PTL-D01 ManThink Technology IoT Terminal Application Layer General Protocol V1.5

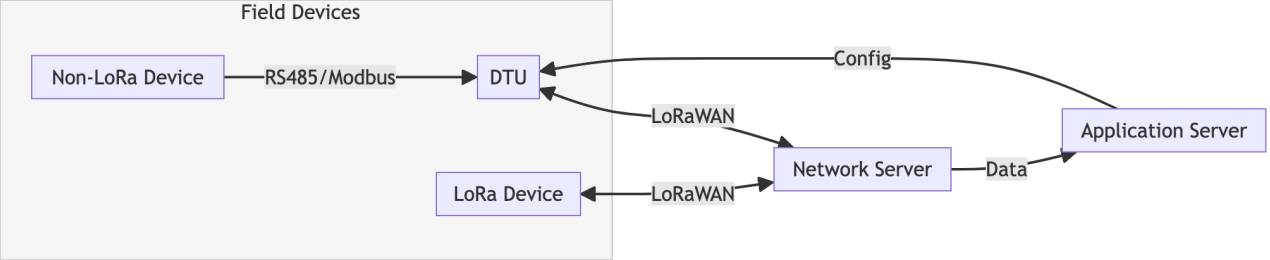
Document Overview

Purpose and Scope

This protocol is designed to establish a direct communication standard between Internet of Things (IoT) terminal devices and application servers. It covers the uplink data protocol for LoRaWAN devices and a proprietary parameter configuration protocol for Manthink Technology equipment. The design of this protocol gives full consideration to the following aspects:

- **Compatibility**: Support for the integration of multiple device models.
- **Scalability**: Reservation of space for future functional expansions.
- **Reliability**: A comprehensive mechanism to ensure data transmission integrity.

Protocol Architecture



Core Terminology Definitions

Device Types

Term	Definition
DTU	A device that collects data from terminal devices via RS-485 and forwards it through the LoRaWAN network.
Terminal Device	The end-node sensor, such as an electricity meter, water meter, or temperature and humidity sensor.
LoRaWAN Device	A device that possesses both sensing capabilities and an integrated LoRaWAN communication module (e.g., the KS52 temperature and humidity meter).

Network Elements

Term	Technical Description
Confirm Packet	A data packet that requires an acknowledgment (ACK) and is retransmitted upon failure (default of 5 retries).

Term	Technical Description
Unconfirm Packet	A data packet that does not require acknowledgment (default of 1 transmission).
Class A/C	**Class A**: Battery-powered devices that only open a receive window after an uplink transmission.
Class C: Devices with a constant power supply that continuously listen for downlink messages.	

LoRaWAN Network Specifications

Port Allocation Strategy

Port Range	Purpose	Data Type	Description
11-50	Regular sensor data	Uplink	It is recommended to assign fixed ports based on the device type.
51	Transparent mode	Bidirectional	Raw data forwarding for DTUs.
52-199	Extended ports	Bidirectional	Reserved for future use.
201	Firmware upgrades	Downlink	Reserved by the system.
214	configuration	Bidirectional	Proprietary to Manthink devices.

Advanced Network Functions

ADR Mechanism

LoRaWAN nodes are equipped with Adaptive Data Rate (ADR) functionality, which allows them to adjust their communication speed based on the actual network environment to maximize network utilization. This protocol standard requires that the local ADR function be enabled on the node side to automate rate adjustments.

Time Synchronization

- **Precision Requirement**: Device clock deviation must be less than 5 seconds per day.
- **Implementation Method**:

a. MAC layer `DeviceTimeReq` command (recommended). b. Application layer timestamp protocol (alternative).

Data Reliability Strategy

Data Type	Transmission	Retransmission	Applicable Scenario
Periodically	Unconfirm	1 transmission	Data that can tolerate loss, such as temperature and humidity readings.
Alarm events	Confirm	5 retries	Critical events such as fire alarms.

Data Type	Transmission	Retransmission	Applicable Scenario
Configuration	Confirm	3 retries + application layer confirmation	Important downlink commands, such as parameter modifications.

Uplink Data Protocol Specification

General Data Format

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Version ID (1B) | Control Word (1B) | Data Content (NB) |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Parameter Configuration (Manthink Proprietary)

This guide is applicable for modifying the parameters of Manthink Technology's LoRaWAN devices, which is accomplished by sending commands through the Network Server (NS) platform.

Prerequisites

- The device must be within the coverage area of a gateway, and the gateway must be connected to the NS platform.
- **Class A devices**: Parameters can only be sent after the device has performed an uplink.
- **Class C devices**: Parameters can be sent at any time.

Communication Specifications

Basic Parameters

Item	Description
Communication Port	214 (used for both parameter modification and replies).
Success Response	`[Function Code & 0x7F] 01 00`
Failure Response	`[Function Code & 0x5F] 01 [Failure Code]`
No Response	The device did not receive the sent parameters.

Command Format

General Command Structure

Field	Bytes	Description
Function Code	1	`0xC2` (frequency), `0xCF` (business parameters), etc.
Data size	1	2+n , size of the frame beside of Function Code and DataSize (2 bytes)
Start Address	1	Refer to the specific device's documentation.
Byte Length	1	n, size of Content to modify

Field	Bytes	Description
Content to Modify	n	Corresponds to the byte length, little-endian.

Reading Parameters

- **Request Format**: `0x8F 0x02 [Start Address] [Bytes to Read]`
- **Response Format**: `0x2F [n+2] [Parameter Address] [n] [Parameter Value]`

Example:

- **Request**: `0x8F 0x02 0x02 0x04` (Read 4 bytes starting from address 2).
- **Response**: `0x2F 0x06 0x02 0x04 A3 74 00 00`

Modifying Parameters

- **Request Format**: `0xCF [n+2] [Start Address] [n] [Parameter Value]`
- **Response Format**: `[0x6F/0x4F] 0x01 [Result Code]` (`0x6F`=Success, `0x4F`=Failure)

Example:

- **Request**: `0xCF 0x03 0x08 0x01 0x03` (Modify the value at address 8 to `0x03`).
- **Success Response**: `0x6F 0x01 0x00`
- **Failure Response**: `0x4F 0x01 0x05`

Special Commands

Frequency Modification (General)

`0xC2 0x2A [Start Address] [Byte Length] [Parameter Value]`

Frequency Parameter Table

Parameter	Address	Bytes	Format Description	Example Value (Hz)
channelMap	6	2	Bit enable	`0x00FF`
Frequency 1	8	4	Little-endian	470300000 (`0x1C083560`)
DRRange1	12	1	DR5-DR2	`0x52`
...	...	...	... (8 frequencies)	...

Complete Example:

```
C2 2A 08 28
60 35 08 1C 52
A1 42 0B 1C 52
E2 4F 0E 1C 52
23 5D 11 1C 52
60 6A 14 1C 52
A1 77 17 1C 52
E2 84 1A 1C 52
23 92 1D 1C 52
```

- **Success Response**: `0x620100`

Device Reset

`0xCF 0x03 0x09 0x01 0x01`

- **Success Response**: `0x6F0100`

Heartbeat Interval Modification

`0xCF 0x04 0x3A 0x02 [Minutes (Little-endian)]`

Example (1800 minutes = `0x0708`):
`0xCF 0x04 0x3A 0x02 0x08 0x07`

- **Success Response**: `0x6F0100`
- **Note**: A device reset is required for this to take effect.

Transparent Transmission Mode Specification

Technical Characteristics

- **Port Number**: 51 (fixed).
- **Data Format**: Raw byte stream, with no encoding or decoding.

Application Scenarios

1. Transparent data transmission for traditional RS-485 devices.
2. Compatibility mode for third-party protocols.
3. Raw data capture during debugging.

Appendix

Version Change Log

Technical Support

- 🌐 **Website**: www.manthink.cn
- ✉ **Technical Support Email**: info@manthink.cn
- ☎ **Emergency Contact**: +86-15810684257