованием механических элементов

Видеоинструкция

Управляющая программа\*

```
1 #include <Wire.h> // Подключаем библиотеку Wire
2 #include "Adafruit_TCS34725.h" // Подключаем библиотеку Adafruit_TCS34725
3
4 // Создаем объект и передаем данные о настройке усиления и время преобразования
5 Adafruit_TCS34725 tcs = Adafruit_TCS34725(TCS34725_INTEGRATIONTIME_154MS,
6 TCS34725_GAIN_1X);
7 void setup()
8 {
9   Serial.begin(9600);
10 }
11
12 void loop()
13 {
14   uint16_t r, g, b, c, colorTemp, lux; // Создаем переменные
15   tcs.getRawData(&r, &g, &b, &c); // Получение данных
16   lux = tcs.calculateLux(r, g, b); // вычисление освещенности
17   Serial.print("Lux: ");
18   Serial.print(lux, DEC); // освещенность
19   Serial.print(" - ");
20   Serial.print("R: ");
21   Serial.print(r, DEC); // красный компонент
22   Serial.print(" ");
23   Serial.print("G: ");
24   Serial.print(g, DEC); // зеленый компонент
25   Serial.print(" ");
26   Serial.print("B: ");
27   Serial.print(b, DEC); // синий компонент
28   Serial.println(" ");
29   delay(30);
30 }
```

компиляция успешна

```
[00:00:20.214] Lux: 27553 - R: 46471 G: 52980 B: 55992
[00:00:20.400] Lux: 29159 - R: 45260 G: 53200 B: 54809
[00:00:20.586] Lux: 26836 - R: 45330 G: 51886 B: 55119
[00:00:20.771] Lux: 29889 - R: 46830 G: 54324 B: 55539
[00:00:20.957] Lux: 30736 - R: 45634 G: 54019 B: 54255
[00:00:21.142] Lux: 27382 - R: 46035 G: 52288 B: 54927
```

05м: 40с: 450мс

# Подключение датчика акселерометра

Технология.  
Моделирование роботов с использованием механических элементов

Видеоинструкция

Оборудование

- Колесо
- Мобильный манипулятор
- Компоненты-датчики
- Датчик температуры и влажности
- Датчик дистанции
- Оптический датчик
- Датчик освещенности
- Акселерометр
- Датчик наклона
- ИК-приемник

00м : 07с : 096мс

The image shows a virtual simulation environment for connecting an accelerometer sensor to an Arduino Uno board. The interface includes a toolbar with various icons (trash, mouse, hand, settings, document) and a right-hand panel titled "Оборудование" (Equipment) listing various sensors. The central workspace shows an Arduino Uno board connected to a breadboard with an accelerometer sensor. Wires connect the sensor's pins to the breadboard and the Arduino's digital pins. A video instruction icon is visible in the top right corner, and a timer shows 00m : 07s : 096ms.

# Программный код для датчика акселерометра

Технология. Моделирование роботов с использованием механических элементов

Видеоинструкция



Управляющая программа

```
1 #include <Wire.h> // Подключаем библиотеку Wire
2 #include "Adafruit MPU6050.h" // Подключаем библиотеку Adafruit MPU6050.h
3
4
5 Adafruit MPU6050 mpu ;
6
7 void setup() {
8
9   Serial.begin(9600);
10 }
11
12 void gyro()
13 {
14   sensors_event_t a, g, temp;
15   mpu.getEvent(&a, &g, &temp);
16   Serial.print("Acceleration X: ");
17   Serial.print(a.acceleration.x);
18   Serial.print(", Y: ");
19   Serial.print(a.acceleration.y);
20   Serial.print(", Z: ");
21   Serial.print(a.acceleration.z);
22   Serial.println(" m/s^2");
23   Serial.print("Rotation X: ");
24   Serial.print(g.gyro.x);
25   Serial.print(", Y: ");
26   Serial.print(g.gyro.y);
27   Serial.print(", Z: ");
28   Serial.print(g.gyro.z);
29   Serial.println(" rad/s");
30 }
31
32
33 void loop() {
34   gyro();
35   delay(60);
36 }
```

00м: 07с: 096мс

компиляция успешна