**Мотор 28BYJ-48 и драйвер ULN2003**

Описание

28BYJ-48 — это шаговый двигатель. Это бесколлекторный двигатель, вращение вала которого осуществляется шагами (дискретное перемещение)

ULN2003 — это интегральная микросхема, представляющая собой драйвер для управления высоковольтными и высокотоковыми нагрузками от низковольтного источника питания, например, микроконтроллера. Она содержит восемь независимых транзисторных ключей с открытым коллектором.





Основные характеристики

Тип: Шаговый двигатель (28BYJ-48)

Шаг: 5.625° (200 шагов на оборот) — это означает, что для полного оборота требуется 200 импульсов управления. Некоторые источники указывают 48 шагов, но это не совсем точно. Если рассматривать микрошаги, то количество шагов может быть значительно больше.

Размер: Компактный размер, подходит для проектов с ограниченным пространством.

Напряжение питания: обычно 5В, но следует уточнять в спецификации конкретного экземпляра, т.к. могут быть незначительные отклонения.

Ток: обычно небольшой ток, что делает его энергоэффективным. Точный ток нужно искать в спецификации.

Управление: требует специализированного драйвера, так как сам по себе он не имеет достаточной мощности для управления (за исключением очень малых нагрузок). Часто используется вместе с драйвером ULN2003 или подобными.

Драйвер: без драйвера 28BYJ-48 работать не будет. Он нуждается в драйвере для управления током и напряжением. ULN2003 один из немногих вариантов.

Микрошаги: Возможность использования микрошагов (subdivision) для более плавного и точного вращения, но это требует более сложного управления.

Тип: Драйвер с открытым коллектором (ULN2003). Это означает, что выходные транзисторы находятся в состоянии высокого импеданса, когда не активны. Для включения нагрузки необходимо подать низкий уровень на вход драйвера.

Количество каналов: 8 независимых каналов.

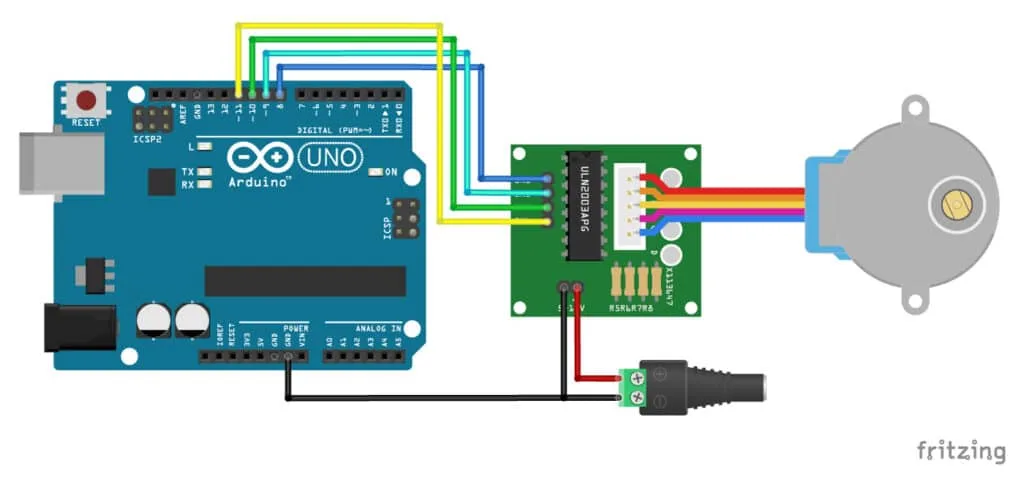
Напряжение питания: обычно 5В, но некоторые варианты могут поддерживать и до 36В. Важно проверить спецификации конкретного варианта.

Выходной ток: обычно до 500 мА на канал, но это зависит от температуры окружающей среды и условий эксплуатации. Более высокие токи могут потребовать радиатора.

Защита: встроенная защита от перенапряжения и перегрузки по току.

Применение: Управление шаговыми двигателями, реле, соленоидами, светодиодами (с соответствующими токоограничивающими резисторами) и другими нагрузками.

Подключение



Модуль драйвера шагового двигателя ULN2003 подключается к Arduino контактами IN1 – IN4 к D8 – D11 соответственно. Вал шагового двигателя в результате выполнения будет делать один полный оборот в одну сторону, затем в обратную.

VCC: подключается к положительному полюсу источника питания

GND: подключается к отрицательному полюсу источника питания.

Входной пин: подключается к выходу микроконтроллера или другого устройства управления. Логический 0 активирует выходной транзистор, логический 1 - деактивирует.

Выходной пин (например, OUT1): подключается к нагрузке через ограничительный резистор и (чаще всего) диод. Диод подключается параллельно нагрузке, катод к выходу ULN2003, анод к нагрузке. Подтягивающий резистор: подключается между выходным пином (OUT1) и VCC. Значение резистора зависит от нагрузки и обычно указывается в DataSheet. Диоды: защищают ULN2003 от обратных токов, возникающих при выключении индуктивных нагрузок. Необходимо использовать диоды с достаточным обратным напряжением.

## Библиотеки

**Stepper.h** — это библиотека для управления шаговым униполярным и биполярным двигателем с помощью плат Arduino. Для использования библиотеки нужно открыть «Менеджер библиотек» в Arduino IDE и установить её оттуда. **Пример использования:** 1. Включить библиотеку #include <Stepper.h>; 2. Определить количество шагов, за которое двигатель проходит полный поворот: const int stepsPerRevolution; 3. Создать экземпляр класса с указанным количеством шагов и выводами, к которым подключается двигатель: Stepper myStepper = Stepper(steps, pin1, pin2);

## Программа на драйвер с пошаговым двигателем 28BYJ-48:

//Includes the Arduino Stepper Library

#include <Stepper.h>

// Defines the number of steps per rotation

const int stepsPerRevolution = 2038;

// Creates an instance of stepper class

// Pins entered in sequence IN1-IN3-IN2-IN4 for proper step sequence

Stepper myStepper = Stepper(stepsPerRevolution, 8, 10, 9, 11);

void setup()

{

// Nothing to do (Stepper Library sets pins as outputs)

}

void loop()

{

// Rotate CW slowly at 5 RPM

myStepper.setSpeed(5);

myStepper.step(stepsPerRevolution);

delay(1000);

// Rotate CCW quickly at 10 RPM

myStepper.setSpeed(10);

myStepper.step(-stepsPerRevolution);

delay(1000);

}

Объяснение программного кода

Эскиз начинается с включения встроенной библиотеки шаговых программ.

#include <Stepper.h> Далее определяется константа stepsPerRevolution, которая содержит количество «шагов», которые делает двигатель за один оборот. В нашем случае это 2038.

const int stepsPerRevolution = 2038; Последовательность шагов однополярного шагового двигателя 28BYJ-48: IN1-IN3-IN2-IN4. Мы будем использовать эту информацию для управления двигателем, создав экземпляр библиотеки шагового двигателя myStepper с последовательностью контактов 8, 10, 9, 11.

Убедитесь, что вы всё сделали правильно, иначе двигатель не будет работать должным образом.

Stepper myStepper = Stepper(stepsPerRevolution, 8, 10, 9, 11);

Поскольку библиотека шагового двигателя внутренне настраивает четыре управляющих вывода как выходные, в функции настройки ничего не нужно настраивать, поэтому она остаётся пустой.

void setup() {

}

В функции цикла мы используем функцию setSpeed() для указания скорости, с которой должен двигаться шаговый двигатель, и функцию step() для указания количества шагов.

Передача отрицательного числа в функцию step() заставляет двигатель вращаться в противоположном направлении.

Первый фрагмент кода заставляет двигатель очень медленно вращаться по часовой стрелке, а второй — очень быстро вращаться против часовой стрелки.