

Потенциометр (Трема-модуль v2.0)



Общие сведения:

[Трема-модуль Потенциометр](#) отлично подойдет для управления различными проектами. Исполнен в линейке Трема-модулей, благодаря этому вам не придется ничего паять и собирать на макетной плате.

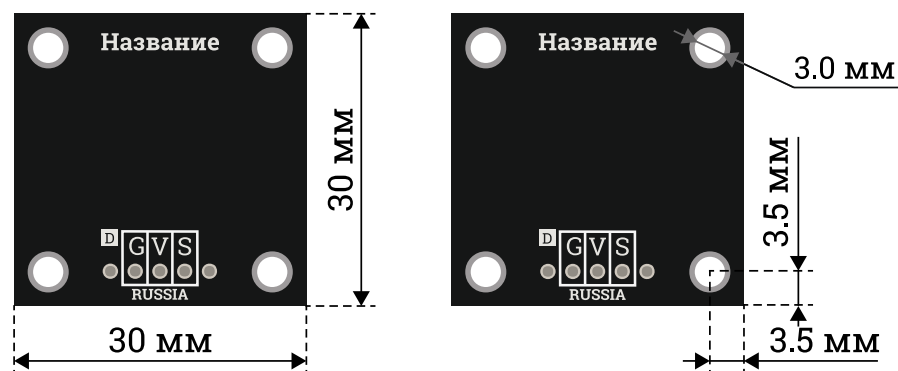
Удобно использовать для настройки скорости вращения двигателя, угла поворота сервопривода, яркости светодиода и многого другого.

Если вы хотите управлять потенциометром линейным движением, а не вращательным, то можете воспользоваться аналогичным вариантом [Трема-модуль ползунковый потенциометр](#).

Спецификация:

- Сопротивление: 10 кОм
- Тип потенциометра: линейный
- Угол поворота движка: 300°

Все модули линейки "Тема" выполнены в одном формате



Подключение:

В комплекте имеется кабель для быстрого и удобного подключения к [Trema Shield](#).

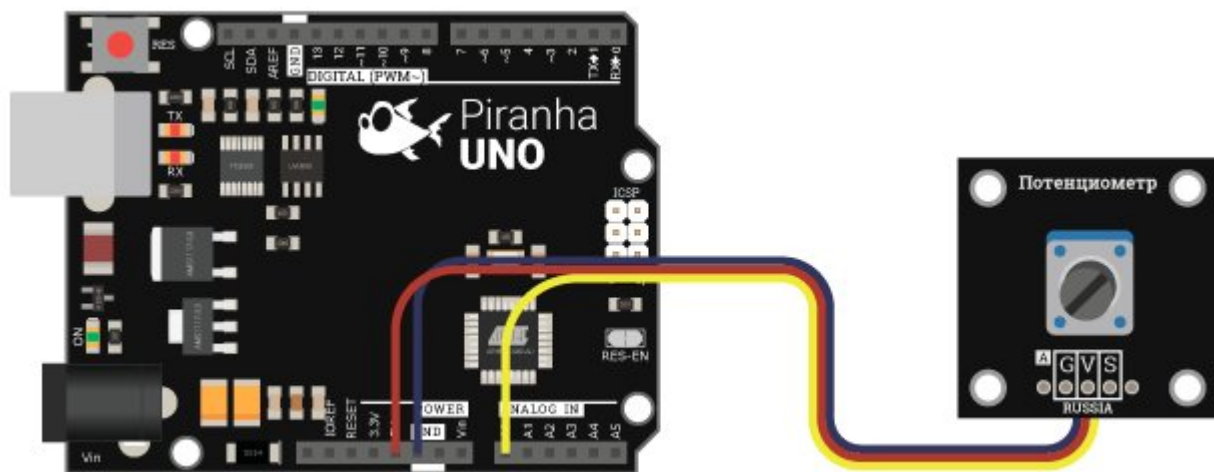
На выходе из датчика мы имеем аналоговый сигнал

Датчик подключается к любому из аналоговых входов.

Модуль удобно подключать 3 способами, в зависимости от ситуации:

Способ - 1 : Используя проводной шлейф и Piranha UNO

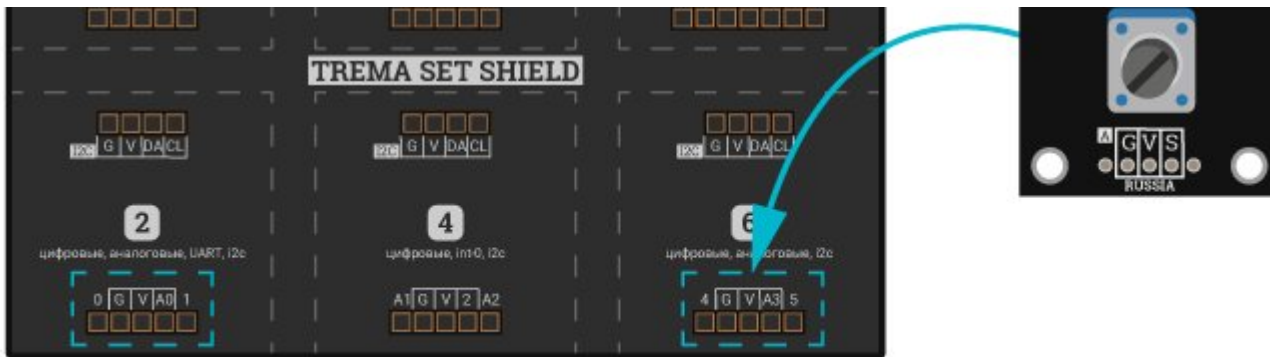
Используя провода «Папа – Мама», подключаем напрямую к контроллеру Piranha UNO



Способ - 2 : Используя Trema Set Shield

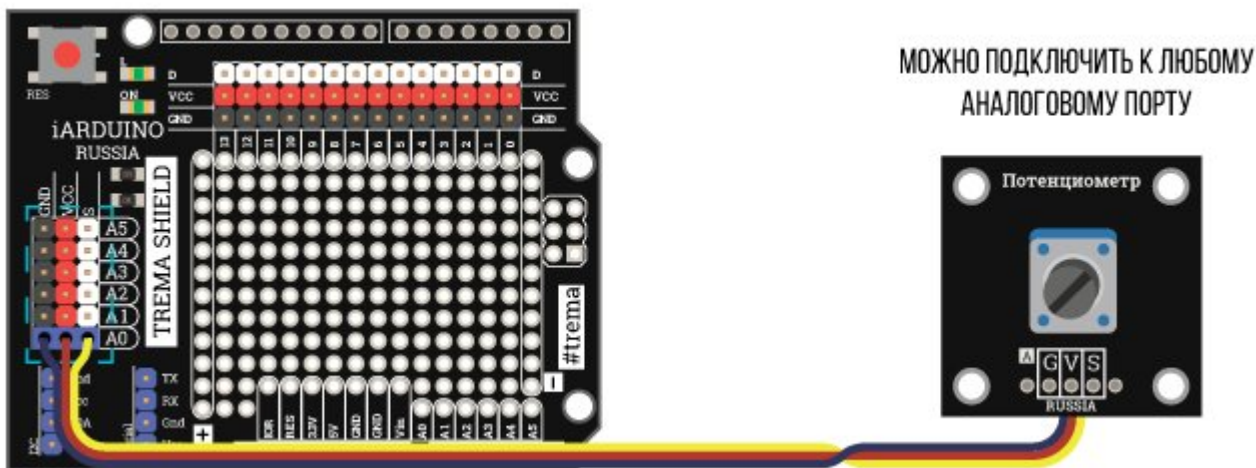
Модуль можно подключить к любому из аналоговых входов Trema Set Shield.





Способ - 3 : Используя проводной шлейф и Shield

Используя 3-х проводной шлейф, к Trema Shield, Trema-Power Shield, Motor Shield, Trema Shield NANO и тд.



Примеры:

Регулировка потенциометром угла поворота сервопривода.

```
#include <Servo.h> // Подключаем библиотеку для работы с сервоприводом
Servo servo1; // Объявляем переменную servo типа "servo1"
int8_t potencPin = A0; // Указываем номер вывода, к которому подключен потенциометр
```

```
int16_t Volume = 0; // Определяем переменную для значений, считанных с потенциометра
uint8_t konvert = 0; // Определяем переменную для значений, которые будут передаваться как ШИМ-сигнал
void setup() {
    servo1.attach(3); // привязываем сервопривод к выводу 3
}
void loop() {
    Volume = analogRead(potencPin); // Считываем значения с потенциометра
    konvert = map(Volume, 0, 1023, 0, 180); // Преобразуем полученный интервал значений с потенциометра в интервал для ШИМ-сигнала
    servo1.write(konvert); // Подаём на вывод сервопривода значение угла поворота
}
```