

Конструктор для сборки робота с камерой

Модель: Yahboom Raspberry Pi 4B G1 Tank



Руководство по эксплуатации и программированию

Содержание

1 Дистанционное управление.....	3
2 Подключение и использование камеры	10
3 Доступ к системе raspberry pi.....	12
4 Базовый урок: color_led.....	13



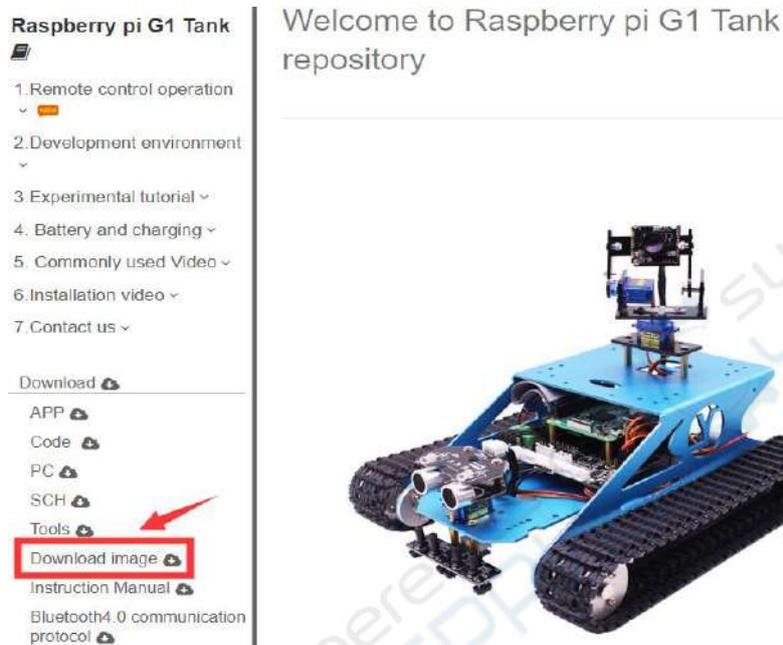
1 ДИСТАНЦИОННОЕ УПРАВЛЕНИЕ

Имя пользователя: pi

Пароль: yahboom

IP: 192.168.50.1

1. После сборки робота G1 Tank следует скачать приложение Raspberry Pi. Для этого перейдите по ссылке <https://www.yahboom.net/study/G1-T-PI>, выберите пункт меню «Download ZIP» и сохраните архив на жестком диске компьютера.



2. Распакуйте архив **Yahboom_G1_Tank_image.zip**. В результате распаковки будет создана папка **Yahboom_G1_Tank_image** с файлом **Yahboom_G1_Tank_image.img**.

3. Скопируйте этот файл на SD-карту Raspberry Pi.

Процесс копирования файла с образом описан на сайте по указанной ниже ссылке. Внимательно прочитайте приведенную на сайте информацию.



Raspberry pi G1 Tank



1. Remote control operation
2. Development environment
3. Experimental tutorial
4. Battery and charging
5. Commonly used Video
6. Installation video
7. Contact us
8. APP Download

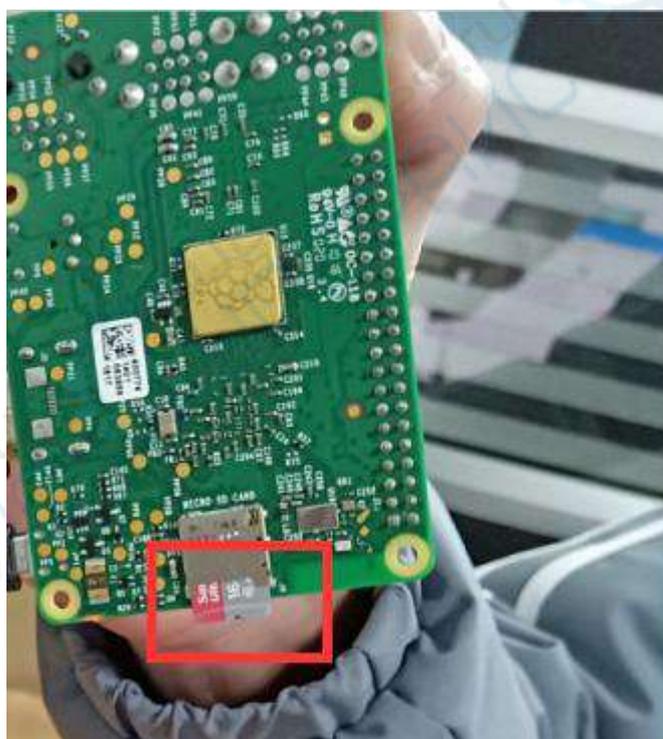
Download

- APP(Android)
- Code
- PC
- Expansion Board Manual
- Tools
- Download image
- Instruction Manual
- Bluetooth4.0 communication protocol
- USB camera firmware

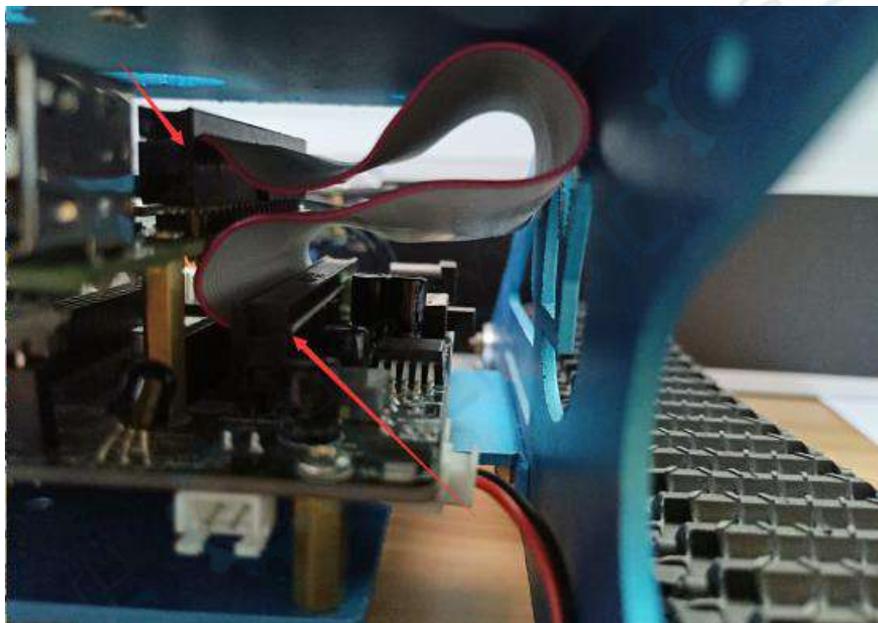
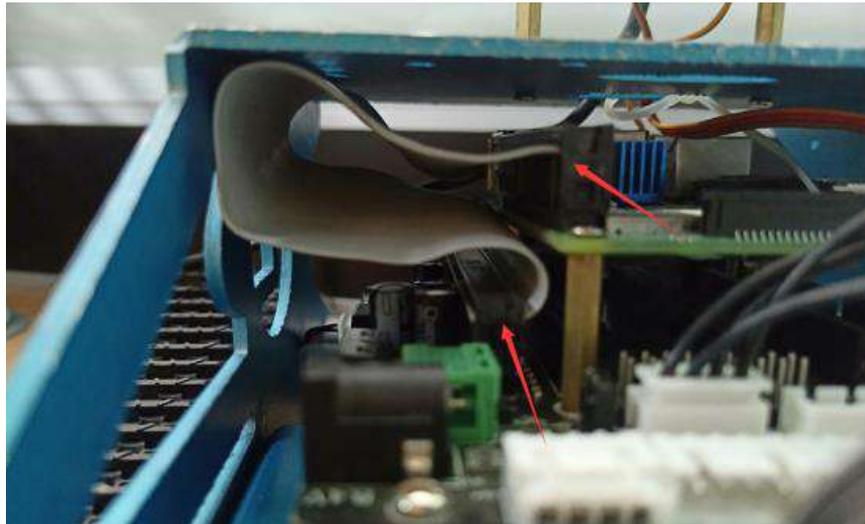
Welcome to Raspberry pi G1 Tank repository



4. После завершения копирования вставьте SD-карту в слот на задней части Raspberry Pi, как показано на рисунке.



5. Подключите плату Raspberry Pi к плате расширения 4WD 40-контактной шиной, как показано на рисунке (обязательно проверьте правильность подключения).



6. Пользователи Android могут отсканировать QR-код или найти приложение «YahboomRobot» в Google Play Market.

Пользователи iOS могут отсканировать QR-код камерой или найти приложение «YahboomRobot» в App Store.



Примечание: после установки приложения необходимо предоставить все разрешения для приложения YahboomRobot.

7. После установки приложения активируйте Bluetooth в телефоне и включите питание робота. Красный индикатор модуля Bluetooth начинает мигать.

Примечание: необходимо выждать паузу, поскольку роботу требуется некоторое время для запуска модулей. Тройной звуковой сигнал и вращение сервопривода сигнализирует о запуске робота, после чего можно перейти к следующему шагу.

8. Запустите приложение **YahboomRobot**. В открывшемся интерфейсе нажмите кнопку меню в левом верхнем углу и выберите нужное устройство.



9. Выберите «G1 Tank» для открытия интерфейса удаленного управления.



10. В приложении откроется приведенное ниже окно. Нажмите на значок «Bluetooth» в правом верхнем углу окна.



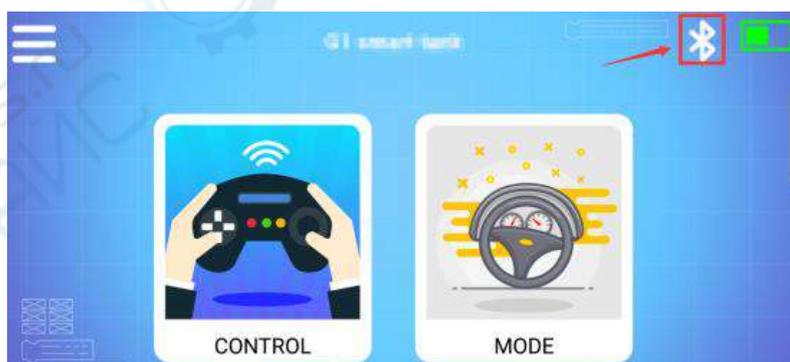
11. Телефон будет автоматически подключаться к роботу, когда будет находится рядом с ним.



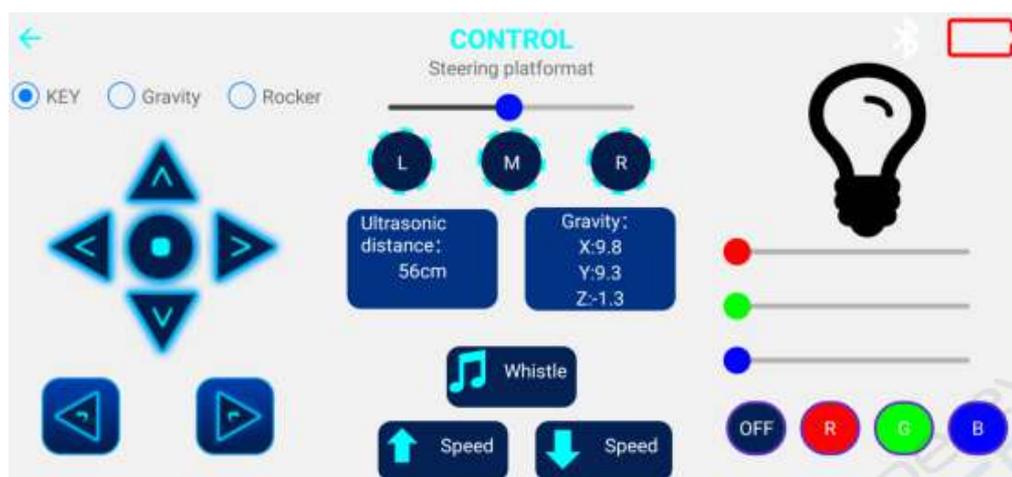
12. После успешного подключения на экране отобразится соответствующее сообщение, а на модуле Bluetooth будет гореть красный индикатор. Нажмите кнопку «OK».



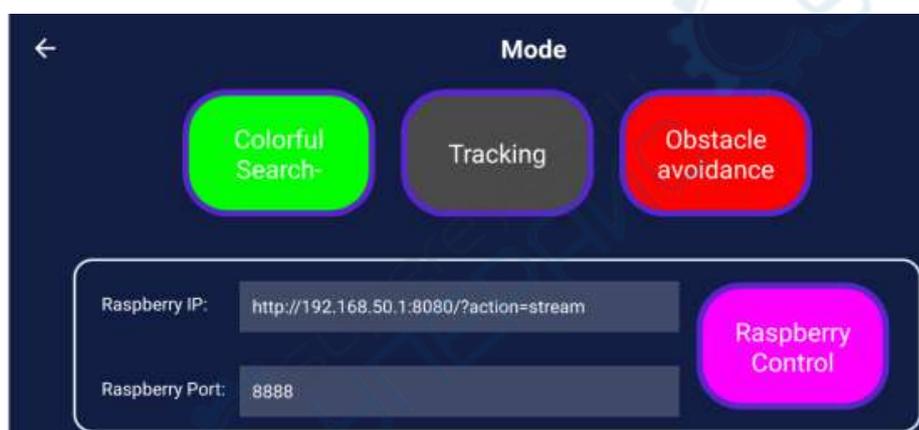
Откроется следующее окно:



13. Выберите пункт «CONTROL». Дождитесь обмена ультразвуковыми данными, поскольку это является подтверждением нормальной передачи данных по Bluetooth. Теперь робот доступен для управления.



14. Нажмите кнопку «MODE». Откроется следующее окно.



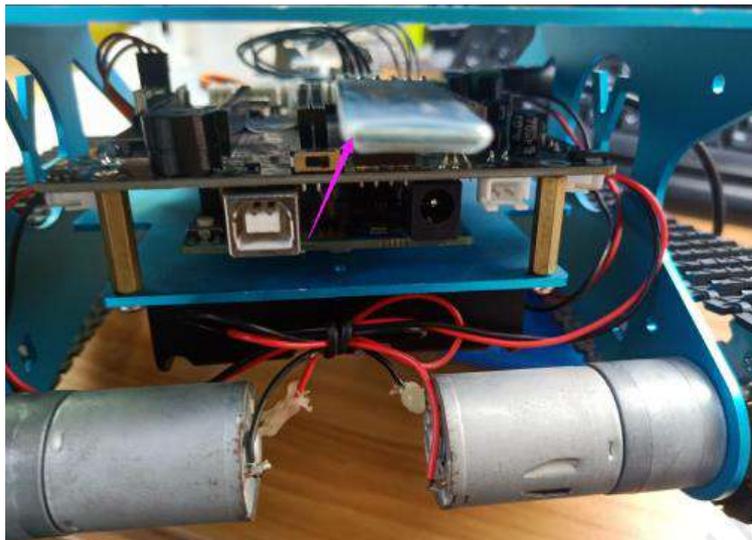
Необходимо строго соблюдать порядок действий, в противном случае могут наблюдаться проблемы с подключением.

Примечания:

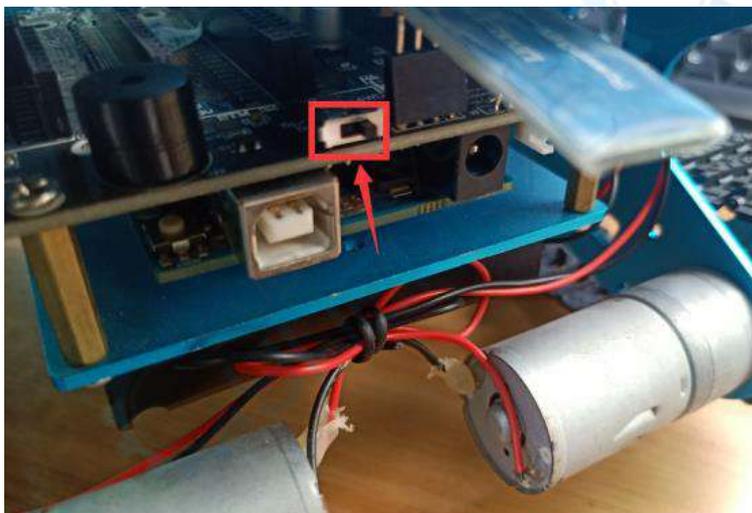
- Для обеспечения нормальной работы робот должен быть заряжен.



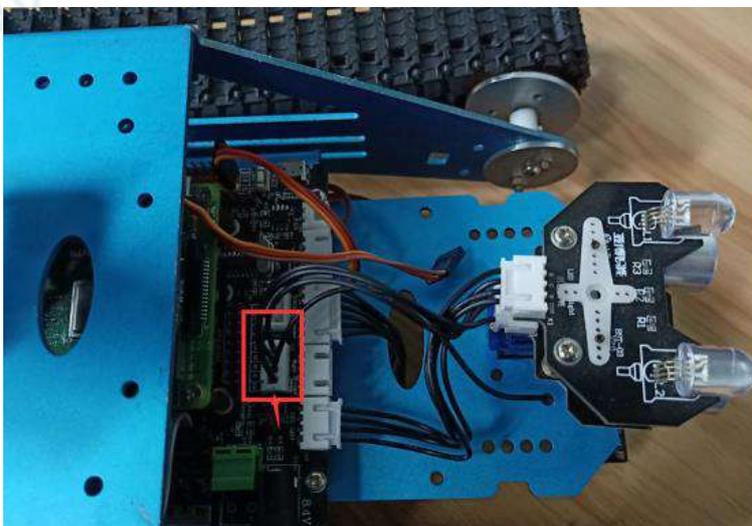
- Модуль Bluetooth должен быть правильно вставлен в плату расширения робота, как показано ниже.



- Переключатель 5V/Arduino Download Switch на плате расширения должен быть включен в положение «OFF», как показано на рисунке.



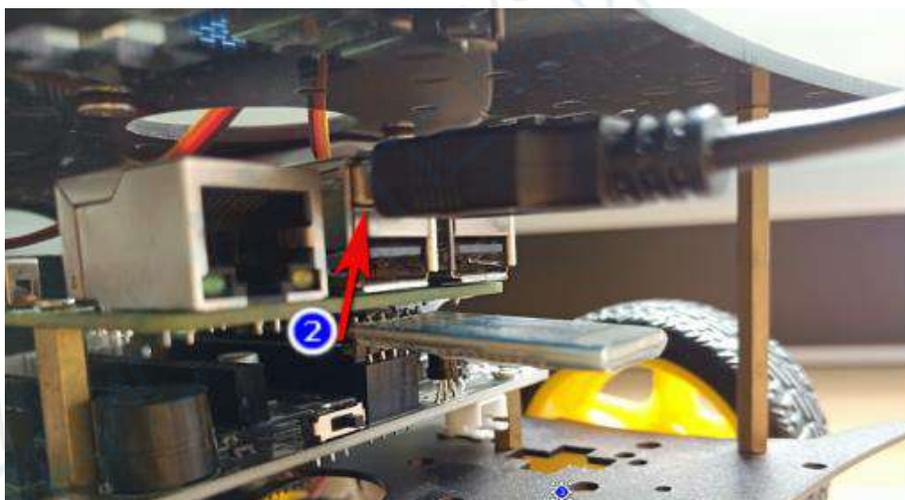
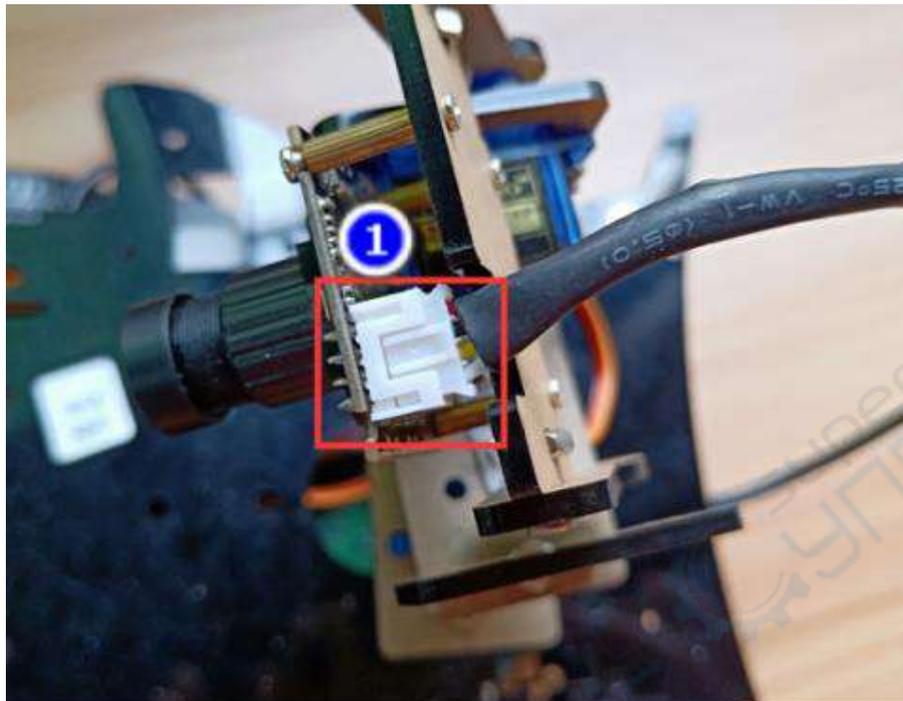
- Ультразвуковой модуль должен быть подключен, как показано на рисунке.



Прочитайте руководство для ознакомления с функцией удаленного управления по Bluetooth.

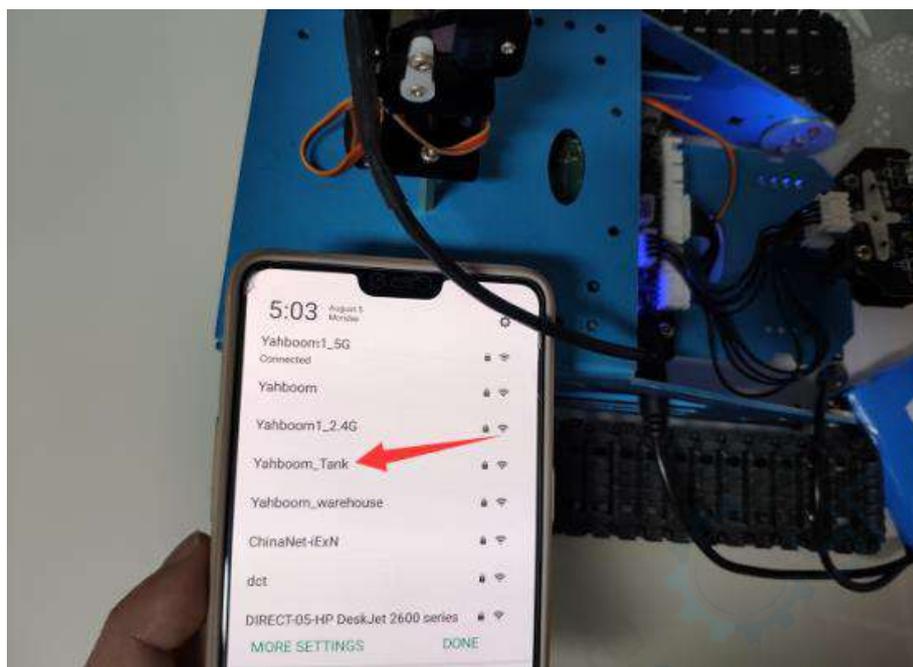
2 ПОДКЛЮЧЕНИЕ И ИСПОЛЬЗОВАНИЕ КАМЕРЫ

Камеру следует подключить к плате Raspberry Pi в соответствии с приведенным ниже рисунком:



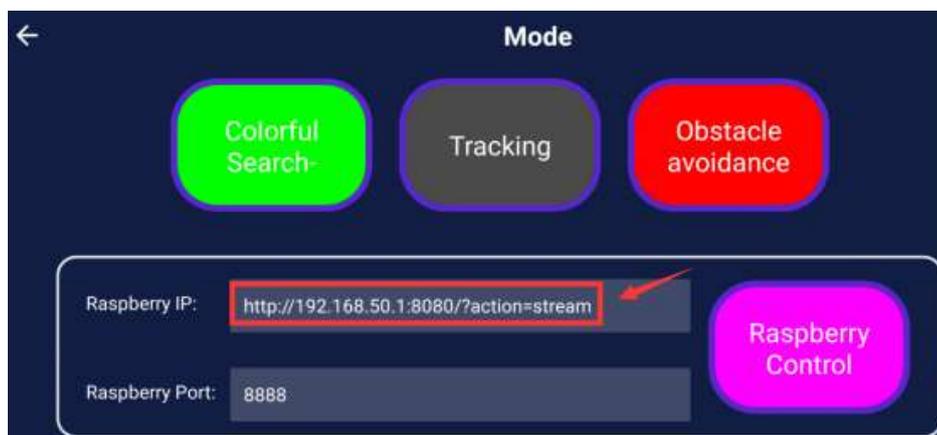
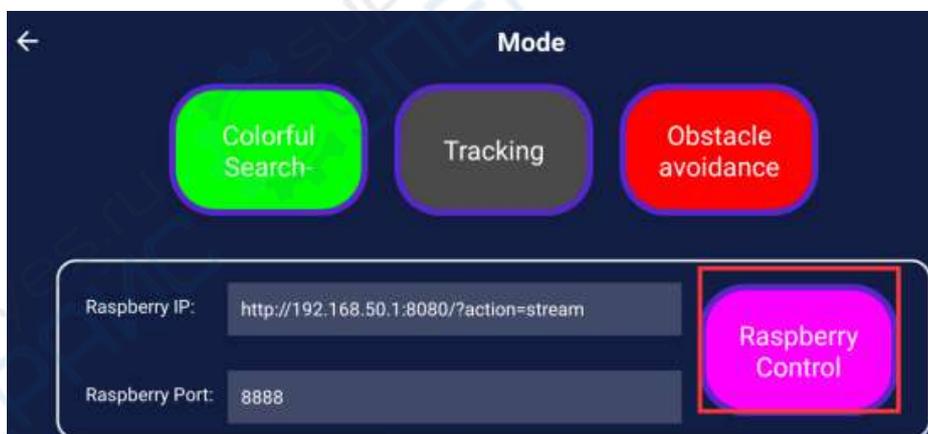
Подключение к роботу по Wi-Fi производится с использованием следующих данных:

Имя пользователя: Yahboom_Tank
Пароль: 12345678



Данная сеть Wi-Fi используется только для передачи видеоданных и к ней нет доступа из внешней сети.

После подключения по Wi-Fi нажмите кнопку «**RaspberryControl**» для просмотра изображения с камеры на экране мобильного телефона.



3 ДОСТУП К СИСТЕМЕ RASPBERRY PI

Примечание: по умолчанию используются следующие данные:

IP-адрес:192.168.50.1

Имя пользователя: pi

Пароль: yahboom

ПК должен быть подключен к роботу по WI-FI.

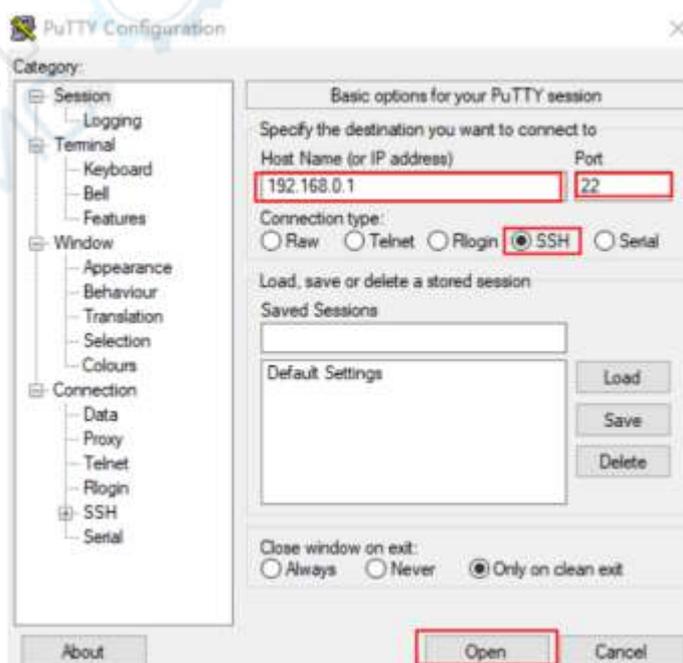
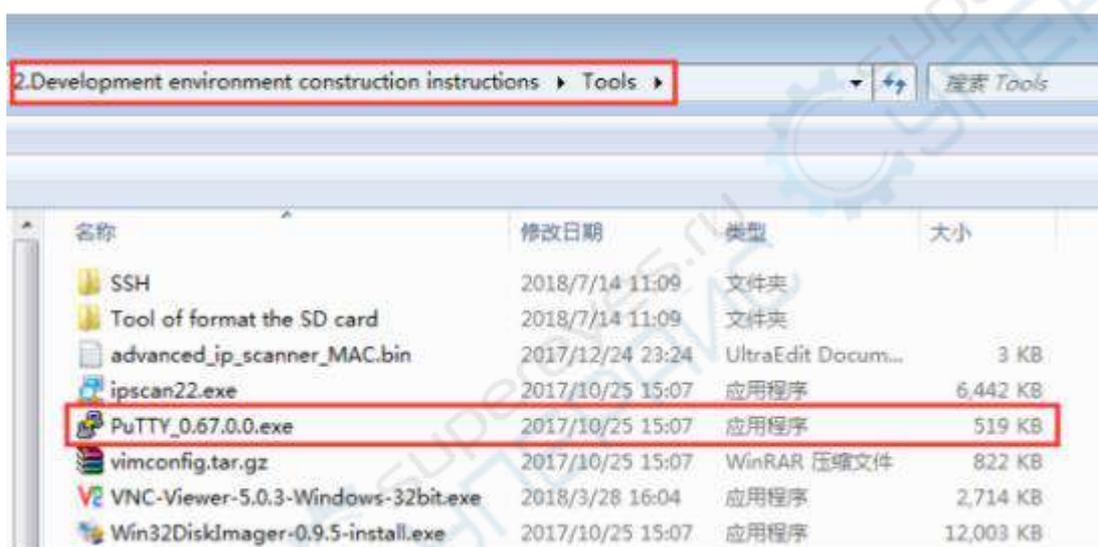
Идентификатор сети WI-FI: Yahboom_Tank

Пароль: 12345678

Удаленное подключение с помощью PuTTY

Подключиться к системе Raspberry Pi можно с помощью программы PuTTY.

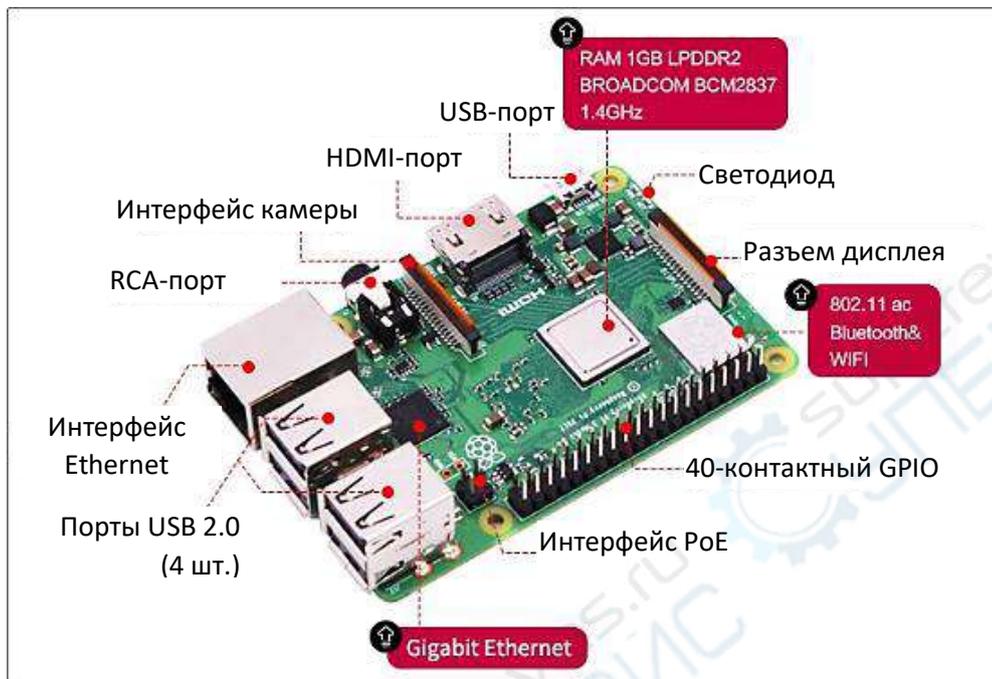
Примечание: программа находится в папке Tools.



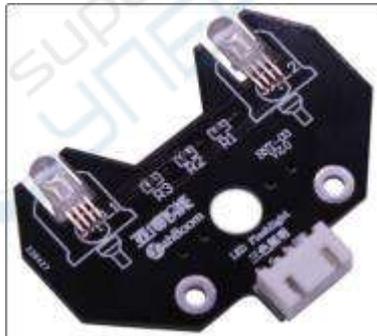
4 БАЗОВЫЙ УРОК: COLOR_LED

ColorLED.c
ColorLED.py

1. Подготовка



Плата Raspberry Pi



Модуль RGB

2. Цель эксперимента

После запуска программы светодиод продемонстрирует 7 различных цветов.

3. Описание эксперимента

RGB-модуль оснащен тремя светодиодами (красный, зеленый, синий). Плата позволяет смешивать различные цвета (256x256x256) путем управления их интенсивностью свечения. Согласно принципиальной схеме, лампа RGB представляет собой светодиод с общим катодом, один контакт которого подключается к GND, а остальные три контакта соответственно подключаются к портам 3, 2, 5 платы Raspberry Pi. Каждый светодиод необходимо подключить последовательно с токоограничительным резистором номиналом 220 Ом. Светодиодом можно управлять, подавая соответствующий уровень на плату.

4. Этапы эксперимента

4-1. Схемы:

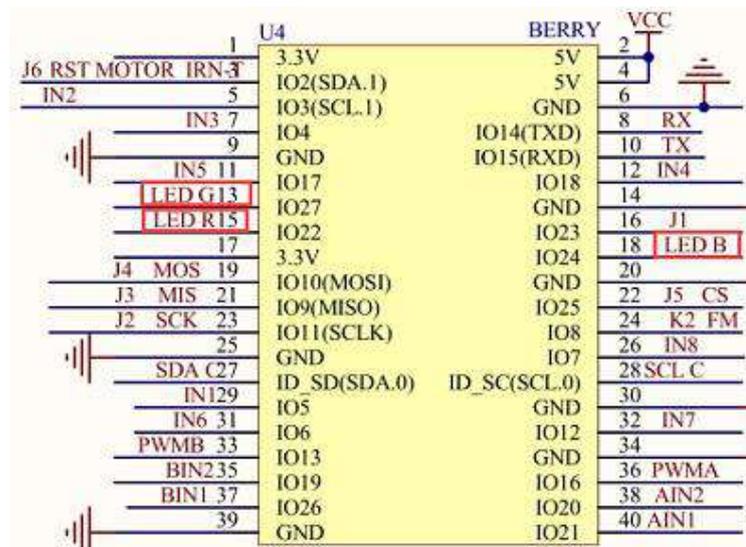


Схема интерфейса Raspberry Pi

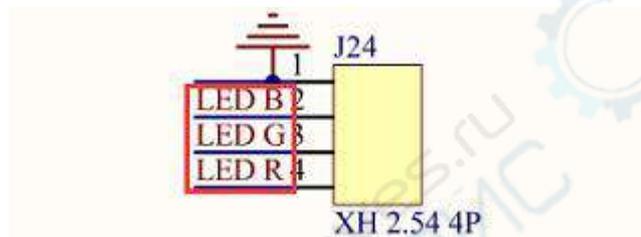


Схема интерфейса RGB-модуля

wiringPi	BCM	Функция	Физические контакты		Функция	BCM	wiringPi
		3.3V	1	2	5V		
8	2	SDA.1	3	4	5V		
9	3	SCL.1	5	6	GND		
7	4	GPIO.7	7	8	TXD	14	15
		GND	9	10	RXD	15	16
0	17	GPIO.0	11	12	GPIO.1	18	1
2	27	GPIO.2	13	14	GND		
3	22	GPIO.3	15	16	GPIO.4	23	4
		3.3V	17	18	GPIO.5	24	5
12	10	MOSI	19	20	GND		
13	9	MISO	21	22	GPIO.6	25	6
14	11	SCLK	23	24	CE0	8	10
		GND	25	26	CE1	7	11
30	0	SDA.0	27	28	SCL.0	1	31
21	5	GPIO.21	29	30	GND		
22	6	GPIO.22	31	32	GPIO.26	12	26
23	13	GPIO.23	33	34	GND		
24	19	GPIO.24	35	36	GPIO.27	16	27
25	26	GPIO.25	37	38	GPIO.28	20	28
		GND	39	40	GPIO.29	21	29

Распиновка 40-контактного интерфейса. Сравнительная таблица

4-2. В соответствии со схемой:

- LED_R-----15 (физический контакт)----- 3 (wiringPi)
- LED_G-----13 (физический контакт)----- 2 (wiringPi)
- LED_B-----18 (физический контакт)----- 5 (wiringPi)

Примечание: при написании кода будет использоваться библиотека wiringPi.

4-3. Информация о коде

Перед компиляцией кода можно просмотреть изменения режима и уровня состояния контактов с помощью команды `gpio readall`.

```
root@raspberrypi:/home/pi/SmartCar# ls
ColorLED ColorLED.c initpin.sh
root@raspberrypi:/home/pi/SmartCar# gpio readall
```

Pi 3											
BCM	wPi	Name	Mode	V	Physical	V	Mode	Name	wPi	BCM	
		3.3v			1	2		5v			
2	8	SDA.1	IN	1	3	4		5v			
3	9	SCL.1	IN	1	5	6		0v			
4	7	GPIO. 7	IN	1	7	8	1	TxD	15	14	
		0v			9	10	1	RxD	16	15	
17	0	GPIO. 0	IN	0	11	12	1	GPIO. 1	1	18	
27	2	GPIO. 2	IN	0	13	14		0v			
22	3	GPIO. 3	IN	0	15	16	0	GPIO. 4	4	23	
		3.3v			17	18	0	IN	GPIO. 5	5	24
10	12	MOST	TN	1	19	20		0v			
9	13	MISO	TN	1	21	22	1	GPIO. 6	6	25	
11	14	SCLK	TN	1	23	24	1	CE0	10	8	
		0v			25	26	1	CE1	11	7	
0	30	SDA.0	IN	1	27	28	1	SCL.0	31	1	
5	21	GPIO.21	IN	1	29	30		0v			
6	22	GPIO.22	IN	1	31	32	0	GPIO.26	26	12	
13	23	GPIO.23	IN	0	33	34		0v			
19	24	GPIO.24	IN	0	35	36	0	GPIO.27	27	16	
26	25	GPIO.25	IN	0	37	38	0	GPIO.28	28	20	
		0v			39	40	0	GPIO.29	29	21	

A. Код .c

- Сначала необходимо скомпилировать файлы в системе Raspberry Pi (примечание: следует добавить `-lwiringPi` к файлу библиотеки).
Необходимо ввести команды: `gcc ColorLED.c -o ColorLED -lwiringPi`
- Затем следует запустить скомпилированный запускаемый файл в системе Raspberry Pi.
Введите команду `./ColorLED`

```
pi@ya:~/SmartCar$ gcc ColorLED.c -o ColorLED -lwiringPi
pi@ya:~/SmartCar$ ./ColorLED
```

- Можно прервать процесс одновременным нажатием `ctrl+c`. В данном случае, ядру Linux будет отправлена команда прерывания текущего процесса, однако состояние соответствующего контакта является неопределенным, поэтому необходимо запустить скрипт инициализации всех контактов.
- Для просмотра режима и состояния всех контактов введите команду `gpio readall`.
Изменение состояния соответствующего контакта будет отображено в окне.

```

root@raspberrypi:/home/pi# gpio readall
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| BCM | wPi | Name | Mode | V | Physical | V | Mode | Name | wPi | BCM |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 2 | 8 | 3.3v | | | 1 | 2 | | | 5v | | |
| 3 | 9 | SDA.1 | IN | 1 | 3 | 4 | | | 5v | | |
| 4 | 7 | SCL.1 | IN | 1 | 5 | 5 | | | 0v | | |
| 4 | 7 | GPIO.7 | IN | 1 | 7 | 8 | 1 | IN | TxD | 15 | 14 |
| | | | | | 9 | 10 | 1 | IN | RxD | 16 | 15 |
| 17 | 0 | GPIO.0 | IN | 0 | 11 | 12 | 1 | IN | GPIO.1 | 1 | 18 |
| 27 | 2 | GPIO.2 | OUT | 1 | 13 | 14 | | | 0v | | |
| 22 | 3 | GPIO.3 | OUT | 0 | 15 | 16 | 0 | IN | GPIO.4 | 4 | 23 |
| | | | | | 17 | 18 | 0 | OUT | GPIO.5 | 5 | 24 |
| 10 | 12 | 3.3v | | | 19 | 20 | | | 0v | | |
| 9 | 13 | MOSI | IN | 1 | 21 | 22 | 1 | IN | GPIO.6 | 6 | 25 |
| 11 | 14 | MISO | IN | 1 | 23 | 24 | 1 | IN | CE0 | 10 | 8 |
| | | | | | 25 | 26 | 1 | IN | CE1 | 11 | 7 |
| 0 | 30 | SCL.0 | IN | 1 | 27 | 28 | 1 | IN | SCL.0 | 31 | 1 |
| 5 | 21 | GPIO.21 | IN | 1 | 29 | 30 | | | 0v | | |
| 6 | 22 | GPIO.22 | IN | 1 | 31 | 32 | 0 | IN | GPIO.26 | 26 | 12 |
| 13 | 23 | GPIO.23 | IN | 0 | 33 | 34 | | | 0v | | |
| 19 | 24 | GPIO.24 | IN | 0 | 35 | 36 | 0 | IN | GPIO.27 | 27 | 16 |
| 26 | 25 | GPIO.25 | IN | 0 | 37 | 38 | 0 | IN | GPIO.28 | 28 | 20 |
| | | | | | 39 | 40 | 0 | IN | GPIO.29 | 29 | 21 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| BCM | wPi | Name | Mode | V | Physical | V | Mode | Name | wPi | BCM |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

```

Примечание: скрипт инициализации `initpin.sh` находится в папке `SmartCar/python`.

Необходимо ввести следующие команды:

```

chmod 777 initpin.sh
./initpin.sh

```

```

pi@raspberrypi:~/SmartCar $ sudo chmod 777 initpin.sh
pi@raspberrypi:~/SmartCar $ ./initpin.sh

```

В. Для кода python

- Для запуска кода Python необходимо ввести следующую команду:

```
python ColorLED.py
```

```

pi@raspberrypi:~/python $ python ColorLED.py

```

- Можно прервать процесс одновременным нажатием `ctrl+c`. В данном случае, ядру Linux будет отправлена команда прерывания текущего процесса, однако состояние соответствующего контакта является неопределенным, поэтому необходимо запустить скрипт инициализации всех контактов.
- Введите команду: `chmod 777 initpin.sh`

```
./initpin.sh
```

```

pi@raspberrypi:~/SmartCar $ sudo chmod 777 initpin.sh
pi@raspberrypi:~/SmartCar $ ./initpin.sh

```

После выполнения описанных шагов эксперимент будет завершен.

- Результат работы программы: будет наблюдаться переключение цветов раз в секунду.