

Hardware Control course--Drive LCD

1. Learning target

In this course, we will learn how to use drive LED on expansion board.

2. Principle of experimental

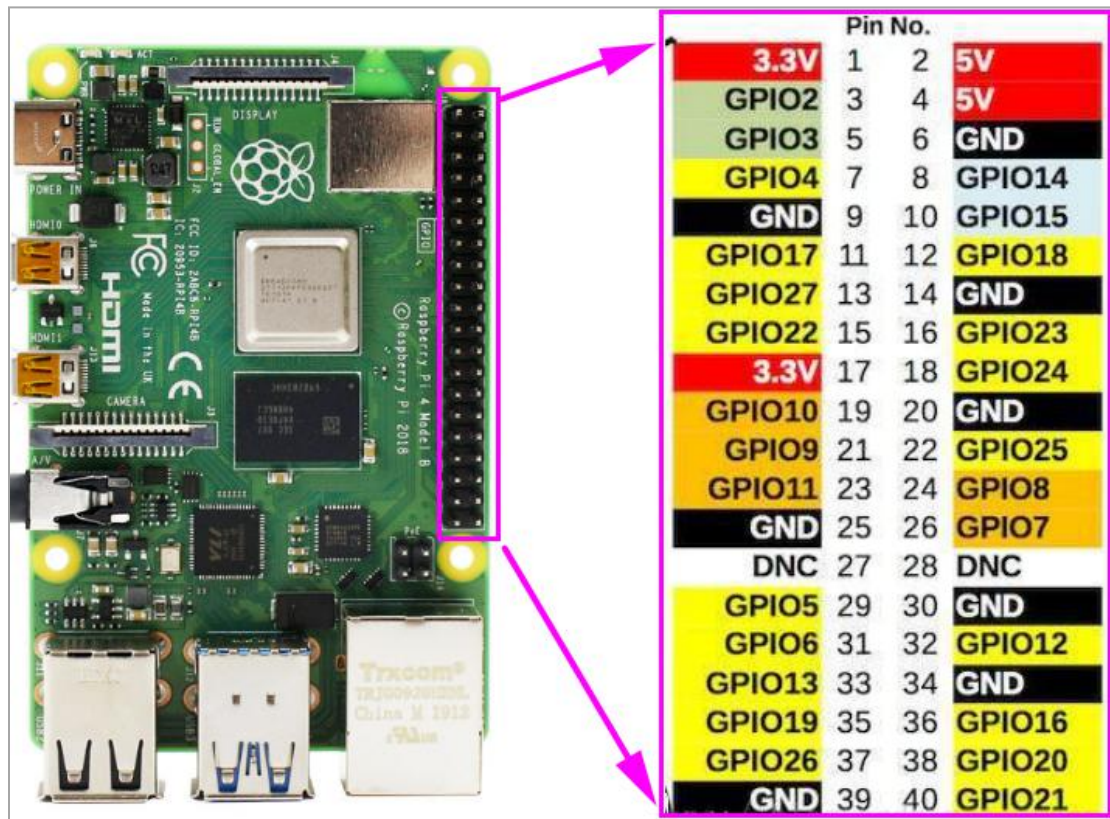
According to the hardware interface manual, LED1 and LED2 are directly driven by the physical pins 40 and 38 of the Raspberry Pi board.

Classification	Function	Pi	BOARD	BCM	Remark
Tracking module	Left 1	GPIO.2	13	27	
	Left 2	GPIO.3	15	22	
	Right 1	GPIO.0	11	17	
	Right 2	GPIO.7	7	4	
Infrared obstacle avoidance module	Left	MISO	21	9	
	Right	MOSI	19	10	
Infrared obstacle avoidance module switch	Turn on infrared obstacle avoidance	GPIO.6	22	25	
Ultrasonic module	Echo	GPIO.5	18	24	
	Trig	GPIO.4	16	23	
Buzzer	Buzzer	GPIO.26	32	12	
Infrared receiving sensor	Infrared receiver	GPIO.27	36	16	
LED1 (red)	red light	GPIO.29	40	21	
LED2 (blue)	blue light	GPIO.28	38	20	
MCU coprocessor	SCL	SCL.1	5	3	Raspberry Pi communicates with MCU through I2C to drive motors and servos
	SDA	SDA.1	3	2	
The motor and servo ports (S1-S4) are driven by the bottom MCU .					

3. Coding method

In this course, we use BOARD coding method.

We need to control the 40,38 pins of the Raspberry Pi board.



wiringPi	BCM	Function	BOARD		Function	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. About code

Path: /home/pi/Yahboom_project/Raspbot/2.Hardware Control course/08.Drive LED/Drive LED.ipynb

1) Import time and GPIO library

```
#-*- coding:UTF-8 -*-
import RPi.GPIO as GPIO
import time
```

2) Set the GPIO coding mode, set LED1 and LED2 to output mode

```
#Set the pin coding mode to BOARD coding mode
GPIO.setmode(GPIO.BOARD)

#Ignore warning
GPIO.setwarnings(False)

LED1 = 40 #Define the pin of LED1 (red)
LED2 = 38 #Define the pin of LED1 (blue)

Set the pins of LED1 and LED2 to output mode

GPIO.setup(LED1,GPIO.OUT)
GPIO.setup(LED2,GPIO.OUT)
```

3) Set high level to light on LED

```
Light up

GPIO.output(LED1, GPIO.HIGH)
GPIO.output(LED2, GPIO.HIGH)
```

4) Set low level to light off LED

```
Light off

GPIO.output(LED1, GPIO.LOW)
GPIO.output(LED2, GPIO.LOW)
```

5. Running code

Click the button shown in the figure below to run the program on the Jupyter Lab interface



6. Experimental phenomena

When we run the program that lights up the LED light, LED1 and LED2 light up red and blue respectively.

When we run the program that lights off the LED light, the two LED lights are off.