

类别	内容
关键词	LoRaWAN、AT 模组、深度开发
摘要	LoRaWAN AT 模块深度开发指南.

修订历史

版本	日期	原因
1.0.0	2022/07/18	• 创建文档

目 录

1. 简介	1
1.1 功能简介	1
2. 引脚描述	2
2.1 控制引脚	3
2.1.1 ISP	3
2.1.2 BUSY	4
2.1.3 DEFAULT	5
2.1.4 RESET	5
2.1.5 MODE	5
2.1.6 STATE	6
2.1.7 ACK	7
2.1.8 SLEEP/WAKE	8
2.2 IO 引脚	8
2.3 ADC 引脚	8
2.4 UART 引脚	8
2.5 PWM 引脚	8
3. 推荐电路	10
4. 串口通信规范	11
4.1 AT 通信模式	11
4.2 AT 指令格式	11
4.2.1 AT 指令	11
4.2.2 URC 指示	12
4.2.3 响应错误	12
4.3 透传通信模式	12
4.4 透传通信格式	12
4.4.1 发送数据	12
4.4.2 输出格式 0	13
4.4.3 输出格式 1	14
4.4.4 输出格式 2	15
4.5 串口通信约定	17
4.6 串口分帧格式	17
4.7 协议栈错误码	17
5. 频段参数规范	19
5.1 CN470 - 510MHz	19
5.1.1 通信速率编码	19
5.1.2 发射功率编码	19
5.1.3 通信帧长定义	19

6. 终端 AT 指令集	21
6.1 AT 协议栈指令集	21
6.1.1 AT+CLASS	22
6.1.2 AT+MODE	23
6.1.3 AT+ID	23
6.1.4 AT+APPKEY	24
6.1.5 AT+SKEY	24
6.1.6 AT+ADDRESS	25
6.1.7 AT+ADR	26
6.1.8 AT+CHMASK	26
6.1.9 AT+PORT	27
6.1.10 AT+DR	27
6.1.11 AT+POWER	28
6.1.12 AT+TRANSJOIN	28
6.1.13 AT+JOIN	29
6.1.14 AT+NETSTS	31
6.1.15 AT+TX	31
6.1.16 AT+RX2	32
6.1.17 AT+RXCNT	33
6.1.18 AT+LINKCHECK	33
6.1.19 AT+DATLENGTH	34
6.1.20 AT+BUSYCHECK	34
6.1.21 AT+MULTICASTADD	35
6.1.22 AT+MULTICASTDEL	35
6.1.23 AT+MULTICASTLIST	36
6.1.24 AT+CSMA	36
6.1.25 AT+BACKOFF	36
6.1.26 AT+PERIODTX	37
6.1.27 AT+AUTOACK	38
6.1.28 AT+VOLTAGE	38
6.1.29 AT+ACKTIME	39
6.1.30 AT+CONFIRMED	40
6.1.31 AT+NBTRANS	40
6.1.32 AT+TXSTR	41
6.1.33 AT+TXHEX	41
6.1.34 AT+TRANSOUT	42
1. 格式 0 (FORMAT=0)	42
2. 格式 1 (FORMAT=1)	42
3. 格式 2 (FORMAT=2)	42
6.1.35 中继指令集	43
1. AT+REP	44
2. AT+REPMODE	44

3. AT+REPWHLISTADD	45
4. AT+REPWHLISTDEL	45
5. AT+REPWHLIST	46
6. AT+REPFREQADD	46
7. AT+REPFREQDEL	47
8. AT+REPFREQ	47
9. AT+REPFREQLIVE	48
6.2 AT 外设操作指令集	48
6.2.1 AT+GPIO	49
6.2.2 AT+PWM	49
6.2.3 AT+ADC	50
6.3 AT 其他功能指令集	50
6.3.1 AT	51
6.3.2 AT+E	51
6.3.3 AT+H	51
6.3.4 AT+L	52
6.3.5 AT+SLEEP	53
6.3.6 AT+DEFAULT	53
6.3.7 AT+SAVE	54
6.3.8 AT+RST	54
6.3.9 AT+UPGRADE	54
6.3.10 AT+BAUD	55
6.3.11 AT+CHTIMEOUT	55
6.3.12 AT+DEVINFO	56
6.4 AT 测试指令集	56
6.4.1 AT+LORATEST	56
6.4.2 AT+TEST	57
1. AT+TEST=?	57
2. AT+TEST=TXCW	57
3. AT+TEST=TXLORA	57
4. AT+TEST=TXCFG	58
5. AT+TEST=RXCFG	58
6. AT+TEST=TXPKT	59
7. AT+TEST=TXSTR	59
8. AT+TEST=PUBLIC	60
9. AT+TEST=RXCONTINUE	60
10. AT+TEST=RSSI	61
11. AT+TEST=LWCAP	61
6.5 AT 服务器 URC 指示列表	62
6.5.1 +TX	62
6.5.2 +RX	63
6.5.3 +LINKCHECK	63

6.5.4 +JOIN	64
7. 终端时序参考范例	65
7.1 启动时序	65
7.2 低功耗时序	65
7.3 串口帧分割时序	66
7.4 AT 指令时序	66
7.5 串口发送时序	66
7.6 数据接收时序	66
7.7 节点透传时序	67
8. 终端 AT 使用范例	68
8.1 终端激活	68
8.1.1 OTAA	68
8.1.2 ABP	69
8.2 信道修改	69
8.3 数据收发	69
8.3.1 数据发送	70
8.3.2 数据接收	71
8.4 MAC 指令请求	71
8.5 多播使用	71
8.6 中继使用	72
9. 终端透传模式使用范例	74
9.1 数据分帧	74
9.2 终端配置	74
9.2.1 启动配置	74
9.2.2 运行中配置	74
9.3 数据收发	74
10. 如何升级	76
10.1 串口升级	76
10.1.1 进入等待升级模式	76
10.1.2 升级	76
11. 低功耗处理	77
12. 免责声明	78

1. 简介

广州致远电子有限公司 ZSL420-EVB demo 板是标准 LoRaWAN CN 470 - 510MHz 终端。完全实现 LoRaWAN™ Specification v1.0.2 Class A/C 规范和 LoRaWAN™ 1.0.2 Regional Parameters Revision: B CN 470-510MHz 频段参数。采用 UART 作为通信控制方式，包含串口透传与 AT 两种模式，方便用户快速开发基于 LoRaWAN 协议应用。

1.1 功能简介

用户可先将终端设备切换至 AT 模式，配置入网等参数，再切换回透传模式，透传用户的收发数据。为了在透传模式获取当前终端设备状态，终端设备也提供了 STATE、BUSY 等反馈引脚，用于判断当前终端设备是否入网、是否忙碌等。

为了降低功耗，用户可将终端设备设置为休眠模式，需要再发数据时，唤醒终端设备。当然，终端设备也提供周期上报功能，用户 MCU 可继续休眠，终端设备接收到数据后，通过 ACK 引脚唤醒用户 MCU，即实现空中唤醒功能，降低整机功耗。

终端设备也支持 Xmode 升级功能，以及 LoRaWAN 的组播功能，功能示意如图 1.1 所示。

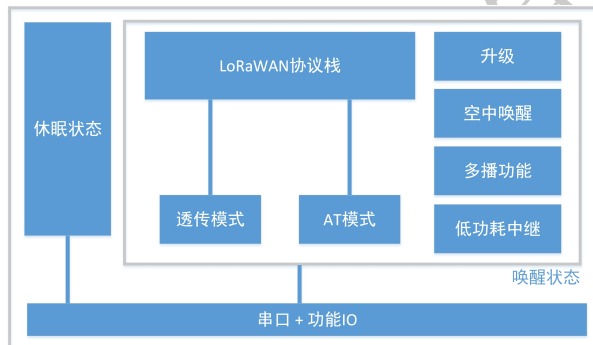


图 1.1 终端设备功能示意图

本文旨在描述 ZSL420-EVB 相关规范及应用开发指南。

2. 引脚描述

ZSL42x LoRaWAN 固件芯片引脚如 图 2.1 所示。

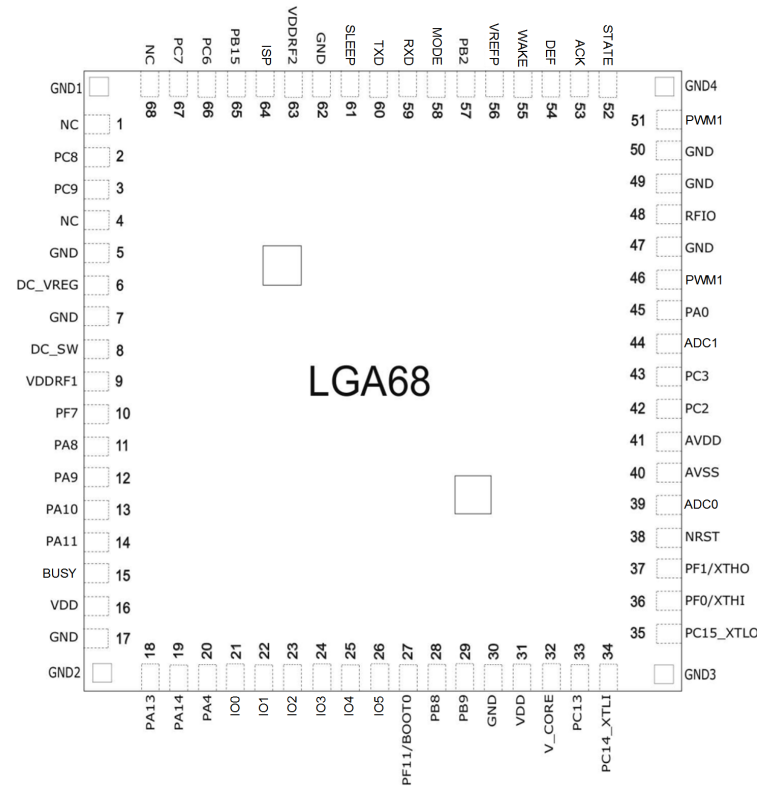


图 2.1 ZSL42x LoRaWAN 固件芯片引脚图

LoRaWAN 固件定义的引脚功能如 表 2.1 所示。

表 2.1 LoRaWAN 固件定义的引脚及其功能

引脚号	引脚名称	类型	用户 MCU 引脚处理建议
5	GND	S	电源地
6	DC_VREG	-	需要接外部 15uH 电感
7	GND	S	电源地
8	DC_SW	-	需要接外部 15uH 电感
9	VDD_RF1	S	无线收发器供电引脚 1
15	BUSY	O	指示终端设备忙状态引脚
16	VDD	S	芯片供电电源
17	GND	S	电源地
18	SWDIO	I/O	SWD 调试引脚
19	SWCLK	I/O	SWD 调试引脚
21	IO5	I/O	LoRaWAN 固件的 IO5 引脚，可输入、输出
22	IO4	I/O	LoRaWAN 固件的 IO4 引脚，可输入、输出
23	IO3	I/O	LoRaWAN 固件的 IO3 引脚，可输入、输出
24	IO2	I/O	LoRaWAN 固件的 IO2 引脚，可输入、输出
25	IO1	I/O	LoRaWAN 固件的 IO1 引脚，可输入、输出
26	IO0	I/O	LoRaWAN 固件的 IO0 引脚，可输入、输出

续上表

引脚号	引脚名称	类型	用户 MCU 引脚处理建议
27	PF11/BOOT0	I	外接下拉电阻
30	GND	S	电源地
31	VDD	S	芯片供电电源
32	V_CORE	S	芯片内核电压输出引脚，接 1uF 电容到 GND
34	XTLI	I/O	芯片 MCU 内核外部 32.768K 晶体引脚
35	XTLO	I/O	芯片 MCU 内核外部 32.768K 晶体引脚
38	RST	I	复位，至少保持 100ns 低电平复位
39	ADC0	I	LoRaWAN 固件 ADC 引脚
40	AVSS	S	模拟电源地
41	AVDD	S	模拟供电电源
44	ADC1	I	LoRaWAN 固件 ADC 引脚
46	PWM2	O	LoRaWAN 固件 PWM 引脚
47	GND	S	电源地
48	RFIO	I/O	射频 RF 信号输入输出
49	GND	S	电源地
50	GND	S	电源地
51	PWM1	O	LoRaWAN 固件 PWM 引脚
52	STATE	O	入网、数据收发状态指示引脚
53	ACK	O	接收到数据通知引脚
54	DEF	I	恢复出厂配置引脚
55	WAKE	I	唤醒引脚
58	MODE	I	透传与 AT 模式切换引脚
59	RXD	I	串口接收引脚
60	TXD	O	串口发送引脚
61	SLEEP	I	进入休眠模式引脚（模块空闲时，至少拉低 10ms）
62	GND	S	电源地
63	VDD_RF2	S	无线收发器供电引脚 2
64	ISP	I	进入升级状态引脚（上电或复位，至少拉低 10ms）

注：

1. DEF 引脚内部上拉，芯片复位或上电之前拉低该引脚则芯片参数恢复出厂设置。

此次只讲述与应用开发相关的功能使用和低功耗模式下的状态，其他引脚可参考相关数据手册。终端设备在低功耗模式下引脚状态将保持进入低功耗模式前的状态。

注：文档中的节点、终端设备、AT 服务器均指 ZSL42x LoRaWAN 固件节点。

下面介绍一下 LoRaWAN 固件中，一些重要的功能引脚。

2.1 控制引脚

2.1.1 ISP

ISP 为输入引脚，内部输入上拉。功能为指示是否利用内部集成的 bootloader 对应用代码进行升级，输入有效时间范围为终端复位后的初始化过程。当输入高电平或 ISP 脚悬空时执行应用代码，如图 2.2 所示：

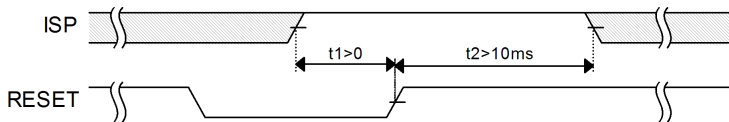


图 2.2 ISP 引脚控制使用应用代码时序

当 ISP 脚输入低电平时表示进入升级模式。需要在 ZSL420-EVB 复位前拉低，并在复位结束（即 RESET 拉高）后至少保持 10ms 时间，如 图 2.3 所示：

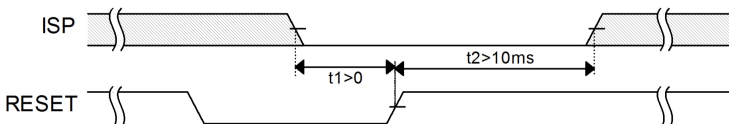


图 2.3 ISP 引脚升级使用时序

在低功耗模式下主控制器可以调整与终端设备相连接的引脚状态使得达到最低功耗，推荐调整为输出高电平或输入上拉，此操作并不会影响终端的运行。

2.1.2 BUSY

BUSY 输出终端设备是否空闲，BUSY 输出低电平表示终端设备忙碌，此时不可操作终端设备；当 BUSY 输出高电平，表示设备空闲，此时可以使用串口发送数据或 AT 指令操作终端设备。

终端设备复位后 BUSY 会自动切换至忙碌状态，终端设备初始化完成再切换至空闲状态。因此可利用 BUSY 引脚状态来判断终端设备是否初始化完成，如 图 2.4 所示：

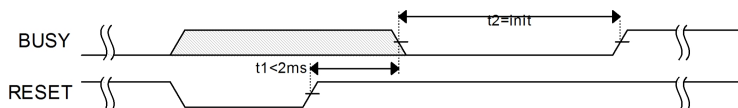


图 2.4 BUSY 引脚在复位时的应用时序

如上图所示，在复位完成后到 BUSY 引脚输出状态有效之间有一个大约 2ms 内的间隔 t_1 ，之后 BUSY 引脚被拉低直到终端设备初始化完成，时间为 t_2 。若需要消除 BUSY 引脚的无效状态时间 t_1 ，可以在 BUSY 引脚加外部下拉，维持 BUSY 无效时的状态。

在终端设备正常运行时，BUSY 在上行数据时被拉低，在上行结束后 BUSY 引脚拉高，以 AT 上行数据为例，如 图 2.5 所示：

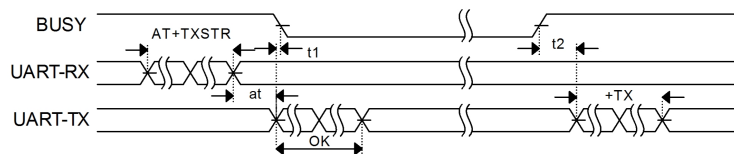


图 2.5 BUSY 引脚在数据上行的应用时序

如上图所示，在 AT+TXSTR 指令处理后返回响应的同时 BUSY 引脚也被拉低指示设备忙碌，其中时间差 t_1 基本可忽略。在整个上行流程结束后 BUSY 拉高表示终端设备空闲，在

延时 t_2 后输出了本次上行的结果，时间段 t_2 的定义与描述可参考 ACK 引脚。

同理 BUSY 引脚也可用于判断终端设备的低功耗模式，当设备进入低功耗模式时 BUSY 引脚被拉低，指示终端设备不能再接收串口数据，当设备唤醒后 BUSY 引脚被拉高指示终端设备空闲。如 图 2.6 所示：

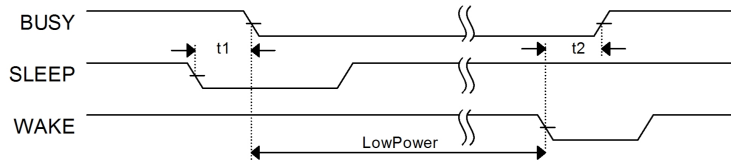


图 2.6 BUSY 引脚在低功耗模式的应用时序

上图中时间 t_1 、 t_2 将分别在 SLEEP 引脚及 WAKE 引脚章节详细描述。

2.1.3 DEFAULT

DEFAULT 为输入引脚，内部输入上拉。功能为指示是否恢复默认配置，输入有效时间范围为终端复位后的初始化过程。当 DEFAULT 输入高电平或悬空时不恢复参数，加载用户存储数据。当 DEFAULT 输入低电平时表示恢复默认参数。需要在终端复位前拉低，并保持至初始化化结束，如 图 2.7 所示：

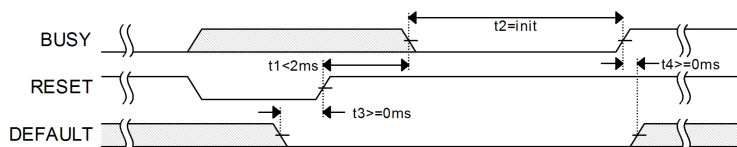


图 2.7 使用 DEFAULT 引脚恢复默认配置时序

2.1.4 RESET

RESET 作为终端设备复位脚，拉低 RESET 脚可以复位整个终端设备，如 图 2.8 所示：

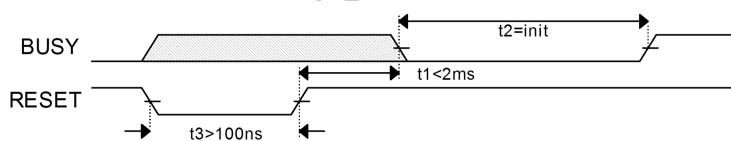


图 2.8 RESET 引脚时序

2.1.5 MODE

MODE 为输入引脚，内部输入上拉。该引脚用来控制终端设备模式的动态切换，当输入高电平时终端设备为透传模式，当输入低电平时终端设备为 AT 模式。MODE 引脚需要一直保持，在发生模式切换时所有的引脚状态保持不变。

切换模式后当前缓存的 UART 接收数据将被清除，已进入协议栈处理的消息不受模式切换影响。例如数据上行，单次入网操作。处理结果将使用与处理结束时的模式相对应的方法输出。例如在透传模式自动入网过程过切换至 AT 模式，在本次入网结束后会使用 +JOIN 指令输出。

MODE 脚可在任何时候切换，切换可在极短的时间内识别，但若需要使用 UART 操作设备最好需要等待 BUSY 输出空闲。

2.1.6 STATE

STATE 用于指示终端设备状态，在复位后 STATE 指示终端设备是否入网，输出高表示终端设备已经成功入网，输出低表示终端设备未入网。在复位初始化结束后到终端入网前 STATE 将一直保持低电平，因此可以直接使用 STATE 脚判断是否入网成功，AT 模式下如图 2.9 所示：

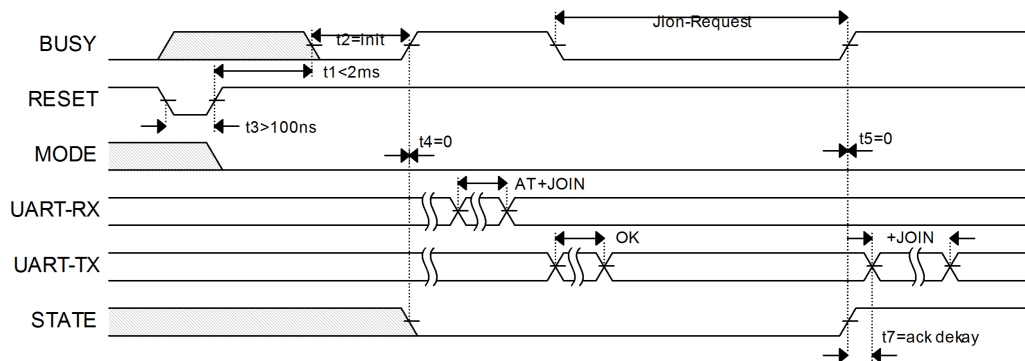


图 2.9 AT 模式下 STATE 脚指示入网状态时序

透传模式复位初始化后自动入网，可以和 BUSY 脚结合一起判断当前是否还在尝试入网及入网结果，如图 2.10 所示：

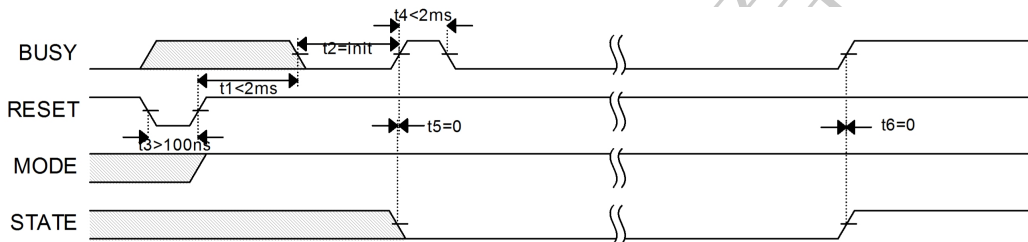


图 2.10 透传模式下 STATE 脚指示入网状态时序

如上图所示，BUSY 脚拉高后 STATE 脚为高表明入网成功，STATE 脚拉低表明终端设备入网失败。入网成功后 STATE 脚则变化为指示最后一次上行过程的执行结果，STATE 脚输出高则表示最后一次上行结果为成功，STATE 脚输出低则表示最后一次上行结果为失败。STATE 脚将保持输出直到再次启动上行，再次启动上行时 STATE 状态清除，输出高电平。以 AT 模式为例，如图 2.11 所示：

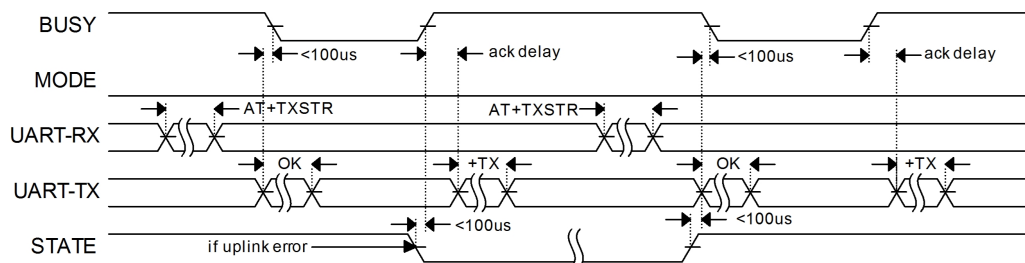


图 2.11 STATE 脚指示数据传输状态时序

在启动传输时，若启动失败则会直接将 STATE 拉低表明当前启动错误。在 AT 模式下会输出 AT 指令响应错误，透传模式下不会有任何信息输出。以透传模式为例，如图 2.12

所示:

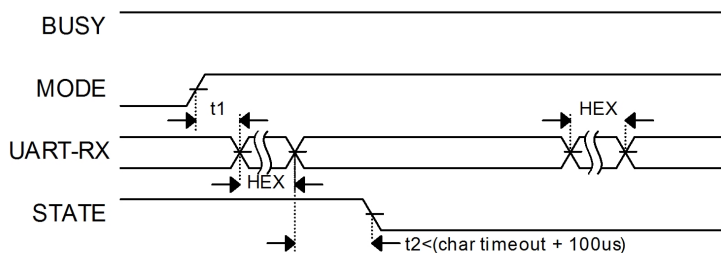


图 2.12 STATE 脚指示上行启动失败时序

其中字符超时时间为透传模式下分帧的依据, 可通过 AT+CHTIMEOUT 设置。

在终端运行中如果发送 AT+JOIN 重新入网, 则 STATE 脚将回到指示网络连接的状态, 在启动入网时 STATE 脚将被拉低, 直到重新入网成功, 如图 2.13 所示:

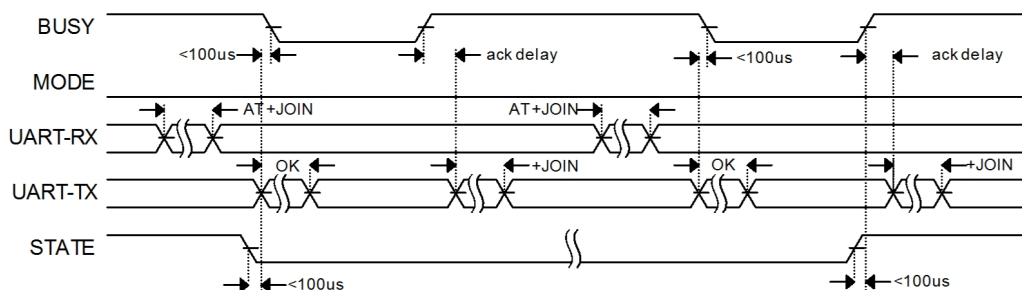


图 2.13 STATE 脚指示重入网状态时序

2.1.7 ACK

ACK 引脚为输出引脚, 终端设备在 AT 模式发送 URC 指示及透传模式输出消息时, ACK 引脚都会输出一个低脉冲。以 AT 模式数据发送为例, 如图 2.14 所示:

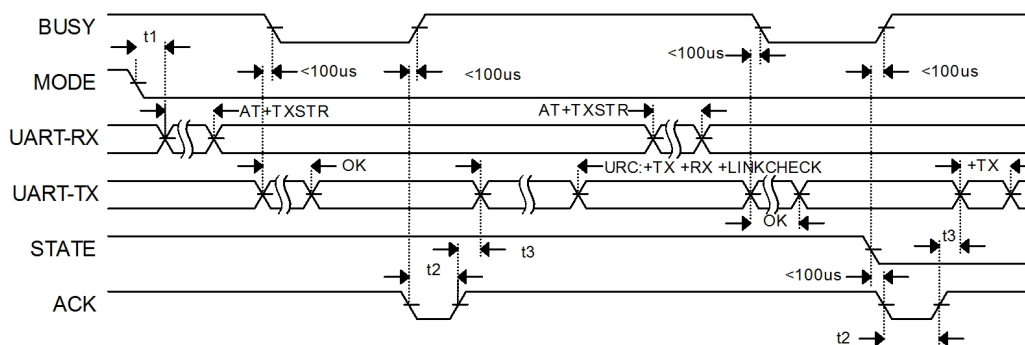


图 2.14 ACK 脚状态时序

上图中 t2 为 ACK 脉冲宽度, 默认为 1ms; t3 为输出延时, 默认为 0ms, 这两个值可以通过 AT+ACKTIME 指令修改。

AT 模式入网结果 +JOIN 指令输出前也会输出 ACK 脉冲, 和图 2.14 相同。在数据传输时一次传输可能输出多个 URC 指令, 但只会输出一个 ACK 脉冲, 所有 URC 指示在脉冲后连续输出。

透传模式下的 ACK 引脚时序和 AT 模式相同, 在消息输出前输出 ACK 脉冲。

2.1.8 SLEEP/WAKE

SLEEP/WAKE 引脚为输入引脚，默认输入上拉。SLEEP 用来控制终端进入睡眠模式，WAKE 用于唤醒处于低功耗模式的终端设备。控制时序如 图 2.15 所示：

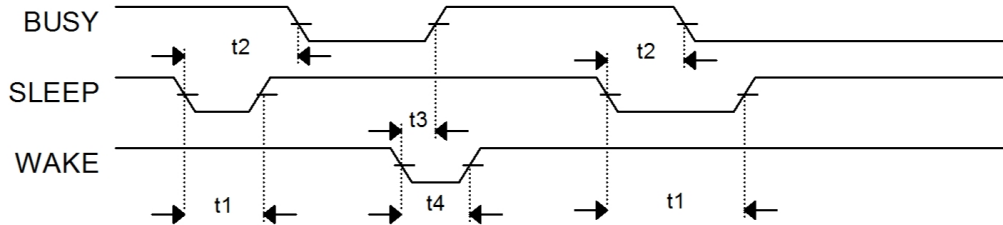


图 2.15 SLEEP/WAKE 引脚控制时序

SLEEP 保持时间 t_1 ，在 BUSY 指示空闲时需要持续至少 6ms，保证终端成功检测到 SLEEP 输入。在 BUSY 指示为忙碌时 SLEEP 拉低时间不可控，若需要在 BUSY 指示忙碌时操作 SLEEP，则最好一直保持 SLEEP 电平。

SLEEP 进入时间 t_2 的具体时间不可控，一般在 SLEEP 识别之后的几十微秒到几十毫秒之内，因为终端需要做相关配置和参数存储。虽然 t_2 时间不确定，但在 BUSY 拉低后就表明开始执行低功耗进入，此时不应该再操作终端设备。

WAKE 保持时间 t_4 需要至少保持 1ms 的时间。终端在通过 WAKE 唤醒后将不会再自动进入低功耗模式，如需再次进入需要再使用 SLEEP 或 AT 指令 AT+SLEEP。

WAKE 唤醒时间 t_3 为几十微妙到十几毫秒的一个范围。当 BUSY 恢复到空闲状态时则表明设备已唤醒并恢复，此时可继续操作终端。

2.2 IO 引脚

通用 IO 有 IO0~IO5 共计 6 个，所有 IO 在第一次上电时默认状态均为输入下拉。IO 可通过 AT 指令 AT+GPIO 配置输入输出方向、配置输出状态、获取输入状态。IO 在配置完成后若使用 AT+SAVE 保存，则在下次上电后自动恢复为配置值，否则恢复为默认值。

在低功耗模式下及唤醒后所有 IO 保持状态不变。

2.3 ADC 引脚

ADC 引脚为模拟输入引脚，连接 ADC 外设，为 AT+ADC 指令获取 ADC 值的输入通道。在低功耗模式下该引脚的配置不改变。

2.4 UART 引脚

即串口收发引脚，可先通过 AT 命令配置，之后可直接透传数据。

2.5 PWM 引脚

PWM 引脚为输出引脚，用来输出 AT+PWM 控制的 PWM 波形，在禁能 PWM 时也可用来作为通用输出的 IO 口。

该引脚默认配置为禁能 PWM、输出低电平。由于 PWM 终端不能在低功耗模式下工作，因此如果使能了 PWM 则终端不会进入低功耗模式，配置的 PWM 不会保存，在重启后默认

加载 PWM 禁能和禁能输出电平。使用 AT+PWM 配置的禁能输出电平会保存在 Flash 中，上电后恢复。

广州致远电子股份有限公司

3. 推荐电路

ZSL42x LoRaWAN 固件芯片推荐如 图 3.1 所示。其中这里只涉及通过串口配置参数、数据透传，当需要其他功能时，按照上述功能引脚描述增加引脚。

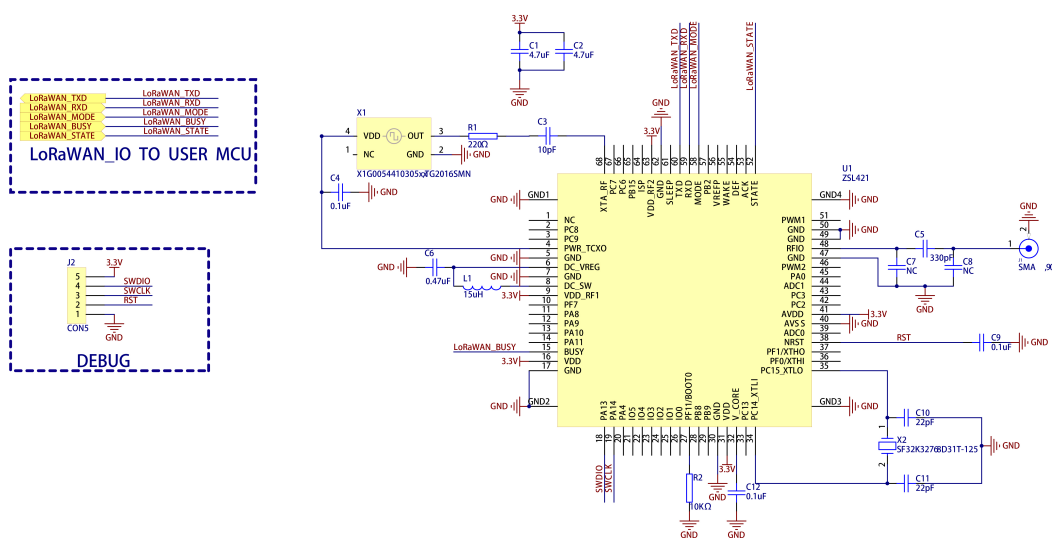


图 3.1 ZSL42x 推荐电路

注:

1. 在透传模式下, 可通过 `BUSY` 判断模式是否忙碌 (低电平忙碌), 通过 `STATE` 引脚判断数据发送是否异常。

特别的，对于一些特定的场景，如低功耗等，需要增加一些额外的引脚，即：

- 模块设置为定时上报，当用户 MCU 休眠后，模块通过 **ACK** 引脚唤醒用户 MCU，通知用户 MCU 接收数据，实现空中唤醒功能，降低整机功耗。
- 当需要做低功耗处理时，将 **WAKE**、**SLEEP** 引脚接上，当模块空闲时，拉低 **SLEEP** 引脚 10ms，即可进入休眠模式，通过 **WAKE** 脚下降沿唤醒 MCU。
- 可在芯片上电后复位，拉低 **DEF** 引脚 10ms，即可恢复出厂设置。

4. 串口通信规范

4.1 AT 通信模式

AT 指令是从主控制器设备（例如 PC）向终端设备（例如 ZSL420-EVB demo 板）发送的一串特殊格式的数据。一般将主控制器设备称为客户端，将终端设备称为服务器。顾名思义，服务器是提供某种“服务”的设备，“服务”可以是某些有意义的数据，也可以是执行某些动作。客户端通过 AT 指令向服务器提出请求，请求服务器的某种服务。在本文档中 AT 服务器指 ZSL420-EVB demo 板，AT 客户端指 PC 设备或其他主控设备。

客户端与服务器的通讯过程为：客户端向服务器发送 AT 指令，服务器在接收到指令后执行指令对应的操作，最后将执行的结果（一般将其称为响应）返回，通讯过程见 图 4.1。

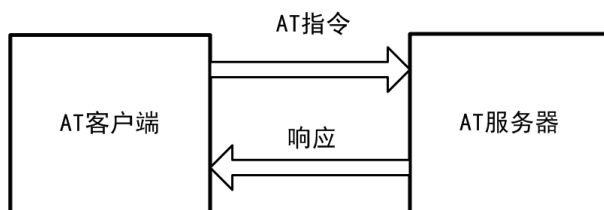


图 4.1 AT 客户端请求指令流程

此外，在一些特殊情况下，服务器会主动向客户端发送一串特定格式的字符串（一般将其称为 URC 指示），以上报当前的情况，如上报 LoRaWAN 服务器的主动下行，通讯过程见 图 4.2。

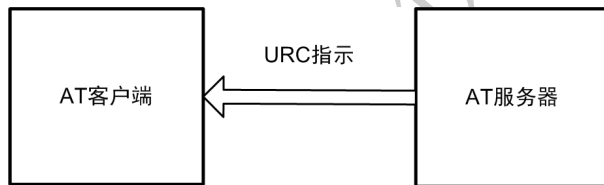


图 4.2 AT 服务器发送 URC 指示

注：为方便叙述，在本文档中将 ZSL420-EVB 称为 AT 服务器，将主控制器设备称为 AT 客户端。

4.2 AT 指令格式

4.2.1 AT 指令

AT 指令的完整格式为：前缀 + 指令名 + 请求 + 参数 + 结束符。其中前缀固定为 AT；结束符固定为“\r\n”；指令名用来区分不同的 AT 指令，例如 BAUD 为波特率等等；请求类型一共分为测试、查询、设置、执行四类，其中测试、执行、查询不带参数，设置可带参数。格式如 表 4.1 所示。

表 4.1 AT 指令格式

请求	格式	描述	举例
测试	AT+<CMD>=?	用来查询指令的使用方法	AT+TEST=?
查询	AT+<CMD>?	用来查询参数	AT+BAUD?
设置	AT+<CMD>=<PARAMS>	用来设置参数	AT+BAUD=115200
执行	AT+<CMD>	执行某一操作	AT+L

并不是每一指令均实现了这 4 个请求，一般只实现了一个或几个，根据具体的指令决定。

4.2.2 URC 指示

URC 指示一般用来返回异步操作的结果和主动上报接收的数据，其格式一般为：“+<CMD>,<PARAMS>”。其中“<CMD>”为对应的 AT 指令，例如“+TX,<PARAMS>”是对 AT 指令“AT+TX=<PARAMS>”执行结果的上报。

```
AT+TX=1,0
OK
+TX,"OK",2,1,0
```

4.2.3 响应错误

响应错误码主要是在指令执行期间遇见的指令解析错误、指令参数错误、指令执行错误等。其返回值与含义如表 4.2 所示，错误码对应含义可通过 AT+H 查询，如表 6.98 所示，

表 4.2 返回值与响应结果的关系

响应结果字符串	含义
“OK\r\n”	成功处理，无错误
“ERROR,-1\r\n”	有一个普通错误
“ERROR,-2\r\n”	AT 指令格式错误或不能执行
“ERROR,-3\r\n”	AT 指令参数格式错误
“ERROR,-4\r\n”	AT 指令参数内容解析错误

4.3 透传通信模式

透传模式下终端转发应用数据，数据以 HEX 格式进行传输，主控制器直接使用串口发送待上行至服务器的应用数据并从串口接收服务器的下行数据。该模式简化了数据交互过程。对于下行数据输出格式，主控制器可以使用 6.1.34 节所示的 AT+TRANSOUT 进行配置

4.4 透传通信格式

4.4.1 发送数据

串口数据直接采用 HEX 格式发送。参考伪代码如程序清单 4.1 所示。

程序清单 4.1 透传模式发送数据参考

```

1  /**
2   * \brief 透传模式下写入数据
3   *
4   * \param [in] p_uart   : 平台串口设备
5   * \param [in] p_buffer : 待发送的数据缓存区地址
6   * \param [in] size     : 数据大小
7   *
8   * \return 实际写入个数
9   */
10 static int __lm400tu_trans_write (void *p_uart, void *p_buffer, uint8_t size)
11 {
12     int error;
13     uint8_t index = 0;
14
15     /* 检查最大帧长 */
16     error = __lm400tu_max_nbytes_check(size);
17     if (error != OK) {
18         return 0;
19     }
20
21     /* 等待终端空闲 */
22     while (BUSY == 0);
23
24     /* 循环写入 */
25     for (index = 0; index < size; index++) {
26         while (__uart_idle(p_uart) == false);
27         error = __uart_poll_putchar(p_uart, p_buffer[index]);
28         if (error == ERROR) {
29             break;
30         }
31     }
32     return index;
33 }

```

4.4.2 输出格式 0

在格式 0 时，只输出收到的应用数据负载，不输出任何其他信息，接收可参考伪代码 程序清单 4.2。

程序清单 4.2 输出格式 0 数据接收

```

1  /** \brief 接收数据缓存区 */
2  static uint8_t __g_data[512] = 0;
3  /** \brief 接收数据大小 */
4  static uint16_t __g_size = 0;
5  /** \brief 接收完成标志 */
6  static bool __g_flag = 0;
7  /** \brief 软件定时器 */
8  static timer_t __g_timer;
9
10 /**
11  * \brief 字符超时定时器回调
12  */
13 static void __char_timeout (void)
14 {
15     __timer_stop(&__g_timer);
16
17     __g_flag = true;
18 }

```

```
19
20  /**
21   * \brief 串口接收中断处理
22   */
23  static void __uart_rx_interrupt (void)
24  {
25      __g_data[__g_size++] = __uart_rx_data();
26
27      __timer_restart(&__g_timer, CHAR_TIMEOUT);
28  }
29
30  /**
31   * \brief 串口接收中断处理
32   */
33  static void __lm400tu_trans_read (void)
34  {
35      if (__g_flag == false) {
36          return;
37      }
38
39      /* 接收数据 __g_data, 长度 __g_size */
40
41      __g_flag = false;
42      __g_size = 0;
43  }
```

4.4.3 输出格式 1

格式 1 的输出数据如 表 4.3 所示。

表 4.3 透传模式下数据输出格式 1

1 Byte	4 Byte	1 Byte	0 ~ N Byte
状态，其值如 表 4.5 所示	设备地址	数据端口	数据负载

其中对于长度不为 1 字节的数据采用低字节在前，高字节在后的顺序发送。输出解析可参考伪代码 程序清单 4.3 。

程序清单 4.3 输出格式 1 解析参考

```
1  /**
2   * \brief 透传输出格式 1
3   */
4  struct __trans_out1 {
5      uint8_t  status;
6      uint32_t address;
7      uint8_t  port;
8      uint8_t  data[242];
9      uint8_t  size;
10 };
11
12 /**
13  * \brief 输出格式 1 解析
14  */
15 #define __GET_32UINT(p_dat) (((p_dat)[3] << 24) | \
16                             ((p_dat)[2] << 16) | \
17                             ((p_dat)[1] << 8) | \
18                             (p_dat)[0])
```

```
19
20  /**
21   * \brief 透传模式下数据发送
22   *
23   * \param [out] p_out      : 输出信息
24   * \param [in]  p_buffer   : 接收到的数据缓存区地址
25   * \param [in]  size       : 接收数据大小
26   *
27   * \return 状态
28   */
29  static int __lm400tu_trans_out1_parser (struct __trans_out1 *p_out,
30                                         uint8_t             *p_buffer,
31                                         uint16_t             size)
32  {
33      uint16_t index = 0;
34
35      /* status */
36      if (size == 0) {
37          return ERROR;
38      } else {
39          p_out->status = p_buffer[index++];
40      }
41
42      /* address */
43      if ((size - index) >= 4) {
44          p_out->address = __GET_32UINT(&p_buffer[index]);
45          index += 4;
46      }
47
48      /* port & data */
49      if ((size - index) > 0) {
50          p_out->port = p_buffer[index++];
51          p_out->size = size - index;
52          memcpy(p_out->data, &p_buffer[index], size - index);
53      }
54
55      return OK;
56  }
```

4.4.4 输出格式 2

格式 2 的输出数据如 表 4.4 所示。

表 4.4 透传模式下数据输出格式 2

1 Byte	4 Byte	4 Byte	2 Byte	1 Byte	1 Byte	1 Byte	0 ~ N Byte
状态，其值如 表 4.5 所示	下行计数器	设备地址	RSSI	SNR	帧类型，同 表 6.133 中的 TYPE	数据端口	数据负载

其中对于长度不为 1 字节的数据采用低字节在前，高字节在后的的顺序发送。输出解析可参考伪代码 程序清单 4.4 。

程序清单 4.4 输出格式 2 解析参考

```
1  /**
2   * \brief 透传输出格式 2
3   */
4  struct __trans_out2 {
```

```

5      uint8_t    status;
6      uint32_t   counter;
7      uint32_t   address;
8      int16_t    rssi;
9      int8_t     snr;
10     uint8_t     type;
11     uint8_t     port;
12     uint8_t     data[242];
13     uint8_t     size;
14 };
15
16 /**
17  * \brief 获取 32 位数
18  */
19 #define __GET_32bit(p_dat)  (((p_dat)[3] << 24) | \
20                             ((p_dat)[2] << 16) | \
21                             ((p_dat)[1] << 8)  | \
22                             (p_dat)[0])
23
24 /**
25  * \brief 获取 16 位数
26  */
27 #define __GET_16bit(p_dat)  (((p_dat)[1] << 8) | \
28                             (p_dat)[0])
29
30 /**
31  * \brief 输出格式 2 解析
32  *
33  * \param [out] p_out      : 输出信息
34  * \param [in]  p_buffer   : 接收到的数据缓存区地址
35  * \param [in]  size       : 接收数据大小
36  *
37  * \return 状态
38  */
39 static int __lm400tu_trans_out2_parser (struct __trans_out2 *p_out,
40                                         uint8_t             *p_buffer,
41                                         uint16_t             size)
42 {
43     uint16_t index = 0;
44
45     /* status */
46     if (size == 0) {
47         return ERROR;
48     } else {
49         p_out->status = p_buffer[index++];
50     }
51
52     /* counter/address/rssi/snr/type 固定存在 */
53     if ((size != 1) && ((size - index) < 12)) {
54         return ERROR;
55     }
56
57     /* counter */
58     p_out->counter = __GET_32bit(&p_buffer[index]);
59     index += 4;
60     /* address */
61     p_out->address = __GET_32bit(&p_buffer[index]);
62     index += 4;
63     /* rssi */
64     p_out->rssi = __GET_16bit(&p_buffer[index]);
65     index += 2;
66     /* snr */
67     p_out->snr = p_buffer[index++];
68     /* type */

```

```
68     p_out->type      = p_buffer[index++];
69
70     /* port & data 有应用负载时存在 */
71     if ((size - index) > 0) {
72         p_out->port     = p_buffer[index++];
73         p_out->size     = size - index;
74         memcpy(p_out->data, &p_buffer[index], size - index);
75     }
76
77     return OK;
78 }
```

4.5 串口通信约定

- UART 通信默认参数为波特率 115200、8 数据位、1 停止位、无校验位，其中仅波特率可通过 AT 指令修改；
- 所有 AT 指令、URC 指示均以“\r\n”作为结束符；
- 一帧数据或 AT 指令最大长度不得超过 550 字节，其中包括结束符；
- 所有 AT 指令均有应答；
- 异步操作指令会先返回启动结果，执行结果通过 URC 指示返回；
- 指令大小写敏感；
- AT 指令的参数中十六进制数据格式以 #X 起始，例如：#X0123 表示 0x01、0x23；
- AT 指令的参数中字符串数据用“”包含，例如：“string”表示字符串 string；
- 串口通信字符超时默认为 10ms，字符间隔超过 10ms 即认为接收完成。

4.6 串口分帧格式

AT 模式下一帧即一个 AT 指令或 AT 响应、URC 指示，指令均以“\r\n”作为结束符。

透传模式下输出、输入都用字符超时时间作为一帧传输完成的依据，在发生字符超时后表示数据传输完成。

4.7 协议栈错误码

协议栈的错误码主要是在异步操作的结果返回，其值与含义定义如表 4.5 所示：

表 4.5 协议栈错误码

错误码	含义
1	参数无效
2	设备当前忙碌
3	该操作不支持
4	请求为空
5	设备未入网
6	网络错误
7	请求已存在
8	无应答
9	资源不足
10	下行帧接收错误

续上表

错误码	含义
11	下行帧 MHDR 错误
12	下行帧设备地址错误
13	下行帧 MIC 错误
14	下行帧 FCNT 异常
15	下行帧接收超时
16	下行帧多播错误
17	上行帧长度错误
18	上行帧发送错误
19	上行帧信道退让
20	上行帧发送超时
21	上行帧信道活跃
22	在发送或接收中被高优先级事件打断
23	未找到
24	信道掩码错误
25	协议栈入网复位
26	信道错误，无可信道
27	溢出错误
28	JoinNonce 错误
29	速率错误，当前速率应用及 CMD 均不能发送
30	下行帧指令存放冲突

5. 频段参数规范

5.1 CN470 - 510MHz

5.1.1 通信速率编码

CN470 - 510MHz 频段速率编码与 LoRa 速率的对应关系如 表 5.1 所示。

表 5.1 CN470 - 510MHz 频段速率编码

DR	配置	速率 bit/s
0	LoRa: SF12 / 125 kHz	250
1	LoRa: SF11 / 125 kHz	440
2	LoRa: SF10 / 125 kHz	980
3	LoRa: SF9 / 125 kHz	1760
4	LoRa: SF8 / 125 kHz	3125
5	LoRa: SF7 / 125 kHz	5470
6:15	RFU	

5.1.2 发射功率编码

CN470 - 510MHz 频段发射功率编码与发送功率的对应关系如 表 5.2 所示。

表 5.2 CN470 - 510MHz 频段发射功率编码

TXPower	配置
0	MaxEIRP
1	MaxEIRP - 2dB
2	MaxEIRP - 4dB
3	MaxEIRP - 6dB
4	MaxEIRP - 8dB
5	MaxEIRP - 10dB
6	MaxEIRP - 12dB
7	MaxEIRP - 14dB
8:15	RFU

对于 CN470 - 510MHz 频段，其 MaxEIRP 默认定义为 +19.15dBm，对于终端节点 ERP = EIRP - 2.15dB

5.1.3 通信帧长定义

CN470 - 510MHz 频段速率编码与允许的最大发送帧长对应关系如 表 5.3 所示。

表 5.3 CN470 - 510MHz 频段速率编码

DR	允许中继的最大允许长度	不允许中继的最大长度
0	51	51
1	51	51
2	51	51
3	115	115
4	222	242
5	222	242
6:15	RFU	RFU

在表 5.3 中的最大长度为数据帧中不包含 MAC 指令的长度,若协议栈中存在 MAC 指令请求或请求应答则需要减去相应长度。当前允许传输的最大长度可以使用 AT+DATLENGTH 指令获取。

ZSL420 使用内置晶振,在硬件上限制了 ZSL420 的最大发送帧长,对比表 5.3 和表 5.4。

表 5.4 设备的最大发送帧长

DR	ZSL420	ZSL421
0	x	255
1	x	255
2	10	255
3	60	255
4	230	255
5	255	255
6:15	RFU	RFU

注意



为满足 CN470 - 510MHz 频段的最大发送帧长需求, ZSL420 的速率编码只能设置为 DR4 或 DR5 且不能开启 ADR 功能。

6. 终端 AT 指令集

6.1 AT 协议栈指令集

协议栈 AT 指令指令如 表 6.1 、表 6.2 所示。

表 6.1 LoRaWAN 协议栈 AT 指令列表 1

AT 指令	指令名称	测试 (=?)	查询 (?)	设置 (=)	执行 (无)	AT+SAVE 存储有效
AT+CLASS	终端设备类型		√	√		√
AT+MODE	终端激活模式		√	√		√
AT+ID	OTAA 模式设备 ID 信息		√	√		√
AT+APPKEY	OTAA 模式 APPKEY		√	√		√
AT+SKEY	ABP 模式秘钥		√	√		√
AT+ADDRESS	ABP 模式设备地址		√	√		√
AT+ADR	终端 ADR 功能支持		√	√		√
AT+CHMASK	终端信道掩码设置		√	√		√
AT+PORT	终端当前通信端口		√	√		√
AT+DR	终端当前速率		√	√		仅 ADR 禁能时存储
AT+POWER	终端当前发射功率		√	√		仅 ADR 禁能时存储
AT+JOIN	入网参数保存与入网设置		√	√	√	√
AT+TX	终端发送请求			√		
AT+RX2	终端 RX2 窗口配置		√	√		√
AT+RXCNT	终端接收窗口数量		√	√		√
AT+LINKCHECK	缓存 LinkCheckReq				√	
AT+DATLENGTH	当前允许传输的最大字节		√			
AT+BUSYCHECK	协议栈忙碌检测		√			
AT+NETSTS	终端网络连接状态		√			
AT+MULTICASTADD	添加多播设备			√		√
AT+MULTICASTDEL	删除多播设备			√		√
AT+MULTICASTLIST	多播设备列表查询		√			
AT+CSMA	信道活跃检测功能支持		√	√		√
AT+BACKOFF	终端重发退让参数设置		√	√		√
AT+PERIODTX	终端设备周期发送设置			√		√

表 6.2 LoRaWAN 协议栈 AT 指令列表 2

AT 指令	指令名称	测试 (=?)	查询 (?)	设置 (=)	执行 (无)	AT+SAVE 存储有效
AT+AUTOACK	终端自动应答设置		√	√		√

续上表

AT 指令	指令名称	测试 (=?)	查询 (?)	设置 (=)	执行 (无)	AT+SAVE 存储有效
AT+VOLTAGE	终端电量测量参数设置		√	√		√
AT+ACKTIME	ACK 引脚配置		√	√		√
AT+CONFIRMED	配置上行帧类型		√	√		√
AT+NBTRANS	配置上行帧传输次数		√	√		√
AT+TXSTR	发送字符串			√	√	
AT+TXHEX	发送 HEX 数据			√	√	
AT+TRANSJOIN	透传模式自动入网次数		√	√		√
AT+REP	中继器开关		√	√		√
AT+REPMODE	设置终端工作模式		√	√		√
AT+REPWHLISTADD	添加中继器白名单			√		√
AT+REPWHLISTDEL	删除中继器白名单			√		√
AT+REPWHLIST	查询中继器白名单		√			
AT+REPFREQADD	添加中继信道			√		√
AT+REPFREQDEL	删除中继信道			√		√
AT+REPFREQ	查询中继信道		√			
AT+REPFREQLIVEQ	查询中继活跃信道		√			

6.1.1 AT+CLASS

使用该指令来查询及设置设备类。指令格式如 表 6.3 所示，指令参数如 表 6.4 所示。

表 6.3 AT+CLASS 指令使用方法

条目	指令	应答
设备类查询	AT+CLASS?	CLASS_VAL
设备类设置	AT+CLASS=CLASS_VAL	OK

表 6.4 AT+CLASS 参数项 CLASS_VAL 说明

CLASS_VAL	说明
A	Class A 设备
C	Class C 设备

使用示例：

```
AT+CLASS=A //设置设备为 A 类
OK //设置成功
AT+CLASS=C //设置设备为 C 类
OK //设置成功
AT+CLASS? //查询当前设备类型
C //返回当前设备类型为 C 类
OK //查询成功
```

6.1.2 AT+MODE

使用该指令来查询及设置设备入网方式。指令格式如 表 6.5 所示，指令参数如 表 6.6 所示。

表 6.5 AT+MODE 指令使用方法

条目	指令	应答
入网模式查询	AT+MODE?	MODE_VAL
入网模式设置	AT+MODE=MODE_VAL	OK

表 6.6 AT+MODE 参数项 MODE_VAL 说明

MODE_VAL	说明
OTAA	空中激活模式
ABP	本地激活模式

使用示例：

```
AT+MODE=ABP      //设置设备激活模式为 ABP 模式
OK               //设置成功
AT+MODE=OTAA     //设置设备激活模式为 OTAA 模式
OK               //设置成功
AT+MODE?         //查询当前设备激活模式
OTAA            //返回当前激活模式为 OTAA
OK               //查询成功
```

6.1.3 AT+ID

使用该指令来查询及设置设备在 OTAA 激活模式中使用的 APPEUI、DEVEUI。指令格式如 表 6.7 所示，指令参数如 表 6.8 所示。

表 6.7 AT+ID 指令使用方法

条目	指令	应答
ID 查询	AT+ID?	DEVEUI_VAL,APPEUI_VAL
ID 设置	AT+ID=TYPE_STR,ID_VAL	OK


表 6.8 AT+ID 参数说明

参数	说明
TYPE_STR	1、“DEVEUI”：设备唯一识别码；2、“APPEUI”：应用唯一识别码
ID_VAL	十六进制数据，#X0123456789ABCDEF。当参数 TYPE_STR 为“DEVEUI”时，该值为设备 DEVEUI；当参数 TYPE_STR 为“APPEUI”时，该值为设备 APPEUI
DEVEUI_VAL	输出的 DEVEUI，十六进制数据，#X0123456789ABCDEF
APPEUI_VAL	输出的 APPEUI，十六进制数据，#X0123456789ABCDEF

使用示例：

```
AT+ID="DEVEUI",#X1122334455667788 //设置设备唯一识别码, DEVEUI
OK //设备唯一识别码设置成功
AT+ID="APPEUI",#XAABBCCDDEEFF0011 //设置应用唯一识别码, APPEUI
OK //应用唯一识别码设置成功
AT+ID? //查询设备 ID
#X1122334455667788,#XAABBCCDDEEFF0011 //返回的当前 ID 信息
OK //查询成功
```

注意

 **#X** 前缀表示 16 进制的数据串。AT+ID 指令只有在 OTAA 激活模式下有效，当处于 ABP 模式时该指令会返回错误

6.1.4 AT+APPKEY

使用该指令来查询及设置设备在 OTAA 激活模式中使用的应用根秘钥 APPKEY。指令格式如表 6.9 所示，指令参数如表 6.10 所示。

表 6.9 AT+APPKEY 指令使用方法

条目	指令	应答
应用根秘钥查询	AT+APPKEY?	APPKEY_VAL
应用根秘钥设置	AT+APPKEY=APPKEY_VAL	OK


表 6.10 AT+APPKEY 参数说明

参数	说明
APPKEY_VAL	应用根秘钥，十六进制，#X 开头，例如 #X00112233445566778899AABBCCDDEEFF

使用示例：

```
AT+APPKEY=#X00112233445566778899AABBCCDDEEFF //设置应用根秘钥
OK //设置成功
AT+APPKEY? //查询应用根秘钥
#X00112233445566778899AABBCCDDEEFF //当前应用根秘钥
OK //查询成功
```

注意

 AT+APPKEY 指令只有在 OTAA 激活模式下有效，当处于 ABP 模式时该指令会返回错误

6.1.5 AT+SKEY

使用该指令来设置设备在 ABP 模式下的会话秘钥及查询设备在 ABP 模式下的会话秘钥。指令格式如表 6.11 所示，指令参数如表 6.12 所示。

表 6.11 AT+SKEY 指令使用方法

条目	指令	应答
ABP/OTAA 模式会话密钥查询	AT+SKEY?	APPSKEY_VAL,NWKSKEY_VAL
ABP 模式会话密钥设置	AT+SKEY=TYPE_STR,SKEY	OK

表 6.12 AT+SKEY 参数说明

参数	说明
TYPE_STR	1、“APPSKEY”：应用会话密钥; 2、“NWKSKEY”：网络会话密钥
SKEY	密钥值，十六进制，#X 开头，例如 #X00112233445566778899AABBCCDDEEFF。当参数 TYPE_STR 为“APPSKEY”时，该值为应用会话密钥 APPSKEY；当参数 TYPE_STR 为“NWKSKEY”时，该值为网络会话密钥 NWKSKEY
APPSKEY_VAL	应用密钥值，十六进制，#X 开头
NWKSKEY_VAL	网络密钥值，十六进制，#X 开头

使用示例：

```
AT+SKEY="NWKSKEY",#X00112233445566778899AABBCCDDEEFF //设置设备网络会话密钥
OK //设置成功
AT+SKEY="APPSKEY",#X00112233445566778899AABBCCDDEEFF //设置设备应用会话密钥
OK //设置成功
AT+SKEY? //查询设备会话密钥
#X00112233445566778899AABBCCDDEEFF,#X00112233445566778899AABBCCDDEEFF //设备密钥
OK //查询成功
```

注意

⚠ AT+SKEY 指令只有在 ABP 激活模式下有效，当处于 OTAA 模式时使用该指令会返回错误。

6.1.6 AT+ADDRESS

使用该指令来设置设备在 ABP 模式下的设备地址及查询设备在 ABP、OTAA 模式下的设备地址。指令格式如 表 6.13 所示，指令参数如 表 6.14 所示。

表 6.13 AT+ADDRESS 指令使用方法

条目	指令	应答
ABP/OTAA 模式设备地址查询	AT+ADDRESS?	VAL
ABP 模式设备地址设置	AT+ADDRESS=VAL	OK

表 6.14 AT+ADDRESS 参数说明

参数	说明
VAL	设备地址，十六进制，#X 开头，例如设备地址为 0x12345678，则输入 #X12345678

使用示例：

```
AT+ADDRESS=#X11223344 //设置设备地址为 0x11223344
OK //设备地址设置成功
AT+ADDRESS? //查询设备地址
#X11223344 //设备地址
OK //查询成功
```

6.1.7 AT+ADR

使用该指令来查询及设置设备 ADR 功能是否支持。指令格式如 表 6.15 所示，指令参数如 表 6.16 所示。

表 6.15 AT+ADR 指令使用方法

条目	指令	应答
ADR 查询	AT+ADR?	ADR_EN
ADR 设置	AT+ADR=ADR_EN	OK

表 6.16 AT+ADR 参数说明

参数	说明
ADR_EN	0: 禁能/1: 使能

使用示例：

```
AT+ADR=1 //使能 ADR
OK //设置成功
AT+ADR? //查询当前 ADR 使能情况
1 //当前 ADR 状态为使能
OK //查询成功
```

注意

 ZSL420 禁止使用 ADR 功能，设置使能将返回错误，可参考 5 章节。

6.1.8 AT+CHMASK

使用该指令来查询及设置设备当前所使用的信道。指令格式如 表 6.17 所示，指令参数如 表 6.18 所示。该信道掩码同时也是入网所使用的信道掩码。

以 CN470 频段为例：起始频点为：470.3MHz，间隔 0.2M，终止频点为 489.3MHz，因而总共 96 个信道，为了方便设置，我们使用十六进制数的每一位代表一个信道，因而该指令为 AT+CHMASK=#XXXXXXXXXXXXXXXXXXXX，总共 12 个字节（12*8=96 信道），最低位代表 470.3M，如想使用 471.5M~472.9M，那么指令为：AT+CHMASK=#X000000000000000000003FC0。

表 6.17 AT+CHMASK 指令使用方法

条目	指令	应答
通信信道掩码查询	AT+CHMASK?	LRMASK
通信信道掩码设置	AT+CHMASK=LRMASK	OK


表 6.18 AT+CHMASK 参数说明

参数	说明
LRMASK	信道掩码值，十六进制，#X 开头，指示 96 个上行信道。如 #X0000000000000000000000000000FF

使用示例：

```
AT+CHMASK=#X0000000000000000000000000000FF //设置 0-7 号信道，即：470.3MHz ~ 471.7MHz
OK //设置成功
AT+CHMASK? //查询当前的通信信道
#X0000000000000000000000000000FF //当前通信信道为 0-7
OK //查询成功
```

注意

 通信信道掩码不能设置为全 0，否则将返回错误。

6.1.9 AT+PORT

使用该指令来查询及设置设备上端口，端口有效值为 1~223。指令格式如 表 6.19 所示，指令参数如 表 6.20 所示。

表 6.19 AT+PORT 指令使用方法

条目	指令	应答
上行端口查询	AT+PORT?	NUM
上行端口设置	AT+PORT=NUM	OK

表 6.20 AT+PORT 参数说明

参数	说明
NUM	上行端口号 (1~223)

使用示例：

```
AT+PORT=99 //设置上行帧端口为 99
OK //设置成功
AT+PORT? //查询当前上行帧端口
99 //当前上行帧端口为 99
OK //查询成功
```

6.1.10 AT+DR

使用该指令来查询及设置设备的上行数据速率 (仅 ADR 禁能才能设置，若有保存至 flash 则会自动恢复，将影响入网使用的速率)。指令格式如 表 6.21 所示，指令参数如 表 6.22 所示。

表 6.21 AT+DR 指令使用方法

条目	指令	应答
数据速率查询	AT+DR?	VALUE
数据速率设置	AT+DR=VALUE	OK

表 6.22 AT+DR 参数说明

参数	说明
VALUE	DR 值, 可选参数为 0~5, 分别对应 SF12/125KHz-SF7/125KHz, 可参考 5 章节

使用示例:

```
AT+DR=5      //设置上行数据速率为 DR5, SF7/BW125KHz
OK           //设置成功
AT+DR?       //查询当前上行数据速率
5            //当前上行数据速率为 DR5
OK           //查询成功
```

注意

⚠ ZSL420 速率只能设置为 DR4 或 DR5, 否则会返回错误, 可参考 5 章节。

6.1.11 AT+POWER

使用该指令来查询及设置设备的发送功率 (仅 ADR 禁能才能设置, 若有保存至 flash 则会自动恢复, 将影响入网使用的功率)。指令格式如 表 6.23 所示, 指令参数如 表 6.24 所示。

表 6.23 AT+POWER 指令使用方法

条目	指令	应答
发射功率查询	AT+POWER?	VALUE
发射功率设置	AT+POWER=VALUE	OK

表 6.24 AT+POWER 参数说明

参数	说明
VALUE	功率编码值, 可选值为 0~7, 其中与功率转换公式可参考 5 章节

使用示例:

```
AT+POWER=0   //设置发送功率编码为 0(17dB)
OK           //设置成功
AT+POWER?    //查询当前发送功率
0            //当前发送功率为 0(17dB)
OK           //查询成功
```

6.1.12 AT+TRANSJOIN

当终端设备进入透传模式后, 若终端设备处于连接丢失状态则会自动入网, 使用该指令来指示透传模式下自动入网的最大尝试次数, 入网结果可以通过 AT+NETSTS 指令查询。指

令格式如表 6.25 所示，指令参数如表 6.26 所示。

表 6.25 AT+TRANSJOIN 指令使用方法

条目	指令	应答
透传模式入网尝试次数查询	AT+TRANSJOIN?	MAX_TIMES
透传模式入网尝试次数设置	AT+TRANSJOIN=MAX_TIMES	OK

表 6.26 AT+TRANSJOIN 参数说明

参数	说明
MAX_TIMES	范围 0 ~ 255, 0 表示禁止自动入网，默认自动入网次数: 3

使用示例：

```
AT+TRANSJOIN=2 //设置透传模式下自动入网最多尝试 2 次
OK //设置成功
AT+TRANSJOIN? //透传模式下自动入网次数查询
2 //当前最多尝试 2 次
OK
```

6.1.13 AT+JOIN

在 OTAA 模式下，使用该指令启动设备入网操作，以及设置 OTAA 模式下是否保存入网状态参数，指令格式如表 6.27 所示。

- 入网状态参数不保存模式：设备入网后，掉电或复位 **需要**重新发起入网。
- 入网状态参数保存模式：设备入网后，掉电或复位 **不需要**重新发起入网，即可上电就数据收发。

注意



恢复出厂后，默认为不保存模式。

选择使用入网参数保存模式，每次设备入网成功都会自动保存入网状态参数，如网络会话密钥、ADDRESS 和网络状态等，设备重新上电或复位后不需要重新入网即可上行数据至云端。在使用该模式前，应确保设备的类型 (CLASS)、入网方式、APPEUI、DEVEUI、APPKEY 参数已经保存完毕，如对这些参数进行了修改，应使用 AT+SAVE 保存后再执行入网操作。

表 6.27 AT+JOIN 指令使用方法

条目	指令	应答
入网模式查询	AT+JOIN?	MODE_VAL
启动入网激活	AT+JOIN	OK
设置入网模式并启动入网激活	AT+JOIN=MODE_VAL	OK

表 6.28 AT+JOIN 参数项 MODE_VAL 说明

MODE_VAL	说明
0	入网参数不保存模式
1	入网参数保存模式

执行一次 AT+JOIN=MODE_VAL 后，会自动保存入网参数保存模式。例如执行了一次 AT+JOIN=1，此后每次使用 AT+JOIN 指令入网都会保存入网参数，直到执行 AT+JOIN=0 退出入网参数保存模式。

入网操作为异步操作，应答“OK”表示启动成功，入网成功或失败会通过 URC 指令 +JOIN 发送给 AT 客户端，该 URC 指令格式和参数可参考 6.5.4 章节。

执行一次 AT+JOIN、AT+JOIN=0 或 AT+JOIN=1 则为发送一次 Join-request，入网失败不会继续尝试。

使用示例 1:

```
AT+JOIN=0           //设置入网参数不保存模式并执行入网操作
OK                  //启动成功
+JOIN,"Connect"     //入网成功不保存入网参数

AT+JOIN?            //查询当前入网参数保存模式
0                   //当前为入网参数不保存模式
OK                  //查询成功

AT+JOIN             //开始执行入网操作
OK                  //启动成功
+JOIN,"Connect"     //入网成功不保存入网参数
```

使用示例 2:

```
AT+JOIN=1           //设置入网参数保存模式并执行入网操作
OK                  //启动成功
+JOIN,"Connect"     //入网成功并保存入网参数

AT+JOIN?            //查询当前入网参数保存模式
1                   //当前为入网参数保存模式
OK                  //查询成功

AT+JOIN             //开始执行入网操作
OK                  //启动成功
+JOIN,"Connect"     //入网成功并保存入网参数
```

注意



1. AT+JOIN 指令只有在 OTAA 激活模式下有效，当处于 ABP 模式时该指令会返回错误。
2. AT+JOIN 与 AT+JOIN=MODE_VAL 的区别：AT+JOIN 以当前入网参数保存模式执行入网操作而不设置入网参数保存模式，AT+JOIN=MODE_VAL 既设置入网参数保存模式也执行入网操作。
3. 使用入网参数保存模式，每次入网成功都会保存入网参数到 Flash 中。为降低 Flash 的擦写次数提高使用寿命，在择使用入网参数保存模式时应尽量减少不必要的入网操作。

6.1.14 AT+NETSTS

使用该指令获取当前终端的连接状态。指令格式如 表 6.29 所示，指令参数如 表 6.30 所示。

表 6.29 AT+NETSTS 指令使用方法

条目	指令	应答
查询当前连接状态	AT+NETSTS?	STATUS

表 6.30 AT+NETSTS 参数项 STATUS 说明

STATUS	说明
“JOIN”	正在执行搜网流程
“CONNECT”	处于连接模式
“DISCONNECT”	处于连接丢失模式
“LOSE”	连接丢失, 将在自动重试入网

使用示例：

```
AT+NETSTS?           //查询当前连接状态
"CONNECT"           //连接状态
OK                  //查询成功
```

6.1.15 AT+TX

使用该指令来请求数据发送，指令格式如 表 6.31 所示，指令参数如 表 6.32 所示。

表 6.31 AT+TX 指令使用方法

条目	指令	应答
发送数据帧	AT+TX=TYPE,RETRY,DAT	OK

表 6.32 AT+TX 参数说明

参数	说明
TYPE	帧类型：0：非确认帧；1：确认帧；2：私有帧
RETRY	确认帧为最大重发次数，非确认帧为发送次数；为 0 时确认帧表示不重发，非确认帧表示不发送
DAT	待发送的数据。可选参数，不存在时发送空数据。字符串数据格式：“dat”；HEX 数据格式：#Xxx

发送操作为异步操作，应答“OK”表示启动成功，协议栈上行成功或失败会通过 URC 指令 +TX 发送给 AT 客户端，该 URC 指令格式和参数可参考 6.5.1 章节。

使用示例：

```
AT+TX=0,1,"hello!" //非确认帧发送字符串 hello! 到服务器
OK //发送启动成功
+TX,"OK",0,0,1 //上行成功，帧计数器为 0，未收到 ACK，发送 1 次
AT+TX=0,1,#X000102 //非确认帧发送 HEX 数据 0x00 0x01 0x02 到服务器
OK //发送启动成功
+TX,"OK",1,0,1 //上行成功，帧计数器为 1，未收到 ACK，发送 1 次
AT+TX=1,0 //确认帧发送空帧
OK //发送启动成功
+TX,"OK",2,1,0 //上行成功，帧计数器为 2，收到 ACK，重发 0 次
+RX,"OK",0,#X076DF072,-50,8,3,0,0 //收到的应答数据
```

注意

⚠ 使用该指令发送 RETRY 次非确认帧数据，帧计数器只增加 1 次并不会增加 RETRY 次，与确认帧的重发机制一致。

6.1.16 AT+RX2

使用该指令来设置和获取当前终端 RX2 窗口速率、频率、使用参数类型，指令格式如表 6.33 所示，指令参数如表 6.34 所示。

表 6.33 AT+RX2 指令使用方法

条目	指令	应答
RX2 窗口查询	AT+RX2?	USING,USER_DR,USER_FREQ
RX2 窗口配置	AT+RX2=USING,USER_DR,USER_FREQ	OK

表 6.34 AT+RX2 参数说明

参数	说明
USING	1: 使用用户设置的参数，0: 使用由协议栈和服务器共同维护的参数
USER_DR	用户设置的 DR 值，可选参数为 0~5，分别对应 SF12/125KHz-SF7/125KHz，可参考 5 章节。该参数为可选参数，存在时将同步保存的用户参数，与 USING 无关；不存在时对于 USING 为 1 将使用上次设定的值
USER_FREQ	使用的信道频率，可选频率为 500.3~509.7，单位 MHz。该参数为可选参数，存在时将同步保存的用户参数，与 USING 无关；不存在时对于 USING 为 1 将使用上次设定的值

在初次上电和 OTAA 模式入网后将恢复到使用协议栈和服务器共同维护的参数，即 USING = 0。

使用示例：

```
AT+RX2=1,0,505300000 //配置 RX2 窗口使用用户设置的参数: DR0, 505.3MHz
OK //配置成功
AT+RX2? //查询当前窗口配置情况
1,0,505300000 //当前 RX2 窗口使用用户设置的参数, 用户设置参数为 DR0, 505.
→3MHz
OK //查询成功
```

6.1.17 AT+RXCNT

该指令用来查询及设置终端设备上行后开启窗口的数量, 允许使用该指令裁剪用不到的窗口以降低功耗、缩短上行流程时间。指令格式如表 6.35 所示, 指令参数如表 6.36 所示。

表 6.35 AT+RXCNT 指令使用方法

条目	指令	应答
接收窗口数量查询	AT+RXCNT?	VALUE
接收窗口数量配置	AT+RXCNT=VALUE	OK

表 6.36 AT+RXCNT 参数说明

参数	说明
VALUE	窗口数量, 取值范围 0-2。0: RX1、RX2 窗口均不开启; 1: 只开启 RX1, 关闭 RX2 窗口; 2: RX1、RX2 窗口均正常开启

使用示例:

```
AT+RXCNT=2 //配置 RX1、RX2 窗口正常打开
OK //配置成功
AT+RXCNT? //查询当前窗口配置情况
2 //当前 RX1、RX2 窗口正常打开
OK //查询成功
```

注意



1. Class C 设备 RX1 和 RX2 窗口禁止关闭, 否则会返回错误。
2. 在设置 AT+CLASS=C 的同时 RX1 和 RX2 窗口会自动设置为开启, 在使用中应注意 RX1 和 RX2 窗口的开关变化。

6.1.18 AT+LINKCHECK

使用该指令来缓存一个 LoRaWAN LinkCheckReq MAC 指令。指令格式如表 6.37 所示。

表 6.37 AT+LINKCHECK 指令使用方法

条目	指令	应答
缓存 LinkCheckReq	AT+LINKCHECK	OK

该指令的作用仅为向 MAC 指令缓存区中添加一条 LinkCheckReq 指令, 并不会启动发送。发送需要使用 AT+TX 发起上行请求, 在上行时会将 MAC 指令缓存区中所有的

指令携带发送。LinkCheckReq 的应答指令 LinkCheckAns 会在终端收到后采用 URC 指示 +LINKCHECK 发送给 AT 客户端，该 URC 指令格式和参数可参考 6.5.3 章节。完整指令上行流程可参考 8.4 章节。

```
AT+LINKCHECK      //缓存 LinkCheckReq 指令
OK                //指令缓存成功
```

6.1.19 AT+DATLENGTH

使用该指令获取当前终端可以发送的负载数据长度。根据 LoRaWAN 规范，不同速率下数据负载的上限不同，对于开启 ADR 的终端设备，由于速率受服务器调控，因此应该注意使用该指令获取当前允许的最大发送长度，避免发送失败和数据被截断。指令格式如表 6.38 所示，指令参数如表 6.39 所示。

表 6.38 AT+DATLENGTH 指令使用方法

条目	指令	应答
查询当前可以发送的负载长度	AT+DATLENGTH?	LENGTH

表 6.39 AT+DATLENGTH 参数说明

参数	说明
LENGTH	当前可以发送的负载长度，若为 0 则表明指令缓存区内缓存的指令长度超过了 15 个字节，下一帧不能携带数据，此时应该发上行确认帧清空缓存区

使用示例：

```
AT+DATLENGTH?    //查询当前可以发送的负载长度
115              //当前允许的最大负载长度为 115 字节
OK               //查询成功
```

注意

⚠️ 最大负载长度和当前终端速率及缓存的 MAC 指令长度有关

6.1.20 AT+BUSYCHECK

使用该指令查询协议栈是否忙碌。指令格式如表 6.40 所示，指令参数如表 6.41 所示。

表 6.40 AT+BUSYCHECK 指令使用方法

条目	指令	应答
查询协议栈是否忙碌	AT+BUSYCHECK?	BUSY

表 6.41 AT+BUSYCHECK 参数说明

参数	说明
BUSY	1: 忙碌、0: 空闲

使用示例:

```
AT+BUSYCHECK? //查询协议栈是否忙碌
0 //空闲
OK //查询成功
```

6.1.21 AT+MULTICASTADD

使用该指令来添加一个多播设备，当前最多支持 4 个多播设备。指令格式如 表 6.42 所示，指令参数如 表 6.43 所示。

表 6.42 AT+MULTICASTADD 指令使用方法

条目	指令	应答
多播设备添加	AT+MULTICASTADD=ADDR,NWKS,APPS,COUNT	OK

表 6.43 AT+MULTICASTADD 参数说明

参数	说明
ADDR	多播设备地址
NWKS	多播设备网络会话密钥
APPS	多播设备应用会话密钥
COUNT	多播设备初始帧计数器值

Class C 使用示例:

```
AT+MULTICASTADD=#X063C65E9,#X00112233445566778899AABBCCDDEEFF,
->#X00112233445566778899AABBCCDDEEFF,0 //添加多播设备 0x063C65E9
OK //添加成功
```

6.1.22 AT+MULTICASTDEL

使用该指令来删除一个多播设备。指令格式如 表 6.44 所示，指令参数如 表 6.45 所示。

表 6.44 AT+MULTICASTDEL 指令使用方法

条目	指令	应答
多播设备删除	AT+MULTICASTDEL=ADDR	OK

表 6.45 AT+MULTICASTDEL 参数说明

参数	说明
ADDR	多播设备地址，当未添加该多播设备时会返回错误

使用示例:

```
AT+MULTICASTDEL=#X063C65E9 //删除多播设备 0x063C65E9
OK //删除成功
```

6.1.23 AT+MULTICASTLIST

使用该指令来获取所有支持的多播设备地址。指令格式如 表 6.46 所示，指令参数如 表 6.47 所示。

表 6.46 AT+MULTICASTLIST 指令使用方法

条目	指令	应答
多播设备列表查询	AT+MULTICASTLIST?	ADDRESS

表 6.47 AT+MULTICASTLIST 参数说明

参数	说明
ADDRESS	多播设备地址，#X....., #X.....

使用示例：

```
AT+MULTICASTLIST?           //查询多播设备
#X063C65E1, #X063C65E2, #X063C65E3 //支持多播设备 0x063C65E1, 0x063C65E2, 0x063C65E3
OK                           //查询成功
```

6.1.24 AT+CSMA

使用该指令来查询及设置设备是否使能信道活跃退让。指令格式如 表 6.48 所示，指令参数如 表 6.49 所示。

表 6.48 AT+CSMA 指令使用方法

条目	指令	应答
CSMA 查询	AT+CSMA?	CSMA_EN
CSMA 设置	AT+CSMA=CSMA_EN	OK

表 6.49 AT+CSMA 参数说明

参数	说明
CSMA_EN	0: 禁能，发送时直接发送;1: 使能，每次发送前先检测信道是否活跃，若信道活跃则进行退让

使用示例：

```
AT+CSMA=1           //使能 CSMA
OK                   //设置成功
AT+CSMA?             //查询当前 CSMA 设置情况
1                     //当前 CSMA 状态为使能
OK
```

6.1.25 AT+BACKOFF

使用该指令来设置终端重发退让时间计算参数，所谓的重发是指终端设备在没有收到应答时进行的再次发送，退让时间为本次接收超时时刻点到下一次发送时间，该设置也适用

于非确认帧。指令格式如 表 6.50 所示，指令参数如 表 6.51 所示。

表 6.50 AT+BACKOFF 指令使用方法

条目	指令	应答
退让计算参数查询	AT+BACKOFF?	MAX
退让计算参数设置	AT+BACKOFF=MAX	OK


表 6.51 AT+BACKOFF 参数说明

参数	说明
MAX	最大退让时间，单位 ms，默认值为 1000。在条件允许的情况下尽量调大该值，这样有利于解决信道冲突和拥堵。设置为 0 时表示不退让

使用示例：

```
AT+BACKOFF=20000 //设置最大退让时间为 20 秒
OK //设置成功
AT+BACKOFF? //最大退让时间查询
20000 //当前最大退让时间为 20 秒
OK
```

注意

 该重发时间是退让时间的最大值，并不是一定会退让那么长的时间。实际重发退让时间在 0~MAX 之间。

6.1.26 AT+PERIODTX

用户可使用该指令来设置终端设备周期性上报。上报采用非确认帧，上报数据包括用户自定义固定数据，AD0，AD1，以及全部的 IO 值，若终端设备接收到服务器下发数据后，通过 ACK 输出脉冲通知用户，之后并通过串口发送引脚输出接收数据。启动周期发送后也可以正常发送数据。指令格式如 表 6.52 所示，指令参数如 表 6.53 所示。

表 6.52 AT+PERIODTX 指令使用方法

条目	指令	应答
周期上报设置	AT+PERIODTX=EN,PERIOD,DR,PORT,APP_DAT	OK

表 6.53 AT+PERIODTX 参数说明

参数	说明
EN	使能或禁能，其值为 ON 或 OFF
PERIOD	可选参数，使能时必须包含。发送周期，范围 0~4294967295，单位 ms。此值是 RX2 和 TX 的间隔 (TX-RX1-RX2-TX.....)
PORT	可选参数，使能时必须包含。数据使用端口号
APP_DAT	可选参数，需携带的应用数据。字符串数据格式：“dat”；HEX 数据格式：#Xxx，使用方式同 AT+TX 指令

周期上行的数据格式如 表 6.54 所示。

表 6.54 AT+PERIODTX 指令上报的数据格式

字节	1 ~ len	len + 1	len + 2	len + 3	len + 4	len + 5
内容	APP_DAT	AD0(LSB)	AD0(MSB)	AD1(LSB)	AD1(MSB)	引脚状态 IO5(bit5) - IO0(bit0)

注意

⚠ AT+PERIODTX 指令上报的数据格式中 AD 与 IO 的配置数据占用了 5 个字节的发送数据长度，即可携带的应用数据（APP_DAT）长度为当前速率下的最大允许发送帧长（可使用 AT+DATLENGTH? 获取）再减去 5 个字节。

使用示例：

```

AT+PERIODTX=ON,10000,20,#X11223344 //开启周期上行（用户数据为 0x11,0x22,
→0x33,0x44），周期为 10s，端口号 20
OK //开启成功
+RX,"OK",1,#X076DF072,-53,6,3,0,0 //收到一个下行帧
+RX,"OK",2,#X076DF072,-50,9,3,0,0 //收到一个下行帧
+RX,"OK",3,#X076DF072,-50,9,3,0,0 //收到一个下行帧
+RX,"OK",4,#X076DF072,-53,8,3,0,0,1,#X12 //收到一个下行帧
AT+PERIODTX=OFF //关闭周期上行
OK //关闭成功
    
```

6.1.27 AT+AUTOACK

该指令用来设置终端是否开启自动应答，若打开，则终端在 Pending 置位、收到确认帧、需要上行帧时自动上行一个非确认帧。指令格式如 表 6.55 所示，指令参数如 表 6.56 所示。

表 6.55 AT+AUTOACK 指令使用方法

条目	指令	应答
AUTOACK 查询	AT+AUTOACK?	ENABLE
AUTOACK 设置	AT+AUTOACK=ENABLE	OK

表 6.56 AT+AUTOACK 参数说明

参数	说明
ENABLE	0: 禁能;1: 使能

使用示例：

```

AT+AUTOACK=1 //使能 AUTOACK
OK //设置成功
AT+AUTOACK? //查询当前 AUTOACK 设置情况
1 //当前 AUTOACK 状态为使能
OK
    
```

6.1.28 AT+VOLTAGE

使用该指令来设置终端电池最高电压和 ADC 基准电压，用来计算终端当前电量，适配协议栈的 DevStatusReq 指令。指令格式如 表 6.57 所示，指令参数如 表 6.58 所示。

表 6.57 AT+VOLTAGE 指令使用方法

条目	指令	应答
电压参数查询	AT+VOLTAGE?	VDD,VREF
电压参数设置	AT+VOLTAGE=VDD,VREF	OK

表 6.58 AT+VOLTAGE 参数说明

参数	说明
VDD	电池供电电压最大值, 取值范围 0 或 1800~3300 单位 mv, 例如 3.3V = 3300; 当设置为 0 时表示终端使用外部电源供电
VREF	ADC 基准电压, 即 VREFH 引脚接入的电压值, 取值范围 1800~3300

使用示例:

```
AT+VOLTAGE=0,2500 //当前使用外部电源供电, 不需要检测电量, ADC 基准电压为 2.5V
OK //设置成功
AT+VOLTAGE? //电压参数查询
0,2500 //当前使用外部电源, ADC 基准电压为 2.5V
OK
```

6.1.29 AT+ACKTIME

无论是 AT 模式还是透传模式, 在终端设备接收到服务器的下行数据时会先使用 ACK 引脚输出脉冲后再输出下行数据。使用该指令可以设置 ACK 脉冲宽度, 和 ACK 脉冲输出完成到数据输出的间隔时间。指令格式如 表 6.59 所示, 指令参数如 表 6.60 所示。

表 6.59 AT+ACKTIME 指令使用方法

条目	指令	应答
ACK 引脚参数查询	AT+ACKTIME?	WIDTH,DELAY
ACK 引脚参数设置	AT+ACKTIME=WIDTH,DELAY	OK

表 6.60 AT+ACKTIME 参数说明

参数	说明
WIDTH	ACK 输出脉冲宽度, 范围 0~10000, 单位毫秒
DELAY	ACK 脉冲输出完成后到串口开始输出数据的延时时间, 范围 0~65535, 单位毫秒

使用示例:

```
AT+ACKTIME=10,20 //设置脉冲宽度 10ms, 脉冲后 20ms 开始输出
OK //设置成功
AT+ACKTIME? //ACK 参数查询
10,20 //当前脉冲宽度 10ms, 脉冲后 20ms 开始输出
OK
```

6.1.30 AT+CONFIRMED

该指令用来指示透传模式及 AT 模式下 AT+TXSTR、AT+TXHEX 指令发送数据使用的数据类型。指令格式如 表 6.61 所示，指令参数如 表 6.62 所示。

表 6.61 AT+CONFIRMED 指令使用方法

条目	指令	应答
上行帧类型查询	AT+CONFIRMED?	CONFIRMED
上行帧类型设置	AT+CONFIRMED=CONFIRMED	OK

表 6.62 AT+CONFIRMED 参数说明

参数	说明
CONFIRMED	0：非确认帧；1：确认帧

使用示例：

```
AT+CONFIRMED=1 //设置透传模式及 AT 模式下 AT+TXSTR、AT+TXHEX 指令使用确认帧
OK //设置成功
AT+CONFIRMED? //上行使用的帧类型查询
1 //当前使用确认帧
OK
```

6.1.31 AT+NBTRANS

该指令用来指示透传模式及 AT 模式下 AT+TXSTR、AT+TXHEX 指令发送数据的最大发送次数。指令格式如 表 6.63 所示，指令参数如 表 6.64 所示。

表 6.63 AT+NBTRANS 指令使用方法

条目	指令	应答
上行帧传输次数查询	AT+NBTRANS?	NBTRANS
上行帧传输次数设置	AT+NBTRANS=NBTRANS	OK

表 6.64 AT+NBTRANS 参数说明

参数	说明
NBTRANS	范围 0~255，0 表示使用 LoRaWAN 协议规定的默认值

使用示例：

```
AT+NBTRANS=2 //设置透传模式及 AT 模式下 AT+TXSTR、AT+TXHEX 指令数据发送 2 次
OK //设置成功
AT+NBTRANS? //当前上行帧发送次数查询
2 //当前数据发送 2 次
OK
```

注意

该参数与 AT+TX 指令中的 RETRY 参数功能相似，发送确认帧时为最大重发次数，发送非确认帧时为发送次数。

6.1.32 AT+TXSTR

使用该指令来请求发送字符串数据，指令为 AT+TX 指令的精简指令，发送的数据类型由 AT+CONFIRMED 配置，发送数据次数由 AT+NBTRANS 配置，指令格式如 表 6.65 所示，指令参数如 表 6.66 所示。

表 6.65 AT+TXSTR 指令使用方法

条目	指令	应答
发送字符串	AT+TXSTR=DAT	OK
发送空帧	AT+TXSTR	OK

表 6.66 AT+TXSTR 参数说明

参数	说明
DAT	待发送的数据。格式：“dat”

发送操作为异步操作，应答“OK”表示启动成功，协议栈上行成功或失败会通过 URC 指令 +TX 发送给 AT 客户端，该 URC 指令格式和参数可参考 6.5.1 章节。

使用示例：

```
AT+TXSTR="hello!"           //发送字符串 hello! 到服务器
OK                           //发送启动成功
+TX,"OK",0,0,1              //上行成功，帧计数器为 0，未收到 ACK，发送 1 次
```

6.1.33 AT+TXHEX

使用该指令来请求发送 HEX 格式数据，指令为 AT+TX 指令的精简指令，发送的数据类型由 AT+CONFIRMED 配置，发送数据次数由 AT+NBTRANS 配置，指令格式如 表 6.67 所示，指令参数如 表 6.68 所示。

表 6.67 AT+TXHEX 指令使用方法

条目	指令	应答
发送 HEX 格式数据	AT+TXHEX=DAT	OK
发送空帧	AT+TXHEX	OK

表 6.68 AT+TXHEX 参数说明

参数	说明
DAT	待发送的数据。格式：#X000102

发送操作为异步操作，应答“OK”表示启动成功，协议栈上行成功或失败会通过 URC 指令 +TX 发送给 AT 客户端，该 URC 指令格式和参数可参考 6.5.1 章节。

使用示例：

AT+TXHEX=#X000102
OK
+TX,"OK",0,0,1

//发送 HEX 数据 0x00, 0x01, 0x02 到服务器
//发送启动成功
//上行成功, 帧计数器为 0, 未收到 ACK, 发送 1 次

6.1.34 AT+TRANSOUT

使用该指令来配置透传模式上行结果及下行数据输出格式。指令格式如 表 6.69 所示，指令参数如 表 6.70 所示。

表 6.69 AT+TRANSOUT 指令使用方法

条目	指令	应答
透传模式输出格式查询	AT+TRANSOUT?	FORMAT
透传模式输出格式设置	AT+TRANSOUT=FORMAT	OK

表 6.70 AT+TRANSOUT 参数说明

参数	说明
FORMAT	格式类型

1. 格式 0(FORMAT=0)

使用示例：

/.../
AT+TRANSOUT=0
OK
AT+TRANSOUT?
0
OK
/.../
0001020304
000102

//设置模式引脚切换至 AT 模式
//设置透传模式输出格式 0
//设置成功
//透传模式输出格式查询
//当前使用透传模式输出格式 0

//设置模式引脚切换至透传模式，串口配置为 HEX 格式发送
//发送 HEX 数据 0x00, 0x01, 0x02, 0x03, 0x04
//收到服务器下发的数据 0x00, 0x01, 0x02

2. 格式 1(FORMAT=1)

使用示例：

/.../
AT+TRANSOUT=1
OK
AT+TRANSOUT?
1
OK
/.../
0001020304
0043234202010001
0001020304
00

//设置模式引脚切换至 AT 模式
//设置透传模式输出格式 1
//设置成功
//透传模式输出格式查询
//当前使用透传模式输出格式 1

//设置模式引脚切换至透传模式，串口配置为 HEX 格式发送
//发送 HEX 数据 0x00, 0x01, 0x02, 0x03, 0x04
//收到服务器下发的数据 0x00, 0x01, 设备地址为 0x12422343，端口号为 0x01
//发送 HEX 数据 0x00, 0x01, 0x02, 0x03, 0x04
//成功，没有下行数据

3. 格式 2(FORMAT=2)

使用示例：


```
/. . ./ //设置模式引脚切换至 AT 模式
AT+TRANSOUT=2 //设置透传模式输出格式 2
OK //设置成功
AT+TRANSOUT? //透传模式输出格式查询
2 //当前使用透传模式输出格式 2
OK
/. . ./ //设置模式引脚切换至透传模式，串口配置为 HEX 格式发送
0001020304 //发送 HEX 数据 0x00, 0x01, 0x02, 0x03, 0x04
000100000043234212C5FF0703010001 //收到服务器下发的数据 0x00, 0x01, 设备地址为
0x12422343, 端口号为 0x01, 帧计数器 0x00000001, RSSI=-59, SNR=7, 帧为非确认下行帧
0001020304 //发送 HEX 数据 0x00, 0x01, 0x02, 0x03, 0x04
00 //成功，没有下行数据
```

6.1.35 中继指令集

- 中继器支持 Class A 操作。
- 中继器支持在 Class A 运行的终端设备。

注意



中继功能仅支持 Class A 设备。

在实际应用中，由于终端节点与网关之间的距离过大，导致终端与网关无法通信。因此我们增加了 LoRaWAN 中继，使用中继器转发终端数据连接网关通信。

在 LoRaWAN 中继功能中，中继器与终端都可以作为数据节点与网关通信。中继器为 Class A 设备无法接收与转发云端主动下发的数据。其中，终端设备具有一般模式，中继模式和自动切换模式三种终端工作模式。一般模式，主要应用于终端与网关距离固定且较近，连接信号良好的环境下；中继模式，应用于终端与网关距离固定，但是相距较远连接信号差的环境下；自动切换模式，应用于终端与网关距离不固定，连接信号时好时坏的环境下（该模式仅支持 OTAA 激活模式的终端使用）。终端多种工作模式的支持，满足了各种应用场景下的不同使用需求。

终端工作模式：

- 一般模式：设备只能使用一般信道（通过 AT+CHMASK 指令设置的信道）与网关通信。
- 中继模式：设备只能使用中继信道，通过中继器转发数据与网关入网通信，入网成功标记该中继信道为中继活跃信道，之后使用该中继活跃信道通过中继器转发数据与网关通信。
- 自动切换模式：设备优先使用一般模式尝试入网，入网成功后将使用一般模式与网关通信。一般模式入网失败则改用中继模式，通过中继器连接网关尝试入网，入网成功后将使用中继模式与网关通信。重新申请入网时，再从一般模式开始申请入网。

在中继网络中，存在多中继器与多终端的情况，多个中继器可能会同时向网关转发同一个终端数据，这造成了极大网络干扰。因此中继器添加了白名单功能，过滤掉非白名单中的终端数据。这样便可以解决因中继器转发所有的终端数据而造成的网络干扰问题。

中继 AT 指令使用参考 8.6 章节。

1. AT+REP

该指令用来设置设备是否使能为中继器，若使能，设备将作为中继器工作，否则设备作为终端工作。指令格式如 表 6.71 所示，指令参数如 表 6.72 所示。

表 6.71 AT+REP 指令使用方法

条目	指令	应答
REP 查询	AT+REP?	ENABLE
REP 设置	AT+REP=ENABLE	OK

表 6.72 AT+REP 参数说明

参数	说明
ENABLE	0: 禁能/1: 使能

使用示例：

```
AT+REP=1      //使能中继器
OK            //操作成功
AT+REP?       //查询当前 REP 设置情况
1            //当前 REP 状态为使能
OK
```

2. AT+REPMODE

该指令用来设置终端工作模式 (仅 REP 禁能才能设置)。指令格式如 表 6.73 所示，指令参数如 表 6.74 所示。

表 6.73 AT+REPMODE 参数说明

条目	指令	应答
REPMODE 查询	AT+REPMODE?	VALUE
REPMODE 设置	AT+REPMODE=VALUE	OK

表 6.74 AT+REPMODE 参数说明

参数	说明
VALUE	0: 一般模式，1: 中继模式，2: 自动切换模式

使用示例：

```
AT+REPMODE=1  //设置终端使用中继模式
OK            //操作成功
AT+REPMODE?   //查询当前 REPMODE 设置情况
1            //当前终端工作模式为中继模式
OK
```

注意



当终端工作模式设置为非一般模式时，设备 ADR 会禁能。

3. AT+REPWHLISTADD

该指令用来添加中继器的白名单 (仅 REP 使能才能设置, 若白名单为空中继电器默认转发所有终端数据, 否则只转发白名单中的终端的数据), 最多可添加 20 个。指令格式如 表 6.75 所示, 指令参数如 表 6.76 所示。

表 6.75 AT+REPWHLISTADD 指令使用方法

条目	指令	应答
REPWHLISTADD 设置	AT+REPWHLISTADD=VALUE	OK

表 6.76 AT+REPWHLISTADD 参数说明

参数	说明
VALUE	<ol style="list-style-type: none">需要添加的终端为 ABP 激活模式时, 则输入终端的 ADDR, 例如终端的 ADDR 为 0x12345678, 则输入 #X12345678。需要添加的终端为 OTAA 激活模式时, 则输入终端的 DEVEUI, 例如终端的 DEVEUI 为 0x1111111111111111, 则输入 #X1111111111111111。

使用示例:

```
AT+REPWHLISTADD=#X12345678,#X1111111111111111 //添加了 2 个终端设备到中继器白名单。  
OK //操作成功
```

注意



中继器白名单最多可添加 20 个设备。

4. AT+REPWHLISTDEL

该指令用来删除中继器白名单 (仅 REP 使能才能设置), 可以一次删除全部, 也可以选择任意个删除。指令格式如 表 6.77 所示, 指令参数如 表 6.78 所示。

表 6.77 AT+REPWHLISTDEL 指令使用方法

条目	指令	应答
REPWHLISTDEL 设置	AT+REPWHLISTDEL=VALUE	OK

表 6.78 AT+REPWHLISTDEL 参数说明

参数	说明
VALUE	<ol style="list-style-type: none">删除全部:ALL任意选择删除: 需要的删除的终端 ADDR 或 DEVEUI, 最多可连续输入 20 个

使用示例:

```
AT+REPWHLISTDEL=#X12345678,#x1111111111111111 //选择删除
OK //操作成功
AT+REPWHLISTDEL=ALL //删除中继器白名单中所有设备
OK //操作成功
AT+REPWHLIST? //查询中继器白名单
0 //中继白名单为空
OK
```

5. AT+REPWHLIST

该指令用来查询中继器白名单。指令格式如 表 6.79 所示，指令参数如 表 6.80 所示。

表 6.79 AT+REPWHLIST 指令使用方法

条目	指令	应答
REPWHLIST 查询	AT+REPWHLIST?	VALUE

表 6.80 AT+REPWHLIST 参数说明

参数	说明
VALUE	0(白名单为空)，终端设备的 ADDR 或 DEVEUI

使用示例：

```
AT+REPWHLIST? //查询中继器白名单
#X1111111111111111,#X11111111 //DEVEUI,ADDR
OK //设置成功
```

6. AT+REPFREQADD

该指令用来添加中继器和终端的中继信道(仅有中继器、终端中继模式、终端自动切换模式可设置)，中继器只可以设置 1 个中继信道，终端最多设置 5 个中继信道。指令格式如 表 6.81 所示，指令参数如 表 6.82 所示。

表 6.81 AT+REPFREQADD 指令使用方法

条目	指令	应答
REPFREQADD 设置	AT+REPFREQADD=FREQ	OK

表 6.82 AT+REPFREQADD 参数说明

参数	说明
FREQ	中继信道，单位 Hz，范围 470.3M~489.3M

使用示例：

```
AT+REP=1 //设置设备为中继器
OK //操作成功
AT+REPFREQADD=470300000 //设置中继器中继信道
OK //操作成功
```

```
AT+REP=0 //设置设备为终端
OK //操作成功
AT+REPMODE=1 //设置终端为一般工作模式
OK //操作成功
AT+REPFREQADD=470300000,471100000,470500000 //设置中继器中继信道
OK
```

注意

⚠ 终端为 OTAA 激活模式时最多可以设置 5 个中继信道，如果终端为 ABP 激活模式，终端只可设置一个中继信道。

7. AT+REPFREQDEL

该指令用来删除终端的中继信道 (仅 REP 禁能，终端 OTAA 激活模式能设置)。指令格式如 表 6.83 所示，指令参数如 表 6.84 所示。

表 6.83 AT+REPFREQDEL 指令使用方法

条目	指令	应答
REPFREQDEL 设置	AT+REPFREQDEL=VALUE	OK

表 6.84 AT+REPFREQDEL 参数说明

参数	说明
VALUE	1. 删除全部:ALL 2. 删除任意个: 中继信道，单位 Hz，范围 470.3M~489.3M

使用示例：

```
AT+REPFREQDEL=470300000,471100000 //选择删除中继信道
OK //操作成功
AT+REPFREQDEL=ALL //删除终端所有中继信道
OK //操作成功
AT+REPFREQ? //查询中继信道
0 //中继信道为空
OK
```

注意

- ⚠
1. 当设备为中继器或者终端 ABP 激活模式时，只有一个中继信道，因此不可以直接删除信道，但可以通过 AT+REPFREQADD 指令修改。
 2. 当终端为中继模式或者自动切换模式时不可以将中继信道全部删除，至少要留一个中继信道。
 3. 当终端为一般模式时可以将中继信道全部删除，但是再次切换回非一般模式时需要添加至少一个中继信道。

8. AT+REPFREQ

该指令用来查询中继信道。指令格式如 表 6.85 所示，指令参数如 表 6.86 所示。

表 6.85 AT+REPFREQ 指令使用方法

条目	指令	应答
REPFREQ 设置	AT+REPFREQ?	FREQ

表 6.86 AT+REPFREQ 参数说明

参数	说明
FREQ	中继信道，单位 Hz，范围 470.3M~489.3M，0(中继信道为空)

使用示例：

```
AT+REPFREQ?           //查询中继信道
470300000,471100000    //中继信道 Hz
OK
```

9. AT+REPFREQLIVE

该指令用来查询中继活跃信道，该中继活跃信道是当前终端与中继器通信使用的信道。指令格式如 表 6.87 所示，指令参数如 表 6.88 所示。

表 6.87 AT+REPFREQLIVE 指令使用方法

条目	指令	应答
REPFREQLIVE 设置	AT+REPFREQLIVE?	FREQ

表 6.88 AT+REPFREQLIVE 参数说明

参数	说明
FREQ	中继信道，单位 Hz，范围 470.3M~489.3M，0(没有中继活跃信道)

使用示例：

```
AT+REPFREQLIVE?       //查询中继活跃信道
470300000              //中继活跃信道 Hz
OK
```

注意



1. 当终端在 ABP 激活模式下，中继活跃信道即为其中继信道。
2. 如果终端在自动切换模式下成功入网通信，此时查询中继活跃信道为 0，表示终端当前使用的是一般信道通信，否则为使用中继信道通信。

6.2 AT 外设操作指令集

AT 外设操作指令集如 表 6.89 所示。

表 6.89 AT 外设操作指令集列表

AT 指令	指令名称	测试 (=?)	查询 (?)	设置 (=)	执行 (无)	AT+SAVE 存储有效
AT+GPIO	GPIO 配置			√		√
AT+PWM	PWM 配置			√		仅存储和恢复 关闭状态
AT+ADC	ADC 数据读取			√		

6.2.1 AT+GPIO

使用该指令来查询及设置 ZSL420-EVB IO。指令格式如表 6.90 所示，指令参数如表 6.91 所示，IO 配置后若保存 AT 参数则在下次上电时会自动加载当前配置，所有引脚初始配置为输入下拉。

表 6.90 AT+GPIO 指令使用方法

条目	指令	应答
IO 引脚电平查询	AT+GPIO=1,PIN	LEVEL
配置 IO 输出模式	AT+GPIO=0,PIN,0,OUT_LEVEL	OK
配置 IO 输入模式	AT+GPIO=0,PIN,1,PULL	OK

表 6.91 AT+GPIO 参数说明

参数	说明
PIN	IO 编号 0 - 5
LEVEL	IO 当前电平，若 IO 为输入则为输入电平，若 IO 为输出则为输出电平
OUT_LEVEL	输出电平，0-低电平，非 0-高电平。参数可选，不存在时默认输出低电平
PULL	上下拉模式，0-下拉；1-上拉；2-浮空。参数可选，不存在时默认下拉

使用示例：

```
AT+GPIO=0,1,0,1 //设置 IO1 为输出高电平
OK //设置成功
AT+GPIO=1,1 //查询 IO1 的电平状态
1 //高电平
OK //查询成功
```

6.2.2 AT+PWM

使用该指令来设置 ZSL420-EVB PWM 输出。需要注意的是 ZSL420-EVB 的两个通道的 PWM 频率相同，占空比不同，并且由于两个 PWM 共用同一时钟源，如果改变其中一路的输出频率，另一路的输出频率也会随之改变。两个 PWM 可以以不同的占空比独立启动输出或关闭输出。指令格式如表 6.92 所示，指令参数如表 6.93 所示，开启 PWM 后终端设备将不会进入休眠模式，终端设备复位或重启后 PWM 脚将恢复至空闲电平，空闲电平默认为输出低电平。空闲电平可配置，配置后若保存则下次自动加载。

表 6.92 AT+PWM 指令使用方法

条目	指令	应答
PWM 关闭	AT+PWM=CHAN,0,IDLE_LEVEL	OK
PWM 配置	AT+PWM=CHAN,1,FREQ,DUTY	OK

表 6.93 AT+PWM 参数说明

参数	说明
CHAN	ZSL420-EVB PWM 通道号, 0-1
IDLE_LEVEL	引脚空闲电平, 0: 低电平, 1: 高电平
FREQ	输出频率, 单位 Hz, 范围 62~800000Hz
DUTY	输出占空比, 范围 1 - 99

使用示例:

```
AT+PWM=0,1,1000,50 //设置 PWM0 输出 1KHz,50% 占空比 PWM
OK //设置成功
AT+PWM=0,0,0 //关闭 PWM0, 设置引脚为低电平
OK //关闭成功
```

6.2.3 AT+ADC

使用该指令来查询 ZSL420-EVB ADC 采样电压的 AD 值, 返回为 12 位的 AD 转换值, 需要自行转换为电压值, 终端设备 ADC 的参考电压为内部参考电压, 终端设备 ADC 精度为 12 位采样精度。指令格式如表 6.94 所示, 指令参数如表 6.95。

表 6.94 AT+ADC 指令使用方法

条目	指令	应答
ADC 采样	AT+ADC=CHAN	VOLTAGE

表 6.95 AT+ADC 参数说明

参数	说明
CHAN	ZSL420-EVB ADC 通道号, 0-1
VOLTAGE	返回的采样电压的 AD 值

使用示例:

```
AT+ADC=1 //采集 ADC1 的电压
1463 //返回电压 AD 值
OK //采集成功
```

6.3 AT 其他功能指令集

AT 其他功能指令集如表 6.96 所示。

表 6.96 AT 其他功能指令集列表

AT 指令	指令名称	测试 (=?)	查询 (?)	设置 (=)	执行 (无)	AT+SAVE 存储有效
AT	AT 测试				√	
AT+E	AT 回显			√		
AT+H	查看错误码含义				√	
AT+L	AT 指令列表				√	
AT+SLEEP	节点睡眠使能				√	
AT+DEFAULT	恢复至默认配置				√	
AT+SAVE	AT 配置保存				√	
AT+RST	节点复位				√	
AT+UPGRADE	开启固件升级		√		√	
AT+BAUD	串口波特率		√	√		√
AT+CHTIMEOUT	字符超时时间		√	√		√
AT+DEVINFO	设备版本信息		√			

6.3.1 AT

使用该指令可以测试 AT 框架工作是否正常。

使用示例：

```
AT
OK
```

6.3.2 AT+E

使用该指令来设置 AT 服务器是否回显，指令只支持设置，指令格式如 表 6.97 所示。

表 6.97 AT+E 指令使用方法

功能	指令	应答
使能回显	AT+E=1	OK
禁能回显	AT+E=0	OK

使用示例：

```
AT+E=0    //禁能指令头回显
OK        //设置成功
AT+E=1    //使能指令头回显
OK        //设置成功
```

6.3.3 AT+H

使用该指令来查看终端返回错误号的含义，指令格式如 表 6.98 所示。

表 6.98 AT+H 指令使用方法

功能	指令	应答
查看错误码含义	AT+H	OK

使用示例：

```
AT+H
ERROR NUM :
ERROR -1:  NORMAL ERROR
ERROR -2:  CMD MATCH FAILED
ERROR -3:  CHECK ARGS FORMAT FAILED
ERROR -4:  PARSE ARGS FAILED
OK
```

6.3.4 AT+L

使用 AT+L 指令来获取当前 AT 服务器支持的所有指令和指令参数格式。指令格式如表 6.99 所示。

表 6.99 AT+L 指令使用方法

功能	指令	应答
获取支持的指令列表	AT+L	OK

使用示例：

```
AT+L
Commands list :
AT
AT+E=<echo_mode>
AT+L
AT+H
AT+CLASS=<A/C>
AT+MODE=<OTAA/ABP>
AT+ID=<"DEVEUI"/"APPEUI">,<#X0000000000000000>
AT+APPKEY=<#X00000000000000000000000000000000>
AT+SKEY=<"APPSKEY"/"NWSKEY">,<#X00000000000000000000000000000000>
AT+ADDRESS=<#X00000000>
AT+JOIN
AT+ADR=<0/1>
AT+PORT=<1.223>
AT+DR=<0.Max>
AT+POWER=<Power Number>
AT+TX=<type:0/1/2>,<retry:0...>,[<string>/#X..]
AT+RX2=<1:user/0:stack>,[user DR:Code],[user frequency:Hz]
AT+RXCNT=<val>
AT+LINKCHECK
AT+DATLENGTH
AT+BUSYCHECK
AT+NETSTS
AT+MULTICASTADD=<addr:#X>,<nwks:#X>,<apps:#X>,<count(e.g.,10)>,[dr],[chan],
→[periodicity]
AT+MULTICASTDEL=<addr:#X...>
AT+MULTICASTLIST
AT+CSMA=<enable:0/1>
```

```

AT+CHMASK=<#Xxxx...>,[mode]
AT+BACKOFF=<max-backoff:ms>
AT+PERIODTX=<ON/OFF>,[period : ms],[port],[string"/#X..]
AT+AUTOACK=<enable:0/1>
AT+VOLTAGE=<VDD:mV/0-external power>,<ADC-Vref:mv>
AT+ACKTIME=<ACK_pin_width:ms>,<ACK_out_delay:ms>
AT+CONFIRMED=<0:Unconfirmed/1:confirmed>
AT+NBTRANS=<number of transmissions, 0~255>
AT+TXSTR=<string:"...">
AT+TXHEX=<hex:#Xxxx>
AT+TRANSJOIN=<number of Join-Request, 0~255, 0 : close>
AT+TRANSOUT=<message output mode, 0 ~ 2>
AT+GPIO=<0/1 set/get>,<pin>,[0/1 out/in],[out_level/in_pull]
AT+PWM=<number>,<0/1 close/open>,<idle_level/frequency>,[duty %]
AT+ADC=<number>
AT+SLEEP
AT+DEFAULT
AT+SAVE
AT+RST
AT+UPGRADE
AT+BAUD=<baud 4800/9600/14400/19200/38400/57600/115200/230400>
AT+CHTIMEOUT=<UART char timeout:ms>
AT+DEVINFO
AT+LORATEST=<ON/OFF>
AT+TEST=<reference AT+TEST=?>,[],[],[],[],[],[],[]
OK

```

6.3.5 AT+SLEEP

使用该指令允许设备进入睡眠模式，设备会根据当前状态选择低功耗模式，可能不会立刻进入低功耗，但当设备处在空闲时会进入低功耗模式。当设备处于低功耗模式时无法接收 AT 指令，只有通过唤醒引脚唤醒后才可继续接收 AT 指令。当使用 WAKE 引脚唤醒后设备不会再进入低功耗模式，即使设备处于空闲。若需要设备休眠则需要继续发送 AT+SLEEP 指令。在 Class A 类设备未使能周期发送时将进入 RAM 掉电的低功耗模式，因此 Class A 类未使能周期发送的设备在进入低功耗模式时会自动备份 AT 配置到 flash 中。由于 ClassC 设备使用场景不考虑功耗问题，因而即使发送了该指令，也不会进入睡眠模式。指令格式如表 6.100 所示。

表 6.100 AT+SLEEP 指令使用方法

功能	指令	应答
使能进入睡眠模式	AT+SLEEP	OK

使用示例：

```

AT+SLEEP //允许终端进入低功耗模式
OK      //设置成功

```

6.3.6 AT+DEFAULT

使用该指令将所有配置全部还原至默认值，但不会还原 flash 备份区域，若需要将备份区域还原，还需要调用 AT+SAVE 指令。指令格式如表 6.101 所示。

表 6.101 AT+DEFAULT 指令使用方法

功能	指令	应答
还原配置	AT+DEFAULT	OK

使用示例:

```
AT+DEFAULT    //还原配置为默认值
OK           //还原成功
```

6.3.7 AT+SAVE

使用该指令将所有配置保存至 flash 备份区域, 设备复位或重启后将会自动加载。指令格式如 表 6.102 所示。

表 6.102 AT+SAVE 指令使用方法

功能	指令	应答
参数存储	AT+SAVE	OK

使用示例:

```
AT+SAVE
OK
```

6.3.8 AT+RST

使用该指令会将设备复位, 指令会先返回执行结果再复位。指令格式如 表 6.103 所示。

表 6.103 AT+RST 指令使用方法

功能	指令	应答
设备复位	AT+RST	OK

使用示例:

```
AT+RST
OK
```

6.3.9 AT+UPGRADE

使用该指令设备将会进入固件升级模式或查询当前状态。升级时多次间隔 10ms 发送直到收到 OK 后, 才表明进入升级状态, 升级固件使用 xmode 协议传输。指令格式如 表 6.104 所示。

表 6.104 AT+UPGRADE 指令使用方法

功能	指令	应答
状态查询	AT+UPGRADE?	APP/BOOT
设备升级	AT+UPGRADE	OK

使用示例：

```
AT+UPGRADE      // 进入升级状态
OK
AT+UPGRADE?     // 查询状态
APP             // 当前处于应用状态
OK
AT+UPGRADE?     // 查询状态
BOOT           // 当前处于 Boot 状态
OK
```

6.3.10 AT+BAUD

使用该指令来设置和获取设备的串口波特率，在设置波特率时将会用原波特率返回执行结果，执行结果为执行成功时，返回后将更新串口波特率。指令格式如表 6.105 所示，指令参数如表 6.106 所示。

表 6.105 AT+BAUD 指令使用方法

功能	指令	应答
波特率获取	AT+BAUD?	BAUD
波特率设置	AT+BAUD=BAUD	OK

表 6.106 AT+BAUD 指令参数说明

参数	描述
BAUD	串口波特率，支持 4800/9600/14400/19200/38400/57600/115200/230400

使用示例：

```
AT+BAUD=115200 //设置当前波特率为 115200
OK             //设置成功
AT+BAUD?      //查询当前波特率
115200        //当前波特率为 115200
OK            //查询成功
```

6.3.11 AT+CHTIMEOUT

使用该指令来设置和获取串口接收数据时的超时时间。当接收过程中字符间隔时间超过设定时间时认定为发生超时，在透传模式下发生超时则认为一帧数据传输完成。指令格式如表 6.107 所示，指令参数如表 6.108 所示。

表 6.107 AT+CHTIMEOUT 指令使用方法

功能	指令	应答
字符超时时间获取	AT+CHTIMEOUT?	TIMEOUT
字符超时时间设置	AT+CHTIMEOUT=TIMEOUT	OK

表 6.108 AT+CHTIMEOUT 指令参数说明

参数	描述
TIMEOUT	字符超时时间，单位 ms，最小为 4ms

使用示例：

```
AT+CHTIMEOUT=10 //设置当前超时时间为 10ms
OK //设置成功
AT+CHTIMEOUT? //查询当前超时时间
10 //当前超时时间为 10ms
OK //查询成功
```

6.3.12 AT+DEVINFO

使用该指令查询设备的型号及版本号。指令格式如 表 6.109 所示。

表 6.109 AT+DEVINFO 指令使用方法

功能	指令	应答
设备版本号查询	AT+DEVINFO?	OK

使用示例：

```
AT+DEVINFO?
ZLG-ZSL420,LoRaWAN,V1.0.0,V1.3.0 //设备名称，设备类型，硬件版本号，软件版本号
OK
```

6.4 AT 测试指令集

LoRa 测试 AT 指令指令如 表 6.110 所示。

表 6.110 LoRa 测试 AT 指令列表

AT 指令	指令名称	测试 (=?)	查询 (?)	设置 (=)	执行 (无)	AT+SAVE 存储有效
AT+LORATEST	LORA 测试使能			√		
AT+TEST	测试	√		√		

6.4.1 AT+LORATEST

使用该指令来开启或关闭 LoRa 测试，在测试模式时不会进入低功耗。测试模式的参数不会保存。处于测试模式时不能操作其他指令（包括协议栈、通用指令等），若需要使用协议栈指令需要先退出测试模式。指令格式如 表 6.111 所示。

表 6.111 AT+LORATEST 指令使用方法

功能	指令	应答
进入测试模式	AT+LORATEST=ON	OK
退出测试模式	AT+LORATEST=OFF	OK

使用示例：

```
AT+LORATEST=ON           //进入测试模式
OK                         //进入成功
AT+LORATEST=OFF          //退出测试模式
OK                         //退出成功
```

6.4.2 AT+TEST

该指令必须在测试模式下使用，可使用的测试模式及参数如下所示。

1. AT+TEST=?

使用该指令可以查看 AT+TEST 指令的使用方法。使用示例如下：

```
AT+TEST=?
AT+TEST=TXCW,<freq-Hz>,<power-dB>
AT+TEST=TXLORA
AT+TEST=TXCFG,<freq-Hz>,<sf 7-12>,<bw-Hz>,<tx polarity>,<power-dB>
AT+TEST=RXCFG,<freq-Hz>,<sf 7-12>,<bw-Hz>,<rx polarity>
AT+TEST=TXPKT,<#Xxx>,<count>,<interval>
AT+TEST=TXSTR,<"string">,<count>,<interval>
AT+TEST=PUBLIC,<0/1 private/public>
AT+TEST=RXCONTINUE,[0/1 1: don't print, default 0]
AT+TEST=RSSI
AT+TEST=LWCAP
OK
```

2. AT+TEST=TXCW

使用该指令来发送 FSK 载波信号，指令格式如 表 6.112 所示，指令参数如 表 6.113 所示。

表 6.112 AT+TEST=TXCW 指令使用方法

功能	指令	应答
发送 FSK 载波信号	AT+TEST=TXCW,FREQ,POWER	OK

表 6.113 AT+TEST=TXCW 指令参数说明

参数	描述
FREQ	频率，范围 470M ~ 510M Hz
POWER	功率，范围 -2 ~ 20 dB

使用示例：

```
AT+LORATEST=ON           //进入测试模式
OK                         //进入成功
AT+TEST=TXCW,470000000,5 //发送频点为 470M FSK 载波信号，强度为 5dB
OK                         //发送成功
```

3. AT+TEST=TXLORA

使 LoRa 射频芯片进入 LoRa 测试模式，该指令没有参数，直接发送 AT+TEST=TXLORA 即可。

使用示例:

```
AT+LORATEST=ON      //进入测试模式
OK                  //进入成功
AT+TEST=TXLORA       //进入 LoRa 测试模式
OK                  //进入成功
```

4. AT+TEST=TXCFG

使用该指令来配置 LoRa 射频发送参数, 指令格式如表 6.114 所示, 指令参数如表 6.115 所示。

表 6.114 AT+TEST=TXCFG 指令使用方法

功能	指令	应答
测试模式射频配置	AT+TEST=TXCFG,FREQ,SF,BW,TX_POLARITY,POWER	OK

表 6.115 AT+TEST=TXCFG 指令参数说明

参数	描述
FREQ	射频芯片使用的中心频点, 单位 Hz, 范围 470M~510M
SF	扩频因子, 单位 sf, 范围 7-12
BW	带宽, 单位 Hz, 125000/250000/500000
TX_POLARITY	发送极性翻转, 1 使能极性翻转, 0 禁能极性翻转
POWER	发送功率, 单位 dB, 范围 -2 - 20

使用示例:

```
AT+TEST=TXCFG,470300000,7,125000,1,17 //配置射频发送参数
OK                                       //设置成功
```

5. AT+TEST=RXCFG

使用该指令来配置 LoRa 射频接收参数, 指令格式如表 6.116 所示, 指令参数如表 6.117 所示。

表 6.116 AT+TEST=RXCFG 指令使用方法

功能	指令	应答
测试模式射频配置	AT+TEST=RXCFG,FREQ,SF,BW,RX_POLARITY	OK

表 6.117 AT+TEST=RXCFG 指令参数说明

参数	描述
FREQ	射频芯片使用的中心频点, 单位 Hz, 范围 470M~510M
SF	扩频因子, 单位 sf, 范围 7-12
BW	带宽, 单位 Hz, 125000/250000/500000
RX_POLARITY	接收极性翻转, 1 使能极性翻转, 0 禁能极性翻转

使用示例：

```
AT+TEST=RXCFG,470300000,7,125000,1 //配置射频接收参数
OK //设置成功
```

6. AT+TEST=TXPKT

使用该指令来发送 HEX 数据，指令格式如 表 6.118 所示，指令参数如 表 6.119 所示。

表 6.118 AT+TEST=TXPKT 指令使用方法

功能	指令	应答
发送 HEX 数据	AT+TEST=TXPKT,DAT,COUNT,INTERVAL	OK

表 6.119 AT+TEST=TXPKT 指令参数说明

参数	描述
DAT	HEX 数据，格式 #Xxx
COUNT	发送次数，范围（1~2147483647）
INTERVAL	每两次发送之间的时间间隔，单位 ms，范围（1~2147483647）

使用示例：

```
AT+TEST=TXPKT,#X0102,2,1000 //发送数据 0x01,x002，发送两次，间隔 1000ms
OK //设置成功
+TXDONE //所有数据包发送完成
```

7. AT+TEST=TXSTR

使用该指令来发送字符串数据，指令格式如 表 6.120 所示，指令参数如 表 6.121 所示。

表 6.120 AT+TEST=TXSTR 指令使用方法

功能	指令	应答
发送字符串数据	AT+TEST=TXSTR,DAT,COUNT,INTERVAL	OK

表 6.121 AT+TEST=TXSTR 指令参数说明

参数	描述
DAT	字符串数据数据，格式“string”
COUNT	发送次数，范围（1~2147483647）
INTERVAL	每两次发送之间的时间间隔，单位 ms，范围（1~2147483647）

使用示例：

```
AT+TEST=TXSTR,"string",2,1000 //发送字符串 string，发送两次，间隔 1000ms
OK //设置成功
+TXDONE //所有数据包发送完成
```

8. AT+TEST=PUBLIC

使用该指令来设置是否使能公网，指令格式如 表 6.122 所示。

表 6.122 AT+TEST=PUBLIC 指令使用方法

功能	指令	应答
使能公网	AT+TEST=PUBLIC,1	OK
禁能公网	AT+TEST=PUBLIC,0	OK

使用示例：

```
AT+TEST=PUBLIC,1 //设置使能公网
OK //设置成功
```

9. AT+TEST=RXCONTINUE

使用该指令来开启连续接收模式（接收前需通过 AT+TEST=RXCFG 配置 LoRa 射频接收参数），指令格式如 表 6.123 所示。

表 6.123 AT+TEST=RXCONTINUE 指令使用方法

功能	指令	应答
开启连续接收, 并打印接收结果	AT+TEST=RXCONTINUE	OK
开启连续接收, 不显示结果	AT+TEST=RXCONTINUE, 1	OK

如果使能打印接收结果，接收到数据后按照如下格式输出：

表 6.124 连续接收模式数据上报格式

功能	格式
数据上报	+RSSI:VAL1,+SNR:VAL2,+LEN:VAL3,+HEX:VAL4
错误上报	ERROR

上报数据参数如 表 6.125 所示。

表 6.125 连续接收模式数据上报参数说明

参数	描述
VAL1	接收信道强度 RSSI
VAL2	接收信噪比 SNR
VAL3	接收的数据长度，字节数
VAL4	接收到的数据，按照 HEX 格式输出，#Xxxxx

使用示例：

```
AT+TEST=RXCONTINUE //开启连续接收
OK //开启成功
+RSSI:-80,+SNR:6,+LEN:4,+HEX:#X00112233 //接收到数据
```

10. AT+TEST=RSSI

使用该指令来实时获取当前空中的噪声信号强度（接收前需通过 AT+TEST=RXCFG 配置 LoRa 射频接收参数），指令格式如 表 6.126 所示。

表 6.126 AT+TEST=RSSI 指令使用方法

功能	指令	应答
获取空中噪声信号强度	AT+TEST=RSSI	RSSI

使用示例：

```
AT+TEST=RXCONTINUE,1 // 开启连续接收，不显示结果
AT+TEST=RSSI // 获取当前空中的噪声信号强度
-100 // 信号强度 (dBm)
OK // 成功
AT+TEST=RSSI // 获取当前空中的噪声信号强度
-102 // 信号强度 (dBm)
OK // 成功
```

11. AT+TEST=LWCAP

使用该指令来开启 LoRaWAN 抓包模式，指令格式如 表 6.127 所示。

表 6.127 AT+TEST=LWCAP 指令使用方法

功能	指令	应答
开启 LoRaWAN 抓包模式	AT+TEST=LWCAP	OK

接收到数据后按照如下格式输出：

表 6.128 LoRaWAN 抓包模式数据上报格式

功能	格式
Join Request	+TYPE:0,+AppEUI:#Xxx...,+DevEUIE:#Xxx...,+DevNonce:#Xxxxx
Join Accept	+TYPE:1,+HEX:#Xxx...
Unconfirmed Data Up	+TYPE: 2,+DevAddr:#Xxx..., +FCtrl:#Xxx, +FCnt:#Xxxxx, +FPort:port, +HEX:#Xxx...
Unconfirmed Data Down	+TYPE: 3,+DevAddr:#Xxx..., +FCtrl:#Xxx, +FCnt:#Xxxxx, +FPort:port, +HEX:#Xxx...
Confirmed Data Up	+TYPE: 4,+DevAddr:#Xxx..., +FCtrl:#Xxx, +FCnt:#Xxxxx, +FPort:port, +HEX:#Xxx...
Confirmed Data Down	+TYPE: 5,+DevAddr:#Xxx..., +FCtrl:#Xxx, +FCnt:#Xxxxx, +FPort:port, +HEX:#Xxx...
Proprietary	+TYPE:7,+HEX:#Xxx...

使用示例：

```
AT+TEST=RXCFG,471500000,9,125000,0 //配置射频
OK //配置成功
AT+TEST=LWCAP //开启抓包
OK //开启成功
+TYPE:0,+AppEUI:#X47297b84771aaed9,+DevEUI:#Xac5bfda245317055,+DevNonce:#X9382
AT+TEST=RXCFG,501500000,9,125000,1 //配置射频
OK //配置成功
AT+TEST=LWCAP //开启抓包
OK //开启成功
+TYPE:1,+HEX:#X6f621797cd1577ebf6cd727e3dac22e6
```

6.5 AT 服务器 URC 指示列表

AT 服务器即 LoRaWAN 终端设备会在某些时候主动向 AT 客户端发送消息，在 AT 中称为 URC 指示。主要体检在三种情况下：

- 收到 LoRaWAN 服务器的下行指示；
- 收到 LoRaWAN 服务器的 MAC 指令请求应答；
- 部分异步操作执行结果通知。

当前 AT 服务器支持的 URC 指示列表如 表 6.129 所示。

表 6.129 AT 服务器 URC 指示列表

URC 指示	描述
+TX	AT+TX 指令发起的上行请求执行结果
+RX	上报接收到的数据及相关信息
+LINKCHECK	LinkCheckReq 指令请求结果及 LinkCheckAns 指令上报
+JOIN	OTAA 模式下 AT+JOIN 指令执行结果及网络连接状态上报

6.5.1 +TX

AT 服务器使用该指令向 AT 客户端报告本次上行的执行结果和本次上行的相关信息，指令格式分为两种，格式如 表 6.130 所示，指令参数如 表 6.131 所示。

表 6.130 URC 指令 +TX 格式

条目	指令
上行请求执行错误	+TX, “ERROR” ,ERR_CODE
上行请求执行成功	+TX, “OK” ,COUNT,ACK,TIMES

表 6.131 URC 指令 +TX 参数说明

参数	说明
ERR_CODE	上报的错误值，可参考 4.7 章节
COUNT	当前上行帧的帧计数器
ACK	本次上行是否有下行应答帧
TIMES	确认帧为重发次数，非确认帧为发送次数

使用示例:

```
AT+TX=0,1,"hello!" //非确认帧发送字符串 hello! 到服务器, 发送一次
OK //请求发送成功
+TX,"OK",2,0,1 //上行成功, 帧计数器为 2, 未收到 ACK, 发送 1 次
AT+TX=1,0,"hello!" //确认帧发送字符串 hello! 到服务器, 不重发
OK //请求发送成功
+TX,"ERROR",15 //上行错误, 接收下行应答帧超时
```

6.5.2 +RX

AT 服务器使用该指令向 AT 客户端报告接收到的数据帧及数据帧的相关内容, 指令格式分为两种, 格式如 表 6.132 所示, 指令参数如 表 6.133 所示。

表 6.132 URC 指令 +RX 格式

条目	指令
接收解析失败	+RX, "ERROR",ERR_CODE
接收解析成功	+RX, "OK" ,COUNT,ADDRESS,RSSI,SNR,TYPE,PENDING,SHOULD_UP,PORT,DAT

表 6.133 URC 指令 +RX 参数说明

参数	说明
ERR_CODE	接收错误时 LoRaWAN 协议栈上报的错误值, 可参考 4.7 章节
COUNT	当前下行帧的帧计数器
ADDRESS	下行帧携带的地址信息
RSSI	LoRa 接收器信号强度
SNR	接收信噪比
TYPE	帧类型, LoRaWAN 规范定义的帧类型值。非确认下行帧 (3), 确认下行帧 (5), 私有帧 (7)
PENDING	指示下行帧 Pending 标志
SHOULD_UP	指示是否需要上行帧
PORT	数据端口号, 无数据负载时不存在此项
DAT	数据负载, 十六进制数据 (#X), 无数据负载时不存在此项

使用示例:

```
AT+TX=1,1 //请求一个上行确认帧, 未收到应答时最多重发一次
OK //请求发送成功
+TX,"OK",0,1,0 //上行成功, 帧计数器为 0, 收到 ACK, 重发 0 次
+RX,"OK",1,#X02021511,-59,7,3,0,0 //收到一个下行帧, 帧计数器为 1, 帧地址为 0X02021511,
RSSI=-59, SNR=7, 帧类型为非确认帧下行, 无数据负载 (此帧为之前上行确认帧的应答帧)
```

6.5.3 +LINKCHECK

AT 服务器使用该指令向 AT 客户端报告 LinkCheckAns 指令, 指令格式分为两种, 格式如 表 6.134 所示, 指令参数如 表 6.135 所示。

表 6.134 URC 指令 +LINKCHECK 格式

条目	指令
LinkCheckReq 请求失败	+LINKCHECK, "ERROR",ERR_CODE
LinkCheckReq 请求成功	+LINKCHECK, "OK",MARGIN,GWCNT

表 6.135 URC 指令 +LINKCHECK 参数说明

参数	说明
ERR_CODE	LinkCheckReq 请求失败时 LoRaWAN 协议栈上报的错误值，可参考 4.7 章节
MARGIN	通信链路调制裕度（0-254dB）
GWCNT	收到 LinkCheckReq 请求的网关数量

使用示例：

```

AT+LINKCHECK           //缓存 LinkCheckReq 指令
OK                     //指令缓存成功
AT+TX=1,1              //上行确认帧空帧，清空 MAC 指令缓存区
OK                     //上行帧请求成功
+TX,"OK",2,1,0         //上行帧执行成功
+LINKCHECK,"OK",16,1   //LinkCheckReq 指令请求成功，Margin=16，
Gwcnt=1
+RX,"OK",3,#X02021511,-59,9,3,0,0 //收到一个下行帧并成功解析，下行帧无数据负载

```

6.5.4 +JOIN

AT 服务器使用该指令向 AT 客户端报告网络连接状。使用 ABP 激活模式的终端不会有该 URC 指示，因为 ABP 模式初始化完即表示入网成功。指令格式如 表 6.136 所示，指令参数如 表 6.30 所示：

表 6.136 URC 指令 +JOIN 格式

条目	指令
网络连接状态报告	+JOIN,STATUS

使用示例：

```

AT+JOIN                //开启搜网
OK                     //开启搜网成功
+JOIN,"Connect"        //连接成功

```

7. 终端时序参考范例

本章节描述了终端设备使用过程中所遵循的控制时序，用户可参考如下时序，进行产品相关接口时序设计。

7.1 启动时序

终端设备每次上电需要一段的初始化时间，在这段时间内，用户不能对终端设备进行任何操作。终端设备上电时序如 图 7.1 所示：

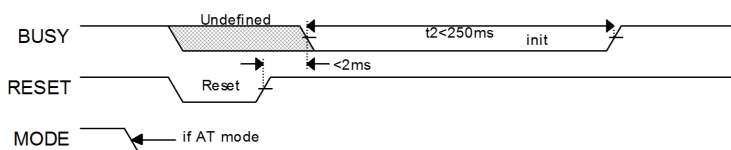


图 7.1 复位启动时序

终端设备上电或者复位后，用户需要等待一段时间 t_2 ，典型值 200ms。

用户在初始化终端设备前，需要给终端设备的 MODE 引脚提供一个电平用来指示初始化结束后的默认模式（模式可在运行中切换）。终端设备在初始化完成前，不会响应用户任何操作（比如发送串口数据或者 I/O 操作等）。

终端设备上电前后，BUSY 缺省输出低电平（终端设备忙），待终端设备完成初始化后，BUSY 信号会输出高电平，用户检测到高电平后，可以开始相关操作。

注意



注意 t_2 时间为终端设备硬件初始化时间，不包括透传模式下的自动搜网时间。

7.2 低功耗时序

终端设备可以使用 SLEEP 脚控制终端设备进入睡眠模式，为了保证 SLEEP 在任何时候被检测到，SLEEP 需要保持 100ms，其后当 BUSY 脚输出忙碌表示终端设备进入睡眠模式。在睡眠模式时 SLEEP 脚输入状态无效，时序如 图 7.2 所示：

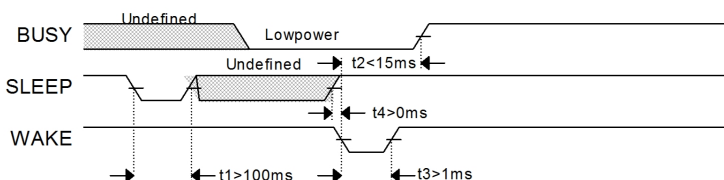


图 7.2 低功耗进入退出时序

终端设备进入睡眠状态后，用户需要通过 WAKE 引脚将其唤醒，终端设备接收到有效的唤醒信号后（需要保持至少 1ms 时间），需要一段时间 ($t_2 \leq 15ms$) 才能完全醒来。在终端设备被完全唤醒前，终端设备将不能接收来自用户的数据。在终端设备唤醒并完全恢复后 BUSY 将输出空闲，此时可正常操作终端设备。

在输入 WAKE 信号前需要将 SLEEP 信号拉高，避免唤醒后又立刻进入睡眠模式。

7.3 串口帧分割时序

在 AT 模式下串口使用“`\r\n`”作为指令的分割依据，而在透传模式下，终端设备将采用串口超时分帧机制。即终端设备在超过设定时间 t_1 内（默认 10ms 可以通过指令 `AT+CHTIMEOUT` 设置，参考 6.3.11），没有接收到任何新的串口数据，终端设备认为该帧已经结束，开始处理这个帧。时序如图 7.3 所示：

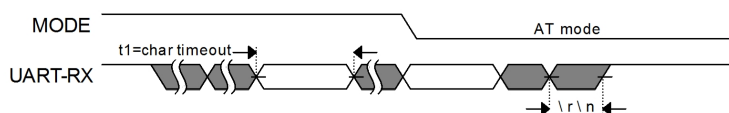


图 7.3 串口帧分割时序

7.4 AT 指令时序

在 AT 模式下，用户发出指令，终端设备收到后，就会立即解析该指令，并给出相应的响应结果。从用户发出指令到收到终端设备响应需要一定时间（ t_1 ），时序如图 7.4 所示：

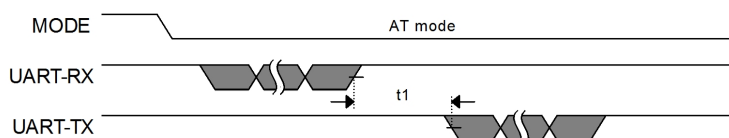


图 7.4 指令时序

命令响应的时间 t_1 和具体指令功能、指令参数有关。

7.5 串口发送时序

无论是 AT 模式还是透传模式，用户发送串口数据，需要遵循 BUSY 信号，数据发送的时序如图 7.5 所示：

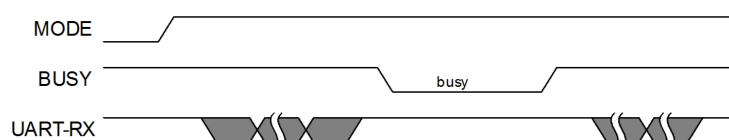


图 7.5 串口发送时序

只有在 BUSY 为高的情况，写入数据才是安全的。

7.6 数据接收时序

当终端设备在透传模式串口输出或 AT 模式接收到服务器的下行数据时，终端设备会通过 ACK 引脚通知用户。ACK 先输出一个低脉冲后再延迟一定时间后将数据发送给用户，时序如图 7.6 所示：

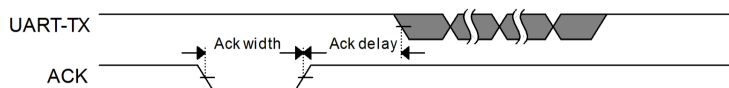


图 7.6 数据接收时序

ACK 脉冲宽度 “Ack width”、延迟时间 “Ack delay” 可以通过 AT+ACKTIME 指令配置，参考 6.1.29。默认脉冲宽度为 1ms，延迟为 0ms。

7.7 节点透传时序

图 7.7 描述了在终端设备已经成功加入网络后，透传模式下终端设备进行用户数据交互的流程，AT 模式下时序相似。

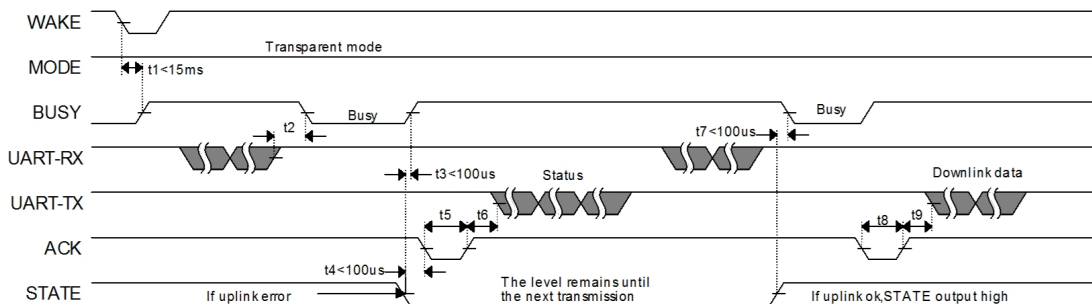


图 7.7 节点透传时序

1. 用户拉低 WAKE，唤醒终端设备。
2. MODE 引脚保持为高电平，指示处于透传模式。
3. 检测 BUSY，等待 BUSY 输出空闲。
4. 发送应用数据，最后一个字节发送完成后保持串口在字符超时时间内空闲。
5. 等待 BUSY 拉低表示开始按照 LoRaWAN 规范转发上行数据，此时 STATE 已被清除，指示正常。
6. 等待 BUSY 拉高表示传输完成。
7. 读取 STATE，STATE 输出低表示上行发生错误，STATE 保持高电平则表示上行成功。
8. 若检测到 ACK 脉冲则表示串口有数据等待输出，数据将在 ACK 延时后输出。
9. 重复 3 - 8 将待发送数据发送完成

8. 终端 AT 使用范例

8.1 终端激活

接入 LoRaWAN 服务器的终端设备无论采用哪种激活方式都需要配置信道掩码、终端类型这两项参数。终端设备在上电后需要首先配置信道掩码、终端类型，最后根据实际入网方式选择激活方式并配置该激活方式需要的鉴权信息：

- AT+CLASS
- AT+CHMASK

公共配置使用范例如下：

```
AT+CLASS=A           //设置设备为 A 类
OK                   //设置成功
AT+CHMASK=#X0000000000000000000000003FC0
OK                   //使能第 6~13 信道,
                   //使能成功
```

8.1.1 OTAA

在公共部分配置完成后对于 OTAA 模式还需要配置为 OTAA 激活并设置设备 ID 和 APPKEY，使用到的指令如下所示：

- AT+MODE
- AT+ID
- AT+KEY

使用范例如下：

```
AT+MODE=OTAA         //配置为 OTAA 激活
OK                   //配置成功
AT+ID="DEVEUI",#X1122334455667788
OK                   //配置 DEVEUI
AT+ID="APPEUI",#XAABBCCDDEEFF0011
OK                   //配置 APPEUI
AT+APPKEY=#X00112233445566778899AABBCCDDEEFF
OK                   //配置 APPKEY
OK                   //配置成功
```

- AT+JOIN

使用范例如下：

```
/*
 * 使用入网参数不保存模式
 */
AT+JOIN=0           //设置入网参数不保存模式并执行入网操作，之后可以使用 AT+JOIN 入网
OK                 //启动成功
+JOIN,"CONNECT"     //入网成功，入网参数不保存

/*
 * 使用入网参数保存模式
 */
AT+SAVE             //在使用入网参数保存模式入网前需要先保存以上设置的参数，已保存过且未
                    //改动则不需要再次保存
OK                 //保存成功
```

```

AT+JOIN=1          //设置入网参数保存模式并执行入网操作，之后使用 AT+JOIN 即可实现入网
与保存入网参数
OK                //启动成功
+JOIN,"Connect"    //入网成功并保存入网参数

```

注意



1. 入网参数默认为不保存模式。
2. 使用入网参数保存模式入网前，需要保证当前入网设备的 AT 配置参数已经更新保存到 Flash 中。这是因为设备是以当前参数入网的，如信道掩码、DEVEUI 等参数。如不保存这些参数，在设备重新上电后这些参数与入网时的使用设备参数不一致，即使已经保存了入网参数也会出现上行数据错误，如信道掩码被改变无法上行数据到网关，DEVEUI 被改变无法上行数据到当前 DEVEUI 的云端节点设备。

8.1.2 ABP

在公共部分配置完成后对于 ABP 模式还需要配置为 ABP 激活并设置设备会话密钥和设备地址，使用到的指令如下所示：

- AT+MODE
- AT+SKEY
- AT+ADDRESS

使用范例如下：

```

AT+MODE=ABP          //配置为 ABP 激活
OK                  //配置成功
AT+SKEY="NWKSKEY",#X00112233445566778899AABBCCDDEEFF //配置 NWKSKEY
OK                  //配置成功
AT+SKEY="APPSKEY",#X00112233445566778899AABBCCDDEEFF //配置 APPSKEY
OK                  //配置成功
AT+ADDRESS=#X11223344 //配置 ADDRESS
OK                  //配置成功

```

8.2 信道修改

终端设备可以使用 AT+CHMASK 指令查询、修改当前的通信信道掩码，该掩码值可能被 MAC 指令 LinkADDRReq 修改。

```

AT+CHMASK=#X00000000000000000000000000FF //设置 0-7 号信道
OK                  //设置成功

```

8.3 数据收发

终端设备入网后可以采用 AT+TX 指令进行数据发送，终端设备收到下行数据时会采用 URC 指示报告给 AT 客户端，AT 服务器提供了部分指令可以配置终端发送速率、功率，在使能 ADR 的情况下服务器会自动调整终端设备的功率与速率。

8.3.1 数据发送

- 在发送前首先检查当前协议栈是否忙碌、终端设备是否成功入网，对于空闲的已入网设备，可以发起上行数据操作。

```
AT+BUSYCHECK?           //查询终端设备是否忙碌
0                         //空闲
OK                        //查询成功
AT+NETSTS?               //查询网络状态
"CONNECT"                //已连接
OK                        //查询成功
```

- 有些应用场景可能不需要 ADR 功能或需要严格控制发送速率和功率，对于这部分应用在发送之前可以配置发送速率和功率。

```
AT+ADR=0                 //关闭 ADR 功能，在 ADR 打开时无法配置速率和功率
OK                        //关闭成功
AT+DR=3                  //配置速率为 SF9/125KHz
OK                        //配置成功
AT+POWER=0               //配置发送功率为最大值
OK                        //配置成功
```

- 配置是否开启 CSMA 功能。若禁能则发送前不会检查信道是否活跃，直接开启发送。

```
AT+CSMA=1                //使能 CSMA 功能
OK                        //设置成功
```

- 配置使用的通信端口。

```
AT+PORT=10               //设置上行帧端口为 10
OK                        //设置成功
```

- 检查当前允许的最大发送长度，若长度为 0 表示缓存的指令超过 15 字节，下一帧无法携带数据，此时应该上行一帧确认帧空帧清空当前指令缓存区。若当前待发送数据长度大于允许长度则应该将数据分包发送，不推荐使用调高速率的方式来获取较大的传输负载长度，因为终端单方面增加速率可能导致网关收不到数据。

```
AT+DATLENGTH?            //查询当前可以发送最大数据长度
0                         //当前不能携带数据
OK                        //查询成功
AT+TX=1,1                //上行一个确认帧空帧，最大重发一次
+TX,"OK",0,1,0           //发送成功，帧计数器为 0，收到应答，未重发
+RX,"OK",0,#X0203071B,-61,6,3,0,0 //接收到帧，帧计数器为 0，非确认帧，空帧
AT+DATLENGTH?            //查询当前可以发送最大数据长度
115                       //当前允许的最大上行负载为 115 字节
OK                        //查询成功
```

- 上行发送数据。

```
AT+TX=0,1,"hello!"       //发送字符串 hello!，采用非确认帧，发送一次
OK                        //启动发送成功
+TX,"OK",1,0,1           //发送成功，帧计数器为 1，没有应答，发送一次

AT+TX=1,1,#X0102          //发送十六进制 0x01 0x02，采用确认帧，最多重发一次
OK                        //启动发送成功
+TX,"OK",2,1,0           //发送成功，帧计数器为 2，收到应答，未重试
+RX,"OK",1,#X0203071B,-59,6,3,0,0 //接收到帧，帧计数器为 1，非确认帧，空帧
```

```

AT+TX=0,1          //发送空帧，采用非确认帧，发送一次
OK                 //启动发送成功
+TX,"OK",3,0,1     //发送成功，帧计数器为 3，没有应答，发送一次

```

8.3.2 数据接收

对于 Class A 设备要接收数据必须要启动上行来打开接收窗口，对于 Class C 设备可以直接接收服务器下发的数据。终端设备接收到数据后采用 URC 指示报告给 AT 客户端。终端设备开启接收不需要额外操作。

LoRaWAN 完整的发送流程为 TX->RX1->RX2，对于某些觉得一个上行周期太长或需要裁剪接收窗口的应用可以使用 AT+RXCNT 指令裁剪接收窗口。

```

AT+RXCNT=2          //RX1、RX2 正常开启
OK                 //设置成功
AT+RXCNT=1          //只开启 RX1
OK                 //设置成功
AT+RXCNT=0          //RX1、RX2 都不开启，Class A 将无法接收下行数据
OK                 //设置成功

```

8.4 MAC 指令请求

发送 MAC 指令时一般推荐采用确认帧发送，部分服务器可能不识别采用非确认帧发送 MAC 指令。在缓存需要发送的指令后立刻调用 AT+TX 发送，以 LinkCheckReq 为例：

```

AT+LINKCHECK        //缓存 LinkCheckReq 指令
OK                 //指令缓存成功
AT+TX=1,1           //上行确认帧空帧，清空 MAC 指令缓存区
OK                 //上行帧请求成功
+TX,"OK",2,1,0      //上行帧执行成功
+LINKCHECK,"OK",16,1 //LinkCheckReq 指令请求成功，Margin=16，Gwcnt=1
+RX,"OK",3,#X02021511,-59,9,3,0,0 //收到一个下行帧并成功解析，下行帧无数据负载

```

8.5 多播使用

针对多播设备主要实现了多播设备的添加（AT+MULTICASTADD）、删除（AT+MULTICASTDEL）、查询（AT+MULTICASTLIST）。只需要将对应的多播设备添加到终端，之后多播数据也会通过 +RX 指示报告给 AT 客户端，可以使用 +RX 中携带的地址信息区分该下行帧为单播还是多播设备。最大只能添加 4 个多播设备。

```

AT+MULTICASTDEL=#X063C65E9 //删除多播设备 0x063C65E9
OK                           //删除成功
AT+MULTICASTADD=#X063C65E9,#X00112233445566778899AABBCCDDEEFF,
↪#X00112233445566778899AABBCCDDEEFF,0 //添加多播设备 0x063C65E9
OK                           //添加成功
AT+MULTICASTLIST?           //查询多播设备
#X063C65E9                 //支持多播设备 0x063C65E9
OK                           //查询成功
+RX,"OK",0,#X063C65E9,-72,8,3,0,0,1,#X01 //收到多播数据

```

8.6 中继使用

1. 中继功能仅支持工作在 CLASS A 的设备，在设置中继功能之前请先将中继器与终端设置为 CLASS A 模式。
2. 中继器为 OTAA 激活模式时，需要在中继器成功入网后才能转发终端数据。
3. 在中继器需要添加 OTAA 类型终端到中继白名单功能时：中继器需要先添加 OTAA 类型终端的 DevEUI 到白名单中，终端再使用 AT+JOIN 指令通过中继入网，最后在所有终端都入网后中继器使用 AT+SAVE 保存参数。（终端入网后中继器会获取终端的 DevAddr 作为此后终端上行数据的白名单，因为在 LoRaWAN 协议规范中，终端在入网后的上下行数据中不会携带 DevEUI 而是携带由网络服务器分配的 DevAddr。所以在所有终端都通过中继器入网成功后需要使用 AT+SAVE 指令保存终端的 DevAddr 数据，否则中继器在掉电之后将无法再转发终端上行数据。）
4. 中继器与终端的 DR 设置需要一致，中继器才能正常转发终端数据。

• 中继器设置

```
AT+REP=1 //将设备设置为中继器
OK
AT+REPWHLISTADD=#X1111111111111111 //添加终端 DEVEUI 到中继器白名单
//中继器初始默认白名单为空
OK
AT+REPWHLIST? //查询中继器白名单
#X1111111111111111 //中继器白名单中的终端 DEVEUI
OK
AT+REFREQADD=471100000 //修改中继器中继信道为 471100000,
//中继器初始默认中继信道为 473700000
OK
AT+REFREQ? //查询中继器中继信道
471100000 //中继器正在使用的中继信道
OK
AT+JOIN //当中继器为 OTAA 激活模式需要申请入网
OK
+JOIN,"CONNECT" //中继器入网成功
```

• 终端设置

```
//该终端为 OTAA 激活模式，DEVEUI 为 0X1111111111111111
//将设备设置为终端
AT+REP=0
OK
AT+REPMODE=1 //设置终端使用中继模式
OK
AT+REFREQADD=471100000 //添加终端中继信道为 471100000,
//终端初始默认中继信道为 473700000
OK
AT+REFREQ? //查询终端中继信道
473700000,471100000 //终端中继信道
OK
AT+JOIN //申请入网
OK
+JOIN,"CONNECT" //终端通过中继器入网成功
AT+REFREQLIVE? //查看中继活跃信道
471100000 //当前终端中继活跃信道
OK
AT+TX=1,1,#X0102 //发送十六进制 0x01 0x02，采用确认帧，最多重发一次
OK //启动发送成功
+TX,"OK",4,1,0 //发送成功，帧计数器为 4，收到应答，未重试
```

```
+RX,"OK",4,#X00001CED,-63,11,3,0 //收到一个下行帧并成功解析，下行帧无数据负载
```

广州致远电子股份有限公司

9. 终端透传模式使用范例

透传模式的定位为转发应用数据，将串口接收到的数据通过 LoRaWAN 协议转发，将 LoRaWAN 协议栈收到的数据转发给串口发送。因此透传模式的所有工作参数均需要切换到 AT 模式下使用 AT 指令进行配置。理想的工作模式为终端长时间处于透传模式，当出现异常或需要修改配置时临时切换至 AT 模式通过 AT 指令进行相关操作或配置，完成后退回到透传模式。但终端也可以一直工作于 AT 模式，通过 AT+TX、AT+TXSTR、AT+TXHEX 发送数据。

注意



MODE 引脚拉低，才能进入 AT 模式进行配置。

9.1 数据分帧

透传模式下的分帧方式为字符超时，即发生字符超则表示当前帧传输完成。字符超时时间由 AT 指令 AT+CHTIMEOUT 配置。

在透传模式下需要特别注意每帧数据的长度，因为终端处于输出格式 0 时将不会输出错误状态。

帧长与速率的关系可参考 5 章节，以 CN470-510MHz 频段为例，其关系如表 5.3 所示。由于在透传模式下自动应答服务器的确认下行帧，因此除应用层主动发起的指令请求外，基本不存在 MAC 指令缓存，当 ADR 禁能时速率固定，允许帧长基本固定；在 ADR 使能时速率不固定，帧长可能会有变动。无论何时，当前支持的帧长都可以用 AT+DATLENGTH 指令获取。

9.2 终端配置

9.2.1 启动配置

若设备为首次启动，建议在启动前将 MODE 引脚设置为 AT 模式，方便启动后第一时间进行配置。

参数配置全部使用 AT 指令的方式进行配置。可参考 6 章节及 8 章节。配置完成后可使用 AT+SAVE 将参数存储，下一次启动后可以不再配置。

9.2.2 运行中配置

在终端设备运行中可以随时使用 MODE 引脚切换至 AT 模式，切换后可使用 AT 指令进行参数配置和查询。配置完成后可使用 AT+SAVE 将参数存储。

9.3 数据收发

若未对设备进行配置，则先对设备需要的参数进行配置，以 ZSL420-EVB 设备（ZSL42x 评估板）为例，按下按键 SW2 (MODE) 脚后，即进入 AT 模式，如图 9.1 所示进行配置。注意串口助手按照 AT 通信约定配置。

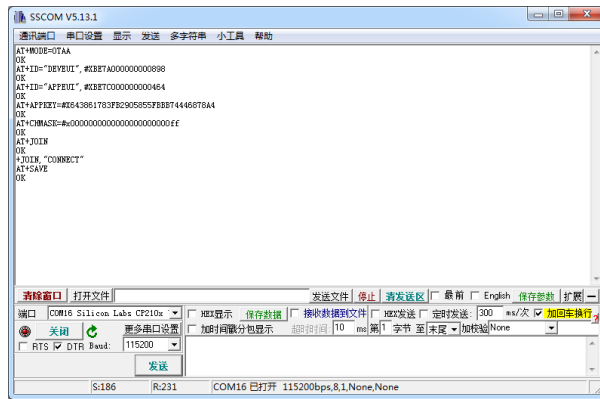


图 9.1 参数配置参考

松开按键 SW2 (MODE) 后进入透传模式，直接发送 HEX 格式数据，就可在服务器端看见串口发送的数据，如 图 9.2 所示。

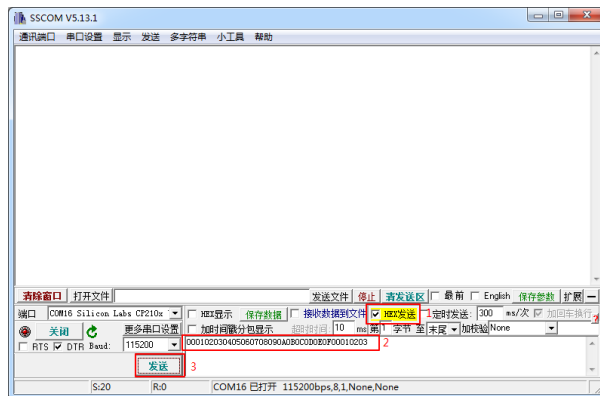


图 9.2 透传模式发送数据

服务器下发的数据也直接就通过串口发送，以输出格式 0 为例，如 图 9.3 所示，终端接收到的数据为 0x03, 0x02, 0x01。

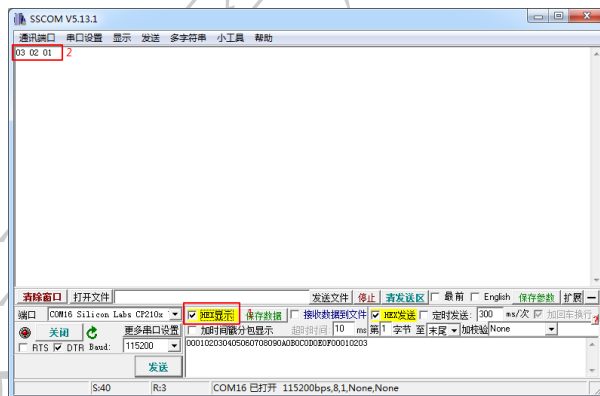


图 9.3 透传模式接收数据

10. 如何升级

10.1 串口升级

10.1.1 进入等待升级模式

终端可以通过两种方式进入等待升级状态：

- 拉低 ISP，终端设备复位后，进入 bootloader 等待升级命令，发送 AT+UPGRADE 升级命令开始等待固件接收升级；
- 在 AT 模式，发送 AT+UPGRADE，进入 bootloader 等待升级命令，发送 AT+UPGRADE 升级命令开始等待固件接收升级。

10.1.2 升级

多次尝试间隔 100ms 左右发送 AT+UPGRADE 后，等待接收到“OK”后，再通过 Xmode 协议将指定固件发送给终端升级，流程如图 10.1 所示。

注意



终端 Xmode 接收传输使用 CRC16 校验，并且支持 Xmode 128 字节及 1024 字节传输。

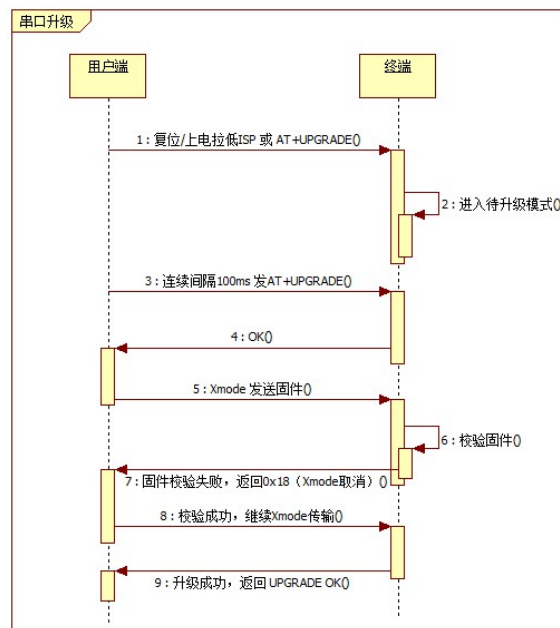


图 10.1 串口升级流程

11. 低功耗处理

用户通过拉低 SLEEP 引脚 100ms 左右, 可以使终端设备进入睡眠。终端设备进入睡眠后, 各引脚状态如表 11.1 所示。

表 11.1 终端设备睡眠的引脚状态

引脚	引脚功能	引脚方向	引脚状态	用户 MCU 引脚处理建议
RESET	复位引脚	Input	PULL-UP	拉高或浮空输入
TX	串口发送	Input	PULL-UP	浮空输入
RX	串口接收	Input	PULL-UP	浮空输入
MODE	模式切换脚	Input	PULL-UP	浮空输入
SLEEP	休眠引脚	Input	PULL-UP	拉低
WAKE	唤醒引脚	Input	PULL-UP	拉高或浮空输入
BUSY	指示终端设备忙 状态引脚	Output	LOW	浮空输入
STATE	入网、数据收发 状态指示引脚	Output	HIGH/LOW	浮空输入
ACK	终端设备接收数 据通知	Output	HIGH	浮空输入
DEF	恢复出厂设置	Input	PULL-UP	拉高或浮空输入
ISP	进入升级	Input	PULL-UP	拉高或浮空输入
IO 脚	终端设备 IO 脚	Input/ Output	PULL-UP/用户配 置	如果用户 MCU 配置了该引 脚, 以实际为准

12. 免责声明

应用信息：本应用信息适用于嵌入式产品的开发设计。客户在开发产品前，必须根据其产品特性给予修改并验证。

专业. 专注成就梦想

Dreams come true with professionalism and dedication.

广州致远电子股份有限公司

更多详情请访问
www.zlg.cn

欢迎拨打全国服务热线
400-888-4005

