

# Infrared sensor

## Infrared sensor

- 1、software-hardware
- 2、Brief principle
  - 2.1、Hardware schematic diagram
  - 2.2、Physical connection diagram
  - 2.3、Principle of control
- 3、Engineering configuration
  - 3.1、Notes
  - 3.2、Pin configuration
- 4、Main Function
  - User function
- 5、Experimental phenomenon

This tutorial demonstrates how to print the converted values of two infrared obstacle avoidance module ADMs via **serial port (USART1)** .

## 1、software-hardware

- **STM32F103CubeIDE**

- **STM32 robot expansion board**

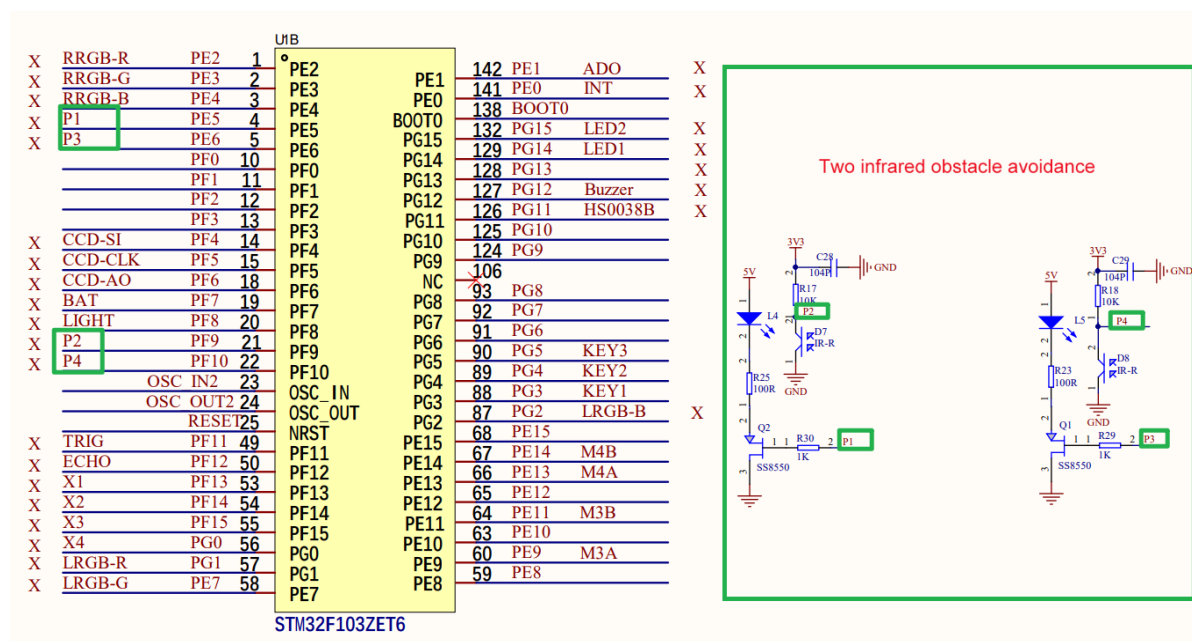
Two infrared obstacle avoidance modules are integrated on the development board

- **Type-C cable or ST-Link**

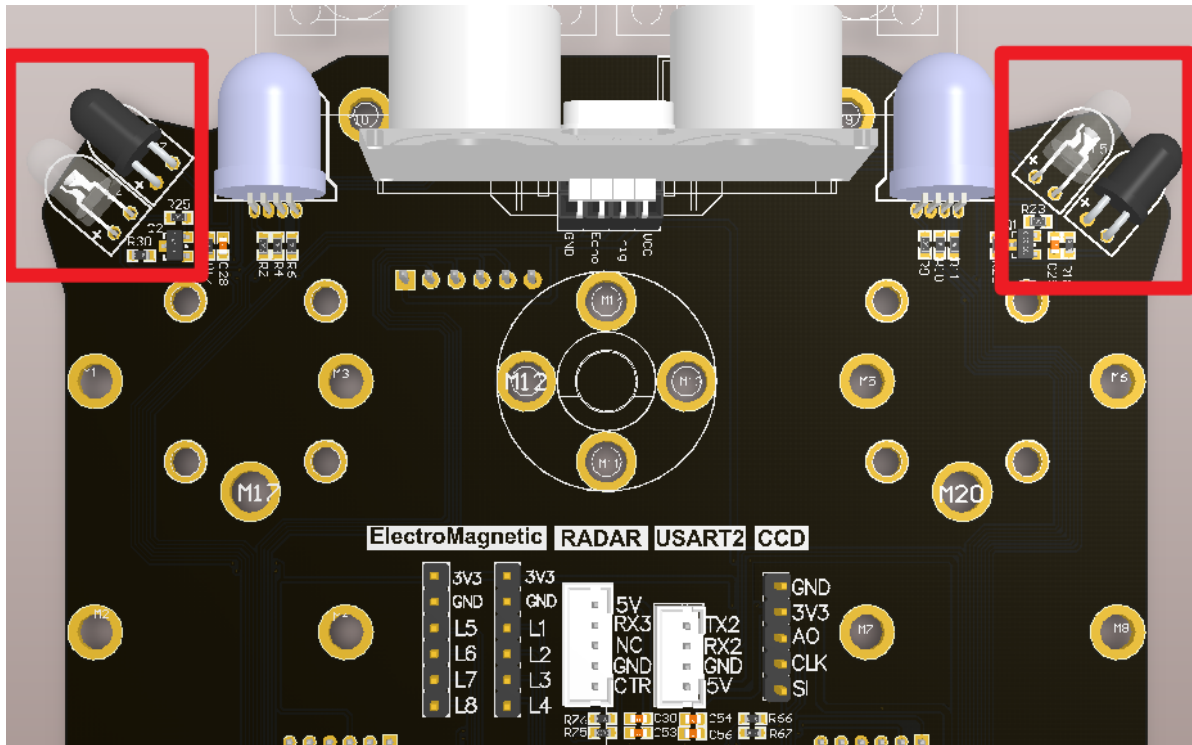
Download or simulate the program of the development board

## 2、Brief principle

### 2.1、Hardware schematic diagram



## 2.2、Physical connection diagram



## 2.3、Principle of control

The analog voltage output of infrared obstacle avoidance module can be converted to digital value by two ADC conversion. The range is 0 to  $2^{12}-1$  (that is, 0 to 4095).

Infrared obstacle avoidance (development board integration)	Corresponding pin
P1	PE5 (Control the left obstacle avoidance module switch)
P2	PF9 (ADC3_IN7)
P3	PE6 (Control the switch of the right obstacle avoidance module)
P4	PF10 (ADC3_IN7)

## 3、Engineering configuration

**Project Configuration:** Prompts for configuration options in the STM32CubeIDE project configuration process

### 3.1、Notes

Omitted project configuration: **New project, chip selection, project configuration, SYS for pin configuration, RCC configuration, clock configuration, and project configuration** content

The project configuration part, which is not omitted, is the key point to configure in this tutorial.

Please refer to [2, development environment construction and use: STM32CubeIDE installation - Use] to understand how to configure the omitted part of the project

## 3.2、Pin configuration

- GPIO

The screenshot shows the STM32CubeIDE interface with the 'Pinout & Configuration' tab selected. The left sidebar shows the 'System Core' category with 'GPIO' highlighted. The main area displays the 'GPIO Mode and Configuration' window. The 'Configuration' section shows a table of pins with the following data:

Pin Name	Signal on Pin	GPIO output I...	GPIO mode	GPIO Pull-up...	Maximum ou...	User Label	Modified
PE5	n/a	Low	Output Push ...	No pull-up an...	Medium	Left_Switch_I...	✓
PE6	n/a	Low	Output Push ...	No pull-up an...	Medium	Right_Switch...	✓

Below the table, the 'PE5 Configuration' section shows the following settings:

- GPIO output level: Low
- GPIO mode: Output Push Pull
- GPIO Pull-up/Pull-down: No pull-up and no pull-down
- Maximum output speed: Medium
- User Label: Left\_Switch\_Iravid

- USART

The screenshot shows the STM32CubeIDE interface with the 'Pinout & Configuration' tab selected. The left sidebar shows the 'Connectivity' category with 'USART1' highlighted. The main area displays the 'USART1 Mode and Configuration' window. The 'Mode' section shows the following settings:

- Mode: Asynchronous
- Hardware Flow Control (RS232): Disable

The 'Configuration' section shows the 'Parameter Settings' tab selected. The 'Basic Parameters' section shows the following settings:

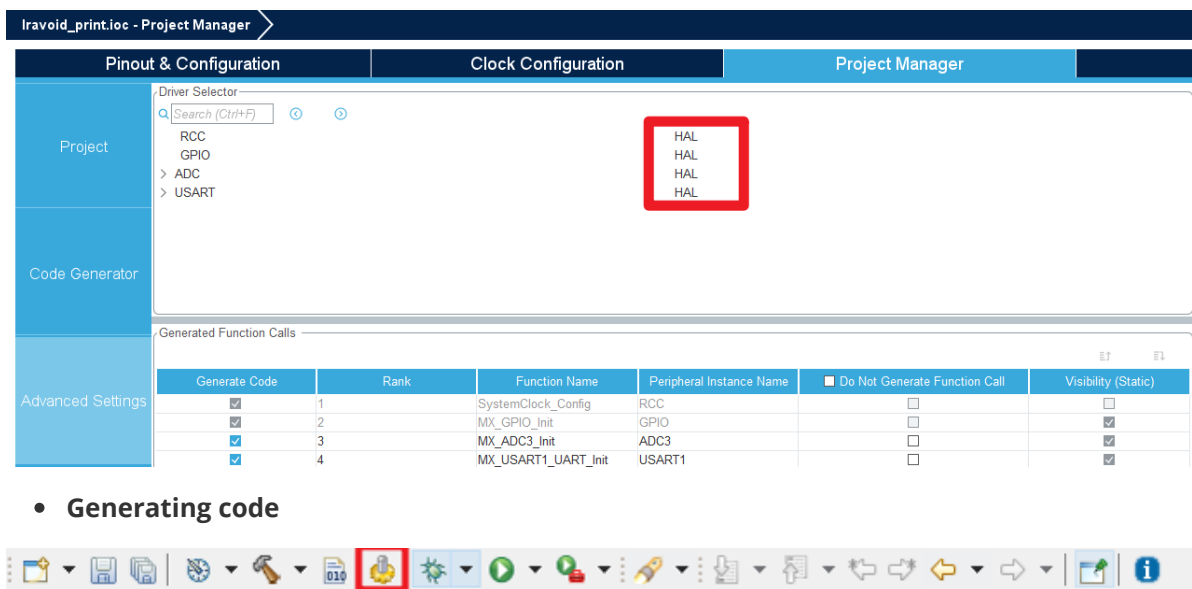
- Baud Rate: 115200 Bits/s
- Word Length: 8 Bits (including Parity)
- Parity: None
- Stop Bits: 1

The 'Advanced Parameters' section shows the following settings:

- Data Direction: Receive and Transmit
- Over Sampling: 16 Samples

- ADC





#### • Generating code

## 4、 Main Function

This paper mainly introduces the functional code written by users. \*\* Detailed code can be opened by yourself in the project file we provide, and enter the Bsp folder to view the source code. \*\*.

### User function

Many of the common HAL library functions were covered in Chapter 3, but they will not be covered here.

#### 函数: Adc\_Get\_Iravoid

Function prototypes	uint16_t Adc_Get_Iravoid(uint32_t ch)
Functional Description	The converted values of the infrared sensor ADC are collected
Input parameters	<b>ch</b> : channel
Return value	ADC converted value

#### 函数: Get\_Iravoid\_Data

Function prototypes	void Get_Iravoid_Data(uint16_t *left_data,uint16_t *right_data)
Functional Description	Print the converted values of the two infrared sensor ADCs
Input parameters1	<b>left_data</b> : Infrared sensor values on the left
Input parameters2	<b>right_data</b> : Infrared sensor values on the right
Return value	None

## 5、 Experimental phenomenon

After downloading the program successfully, press the RESET button of the development board to observe the phenomenon of serial debugging assistant

Program download can refer to [2, development environment construction and use: program download and simulation]

### phenomenon:

The serial port continuously prints the converted values of the two infrared sensor ADCs with a time interval of 0.3 seconds.

