# Buzzer control



This tutorial demonstrates: controlling the onboard active buzzer on the development board through the output function of **GPIO**.

## 1. Software-Hardware

- **STM32F103CubeIDE**
- **STM32 Robot Development Board**

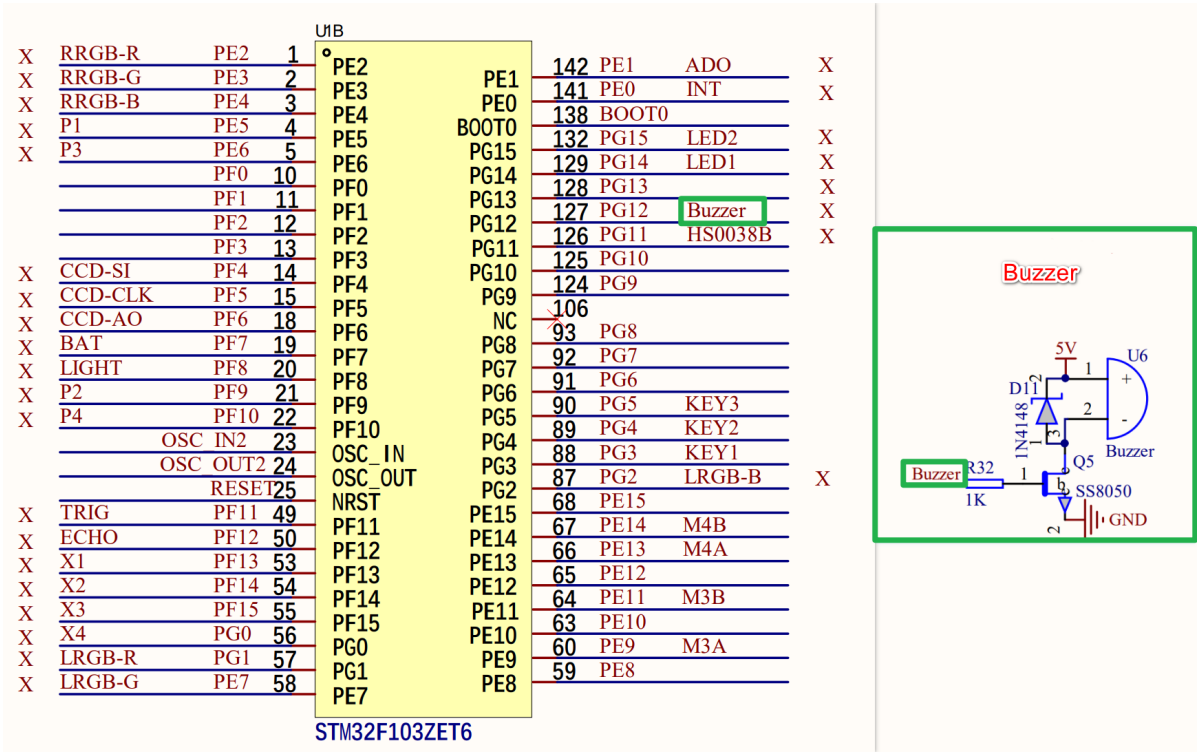
  GPIO: chip internal peripherals

  Active buzzer: onboard
- **Type-C data cable or ST-Link**

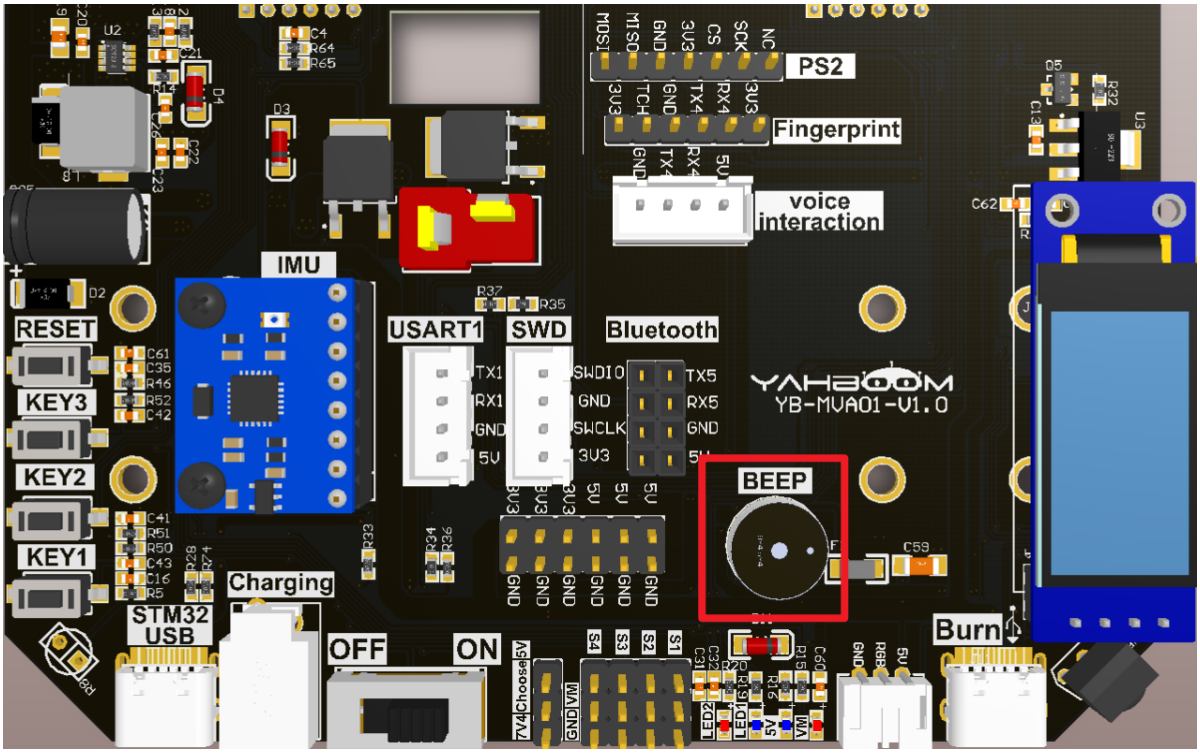
  Download programs or simulate the development board

## 2. Brief principle

### 1. Hardware schematic diagram




## 2. Physical connection diagram




## 3. Control principle

Control the buzzer sound by controlling the high and low levels of the buzzer pin.

**Buzzer: High level sounds, low level does not sound**

| Buzzer (schematic name) | control pin | Function |
| --- | --- | --- |
| Buzzer | PG12 | Control the buzzer sound |

- **buzzer**

| buzzer | Features |
| --- | --- |
| Active buzzer | It has its own internal oscillation source, and the sound can be controlled by setting high or low levels, and the sound frequency is fixed. |
| Passive buzzer | There is no internal oscillation source, and different oscillation pulse frequencies are provided to emit non-frequency sounds. |

# 3. Project configuration

**Project configuration: Prompt configuration options during STM32CubeIDE project configuration**

## 1. Description

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

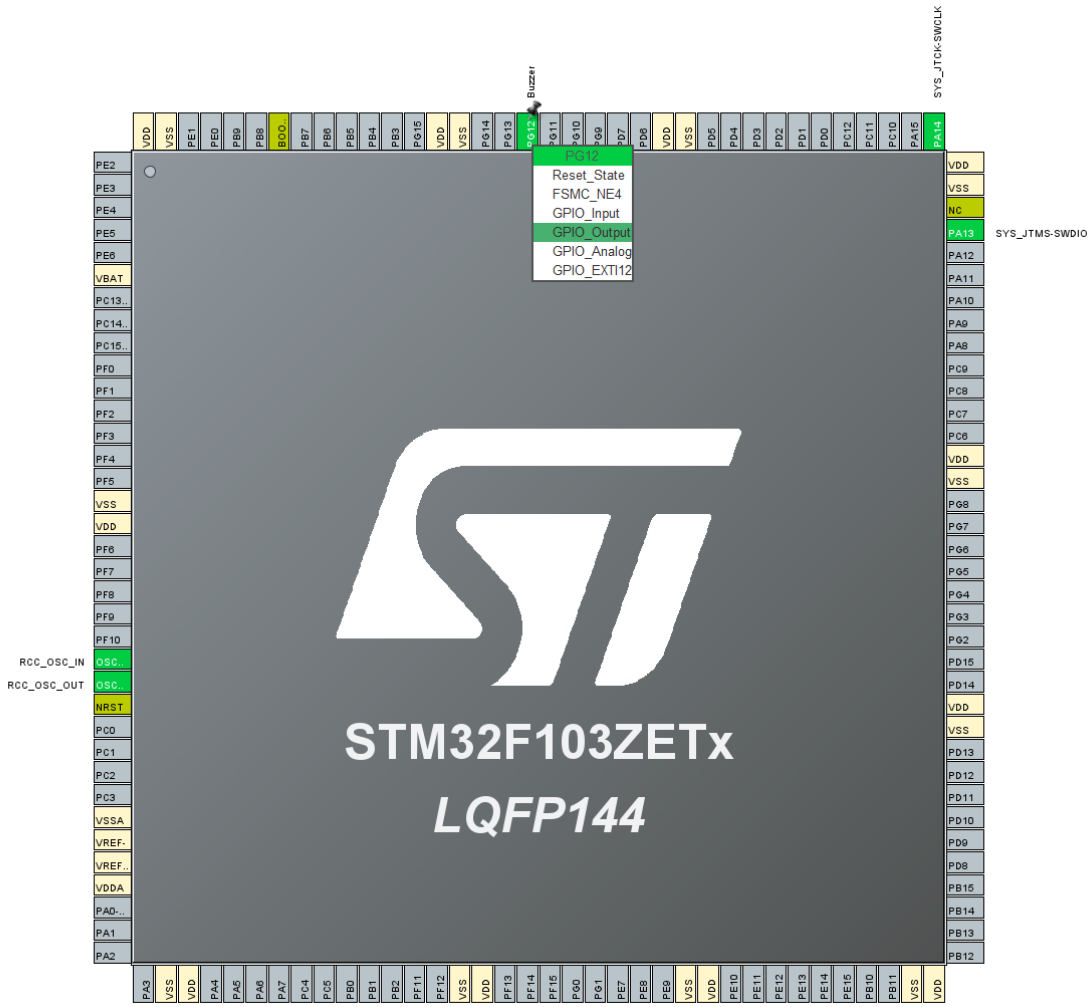
The project configuration part that is not omitted is the key point that needs to be configured in this tutorial.

```
Please refer to [2. Development environment construction and use: STM32CubeIDE installation and use] to understand how to configure the omitted parts of the project.
```

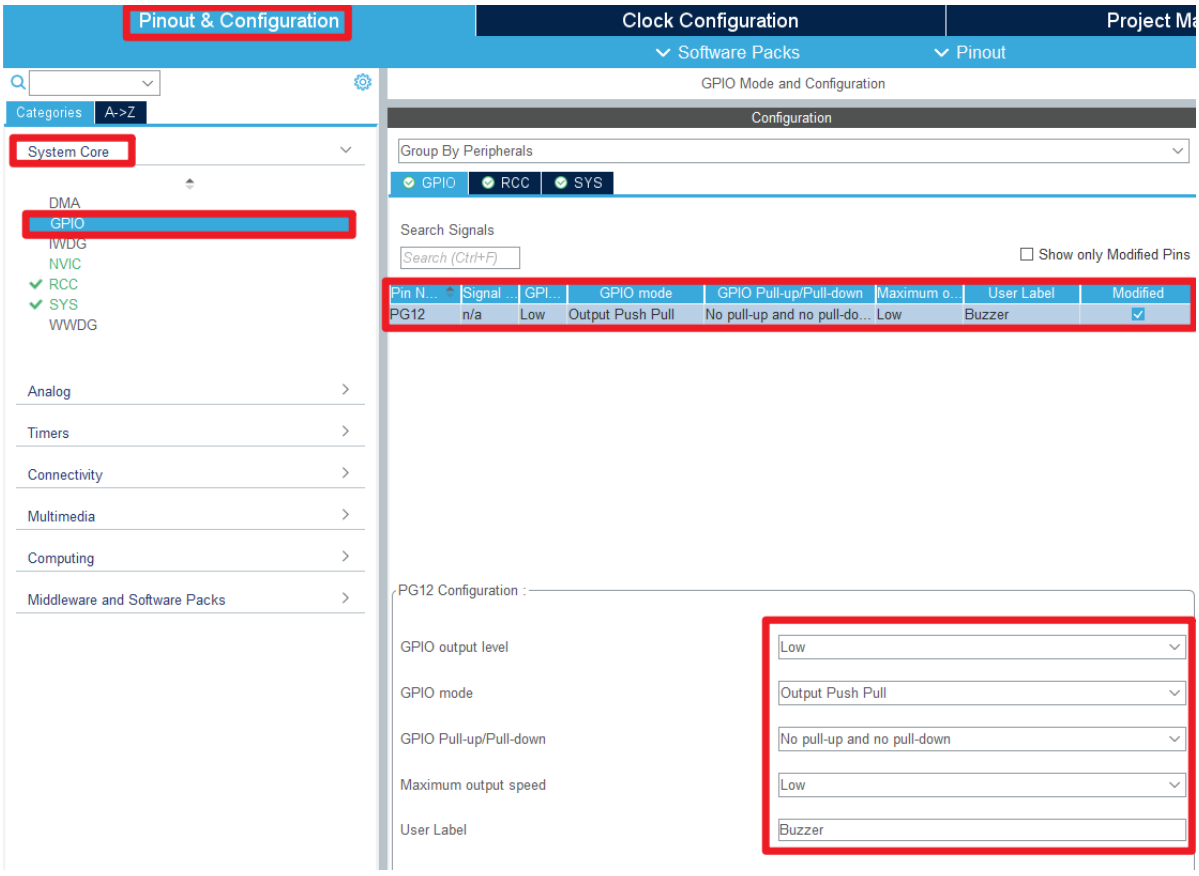
## 2. Pin configuration

- **Configure specified pin function**

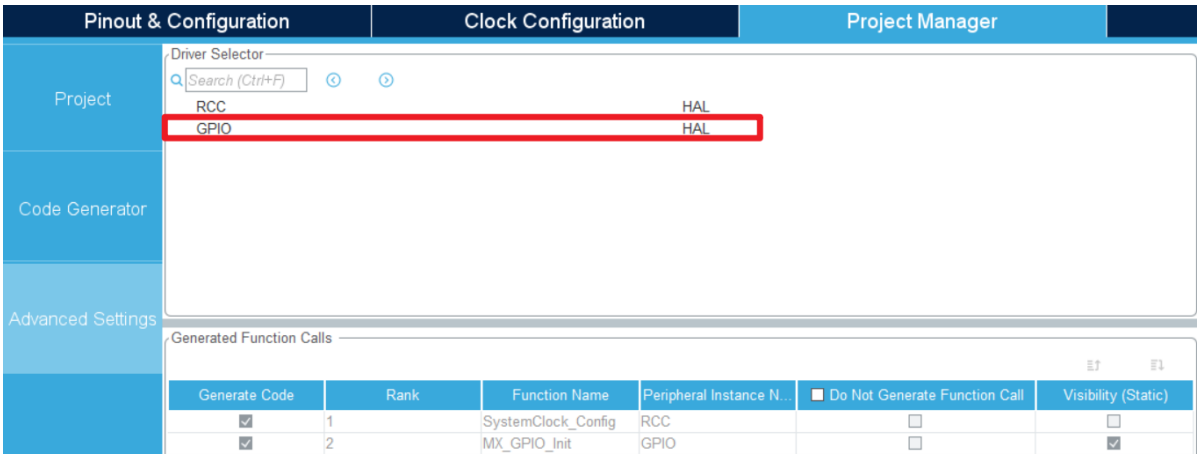
You can directly select the corresponding pin number in the pin view, and the corresponding options will appear when you left-click the mouse.



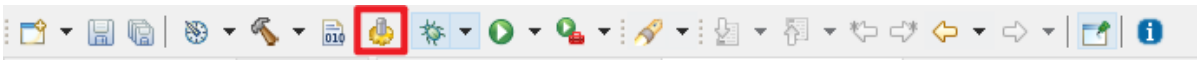

- **GPIO**




- **Advanced Settings**




- **Generate code**



# 4. Main functions

It mainly introduces the functional code written by the user. For detailed code, you can open the project file provided by us yourself and enter the Bsp folder to view the source code.

## User function

The HAL library functions that have been introduced in the previous tutorial will not be introduced again in the tutorial!

```
If you want to find the HAL library and LL library function analysis involved in the entire tutorial, you can view the documents in the folder [8. STM32 Manual: STM32F1_HAL Library and LL Library_User Manual]
```

> **function： Set_Buzzer**

| **function prototype** | **void Set_Buzzer(uint8_t i)** |
| --- | --- |
| Function description | control buzzer |
| Input parameters | **i**: switch (0: off, 1: on) |
| return value | none |

# 5. Experimental phenomena

After downloading the program successfully, press the RESET button of the development board and observe the development board phenomenon!

```
For program download, please refer to [2. Development environment construction and use: program download and simulation]
```

Phenomenon:

**Buzzer**: The time interval between sounding and not sounding is 0.2 seconds

For experimental phenomena, please see [Buzzer Control_Experimental Phenomenon.mp4]