Lesson 5 Auto Follow

CONTENT

Ⅰ. Brief Introduction

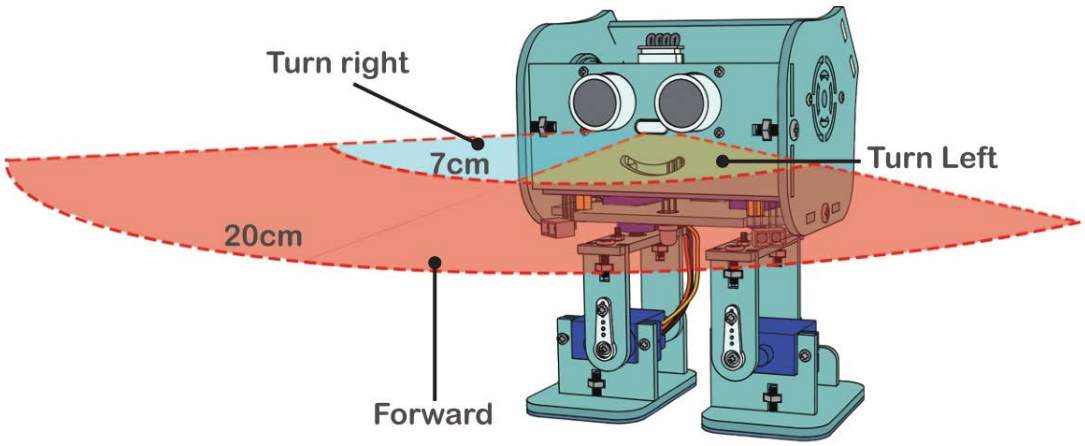
In this chapter you will learn the principle of the auto follow mode of penguin bot and how to write program to control penguin bot to automatically follow an object.

Ⅱ. Principle of Auto Follow

HC-SR04 Ultrasonic Sensor Module

ST188 infrared sensor

Auto Follow Mode will use ultrasonic sensor and ST188 infrared sensor.



Tips: If you have any questions or run into any problems during assembling and testing Penguin Bot please feel free to contact us at service@elegoo.com or euservice@elegoo.com (Europe customers).

Ultrasonic sensor detects the distance of the obstacle in front and ST188 detects whether there are obstacles on the left and right side.

If Penguin Bot detects obstacle on the left side it will turn left and if Penguin Bot detects obstacle on the right side it will turn right. If the distance from the obstacle is more than 20 cm, Penguin Bot will stop moving. Otherwise it will continue moving forwards.

Ⅲ. Write Program of Auto Follow

The sketch used in this chapter is saved in below path and please refer to Upload Penguin Bot program and upload the codes.

\ELEGOO Penguin Bot V2.0\Penguin Bot Function Introduction\ Lesson 5 Auto Follow\Follow\ Follow.ino

Code reviews:

```
* The following is the core part of the code for this case (please
refer to the corresponding source project for details)

#include "Oscillator.h"
#include <Servo.h>

 #define YL_PIN 10
 #define YR_PIN 9
 #define RL_PIN 12
 #define RR_PIN 6
 #define ECHO_PIN 4
 #define TRIG_PIN 5
 #define ST188_R_PIN A1
 #define ST188_L_PIN A0
void followMode()
{
  double distance_value = getDistance();

   if (distance_value >= 1 && distance_value <= 500)
   {
      int st188Val_L = analogRead(ST188_L_PIN);
      int st188Val_R = analogRead(ST188_R_PIN);
      if (st188Val_L >= 400 && st188Val_R >= 400)
```

Macro definition: hardware device IO interface

Get: Ultrasonic sensor data

Judgment: Whether the ultrasonic data is valid

Get: ST188 data

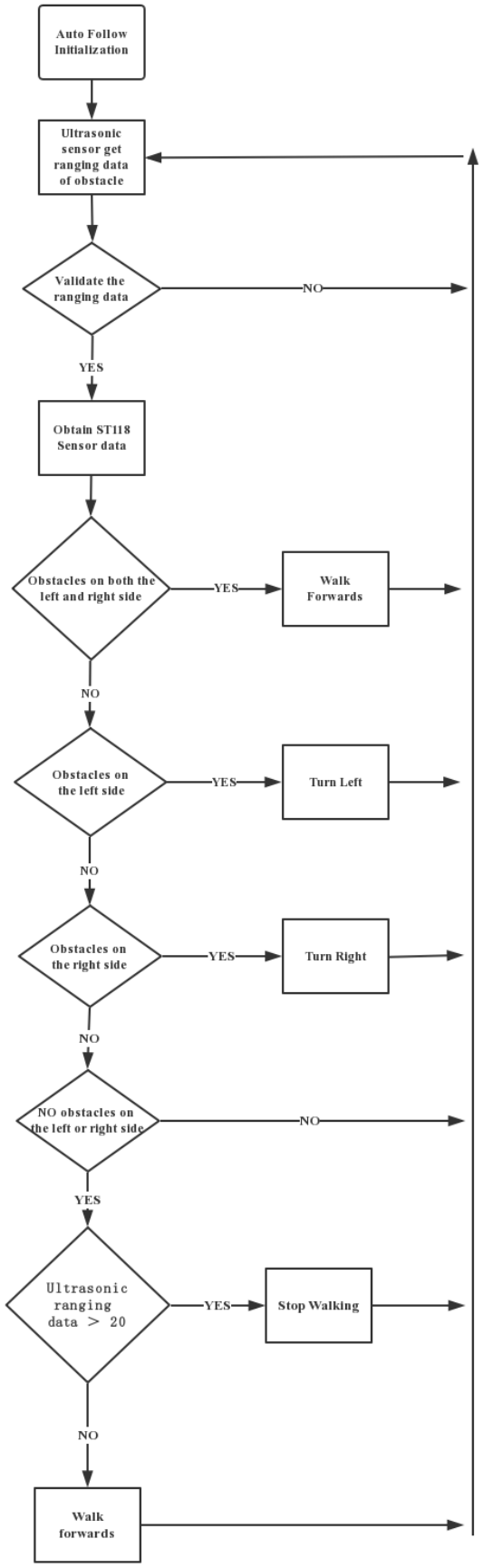
```
        {
            walk(1, t * 3, 1);
        }
        else if (st188Val_L >= 400 && st188Val_R < 400)
        {
            turn(1, t * 4, -1);
        }
        else if (st188Val_L < 400 && st188Val_R >= 400)
        {
            turn(1, t * 4, 1);
        }
        else if (st188Val_L < 400 && st188Val_R < 400)
        {
            if (distance_value > 20){ homes(0);}
            else walk(1, t * 3, 1);
        }
    }
    else homes(0);
}
void setup()
{
    pinMode(ECHO_PIN, INPUT);
    pinMode(TRIG_PIN, OUTPUT);
    servo[0].attach(RR_PIN);
    servo[1].attach(RL_PIN);
    servo[2].attach(YR_PIN);
    servo[3].attach(YL_PIN);
    homes(100);
}
void loop()
{
    followMode();
}
```

Judgment: Are there obstacles on the left and right sides

Judgment: There are objects close to 20cn in the positive direction

Execution: Follow mode

Workflow of auto follow



Tips: If you have any questions or run into any problems during assembling and testing Penguin Bot please feel free to contact us at service@elegoo.com or euservice@elegoo.com (Europe customers).

From the workflow above we know that Penguin Bot will first detect whether there are obstacles on both sides and if yes it will continue walking forwards; If Penguin Bot detects obstacle on the left side it will turn left and if Penguin Bot detects obstacle on the right side it will turn right. If the distance from the obstacle is more than 20 cm, Penguin Bot will stop moving. Otherwise it will continue moving forwards.