



HCRTOS SDK flash的通用otp读写文档

!!! flash的otp区域有永久写入不可更改的特性，请在批量操作的之前，确保otp写入区域和写入数据正确，再开启otp锁定功能函数!!!

1. 文档履历

2. 文档履历

版本号	日期	制/修订人	制/修订记录
1.0	2023.08.02	邱浩佳	新增flash的otp读写指南

HCRTOS SDK flash的通用otp读写文档

!!! flash的otp区域有永久写入不可更改的特性，请在批量操作的之前，确保otp写入区域和写入数据正确，再开启otp锁定功能函数!!!

1. 文档履历
2. 文档履历
3. 概述
 - 3.1 编写目的
 - 3.2 读者对象
4. 模块介绍
5. 模块测试用例与Sample Code
 - 5.1 模块功能的开启
 - 5.2 example的使用
 - 5.3 flash otp example相关代码
6. 如何添加自己的flash型号读写otp
7. otp操作命令与通用不一致

3. 概述

3.1 编写目的

介绍和指导hcrtos的flash读写otp的操作。该代码支持通用otp命令，针对特殊读写otp操作命令的flash，需要根据flash文档进行代码更改。

3.2 读者对象

hcrtos sdk的开发工程师。

4. 模块介绍

- 本模块包括flash的otp的读写接口使用介绍

5. 模块测试用例与Sample Code

5.1 模块功能的开启

> Components > Cmds

```
There is no help available for this option.
Prompt: Test spi controller
Location:
  -> Components
    -> Cmds (BR2_PACKAGE_CMD5 [=y])
      -> spi operations (CONFIG_CMD5_SPI [=y])
Defined at source:62
Depends on: BR2_PACKAGE_CMD5 [=y] && CONFIG_CMD5_SPI [=y]
Selected by [m]:
  - BR2_PACKAGE_CMD5 [=y] && CONFIG_CMD5_SPI [=y] && m
```



```

2  {

3      uint8_t register_status = 0;

4      uint8_t flash_otp_wdata[256] = { 0 };

5      uint8_t flash_otp_rdata[256] = { 0 };

6

7      /* otp data to write */

8      memset(flash_otp_wdata, 0xcc, sizeof(flash_otp_wdata));

9

10     /* erase enable */

11     hc_norflash_send_one_cmd(0x06);

12     hc_norflash_read_register(0x05, &register_status);

13

14     /* erase */

15     hc_norflash_erase_OTP(flash_name);

16     /* wait erase finish */

17     hc_norflash_read_register(0x05, &register_status);

18     while ((register_status >> 0x0) & 0x01) {

19         hc_norflash_read_register(0x05, &register_status);

20     }

21

22     /* write enable */

23     hc_norflash_send_one_cmd(0x06);

24     hc_norflash_read_register(0x05, &register_status);

25

26     /* write */

27     hc_norflash_write_OTP(flash_name, flash_otp_wdata,
sizeof(flash_otp_wdata));
28     /* wait write finish */

29     hc_norflash_read_register(0x05, &register_status);

30     while ((register_status >> 0x0) & 0x01) {

```

```

31         hc_norflash_read_register(0x05, &register_status);

32     }

33
34     /* read otp data */

35     nor_flash_read_OTP(flash_name, flash_otp_rdata,
sizeof(flash_otp_rdata));
36
37     /* func: nor_flash_lock_OTP

38     * warning !!! if this function run, the otp area can't be write
forever
39     */

40     //hc_norflash_lock_OTP(flash_name);

41
42     return 0;

43 }

```

注意!!!：hc_norflash_lock_OTP函数默认被注释，需要将otp区域锁定才取消注释，一旦otp区域被锁定，再也无法写入，只能读取，执行这个锁定操作时，确保写入区域和写入数据的正确！！

6. 如何添加自己的flash型号读写otp

代码位置位于：components/cmds/source/spi/spidev_test.c

```

struct flash_type {
    char name[128];
    uint32_t id;
    int8_t read_id_cmd;          int8_t read_id_dummy_num;          uint32_t id_len;
    int8_t read_unique_id_cmd;   int8_t read_unique_id_dummy_num;   uint32_t unique_id_len; int32_t uniqueid_read_start_addr;
    int8_t read_otp_cmd;         int8_t read_otp_dummy_num;         uint32_t otp_addr;
    int8_t enter_otp_mode_cmd;   int8_t exit_otp_mode_cmd;
    int8_t write_otp_cmd;
    int8_t erase_otp_cmd;
};

```

根据flash文档，将对应的结构体参数给补全，并添加到struct flash_type FLASH_TYPE[]结构体中。

例如：

```
1  {  
2      "BY25Q128",  
3      0x684018,  
4      0x9f, 0, 0x03,  
5      0x4b, 4, 16, NONE,  
6      0x48, 1, 0x3000,  
7      NONE, NONE,  
8      0x42,  
9      0x44,  
10 },
```

7. otp操作命令与通用不一致

需要根据flash文档描述的otp命令进行编写。