

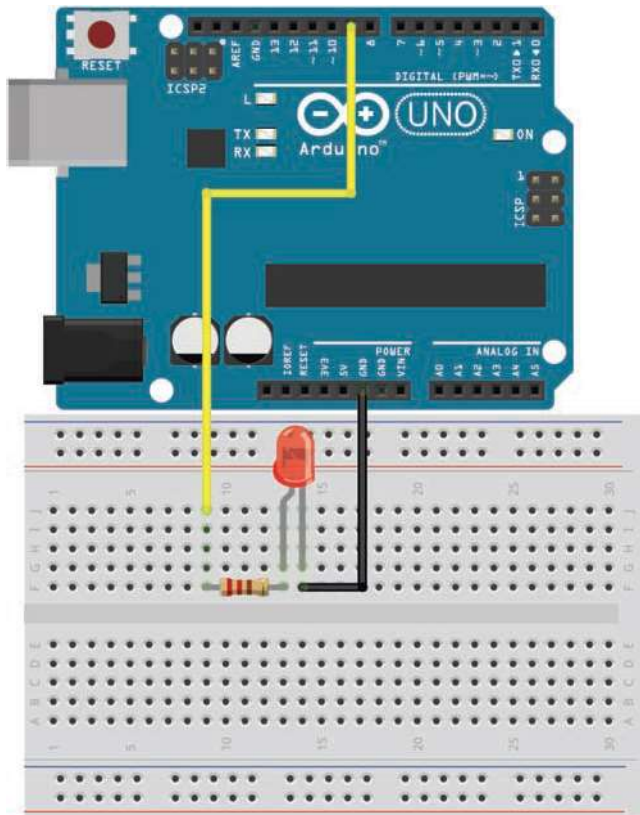


Рис. 2.2. Подключение светодиода к плате Arduino Uno (рисунок создан в программе Fritzing)

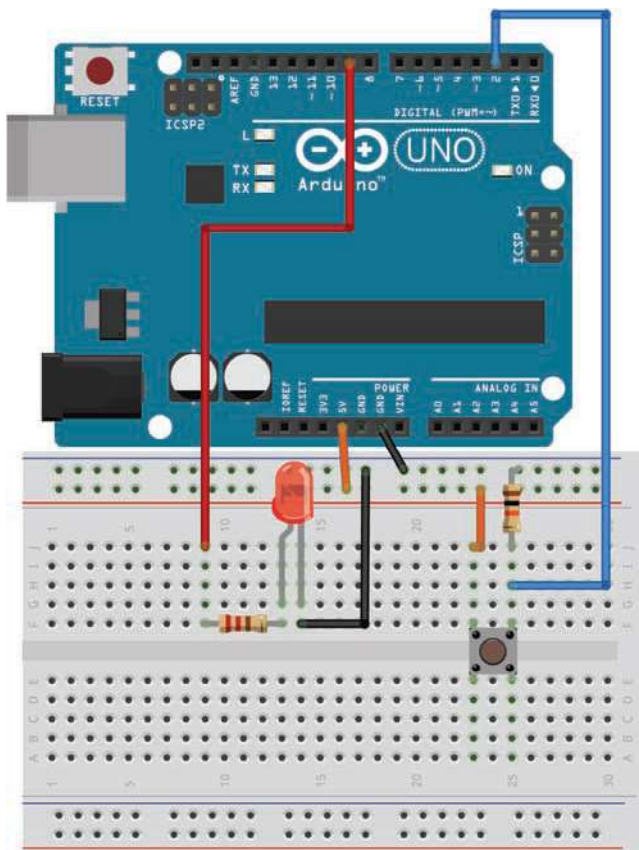


Рис. 2.6. Монтажная схема подключения кнопки к плате Arduino (рисунок создан в программе Fritzing)

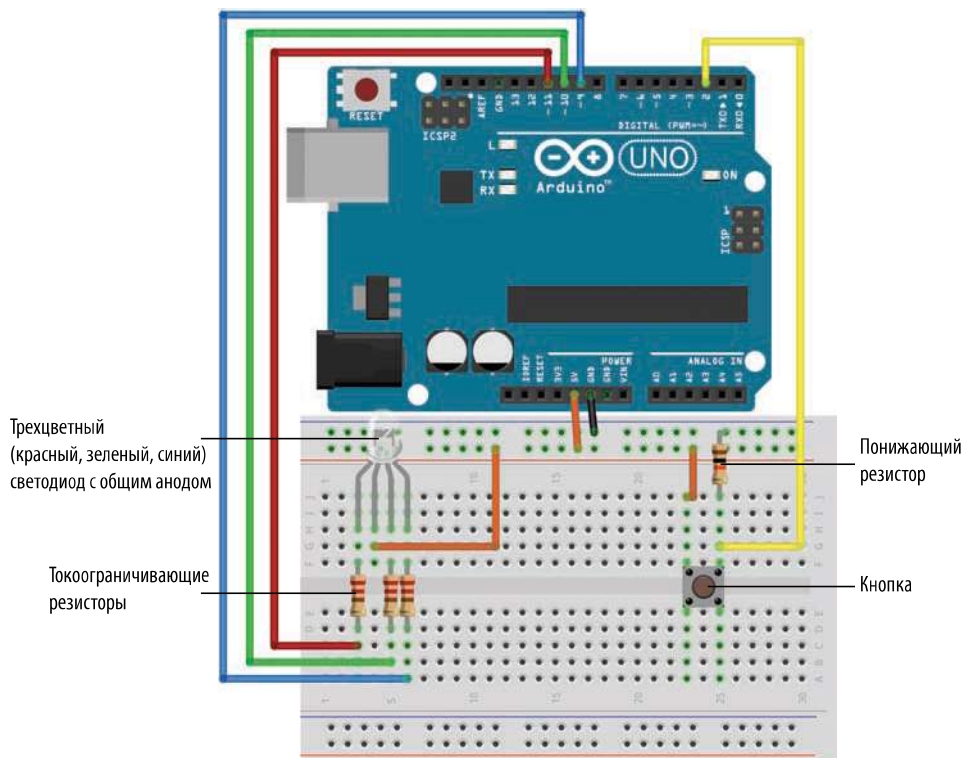


Рис. 2.8. Монтажная схема ночника на трехцветном светодиоде (рисунок создан в программе Fritzing)

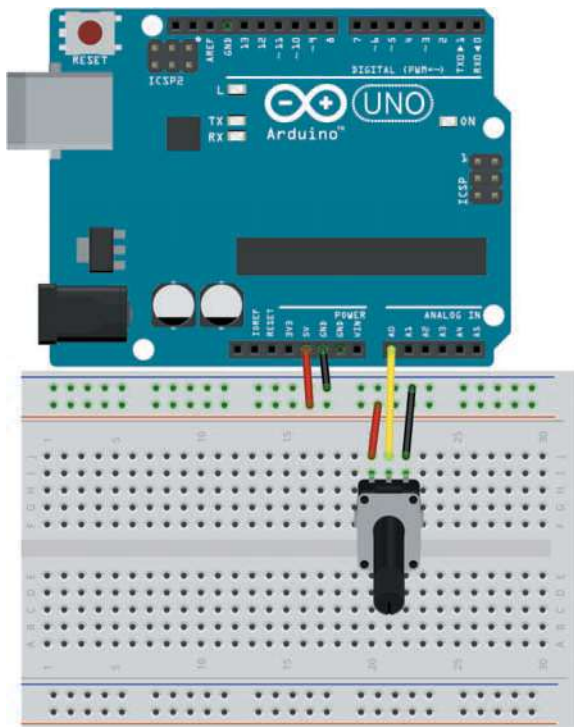


Рис. 3.3. Монтажная схема подключения потенциометра к плате Arduino (рисунок создан в программе Fritzing)

в целом они могут быть любого размера и любой формы. Но все потенциометры имеют три вывода. Один из крайних выводов подключается на землю, а другой на контакт положительного питания 5 В платы Arduino. Потенциометры симметричны, поэтому не имеет значения, какой из его крайних выводов подключать к земле, а какой к питанию 5 В. Средний вывод потенциометра подключается к контакту A0 разъема аналоговых входов платы Arduino. Это иллюстрируется монтажной схемой на рис. 3.3.

Вращение ручки потенциометра меняет напряжение на его среднем выводе (которое подается на аналоговый вход 0 платы Arduino) в диапазоне от 0 до 5 В. В этом можно убе-

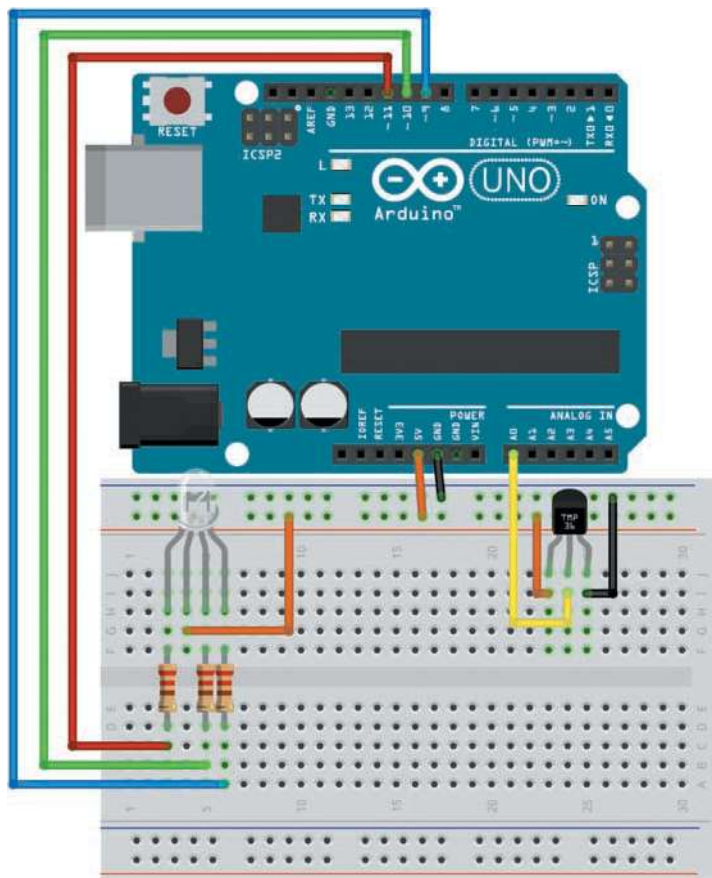


Рис. 3.8. Подключение датчика температуры к плате Arduino (рисунок создан в программе Fritzing)

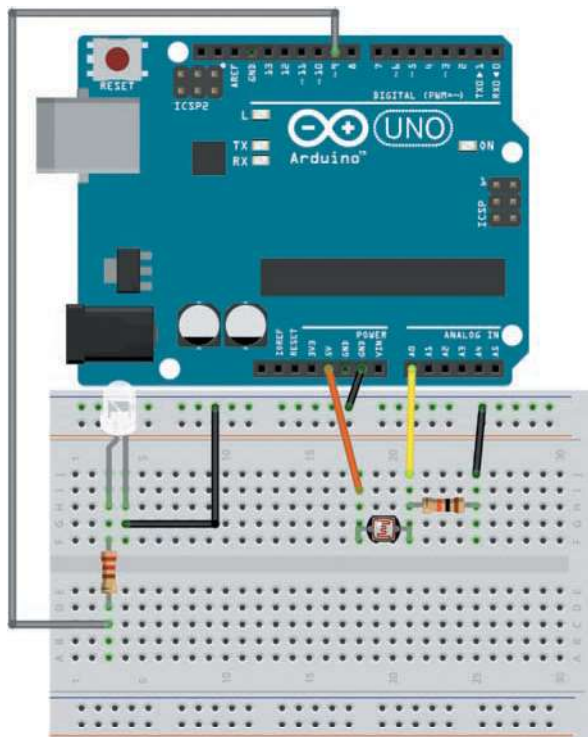


Рис. 3.11. Делитель напряжения с фоторезистором вместо одного из резисторов (рисунок создан в программе Fritzing)

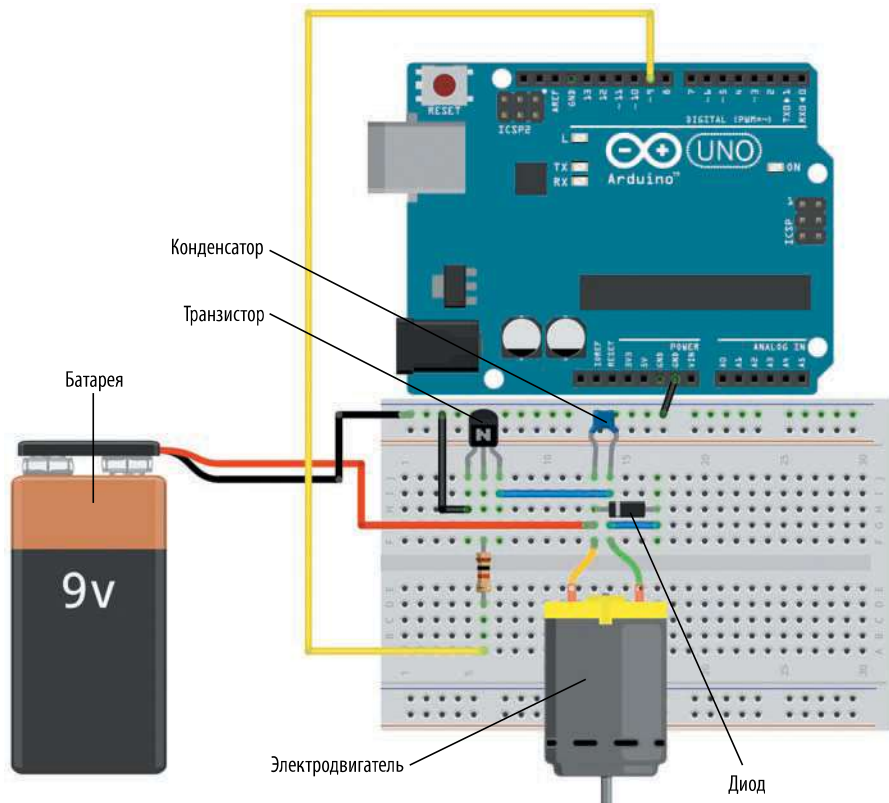


Рис. 4.3. Монтажная схема подключения электродвигателя к плате Arduino (рисунок создан в программе Fritzing)

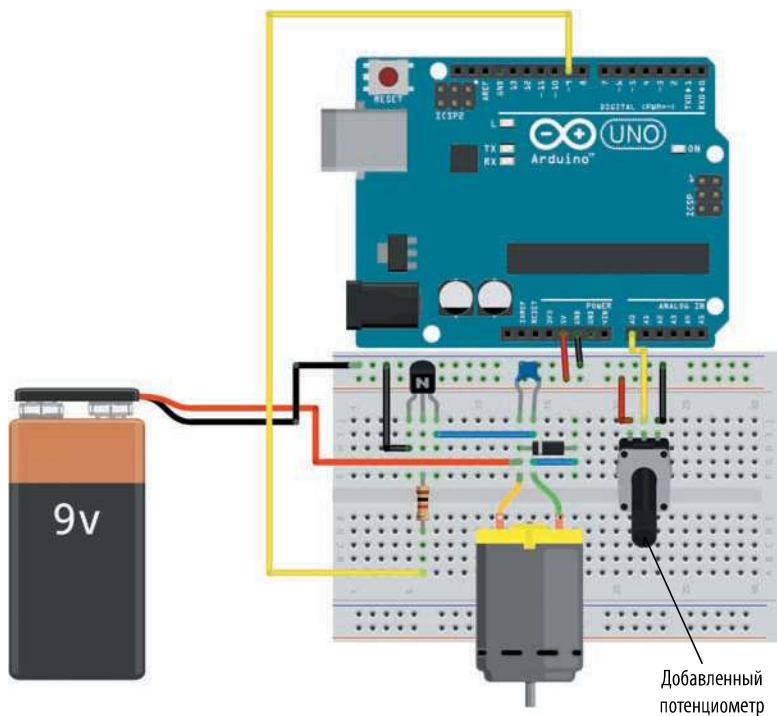


Рис. 4.4. Подключение потенциометра (рисунок создан в программе Fritzing)

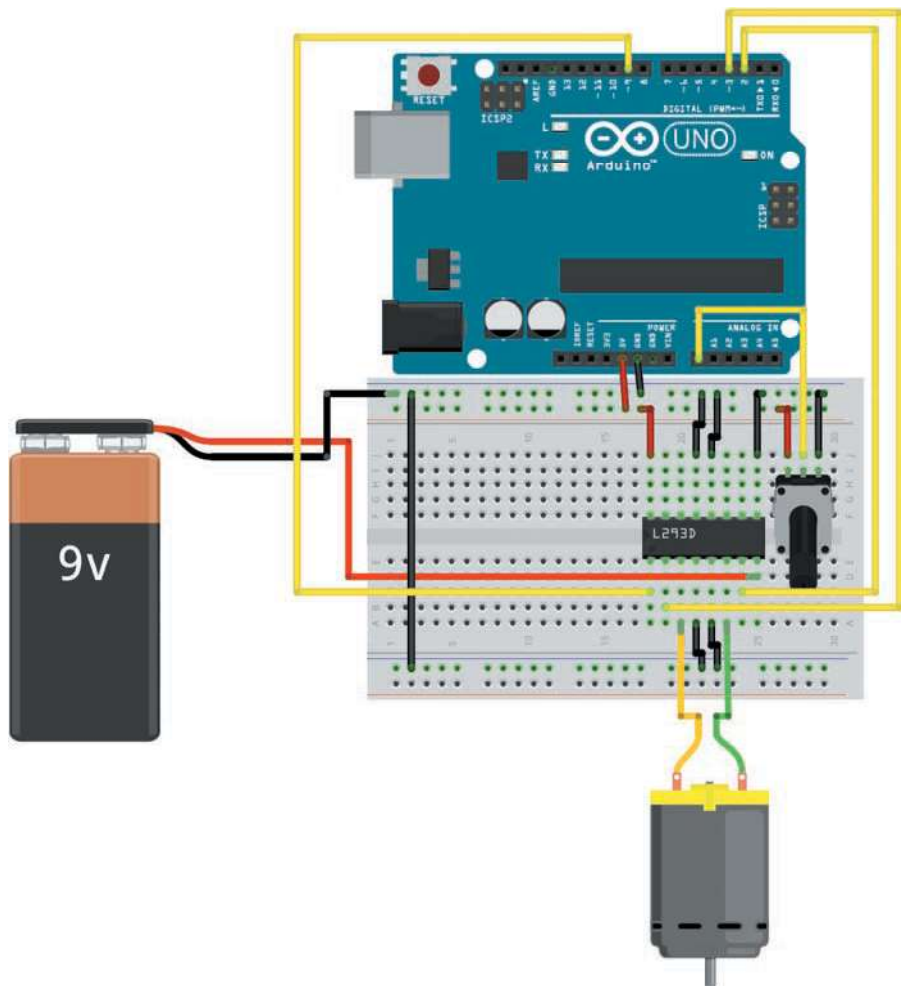


Рис. 4.7. Монтажная схема подключения драйвера Н-моста (рисунок создан в программе Fritzing)

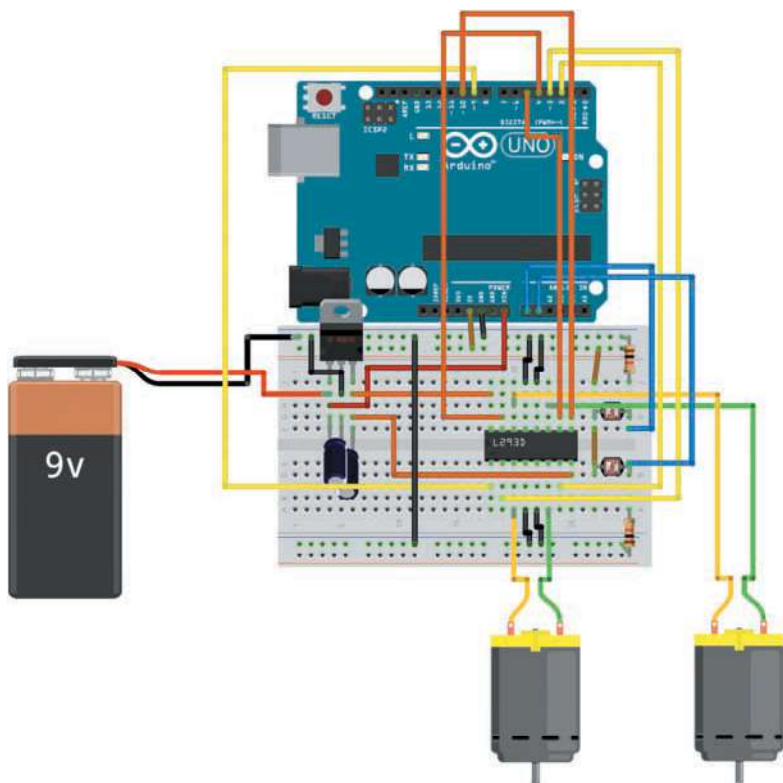


Рис. 4.11. Монтажная схема управления роботом (рисунок создан в программе Fritzing)

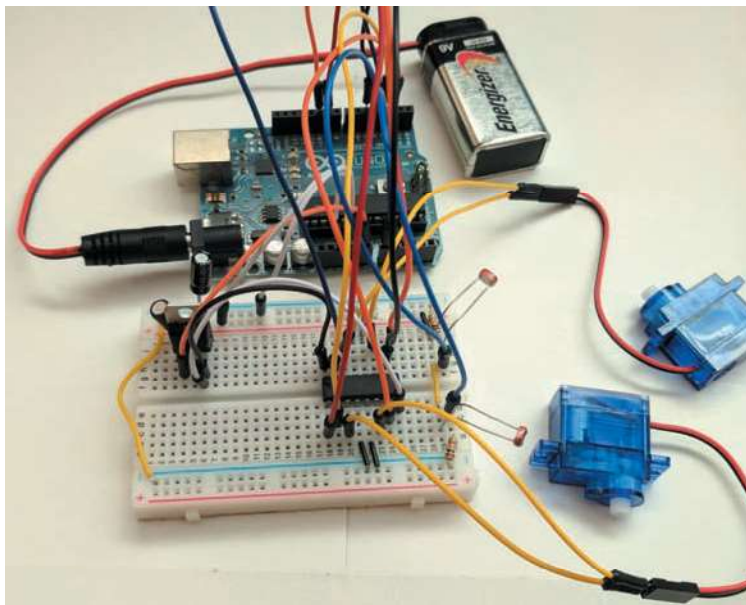


Рис. 4.12. Собранная схема управления самоходным роботом

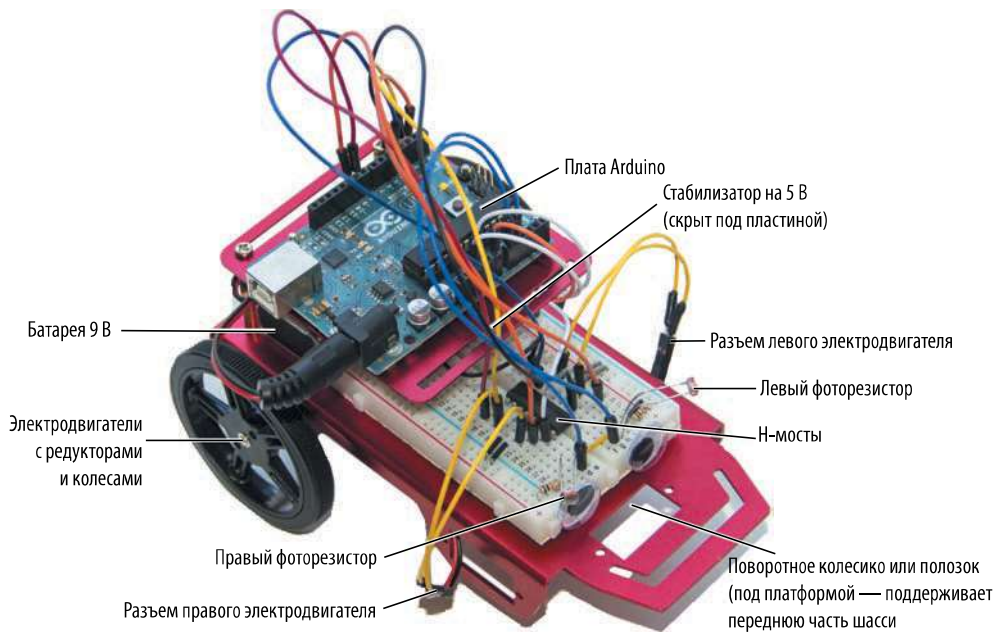


Рис. 4.13. Полностью собранное самоходное шасси

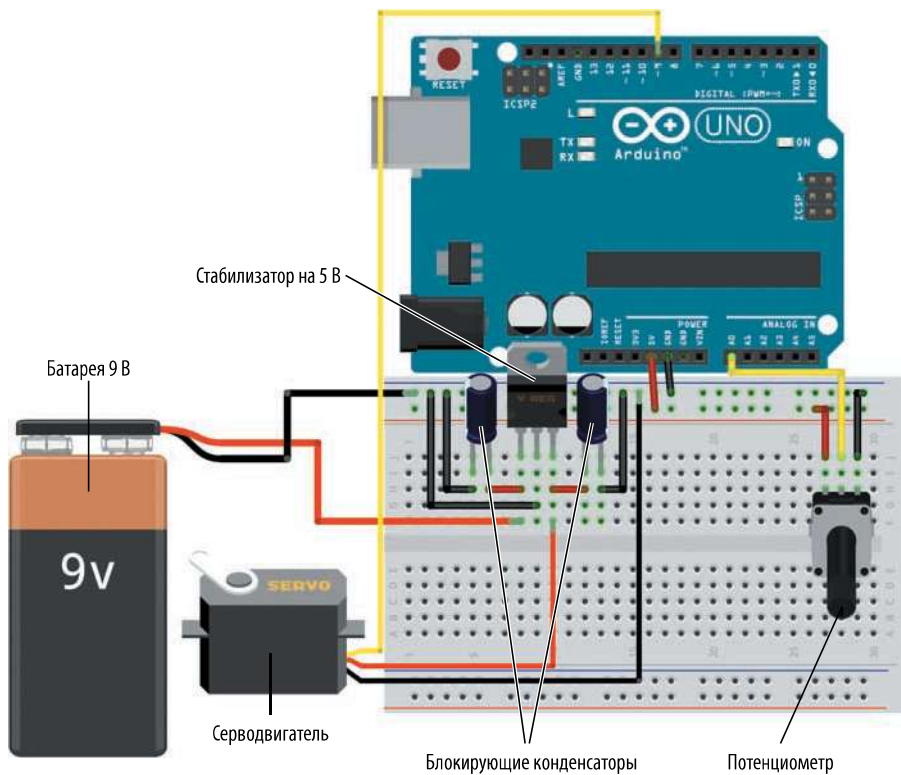


Рис. 5.3. Монтажная схема подключения сервопривода (рисунок создан в программе Fritzing)

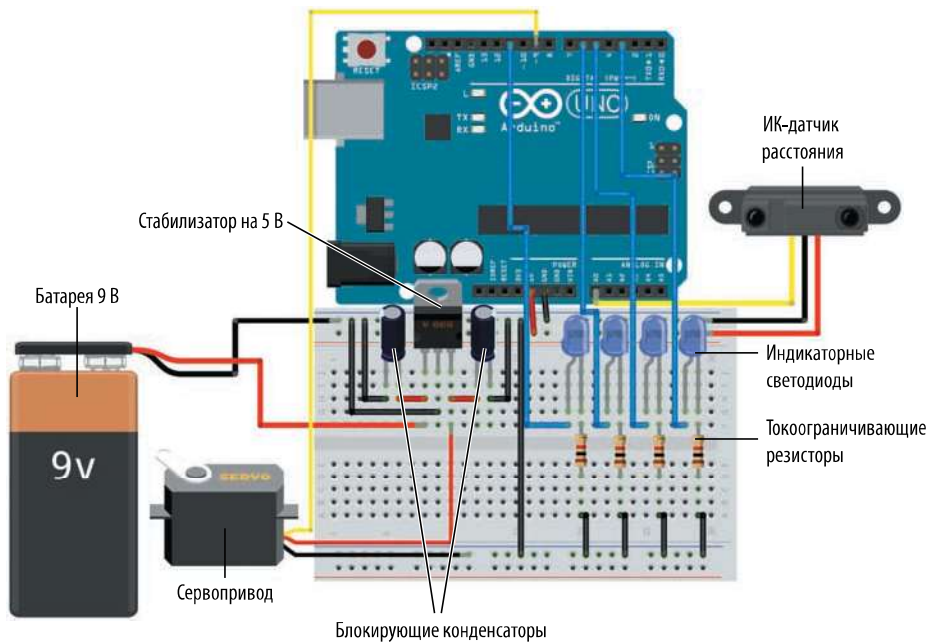
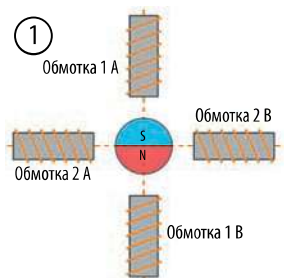
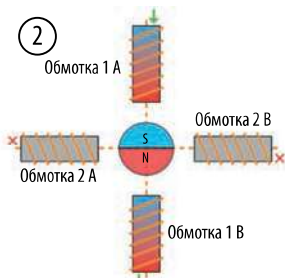


Рис. 5.5. Монтажная схема проекта сканирующего ИК-дальномера (рисунок создан в программе Fritzing)



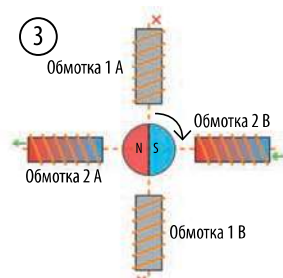
Питание не подается

Ток через обмотки не протекает.
Ротор можно свободно вращать рукой.



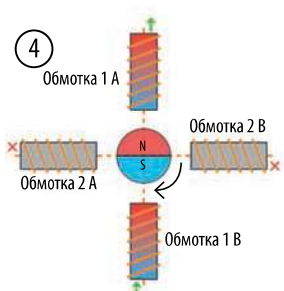
Питание подается

Ток протекает через обмотку 1, индуцируя магнитное поле.
Ротор фиксируется в текущем положении.



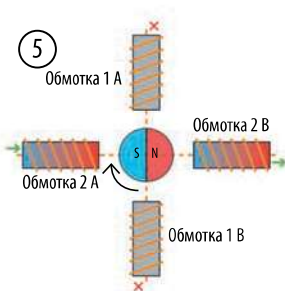
Питание подается, ротор вращается

Ток протекает через обмотку 2, индуцируя новое магнитное поле, сдвинутое по отношению к предыдущему.
Ротор в виде постоянного магнита вращается вслед за этим полем.



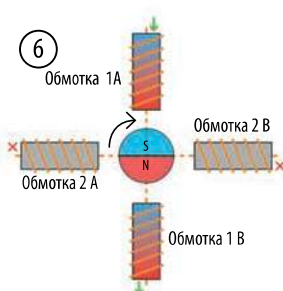
Питание подается, ротор вращается

Ток протекает через обмотку 1, но в противоположном направлении по сравнению с шагом 2.
Созданное магнитное поле также имеет обратную полярность.
Ротор вращается вслед за этим полем.



Питание подается, ротор вращается

Ток протекает через обмотку 2, но в противоположном направлении по сравнению с шагом 3.
Созданное магнитное поле также имеет обратную полярность, чем магнитное поле на шаге 3.
Ротор вращается вслед за этим полем.



Питание подается, ротор вращается

Повторяется шаг 2.
Ротор снова вращается, возвращаясь в исходное положение.

Рис. 5.7. Последовательность операций шагового электродвигателя

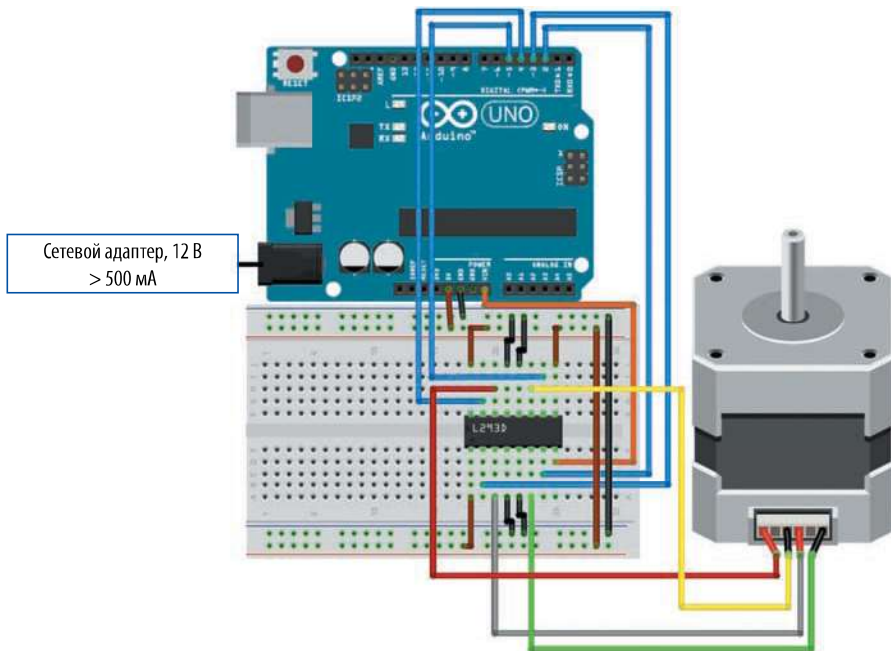


Рис. 5.9. Монтажная схема подключения Н-моста и шагового электродвигателя (рисунок создан в программе Fritzing)

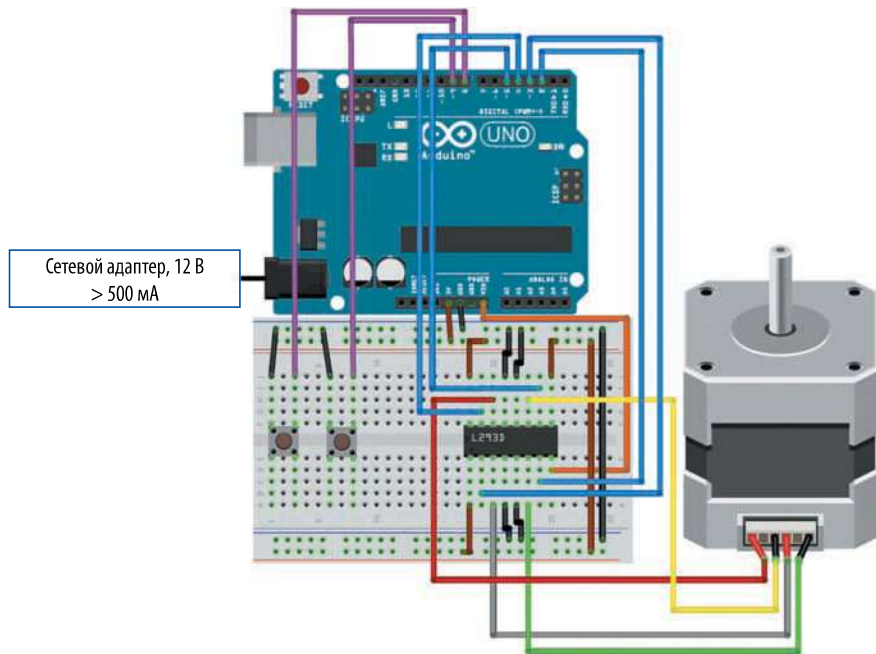


Рис. 5.10. Монтажная схема хронографа (рисунок создан в программе Fritzing)

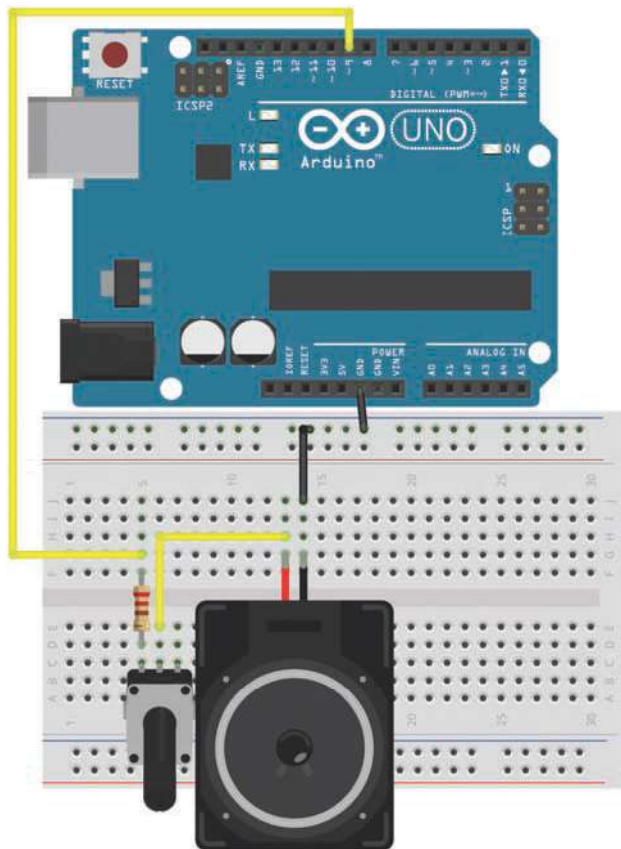


Рис. 6.5. Монтажная схема подключения динамика к плате Arduino
(рисунок создан в программе Fritzing)

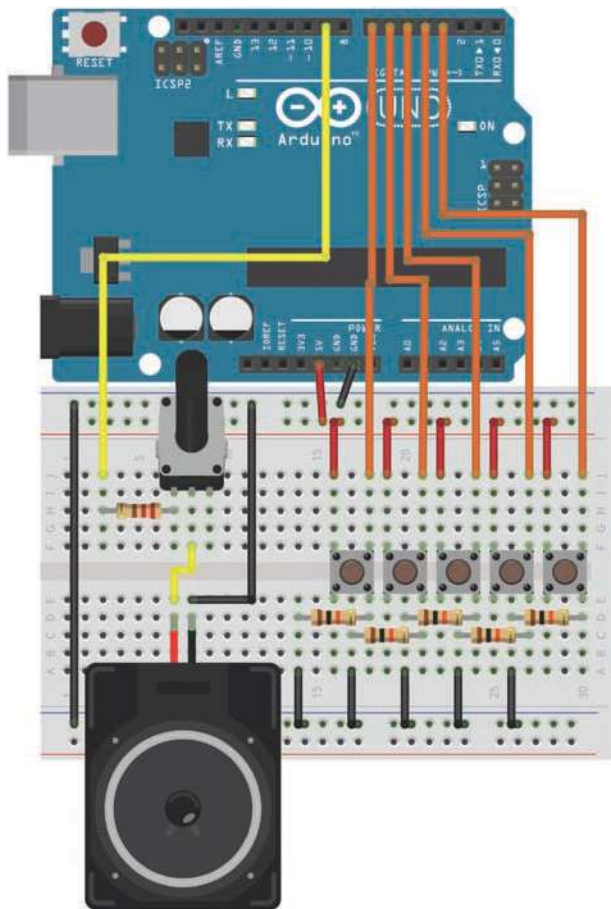


Рис. 6.6. Монтажная схема пианино на Arduino (рисунок создан в программе Fritzing)

Как мы узнали ранее, функция `Serial.begin()` вызывается один раз в начале программы в функции `setup()`, чтобы подготовить последовательный порт для работы. После этого можно беспрепятственно вызывать функции `Serial.print()` и `Serial.println()` для отправки данных в последовательный порт. Единственная разница между этими двумя функциями состоит в том, что функция `Serial.println()` добавляет в конце символ перевода строки, чтобы следующий текст отображался на новой строке. Чтобы поэкспериментировать с этой функциональностью, соберите простую схему, подключив потенциометр к контакту A0 платы Arduino, как показано на рис. 7.6.

Далее создайте новый скетч и скопируйте в него код из листинга 7.1. Данная программа считывает текущее значение выходного сигнала потенциометра и отображает его в окне монитора порта в виде исходного значения и значения в процентах.

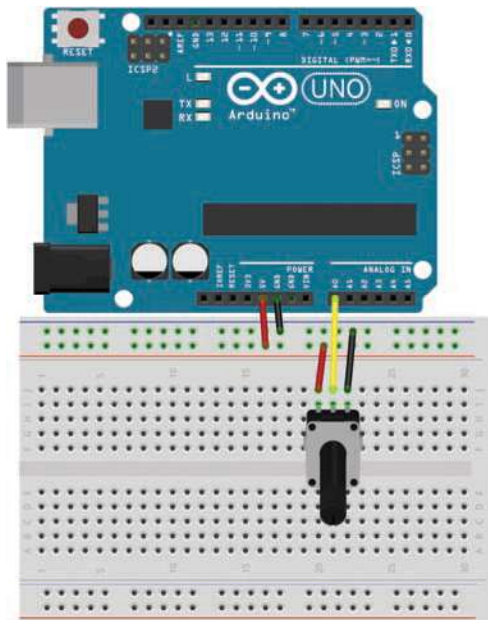


Рис. 7.6. Монтажная схема подключения потенциометра к плате Arduino (рисунок создан в программе Fritzing)

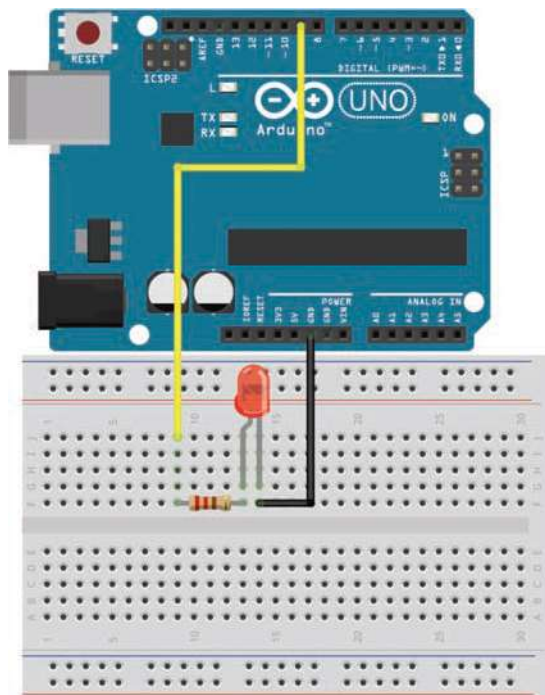


Рис. 7.10. Подключение светодиода к контакту 9 платы Arduino (рисунок создан в программе Fritzing)

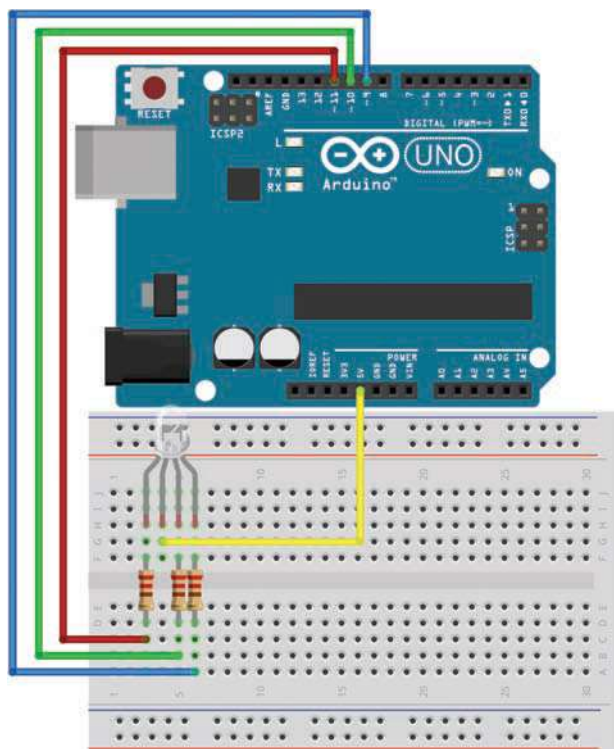
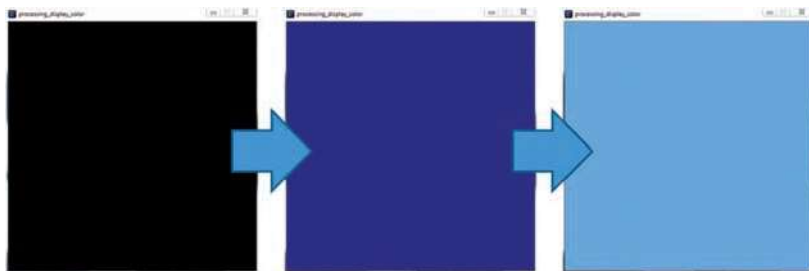


Рис. 7.11. Подключение разноцветного светодиода к плате Arduino (рисунок создан в программе Fritzing)



Увеличивающиеся аналоговые значения

Рис. 7.13. Пример исполнения скетча Processing для изменения цвета окна

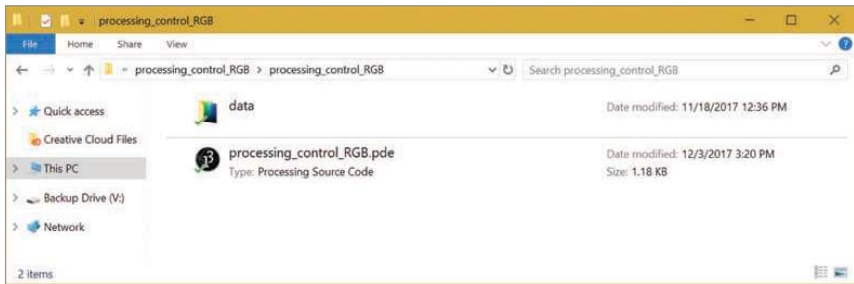


Рис. 7.14. Структура папок проекта Processing

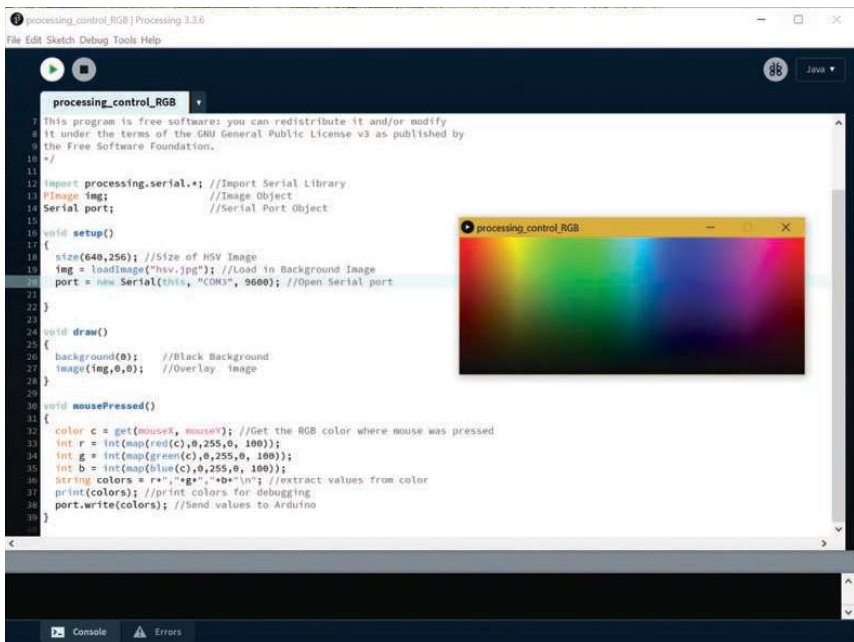


Рис. 7.15. Окно программы Processing для выбора цвета

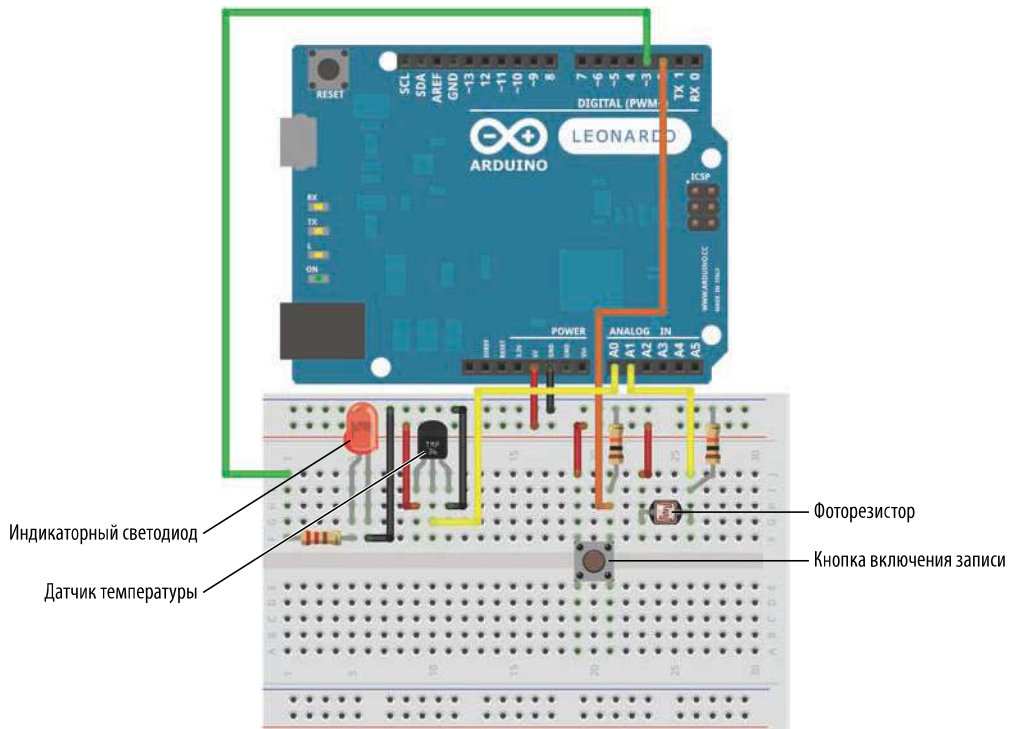


Рис. 8.1. Монтажная схема датчика температуры и освещенности (рисунок создан в программе Fritzing)

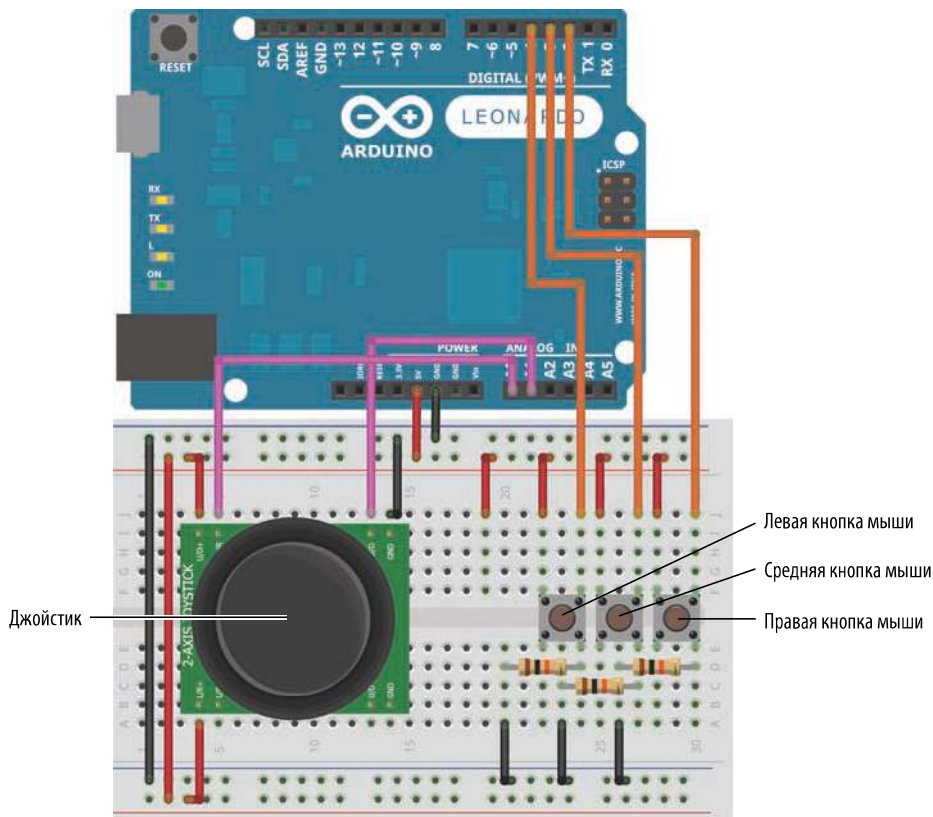


Рис. 8.2. Схема для эмуляции мыши с помощью Arduino Leonardo (рисунок создан в программе Fritzing)

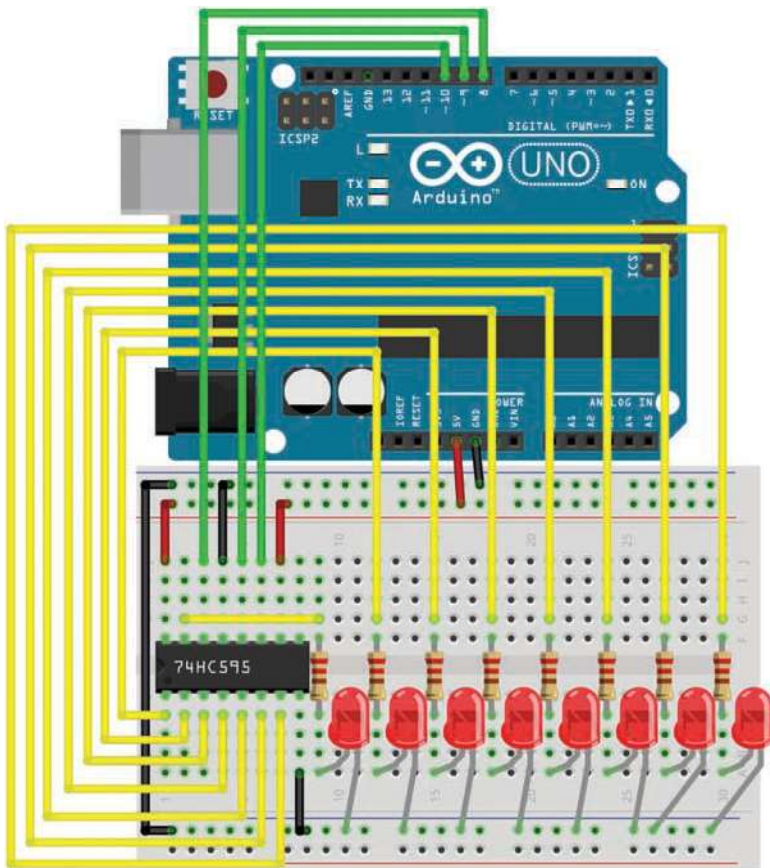


Рис. 9.4. Монтажная схема подключения к плате Arduino сдвигового регистра и восьми светодиодов (рисунок создан в программе Fritzing)

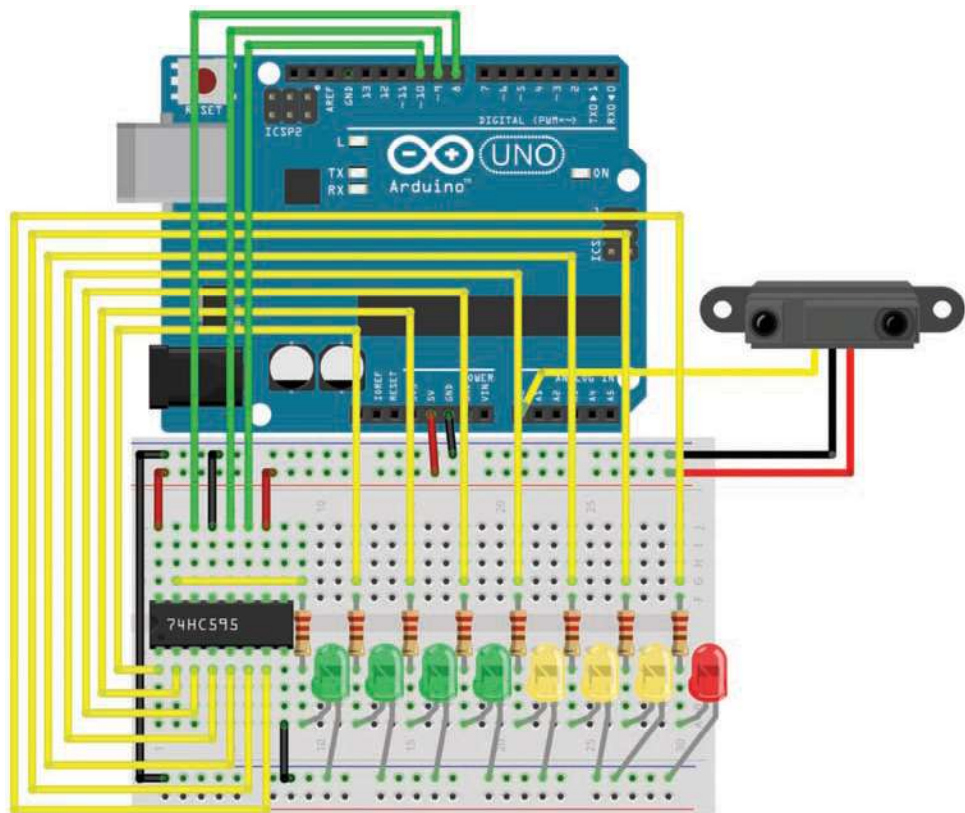


Рис. 9.7. Монтажная схема гистограммы, реагирующей на показания датчика расстояния (рисунок создан в программе Fritzing)

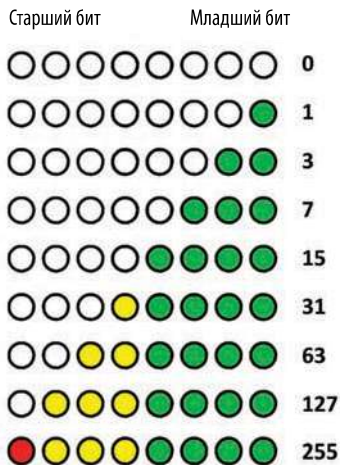


Рис. 9.8. Десятичные значения, необходимые для включения требуемого количества светодиодов

иллюстрация десятичных значений, соответствующих включению необходимого количества светодиодов.

Как мы узнали в *главе 3*, диапазон пригодных для использования значений, выдаваемых ИК-датчиком расстояния, может быть меньше, чем весь доступный 10-битовый диапазон. (У меня максимальное рабочее значение было приблизительно 500, но значение для вашей схемы, скорее всего, будет другим.) Минимальное значение также может быть больше нуля. Лучше всего протестировать ваш датчик в рабочих условиях и установить подходящие значения на практике. Все десятичные представления состояний гистограммы можно сохранить в массиве из

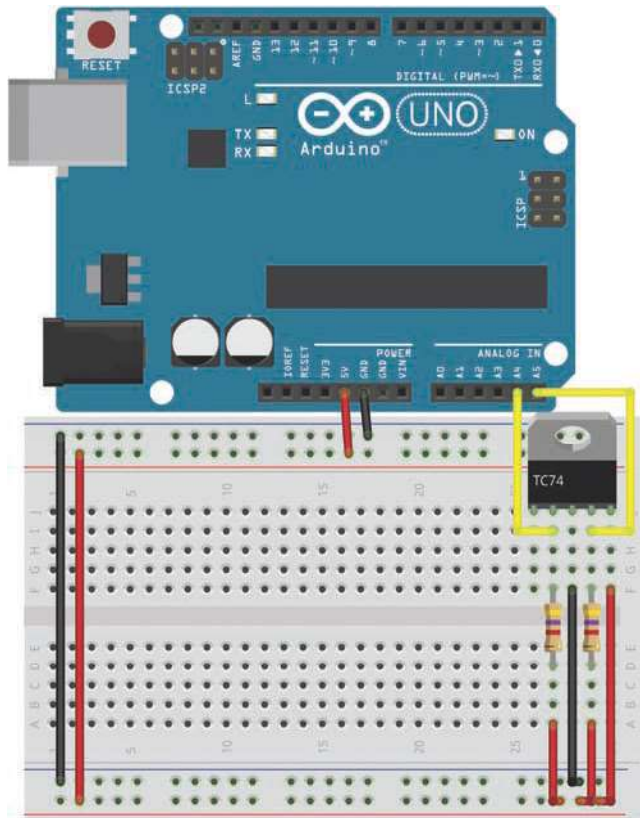


Рис. 10.4. Подключение датчика температуры TC74 к плате Arduino (рисунок создан в программе Fritzing)

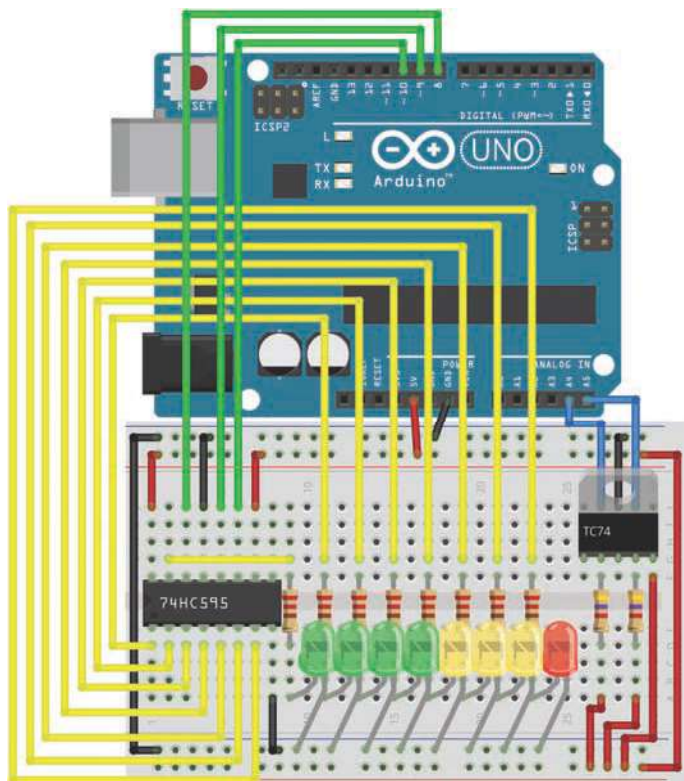


Рис. 10.8. Монтажная схема датчика температуры с выводом на гистограмму (рисунок создан в программе Fritzing)

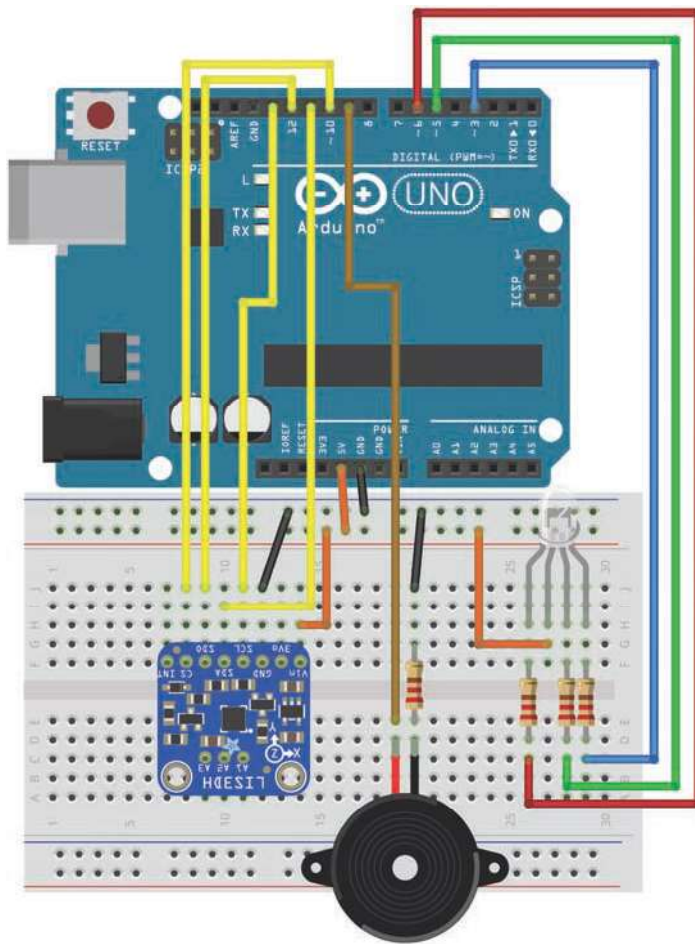


Рис. 11.10. Подключение пьезоэлемента для создания схемы музыкального инструмента (рисунк созд в программе Fritzing)

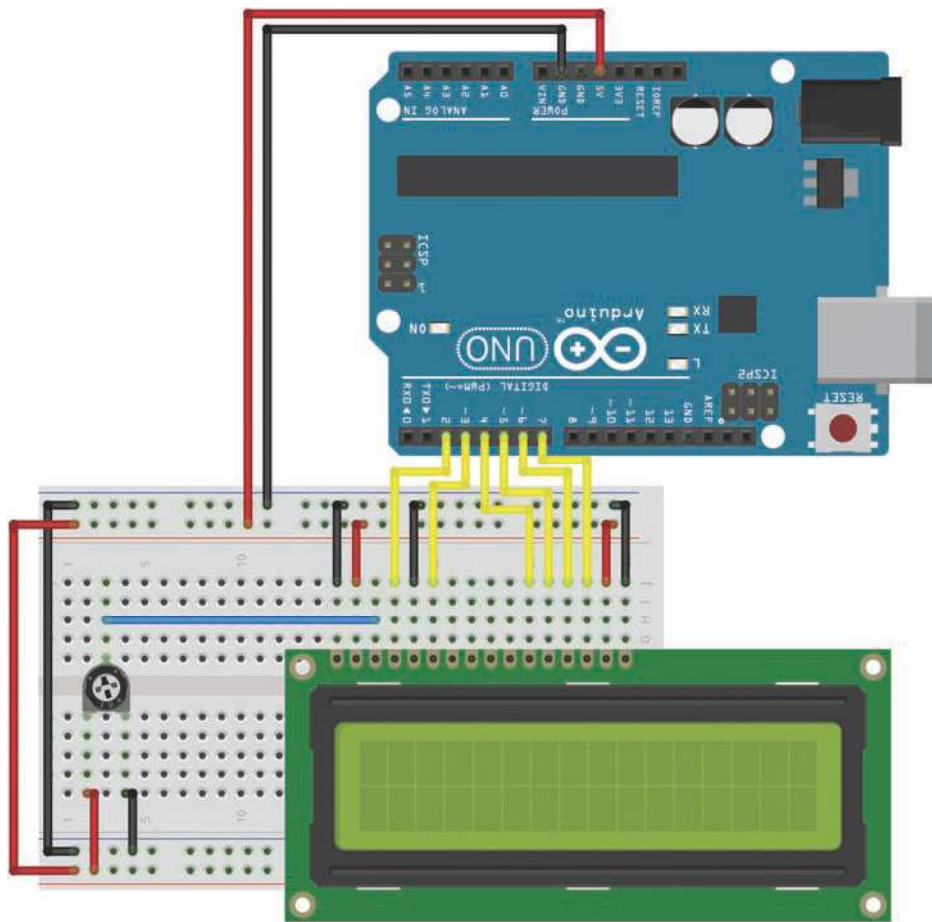


Рис. 12.2. Подключение ЖКД к плате Arduino (рисунок создан в программе Fritzing)

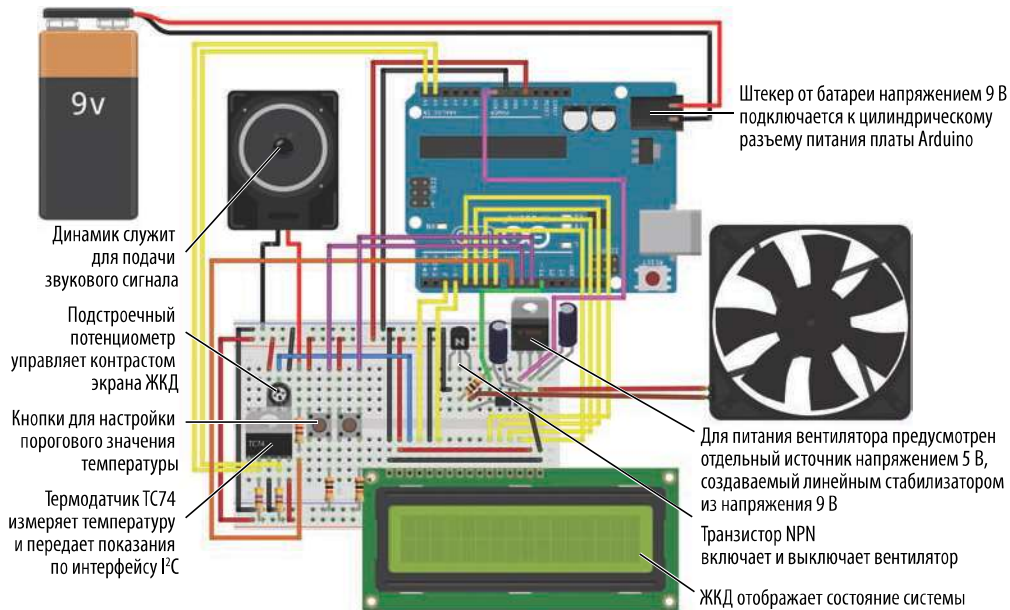


Рис. 12.5. Монтажная схема регулятора температуры с ЖКД (рисунок создан в программе Fritzing)

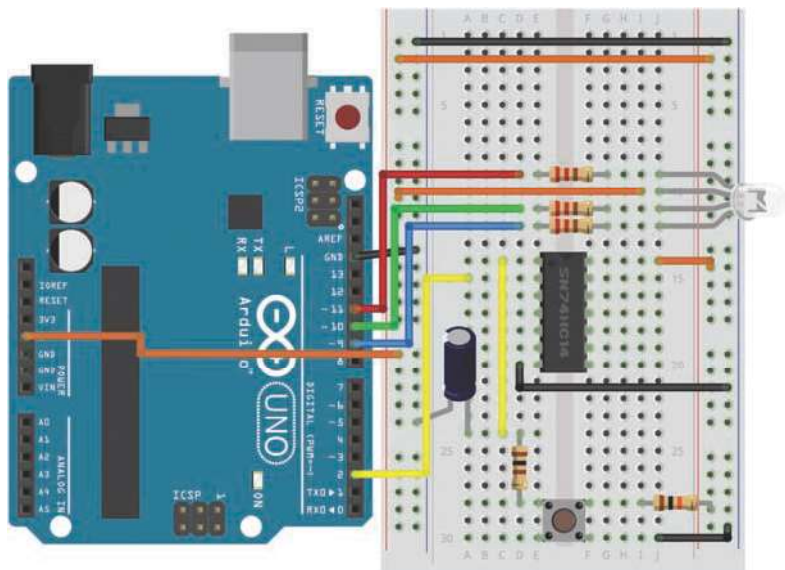


Рис. 13.8. Монтажная схема аппаратной защиты от дребезга (рисунок создан в программе Fritzing)

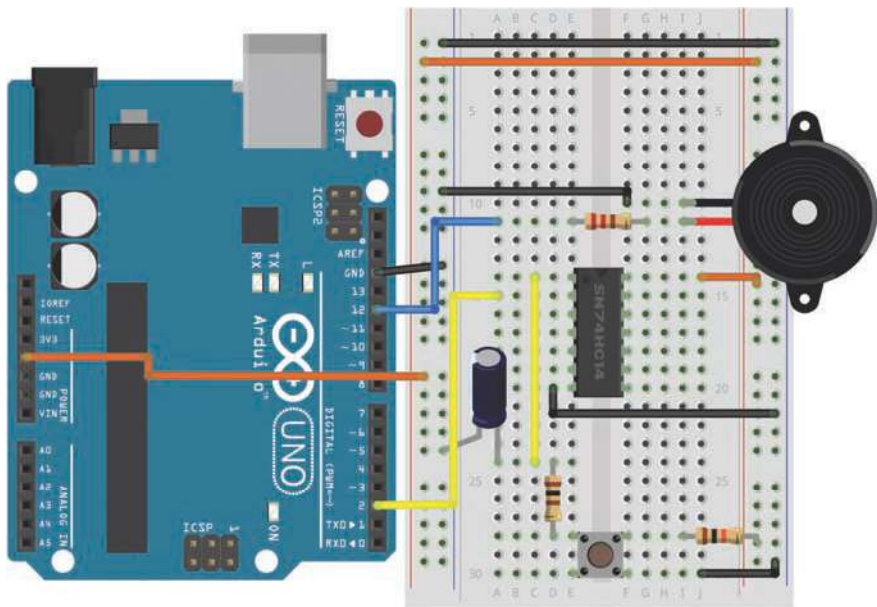


Рис. 13.9. Монтажная схема музыкального инструмента (рисунок создан в программе Fritzing)

Это такой же микроконтроллер производства компании Atmel/Microchip, который установлен на плате Leonardo, в большой мере совместимой с Feather, используемой в этой главе. Но между этими платами есть одно важное различие: плата Feather работает с логическими уровнями напряжением 3,3 В. Иными словами, для этой платы высоким логическим уровнем является напряжение 3,3 В, а не 5 В, как для платы Leonardo. Чтобы подключить к этой плате датчики или периферийные устройства с высоким логическим уровнем 5 В, нужно осуществить преобразование уровней, или же просто выбрать устройства с рабочим напряжением 3,3 В. Плата Feather — это, по сути, просто плата Arduino на микроконтроллере 32U4, объединенная с модулем BTLE в конструкции. В действительности такая компоновка ничем не отличается от платы Arduino с подключенным шилдом BTLE, но обеспечивает более компактное решение благодаря размещению обоих элементов на одной печатной плате.

Считывание датчика через канал BTLE

Для начала возьмем самый простой датчик — потенциометр. Подключите его к плате Feather, как показано на монтажной схеме на рис. 16.1. В частности, внешние контакты потенциометра подключаются к шинам 3,3 В и земли

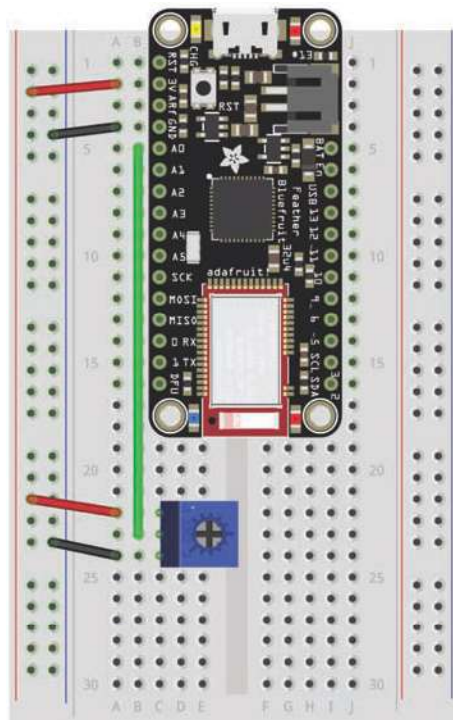


Рис. 16.1. Плата Feather BTLE с подключенным к ней потенциометром (рисунок создан в программе Fritzing)

можно реализовать простейшую систему распознавания команд на естественном языке.

Более сложные методы обработки естественного языка, подобные выполняемым приложением виртуального цифрового помощника с речевым управлением на вашем смартфоне, используют машинное обучение и сложные языковые модели, которые выходят за рамки этой книги. Но, просто проверяя на наличие определенных ключевых слов в строке и не требуя точного совпадения всей строки, можно будет отправлять команды типа «Включить светодиод» и «Выключить светодиод», чтобы реализовать простую систему управления Arduino с помощью команд естественного языка.

На каждой итерации главного цикла `loop()` можно вызывать функцию `btle.available()` для проверки наличия входящих данных (листинг 16.6). При

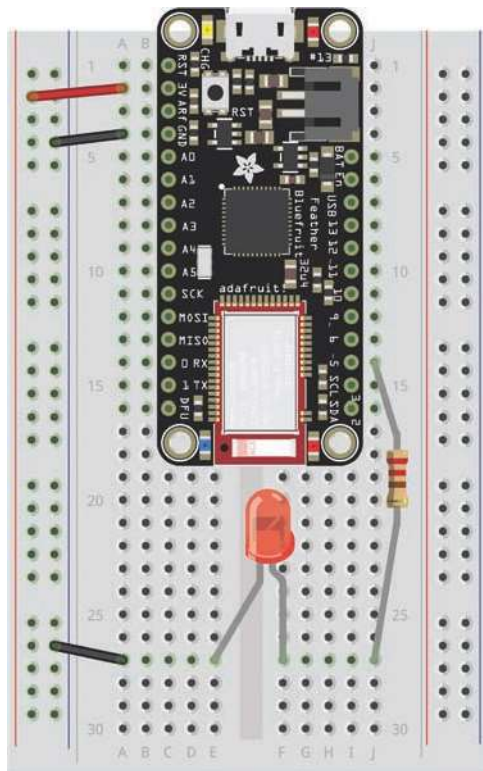


Рис. 16.9. Плата Feather BTLE с подключенным к ней светодиодом (рисунок создан в программе Fritzing)

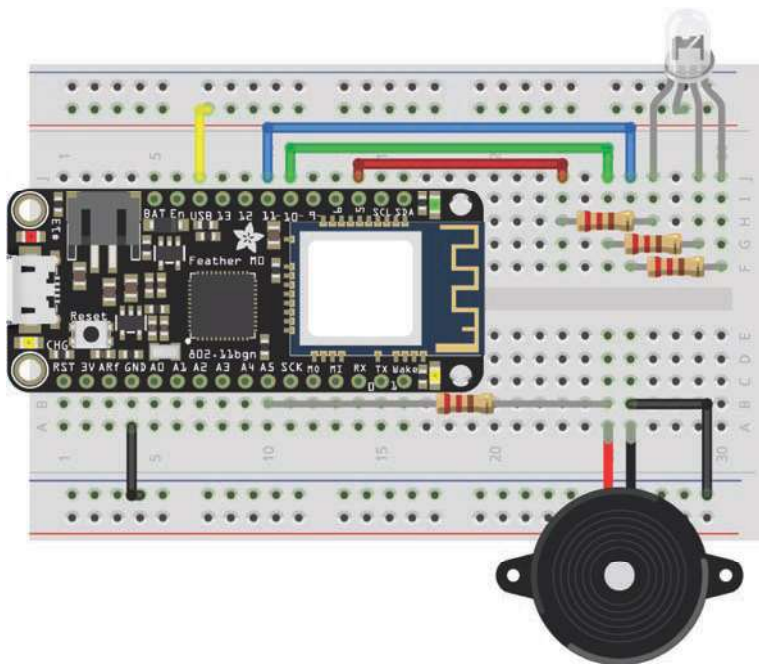


Рис. 17.2. Веб-сервер на плате Arduino Feather с подключенными к ней RGB-светодиодом и пьезозуммером (рисунок создан в программе Fritzing)

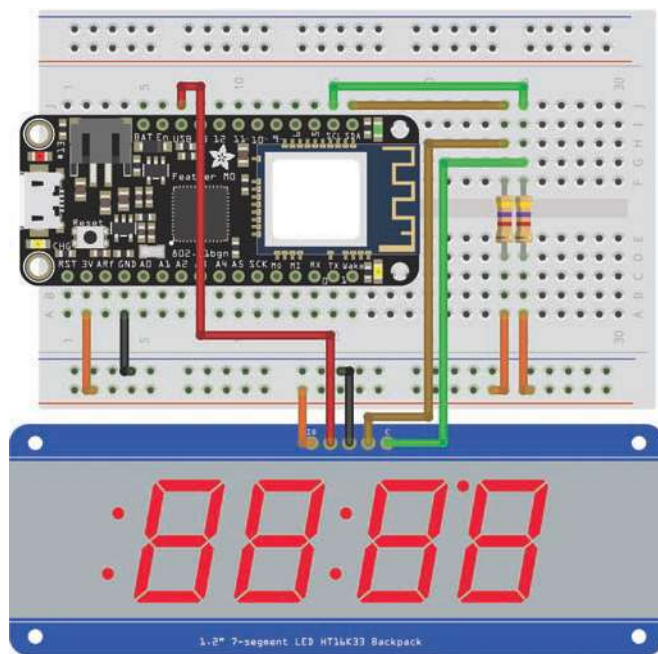


Рис. 17.19. Плата Arduino Feather с подключенным светодиодным дисплеем (рисунок создан в программе Fritzing)