|  |
| --- |
| OpenCPU |
| SDK FTP API 说明 |
| 版本1.0日期2025-02-25 |

版权声明

版权所有 © 深圳市有方科技股份有限公司2025。深圳市有方科技股份有限公司保留所有权利。

未经深圳市有方科技股份有限公司书面同意，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

是深圳市有方科技股份有限公司所有商标。

本文档中出现的其他商标，由商标所有者所有。

说明

本文档的使用对象为系统工程师，开发工程师及测试工程师。

本设计指南为客户产品设计提供支持，客户须按照本文中的规范和参数进行产品设计和调试。如因客户操作不当造成的人身伤害和财产损失，有方概不承担责任。

由于产品版本升级或其它原因，本文档内容会在不预先通知的情况下进行必要的更新。

除非另有约定，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

深圳市有方科技股份有限公司为客户提供全方位的技术支持，任何垂询请直接联系您的客户经理或发送邮件至以下邮箱：

Sales@neoway.com

Support@neoway.com

公司网址：http://www.neoway.com

目录

[关于本文档 iv](#_Toc191391823)

[范围 iv](#_Toc191391824)

[读者对象 iv](#_Toc191391825)

[修订记录 iv](#_Toc191391826)

[符号约定 iv](#_Toc191391827)

[1 概述 5](#_Toc191391828)

[2 数据结构 6](#_Toc191391829)

[2.1 nwy\_ftp\_handle\_t 6](#_Toc191391830)

[2.2 nwy\_ ftp\_put\_mode\_e 6](#_Toc191391831)

[2.3 nwy\_ftp\_param\_t 6](#_Toc191391832)

[2.4 nwy\_ ssl\_ftp\_type\_e 6](#_Toc191391833)

[2.5 nwy\_ftp\_ssl\_cfg\_t 6](#_Toc191391834)

[2.6 ftp 事件nwy\_ftp\_event\_e 7](#_Toc191391835)

[2.7 回调函数定义 8](#_Toc191391836)

[3 函数 9](#_Toc191391837)

关于本文档

范围

本文档对应产品为N706B系列（RTOS系统）。

读者对象

本文档的使用对象为系统工程师，开发工程师及测试工程师。

修订记录

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 日期 | 变更 | 作者 |
| 1.0 | 2025-01 | 初始版本 | Liu Feifan |

符号约定

|  |  |
| --- | --- |
| 符号 | 含义 |
|  | 危险或警告，用户必须遵从的规则，否则会造成模块或客户设备不可逆的故障损坏，甚至可能造成人员身体伤害。 |
|  | 注意，警示用户使用模块时应该特别注意的地方，如不遵从，模块或客户设备可能出现故障。 |
|  | 说明或提示，提供模块使用的意见或建议。 |

# 概述

本文档介绍了OpenCPU SDK中FTP/SSL FTP操作相关的API。

# 数据结构

## nwy\_ftp\_handle\_t

typedef void \*nwy\_ftp\_handle\_t;

## nwy\_ ftp\_put\_mode\_e

typedef enum

{

FTP\_STOR\_MODE=1,//在服务器上创建一个文件以写入数据，如果该文件存在，则覆盖源文件

FTP\_APPE\_MODE ,//在服务器上创建一个文件以写入数据，如果该文件存在，则附加到文件的末尾

}nwy\_ftp\_put\_mode\_e;

## nwy\_ftp\_param\_t

typedef struct{

unsigned short cid;

int timeout\_s;

char \*host;

int port;

char \*username;

char \*password;

unsigned char mode;//0 pasv 1:port

nwy\_ftp\_resultcb cb;//callback function

}nwy\_ftp\_param\_t;

## nwy\_ ssl\_ftp\_type\_e

typedef enum

{

SSLFTP\_TYPE\_EXPLICIT =1,//显示ssl

SSLFTP\_TYPE\_IMPLICIT //隐示ssl

}nwy\_ssl\_ftp\_type\_e;

## nwy\_ftp\_ssl\_cfg\_t

typedef struct{

nwy\_ssl\_ftp\_type\_e ftps\_type;

nwy\_ssl\_conf\_t ssl\_config;

}nwy\_ftp\_ssl\_cfg\_t;

## ftp 事件nwy\_ftp\_event\_e

typedef enum

{

NWY\_FTP\_EVENT\_DNS\_ERR,

NWY\_FTP\_EVENT\_OPEN\_FAIL,

NWY\_FTP\_EVENT\_CLOSED,

NWY\_FTP\_EVENT\_LOGOUT,

NWY\_FTP\_EVENT\_DATA\_GET,

NWY\_FTP\_EVENT\_DATA\_GET\_ERROR,

NWY\_FTP\_EVENT\_DATA\_OPEND,

NWY\_FTP\_EVENT\_DATA\_CLOSED,

NWY\_FTP\_EVENT\_DATA\_SETUP\_ERROR,

NWY\_FTP\_EVENT\_DATA\_GET\_FINISHED,

NWY\_FTP\_EVENT\_LOGIN,

NWY\_FTP\_EVENT\_SIZE,

NWY\_FTP\_EVENT\_PASS\_ERROR,

NWY\_FTP\_EVENT\_FILE\_NOT\_FOUND,

NWY\_FTP\_EVENT\_FILE\_SIZE\_ERROR,

NWY\_FTP\_EVENT\_FILE\_DELE\_SUCCESS,

NWY\_FTP\_EVENT\_FILE\_DELE\_ERROR

NWY\_FTP\_EVENT\_DATA\_PUT\_FINISHED,

NWY\_FTP\_EVENT\_DATA\_PUT\_ERROR,

NWY\_FTP\_EVENT\_CLOSED\_SSL\_FAIL,

NWY\_FTP\_EVENT\_SSL\_CONNECT\_FAIL,

NWY\_FTP\_EVENT\_SSL\_CONNECTED,

NWY\_FTP\_EVENT\_UNKOWN

}nwy\_ftp\_event\_e;

|  |  |
| --- | --- |
| 事件id | 说明 |
| NWY\_FTP\_EVENT\_DNS\_ERR, | DNS 解析错误 |
| NWY\_FTP\_EVENT\_OPEN\_FAIL | 控制链路Socket 打开失败 |
| NWY\_FTP\_EVENT\_CLOSED | 控制链路Socket 关闭 |
| NWY\_FTP\_EVENT\_LOGOUT | 退出ftp登录 |
| NWY\_FTP\_EVENT\_DATA\_GET | 收到ftp get 数据 |
| NWY\_FTP\_EVENT\_DATA\_GET\_ERROR | ftp get 错误 |
| NWY\_FTP\_EVENT\_DATA\_OPEND | ftp 数据链路打开成功 |
| NWY\_FTP\_EVENT\_DATA\_CLOSED | ftp 数据链路关闭 |
| NWY\_FTP\_EVENT\_DATA\_SETUP\_ERROR | ftp 数据链路打开失败 |
| NWY\_FTP\_EVENT\_DATA\_GET\_FINISHED | ftp get 完成 |
| NWY\_FTP\_EVENT\_LOGIN | ftp登录成功 |
| NWY\_FTP\_EVENT\_SIZE | ftp获取文件大小 |
| NWY\_FTP\_EVENT\_PASS\_ERROR | ftp密码错误 |
| NWY\_FTP\_EVENT\_FILE\_NOT\_FOUND | 服务器没有发现要获取的文件 |
| NWY\_FTP\_EVENT\_FILE\_SIZE\_ERROR | ftp 文件大小获取失败 |
| NWY\_FTP\_EVENT\_FILE\_DELE\_SUCCESS | 服务器删除文件成功 |
| NWY\_FTP\_EVENT\_DATA\_PUT\_FINISHE | ftp 上传文件完成 |
| NWY\_FTP\_EVENT\_DATA\_PUT\_ERROR | ftp put 失败 |
| NWY\_FTP\_EVENT\_CLOSED\_SSL\_FAIL | ssl 关闭失败 |
| NWY\_FTP\_EVENT\_SSL\_CONNECT\_FAIL | ssl连接失败 |
| NWY\_FTP\_EVENT\_SSL\_CONNECTED | ssl 连接成功 |
| NWY\_FTP\_EVENT\_UNKOWN | ssl 内部错误 |

## 回调函数定义

typedef struct{

unsigned short channel;//数据通道

nwy\_ftp\_handle\_t ftp\_handle; //ftp 客户端句柄

unsigned int event; /\*事件id \*/

unsigned int data\_len; /\*获取接收的数据长度 \*/

char\* data; /\* 获取的数据内容\*/

}nwy\_ftp\_result\_t;

typedef void (\*nwy\_ftp\_resultcb)(nwy\_ftp\_result\_t \*event);

# 函数

本章节提供FTP相关的操作接口。

此接口函数定义位于nwy\_ftp\_api.h，用于进行FTP相关操作。

加密模式的FTP需要调用ssl 配置文件nwy\_ssl\_config.h。

* + - 1. nwy\_ftp\_login

|  |  |
| --- | --- |
| 格式 | nwy\_ftp\_handle\_t nwy\_ftp\_login(nwy\_ftp\_param\_t \*param，nwy\_ftp\_ssl\_cfg\_t \*sslcfg) |
| 描述 | 登录ftp/sslftp服务器 |
| 参数说明 | nwy\_ftp\_param\_t \*param：ftp 参数  unsigned short channel :数据通道（1~7），每一个通道支持建立一个ftp客户端  int timeout\_s： 超时时间  char \*host: 服务器地址或者域名  int port：服务器端口  char \*username：登录用户名  char \*password：登录密码  unsigned char mode：0 被动模式 1:主动模式  nwy\_ftp\_resultcb cb：ftp状态回调函数  nwy\_ftp\_ssl\_cfg\_t \*sslcfg： ssl ftp 配置参数,ssl配置查看头文件nwy\_ssl\_config.h，参数NULL为不加密模式 |
| 返回值 | 返回ftp客户端句柄：  返回NULL表示错误，非NULL 表示成功 |
| 注意事项 | 最多支持两个ftp客户端同时在线 |

* + - 1. nwy\_ftp\_logout

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_ftp\_logout (nwy\_ftp\_handle\_t handle); |
| 描述 | 注销ftp/sslftp连接 |
| 参数说明 | handle：ftp/sslftp 客户端句柄 |
| 返回值 | 检查nwy\_error\_e |

* + - 1. nwy\_ftp\_get

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_ftp\_get(nwy\_ftp\_handle\_t handle,nwy\_ftp\_get\_param\_t \*ftp\_get\_param) |
| 描述 | 从ftp/sslftp服务器下载文件 |
| 参数说明 | nwy\_ftp\_handle\_t handle ：ftp/sslftp 客户端句柄  typedef struct{  char \* remotefile;  unsigned char type;  int offset;  int len;  }nwy\_ftp\_get\_param\_t;  结构体参数类型：  char \* remotefile：要获取的文件名  unsigned char type： 1、ascii 2、binary  int offset：文件偏移量  int len：当前要获取的文件长度； |
| 返回值 | 检查nwy\_error\_e |
| 注意事项 | 参数 offset 和len 为0 默认为全包下载。  整包数据大小len 可以通过调用接口函数nwy\_ftp\_filesize后从回调事件NWY\_FTP\_EVENT\_SIZE的响应中获取 |

* + - 1. nwy\_ftp\_put

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_ftp\_put(nwy\_ftp\_handle\_t handle, nwy\_ftp\_put\_param\_t \*ftp\_put\_param); |
| 描述 | 上传数据到服务器指定的文件。 |
| 参数 | nwy\_ftp\_handle\_t handle ：ftp/sslftp 客户端句柄  结构体参数类型：  typedef struct{  unsigned char eof;  char \*remotefile;  unsigned char \*data;  int len;  unsigned char type;  nwy\_ftp\_put\_mode\_e mode;  }nwy\_ftp\_put\_param\_t;  unsigned char eof：数据上传是否结束。0x00：表示没结束，0x01：表示结束  char \*remotefile: 上传数据到服务器指定的文件名  unsigned char \*data：需要写入文件的数据  int len：写入的数据长度（不大于8096）  unsigned char type： 1、ascii 2、binary  nwy\_ftp\_put\_mode\_e mode：  1 在服务器上创建一个文件以写入数据，如果该文件存在，则覆盖源文件  2 在服务器上创建一个文件以写入数据，如果该文件存在，则附加到文件的末尾 |
| 返回值 | 检查nwy\_error\_e |

* + - 1. nwy\_ftp\_get\_filesize

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_ftp\_get\_filesize(nwy\_ftp\_handle\_t handle, const char\* filename); |
| 描述 | 获取服务器上文件长度 |
| 参数说明 | nwy\_ftp\_handle\_t handle：ftp/sslftp 客户端句柄  const char\* remotefile: 要获取的文件名 |
| 返回值 | 检查nwy\_error\_e |

* + - 1. nwy\_ftp\_del\_file

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_ftp\_del\_file(nwy\_ftp\_handle\_t handle, const char \*remotefile); |
| 描述 | 删除服务器上文件 |
| 参数说明 | handle ：ftp/sslftp 客户端句柄  remotefile: 要删除的文件名 |
| 返回值 | 检查nwy\_error\_e |