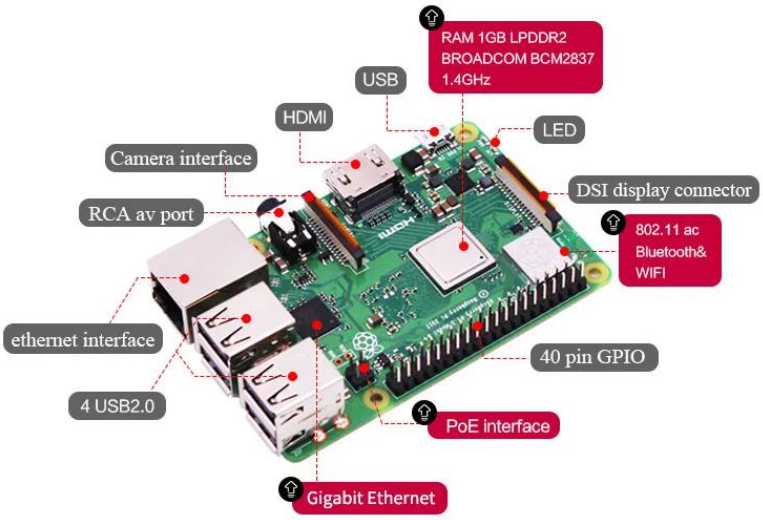
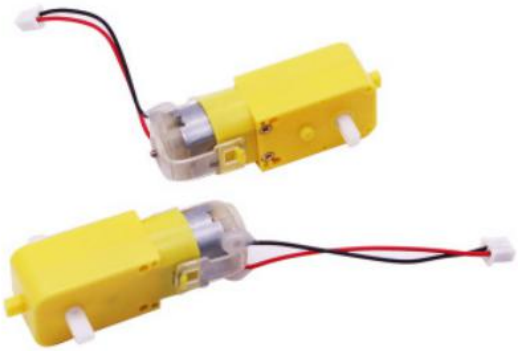
3.Raspberry Pi platform ------- Car Run

1)Preparation



1-1 Raspberry Pi board



1-2 4 motors

2) Purpose of Experimental

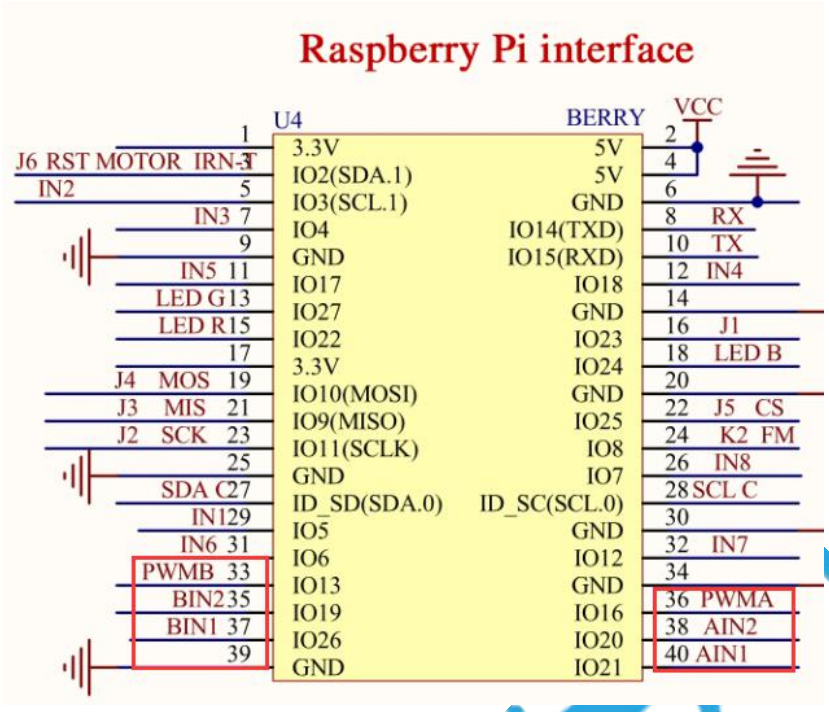
After running the CarRun executable in the Raspberry Pi system. After the car is still for 2 s, advance 1 s ,back 1 s,turn left 2 s,turn right 2 s, turn left in place 3 s, turn right in place 3 s, stop 0.5s.

3)Principle of experimental

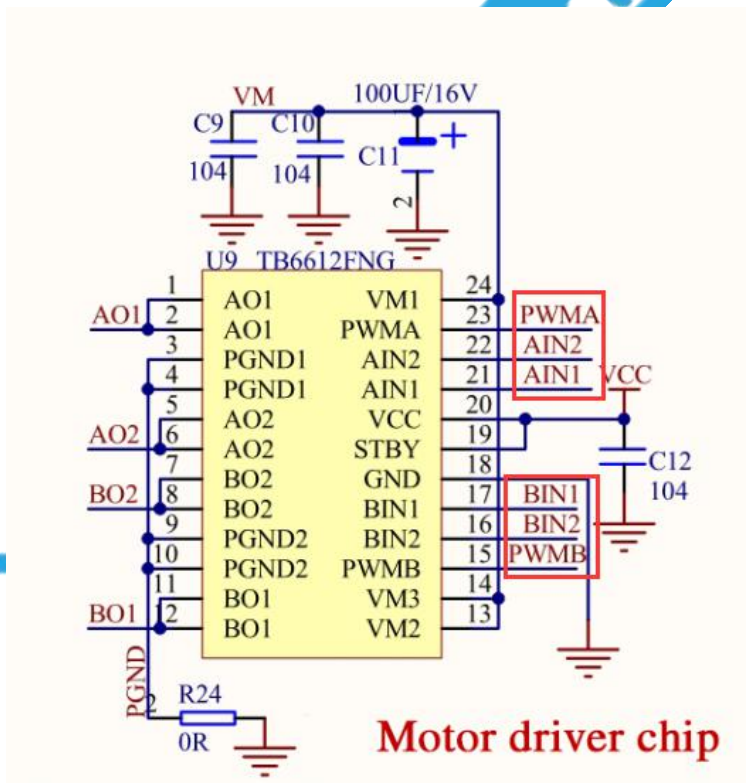
We use the TB6612FNG driver chip to drive the motor. Control the forward, reverse, and stop of the motor by controlling the level of AIN1, AIN2, BIN1, BIN2, PWMA, and PWMB of the driver chip.
In this experiment, we control robot car advance by controlling AIN1 to be high level, AIN2 is low level, BIN1 is high level, BIN2 is low level. And we control speed of the robot car by controlling PWMA, PWMB(0-255). One-channel PWM control the speed of the motor on one side of the robot car.

4)Experimental Steps

4-1 About the schematic



4-1 Raspberry Pi interface circuit diagram



4-2 Motor drive chip---TB6612FNG

wiringPi	BCM	Funtion	Physical pin		Funtion	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

4-3 Raspberry Pi 40 pins comparison table

4-2 According to the circuit schematic:

AIN1-----40(Physical pin)----29(wiringPi)
AIN2-----38(Physical pin)----28(wiringPi)
PWMA-----36(Physical pin)----27(wiringPi)
BIN1-----37(Physical pin)----25(wiringPi)
BIN2-----35(Physical pin)----24(wiringPi)
PWMB-----33(Physical pin)----23(wiringPi)

(Note: We use the wiringPi library to write code.)

4-3 About the code

Please view .py and.c file

A. For .c code

(1) We need to compile this file in the Raspberry Pi system. (Note: we need to add -lwiringPi to the library file.)

We need to input: gcc CarRun.c -o CarRun -lwiringPi

(2)We need to run the compiled executable file in the Raspberry Pi system.We need to input: ./CarRun

```
pi@yahboom4wd:~/SmartCar $ gcc CarRun.c -o CarRun -lwiringPi
pi@yahboom4wd:~/SmartCar $ ./CarRun
```

(3)We can input: ctrl+c to stop this process, which mean is send a signal to the linux kernel to terminate the current process, but the state of the relevant pin is uncertain at this time, we also need to run a script to initialize all pins.
You need to input:
chmod 777 initpin.sh
./initpin.sh

```
pi@yahboom4wd:~/SmartCar $ sudo chmod 777 initpin.sh
pi@yahboom4wd:~/SmartCar $ ./initpin.sh
```

B. For python code

1) We need to input following command to run python code.
python CarRun.py

```
pi@yahboom4wd:~/python $ python CarRun.py
```

2) We can input: ctrl+c to stop this process, which mean is send a signal to the linux kernel to terminate the current process, but the state of the relevant pin is uncertain at this time, we also need to run a script to initialize all pins.
3) You need to input: chmod 777 initpin.sh
./initpin.sh

```
pi@yahboom4wd:~/SmartCar $ sudo chmod 777 initpin.sh
pi@yahboom4wd:~/SmartCar $ ./initpin.sh
```

After completing the above steps, the experiment is over.

5) Experimental phenomenon

When we run the program, car will advance 1 s ,back 1 s,turn left 2 s,turn right 2 s, spin left 3 s, spin right in place 3 s, stop 0.5s.