

# H1 读取芯片型号方法

通过读取efuse来获取具体的芯片型号, 所以如果芯片出货前没有经过efuse烧录, 是无法正确获取具体的芯片型号.

目前 `hcLinux` 或 `hcrtos` 均支持.

## H2 HCRTOS

### H3 使用命令

`menuconfig` 所需要打开的开关

- `CONFIG_CMDS_EFUSE_DUMP`

上电后输入以下命令 `efuse chip` 即可, 如下所示

```
1 hc1600a@dbD3100v10# efuse chip
2 chip: D3100
```

### H3 源码方式调用

主要是通过 `open("/dev/efuse", O_RDWR);`, 然后再利用 `ioctl` 命令 `EFUSE_GET_CHIPID` 来获取具体芯片型号.

示例如下所示

```
1 #include <fcntl.h>
2 #include <sys/ioctl.h>
3 #include <hcuapi/efuse.h>
4 #include <hcuapi/chipid.h>
5
6 int console_get_chip_id(int argc, char **argv)
7 {
8     int fd, ret;
```

```
9     enum HC_CHIPID chip;
10
11     fd = open("/dev/efuse", O_RDWR);
12     if (fd < 0) {
13         printf("[error] cannot open /dev/efuse, ret:%d\r\n", fd);
14         return -1;
15     }
16
17     ret = ioctl(fd, EFUSE_GET_CHIPID, (uint32_t)&chip);
18     if(ret < 0){
19         printf("[error] ioctl CMD EFUSE_GET_CHIP_ID, ret:%d\r\n", fd);
20         return -1;
21     }
22
23     switch(chip){
24         case HICHIP_A3000: printf("chip: A3000\n"); break;
25         case HICHIP_A3100: printf("chip: A3100\n"); break;
26         case HICHIP_A3200: printf("chip: A3200\n"); break;
27         case HICHIP_A3300: printf("chip: A3300\n"); break;
28         case HICHIP_A5000: printf("chip: A5000\n"); break;
29         case HICHIP_A5100: printf("chip: A5100\n"); break;
30         case HICHIP_A5200: printf("chip: A5200\n"); break;
31         case HICHIP_B3100: printf("chip: B3100\n"); break;
32         case HICHIP_B3200: printf("chip: B3200\n"); break;
33         case HICHIP_C3000: printf("chip: C3000\n"); break;
34         case HICHIP_C3100: printf("chip: C3100\n"); break;
35         case HICHIP_C5000: printf("chip: C5000\n"); break;
36         case HICHIP_C5200: printf("chip: C5200\n"); break;
37         case HICHIP_D3000: printf("chip: D3000\n"); break;
38         case HICHIP_D3100: printf("chip: D3100\n"); break;
39         case HICHIP_D3200: printf("chip: D3200\n"); break;
40         case HICHIP_D5000: printf("chip: D5000\n"); break;
41         case HICHIP_D5200: printf("chip: D5200\n"); break;
42         case HICHIP_E3000: printf("chip: E3000\n"); break;
43         case HICHIP_E3100: printf("chip: E3100\n"); break;
44         default : printf("chip: NOT KNOWN CHIP ID\n"); break;
45     }
46     close(fd);
47     return 0;
48 }
```

## H2 HCLINUX

### H3 源码方式调用

用户层通过加载 `hc_efuse.ko`, 然后访问 `open("/dev/efuse", O_RDWR);`, 利用 `ioctl` 命令 `EFUSE_GET_CHIPID` 来获取具体芯片型号.

```
1  #include <stdio.h>
2  #include <stdint.h>
3  #include <fcntl.h>
4  #include <unistd.h>
5  #include <sys/ioctl.h>
6  #include <hcuapi/efuse.h>
7  #include <hcuapi/chipid.h>
8
9  int main(int argc, char **argv)
10 {
11     int fd, ret;
12     enum HC_CHIPID chip;
13
14     system("modprobe hc_efuse"); // 通过modprobe 加载 hc_efuse 的驱动模块
15
16     usleep(5000);
17
18     fd = open("/dev/efuse", O_RDWR);
19     if (fd < 0) {
20         printf("[error] cannot open /dev/efuse, ret:%d\r\n", fd);
21         return -1;
22     }
23
24     ret = ioctl(fd, EFUSE_GET_CHIPID, (uint32_t)&chip);
25     if(ret < 0){
26         printf("[error] ioctl CMD EFUSE_GET_CHIP_ID, ret:%d\r\n", fd);
27         return -1;
28     }
29
30     switch(chip){
31         case HICHIP_A3000: printf("chip: A3000\n"); break;
32         case HICHIP_A3100: printf("chip: A3100\n"); break;
33         case HICHIP_A3200: printf("chip: A3200\n"); break;
34         case HICHIP_A3300: printf("chip: A3300\n"); break;
35         case HICHIP_A5000: printf("chip: A5000\n"); break;
36         case HICHIP_A5100: printf("chip: A5100\n"); break;
37         case HICHIP_A5200: printf("chip: A5200\n"); break;
38         case HICHIP_B3100: printf("chip: B3100\n"); break;
39         case HICHIP_B3200: printf("chip: B3200\n"); break;
40         case HICHIP_C3000: printf("chip: C3000\n"); break;
41         case HICHIP_C3100: printf("chip: C3100\n"); break;
```

```
42         case HICHIP_C5000: printf("chip: C5000\n"); break;
43         case HICHIP_C5200: printf("chip: C5200\n"); break;
44         case HICHIP_D3000: printf("chip: D3000\n"); break;
45         case HICHIP_D3100: printf("chip: D3100\n"); break;
46         case HICHIP_D3200: printf("chip: D3200\n"); break;
47         case HICHIP_D5000: printf("chip: D5000\n"); break;
48         case HICHIP_D5200: printf("chip: D5200\n"); break;
49         case HICHIP_E3000: printf("chip: E3000\n"); break;
50         case HICHIP_E3100: printf("chip: E3100\n"); break;
51         default : printf("chip: NOT KNOWN CHIP ID\n"); break;
52     }
53     close(fd);
54     return 0;
55 }
56
```