LED blinking routine

Introduction

This example is the first example of the SDK and also the simplest one, similar to the first program a programmer encounters, Hello World.

Its main function is to make the red LED in the onboard RGB-LED flash continuously.

Hardware Description

![LED circuit schematic](../../docs/figures/01_basic_led_blink/led_sch1.png)

![LED circuit schematic](../../docs/figures/01_basic_led_blink/led_sch2.png)

As shown in the figure above, RBG-LED is a common anode LED, and the cathode is connected to pins 30, 31, and 32 of the microcontroller, where the red LED corresponds to pin 30. The microcontroller pin outputs a low level to light up the LED, and a high level output will turn off the LED.

The position of the LED on the development board is shown in the following figure:

![LED position](../../docs/figures/01_basic_led_blink/obj.png)

Software Description

The source code of the flash light is located in `/examples/01_basic_led_blink/applications/main.c`. First, a macro `LED_PIN` is defined, which corresponds to the red pin `30` of the LED.

```c
/*
using RED LED in RGB */ #define LED_PIN
(30)
```

In the main function, configure the pin as output mode, and in the following while loop, periodically (500 milliseconds) switch the LED and output some log information.

```c
int main(void) {

    unsigned int count = 1;

    /* Set LED pin to output mode */
    rt_pin_mode(LED_PIN, PIN_MODE_OUTPUT);

    while (count > 0) {

        /*LED light on*/
        rt_pin_write(LED_PIN, PIN_LOW);
        LOG_D("led on, count: %d", count); rt_thread_mdelay(500);


        /* LED light off*/
        rt_pin_write(LED_PIN, PIN_HIGH);
        LOG_D("led off");
```

```
            rt_thread_mdelay(500);

            count++;
        }

        return 0;
}``
```

Run

Compile & Download

 - **MDK**: Double-click `project.uvprojx` to open the MDK5 project and execute compilation.

After compiling, download the firmware to the development board.

running result

Press the reset button to restart the development board and observe the actual effect of the RBG-LED on the development board. After normal operation, the red LED will flash periodically, as shown in the following figure:

 ![RGB red light turns on](../../docs/figures/01_basic_led_blink/led_on.png)

At this time, you can also use the terminal tool on the PC to open the `uart0` serial port of the development board and set `115200-8-1-N`. The running log information of the development board can be output in real time.

```shell
[D/main] led on, count: 1
[D/main] led off
[D/main] led on, count: 2
[D/main] led off
[D/main] led on, count: 3
[D/main] led off
[D/main] led on, count: 4
[D/main] led off
[D/main] led on, count: 5
[D/main] led off
[D/main] led on, count: 6
[D/main] led off
[D/main] led on, count: 7
[D/main] led off
[D/main] led on, count: 8
[D/main] led off
[D/main] led on, count: 9
[D/main] led off
[D/main] led on, count: 10
```

Precautions

If you want to modify the LED_PIN macro definition, you can refer to the /drivers/pin_map.c file, which defines other pin numbers of the microcontroller.

References

- "General GPIO Device Application Notes": docs/AN0002-RT-Thread-General GPIO Device Application Notes.pdf
- "RT-Thread Programming Guide": docs/RT-Thread Programming Guide.pdf