



深圳市海凌科电子有限公司

HLK-N10 平台扩展 AT 命令

使用前说明

本书采用的约定及标志如下。

1. 符号约定

带尖括号“< >”表示键名、按钮名以及操作员从终端输入的信息；带方括号“[]”表示人机界面、菜单条、数据表和字段名等，多级菜单用“→”隔开。如 [文件→新建→文件夹] 多级菜单表示 [文件] 菜单下的 [新建] 子菜单下的 [文件夹] 菜单项。

2. 键盘操作约定

格式	意义
加尖括号的字符	表示键名、按钮名。如 < Enter >、< Tab >、< Backspace >、< a > 等分别表示回车、制表、退格、小写字母 a
< 键 1+键 2 >	表示在键盘上同时按下几个键。如 < Ctrl+Alt+A > 表示同时按下“Ctrl”、“Alt”、“A”这三个键
< 键 1, 键 2 >	表示先按第一键，释放，再按第二键。如 < Alt, F > 表示先按 < Alt > 键，释放后，紧接着再按 < F > 键

3. 鼠标操作约定

格式	意义
单击	快速按下并释放鼠标的左键
双击	连续两次快速按下并释放鼠标的左键
右击	快速按下并释放鼠标的右键
拖动	按住鼠标的左键不放，移动鼠标

4. 标志

小心、注意、警告、危险前使用符号“”。

说明、提示、小窍门前使用符号“”。

目录

概要.....	1
1 平台自定义扩展 AT.....	1
1.1 快速深睡+FASTOFF.....	1
1.2 快速断电+CPOF.....	1
1.3 RESET 按键生效时长+RESETCTL.....	1
1.4 NV 动态配置+NV.....	2
1.5 查询版本相关信息.....	3
1.6 测试命令+TEST.....	4
1.7 设置 ERROR 上报模式+CMEE.....	5
1.8 设备软重启+RB/NRB.....	6
1.9 设备冷重启+COLDREB.....	6
1.10 设备恢复出厂设置+RESET.....	7
1.11 人为主动断言+ASSERT.....	7
1.12 开关硬看门狗+WDT.....	7
1.13 设置 SIM 卡电压值+SIMVCC.....	8
1.14 AT 串口设置+UARTSET.....	8
1.15 省电锁+WORKLOCK.....	9
1.16 用户快速链接释放+XYRAI.....	10
1.17 动态开关+STANDBY.....	10
1.18 省电模式主动上报+POWERDOWN(仅用于带外部 MCU 产品).....	11
1.19 上电完成主动上报+POWERON (仅用于带外部 MCU 产品).....	11
1.20 断电时间补偿+OFFTIME(仅用于带外部 MCU 产品).....	12
1.21 低电压告警+LOWVBAT.....	12
1.22 电压查询+VBAT.....	13
1.23 SIM 卡类型主动上报+UICCTYPE.....	13
1.24 内存动态查看+NUESTATS.....	13
1.25 FOTA 升级开关+ FOTACTR.....	14
1.26 透传数据 ATD*98/ATD*99(仅用于内部调试).....	14
2 外部适配扩展 AT.....	16
2.1 切换深睡唤醒主动上报+NPSMR.....	16
2.2 世界时间配置+CCLK.....	17
2.3 世界时间获取方式+NITZ.....	17
2.4 AT 串口设置+NATSPEED.....	18
2.5 设置日志等级+NLOGLEVEL.....	19
2.6 本地 FOTA 升级 +NFWUPD.....	19
3 平台 SOCKET 扩展命令.....	21
3.1 DNS 设置+XDNSCFG.....	21
3.2 DNS 解析+XDNS.....	21
3.3 数据模式设置+XDTMODE.....	22
3.4 开启服务+XSOPEN.....	22
3.5 链接状态上报+XSSTATE.....	23
3.6 发送数据+XSSEND.....	23
3.7 查询发送状态+SEQUENCE.....	25

3.8 下行数据主动上报+XSNMI.....	25
3.9 读取下行数据+XSREAD.....	26
3.10 关闭服务+XSCLOSE.....	27
3.11 典型示例.....	28
3.12 注意事项.....	28
4 SOCKET 扩展命令.....	29
4.1 开启服务+NSOCR.....	29
4.2 TCP 开启服务+NSOCO.....	29
4.3 UDP 发送数据+NSOST.....	30
4.4 TCP 发送数据+NSOSD.....	31
4.5 上行报文发送状态主动上报+SEQUENCE.....	32
4.6 查询 SOCKET 链接状态 +NSOSTATUS.....	33
4.7 下行数据的上报模式设置+NSONMI.....	33
4.8 下行数据的上报+NSONMI.....	34
4.9 接收数据+NSORF.....	35
4.10 关闭服务+NSOCL.....	36
4.11 注意事项.....	36
4.12 典型示例.....	36
5 天翼云扩展 AT.....	38
5.1 CDP 设置与查询+NCDP.....	38
5.2 注册去注册+QLWSREGIND.....	39
5.3 注册状态主动上报+QLWEVTIND.....	39
5.4 注册模式+QREGSWT.....	40
5.5 发送信息+NMGS.....	40
5.6 发送信息+NMGSEXT.....	41
5.7 发送信息指示+NSMI.....	41
5.8 接收信息+NMGR.....	42
5.9 新信息指示+NNMI.....	42
5.10 查询已发送信息+NQMGS.....	43
5.11 查询已接收信息+NQMGR.....	43
5.12 注意事项.....	44
5.13 典型示例.....	44
6 ONENET 扩展 AT.....	45
6.1 创建 ONENET 通信套件实例+MIPLCREATE.....	45
6.2 删除 ONENET 通信套件实例+MIPLDELETE.....	46
6.3 增加 LwM2M 实例+MIPLADDOBJ.....	46
6.4 删除 LwM2M 实例+MIPLDELOBJ.....	47
6.5 发起注册请求+MIPLOPEN.....	47
6.6 发送注销请求+MIPLCLOSE.....	48
6.7 DISCOVER 请求的响应+MIPLDISCOVERRSP.....	48
6.8 主动上报 DISCOVER 请求+MIPLDISCOVER.....	49
6.9 OBSERVE 请求的响应+MIPLOBSERVERSP.....	49
6.10 主动上报 OBSERVE 请求+MIPLOBSEVE.....	50
6.11 READ 请求的响应+MIPLREADRSP.....	51
6.12 主动上报 READ 请求+MIPLREAD.....	52
6.13 WRITE 请求的响应+MIPLWRITERSP.....	53

6.14	主动上报 WRITE 请求+MIPLWRITE.....	53
6.15	EXECUTE 请求的响应+MIPLXECUTERSP.....	54
6.16	主动上报 EXECUTE 请求+MIPLXECUTE.....	55
6.17	WRITE 属性请求的响应+MIPLPARAMETERRSP.....	55
6.18	主动上报 WRITE 属性请求+MIPLPARAMETER.....	56
6.19	NOTIFY 数据到 ONENET 云端+MIPLNOTIFY.....	56
6.20	发送更新请求+MIPLUPDATE.....	58
6.21	查询当前 ONENET 通信套件版本+MIPLVER.....	58
6.22	主动上报 EVENT 事件通知+MIPLXEVENT.....	59
6.23	注意事项.....	59
7	平台工具类扩展 AT.....	60
7.1	PING 包指令+NPING.....	60
7.2	速率灌包测试+XYPERF.....	60
7.3	SOCKET 的灌包+XDSEND.....	61
7.4	用户 DEMO 演示命令+TEST.....	62
7.5	内存信息获取命令+MEMSTATS.....	63
8	ERR 错误码.....	65
8.1	GENERAL ERRORS (27.007).....	65
8.2	GENERAL ERRORS (27.005).....	65
8.3	XYENHANCED ERRORS (HLK-N10).....	66
	版本历史.....	67

概要

本文描述的是海凌科的 NBIOT 芯片平台提供的扩展 AT 命令，以满足大多数客户的共性需求。对于客户自行定义的扩展 AT 命令，自行进行按照示例进行扩展开发即可。

1 平台自定义扩展 AT

1.1 快速深睡+FASTOFF

语法结构

命令	响应
AT+FASTOFF=<abnormal off>	OK
	+CME ERROR:<err>

接口说明

该命令用于控制芯片毫秒级进入深睡 (deepsleep) 模式，流程不可逆转，必须收到+POWERDOWN 主动上报后，芯片才能再次响应用户事件。

该命令流程不执行 PSM 流程，所以芯片再次上电时，NB 协议栈执行 Attach 流程。对于模组类用户，如果不关心 PSM 流程，底板 MCU 可以发送 AT+FASTOFF 来控制芯片进入快速深睡。

需要注意的是，支持断电补偿 OFFTIME 机制的深睡，不能使用 AT+FASTOFF/CPOF，只能使用 +WORKLOCK 指令进入深睡。

取值说明

<abnormal off>:

- 0: 默认值，保存用户数据和平台数据到 Flash 中，并立即进入深睡，支持深睡唤醒；
- 1: 直接断电，不保存NV到flash中，立即进入深睡，支持深睡唤醒。目前仅限于功耗测试中，正常开发时不能使用该参数。
- 2: 内部保存平台和用户数据到 Flash 后，关闭时钟，但不进入 DEEPSLEEP 深睡，不支持深睡唤醒。芯片用户收到 "+POWERDOWN:" 主动上报后，必须执行断电操作，避免因用户底板带有大电容造成的深睡后放电过慢，再次上电后状态异常的问题。

1.2 快速断电+CPOF

与 AT+FASTOFF 功能完全一致，具体使用请参考+FASTOFF 指令。

1.3 RESET 按键生效时长+RESETCTL

语法结构

命令	响应
AT+RESETCTL=<mode>	OK
	+CME ERROR:<err>
AT+RESETCTL?	+RESETCTL:<mode>
	OK

接口说明

RESET 按键触发重启的生效时长配置。重启后，NB 的工作态 NV 仍然有效。

取值说明

<mode>:

0, 表示按键超过 20 毫秒复位信号生效; 默认值。

1, 表示按键超过 6 秒复位信号生效, 小于 6s 为唤醒信号。

1.4 NV 动态配置+NV

语法结构

命令	响应
AT+NV=<action>[,<param>[,<val>]]	OK +CME ERROR:<err>

接口说明

用于设置/查询各种出厂 NV 参数, 仅用于调试, 不得把 AT+NV 作为正常的 AT 代码进行开发使用。为了方便用户进行功耗测试, 减少输入的 AT 命令, “AT+NV=SET,POWERTEST,1” 用于仪表的功耗测试, “AT+NV=SET,POWERTEST,2” 用于 SDK 版本的功耗测试。由于功耗测试期间, 会关闭 log 输出、关闭 URC 输出, 所以仅能通过 AT 请求查询一些信息。

取值说明

<action>: 对 NV 参数的操作, 取值如下:

取值	含义
SET	设置某出厂NV参数
GET	读取某出厂NV参数
SAVE	保存出厂NV到flash中
?	查询支持哪些出厂NV参数的动态可配置

<param>: NV 参数, 可查看《海凌科 HLK-N10 平台开发指南》中的“出厂 NV 重要参数”章节。

为了方便用户记忆, 额外添加了一些非出厂 NV 的参数查询与设置。

取值	含义
MEM	查询ARM和DSP核运行期间堆内存剩余空间的最小值, 以协助决定是否需要开大堆空间, 防止内存申请不到
ARMSTACK	查询ARM核所有线程的栈空间使用情况, 以协助决定是否在栈空间溢出风险
FACTORY	查询用户配置的关键出厂NV的具体值
VER	查询内外部所有版本号, 以快速找到对应的源文件
RATETEST	仅用于下行灌包时, 设置不透传到M3核, 以解决多余的上限ICMP报文干扰问题
DEMOTEST	用于设置启用哪些DEMO任务, 位图方式表示
PRODUCTVER	用于设置product version信息
MODULVER	用于设置module version信息。注: 格式需参照XXX-XXXXX
HARDVER	用于设置hardware version信息
VERSIONEXT	用于设置external version信息

<val>: 可选参数, 无固定取值, 用户可设置为空

典型示例

请求: AT+NV=SET,IPALIVE,1

响应: OK

请求: AT+NV=SAVE,1,1,1 //保存 dsp_varnv_save,dsp_invarnv_save,user_nv_save 对应的 nv

响应: REBOOTING

请求: AT+NV=GET, MEM

响应: ARM:27888;DSP:71184

响应: OK

请求: AT+NV=GET, ARMSTACK

响应: Swt_TaskStackSize:720

Swt_TaskStackRemaing:164

Swt_TaskPeakUsedSize:164

...

OK

请求: AT+NV=GET, FACTORY

响应: VERTYPE:2, WORKMODE:0, KEEPALIVE:0, STANDBY:1, OFFTIME:1, POWEROFF:1, WFI:1,

OK

1.5 查询版本相关信息

语法结构

命令	响应
AT+CGMR	<Software Version:软件版本号> OK
	+CME ERROR:<err>
AT+CGMI	<厂商代码> OK
	+CME ERROR:<err>
AT+CGMM	<模组版本> OK
	+CME ERROR:<err>
AT+NV=GET,HARDVER	<Hardware Version:硬件版本号> OK
	+CME ERROR:<err>
AT+NV=GET,EXTVER	<Software Version:软件版本号> OK
	+CME ERROR:<err>
AT+SWVER=<verval>	OK
	+CME ERROR:<err>

AT+SWVER	<Software Version:软件版本号> OK
	+CME ERROR:<err>
AT+HVER=<verval>	OK
	+CME ERROR:<err>
AT+HVER	<Hardware Version:硬件版本号> OK
	+CME ERROR:<err>

接口说明

查询软件版本号及硬件版本号前缀符合运营商要求，前缀格式如下“典型示例”。

取值说明

<verval>：版本信息

软件版本信息可设置 28 字节可打印字符，尾部为结束符“\0”；

硬件版本信息可设置 20 字节可打印字符，尾部为结束符“\0”；

<err>：错误码，参看 err 章节

典型示例

请求：AT+NV=GET,EXTVER

响应：Software Version:V1100B01305R00C0002

响应：OK

请求：AT+NV=GET,HARDVER

响应：Hardware Version:XYM110_HW_V1.0

响应：OK

请求：AT+SWVER=V1100B01306R00C0023

响应：OK

请求：AT+SWVER

响应：Software Version: V1100B01305R00C0023

请求：AT+HVER=XYM110_HW_V1.9

响应：OK

请求：AT+HVER

响应：Hardware Version: XYM110_HW_V1.9

1.6 测试命令+TEST

语法结构

命令	响应
AT+TEST=<act>,<address>,<length>[,<data>]	OK

接口说明

该命令用于读取或写入任何可用的内存空间，包括 RAM、flash、寄存器等。

典型示例

请求: AT+TEST=READ,0X27005000,0X1000
 响应: OK
 请求: AT+TEST=WRITE,0xA0058000,4,0X1234
 响应: OK

1.7 设置 ERROR 上报模式+CMEE

语法结构

命令	响应
AT+CMEE=<error_mode>	OK

接口说明

该命令是 3GPP 命令，用于设置 ERROR 上报模式。M3 核处理完会传给 DSP 核处理。

取值说明

<error_mode>:

- 0: AT 命令报错时，只响应\r\nERROR\r\n，无错误码。
- 1: AT 命令报错时，响应\r\n+CME ERROR: %d\r\n(与 Onenet 业务相关则回复\r\n+CIS ERROR: %d\r\n)，带上错误码。
- 2: AT 命令报错时，回复\r\n+CME ERROR: %d\r\n(与 Onenet 业务相关则回复\r\n+CIS ERROR: %d\r\n)，带上错误原因描述。

典型示例

请求: AT+CMEE=0
 响应: OK
 请求: AT+CMG
 响应: ERROR
 请求: AT+CMEE=1
 响应: OK
 请求: AT+CMG
 响应: +CME ERROR:4
 请求: AT+MIPLNOTIFY=0,514456,3303,0,5601,4,4,88,0,0
 响应: +CIS ERROR:601
 请求: AT+CMEE=2
 响应: OK
 请求: AT+MIPLNOTIFY=0,514456,3303,0,5601,4,4,88,0,0
 响应: +CIS ERROR:CIS_PARAM_ERROR
 请求: AT+CMG
 响应: +CME ERROR:"operation not supported"

1.8 设备软重启+RB/NRB

语法结构

命令	响应
AT+RB	REBOOTING
AT+NRB	REBOOTING
AT+NRB=<xy_nv_save>,<user_nv>	REBOOTING

接口说明

该命令控制芯片进入软件重启流程，用于设置出厂 NV 配置后，重启生效场景。软重启过程中，用户设置的 UTC 仍然有效，如果用户希望无效，必须在调用软重启接口之前，调用 xy_rtc_timer_delete 接口删除响应的 URC 设备，详细内容请查看《海凌科 HLK-N10 业务平台接口说明》文档。

取值说明

<xy_nv_save>：平台的 NV，包括小区信息、RTC 定时器信息等；

0：不保存

1：保存，默认值

<user_nv>：是否保存用户相关的工作态 NV，默认保存；

0：不保存

1：保存

典型示例

请求：AT+Nv=SET,OFFTIME,0

响应：OK

请求：AT+Nv=SAVE

响应：REBOOTING

请求：AT+RB

响应：REBOOTING

1.9 设备冷重启+COLDREB

语法结构

命令	响应
AT+COLDREB	REBOOTING
AT+COLDREB=<xy_nv_save>,<user_nv>	REBOOTING

接口说明

该命令用于冷重启芯片，控制芯片断电后给所有外设总线等硬件电路重新上电。冷重启过程中，用户设置的 UTC 仍然有效，如果用户希望无效，必须在调用软重启接口之前，调用 xy_rtc_timer_delete 接口删除响应的 URC 设备，详细内容请查看《海凌科 HLK-N10 驱动接口 API 说明》文档。

取值说明

<xy_nv_save>：平台 NV，包括小区信息、RTC 定时器信息等；

0：不保存

1: 保存

<user_nv>: 是否保存用户相关的工作态 NV, 默认保存;

0: 不保存

1: 保存

典型示例

请求: AT+Nv=SET,OFFTIME,0

响应: OK

请求: AT+COLDNRB

响应: REBOOTING

1.10 设备恢复出厂设置+RESET

语法结构

命令	响应
AT+RESET	RESETTING

接口说明

该命令用于在检查到设备发生严重软件异常, 长时间无法恢复到正常状态时, 使用该命令恢复出厂 NV, 并擦除所有工作态 NV。

1.11 人为主动断言+ASSERT

语法结构

命令	响应
AT+ASSERT	+ASSERT
AT+ASSERTCP	

接口说明

该命令供用户主动断言, 其中, ASSERT 用于 M3 核断言; ASSERTCP 用于 DSP 核断言。当为 debug 版本时, 通过这两个命令, 可以导出当前工作的实时现场, 以协助问题定位。Release 版本可以通过该条命令验证系统断言后是否能够正常重启工作, 客户要关注 user_flash_hook_by_reset 接口是否按用户设计预期被执行, 详情可参考《海凌科 HLK-N10 软件开发指南》。

1.12 开关硬看门狗+WDT

语法结构

命令	响应
AT+Nv=SET,WDT,<enable>	OK

接口说明

用于打开/关闭芯片看门狗功能。当关闭 DEBUG 模式 (AT+Nv=SET,CLOSEDEBUG,1) 时, 开启看门狗功能, 若发生死机, 硬件看门狗将复位芯片。重启后只保存出厂 NV, 并擦除所有工作态 NV。

取值说明

<enable>: 开门狗功能选择
 0 表示关闭;
 1 表示打开; 默认值

1.13 设置 SIM 卡电压值+SIMVCC

语法结构

命令	响应
AT+NV=SET,SIMVCC,<val>	OK

接口说明

对应 NV 参数 sim_vcc_ctrl, 该 NV 用于选择 SIM 卡电压值, 支持 1.8V 和 3V 两种电压的动态和静态配置。

如果用户需要每次开机皆从 1.8V 到 3V 进行自适应, sim_vcc_ctrl 参数设为 2 即可。

平台不支持 SIM 卡的热插拔功能, 仅在断电上电场景下, 才会执行 SIM 卡自适应动作; 其他场景, 皆使用之前保存的值, 即<val>的 bit[4-6]。

需要注意, 芯片内部晶振和 SIM 卡共用 VDDIO 供电, 故启用自适应需要晶振也支持 1.8V; 若晶振不支持 1.8 伏, 则可能导致异常死机。

取值说明

<val>: 整型, 取值范围[0,3]。按照位数进行细分。详细取值见下表。

bit[0-2]	Bit[3]	Bit[4-6]	Bit[7]	描述
000	--	--	--	选择类型, 3V
001	--	--	--	选择类型, 1.8V
010	--	--	--	选择类型, 优先 1.8V
011	--	--	--	选择类型, 优先 3V
--	Reserved	--	--	预留位
--	--	010	--	当前 SIM 卡电压 3V
--	--	100	--	当前 SIM 卡电压 1.8v
--	--	--	Reserved	内部调试位, 未开放

1.14 AT 串口设置+UARTSET

语法结构

命令	响应
AT+UARTSET=<baud_rate>[,<store>[,<openstandby>]]	OK +CME ERROR:<err>

接口说明

该命令用于设置 AT 口的波特率等参数，默认波特率为 9600。目前平台仅支持波特率的设置，其他参数使用平台默认值。该命令支持动态配置波特率，当 store 为 0 时，动态配置波特率生效，要求对端也切换成对应的波特率；当 store 为 1 时，将设置的波特率除以 2400 后，保存到出厂 NV 的 uart_rate 参数里，且重启生效。

对于动态切换波特率场景，由于需要底板 MCU 与芯片同时切换，存在时间差问题，不能保证“OK”被底板 MCU 准确收到。为了防止切换失败，建议用户发送完切换命令后，再通过“AT”“OK”进行握手。

由于高波特率下，STANDBY 唤醒后 BBPLL 稳定需要 3 毫秒左右，从而造成数据丢失。针对动态切换波特率场景，支持用户通过参数来动态开关 standby。若用户不赋值<openstandby>参数，则系统检测到<store>为 0 时，根据波特率动态开关 STANDBY，波特率小于等于 19200 时，开启 STANDBY，波特率高于 19200 时，关闭 STANDBY 模式，避免数据丢失。

由于需要考虑容错机制，不建议启用动态波特率配置。

取值说明

<baud_rate>: 波特率，目前支持最高 115200

<store>: 是否动态波特率，默认值为 0，即动态切换波特率；

1 表示固定波特率保存到 NV 中，且立即重启；

0 表示动态波特率生效，需要对方向同步切换波特率

<openstandby>: 是否打开 standby 睡眠，仅当 store 设为 0 时方有效；默认为动态开关。

1 表示启动 STANDBY 睡眠机制，

0 表示关闭 STANDBY 睡眠机制

典型示例-动态修改波特率

AT+UARTSET=115200,0 //动态修改波特率，进而 OK 无法在对端显示，对端需同步切换为 115200 方可，且临时关闭 STANDBY

AT+NPING=139.224.112.6,32,1,30,1 //在 115200 波特率上进行 AT 传输
 OK

AT+UARTSET=9600,0 //动态修改波特率，进而 OK 无法在对端显示，对端需同步切换为 9600 方可，且打开 STANDBY

AT+NPING=139.224.112.6,32,1,30,1 //在 9600 波特率上进行 AT 传输
 OK

典型示例-修改 NV 默认波特率配置

AT+UARTSET=115200,1 //保存波特率配置，且自动重启

REBOOTING //芯片重启后主动上报开机回码

AT+NPING=139.224.112.6,32,1,30,1 //在 115200 波特率上进行 AT 传输
 OK

AT+NV=SAVE

REBOOTING //芯片重启开机回码

1.15 省电锁+WORKLOCK

语法结构

命令	响应
----	----

AT+WORKLOCK=<enable>[,<type>]	OK
	+CME ERROR:<err>

接口说明

该扩展 AT 命令为芯片工作锁，目的是在有业务流程进行时禁止芯片进入 DEEPSLEEP 深睡模式，以确保用户流程不被打断，且提高网络数据传输的实时性。待用户流程结束后，再输入“AT+WORKLOCK=0”，芯片内部会执行完整的 PSM 睡眠流程，随后芯片进入 DEEPSLEEP 深睡模式。

该条命令常用于需要支持 PSM 功能的产品，对于 OPENCPU 用户，则必须使用该条命令进行睡眠控制。其中 type 参数为可选参数，供用户区分不同的任务锁，具体值用户可自由设置；0 或缺省，表示不细分各个任务锁。为了确保正常深睡，所有模组必须确保锁的配对使用。具体使用请参阅《海凌科 HLK-N10 低功耗开发指南》。

取值说明

<enable>：申请/释放锁

1 表示申请锁，默认值

0 表示释放锁

<type>：锁类型，0 表示不细分锁类型，1/2/3...等值用于供用户针对每个个体任务设置不同的类型值

<err>：错误码，参看 err 章节。

典型示例

请求：AT+WORKLOCK=0

响应：OK

主动上报：+POWERDOWN:8824 //TAU 到期时间偏移 8824 秒

1.16 用户快速链接释放+XYRAI

语法结构

命令	响应
AT+XYRAI=<remote_ip>,<remote_port>	OK +CME ERROR:<err>

接口说明

该扩展 AT 命令供用户触发链接释放流程，平台收到该条命令后，发送一个空的 UDP 报文给网络侧，并携带 RAI=1 指示，以通知基站进行快速链接释放。用户使用该条命令时，需要确保当前 NB 协议栈处于链接态，否则会造成一次多余的链接建立和释放的流程。

取值说明

<remote_ip>：点分十进制的 IP 地址或域名，域名不超过 20 个字节，如“10.0.122.10”；

<remote_port>：服务器的端口号；

1.17 动态开关+STANDBY

语法结构

命令	响应
----	----

AT+STANDBY=<enable>[,time]	OK
	+CME ERROR:<err>

接口说明

该命令用于开/关 STANDBY 睡眠，设置睡眠的阈值。STANDBY 睡眠机制会关闭串口的 BBPLL 时钟，进而造成下次 AT 唤醒时，需要 2 毫秒左右的稳定时长，导致 AT 命令有丢失情况发生。

为了方便用户调试和开发，该命令供用户动态开关 standby。其中<time>为保持 standby 关闭时间。

对于整机产品开发，建议用户不要使用该命令，具体开发流程，参考《海凌科 HLK-N10 低功耗开发指南》。

取值说明

<enable>：是否打开 standby 开关

1 表示打开 standby 睡眠机制；默认值

0 表示关闭

<time>：保持 standby 关闭时间，单位秒，仅在 enable 取值为 0 时有效；

<err>：错误码，参看 err 章节。

1.18 省电模式主动上报+POWERDOWN(仅用于带外部 MCU 产品)

语法结构

命令	响应
	+POWERDOWN=<next TAU offset>,<next RTC wakeup offset>

接口说明

仅用于带底板 MCU，且支持断电模式的产品。ARM 核检测到当前模组可以进入深睡后，发送命令给外部底板，通知底板可以执行断电操作。若底板采取断电策略，则在下次给模组上电后，需要发送 OFFTIME 主动上报给模组，以便模组进行 RTC 时间补偿。

取值说明

<next TAU offset>：整型，单位秒，PSM 的 TAU 超时偏移，即多少秒后 TAU 将超时；该参数为 0 时，表示无效的 TAU 时长。

<next RTC wakeup offset>：RTC 硬件定时器下一个唤醒时刻点偏移，如 TAU 时刻点、用户硬定时、FOTA 定时等；该参数为-1 时，表示未设置 RTC 定时事件。

1.19 上电完成主动上报+POWERON (仅用于带外部 MCU 产品)

语法结构

命令	响应
	+POWERON:<cause>

接口说明

仅用于存在外部底板 MCU 时，且伴随 SIM 卡初始化完成时主动上报“^SIMST:”。系统启动后上报启动原因给外部用户，正常情况下，用户收到该主动上报后，表明芯片已完成 SIM 卡初始化，能够开始使用 3GPP 协议栈。

对于值 2/5/6 三个异常重启情况，外部 MCU 运行过程中收到该主动上报值后，表明芯片发生重启行为，此时双方的状态机已经不一致，需要针对具体的重启原因值进行排查，或者进行整机重启复位。

取值说明

<cause>:

取值	含义	备注
0	POWER ON	正常的上电动作
1	PIN RESET	PIN复位，供外部进行芯片的硬复位
2	SOFT RESET	软重启功能
3	UTC WAKEUP	UTC唤醒动作
4	External PIN WAKEUP	外部PIN唤醒，常与数据通信绑定
5	WDT RESET	硬件看门狗重启
6	UNKNOWN	未知的异常重启，如电容放电不充分等

1.20 断电时间补偿+OFFTIME(仅用于带外部 MCU 产品)

语法结构

命令	响应
AT+OFFTIME=<off time>	OK

接口说明

仅用于存在外部底板 MCU 时。外部 MCU 若对 NB 模块进行断电操作，当再次给 NB 模块上电后，需要通过该条命令告知 NB 模块期间断电时长，以便 NB 模块更新内部的 RTC 定时器。

若发生整机异常断电，底板 MCU 无法准确获知芯片的断电时长，则输入“AT+OFFTIME=0”即可，通知芯片放弃断电时长补偿，直接进行 ATTACH 正常开机操作。

需要注意的是，支持断电补偿机制的深睡，不能使用 AT+FASTOFF/CPOF，只能使用 AT+WORKLOCK 机制进入深睡。

取值说明

<off time>: 外部 MCU 断电 NB 模块的秒数

1.21 低电压告警+LOWVBAT

语法结构

命令	响应
	+LOWVBAT:<vbat val>

接口说明

用于上报低压告警，以通知底板 MCU 更换电池或其他策略。该主动上报的触发条件是 min_mVbat 值有效，该出厂 NV 参数指示产品的最低工作电压。海凌科平台在开机初始化及 flash 的擦写接口处进行了低电压的检测，并且提供了 xy_check_low_vBat 接口供用户二次开发时在合适点调用，以检查是否低电压。

取值说明

<vbat val>：产品的低压工作下限，目前芯片的工作电压为 3.1~4.2V

1.22 电压查询+VBAT

语法结构

命令	响应
AT+VBAT=?	+VBAT:<val>

接口说明

用于查询当前电压值，单位为毫伏。

1.23 SIM 卡类型主动上报+UICCTYPE

该命令仅用于调试阶段。

语法结构

命令	响应
	+UICCTYPE:<type>

接口说明

当 PDP 激活成功后，主动上报，指示当前 SIM 卡的类型，仅用于调试。

取值说明：

<type>：SIM 卡类型，取值如下：

- 0：无卡
- 1：TELECOM，电信卡
- 2：MOBILE，移动卡
- 3：UNICOM，联通卡

1.24 内存动态查看+NUESTATS

语法结构

命令	响应
AT+NUESTATS=<param>	OK

接口说明

该功能与平台提供的“AT+MEMSTATS”功能相似，用于查看系统运行时的堆和栈动态统计信息。

取值说明

<param>: 为 APPSMEM 或 ARMMEM, 查询用户可用内存, 正常情况下上报相关内存信息如下:

AT+NUESTATS=APPSMEM

NUESTATS:APPSMEM, Current Allocated: //目前已分配内存大小

NUESTATS:APPSMEM, Total Free: 68497 //总剩余内存大小

NUESTATS:APPSMEM, Max Free: 68185 //剩余最大连续内存块大小

NUESTATS:APPSMEM, Number Allocs: 194 //历史总内存分配次数

NUESTATS:APPSMEM, Number Frees: 159 //历史总内存释放次数

OK

1.25 Fota 升级开关+ FOTACTR

语法结构

命令	响应
AT+FOTACTR=<n>	OK +CME ERROR:<err>
AT+FOTACTR?	+FOTA:<n>

接口说明

该命令用于打开/关闭 Fota 自动升级功能。打开状态下, 如果云平台存在 Fota 升级任务的前提下, 将自动触发 Fota 升级任务; 关闭状态下, 终端将忽略云平台发起的 Fota 操作, 需要用户手动开启此开关或者向私有 Fota 服务器发起连接。

取值说明

<n>: 默认 0, 即打开 FOTA 自动升级功能

取值	含义
0	打开fota自动升级功能
1	关闭fota自动升级功能

< err>: 错误码, 参看 err 章节。

1.26 透传数据 ATD*98/ATD*99(仅用于内部调试)

语法结构

命令	响应
ATD*98/ATD*99	CONNECTING OK +CME ERROR:<err>
+++	OK +CME ERROR:<err>

接口说明

该扩展命令修改于 PPP 协议拨号上网模式, 通过命令切换串口状态, 实现数据透传功能。

取值说明

ATD*98/ATD*99: AT 命令模式下, 发 ATD*98/r/n 或 ATD*99/r/n, 切换到数据透传模式;

+++ : 结束透传模式, 输入连续的 “+++” 将退出透传模式, 进入 AT 命令模式。

注意用户数据中不能出现 “+++” 否则会结束透传模式。

CONNECTING: 进入透传模式的上报指示

<err>: 错误码

典型示例

```
ATD*98          //切换到透传模式状态
CONNECTING      //已切换至透传模式, 用户可进行透传数据操作
OK
QWERTYUIOPASDFGHJKLZXCVB
QWERTYUIOPASDreasxVB
QWERTYUIOPASXCVBNM+++ //用户数据后的 “+++” 用于结束透传模式
OK

ATD*99          //切换到透传模式状态
CONNECTING
OK
QWERTYUIOPASXCVBNM
QWERTYUIOPASDreasxVB
+++ //用户可在最后一组数据发送结束后, 发 “+++” 结束透传模式
OK
```

注意事项

- 切换到透传模式的命令, 需要加上**后缀\r\n**, 结束透传模式只需发送三字节 **“+++”**, 具体发送方式要根据用户使用的 AT 调试工具;
- 目前仅用于内部调试, 不提供用户使用

2 外部适配扩展 AT

2.1 切换深睡唤醒主动上报+NPSMR

语法结构

命令	响应
AT+NPSMR=<n>	OK +CME ERROR:<err>
AT+NPSMR?	+ NPSMR:<n>,<mode>

接口说明

该命令用于切换深睡/唤醒的主动上报的信息内容。

取值说明

<n>:

取值	含义
1	打开+NPSMR主动上报, 深睡时上报 "+NPSMR:1" ; 唤醒时上报 "+NPSMR:0"
0	关闭+NPSMR主动上报, 深睡时上报 "+POWERDOWN:<nextwakeupoffset>" ; 唤醒时上报 "+POWERON:<case>"

<mode>: 预留参数, 暂无意义

<err>: 错误码, 参看 err 章节。

典型示例

AT+NV=SET,IPALIVE,0 //关闭 IP 地址始终有效不变的功能, 该命令在产线直接修改 NV, 不需要再发此命令

OK

/*****期间可以进行若干条业务 AT 命令交互, 如 socket 通信*****/

AT+WORKLOCK=0 //通知执行 PSM 省电, 芯片将进入深睡

OK

+NPSMR:1 //主动上报命令, 表示芯片已经进入深睡

/*****底板 MCU 给芯片断电, 直到下次再需使用时再给芯片上电*****/

AT+OFFTIME=120 //开启时间补偿, 再次上电必须给出补偿

OK

/*****期间可以进行若干条业务 AT 命令交互, 如 socket 通信*****/

AT+WORKLOCK=0 //通知执行 PSM 省电, 芯片将进入深睡

OK

+NPSMR:1 //主动上报命令, 表示芯片已经进入深睡

2.2 世界时间配置+CCLK

语法结构

命令	响应
AT+CCLK=<time>	OK
	+CME ERROR:<err>
AT+CCLK?	+CCLK:<time> OK

接口说明

用于供用户动态设置世界时间，并提供实时查询命令。平台内部会在 Attach 时获取。

取值说明

<time>: 时间，必选参数，格式为"yy/MM/dd,hh:mm:ss±zz"，例如"19/05/06,22:10:00+08",最大年份不超过 99，将自动转为不超过 2099；其中，±zz 为子时区，每个子时区 15 分钟。

典型实例

```
AT+NITZ=0
OK
```

```
AT+CCLK=19/03/30,10:50:30+32
OK
```

```
AT+CCLK?
+CCLK:19/3/30,10:50:33+32
OK
```

```
AT+NITZ=0
OK
```

```
AT+CCLK=19/03/30,11:29:0+8
OK
```

```
AT+CCLK?
+CCLK:19/3/30,13:29:1+8
OK
```

2.3 世界时间获取方式+NITZ

语法结构

命令	响应
AT+NITZ=<mode>	OK
	+CME ERROR:<err>
AT+CCLK?	+NITZ:<mode> OK

接口说明

该命令用于配置世界时间的设置方式。

取值说明

<mode>: 更新世界时间的模式，默认值为 1

- 0 表示由用户通过 CCLK 命令进行本地设置；
- 1 表示由 3GPP 的 “+CTZEU:” 主动上报来准实时更新；

典型实例

```
AT+NITZ=0
OK
```

```
AT+CCLK=19/03/30,11:29:0+8
OK
```

```
AT+CCLK?
+CCLK:19/3/30,11:29:1+8
OK
```

2.4 AT 串口设置+NATSPEED

语法结构

命令	响应
AT+NATSPEED=<baud_rate>,<timeout>,<store>,<sync_mode>[,<stopbits>[,<parity>[,<xonxoff>]]]	OK
	+CME ERROR:<err>

接口说明

该命令用于设置 AT 口的波特率等参数，默认为 9600 波特率。**不建议使用该命令更改 AT 口参数，建议使用 “AT+UARTSET” 命令。** 目前平台仅支持设置 <baud_rate> 参数，其他参数使用平台默认值。该命令支持动态配置波特率，当 store 为 0 时，动态配置波特率生效，要求对端也切换成对应的波特率；当 store 为 1 时，将设置的波特率除以 2400 后，保存到出厂 NV 的 uart_rate 参数里，且重启生效。

对于动态切换波特率场景，由于需要底板 MCU 与芯片同时切换，存在时间差问题，不能保证 “OK” 被底板 MCU 准确收到。为了防止切换失败，建议用户发送完切换命令后，通过 “AT” “OK” 进行握手。

默认支持停止位，且不可更改。

由于需要考虑容错机制，不建议用户启用动态波特率配置。

取值说明

<baud_rate>：波特率，目前支持最高 115200

<timeout>：该参数**目前未启用**

<store>：是否动态配置波特率，默认为 1，即设置固定波特率，

1，表示设置固定波特率，且保存到 NV 中，且立即重启；

0，表示动态配置波特率生效，需要对方向同步切换波特率

<err>：错误码，参看 err 章节。

典型示例-动态修改波特率

```
AT+NATSPEED=115200,0,0 //动态修改波特率，进而 OK 无法在对端显示，对端需同步切换为 115200 方可
```

```
AT+NPING=139.224.112.6,32,1,30,1 //在 115200 波特率上进行 AT 传输
OK
```

AT+NATSPEED=9600,0,0 //动态修改波特率, 进而 OK 无法在对端显示, 对端需同步切换为 9600 方可

AT+NPING=139.224.112.6,32,1,30,1 //在 9600 波特率上进行 AT 传输
OK

典型示例-修改 NV 默认波特率配置

AT+NATSPEED=**115200** //默认为 NV 保存,自动重启生效
REBOOTING

AT+NPING=139.224.112.6,32,1,30,1 //在 **115200 波特率**上进行 AT 传输
OK
AT+NVS=SAVE
REBOOTING

2.5 设置日志等级+NLOGLEVEL

语法结构

命令	响应
AT+NLOGLEVEL=<level>	OK +CME ERROR:<err>
AT+NLOGLEVEL?	+LOGLEVEL:<level> OK +CME ERROR:<err>

接口说明

该命令设置日志级别, 指示在日志工具上的日志输出情况。

取值说明

<level>: 日志级别要求

NONE: 不打印日志

NORMAL: 打印所有核的日志, , 默认值;

M3UNCLEAR: 仅打印平台 M3 核日志

USER: 仅打印用户相关日志

2.6 本地 FOTA 升级 +NFWUPD

语法结构

命令	响应
AT+NFWUPD=<cmd>[,<sn>,<len>,<d ata>,<crc>]	OK +CME ERROR:<err>
AT+NFWUPD=?	+NFWUPD:(list of supported <cmd>s) OK +CME ERROR:<err>

接口说明

此命令支持固件更新。它允许包下载，包验证和固件升级。在升级固件之前，需要先下载软件包。下载完成后，发送包有效性 cmd。假如差分包合法就可以升级，否则返回错误和打印原因。在差分包校验完成后，发送固件升级命令。如果没有对差分包进行校验，拒绝执行固件升级命令。

取值说明

<cmd>: 差分包处理命令

0: 擦除 FLASH 并初始化

1: <sn>,<len>,<data>,<crc>; 下载一个包段。包段是 FOTA 包的连续段。段可以是任意长度(小于 512 字节), 但是必须按顺序提供。

2: 校验差分包

5: 启动升级

<sn>: 序号。它从 0 开始, 每个包段递增 1

<len>: 数据的字节长度, 应该为 32/64/128/256/512。

<data>: 此数据包传输的数据, 采用十六进制字符串格式。

<crc>: 包段二进制数据的 CRC。CRC 以十六进制字符串的形式发送。crc 是包段中每个字节的 xor8。

3 平台 Socket 扩展命令

3.1 DNS 设置+XDNSCFG

语法结构

命令	响应
AT+XDNSCFG=<pri_dns>[,<sec_dns>]	OK +CME ERROR:<err>
AT+ XDNSCFG?	+XDNSCFG: <pri_dns>[,<sec_dns>] OK

接口说明

该命令用于手动设置 DNS 服务器。

取值说明

<pri_dns>: 主 DNS 服务器地址, 点分十进制格式

<sec_dns>: 辅 DNS 服务器地址, 点分十进制格式

典型示例

```
AT+XDNSCFG=8.8.8.8,114.114.114.114
OK
```

3.2 DNS 解析+XDNS

语法结构

命令	响应
AT+XDNS=<domain>	OK +XDNS:<ip address> +CME ERROR:<err>

接口说明

该命令用于获取URL的IP地址, 若上报网络IP时无DNS, 则会使用NV默认的地址。该命令支持异步和同步两种查询机制。当vertype是CMIOT_VER时, 为异步查询机制, AT命令返回OK, 且上报查询的结果,如 "+XDNS:111.62.246.155" 或者 "+XDNS:QUERY_DNS_FAILED"。当vertype非CMIOT_VER时, 为同步查询机制: AT命令查询为阻塞时, 在AT未返回结果前链路都是阻塞的。

取值说明

<domain>: URL 的域名, 长度不超过 100 字节

<ip address>: URL 的 IP 地址

典型示例

```
AT+XDNS="www.baidu.com"
OK
+XDNS:183.232.231.172
```

3.3 数据模式设置+XDTMODE

语法结构

命令	响应
AT+XDTMODE=<mode>	OK +CME ERROR:<err>
AT+XDTMODE?	+XDTMODE:<mode>

接口说明

数据模式设置，目前平台不支持码流格式。对于字符串型，数据内容将会使用 "" 进行传递，以避免双引号、逗号等特殊字符的干扰。优先推荐使用 HEX 字符串类型。

取值说明

<mode>: 数据格式

- 1: 字符串，且需用 "" 将数据括起来
- 2: hex 字符串
- 3: 码流 (暂不支持)

发送字符串数据，如果数据自带逗号，需要将数据用英文双引号引起来 AT+XSSEND=0,7,"12,3456"

<err>: 错误码，参看 err 章节。

典型示例

```
AT+XDTMODE=2          //发送 data 为 Hex 格式
OK
AT+XSSEND=0,6,3AC2884E0D0A    //在发送十六进制数据下，数据中可以存在 0D0A
OK
AT+XSSEND=0,6,3AC2884E7F46,5
OK
+NSOSTR:0,5,1
AT+XDTMODE=1          //发送 data 为字符串格式
OK
AT+XSSEND=0,6,"ascv12"    //发送字符串数据
OK
```

3.4 开启服务+XSOPEN

语法结构

命令	响应
AT+XSOPEN=<type>,<remote_ip>,<remote_port>[,<local_port>]	+XSOPEN:<socket_id> OK +XSSTATE:<socket_id>,<state>
	+CME ERROR:<err>
AT+XSOPEN?	+XSOPEN:<type>,<remote_ip>,<remote_port>[,<local_port>] OK

接口说明

使用该命令打开 TCP/UDP 连接。查询命令可主动查询已创建的 socket 链路信息。

取值说明

<socket_id>: 整数类型, 0-5, 链路号, 目前仅支持 0 和 1;

<state>: 链路状态

1 表示连接态;

0 表示断开连接。

<type>: 传输类型

取值	含义
0	TCP
1	UDP

<remote_ip>: 服务器地址或者域名, 点分十进制格式, 域名不超过 20 个字节;

<remote_port>: 服务器的 port;

<local_port>: 可选参数, 本地的 port, 默认值为 0, 表示由 TCPIP 自由选择。

<err>: 错误码, 参看 err 章节。

典型示例

```
AT+XSOPEN=0,139.224.112.6,10003
```

```
+XSOPEN:0
```

```
OK
```

```
+XSSTATE:0,1
```

AT+XSOPEN? //已创建两路 socket 后, 发送查询命令

```
+XSOPEN:0,0,139.224.112.6,10027
```

```
+XSOPEN: 1,1,139.224.112.6,10025
```

```
OK
```

3.5 链接状态上报+XSSTATE

语法结构

命令	响应
	+XSSTATE:<socket_id>,<state>

接口说明

主动上报方式通知对应的 socket_id 的链接状态变化

取值说明

<socket_id>: 整数类型, 0-5, 链路号, 目前仅支持 0 和 1;

<state>: 链路状态

1 表示连接态;

0 表示断开连接。

3.6 发送数据+XSSEND

语法结构

命令	响应
AT+XSSEND=<socket_id>,<size>,<data>[,<sequence>]	OK [+NSOSTR:<sockid>,<sequence>,<status>] +CME ERROR:<err>
AT+XSSEND?	+XSSEND:<socket_id>,<size> OK

接口说明

底板 MCU 发送数据请求，最终通过 NB 协议栈发送给远端服务器。查询命令可查看所有 socket 链路发送上行数据情况。

取值说明

<socket_id>: 整数类型, 0-5, 链路号, **目前仅支持 0 和 1**;

<size>: 有效数据长度, 目前最大限制 1400 字节

<data>: 数据流, 格式由参数+XDTMODE 指令设置决定;

<sequence>: 是否需要确认的数据包的发送状态,

0 或空: 表示无需确认,

1-255: 表示需要确认数据发送成功与否状态。

需要确认发送状态的情况下, 未收到主动上报 "+NSOSTR:<sockid>,<sequence>,<status>" , 表明数据还在发送, 不可再次发送具有相同 sockid 和 sequence 的数据命令

<Sockid>: 由发送数据时传入的 socket 决定。

<Status>: 数据发送状态,

0: 失败,

1: 成功

<err>: 错误码, 参看 err 章节。

典型示例

AT+XDTMODE=2 //发送 data 为 Hex 格式

OK

AT+XSSEND=0,6,3AC2884E0D0A //在发送十六进制数据下, 数据中可以存在 0D0A

OK

AT+XSSEND=0,6,3AC2884E7F46,5

OK

+NSOSTR:0,5,1

AT+XDTMODE=1 //发送 data 为字符串格式

OK

AT+XSSEND=0,6,"ascv12"

OK

AT+XSSEND? //已创建两路 socket 情况下, 发送查询命令

+XSSEND:0,15

+XSSEND: 1,0

OK

3.7 查询发送状态+SEQUENCE

语法结构

命令	响应
AT+SEQUENCE=<socket>,<sequence>	<socket>,<sequence>,<status> OK
	+CME ERROR:<err>

接口说明

底板 MCU 发送数据请求，查询数据发送状态。

取值说明

<socket>：要查询的指定 socket_id，

<sequence>：需要查询的 sequence

<status>：查询到的状态，

0：发送不成功，

1：发送成功，

-1：未使用到该 sequence。

典型示例

```
AT+SEQUENCE=0,5
1
OK
AT+SEQUENCE=0,150
-1
OK
```

3.8 下行数据主动上报+XSNMI

语法结构

命令	响应
AT+XSNMI=<report_mode>	OK
	+CME ERROR:<err>
	+XSRECV:<socket_id>,<len>,<data>
AT+XSNMI?	+XSNMI:<report_mode>

接口说明

设置下行数据主动上报模式，其中+XSRECV 为主动上报命令。

取值说明

<report_mode>：整数类型，支持 0~1，默认为 0：

0 下行数据不缓存。上报格式：显示链接指示、有效数据长度及数据流

上报信息：+XSRECV:<socket_id>,<len>,<data>

1 下行数据缓存。上报格式：仅显示指示链接指示、有效数据长度

上报信息: +XSRECV:<socket_id>,<len>, 缓存一个字节情况下, 最多缓存 180 次, 否则容易申请不到内存。

<socket_id>: 整数类型, 0-5, 能支持的链接数, 目前仅支持 0 和 1

<len>: 有效数据长度, 目前单个下行数据长度最大限制 1400 个字节。

<data>: 数据, 数据类型由指令+XDTMODE 设置值决定, 默认为 HEX 格式数据:

- 1 发送上行数据格式为字符串, 下行数据若为字符串类型, 读取格式为"字符串", 字符串数据放入双引号中;
- 2 默认模式。发送上行数据格式为 Hex, 下行数据若字符串类型, 则读取时将数据流转换为 ASCII 码类型上报。

典型示例

```
AT+XDTMODE=1
OK
AT+XSNMI=0
OK
+XSRECV:0,10,"aassdffgg"
+XSRECV:0,10,"aassdffgg"
AT+XDTMODE=2
OK
AT+XSNMI=1
OK
+XSRECV:0,9
AT+XSNMI=0
OK
+XSRECV:0,9,302031322032312034
```

注意事项

<report_mode>参数设置后掉电不保存, 默认为 0, 下行数据不缓存模式下, 上报信息: +XSRECV:<socket_id>,<len>,<data>。下行数据发送长度超过 4000 后, 将主动上报"+XSRECV:drop<len> bytes pkt"信息提示用户将不缓存之后的下行数据。该参数与+XDTMODE 指令中<mode>参数配合使用, 决定下行数据的显示格式。

<mode>为 1 且<report_mode>为 1 时, 下行数据为字符串, 接收端显示 data 格式为"data";

<mode>为 2 且<report_mode>为 1 时, 下行数据为字符串, 接收端显示 data 格式为 data, data 是将数据流转换为 ASCII 码类型。

3.9 读取下行数据+XSREAD

语法结构

命令	响应
AT+XSREAD=<socket_id>,<req_len>	OK +XSSTATE:<socket_id>,<read_len>,<data_buf><remaining_length>
	+CME ERROR:<err>

接口说明

读取下行数据流。新增了缓存功能, 可根据需求调整<read_len>大小依次读取缓存数据流以及查看剩余数据长度。

取值说明

- <socket_id>: 整数类型, 0-5, 有效链路号, 目前仅支持 0 和 1。
- <req_len>: 数据的十进制长度, 本次请求读取的数据长度。
- <read_len>: 数据的十进制长度, 读取到数据的长度。
- <data_buf>: 数据流, 数据显示格式由 "+XDTMODE" 指令的<mode>参数决定, 若是字符串类型, 则会将内容转换为 ASCII 码。
- <remaining_length>: 缓存中剩余数据的长度。
- <err>: 错误码, 参看 err 章节。

典型示例

```
+XSRECV:0,12
AT+XSREAD=0,4
0,4,1E34A6B8,8
OK
AT+XSREAD=0,5
0,5,1E34A6B83D,3
OK
AT+XSREAD=0,5
0,3,789C1E,0
OK
```

注意事项

- 在不读状态下, 缓存能力达 4000 字节大小, 若最后一个数据包的长度与已缓存长度之和大于 3999, 则最后一个数据包丢掉;
- 若此请求读取长度大于已缓存数据流长度, 则将读取缓存中全部数据流, 并且返回结果中的<read_len>为实际读取长度, 而不是用户请求读取的长度。

3.10 关闭服务+XSCLOSE

语法结构

命令	响应
AT+XSCLOSE=<socket_id>	OK +XSSTATE:<socket_id>,<state>
	+CME ERROR:<err>
AT+XSCLOSE?	+XSCLOSE: (Socket<socket_id>,<state>), OK

接口说明

查询命令可查看已关闭的 socket 链路情况, 主动上报所有链路的 state。

取值说明

- <socket_id>: 整数类型, 0-5, 能支持的链接数, 目前仅支持 0 和 1;
- <state>: 链路状态
 - 1 表示连接态;
 - 0 表示断开连接
- <err>: 错误码, 参看 err 章节。

3.11 典型示例

```
AT+XSOPEN=1,139.224.112.6,10003
+XSOPEN:0
OK
+XSSTATE:0,1
AT+XDTMODE=2
OK
AT+XSSEND=0,6,3AC2884E7F46
OK
AT+XSNMI=0
OK
+XSRECV:0,9,302031322032312034
AT+XSCLOSE=0
OK
+XSSTATE:0,0 //socket_id,state
AT+XSCLOSE? //关闭 socket id 为 0 链路
+XSCLOSE:
(Socket0,0),
(Socket1,1),
OK
AT+XSCLOSE=1
OK
AT+XSCLOSE? //关闭 socket id 为 1 链路
+XSCLOSE:
(Socket0,0),
(Socket1,0),
OK
```

3.12 注意事项

- 由于内部 Socket 数量有限，仅支持 2 路同时工作，且内部释放 Socket 资源需要一定时间，建议在使用 UDP/TCP Socket 做完业务之后再执行关闭操作，避免短时间内重复创建/收发/关闭 Socket，以免 Socket 资源短时间内耗尽导致链路再次创建时失败；
- 请避免连续快速输入待发送数据且速度高于底层发送速度，否则可能引起发送缓冲区满而造成数据发送错误；
- 芯片内部可缓存一定接收数据量，当有数据到达且为手动接收模式时，需等待主动上报下行数据信息后再行读取数据；当需要连续接收大量数据时，建议将波特率调高以加快接收速度。
- 对于数据传输的扩展 AT 命令，目前不建议用户使用 ASCII 模式进行内容传递。如果用户使用 ASCII 传输数据，则必须用 "" 把 ASCII 字符串包起来。
- TCP 的 socket 扩展 AT，由于 close 接口非阻塞，会立即回复 "+XSSTATE:0,0"，故要求延迟十秒左右后再重新创建连接，否则服务器会拒绝。

4 Socket 扩展命令

4.1 开启服务+NSOCR

语法结构

命令	响应
AT+NSOCR=<type>,<protocol>,<listenport>[,<receive control>]	+NSOCR:<socket> OK
	+CME ERROR:<err>

接口说明

该命令是创建 UDP/TCP socket。对于 TCP 创建，此时并未 connect，还需+NSOCO 命令完成。

取值说明

<type>：传输类型，支持 DGRAM 和 STREAM 两种类型。

<protocol>：标准互联网协议定义，比如：UDP 是 17，TCP 是 6。

<listen port>：本地的 port，0 表示由模组随机分配；

<socket>：链路号，整型，取值范围[0,1]。

<receive control>：

1 表示接收传入的消息，默认值

0 表示忽略传入的消息

<err>：错误码，参看 err 章节。

典型示例

```
AT+NSOCR=DGRAM,17,10004,1
+NSOCR:0
OK
```

```
AT+NSOCR=STREAM,6,10005,1
+NSOCR:1
OK
```

4.2 TCP 开启服务+NSOCO

该命令仅在操作 TCP 时才需要使用。

命令	响应
AT+NSOCO=<socket>,<remote_addr>,<remote_port>	OK
	+CME ERROR:<err> +NSOCLI:<socket>

接口说明

底板 MCU 发送数据请求，TCP 完成+NSOCR 后，要进行 connect 服务器操作。

取值说明

<socket>：由 NSOCR 创建 socket 时返回的 socket_id

<remote_addr>: 服务器 IP 地址
 <remote_port>: 服务器端口号, 范围在 0-65535
 +NSOCLI: 上报并关闭异常连接的 TCP 链路。

典型示例

```
AT+NSOCR=STREAM,6,10005,1,2
+NSOCR:0
OK
```

```
AT+NSOCO=0,139.224.112.6,10005
OK
```

```
AT+NSOCO=0,139.224.112.6,10006
+CME ERROR:8009
+NSOCLI:0
```

4.3 UDP 发送数据+NSOST

语法结构

命令	响应
AT+NSOST=<socket>,<remote_addr>,<remote_port>,<length>,<data>[,<sequence>]	<socket>,<length> OK [+NSOSTR:<sockid>,<sequence>,<status>]
	+CME ERROR: <err>

接口说明

底板 MCU 发送数据请求, 最终通过 NB 协议栈发送给远端服务器。

取值说明

- <socket>: 由创建 socket 时决定的 socket_id
- <remote_addr>: IPv4 点分十进制 IP 地址
- <remote_port>: 远程端口, 范围在 0-65535.
- <length>: 为<data>的十进制长度, 最大长度 1400
- <data>: 十六进制字符串格式的数据
- <sequence>: 是否需要确认的数据包的发送状态,
 - 0 或 NULL: 无需确认,
 - 1-255: 需要确认数据发送成功与否状态。需要确认发送状态的情况下, 在未收到主动上报信息 "+NSOSTR:<sockid>,<sequence>,<status>" 则表明数据还在发送, 此时不可再次发送具有相同 sockid 和 sequence 的数据命令。
- <Sockid>: 由发送数据时传入的 socket 决定。
- <Status>: 数据发送状态,
 - 0: 发送失败,
 - 1: 发送成功

典型示例

```
AT+NSOST=0,139.224.112.6,10005,2,AB30
0,2
```

OK

```
AT+NSOST=0,139.224.112.6,10005,2,AB30,5
0,2
OK
+NSOSTR:0,5,1
```

备注

- 接收数据的最大长度为 1400 bytes.
- 仅支持 IPv4.
- 目前仅支持 hex 字符串格式的 data.

4.4 TCP 发送数据+NSOSD

语法结构

命令	响应
AT+NSOSD=<socket>,<length>,<data>[,<flag>[,<sequence>]]	<socket>,<length> OK [+NSOSTR:<sockid>,<sequence>,<status>] >]
	+CME ERROR:<err>

接口说明

底板 MCU 发送 TCP 数据请求，最终通过 NB 协议栈发送给远端服务器。

取值说明

<socket>：由创建 socket 时决定的 socket_id

<length>：要发送的数据的十进制长度，目前最大限制 1400

<data>：以十六进制字符串格式发送的数据

<flag>：定义发送消息类型，选填，为十六进制字符串，可选参数如下：

- 0x100:异常消息，高优先级发送
- 0x200:在下一条消息后指示释放
- 0x400:指示在回复下一条消息后释放

<sequence>：是否需要确认的数据包的发送状态，

0 或 NULL：无需确认，

1-255：需要确认数据发送成功与否状态。需要确认发送状态的情况下，在未收到主动上报信息 "+NSOSTR:<sockid>,<sequence>,<status>" 则表明数据还在发送，此时不可再次发送具有相同 sockid 和 sequence 的数据命令。

<sockid>：发送数据的 socket_id，由发送数据时传入的 socket 决定。

<status>：数据发送状态，

- 0：发送失败，
- 1：发送成功

典型示例

```
AT+NSOSD=0,2,4444,0x200,1
0,2
```

```
OK
+NSOSTR:0,1,1

AT+NSOSD=0,2,4444,0x200
0,2
OK
```

4.5 上行报文发送状态主动上报+SEQUENCE

语法结构

命令	响应
	+NSOSTR:<socket_id>,<seq_num>,<status>

接口说明

主动上报方式通知 socket 上行报文是否已被物理层从空口成功发送。

取值说明

- <socket_id>: 创建 socket 成功后返回的 socket_id, 发送数据时传入的 socket_id
- <seq_num>: socket 发送上行数据时输入的 sequence 参数, 范围 0<= seq_num<=255
- <status>: 发送状态
 - 0 表示失败
 - 1 表示成功

使用说明

用户在 socket 通信添加 sequence 的需求, 以识别上行报文是否已被物理层从空口发送成功。需使用平台扩展的 send2\sendto2 接口, 实现上行数据发送, 具体原型如下:

```
send2(int s, const void *data, size_t size, int flags,int seq)

sendto2(int s, const void *data, size_t size, int flags,const struct sockaddr *to, socklen_t tolen,int seq)
```

用户可根据场景需求, 对 sequence 上报参数处理, 但须严格遵守平台对 socket_id 和 seq_num 参数的解析处理。成功解析后的 socket_id 和 seq_num 参数, 用户可自行处理, 或使用平台主动上报方式判断上行数据被物理层从空口发送的状态。

注意事项

- 1.发送接口写入参数 seq 的情况下, 未收到主动上报 "+NSOSTR:<sockid>,<sequence>,<status>" , 表明数据还在发送, 不可再次发送具有相同 sockid 和 sequence 的数据命令;
- 2.发送接口中参数 seq 为 NULL 或 0 情况下, 不主动上报" +NSOSTR:";
- 3.平台接收协议栈发来的组包两个字节进行解析, 高 8 位为 socket_id,低 8 位为 seq_num, 用户须遵守解析流程。

4.6 查询 socket 链接状态 +NSOSTATUS

语法结构

命令	响应
AT+NSOSTATUS=<socket_id>	+NSOSTATUS:<socket id>,<status> OK
AT+NSOSTATUS	+NSOSTATUS:<socket id>,<status> OK
AT+NSOSTATUS?	+NSOSTATUS: (0-sock_num) OK

接口说明

获取平台所支持 socket 当前状态。执行命令返回所有支持的套接字状态的列表。

取值说明

<socket_id>: 整数类型, 0-5, 能支持的链接数, 目前仅支持 0 和 1;

<state>: 连接状态,

1 表示连接态;

0 表示断开连接。

注意事项

发送 AT+NSOSTATUS 查询时, 将上报平台所允许建链个数的状态 0 或 1。

典型示例

```
AT+NSOSTATUS=0
+NSOSTATUS:0,1
OK
AT+NSOSTATUS
+NSOSTATUS:0,1
+NSOSTATUS:1,0
OK
AT+NSOSTATUS?
+NSOSTATUS:(0-1)
OK
```

4.7 下行数据的上报模式设置+NSONMI

语法结构

命令	响应
AT+NSONMI=<mode>	OK
AT+NSONMI?	+NSONMI:<mode> OK
AT+NSONMI=?	+NSONMI:(0,1,2,3) OK

接口说明

下行数据主动上报模式设置。

取值说明

<mode>上报模式。

- 0: 下行数据无提示, 只存储 4000 个字节, 多余的丢弃, 并开始主动上报“+NSONMI:drop <len> bytes pkt”提示信息, 响应一个字节时, 最多缓存 180 次;
- 1: 下行数据提示, 第一个下行数据来临时, 会主动上报, 内容为 “+NSONMI:<socketid>,<length>”, 最多存储 4000 个字节, 多余的丢弃, 同一路 socket 的数据未读取完毕, 第二个下行数据不会主动上报提示, 直到读完一个节点的全部数据, 再上报下一个数据节点的数据信息 “+NSONMI:<socketid>,<length>”, 最终到所有数据都读完为止, 状态重置, 如果下一个节点没有数据, 则不会上报; 响应一个字节时, 最多缓存 180 次, 否则容易申请不到内存;
- 2: 下行数据提示, 内容为 “+NSONMI:<socket>,<remote_addr>,<remote_port>,<length>,<data>”, 不储存数据;
- 3: 下行数据提示, 内容为+NSONMI: <socket>,<length>,<data>, 不储存数据;

注意事项

下行数据发送长度超过 4000 后, 将主动上报“+NSONMI:drop <len> bytes pkt”信息提示用户将不缓存之后的下行数据。

可通过发送 “AT+NSONMI?” 查询当前<mode>值。

如: AT+NSONMI?

+NSONMI:1

OK

典型示例

AT+NSONMI=2

OK

+NSONMI:0,139.224.112.6,10005,10,31313232323231313131

4.8 下行数据的上报+NSONMI

语法结构

命令	响应
	+NSONMI:<socket>,<length>

接口说明

主动上报方式通知 socket 新接收到下行数据, 以便底板通过+NSORF 来读取。上报模式可由 AT+NSONMI 参数选择。

取值说明

<socket>: 由创建 socket 时决定的 socket_id (AT+NSOCR)。

<length>: 最近一条下行数据的数据长度, 最大为 1400 字节。

典型示例

+NSONMI:0,139.224.112.6,10005,10,31313232323231313131

注意事项

多个下行数据发送总长度超过 4000 后，将主动上报“+NSONMI:drop <len> bytes pkt”信息提示用户将不缓存之后的下行数据。单个下行数据的最大长度为 1400 字节。缓存一个字节情况下，最多缓存 180 次，否则容易申请不到内存。

4.9 接收数据+NSORF

语法结构

命令	响应
AT+NSORF=<socket>,<req_length>	<socket>,<ip_addr>,<port>,<length>,<data>,<remaining_length> OK [,+NSONMI:socket,length_next]
	+CME ERROR:<err>

接口说明

主动读取收到的下行数据报文，由底板 MCU 对接收到的数据进行处理，该条 AT 命令是针对数据当中的数据节点操作，操作完后才会上报下一个数据节点的数据信息，直到数据节点为 NULL。

取值说明

<socket>：由创建 socket 时决定的 socket_id

<req_length>：读取数据的十进制长度，读取对应 socket 的数据，如 req_length 大于等于所存的数据长度，则会全部读出来，否则只读取指定长度的数据，剩余的存储起来。

<ip_addr>：远端 IP 地址

<port>：远端端口，范围为 0-65535。

<length>：接收到的数据的十进制 byte 长度

<remaining_length>：剩余未读取的数据的长度

<data>：接收到的十六进制格式的数据

<length_next>：下一个数据节点的数据长度

典型示例

+NSONMI:0,3//第一个下行数据，数据上报模式为 1 时

AT+NSORF=0,1

+NSORF:0,139.224.112.6,10005,1,68,2

OK

AT+NSORF=0,1

+NSORF:0,139.224.112.6,10005,1,74,1

OK

AT+NSORF=0,1

+NSORF:0,139.224.112.6,10005,1,74,20//读完第一个节点的数据后，此长度为第二个数据节点的数据长度

OK

+NSONMI:0,20//直到读完上一个数据节点的数据才上报第二个数据节点的数据信息，数据上报模式为 1 时

```
AT+NSORF=0,1
+NSORF:0,139.224.112.6,10005,1,70,19
OK
```

备注：仅支持 IPv4。

4.10 关闭服务+NSOCL

语法结构

命令	响应
AT+NSOCL=<socket>	OK +NSOCLl:<socket_id>
	+CME ERROR:<err>

接口说明

该指令用于关闭指定的链路。

取值说明

<socket>：由创建 socket 时决定的 socket_id

<socket_id>：关闭 socket 链接时，上报关闭链路

<err>：错误码，参看 err 章节。

4.11 注意事项

目前平台内部会自动进行 IP 地址的获取，主动上报 "+CGEV:PDN ACT" 和 "+CGEV:PDN DEACT"，分别表示获取到 IP 地址或 IP 地址无效，用户需要监听这两个主动上报，以进行自身的 AT 命令流程设计有限，仅支持 2 个同时工作，且内部释放 Socket 资源需要一定时间，建议在使用 UDP/TCP Socket 做完业务之后再执行关闭操作，避免短时间内重复创建/收发/关闭 Socket，以免 Socket 资源短时间内耗尽导致 +NSOCR 再次创建时失败；

当连续快速输入待发送数据且速度高于底层发送速度时，可能引起发送缓冲区满而引起发送错误，请间隔数秒后再次发送数据；

模组内部可缓存一定接收数据量，当有数据到达且为手动接收模式时，需及时读取数据，否则可能引起后续到达数据丢失；当需要连续接收大量数据时，建议将波特率调高以加快接收速度。

对于数据传输的扩展 AT 命令，目前不建议用户使用 ASCII 模式进行内容传递，因为无法控制 ASCII 字符串中的特殊字符，如 ""，等标识符，容易出异常。建议用户使用十六进制 ASCII 码来描述 ASCII 字符串。

4.12 典型示例

```
AT+CGATT?           //查询是否已附着，即获取到外网 IP 地址
+CGATT:1           //1 表示已附着；若为 0，需要循环查询
OK
AT+NSOCR=DGRAM,17,10005 //创建 UDP 的 socket 连接
0
OK
AT+NSOST=0,139.224.112.6,10005,5,1234567890 //发送 UDP 数据报文
```

```
OK
+NSONMI:0,4          //主动上报命令, 表示有 4 字节的数据上报
AT+NSORF=0,4        //接受 4 字节的数据
0,139.224.112.6,10005,4,1234ABCD,0
OK
AT+NSOCL=0          //关闭创建的 socket 连接
OK
+NSOCLI:0

AT+WORKLOCK=0
OK
+POWERDOWN: 5360    //主动上报命令, 表示芯片已经进入深睡,
```

5 天翼云扩展 AT

5.1 CDP 设置与查询+NCDP

语法结构

命令	响应
AT+NCDP=<ip_addr>[,<port>]	OK
	+CME ERROR:<err>
AT+NCDP?	+NCDP:<ip_addr>,<port>
	OK
	+CME ERROR:<err>

接口说明

用于设置和查询CDP (Connected Device Platform) 服务器的IP地址和端口。

AT+NCDP=<ip_addr>[,<port>],

取值说明

<ip_addr>: IPv4 地址, IP 地址是以点分格式 (0-255) 。

<port>: 端口号, 无符号整数, 范围是 0-65535。若未指定端口号则使用先前设置端口。默认值 0, 使用默认端口 (5683) 。

典型示例

```
AT+NCDP=192.168.5.1
OK
AT+CFUN=0
OK
+CGEV:ME PDN DEACT 0
+CGEV:ME DETACH

AT+CFUN=1
OK
^SIMST:1
+CEREG:2,,
+UICCTYPE:2
+CGEV:ME PDN ACT 0
+XYIPDNS:1,0,"IP","100.67.43.102","", "211.136.17.107","211.136.20.203"
+CEREG:1,3A05,099AEDB7,9
+CTZEU:+32,0,2019/10/12,07:01:19
AT+NCDP?
+NCDP:192.168.5.1,5683
OK
```

备注

- 仅支持 IPv4。
- 设置值掉电保存。
- 在设置 IMEI 号后, 此命令可用。
- 当发送失败时, 必须执行命令 (AT+CFUN=0) 将强制模组进入此状态。

- IP 地址可用十进制、八进制或十六进制。
- 设置更改将在附着成功之前生效。
- 建议使用+NRB 命令保存 CDP 服务器设置。

5.2 注册去注册+QLWSREGIND

语法结构

命令	响应
AT+QLWSREGIND=<type>	OK
	+CME ERROR:<err>

接口说明

用于注册/去注册 CDP 平台，仅当配置为手动注册时使用。当配置为自动注册时，无需使用该条 AT 命令。目前平台不支持自动注册，必须使用该条命令手动进行注册/去注册。

取值说明

- <type>: 注册/去注册平台参数
- 0 表示注册 CDP 平台；
 - 1 表示去注册 CDP 平台；

典型实例

```
AT+QLWSREGIND=0
OK
```

5.3 注册状态主动上报-+QLWEVTIND

语法结构

命令	响应
	+QLWEVTIND:<type>

接口说明

注册状态主动上报

取值说明

- <type>: 状态指示值
- 0 Register completed
 - 1 Deregister completed
 - 2 Registration status updated
 - 3 Object 19/0/0 observe completed
 - 4 Bootstrap completed
 - 5 5/0/3 resource observe completed
 - 6 Notify the device to receive update package URL
 - 7 Notify the device download has been completed
 - 8 Register timeout
 - 9 Cancel object 19/0/0 observe

典型实例

```
AT+QLWSREGIND=0
OK
```

```
+QLWEVTIND:0
+QLWEVTIND:3
AT+QLWSREGIND=1
OK
+QLWEVTIND:1
```

5.4 注册模式+QREGSWT

语法结构

命令	响应
AT+QREGSWT=<type>	OK
	+CME ERROR:<err>
AT+QREGSWT?	+QREGSWT:<type>

接口说明

该命令用于配置 CDP 平台的注册模式。当设置为手动模式时，需要通过+QLWSREGIND 进行手动的注册/去注册。目前平台不支持自动注册，必须使用+QLWSREGIND 命令手动进行注册/去注册。

取值说明

<type>>: 注册模式
 0 表示手动注册;
 1 表示自动注册(不支持);

典型实例

```
AT+QREGSWT=0
OK
```

5.5 发送信息+NMGS

语法结构

命令	响应
AT+NMGS=<length>,<data>	OK
	+CME ERROR:<err>

接口说明

通过 CDP 服务器将信息从终端发送到网络。

取值说明

<length>: 信息的十进制长度，目前支持单次发送数据最大长度 1000 字节。
 <data>: 要传输的数据，以十六进制字符串格式表示

典型示例

```
AT+NMGS=3,AA11BB
OK
```

备注

- 1.最大数据长度为 1000 字节。
- 2.同一时刻只能缓冲八条消息（即收到下行数据前，最多缓存八个消息）。

5.6 发送信息+NMGSEXT

语法结构

命令	响应
AT+NMGSEXT=<msg_type><length>,<data>	OK +CME ERROR:<err>

接口说明

通过 CDP 服务器将信息从终端发送到网络。

取值说明

<msg_type>: 信息数据包格式。

0 为 CON,

1 为 NON

<length>: 信息的十进制长度, 最大支持 1000 字节。

<data>: 要传输的数据, 以字符串格式表示。

典型示例

```
AT+NMGS=0,3,"A12"
OK
```

备注

- 1.最大数据长度为 1000 字节。
- 2.同一时刻只能缓冲八条消息。

5.7 发送信息指示+NSMI

语法结构

命令	响应
AT+NSMI=<status>	OK +CME ERROR:<err>
AT+NSMI?	+NSMI:<status> OK
	+NSMI:<result>

接口说明

该命令设置是否开启上行数据发送的状态主动上报。若开启后, 当发送完上行数据, 收到云平台的确
 认消息后, 会上报"+NSMI:<status>[,<seq_num>]"主动上报给外部。其中, <seq_num>与 AT+NMGS
 命令中的可选参数保持一致, 若没有, 则无需携带。

取值说明

<status>: 0 表示不指示; 1 表示主动上报指示

<result>: 发送状态

SENT: 发送成功

DISCARDED: 发送失败

典型示例

```
AT+NSMI=1
OK
AT+NSMI?
+NSMI:1
OK
```

5.8 接收信息+NMGR

语法结构

命令	响应
AT+NMGR	+NMGR:<length>,<data> OK
	+CME ERROR:<err>

接口说明

从 CDP 服务器接收信息。

该命令返回最早的缓冲信息，并从缓冲区中删除它们，若无信息则无响应。若打开了新信息指示 (AT+NNMI)，则通过该命令将无法接收到信息。

取值说明

<length>: 数据长度，十进制格式，最大支持 1000 字节

<data>: 接收到的数据，以十六进制字符串格式表示

典型示例

```
AT+NMGR
+NMGR:5,48656C6C6F
OK
```

备注

- 1.最大数据长度为 1000 字节。
- 2.同一时刻只能缓冲八条消息。

5.9 新信息指示+NNMI

语法结构

命令	响应
AT+NNMI=<status>	OK
	+CME ERROR:<err>
AT+NNMI?	+NNMI:<status> OK
	+NNMI:<length>,<data>

接口说明

该命令设置或获取发送的新信息的指示。

取值说明

<status>: 是否主动上报

0 不上报, 默认值

1 收到下行数据后主动上报数据长度和内容, 上报+NNMI:<length>,<data>

2 收到下行数据后, 主动上报+NNMI

<length>: 信息的十进制长度

<data>: 要传输的数据, 以十六进制字符串格式表示

典型示例

```
AT+NNMI=1
OK
AT+NNMI?
+NNMI:1
OK
```

5.10 查询已发送信息+NQMGS

语法结构

命令	响应
AT+NQMGS	+NQMGS:PENDING=<pending>,SENT=<sent>,ERROR=<error> OK +CME ERROR:<err>

接口说明

该命令用于查询发送给 CDP 服务器的上游信息的状态。

取值说明

<pending>: 当已注册完成+QLWEVTIND:3, 在上游缓冲区中等待发送的信息数

<sent>: 自终端启动以来发送到 NB-IoT 堆栈的上行链路信息总数

<error>: 终端启动后由于错误而无法由终端发送的信息数

典型示例

```
AT+NQMGS
+NQMGS:PENDING=1,SENT=34,ERROR=0
OK
```

5.11 查询已接收信息+NQMGR

语法结构

命令	响应
AT+NQMGR	+NQMGR:BUFFERED=<buffered>,RECEIVED=<received>,DROPPED=<dropped> OK +CME ERROR:<err>

接口说明

该命令用于查询从 CDP 服务器接收的下游信息的状态。

取值说明

- <buffered>: 在下游缓冲区中等待读取的信息数
- <received>: 自终端启动以来终端接收的信息总数
- <dropped>: 终端启动后终端丢弃的信息数

典型示例

```
AT+NQMGR
+NQMGR:BUFFERED=0,RECEIVED=34,DROPPED=2
OK
```

5.12 注意事项

目前平台内部会自动进行 IP 地址的获取，主动上报 “+CGEV:ME PDN ACT” 表示获取到 IP 地址和 “+CGEV:ME PDN DEACT” 表示 IP 地址无效，用户需要监听这两个主动上报，以进行自身的 AT 命令流程设计。

5.13 典型示例

```
AT+CGATT?           //查询是否已附着，即获取到外网 IP 地址
+CGATT:1           //1 表示已附着；若为 0，需要循环查询
OK
AT+NNMI=2          //打开新数据上报开关，当有新数据接收时，主动上报+NNMI
OK
AT+NMGS=16,00000200200200202020200300AA11BB           //发送数据
OK
+NNMI              //主动上报命令，表示收到下行数据
AT+NMGR            //读取数据
+NMGR:5,23656D5C6F
OK
AT+NQMGR           //查询接收的数据情况
+NQMGR:BUFFERED=0,RECEIVED=5,DROPPED=0
OK
AT+NQMGS           //查询发送的数据情况
+NQMGS:PENDING=1,SENT=5,ERROR=0
OK
AT+WORKLOCK=0      //深睡省电
OK
+POWERDOWN:84481   //主动上报命令，表示芯片已经进入深睡
```

6 Onenet 扩展 AT

6.1 创建 ONENET 通信套件实例+MIPLCREATE

语法结构

命令	响应
AT+MIPLCREATE=<totalsize>,<config>,<index>,<currentsize>,<flag>	+MIPLCREATE:<ref> OK
	+CIS ERROR:<errid>
AT+MIPLCREATE	+MIPLCREATE:<ref> OK
	+CIS ERROR:<errid>

接口说明

该指令用于创建一个基础通信套件的实例。

设置命令是设置配置文件，该配置文件存储在 NV 中，芯片出厂时 NV 为空，因此第一次使用时必须先使用设置命令设置配置文件，第二次使用时候如果仍使用上次的配置文件，可以直接使用

“AT+MIPLCREATE”命令创建实例，**配置文件需要用户自己生成**，详情见《海凌科 HLK-N10 onenet 中移云开发及调试指南》。

取值说明

<totalsize>: config 文件的总长度。

<config>: 配置文件，参见配置文件格式。

<index>: 配置文件的序号，考虑到 AT 指令长度有限，一个完整的配置文件未必能在一条 AT 指令中发送完成，可以将内容切分成多段，比如分为 N 段，则从前到后按照降序依次分配序号为 N-1~0，按照从大到小序号的顺序每段调用一次 AT 指令，因此当 index 为 0 时该条指令为最后一条配置消息。

<currentsize>: 当前指令所包含的配置文件长度。

<flag>: 消息标识

0: 最后一条消息

1: 第一条消息

2: 中间消息

<ref>: 消息正确接收返回一个创建完成的基础通信套件的一个实例标识，类型为一个无符号整数。

注意: 如果发送的是 index 不为 0 的消息，回复 OK 代表消息正确接收，只有发送<index>=0 的消息才会返回<ref>参数，该回复代表消息正确接收

<errid>: 返回错误，取值如下

601 参数错误

602 状态错误

100 未知错误

典型示例

```
AT+MIPLCREATE=83,130053f10003f20045040011800005434d494f54000000000123138332e32
33302e34302e33393a35363833001d41757468436f64653a77616e6768616e3b50534b3a77616e67
68616e3bf30008e400c80000,0,83,0
+MIPLCREATE:0
OK

AT+NV=SAVE
REBOOTING
```

6.2 删除 ONENET 通信套件实例+MIPLDELETE

语法结构

命令	响应
AT+MIPLDELETE=<ref>	OK
	+CIS ERROR:<errid>

接口说明

该指令用于删除一个基础通信套件的实例

取值说明

<ref>: 基础通信套件的一个实例标识, 类型为一个无符号整数

<errid>: 返回错误, 取值如下

601 参数错误

602 状态错误

100 未知错误

典型示例

```
AT+MIPLDELETE=0
OK
```

6.3 增加 LwM2M 实例+MIPLADDOBJ

语法结构

命令	响应
AT+MIPLADDOBJ=<ref>,<objectid>,<instancecount> ,<instancebitmap>,<attributecount>,<actioncount>	OK
	+CIS ERROR:<errid>

接口说明

该指令用于添加 LwM2M 的一个 object

取值说明

<ref>: 基础通信套件的一个实例标识, 无符号整型

<objectid>: 对象 id

<instancecount>: 实例个数

<instancebitmap>: 实例位图, 字符串格式, 每一个字符表示为一个实例, 其中 1 表示可用, 0 表示不可用。例如当前添加的 object 有 5 个实例, 其中, 1, 3 可用, 则实例位图为 00101

<attributecount>: 属性个数

<actioncount>: 操作个数

<errid>: 返回错误, 取值如下

601 参数错误

602 状态错误

100 未知错误

典型示例

```
AT+MIPLADDOBJ=0,3311,2,"11",4,3
OK
```

6.4 删除 LwM2M 实例+MIPLDELOBJ

语法结构

命令	响应
AT+MIPLDELOBJ=<ref>,<objectid>	OK
	+CIS ERROR:<errid>

接口说明

该指令用于删除 LwM2M 的一个 object

取值说明

<ref>: 基础通信套件的一个实例标识, 类型为一个无符号整数

<objectid>: 对象 id

<errid>: 返回错误, 取值如下

601 参数错误

602 状态错误

100 未知错误

典型示例

```
AT+MIPLCLOSE=0
OK
```

6.5 发起注册请求+MIPLOPEN

语法结构

命令	响应
AT+MIPLOPEN=<ref>,<lifetime>[,<timeout>]	OK
	+CIS ERROR:<errid>

接口说明

该指令向平台发起注册请求。

取值说明

<ref>: 基础通信套件的一个实例标识, 类型为一个无符号整数

- <lifetime>: 生命周期, 单位为秒 (最短 120s)
- <timeout>: 注册的超时时长, 单位为 s (该功能开发中)。
- <errid>: 返回错误, 取值如下
 - 601 参数错误
 - 602 状态错误
 - 100 未知错误

典型示例

```
AT+MIPLOPEN=0,3000,30
OK
```

6.6 发送注销请求+MIPLCLOSE

语法结构

命令	响应
AT+MIPLCLOSE=<ref>	OK +MIPLEVENT:0,15
	+CIS ERROR:<errid>

接口说明

该指令向平台发起注销请求

取值说明

- <ref>: 基础通信套件的一个标识, 无符号整型
- <errid>: 返回错误, 取值如下
 - 601 参数错误
 - 602 状态错误
 - 100 未知错误

典型示例

```
AT+MIPLCLOSE=0
OK
```

6.7 Discover 请求的响应+MIPLDISCOVERRSP

语法结构

命令	响应
AT+MIPLDISCOVERRSP=<ref>,<msgid>,<result>,<length>,<value string>	OK
	+CIS ERROR:<errid>

接口说明

该指令设置基础通信套件获取到的指定 object 的所有属性。

取值说明

<ref>: 基础通信套件的一个标识, 无符号整型

<msgid>: 消息 id

<result>: 发现操作的结果; 可返回代码如下:

- 1 2.05 Content 操作正确完成
- 11 4.00 Bad Request
- 12 4.01 Unauthorized
- 13 4.04 Not Found
- 14 4.05 Method Not Allowed
- 15 4.06 Not Acceptable

<length>: 返回 valuestring 的长度

<valuestring>: object 的属性要求, 多个属性放置双引号""中, 使用分号";" 隔开"5850;5851;5706;5805"

<errid>: 返回错误, 取值如下

- 601 参数错误
- 602 状态错误
- 100 未知错误

典型示例

```
AT+MIPLDISCOVERRSP=0,7160,1,19,"5850;5851;5706;5805"
OK
```

6.8 主动上报 Discover 请求+MIPLDISCOVER

语法结构

命令	响应
	+MIPLDISCOVER:<ref>,<msgid>,<objectid>

接口说明

该指令通知 MCU 需要获取指定 object 的属性。

取值说明

<ref>: 基础通信套件的标识, 无符号整型

<msgid>: 消息 id

<object>: 指定的 object 对象 id

典型示例

```
+MIPLDISCOVER:0,7160,3311
```

6.9 Observe 请求的响应+MIPOBSERVERSP

语法结构

命令	响应
AT+MIPOBSERVERSP=<ref>,<msgid>,<result>	OK
	+CIS ERROR:<errid>

接口说明

该指令通知基础通信套件观测指令是否有效。当应用程序收到+MIPLOBSEVERSP 主动上报消息后，需要去验证该请求是否有效。调用该命令时将 msgid 置为对应+MIPLOBSEVERSP 消息携带的 msgid。该功能也包括 cancelobserve 的回复。

取值说明

<ref>: 基础通信套件标识, 无符号整型

<msgid>: 消息 id

<result>: 观测消息或者取消观察消息是否成功

- 1 2.05Content 操作正确完成
- 11 4.00 Bad Request
- 12 4.01 Unauthorized
- 13 4.04 Not Found
- 14 4.05 Method Not Allowed
- 15 4.06 Not Acceptable

<errid>: 返回错误, 取值如下

- 601 参数错误
- 602 状态错误
- 100 未知错误

典型示例

```
AT+MIPLOBSEVERSP=0,95545,1
OK
```

6.10 主动上报 Observe 请求+MIPLOBSEVERSP

语法结构

命令	响应
	+MIPLOBSEVERSP:<ref>,<msgid>,<flag>,<objectid>,<instanceid>[,<resourceid>]

接口说明

该指令是一个主动上报观测请求消息。

取值说明

<ref>: 基础通信套件的标识, 固定为 0。

<msgid>: 消息 id, 标识该观测请求消息

<flag>:

- 1 为添加观测,
- 0 为取消观测

<objectid>: 对象 id

<instanceid>: 实例 id, 如果为 '-1', 则代表观测该 object 下所有 instance 下的所有资源

<resourceid>: 资源 id, 如果为 '-1', 则代表观测该 instance 下的所有资源返回值

典型示例

```
+MIPLOBSEVERSP:0,95545,1,3311,0,-1
```

6.11 Read 请求的响应+MIPLREADRSP

语法结构

命令	响应
AT+MIPLREADRSP=<ref>,<msgid>,<result>[,<objectid>,<instanceid>,<resourceid>,<valuetype>,<len>,<value>,<index>,<flag>]	OK +CIS ERROR:<errid>

接口说明

该指令通知基础通信套件一个属性变化。

取值说明

<ref>: 基础通信套件的一个标识, 固定为 0.

<msgid>: 消息 id

<result>: 写入操作的结果; 可返回代码如下:

- 2: 2.04 Changed 写操作正确完成
- 11: 4.00 Bad Request
- 12: 4.01 Unauthorized
- 13: 4.04 Not Found
- 14: 4.05 Method Not Allowed

<objectid>: 对象 id

<instanceid>: 实例 id

<resourceid>: 资源 id

<valuetype>: 发送的数据类型

<len>: 上报的数据长度, 单位字节

<value>: 发送的数据值

<index>: 指令序号。若某个 Notify 操作需要 N 条消息组合为一完整指令, 则 index 从 N-1 至 0 降序编号, 当 index 编号为 0 时表示本次 Notify 指令结束。

<flag>: 消息标识

- 0: 最后一条消息
- 1: 第一条消息
- 2: 中间消息

发送的数据类型要求如下:

string=1, //字符串型, <value>加双引号, 如 "abc"; <length>参数为字符串长度, 即双引号内字符串长度, 不包括双引号, 长度小于 1400。

opaque=2, //不透明类型, <value>不加双引号, <length>参数为字节长度, 长度小于 160。

integer=3, //整型, <length>为该整型占用的字节数, 可选 2 字节、4 字节、8 字节

float=4, //浮点型, <length>为该 float 占用的字节数, 当前可选 4 字节和 8 字节

bool=5, //布尔型, 0 代表 false, 1 代表 true, <length>参数为 1。

举例如下:

如果是 string 类型的 value, 则该指令如下

AT+MIPLREADRSP=<ref>,<msgid>,<result>[,<objectid>,<instanceid>,<resourceid>,<index>,<flag>]<CR>

如果是 opaque 类型的 value, 则该指令如下

AT+MIPLREADRSP=<ref>,<msgid>,<result>[,<objectid>,<instanceid>,<resourceid>,<index>,<flag>]<CR>

如果是 interger 类型的 value, 则该指令如下

AT+MIPLREADRSP=<ref>,<msgid>,<result>[,<objectid>,<instanceid>,<resourceid>,<index>,<flag>]<CR>

如果是 float 类型的 value, 则该指令如下

AT+MIPLREADRSP=<ref>,<msgid>,<result>[,<objectid>,<instanceid>,<resourceid>,<index>,<flag>]<CR>

如果是 bool 类型的 value, 则该指令如下

AT+MIPLREADRSP=<ref>,<msgid>,<result>[,<objectid>,<instanceid>,<resourceid>,<index>,<flag>]<CR>

<ackid>: MCU 指定该消息以 CON 形式上报, 如果该条 Notify 消息成功, 则基础通信套件会收到 +MIPLEVENT 返回。当所有 inst_id 要 notify 时, 一个实例对应一个 ackid, 不能相等, 当一个 instance 下的所有 res_id 都要 notify 时, 该 instance 下的 res_id 对应一个 ackid, 当指定 inst_id 和 res_id 则对应一个 ackid, 短时间内 ackid 不能相同。

<errid>: 返回错误, 取值如下

- 601 参数错误
- 602 状态错误
- 100 未知错误

典型示例

AT+MIPLREADRSP=0,64927,1,3311,0,5851,3,2,800,0,0
 OK

备注

如果对于同一条消息, 只有一条 notify 指令, 则 index 和 flag 都为 0。一组上报消息最多只能上报一个 instance 的信息, 如果需要上报多个 instance 的信息, 需要按照 instance 对消息进行分组上报。

6.12 主动上报 Read 请求+MIPLREAD

语法结构

命令	响应
	+MIPLREAD:<ref>,<msgid>,<objectid>,<instanceid>,<resourceid>

接口说明

该指令主动上报 read 请求信息。

取值说明

- <ref>: 基础通信套件的标识, 无符号整型
- <msgid>: 消息 id, 标识该读取请求消息
- <objectid>: 对象 id
- <instanceid>: 实例 id, 如果为"-1",则代表需要读取该 object 下的所有资源

<resourceid>: 资源 id, 如果为 '-1', 则代表需要读取该 instance 下的所有资源返回值

典型示例

+MIPLREAD:0,19397,3311,0,5851

6.13 Write 请求的响应+MIPLWRITERSP

语法结构

命令	响应
AT+MIPLWRITERSP=<ref>,<msgid>,<result>	OK
	+CIS ERROR:<errid>

接口说明

该指令通知基础通信套件写入的消息结果。用户接到+MIPLWRITE 消息后, 需要去写入需要的资源值, 同时使用该消息通知基础通信套件写入的结果。调用该命令时 msgid 值为对应+MIPLWRITE 消息携带的 msgid。

取值说明

<ref>: 基础通信套件的一个标识, 类型为一个无符号整数。

<msgid>: 消息 id。

<result>: 写入操作的结果;

- 2 2.04Changed 写操作正确完成
- 11 4.00 Bad Request
- 12 4.01 Unauthorized
- 13 4.04 Not Found
- 14 4.05 Method Not Allowed

<errid>: 返回错误, 取值如下

- 601 参数错误
- 602 状态错误
- 100 未知错误

典型示例

AT+MIPLWRITERSP=0,58067,2
 OK

备注: 同一个消息 id 的 write 返回只需要一次结果返回。

6.14 主动上报 Write 请求+MIPLWRITE

语法结构

命令	响应
	+MIPLWRITE:<ref>,<msgid>,<objectid>,<instanceid>,<resourceid>,<valuetype>,<len>,<value>,<flag>,<index>

接口说明

该指令是一个写操作请求消息的主动上报。

取值说明

- <ref>: 基础通信套件的标识, 无符号整型
- <msgid>: 消息 id, 标识该写入请求消息
- <objectid>: 对象 id
- <instanceid>: 实例 id
- <resourceid>: 资源 id
- <valuetype>: 待写入的数据类型, 要求如下:
 - string =1, //字符串型, value 内容将会使用 "" 进行传递
 - opaque=2, //不透明类型
 - integer=3, //整型
 - float=4, //浮点型
 - bool=5, //布尔型
- <len>: 待写入的数据长度
- <value>: 待写入的数据值
- <flag>: 如果一条写入的消息 value 需要分多条指令上报, 则消息标识需要根据当前的位置填写
 - 0: 最后一条消息
 - 1: 第一条消息
 - 2: 中间消息
- <index>: 写操作的序号, 从大到小递减, 比如 N, N-1, ..., 0, 最后一条的时候为 0。

典型示例

+MIPLWRITE:0,58067,3311,0,5851,2,1,05,0,0

备注: 如果该 msgid 对应的消息只有一条, 且不分条, 则<flag>和<value>都为 0。

6.15 Execute 请求的响应+MIPLEXECUTERSP

语法结构

命令	响应
AT+MIPLEXECUTERSP=<ref>,<msgid>,<result>	OK
	+CIS ERROR:<errid>

接口说明

用户接到+MIPLEXECUTE 消息后, 需要去执行请求的动作, 同时使用该消息通知基础通信套件执行的结果。调用该命令时将 msgid 置为对应+MIPLEXECUTE 消息携带的 msgid。

取值说明

- <ref>: 基础通信套件的标识, 无符号整型。
- <msgid>: 消息 id。
- <result>: 执行操作的结果;
 - 2 2.04Changed 执行操作正确完成
 - 11 4.00 Bad Request
 - 12 4.01 Unauthorized

- 13 4.04 Not Found
- 14 4.05 Method Not Allowed

<errid>: 返回错误, 取值如下

- 601 参数错误
- 602 状态错误
- 100 未知错误

6.16 主动上报 Execute 请求+MIPLEXECUTE

语法结构

命令	响应
	+MIPLEXECUTE:<ref>,<msgid>,<objectid>,<instanceid>,<resourceid>[,<len>,<arguments>]

接口说明

该指令是一个执行操作请求消息主动上报。

取值说明

- <ref>: 基础通信套件的标识, 无符号整型
- <msgid>: 消息 id, 标识该执行请求消息
- <objectid>: 对象 id
- <instanceid>: 实例 id
- <resourceid>: 资源 id
- <len>: 参数的字符长度
- <arguments>: 执行的字符串格式参数

6.17 Write 属性请求的响应+MIPLPARAMETERRSP

语法结构

命令	响应
AT+MIPLPARAMETERRSP=<ref>,<msgid>,<result>	OK
	+CIS ERROR:<errid>

接口说明

用户接到+MIPLPARAMETER 消息后, 需要去执行请求的动作, 同时使用该消息通知基础通信套件执行的结果。调用该命令时 msgid 值为对应+MIPLPARAMETER 消息携带的 msgid。

取值说明

- <ref>: 基础通信套件的标识, 无符号整型
- <msgid>: 消息 id
- <result>: 设置订阅参数操作的结果;
 - 2 2.04Changed 操作正确完成
 - 11 4.00 Bad Request

- 12 4.01 Unauthorized
- 13 4.04 Not Found
- 14 4.05 Method Not Allowed

<errid>: 返回错误, 取值如下

- 601 参数错误
- 602 状态错误
- 100 未知错误

6.18 主动上报 Write 属性请求+MIPLPARAMETER

语法结构

命令	响应
	+MIPLPARAMETER:<ref>,<msgid>,<objectid>,<instanceid>,<resourceid>,<len>,<parameter>

接口说明

该指令是一个设置策略参数请求消息主动上报。

取值说明

<ref>: 基础通信套件的标识, 无符号整型

<msgid>: 消息 id, 标识该观测请求消息

<objectid>: 对象 id

<instanceid>: 实例 id, 如果为 '-1', 则代表观测该 object 下所有 instance 下的所有资源

<resourceid>: 资源 id, 如果为 '-1', 则代表观测该 instance 下的所有资源

<len>: 参数长度

<parameter>: 策略参数, 格式为字符串形式, 包括如下策略:

pmin=xxx; pmax=xxx; gt=xxx; lt=xxx; stp=xxx

6.19 Notify 数据到 ONENET 云端+MIPLNOTIFY

语法结构

命令	响应
AT+MIPLNOTIFY=<ref>,<msgid>,<objectid>,<instanceid>,<resourceid>,<valuetype>,<len>,<value>,<index>,<flag>[,<ackid>]	OK
	+CIS ERROR:<errid>

接口说明

该指令通知基础通信套件一个数值变化。

取值说明

<ref>: 基础通信套件的标识, 无符号整型

<msgid>: 消息 id

<objectid>: 对象 id

<instanceid>: 实例 id, -1 表示 notify 所有的 instance 资源

<resourceid>: 资源 id, -1 表示 notify 选定的 instance 下的所有资源

<valuetype>: 发送的数据类型

发送的数据类型要求如下:

string =1, //字符串型, <value>两端加双引号, 如 "abc"; <length>参数为字符串长度, 即双引号内字符串长度, 长度小于 1400, 不包括双引号。

opaque=2, //不透明类型, 十六进制 ASCII 数据, <value>两端不加双引号, <length>参数为字节长度, 长度小于 160。

integer=3, //整型, <length>为该整型占用的字节数, 可选 2 字节、4 字节、8 字节

float=4, //浮点型, <length>为该 float 占用的字节数, 当前可选 4 字节和 8 字节

bool=5, //布尔型, 0 代表 false, 1 代表 true, <length>参数为 1。

<len>: 上报的数据长度

<value>: 发送的数据值

<index>: 指令序号。若某个 Notify 操作中, 有 M 个 instance, 每个 instance 有 N 条消息组合为一完整指令, 则 index 从 N-1 至 0 降序编号, 当 index 编号为 0 时表示本次 Notify 指令结束, 执行 M 次。

<flag>: 消息标识

1: 第一条消息

2: 中间消息

0: 最后一条消息

<result>: 观测消息或者取消观察消息是否成功

1 2.05Content 操作正确完成

11 4.00 Bad Request

12 4.01 Unauthorized

13 4.04 Not Found

14 4.05 Method Not Allowed

15 4.06 Not Acceptable

<ackid>: MCU 指定该消息以 CON 形式上报, 如果该条 Notify 消息成功, 则基础通信套件会将收到的返回, 用+MIPLEVENT 返回。当所有 inst_id 要 notify 时, 一个实例对应一个 ackid, 不能相等, 当一个 instance 下的所有 res_id 都要 notify 时, 该 instance 下的 res_id 对应一个 ackid, 当指定 inst_id 和 res_id 则对应一个 ackid, 短时间内 ackid 不能相同。

<errid>: 返回错误, 取值如下

601 参数错误

602 状态错误

100 未知错误

典型示例

```
AT+MIPLNOTIFY=0,116055,3311,0,5850,5,1,0,3,1
OK
```

备注

如果对于同一条消息, 只有一条 notify 指令, 则 index 和 flag 都为 0, 一组上报消息最多只能上报一个 instance 的信息, 如果需要上报多个 instance 的信息, 需要按照 instance 对消息进行分组上报。

6.20 发送更新请求+MIPLUPDATE

语法结构

命令	响应
AT+MIPLUPDATE=<ref>,<lifetime>,<withObjectFlag>	OK +CIS ERROR:<errid>

接口说明

该指令通知基础通信套件发送主动更新注册信息。

取值说明

<ref>: 基础通信套件的标识, 无符号整型

<lifetime>: 更新的 lifetime 值, 单位为 s, 如果为 0 则表示使用默认的 lifetime 值, 默认为 24 小时。
 (最短 120s)

<withObjectFlag>: 是否需要同时更新注册的 Object 对象

<errid>: 返回错误, 取值如下

601 参数错误

602 状态错误

100 未知错误

典型示例

```
AT+MIPLUPDATE=0,3000,0
OK
```

6.21 查询当前 ONENET 通信套件版本+MIPLVER

语法结构

命令	响应
AT+MIPLVER?	+MIPLVER:2.3.0 OK +CIS ERROR:<errid>

接口说明

该指令获取基础通信套件版本。

取值说明

+MIPLVER: 基础通信套件的版本号, 如 “10.01”

<errid>返回错误, 取值如下

601 参: 数错误

602 状态错误

100 未知错误

典型示例

```
AT+MIPLVER?
+MIPLVER:2.3.0
OK
```

6.22 主动上报 EVENT 事件通知+MIPLEVENT

语法结构

命令	响应
	+MIPLEVENT:<ref>,<evtid>[,<extend>,<ackid>]

接口说明

该指令上报一个状态事件。

取值说明

<ref>: 基础通信套件的标识, 无符号整型

<evtid>: 事件 id, 十进制显示。

<ackid>: notify 成功时, 回复的 ackid。

<extend>: 扩展参数, CIS_EVENT_RESPONSE_FAILED 和 CIS_EVENT_RESPONSE_FAILED 事件时会携带该对应指令的 msgid; CIS_EVENT_UPDATE_NEED 事件时会携带 LIFETIME 剩余时间(单位秒), 详见《海凌科 HLK-N10 业务平台接口说明》文档。

CIS_EVENT_BASE	Ref,(cis_evt_t)0x00
CIS_EVENT_BOOTSTRAP_START	0,1
CIS_EVENT_BOOTSTRAP_SUCCESS	0,2
CIS_EVENT_BOOTSTRAP_FAILED	0,3
CIS_EVENT_CONNECT_SUCCESS	0,4
CIS_EVENT_CONNECT_FAILED	0,5
CIS_EVENT_REG_SUCCESS	0,6
CIS_EVENT_REG_FAILED	0,7
CIS_EVENT_REG_TIMEOUT	0,8
CIS_EVENT_LIFETIME_TIMEOUT	0,9
CIS_EVENT_STATUS_HALT	0,10
CIS_EVENT_UPDATE_SUCCESS	0,11
CIS_EVENT_UPDATE_FAILED	0,12
CIS_EVENT_UPDATE_TIMEOUT	0,13
CIS_EVENT_UPDATE_NEED	0,14
CIS_EVENT_UNREG_DONE	0,15
CIS_EVENT_RESPONSE_FAILED	0,20
CIS_EVENT_RESPONSE_SUCCESS	0,21
CIS_EVENT_NOTIFY_FAILED	0,25
CIS_EVENT_NOTIFY_SUCCESS	0,26

典型示例

+MIPLEVENT:0,20,204313

6.23 注意事项

对于数据传输的扩展 AT 命令, 目前不建议用户使用 ASCII 模式进行内容传递。如果用户使用 ASCII 格式数据, 则必须用 "" 把 ASCII 字符串包起来, 以避免内部包含的特殊字符。

7 平台工具类扩展 AT

7.1 Ping 包指令+NPING

语法结构

命令	响应
AT+NPING=<host>,<data_len>,<ping_num>,<time_out>,<interval_time>[,<rai>]	OK +CME ERROR:<err>

接口说明

底板 MCU 发送数据请求，最终通过 NB 协议栈发送给远端服务器。若需要进行 RAI=1 的上行数据发送完毕的释放测试，host 地址随便写一个，确保目标 IP 地址不可达即可，进而也就没有下行应答数据回复了。

若需要在 PING 过程中停止 PING 包，则输入“AT+NPINGSTOP”即可停止，ping 包业务没完成，不可再次发起 ping 包业务。

取值说明

<host>: 目标 ip 地址

<data_len>: icmp 数据长度，最大为 1400 字节

<ping_num>: ping 的次数，单位为字节

<time_out>: 等待 ping 应答结果的超时时间，超时后判为无应答，若为 0，则不会发起 ping 业务，返回 8001 参数错误。

<interval_time>: 两次 ping 的间隔时间，

<rai>: 可选参数，默认 0。

0 表示缺省；

1 表示发送完上行报文后即可释放链接；

2 表示发送完下行报文后即可释放链接；

典型示例

AT+NPING=139.224.112.6,64,4,20,21

OK

ADMIN_TRACE_INFO	ADMIN_TRACE_LEVEL_1	ADMIN	PS_OM	00:01:56:799	start send ping
ADMIN_TRACE_INFO	ADMIN_TRACE_LEVEL_1	ADMIN	PS_OM	00:01:56:799	ping ipv4 datalen:64, timeout:20, interval_time:21
ADMIN_TRACE_INFO	ADMIN_TRACE_LEVEL_1	ADMIN	PS_OM	00:02:00:969	reply from 139.224.112.6, 64 bytes 4168 ms 112 ttl
ADMIN_TRACE_INFO	ADMIN_TRACE_LEVEL_1	ADMIN	PS_OM	00:02:16:864	reply from 139.224.112.6, 64 bytes 15894 ms 112 ttl
ADMIN_TRACE_INFO	ADMIN_TRACE_LEVEL_1	ADMIN	PS_OM	00:02:32:224	reply from 139.224.112.6, 64 bytes 15359 ms 112 ttl
ADMIN_TRACE_INFO	ADMIN_TRACE_LEVEL_1	ADMIN	PS_OM	00:02:46:341	reply from 139.224.112.6, 64 bytes 14116 ms 112 ttl
ADMIN_TRACE_INFO	ADMIN_TRACE_LEVEL_1	ADMIN	PS_OM	00:02:46:343	statistics: ping num:4, reply:4, longest_rtt:15894ms, shortest_rtt:4168ms, average_time:12384ms

7.2 速率灌包测试+XYPERF

语法结构

命令	响应
AT+XYPERF=<host>,<port>,<net_type>,<size>,<band width>,<time>,<param>	OK +CME ERROR:<err>

接口说明

该命令用于 XYPERF 灌包, 仅用于速率灌包测试, 不可用于开发, 灌包尚未结束时, 不能发第二次灌包。灌包速率由包大的小和带宽决定。

取值说明

<host>: 服务器 IP 地址

<port>: 服务器端口号

<net_type>: 协议类型, 目前只支持 udp

<size>: 包的大小

<bandwidth>: 带宽, 可支持数字带上 k(K)、m(M)、g(G), 例如 1k, 2M 或者直接数字, 例如 1024、2048, 最小值不能低于 1024

<time>: 灌包时长, 单位秒

<param>: 固定为 1

典型示例

AT+XYPERF=139.224.112.6,10003,udp,1024,2k,30,1
OK

```

351951 2019-11-12 16:00:49.191 L2 / / Em_data_trans
351952 2019-11-12 16:00:49.191 L2 / / Secur_Protected_Nas_msg
351953 2019-11-12 16:00:49.191 NAS DEBUG LNB_RRC_DATA_REQ
351954 2019-11-12 16:00:49.191 L2 / / MultiConnCtrlReq-Req
351955 2019-11-12 16:00:49.191 L2RC DEBUG MSG_LNB_RLC_AM_DATA_REQ
351956 2019-11-12 16:00:49.191 LPHY DEBUG PHY: RfGainControlPlant ampLinearBB=13, powMMS=338, LNA=46, MX=80, PGA=230, CIC=180, ACI=150, powAMT=
351957 2019-11-12 16:00:49.191 LPHY DEBUG PHY: RfGainControlPlant DeltaRfGainDb10=0, DeltaCicGainDb10=1, DeltaAciGainDb10=24, DeltaBBFowDb10=73
351958 2019-11-12 16:00:49.191 USER_LOG WARNING ip data write to dsp, mem addr=0x20015100
351959 2019-11-12 16:00:49.191 USER_LOG WARNING ip data write to dsp, mem addr=0x20015100
351960 2019-11-12 16:00:49.191 USER_LOG WARNING xyperf finished
351961 2019-11-12 16:00:49.191 LPHY DEBUG PHY: RfGainControlPlant ampLinearBB=18, powMMS=648, LNA=46, MX=80, PGA=230, CIC=180, ACI=150, powAMT=
351962 2019-11-12 16:00:49.191 LPHY DEBUG PHY: RfGainControlPlant DeltaRfGainDb10=0, DeltaCicGainDb10=1, DeltaAciGainDb10=23, DeltaBBFowDb10=70
351963 2019-11-12 16:00:49.191 LPHY DEBUG PHY: PHY Conf LIC eventId=8, Cr=0, subframeCount =31, NetworkTime=(263, 703, 1), Sfo=(299, 8, 33), Freq
351964 2019-11-12 16:00:49.191 LIC DEBUG BBPLL: ulCurRegVal:6710886, ulSetRegVal:6710886, slDetal: 200, sFreqErr: 560, ulPrequency: 953800000
351965 2019-11-12 16:00:49.191 LPHY DEBUG PHY: T1LSNRdB_PSS=3515, T1LSNRdB_PSS=3086, gainK1_RF_LNA=46, gainK1_RF_MX=80, gainK1_RF_PGA=230, gainK
351966 2019-11-12 16:00:49.191 LIC DEBUG LIC: Meas NRS ServingCell B1FearIcon=3738, poi=37, Nrsrp=910, Nrsrq=187, Rssi=723, Sinc=68, NormalCv
351967 2019-11-12 16:00:49.191 LPHY DEBUG PHY: LIC Config PHY EventID=6, StartTime=(263, 704, 1), EndTime=(263, 706, 2), Sfo=(39, 299, 8, 33), Fr
351968 2019-11-12 16:00:49.191 LPHY DEBUG PHY: DFE open local CurProtina=(224, 403, 5, 968), Start=(224, 403, 6, 1387), Dfe Interrupt=(224, 403, 7, 427),
351969 2019-11-12 16:00:49.191 LPHY DEBUG PHY: enhanced CHE, connectFlg 1, openDfe countInsub:1887, sfo countInsub:33 openDfe countInSub:1387
351970 2019-11-12 16:00:49.191 LIC WARNING LIC: Cmt Config npdch StartTime=2700153, NpdchStartTime=2700161, NpdchEndTime=2700182, NpdchInitSt
    
```

7.3 SOCKET 的灌包+XDSEND

该命令仅用于速率灌包测试, 不可用于开发。

语法结构

命令	响应
AT+XDSEND=<socket_id>,<datalen_data>	OK
	+CME ERROR:<err>

接口说明

该命令用于 XDSEND 灌包, 该灌包命令的串口波特率需要设置成 115200。灌包前需要使用命令 “AT+XSOPEN=<type>,<remote_ip>,<remote_port>[,<local_port>]” 创建 UDP socket, 再发送灌包 AT 命令, 无需等待灌包 AT 命令的相应结果, 等待 20ms 直接再发下一条, 见下图。

取值说明

<socket_id>: 链路 ID 号, 由创建 socket 返回的 socketid。

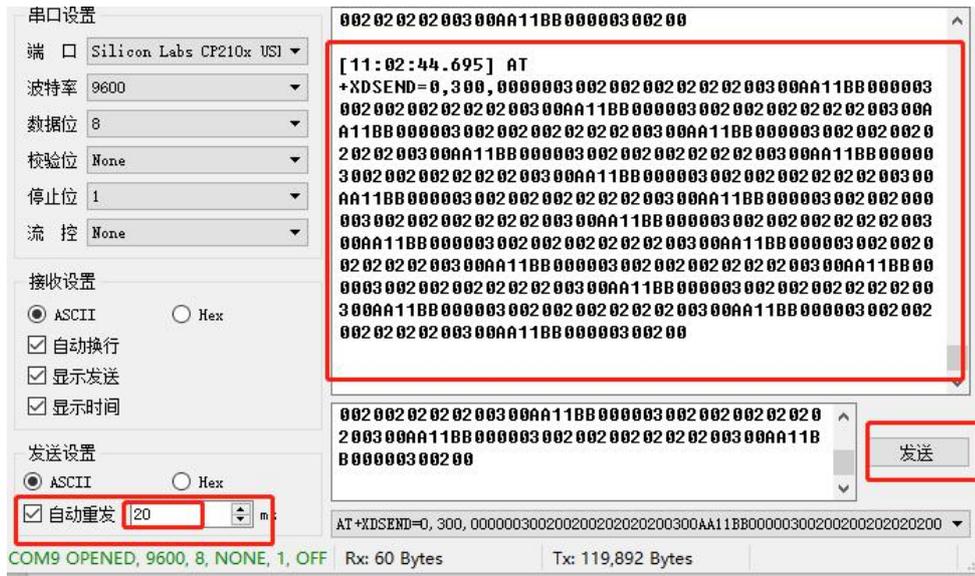
<datalen>: 灌包数据长度 (十六进制)

<data>: 十六进制灌包数据

典型示例

AT+XSOPEN=1,139.224.112.6,10005
+XSOPEN:0
OK

```
+XSSTATE:0,1
AT+XDSEND=0,300,000000300200200202020200300AA11BB000003002002002020200300AA
11BB000003002002002020200300AA11BB000003002002002020200300AA11BB000003002
002002020200300AA11BB000003002002002020200300AA11BB000003002002002020200
300AA11BB000003002002002020200300AA11BB000003002002002020200300AA11BB0000
03002002000003002002002020200300AA11BB000003002002002020200300AA11BB000003
002002002020200300AA11BB000003002002002020200300AA11BB000003002002002020
200300AA11BB000003002002002020200300AA11BB000003002002002020200300AA11BB0
00003002002002020200300AA11BB000003002002002020200300AA11BB00000300200
```



7.4 用户 DEMO 演示命令+TEST

语法结构

命令	响应
AT+TEST=<NAME>	OK
	+CME ERROR:<err>

接口说明

用于调试各种 DEMO 参考代码。

取值说明

命令格式	功能源文件	备注
AT+TEST=CLIENT	at_socket_demo.c	基于扩展AT命令的socket用户端参考代码，实现与远端服务器的建链、数据交互、释放等流程。
AT+TEST=SERVER	at_server_demo.c	AT服务端参考代码，供用户实现外部MCU与海凌科芯片间的扩展AT命令信息交互。
AT+TEST=RTC	rtc_task_demo.c	RTC定时器的开发参考，包括省电锁的使用
AT+TEST=TASK	rtc_task_demo.c	用户周期性任务的参考代码，包括第一次上电及深睡唤醒的参考建议；

AT+TEST=GPIO	gpio_demo.c	gpio的开发参考，以中断方式体现，一种是外部MCU触发中断给海凌科芯片；一种是海凌科芯片触发中断给外部MCU
AT+TEST=ENCODE,<data>	at_encode_demo.c	用于编解码的参考代码，编解码的线程栈需要3k为宜，data为填入的编码数据内容

典型示例

```
AT+TEST=RTC
OK
```

备注：如果遇到省电相关异常，可通过 AT+CPSMS=0 关闭 PSM 省电。

7.5 内存信息获取命令+MEMSTATS

语法结构

命令	响应
AT+MEMSTATS=<Option>	成功：返回内存信息 失败：返回错误类型

接口说明

用于返回内存 DSP 或者 ARM 核（取决于输入参数）的堆或栈使用情况。

取值说明

命令格式	功能源文件	备注
AT+MEMSTATS=DSPMEM	los_heap.c	基于Freertos的堆内存分配源代码，加入内存堆使用情况调试信息（查询dpsmem请用 AT+NUESTATS=DSPMEM）
AT+MEMSTATS=ARMMEM	heap5.c	基于LitOS的堆内存分配源代码，加入内存堆使用情况调试信息
AT+MEMSTATS=ARMSTACK	los_task.c	基于LitOS的任务信息获取源代码，加入内存栈使用情况调试信息
AT+MEMSTATS=ALLMEM	os_adapt.c	获取flash和ram总大小以及剩余空间。

典型示例

例 1:

```
AT+MEMSTATS=DSPMEM
TotalHeapSize:129024           DSP 堆总大小
CurrentTotalRemainSize:72720   堆剩余大小
CurrentMaxFreeBlockSize:33752  当前堆中可申请最大的内存块大小
HistotyMinimumRemainSize:65744 堆历史剩余最小值
OK
```

例 2:

```
AT+MEMSTATS=ARMMEM
NUESTATS:APPSMEM, Current Allocated:  目前已分配内存大小
NUESTATS:APPSMEM, Total Free: 68497   总剩余内存大小
```

NUESTATS:APPSMEM, Max Free: 68185 剩余最大连续内存块大小
NUESTATS:APPSMEM, Number Allocs: 194 历史总内存分配次数
NUESTATS:APPSMEM, Number Frees: 159 历史总内存释放次数
OK

例 3:

AT+MEMSTATS=ALLMEM

TotalSizeOfFlash:405504 FLASH Text 总大小
FlashRemaining:61419 FLASH Text 剩余空间
TotalSizeOfDataAndText:106496 RAM 中 Data 和 Text 区域总大小
DataAndTextRemaining:32036 RAM 中 Data 和 Text 剩余空间
TotalSizeOfBSS:61440 BSS 段总大小
BSSRemaining:13912 BSS 段剩余空间
OK

例 4:

AT+MEMSTATS=ARMSTACK

+wdt_task,StackSize:1024 堆栈总大小
+wdt_task,StackRemaing:164 当前堆栈使用量
+wdt_task,PeakUsedSize:504 堆栈峰值使用量
OK

8 err 错误码

目前平台提供的错误码包括：3GPP 标准 AT 命令的错误码，范围 0-500；应用平台级扩展错误码，范围为 8000-9000；型号机客户自定义的扩展错误码，范围为大于 9000。

8.1 General Errors (27.007)

Code of <err>	Description
3	Operation not allowed
4	Operation not supported
5	Need to enter PIN
23	Memory failure
30	No network service
50	Incorrect parameters
51	Command implemented but currently disabled
52	Command aborted by user
100	Unknown
159	Uplink busy/flow control

8.2 General Errors (27.005)

Code of <err>	Description
300	ME failure
301	SMS service of ME reserved
302	Operation not allowed
303	Operation not supported
304	Invalid PDU mode parameter
305	Invalid text mode parameter
310	USIM not inserted
311	USIM PIN required
312	PH-USIM PIN required
313	USIM failure
314	USIM busy
315	USIM wrong
316	USIM PUK required
317	USIM PIN2 required
318	USIM PUK2 required
320	Memory failure
321	Invalid memory index
322	Memory full
330	SMSC address unknown
331	No network service
332	Network timeout
340	No +CNMA acknowledgement expected
500	Unknown ERR

8.3 XYEnhanced Errors (HLK-N10)

Code of <err>	Description
8000	需要进行设备重启
8001	输入的AT命令参数不合法
8002	操作不允许，通常为状态机不符合
8003	AT命令丢失或字符脏，常见于standby唤醒时PLL不稳造成的
8004	当前正在进行FOTA升级，不接受其他操作控制
8005	暂未使用
8006	等待AT应答超时，需要将等待时长设大
8007	AT通道忙，上一条AT请求尚未处理完
8008	当前为单核模式，仅M3核正常工作，错误的发送需要DSP核处理的AT命令
8009	NB网络异常，tcpip网路不通畅
8010	发生断电，需要输入“AT+OFFTIME=<time>”断电时长补偿，仅在poweroff_enable和offtime皆为1情况下才会上报该错误
8012	前缀无效。如：AT+XXX=?QQQ和AT+XXX?QQQ
9000	用户自定义扩展错误码起始值

版本历史

资料版本	时间	资料更新说明
V1.0	2020/7/18	初始版本