Internal temperature sensor

Internal temperature 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、Tips
  - 3.2、Pin configuration
- 4、Main functions
  - HAL library function parsing
- 5、Experimental phenomenon

This tutorial demonstrates: getting real-time temperature from internal temperature sensor via **ADC (ADC1_IN16)** and printing battery voltage information via **serial port (UASRT1)** .

1、Software-Hardware

- **STM32F103CubeIDE**
- **STM32机 robot expansion board**

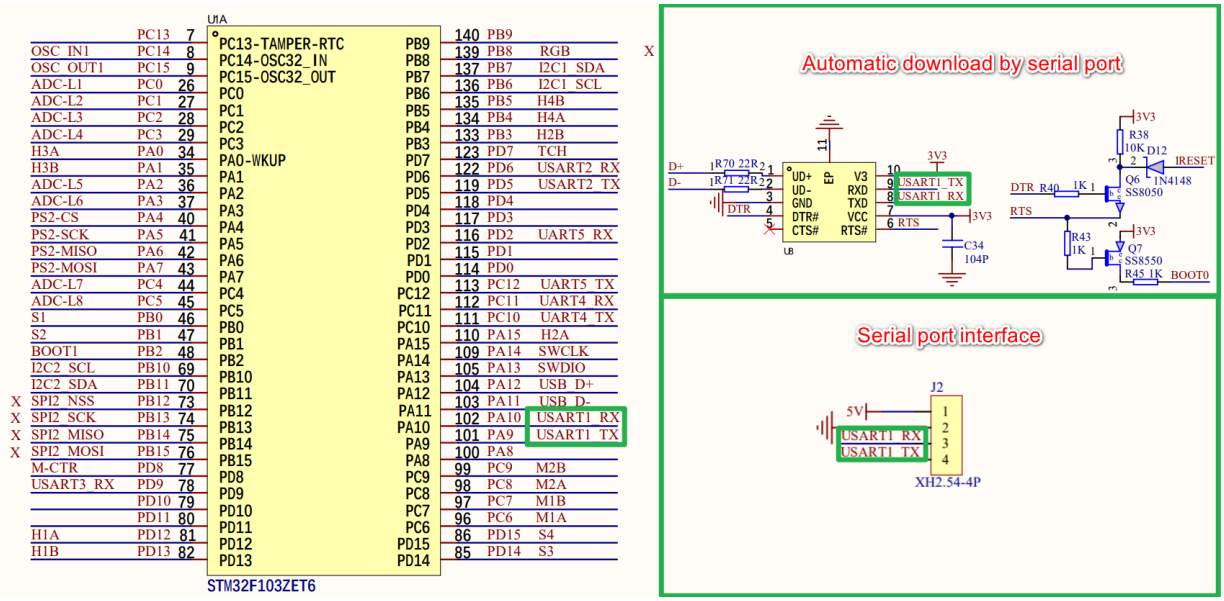
  ADC, USART, internal temperature sensor: chip internal peripherals
- **Type-C cable or ST-Link**

  Download or simulate the program of the development board

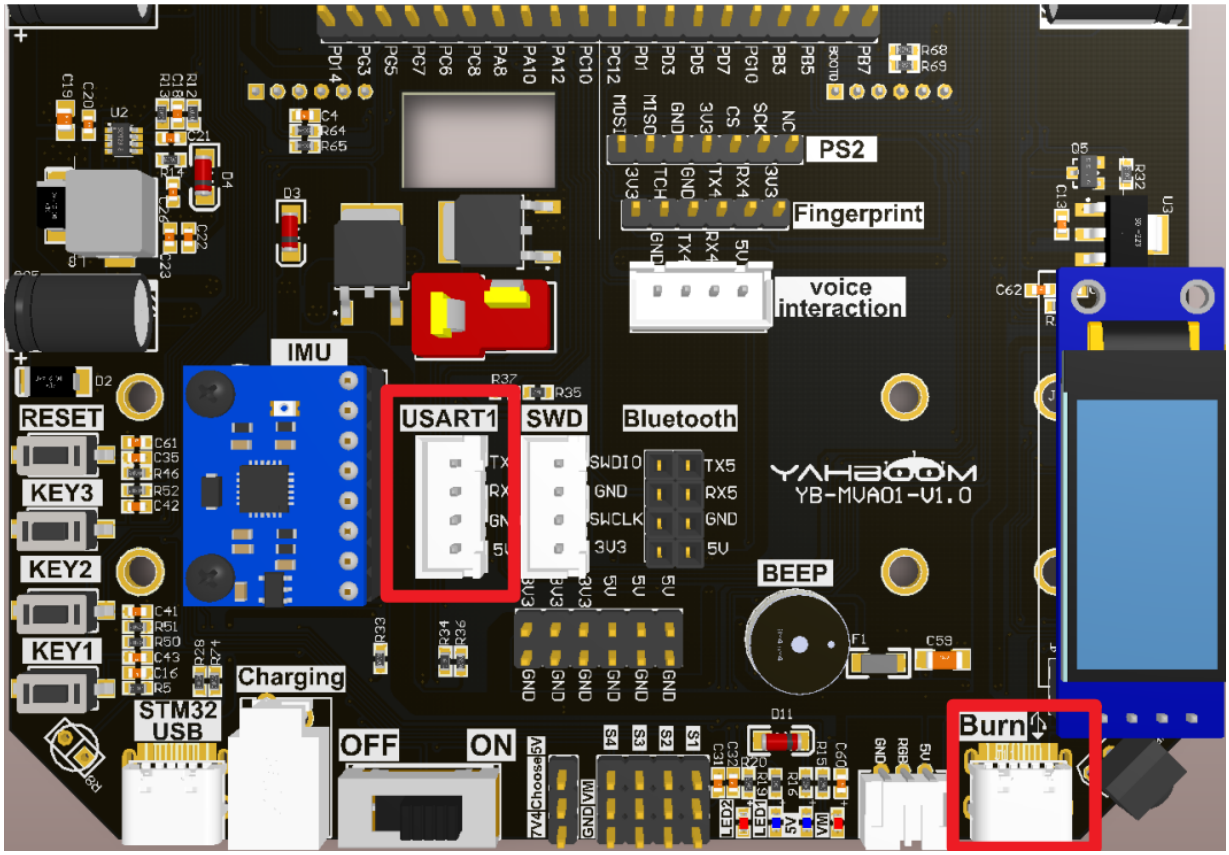
2、Brief principle

2.1、Hardware schematic diagram

由于ADC、USART、内部温度传感器都是芯片内部外设，这里只展示串口相关接口



2.2、Physical connection diagram



2.3、Principle of control

The converted value of internal temperature is obtained through a single-channel ADC, and the value is converted into the actual temperature information and printed out through the serial port.

- **Internal temperature sensor**

The STM32 has an internal temperature sensor that can be used to measure the temperature of the CPU and surrounding devices.

- **ADC conversion**

```
Adc-related knowledge can be referred to [3. Development Board Basic Tutorial: Voltage Detection]
```

The internal temperature sensor is internally connected to the ADC1_IN16 input channel, which converts the voltage output of the sensor into a digital value.

$$V = \frac{Value_{The\ number\ converted\ by\ the\ ADC} * （3.3）}{4096}$$

Actual temperature:

$V_{25} = V_{SENSE}$: $At\ 25\ degrees\ Celsius$； Avg_Slope : $Average\ slope\ of\ the\ temperature\ versus\ V_{SENSE}\ curve$。

（Typical values are used here, and the calculated temperature is not exact, but it is possible to observe the temperature change）

$$T_{℃} = \frac{V_{25} - V_{SENSE}}{Avg\_Slope} + 25.0 = \frac{1.43 - V}{0.0043} + 25.0$$

3、Engineering configuration

Project Configuration: Prompt for configuration options during STM32CubeIDE project configuration

3.1、Tips

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

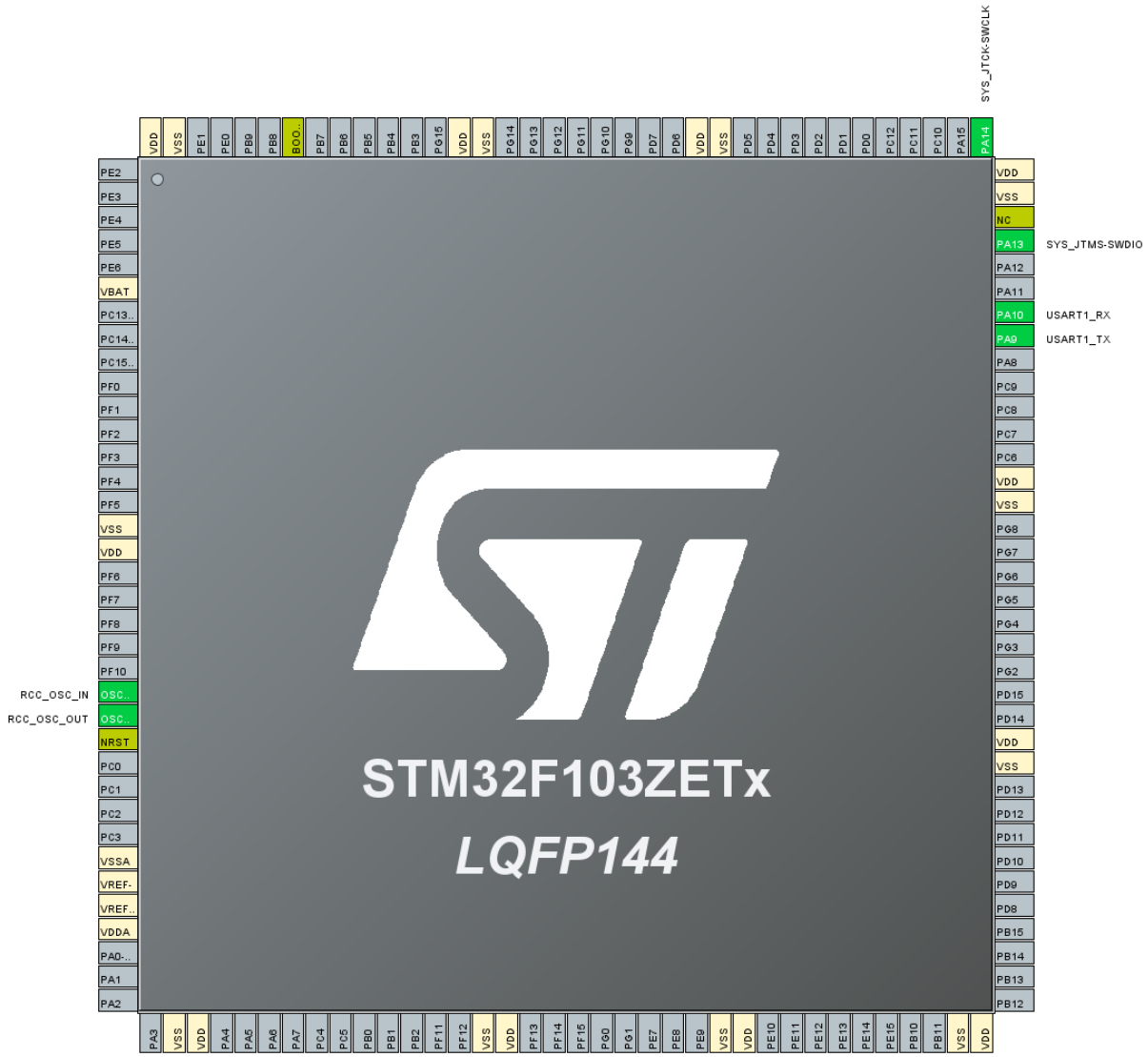
The project configuration part, which is not omitted, is the key point to configure in this tutorial.

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

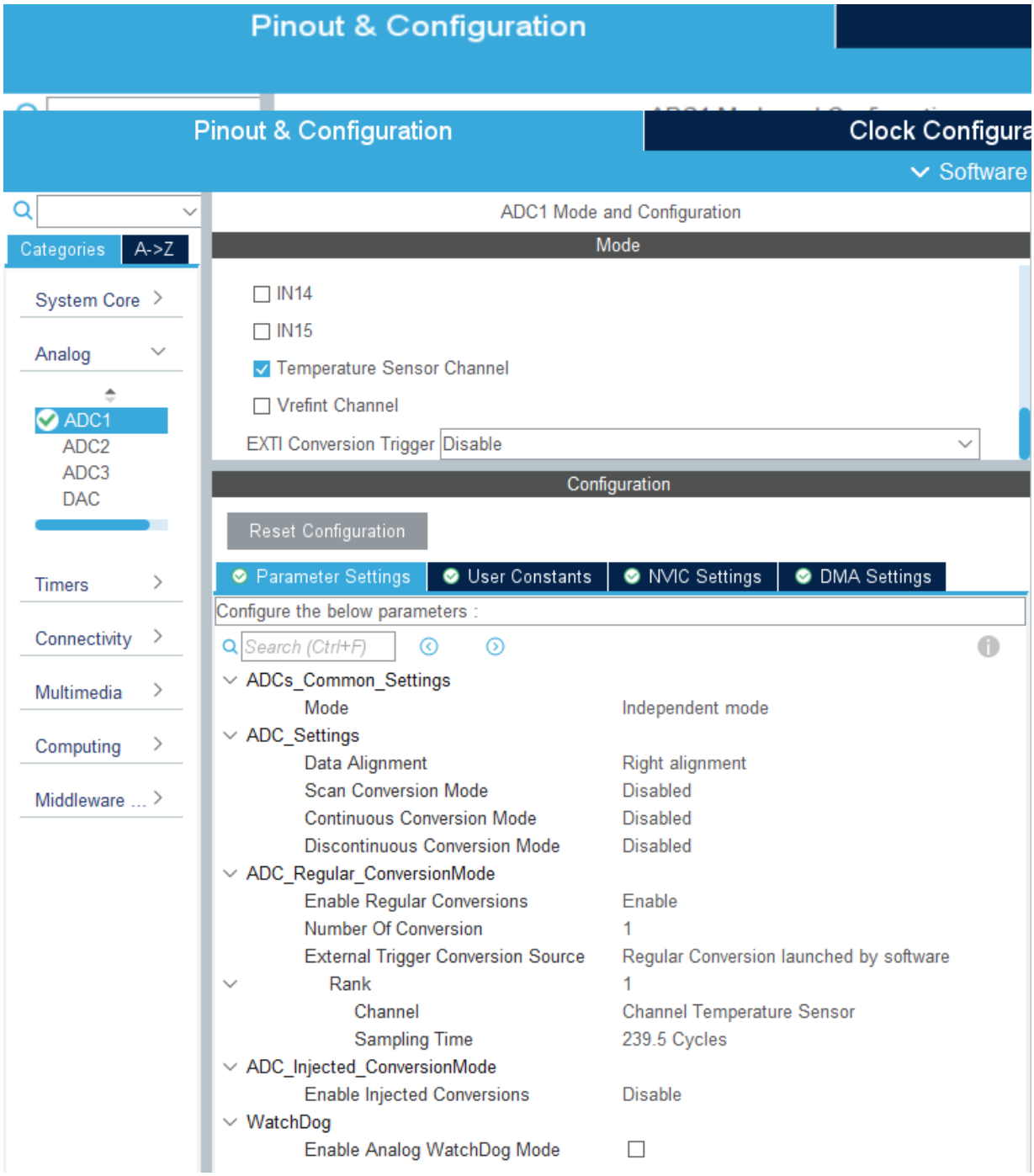
3.2、Pin configuration

- **Configure the specified pin function**

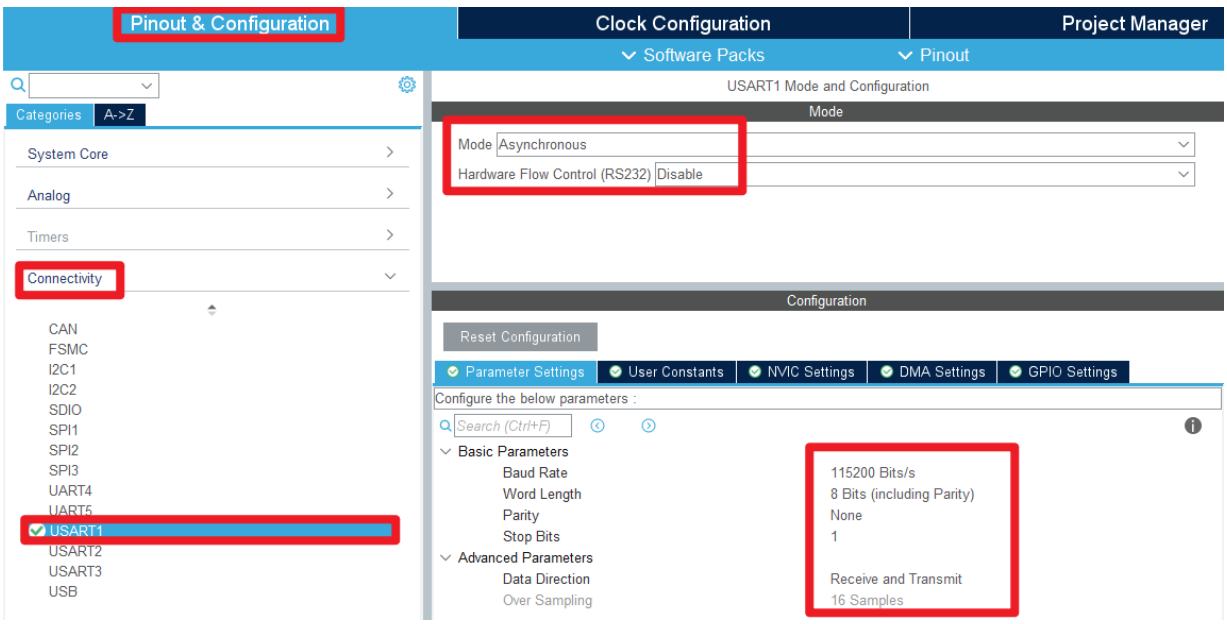
You can directly select the corresponding pin number in the pin view, and the corresponding option will appear when the mouse is left clicked

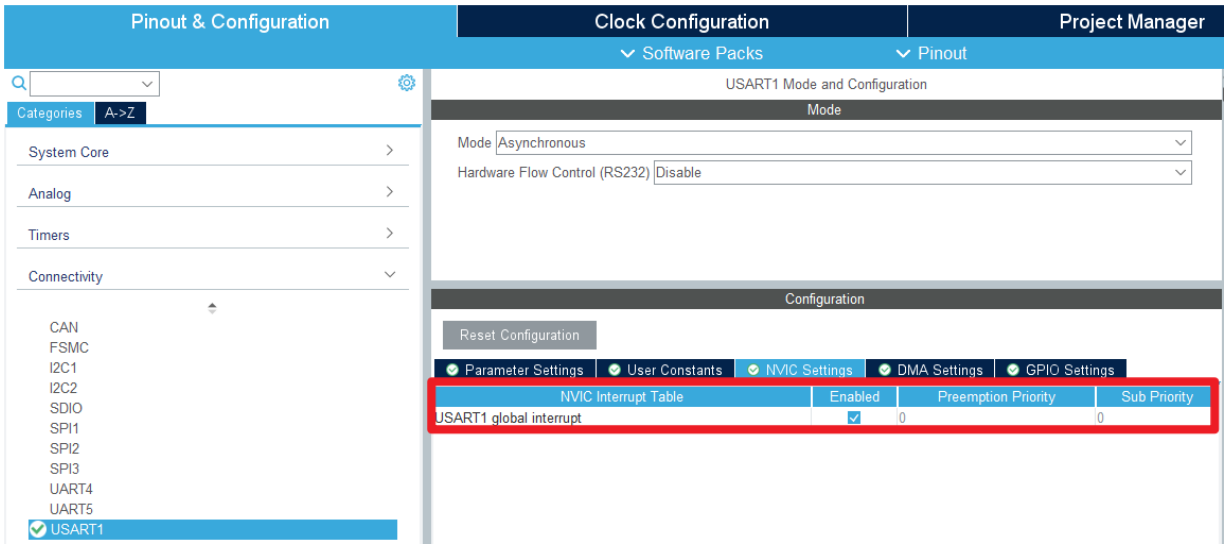


- **ADC**

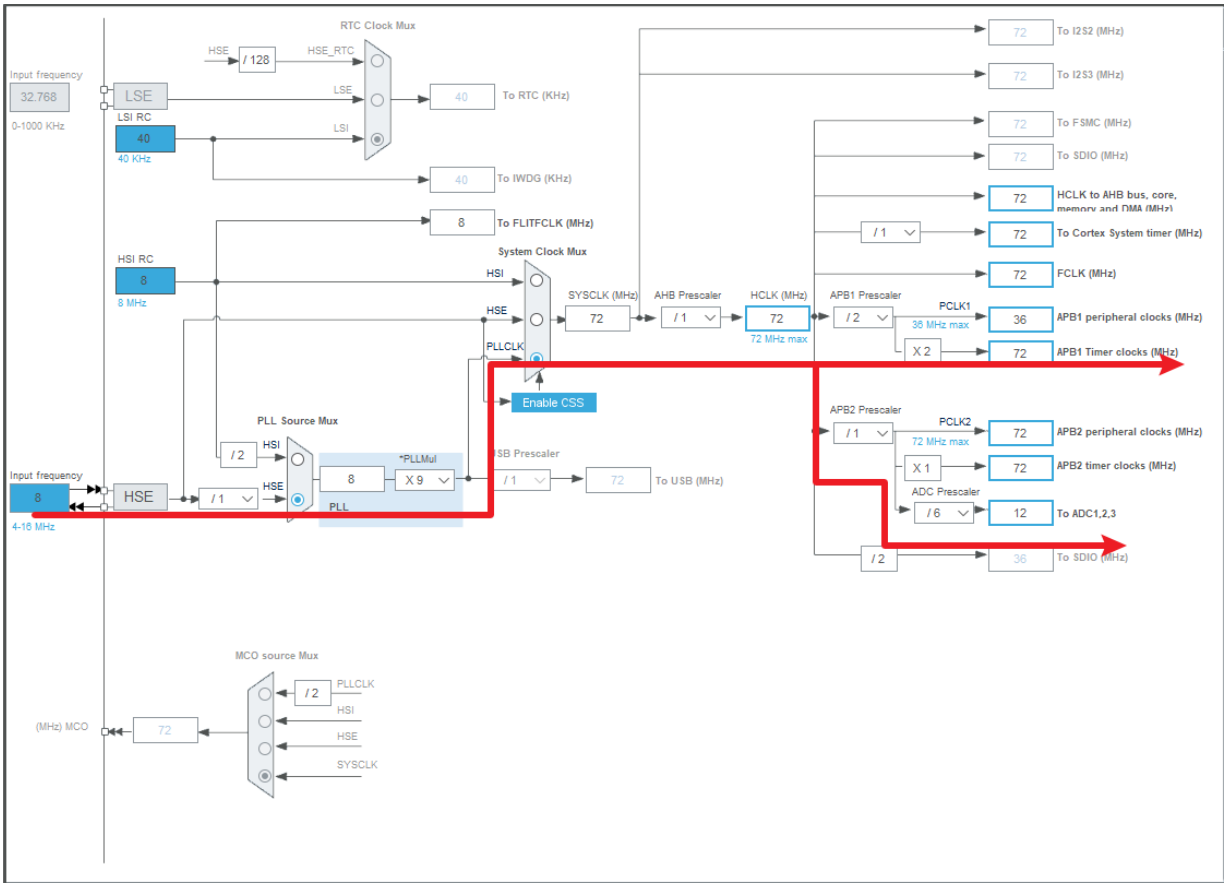

Pinout & Configuration
Pinout & Configuration
Clock Configura
Software
ADC1 Mode and Configuration
Categories
A->Z
Mode
System Core
Analog
IN14
IN15
Temperature Sensor Channel
Vrefint Channel
ADC1
ADC2
ADC3
DAC
EXTI Conversion Trigger
Disable
Configuration
Reset Configuration
Timers
Parameter Settings
User Constants
NVIC Settings
DMA Settings
Connectivity
Configure the below parameters :
Search (Ctrl+F)
Multimedia
ADCs_Common_Settings
Mode
Independent mode
Computing
ADC_Settings
Data Alignment
Right alignment
Middleware ...
Scan Conversion Mode
Disabled
Continuous Conversion Mode
Disabled
Discontinuous Conversion Mode
Disabled
ADC_Regular_ConversionMode
Enable Regular Conversions
Enable
Number Of Conversion
1
External Trigger Conversion Source
Regular Conversion launched by software
Rank
1
Channel
Channel Temperature Sensor
Sampling Time
239.5 Cycles
ADC_Injected_ConversionMode
Enable Injected Conversions
Disable
WatchDog
Enable Analog WatchDog Mode

- **USART**

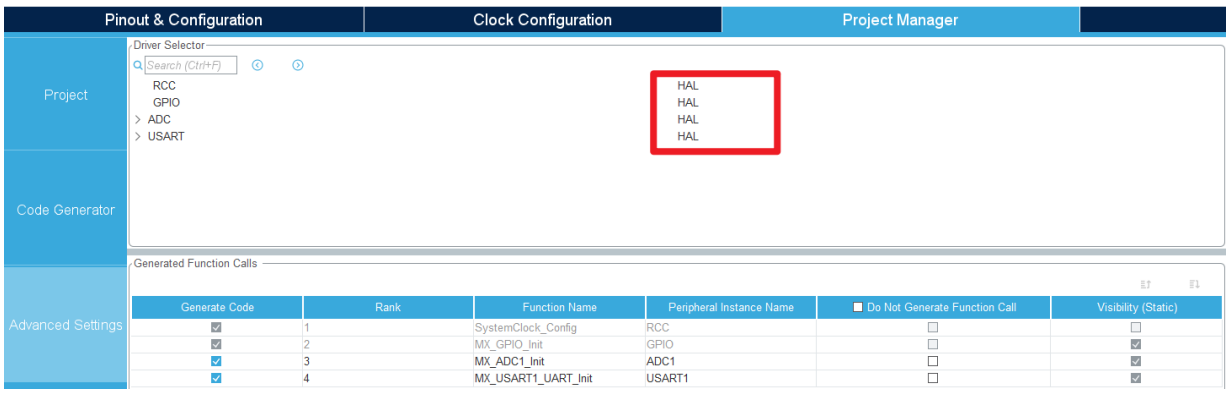

Pinout & Configuration
Clock Configuration
Project Manager
Software Packs
Pinout
USART1 Mode and Configuration
Categories
A->Z
Mode
System Core
Mode
Asynchronous
Analog
Hardware Flow Control (RS232)
Disable
Timers
Connectivity
Configuration
CAN
FSMC
I2C1
I2C2
SDIO
SPI1
SPI2
SPI3
UART4
UART5
USART1
USART2
USART3
USB
Reset Configuration
Parameter Settings
User Constants
NVIC Settings
DMA Settings
GPIO Settings
Configure the below parameters :
Search (Ctrl+F)
Basic Parameters
Baud Rate
115200 Bits/s
Word Length
8 Bits (including Parity)
Parity
None
Stop Bits
1
Advanced Parameters
Data Direction
Receive and Transmit
Over Sampling
16 Samples



- **Clock Configuration**



- **Advanced Settings**



- **Generating code**

4、Main functions

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.** .

HAL library function parsing

The HAL library functions that were covered in the previous tutorial will not be covered

```
For information on how to parse the HAL and LL libraries that are covered throughout this tutorial, check out the documentation under the STM32 Manual: STM32F1_HAL and LL Library_ User Manual
```

function：HAL_ADCEx_Calibration_Start

Function prototypes	**HAL_StatusTypeDef HAL_ADCEx_Calibration_Start(ADC_HandleTypeDef* hadc)**
Functional Description	Start the calibration process of the ADC
Input parameters	**hadc**：ADC handle address
Return value	**HAL status value**：HAL_OK、HAL_ERROR、HAL_BUSY、HAL_TIMEOUT

5、Experimental phenomenon

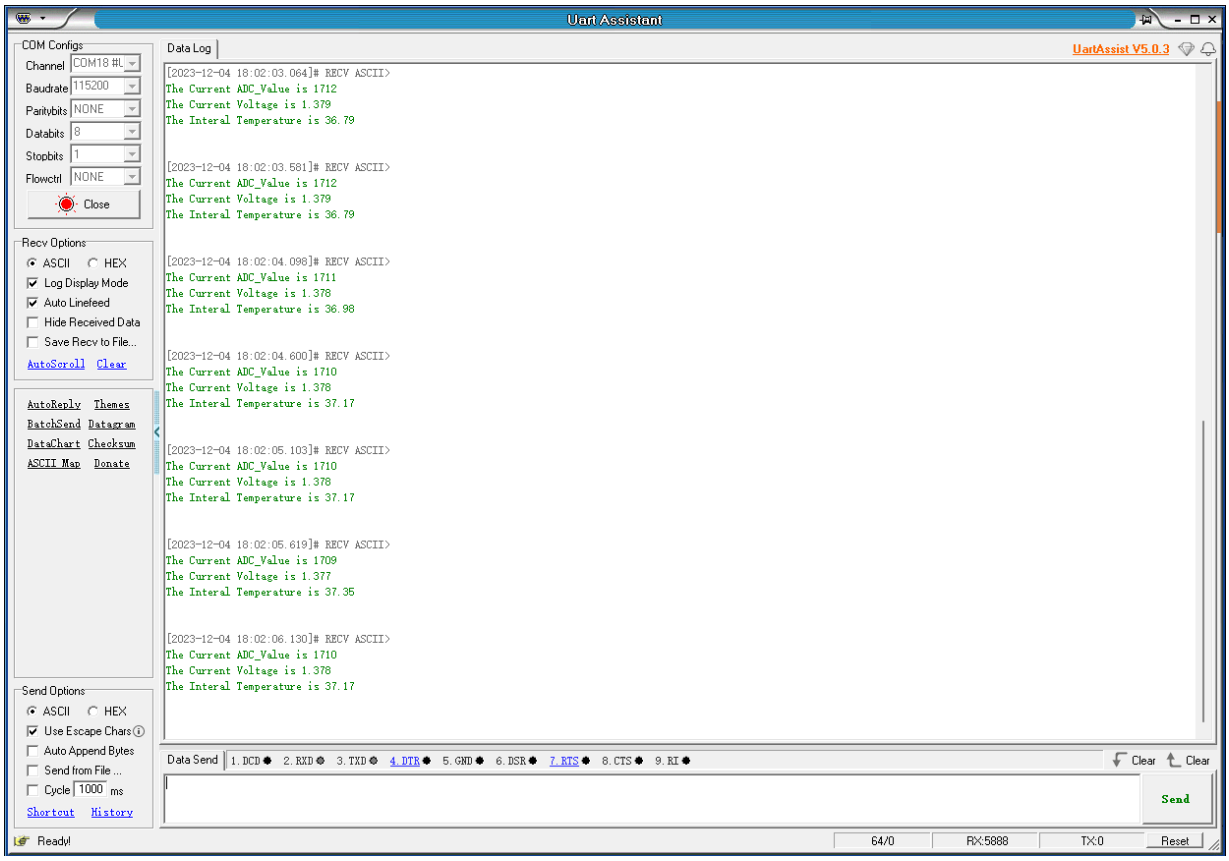
After downloading the program successfully, press the RESET button of the development board to open the serial debugging assistant to observe the phenomenon

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

phenomenon：

The serial debugging assistant will print the following information:

1. The value converted by the current ADC;
2. The voltage converted according to the ADC value;
3. Real-time temperature information from internal temperature sensor.


Uart Assistant
COM Configs
Channel COM18 #L
Baudrate 115200
Paritybits NONE
Databits 8
Stopbits 1
Flowctrl NONE
Close
Recv Options
ASCII HEX
Log Display Mode
Auto Linefeed
Hide Received Data
Save Recv to File...
AutoScroll Clear
AutoReply Themes
BatchSend Datagram
DataChart Checksum
ASCII Map Donate
Send Options
ASCII HEX
Use Escape Chars
Auto Append Bytes
Send from File ...
Cycle 1000 ms
Shortcut History
Data Log
UartAssist V5.0.3
[2023-12-04 18:02:03.064]# RECV ASCII>
The Current ADC_Value is 1712
The Current Voltage is 1.379
The Interal Temperature is 36.79
[2023-12-04 18:02:03.581]# RECV ASCII>
The Current ADC_Value is 1712
The Current Voltage is 1.379
The Interal Temperature is 36.79
[2023-12-04 18:02:04.098]# RECV ASCII>
The Current ADC_Value is 1711
The Current Voltage is 1.378
The Interal Temperature is 36.98
[2023-12-04 18:02:04.600]# RECV ASCII>
The Current ADC_Value is 1710
The Current Voltage is 1.378
The Interal Temperature is 37.17
[2023-12-04 18:02:05.103]# RECV ASCII>
The Current ADC_Value is 1710
The Current Voltage is 1.378
The Interal Temperature is 37.17
[2023-12-04 18:02:05.619]# RECV ASCII>
The Current ADC_Value is 1709
The Current Voltage is 1.377
The Interal Temperature is 37.35
[2023-12-04 18:02:06.130]# RECV ASCII>
The Current ADC_Value is 1710
The Current Voltage is 1.378
The Interal Temperature is 37.17
Data Send
1.DCD 2.RXD 3.TXD 4.DTR 5.GND 6.DSR 7.RTS 8.CTS 9.RI
Clear Clear
Send
Ready!
64/0 RX:5888 TX:0 Reset