Radar following

Radar following
- Hardware connection
- Control principle
- Main Code
- Program flow chart
- Experimental phenomenon

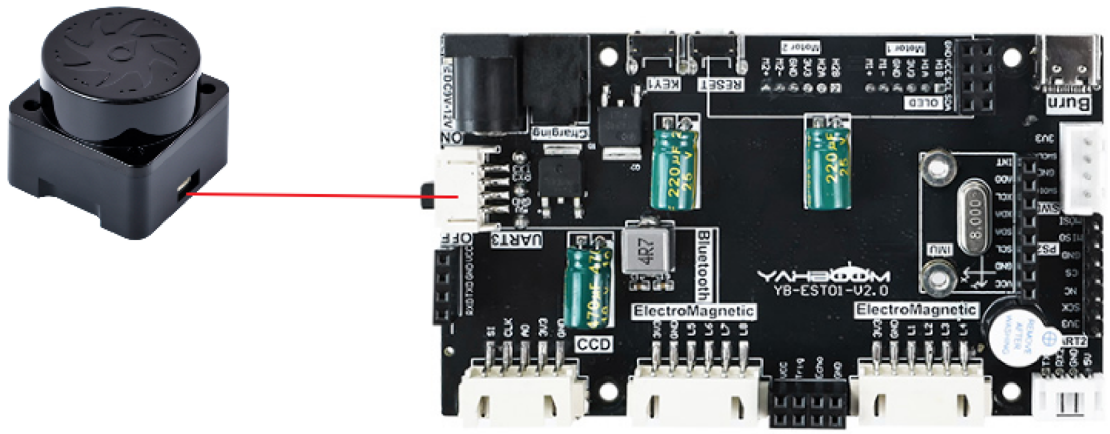
The tutorial mainly demonstrates the following function of the balance car combined with the Tmini-Plus radar.

```
The tutorial only introduces the standard library project code
```

Hardware connection

Since we have configured a special connection line, we only need to install it to the corresponding interface.

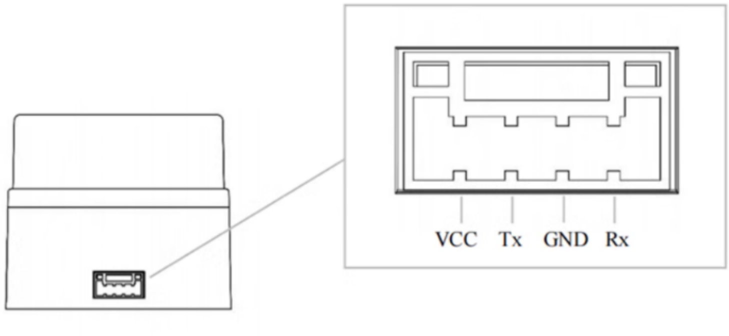
Peripherals	Development board
Tmini-Plus radar: VCC	5V
Tmini-Plus radar: TXD	PC10
Tmini-Plus radar: RXD	PC11
Tmini-Plus radar: GND	GND



Control principle

The program analyzes the radar data and determines the direction of movement of the balance car based on the distance information at the specified angle.

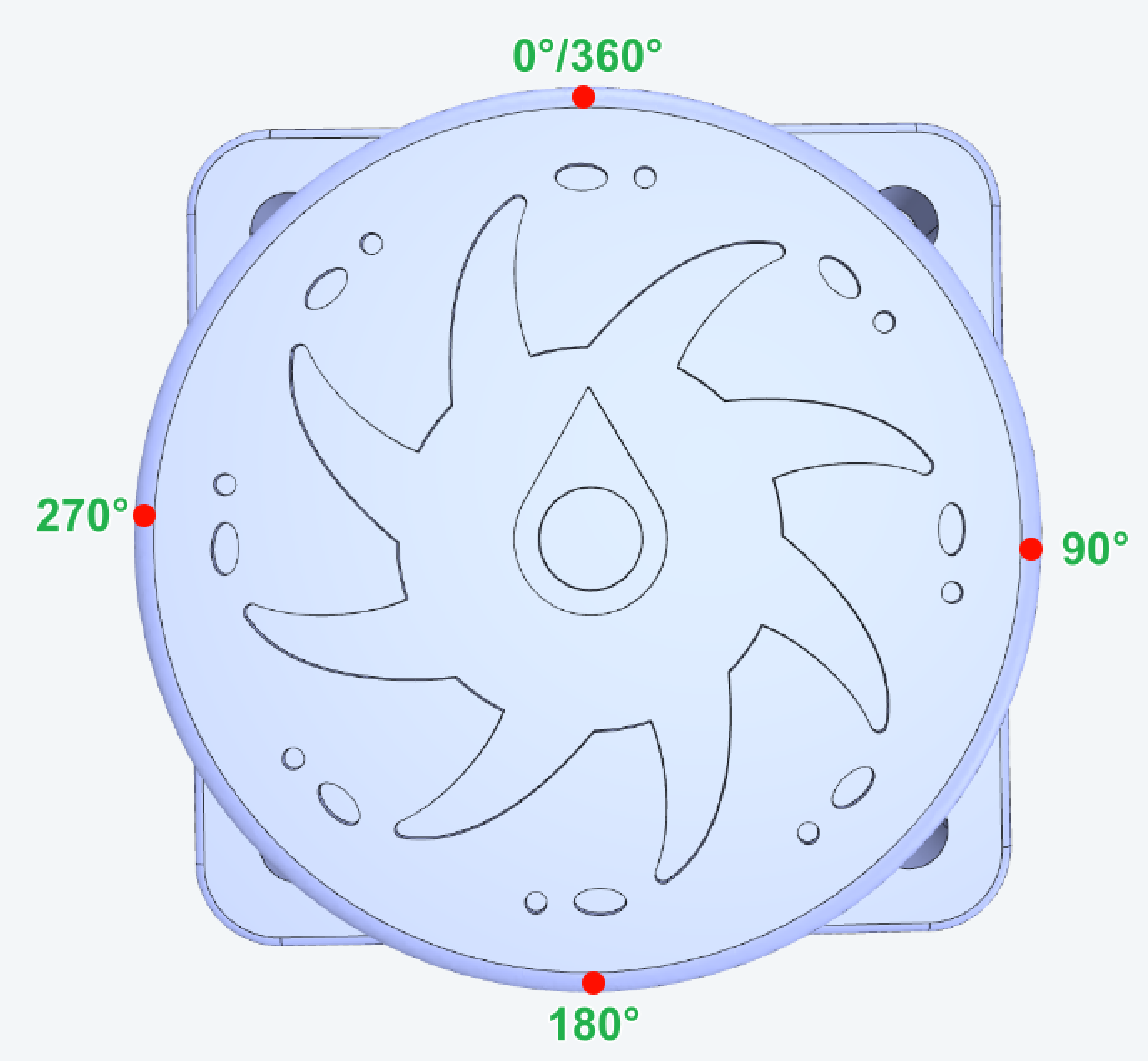
- Tmini-Plus radar



Product name	**Tmini-Plus radar**
Scanning frequency	6-12Hz
Sampling frequency	4000 times/s
Measuring radius	Black object: 12m
Minimum measuring distance	0.05m
Ranging principle	TOF ranging
Scanning angle	360°
Communication interface	Standard asynchronous serial port (UART) 1. Baud rate: 230400 2. Data bits: 8 3. Check bit: None 4. Stop bit: 1
ROS support	ROS1/ROS2
Windows support	Host computer

Radar Angle Distribution

The arrow in the center of the radar points to 0°/360°, and the angle increases clockwise.



> **Communication Protocol**

```
For detailed information, please refer to the "T_Mini_Plus Manual"
```

Main Code

The tutorial mainly explains the code for the radar following function. For detailed code, please refer to the corresponding project file.

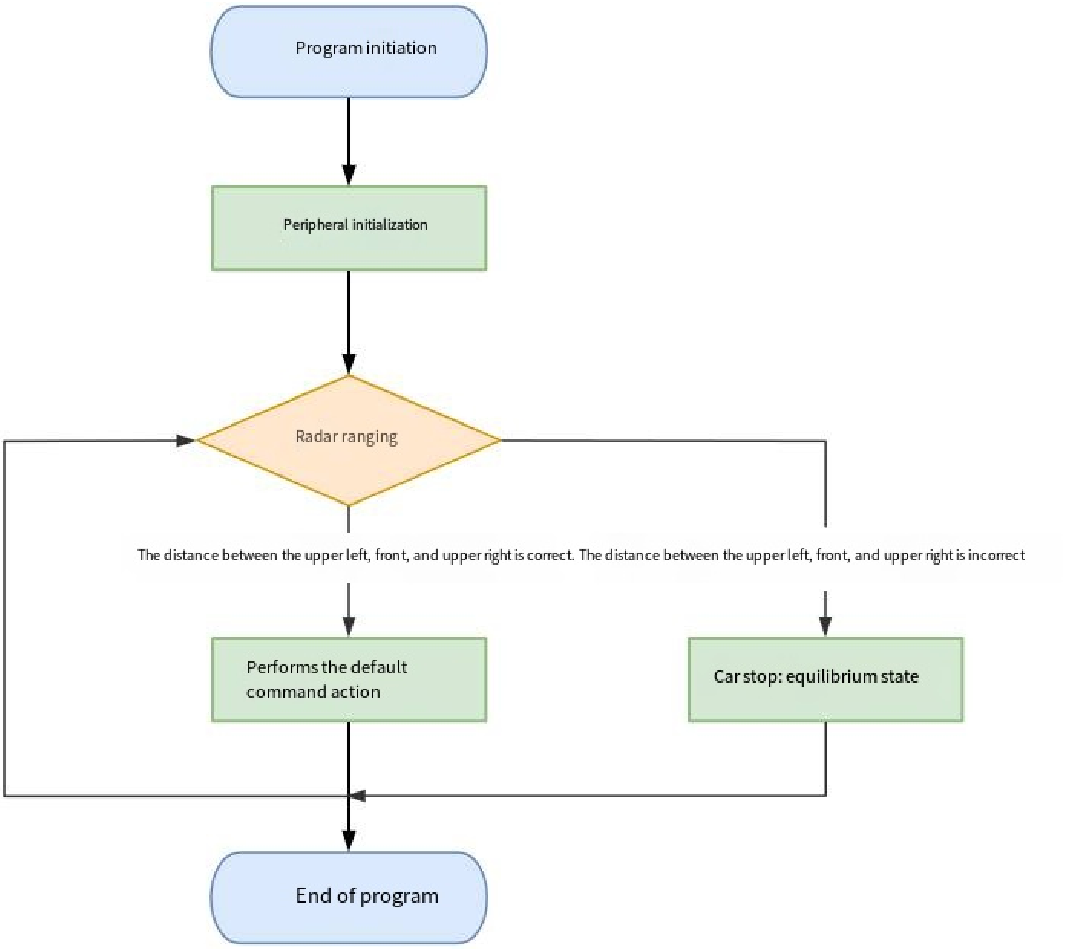
> **Car_Follow**

The distance data of the radar at 0°, 45°, and 315° is used to determine the position of the following object and control the movement of the balance car.

```
void Car_Follow(void)
{
    float get_data_mid = Tminidis[0];

    float get_data_LL = Tminidis[315];

    float get_data_RR = Tminidis[45];

    printf("data = %.2f\t %.2f\t %.2f\r\n",get_data_mid,get_data_LL,get_data_RR);

    // positive direction
    if(get_data_mid < 130 && get_data_mid>0)
    {
        //The car moves backward
        Move_X = -15;
        Move_Z = 0;
    }
    else if (get_data_mid > 220 && get_data_mid < 350)
    {
        //The car moves forward
        Move_X = 15;
        Move_Z = 0;
    }

    // Left direction
    else if (get_data_LL < 340 && get_data_mid>200)
    {
        // Left rotation of the car
        Move_X = 0;
        Move_Z = -450;
    }

    // Right direction
    else if (get_data_RR < 340 && get_data_mid>200)
    {
        //The car turns right
        Move_X = 0;
        Move_Z = 450;
    }

    else//Not following
    {
        Move_X = 0;
        Move_Z = 0;
    }
}
```

Program flow chart

Briefly introduce the process of function implementation:


Program initiation
Peripheral initialization
Radar ranging
The distance between the upper left, front, and upper right is correct. The distance between the upper left, front, and upper right is incorrect
Performs the default command action
Car stop: equilibrium state
End of program

Experimental phenomenon

> ### Software code

The Balance_Radar_Follow.hex file generated by the project compilation is located in the OBJ folder of the Balance_Radar_Follow project. Find the Balance_Radar_Follow.hex file corresponding to the project and use the FlyMcu software to download the program into the development board.

```
Product supporting data source code path: Attachment → Source code summary →
5.Balanced_Car_Extended → 14.Balance_Radar_Follow
```

> ### Experimental phenomenon

After the program is started, press KEY1 according to the OLED prompt to start the radar following function of the balance car: OLED displays start control!

The radar detects the following object in front (0mm<following object distance<130mm): the car moves backward

The radar detects the following object in front (200mm<following object distance<340mm): move forward

The radar detects the following object on the left (the distance to the left of the following object <340mm, the distance in front>200mm): the car turns left

The radar detects the following object on the right (the distance to the right of the following object <340mm, the distance in front>200mm): the car turns right

```
The program has voltage detection. If the voltage is less than 9.6V, a low voltage alarm is triggered and the buzzer will sound.
Common situations that trigger voltage alarms:
1. The power switch of the development board is not turned on, and only the Type-C data cable is connected for power supply
2. The battery pack voltage is lower than 9.6V and needs to be charged in time
```