

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Рязанский государственный радиотехнический университет имени В.Ф.
Уткина»

**Пояснительная записка к курсовой работе по теме:
«Треккер солнца»**

Выполнили: ст. гр. 233
Машкова Е. А.
Минюков И.А.
Проверил:
Борисов А.Г.

Рязань 2025

Оглавление

Сценарий применения	3
Аппаратная часть прототипа	3
Текст программы	5

Сценарий применения

Солнечная энергия становится все более важным источником возобновляемой энергии в условиях современных вызовов, связанных с изменением климата и истощением традиционных энергетических ресурсов. Эффективность солнечных панелей напрямую зависит от их ориентации относительно солнца, что приводит к разработке различных систем, направленных на максимизацию поглощения солнечного света. Одним из наиболее перспективных решений являются солнечные трекеры, которые автоматически изменяют угол наклона солнечных панелей в зависимости от положения солнца на небе.

Солнечный трекер может быть использован в энергетике, робототехнике, а также в образовательных целях для изучения принципов работы с датчиками и микроконтроллерами.

Энергетика: солнечный трекер значительно повысит эффективность солнечных панелей, что позволит сократить затраты на добычу солнечной энергии

Робототехника: используя солнечный трекер можно в разы уменьшить, а, следовательно, облегчить портативные солнечные панели для роботизированных платформ

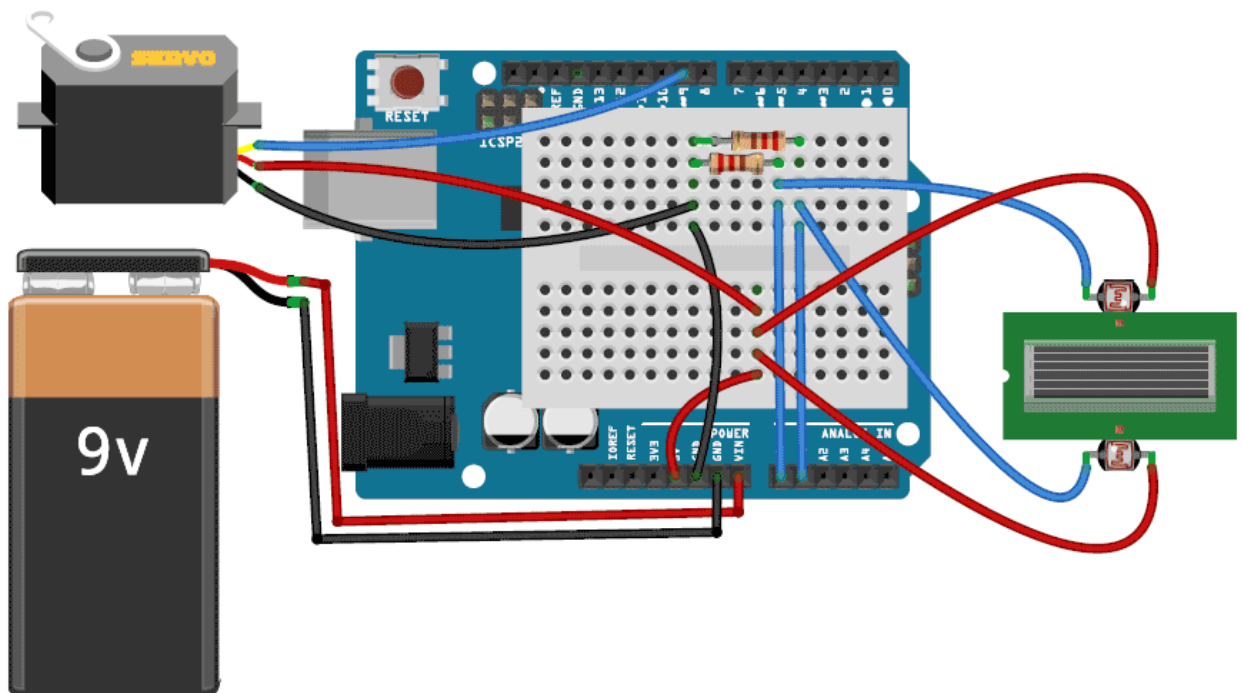
Образование: устройство может служить наглядным пособием для изучения принципов работы фоторезисторов, солнечных панелей и микроконтроллеров.

Аппаратная часть прототипа

1. Плата Arduino Uno
2. Сервомотор sg90
3. Солнечная панель
4. Фоторезистор (2 шт.)

5. Резистор 10 кОм (2 шт.)
6. Батарея (от 6 до 12 В)

В этом проекте фоторезисторы будут работать в качестве детекторов света. Когда на фоторезистор начинает падать свет его сопротивление уменьшается. Поэтому фоторезисторы так часто используются в различных детекторах света или темноты. В нашем проекте два фоторезистора будут помещены на обоих концах солнечной панели, а сервомотор будет использоваться для поворота солнечной панели. Сервомотор будет поворачивать солнечную панель в направлении того фоторезистора, чье сопротивление будет меньше, что будет означать что на него падает больше солнечного света. Если на оба фоторезистора будет падать одинаковое количество солнечного света, сервомотор не будет поворачивать солнечную панель. То есть сервомотор будет пытаться повернуть солнечную панель в такое положение, чтобы оба фоторезистора имели примерно одинаковое сопротивление, что будет означать что на них падает примерно одинаковое количество солнечного света. Если же сопротивление одного фоторезистора становится меньше, чем сопротивление другого, то сервомотор будет поворачивать солнечную панель в направлении этого фоторезистора.



Текст программы

При написании программы для нашей солнечной панели, следующей за Солнцем, первым делом необходимо подключить библиотеку для сервомотора. Далее инициализируем переменную для хранения начальной позиции сервомотора. Также инициализируем переменные для считывания данных с фоторезисторов и контакт, к которому подключен сервомотор.

```
#include <Servo.h> //подключение библиотеки для работы с сервомотором
```

```
Servo sg90; //даем имя нашему сервомотору, назовем его sg90
```

```
int initial_position = 90; //переменная для хранения начальной позиции сервомотора  
int LDR1 = A0; //контакт, к которому подключен первый фоторезистор
```

```
int LDR2 = A1; // контакт, к которому подключен второй фоторезистор
```

```
int error = 5; //переменная для хранения ошибки
```

```
int servopin=9;
```

Команда `sg90.attach(servopin)` «подсоединяет» сервомотор к контакту 9 платы Arduino. Далее устанавливаем контакты, к которым подключены фоторезисторы, в режим ввода данных. Затем устанавливаем сервомотор в начальную позицию (90 градусов).

```
void setup()
```

```
{
```

```
sg90.attach(servopin); // присоединяет сервомотор к контакту 9
```

```
pinMode(LDR1, INPUT); // на ввод данных
```

```
pinMode(LDR2, INPUT); // на ввод данных
```

```

    sg90.write(initial_position); //поворачиваем сервомотор в начальную
позицию (90 градусов)

    delay(2000); // задержка 2 секунды

}

int R1 = analogRead(LDR1); // считывание значения с фоторезистора 1
int R2 = analogRead(LDR2); // считывание значения с фоторезистора 2

int diff1= abs(R1 - R2); // расчет разницы между значениями с
фоторезисторов

int diff2= abs(R2 - R1);

if((diff1 <= error) || (diff2 <= error)) {

    //если разница меньше величины ошибки/погрешности (error) то не
делаем ничего }

else { if(R1 > R2)

{

    initial_position = --initial_position; //поворачиваем серводвигатель в
направлении 0градусов

}

if(R1 < R2)

{

    initial_position = ++initial_position; //поворачиваем серводвигатель в
направлении 180 градусов

} }

```