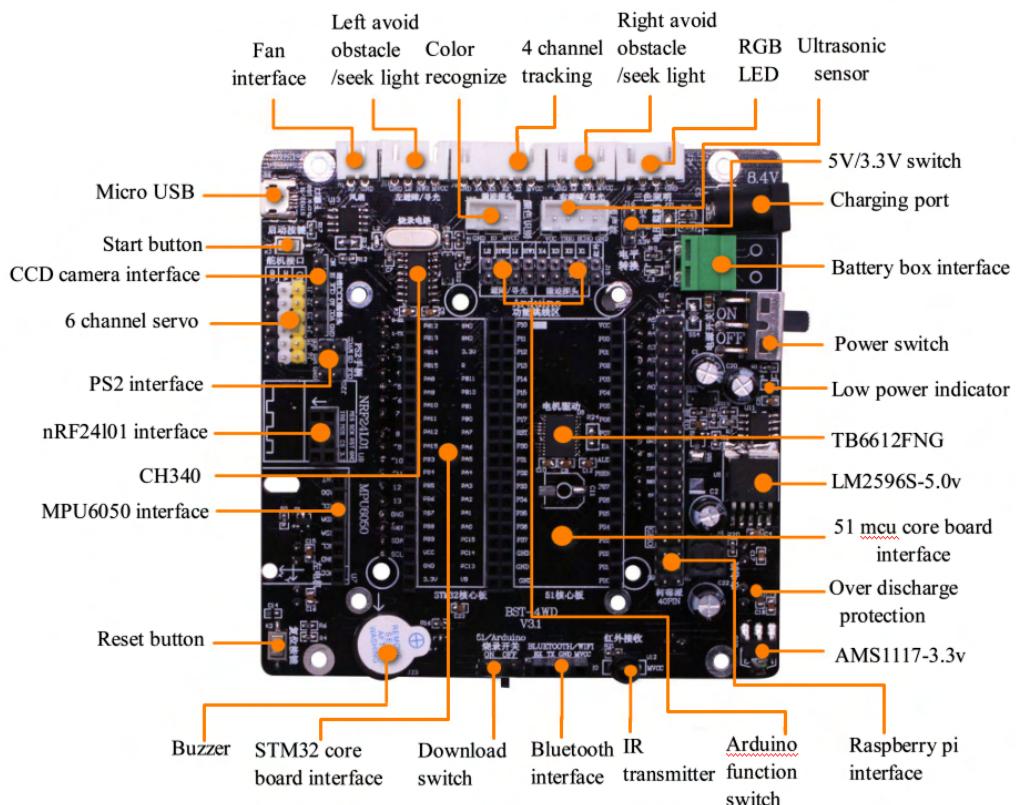
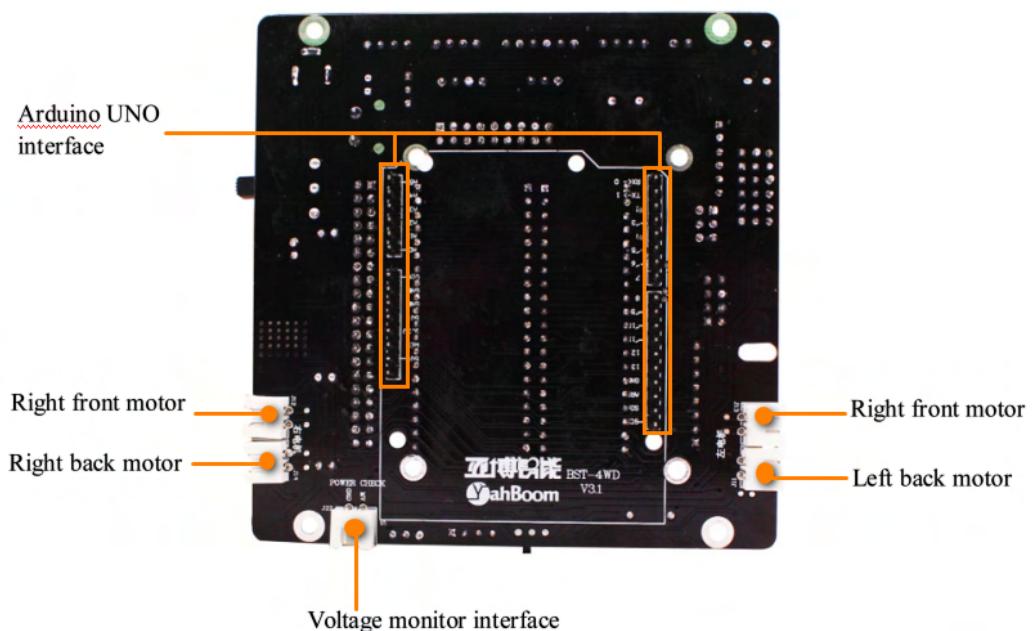


## 4WD expansion board manual

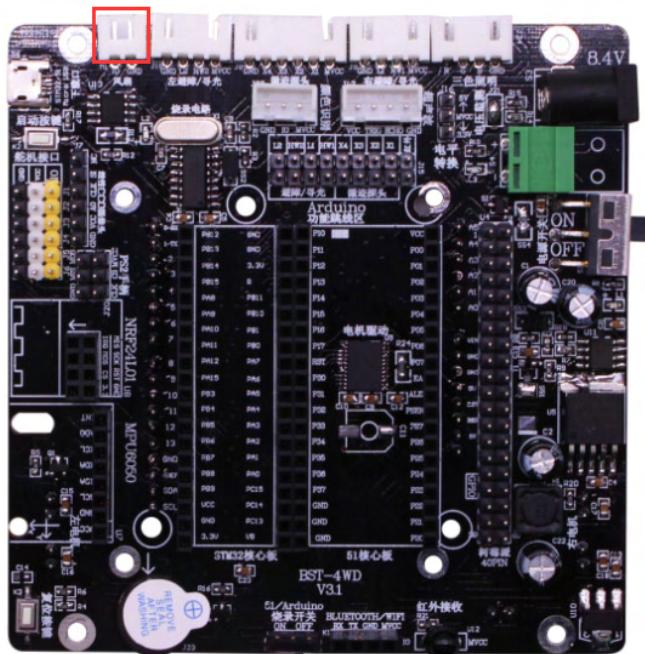
Front:



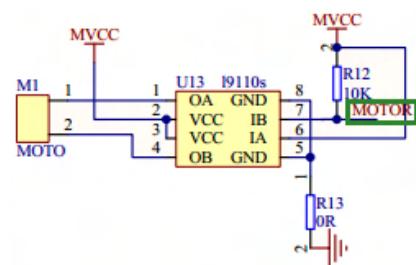
Back:



1. Fan interface:

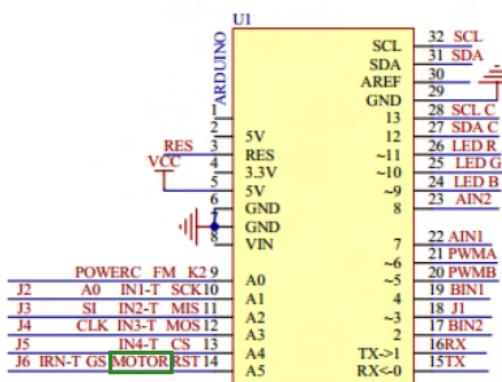


## 1-1 position



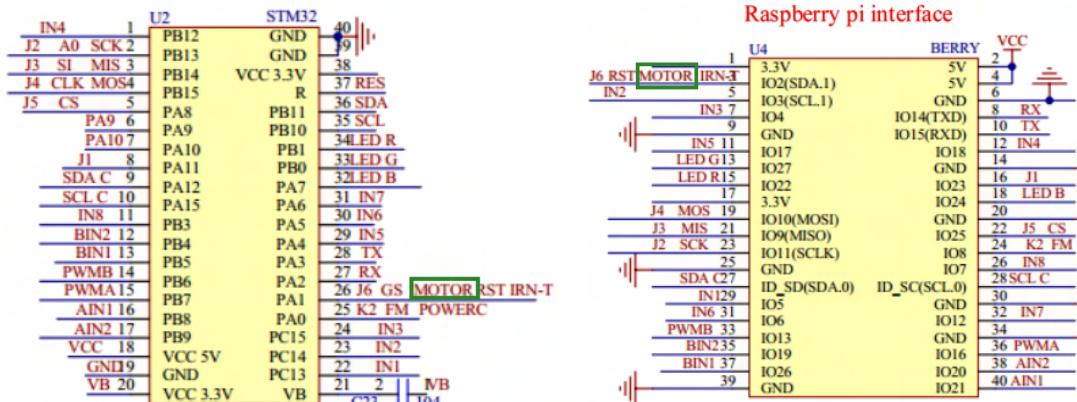
51 mcu core board interface

## Arduino UNO interface



IN1	1	U3		STC	VCC
IN2	2	P1.0		40	39 J6 RST
IN3	3	P1.1		38	J5 CS
IN4	4	P1.2		37	J4 MOS
IN5	5	P1.3		36	J3 MIS
IN6	6	P1.4		35	J2 SCK
IN7	7	P1.5		34	J1
IN8	8	P1.6		33	SDA
		P1.7		32	SCL
51 RES	9	RST			
PA10 TX	10	RXD		31	LED R
PA9 RX	11	TXD		30	LED G
IRN	12	INT0		29	LED B
	13	INT1			
	14	T0		28	K2 FM
	15	T1		27	
SDA C	16	WR		26	PWMB
SCl C	17	RD		25	BIN2
	18	XTAL2		24	BIN1
	19	XTAL1		23	AIN1
	20	GND		22	AIN2
				21	PWMA

### STM32 core board interface



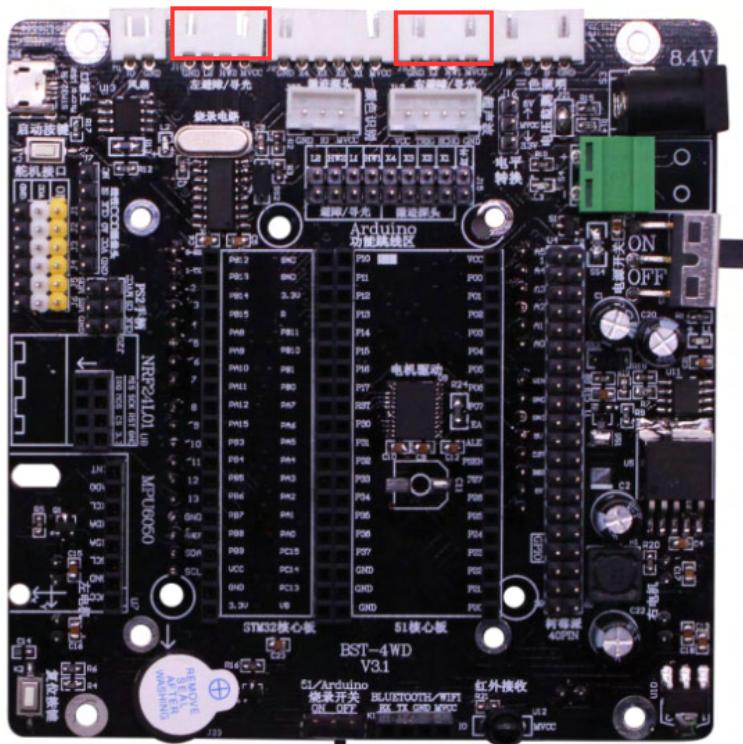
1-2 Schematic diagram

The interface has two pins: GND, IO port. When the IO port gets high level, the fan can rotate.

Pin table:

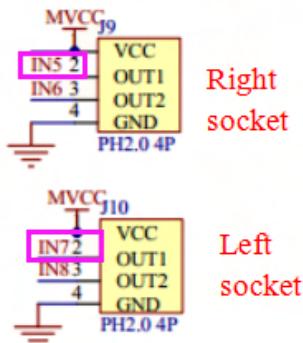
Module interface	Arduino	51controller	STM32	Raspberry Pi
IO port	A5	P0.0	PA1	IO2

### 2.Left and right infrared obstacle avoidance (light seeking) interface:

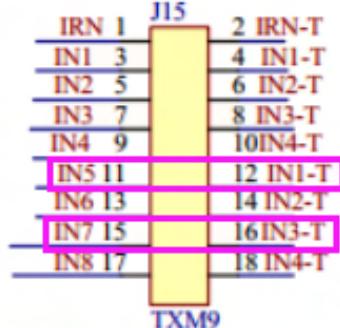


2-1 Position

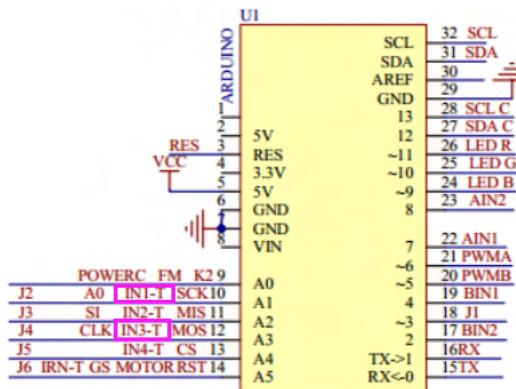
**Left IR avoid obstacle/seek light  
Module interface**



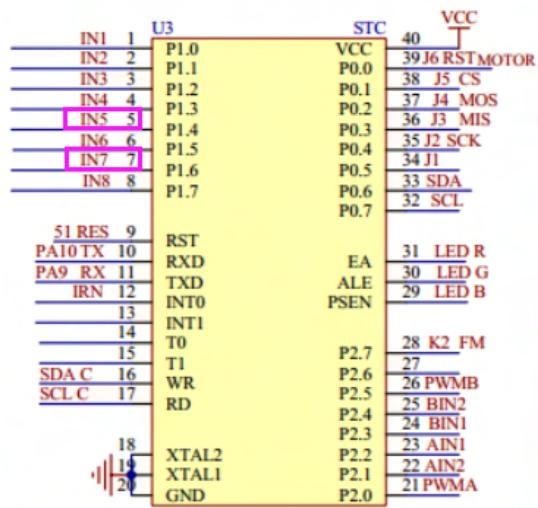
## Arduino function switch



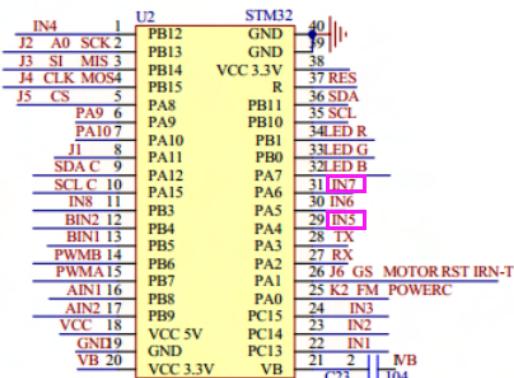
## Arduino UNO interface



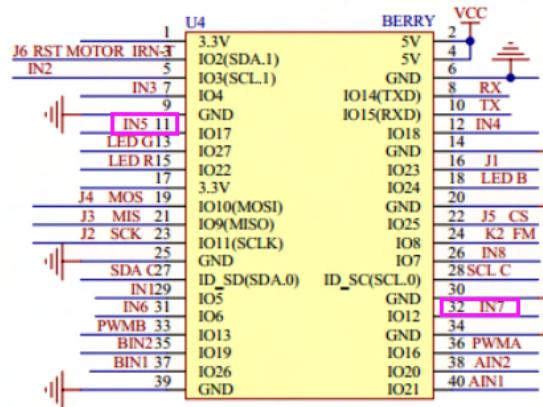
## 51 mcu core board interface



STM32 core board interface



## Raspberry pi interface



## 2-2 Schematic diagram

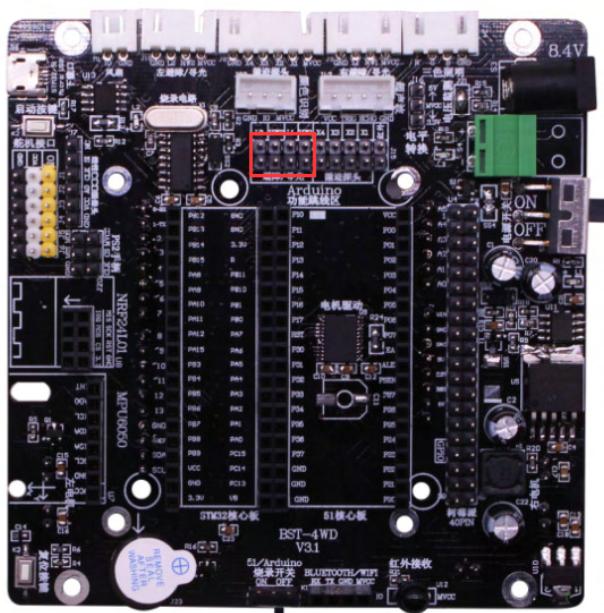
The interface has four pins: VCC, GND, and two OUT ports. Under normal working conditions, VCC is 5V voltage. We can connect the infrared obstacle avoidance sensor module to these two joints, and judge whether the front is an obstacle by detecting the level

of the two OUT1 ports.

Pin table:

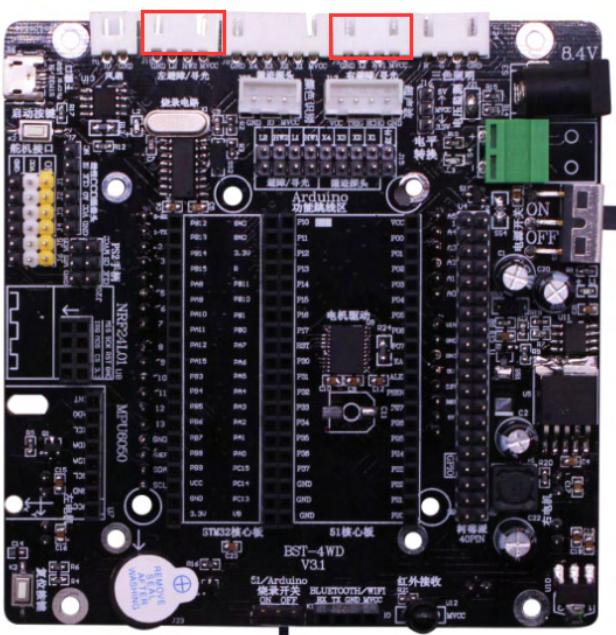
Module interface		Arduino	51 controller	STM32	Raspberry Pi
IN5(OUT1_R)	A1	P1.4	PA4	IO17	
IN7(OUT1_L)	A3	P1.6	PA6	IO12	

Note: When using the Arduino core controller, the corresponding IN5 and in1-t,IN7 and in3-t need to be connected with the jumper cap.



2-3 Position with jumper cap

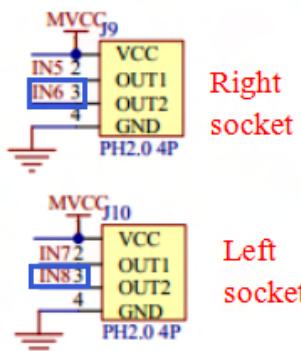
3. Light seeking (Left and right infrared obstacle avoidance) interface: :



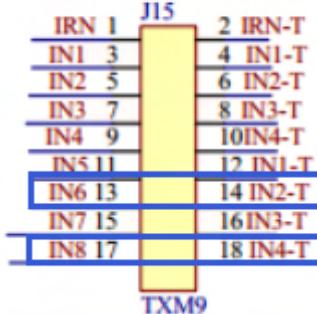
## 3-1 Position

Left IR avoid obstacle/seek light

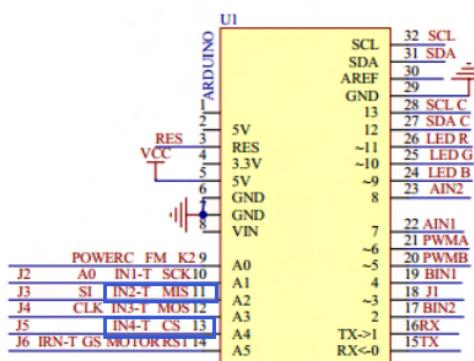
## Module interface



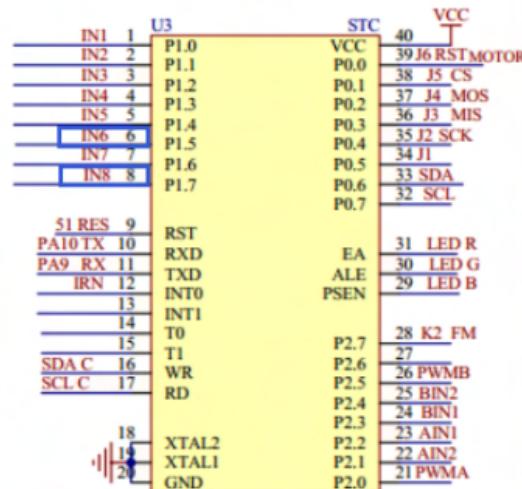
## Arduino function switch



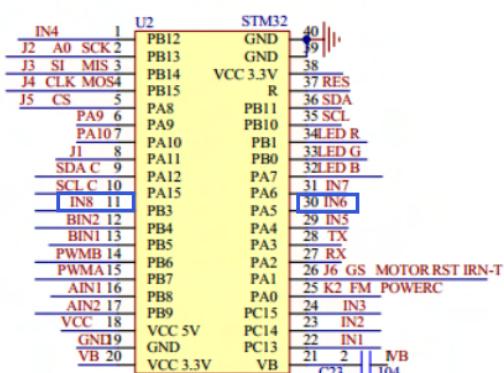
## Arduino UNO interface



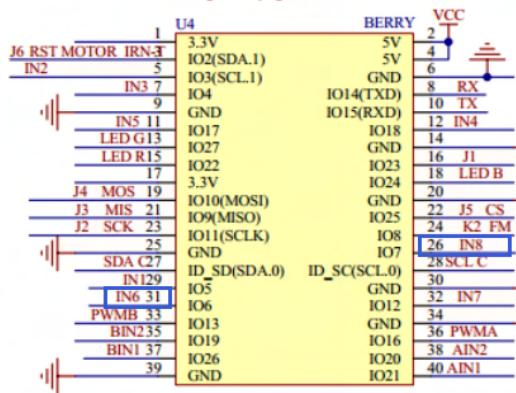
51 mcu core board interface



STM32 core board interface



## Raspberry pi interface



### 3-2 Schematic diagram

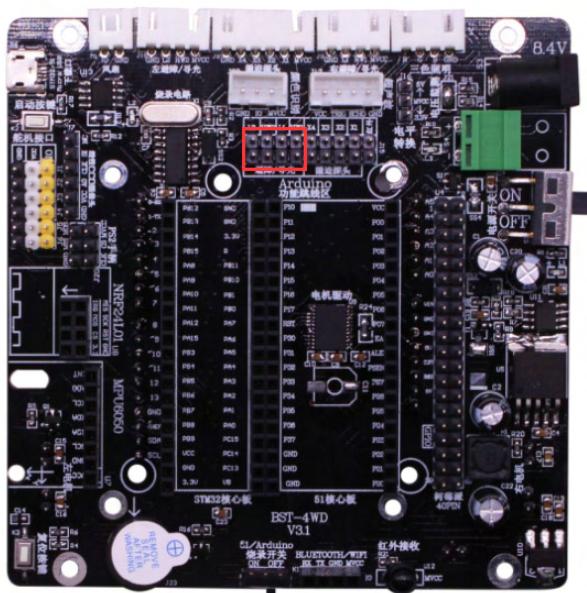
The interface has four pins: VCC, GND, and two OUT ports. Under normal working

conditions, VCC is 5V voltage. We can connect the infrared light-seeking sensor module to these two interfaces, and judge whether there is light by detecting the level of the two OUT2 ports.

Pin table:

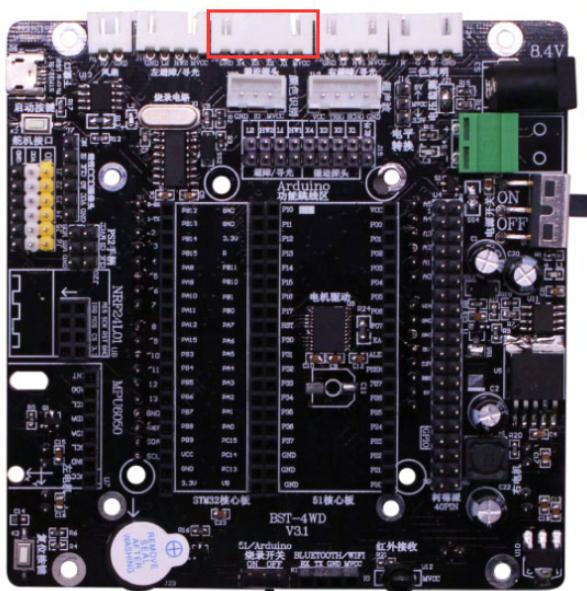
Module interface	Arduino	51controller	STM32	Raspberry Pi
IN6(OUT2_R)	A2	P1.5	PB3	IO6
IN8(OUT2_L)	A4	P1.7	PA5	IO7

**Note:** When using the Arduino core controller, the corresponding IN6 and in2-t, IN8 and in4-t need to be connected with the jumper cap.



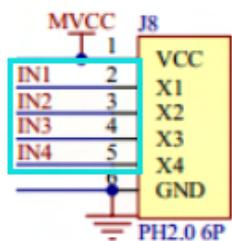
3-3 Position with jumper cap

#### 4. 4-Channel tracking interface

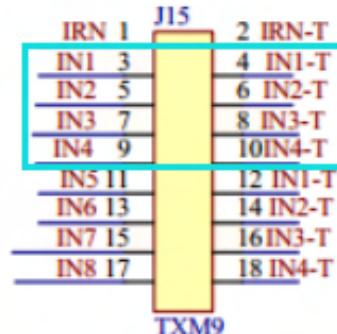


## 4-1 Position

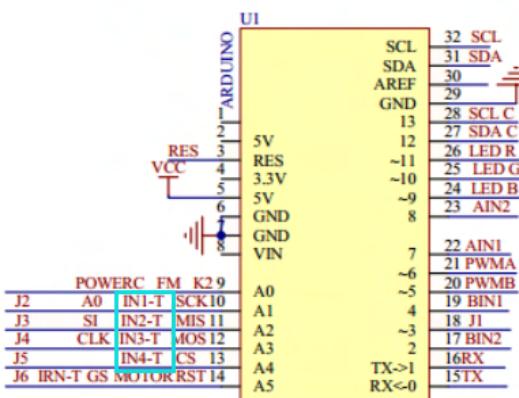
### 4 channel tracking Module interface



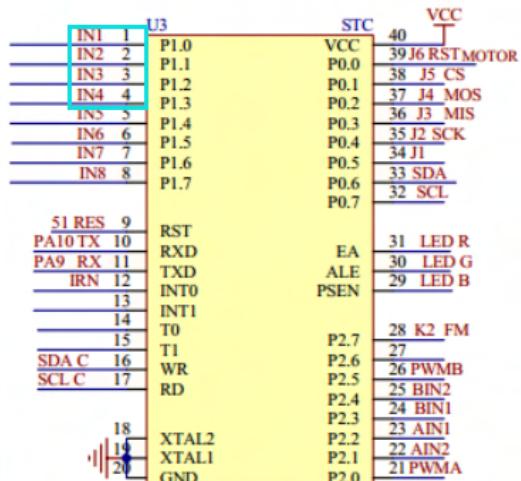
### Arduino function switch



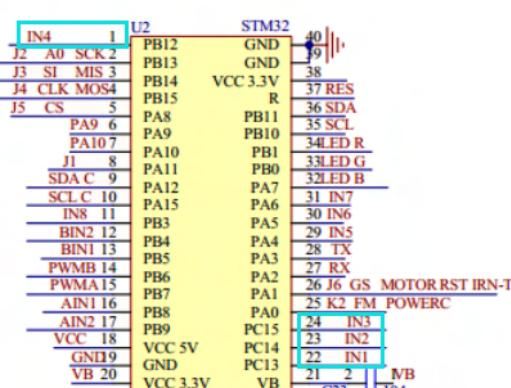
### Arduino UNO interface



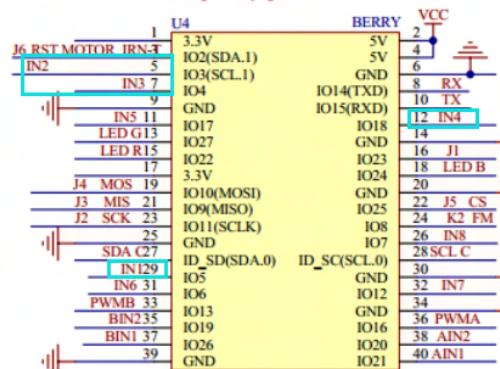
### 51 mcu core board interface



### STM32 core board interface



### Raspberry pi interface



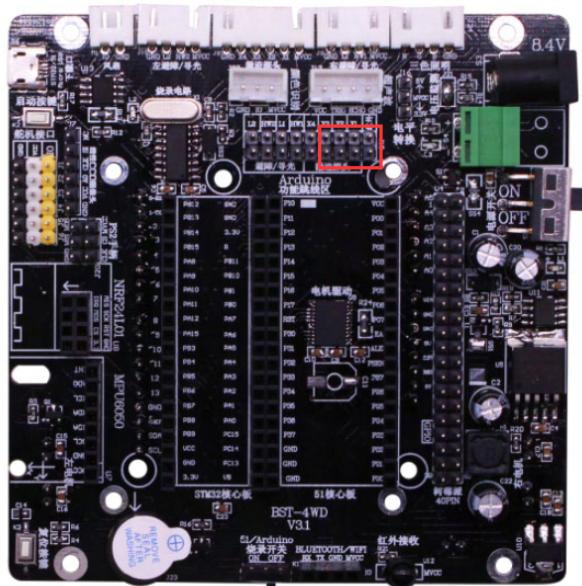
## 4-2 Schematic diagram

The interface has six pins: VCC, GND, IN1, IN2, IN3, and IN4. VCC is 5V voltage under normal operation. We can connect a four-channel tracking module here, and judge whether it is on the black track by checking the level of IN1, IN2, IN3 and IN4 ports.

Pin table:

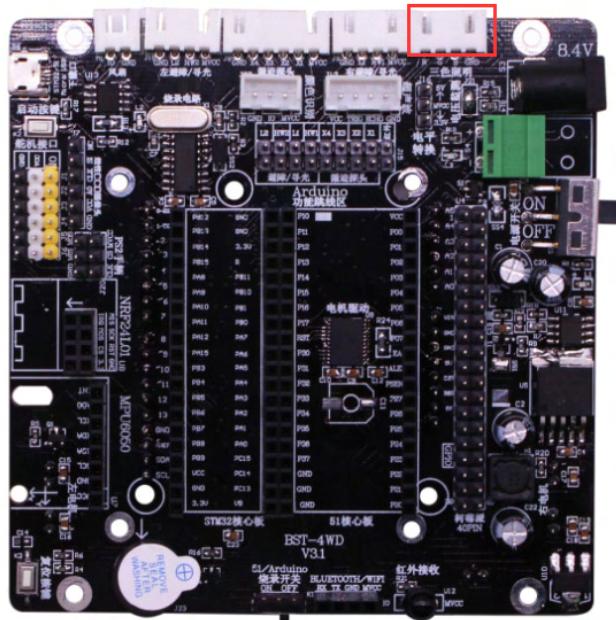
Module interface	Arduino	51controller	STM32	Raspberry
IN1	A1	P1.0	PC13	IO5
IN2	A2	P1.2	PC14	IO3
IN3	A3	P1.3	PC15	IO4
IN4	A4	P1.4	PB12	IO18

Note: When using the Arduino core controller, the corresponding IN1 and in1-t, IN2 and in2-t, IN3 and in3-t, IN4 and in4-t need to be connected with the jumper cap.



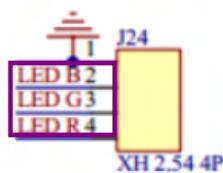
4-3 Position with jumper cap

#### 5.RGB LED module interface

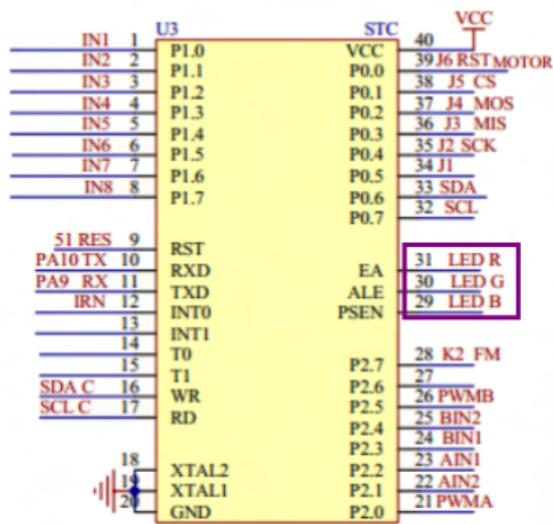
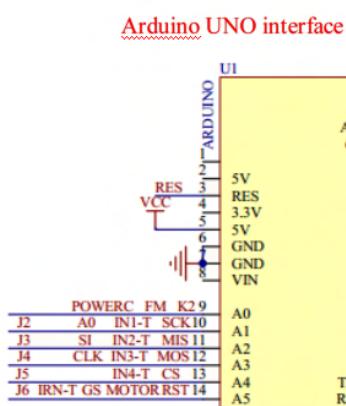


5-1 Position

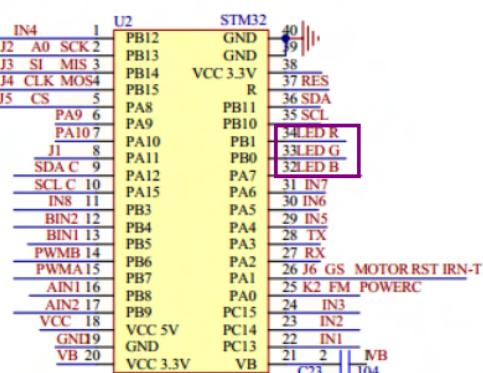
### RGB LED Module interface



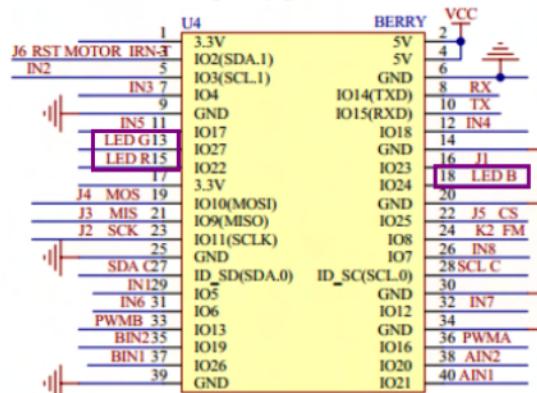
### 51 mcu core board interface



### STM32 core board interface



### Raspberry pi interface



5-2 Schematic diagram

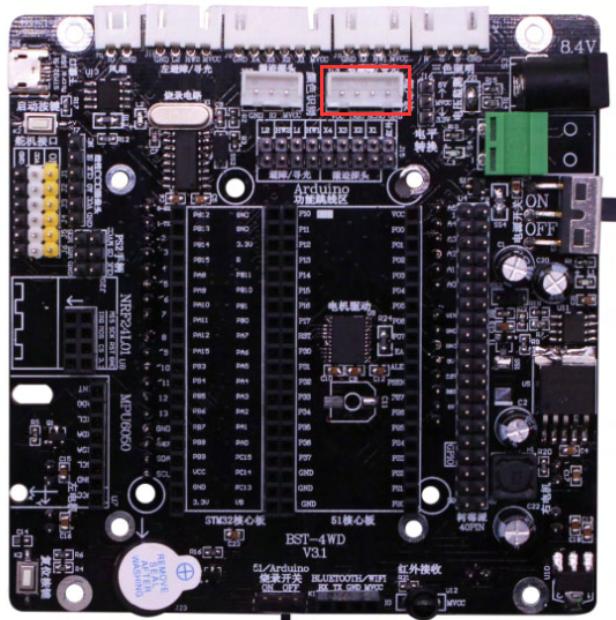
The interface has four pins: GND, LED R, LEDG, LEDB. When the LED R, LED G, and LED B pins are respectively turned on, the corresponding color lights (red, green, and blue) can be lit. When any two of the pins (or three pins) are at a high level, the RGB light will appear in a mixture of two colors (three colors).

#### Pin table:

Module interface	Arduino	51 controller	STM32	Raspberry Pi
------------------	---------	---------------	-------	--------------

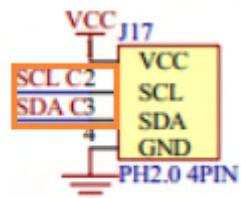
LED R	~11	P4.6 (EA)	PB1	IO22
LED G	~10	P4.5 (ALE)	PB0	IO27
LED B	~9	P4.4 (PSEN)	PA7	IO24

## 6. Ultrasonic module interface

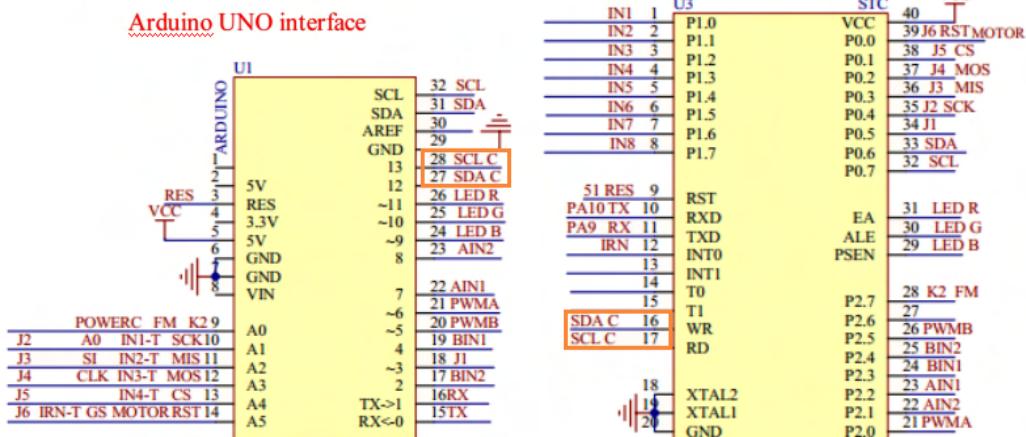


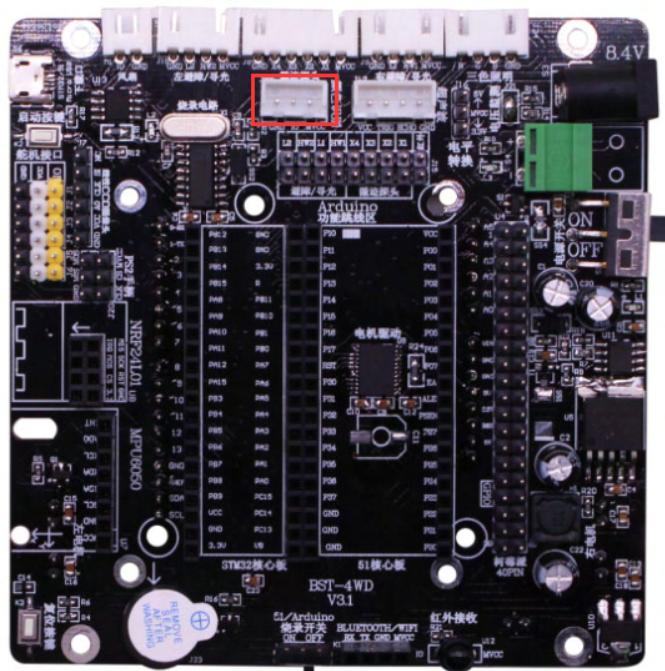
6-1 Position

### Ultrasonic sensor Module interface



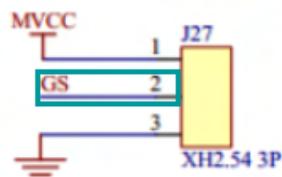
### 51 mcu core board interface



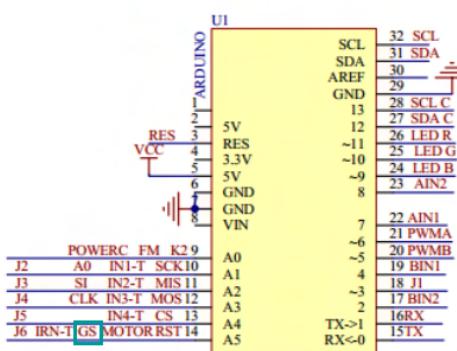


7-1 Position

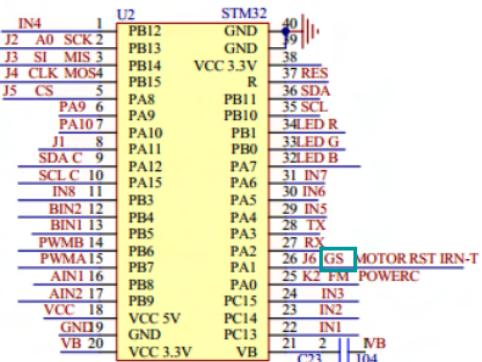
### Color recognize Module interface



Arduino UNO interface



STM32 core board interface



7-2 Schematic diagram

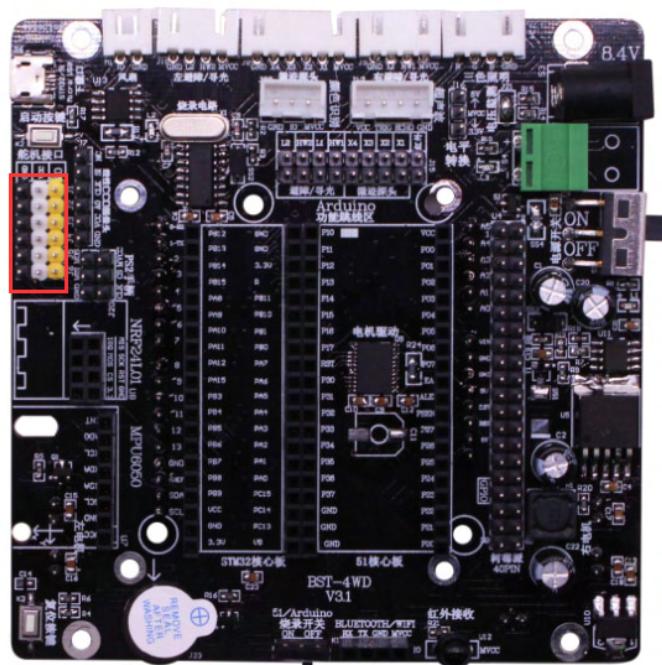
The interface has three pins: VCC, GND, and IO. VCC is 5V under normal operating conditions.

Pin table:

Module interface	Arduino	STM32
GS	A5	PA1

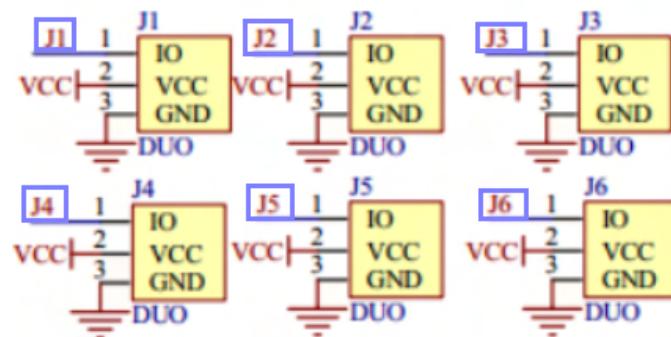
Note: This interface only supports Arduino and STM32 versions. The fan and infrared remote jumper caps need to be removed during use (explained later).

## 8. Servo interface

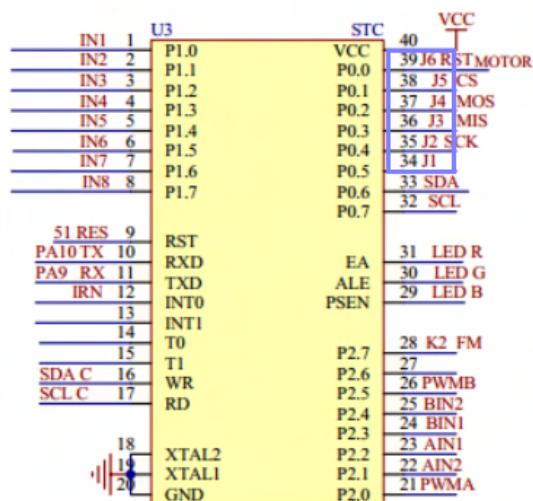
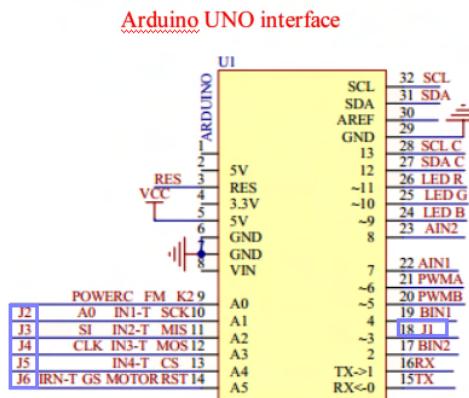


### 8-1 Position

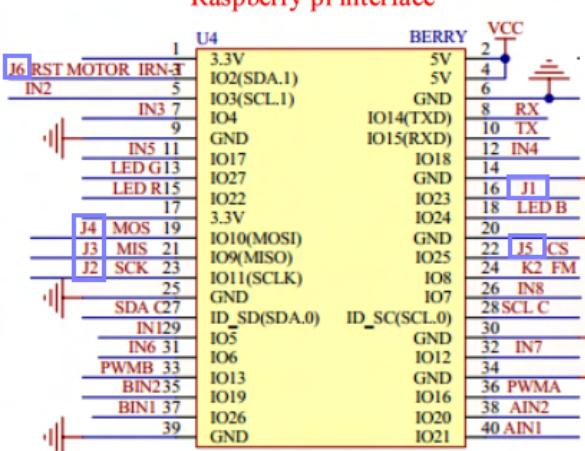
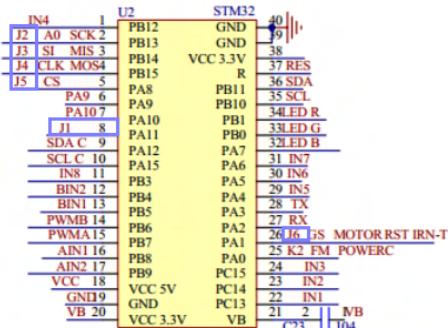
#### Servo interface



### 51 mcu core board interface



### STM32 core board interface



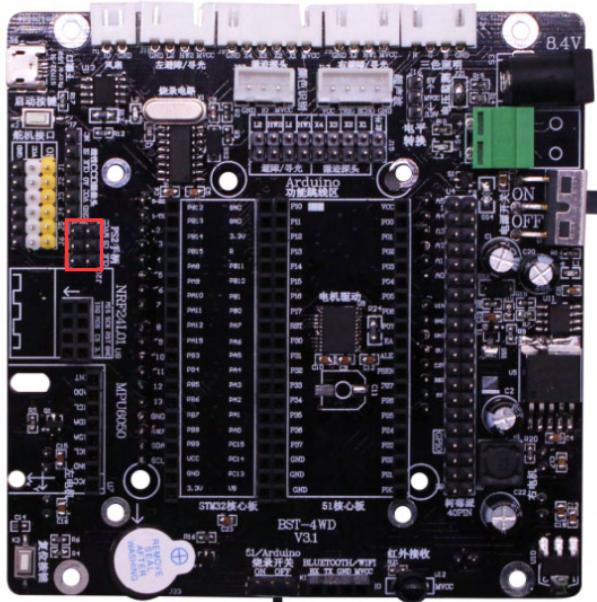
### 8-2 Schematic diagram

There are 6 servo interfaces, each with three pins: VCC, GND, and IO. VCC is 5V under normal operating conditions. It can simultaneously output 6 PWMs independently to control 6 servos.

### Pin table:

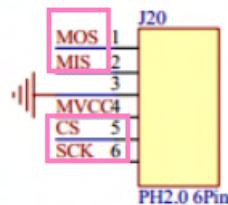
Module interface	Arduino	51 controller	STM32	Raspberry Pi
J1	~3	P0.5	PA11	IO23
J2	A1	P0.4	PB13	IO11
J3	A2	P0.3	PB14	IO9
J4	A3	P0.2	PB15	IO10
J5	A4	P0.1	PA8	IO25
J6	A5	P0.0	PA1	IO2

## 9. PS2controller receiving module interface



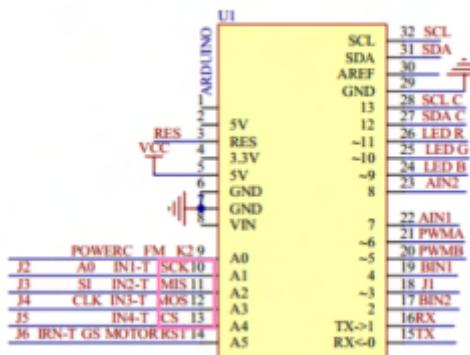
9-1 Position

### PS2 receive Module interface

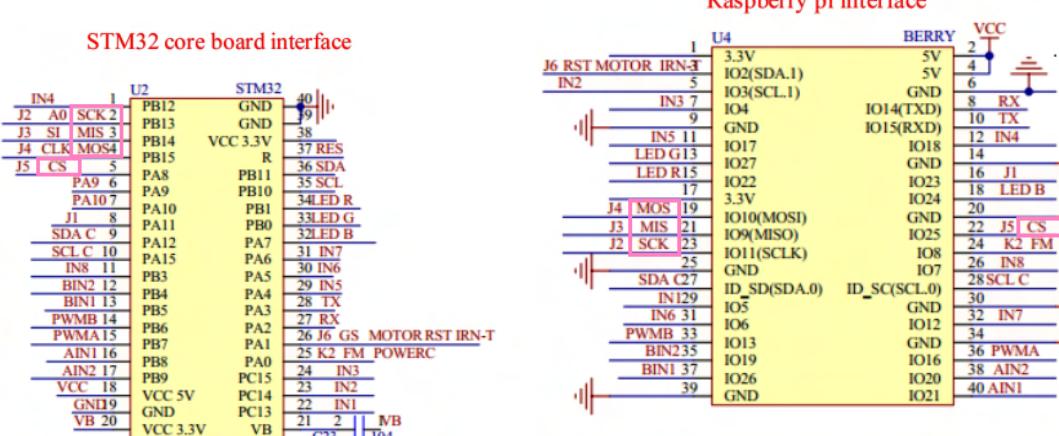


### 51 mcu core board interface

#### Arduino UNO interface



	IN1	U3	STC	VCC
	1	P1.0	VCC	39 J6 RST MOTOR
	2	P1.1	P0.0	38 J6 CS
	3	P1.2	P0.1	37 J6 MOS
	4	P1.3	P0.2	36 J6 MIS
	5	P1.4	P0.3	35 J2 SCK
	6	P1.5	P0.4	34 J1
	7	P1.6	P0.5	33 SDA
	8	P1.7	P0.6	32 SCL
			P0.7	
		51 RES 9	RST	31 LED R
		PA10 TX 10	EA	30 LED G
		PA9 RX 11	TXD	29 LED B
		IRN 12	ALE	
		13	PSEN	
		INT0		
		14		28 K2 FM
		T0	P2.7	27
		15	P2.6	26 PWMB
		21 PWMRA	P2.5	25 BIN2
		SDA C 16	P2.4	24 BIN1
		SCL C 17	P2.3	23 AIN1
			P2.2	22 AIN2
			P2.1	21 PWMA
			P2.0	



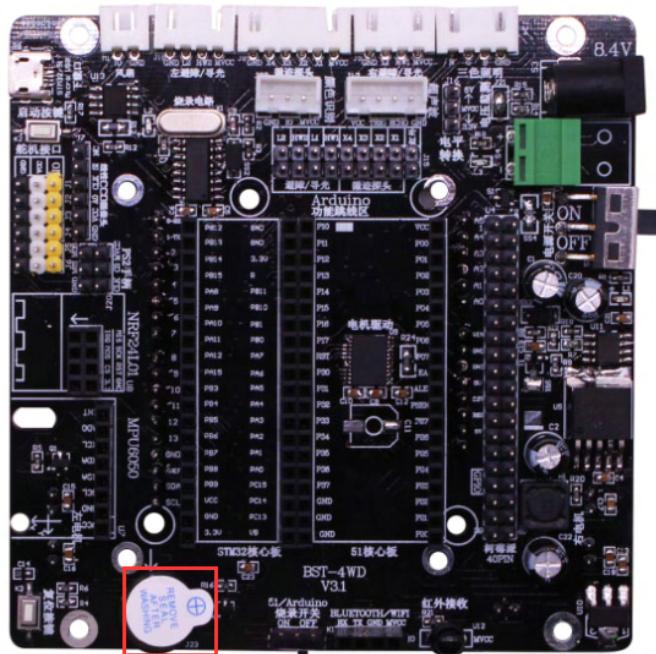
## 9-2 Schematic diagram

The interface has six pins: VCC, GND, MOS, MIS, CS, SCK. Under normal working conditions, VCC is 5V, and the SPI communication method is adopted between the receiver and the core controller.

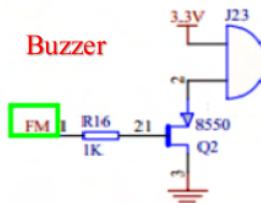
Pin table:

Module interface	Arduino	51controller	STM32	Raspberry Pi
CS	A4	P0.1	PA8	IO25
SCK	A1	P0.4	PB13	IO11
MOS	A3	P0.2	PB15	IO10
MIS	A2	P0.3	PB14	IO9

## 10.Buzzer

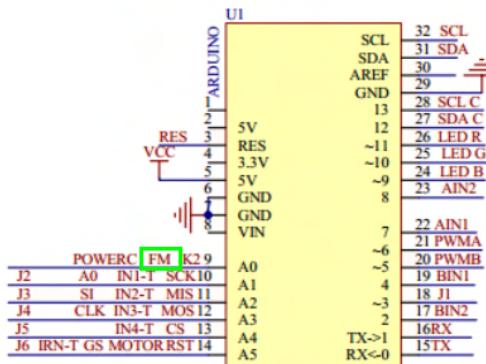


## 10-1 Position



51 mcu core board interface

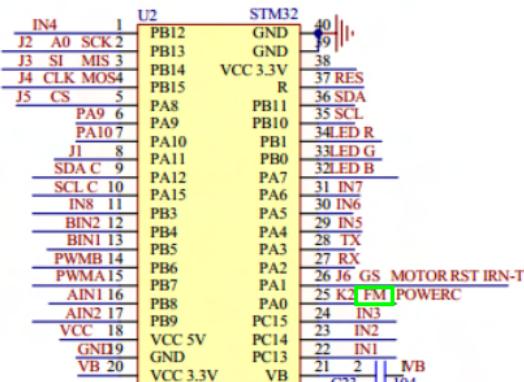
## Arduino UNO interface



## STM32 core board interface

IN1	1	U3	STC	VCC
IN2	2	P1.0	VCC	40
IN3	3	P1.1	P0.0	39 J6 RST MOTOR
IN4	4	P1.2	P0.1	38 J5 CS
IN5	5	P1.3	P0.2	37 J4 MOS
IN6	6	P1.4	P0.3	36 J3 MIS
IN7	7	P1.5	P0.4	35 J2 SCK
IN8	8	P1.6	P0.5	34 J1
		P1.7	P0.6	33 SDA
			P0.7	32 SCL
51 RES 9				
PA10	TX 10	RST		
PA9	RX 11	RXD	EA	31 LED R
		TXD	ALE	30 LED G
		INT0	PSEN	29 LED B
		INT1		
		T0	P2.7	28 K2 EM
		T1	P2.6	27
		WR	P2.5	26 PWM_B
		RD	P2.4	25 BIN2
			P2.3	24 BIN1
		XTAL2	P2.2	23 AIN1
		XTAL1	P2.1	22 AIN2
		GND	P2.0	21 PWMA
18				
	14			
	20			

## Raspberry pi interface



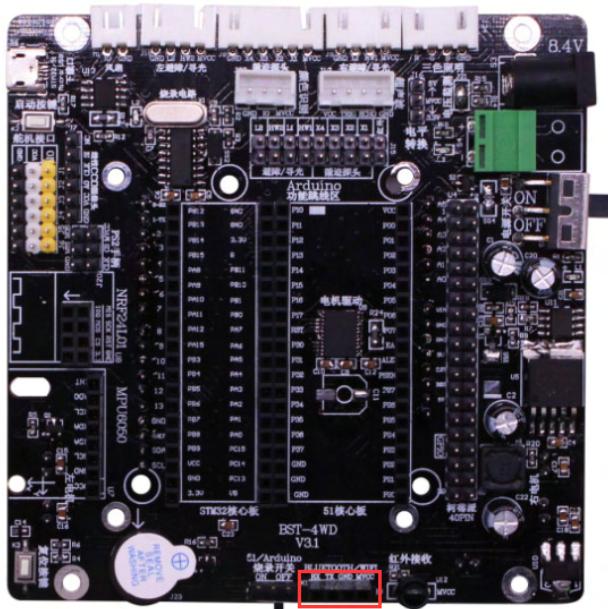
## 10-2 Schematic diagram

The active buzzer is used here.

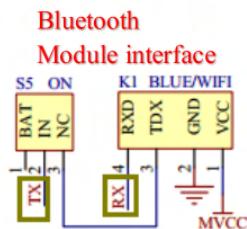
## Pin table:

Module interface	Arduino	51controller	STM32	Raspberry Pi
FM	A0	P2.7	PA0	IO8

## 11.Bluetooth/WIFI module interface



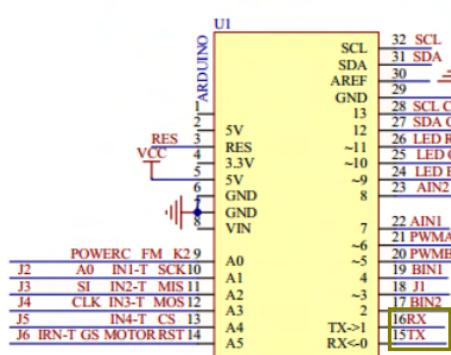
11-1 Position



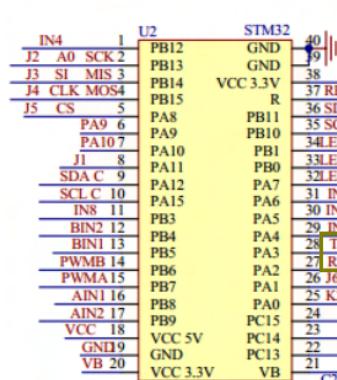
51 mcu core board interface

IN1	U3	STC	VCC
IN2	P1.0	VCC	39 J6 RSTMOTOR
IN3	P0.0	P0.0	38 J5 CS
IN4	P1.2	P0.1	37 J4 MOS
IN5	P1.3	P0.2	36 J3 MIS
IN6	P1.4	P0.3	35 J2 SCK
IN7	P1.5	P0.4	34 J1
IN8	P1.6	P0.5	33 SDA
	P1.7	P0.6	32 SCL
		P0.7	
	51 RES	RST	
	PA10 TX	RXD	31 LED R
	PA9 RX	TXD	30 LED G
	IRN 12	INT0	29 LED B
		INT1	
	13	T0	28 K2 FM
	PA10 SDA C	T1	27
	PA9 SCL C	WR	26 PWMB
	IRN 12	RD	25 BIN2
			24 BIN1
	14		23 AIN1
	15		22 AIN2
	16		21 PWMA
	SCL C		20 PWMB
	17		19 BIN1
			18 J1
	18		17 BIN2
	XTAL2		16 J2
	XTAL1		15 TX
	GND		14 IRN-T GS MOTOR RST

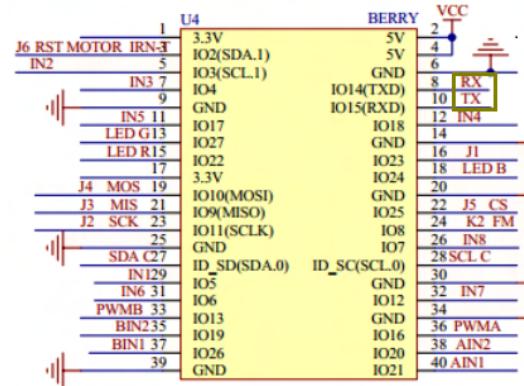
Arduino UNO interface



STM32 core board interface



## Raspberry pi interface



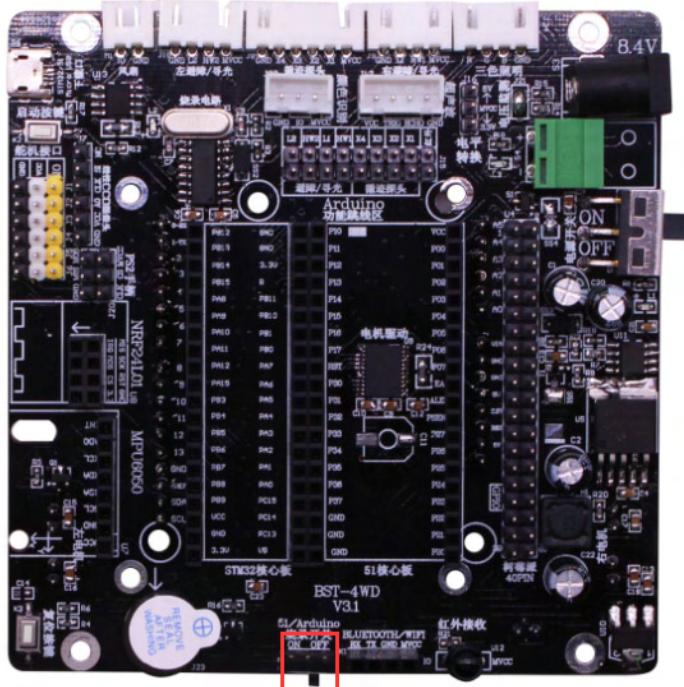
## 11-2 Schematic diagram

The interface has four pins: VCC, GND, TX, RX. VCC is 5V under normal working conditions, where Bluetooth uses serial communication.

## Pin table:

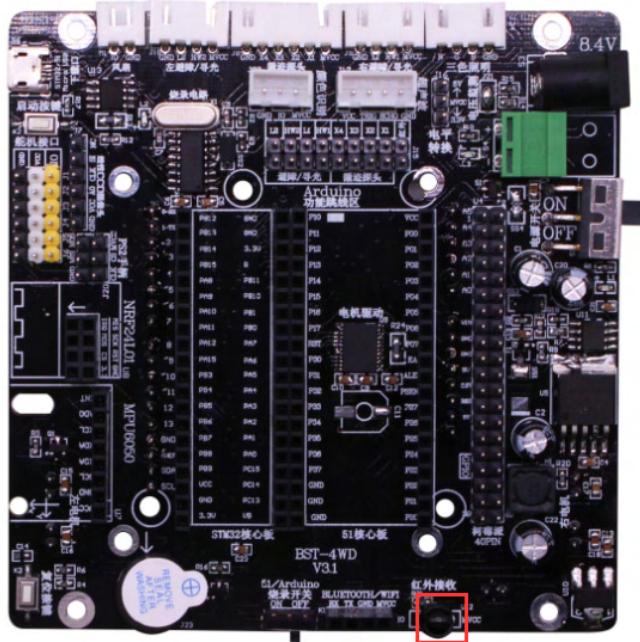
Module interface	Arduino	51controller	STM32	Raspberry Pi
TX	0	P3.0	PA3	IO15
RX	1	P3.1	PA2	IO14

Note: When using the serial port, for example: print data, burn the program through the CH340 serial port, you need to unplug the Bluetooth module. When using the Bluetooth module, you need to turn the programming switch next to the interface to OFF.



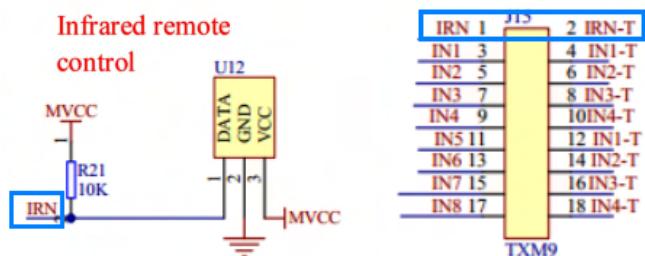
### 11-3 Switch position for burning

## 12. Infrared receiver

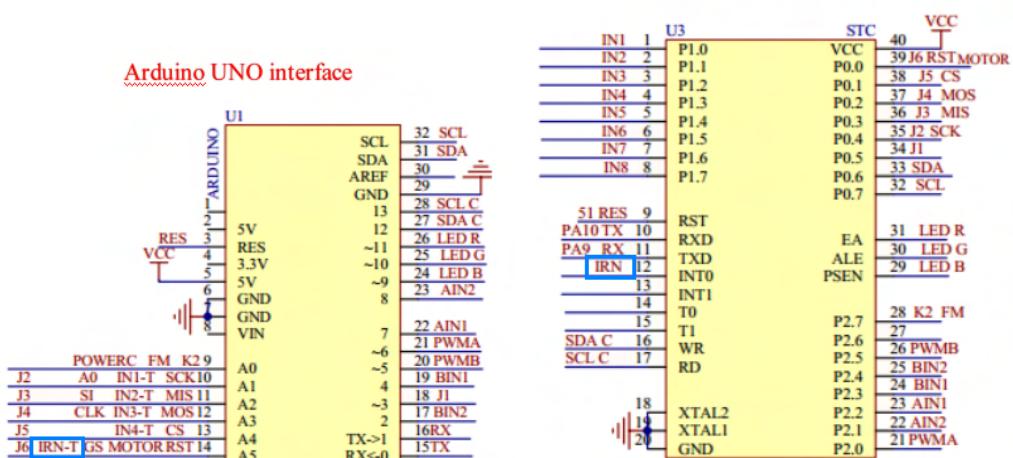


12-1 Position

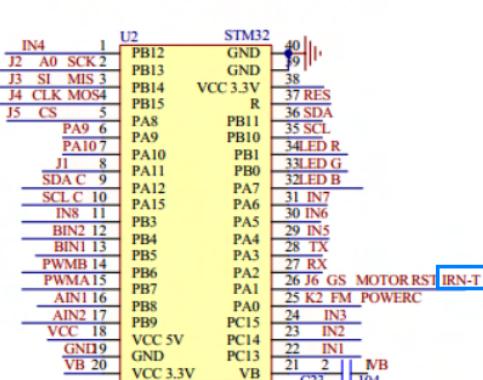
Arduino function switch



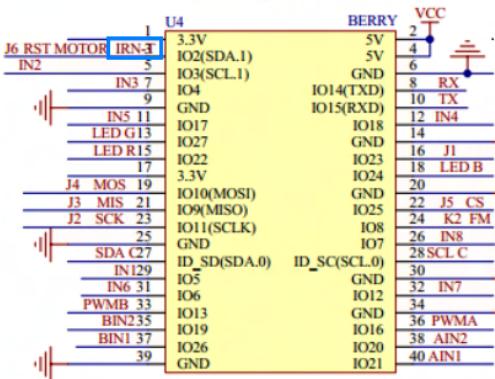
### 51 mcu core board interface



### STM32 core board interface



### Raspberry pi interface



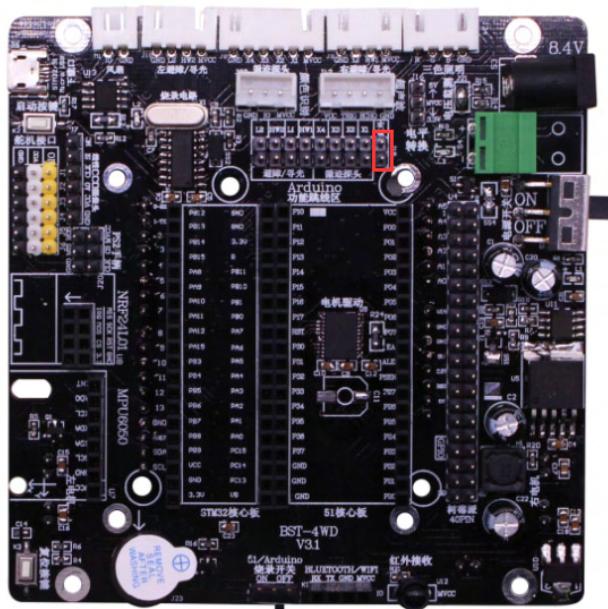
### 12-2 Schematic diagram

The infrared remote control function needs to use our dedicated remote control to be implemented normally.

Pin table:

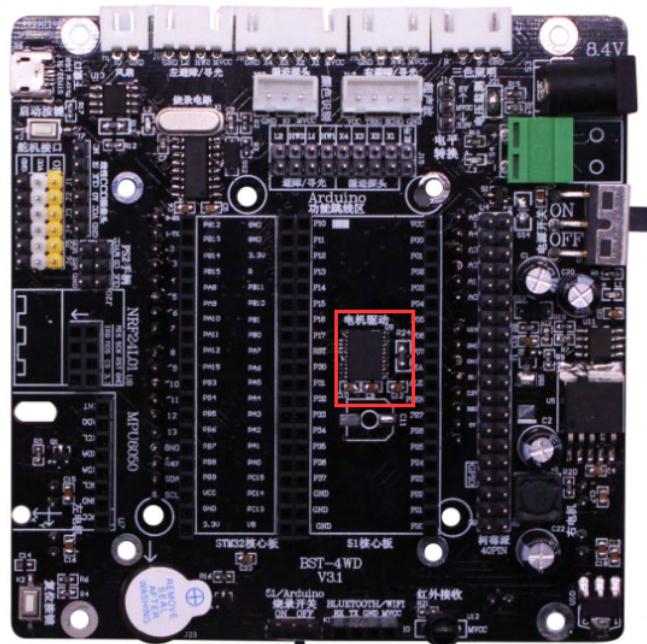
Module interface	Arduino	51controller	STM32	Raspberry Pi
IRN	A5	P3.2(INT0)	PA1	IO2

Note: When using the infrared remote control, you need to connect the infrared jumper cap on the expansion board. And unplug the grayscale (color recognition module) and the small fan.

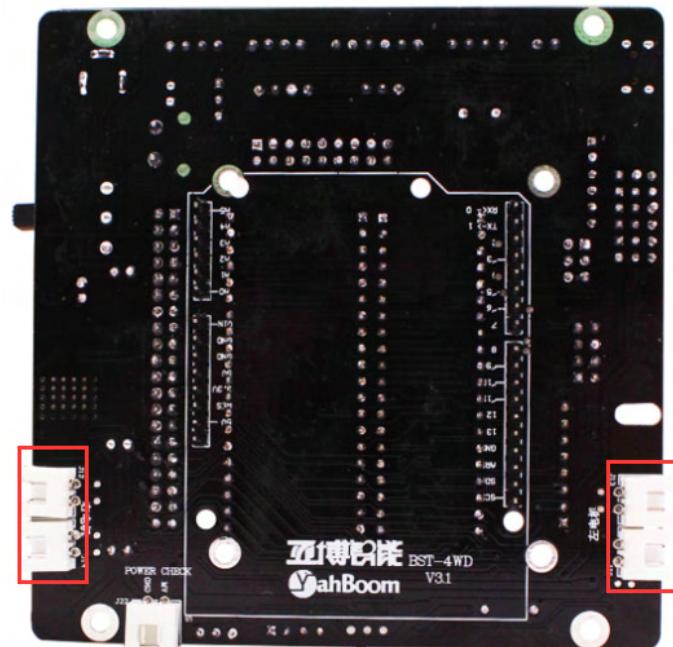


12-3 Position with jumper cap

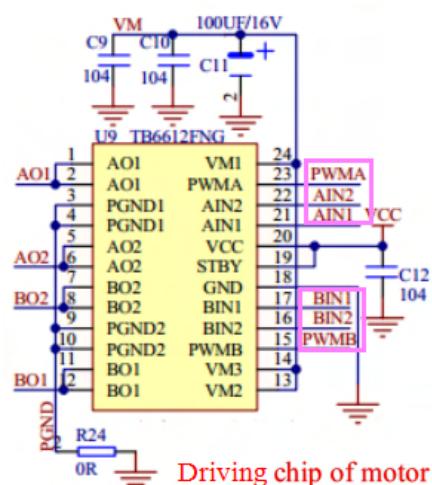
### 13. Motor driver circuit/interface



13-1 Motor driver chip position

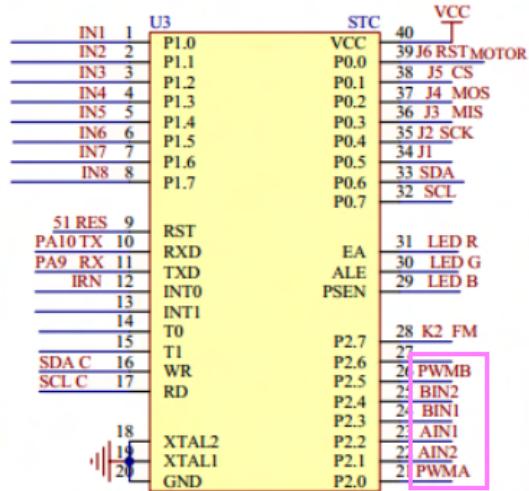
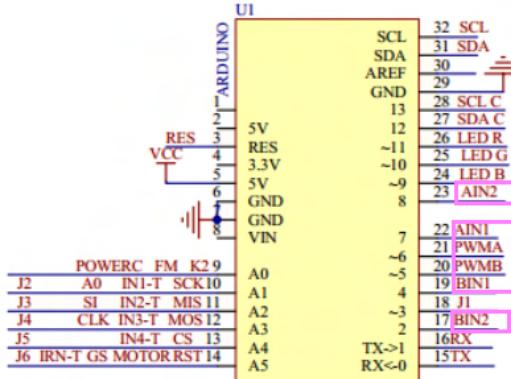


13-1 Motor interface position

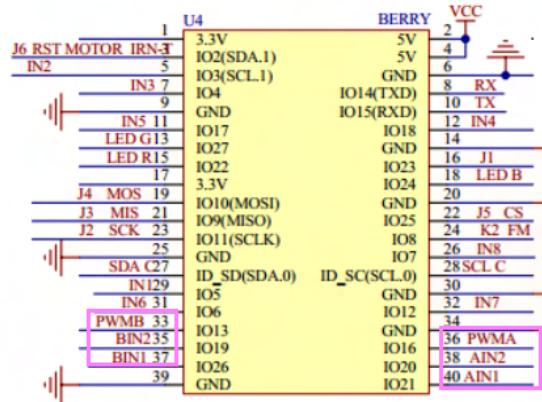
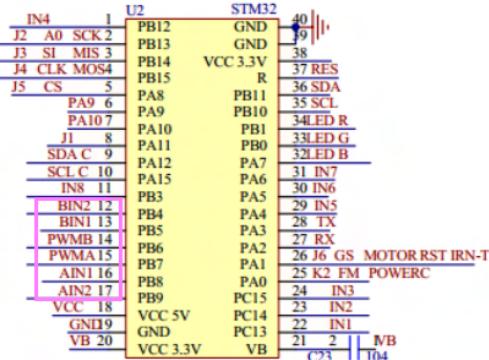


### 51 mcu core board interface

#### Arduino UNO interface



#### STM32 core board interface



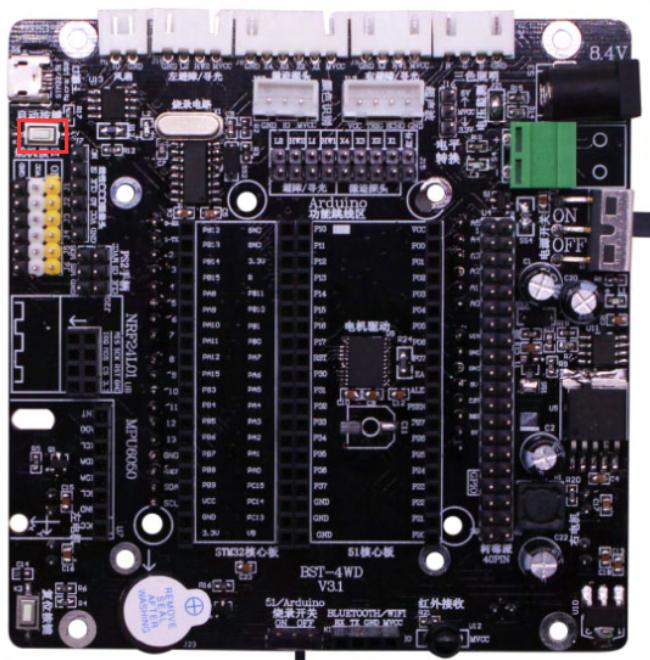
13-3 Schematic diagram

Control two motors with two PWM to adjust the direction and speed of the motor.

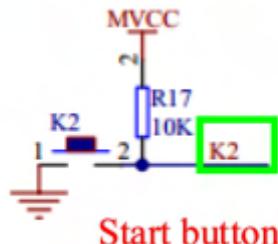
#### Pin table:

Module interface	Arduino	51controller	STM32	Raspberry Pi
PWMA	~6	P2.0	PB7	IO16
PWMB	~5	P2.5	PB6	IO13
AIN1	7	P2.2	PB8	IO21
AIN2	8	P2.1	PB9	IO20
BIN1	4	P2.3	PB5	IO26
BIN2	2	P2.4	PB4	IO19

#### 14. Start button



14-1 Position

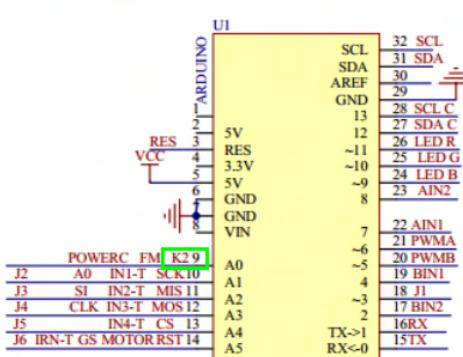


Start button

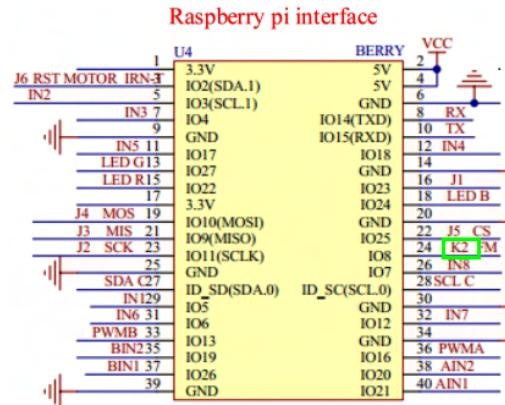
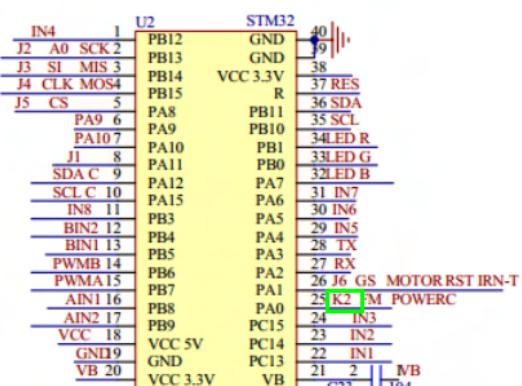
51 mcu core board interface

	IN1	1	U3	STC	VCC
	IN2	2	P1.0	40	39 J6 RST MOTOR
	IN3	3	P1.1	39	J5 CS
	IN4	4	P1.2	38	J4 MOS
	IN5	5	P1.3	37	J3 MIS
	IN6	6	P1.4	36	J2 SCK
	IN7	7	P1.5	35	J1
	IN8	8	P1.6	34	SDA
			P1.7	33	SCL
			RST	32	
			PA10 TX	27	K2 FM
			PA9 RX	27	LED R
			IRN 12	27	LED G
			INT0	27	LED B
			INT1	27	
			T0	27	
			SDA C	27	
			WR	27	
			RD	27	
			XTAL2	27	
			XTAL1	27	
			GND	27	

Arduino UNO interface



### STM32 core board interface



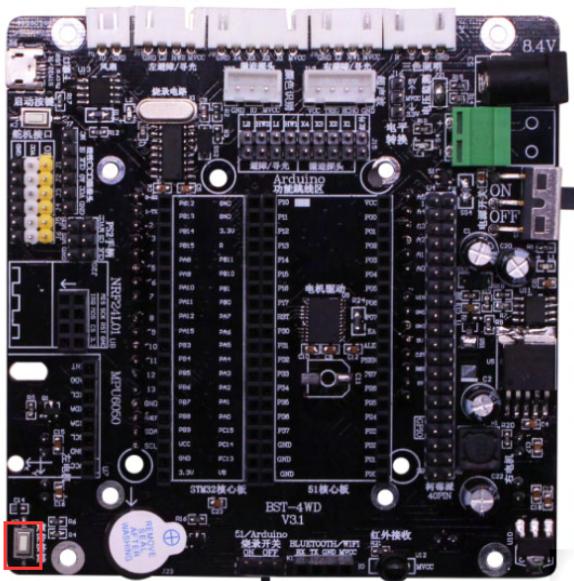
14-2 Schematic diagram

This is a start button that can be defined to turn on a feature.

Pin table:

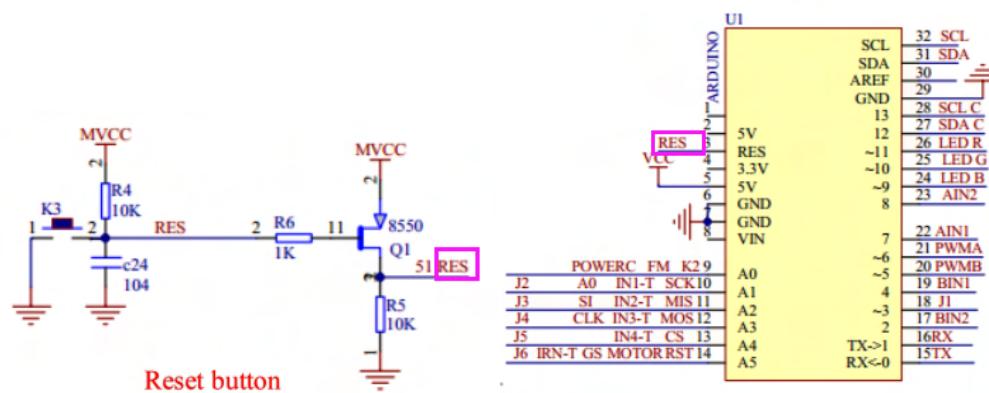
Module interface	Arduino	51controller	STM32	Raspberry Pi
K2	A0	P2.7	PA0	IO8

### 15. Reset button



15-1 Position

### Arduino UNO interface



### 51 mcu core board interface

	IN1	1	U3	STC	40	VCC
IN2	2	P1.0	VCC	39	J6 RST MOTOR	
IN3	3	P1.1	P0.0	38	J5 CS	
IN4	4	P1.2	P0.1	37	J4 MOS	
IN5	5	P1.3	P0.2	36	J3 MIIS	
IN6	6	P1.4	P0.3	35	J2 SCK	
IN7	7	P1.5	P0.4	34	J1	
IN8	8	P1.6	P0.5	33	SDA	
		P1.7	P0.6	32	SCL	
			P0.7			
51 RES	9	RST				
PA10 TX	10	RXD	EA	31	LED R	
PA9 RX	11	TXD	ALE	30	LED G	
IRN	12	INT0	PSEN	29	LED B	
	13	INT1				
	14	T0		28	K2 FM	
SDA C	16	T1	P2.7	27		
SCL C	17	WR	P2.6	26	PWMB	
		RD	P2.5	25	BIN2	
			P2.4	24	BIN1	
			P2.3	23	AIN1	
			P2.2	22	AIN2	
			P2.1	21	PWMA	
	18	XTAL2				
	19	XTAL1				
	20	GND	P2.0			

### STM32 core board interface

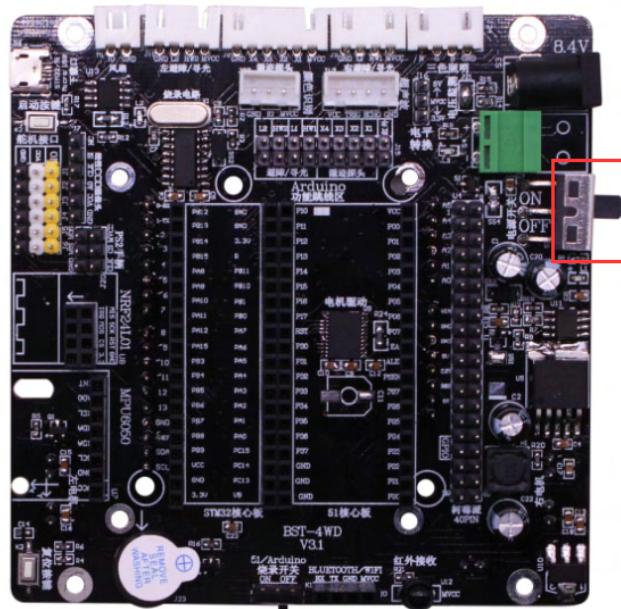
	IN4	1	U2	STM32	40	
J2	A0	SCK 2	PB12	GND	39	
J3	SI	MIS 3	PB13	GND	38	
J4	CLK	MOS4	PB14	VCC 3.3V		
J5	CS	5	PB15	R	37	RES
			PA9	PA8	36	SDA
			PA10	PA9	35	SCL
			J1	PA10	34	LED R
			PA11	PA11	33	LED G
			PA12	PA12	32	LED B
			PA15	PA15	31	IN7
			PA16	PA16	30	IN6
			PA17	PA17	29	IN5
			PA18	PA18	28	TX
			PA19	PA19	27	RX
			PA20	PA20	26	J6 GS MOTOR RST IRN-T
			PA1	PA1	25	K2 FM POWERC
			PA0	PA0	24	IN3
			PC15	PC15	23	IN2
			PC14	PC14	22	IN1
			PC13	PC13	21	2 WB
			VB	VB	C23	T04
			VCC 3.3V	VCC 3.3V		
			GND	GND		
			VB	VB		

### 15-2 Schematic diagram

This is a reset button that re-energizes the expansion board, which is equivalent to restarting the expansion board and the core controller.

**Note:** The Raspberry Pi core controller does not need to connect to this button.

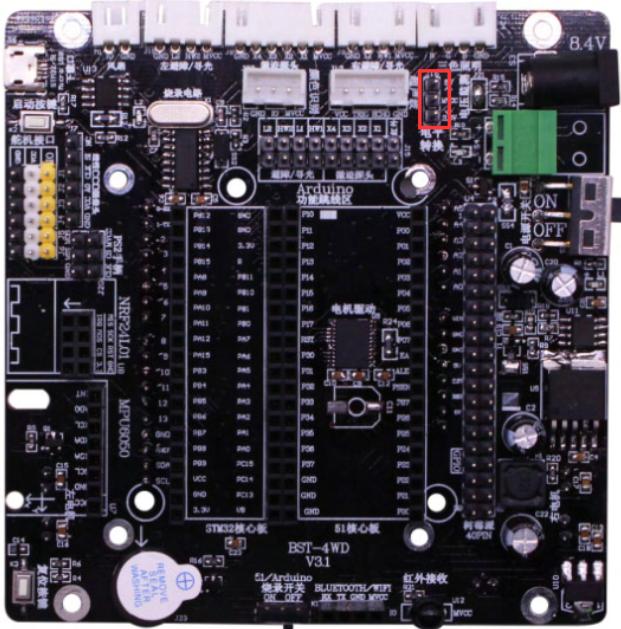
### 16. Power button



16-1 Position

It is used to control the power switch of the expansion board.

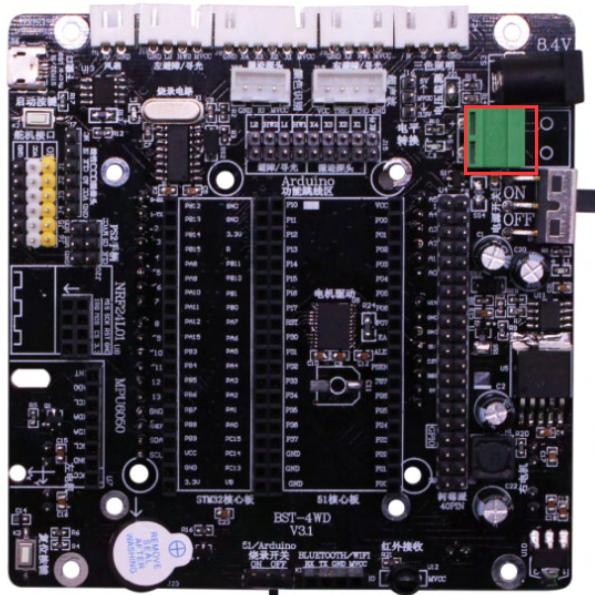
### 17. 5V/3.3V power switch



17-1 Position

If the top two pin headers are connected by a jumper cap, it is a 5V power supply; if the lower two pin headers are connected, it is a 3.3V power supply.

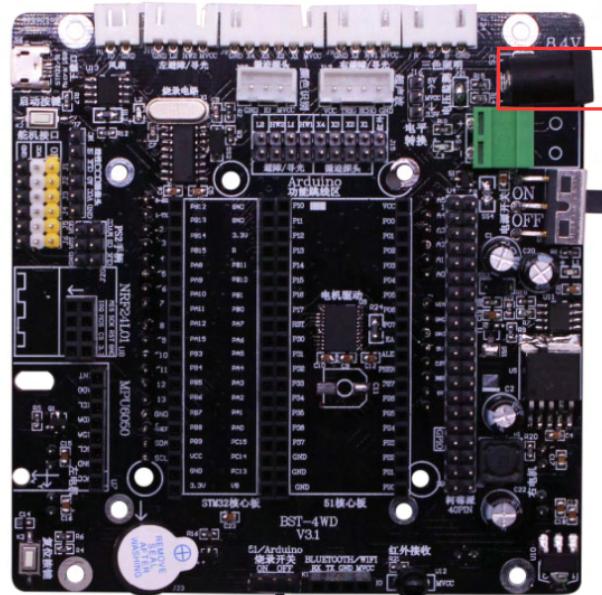
### 18. Battery box interface



### 18-1 Position

Plug the special battery box here to supply power to the expansion board. The power supply voltage of the expansion board cannot exceed 12.6v.

### 19. Charging socket



### 19-1 Position

Here you can charge the battery with a 12.6V in-line charger.