

# CherryUSB 主从协议栈移植

[芯片通用移植指南 — CherryUSB 1.1.0 文档](#)

## 前期准备

- 一个可以打印的工程（主机需要带os），USB 使用官方例程可以工作再来用，默认使用printf，确认是否printf正常
- 确定usb ip
- 确定 USB 引脚如何配置，USB 时钟如何配置，USB 寄存器基地址，USB 中断号，usb 中断函数入口名称，如何注册中断，USB 硬件资源
- IP 专用的一些宏的配置
- 是否带 cache

## 移植

- 带 cache的处理方式：必须指定 nocacheable 的 ram
- usb\_config宏的一些说明
- 单模式的移植
- 主从切换的处理

移植过程中的一些报错处理

- Ep addr xx overflow
- Ignore DQD in fs
- Interface num overflow
- Endpoint num overflow
- Failed to enable port

## errno

```
#define USB_ERR_NOMEM      1
#define USB_ERR_INVALID   2
#define USB_ERR_NODEV     3
#define USB_ERR_NOTCONN   4
#define USB_ERR_NOTSUPP   5
#define USB_ERR_BUSY      6
#define USB_ERR_RANGE     7
#define USB_ERR_STALL     8
#define USB_ERR_BABBLE    9
#define USB_ERR_NAK      10
#define USB_ERR_DT       11
#define USB_ERR_IO       12
#define USB_ERR_SHUTDOWN 13
#define USB_ERR_TIMEOUT  14
```

# RT-Thread 移植

---

```
#include "rtthread.h"

#define CHERRYUSB_VERSION      0x010100
#define CHERRYUSB_VERSION_STR "v1.1.0"

/* ===== USB common Configuration ===== */

#define CONFIG_USB_PRINTF(...) rt_kprintf(__VA_ARGS__)

#define usb_malloc(size) rt_malloc(size)
#define usb_free(ptr)    rt_free(ptr)
```

memcpy 需要改成 rt\_memcpy