

Набор «Малина»

Электронная версия брошюры из набора «Малина». Здесь собраны все исходные коды экспериментов, подсказки и хаки по прохождению набора.



Подключение и запуск

Если под рукой нет внешнего монитора, [настрой SSH](#).

Настройка Wi-Fi

Подключись к Raspberry Pi по SSH.

Открой в текстовом редакторе «nano» файл настроек Wi-Fi. Для этого введи команду

```
~ $ sudo nano /etc/wpa_supplicant/wpa_supplicant.conf
```

и нажми Enter.

Откроется файл, в конце добавь параметры сети.

```
network={
    ssid="имя твоего wi-fi"
    psk="пароль твоего wi-fi"
}
```

Закрой файл сочетанием клавиш ctrl+X (cmd+X для Mac OS) на клавиатуре. Редактор спросит, сохранить ли изменения. Нажми клавишу Y (да), а затем Enter. Выполни перезагрузку системы командой `sudo reboot`

```
~ $ sudo reboot
```

После перезагрузки отсоедини кабель Ethernet и работай с Расберри через Wi-Fi.

Эксперименты

3. Маячок

```
i = 0
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)
GPIO.setup(17, GPIO.OUT)

i = 0
while (i < 5):
    i = i + 1
    time.sleep(0.5)
    GPIO.output(17, GPIO.HIGH)
    time.sleep(0.5)
    GPIO.output(17, GPIO.LOW)

GPIO.cleanup()
```

4. Выключатель

```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)
GPIO.setup(2, GPIO.IN)

while (True):
    time.sleep(0.5)
    button = GPIO.input(2)
    print(button)
```

5. Передай переменную button в функцию output, чтобы управлять светодиодом.

```
while (True):
    button = GPIO.input(2)
    GPIO.output(24, button)
```

Не забудь добавить строку инициализации пина на выход.

```
GPIO.setup(24, GPIO.OUT)
```

5. Переключатель

```
import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BCM)
GPIO.setup(8, GPIO.IN)
GPIO.setup(24, GPIO.OUT)
GPIO.setup(26, GPIO.OUT)

while (True):
    button = GPIO.input(8)
    if (button == False):
        GPIO.output(24, GPIO.HIGH)
        GPIO.output(26, GPIO.LOW)
    else:
        GPIO.output(24, GPIO.LOW)
        GPIO.output(26, GPIO.HIGH)
```

6. Управление яркостью

```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)
GPIO.setup(18, GPIO.OUT)

pwm = GPIO.PWM(18, 1000)
dutyCycle = 50
pwm.start(dutyCycle)

while (True):
    time.sleep(0.01)
    dutyCycle = dutyCycle + 1
    if (dutyCycle > 100):
        dutyCycle = 0
    pwm.ChangeDutyCycle(dutyCycle)
```

7. Панель управления светом

```
import RPi.GPIO as GPIO

def isPressed(btn, led):
    if (GPIO.input(btn) == False):
        GPIO.output(led, GPIO.HIGH)
    else:
        GPIO.output(led, GPIO.LOW)

button1 = 3
button2 = 4
led1 = 14
led2 = 15

GPIO.setmode(GPIO.BCM)
GPIO.setup(button1, GPIO.IN)
GPIO.setup(button2, GPIO.IN)
GPIO.setup(led1, GPIO.OUT)
GPIO.setup(led2, GPIO.OUT)

while (True):
    isPressed(button1, led1)
    isPressed(button2, led2)
```

8. Массивная оптимизация

```
import RPi.GPIO as GPIO
import time
```

```
GPIO.setmode(GPIO.BCM)
leds = [12, 13, 14, 18]
for led in leds:
    GPIO.setup(led, GPIO.OUT)
    GPIO.output(led, GPIO.HIGH)
time.sleep(3)
for led in leds:
    GPIO.output(led, GPIO.LOW)
GPIO.cleanup()
```

9. Web-сервер

```
from flask import Flask
app = Flask('simpleServer')
@app.route('/')
def index():
    return 'Hello Amperka'
app.run(debug=True, port=3000, host='0.0.0.0')
```

Если при перезапуске сервера консоль ответит сообщением «socket.error: [Errno 48] Address already in use», набери команду

```
ps -fA | grep python
```

В консоль выведется список запущенных процессов. Найди строку с файлом `landing.py`. Во второй колонке будет указан номер процесса. Отключи его командой

```
kill -9 xxx
```

Где `xxx` — номер процесса. Команда `kill` «убивает» процесс, иногда это называют «убить девяткой».

Запусти сервер заново.

10. Landing page

Ссылка на архив с файлами web-сервера.

http://amperka.github.io/malina_support/web-server.zip

```
from flask import Flask, send_file
app = Flask('landingPage')
@app.route('/')
def index():
    return send_file('landing.html')
@app.route('/images/<filename>')
def get_image(filename):
    return send_file('images/'+filename)
app.run(debug=True, port=3000, host='0.0.0.0')
```

11. Интернет свет

```
from flask import Flask, send_file
import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BCM)
led = 18
GPIO.setup(led, GPIO.OUT)
app = Flask('lightControl')

@app.route('/')
def index():
    return send_file('light.html')

@app.route('/images/<filename>')
def get_image(filename):
    return send_file('images/'+filename)

@app.route('/turnOn')
def turnOn():
    GPIO.output(led, GPIO.HIGH)
    return 'turnedOn'

@app.route('/turnOff')
def turnOff():
    GPIO.output(led, GPIO.LOW)
    return 'turnedOff'

app.run(debug=True, port=3000, host='0.0.0.0')
```

12. Обратная связь

```
from flask import Flask, send_file
from flask_socketio import SocketIO
import RPi.GPIO as GPIO

app = Flask('feedback')
socketio = SocketIO(app)

GPIO.setmode(GPIO.BCM)
btn = 2
GPIO.setup(btn, GPIO.IN)

@app.route('/')
def index():
    return send_file('feedback.html')

@app.route('/images/<filename>')
def get_image(filename):
    return send_file('images/'+filename)

@socketio.on('isPressed')
def checkButton(receivedData):
    if (GPIO.input(btn) == False):
        socketio.emit('button', 'pressed')
    else:
        socketio.emit('button', 'released')

socketio.run(app, port=3000, host='0.0.0.0')
```

13. Погодный фиджет

```
import requests, json

url = 'http://api.openweathermap.org/data/2.5/forecast'
```

```

payload = {
    'lat': 'широта_твоего_города',
    'lon': 'долгота_твоего_города',
    'units': 'metric',
    'appid': 'твой_ключ'
}

res = requests.get(url, params=payload)
data = json.loads(res.text)
next3hours = data['list'][0]

if ('clouds' in next3hours):
    print 'clouds:', next3hours['clouds']['all'], '%'
if ('rain' in next3hours):
    print 'rain:', next3hours['rain']['3h'], 'mm'
if ('snow' in next3hours):
    print 'snow:', next3hours['snow']['3h'], 'mm'

print 'temp:', next3hours['main']['temp'], 'C'

```

Наш вариант погодного фиджета:

amperka.github.io/malina_support/weather.py

14. Бот ВКонтакте

```

# -*- coding: utf-8 -*-
import time
import vk_api

vk = vk_api.VkApi(token = 'твой_токен')

param = {
    'out': 0,
    'count': 1,
    'time_offset': 5
}

def write_msg(user_id, msg):
    vk.method('messages.send', {
        'user_id': user_id,
        'message': msg
    })

while True:
    response = vk.method('messages.get', param)
    items = response['items']
    if items:
        param['last_message_id'] = items[0]['id']
        for i in items:
            write_msg(i['user_id'], i['body'])
    time.sleep(1)

```

15. Автозапуск и расписания

Одной строкой можно выполнить сразу несколько команд.

```
@reboot cd web-server; python light.py
```

— перейти в директорию web-server и выполнить из неё скрипт light.py

17. Кинотеатр

amperka.github.io/malina_support/omx-web.zip

18. Поставь торрент заранее

```
# -*- coding: utf-8 -*-
import time
import vk_api
from deluge_client import DelugeRPCClient

vk = vk_api.VkApi(token = 'твой_токен')
values = {'out': 0, 'count': 1, 'time_offset': 60}

client = DelugeRPCClient(
    '127.0.0.1',
    58846,
    'pi',
    'raspberry'
)

client.connect()

def write_msg(user_id, s):
    vk.method('messages.send', {
        'user_id': user_id,
        'message': s
    })

while True:
    response = vk.method('messages.get', values)
    if response['items']:
        item = response['items'][0]
        values['last_message_id'] = item['id']
        url = item['body']
        res = client.call('core.add_torrent_url', url, {
            'move_completed_path': '/home/pi/Torrents'
        })
        write_msg(item[u'user_id'], u'ок!')
        time.sleep(1)
```

Включить все светодиоды на Облаке

```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)
leds = [10, 12, 13, 14, 15, 16, 17, 18, 19, 21, 24, 26]

for led in leds:
    GPIO.setup(led, GPIO.OUT)
    GPIO.output(led, GPIO.HIGH)

time.sleep(3)

for led in leds:
    GPIO.output(led, GPIO.LOW)

GPIO.cleanup()
```