

DLSx0 数传中继器用户手册

(文档版本 V1.0.0)



- UART \leftrightarrow UART/LoRA/LTE
- LoRA \leftrightarrow LoRA/UART/LTE
- LTE \leftrightarrow UART/LoRA
- 数传、中继、网关
- 数据存储器
- 传感器无线采发仪

河北稳控科技有限公司

2021年7月

目录

| | |
|-------------------------|----|
| 封面 | 1 |
| 概述 | 4 |
| 应用领域 | 4 |
| 产品选型 | 4 |
| 参数/性能指标 | 5 |
| 结构组成/安装尺寸 | 6 |
| 结构组成 | 6 |
| 安装尺寸 | 6 |
| 指示灯 | 7 |
| 接口定义 (DLS10_1U2/4) | 7 |
| 1. 开始使用 | 7 |
| 1.1 安装电池/连接电源 | 7 |
| 1.2 数字接口 | 7 |
| 1.3 查看设备基本信息 | 8 |
| 1.4 使用\$SETPTool 工具读写参数 | 8 |
| 1.5 DLSx0 工作模式 | 9 |
| 1.5.1 工作模式 | 9 |
| 1.5.2 状态机 | 9 |
| 1.5.3 前导码 | 10 |
| 1.5.4 LoRA 时间参数 | 10 |
| 1.5.5 工作模式性能对比 | 11 |
| 1.6 LoRA 频道与中心频率 | 11 |
| 1.7 读取内部存储的数据 | 11 |
| 2. 通讯协议 | 11 |
| 2.1 寄存器 (参数) 汇总 | 11 |
| 2.1.1 寄存器 (读/写) | 11 |
| 2.1.2 寄存器 (只读) | 13 |
| 2.2 参数读写协议 | 13 |
| 2.2.1 MODBUS 协议 | 13 |
| 2.2.2 AABBB 协议 | 15 |

| | |
|---------------------------------------|----|
| 2.2.3 基于设备地址的\$字符串协议..... | 15 |
| 2.2.4 其它指令..... | 16 |
| 2.3 超级指令@REST..... | 16 |
| 2.4 数据转发协议..... | 17 |
| 2.4.1 转发数据包结构..... | 17 |
| 2.4.2 数据转发规则..... | 17 |
| 2.4.3 LoRA 数据接收映射..... | 18 |
| 2.5 存储数据协议..... | 18 |
| 3. DLS 应用实例..... | 20 |
| 3.1 UART (RS232/RS485)、LoRA 互转..... | 20 |
| 3.2 LoRA 中继-使用群组码实现..... | 20 |
| 3.3 LoRA 中继-使用频道实现..... | 21 |
| 3.4 修改网络中指定设备的参数..... | 21 |
| 3.5 与其它厂商 LoRA 设备匹配..... | 21 |
| 3.6 LoRA 网关跨区域转发..... | 21 |
| 3.7 使用 DLS1x 与 VSxxx 设备的 LoRA 匹配..... | 21 |
| 4. 常见问题..... | 23 |
| 4.1 UART 通讯问题..... | 23 |
| 4.2 参数访问相关问题..... | 23 |

概述

感谢您选择我们的产品！

DLSxx 是内置电池以及 LoRA、LTE (4G) 无线的低功耗数据转发器，通过预置的转发协议可实现数据透明传输、中继接力、网关转换功能。采用休眠+无线数据唤醒技术实现了内置电池条件下长时间自主工作。内置数据存储功能，可将接收的数据存储于 Flash，以备随时读取查看。具有数据实时转发和定时启动集中转发功能。开放所有通讯参数，可与其它厂商同类产品融合。

注：本手册适用于 DLSx0_xxxx (详见“产品选型”)。



应用领域

- ◆ 自动化测控现场 LoRA-4G 工作站
- ◆ 串口设备转无线
- ◆ LoRA 中继、换频
- ◆ 不同 LoRA 设备匹配
- ◆ 无线数据记录仪
- ◆ LoRA-4G 网关

产品选型

型号标识：DLS-AB_xUXX

- DLS：产品类型标识，固定为 DLS (Double L System)，L 是指 LoRA 和 LTE。
- A：数字表示的内置 LoRA 数量，可以为 0、1、2。
- B：数字表示的内置 LTE 数量，或以为 0、1、2。
- xUXX：UART 接口定义码。例：2UT2 表示 2 个 UART 接口，分别为 TTL 和 RS232 电平。
 - ◆ xU：UART 接口的数量，可以为 1 或者 2。
 - ◆ XX：UART 接口的电平类型，2 表示 RS232，4 表示 RS485，T 表示 TTL。

型号推荐

| 型号 | 数字接口数量 | | | 传感器接口 | | | 内置电池 |
|------------|--------|------|-----|-------|----|----|-----------|
| | UART | LoRA | LTE | 振弦 | 电压 | 电流 | |
| DLS10_1U2 | 1 | 1 | 0 | | | | 1800mAH*3 |
| DLS10_1U4 | 1 | 1 | 0 | | | | 1800mAH*3 |
| DLS10_2U24 | 2 | 1 | 0 | | | | 1800mAH*3 |
| DLS20_1U2 | 1 | 2 | 0 | | | | 1800mAH*3 |
| DLS11_1U2 | 1 | | | | | | 1800mAH*3 |
| DLS11_1U4 | 1 | 1 | 1 | | | | 1800mAH*3 |
| DLS11_2U24 | 2 | 1 | 1 | | | | 1800mAH*3 |

根据型号定义规则，可选购指定任意型号，请联系我们。

内置电池是指内部可安装的电池最大容量。内置电池不是设备标配选项。

参数/性能指标

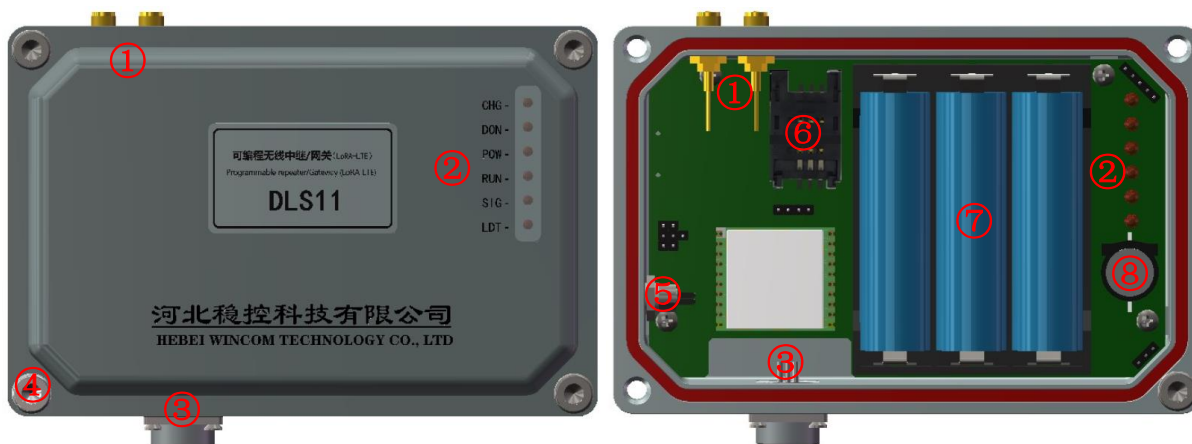
测试条件和环境：如无特殊说明，以下指标均在室温 25℃ 条件下测得。

| 项目 | 条件 | 范围 | | | 单位 |
|-------------------|--|----------|--------|--------|-----------------|
| | | 最小 | 标准 | 最大 | |
| 外形尺寸 | 148x98x43 (长 x 宽 x 厚) | | | | mm ³ |
| 防护等级 | IP65 | | | | |
| 供电方式 | 1~3 节 18650 锂电池+外部充电接口 | | | | |
| 电池容量 | 单节电池 2500mAh, 3 节 7500mAh | | | | |
| 外接电源 | | 5.5 | 12 | 24 | V |
| 功耗 (电池供电) | 待机侦听, 无线唤醒模式 | | 8 | | uA |
| | 实时接收模式 | | 17.5 | | mA |
| | LoRA 数据发送 100mW | 25 | | 125 | mA |
| | LoRA 数据发送 500mW | | | | |
| | 瞬态峰值平均值 | | 1.2 | | A |
| 工作时长 ^① | 内置电池 (3 节) 每分钟转发一次数据 | | 3 | | 年 |
| 温度 | 使用温度 | -20 | | 80 | ℃ |
| | 存储温度 | -60 | | 120 | |
| 内部存储 | 存储容量 | 4 | | | MByte |
| | 最大保存数据条数 | 16384 | | | 条 |
| LoRA | 频段 433MHz | 420 | | 450 | MHz |
| | 频段 868MHz | 854 | | 884 | MHz |
| | 频道数量 | | 15 | | |
| | 通讯速率 | 300 | 2604 | 37500 | bps |
| LTE | LTE-TDD B38/B39/B40/B41 LTE-FDD B1/B3/B8 TD-SCDMA B34/B39 UMTS/HSDPA/HSPA+ B1/B8 CDMA 1X/EVDO BC0 GSM/GPRS/EDGE900/1800 MHz | | | | |
| 传感器量程 | 振弦频率 | 300~6000 | | | Hz |
| | 温度 | -20~120 | | | ℃ |
| | 电压测量 | 0~10 | | | V |
| | 电流测量 | 0~20 | | | mA |
| UART 通讯速率 | | 1200 | 115200 | 460800 | bps |

注①：未考虑电池自放电及复杂网络时频繁唤醒造成的电量损失。

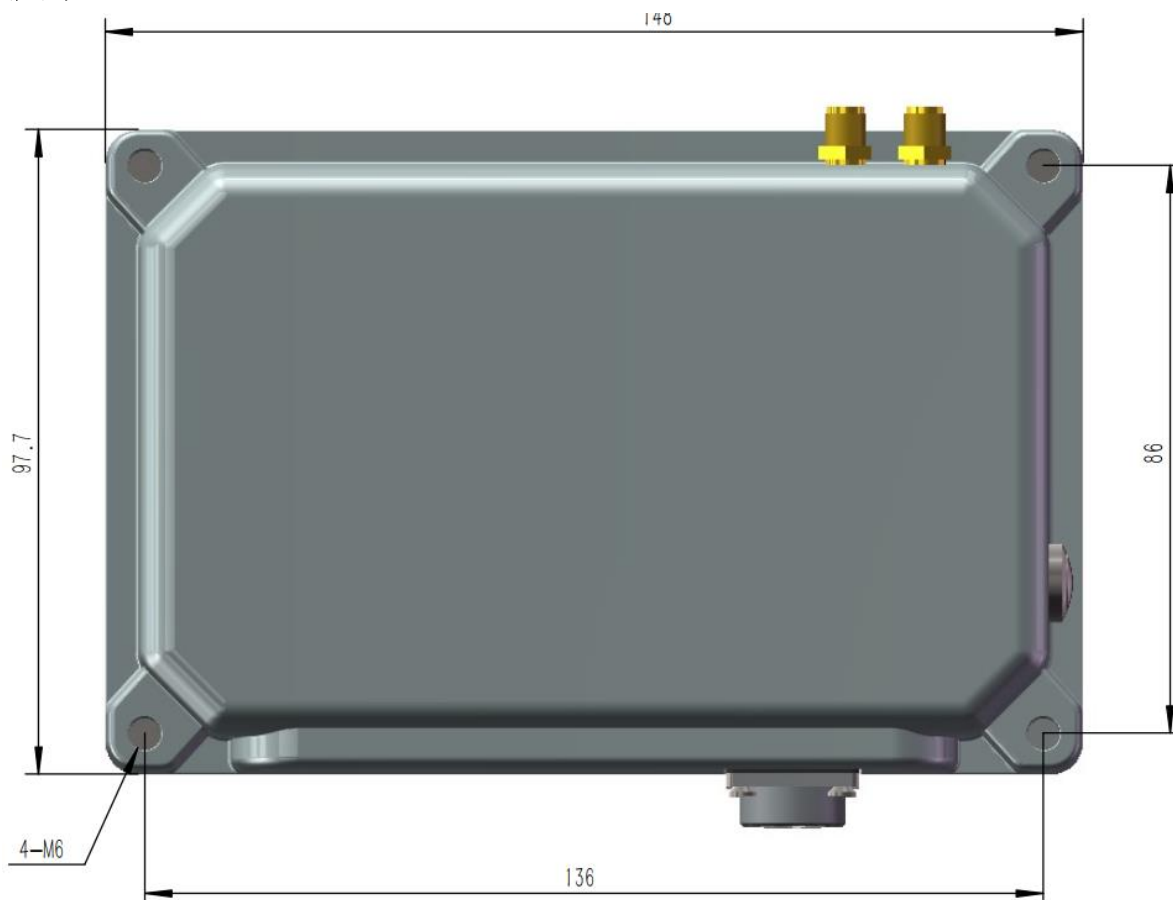
结构组成/安装尺寸

结构组成



- ①天线
- ②指示灯
- ③电源&通讯接口
- ④上盖固定螺丝
- ⑤测试按键
- ⑥SIM 卡座
- ⑦内置电池仓
- ⑧纽扣电池

安装尺寸



安装尺寸图 (底视图)

指示灯

指示灯功能说明

| 标识 | 名称 | 状态 | 描述说明 | 备注说明 |
|-----|---------|-------|---------------|---------|
| CHG | 正在充电 | 常亮 | 正在充电 | |
| DON | 充电完成 | 常亮 | 已充满 | |
| POW | 电源指示 | 常亮 | 正在工作 | 仅 DLSx1 |
| | | 熄灭 | 已休眠 | |
| RUN | 运行状态 | 闪烁 | 正在工作 | 仅 DLSx1 |
| | | 熄灭 | 已休眠 | |
| SIG | LTE 信号 | 常亮 | 正在搜索网络 | 仅 DLSx1 |
| | | 0.6Hz | 已注册网络 | |
| | | 2.5Hz | 已建立数据连接 | |
| | | 熄灭 | LTE 未工作 | |
| LDT | LoRA 数据 | 1Hz | 非休眠状态 | |
| | | 快闪 | 正在接收或者发送数据 | |
| | | 短亮 | 已休眠, 正在检测唤醒信号 | |

接口定义 (DLS10_1U2/4)

| 引脚编号 | 符号 | 名称 | 备注说明 |
|------|-------|---------------------|------------|
| 1 | VIN | 外部电源输入+ | DC5.5V~24V |
| 2 | GND | 电源- | |
| 3 | GND | 电源- | |
| 4 | TXD/A | RS232 发送/RS485 (D+) | |
| 5 | RXD/B | RS232 接收/RS485 (D-) | |

1. 开始使用

1.1 安装电池/连接电源

DLSxx 使用内置电池和(或)外部电源工作。可以仅安装 1~3 节 18650 型锂电池, 或者仅使用外部供电, 也可以内置电池和外部供电同时存在, 此时 DLSxx 的电能完全来自外部电源, 同时外部电源也为内置电池充电。

安装内置电池的方法和步骤为: (1) 打开 DLSxx 设备上盖, 将电池安装于电池座内, 注意电池正负极。

1.2 数字接口

DLSxx 的任意数字接口均可完成参数的读取与修改, 上位机(一般为计算机)使用对应电平类型的数字接口或者通过 LoRA 接口与 DLS 建立连接即可, 数字接口的接收缓存均为 200 字节。

需要注意的是: 无论使用哪种数字接口, 上位机的接口参数必须与 DLS 一致。UART 接口参数包括通讯速率、数据位、校验位、停止位, LoRA 接口参数包括射频频率(频道)、扩频因子、编码率、信道带宽。

DLS 通讯接口默认参数

| 接口 | 参数名称 | 默认值 | 单位 |
|------|------|--------|-----|
| UART | 通讯速率 | 115200 | bps |

| | | | |
|------|------|----------------|---|
| | 校验位 | N | |
| | 数据位 | 8 | 位 |
| | 停止位 | 1 | 位 |
| LoRA | 频道 | 7 (434/868MHz) | |
| | 扩频因子 | 8 | |
| | 编码率 | 2 | |
| | 信道带宽 | 7 | |

DLS1x 内部有一路 LoRA 接口和两路独立的 UART 接口。采用“时分复用”技术将一路 LoRA 接口扩展为频道 A 和频道 B (LoRA-A 和 LoRA-B)。LoRA-A 和 LoRA-B 共用参数 LA_SF、LA_CR、LA_BW 和 LA_POW。详见“寄存器 (参数) 汇总”。

DLS2x 内部有两路独立的 LoRA 接口和两路独立的 UART 接口。LoRA-A 和 LoRA-B 各自拥有专用的 LxSF、Lx_CR、Lx_BW 和 Lx_POW 参数。详见“寄存器 (参数) 汇总”。

1.3 查看设备基本信息

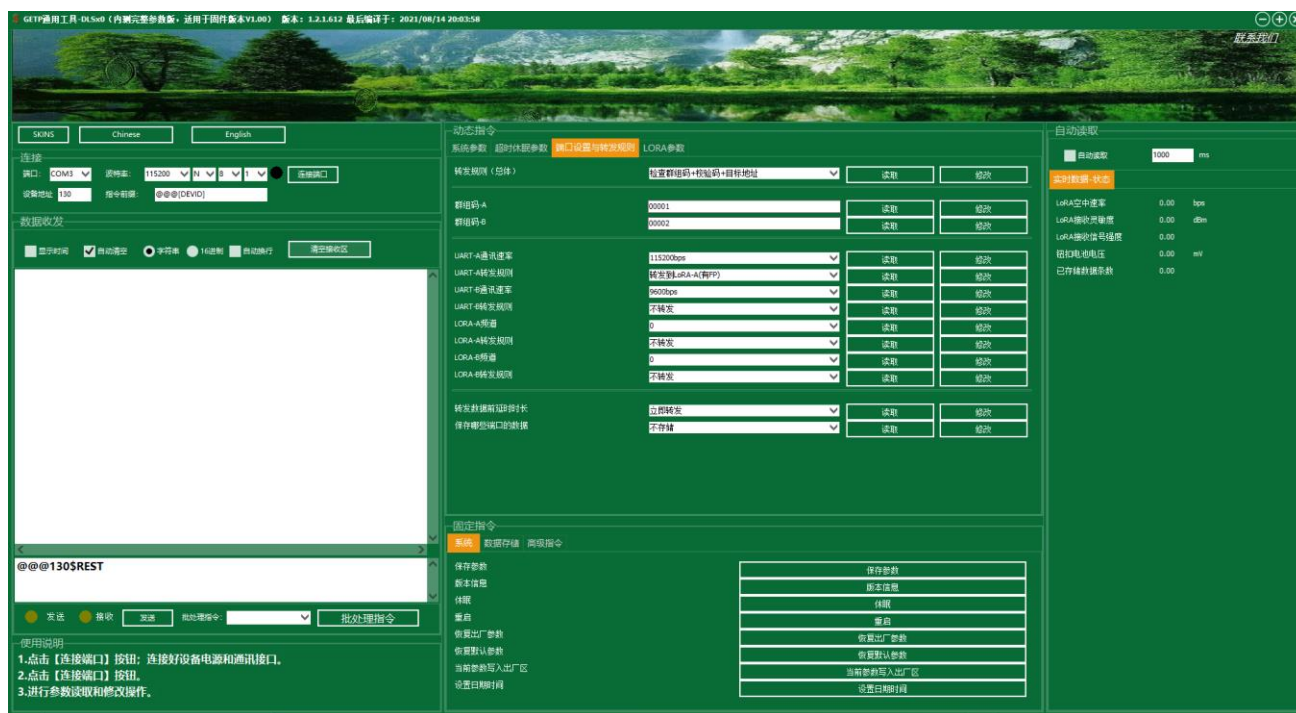
在上电启动时 DLSx0 会主动通过 UART-A 输出设备的基本信息，在设备运行过程中，向其发送 \$INFO 指令获取设备基本信息。基本信息如下所示：

| | |
|---------------------|---------------------|
| TYPE: DLS10 | 设备型号 |
| HW1.00 SF:1.00 | 硬件版本, 固件版本 |
| ADDR:129(81H) | 设备地址 |
| GAID:1 GBID:2 | 群组码 A、群组码 B |
| LACH:7 LBCH:7 | LoRA-A 频道、LoRA-B 频道 |
| BSFR: 854MHz | 射频频率 |
| SN=F628C56F0327CFE7 | 设备序列号 (唯一识别码) |
| FP=@## IP=@## | 数据转发前导符号, 字符串指令前导符号 |

1.4 使用 \$SETPTool 工具读写参数

\$SETPTool 是通用的设备测试、参数读写工具，适用于我公司绝大部分设备。如果要编写自己的测试工具，可参考“2. 通讯协议”章节说明。

\$SETPTool For DLSx0 的主界面如下图所示。



按照主界面左下角操作提示即可进行参数的读取、修改，以及设备的实时数据的自动读取。关于 \$SETPTool 更加详细的使用说明，详见“通用参数配置工具 SETPTool 使用说明.pdf”文件。

1. 5DLSx0 工作模式

1.5.1 工作模式

DLSx0 有实时接收和超时休眠两种工作模式，修改寄存器 WKMOD 为 0 表示工作于实时接收模式，为 1 表示工作于超时休眠模式。

- **实时接收模式：**DLS 设备的 LoRA-x 接口一直处于接收状态。在实时接收工作模式下，DLS1x 设备仅 LoRA-A 有效，而 DLS2x 的两路 LoRA 均有效。
- **超时休眠模式：**当无操作超过预定的时长后，设备进入空闲状态节省电能，若长时间无数据交互时进一步的进入停机状态。停机状态具有最低的电流消耗。

1.5.2 状态机

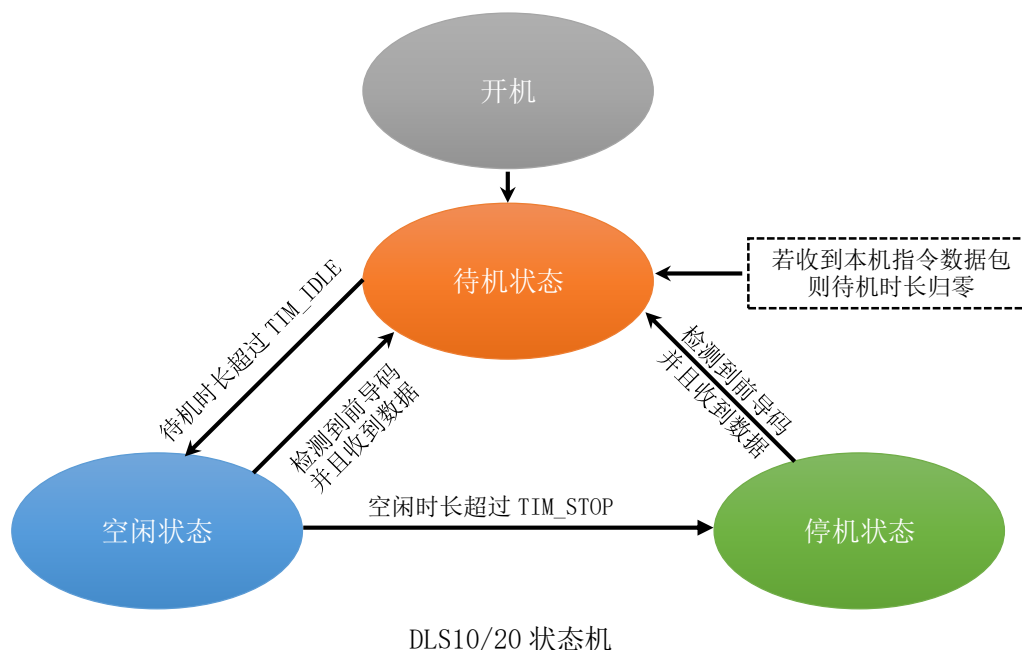
工作于超时休眠模式的设备有三种状态机（待机、空闲、停机）。

- **待机状态：**DLSx0 加电启动后的默认状态，在此状态下，DLSx0 连续的侦听 LoRA-A 和 LoRA-B 是否接收到了有效的前导码^①信号(时间间隔约 10ms)，当接收到时切换至接收模式直到数据接收完毕。
- **空闲状态：**处于待机状态的设备运行时长超过 TIM_IDLE 规定的时长后自动进入空闲状态。空闲状态具有较低的电流消耗。在此状态下，DLS 设备的 UART 正常工作，LoRA 处于休眠侦听状态，每间隔 TIM_WUT 规定的时长进行一次前导码侦听。当接收到 UART 数据或者 LoRA 前导码时退出空闲状态进入待机状态。
- **停机状态：**处于空闲状态的设备运行时长超过 TIM_STOP 规定的时长后自动进入停机状态。停机状态具有最低的电流消耗。在此状态下，DLS 设备的 UART 处于省电状态，LoRA 处于休眠侦听状态，每间隔 TIM_WUT 规定的时长进行一次前导码侦听。当接收到 UART 数据或者 LoRA 前导码时退出停机状态进入待机状态。需要注意的是：处于停机状态的设备所接收到的第一包 UART 数据会不完整，此不完整的数据包会被直接丢弃（不作任何处理）。

当处于空闲或者停机状态的设备被数字接口的数据接收事件唤醒后，对接收到的数据进行处理（转发、存储、执行指令等），处理完毕后会立即再次进入空闲状态。若接收到的数据是针对设备本身的指令，则会

执行指令并自动切换到待机状态并设置待机状态累计时长为 0（即：等待 TIM_IDLE 时长后才会再次进入空闲状态）。

①前导码：LoRA 通讯之前由发送方主动发送的一串同步信号，同步信号之后才是真正要发送的数据。



1.5.3 前导码

前导码是 LoRA 无线技术规定的必要内容，一个完整的通讯过程由发送方发起，发送方首先发送若干前导码，接收方在检测到前导码时进行时钟同步并准备开始接收数据。LoRA 协议规定前导码个数不得低于 6 个。

处于实时接收工作模式的 DLSx0 设备，可以接收任意长度前导码，故此对发送方的前导码个数没有要求（最少 6 个）。当设备工作于超时休眠工作模式时，LoRA 处于前导码侦听模式，当工作于待机状态时大约每间隔 10mS 侦听一次前导码，而处于空闲或者停机状态的 DLSx0 设备每间隔 TIM_WUT 规定的时长才会进行一次短促的前导码侦听，故此需要发送方在发送数据时先发送足够多的前导码（足够时长）。在 DLSx0 设备中，用寄存器 TIM_WUT 设置侦听的时间间隔，用 TIM_SPB 设置发送数据时发送前导码的持续时长。为了保证处于空闲或者停机状态的 DLS 设备能够侦听到前导码，发送方的 TIM_SPB 时长不得低于接收方 TIM_WUT 时长。

TIM_WUT 的默认值为 1000mS，TIM_SPB 的默认值为 2000mS。

1.5.4 LoRA 时间参数

DLSx0 设备有多个时间寄存器（参数），说明如下：

TIM_IDLE：详见“状态机”。

TIM_STOP：详见“状态机”。

TIM_WUT：详见“前导码”。

TIM_SPB：详见“前导码”。

TIM_WRD：接收超时时长。此参数用于设置接收数据的超时时长，自侦听到首个前导码后开始计时。当设置为 0 时表示自动设置为 (TIM_SPB+256 字节传输时长)*2。

TIM_FDL：数据转发延时时长。当接收到任意数字接口的数据包后，延时多长时间再进行数据转发。

1.5.5 工作模式性能对比

| 工作模式 | 实时接收 工作模式 | 超时休眠 (永不休眠) | 超时休眠 (仅空闲) | 超时休眠 (停机) |
|---|--------------|----------------|---------------|----------------------|
| 进入条件 参数设置 | WKMOD=0 | WKMOD=1 | WKMOD=1 | WKMOD=1 |
| | TIM_IDLE=xx | TIM_IDLE=0 | TIM_IDLE>0 | TIM_IDLE>0 |
| | TIM_STOP=xx | TIM_STOP=xx | TIM_STOP=0 | TIM_STOP>0 |
| 电流 | 25mA | 25mA | 13mA | 5uA |
| LoRA 通道数量 | 1 | 2 | 2 | 2 |
| 接收检测 时间间隔 | 不间断 | 20mS | TIME_WUD | TIME_WUD |
| 优点 | 时效最高 | 双通道 时效较高 | 较省电 | 极省电 |
| 缺点 | 耗电量大 仅单通道 | 耗电量大 | 时效性差 | 时效性差 接收首包 UART 无效 |
| 对发送方 前导码的要求 | 无要求 | >20mS | >TIME_WUD | >TIME_WUD |
| <p>参数“xx”表示不关心。 为了提高数据传输时效性，应根据接收方对前导码的长度要求设置“发送前导码时长”TIM_SPB 参数</p> | | | | |

1.6 LoRA 频道与中心频率

DLS 使用频道来设置不同的射频中心频率。

$$\text{中心频率 MHz} = \text{基频 MHz} + (\text{频道} * 2\text{MHz})$$

出厂时，DLS 的 LoRA 基频已设置为 420 或者 854MHz，可以使用 \$STRF=xxx 重新设置基频。

例如：\$STRF=420

注意：不得修改 LoRA 的基频（除非为了与其它厂商的 LoRA 进行匹配或者极为特殊的用途）。

1.7 读取内部存储的数据

使用专用的 \$GTDA=xxxx 可获取指定已存储的指定编号的数据。返回的数据包协议详见“2. 通讯协议”章节说明。

2. 通讯协议

2.1 寄存器（参数）汇总

2.1.1 寄存器（读/写）

寄存器汇总表（读/写）

| 寄存器 地址 | 符号 | 名称 | 取值范围 | 默认值 | 单位 | 备注 |
|-----------|----------|--------|---------|------------|----|-------|
| 0 | DEV_ID | 设备地址 | 1~255 | 0x81 (129) | | |
| 1 | GAID | 群组码 A | 1~255 | 1 | | |
| 2 | GBID | 群组码 B | 1~255 | 2 | | |
| 3 | WKMOD | 工作模式 | 0: 实时接收 | 1 | | |
| | | | 1: 超时休眠 | | | |
| 5 | TIM_IDLE | 空闲状态超时 | 0: 从不 | 10 | 秒 | 仅超时休眠 |

| | | | | | | |
|----|----------|--------------------------|-----------|--------|-------|--------------------------|
| | | | 1~65535 | | | 模式有效 |
| 6 | TIM_STOP | 休眠状态超时 | 0: 从不 | 10 | 秒 | 仅超时休眠模式有效 |
| | | | 1~65535 | | | |
| 7 | BBAT_CHG | 钮扣电池充电开关 | 0: 从不充电 | 1 | | |
| | | | 1: 电压低时充电 | | | |
| 8 | BBAT_LOW | 钮扣电池低电量电压 | 0~65535 | 2400 | mV | |
| 9 | FW_RULE | 转发规则 (总) ^① | 0~7 | 7 | | |
| 10 | UA_BAUD | UART-A 通讯速率 ^② | 12~4608 | 1152 | 百 bps | |
| 12 | UA_FWR | UART-A 转发规则 ^③ | | 0x10 | | 转发到 LoRA-A |
| 15 | UB_BAUD | UART-B 通讯速率 ^② | 12~4608 | | | |
| 17 | UB_FWR | UART-B 转发规则 ^③ | | 0x40 | | 转发到 LoRA-B |
| 21 | LORA_SF | LoRA-A 扩频因子 | 6~12 | 8 | | LoRA-A LoRA-B 共用参数 |
| 22 | LORA_CR | LoRA-A 编码率 | 1~4 | 2 | | |
| 23 | LORA_BW | LoRA-A 信道带宽 | 0~9 | 7 | | |
| 25 | LORA_POW | LoRA-A 发射功率 | 0~15 | 10 | | |
| 24 | LA_CH | LoRA-A 频道 | 0~15 | 7 | | |
| 26 | LA_FWR | LoRA-A 转发规则 ^③ | | 0x01 | | 转发到 UART-A |
| 34 | LB_CH | LoRA-B 频道 | 0~15 | 7 | | |
| 36 | LB_FWR | LoRA-B 转发规则 ^③ | | 0x04 | | 转发到 UART-B |
| 40 | TIM_WUD | 唤醒侦听时间间隔 | 0: 不侦听 | 1000 | 毫秒 | 仅超时休眠模式有效 |
| | | | 50~65535 | | | |
| 41 | TIM_WRD | 接收超时时长 | 0~65535 | 10000 | 毫秒 | |
| 42 | TIM_SPB | 发送前导码时长 | 0~65535 | 2000 | 毫秒 | |
| 43 | TIM_FDL | 数据转发延时时长 | 0~65535 | 100 | 毫秒 | |
| 44 | DAT_SAVE | 是否保存数据 ^④ | | 0x0000 | | 不存储 |
| 46 | DBG_MSG | 输出调试信息 | 0: 不输出 | 0 | | |
| | | | 1: 输出 | | | |

(1) 转发规则 (总) 寄存器

| 位 | 名称 | 说明 |
|---------|-----------------------|--------------------------|
| bit15:3 | 保留 | |
| bit2 | 转发前是否检查数据包的目标地址合法性 | 不合法的数据包不会被转发, 详见“数据转发协议” |
| bit1 | 转发前是否检查数据包的 FP 校验码合法性 | |
| bit0 | 转发前是否检查数据包的群组码合法性 | |

(2) UART-x 通讯速率寄存器

| 位 | 名称 | 参数 | 单位 |
|----------|------|---------|-------|
| bit15:14 | 校验位 | 0: 无校验 | |
| | | 1: 奇校验 | |
| | | 2: 偶校验 | |
| bit13:0 | 通讯速率 | 12~4608 | 百 bps |

(3) 端口数据转发规则寄存器 (目标端口设置)

每个数字接口均有一个 xxxx_FWR(Forward Rule) 寄存器用于设置此接口接收到数据后将数据转发到哪个(或者哪几个)数字接口。这个寄存器从低位向高位每两位代表一个目标端口, bit0/1 代表 UART-A, bit2/3 代表 UART-B, bit4/5 代表 LoRA-A, bit6/7 代表 LoRA-B。每两位中, 低位表示是否从此接口输出数据包, 高位表示输出数据时是否带有转发协议头 FP (ForwardPrefix)。

数据转发寄存器 xxxx_FWR

| bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
|-------------|------|-------------|------|-------------|------|-------------|------|
| 从 LoRA-B 输出 | | 从 LoRA-A 输出 | | 从 UART-B 输出 | | 从 UART-A 输出 | |
| 是否带 FP | 是否输出 | 是否带 FP | 是否输出 | 是否带 FP | 是否输出 | 是否带 FP | 是否输出 |

例如:

设置 UA_FWR=00010000B(0x10) 表示 UART-A 收到数据后从 LoRA-A 转发出去, 转发时不带 FP。

设置 LA_FWR=00000001B(0x01) 表示 LoRA-A 收到数据后从 UART-A 转发出去, 转发时不带 FP。

(4) 数据保存寄存器 DAT_SAVE

| 位 | 名称 | 说明 |
|---------|-------------------|-------------------|
| bit15:4 | 保留 | |
| bit3 | LoRA-B 接收到数据后是否存储 | 若数据内容是给本机的指令, 不存储 |
| bit2 | LoRA-A 接收到数据后是否存储 | |
| bit1 | UART-B 接收到数据后是否存储 | |
| bit0 | UART-A 接收到数据后是否存储 | |

2.1.2 寄存器 (只读)

寄存器汇总表 (只读)

| 寄存器地址 | 符号 | 名称 | 取值范围 | 默认值 | 单位 |
|-------|-----------|----------------|----------|-----|-----|
| 70 | RTC_YM | 日期时间-年月 | BCD 码 | | |
| 71 | RTC_DH | 日期时间-日时 | | | |
| 72 | RTC_MS | 日期时间-分秒 | | | |
| 74 | RF_BAUD | 空中速率① | 12~37500 | | bps |
| 75 | SENS | LoRA 接收灵敏度 | | | |
| 76 | RSSI | 检测到的 LoRA 信号强度 | | | |
| 79 | VCC_MV | 工作电压 | | | mV |
| 80 | BBAT_MV | 钮扣电池电压 | | | mV |
| 91 | SAVED_NUM | 已存储数据条数 | | | |

(1) 空中速率寄存器 RF_BAUD (只读)

这个寄存器存储的是 DLS 根据 LoRA 的扩频因子、编码率、信道带宽计算得到的射频传输速率。

2.2 参数读写协议

DLSx0 支持基于设备地址的 MODBUS 协议、自定义的 AABB 协议以及字符串指令集协议, 使用这些通讯协议可对寄存器 (参数) 进行访问。

2.2.1 MODBUS 协议

DLSxx 支持 MODBUS 的 03、04、06 指令码。

(1) 03 (0x03) /03 (0x04) 指令码: 读取多个连续的寄存器数据, 指令格式如下
指令数据帧结构

| 地址码 | 功能码 0x03 | 开始地址 | 寄存器数量 | CRC 校验 |
|------|----------|------|-------|--------|
| 1 字节 | 1 字节 | 2 字节 | 2 字节 | 2 字节 |

返回数据帧结构

| 地址码 | 功能码 0x03 | 数据长度 | 数据 | CRC 校验 |
|------|----------|------|------|--------|
| 1 字节 | 1 字节 | 2 字节 | n 字节 | 2 字节 |

例：读取地址为 129 的设备寄存器值，寄存器开始地址为 0，连续读取 10 个寄存器

主机发送指令：0x81 0x03 0x00 0x00 0x00 0x0A 0xDA 0x0D

从机返回应答：81 03 14 00 81 00 01 00 02 00 01 00 01 00 0A 00 0A 00 01 09 60 00 07 A1 EF (下划线为读取到的 10 个寄存器值)

读取多个连续寄存器时，单次读取不要超过 32 个寄存器，不要试图读取不存在的寄存器。

(2) 06 (0x06) 指令码：修改单个寄存器的值，指令格式如下

指令数据帧结构

| 地址码 | 功能码 0x06 | 寄存器地址 | 寄存器值 | CRC 校验 |
|------|----------|-------|------|--------|
| 1 字节 | 1 字节 | 2 字节 | 2 字节 | 2 字节 |

返回数据帧结构

| 地址码 | 功能码 0x06 | 寄存器地址 | 寄存器值 | CRC 校验 |
|------|----------|-------|------|--------|
| 1 字节 | 1 字节 | 2 字节 | 2 字节 | 2 字节 |

例：将地址为 129 的设备中的寄存器 8 的值修改为 100

主机发送指令：0x81 0x06 0x00 0x08 0x09 0x60 0x11 0xB0

从机返回应答：0x81 0x06 0x00 0x08 0x09 0x60 0x11 0xB0

(3) 校验码算法

CRC16-MODBUS 算法：

```

unsigned int crc16(unsigned char *dat, unsigned int len)
{
    unsigned int crc=0xffff;
    unsigned char i;
    while(len!=0)
    {
        crc^=*dat;
        for(i=0;i<8;i++)
        {
            if((crc&0x0001)==0)
                crc=crc>>1;
            else
            {
                crc=crc>>1;
                crc^=0xa001;
            }
        }
        len--;
        dat++;
    }
    return crc;
}

```


2.2. 2AABB 协议

读取寄存器:

| | 帧头 | 设备地址 | 寄存器地址 | 校验和 |
|--------------|-------|------|-------|------|
| 16 进制 | AA BB | 1 字节 | 1 字节 | 1 字节 |

设备响应:

| | 帧头 | 设备地址 | 寄存器地址 | 寄存器值 | 校验和 |
|--------------|-------|------|-------|------|------|
| 16 进制 | AA BB | 1 字节 | 1 字节 | 2 字节 | 1 字节 |

例如: 读取地址为 129 的 DLS 设备的寄存器 10 的值。

向设备发送(HEX): AA BB 81 0A F0

设备返回(HEX): AA BB 81 0A 04 80 74, 0x0480 是 10 进制的 1152。

修改寄存器:

| | 帧头 | 设备地址 | 寄存器地址 | 寄存器数据 | 校验和 |
|--------------|-------|------|-------------|-------|------|
| 16 进制 | AA BB | 1 字节 | 1 字节 0x80 | 2 字节 | 1 字节 |

设备响应:

| | 帧头 | 设备地址 | 寄存器地址 | 寄存器值 | 校验和 |
|--------------|-------|------|-------|------|------|
| 16 进制 | AA BB | 1 字节 | 1 字节 | 2 字节 | 1 字节 |

例如: 修改地址为 129 的 DLS 设备的寄存器 10 的值为 1152 (0x0480)。

向设备发送: AA BB 81 8A 04 80 F4

设备返回: AA BB 81 0A 04 80 74

注意: AABB 协议仅可访问地址为 0~127 的寄存器。

提示: AABB 协议中, 0xFF 为通用地址。

2.2.3 基于设备地址的\$字符串协议

(1) 读取单个寄存器

| 固定前缀 | 设备地址 | 读取指令 | 寄存器地址 | 结束符号 |
|------|------|---------|-------|-----------|
| @@@ | xxx | \$GETP= | xxx | \r\n(回车符) |

固定前缀: 固定为“@@@”。此前缀可使用专用指令进行修改, 详见“其它指令”。

设备地址: 3 个数字字符表示的设备地址, 如“129”、“001”。

读取指令: 固定为“\$GETP=”。

寄存器地址: 数字字符表示的要读取的寄存器地址。

例如: 读取设备地址为 129 的 DLSx0 寄存器 10 的值

向 DLS 发送: @@@129\$GETP=10

DLS 返回: \$REG[10]=01152

(2) 修改单个寄存器

| 固定前缀 | 设备地址 | 修改指令 | 寄存器地址 | 寄存器值 | 结束符号 |
|------|------|---------|-------|-------|-----------|
| @@@ | xxx | \$SETP= | xxx | , xxx | \r\n(回车符) |

固定前缀: 固定为“@@@”。此前缀可使用专用指令进行修改, 详见“其它指令”。

设备地址: 3 个数字字符表示的设备地址, 如“129”、“001”。

修改指令: 固定为“\$SETP=”。

寄存器地址: 数字字符表示的要读取的寄存器地址。

寄存器值: 数字字符表示的寄存器值

文档版本 V1.00 适用于固件版本 V1.00

电话: 0316-3093523 010-61591202

例如：修改设备地址为 129 的 DLSx0 寄存器 10 的值为 96。

向 DLS 发送：@@@129\$SETP=10,96

DLS 返回：OK

(3) 读取多个寄存器

| 固定前缀 | 设备地址 | 读取指令 | 起始寄存器地址 | 寄存器个数 | 结束符号 |
|------|------|---------|---------|-------|-----------|
| @@@ | xxx | \$GETP= | xxx | ,xxx | \r\n(回车符) |

例如：读取设备地址为 129 的 DLSx0 寄存器,从寄存器 0 开始,连续读取 10 个寄存器

向 DLS 发送：@@@129\$GETP=0,10

DLS 返回：\$REG[00]=00129,00001,00002,00001,00001,00005,00005,00001,03200,00000

(4) 修改多个寄存器

| 固定前缀 | 设备地址 | 修改指令 | 寄存器地址 | 寄存器值 | 结束符号 |
|------|------|---------|-------|-------------|-----------|
| @@@ | xxx | \$SETP= | xxx | ,xxx,xxx,…… | \r\n(回车符) |

例如：修改设备地址为 129 的 DLSx0 寄存器,从寄存器 0 开始,连续修改 5 个寄存器

向 DLS 发送：@@@129\$SETP=0,129,1,2,1,1

DLS 返回：OK

2.2.4 其它指令

指令格式(结构)

| 固定前缀 | 设备地址 | 指令字符串 | 结束符号 |
|------|------|-------|-----------|
| @@@ | xxx | xxxx | \r\n(回车符) |

| 指令 | 功能描述 |
|-------------|--------------------------------------|
| \$INFO | 读取设备基本信息 |
| \$REST | 重启 |
| \$SLEP | 进入休眠模式 |
| \$RSTP | 恢复为出厂参数 |
| \$STFC | 将当前参数写入到出厂区 |
| \$STDF | 恢复为默认参数 |
| \$GTDA=xxx | 读取已存储的第 xxx 条数据 |
| \$STNM=0 | 清空已存储的所有数据 |
| \$STDT=xxxx | 设置日期时间。例如：\$STDT=2015/12/21 18:37:05 |
| \$STFP=xxxx | 设置数据转发协议前缀,xxxx 固定为 4 字符 |
| \$GTFP | 读取数据转发协议前缀 |
| \$STIP=xxx | 设置字符串指令前缀,xxx 固定为 3 字符 |
| \$GTIP | 读取字符串指令前缀 |
| \$STRF=xxx | 设置 LoRA 基频频率。例：\$STRF=420,\$STRF=854 |

2.3 超级指令@REST

超级指令是不带有设备地址的字符串指令,仅当设备无法正常通讯、指令前导码不正确或者其它无法预知的错误导致设备无法正常工作时使用。向 DLS 设备发送此指令后,设备依次进行恢复参数为默认值、重写

出厂参数区、重写指令前导符号、重写数据转发前导符号的操作，此指令执行完成后会自动重新启动。

2.4 数据转发协议

数据转发是指当某个数字接口接收到数据后从哪个（或者哪些）数字接口再次发送出去。DLS 有 4 个数字接口，分别是 UART-A、UART-B、LoRA-A、LoRA-B。

2.4.1 转发数据包结构

不同的 DLSx0 之间使用特定的数据包结构来完成数据包的定向转发传输，一个完整的数据包由数据转发前缀 FP 和要转发的数据两部分构成。

数据转发前缀 FP (ForwardPrefix) 是包含了数据转发路由信息的预定义前缀，包含有发送方、接收方、群组码、数据包长度等信息。FP 一般由 DLS 设备自行生成和维护，用户无需关心。

DLS 转发数据包结构

| 数据转发前缀 FP | | | | | | | 要转发的数据 | |
|-----------|-------|------|------|--------|------|------|----------|----------|
| FP 识别码 | 群组 ID | 起始地址 | 目标地址 | 转发次数 | 转发路径 | 校验和 | 数据包长度 | XXXXXXXX |
| @#0# | 1 字节 | 1 字节 | 1 字节 | 1 字节 n | n 字节 | 1 字节 | 1 字节 (m) | m 字节 |

FP 识别码：@#0#，识别 FP 的关键信息。

群组 ID：最近一次发送数据时与设备端口对应的群组 ID 值。

起始地址：本条数据的发起设备地址值。

目标地址：本条数据最终的目的设备地址。

转发次数：本条数据已经经过了几次转发（已经 n 次）。

转发路径：长度 n 字节，每个字节依次保存了本条数据每次被转发时的设备地址值。

校验和：前面所有数据的和校验值，仅此值正确时才会处理此数据包。

数据包长度：协议前缀之后的数据包字节数。

2.4.2 数据转发规则

(1) 处于同一区域内的每台设备均有唯一的地址（同一区域不得有相同地址的设备）。合法的设备地址为 0x01~0xFE。0xFF 等效于任意地址值，0x00 等效于不存在的地址值。

(2) 每台设备均有两个 UART (UART-A/UART-B) 数字接口和两个 LoRA (LoRA-A/LoRA-B) 数字接口，各端口均有独立的通讯参数。每台设备均有两个群组 ID 码 (GroupA_ID/GroupB_ID，简称 GAID 和 GBID)。

(3) 被传输的数据包均带有协议前缀 FP (ForwardPrefix)，FP 包含有数据包的发送起始设备地址、数据包目的设备地址、数据包群组 ID、数据包转发路径记录等信息。当接收到无 FP 的数据包时会自动为其添加。添加规则为：

群组号=接收到数据的端口对应的群组 ID 号（接收数据的端口为 xxxxA 时为 GAID，接收数据的端口为 xxxxB 时为 GBID）。

起始设备地址=0xFF。 **目标设备地址**=0xFF。

注：其它设备使用 LoRA 发来的数据包仅可被 DLS10 的 LORA-A 接收到。

(4) 无论哪个数字接口，当接收到数据包后会判断 FP 中的群组 ID 是否与自身的两个群组 ID 之一相同（匹配），若不同则丢弃处理，若相同则会首先判断数据包是否是用于自身的指令，若是则执行指令然后丢弃数据包，若不是则按照转发规则对数据包进行转发。

数据包转发规则为：

每个数字接口均可指定在收到数据后将数据转发到哪个（或者哪几个）数字端口以及是否输出数据时带有 FP。转发数据前会根据数据输出的端口自动更新 FP 中的群组 ID 值。

(5) 当接收到的数据包 FP 中的目的地址与本设备地址相同时，在转发前会修改 FP 中的目标设备地址为 0x00。其它设备在接收到数据包后会判断 FP 中的目的地址是否为 0x00，是则立即丢弃数据包（即：数据包

转发终止)。

例：16 进制数据包 40 23 40 23 02 81 85 02 81 82 D3 03 31 32 33

40 23 40 23：协议前缀识别码@##

02：本包数据最后一次被发送时使用的群组 ID 为 02。

81：本包数据的发起设备地址为 0x81。

85：本包数据是发送的终点是设备 0x85。

02：本包数据已经被转发了 02 次。

81 82：本包数据第一次由设备 0x81 转发，第二次（最近一次）是被设备 0x82 转发出来的。

D3：之前所有字节的和校验。

03：本包数据实际内容为 03 个字节。

31 32 33：本包数据实际内容为 0x31 0x32 0x33。

2.4.3 LoRA 数据接收映射

当 LoRA-A 与 LoRA-B 的频道相同时，两个通道均有可能接收到 LoRA 数据包，为了维持数据转发逻辑的正确性，DLS 在接收到 LoRA 数据包后首先进行如下处理。

■ 检查数据包是否带有正确的 FP 前导符号（默认为“@##”）

➢ 有：继续检查 FP 中的群组码是否适用于本设备

◆ 是：则将数据包映射到与群组号对应的 LoRA-x。

◆ 否：维持现状（不进行映射处理）。

➢ 无：将接收到的 LoRA 数据包映射到 LoRA-A。

LoRA 接收到的数据映射的意思是：无论 LoRA 数据包是由 LoRA-A 或者 LoRA-B 接收到的，强制的将接收到的数据包按照上述规则更新到 LoRA-A 或者 LoRA-B 的接收缓存，然后进一步按照数据转发规则处理。

2.5 存储数据协议

寄存器 DAT_SAVE 用来设置数字接口收到数据后是否存储到内部 Flash 芯片。Bit0 代表 UART-A，bit1 代表 UART-B，bit2 代表 LoRA-A，bit3 代表 LoRA-B。

例如：DAT_SAVE=0x0001 表示 UART-A 收到数据后存储到 Flash 芯片。DAT_SAVE=0x000F 表示无论哪个数字接口收到的数据都存储。

存储于 Flash 芯片的数据可使用“\$GTDA=数据号,数据协议码”指令读取，数据协议码为 0 时表示输出数据时不带有数据的数据记录标识和基本信息，数据协议码为 1 时表示输出数据时带有数据的数据记录标识和基本信息。DLS 输出的数据格式如下。

已存储的数据包结构

| 数据记录标识 | 数据基本信息 | 数据内容 |
|----------|----------|----------|
| DATxxxx= | XXXXXXXX | XXXXXXXX |

| 数据基本信息结构 | | | | | | | | | |
|----------|------|------|-------|------|------|--------|------|------|-------|
| 数据包长度 | 日期时间 | 端口号 | 群组 ID | 起始地址 | 目标地址 | 转发次数 | 转发路径 | 和校验 | 数据包长度 |
| 1 字节 | 6 字节 | 1 字节 | 1 字节 | 1 字节 | 1 字节 | 1 字节 n | n 字节 | 1 字节 | 1 字节 |

数据记录标识：固定为“DAT+4 位数字字符”（8 字节），数字字符表示本条数据是内部存储的第几条数据。

数据包长度：包含此字节在内及其之后所有数据的总字节数

日期时间：BCD 码表示的年月日时分秒。

端口号：本条数据是哪个数字接口接收并存储的。0 表示 UART-A，1 表示 UART-B，2 表示 LoRA-A，3 表示 LoRA-B。

群组 ID、起始地址、目标地址、转发次数、转发路径、校验和、数据包长度、数据内容含意与 FP 相同，详见“转发数据包结构”小节。

例如：

向 DLS 发送：@@@129\$GTDA=79,1

DLS 返回 (HEX)：44 41 54 30 30 37 39 3D 14 16 01 12 01 45 17 00 01 FF FF 00 C5 06 31 32 33 34 35 36

44 41 54 30 30 37 39 3D：DAT0079=

14：本字节及之后所有数据字节数，0x14=20 字节。

16 01 12 01 45 17：本条数据的存储时间为 2016-01-12 01:45:17

00：本条数据是 UART-A 接收并存储的

01 FF FF 00 C5 06：本条数据接收时的 FP，0x06 表示实际的数据内容为 6 个字节。详见“FP（转发协议前缀）结构”。

31 32 33 34 35 36：实际的数据内容。

例如：

向 DLS 发送：@@@129\$GTDA=79

DLS 返回 (HEX)：31 32 33 34 35 36

注：读取第 0 条数据时，DLS 会返回最后存储的一条数据，例如：向设备发送“@@@129\$GTDA=0,1”，设备返回“DAT0123=……”，表示当前已经存储了 123 条数据。

3. DLS 应用实例

以下实例均以 DLS 默认参数为基础。

3.1 UART (RS232/RS485)、LoRA 互转

本实例实现两个 UART 设备之间的无线透明传输（代替 UART 之间的物理线路）。设备 I、设备 II 均为 UART 接口。



DLS (1#) 参数设置：设置 UART-A 通讯参数与设备 I 一致。

DLS (2#) 参数设置：设置 UART-B 通讯参数与设备 II 一致。

注意事项：根据需要设置是否存储数据。

3.2 LoRA 中继-使用群组码实现

本实例使用 DLS 中继接力的方式实现两个 UART 设备的远距离传输。



数据传输举例：

★设备 I 由 UART 发送：“123” (0x31 0x32 0x33)

DLS (1#) 接收到 UART-A 数据“123”，为其增加 FP，因参数 UA_FWR=0x0030，故此 UART-A 接收到的数据转发到 LoRA-A，转发时带有 FP，则经由 DLS (1#) 的 LoRA-A 发出的数据内容为：

```
40 23 40 23 01 FF FF 01 81 47 03 31 32 33
```

DLS (2#) 的 LoRA-A 接收到数据包，并从 LoRA-B 发出，发送内容为：

```
40 23 40 23 02 FF FF 02 81 81 CA 03 31 32 33
```

.....

★DLS (4#) 的 LoRA-A 接收到数据包 40 23 40 23 03 FF FF 04 81 81 81 81 CF 03 31 32 33，因为 LA_FWR=0x0001，故此数据转发到 UART-A，转发时不带 FP，DLS (4#) 的 UART-A 发送的内容为：

31 32 33，即设备 II 的 UART 接收到“123”。

有关数据转发前缀 FP 的解释说明，详见“数据转发协议”章节。

由设备 II 的 UART 发出的数据同样可以由 DLS 中转最终到达设备 I，数据传输过程与上述描述基本相同。

本实例利用了 DLS 设备的群组码匹配转发规则，当任意一台 DLS 设备通过 LoRA-x 发送数据时，其它设备均会接收到数据，但仅群组码匹配的 DLS 设备才会进行进一步的数据转发，从而实现了数据包的定向串行传输，延长了 LoRA 通讯距离。

利用群组码实现数据中转的方法可以实现延长 LoRA 通讯距离的目的，但任意一台设备进行 LoRA 发送时会同时唤醒所有 DLS 设备，会造成一定的电量损失。

3.3 LoRA 中继-使用频道实现

本实例使用 DLS 中继接力的方式实现两个 UART 设备的远距离传输。



本方案的数据转发流程与“LoRA 中继-使用群组码实现”完全相同。

本方案的优点是：在某一台 DLS 设备发送 LoRA 数据包时，仅与它频道相同的 DLS 设备会被唤醒，实现了数据中转并降低了整个中转网络的功耗。

3.4 修改网络中指定设备的参数

本实例修改地址为 131 设备的寄存器 10 的值为 1152。

实例“LoRA 中继-使用群组码实现”或者“LoRA 中继-使用频道实现”中，设置 1#~4#设备的地址分别为 129、130、131、132。

- (1) 设备 I 为计算机，计算机通过 UART 接口向 DLS (1#) 发送字符串指令“@@@131\$SETP=10,1152”。
- (2) DLS (1#) 经由 LoRA-B→DLS (2#) LoRA-A→DLS (2#) LoRA-B→DLS (3#) LoRA-A。
- (3) DLS (3#) 接收到数据后执行指令并原路返回“OK”。

因为数据内容是针对 DLS (3#) 的指令，故此不再继续转发（详见“数据转发协议”）。

3.5 与其它厂商 LoRA 设备匹配

不同 LoRA 设备之间数据传输的必要条件是通讯参数完全一致，这些参数包括扩频因子 SF、编码率 CR、信道带宽 BW 以及中心频率 FRE（通道 CH）。

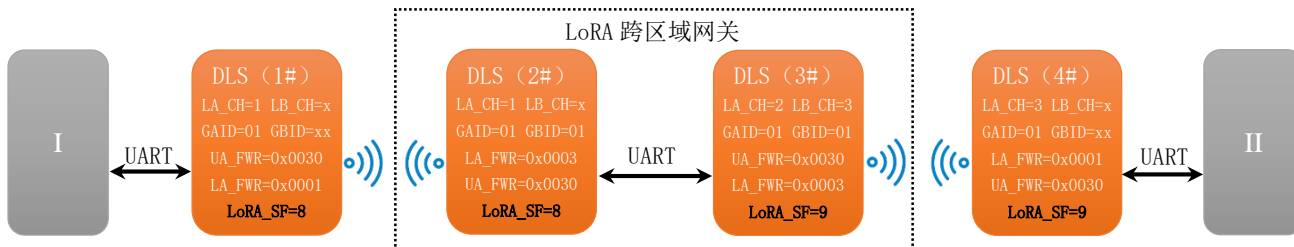
当要使用其它厂商的 LoRA 设备与 DLSx0 通讯时，必须先获取上述 4 个参数值，然后在 DLSx0 中修改对应寄存器即可。

DLS 的中心频率设置，请详见“LoRA 频道与中心频率”。

如果其它 LoRA 设备发送数据时前导码时长小于 50ms，则 DLS 必须工作于实时接收工作模式。

3.6 LoRA 网关跨区域转发

DLS10 进行数据转发时，必须使用完全相同的扩频因子 SF、编码率 CR、信道带宽 BW。使用相同 3 参数的多台设备构成的数据转发网络称为一个“区域”。本应用实例实现跨“区域”的数据转发。

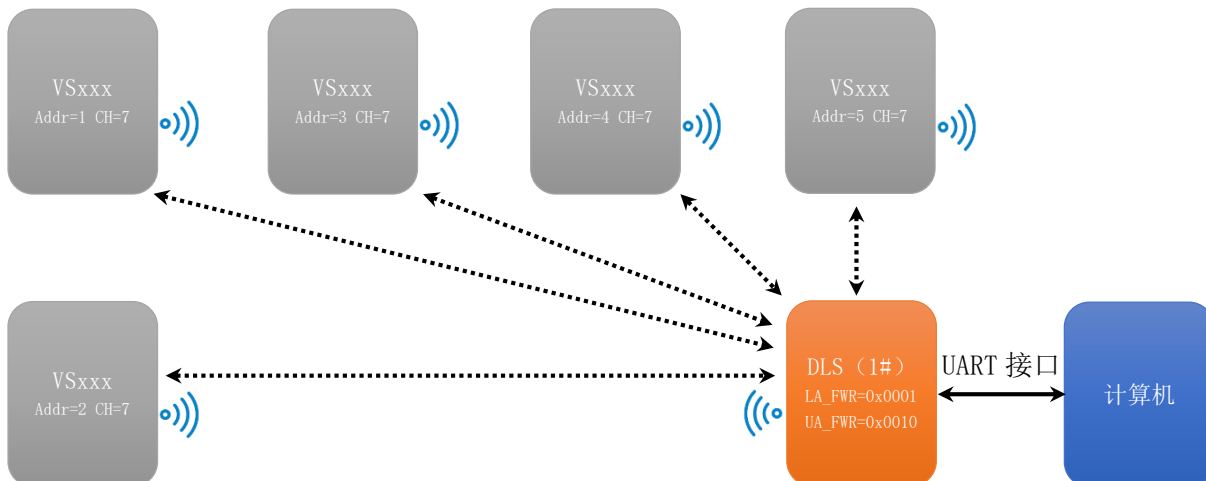


3.7 使用 DLS1x 与 VSxxx 设备的 LoRA 匹配

VSxxx 是具有 LoRA 数据发送功能的仪器，其 LoRA 默认参数与 DLS 完全相同。

VSxxx 的射频参数寄存器分别为扩频因子（283）、编码率（284）、信道带宽（285）以及频道（286），必须保证这 4 个参数与 DLS 相应参数值完全相同。另外 VSxxx 的寄存器 289 决定了 LoRA 发送数据时的前导码时长，默认值为 5 秒（此值应尽量小，但不得小于 DLS 的 TIME_WUD）。

(1) 使用 DLS10 做为 VSxxx 的 LoRA 接收器



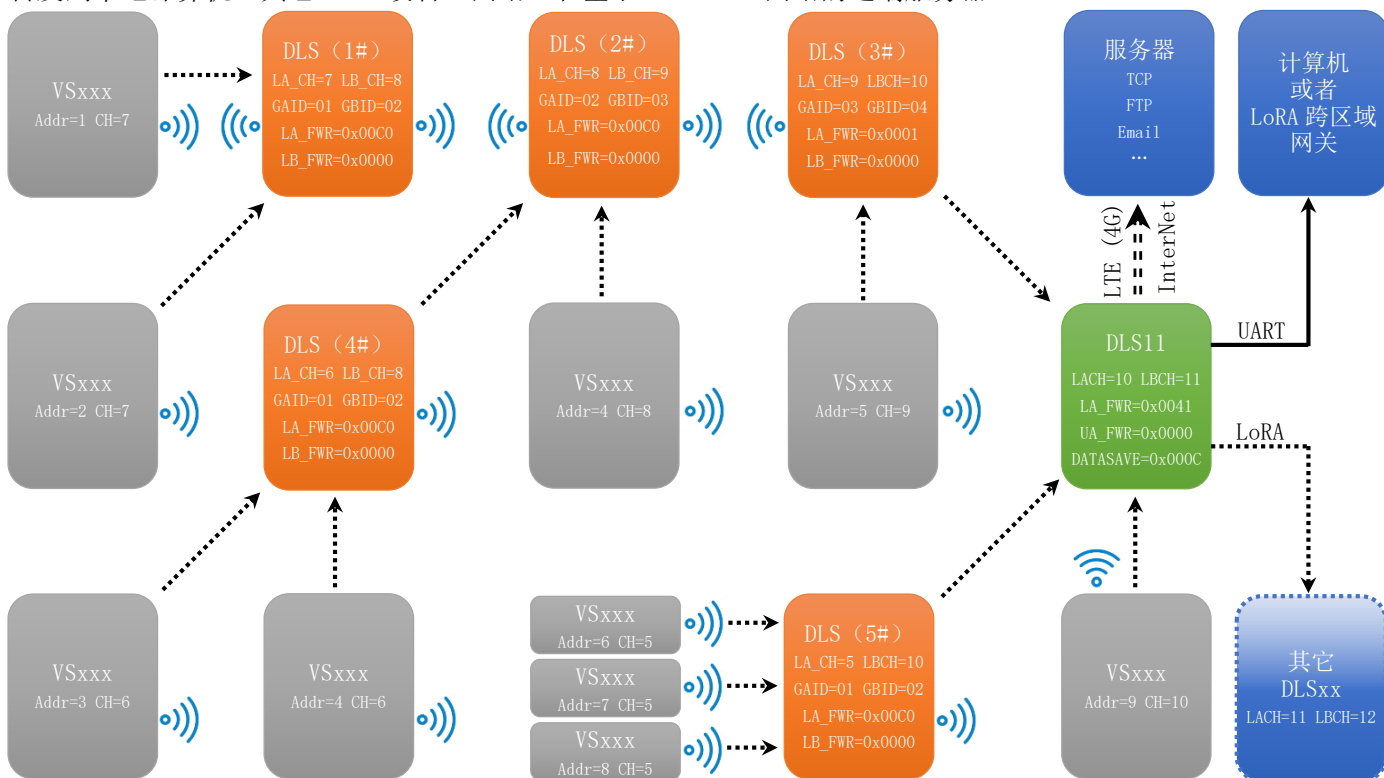
(2) 延长 VSxxx 的 LoRA 通讯距离

本应用实例使用数台 DLS10 延长 VSxxx 设备的 LoRA 通讯距离。



(3) 多台 VSxxx 设备数据汇集远传

此应用实例构建了一个单向传输数据的现场 LoRA 网络，汇集所有 VSxxx 设备的数据到 DLS11，最终将数据转发到本地计算机、其它 LoRA 设备（网络）和基于 InterNet 网络的远端服务器。



4. 常见问题

4.1 UART 通讯问题

使用 UART 接口时一定要确认收发双方的通讯参数完全一致，包括通讯速率、数据位、校验位、停止位参数。

DLS 在上电时会主动输出设备基本信息，若与之连接的上位机可以正常接收到基本信息则说明通讯参数正确，若无法收到或者接收到“乱码”则应修改上位机通讯参数，默认情况下 DLS 的通讯参数为 115200, N, 8, 1。

4.2 参数访问相关问题

对于 DLS 的参数访问必须是基于设备地址的指令 (MDOBUS、AABB、字符串)，所以首先要确认指令中的设备地址是否正确。

当使用字符串指令时，一定要确认指令的 3 个前导符号是否正确。指令前导符号可以从上电信息中获取，详见“查看设备基本信息”。

若上电信息中的指令前导符号为“乱码”或者“空”时，可使用“@REST”超级指令恢复设备为出厂状态。

4.3