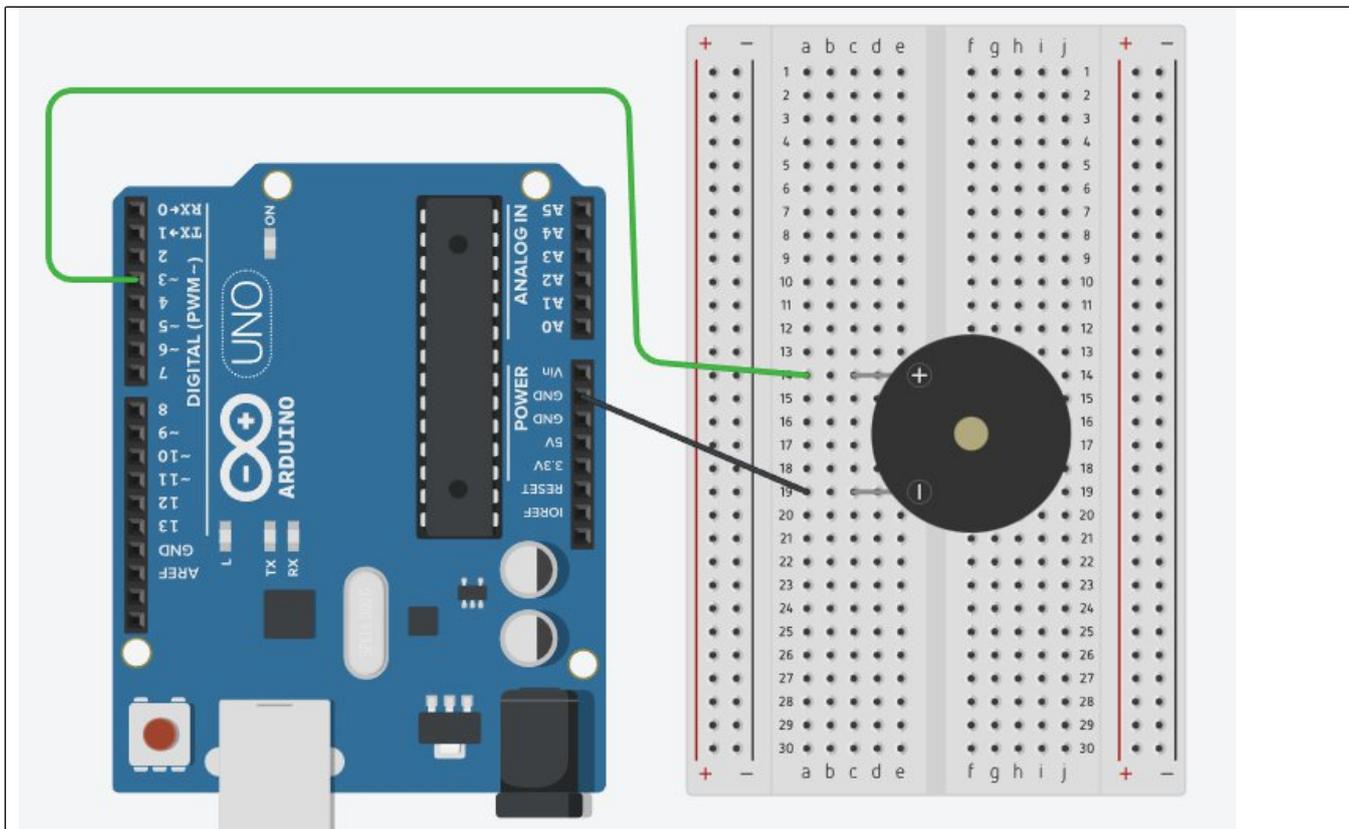


Лабораторная работа № 5. Пьезодинамик

Цель: Знакомство с принципом работы пьезодинамика

Пьезодинамик — электромеханический преобразователь, одним из разновидностей которого является пьезоизлучатель звука, который также называют пьезодинамиком, просто звонком или английским **buzzer**. Пьезодинамик переводит электрическое напряжение в колебание мембраны. Эти колебания и создают звук.

Подключение пьезодинамика к Arduino.



Рабочая программа

```
Код ▶ Начать моделирование Отправить
```

Текст ▼ [иконка загрузки] [иконка сохранения] AA ▼ 1 (Arduino Uno R3) ▼

```
1 int p = 3; //объявляем переменную с номером пина, на который мы
2 //подключили пьезоэлемент
3 void setup()
4 {
5   pinMode(p, OUTPUT); //объявляем пин как выход
6 }
7 void loop()
8 {
9   tone(p, 500); //включаем на 500 Гц
10  delay(100); //ждем 100 мс
11  tone(p, 1000); //включаем на 1000 Гц
12  delay(100); //ждем 100 мс
13 }
```

Практическая часть

1) Функция генерации прямоугольной волны заданной частоты - `tone()`.

Описание:

Данная функция генерирует на порту вход / выход прямоугольную волну, заданной частоты с 50% рабочим циклом. К порту подключается пьезо или другой динамик для воспроизведения сигнала.

Воспроизводиться одновременно может только один сигнал. При повторном вызове `tone()` с тем же номером порта будет устанавливаться новая частота сигнала.

Синтаксис:

`tone(pin, f)` — либо, `tone(pin, f, d)`

Параметры:

`pin`-номер пина,

`f`- частота сигнала [Гц]

`d` - длительность сигнала [мс] (может быть не задана, в этом случае сигнал будет генерироваться, пока не будет вызвана функция `noTone()`),

2. Функция пропорционального переноса - `map()`,

Синтаксис:

`map(value, x, y, a, b)`

Параметры:

`value` - значение для переноса,

`x` - нижняя граница текущего диапазона,

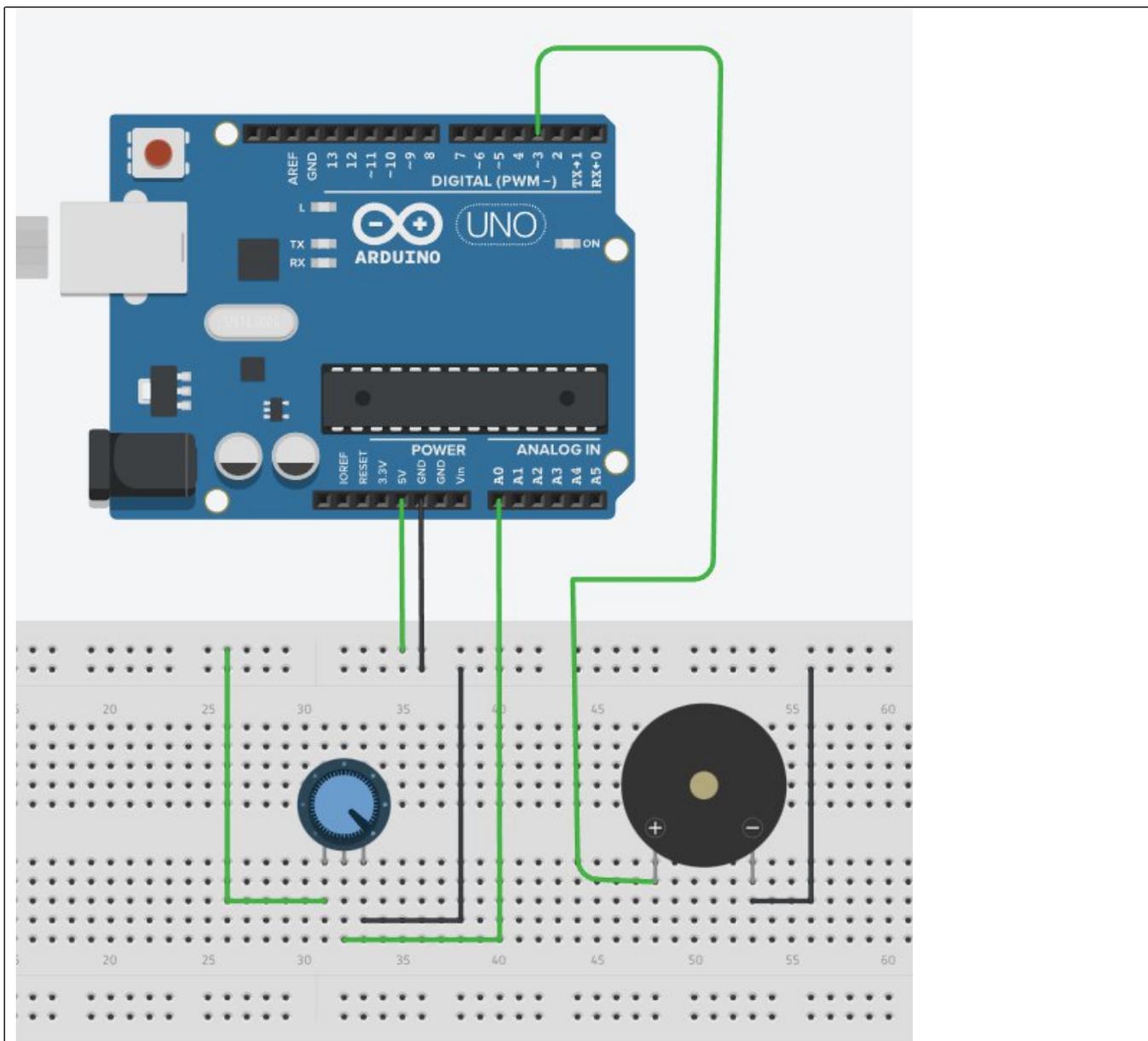
`y` - верхняя граница текущего диапазона,

`a` - нижняя граница нового диапазона, в который переносится значение,

`b` - верхняя граница нового диапазона, в который переносится значение.

Функция возвращает значение в новом диапазоне.

Схема подключения



Рабочая программа

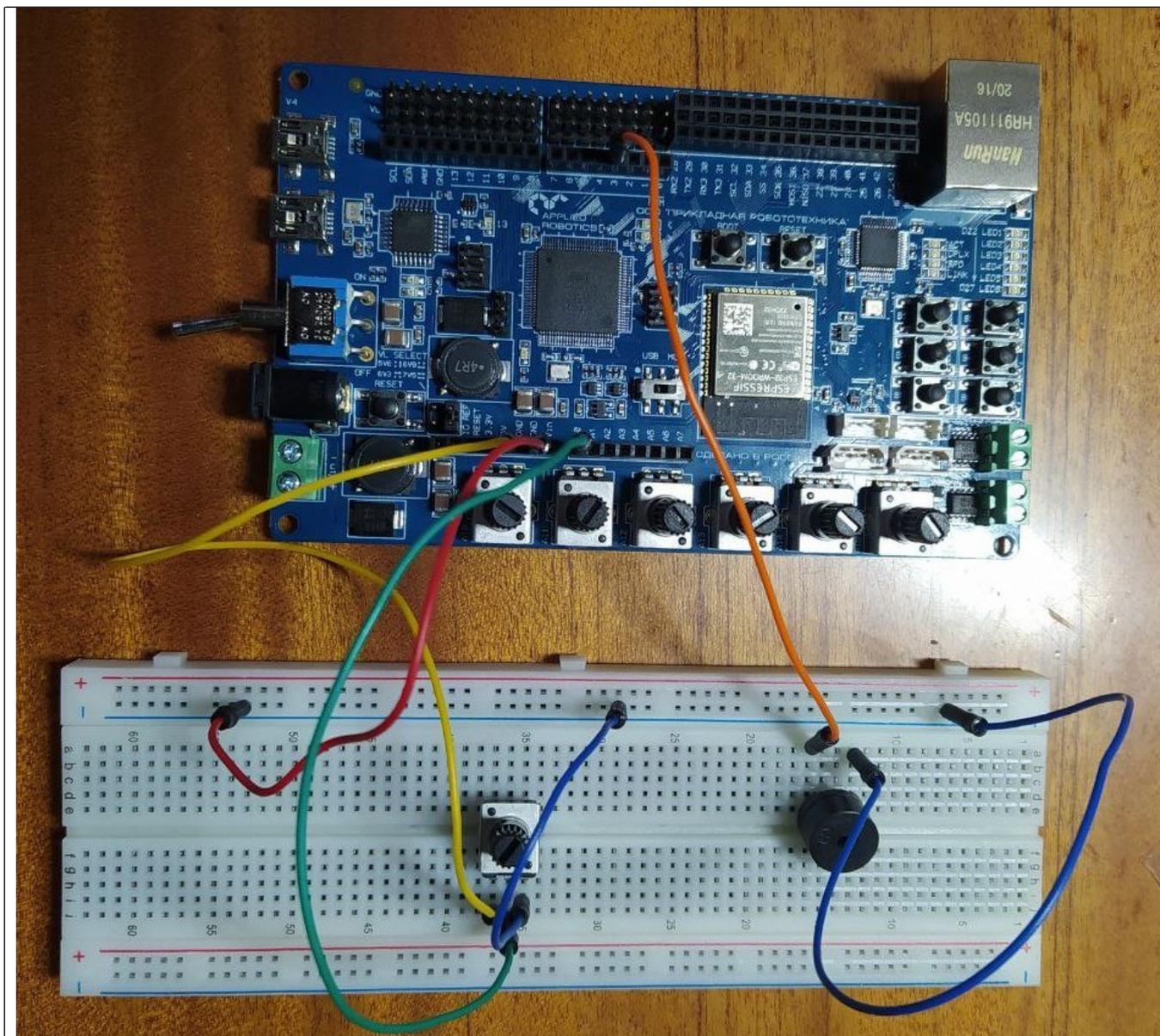
```
Код ▶ Начать моделирование Отправить
```

Текст

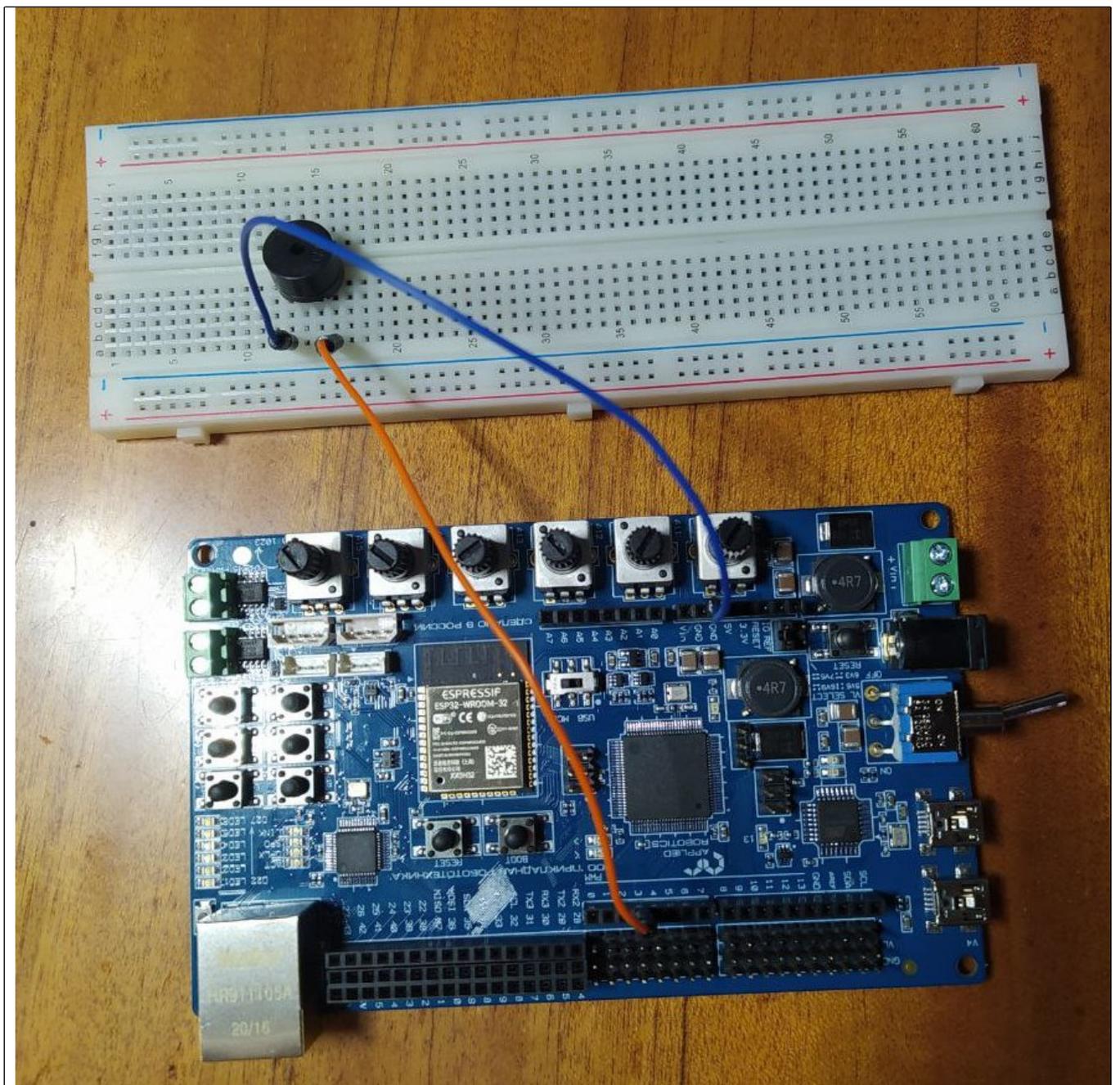
```
1
2 void setup()
3 {
4   pinMode(3, OUTPUT);
5   pinMode(A0, INPUT);
6   Serial.begin(9600);
7 }
8
9 void loop()
10 {
11   int x = analogRead(A0);
12   int y = map(x, 0, 1023, 3500, 4500); // преобразование сигнала
13   Serial.println(y); // печатается значение частоты сигнала
14   tone(3, y, 20); // задается длительность сигнала 20 мс
15 }
```

Покрутите ручку потенциометра и послушайте как изменяется высота сигнала

Схема подключения для микроконтроллера FTmega2560



Составьте схему подключения для микроконтроллера FTmega2560 с использованием встроенного потенциометра и выполните программу.



Рабочая программа

```
void setup()
{
  pinMode(3, OUTPUT);
  pinMode(A10,INPUT);
  Serial.begin(9600);
}

void loop()
{
  int x = analogRead(A10);
  int y = map(x,0, 1023, 3500, 4500); // преобразование сигнала
  Serial.println(y); // печатается значение частоты сигнала
  tone(3, y, 20); // задается длительность сигнала 20 мс
}
```

Список контрольных вопросов:

1. Каким образом пьезодинамик выдаёт звук?
2. Как изображается пьезодинамик на электрической схеме?
3. От чего зависит частота звучания пьезодинамика?
4. Как можно регулировать громкость звучания пьезодинамика?
5. В чём измеряется громкость звука?
6. Как называется максимальная частота, которую может выдать пьезодинамик? В чём она измеряется?
7. Как работает функция tone?
8. Какую скважность имеет подаваемая функцией tone волна?
9. Сколько сигналов может одновременно воспроизводиться функцией tone?
10. Какие параметры имеет функция tone?
11. Для чего нужна команда noTone?
12. Как работает функция map?.
13. Какие параметры имеет функция map?
14. Какое значение возвращает функция map?

Список дополнительных задач:

1. Изменить программу так, чтобы регулировать звучание можно было программно.
2. Изменить диапазон частоты выдаваемого звука.
3. Поменять направление увеличения частоты звука пьезодинамика на потенциометре.
4. Изменить программу и использовать в ней функцию noTone.