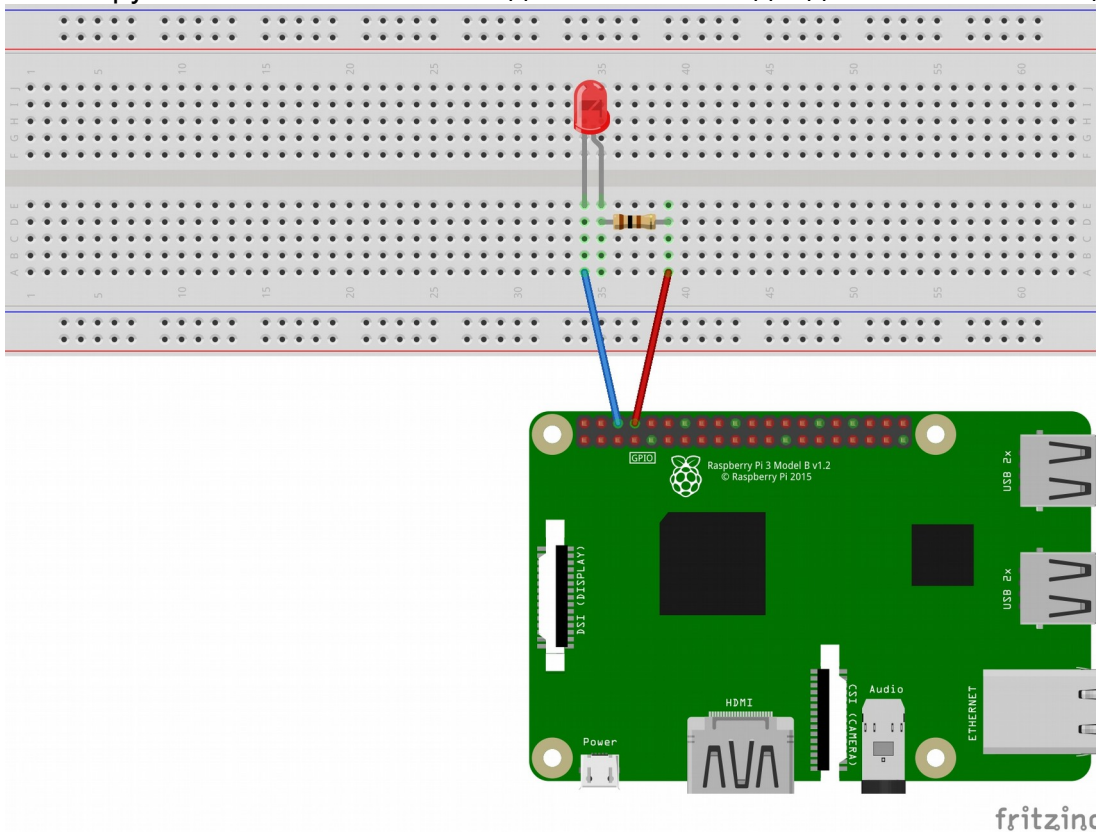


## 03. Разработка программ на языке **Ruby**

Практические задания.

### № 03.1

1. Выключите Raspberry Pi.
2. Смонтируйте на макетной плате подключение светодиода по этой схеме подключения:



Если к физическим контактам [4] (+5V) и [6] (GND) подключен вентилятор, то для подключения светодиода можно воспользоваться физическими контактами [12] (BCM18) и [14] (GND).

3. Внимательно сверьте правильность всех смонтированных соединений на макетной плате со схемой подключения.
4. Включите Raspberry Pi.

### № 03.2

1. Перейдите в каталог `/home/pi/projects`  
`cd ~/projects`
2. Запустите готовый пример программы на **Ruby**, которая управляет светодиодом, для чего выполните в терминальном окне команду  
`./test_led.rb`
3. Понаблюдайте результат её работы. Если светодиод не включается, проверьте правильность подключения, плотность контактов, логический номер контакта в программе.

### № 03.3

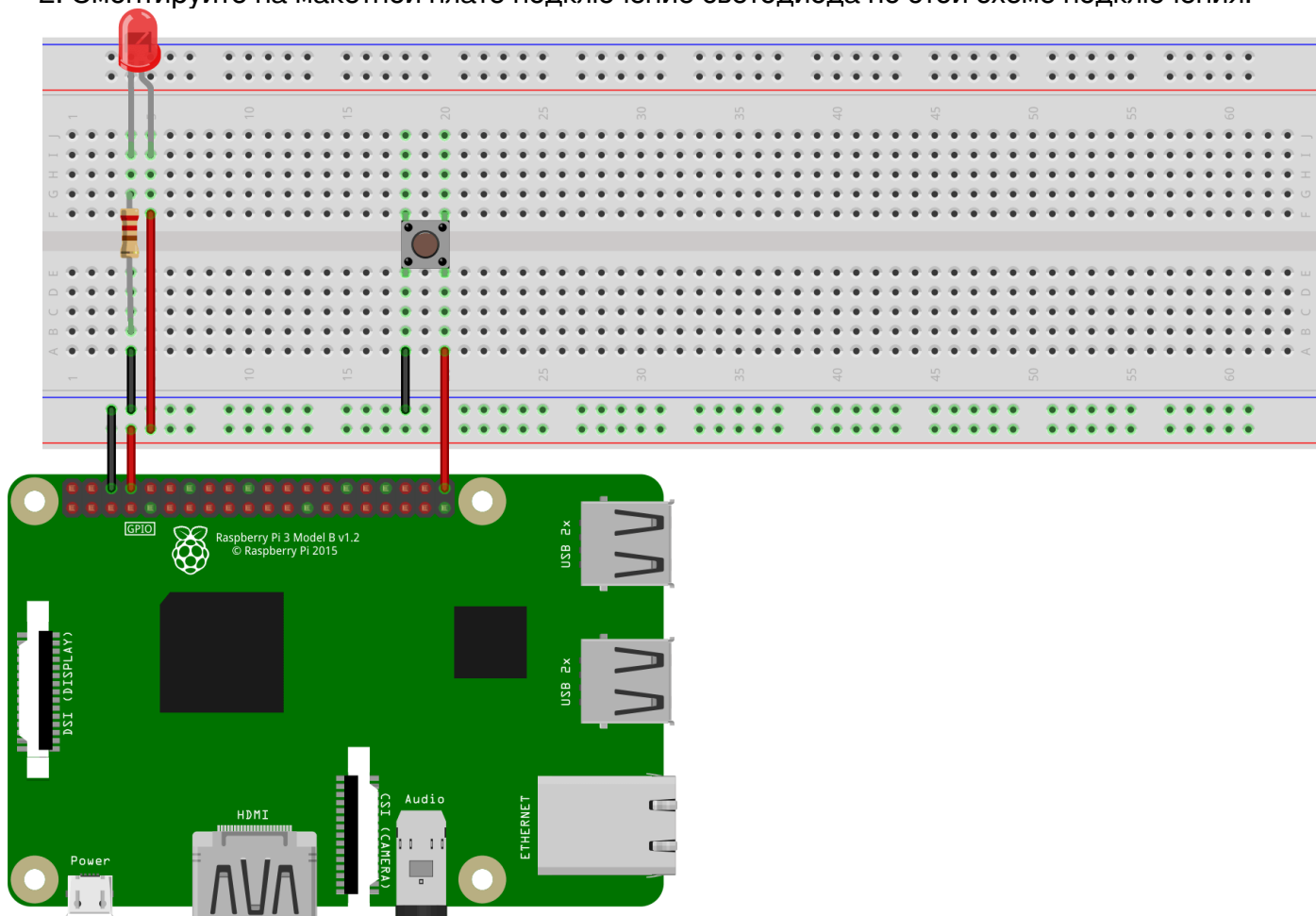
1. В главном меню выберите раздел «Программирование» и запустите из него редактор **Geany**.
2. Откройте командный файл `test_led.rb`, для чего: в меню редактора **Geany** выберите раздел «Файл», а в нём — пункт «Открыть». В открывшемся окне «Открыть файл» выберите домашний каталог `pi`, в нём — подкаталог `projects`, а в нём — файл `test_led.rb`.
3. Запустите его на выполнение из раздела меню «Сборка», пункт «Execute» (выполнить).
4. Понаблюдайте результат её работы в открывшемся терминальном окне и на макетной плате.
5. Закройте терминальное окно.

#### № 03.4

1. В редакторе **Geany** сохраните программу `test_led.rb` как `sos.rb`.
2. Измените её так, чтобы светодиод постоянно подавал световой сигнал о помощи «SOS» азбукой Морзе, как в предыдущем практическом задании.
3. Запустите программу на выполнение и проверьте результат её работы на макетной плате.
4. Измените программу так, чтобы на экран в терминальном окне параллельно со вспышками светодиода выдавались «точки» и «тире».
5. Запустите программу на выполнение и проверьте результат её работы в открывшемся терминальном окне и на макетной плате.

#### № 03.5

1. Выключите Raspberry Pi.
2. Смонтируйте на макетной плате подключение светодиода по этой схеме подключения:



fritzing

3. Создайте в редакторе Geany новый файл `button.rb` и введите в него с клавиатуры текст программы:

```
#!/usr/bin/ruby
```

```

require "button"                # подключить библиотеку button.rb

button = Button.new(21)         # создать объект класса Button

button.wait_for_press           # ожидать нажатия на кнопку
if button.long_press?           # было длинное нажатие?
  print "Длинное!\n"
else
  print "Короткое.\n"
end

button.wait_for_presses(2)      # ожидать 2 нажатия на кнопку
if button.double_press?         # было двойное нажатие?
  print "Двойное...\n"
else                             # было одиночное нажатие
  print "2 одинарных\n"
end

```

4. Запустите программу на выполнение и проверьте результат её работы.
5. Измените программу так, чтобы по короткому нажатию на кнопку светодиод включался, а по долгому — выключался.
6. Проверьте результат выполнения программы.
7. Сохраните эту программу под именем **control.rb**.
8. Измените программу так, чтобы с помощью разных типов нажатий на кнопку можно было перезагружать и выключать Raspberry Pi.

#### № 03.6 — Домашнее задание

1. Какие приспособления из самых разных предметов могут служить «кнопкой»?