|  |
| --- |
| OpenCPU |
| SDK Network API说明 |
| 版本1.0日期2025-03-03 |

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

本文档对应产品为N706B。

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 v](#_Toc191908621)

[范围 v](#_Toc191908622)

[读者对象 v](#_Toc191908623)

[修订记录 v](#_Toc191908624)

[符号约定 v](#_Toc191908625)

[1 范围 6](#_Toc191908626)

[2 数据结构 7](#_Toc191908627)

[2.1 Enum 7](#_Toc191908628)

[2.2 结构体、联合体 13](#_Toc191908629)

[3 APIs 22](#_Toc191908630)

[3.1 nwy\_nw\_mode\_get 22](#_Toc191908631)

[3.2 nwy\_nw\_mode\_set 22](#_Toc191908632)

[3.3 nwy\_nw\_lock\_get 23](#_Toc191908633)

[3.4 nwy\_nw\_lock\_set 23](#_Toc191908634)

[3.5 nwy\_nw\_priband\_get 23](#_Toc191908635)

[3.6 nwy\_nw\_priband\_set 23](#_Toc191908636)

[3.7 nwy\_nw\_ims\_get 24](#_Toc191908637)

[3.8 nwy\_nw\_ims\_set 24](#_Toc191908638)

[3.9 nwy\_nw\_radio\_get 24](#_Toc191908639)

[3.10 nwy\_nw\_radio\_set 24](#_Toc191908640)

[3.11 nwy\_nw\_edrx\_get 25](#_Toc191908641)

[3.12 nwy\_nw\_edrx\_set 25](#_Toc191908642)

[3.13 nwy\_nw\_psm\_get 25](#_Toc191908643)

[3.14 nwy\_nw\_psm\_set 25](#_Toc191908644)

[3.15 nwy\_nw\_config\_get 26](#_Toc191908645)

[3.16 nwy\_nw\_config\_set 26](#_Toc191908646)

[3.17 nwy\_nw\_operator\_get 26](#_Toc191908647)

[3.18 nwy\_nw\_csq\_get 27](#_Toc191908648)

[3.19 nwy\_nw\_signal\_get 27](#_Toc191908649)

[3.20 nwy\_nw\_regstatus\_get 27](#_Toc191908650)

[3.21 nwy\_nw\_cellinfo\_get 27](#_Toc191908651)

[3.22 nwy\_nw\_scan 28](#_Toc191908652)

[3.23 nwy\_nw\_manual\_select 28](#_Toc191908653)

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 | 2025-02 | 初始版本 | Zhang Yanhong |

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。 |

# 范围

本文档介绍了OpenCSDK中Network相关的API，主要包括以下几个方面：

* 锁网络制式、频段、频点
* 进出飞行模式
* 获取网络信号信息
* 获取周围网络小区信息
* 进行PSM等低功耗操作

# 数据结构

## Enum

由于要兼容2.3.4.5G的不同网络信息，数据结构过多，建议直接参考.h文件。

typedef enum {

NWY\_NW\_MODE\_MASK\_AUTO = 0x0,//unlocked mode

NWY\_NW\_MODE\_MASK\_CDMA\_1X = 0x1,

NWY\_NW\_MODE\_MASK\_CDMA\_HDR = 0x2,

NWY\_NW\_MODE\_MASK\_GSM = 0x4,

NWY\_NW\_MODE\_MASK\_WCDMA = 0x8,

NWY\_NW\_MODE\_MASK\_LTE = 0x10,//EUTRAN, Not included CATM1 and NB

NWY\_NW\_MODE\_MASK\_TDSCDMA = 0x20,

NWY\_NW\_MODE\_MASK\_NR = 0x40,

NWY\_NW\_MODE\_MASK\_CATM = 0x80,

NWY\_NW\_MODE\_MASK\_NB = 0x100,

} nwy\_nw\_mode\_mask\_e;

typedef enum {

NWY\_NW\_RAT\_NONE = 0, //not reg status

NWY\_NW\_RAT\_GSM = 1,

NWY\_NW\_RAT\_WCDMA = 2,

NWY\_NW\_RAT\_LTE = 3, //EUTRAN, Not included CATM1 and NB

NWY\_NW\_RAT\_CATM = 4,

NWY\_NW\_RAT\_NBIoT = 5,

NWY\_NW\_RAT\_EUTRA\_TO\_5GCN = 6, //LTE, core network is 5G

NWY\_NW\_RAT\_NR\_TO\_5GCN = 7, //NR, core network is 5G

NWY\_NW\_RAT\_NR\_TO\_EPS = 8, //NR, core network is 4G

NWY\_NW\_RAT\_EUTRA\_NR\_DUAL = 9, //LTE + NR

NWY\_NW\_RAT\_CDMA = 10,

NWY\_NW\_RAT\_HDR = 11,

NWY\_NW\_RAT\_TDSCDMA = 12,

NWY\_NW\_RAT\_MAX = 0xff

} nwy\_nw\_rat\_type\_e;

typedef enum {

NWY\_NW\_LOCK\_MODE\_UNLOCK = 0, //UNLOCK all

NWY\_NW\_LOCK\_MODE\_LOCK\_BAND = 1, //lock band

NWY\_NW\_LOCK\_MODE\_LOCK\_FREQ = 2, //lock freq

NWY\_NW\_LOCK\_MODE\_LOCK\_PCI = 3 //lock cell

} nwy\_nw\_lockmode\_e;

typedef enum {

NWY\_NW\_PRI\_BAND\_UNLOCK = 0,

NWY\_NW\_PRI\_BAND\_LOCK = 1,

NWY\_NW\_PRI\_BAND\_MAX

} nwy\_nw\_priband\_mode\_e;

typedef enum {

NWY\_NW\_RADIO\_MIN\_MODE = 0, //min mode, close rf/sim

NWY\_NW\_RADIO\_NORMAL\_MODE = 1, //normal mode, open rf and sim

NWY\_NW\_RADIO\_FLIGHT\_MODE = 4, //flight mode, close rf

} nwy\_nw\_radio\_mode\_e;

typedef enum {

NWY\_NW\_EDRX\_MODE\_DISABLE = 0,

NWY\_NW\_EDRX\_MODE\_ENABLE = 1,

NWY\_NW\_EDRX\_MODE\_ENABLE\_REPORT = 2,

NWY\_NW\_EDRX\_MODE\_RESET\_SETTING = 3,

} nwy\_nw\_edrx\_mode\_e;

typedef enum {

NWY\_NW\_PSM\_MODE\_DISABLE = 0,

NWY\_NW\_PSM\_MODE\_ENABLE = 1,

NWY\_NW\_PSM\_MODE\_RESET = 2,

} nwy\_nw\_psm\_mode\_e;

typedef enum {

NWY\_NW\_CONFIG\_RW\_NETAUTO = 0,

NWY\_NW\_CONFIG\_RW\_UEMODE = 1,

NWY\_NW\_CONFIG\_RW\_DATAONLY = 2,

NWY\_NW\_CONFIG\_RW\_ANTAUX = 3,

NWY\_NW\_CONFIG\_R\_BAND\_CAP =4,

NWY\_NW\_CONFIG\_RW\_NITZ\_ONOFF = 5,

} nwy\_nw\_config\_type\_e;

typedef enum {

NWY\_NW\_SPN\_ENC\_GSM7 = 0x00, /\* 7bit code \*/

NWY\_NW\_SPN\_ENC\_UCS2 = 0x01, /\* UCS2 code \*/

}nwy\_nw\_spn\_enc\_e;

typedef enum {

NWY\_NW\_SERVICE\_NONE = 0, //out of service

NWY\_NW\_SERVICE\_LIMITED = 1,

NWY\_NW\_SERVICE\_FULL = 2,

}nwy\_nw\_service\_type\_e;

typedef enum {

NWY\_NW\_ROAM\_STATE\_OFF = 0,

NWY\_NW\_ROAM\_STATE\_ON = 1,

}nwy\_nw\_roam\_state\_e;

typedef enum {

NWY\_BANDWIDTH\_ENUM\_TYPE\_MIN = -1,

NWY\_LTE\_BW\_NRB\_6\_V01 = 0, /\*\*< LTE 1.4 MHz bandwidth \n \*/

NWY\_LTE\_BW\_NRB\_15\_V01 = 1, /\*\*< LTE 3 MHz bandwidth \n \*/

NWY\_LTE\_BW\_NRB\_25\_V01 = 2, /\*\*< LTE 5 MHz bandwidth \n \*/

NWY\_LTE\_BW\_NRB\_50\_V01 = 3, /\*\*< LTE 10 MHz bandwidth \n \*/

NWY\_LTE\_BW\_NRB\_75\_V01 = 4, /\*\*< LTE 15 MHz bandwidth \n \*/

NWY\_LTE\_BW\_NRB\_100\_V01 = 5, /\*\*< LTE 20 MHz bandwidth \n \*/

NWY\_NR5G\_BW\_NRB\_6\_V01 = 6, /\*\*< NR5G 1.4 MHz bandwidth \n \*/

NWY\_NR5G\_BW\_NRB\_15\_V01 = 7, /\*\*< NR5G 3 MHz bandwidth \n \*/

NWY\_NR5G\_BW\_NRB\_25\_V01 = 8, /\*\*< NR5G 5 MHz bandwidth \n \*/

NWY\_NR5G\_BW\_NRB\_50\_V01 = 9, /\*\*< NR5G 10 MHz bandwidth \n \*/

NWY\_NR5G\_BW\_NRB\_75\_V01 = 10, /\*\*< NR5G 15 MHz bandwidth \n \*/

NWY\_NR5G\_BW\_NRB\_100\_V01 = 11, /\*\*< NR5G 20 MHz bandwidth \*/

NWY\_BANDWIDTH\_ENUM\_TYPE\_MAX = 2147483647

}nwy\_bandwidth\_type\_e;

typedef enum {

NWY\_CDMA\_PILOT\_TYPE\_ENUM\_MIN\_ENUM\_VAL\_V01 = -2147483647, NWY\_CDMA\_PILOT\_CURR\_ACT\_PLT\_V01 = 0x00,

NWY\_CDMA\_PILOT\_NEIGHBOR\_PLT\_V01 = 0x01,

NWY\_CDMA\_PILOT\_TYPE\_ENUM\_MAX\_ENUM\_VAL\_V01 = 2147483647

}nwy\_cdma\_pilot\_e;

typedef enum {

NWY\_CDMA\_CLASS\_NONE = -1,

NWY\_CDMA\_BC0 = 0, /\*\*< Band Class 0: U.S. Cellular band (800 MHz). \*/

NWY\_CDMA\_BC1 = 1, /\*\*< Band Class 1: U.S. \*/

NWY\_CDMA\_BC3 = 3, /\*\*< Band Class 3: Japanese Cellular band (800 MHz). \*/

NWY\_CDMA\_BC4 = 4, /\*\*< Band Class 4: Korean PCS band (1800 MHz). \*/

NWY\_CDMA\_BC5 = 5, /\*\*< Band Class 5 (450 MHz). \*/

NWY\_CDMA\_BC6 = 6, /\*\*< Band Class 6 (2 GHz). \*/

NWY\_CDMA\_BC7 = 7, /\*\*< Band Class 7 (Upper 700 MHz). \*/

NWY\_CDMA\_BC8 = 8, /\*\*< Band Class 8 (1800 MHz). \*/

NWY\_CDMA\_BC9 = 9, /\*\*< Band Class 9 (900 MHz). \*/

NWY\_CDMA\_BC10 = 10, /\*\*< Band Class 10 (Second 800 MHz). \*/

NWY\_CDMA\_BC11 = 11, /\*\*< Band Class 11: European PAMR Band (400 MHz). \*/

NWY\_CDMA\_BC12 = 12, /\*\*< Band Class 12: PAMR Band (800 MHz). \*/

NWY\_CDMA\_BC13 = 13, /\*\*< Band Class 13: currently undefined. \*/

NWY\_CDMA\_BC14 = 14, /\*\*< Band Class 14 (U.S. PCS 1.9 GHz Band). \*/

NWY\_CDMA\_BC15 = 15, /\*\*< Band Class 15 (1700-2100 MHz -AWS). \*/

NWY\_CDMA\_BC16 = 16, /\*\*< Band Class 16 (U.S. 2.5 GHz). \*/

NWY\_CDMA\_BC17 = 17, /\*\*< Band Class 17 (U.S. 2.5 GHz Forward Link only band). \*/

NWY\_CDMA\_BC18 = 18, /\*\*< Band Class 18 (700 MHz Public Safety Broadband). \*/

NWY\_CDMA\_BC19 = 19, /\*\*< Band Class 19 (Lower 700 MHz band).\*/

NWY\_CDMA\_BC\_MAX = 20,

/\*\*< Upper boundary for CDMA band classes. \*/

}nwy\_band\_cdma\_class\_e;

typedef enum {

NWY\_GSM\_NONE = -1,

NWY\_GSM\_450 = 0, /\*\*< GSM 450 band (450 MHz). \*/

NWY\_GSM\_480 = 1, /\*\*< GSM 480 band (480 MHz). \*/

NWY\_GSM\_750 = 2, /\*\*< GSM 750 band (750 MHz). \*/

NWY\_GSM\_850 = 3, /\*\*< GSM 850 band (850 MHz). \*/

NWY\_GSM\_EGSM\_900 = 4, /\*\*< GSM Extended GSM (E-GSM) 900 band (900 MHz). \*/

NWY\_GSM\_PGSM\_900 = 5, /\*\*< GSM Primary GSM (P-GSM) 900 band (900 MHz). \*/

NWY\_GSM\_RGSM\_900 = 6, /\*\*< GSM Railways GSM (R-GSM) 900 band (900 MHz). \*/

NWY\_GSM\_DCS\_1800 = 7, /\*\*< GSM DCS band (1800 MHz). \*/

NWY\_GSM\_PCS\_1900 = 8, /\*\*< GSM PCS band (1900 MHz). \*/

}nwy\_band\_gsm\_class\_e;

typedef enum {

NWY\_LTE\_CLASS\_NONE = 0,/\* LTE bands\*/

NWY\_LTE\_EUTRAN\_BAND1 = 1, /\*\*< UL:1920-1980; DL:2110-2170. \*/

NWY\_LTE\_EUTRAN\_BAND2 = 2, /\*\*< UL:1850-1910; DL:1930-1990. \*/

NWY\_LTE\_EUTRAN\_BAND3 = 3, /\*\*< UL:1710-1785; DL:1805-1880. \*/

NWY\_LTE\_EUTRAN\_BAND4 = 4, /\*\*< UL:1710-1755; DL:2110-2155. \*/

NWY\_LTE\_EUTRAN\_BAND5 = 5, /\*\*< UL: 824-849; DL: 869- 894. \*/

NWY\_LTE\_EUTRAN\_BAND6 = 6, /\*\*< UL: 830-840; DL: 875-885. \*/

NWY\_LTE\_EUTRAN\_BAND7 = 7, /\*\*< UL:2500-2570; DL:2620-2690. \*/

NWY\_LTE\_EUTRAN\_BAND8 = 8, /\*\*< UL: 880-915; DL: 925-960. \*/

NWY\_LTE\_EUTRAN\_BAND9 = 9, /\*\*< UL:1749.9-1784.9; DL:1844.9-1879.9. \*/

NWY\_LTE\_EUTRAN\_BAND10 = 10, /\*\*< UL:1710-1770; DL:2110-2170. \*/

NWY\_LTE\_EUTRAN\_BAND11 = 11, /\*\*< UL:1427.9-1452.9; DL:1475.9-1500.9. \*/

NWY\_LTE\_EUTRAN\_BAND12 = 12, /\*\*< UL:698-716; DL:728-746. \*/

NWY\_LTE\_EUTRAN\_BAND13 = 13, /\*\*< UL: 777-787; DL: 746-756. \*/

NWY\_LTE\_EUTRAN\_BAND14 = 14, /\*\*< UL: 788-798; DL: 758-768. \*/

/\* Reserved for BAND 15, 16 \*/

NWY\_LTE\_EUTRAN\_BAND17 = 17, /\*\*< UL: 704-716; DL: 734-746. \*/

NWY\_LTE\_EUTRAN\_BAND18 = 18, /\*\*< UL: 815-830; DL: 860-875. \*/

NWY\_LTE\_EUTRAN\_BAND19 = 19, /\*\*< UL: 830-845; DL: 875-890. \*/

NWY\_LTE\_EUTRAN\_BAND20 = 20, /\*\*< UL: 832-862; DL: 791-821. \*/

NWY\_LTE\_EUTRAN\_BAND21 = 21, /\*\*< UL: 1447.9-1462.9; DL: 1495.9-1510.9. \*/

NWY\_LTE\_EUTRAN\_BAND23 = 23, /\*\*< UL: 2000-2020; DL: 2180-2200 \*/

NWY\_LTE\_EUTRAN\_BAND24 = 24, /\*\*< UL: 1626.5-1660.5; DL: 1525 -1559. \*/

NWY\_LTE\_EUTRAN\_BAND25 = 25, /\*\*< UL: 1850-1915; DL: 1930 -1995 . \*/

NWY\_LTE\_EUTRAN\_BAND26 = 26, /\*\*< UL: 814-849; DL: 859 -894 . \*/

NWY\_LTE\_EUTRAN\_BAND27 = 27, /\*\*< UL: 807.5 - 824; DL: 852 - 869 \*/

NWY\_LTE\_EUTRAN\_BAND28 = 28, /\*\*<UL: 703-748; DL: 758-803 . \*/

NWY\_LTE\_EUTRAN\_BAND29 = 29, /\*\*< UL: 1850-1910 or 1710 - 1755; DL: 716-728. \*/

NWY\_LTE\_EUTRAN\_BAND30 = 30, /\*\*< UL: 2305 - 2315 ; DL: 2350 - 2360 \*/

NWY\_LTE\_EUTRAN\_BAND31 = 31, /\*\*< UL: 452.5 - 457.5 ; DL: 462.5 - 467.5 \*/

NWY\_LTE\_EUTRAN\_BAND32 = 32, /\*\*< DL: 9920 - 10359 \*/

NWY\_LTE\_EUTRAN\_BAND33 = 33, /\*\*< UL: 1900-1920; DL: 1900-1920. \*/

NWY\_LTE\_EUTRAN\_BAND34 = 34, /\*\*< UL: 2010-2025; DL: 2010-2025. \*/

NWY\_LTE\_EUTRAN\_BAND35 = 35, /\*\*< UL: 1850-1910; DL: 1850-1910. \*/

NWY\_LTE\_EUTRAN\_BAND36 = 36, /\*\*< UL: 1930-1990; DL: 1930-1990. \*/

NWY\_LTE\_EUTRAN\_BAND37 = 37, /\*\*< UL: 1910-1930; DL: 1910-1930. \*/

NWY\_LTE\_EUTRAN\_BAND38 = 38, /\*\*< UL: 2570-2620; DL: 2570-2620. \*/

NWY\_LTE\_EUTRAN\_BAND39 = 39, /\*\*< UL: 1880-1920; DL: 1880-1920. \*/

NWY\_LTE\_EUTRAN\_BAND40 = 40, /\*\*< UL: 2300-2400; DL: 2300-2400. \*/

NWY\_LTE\_EUTRAN\_BAND41 = 41, /\*\*< UL: 2496-2690; DL: 2496-2690. \*/

NWY\_LTE\_EUTRAN\_BAND42 = 42, /\*\*< UL: 3400-3600; DL: 3400-3600. \*/

NWY\_LTE\_EUTRAN\_BAND43 = 43, /\*\*< UL: 3600-3800; DL: 3600-3800. \*/

NWY\_LTE\_EUTRAN\_BAND46 = 46, /\*\*< DL: 5150-5925 MHz. \*/

NWY\_LTE\_EUTRAN\_BAND47 = 47, /\*\*< DL: ? MHz. \*/

NWY\_LTE\_EUTRAN\_BAND48 = 48, /\*\*< DL: ? MHz. \*/

NWY\_LTE\_EUTRAN\_BAND49 = 49, /\*\*< UL: ?; DL: ? \*/

NWY\_LTE\_EUTRAN\_BAND53 = 53, /\*\*< 2483.5 - 2495 MHz TDD \*/

NWY\_LTE\_EUTRAN\_BAND59 = 59, /\*\*< 1785 - 1805MHz TDD \*/

NWY\_LTE\_EUTRAN\_BAND67 = 67, /\*\*< UL: ?; DL: ? \*/

NWY\_LTE\_EUTRAN\_BAND66 = 66, /\*\*< UL: 1710-1780; DL: 2110-2200 \*/

NWY\_LTE\_EUTRAN\_BAND68 = 68, /\*\*< UL: ?; DL: ? \*/

NWY\_LTE\_EUTRAN\_BAND71 = 71, /\*\*< UL: ?; DL: ? \*/

NWY\_LTE\_EUTRAN\_BAND125 = 125, /\*\*< DL: 64835 - 64859 \*/

NWY\_LTE\_EUTRAN\_BAND126 = 126, /\*\*< DL: 64860 - 64974 \*/

NWY\_LTE\_EUTRAN\_BAND127 = 127, /\*\*< DL: 64975 - 64999 \*/

NWY\_LTE\_EUTRAN\_BAND250 = 250, /\*\*< UL: 3400-3800; DL: 3400 -3800 . \*/

NWY\_LTE\_EUTRAN\_BAND252 = 252, /\*\*< DL: 5150 - 5250 \*/

NWY\_LTE\_EUTRAN\_BAND255 = 255, /\*\*< DL: 5725 - 5850 \*/

}nwy\_band\_lte\_class\_e;

typedef enum {

NWY\_WCDMA\_CLASS\_NONE = 0,

NWY\_WCDMA\_I\_IMT\_2000 = 1, /\*\*< WCDMA Europe, Japan, and China IMT 2100 band. W1\*/

NWY\_WCDMA\_II\_PCS\_1900 = 2, /\*\*< WCDMA U.S. PCS 1900 band. W2\*/

NWY\_WCDMA\_III\_1800 = 3, /\*\*< WCDMA Europe and China DCS 1800 band. W3\*/

NWY\_WCDMA\_IV\_1700 = 4, /\*\*< WCDMA U.S. 1700 band. W4\*/

NWY\_WCDMA\_V\_850 = 5, /\*\*< WCDMA U.S. 850 band. W5\*/

NWY\_WCDMA\_VI\_800 = 6, /\*\*< WCDMA Japan 800 band. W6\*/

NWY\_WCDMA\_VII\_2600 = 7, /\*\*< WCDMA Europe 2600 band. W7\*/

NWY\_WCDMA\_VIII\_900 = 8, /\*\*< WCDMA Europe and China 900 band. W8\*/

NWY\_WCDMA\_IX\_1700 = 9, /\*\*< WCDMA Japan 1700 band. W9\*/

/\* Reserved W10 for WCDMA BC10-1700 band classes. \*/

NWY\_WCDMA\_XI\_1500 = 11, /\*\*< WCDMA 1500 band. W11\*/

NWY\_WCDMA\_XIX\_850 = 19, /\*\*< WCDMA Japan 850 band. W19\*/

}nwy\_band\_wcdma\_class\_e;

typedef enum {

NWY\_TDS\_CLASS\_NONE = 0, /\* TD-SCDMA bands \*/

NWY\_TDS\_BANDA = 1, /\*\*< TDS Band A 1900-1920 MHz, 2010-2020 MHz \*/

NWY\_TDS\_BANDB = 2, /\*\*< TDS Band B 1850-1910 MHz, 1930-1990 MHz \*/

NWY\_TDS\_BANDC = 3, /\*\*< TDS Band C 1910-1930 MHz \*/

NWY\_TDS\_BANDD = 4, /\*\*< TDS Band D 2570-2620 MHz \*/

NWY\_TDS\_BANDE = 5, /\*\*< TDS Band E 2300-2400 MHz \*/

NWY\_TDS\_BANDF = 6, /\*\*< TDS Band F 1880-1920 MHz \*/

}nwy\_band\_tdscdma\_class\_e;

typedef enum

{

NWY\_NR5G\_CLASS\_NONE = 0,

/\*NR bands in FR1\*/

NWY\_NR5G\_BAND1 = 1, /\*\*< UL: 1920 - 1980 MHz, DL: 2110 - 2170 MHz, FDD\*/

NWY\_NR5G\_BAND2 = 2, /\*\*< 1850 - 1910 MHz 1930 - 1990 MHz FDD \*/

NWY\_NR5G\_BAND3 = 3, /\*\*< 1710 - 1785 MHz 1805 - 1880 MHz FDD \*/

NWY\_NR5G\_BAND5 = 5, /\*\*< 824 - 849 MHz 869 - 894MHz FDD \*/

NWY\_NR5G\_BAND7 = 7, /\*\*< 2500 - 2570 MHz 2620 - 2690 MHz FDD \*/

NWY\_NR5G\_BAND8 = 8, /\*\*<880 - 915 MHz 925 - 960 MHz FDD \*/

NWY\_NR5G\_BAND12 = 12, /\*\*<729 - 746 MHz 699 - 716 MHz FDD \*/

NWY\_NR5G\_BAND13 = 13, /\*\*<777 - 787 MHz 746 MHz - 756 MHz FDD \*/

NWY\_NR5G\_BAND14 = 14, /\*\*<758 - 768 MHz 788 - 798 MHz FDD \*/

NWY\_NR5G\_BAND18 = 18, /\*\*<815 - 830 MHz 860 - 875 MHz FDD \*/

NWY\_NR5G\_BAND20 = 20, /\*\*< 832 - 862 MHz 791- 821MHz FDD \*/

NWY\_NR5G\_BAND25 = 25, /\*\*< 1930 - 1995 MHz 1850 - 1915 MHz FDD\*/

NWY\_NR5G\_BAND26 = 26, /\*\*<814 - 849 MHz 859 - 894 MHz FDD \*/

NWY\_NR5G\_BAND28 = 28, /\*\*< 703 - 748 MHz 758 - 803 MHz FDD \*/

NWY\_NR5G\_BAND29 = 29, /\*\*<N/A 717 MHz - 728 MHz SDL \*/

NWY\_NR5G\_BAND30 = 30, /\*\*<2305 - 2315 MHz 2350 - 2360 MHz FDD \*/

NWY\_NR5G\_BAND34 = 34, /\*\*< 2010 - 2025 MHz TDD \*/

NWY\_NR5G\_BAND38 = 38, /\*\*< 2570 - 2620 MHz 2570 - 2620 MHz TDD\*/

NWY\_NR5G\_BAND39 = 39, /\*\*< 1880 - 1920 MHz TDD \*/

NWY\_NR5G\_BAND40 = 40, /\*\*< 2300 - 2400 MHz TDD \*/

NWY\_NR5G\_BAND41 = 41, /\*\*<2496 - 2690 MHz 2496 - 2690 MHz TDD \*/

NWY\_NR5G\_BAND46 = 46, /\*\*< 5150-5925 MHz TDD \*/

NWY\_NR5G\_BAND48 = 48, /\*\*< 3550 - 3700 MHz TDD \*/

NWY\_NR5G\_BAND50 = 50, /\*\*< 1432 - 1517 MHz 1432 - 1517 MHz TDD \*/

NWY\_NR5G\_BAND51 = 51, /\*\*< 1427 - 1432 MHz 1427 - 1432 MHz TDD \*/

NWY\_NR5G\_BAND53 = 53, /\*\*< 2483.5 - 2495 MHz TDD \*/

NWY\_NR5G\_BAND65 = 65, /\*\*< 2110 - 2200 MHz 1920 - 2010 MHz FDD \*/

NWY\_NR5G\_BAND66 = 66, /\*\*< 1710 - 1780 MHz 2110 - 2200 MHz FDD\*/

NWY\_NR5G\_BAND70 = 70, /\*\*<1695 - 1710 MHz 1995- 2020 MHz FDD \*/

NWY\_NR5G\_BAND71 = 71, /\*\*< 663 - 698 MHz 617 - 652 MHz FDD \*/

NWY\_NR5G\_BAND74 = 74, /\*\*< 1427 -1470 MHz 1475 - 1518 MHz FDD \*/

NWY\_NR5G\_BAND75 = 75, /\*\*< N/A 1432 - 1517 MHz SDL\*/

NWY\_NR5G\_BAND76 = 76, /\*\*<N/A 1427 - 1432 MHz SDL \*/

NWY\_NR5G\_BAND77 = 77, /\*\*< 3.3 - 4.2 GHz 3.3 - 4.2 GHz TDD \*/

NWY\_NR5G\_BAND78 = 78, /\*\*< 3.3 - 3.8 GHz 3.3 - 3.8 GHz TDD \*/

NWY\_NR5G\_BAND79 = 79, /\*\*< 4.4 - 5.0 GHz 4.4 - 5.0 GHz TDD\*/

NWY\_NR5G\_BAND80 = 80, /\*\*<1710 - 1785 MHz N/A SUL \*/

NWY\_NR5G\_BAND81 = 81, /\*\*< 880 - 915 MHz N/A SUL \*/

NWY\_NR5G\_BAND82 = 82, /\*\*< 832 - 862 MHz N/A SUL \*/

NWY\_NR5G\_BAND83 = 83, /\*\*< 703 - 748 MHz N/A SUL\*/

NWY\_NR5G\_BAND84 = 84, /\*\*<1920 - 1980 MHz N/A SUL \*/

NWY\_NR5G\_BAND85 = 85, /\*\*< 2496 - 2690 MHz N/A SUL \*/

NWY\_NR5G\_BAND86 = 86, /\*\*< 1710 - 1780 MHz SUL \*/

NWY\_NR5G\_BAND91 = 91, /\*\*< 832 - 862 MHz 1427 - 1432 MHz FDD \*/

NWY\_NR5G\_BAND92 = 92, /\*\*< 832 - 862 MHz 1432 - 1517 MHz FDD \*/

NWY\_NR5G\_BAND93 = 93, /\*\*< 880 - 915 MHz 1427 - 1432 MHz FDD \*/

NWY\_NR5G\_BAND94 = 94, /\*\*< 880 - 915 MHz 1432 - 1517 MHz FDD \*/

/\*NR bands in FR2\*/

NWY\_NR5G\_BAND257 = 257, /\*\*< 26.5 -29.5 GHz 26.5 -29.5 GHz TDD \*/

NWY\_NR5G\_BAND258 = 258, /\*\*< 24.25 - 27.5 GHz 24.25 - 27.5 GHz TDD \*/

NWY\_NR5G\_BAND259 = 259, /\*\*< N/A N/A TDD \*/

NWY\_NR5G\_BAND260 = 260, /\*\*< 37-40 GHz 37-40 GHz TDD \*/

NWY\_NR5G\_BAND261 = 261, /\*\*< 27.5-28.3 GHz TDD \*/

}nwy\_band\_nr5g\_class\_e;

typedef enum {

NWY\_NW\_NET\_STAT\_UNKNOWN = 0, /\* Network state unknown \*/

NWY\_NW\_NET\_STAT\_AVAILABLE, /\* Network is available for registration \*/

NWY\_NW\_NET\_STAT\_CURRENT, /\* Network currently registered \*/

NWY\_NW\_NET\_STAT\_FORBIDDEN, /\* Network forbidden registraiton \*/

}nwy\_nw\_status\_e;

typedef enum {

NWY\_NW\_SELECT\_DEREG = 0,

NWY\_NW\_SELECT\_REG\_AUTO = 1,

NWY\_NW\_SELECT\_REG\_MANU = 2,

NWY\_NW\_SELECT\_REG\_MANU\_AUTO = 3,// if manual selection fails, automatic mode is entered

}nwy\_nw\_select\_mode\_e;

typedef enum {

NWY\_NW\_REGS\_NONE = -1,

NWY\_NW\_REGS\_DATA\_IND = 0, // data register state event

NWY\_NW\_REGS\_VOICE\_IND = 1, // voice register state event

}nwy\_nw\_regs\_ind\_type\_e;

typedef enum{

NWY\_NW\_CFG\_DISABLE = 0,

NWY\_NW\_CFG\_ENABLE = 1,

} nwy\_nw\_cfg\_switch\_type\_e;

typedef enum {

NWY\_NW\_PS\_MODE\_II = 0, //UE only register EPS service,data centric

NWY\_NW\_CS\_PS\_MODE\_I = 1, //UE registers both EPS/CS services, voice centric

NWY\_NW\_CS\_PS\_MODI\_II = 2, //UE registers both EPS/CS services, data centric

NWY\_NW\_PS\_MODE\_I = 3, //UE only register EPS service,voice centric

} nwy\_nw\_cfg\_uemode\_type\_e;

typedef enum{

NWY\_NW\_GET\_SCELL = 0, //serving cell

NWY\_NW\_GET\_ALL\_CELL = 1, //serving cell + neighborhood cell

} nwy\_nw\_cellinfo\_mode\_e;

## 结构体、联合体

typedef struct {

int band\_nums;

uint32 band[NWY\_NW\_LOCK\_FREQ\_MAX\_NUM];

}nwy\_nw\_lock\_band\_t;

typedef struct {

int freq\_nums;

uint32 freq[NWY\_NW\_LOCK\_FREQ\_MAX\_NUM];

}nwy\_nw\_lock\_freq\_t;

typedef struct {

uint32 freq;

int cell\_nums;

uint32 pci[NWY\_NW\_LOCK\_PCI\_MAX\_NUM];

}nwy\_nw\_lock\_pci\_info\_t;

typedef struct {

int lock\_pci\_nums;

nwy\_nw\_lock\_pci\_info\_t lock\_pci\_info[NWY\_NW\_LOCK\_PCI\_MAX\_NUM];

}nwy\_nw\_lock\_pci\_t;

typedef union {

nwy\_nw\_lock\_band\_t lock\_band;

nwy\_nw\_lock\_freq\_t lock\_freq;

nwy\_nw\_lock\_pci\_t lock\_pci;

}nwy\_nw\_lock\_info\_u;

typedef struct {

nwy\_nw\_lockmode\_e mode;

nwy\_nw\_mode\_mask\_e rat;

nwy\_nw\_lock\_info\_u lock\_info;

} nwy\_nw\_lock\_t;

typedef struct {

nwy\_nw\_priband\_mode\_e mode;

nwy\_nw\_mode\_mask\_e rat;

int band\_nums;

int band[NWY\_NW\_PRI\_BAND\_MAX\_NUM];

} nwy\_nw\_priband\_t;

typedef struct {

nwy\_nw\_edrx\_mode\_e mode;

nwy\_nw\_mode\_mask\_e rat; //radio access technology,ex.gsm/wcdma..

uint8 requested\_eDRX\_value; //see Extended DRX parameters information element in 3GPP TS 24.008 Table 10.5.5.32

uint8 nw\_provided\_eDRX\_value;

uint8 Paging\_time\_window;

} nwy\_nw\_edrx\_info\_t;

typedef struct {

nwy\_nw\_psm\_mode\_e mode;

uint8 requested\_Periodic\_rau; //T3312, see the GPRS Timer 3 IE in 3GPP TS 24.008 Table 10.5.163a

uint8 requested\_gprs\_ready\_timer; //T3314, see the GPRS Timer IE in 3GPP TS 24.008 Table 10.5.172

uint8 requested\_Periodic\_tau; //T3412, see the GPRS Timer 3 IE in 3GPP TS 24.008 Table 10.5.163a

uint8 requested\_active\_timer; //T3324, see the GPRS Timer 2 IE in 3GPP TS 24.008 Table 10.5.163

} nwy\_nw\_psm\_info\_t;

typedef struct {

char long\_eons[NWY\_NW\_EONS\_MAX\_LEN + 1]; //long operator name,ex."CHINA MOBILE"

char short\_eons[NWY\_NW\_EONS\_MAX\_LEN + 1]; //short operator name,ex."CMCC"

char mcc[NWY\_NW\_MNC\_MCC\_LEN + 1]; //Mobile Country Code

char mnc[NWY\_NW\_MNC\_MCC\_LEN + 1]; //Mobile Network Code

char spn[NWY\_NW\_SPN\_LEN\_MAX + 1]; //Servoce Provider Name,0xFF - InValid

nwy\_nw\_spn\_enc\_e spn\_enc;

}nwy\_nw\_operator\_t;

typedef struct {

int rssi;

int ber;

}nwy\_nw\_gsm\_signal\_info\_t;

typedef struct {

int rssi;

int ecio;

}nwy\_nw\_cdma\_hdr\_signal\_info\_t;

typedef struct {

int rscp;

int ecio;

int rssi;

}nwy\_nw\_wcdma\_signal\_info\_t;

typedef struct {

int rsrp;

int rsrq;

int rssi;

int sinr;

}nwy\_nw\_lte\_signal\_info\_t;

typedef struct {

int rsrp;

int rsrq;

int rssi;

int sinr;

}nwy\_nw\_nr\_signal\_info\_t;

typedef struct {

int ecno;

int rscp;

int rssi;

}nwy\_nw\_tdscdma\_signal\_info\_t;

typedef union{

nwy\_nw\_gsm\_signal\_info\_t gsm\_signal\_info;

nwy\_nw\_wcdma\_signal\_info\_t wcdma\_signal\_info;

nwy\_nw\_cdma\_hdr\_signal\_info\_t cdmd\_hdr\_signal\_info;

nwy\_nw\_lte\_signal\_info\_t lte\_signal\_info;

nwy\_nw\_nr\_signal\_info\_t nr\_signal\_info;

nwy\_nw\_tdscdma\_signal\_info\_t tdscdma\_signal\_info;

}nwy\_nw\_rat\_signal\_info\_u;

typedef struct {

nwy\_nw\_rat\_type\_e rat;

nwy\_nw\_rat\_signal\_info\_u rat\_signal\_info;

}nwy\_nw\_signal\_info\_t;

typedef struct {

nwy\_nw\_rat\_type\_e rat\_type; //Radio technology; see #NWY\_nw\_radio\_tech\_t

nwy\_nw\_service\_type\_e regs\_state; //Registration state

nwy\_nw\_roam\_state\_e roam\_state;

}nwy\_nw\_common\_regs\_type\_t;

typedef struct {

uint8 mm\_regs\_reject\_info\_valid; //mm:Mobile Management

uint8 mm\_regs\_rej\_cause;

uint8 sm\_regs\_reject\_info\_valid; //sm:Session Management

uint8 sm\_regs\_rej\_cause;

int rej\_srv\_domain;

}nwy\_nw\_res\_reject\_info\_t;

/\*The core network contains two parts, the CS domain, and the PS domain \*/

typedef struct {

uint8 cs\_regs\_valid;

nwy\_nw\_common\_regs\_type\_t cs\_regs;/\* CS: 'Circuit Switching', handles voice calls in 2.3G net \*/

uint8 ps\_regs\_valid;

nwy\_nw\_common\_regs\_type\_t ps\_regs;/\* PS: 'Packet Switching', handles data sessions in 2.3.4G net \*/

nwy\_nw\_res\_reject\_info\_t reject\_info;

}nwy\_nw\_regstatus\_t;

typedef struct {

uint16 mcc; /\*A 16-bit integer representation of MCC. Range: 0 to 999.\*/

uint16 mnc; /\*A 16-bit integer representation of MNC. Range: 0 to 999.\*/

uint8 mnc\_includes\_pcs\_digit;s

}nwy\_nw\_plmn\_id\_type\_t;

typedef struct {

nwy\_cdma\_pilot\_e pilot\_type;

uint16 sid;

/\*System ID. Range: 0 to 32767. \*/

uint16 nid;

/\* Network ID. Range: 0 to 65535.\*/

uint16 base\_id;

/\* Base station ID. \*/

nwy\_nw\_plmn\_id\_type\_t plmn;

nwy\_band\_cdma\_class\_e band;

/\* Active band of the current system. \*/

int32 channel;

/\* Active channel of the current system. \*/

uint16 pilot\_pn;

/\*Pilot PN sequence offset index. Range: 0 to 511.\*/

uint16 pilot\_strength;

/\*Strength of the pilot (in dB). Range: 0 to 64\*/

uint32 base\_lat;

/\*Latitude of the current base station in units of 0.25 sec.\*/

uint32 base\_long;

/\*Longitude of the current base station in units of 0.25 sec.\*/

uint64 time\_stamp;

/\*Time (in milliseconds) from the start of GPS time when the measurement was taken.\*/

int32 TxPower; /\* Tx power value in 1/10 dBm resolution. \*/

int32 RxPower; /\* Rx power value in 1/10 dBm resolution. \*/

}nwy\_cdma\_bs\_info\_t ;

typedef struct {

uint32 bs\_len;

nwy\_cdma\_bs\_info\_t bs[NWY\_NW\_CDMA\_POSITION\_MAX];

}nwy\_cdma\_hdr\_cell\_info\_t;

typedef struct {

nwy\_nw\_plmn\_id\_type\_t plmn;

uint32 cell\_id;

uint32 timing\_advance;

uint16 lac; /\* range 0..65535, see 24.008 10.5.1.3 \*/

nwy\_band\_gsm\_class\_e band;

uint16 arfcn; /\*range 0..1023 \*/

uint8 bsic; /\* range 0..63, see 23.003 4.3.2 \*/

uint8 egprs\_support; //1: is edge

int rssi;

int32 TxPower; /\* Tx power value in 1/10 dBm resolution. \*/

int32 RxPower; /\* Rx power value in 1/10 dBm resolution. \*/

}nwy\_gsm\_serv\_cell\_info\_t;

typedef struct {

nwy\_nw\_plmn\_id\_type\_t plmn;

uint8 bsic; /\* range 0..63, see 23.003 4.3.2 \*/

uint16 lac; /\* range 0..65535, see 24.008 10.5.1.3 \*/

uint16 cell\_id;

uint16 arfcn; /\*range 0..1023 \*/

int16 rssi;

uint16 RxLev; //RX\_LEV range 0..63, See 45.008 sect 8.1.4, default: 0

}nwy\_nbr\_gsm\_cell\_info\_t;

typedef struct {

nwy\_nw\_plmn\_id\_type\_t plmn;

uint16 lac;

uint32 cellid;

uint16 uArfcn; /\*Absolute radio frequency channel number \*/

uint16 psc\_cellParameterId; /\*Primary scrambling code\*/

uint16 set; /\*Intrafrequency cell type. \*/

int rscp;

int ecio;

int16 s\_rxlev; /\*Suitable receive level. \*/

}nwy\_nbr\_wcdma\_cell\_info\_t;

typedef struct {

nwy\_nw\_plmn\_id\_type\_t plmn;

uint16 tac;

uint32 cellid;

uint16 pci;

uint32 earfcn;

int rsrp;

int rsrq;

int rssi;

int sinr;

int16 srxlev; /\*Suitable receive level. \*/

}nwy\_nbr\_lte\_cell\_info\_t;

typedef struct {

nwy\_nw\_plmn\_id\_type\_t plmn;

uint16 tac;

uint64 nr\_cellid;

uint32 nr5g\_arfcn; /\*\*< Raster frequency of the serving cell. \*/

uint16 pci; /\*\*< Physical cell ID. \*/

int rsrp;

int rsrq;

int rssi;

int sinr;

int16 srxlev; /\*Suitable receive level. \*/

}nwy\_nbr\_nr\_cell\_info\_t;

typedef struct {

uint16 uarfcn; /\*TD-SCDMA layer frequency. \*/

uint16 lac;

uint32 cellid;

uint16 cpid; /\* TDS cell ID. \*/

uint16 cell\_position; /\*TDS cell position. \*/

int16 rscp;

int16 s\_rxlev; /\*Suitable receive level. \*/

} nwy\_nbr\_tdscdma\_cell\_info\_t;

typedef struct {

nwy\_gsm\_serv\_cell\_info\_t serv\_cell;

uint32 gsm\_nbr\_gsm\_cell\_len;

nwy\_nbr\_gsm\_cell\_info\_t nbr\_gsm\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

uint32 gsm\_nbr\_wcdma\_cell\_len;

nwy\_nbr\_wcdma\_cell\_info\_t nbr\_wcdma\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

uint32 gsm\_nbr\_tdscdma\_cell\_len;

nwy\_nbr\_tdscdma\_cell\_info\_t nbr\_tdscdma\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

uint32 gsm\_nbr\_lte\_cell\_len;

nwy\_nbr\_lte\_cell\_info\_t nbr\_lte\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

uint32 gsm\_nbr\_nr\_cell\_len;

nwy\_nbr\_nr\_cell\_info\_t nbr\_nr\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

}nwy\_gsm\_cell\_info\_t;

typedef struct {

nwy\_nw\_plmn\_id\_type\_t plmn;

uint16 cell\_id; /\*Cell identity; as per 3G TS 25.331, 10.3.2.2 (28 bits) \*/

uint16 lac; /\*Location area code \*/

uint8 is\_rac\_present; /\*Boolean indicated is RAC is present\*/

uint8 rac;

uint32 ura; /\*URA identity \*/

uint16 uarfcn; /\*Absolute radio frequency channel number \*/

nwy\_band\_wcdma\_class\_e band;

uint16 psc; /\*Primary scrambling code. \*/

uint8 HSDPASupport; /\*TRUE - serving cell supports HSDPA; FALSE - other. \*/

uint8 HSUPASupport; /\*TRUE - serving cell supports HSUPA; FALSE - other. \sa CCI API Ref Manual \*/

int rscp;

int ecio;

int rssi;

int32 TxPower; /\* Tx power value in 1/10 dBm resolution. \*/

int32 RxPower; /\* Rx power value in 1/10 dBm resolution. \*/

} nwy\_wcdma\_serv\_cell\_info\_t;

typedef struct {

nwy\_wcdma\_serv\_cell\_info\_t serv\_cell;

uint8 wcdma\_nbr\_cell\_len;

nwy\_nbr\_wcdma\_cell\_info\_t wcdma\_nbr\_wcdma\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

uint32 wcdma\_gsm\_nbr\_len;

nwy\_nbr\_gsm\_cell\_info\_t wcdma\_nbr\_gsm\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

uint8 wcdma\_nbr\_lte\_len;

nwy\_nbr\_lte\_cell\_info\_t wcdma\_nbr\_lte\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

uint32 wcdma\_nbr\_nr\_len;

nwy\_nbr\_nr\_cell\_info\_t wcdma\_nbr\_nr\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

}nwy\_wcdma\_cell\_info\_t;

typedef struct {

nwy\_nw\_plmn\_id\_type\_t plmn;

uint32 cell\_id; /\*Cell ID (0xFFFFFFFF indicates cell ID information is not present).\*/

uint16 lac; /\*Location area code. (This field is ignored when cell\_id is not present.)\*/

nwy\_band\_tdscdma\_class\_e band;

uint16 uarfcn; /\*Absolute RF channel number.\*/

uint8 cell\_parameter\_id; /\*Cell parameter ID.\*/

uint8 pathloss; /\*Path loss in units of 1 dB.\*/

uint32 timing\_advance;

int rscp;

}nwy\_tdscdma\_serv\_cell\_info\_t;

typedef struct {

nwy\_nw\_plmn\_id\_type\_t plmn;

uint16 uarfcn; /\*Absolute RF channel number.\*/

uint8 cell\_parameter\_id; /\*Cell parameter ID.\*/

int rscp;

}nwy\_tdscdma\_nbr\_tdscdma\_cell\_info\_t;

typedef struct {

nwy\_tdscdma\_serv\_cell\_info\_t serv\_cell;

uint8 tdscdma\_nbr\_cell\_len;

nwy\_tdscdma\_nbr\_tdscdma\_cell\_info\_t tdscdma\_nbr\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

}nwy\_tdscdma\_cell\_info\_t;

typedef struct {

nwy\_nw\_plmn\_id\_type\_t plmn;

uint16 tac;

uint16 earfcn; /\* EARFCN of the serving cell. Range: 0 to 65535. \*/

uint32 cell\_id;

uint16 pci; /\*Range: 0 to 503.\*/

uint16 band;

uint16 dlBandwidth;

uint8 isTdd; /\* 1--Tdd, 0--Fdd \*/

uint16 cqi; /\*Value range: 0~15, invalid(0xFFFF). \*/

int rsrp;

int rsrq;

int rssi;

int sinr;

int32 TxPower; /\* Tx power value in 1/10 dBm resolution. \*/

int32 RxPower; /\* Rx power value in 1/10 dBm resolution. \*/

}nwy\_lte\_serv\_cell\_info\_t;

typedef struct {

nwy\_lte\_serv\_cell\_info\_t serv\_cell;

uint8 lte\_nbr\_len;

nwy\_nbr\_lte\_cell\_info\_t lte\_nbr\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

uint8 lte\_nbr\_wcdma\_len;

nwy\_nbr\_wcdma\_cell\_info\_t lte\_nbr\_wcdma\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

uint8 lte\_nbr\_gsm\_len;

nwy\_nbr\_gsm\_cell\_info\_t lte\_nbr\_gsm\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

uint8 lte\_nbr\_cdma\_len;

nwy\_cdma\_bs\_info\_t lte\_nbr\_cdma\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

uint8 lte\_nbr\_cdma\_hdr\_len;

nwy\_cdma\_bs\_info\_t lte\_nbr\_cdma\_hdr\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

uint8 lte\_nbr\_nr\_len;

nwy\_nbr\_nr\_cell\_info\_t lte\_nbr\_nr\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

}nwy\_lte\_cell\_info\_t;

typedef struct {

nwy\_nw\_plmn\_id\_type\_t plmn;

uint8 tac[3]; /\*R5G Serving Cell Information\*/

uint64 cell\_id;

uint32 g\_node\_id; // same as LTE eNodeB id,0 means invaild

uint32 arfcn;

nwy\_band\_nr5g\_class\_e band;

nwy\_bandwidth\_type\_e dlBandwidth;

uint16 pci; /\*Physical cell ID. \*/

int rsrp;

int rsrq;

int rssi;

int sinr;

int32 TxPower; /\* Tx power value in 1/10 dBm resolution. \*/

int32 RxPower; /\* Rx power value in 1/10 dBm resolution. \*/

}nwy\_nr\_serv\_cell\_info\_t;

typedef struct {

nwy\_nr\_serv\_cell\_info\_t serv\_cell;

uint8 nr\_nbr\_intra\_len;

nwy\_nbr\_nr\_cell\_info\_t nr\_nbr\_intra\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

uint8 nr\_nbr\_inter\_len;

nwy\_nbr\_nr\_cell\_info\_t nr\_nbr\_inter\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

uint8 nr\_nbr\_lte\_len;

nwy\_nbr\_lte\_cell\_info\_t nr\_nbr\_lte\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

uint8 nr\_nbr\_wcdma\_len;

nwy\_nbr\_wcdma\_cell\_info\_t nr\_nbr\_wcdma\_cell[NWY\_NW\_NBR\_CELLS\_MAX\_NUM];

}nwy\_nr5g\_cell\_info\_t;

typedef struct

{

nwy\_nw\_rat\_type\_e rat;

union{

nwy\_cdma\_hdr\_cell\_info\_t cdma\_hdr\_cell;

nwy\_gsm\_cell\_info\_t gsm\_cell;

nwy\_wcdma\_cell\_info\_t wcdma\_cell;

nwy\_tdscdma\_cell\_info\_t tds\_cell;

nwy\_lte\_cell\_info\_t lte\_cell;

nwy\_nr5g\_cell\_info\_t nr5g\_cell;

}cell\_info;

}nwy\_nw\_cellinfo\_t;

typedef struct {

nwy\_nw\_status\_e net\_status;

nwy\_nw\_mode\_mask\_e net\_rat;

nwy\_nw\_operator\_t net\_name;

}nwy\_nw\_net\_scan\_item\_t;

typedef struct {

int result;

int net\_num;

nwy\_nw\_net\_scan\_item\_t net\_list[NWY\_NW\_SCAN\_LIST\_MAX\_NUM];

}nwy\_nw\_scan\_rsp\_list\_t;

typedef struct {

char mcc[NWY\_NW\_MNC\_MCC\_LEN + 1];

char mnc[NWY\_NW\_MNC\_MCC\_LEN + 1];

nwy\_nw\_mode\_mask\_e net\_rat;

nwy\_nw\_select\_mode\_e select\_mode;

}nwy\_nw\_select\_param\_t;

typedef struct{

int8 onoff;

int8 timer;

} nwy\_nw\_config\_netauto\_t;

typedef struct{

uint16 support\_rat;

uint16 gsm\_band;

uint32 wcdma\_band;

uint64 lte\_band[2];//lte\_band[0] for B1-B64, lte\_band[1] for B65-B128

uint64 nr\_band[5]; //nr\_band[0]for n1-n64;nr\_band[2] for n65-n128 ...

} nwy\_nw\_cfg\_net\_capacity\_t;

stypedef union {

nwy\_nw\_config\_netauto\_t netauto;

nwy\_nw\_cfg\_uemode\_type\_e uemode;

nwy\_nw\_cfg\_switch\_type\_e dataonly;

nwy\_nw\_cfg\_switch\_type\_e antaux;

nwy\_nw\_cfg\_net\_capacity\_t net\_capacity;

nwy\_nw\_cfg\_switch\_type\_e nitz\_onoff;

} nwy\_nw\_config\_info\_u;

# APIs

此接口函数定义位于nwy\_network.h。

用于实现网络注册相关功能，包括网络注册状态、网络制式等信息的获取及设置网络制式。也可通过这些函数接口进行回调函数的注册来监控网络状态的变化。

## nwy\_nw\_mode\_get

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_mode\_get(nwy\_sim\_id\_e sim\_id,int \*p\_mode) |
| 描述 | 获取当前锁定的网络制式。 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID。  p\_mode: 输出参数，网络制式，支持多个NWY\_NW\_MODE\_MASK配置多个网络制式。  输出的16进制数，转化为2进制，每一bit位代表一种网络制式。  例如：  0x4 （0000 0100） 表示当前仅锁定GSM。  0x10（0001 0000） 表示当前仅锁定LTE。  0x14（0001 0010） 表示锁定GSM+LTE。  NWY\_NW\_MODE\_MASK格式详见nwy\_network.h定义。 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |
| 注意事项 | * 该返回值不表示当前网络的注册状态 * 当对应芯片平台仅支持一种网络制式时，始终返回AUTO。 |

## nwy\_nw\_mode\_set

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_mode\_set(nwy\_sim\_id\_e sim\_id,int mode) |
| 描述 | 设置当前网络制式。 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID。  mode: 输入参数，网络制式，支持多个NWY\_NW\_MODE\_MASK配置多个网络制式。  具体说明见nwy\_nw\_mode\_get。  NWY\_NW\_MODE\_MASK格式详见nwy\_network.h定义。 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |
| 注意事项 | 当对应芯片平台仅支持一种网络制式时，底层判断入参有效后，直接返回OK。 |

## nwy\_nw\_lock\_get

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_lock\_get(nwy\_sim\_id\_e simid, nwy\_nw\_lock\_t\* p\_lock\_info) |
| 描述 | 查询锁定到单个或者多个频段、频点、小区。 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID  lock\_info: 输出参数，锁定信息， 详见nwy\_network.h定义 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |

## nwy\_nw\_lock\_set

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_lock\_set(nwy\_sim\_id\_e simid, nwy\_nw\_lock\_t\* p\_lock\_info) |
| 描述 | 锁定到单个或者多个频段、频点、小区。 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID  lock\_info: 输入参数，锁定信息， 详见nwy\_network.h定义 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |

## nwy\_nw\_priband\_get

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_priband\_get(nwy\_sim\_id\_e simid, nwy\_nw\_priband\_t\* p\_priband\_info) |
| 描述 | 查询优先选择band信息。 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID。  priband\_info: 输出参数，优先选择band信息，详见nwy\_network.h定义。 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |

## nwy\_nw\_priband\_set

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_priband\_set(nwy\_sim\_id\_e simid, nwy\_nw\_priband\_t\* p\_prioband\_info) |
| 描述 | 设置优先选择band信息，当在优先选择band驻网失败后，将尝试在全部band能力下驻网。 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID。  priband\_info: 输入参数，优先选择band信息， band的优先顺序在nwy\_nw\_priband\_t结构体的band字段降序排序。详见nwy\_network.h定义。 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |

## nwy\_nw\_ims\_get

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_ims\_get(nwy\_sim\_id\_e simid, uint8\* p\_ims\_state) |
| 描述 | 查询ims配置信息。 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID  ims\_state: 输出参数，ims功能配置状态， 详见nwy\_network.h定义 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |
| 注意事项 | 当芯片平台不支持IMS或者编译选项未打开时，底层判断入参有效后，直接返回-10；（NWY\_GEN\_E\_PLAT\_NOT\_SUPPORT） |

## nwy\_nw\_ims\_set

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_ims\_set(nwy\_sim\_id\_e simid, uint8 ims\_state) |
| 描述 | 设置ims配置信息。 |
| 参数说明 | sim\_id：输入参数，，指定要操作SIM卡的ID  ims\_state: 输入参数，ims功能配置状态， 详见nwy\_network.h定义 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |
| 注意事项 | 当芯片平台不支持IMS或者编译选项未打开时，底层判断入参有效后，直接返回-10；（NWY\_GEN\_E\_PLAT\_NOT\_SUPPORT） |

## nwy\_nw\_radio\_get

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_radio\_get (int \*p\_radio\_mode) |
| 描述 | 获取当前Radio状态 |
| 参数说明 | radio\_mode：输出参数  0：最小功能（turn off radio and SIM power）  1：全功能（默认）  4：关闭模块的发送和接收射频电路（飞行模式） |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |

## nwy\_nw\_radio\_set

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_radio\_set (int radio\_mode) |
| 描述 | 设置当前Radio状态 |
| 参数说明 | radio\_mode：输入参数输出  0：最小功能（turn off radio and SIM power）  1：全功能（默认）  4：关闭模块的发送和接收射频电路（飞行模式） |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |
| 注意事项 | 某些芯片平台对于状态0.4没有做区分，只会关闭射频，并不会使sim卡掉电。 |

## nwy\_nw\_edrx\_get

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_edrx\_get (nwy\_sim\_id\_e simid, nwy\_nw\_edrx\_info\_t \* p\_edrx\_info) |
| 描述 | 获取edrx配置信息 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID  edrx\_info：输出参数，edrx配置信息， 详见nwy\_network.h定义 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |

## nwy\_nw\_edrx\_set

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_edrx\_set (nwy\_sim\_id\_e simid, nwy\_nw\_edrx\_info\_t \* p\_edrx\_info) |
| 描述 | 设置edrx配置信息 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID  edrx\_info: 输入参数，edrx配置信息， 详见nwy\_network.h定义 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |

## nwy\_nw\_psm\_get

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_psm\_get (nwy\_sim\_id\_e simid, nwy\_nw\_psm\_info\_t \* p\_psm\_info) |
| 描述 | 设置psm配置信息 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID  psm\_info: 输出参数，psm配置信息， 详见nwy\_network.h定义 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |

## nwy\_nw\_psm\_set

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_psm\_set (int fun) |
| 描述 | 设置psm配置信息 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID  psm\_info: 输入参数，psm配置信息， 详见nwy\_network.h定义 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |

## nwy\_nw\_config\_get

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_config\_get (nwy\_sim\_id\_e simid, nwy\_nw\_config\_type\_t config\_type, nwy\_nw\_config\_info\_t \* p\_config\_info) |
| 描述 | 查询网络相关配置信息 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID  config\_type: 输入参数，配置类型  config\_info：输出参数，获取的配置信息  详见nwy\_network.h定义 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |

## nwy\_nw\_config\_set

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_config\_get (nwy\_sim\_id\_e simid, nwy\_nw\_config\_type\_t config\_type, nwy\_nw\_config\_info\_t \* p\_config\_info) |
| 描述 | 查询网络相关配置信息 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID  config\_type：输入参数，配置类型  config\_info：输入参数，获取的配置信息  详见nwy\_network.h定义 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |

## nwy\_nw\_operator\_get

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_operator\_get (nwy\_sim\_id\_e sim\_id, nwy\_nw\_operator\_t \*p\_opt\_name) |
| 描述 | 获取当前注册网络的运营商名称信息，包括Long EONS（Enhanced Operator Name String），Short EONS，MCC，MNC，可以参考数据结构nwy\_nw\_operator\_t的定义。 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID  opt\_name：输出参数，返回的运营商名称 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |

## nwy\_nw\_csq\_get

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_csq\_get (nwy\_sim\_id\_e sim\_id, uint8\_t \*p\_csq\_val) |
| 描述 | 获取当前网络的信号强度。 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID。  csq\_val：输出参数，当前网络的信号强度，其值与AT命令CSQ保持一致。（取值范围为0-31，无效值99） |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |

## nwy\_nw\_signal\_get

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_signal\_get (nwy\_sim\_id\_e simid, nwy\_nw\_signal\_info\_t\* p\_signal\_info) |
| 描述 | 获取当前网络的信号相关信息。 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID  signal\_info: 输出参数，当前网络的信号强度，详见nwy\_network.h定义 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |

## nwy\_nw\_regstatus\_get

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_regstatus\_get (nwy\_sim\_id\_e simid, nwy\_nw\_regstatus\_t \*p\_regs\_info) |
| 描述 | 获取当前网络注册信息。 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID  p\_regs\_info: 输出参数，网络注册信息，详见nwy\_network.h定义 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |

## nwy\_nw\_cellinfo\_get

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_cellinfo\_get(nwy\_sim\_id\_e simid, nwy\_nw\_cellinfo\_mode\_e cellinfo\_mode, nwy\_nw\_cellinfo\_t \*p\_cellinfo\_info) |
| 描述 | 获取当前网络的主邻区信息。 |
| 参数说明 | 一个阻塞型函数，获取主小区可以快速返回，最多1s；获取所有小区信息最多阻塞5s。  sim\_id：输入参数，指定要操作SIM卡的ID  cellinfo\_mode：控制获取主小区信息，还是全部信息ss  p\_cellinfo\_info: 输出参数，当前网络的主邻区信息，详见nwy\_network.h定义 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |
| 注意事项 | * 对于大部分平台，获取小区信息，是从modem侧获取正常网络扫描行为结束后存储在本地的小区信息，并不会触发网络扫描行为。 * 刚开机时，由于刚开始进行网络扫描，获取到的信息比较少，属于正常情况。 |

## nwy\_nw\_scan

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_scan(nwy\_sim\_id\_e simid, nwy\_nw\_scan\_cb\_func scan\_cb) |
| 描述 | 手动扫描网络，异步函数，扫描结果通过回调函数返回。 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID  scan\_cb：回调函数，扫描完成后会触发该函数返回扫描结果 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |

## nwy\_nw\_manual\_select

|  |  |
| --- | --- |
| 格式 | nwy\_error\_e nwy\_nw\_manual\_select(nwy\_sim\_id\_e simid, nwy\_nw\_select\_param\_t \* p\_net\_select) |
| 描述 | 手动注册网络，根据nwy\_nw\_scan函数扫描获得的网络列表选取某一个网络进行注册。 |
| 参数说明 | sim\_id：输入参数，指定要操作SIM卡的ID  net\_select:选择注册网络的参数，包括网络的PLMN和无线接入技术 |
| 返回值 | 成功：NWY\_SUCCESS  失败：其他枚举值 |