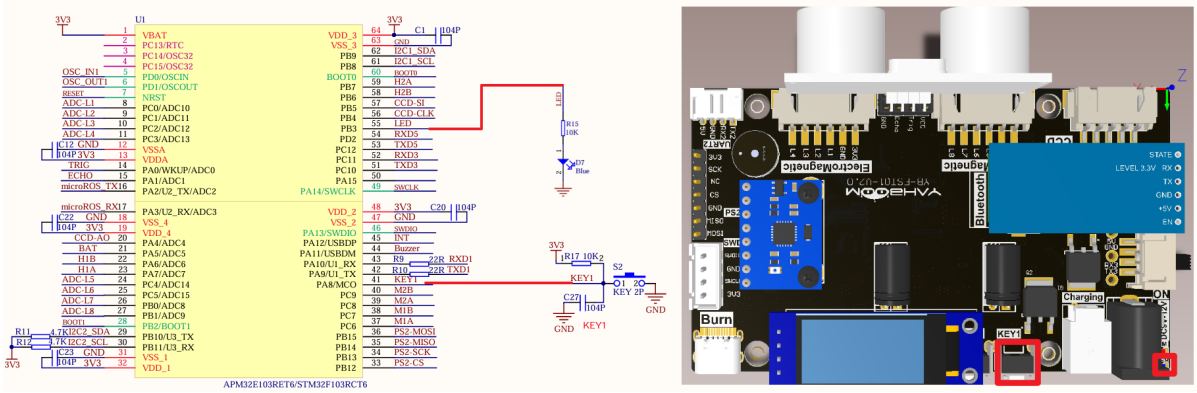
Independent Watchdog (IWDG)

The tutorial combines LED and KEY to demonstrate the timeout reset function of the independent watchdog (IWDG).

Hardware connection



Peripherals	Development board	Description
LED	PB3	The anode of the LED is connected to the development board PB3, and the cathode is connected to the development board GND
KEY1	PA8	One end of the KEY is connected to the PA8 pin, and the other end is connected to GND

Control principle

By using a button to reset the count value of the independent watchdog (IWDG), the dog feeding function is realized and the LED is lit.

- **Watchdog**

The STM32F103ZET6 has two built-in watchdogs (independent watchdog and window watchdog), which are mainly used for system fault detection and recovery.

Watchdog	Function
Independent Watchdog	Used to detect whether the system is running normally
Window Watchdog	Used to detect system failure

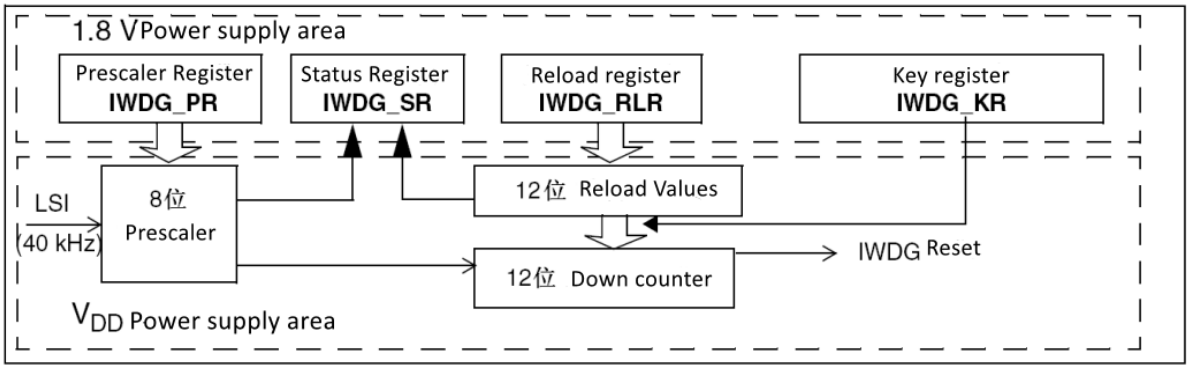
- **Independent Watchdog**

The independent watchdog (IWDG) is driven by a dedicated low-speed clock (LSI), and it is still effective even if the main clock fails;

When the system does not feed the dog (reset the count value) within a certain period of time, the independent watchdog will trigger a reset operation to restart the system.

```
The main function of the independent watchdog is to prevent the system from
entering a dead loop or infinite delay during operation, and to avoid the system
from being unresponsive or freezing for a long time.
```

- **Independent watchdog features**
- Free-running down counter
- After the watchdog is activated, a reset is generated when the counter counts to 0x000
- The clock is provided by an independent RC oscillator (can work in stop and standby modes)



- **Watchdog timeout**: 40kHz input clock (LSI)

Prescaler	PR[2:0]	Minimum time (ms): RL[11:0]=0x000	Maximum time (ms): RL[11:0]=0xFFF
/4	0	0.1	409.6
/8	1	0.2	819.2
/16	2	0.4	1638.4
/34	3	0.8	3276.8
/64	4	1.6	6553.6
/128	5	3.2	13107.2
/256	6 or 7	6.4	26214.4

```
The pre-scaling coefficient set in the tutorial is 256, and the reload value is
156
```

$$T_{Timeout(s)} = \frac{4 * 2^{PR} * Reload}{LSI} = \frac{256 * 156}{40000} \approx 1s$$

Software Configuration

Software Code

Configure the timeout reset function of the independent watchdog (IWDG), without configuring specific pins.

```
Product supporting materials source code path: Attachment → Source Code Summary →
1.Base_Course → 12.IWDG
```

Control Function

The tutorial only briefly introduces the code, and you can open the project source code for details.

> **IWDG_Start**

```
void IWDG_Start(void)
{
    IWDG_WriteAccessCmd(IWDG_WriteAccess_Enable);
    IWDG_SetPrescaler(IWDG_Prescaler_256);
    IWDG_SetReload(155); //1s ≈ (1/40000)*256*(155+1) Watchdog feeding time is
set within 1s
    IWDG_Enable();
    IWDG_ReloadCounter();
}
```

Experimental phenomenon

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

After the program is downloaded successfully: press KEY1 to feed the dog, the LED lights up; if you do not press KEY, the LED goes out, and the serial port keeps printing start!

```
When using the serial port debugging assistant, you need to pay attention to the
serial port settings. If the settings are wrong, the phenomenon may be
inconsistent
```