LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard

Part 1: General

Revision 1.0.1 / 2022-09

Water Supplies Department (WSD)
Hong Kong Applied Science and Technology Research Institute (ASTRI)
The University of Hong Kong (HKU)

Document Status

Document Name	LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard
Document Code	A-1
Author	• Water Supplies Department (WSD) • Hong Kong Applied Science and Technology Research Institute (ASTRI) • The University of Hong Kong (HKU)
Revision Number	1.0.1
Document Status	Released
Contributors	Water Supplies Department: • Tin Tak CHENG • Chi Lun CHAN • Ming Kwong WONG • Samuel K. S. CHUNG Hong Kong Applied Science and Technology Research Institute: • Dr. Abel Z. YANG • Jianxiong WANG • Carlos C. H. CHIU • Henry K. H. FONG • Hang LIU • Lei ZHU • Dr. Miao MA The University of Hong Kong: • Prof. Lawrence K. YEUNG • Dr. Vincent W. L. TAM

Document Revision History

Revision	Date issued	Reviewed by	Approved by	Date approved	Revision type
1.0.0	April 2021	WSD	WSD	April 2021	Initial Release
1.0.1	September 2022	WSD	WSD	September 2022	Minor Revision

This Hong Kong Standard was approved by WSD in September 2022.

This Hong Kong Standard exists in one official version (English).

Table of Contents

Foreword

This document (LHKS001-1:2022) has been prepared by Technical Committee formed by Water Supply Department (WSD), Hong Kong Applied Science and Technology Research Institute (ASTRI), and The University of Hong Kong (HKU).

This document shall be adopted as a Hong Kong standard, either by publication of an identical text or by endorsement, at the latest by September 2022, and conflicting Hong Kong standards shall be withdrawn at the latest by September 2022.

It is possible that some of the elements of this documents may be subject to patent rights. WSD, ASTRI, and HKU shall not be held responsible for identifying any patent rights.

This document has been prepared under a mandate given to ASTRI by WSD.

LHKS001 comprises of the following parts:

- *Part 1: General*
- *Part 2: System Specification*
- *Part 3: Communication Specification*
- *Part 4: Event Specification*
- *Part 5: Security Specification*
- *Part 6: Storage Specification*
- *Part 7: Inspection Specification*

1. Introduction

This document belongs to the LHKS001 series.

This document describes the LoRaWAN-based Wireless Smart Water Metering system structure, function, and performance requirements in a generic way. This is Part 1 of LHKS001.

Additional parts to the series of the standard LHKS001 are:

- *Part 2: System Specification*
- *Part 3: Communication Specification*
- *Part 4: Event Specification*
- *Part 5: Security Specification*
- *Part 6: Storage Specification*
- *Part 7: Inspection Specification*

Since the realm of water metering is going through a period of rapid change, it is likely that this and other parts of the standard will require amendments soon.

2. Normative References

The following documents, as a whole or in part, are referenced in this document and are indispensable for its applications.

[LHKS001-2] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 - Part 2: System Specification, September 2022.

[LHKS001-3] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 - Part 3: Communication Specification, September 2022.

[LHKS001-4] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 - Part 4: Event Specification, September 2022.

[LHKS001-5] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision – Part 5: Security Specification, September 2022.

[LHKS001-6] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision - Part 6: Storage Specification, September 2022.

[LWRP103] LoRaWAN 1.0.3 Regional Parameters, Revision A, LoRa Alliance, July 2018.

[LW103] LoRaWAN Specification, Version 1.0.3, LoRa Alliance, July 2018.

[COP] Code of Practice for the Protection of Workers and Members of Public Against Non-Ionizing Radiation Hazards from Radio Transmitting Equipment.

[HKCA1078] Performance Specification for Radio Equipment Operating in the 920 – 925 MHz Band for the Provision of Public Telecommunications Services, Office of the Communications Authority (OFCA), HKSARG.

3. Terms and Conventions

3.1 Terms

Terms used in this specification are listed in TABLE 3.1.

TABLE 3.1: Terms

Term	Definition
SWM	Smart Water Meter
WA	Water Authority
LoRa™	Long Range modulation technique
LoRaWAN™	Long Range network protocol
MIU	Meter Interface Unit
DCU	Data Concentration Unit
NS	Network Server
GW	Gateway
API	Application Programming Interface
AS	Application Server

3.2 Conventions

The conventions used in this document are listed below.

SHALL - the use of the word 'SHALL indicates a mandatory requirement.
SHOULD - the use of the word 'SHOULD' indicates a requirement for good practice, which should be implemented whenever possible.
MAY - the use of the word 'may' indicates a desirable requirement.

4. LoRaWAN-based Smart Water Metering System General Description

The Smart Water Metering System allows users, including central management authorities, e.g., Water Authority (WA), to access current and historical metering data, status outputs, and alert signals, which is essential for operation management and monitoring.

4.1 System Architecture Terms

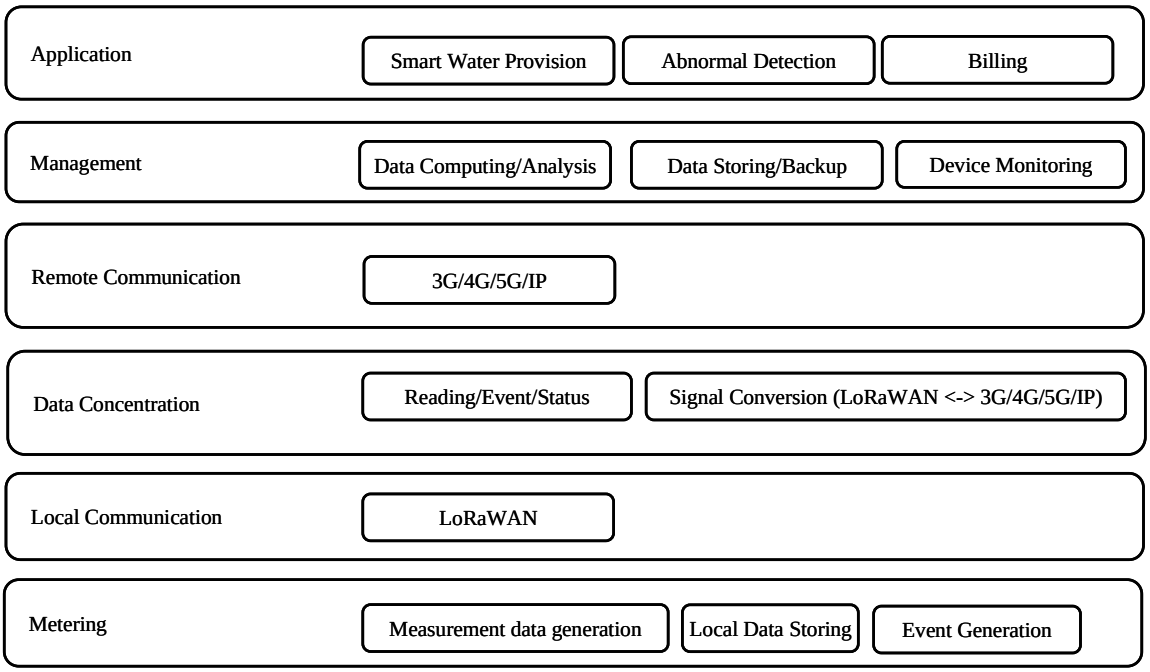

FIG. 4.1: Smart Water Metering System Architecture.

FIG. 4.1 depicts the system architecture of a LoRaWAN-based smart water metering system with its functional elements.

4.2 Functional Element

4.2.1 Metering

Metering SHALL measure water consumption, generate/store local reading records (in Meter Interface Unit (MIU)), and generate loggings and events (e.g., flow leakage, reverse flow, etc.,). The MIU SHALL fulfil the requirements of [LHKS001-2], [LHKS001-6] and [HKCA1078], and its instalment/use SHALL comply with [COP].

4.2.2 Local Communication

Local Communication SHALL provide an intermediate communication network (LoRaWAN wireless network) between a water meter and Data Concentration Unit

(DCU)), e.g., Gateway. The physical and data link layer of the LoRaWAN wireless network used SHALL fulfil the requirements of [LW103] and support the configuration defined in [LWRP103]. The communication protocol between a water meter and gateway is standardized by [LHKS001-3].

4.2.3 Data Concentration

Data Concentration SHALL support meter reading, event and status collection over local communication network, and signal conversion between local and remote communication networks. A Data Concentration Unit, i.e., Gateway, SHALL fulfil the requirements of [LW103], and its instalment/use SHALL comply with [COP].

4.2.4 Remote Communication

Remote Communication SHALL provide an intermediate communication network between Data Concentration Unit (DCU), e.g., Gateway, and the data/device management element, e.g., Network Server. A data/device management element, e.g., Network Server and other devices, SHALL fulfil the requirements of [LW103], [LHKS001-2], [LHKS001-3] and [LHKS001-5].

4.2.5 Management

Data/device Management SHALL support local/remote communication network and device management, and data storing/computing/analysis.

4.2.6 Application

Application SHALL provide Application Programming Interface (API) for different application scenarios.

5. System Topology

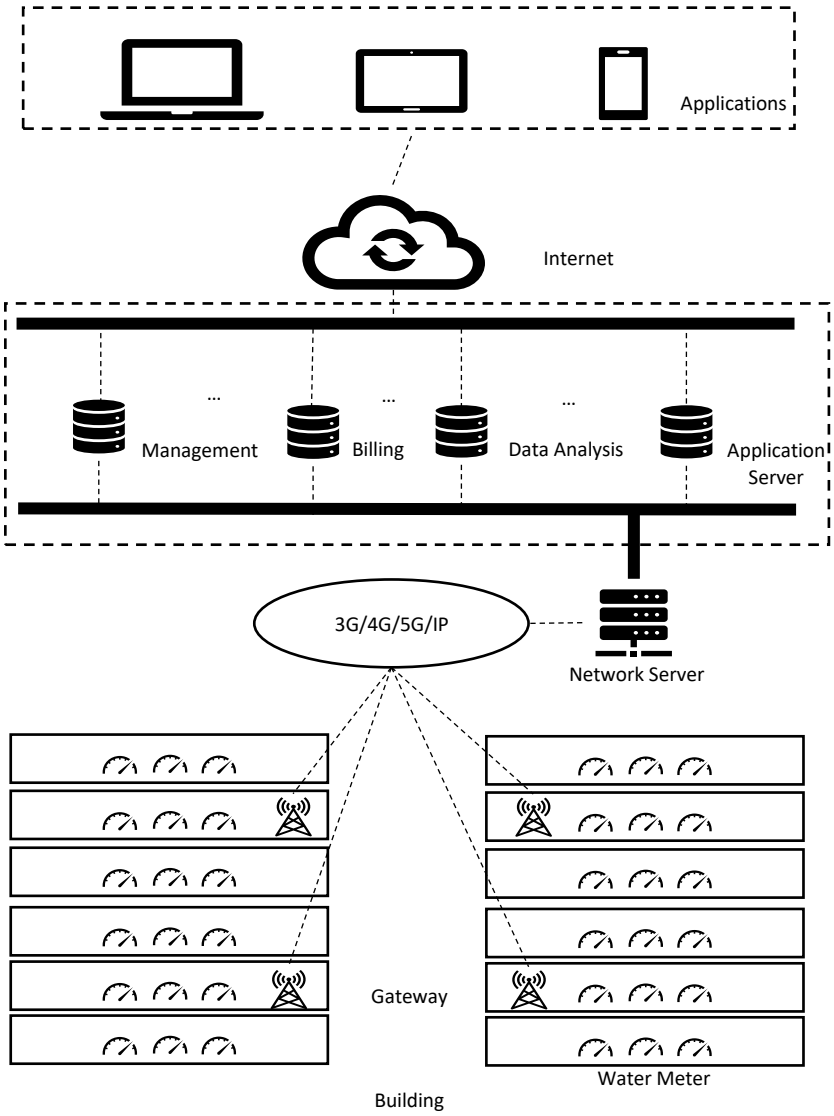

FIG. 5.1: Smart Water Metering System Topology.

FIG. 5.1 shows the system topology. The system consists of water meters, gateways, network server(s), backend servers, and applications.

A water meter is a metering device that measures water consumption measurement, generates index data, caches/logs data, and transmits (to gateway(s) over LoRaWAN uplink [LW103]).

Gateways are intermediaries that allow water meters to transmit data to network server. They receive water meters' messages over LoRaWAN uplinks and forward them to network server through high throughput 3G/4G/5G/IP connections.

Network server filters duplicate messages received from the gateways and send downlink messages to water meters through the most appropriate gateway. In addition, the network server exchanges a set of commands with the water meters for network

administration purposes, e.g., time correction, check connectivity of the water meter, change the data rate, transmit power, and channel of water meters.

Backend servers store and process metering data gathered from network server, so the data can be transferred as billing and fed to other applications.

Applications provide information to users and water authority, so they can deal with information such as water consumption, leakage in a visual format, and the water authority can make intelligent strategies regarding water usage to better conserve and supply water.

LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard

Part 2: System Specification

Revision 1.0.1 / 2022-09

Water Supplies Department (WSD)
Hong Kong Applied Science and Technology Research Institute (ASTRI)
The University of Hong Kong (HKU)

Document Status

Document Name	LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard
Document Code	A-1
Author	• Water Supplies Department (WSD) • Hong Kong Applied Science and Technology Research Institute (ASTRI) • The University of Hong Kong (HKU)
Revision Number	1.0.1
Document Status	Released
Contributors	Water Supplies Department: • Tin Tak CHENG • Chi Lun CHAN • Ming Kwong WONG • Samuel K.S. CHUNG Hong Kong Applied Science and Technology Research Institute: • Dr. Abel Z. YANG • Jianxiong WANG • Carlos C. H. CHIU • Henry K. H. FONG • Hang LIU • Lei ZHU • Dr. Miao MA The University of Hong Kong: • Prof. Lawrence K. YEUNG • Dr. Vincent W.L. TAM

Document Revision History

Revision	Date issued	Reviewed by	Approved by	Date approved	Revision type
1.0.0	April 2021	WSD	WSD	April 2021	Initial Release
1.0.1	September 2022	WSD	WSD	September 2022	Minor Revision

This Hong Kong Standard was approved by WSD in September 2022.

This Hong Kong Standard has one official version (English).

Table of Contents

Foreword

This document (LHKS001-2:2022) has been prepared by Technical Committee formed by Water Supply Department (WSD), Hong Kong Applied Science and Technology Research Institute (ASTRI), and The University of Hong Kong (HKU).

This document shall be adopted as a Hong Kong standard, either by publication of an identical text or by endorsement, at the latest by September 2022, and conflicting Hong Kong standards shall be withdrawn at the latest by September 2022.

It is possible that some of the elements of this documents may be subject to patent rights. WSD, ASTRI, and HKU shall not be held responsible for identifying any patent rights.

This document has been prepared under a mandate given to ASTRI by WSD.

LHKS001 comprises of the following parts:

- *Part 1: General*
- *Part 2: System Specification*
- *Part 3: Communication Specification*
- *Part 4: Event Specification*
- *Part 5: Security Specification*
- *Part 6: Storage Specification*
- *Part 7: Inspection Specification*

1. Introduction

This document belongs to the LHKS001 series.

This document specifies the component and functional requirements of LoRaWAN-based Smart Water Metering system. This is Part 2 of LHKS001.

Additional parts to the series of the standard LHKS001 are:

- *Part 1: General*
- *Part 3: Communication Specification*
- *Part 4: Event Specification*
- *Part 5: Security Specification*
- *Part 6: Storage Specification*
- *Part 7: Inspection Specification*

Since the realm of water metering is going through a period of rapid change, it is likely that this and other parts of the standard will require amendments soon.

2. Normative references

The following documents, as a whole or in part, are referenced in this document and are indispensable for its application.

[LHKS001-1] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1– Part 1: General, September 2022.

[LHKS001-3] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 - Part 3: Communication Specification, September 2022.

[LHKS001-4] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 - Part 4: Event Specification, September 2022.

[LHKS001-5] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 5: Security Specification, September 2022.

[LHKS001-6] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 - Part 6: Storage Specification, September 2022.

[RFC2616] Hypertext Transfer Protocol -- HTTP/1.1.

[LW103] LoRaWAN Specification, Version 1.0.3, LoRa Alliance, July 2018.

[LWRP103] LoRaWAN 1.0.3 Regional Parameters, Revision A, LoRa Alliance, July 2018.

[TLS] IETF RFC 5246, The Transport Layer Security (TLS) Protocol — Version 1.2

3. Terms and Conventions

3.1 Terms

Terms used in this specification are listed in TABLE 3.1.

TABLE 3.1: Terms

Term	Definition
SWM	Smart Water Meter
LoRa™	Long Range modulation technique
LoRaWAN™	Long Range network protocol
MIU	Meter Interface Unit
CIU	Communication Interface Unit
MAC	Medium Access Control
ABP	Activation By Personalization
GW	Gateway
NS	Network Server
DCU	Date Concentration Unit
TLS	Transport Layer Security
AS	Application Server
MQTT	MQ Telemetry Transport or Message Queuing Telemetry Transport
HTTP	Hypertext Transfer Protocol
NTP	Network Time Protocol
PLC	Programmable Logic Controller
DAS	Data Analysis Server
MS	Management Server
BS	Billing Server
EIRP	Effective Isotropic Radiated Power
OTAA	Over-the-Air Activation
RX1	Receive Window 1
RX2	Receive Window 2
RF	Radio Frequency
FCnt	Frame Count
NSK	Network Session Key
ASK	Application Session Key

3.2 Conventions

The conventions used in this document are listed below.

SHALL - the use of the word ‘SHALL indicates a mandatory requirement.
SHOULD - the use of the word ‘SHOULD’ indicates a requirement for good practice, which should be implemented whenever possible.
MAY - the use of the word ‘may’ indicates a desirable requirement.

4. System Functions

4.1 Overview

Smart Water Metering System SHALL include these components listed in TABLE 4.1 at the minimum for the system function.

TABLE 4.1: System function

SN	Function	Sub-SN	Sub-Function	Requirement
F-1	Water consumption measurement	F-1.1	Instantaneous	Generates instantaneous water consumption measurement readings.
		F-1.2	Periodical	Generates 4-hour, daily or application-based measurement readings.
F-2	Data management	F-2.1	Plausibility checking	Checks data value against bound.
		F-2.2	Local storing	Caches measurements within a specified time range.
		F-2.3	Remote storing	Stores full-time data at the backend servers' database.
		F-2.4	Local logging	Generates and maintains logging at water meters.
		F-2.5	Data analysis	Data inspection, cleansing, transforming, and modelling for discovering useful information.
F-3	Index data, status and events generation	F-3.1	Flow rate	Calculates flow rate. Flow rate is the volume of passing water per unit time.
		F-3.2	Accumulated volume	Calculate accumulated volume. Accumulated volume is the total amount of water consumption over a period.
		F-3.3	Maximum and Minimum of water consumption and index	Calculates the maximum and minimum volume, index value, e.g., flow rate, during a period.
		F-3.4	Status	Generates water meter status.
		F-3.5	Event	Detects and generates water metering events.
F-4	Data exchange	F-4.1	Data exchange over LoRaWAN	Exchanges data, i.e., water consumption, index data, status, and events, with the Network Server through gateway over LoRaWAN using specified frequency band.
F-4	Data exchange	F-4.2	Data exchange over Wide Area Network (WAN)	Exchanges data, i.e., water consumption, index data and status, with backend servers over Wireless or Wired WAN, e.g., 3G/4G/5G/IP networks.

TABLE 4.1: System function (Cont'd)

SN	Function	Sub-SN	Sub-Function	Requirement
F-5	System management	F-5.1	Device management	Monitors running statuses of devices (hardware and software). Disconnects a device when tampered or overloaded.
		F-5.2	Data backup	Copies and archives data at backend server to restore it in case of data loss.
		F-5.3	System logging	Generates and maintains information, i.e., debugs, alarms, and errors of the system.
		F-5.4	Network management	Fault analysis, performance management, networks provision, and service quality management.
		F-5.5	Billing	Determines the tariffs and prices for dedicated customer segments and generates bills at the end of each billing cycle.
		F-5.6	Remote upgrade	Updates programable parameters and firmware over LoRaWAN.
F-6	Security	F-6.1	Data security	Conducts data encryption to ensure that the data could only be accessed by authorized individuals, being reliable as well as accurate; and both availability and accessibility satisfy business requirements.
		F-6.2	System security	Sets login access restrictions to the system servers to prevent intruder and unauthorized user to access the system files and programs.
		F-6.3	Network security	Sets a firewall between internal network and the Internet to filter out unexpected intrusions. Set authentication and authorization to restrict access to specific users and approved operations.
F-7	Application	F-7.1	Consumption visualization	Provides visualized water consumption, event, status notifications.
		F-7.2	Data elaboration	Provides elaborated analysis results of defined events, e.g., fault and leakage detection.

5. System Component and Functional Requirement

5.1 System Component

The minimum set of components that a Smart Water Metering System SHALL include are listed in TABLE 5.1.

TABLE 5.1: Smart water metering system components

SN	Component	Sub-SN	Sub-component	Description
C-1	Smart water meter	C-1.1	Metering Interface Unit	Implement measurement functions, generates metering data, events, status, and stores data to data storage.
		C-1.1.1	Clock	Generates timestamp.
		C-1.1.2	Data storage	Provides space for communication buffer, historic values, configurations, events, status, and logging.
		C-1.1.3	Communication Interface Unit (CIU)	Read data from data storage and send data to LoRaWAN gateway over LoRaWAN.
		C-1.1.4	Battery	Provides power for meter.
C-2	Data Concentration Unit (DCU)	C-2.1	Gateway (GW)	Forward packets received from water meter (uplinks) to the Network Server (NS) and transmit packets sent by the Network Server.
		C-2.2	Configuration and Update Server (CUPS)	Provides configuration and software update to Gateway.
C-3	Backend servers	C-3.1	Network Server (NS)	Provides the management of gateways and endpoints, authentication and authorization of endpoints, network encryption and decryption, data routing, adapting data rates, eliminating duplicate packets, and interfacing with applications.
C-3	Backend servers	C-3.2	Application Server (AS)	Handles the LoRaWAN application layer payloads of the associated water meter and provides application-level service to end-user. It also generates all the application layer downlink payloads towards the connected water meters.

TABLE 5.1: Smart water metering system components (Cont'd)

SN	Component	Sub-SN	Sub-component	Description
C-3	Backend servers	C-3.3	Data Analysis Server (DAS)	Analysis the data based on the user application requirements.
		C-3.4	Management Server (MS)	Stores and manages the status of all devices and servers.
		C-3.5	Billing Server (BS)	Generates billing reports for end users.
		C-3.6	Data Processing Server (DPS)	Fetches data received by the DCU and transforms the data into the proper format for database storing.
		C-3.7	Data Query Server (DQS)	Provides web query API functions to support access the data stored in database, and it also generates the CSV or Excel files of a subset of data.

5.2 Component Functional Requirement

5.2.1 Smart water meter

Smart water meter is equipped with a device that allows continuous electronic reading, data processing, record generation/storage and sending the data over the LoRaWAN to the Network Server.

The requirements for each sub-component are listed below.

1) Meter Interfacing Unit (MIU)

   a. SHALL implement function F-1, F-3 in TABLE 5.1,
   b. SHALL support data types defined in [LHKS001-3] TABLE 5.1.

2) Clock

   a. Accuracy of the clock SHOULD be better than 20ppm,
   b. Timestamps are always given in UTC (Coordinated Universal Time),
   c. Water meter SHALL synchronize the internal clock with the Network Server periodically for time correction and heartbeat purposes,
   d. The internal clock SHOULD NOT deviate more than 60 seconds from the absolute time,

3) Communication Interfacing Unit (CIU)

   a. SHALL implement function F-4.1,

b. SHALL support class A communication mode defined in [LW103] and the information elements listed below in TABLE 5.2, SHOULD be provided by the water meter manufacturer:
c. SHALL only use the frequencies defined in TABLE 4.1 of [LHKS001-3] for communicating with Gateway,
d. SHALL have the following information before joining LoRaWAN:
   - DevEUI - a globally unique identifier,
   - AppKey - an AES-128 key.
   - AppEUI[1] - the issue organization SHOULD apply for the Application Server identifier.
e. SHOULD synchronize time with the Network Server at least once every month for keeping connection and rejoining after disconnection.

TABLE 5.2: Information element for CIU

Information element	Mandatory/Optional	Description
MIUID	Mandatory	ID of the Meter Interface Unit
MACVersion	Mandatory	Version of the LoRaWAN supported by the water meter.
MaxEIRP	Mandatory	Maximum EIRP supported by the water meter.
MaxDutyCycle	Optional	Maximum duty cycle supported by the water meter.
RFRegion	Mandatory	RF region name.
Supports32bitFCnt	Optional	Whether a water meter uses 32bit FCnt.

4) Data storage

a. SHALL implement function F-2.2,
b. SHALL be non-volatilememory independent of battery backup,
c. SHOULD retain up to 8 years without any auxiliary power,
d. SHOULD fulfill the requirements defined in [LHKS001-6].

5) Battery

a. SHOULD guarantee a shelf life of 15 years and a capacity life of 10 years under a 4-hour reporting frequency.

5.2.2 Data Concentration Unit (DCU)

The Data Concentration Unit is used to collect metering data at a programmable interval from smart water meters. In addition, the metering data stored in non-volatile memory of the DCU SHALL be transmitted to the Network Server(s) via a high throughput communication network, e.g., 3G/4G/5G/IP.

The requirements for each sub-component are listed below:

[1] Note that AppEUI field of the join-request in LoRaWAN 1.0/1.0.3 is renamed to JoinEUI field in LoRaWAN 1.1. The term JoinEUI is used to refer to the AppEUI in the context of LoRaWAN 1.0/1.0.3 in this standard.

1) Gateway

a. SHALL at least implement the functions below defined in [LW103]:
   - Forward all received LoRaWAN radio packets to the Network Server(s),
   - Decode uplink radio packets and forward them unprocessed to the Network Server(s),
   - For downlinks packets, execute transmission requests by the Network Server(s) without any interpretation of the payload.

b. SHALL support at least 8 concurrent communication frequency channels defined in [LHKS001-3],

c. SHOULD be capable of exchanging data with the Network Server(S) using 3G/4G/IP networks,

d. SHOULD at least support Transport Layer Security (TLS) Protocol [TLS] for communication with the Network Server,

e. SHALL obtain time synchronization/heartbeat messages from the Network Server and reply accordingly,

f. SHOULD regularly contact a separate Configuration and Update Server (CUPS) to check for configuration and software updates,

2) Configuration and Update Server (CUPS)

a. SHOULD provide and maintain different versions of Gateway configurations and software,

b. SHOULD at least support the Transport Layer Security (TLS) Protocol for communication with Gateway.

5.2.3 Backend Servers

1) Network Server

a. SHALL at least implement the following functions defined in [LW103]:
   - Checking the water meter address,
   - Frame authentication and frame counter checks,
   - Acknowledgements,
   - Data rate adaptation,
   - Responding to all MAC layer requests by the water meter,
   - Forwarding uplink application payloads to the appropriate Application Server,
   - Queuing of downlink payloads coming from any Application Server to any water meter connected to the network,
   - Handle Join-request and Join-accept messages between the water meter and the Network Server,

b. SHALL support all commands listed in TABLE 6.1 of [LHKS001-3],

c. SHALL be capable of exchanging data with Gateway using 3G/4G/IP networks,

d. SHALL be capable of exchanging data with other backend servers using IP networks,

e. SHOULD use JSON data format for sending requests and answer messages with the Network Server and Application Server,
f. SHALL support class A devices defined in [LW103],
f. SHALL at least support Over-The-Air Activation (OTAA),
g. SHOULD at least support Transport Layer Security (TLS) Protocol for communication with Gateway and other servers.
h. SHOULD make and manage channel plans and other network parameters centrally to optimize network performance,
i. SHOULD synchronize time with external NTP server over the NTPv4 protocol,
j. SHALL provide time synchronization for Gateway.
k. SHALL implement the Over-The-Air join procedure defined in [LW103],
l. SHALL contain the required information below to process the uplink Join-request frames and generate the downlink Join-accept frames:
    - DevEUI,
    - AppKey,
    - Network Server identifier,
    - Application Server identifier,
    - A way to select the preferred network in case several networks can serve the water meter,
    - The LoRaWAN version of the water meter,
m. SHALL support network and application session key derivation,
n. SHALL communicate the Network Session Key (NSK) of a water meter to the LoRaWAN Network Server and the Application Session Key (ASK) to the corresponding Application Server,
o. SHOULD provide secure provisioning, storage, and usage of root keys,
p. SHOULD at least support Transport Layer Security (TLS) Protocol [TLS] for communication with Gateway and other servers,
q. SHALL synchronize time with external NTP server over the NTPv4 protocol.

2) Application Server

a. SHALL support uplink data decryption and decoding,
b. SHALL support downlink queuing and downlink data encoding and encryption,
c. SHOULD support at least one of application layer communication protocols, e.g., by exchanging JSON messages over MQTT or HTTP, for applications to connect to an Application Server.
d. SHOULD provide repositories for persistently storing and managing collections of data,
e. SHALL synchronize time with external NTP server over the NTPv4 protocol.
f. SHALL provide the protocol or application interface for communicating with other servers.

3) Data Analysis Server

a. SHOULD provide tools for data analysis,

b. SHOULD provide data analysis results according to different requirements,
c. SHOULD store data analysis results at least for 1 year.

4) Management Server

a. SHOULD be able to get the statuses of all devices and servers,
b. SHOULD store the statuses of all devices and servers for at least 3 months,
c. SHOULD report warnings and errors.

5) Billing Server

a. SHOULD generate billing reports for each user,
b. SHOULD store billing results locally for at least 1 year.

6) Data Processing Server

a. SHOULD fetch data from other applications servers,
b. SHOULD transform the data into the appropriate format for storage in the database,
c. SHOULD store data for at least 1 year.

7) Data Query Server

a. SHOULD provide web query APIs for data accessing,
b. SHOULD generate CSV or Excel files for a subset of data.

LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard

Part 3: Communication Specification

Revision 1.0.1 / 2022-09

Water Supplies Department (WSD)
Hong Kong Applied Science and Technology Research Institute (ASTRI)
The University of Hong Kong (HKU)

Document Status

Document Name	LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard
Document Code	A-1
Author	• Water Supplies Department (WSD) • Hong Kong Applied Science and Technology Research Institute (ASTRI) • The University of Hong Kong (HKU)
Revision Number	1.0.1
Document Status	Released
Contributors	Water Supplies Department: • Tin Tak CHENG • Chi Lun CHAN • Ming Kwong WONG • Samuel K.S. Chung Hong Kong Applied Science and Technology Research Institute: • Dr. Abel Z. YANG • Jianxiong WANG • Carlos C. H. CHIU • Henry K. H. FONG • Hang LIU • Lei ZHU • Dr. Miao MA The University of Hong Kong: • Prof. Lawrence K. YEUNG • Dr. Vincent W.L. TAM

Document Revision History

Revision	Date issued	Reviewed by	Approved by	Date approved	Revision type
1.0.0	April 2021	WSD	WSD	April 2021	Initial Release
1.0.1	September 2022	WSD	WSD	September 2022	Minor Revision

This Hong Kong Standard was approved by WSD in September 2022.

This Hong Kong Standard exists in one official version (English).

Table of Contents

Foreword

This document (LHKS001-1:2022) has been prepared by Technical Committee formed by Water Supply Department (WSD), Hong Kong Applied Science and Technology Research Institute (ASTRI), and The University of Hong Kong (HKU).

This document shall be adopted as a Hong Kong standard, either by publication of an identical text or by endorsement, at the latest by April 2022, and conflicting Hong Kong standards shall be withdrawn at the latest by April 2022.

It is possible that some of the elements of this documents may be subject to patent rights. WSD, ASTRI, and HKU shall not be held responsible for identifying any patent rights.

This document has been prepared under a mandate given to ASTRI by WSD.

LHKS001 comprises of the following parts:

- *Part 1: General*
- *Part 2: System Specification*
- *Part 3: Communication Specification*
- *Part 4: Event Specification*
- *Part 5: Security Specification*
- *Part 6: Storage Specification*
- *Part 7: Inspection Specification*

1. Introduction

This document belongs to the LHKS001 series.

This document describes the LoRaWAN-based Wireless Smart Water Metering system structure, function, and performance requirements in a generic way. This is Part 3 of LHKS001.

Additional parts to the series of the standard LHKS001 are:

- *Part 1: General*
- *Part 2: System Specification*
- *Part 4: Event Specification*
- *Part 5: Security Specification*
- *Part 6: Storage Specification*
- *Part 7: Inspection Specification*

Since the realm of water metering is going through a period of rapid change, it is likely that this and other parts of the standard will require amendments soon.

2. Normative References

The following documents, as a whole or in part, are referenced in this document and are indispensable for its applications.

[LHKS001-1] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 1: General, September 2022.

[LHKS001-2] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 2: System Specification, September 2022.

[LHKS001-4] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 4: Event Specification, September 2022.

[LWRP103] LoRaWAN 1.0.3 Regional Parameters, Revision A, LoRa Alliance, July 2018.

[LW103] LoRaWAN Specification, Version 1.0.3, LoRa Alliance, July 2018.

[S32IEEESTD754] IEEE standard 754 for binary floating-point arithmetic.

[EN13757-3] EN 13757-3, Communication systems for meters and remote reading of meters — Part 3: Dedicated application layer.

3. Terms and Conventions

3.1 Terms

Terms used in this specification are listed in TABLE 3.1.

TABLE 3.1: Terms

Term	Definition
SWM	Smart Water Meter
LoRa™	Long Range modulation technique
LoRaWAN™	Long Range network protocol
MAC	Medium Access Control
NS	Network Server
GW	Gateway
OTA	Over-the-Air
ABP	Activation By Personalization
DR	Data Rate
ADR	Adaptive Data Rate
NwkSKey	Network Session Key
AppSKey	App Session Key
Rx1	Receive Window 1
Rx2	Receive Window 2

3.2 Conventions

The conventions used in this document are listed below.

SHALL - the use of the word 'SHALL indicates a mandatory requirement.
SHOULD - the use of the word 'SHOULD' indicates a requirement for good practice, which should be implemented whenever possible.
MAY - the use of the word 'may' indicates a desirable requirement.

4. Frequency Planning

The operation of each water meter SHALL comply with the AS923 channel plan and regional parameters defined in [LWRP103].

The frequency plan definitions are shown in TABLE 4.1.

TABLE 4.1 Frequency plans

Channel	Central Frequency (MHz)	Bandwidth (kHz)	Spreading Factor
0	923.20	125	SF7-SF12
1	923.40	125	SF7-SF12
2	923.60	125	SF7-SF12
3	923.80	125	SF7-SF12
4	924.00	125	SF7-SF12
5	924.20	125	SF7-SF12
6	924.40	125	SF7-SF12
7	924.7475	250	SF7

There are eight channels that can be used for data transmission, i.e., channel 0 to 7. Channels 0-6 have a span of 125 kHz bandwidth, and channel 7 spans 250 kHz bandwidth. Channel 0-6 SHALL be used for LoRa signals using the spreading factors of 7-12, and channel 7 SHALL only be used for LoRa signals with a spreading factor of 7. Uplink and downlink Rx1 [LW103] data SHALL use channel 0-7, and downlink Rx2 [LW103] data SHALL use channel 0 with a spreading factor of 10.

Chanel 0 and 1 SHALL be implemented in every water meter and SHALL NOT be modified through the LoRaWAN NewChannelReq command [LW103].

The data rate used in this specification SHALL be listed in TABLE 4.2.

TABLE 4.2: Data rate code

DataRate	**Modulation**	**SF**	**BW (kHz)**	**bit/s**
0	LoRa	12	125	250
1	LoRa	11	125	440
2	LoRa	10	125	980
3	LoRa	9	125	1760
4	LoRa	8	125	3125
5	LoRa	7	125	5470
6	LoRa	7	250	11000
7	FSK 50 kbps			50000

5. Data Types and Payload Format Definitions

5.1 Data Types

The data types are divided into three categories: measurement, status and event, and device information, and they are listed in TABLE 5.1. The definitions and results of detection for the events are detailed in [LHKS001-4].

TABLE 5.1: Data types

	Sub-class	Data type	1 x per 4 hours	1 x per half hour	1 x per day	On event
Measurement	Instant	Instant forward volume				√
		Instant backward volume				√
		Instant flow rate				√
		Current time			√	√
	Periodic	Half-hour volume		√		
		8 half-hour volumes of past 4 hours	√			
		Maximum and minimum flow rate of past day			√	
Status and Event	Status	Remaining battery life				√
		Most recent reset time				√
		Reset times				√
		Most recent time correction time				√
		Time correction times				√
		Status summary	√			
	Event	Flow leakage				√
		Flow leakage clear				√
		Reverse flow				√
		Reverse flow clear				√
		Tamper				√
		Tamper clear				√
		Removal				√
		Battery low				√
		Battery EOL (End of Life)				√
		Hardware error				√
		Hardware error clear				√
Device Information		Firmware version				√
		Production number				√
		Hardware version				√
		LoRaWAN version				√
		MIU ID				√
		Manufacturer-specific info				

5.2 Data Type Encoding Methods and Payload Format Definition

5.2.1 Encoding Methods

Signed BCD, Floating-Point Binary, Binary integer, ASCII, and the Boolean encoding method SHALL be supported.

1) Signed BCD encoding

The BCD codes that is used in this specification comprise of decimal numbers 1-10 and are listed in TABLE 5.2.

TABLE 5.2: BCD encoding of number 1-10

Decimal Number	BCD Code
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Note that the digit's values of Ah – Eh in any digit position signals are invalid; a hex code Fh in the MSD position signals a negative BCD number in the remaining digits, and in any other digit position it signals an error.

2) Floating-point binary

The floating point in this specification refers to [R32IEEESTD754]. TABLE 5.3 shows the structure of the short, real number representation. The bit on the left is the most significant.

TABLE 5.3: Floating-point number representation

31	30	…	23	22	…	0
Sign	Exponent			Fraction		

The sign of a binary floating-point number is represented by a single bit. A 1 bit indicates a negative number, and a 0 bit indicates a positive number.

Exponents are stored as 8-bit unsigned integers with a bias of 127. The fraction part is normalized.

For example, the floating-point binary value 1101.101 is normalized as 1.101101×2^3 by moving the decimal point 3 positions to the left, and multiplying by 2^3, and the leading 1 is omitted because it is redundant. Thus, the sign is 0, the exponent is 10000010, and the fraction part is 101101.

3) Binary Integer

Binary integer in this specification is based on [EN13757-3], and the leftmost bit is the most significant. A 1 bit indicates a negative number, and a 0 bit indicates a positive number. The code "1000…0000b" signals "invalid". The encoding is shown in TABLE 5.4.

TABLE 5.4: Binary integer encoding number representation

	Byte format							
Byte 1	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Byte 2	Sign	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^9	2^8

4) ASCII

The characters supported by this specification and its ASCII encoding is shown in TABLE 5.5.

TABLE 5.5: Supported characters and the ASCII encoding

Binary	Dec	Hex	Character	Binary	Dec	Hex	Character	Binary	Dec	Hex	Character
010 0000	32	20	space	100 1010	74	4A	J	110 0101	101	65	e
011 0000	48	30	0	100 1011	75	4B	K	110 0110	102	66	f
011 0001	49	31	1	100 1100	76	4C	L	110 0111	103	67	g
011 0010	50	32	2	100 1101	77	4D	M	110 1000	104	68	h
011 0011	51	33	3	100 1110	78	4E	N	110 1001	105	69	i
011 0100	52	34	4	100 1111	79	4F	O	110 1010	106	6A	j
011 0101	53	35	5	101 0000	80	50	P	110 1011	107	6B	k
011 0110	54	36	6	101 0001	81	51	Q	110 1100	108	6C	l
011 0111	55	37	7	101 0010	82	52	R	110 1101	109	6D	m
011 1000	56	38	8	101 0011	83	53	S	110 1110	110	6E	n
011 1001	57	39	9	101 0100	84	54	T	110 1111	111	6F	o
100 0000	64	40	@	101 0101	85	55	U	111 0000	112	70	p
100 0001	65	41	A	101 0110	86	56	V	111 0001	113	71	q
100 0010	66	42	B	101 0111	87	57	W	111 0010	114	72	r
100 0011	67	43	C	101 1000	88	58	X	111 0011	115	73	s
100 0100	68	44	D	101 1001	89	59	Y	111 0100	116	74	t
100 0101	69	45	E	101 1010	90	5A	Z	111 0101	117	75	u

TABLE 5.5: Supported characters and the ASCII encoding (Cont'd)

Binary	Dec	Hex	Character	Binary	Dec	Hex	Character	Binary	Dec	Hex	Character
100 0110	70	46	F	110 0001	97	61	a	111 0110	118	76	v
100 0111	71	47	G	110 0010	98	62	b	111 0111	119	77	w
100 1000	72	48	H	110 0011	99	63	c	111 1000	120	78	x
100 1001	73	49	I	110 0100	100	64	d	111 1001	121	79	y
010 1101	45	2D	-	101 1111	95	5F	underline	111 1010	122	7A	z
010 1110	46	2E	.								

5) Boolean

Boolean in this specification is based on [EN13757-3]. A 1 bit indicates true, and a 0 bit indicates false. The encoding is shown in TABLE 5.6.

TABLE 5.6: Boolean encoding

	Byte format							
Byte 1	1 or 0	1 or 0	1 or 0	1 or 0	1 or 0	1 or 0	1 or 0	1 or 0

5.2.2 Payload Data Structure

The data for the report follows the payload structure in TABLE 5.7, which includes four fields: type, encoding, length and value. All data follow the Type-Encoding-Length-Value payload format for network transmission. The “type” field identifies a data type. The “encoding” fieldidentifies the encoding method of the value field through the 3 lowest bits. The “length” field specifies the data length, and the “value” field stores the encoded data. For example, the “type” field 0x0A indicates the data transmission is half hour volume. The “encoding” field 0x00 indicates BCD encoding is used, the “length” field 0x04 indicates the length of data is 4 bytes, and the “value” field stores the encoded half hour volume.

Note that “type”, “encoding” and “length” fields SHALL use binary integer for encoding. If multiple encoding methods are available for a data type, the specific encoding method to be used is subject to the manufacturer.

If not specified otherwise, the byte order will be Most Significant Bit (MSB) first. Values that are unknown or not available to a device will be marked as 0xFF in all bytes of the value field.

TABLE 5.7: Payload data structure

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value
Half hour volume (0x09)	BCD encoding (0x00)	4 bytes (0x00000004)	Data

Note that the most significant 4 bits of encoding field are reserved for future usage, and each bit SHOULD be set to 0. The encoding field bit setting is shown in TABLE 5.8. The most significant bit is on the left.

TABLE 5.8: Encoding field bit setting

Field value								Encoding method
0	0	0	0	0	0	0	0	BCD encoding
0	0	0	0	0	0	0	1	Floating-point binary encoding
0	0	0	0	0	0	1	0	Binary integer encoding
0	0	0	0	0	0	1	1	ASCII encoding
0	0	0	0	0	1	0	0	Boolean encoding
0	0	0	0	0	1	0	1	No encoding
0	0	0	0	0	1	1	1	Multiple encoding

0	0	0	0	1	1	1	1	Not defined[1]

5.2.3 Payload value format

1) Date and Time

The date and time data structure records the date (year, month and day) and the time (hour and minute), and the format for its payload value is shown in TABLE 5.9. Note that 24-hour format SHALL be used for the hour field.

TABLE 5.9: Date and Time payload value format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (5 Bytes)
Date and Time (0x00)	BCD encoding (0x00)	5 bytes (0x05)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 3 (day)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 4 (month)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 5 (year)	Ten (BCD encoding)				Unit (BCD encoding)			

2) Date

The date data structure records the date (year, month and day), and the format for its payload value is shown in TABLE 5.10.

TABLE 5.10: Date payload value format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (3 Bytes)
Date (0x01)	BCD encoding (0x00)	3 bytes (0x03)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (day)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (month)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 3 (year)	Ten (BCD encoding)				Unit (BCD encoding)			

3) Time

The time data structure records the time (hour and minute), and the format for its payload value is shown in TABLE 5.11. Note that the 24-hour format SHOULD be used for the hour field.

[1] If a meter or network server is required to reply with packet with a encoding method that it does not support, 0x0F SHOULD be used in the encoding field.

TABLE 5.11: Time payload value format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (2 Bytes)
Time (0x02)	BCD encoding (0x00)	2 Bytes (0x02)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			

4) Time Duration

The time duration data structure records the continuous time interval, and the format for its payload value is shown in TABLE 5.12.

TABLE 5.12: Time Duration payload value format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (3 Bytes)
Time Duration (0x03)	BCD encoding (0x00)	3 bytes (0x03)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 3 (day)	Ten (BCD encoding)				Unit (BCD encoding)			

5) Instant Forward Volume

The Instant Forward Volume data structure records the total forward volume since the installation date at the asking time, and its payload value format with the BCD and the floating-point binary are shown in TABLE 5.13 and 5.14 respectively. The volume SHOULD be measured in cubic metres (m^3).

TABLE 5.13: Instant Forward Volume Payload Value Format with BCD Encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (4 Bytes)
Instant Forward Volume (0x04)	BCD encoding (0x00)	4 bytes (0x04)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 2	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 3	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 4	Ten thousand (BCD encoding)				Thousands (BCD encoding)			

TABLE 5.14: Instant Forward Volume Payload Value Format with Floating-point Binary Encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (4 Bytes)
Instant Forward Volume (0x04)	Floating-point binary encoding (0x01)	4 bytes (0x04)	Data

31	30	…	23	22	…	0
Sign	Exponent			Fraction		

6) Instant Backward Volume

The Instant Backward Volume data structure records the total backward volume since the installation date at the asking time, and the format for its payload volume with the BCD and the floating-point binary are shown in TABLE 5.15 and 5.16 respectively. The volume SHOULD be measured in cubic metres (m^3).

TABLE 5.15: The Instant Forward Volume payload value format with BCD encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (4 Bytes)
Instant Backward Volume (0x05)	BCD encoding (0x00)	4 bytes (0x04)	Data

TABLE 5.15: Instant Forward Volume Payload Value Format with BCD Encoding (Cont'd)

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 2	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 3	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 4	Ten thousand (BCD encoding)				Thousands (BCD encoding)			

TABLE 5.16: Instant Forward Volume Payload Value Format with Floating-point Binary Encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (4Bytes)
Instant Backward Volume (0x05)	Floating-point binary encoding (0x01)	4 bytes (0x04)	Data

31	30	…	23	22	…	0
Sign	Exponent			Fraction		

7) Instant Flow Rate

The Instant Flow Rate data structure records the flow rate requested, which equals to the volume (in cubic metres) per hour[2] flowing through the water meter, and the format for its payload value with the BCD and the floating-point binary are shown in TABLE 5.17 and 5.18 respectively. The flow rate SHOULD be measured in cubic metres per hour (m^3/hour).

TABLE 5.17: Instant Flow Rate Payload Value Format with BCD Encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (4 Bytes)
Instant flow rate (0x06)	BCD encoding (0x00)	4 bytes (0x04)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 2	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 3	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 4	Ten thousand (BCD encoding)				Thousands (BCD encoding)			

TABLE 5.18: Instant Flow Rate Payload Value Format with Floating-point Binary Encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (4 Bytes)
Instant flow rate (0x06)	Floating-point binary encoding (0x01)	4 bytes (0x04)	Data

31	30	…	23	22	…	0
Sign	Exponent			Fraction		

8) Half-hour Forward Volume

The Half-hour Forward Volume data structure records the forward volume during half an hour, and the format for its payload value with BCD and the encoding for the floating-point binary are shown in TABLE 5.19 and 5.20 respectively. The volume SHOULD be measured in cubic metres (m^3).

TABLE 5.19: The Half-hour Forward Volume Payload Value Format with BCD Encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (4 Bytes)
Half-hour Forward Volume (0x07)	BCD encoding (0x00)	4 bytes (0x04)	Data

[2] The instant flow rate calculation at the asking time is subjected to water meter manufacturer. One simple method is returning the difference between the meter reading after 1 (or 30) minute(s) and the meter reading of the asking time.

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 2	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 3	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 4	Ten thousand (BCD encoding)				Thousands (BCD encoding)			

TABLE 5.20: Half-hour Forward Volume Payload Value Format with Floating-point Binary Encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (4 Bytes)
Half-hour Forward Volume (0x07)	Floating-point binary encoding (0x01)	4 bytes (0x04)	Data

31	30	…	23	22	…	0
Sign	Exponent			Fraction		

9) Half-hour Backward Volume

The Half-hour Backward Volume data structure records the backward volume during half an hour, and the format for its payload value with the BCD and the floating-point binary are shown in TABLE 5.21 and 5.22 respectively. The volume SHOULD be measured in cubic metres (m^3).

TABLE 5.21: Half-hour Backward Volume Payload Value Format with BCD Encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (4 Bytes)
Half-hour Backward Volume (0x08)	BCD encoding (0x00)	4 bytes (0x04)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 2	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 3	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 4	Ten thousand (BCD encoding)				Thousands (BCD encoding)			

TABLE 5.22: Half-hour Forward Volume Payload Value Format with Floating-point Binary Encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (4 Bytes)
Half-hour Backward Volume (0x08)	Floating-point binary encoding (0x01)	4 bytes (0x04)	Data

31	30	…	23	22	…	0
Sign	Exponent			Fraction		

10) 8 Half-hour Forward Volumes of the past 4 hours

The 8 Half-hour Forward Volumes of the past 4 hours data structure records the 8 half-hour forward volumes for the past 4 hours, and the format for its payload value with the BCD and the floating-point binary are shown in TABLE 5.23 and 5.24 respectively. The volume SHOULD be measured in cubic metres (m^3).

TABLE 5.23: 8 Half-hour Forward Volumes of Past 4 Hours Payload Value Format with BCD Encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (32 Bytes)
8 Half-hour Forward volumes of past 4 hours (0x09)	BCD encoding (0x00)	32 bytes (0x20)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
	Data point 1							
Byte 1	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 2	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 3	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 4	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
	Data point 2							
Byte 5	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 6	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 7	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 8	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
	Data point 3							
Byte 9	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 10	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 11	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 12	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
	Data point 4							
Byte 13	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 14	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 15	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 16	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
	Data point 5							
Byte 17	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 18	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 19	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 20	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
	Data point 6							
Byte 21	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 22	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 23	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 24	Ten thousand (BCD encoding)				Thousands (BCD encoding)			

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
	Data point 7							
Byte 25	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 26	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 27	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 28	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
	Data point 8							
Byte 29	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 30	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 31	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 32	Ten thousand (BCD encoding)				Thousands (BCD encoding)			

TABLE 5.24: 8 Half-hour Forward Volumes of Past 4 Hours Payload Value Format with Floating-point Binary Encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (32 Bytes)
8 Half-hour Forward volumes of past 4 hours (0x09)	Floating-point binary encoding (0x01)	32 bytes (0x20)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
	Data point 1							
Byte 1	Fraction							
Byte 2	Fraction							
Byte 3	Exponent	Fraction						
Byte 4	Sign	Exponent						
	Data point 2							
Byte 5	Fraction							
Byte 6	Fraction							
Byte 7	Exponent	Fraction						
Byte 8	Sign	Exponent						
	Data point 3							
Byte 9	Fraction							
Byte 10	Fraction							
Byte 11	Exponent	Fraction						
Byte 12	Sign	Exponent						
	Data point 4							
Byte 13	Fraction							
Byte 14	Fraction							
Byte 15	Exponent	Fraction						
Byte 16	Sign	Exponent						
	Data point 5							
Byte 17	Fraction							
Byte 18	Fraction							
Byte 19	Exponent	Fraction						
Byte 20	Sign	Exponent						

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
	Data point 6							
	Byte format							
Byte 21	Fraction							
Byte 22	Fraction							
Byte 23	Exponent	Fraction						
Byte 24	Sign	Exponent						
	Data point 7							
Byte 25	Fraction							
Byte 26	Fraction							
Byte 27	Exponent	Fraction						
Byte 28	Sign	Exponent						
	Data point 8							
Byte 29	Fraction							
Byte 30	Fraction							
Byte 31	Exponent	Fraction						
Byte 32	Sign	Exponent						

11) The 8 Half-hour Backward Volumes of the past 4 hours

The 8 Half-hour Backward Volumes of the past 4 hours data structure records the 8 half-hour backward volumes for the past 4 hours, and the format for its payload value with the BCD and the floating-point binary are shown in TABLE 5.25 and 5.26 respectively. The volume SHOULD be measured in cubic metres (m^3).

TABLE 5.25: 8 Half-hour Backward Volumes of Past 4 Hours Payload Value Format with BCD Encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (32 Bytes)
8 Half-hour Backward volumes of past 4 hours (0x0A)	BCD encoding (0x00)	32 bytes (0x20)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
	Data point 1							
Byte 1	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 2	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 3	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 4	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
	Data point 2							
Byte 5	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 6	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 7	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 8	Ten thousand (BCD encoding)				Thousands (BCD encoding)			

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
	Data point 3							
Byte 9	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 10	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 11	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 12	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
	Data point 4							
Byte 13	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 14	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 15	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 16	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
	Data point 5							
Byte 17	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 18	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 19	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 20	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
	Data point 6							
Byte 21	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 22	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 23	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 24	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
	Data point 7							
Byte 25	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 26	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 27	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 28	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
	Data point 8							
Byte 29	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 30	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 31	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 32	Ten thousand (BCD encoding)				Thousands (BCD encoding)			

TABLE 5.26: 8 Half-hour Backward Volumes of Past 4 Hours Payload Value Format with Floating-point Binary Encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (32 Bytes)
8 Half-hour Backward volumes of past 4 hours (0x0A)	Floating-point binary encoding (0x01)	32 bytes (0x20)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
	Data point 1							
Byte 1	Fraction							
Byte 2	Fraction							
Byte 3	Exponent	Fraction						
Byte 4	Sign	Exponent						
	Data point 2							

Byte 5	Fraction	
Byte 6	Fraction	
Byte 7	Exponent	Fraction
Byte 8	Sign	Exponent
	Data point 3	
Byte 9	Fraction	
Byte 10	Fraction	
Byte 11	Exponent	Fraction
Byte 12	Sign	Exponent
	Data point 4	
Byte 13	Fraction	
Byte 14	Fraction	
Byte 15	Exponent	Fraction
Byte 16	Sign	Exponent
	Data point 5	
Byte 17	Fraction	
Byte 18	Fraction	
Byte 19	Exponent	Fraction
Byte 20	Sign	Exponent
	Data point 6	
	Byte format	
Byte 21	Fraction	
Byte 22	Fraction	
Byte 23	Exponent	Fraction
Byte 24	Sign	Exponent
	Data point 7	
Byte 25	Fraction	
Byte 26	Fraction	
Byte 27	Exponent	Fraction
Byte 28	Sign	Exponent
	Data point 8	
Byte 29	Fraction	
Byte 30	Fraction	
Byte 31	Exponent	Fraction
Byte 32	Sign	Exponent

12) Maximum and Minimum Flow Rate of Past Day

The Maximum and Minimum Flow Rate of the past day data structure records the maximum and minimum flow rate for the past day and the corresponding time of occurrence, and the format for its payload value with BCD and the floating-point binary are shown in TABLE 5.27 and 5.28 respectively. The flow rate SHOULD be measured in cubic metres per hour (m^3/hour).

TABLE 5.27: Maximum and Minimum Flow Rate of Past Day Payload Value format with BCD Encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (12 Bytes)
Maximum and minimum flow rate of past day (0x0B)	BCD encoding (0x00)	12 bytes (0x0C)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
	Maximum flow rate of past day							
Byte 1	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 2	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 3	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 4	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
	Minimum flow rate of past day							
Byte 5	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 6	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 7	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 8	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
	Time of occurrence of maximum flow rate of past day							
Byte 9 (min)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 10 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			
	Time of occurrence of minimum flow rate of past day							
Byte 11 (min)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 12 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			

TABLE 5.26: Maximum and Minimum Flow Rate of Past Day Payload Value Format with Half-precision Floating-point Binary Encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (12 Bytes)
8 Half-hour Backward volumes of past 4 hours (0x0B)	Floating-point binary encoding (0x01)	12 bytes (0x0C)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
	Maximum flow rate							
Byte 1	Fraction							
Byte 2	Fraction							
Byte 3	Exponent	Fraction						
Byte 4	Sign	Exponent						
	Minimum flow rate							
Byte 5	Fraction							
Byte 6	Fraction							
Byte 7	Exponent	Fraction						
Byte 8	Sign	Exponent						
	Time of maximum flow rate							
Byte 9 (min)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 10 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			
	Time of minimum flow rate							
Byte 11 (min)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 12 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			

13) Remaining Battery Life

The Remaining Battery Life data structure records the remaining percentage of battery life, and the format for its payload value with BCD and the encoding for the floating-point binary are shown in TABLE 5.27. Note that any number greater than 100 (100% of remaining percentage of battery life) is invalid.

TABLE 5.27: Remaining Battery Life payload value format with binary integer encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (1 Byte)
Remaining battery life (0x0C)	Binary integer encoding (0x02)	1 byte (0x01)	Data

	Byte format							
Byte 1	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

Battery life percentage	Value (Decimal)	Value (Binary)
0%	0	00000000
1%	1	00000001
2%	2	00000010
…	…	…

14) Most Recent Reset Time

The Most Recent Reset Time data structure records the most recent date and time that the water meter was reset, and the format for its payload value is shown in TABLE 5.28.

TABLE 5.28: Most Recent Reset Time Payload Value Format

Type (1Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (5 Bytes)
Most Recent Reset Time (0x0D)	BCD encoding (0x00)	5 bytes (0x05)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 3 (day)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 4 (month)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 5 (year)	Ten (BCD encoding)				Unit (BCD encoding)			

15) Reset Times

The Reset Times data structure records the number of times that the water meter has been reset, and the format for its value format is shown in TABLE 5.29. Only the binary integer applies to this data type.

TABLE 5.29: Reset Times payload value format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (1 Byte)
Reset Times (0x0E)	Binary integer encoding (0x02)	1 byte (0x01)	Data

	Byte format							
Byte 1	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

16) Most Recent Time Correction Time

The Most Recent Time Correction Time data structure records the most recent date and time that the water meter's time was corrected, and the format for its payload value is shown in TABLE 5.30.

TABLE 5.30: Most Recent Time Correction Time Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (5 Bytes)
Most Recent Time Correction Time (0x0F)	BCD encoding (0x00)	5 bytes (0x05)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 3 (day)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 4 (month)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 5 (year)	Ten (BCD encoding)				Unit (BCD encoding)			

17) Time Correction Times

The Time Correction Times data structure records the number of times that the water meter's time has been corrected, and the format for its payload format is shown in TABLE 5.31. Only the binary integers applies to this data type.

TABLE 5.31: Time Correction Times Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (1 Byte)
Time Correction Times (0x10)	Binary integer encoding (0x02)	1 byte (0x01)	Data

	Byte format							
Byte 1	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

18) Flow leakage Event Time

The Flow Leakage Event Time data structure records the time when a flow leakage event is detected, and the format for its payload value is shown in TABLE 5.32.

TABLE 5.32. Flow Leakage Event Time Payload Value Format with BCD Encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (2 Bytes)
Flow leakage Event Time (0x11)	BCD encoding (0x00)	2 bytes (0x02)	Data

	Time	
Byte 1 (minute)	Ten (BCD encoding)	Unit (BCD encoding)
Byte 2 (hour)	Ten (BCD encoding)	Unit (BCD encoding)

19) Flow Leakage Clear Event Time

The Flow Leakage Clear Event Time data structure records the time when a flow leakage clear event is detected, and the format for its payload value is shown in TABLE 5.33.

TABLE 5.33. Flow Leakage Clear Event Time payload value format with BCD encoding

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (2 Bytes)
Flow Leakage Clear Event Time (0x12)	BCD encoding (0x00)	2 bytes (0x02)	Data

	Time	
Byte 1 (minute)	Ten (BCD encoding)	Unit (BCD encoding)
Byte 2 (hour)	Ten (BCD encoding)	Unit (BCD encoding)

20) Reverse Flow Event Time

The Reverse Flow Event Time data structure records the time when a reverse flow event is detected, and the format for its payload value is shown in TABLE 5.34.

TABLE 5.34: Reverse Flow Event Time Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (2 Bytes)
Reverse Flow Event Time (0x13)	BCD encoding (0x00)	2 bytes (0x02)	Data

	Time	
Byte 1 (minute)	Ten (BCD encoding)	Unit (BCD encoding)
Byte 2 (hour)	Ten (BCD encoding)	Unit (BCD encoding)

21) Reverse Flow Clear Event Time

The Reverse Flow Clear Event Time data structure records the time when a reverse flow clear event is detected, and the format for its payload value is in TABLE 5.35.

TABLE 5.35: Reverse Flow Clear Event Time Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (2 Bytes)
Reverse Flow Clear Event Time (0x14)	BCD encoding (0x00)	2 bytes (0x02)	Data

	Time	
Byte 1 (minute)	Ten (BCD encoding)	Unit (BCD encoding)
Byte 2 (hour)	Ten (BCD encoding)	Unit (BCD encoding)

22) Tamper Event Time

The Tamper Event Time data structure records the time when a tamper event is detected, and the format for its payload value is shownin TABLE 5.36.

TABLE 5.36: Tamper Event Time Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (2 Bytes)
Tamper Event Time (0x15)	BCD encoding (0x00)	2 bytes (0x02)	Data

	Time	
Byte 1 (minute)	Ten (BCD encoding)	Unit (BCD encoding)
Byte 2 (hour)	Ten (BCD encoding)	Unit (BCD encoding)

23) Tamper Clear Event Time

The Tamper Clear Event Time data structure records the time when a tamper clear event is detected, and the format for its payload value is shown in TABLE 5.37.

TABLE 5.37: Tamper Clear Event Time Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (2 Bytes)
Tamper Clear Event Time (0x16)	BCD encoding (0x00)	2 bytes (0x02)	Data

	Time	
Byte 1 (minute)	Ten (BCD encoding)	Unit (BCD encoding)
Byte 2 (hour)	Ten (BCD encoding)	Unit (BCD encoding)

24) Battery Low Event Time

The Battery Low Event Time data structure records the time when a battery low event is detected, and the format for its payload value is shown in TABLE 5.38.

TABLE 5.38: Battery Low Event Time Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (2 Bytes)
Battery Low Event Time (0x17)	BCD encoding (0x00)	2 bytes (0x02)	Data

	Time	
Byte 1 (minute)	Ten (BCD encoding)	Unit (BCD encoding)
Byte 2 (hour)	Ten (BCD encoding)	Unit (BCD encoding)

25) Battery EOL Event Time

The Battery EOL Event Time data structure records the time when a battery End of Life (EOL) event is detected, and the format for its payload value is shown in TABLE 5.39.

TABLE 5.39: Battery EOL Event Time Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (2 Bytes)
Battery EOL Event Time (0x18)	BCD encoding (0x00)	2 bytes (0x02)	Data

	Time	
Byte 1 (minute)	Ten (BCD encoding)	Unit (BCD encoding)
Byte 2 (hour)	Ten (BCD encoding)	Unit (BCD encoding)

26) Hardware Error Event Time

The Hardware Error Event Time data structure records the time when a hardware error is detected, and the format for its payload value is shown in TABLE 5.40.

TABLE 5.40: Hardware Error Event Time Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (2 Bytes)
Hardware Error Event Time (0x19)	BCD encoding (0x00)	2 bytes (0x02)	Data

	Time	
Byte 1 (minute)	Ten (BCD encoding)	Unit (BCD encoding)
Byte 2 (hour)	Ten (BCD encoding)	Unit (BCD encoding)

27) Hardware Error Event Clear Time

The Hardware Error Event Clear Time data structure records the time when a hardware error clear is detected, and the format for its payload value is shown in TABLE 5.41.

TABLE 5.41: Hardware Error Event Clear Time Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (2 Bytes)
Hardware Error Event Clear Time (0x1A)	BCD encoding (0x00)	2 bytes (0x02)	Data

	Time	
Byte 1 (minute)	Ten (BCD encoding)	Unit (BCD encoding)
Byte 2 (hour)	Ten (BCD encoding)	Unit (BCD encoding)

28) Status Summary

The Status Summary data structure records a set of status summary, and the format for its payload value is shown in TABLE 5.42.

TABLE 5.42: Status Summary Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (2 Bytes)
Status Summary (0x1B)	Boolean (0x04)	2 bytes (0x02)	Data

	Definition (0-normal, 1-abnormal)							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1	reserved	tamper	Hardware error	Battery EOL	Battery low	Time reset	Reverse flow	Flow leakage
Byte 2	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved

29) Firmware Version

The Firmware Version data structure records the firmware version of the water meter, and the format for its payload value is shown in TABLE 5.43. In the value field, either ASCII or BCD is allowed for this data type.

TABLE 5.43: Firmware Version Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (8 Bytes)
Firmware version (0x1C)	ASCII (0x03) or BCD encoding (0x00)	8 bytes (0x08)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1								
Byte 2								
Byte 3								
Byte 4								
Byte 5								
Byte 6								
Byte 7								
Byte 8								

30) Production Number

The Production Number data structure records the production number of the water meter, and the format for its payload value is shown in TABLE 5.44. In the value field, either ASCII or BCD is allowed for this data type.

TABLE 5.44: Production Number Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (8 Bytes)
Production Number (0x1D)	ASCII (0x03) or BCD encoding (0x00)	8 bytes (0x08)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1								
Byte 2								
Byte 3								
Byte 4								
Byte 5								
Byte 6								
Byte 7								
Byte 8								

31) Hardware Version

The Hardware Version data structure records the hardware version of the water meter, and the format for its payload value is shown in TABLE 5.45. In the value field, either ASCII or BCD is allowed for this data type.

TABLE 5.45: Hardware Version Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (4 Bytes)
Hardware Version (0x1E)	ASCII (0x03) or BCD encoding (0x00)	4 bytes (0x04)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1								
Byte 2								
Byte 3								
Byte 4								

32) LoRaWAN version

The LoRaWAN version data structure records the LoRaWAN version of the water meter using, and the format for its payload value is shown in TABLE 5.46. In the value field, only ASCII is allowed for this data type.

TABLE 5.46: LoRaWAN Version Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (5 Bytes)
LoRaWAN Version (0x1F)	ASCII (0x03)	5 bytes (0x05)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1								
Byte 2								
Byte 3								
Byte 4								
Byte 5								

33) MIU ID

The MIU ID data structure records the ID of the Meter Interface Unit (MIU), and and the format for its payload value is shown in TABLE 5.47. In the value field, either BCD or ASCII is allowed for this data type.

TABLE 5.47: MIU ID Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (8 Bytes)
MIU ID (0x20)	ASCII (0x03) or BCD encoding (0x00)	8 bytes (0x08)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1								
Byte 2								
Byte 3								
Byte 4								
Byte 5								
Byte 6								
Byte 7								
Byte 8								

34) Manufacturer-specific Info

The Manufacturer-specific Info data structure allows different manufacturers for internal testing, and the format for its payload value is shown in TABLE 5.48 In the value field, the encoding method to be used is subject to the manufacturer. Note that Byte 1 is reserved for specifying Manufacturer ID[3], the other fields content and encoding method are subject to the usage purpose of the manufacturer.

TABLE 5.48: Manufacturer-specific Info Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (8 Bytes)
Manufacturer-specific Info (0x21)		8 bytes (0x08)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1	Manufacturer ID (Binary integer encoding)							
Byte 2								
Byte 3								
Byte 4								
Byte 5								
Byte 6								
Byte 7								
Byte 8								

[3] Manufacturer ID is allocated by Water Service Department (WSD).

6. Command

A set of application layer commands can be exchanged between the Network Server and the smart water meter for administration. An application layer command consists of a command identifier (A-CID) of 1 byte, an encoding method of 1 byte, a value field of 1 byte, and an empty or a sequence of parameters. The application layer commands defined by this specification are summarized in TABLE 6.1. In addition to these commands, the MAC layer commands specified in [LW103] SHALL be supported by the Network Server and the smart water meter. Note that for the command to support multiple versions of the encoding methods, only one encoding method SHOULD be used in requesting or answering messages, and the encoding method to be used is subject to the consideration of the manufacturers regarding implementation.

6.1 Command Summary

The commands that SHALL be supported are listed in TABLE 6.1.

TABLE 6.1: Commands

	A-CID	Command	Send by meter	Send by Network server	Description
Measurement	0x61	InstantForwardVolumeGetReq		x	Used by the network server to get the meter instant forward volume.
	0x61	InstantForwardVolumeGetAns	x		Used by the meter to answer the meter instant forward volume.
	0x62	InstantBackwardVolumeGetReq		x	Used by the network server to get the meter instant backward volume.
	0x62	InstantBackwardVolumeGetAns	x		Used by the meter to answer the meter instant backward volume.
	0x63	InstantFlowRateGetReq		x	Used by the network server to get the meter instant flow rate.
	0x63	InstantFlowRateGetAns	x		Used by the meter to answer the meter instant flow rate.
	0x64	HalfHourForwardVolumeGetReq		x	Used by the network server to get the meter half hour forward volume.
	0x64	HalfHourForwardVolumeGetAns	x		Used by the meter to answer the meter half hour forward volume.
	0x65	HalfHourBackwardVolumeGetReq		x	Used by the network server to get the meter half hour backward volume.

TABLE 6.1: Commands (Cont'd)

	A-CID	Command	Send by meter	Send by Network server	Description
Measurement	0x65	HalfHourBackwardVolumeGetAns	x		Used by the meter to answer the meter half hour backward volume.
	0x66	MaxMinFlowRateGetReq		x	Used by the network server to get the meter maximum and minimum flow rate.
	0x66	MaxMinFlowRateGetAns	x		Used by the meter to answer the meter maximum and minimum flow rate.
Status	0x67	RemainingBatteryLifeGetReq		x	Used by the network server to get the meter remaining battery life.
	0x67	RemainingBatteryLifeGetAns	x		Used by the meter to answer the meter remaining battery life.
	0x68	MostRecentResetTimeGetReq		x	Used by the network server to get the meter most recent reset time.
	0x68	MostRecentResetTimeGetAns	x		Used by the meter to answer the meter most recent reset time.
	0x69	ResetTimesGetReq		x	Used by the network server to get the meter reset times.
	0x69	ResetTimesGetAns	x		Used by the meter to answer the meter reset times.
	0x6A	TimeCorrectionTimesGetReq		x	Used by the network server to get the meter time correction times.
	0x6A	TimeCorrectionTimesGetAns	x		Used by the meter to answer the meter time correction times.
	0x6B	StatusSummaryGetReq		x	Used by the network server to get the meter time correction times.
	0x6B	StatusSummaryGetAns	x		Used by the meter to answer the meter status summary.
Device info	0x6C	FirmwareVersionGetReq		x	Used by the network server to get the meter firmware version.
	0x6C	FirmwareVersionGetAns	x		Used by the meter to answer the meter firmware version.
	0x6D	ProductionNumberGetReq		x	Used by the network server to get the meter production number.

TABLE 6.1: Commands (Cont'd)

	A-CID	Command	Send by meter	Send by Network server	Description
Device info	0x6D	ProductionNumberGetAns	x		Used by the meter to answer the meter production number.
	0x6E	HardwareVersionGetReq		x	Used by the network server to get the meter hardware version.
	0x6E	HardwareVersionGetAns	x		Used by the meter to answer the meter hardware version.
	0x6F	LoRaWANVersionGetReq		x	Used by the network server to get the LoRaWAN version the meter being used.
	0x6F	LoRaWANVersionGetAns	x		Used by the meter to answer the LoRaWAN version being used.
	0x70	MIUIDGetReq		x	Used by the network server to get the meter's MIU ID.
	0x70	MIUIDGetAns	x		Used by the meter to answer the meter ID.
	0x71	ManufacturerSpecificInfoGetReq		x	Used by the network server to get the meter's Manufacturer-specific Info.
	0x71	ManufacturerSpecificInfoGetAns	x		Used by the meter to answer the Manufacturer-specific Info.
	0x72	MostRecentTimeCorrectionTimeGetReq		x	Used by the network server to get the meter most recent time correction time.
	0x72	MostRecentTimeCorrectionTimeGetAns	x		Used by the meter to answer the meter most recent time correction time

6.2 Command Payload Value Format

1) InstantForwardVolumeGetReq

The command InstantForwardVolumeGetReq payload value format is shown in TABLE 6.2.

TABLE 6.2: Command InstantForwardVolumeGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0 Byte)
InstantForwardVolumeGetReq (0x61)	No encoding (0x05)	0 byte (0x00)	

2) InstantForwardVolumeGetAns

The command InstantForwardVolumeGetAns payload value format is shown in TABLE 6.3.

TABLE 6.3: Command InstantForwardVolumeGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (6 Bytes)
InstantForwardVolumeGetAns (0x61)	BCD encoding (0x00)	6 bytes (0x04)	Data

Volume	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 2	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 3	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 4	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
Time	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (min)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			

Or,

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (6 Bytes)
InstantForwardVolume GetAns (0x61)	Floating-point binary encoding (0x01)	4 bytes (0x04)	Data

31	30	…	23	22	…	0
Sign	Exponent			Fraction		

Time	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			

3) InstantBackwardVolumeGetReq

The command InstantBackwardVolumeGetReq payload value format is shown in TABLE 6.4.

TABLE 6.4: Command InstantBackwardVolumeGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0 Byte)
InstantBackwardVolumeGetReq (0x62)	No encoding (0x05)	0 byte (0x00)	

4) InstantBackwardVolumeGetAns

The command InstantBackwardVolumeGetAns payload value format is shown in TABLE 6.5.

TABLE 6.5: Command InstantBackwardVolumeGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (6 Bytes)
InstantBackwardVolumeGetAns (0x62)	BCD encoding (0x00)	4 bytes (0x04)	Data

Volume	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 2	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 3	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 4	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
Time	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			

Or,

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (6 Bytes)
InstantBackwardVolume GetAns (0x62)	Floating-point binary encoding (0x01)	4 bytes (0x04)	Data

31	30	…	23	22	…	0
Sign	Exponent			Fraction		

Time	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			

5) InstantFlowRateGetReq

The command InstantFlowRateGetReq payload value format is shown in TABLE 6.6.

TABLE 6.6: Command InstantFlowRateGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0 Byte)
InstantFlowRateGetReq (0x63)	No encoding (0x05)	0 byte (0x00)	

6) InstantFlowRateGetAns

The command InstantFlowRateGetAns payload value format is shown in TABLE 6.7.

TABLE 6.7: Command InstantFlowRateGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (6 Bytes)
InstantFlowRateGetAns (0x63)	BCD encoding (0x00)	4 bytes (0x04)	Data

Volume	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 2	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 3	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 4	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
Time	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			

Or,

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (6 Bytes)
InstantFlowRateGetAns (0x63)	Floating-point binary encoding (0x01)	4 bytes (0x04)	Data

31	30	…	23	22	…	0
Sign	Exponent			Fraction		

Time	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			

7) HalfHourForwardVolumeGetReq

The command HalfHourForwardVolumeGetReq payload value format is shown in TABLE 6.8.

TABLE 6.8: Command HalfHourForwardVolumeGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (5 Bytes)
HalfHourForwardVolumeGetReq (0x64)	BCD encoding (0x00)	5 bytes (0x05)	Data

Date and Time	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 3 (day)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 4 (month)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 5 (year)	Ten (BCD encoding)				Unit (BCD encoding)			

8) HalfHourForwardVolumeGetAns

The command HalfHourForwardVolumeGetAns payload value format is shown in TABLE 6.9.

TABLE 6.9: Command HalfHourForwardVolumeGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (6 Bytes)
HalfHourForwardVolume GetAns (0x64)	BCD encoding (0x00)	4 bytes (0x04)	Data

Volume	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 2	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 3	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 4	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
Time	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			

Or,

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (6 Bytes)
HalfHourForwardVolume GetAns (0x64)	Floating-point binary encoding (0x01)	4 bytes (0x04)	Data

31	30	…	23	22	…	0
Sign	Exponent			Fraction		

Time	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			

9) HalfHourBackwardVolumeGetReq

The command HalfHourBackwardVolumeGetReq payload value format is shown in TABLE 6.10.

TABLE 6.10: Command HalfHourBackwardVolumeGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (5 Bytes)
HalfHourBackwardVolume GetReq (0x65)	BCD encoding (0x00)	5 bytes (0x05)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 3 (day)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 4 (month)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 5 (year)	Ten (BCD encoding)				Unit (BCD encoding)			

10) HalfHourBackwardVolumeGetAns

The command HalfHourBackwardVolumeGetAns payload value format is shown in TABLE 6.11.

TABLE 6.11: Command HalfHourBackwardVolumeGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (6 Bytes)
HalfHourBackwardVolume GetAns (0x65)	BCD encoding (0x00)	4 bytes (0x04)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 2	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 3	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 4	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
Time	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			

Or,

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (6 Bytes)
HalfHourBackwardVolume GetAns (0x65)	Floating-point binary encoding (0x01)	4 bytes (0x04)	Data

31	30	…	23	22	…	0
Sign	Exponent			Fraction		

Time	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			

11) MaxMinFlowRateGetReq

The command MaxMinFlowRateGetReq payload value format is shown in TABLE 6.12.

TABLE 6.12: Command MaxMinFlowRateGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0 Byte)
MaxMinFlowRateGetReq (0x66)	No encoding (0x05)	0 byte (0x00)	

12) MaxMinFlowRateGetAns

The command MaxMinFlowRateGetAns payload value format is shown in TABLE 6.13.

TABLE 6.13: Command MaxMinFlowRateBCDGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (12 Bytes)
MaxMinFlowRateGetAns (0x66)	BCD encoding (0x00)	12 bytes (0x0C)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
	Maximum flow rate of past day							
Byte 1	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 2	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 3	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 4	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
	Minimum flow rate of past day							
Byte 5	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 6	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 7	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 8	Ten thousand (BCD encoding)				Thousands (BCD encoding)			
	Time of occurrence of maximum flow rate of past day							
Byte 9 (min)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 10 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			
	Time of occurrence of minimum flow rate of past day							
Byte 11 (min)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 12 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			

Or,

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (12 Bytes)
MaxMinFlowRateGetAns (0x66)	Multiple encoding (0x07)	12 bytes (0x0C)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
	Maximum flow rate							
Byte 1	Fraction							
Byte 2	Fraction							
Byte 3	Exponent	Fraction						
Byte 4	Sign	Exponent						
	Minimum flow rate							
Byte 5	Fraction							
Byte 6	Fraction							
Byte 7	Exponent	Fraction						
Byte 8	Sign	Exponent						
	Time of maximum flow rate							
Byte 9 (min)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 10 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			
	Time of minimum flow rate							
Byte 11 (min)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 12 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			

13) RemainingBatteryLifeGetReq

The command RemainingBatteryLifeGetReq payload value format is shown in TABLE 6.14.

TABLE 6.14: Command MeterRemainingBatteryLifeGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0 Byte)
RemainingBatteryLifeGetReq (0x67)	No encoding (0x05)	0 byte (0x00)	

14) RemainingBatteryLifeGetAns

The command RemainingBatteryLifeGetAns payload value format is shown in TABLE 6.15. In the value field, the remaining battery life percentage is used, and any number greater than 100 (100% of remaining percentage of battery life) is invalid.

TABLE 6.15: Command RemainingBatteryLifeGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (1 Byte)
RemainingBatteryLifeGetAns (0x67)	Binary integer encoding (0x02)	1 byte (0x01)	Data

	Byte format							
Byte 1	0 (reserved)	2^6	2^5	2^4	2^3	2^2	2^1	2^0

15) MostRecentResetTimeGetReq

The command MostRecentResetTimeGetReq payload value format is shown in TABLE 6.16.

TABLE 6.16: Command MostRecentResetTimeGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0 Byte)
MostRecentResetTimeGetReq (0x68)	No encoding (0x05)	0 byte (0x00)	

16) MostRecentResetTimeGetAns

The command MostRecentResetTimeGetAns payload value format are shown in TABLE 6.17. In the value field, only BCD encoding is allowed for this data type.

TABLE 6.17: Command MostRecentResetTimeGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (5 Bytes)
MostRecentResetTimeGetAns (0x68)	BCD encoding (0x00)	5 bytes (0x05)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 3 (day)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 4 (month)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 5 (year)	Ten (BCD encoding)				Unit (BCD encoding)			

17) ResetTimesGetReq

The command ResetTimesGetReq payload value format is shown in TABLE 6.18.

TABLE 6.18: Command ResetTimesGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0 Byte)
ResetTimesGetReq (0x69)	No encoding (0x05)	0 byte (0x00)	

18) ResetTimesGetAns

The command ResetTimesGetAns payload value format is shown in TABLE 6.19. In the value field, only binary integer encoding is allowed for this data type.

TABLE 6.19: Command ResetTimesGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (1 Byte)
ResetTimesGetAns (0x69)	Binary integer encoding (0x02)	1 byte (0x01)	Data

	Byte format							
Byte 1	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

19) TimeCorrectionTimesGetReq

The command TimeCorrectionTimesGetReq payload value format is shown in TABLE 6.20.

TABLE 6.20: Command TimeCorrectionTimesGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0 Byte)
TimeCorrectionTimesGetReq (0x6A)	No encoding (0x05)	0 byte (0x00)	

20) TimeCorrectionTimesGetAns

The command TimeCorrectionTimesGetAns payload value format is shown in TABLE 6.21. In the value field, only the binary integer encoding is allowed for this data type.

TABLE 6.21: Command TimeCorrectionTimesGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (1 Bytes)
TimeCorrectionTimesGetAns (0x6A)	Binary integer encoding (0x02)	1 bytes (0x02)	Data

	Byte format							
Byte 1	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

21) StatusSummaryGetReq

The command StatusSummaryGetReq payload value format is shown in TABLE 6.22.

TABLE 6.22: Command StatusSummaryGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0 Byte)
StatusSummaryGetReq (0x6B)	No encoding (0x05)	0 byte (0x00)	

22) StatusSummaryGetAns

The command StatusSummaryGetAns payload value format is shown in TABLE 6.23. In the value field, only the binary integer encoding is allowed for this data type.

TABLE 6.23: Command StatusSummaryGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (2 Bytes)
StatusSummaryGetAns (0x6B)	Boolean encoding (0x04)	2 bytes (0x02)	Data

	Definition (0-normal, 1-abnormal)							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1	reserved	Tamper	Hardware error	Battery EOL	Battery low	Time reset	Reverse flow	Flow leakage
Byte 2	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved

23) FirmwareVersionGetReq

The command FirmwareVersionGetReq payload value format is shown in TABLE 6.24.

TABLE 6.24: Command FirmwareVersioneGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0 Byte)
FirmwareVersionGetReq (0x6C)	No encoding (0x05)	0 byte (0x00)	

24) FirmwareVersionGetAns

The command FirmwareVersionGetAns payload value format is shown in TABLE 6.25. In the value field, only the ASCII encoding is allowed for this data type.

TABLE 6.25: Command FirmwareVersionGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (8 Bytes)
FirmwareVersionGetAns (0x6C)	ASCII (0x03) or BCD encoding (0x00)	8 bytes (0x08)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1								
Byte 2								
Byte 3								
Byte 4								
Byte 5								
Byte 6								

Byte 7		
Byte 8		

25) ProductionNumberGetReq

The command ProductionNumberGetReq payload value format is shown in TABLE 6.26.

TABLE 6.26: Command ProductionNumbereGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0 Byte)
ProductionNumberGetReq (0x6D)	No encoding (0x05)	0 byte (0x00)	

26) ProductionNumberGetAns

The command ProductionNumberGetAns payload value format is shown in TABLE 6.27. In the value field, only the ASCII encoding is allowed for this data type.

TABLE 6.27: Command ProductionNumberGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (8 Bytes)
ProductionNumberGetAns (0x6D)	ASCII (0x03) or BCD encoding (0x00)	8 bytes (0x08)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1								
Byte 2								
Byte 3								
Byte 4								
Byte 5								
Byte 6								
Byte 7								
Byte 8								

27) HardwareVersionGetReq

The command HardwareVersionGetReq payload value format is shown in TABLE 6.28.

TABLE 6.28: Command HardwareVersioneGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0 Byte)
HardwareVersionGetReq (0x6E)	No encoding (0x05)	0 byte (0x00)	

28) HardwareVersionGetAns

The command HardwareVersionGetAns payload value format is shown in TABLE 6.29. In the value field, only the ASCII encoding is allowed for this data type.

TABLE 6.29: Command HardwareVersionGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (8 Bytes)
HardwareVersionGetAns (0x6E)	ASCII (0x03) or BCD encoding (0x00)	8 bytes (0x08)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1								
Byte 2								
Byte 3								
Byte 4								
Byte 5								
Byte 6								
Byte 7								
Byte 8								

29) LoRaWANVersionGetReq

The command LoRaWANVersionGetReq payload value format is shown in TABLE 6.30.

TABLE 6.30: Command LoRaWANVersioneGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0 Byte)
LoRaWANVersionGetReq (0x6F)	No encoding (0x05)	0 byte (0x00)	

30) LoRaWANVersionGetAns

The command LoRaWANVersionGetAns payload value format is shown in TABLE 6.31. In the value field, only the ASCII encoding is allowed for this data type.

TABLE 6.31: Command LoRaWANVersionGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (8 Bytes)
LoRaWANVersionGetAns (0x6F)	ASCII (0x03) or BCD encoding (0x00)	8 bytes (0x08)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1								
Byte 2								
Byte 3								
Byte 4								
Byte 5								
Byte 6								
Byte 7								
Byte 8								

31) MIUIDGetReq

The command MIUIDGetReq payload value format is shown in TABLE 6.32.

TABLE 6.32: Command MIUIDeGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0 Byte)
MIUIDGetReq (0x70)	No encoding (0x05)	0 byte (0x00)	

32) MIUIDGetAns

The command MIUIDGetAns payload value format is shown in TABLE 6.33. In the value field, only the ASCII encoding is allowed for this data type.

TABLE 6.33: Command MIUIDGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (8 Bytes)
MIUIDGetAns (0x70)	ASCII (0x03) or BCD encoding (0x00)	8 bytes (0x08)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1								
Byte 2								
Byte 3								
Byte 4								
Byte 5								
Byte 6								
Byte 7								
Byte 8								

33) ManufacturerSpecificInfoGetReq

The command ManufacturerSpecificInfoGetReq payload value format is shown in TABLE 6.34.

TABLE 6.34: Command ManufacturerSpecificInfoGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0 Byte)
ManufacturerSpecificInfo GetReq (0x71)	No encoding (0x05)	0 byte (0x00)	

34) ManufacturerSpecificInfoGetAns

The command ManufacturerSpecificInfoGetAns payload value format are shown in TABLE 6.35. In the value field, only BCD encoding is allowed for this data type.

TABLE 6.35: Command ManufacturerSpecificInfoGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (5 Bytes)
ManufacturerSpecificInfo GetAns (0x71)		8 bytes (0x08)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1	Manufacturer ID (Binary integer encoding)							
Byte 2								
Byte 3								
Byte 4								
Byte 5								
Byte 6								
Byte 7								
Byte 8								

35) MostRecentTimeCorrectionTimeGetReq

The command MostRecentTimeCorrectionTimeGetReq payload value format is shown in TABLE 6.36.

TABLE 6.36: Command MostRecentTimeCorrectionTimeGetReq Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0 Byte)
MostRecentTimeCorrectionTimeGetReq (0x72)	No encoding (0x05)	0 byte (0x00)	

36) MostRecentTimeCorrectionTimeGetAns

The command MostRecentTimeCorrectionTimeGetAns payload value format are shown in TABLE 6.37. In the value field, only BCD encoding is allowed for this data type.

TABLE 6.37: Command MostRecentTimeCorrectionTimeGetAns Payload Value Format

Type (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (5 Bytes)
MostRecentTimeCorrectionTimeGetAns (0x72)	BCD encoding (0x00)	5 bytes (0x05)	Data

	Byte format							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 3 (day)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 4 (month)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 5 (year)	Ten (BCD encoding)				Unit (BCD encoding)			

7. Communication Procedure

7.1 Meter Configurations

Every water meter SHALL start with configurations. The configuration parameters that SHALL be supported are listed in TABLE 7.1. The channel and data rate encoding can be referenced in TABLE 4.1 and 4.2.

TABLE 7.1: Configuration Parameters

Parameter	Value range	Default value
UPLINK_CHANNEL	[0, 1, 2, 3, 4, 5, 6, 7]	0
DOWNLINK_CHANNEL	[0, 1, 2, 3, 4, 5, 6, 7]	0
DATA_RATE	[DR0, DR1, DR2, DR3, DR4, DR5, DR6]	DR5
RECEIVE_DELAY1	>= 1s	1s
RECEIVE_DELAY2	>= RECEIVE_DELAY1+1s	RECEIVE_DELAY1+1s
JOIN_ACCEPT_DELAY1	>= 1s	5s
JOIN_ACCEPT_DELAY2	>= 1s	6s
MAX_FCNT_GAP	>= 0	16384
ADR_ACK_LIMIT	>= 0	64
ADR_ACK_DELAY	>= 0	32
ACK_TIMEOUT	>= 1s	2 +/- 1s
DWELL_TIME	>= 0 ms	400ms
TRANSMISSION_POWER	Uplink: <=14dBm, downlink: <= 27dBm	Uplink: 14dBm, downlink: 14dBm
ADR	0 or 1	1
DUTY_CYCLE	[0%-100%]	1%

The value of parameters UPLINK_CHANNEL, DOWNLINK_CHANNEL, and DATA_RATE SHALL be updated real time by the ADR algorithm if ADR is set to 1. The value of parameters RECEIVE_DELAY1, RECEIVE_DELAY2, JOIN_ACCEPT_DELAY1, JOIN_ACCEPT_DELAY2, MAX_FCNT_GAP, ADR_ACK_LIMIT, ADR_ACK_DELAY, and ACK_TIMEOUT SHALL NOT be changed after initialization from the configuration file when the meter is running. The value of parameters DWELL_TIME, TRANSMISSION_POWER, ADR, DUTY_CYCLE can be changed through LoRaWAN MAC commands[LW103].

7.2 Meter Configurations

The water meter SHALL at least support the OTA Activation procedure defined in [LW103] to join the LoRaWAN. A water meter SHALL go through a new join procedure every time it loses the session context information.

The water meter SHALL broadcast the Join-Req message following the parameters shown in TABLE 7.2:

TABLE 7.2: JoinReq Channel List

Modulation	Bandwidth [kHz]	Channel Frequency [MHz]	Data Rate	Duty cycle
LoRa	125	923.20/923.40	DR2 to DR5	< 1%

If a water meter fails its first try to join the LoRaWAN network, it SHOULD try five times at random time within the same day, and change to one try each day if it continues to fail.

If a water meter loses connection with the Network Server, i.e., does not receive the inquiry command for three consecutive times from the Network Server, it SHALL re-join the network using the lowest data rate.

After joining success, a water meter SHALL request the network time using the MAC command DeviceTimeReq [LW103], and SHOULD synchronize the time periodically, e.g., 1 month.

7.3 Water Meter Data Transmission

For a water meter, after joining successfully, the water meter SHALL report the measurement, event, status and command answers to the Network Server over uplink messages defined in [LW103]. Periodic data SHALL be reported in the base period, event data SHALL be reported when the event occurs, and command answer SHALL be replied in response to the command request. The command request from the Network Server SHALL leverage the downlink receive window specified in [LW103]. The channel, data rate, and transmission power used by a water meter for data transmission SHOULD minimize the total water meter power consumption while satisfying given Packet Successful Rate (PSR), and the LoRa transmission parameter allocation algorithm is out of the scope of this standard. In addition, the Network Server SHOULD ask each water meter's local time every week and correct the time over command if the time difference is larger than 1 minute.

7.3.1 Periodic Reporting

TABLE 7.3 lists the water consumption profile that SHOULD be reported periodically in the base period.

TABLE 7.3: Periodic Report Profiles

Water consumption profile	Period base	Report time
4-hour profile	Per 4 hours	Each water meter SHALL generate a time shift between 10 and 230 minutes[4] for reporting in each 4-hour interval until being asked to change by network server. The starting and ending 10 minutes of each period SHALL NOT be used for packet sending.
Maximum and minimum flow rate of past day	Per day	Each water meter SHALL report within two hours of each day, and a time shift between 10 and 110 minutes SHALL be used for reporting of last day's max and min flow rate until being asked to change by Network Server.
Date and time when the message is sent	Per day	Each water meter SHOULD report within two hours of each day, and a time shift between 10 and 110 minutes SHOULD be used for reporting when the message is sent until being asked to change by Network Server.

The packet format of each water consumption profile is shown in FIG. 7.1. Note that the Maximum and Minimum Flow Rate should be sent with the status in one packet. The 4-hour profile reporting is shown in FIG. 7.2. Note that the manufacturer MAY leverage the 4-hour profile reporting for info reporting specific to the manufacturer.

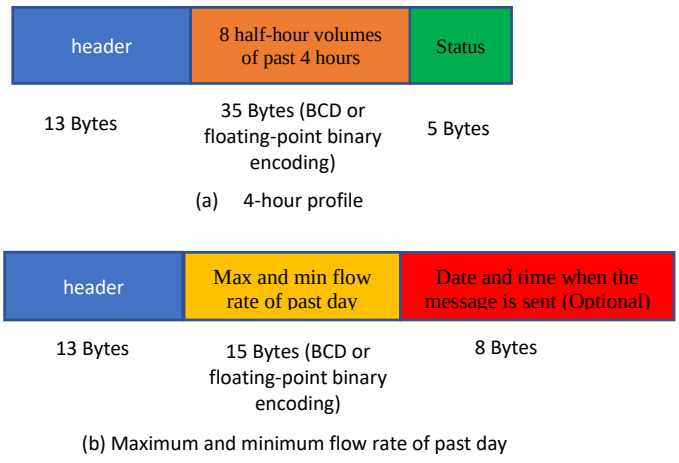

FIG. 7.1 Periodic water consumption profile packet format.

[4] Since the clock in each device is not accurate, the first 10 minutes is not allowed for water consumption profile reporting.

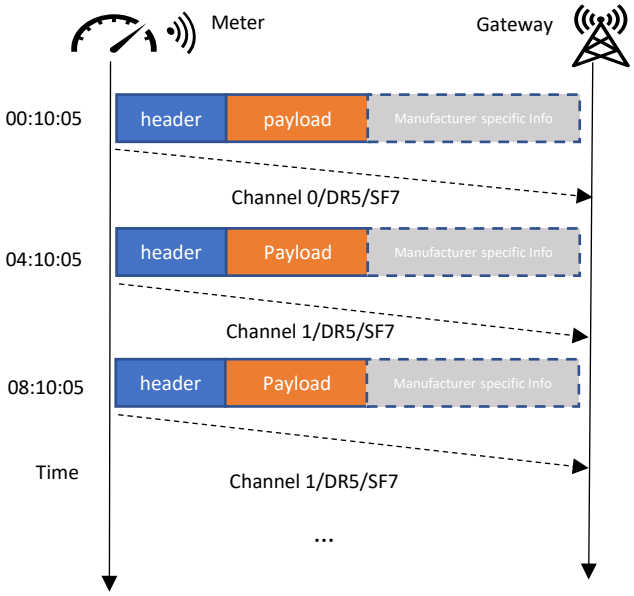

FIG. 7.2 Periodic reporting (4-hour profile).

7.3.2 Event Reporting

When an event is detected by a water meter, the event SHALL be reported over uplink at the earliest possible time to the Network Server through the gateway. The Network Server can ask the water meter for more information by using commands over downlink. The procedure is shown in FIG. 7.3.

The uplink event report packet format can refer to section 5.2.3. The downlink command request packet format is in section 6.

7.3.3 Ask Through Command

Network Server can set/get water meter's measurement data, status, event and device info through commands listed in TABLE 6.1 in section 6. The procedure is shown in FIG. 7.4.

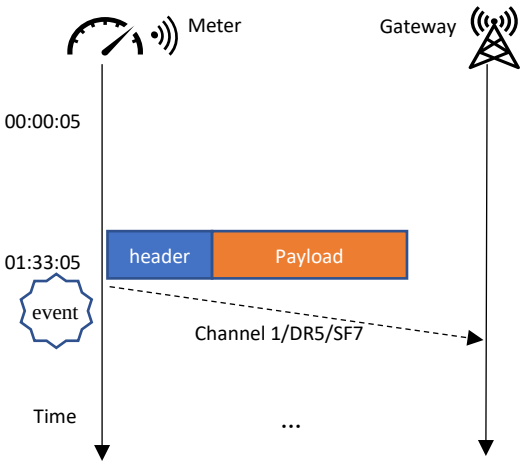

FIG. 7.3 Event reporting.

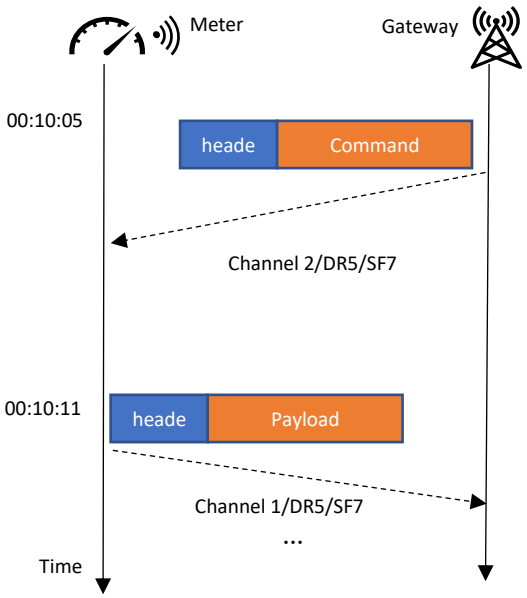

FIG. 7.4 Ask through command.

LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard

Part 4: Event Specification

Revision 1.0.1 / 2022-09

Water Supplies Department (WSD)
Hong Kong Applied Science and Technology Research Institute (ASTRI)
The University of Hong Kong (HKU)

Document Status

Document Name	LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard
Document Code	A-1
Author	• Water Supplies Department (WSD) • Hong Kong Applied Science and Technology Research Institute (ASTRI) • The University of Hong Kong (HKU)
Revision Number	1.0.1
Document Status	Released
Contributors	Water Supplies Department: • Tin Tak CHENG • Chi Lun CHAN • Ming Kwong WONG • Samuel K.S. CHUNG Hong Kong Applied Science and Technology Research Institute: • Dr. Abel Z. YANG • Jianxiong WANG • Carlos C. H. CHIU • Henry K. H. FONG • Hang LIU • Lei ZHU • Dr. Miao Ma The University of Hong Kong: • Prof. Lawrence K. YEUNG • Dr. Vincent W.L. TAM

Document Revision History

Revision	Date issued	Reviewed by	Approved by	Date approved	Revision type
1.0.0	April 2021	WSD	WSD	April 2021	Initial Release
1.0.1	September 2022	WSD	WSD	September 2022	Minor Revision

This Hong Kong Standard was approved by WSD on September 2022.

This Hong Kong Standard exists in one official version (English).

Table of Contents

Foreword

This document (LHKS001-4:2022) has been prepared by Technical Committee formed by Water Supply Department (WSD), Hong Kong Applied Science and Technology Research Institute (ASTRI), and The University of Hong Kong (HKU).

This document shall be adopted as a Hong Kong standard, either by publication of an identical text or by endorsement, at the latest by September 2022, and conflicting Hong Kong standards shall be withdrawn at the latest by September 2022.

It is possible that some of the elements of this documents may be subject to patent rights. WSD, ASTRI and HKU shall not be held responsible for identifying any patent rights.

This document has been prepared under a mandate given to ASTRI by WSD.

LHKS001 comprises of the following parts:

- *Part 1: General*
- *Part 2: System Specification*
- *Part 3: Communication Specification*
- *Part 4: Event Specification*
- *Part 5: Security Specification*
- *Part 6: Data Storage Specification*
- *Part 7: Inspection Specification*

1. Introduction

This document belongs to the LHKS001 series.

This document describes the LoRaWAN-based Wireless Smart Water Metering system structure, function, and performance requirements in a generic way. This is Part 4 of LHKS001.

Additional parts to the series of the standard LHKS001 are:

- *Part 1: General*
- *Part 2: System Specification*
- *Part 3: Communication Specification*
- *Part 5: Security Specification*
- *Part 6: Data Storage Specification*
- *Part 7: Inspection Specification*

Since the realm of water metering is going through a period of rapid change, it is likely that this and other parts of the standard will require amendments soon.

2. Normative References

The following documents, as a whole or in part, are referenced in this document and are indispensable for its applications.

[LHKS001-1] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 1: General, September 2022.

[LHKS001-2] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 2: System Specification, September 2022.

[LHKS001-3] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 3: Communication Specification, September 2022.

[LHKS001-5] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 5: Security Specification, September 2022.

[LHKS001-6] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 6: Data Storage Specification, September 2022.

[LHKS001-7] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 7: Inspection Specification, September 2022.

3. Terms and Conventions

3.1 Terms

Terms used in this specification are listed in TABLE 3.1.

TABLE 3.1: Terms

Term	**Definition**
MIU	Meter Interface Unit
Q1	Minimum Flow Rate
MLFR	Minimum Leak Flow Rate
LFTD	Leak Flow Time Duration
ZCTD	Zero Consumption Time Duration
IRFR	Immediate Reverse Flow Rate
RFTD	Reverse Flow Time Duration
IFFR	Immediate Forward Flow Rate
FFTD	Forward Flow Time Duration
TSP	Tamper Sensing Period
TTD	Tamper Time Duration
CTTD	Clearing Tamper Time Duration

3.2 Conventions

The conventions used in this document are listed below.

SHALL - the use of the word ‘SHALL indicates a mandatory requirement.
SHOULD - the use of the word ‘SHOULD’ indicates a requirement for good practice, which should be implemented whenever possible.
MAY - the use of the word ‘may’ indicates a desirable requirement.

4. Events Definitions and Detections

4.1 Events Introduction

As introduced in [LHKS001-3], the data types are divided into three categories: measurement, status and event, device information. This section covers events only, mainly focusing on their definitions and detections.

Table 4.1 lists all events and their type codes.

TABLE 4.1 Events and Type Codes

Event	**Type Code**
Flow Leakage	0x11
Flow Leakage Clear	0x12
Reverse Flow	0x13
Reverse Flow Clear	0x14
Tamper	0x15
Tamper Clear	0x16
Battery Low	0x17
Battery EOL (end of life)	0x18
Hardware Error	0x19
Hardware Error Clear	0x1A

Note that a status warning variable, i.e., s*tatus_summary*, is defined in [LHKS001-3]. Its payload structure can be found in [LHKS001-3]. Since each bit of the status variable is closely related to the event, its definition is also shown in Table 4.2.

TABLE 4.2 Definition of S*tatus_Summary*

Status	**Definition (0-normal, 1-abnormal)**							
	D7	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
Byte 1	reserved	tamper	Hardware error	Battery EOL	Battery low	Time reset	Reverse flow	Flow leakage
Byte 2	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved

Whenever an event occurs, a warning flag is set, and a packet reporting this event will be immediately sent to the network server. The data structure of every event is described in detail in [LHKS001-3].

Whenever the event is cleared, the warning flag is cleared, and a packet reporting this event clearance will also be immediately sent to the network server [LHKS001-3].

The events are introduced one by one as follows.

4.2 Flow Leakage

4.2.1 Definition

A leakage is most frequently defined as the amount of water which escapes from the pipe network by means other than through a controlled action.

4.2.2 Detection

1) Event Generation

Under normal circumstances without leakage, the residential and commercial properties usually have no water consumption at certain times throughout the whole day. Twenty-four hours (or even longer) continuous flow above the threshold is a strong indicator of flow leakage.

For the purpose of detection, one configuration parameter, i.e., Minimum Leak Flow Rate (MLFR), (a 30-minute average value) is defined. It is the minimum flow rate threshold above which a water meter is considered as non-zero water consumption. The other configuration parameter, i.e., Leak Flow Time Duration (LFTD), is also defined, which is the time duration threshold. Only when the continuous flow exists for a certain time duration, will the flow leakage event be triggered, and will the flow leakage warning flag be set.

Suppose that a water meter reads its water consumption every half an hour. It will check if the condition (Current reading – Previous reading > MLFR) is true for at least LFTD, and then it will confirm flow leakage and set a flow leakage warning flag. Otherwise, no leakage is found.

The default value of MLFR is set as (Q1 * 1 hour), where Q1 denotes the minimum flow rate of a water meter. Besides, the default value of LFTD is set as 7 days (i.e., 168 hours), and the main purpose is to prevent false positive alarm.

2) Event Clearance

A similar method is adopted to clear the leakage warning flag. A configuration parameter, Zero Consumption Time Duration (ZCTD), is defined. It is a time duration threshold with zero water consumption, above which flow leakage no longer exists, and the flow leakage warning flag is cleared.

A water meter reads its water consumption every half an hour. Check if the condition (Current reading – Previous reading ≤ MLFR) is true for at least ZCTD, and then the flow leakage warning flag is cleared and flow leakage no longer exists.

The default value of ZCTD is set as 2 hours.

4.2.3 Configuration Parameters

Three variables are defined, that is, *min_leak_flow* denotes MLFR, *leak_duration* denotes LFTD, and *zero_consume_duration* denotes ZCTD. MLFR supports two types of encoding formats: BCD and floating-point binary, with the length of 2 bytes. The encoding format of LFTD and ZCTD adopts 2-byte binary integer [LHKS001-3]. Table 4.3 shows the configuration parameters for detecting flow leakage.

TABLE 4.3 Configuration Parameters for Flow Leakage Detection

Variable	Meaning	Default value	Encoding format	Length (bytes)	Unit
min_leak_flow	Minimum Leak Flow Rate (MLFR) is the minimum flow rate threshold above which a water meter is considered as non-zero water consumption.	Q1 * 1 hour	BCD Floating-point binary	4	M^3/hour
leak_duration	Leak Flow Time Duration (LFTD) is the time duration threshold with continuous flow.	7 days	BCD	2	day
zero_consume_duration	Zero Consumption Time Duration (ZCTD) is a time duration threshold with zero water consumption, above which the flow leakage warning flag is cleared.	2 hours	BCD	2	hour

4.2.4 Event Packet and Event Log

When flow leakage is detected and when the leakage alarm is cleared, an event packet will be sent immediately to the network server. The payload structure of the event is defined in detail in [LHKS001-3].

An event of leakage detection and leakage alarm clearance will be written into the rotating system log. Refer to [LHKS001-6] for more information.

4.3 Reverse Flow

4.3.1 Definition

Suppose that a water meter reads its water consumption every half an hour. If the current water meter reading is larger than the previous reading from half an hour ago, the flow is forward; and if the current water meter reading is less than the previous reading from half an hour ago, the flow is reverse.

4.3.2 Detection

1) Event Generation

Whenever a reverse flow is found, the amount of reverse flow in the water meter is accumulated over a time period. To prevent false positive, not only is the amount of reverse flow important, but the time duration of reverse flow is also important.

For the purpose of detection, one configuration parameter, i.e., Immediate Reverse Flow Rate (IRFR), is defined. It is a reverse flow rate threshold above which a water meter is immediately confirmed as reverse flow, and the reverse flow warning flag is set. That is, if (|reverse flow| > IRFR), then the reverse flow warning flag is set. The reason for calculating the absolute value of reverse flow is that the reverse flow is negative, so the absolute value of reverse flow is considered.

But if the amount of reverse flow is lower than or equal to IRFR, then the time duration of reverse flow will be . Therefore, the other configuration parameter, i.e., Reverse Flow Time Duration (RFTD), is defined, which is the time duration threshold with a small amount of reverse flow. Only when a small amount of reverse flow lasts for a certain time duration, will the reverse flow event be triggered, and the reverse flow warning flag is set. Put in other words, if both ($0 < |\text{reverse flow}| \leq \text{IRFR}$) and (time duration > RFTD) are true, then the reverse flow warning flag is set.

The default value of IRFR is set as (Q1 * 0.5 hour), and the default value of RFTD is set as 30 minutes.

2) Event Clearance

To clear the reverse flow warning flag, a similar method is used. Again, both the amount of forward flow and the time duration of forward flow are important.

A configuration parameter, i.e., Immediate Forward Flow Rate (IFFR), is defined. It is a forward flow rate threshold above which a water meter is immediately confirmed as forward flow, and the reverse flow warning flag is cleared. Another configuration parameter, i.e., Forward Flow Time Duration (FFTD), is also defined, which is the time duration threshold with a small amount of forward flow, above which the reverse flow warning flag is cleared. The detection works as follows. When the forward flow is found and the amount of forward flow is greater than IFFR, the reverse flow warning flag is immediately cleared. That is, if (forward flow > IFFR), then the reverse flow warning flag is cleared.

Otherwise, if the forward flow exists but the amount of forward flow is smaller than or equal to IFFR, the time duration of the forward flow will be checked. Only when its time duration exceeds FFTD, the reverse flow

warning flag is cleared. That is, if both (0 < forward flow ≤ IFFR) and (time duration > FFTD) are true, then the reverse flow warning flag is cleared.

The default value of IFFR is set as (Q1 * 1 hour), and the default value of FFTD is set as 30 minutes.

4.3.3 Configuration Parameters

Four variables are defined, that is, *imm_reverse_flow* denotes IRFR, *reverse_duration* denotes RFTD, *imm_forward_flow* denotes IFFR, and *forward_duration* denotes FFTD. Both IRFR and IFFR support two types of encoding formats: BCD and floating-point binary, with the same length of 2 bytes. The encoding format of RFTD and FFTD adopts 2-byte binary integer [LHKS001-3]. Table 4.4 shows the configuration parameters for reverse flow detection.

TABLE 4.4 Configuration Parameters for Reverse Flow Detection

Variable	Meaning	Default value	Encoding format	Length (bytes)	Unit
imm_reverse_flow	Immediate Reverse Flow Rate (IRFR) is a reverse flow rate threshold above which a water meter is immediately confirmed as reverse flow.	Q1 * 0.5 hour	BCD Floating-point binary	4	M^3/hour
reverse_duration	Reverse Flow Time Duration (RFTD) is a time duration threshold with a small amount of reverse flow, above which a reverse flow event is triggered.	30 minutes	Binary integer	2	minute
imm_forward_flow	Immediate Forward Flow Rate (IFFR) is a forward flow rate threshold, above which the reverse flow warning flag is immediately cleared.	Q1 * 1 hour	BCD Floating-point binary	4	M^3/hour
forward_duration	Forward Flow Time Duration (FFTD) is a time duration threshold with a small amount of forward flow, above which the reverse flow warning flag is cleared.	30 minutes	Binary integer	2	minute

4.3.4 Event Packet and Event Log

When reverse flow is detected and when reverse alarm is cleared, an event packet will be sent immediately to the network server. The payload structure of the event is defined in detail in [LHKS001-3].

An event of reverse flow detection and reverse alarm clearance will be written into the rotating system log. Refer to [LHKS001-6] for more information.

4.4 Tamper

4.4.1 Definition

Meter tampering is defined as a fraudulent manipulation which may cause incorrect billing of the water consumption by the water supply department.

4.4.2 Detection

1) Event Generation

For the Automatic Meter Reading (AMR) system, a small component, called a Meter Interface Unit (MIU), is installed on the exterior of the water meter. MIU is connected to the water meter and the AMR system can provide half-hour readings.

The MIU module contains sensors for detecting fraudulent manipulation, e.g., magnetic interference. To prevent false positive, only when the tamper status exists for a certain time is the tamper eventtriggered.

To save power, sensors do not operate continuouslybut instead at a regular interval. A configuration parameter, i.e., Tamper Sensing Period (TSP), is defined. It determines how often the sensors detect tamper. Another configuration parameter, i.e., Tamper Time Duration (TTD), is also defined. It is a time threshold above which a tamper event is triggered, and tamper warning flag is set.

The default value of TSP is set as 5 minutes, and the default value of TTD is set as 30 minutes.

2) Event Clearance

To prevent false negative, only when non-tamper status continues for a certain time duration is the tamper warning flag cleared. For this purpose, a configuration parameter, i.e., Clearing Tamper Time Duration (CTTD), is defined. It is a time threshold above which the tamper warning flag is cleared.

The default value of CTTD is set as 30 minutes.

4.4.3 Configuration Parameters

Three variables are defined, that is, *tamper_sensing_period* denotes TSP, *tamper_duration* denotes TTD, and *no_tamper_duration* denotes CTTD. The encoding format of TSP, TTD and CTTD adopts 2-byte binary integer [LHKS001-3]. Table 4.5 shows configuration parameters for detecting tamper or removal.

TABLE 4.5 Configuration Parameters for Tamper Detection

Variable	Meaning	Default value	Encoding format	Length (bytes)	Unit
tamper_sensing_period	Tamper Sensing Period (TSP) determines how often the sensors detect tamper.	5 minutes	Binary integer	2	Minute
tamper_duration	Tamper Time Duration (TTD) a time threshold above which a tamper event is triggered.	30 minutes	Binary integer	2	Minute
no_tamper_duration	Clearing Tamper Time Duration (CTTD) is a time threshold above which the tamper warning flag is cleared.	30 minutes	Binary integer	2	Minute

4.4.4 Event Packet and Event Log

When a tamper is detected and when the tamper alarm is cleared, an event packet will be sent immediately to the network server. The payload structure of the event is defined in detail in [LHKS001-3].

An event of tamper detection and the clearance of the tamper alarm will be written into the rotating system log. Refer to [LHKS001-6] for more information.

4.5 Battery Low

If the current battery life is less than one year, the battery is detected low, and the battery low warning flag is set.

When battery low is detected, an event packet will be sent immediately to the network server. The payload structure of the event is defined in detail in [LHKS001-3].

An event of battery low will be written into the rotating system log. Refer to [LHKS001-6] for more information.

4.6 Battery EOL (End-of-Life)

If the current battery life is less than one month, battery end-of-life (EOL) is detected, and the batter EOL warning flag is set.

When battery end-of-life is detected, an event packet will be sent immediately to the network server. The payload structure of the event is defined in detail in [LHKS001-3].

An event of battery EOL will be written into the rotating system log. Refer to [LHKS001-6] for more information.

4.7 Hardware Error

If any hardware error occurs, the hardware error is detected, and the hardware error warning flag is set. In this situation, the device SHOULD be replaced.

When the hardware error is detected and has been corrected, an event packet will be sent immediately to the network server. The payload structure of the event is defined in detail in [LHKS001-3].

An event of hardware error detection and hardware error clearance will be written into the rotating system log. Refer to [LHKS001-6] for more information.

4.8 Events Summary

TABLE 4.6 summarizes all the detected events. TABLE 4.7 summarizes all the default values of the parameters.

TABLE 4.6 Events Detection Table

Event	Detection	Duration	Status
Flow Leakage	current reading – previous reading > MLFR	LFTD	Alarm
	current reading – previous reading ≤ MLFR	ZCTD	Clear
Reverse Flow	|reverse flow| > IRFR	immediately	Alarm
	0 < |reverse flow| ≤ IRFR	RFTD	
	forward flow > IFFR	immediately	Clear
	0 < forward flow ≤ IFFR	FFTD	
Tamper	detect tamper	TTD	Alarm
	detect no tamper	CTTD	Clear
Battery Low	Current battery life < 1 year	immediately	Alarm
Battery EOL	Current battery life < 1 month	immediately	Alarm
Hardware Error	Find a hardware error	immediately	Alarm
	The hardware error has been corrected	immediately	Clear

TABLE 4.7 Default Values of Parameters

Term	Default Value
Q1	Minimum Flow Rate
MLFR	Q1 * 1 hour
LFTD	7 days
ZCTD	2 hours
IRFR	Q1 * 0.5 hour
RFTD	30 minutes
IFFR	Q1 * 1 hour
FFTD	30 minutes
TSP	5 minutes
TTD	30 minutes
CTTD	30 minutes

5. Commands

5.1 Commands Summary

Table 5.1 summarizes all the commands related to the events parameters setting. As introduced in [LHKS001-3], an application layer command consists of a command identifier (A-CID), which is 1-byte long.

TABLE 5.1 Commands of Parameter Setting

Contents of Command	A-CID	Command name	Send by meter	Send by Network server	Description
Set flow leakage setting	0xA0	SetLeakage		x	Set the parameters of flow leakage.
Ask flow leakage setting	0xA1	AskLeakage		x	Ask the parameters of flow leakage.
Answer flow leakage setting	0xA2	AnsLeakage	x		Answer the parameters of flow leakage.
Set reverse flow setting	0xA3	SetReverse		x	Set the parameters of reverse flow.
Ask reverse flow setting	0xA4	AskReverse		x	Ask the parameter of reverse flow.
Answer reverse flow setting	0xA5	AnsReverse	x		Answer the parameters of reverse flow.
Set tamper setting	0xA6	SetTamper		x	Set the parameters of tamper setting.
Ask tamper setting	0xA7	AskTamper		x	Ask the parameters of tamper setting.
Answer tamper setting	0xA8	AnsTamper	x		Answer the parameters of tamper setting.

5.2 Commands Payload Structure and Value Format

This section introduces the payload structure and value format for each command.

1) Set Flow Leakage Setting

For the flow leakage setting, the format of minimum leak flow rate (MLFR) adopts either BCD encoding or floating-point binary encoding [LHKS001-3].

When MLFR adopts BCD encoding, the encoding field is set as BCD “0x00”; or when MLFR adopts float binary encoding, the encoding field is set as multiple encoding “0x07” since the format of time duration always adopts BCD encoding [LHKS001-3]. Table 5.2 shows the payload structure.

TABLE 5.2: Payload Structure of Set Flow Leakage Setting

A-CID (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (10 bytes)		
			4 bytes	3 bytes	3 bytes
Set flow leakage setting (0xA0)	BCD (0x00) or Multiple (0x07)	10	*min_leak_flow*	*leak_duration*	*zero_consume_duration*

Firstly, BCD encoding (defined for flow rate [LHKS001-3]) for *min_leak_flow* is shown in table 5.3.

TABLE 5.3 BCD Encoding for Flow Rate

	4 bytes							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1	Percentile (BCD encoding)				Thousandths (BCD encoding)			
Byte 2	Unit (BCD encoding)				Tenths (BCD encoding)			
Byte 3	Hundreds (BCD encoding)				Ten (BCD encoding)			
Byte 4	Ten thousand (BCD encoding)				Thousands (BCD encoding)			

Next, the float-point encoding (defined for flow rate [LHKS001-3]) for *min_leak_flow* is shown in table 5.4.

TABLE 5.4 Float-point Encoding for Flow Rate

4 bytes																															
31	30	…							23	22	…																				0
Sign	Exponent									Fraction																					

Lastly, BCD encoding (defined for time duration [LHKS001-3]) for both *leak_duration* and *zero_consume_duration* is shown in table 5.5.

TABLE 5.5 BCD Encoding for Time Duration

	3 bytes							
	D7	D6	D5	D4	D3	D2	D1	D0
Byte 1 (minute)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 2 (hour)	Ten (BCD encoding)				Unit (BCD encoding)			
Byte 3 (day)	Ten (BCD encoding)				Unit (BCD encoding)			

2) Ask Flow Leakage Setting

The command of asking the flow leakage setting is shown in Table 5.6.

TABLE 5.6: Payload Structure: Ask Flow Leakage Setting

A-CID (1 Byte)	Encoding (1 Byte)	Length (1 Byte)
Ask flow leakage setting (0xA1)	No encoding (0x05)	0

3) Answer Flow Leakage Setting

Similarly, the format of minimum leak flow rate (MLFR) adopts either BCD encoding or floating-point binary encoding [LHKS001-3], and the format of time duration adopts BCD encoding [LHKS001-3].

TABLE 5.7 Payload Structure: Answer Flow Leakage Setting

A-CID (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (10 bytes)		
			4 bytes	3 bytes	3 bytes
Answer flow leakage setting (0xA2)	BCD (0x00) or Multiple (0x07)	10	*min_leak_flow*	*leak_duration*	*zero_consume_duration*

4) Set Reverse Flow Setting

For the reverse flow setting, the formats of both immediate reverse flow rate (IRFR) and immediate forward flow rate (IFFR) adopt either BCD encoding or floating-point binary encoding [LHKS001-3].

When IRFR and IFFR adopt BCD encoding, encoding field is set as BCD "0x00"; or when IRFR and IFFR adopt floating-point binary encoding, encoding field is set as multiple encoding "0x07", since the format of time duration always adopts BCD encoding [LHKS001-3].

TABLE 5.8 Payload Structure: Set Reverse Flow Setting

A-CID (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (14 bytes)			
			4 bytes	4 bytes	3 bytes	3 bytes
Set reverse flow setting (0xA3)	BCD (0x00) or Multiple (0x07)	14	*imm_reverse_flow*	*imm_forward_flow*	*reverse_duration*	*forward_duration*

BCD encoding for *imm_reverse_flow* and *imm_forward_flow* is shown as in table 5.3.

Floating-point binary encoding for *imm_reverse_flow* and *imm_forward_flow* is shown in table 5.4.

The formats of both *reverse_duration* and *forward_duration* are shown table 5.5.

5) Ask Reverse Flow Setting

The command of asking the reverse flow setting is shown in Table 5.9.

TABLE 5.9: Payload structure: Ask Reverse Flow Setting

A-CID (1 Byte)	Encoding (1 Byte)	Length (1 Byte)
Ask reverse flow setting (0xA4)	No encoding (0x05)	0

6) Answer Reverse Flow Setting

For the reverse flow setting, the formats of both immediate reverse flow rate (IRFR) and immediate forward flow rate (IFFR) adopt either BCD encoding or floating-point binary encoding [LHKS001-3], and the format of time duration adopts BCD [LHKS001-3].

TABLE 5.10 Payload structure: Answer Reverse Flow Setting

A-CID (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (14 bytes)			
			4 bytes	4 bytes	3 bytes	3 bytes
Answer reverse flow setting (0xA5)	BCD (0x00) or Multiple (0x07)	14	*imm_reverse_flow*	*imm_forward_flow*	*reverse_duration*	*forward_duration*

7) Set Tamper Setting

For the tamper setting, the formats of both tamper sensing period and the time duration adopt BCD encoding [LHKS001-3].

TABLE 5.11 Payload Structure: Set Tamper Setting

A-CID (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (9 bytes)		
			3 bytes	3 bytes	3 bytes
Set tamper setting (0xA6)	BCD (0x00)	9	*tamper_sensing_period*	*tamper_duration*	*no_tamper_duration*

For *tamper_sensing_period*, *tamper_duration* and *no_tamper_duration*, their formats are shown in table 5.5.

8) Ask Tamper Setting

The command of asking the tamper setting is shown in Table 5.12.

TABLE 5.12: Payload Structure: Ask Tamper Setting

A-CID (1 Byte)	Encoding (1 Byte)	Length (1 Byte)
Ask tamper setting (0xA7)	No encoding (0x05)	0

9) Answer Tamper Setting

For tamper setting, the formats of both tamper sensing period and the time duration adopt BCD encoding [LHKS001-3].

TABLE 5.13 Payload Structure: Answer Tamper Setting

A-CID (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (9 bytes)		
			3 bytes	3 bytes	3 bytes
Answer tamper setting (0xA8)	BCD (0x00)	9	*tamper_sensing_period*	*tamper_duration*	*no_tamper_duration*

LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard

Part 5: Security Specification

Revision 1.0.1 / 2022-09

Water Supplies Department (WSD)

Hong Kong Applied Science and Technology Research Institute (ASTRI)

The University of Hong Kong (HKU)

Document Status

Document Name	LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard
Document Code	A-1
Author	• Water Supplies Department (WSD) • Hong Kong Applied Science and Technology Research Institute (ASTRI) • The University of Hong Kong (HKU)
Revision Number	1.0.1
Document Status	Released
Contributors	Water Supplies Department: • Tin Tak CHENG • Chi Lun CHAN • Ming Kwong WONG • Samuel K. S. CHUNG Hong Kong Applied Science and Technology Research Institute: • Dr. Abel Z. YANG • Jianxiong WANG • Carlos C. H. CHIU • Henry K. H. FONG • Hang LIU • Lei ZHU • Dr. Miao MA The University of Hong Kong: • Prof. Lawrence K. YEUNG • Dr. Vincent W.L. TAM

Document Revision History

Revision	Date issued	Reviewed by	Approved by	Date approved	Revision type
1.0.0	April 2021	WSD	WSD	April 2021	Initial Release
1.0.1	September 2022	WSD	WSD	September 2022	Minor Revision

This Hong Kong Standard was approved by WSD in September 2022.

This Hong Kong Standard exists in one official version (English).

Table of Contents

Foreword

This document (LHKS001-5:2022) has been prepared by Technical Committee formed by Water Supply Department (WSD), Hong Kong Applied Science and Technology Research Institute (ASTRI), and The University of Hong Kong (HKU).

This document shall be adopted as a Hong Kong standard, either by publication of an identical text or by endorsement, at the latest by September 2022, and conflicting Hong Kong standards shall be withdrawn at the latest by September 2022.

It is possible that some of the elements of this documents may be subject to patent rights. WSD, ASTRI, and HKU shall not be held responsible for identifying any patent rights.

This document has been prepared under a mandate given to ASTRI by WSD.

LHKS001 comprises of the following parts:

- *Part 1: General*
- *Part 2: System Specification*
- *Part 3: Communication Specification*
- *Part 4: Event Specification*
- *Part 5: Security Specification*
- *Part 6: Data Storage Specification*
- *Part 7: Inspection Specification*

1. Introduction

This document belongs to the LHKS001 series.

This document describes the LoRaWAN-based Wireless Smart Water Metering system structure, function, and performance requirements in a generic way. This is Part 5 of LHKS001.

Additional parts to the series of the standard LHKS001 are:

- *Part 1: General*
- *Part 2: System Specification*
- *Part 3: Communication Specification*
- *Part 4: Event Specification*
- *Part 6: Data Storage Specification*
- *Part 7: Inspection Specification*

Since the realm of water metering is going through a period of rapid change, and it is likely that this and other parts of the standard will require amendments soon.

1. Normative References

The following documents, as a whole or in part, are referenced in this document and are indispensable for its applications.

[LHKS001-1] LoRaWAN -based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 1: General, September 2022.

[LHKS001-2] LoRaWAN -based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 2: System Specification, September 2022.

[LHKS001-3] LoRaWAN -based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 3: Communication Specification, September 2022.

[LHKS001-4] LoRaWAN -based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 4: Event Specification, September 2022.

[LHKS001-6] LoRaWAN -based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 6: Data Storage Specification, September 2022.

[LHKS001-7] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 7: Inspection Specification, September 2022.

[LW103] LoRaWAN Specification, Version 1.0.3, LoRa Alliance, July 2018.

[LWRP103] LoRaWAN 1.0.3 Regional Parameters, Revision A, LoRa Alliance, July 2018.

[LW11] LoRaWAN Specification, Version 1.1, LoRa Alliance, October 2017.

[LWReliable] WEBINAR: LoRaWAN - Providing Secure and Reliable Connectivity, LoRa Alliance, 2020. https://www.youtube.com/watch?v=nHcXPUbZfpc&feature=youtu.be, Watch on YouTube.

[LWWhitePaper] LoRaWAN Security Whitepaper, LoRa Alliance, Feb. 2017.

[LWSecure] LoRaWAN Is Secure (but Implementation Matters), LoRa Alliance, 2020. https://lora-alliance.org/resource-hub/lorawanr-secure-implementation-matters.

[LWFAQ] LoRaWAN Security, Frequently Asked Questions, LoRa Alliance, 2020.

[LWBackend10] LoRaWAN Backend Interfaces Specification, Version 1.0, LoRa Alliance, July 2017.

2. Normative References (Cont'd)

The following documents, as a whole or in part, are referenced in this document and are indispensable for its applications.

[IPSecIETF] IP Security Protocol (IPsec), https://datatracker.ietf.org/wg/ipsec/documents/, IETF Documents.

[RFC5246] The Transport Layer Security (TLS) Protocol – Version 1.2, Aug. 2008.

[RFC2616] Hypertext Transfer Protocol – HTTP 1.1, June 1999.

[RFC2818] HTTP Over TLS, May 2000.

3. Terms and Conventions

3.1 Terms

Terms used in this specification are listed in TABLE 3.1.

TABLE 3.1: Terms

Term	Definition
AppKey	Application Key
AppSKey	Application Session Key
NwkSkey	Network Session Key
DevEUI	Device Extended Unique Identifier
AppEUI	Application Extended Unique Identifier
DevAddr	Device Address
AppAddr	Application Address
AES	Advanced Encryption Standard
OTAA	Over-the-Air-Activation
ABP	Activation by Personalization
DevNonce	Device Nonce
AppNonce	Application Nonce
NetID	Network Identifier
FCnt	Frame Counter
CTR	Counter Mode Encryption
CMAC	Crypto-Message Authentication Code
DoS	Denial of Service
ECB	Electronic Code book
EUI	Extended Unique Identifier
HTTPs	Hyper Text Transfer Protocol Secure
IPsec	Internet Protocol Security
LoRaWAN	Long Range Wide Area Network
MIC	Message Integrity Code
OUI	Organizationally Unique Identifier
TLS	Transport Layer Security
VPN	Virtual Private Network

3.2 Conventions

The conventions used in this document are listed below.

SHALL - the use of the word ‘SHALL indicates a mandatory requirement.
SHOULD - the use of the word ‘SHOULD’ indicates a requirement for good practice, which should be implemented whenever possible.
MAY - the use of the word ‘may’ indicates a desirable requirement.

4. Keys, Personalization and Activation

4.1 Root Keys and Session Keys

Each end-device has one root key, which is the 128-bit long AppKey. The root key is also called a master key. AppKey is shared between the end-device and network server. AppKey SHOULD be unique, random, and difficult to guess.

Whenever an end-device joins a network via Over-The-Air Activation (OTAA), the AppKey is used to derive the session keys: Network Session Key (NwkSKey) and Application Session Key (AppSKey). Both AppSK2ey and NwkSKey are 128 bits, and the AES-128 (Advanced Encryption Standard) is adopted.

These two session keys are renewed at each session. That is, AppKey is static while the sessions keys are dynamically generated. All traffic is protected by the session keys. NwkSKey is used by both the end-device and the network server to prove/verify the packets authenticity and integrity. AppSKey is used by both the end-device and the application server to encrypt/decrypt the application payload. AppSKey can be hidden from the network operator, so that the network operator cannot decrypt the application payloads.

4.2 Device Personalization

The process that AppKey is stored in the device and in the network server is called device personalization. This happens after or just during the production of the end-device [LWReliable].

Each end-device needs to be provisioned with DevEUI, a 64-bit number. EUI denotes Extended Unique Identifier. Each end-device also has its secret AppKey.

A network server SHALL be populated with a list of end-devices, each of which has a DevEUI and an AppKey. The network server is responsible for controlling the end-devices on the network. The AppEUI[1] is a global application identifier in the IEEE EUI-64 address space that identifies the ability to process the join request frame [LW103]. The AppEUI is stored in the end-device before the activation procedure is executed.

4.3 Over-The-Air Activation

The activation of an end-device can be achieved in two ways, either via Over-The-Air Activation (OTAA) when an end-device is deployed or reset, or via Activation By Personalization (ABP) in which the two steps of end-device personalization and activation are done in one step.

[1] Note that AppEUI field of the Join-request in LoRaWAN 1.0.3 [LW103] is renamed to JoinEUI field in LoRaWAN 1.1 [LW11]. In [LW11] a join server is responsible for the join procedure. That is, all the functions done by a join server in [LW11] are done by a network server in [LW103].

OTAA is a handshake between the end-device and the network server. Only end-devices can initiate a join procedure. The end-devices initiate the join procedure every time either when it wants to exchange data with a network server, or when it loses the security session with the network server.

The end-device starts the join procedure by sending a join request message to the network server. The network server sends back a join accept message if the end-device is authorized to join the network.

4.3.1 Join Request Frame

The join request frame is shown in Figure 4.1 [LWReliable].

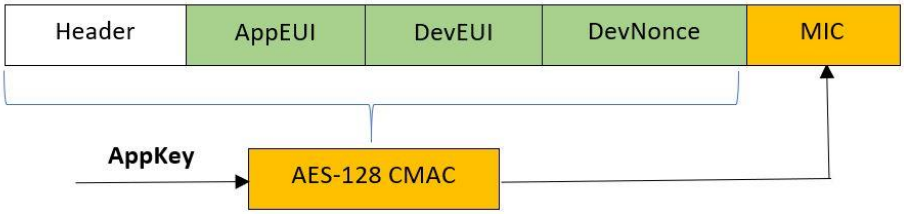

FIGURE 4.1 Join Request Frame

AppEUI is the address of the network server. DevEUI is the identifier of the device. DevNonce is a random value of 2 octets to prevent replay attacks [LW103]. Message Integrity Check (MIC) is calculated by using AES-128 in Cypher-based Message Authentication Code (CMAC) mode with AppKey.

4.3.2 Join Accept Frame

Assuming that the previous frame has been accepted by the network server, meaning that the frame is valid and the Message Integrity Check (MIC) matches, the AppKey matches. Then, the join sever will send an answer, i.e., a join accept frame. The purpose of a join procedure is to make sure that the network server also owns AppKey shared by the end-device.

The frame consists of a AppNonce (i.e., application nonce), a NetID (i.e., network identifier), a DevAddr (i.e., device address) [LW103], and some radio settings. AppNonce is a random value of 3 octets that prevents replay attacks [LW103]. Based on all these contents, MIC is calculated by using AES-128 on CMAC (cypher-based message authentication code) mode, again computed with the AppKey and thenfurther encrypted with the same AppKey. The join accept frame is shown in Figure 4.2 [LWReliable].

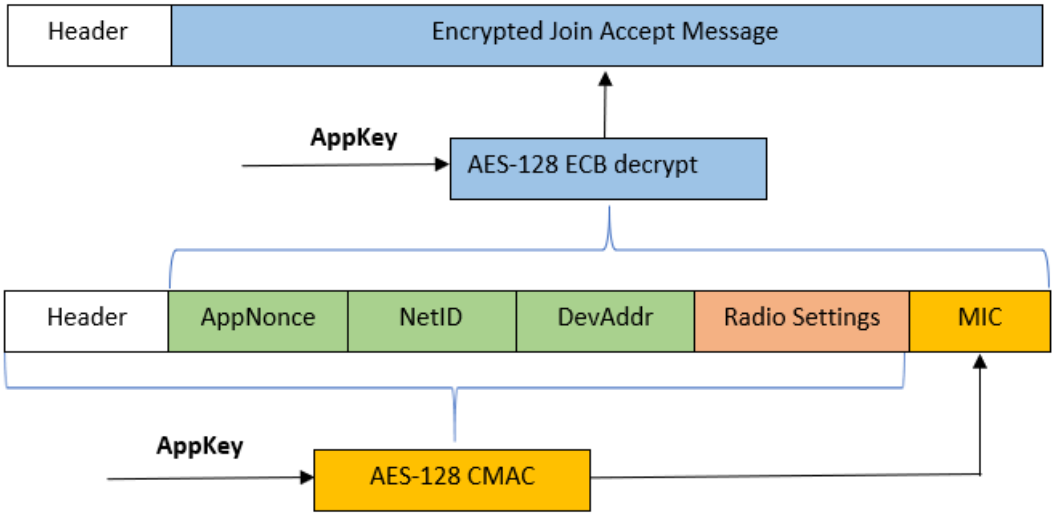

FIGURE 4.2 Join Accept Frame

Upon reception, the end-device will decrypt the message by using AES-128 on ECB (electronic code book) mode with AppKey and then verify the MIC to check if the message is valid.

The network server uses an AES decrypt operation in ECB mode to encrypt the join-accept message, so that the end-device can use an AES encrypt operation to decrypt the message. This way an end-device only has to implement AES encrypt but not AES decrypt [LW103].

Then the device will derive two session keys, NwkKey and AppSkey. Notice that NwkKey and AppSkey are derived using AppKey, NetID, AppNonce and DevNonce. AppNonce is a random value of 3 octets, or some form of unique ID provided by the network server, and DevNonce is a random value of 2 octets received in the join request message [LW103]. Since both AppNonce and DevNonce are included, each new session derives new session keys.

After activation, the following information is stored in each end-device: a device address (DevAddr), an application identifier (AppEUI), a network session key (NwtSKey), an application session key (AppSKey) [LW103].

4.4 Activation by Personalization (ABP)

Activation by personalization, so called ABP, achieves activation and customization in a single step. Activating an end-device by personalization means that the DevAddr and the two session keys, NwkSKey and AppSKey, are directly stored into the end-device instead of the DevEUI, AppEUI and the AppKey [LW103]. The session keys remain the same throughout the lifetime of an ABP end-device.

On one hand, NwkSKey and AppSKey SHOULD be unique, random, and not easily guessable. The reason is that NwkSKey and AppSKey are the only secret that the device will hold for its lifetime.

On the other side, each device SHOULD have a unique set of NwkSKey and AppSKey. Compromising the keys of one device should not compromise the security of the communications of other devices [LW103].

5. Threats, Security Protection and Properties

LoRaWAN basically adopts a security primitive, symmetric cryptography. This choice has been made because symmetric cryptography offers low overhead, and a shorter message is very important for the low power and low-cost devices.

There are three different threats that need to be protected against when security is concerned [LWReliable].

5.1 Threats and Security Protection

5.1.1 Unauthorized Access

To make sure that onlyauthorized devices can access the network, the join procedure uses AES 128 either on CMAC (cypher-based message authentication code) mode or ECB (electronic code book) mode to perform this procedure. Since both the end-device and the network server uses AppKey, such a join procedure offers mutual endpoint authentication between the end device and the network server.

5.1.2 Modification and Spoofing

A Message Integrity Check (MIC) is defined to prevent modification of the data and protect integrity. A MIC (which uses AES-128 on CMAC mode) ensures that the message is complete and has not been altered. As a part of this MIC computation, a device address (identified on a network by DevAddr) which proves that the device is the origin of the message is added. Besides, a frame counter (i.e., FCnt) which protects against replay attacks is added as well. All these security methods protect against modification and spoofing. All security methods happen between the end-device and the network server.

5.1.3 Eavesdropping

In order to prevent eavesdropping between the end-device and the application server, a standard AES-128 in Counter Mode Encryption (CTR) is adopted for encryption. Application payloads are end-to-end encrypted between the end-device and the application server [LW103].

Table 5.1 summarizes all three threats and security protection [LWReliable].

TABLE 5.1 Three types of threats and security protection

Threats	Protecting tools	Security primitives and procedures
Unauthorized Access	Mutual end-point authentication	Join procedure: AES 128 – CMAC AES 128 – ECB
Modification	Integrity protection	MIC message integrity check: • AES 128 – CMAC • Device address (i.e., DevAddr) is part of MIC • Frame counter (i.e., FCnt) is part of MIC
Spoofing	Data origin authentication	
	Replay protection	
Eavesdropping	Data encryption	AES 128 – CTR

5.2 Frame Authentication and Encryption

Frame authentication and encryption happens to **every single frame**. A frame is composed of some link control, a device address (i.e., DevAddr), a frame counter (i.e., FCnt), and Message Integrity Check (MIC), and an encrypted payload. See Figure 5.1 for details [LWReliable].

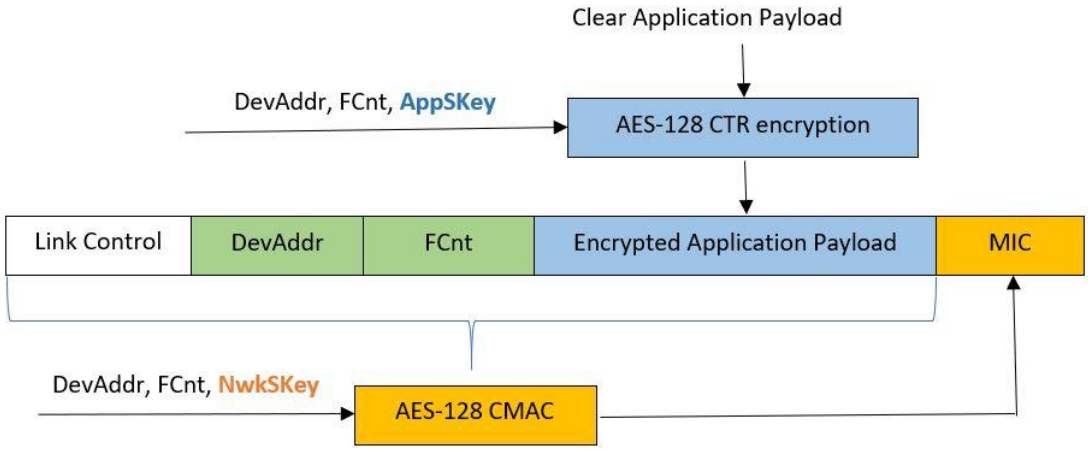

FIGURE 5.1 Frame Structure

The payload from the end-device to the application server is encrypted using AES-128 in Counter Mode Encryption (CTR) mode, and Application Session Key (AppSKey) is used for encryption. Only the application server knows the AppSkey, and therefore, only the application server can decrypt.

Network Session Key (NwkSKey) is used to compute the Message Integrity Check (MIC). If any changes are made to the link control, the device address (i.e., DevAddr), the frame counter (i.e., FCnt), or the encrypted payload, then MIC computation will fail and will not match. Only an entity that has access to NwkSKey can compute and verify the MIC. NwkSKey is shared between the end-device and the network server.

The length of MIC is 32 bits, which means that when a message is changed by an attacker, there is only a probability of

$$p = \frac{1}{2^{32}} = 2.328 * 10^{-10}$$

to have a matching MIC. The probability is extremely small and is considered as relatively impossible. Note that MIC can only be computed and verified with NwkSKey.

5.3 Properties

LoRaWAN security is designed to fit the general LoRaWAN design criteria:

- Low power consumption;
- Low implementation complexity;
- Low cost;
- High scalability.

Three fundamental properties that are supported in LoRaWAN security are:

- Mutual authentication;
- Integrity protection;
- Confidentiality

All these procedures rely on the Advanced Encryption Standard (AES) and use 128-bit cryptographic keys and algorithms.

5.3.1 Mutual Authentication

Table 5.2 summarizes the property of mutual authentication [LWWhitePaper]

TABLE 5.2 Property of Mutual Authentication

Mutual Authentication	
What?	Mutual authentication, also known as two-way authentication, is a security process in which entities authenticate each other before actual communication occurs. Mutual authentication is established between an end-device and the network server. • End device is genuine and authorized. • Network server is genuine and authorized.
How?	Join procedure by Over-the-Air Activation (OTAA). • Both the end-device and the network have the knowledge of AppKey. • Device sends a **join request** with an AES-128 in CMAC (by AppKey). • Network server sends a **join accept** with an AES-128 in CMAC (by AppKey).

5.3.2 Data Integrity

Table 5.3 summarizes the property of data integrity.

TABLE 5.3 Property of Data Integrity

Data Integrity	
What?	Data integrity refers to the reliability and trustworthiness of data during transmission. It also describes the process of preserving the validity and accuracy of data. • The message between end-device and network server is complete and has not been altered.
How?	NwkSKey is adopted for protecting integrity. • NwkSKey is used to compute the Message Integrity Check (MIC) for **every single frame.** • A MIC (which uses AES-128 on CMAC mode) ensures that the message is complete and has not been altered. • NwkSKey is shared only between the end-device and the network server.

5.3.3 Confidentiality

Table 5.3 summarizes the property of confidentiality.

TABLE 5.4 Property of Confidentiality

Confidentiality	
What?	When messages are sent between an end-device and an application server, every single frame is encrypted. Only the application server and the end-device have the key to encrypt/decrypt it. • AppSKey is shared between an end-device and an application server. • AppSKey is hidden from the network operator.Therefore, the network operator is not able to decrypt the application payload.
How?	AppSkey is adopted to provide end-to-end encryption of application payload. • Encryption applies to **every single frame.** • The payload from the end-device to the application server is encrypted using AES-128 in CTR mode, and AppSKey is used for encryption. • Only the application server knows AppSKey. Therefore, only the application server can decrypt it.

6. Implementation Issues

LoRaWAN is secured by design in terms of authentication and encryption. But implementation matters [LWReliable] [LWSecure][LWFAQ].

6.1 Common Security Concerns

Below is a list of some bad but common practices that users SHOULD avoid.

1) The same AppKey is used for multiple end devices.
2) An AppKey is easy to guess.
3) Fixed the security session by using ABP, rather than OTAA, which is more secured.
4) Reusing nonces (i.e., DevNonce and AppNonce) for device activation or reusing frame counters (i.e., FCnt) within a security session.
5) AppKey is not communicated in a secured way from device makers to distributors, or to device owners.

6.2 Follow the Specifications

At implementation, users SHOULD always follow these specifications [LWFAQ].

Each end-device is identified by a 64-bit globally unique identifier, DevEUI, which is assigned either by the manufacturer or the owner of the end-device. Allocation of DevEUI identifiers requires the assignor to have an Organizationally Unique Identifier (OUI) from the IEEE Registration Authority.

Each network server, which is used for authenticating the end-devices, is also identified by a 64-bit globally unique identifier, AppEUI, which is assigned by either the owner or the operator.

Open LoRaWAN networks and private LoRaWAN networks that are collaborating (roaming) with the open networks are identified by a 24-bit globally unique identifier, NetID, assigned by the LoRa Alliance.

When an end-device successfully joins a network, it gets a 32-bit ephemeral device address, DevAddr, assigned by the serving network.

To summarize, users SHOULD always use the assigned identifiers, rather than random ones:

- DevEUI: Follow IEEE OUI assignment,
- AppEUI: Follow IEEE OUI assignment,
- NetID: Follow LoRa Alliance assignment,
- DevAddr: Follow NetID-based assignment by the network server.

6.3 Use OTAA Rather Than ABP

OTAA SHOULD be preferred over ABP for end-devices that require higher levels of security.

AppKey is provisioned by OTAA end-devices. It is used to derive session keys when the OTAA end-device executes a join procedure with the network. Session keys are used by the end-devices to protect the over-the-air traffic. Each new session derives new session keys.

ABP end-devices are not provisioned with the AppKey. Instead, they are provisioned with the DevAddr and the two session keys NwkSKey and AppSKey. The session keys remain the same throughout the lifetime of an ABP end-device.

6.4 Trigger A New Join Only When Necessary

As long as the device does not run out of the frame counters (i.e., FCnt), an end-device can maintain its ongoing session. Since the frame counters are 32 bits, it is not practical for them to run out. Do not reset the frame counters unless a new security session is triggered.

However, there can be times that the end-device loses connectivity and fails to receive downlinks or acknowledgements to its uplinks. Eventually, it may cause a new join, which will trigger a new session. These are the basic situations where a new join and a new session need to be established.

7. Security Issues (not covered by LoRaWAN)

There are some security aspects that are not covered by LoRaWAN [LWReliable].

7.1 Appkey Storage

An end-device has AppKey. And once the AppKey is in place, it SHOULD be security stored against various attacks on the end-device. It SHOULD be practically impossible to extract the keys by any unauthorized entity, because if AppKey is extracted, anyone can impersonate the device and steal the session.

For the device manufacturers, if they have not fully erased the keys, then they SHOULD fully secure their infrastructure.

The network server is a server holding hundreds of keys. A network server SHOULD be protected against any source of attack from the internet, properly securely behind the firewalls with proper access control to prevent unauthorized access.

7.2 Appkey Generation and Delivery

7.2.1 Appkey Generation

The first issue is to generate AppKey for each end-device. The generation is done by the device manufacturer and have to make sure that AppKey is a unique per device. Sharing a same key across multiple devices will lead to a security disaster. That is, if one end device is compromised, then all the other end devices will be compromised as well.

Secondly, the keys SHOULD be cryptographically strong, meaning that they should be hard to guess. Using easy to guess key is a disaster for security, since it basically can break the overall security of the system.

7.2.2 Appkey Delivery

On the network side, after a device is initialized with its AppKey, a network server also needs to have a copy of the same key.

The key delivery SHOULD be properly secured. If it is shared over a file, the file SHOULD be encrypted ideally by the public key of the network server. And if it is shared through application programming interface (API), it SHOULD have extra integrative protection of privacy using Internet Protocol Security (IPsec) [IPSecIETF] or Transport Layer Security (TLS) [RFC5246]. Any compromise in transit will lead the compromise of the overall security.

7.3 Gateway

There are gateways in the middle which help forwarding and receiving packets between the end devices and a network server. A gateway does not play a security role. All of the cryptographic procedures are run at end-to-end basis, which is between an end-device

and the network server. Compromising a gateway would not lead to a compromise of a session.

But in order to prevent any service disruption, the gateway SHOULD be properly secured against any Denial of Service (DoS) attack or any unauthorized access, because failing to do so can cause the gateway offline and can interrupt the traffic.

7.4 Backend Infrastructure and Communication Security

The network architecture includes gateway, network server and application server. These are all infrastructure elements in the architecture. They SHOULD be properly secured using backends in front and communication security [LWBackend10].

Among themselves, they communicate with each other over the Internet, most of the time using public internet, then they SHOULD be properly secured using Internet Protocol Security (IPsec) [IPSecIETF] and Transport Layer Security (TLS) [RFC5246]. Users SHOULD always implement the secured Internet Protocol (IP) links between all the networks elements and the gateways.

The backend interfaces involve control and data signalling among network and application servers. Hyper Text Transfer Protocol Secure (HTTPS) [RFC2616] [RFC2818] and Virtual Private Network (VPN) technologies are used for securing the communication among these critical infrastructure elements. These infrastructure elements SHOULD be properly and securely behind the firewalls, have proper access control, cryptographic security to prevent unauthorized access, and have physical security when applicable.

LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard

Part 6: Data Storage Specification

Revision 1.0.1 / 2022-09

Water Supplies Department (WSD)
Hong Kong Applied Science and Technology Research Institute (ASTRI)
The University of Hong Kong (HKU)

ASTRI

Document Status

Document Name	LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard
Document Code	A-1
Author	• Water Supplies Department (WSD) • Hong Kong Applied Science and Technology Research Institute (ASTRI) • The University of Hong Kong (HKU)
Revision Number	1.0.1
Document Status	Released
Contributors	Water Supplies Department: • Tin Tak CHENG • Chi Lun CHAN • Ming Kwong WONG • Samuel K. S. CHUNG Hong Kong Applied Science and Technology Research Institute: • Dr. Abel Z. YANG • Jianxiong WANG • Carlos C. H. CHIU • Henry K. H. FONG • Hang LIU • Lei ZHU • Dr. Miao MA The University of Hong Kong: • Prof. Lawrence K. YEUNG • Dr. Vincent W.L. TAM

Document Revision History

Revision	Date issued	Reviewed by	Approved by	Date approved	Revision type
1.0.0	April 2021	WSD	WSD	April 2021	Initial Release
1.0.1	September 2022	WSD	WSD	September 2022	Minor Revision

This Hong Kong Standard was approved by WSD in September 2022.

This Hong Kong Standard exists in one official version (English).

Table of Contents

Foreword

This document (LHKS001-6:2022) has been prepared by Technical Committee formed by Water Supply Department (WSD), Hong Kong Applied Science and Technology Research Institute (ASTRI), and The University of Hong Kong (HKU).

This document shall be adopted as a Hong Kong standard, either by publication of an identical text or by endorsement, at the latest by September 2022, and conflicting Hong Kong standards shall be withdrawn at the latest by September 2022.

It is possible that some of the elements of this documents may be subject to patent rights. WSD, ASTRI, and HKU shall not be held responsible for identifying any patent rights.

This document has been prepared under a mandate given to ASTRI by WSD.

LHKS001 comprises of the following parts:

- *Part 1: General*
- *Part 2: System Specification*
- *Part 3: Communication Specification*
- *Part 4: Event Specification*
- *Part 5: Security Specification*
- *Part 6: Data Storage Specification*
- *Part 7: Inspection Specification*

1. Introduction

This document belongs to the LHKS001 series.

This document describes the LoRaWAN-based Wireless Smart Water Metering system structure, function, and performance requirements in a generic way. This is Part 6 of LHKS001.

Additional parts to the series of the standard LHKS001 are:

- *Part 1: General*
- *Part 2: System Specification*
- *Part 3: Communication Specification*
- *Part 4: Event Specification*
- *Part 5: Security Specification*
- *Part 7: Inspection Specification*

Since the realm of water metering is going through a period of rapid change, and it is likely that this and other parts of the standard will require amendments soon.

2. Normative References

The following documents, as a whole or in part, are referenced in this document and are indispensable for its applications.

[LHKS001-1] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 1: General, September 2022.

[LHKS001-2] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 2: System Specification, September 2022.

[LHKS001-3] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 3: Communication Specification, September 2022.

[LHKS001-4] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 4: Event Specification, September 2022.

[LHKS001-5] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 5: Security Specification, September 2022.

[LHKS001-7] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 – Part 7: Inspection Specification, September 2022.

3. Terms and Conventions

3.1 Terms

Terms used in this specification are listed in TABLE 3.1.

TABLE 3.1: Terms

Term	**Definition**
MIU	Meter Interface Unit
Q1	Minimum Flow Rate

3.2 Conventions

The conventions used in this document are listed below.

SHALL - the use of the word ‘SHALL indicates a mandatory requirement.
SHOULD - the use of the word ‘SHOULD’ indicates a requirement for good practice, which should be implemented whenever possible.
MAY - the use of the word ‘may’ indicates a desirable requirement.

4. Data Storage

The Meter Interface Unit (MIU) has a memory space to store data, which has three different types:

- Historical meter data, including volume and flow rate.
- Event log.
- Configuration variables.

In more details, the data stored in memory is shown as follows:

- Half hour volume data SHOULD be available for the last two months.
  - 8 historical volume data per 4 hours.
  - 2880 historical volume data (i.e., 60 days) in total.

- Max and min daily flow rate data SHOULD be available for the last two months.
  - 2 flow rate data per day.
  - 120 historical flow rate data (i.e., 60 days) in total.

- The event log stores the last 100 events.

- Configuration variables including:
  - Configuration parameters of events.
  - Status variables.
  - Device information variables.

5. Data Logging

Historical data is periodically logged in memory by a rotating buffer queue. For each type of data, a queue data structure is adopted. Therefore, there are two queues:

- Half-hour volume queue.
- Daily flow rate queue.

5.1 Half-hour Volume Logging

The payload structures of half-hour volume and date areintroduced in [LHKS001-3]. TABLE 5.1 gives an example of how to store half-hour volume in memory. Other formats are allowed as long as the half-hour volume data can be easily found and read out.

TABLE 5.1 An example: half-hour volumes in memory

Type	**Length (bytes/day)**	**Description**
Date	3	Date (year, month, day)
Time	12	Time (hour, minute) 2 bytes every four hours
Half-hour volume	192	Half-hour volume. 32 bytes every 4 hours
12420 bytes in total for 2 months (or 60 days) historical data		

Two commands are defined: one is to read the half-hour volume and the other is to return it. To read half-hour volume, the command will specify a specific date (i.e., year, month, date) and time (i.e., hour, minute).

Recall that water meter sends half-hour volumes every four hours. A whole day is divided into 6 ranges, starting at 00:00 and each range has 4 hours. Once the time (hour, minute) is specified, a 4-hour range is determined. The water meter will return eight half-hour volumes of that 4-hour range.

The commands are summarized as follows:

- Reading half-hour volumes by specifying date and time is done with the command *VolumeLogRead*.
- Returning half-hour volumes is done with the command *VolumeLogAns*.

5.2 Daily Flow Rate Logging

The payload structures of max and min daily flow rates are introduced in [LHKS001-3]. TABLE 5.2 gives an example of how to store daily flow rate in memory. Other formats are allowed as long as the daily flow rate data can be easily found and read out.

TABLE 5.2 An example: daily flow rates in memory

Type	Length (bytes/day)	Description
Date	3	Date (i.e., year, month, day).
Flow rate	4	Max daily flow rate.
Flow rate	4	Min daily flow rate.
Time	2	Time of max daily flow rate (i.e., hour, minute).
Time	2	Time of min daily flow rate (i.e., hour, minute).
900 bytes in total for 2 months (or 60 days) historical data		

Two commands are defined: one is to read the daily flow rate and the other is to return it. To read the data flow rate, the command will specify a date (i.e., year, month, date). To return daily flow rates, the max and min flow rates and their corresponding time of occurrence are returned.

The commands are summarized as follows:

- Reading max and min daily flow rate by specifying the date is done with the command *FlowRateLogRead.*
- Returning max and min daily flow rate is done with the command *FlowRateLogAns*.

6. Event Logging

All the events are placed into a rotating log, and the log can overwrite previous records. The payload structures of event type, date, and time are introduced in [LHKS001-3]. TABLE 6.1 gives an example of how to store events in memory. Other formats are allowed as long as the event logs can be easily found and read out.

TABLE 6.1 An example: event logs in memory

Type	Length (bytes/event)	Description
Event type	1	Event type.
Event time	5	Date and Time of an event. (i.e., year, month, day, hour, and minute).
600 bytes in total for 100 events		

The memory stores the last 100 events. When more than 100 events occur, the new event log will overwrite the oldest record.

Two commands are defined: one is to read an event log and the other is to return an event log. To read an event log, the command will specify the event type and the specific date. They are listed as follows:

- Reading the event logs by specifying the event type and specific date is done with the command *EventLogRead*.
- Returning the event logs is done with the command *EventLogAns*.

7. Variables

7.1 Configuration Variables

Table 7.1 lists all event-related configuration variables and encryption keys.

TABLE 7.1 Configuration Variables

Name	Default	Description
min_leak_flow	Q1 * 1 hour	Minimum Leak Flow Rate (MLFR) is the minimum flow rate threshold above which a water meter is considered as non-zero water consumption.
leak_duration	7 days	Leak Flow Time Duration (LFTD) is the time duration threshold with continuous flow
zero_consume_duration	2 hours	Zero Consumption Time Duration (ZCTD) is a time duration threshold with zero water consumption, above which the flow leakage warning flag is cleared
imm_reverse_flow	Q1 * 0.5 hour	Immediate Reverse Flow Rate (IRFR) is a flow rate threshold above which a water meter is immediately confirmed as reverse flow.
reverse_duration	30 minutes	Reverse Flow Time Duration (RFTD) is a time duration threshold with reverse flow, above which a reverse flow event is triggered
imm_forward_flow	Q1 * 1 hour	Immediate Forward Flow Rate (IFFR) is a forward flow rate threshold, above which the reverse flow warning flag is immediately cleared.
forward_duration	30 minutes	Forward Flow Time Duration (FFTD) is a time duration threshold with forward flow, above which the reverse flow warning flag is cleared
tamper_sensing_period	5 minutes	Tamper Sensing Period (TSP) determines how often the sensors detect tamper
tamper_duration	30 minutes	Tamper Time Duration (TTD) a time threshold above which a tamper event is triggered
no_tamper_duration	30 minutes	Clearing Tamper Time Duration (CTTD) is a time threshold above which the tamper warning flag is cleared
AES128_key	[0x00…]	All encryption keys used with AES-128 packet encryption are stored in a 16-byte (i.e., 128-bit) array. The default values are all set as zero.

7.2 Other Variables

The status-related variables are listed in TABLE 7.2. The device information variables are listed in TABLE 7.3.

TABLE 7.2 Status-related Variables

Name	**Length (Bytes)**	**Description**
status_summary	1	Each bit denotes an event flag.
battery_percentage	1	Percentage of remaining battery life.
latest_reset_time	5	The most recent time to reset the meter.
reset_times	1	Times of resetting the meter.
latest_timecorrection_time	5	The most recent time to correct meter's time.
timecorrection_times	1	Times of correcting meter's time.

TABLE 7.3 Device Information Variables

Name	**Length (Bytes)**	**Description**
firmware_version	8	Version of firmware.
production_number	8	Production number.
hardware_version	4	Version of hardware.
LoRaWAN_version	5	Version of LoRaWAN.
MIU_ID	8	Identifier of MIU.

8. Commands

8.1 Commands Summary

Table 8.1 summarizes the commands related to event logging and memory. As introduced in [LHKS001-4], an application layer command consists of a command identifier (A-CID), which is 1-byte long.

TABLE 8.1 Commands related to event log and memory

Contents of Command	A-CID	Command name	Send by meter	Send by Network server	Description
Event Log	0xD0	EventLogRead		x	Read event logs with specifying the event type and the date.
	0xD1	EventLogAns	x		Return the event logs.
Half-hour Volume Log	0xD2	VolumeLogRead		x	Read half-hour volume for a specific date and time.
	0xD3	VolumeLogAns	x		Return eight related half-hour volumes.
Daily Flow Rate Log	0xD4	FlowRateLogRead		x	Read max & min daily flow rate for a specific date.
	0xD5	FlowRateLogAns	x		Return max & min daily flow rates and their corresponding time of occurrence.

8.2 Commands Payload Structure and Value Format

This section introduces the payload structure and value format for each command.

1) EventLogRead

The network server sends a command to a water meter to request event logs with specifying the event type and the date. The payload value formats of event type and date are introduced in detail in [LHKS001-3].

Note that type, all the fields of A-CID, encoding, and length SHALL use binary integer encoding [LHKS001-3]. Since the field of event type also uses binary integer encoding, encoding field in the payload structure is set as multiple encoding “0x07”.

TABLE 8.2 Payload structure: event log read

A-CID (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (4 bytes)	
			1 byte	3 bytes
EventLogRead (0xD0)	Multiple (0x07)	4	Event type	Date (year, month, date)

2) EventLogAns

A water meter can return more than one event logs. Again, the encoding field in the payload structure is set as multiple encoding “0x07”.

TABLE 8.3 Payload structure: event log answer

A-CID (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (0~48 bytes)			
			1 byte	3 bytes	2 bytes	… …
EventLogAns (0xD1)	Multiple (0x07)	0~48	Event type	Date (year, month, date)	Time (hour, minute)	Time (hour, minute)

Depending on the number of log events, the length is within the range of [0, 48]. When length is equal to 0, it means no event log is found; and when length is equal to 48 (i.e., the payload length is 51-byte), it means a maximum of 22 event logs are found and returned. If more than 22 event logs are found, a water meter will send more than one packets. The payload value formats of event type, the date and time are introduced in detail in [LHKS001-3].

3) VolumeRead

The network server sends a command to a water meter to request half-hour volumes of a specific date and time. The formats of date and time are introduced in detail in [LHKS001-3].

TABLE 8.4 Payload structure: volume log read

A-CID (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (5 bytes)	
			3 bytes	2 bytes
VolumnLogRead (0xD2)	BCD (0x00)	5	Date (year, month, date)	Time (hour, minute)

4) VolumeLogAns

Recall that a whole day is divided into 6 ranges, starting at 00:00 and each range has 4 hours. Any specified time (i.e., hour, minute) will fall within a certain 4-hour range. Eight half-hour volumes of that 4-hour range will be returned. Half-hour volume can adopt either BCD encoding or floating-point binary encoding method [LHKS001-3].

When half-hour volume adopts BCD, encoding field is set as BCD “0x00”; or when half-hour volume adopts float binary, the encoding field is set as multiple encoding “0x07”, since the formats of date and time always adopt BCD encoding [LHKS001-3].

TABLE 8.5 Payload structure: volume log answer

A-CID (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (37 bytes)					
			3 bytes	2 bytes	4 bytes	4 bytes	…	4 bytes
VolumeLog Ans (0xD3)	BCD (0x00) or Multiple (0x07)	37	Date (year, month, date)	Time (hour, minute)	half-hour volume 1	half-hour volume 2	…	half-hour volume 8

5) FlowRateLogRead

The network server sends a command to a water meter to request max and min daily flow rates on a specific date.

TABLE 8.7 Payload structure: daily flow rate Read

A-CID (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (3 bytes)
FlowRateLogRead (0xD4)	BCD (0x00)	3	Date (year, month, date)

6) FlowRateLogAns

The max and min daily flow rates can adopt either BCD encoding or floating-point binary encoding method. When the max and min daily flow rates adopt BCD encoding, the encoding field is set as BCD “0x00”; or when the max and min daily flow rates adopt float binary encoding, the encoding field is set as Multiple encoding “0x07”, since the formats of date and time always adopt BCD encoding [LHKS001-3].

TABLE 8.8 Payload structure: daily flow rate answer

A-CID (1 Byte)	Encoding (1 Byte)	Length (1 Byte)	Value (15 bytes)				
			3 bytes	4 bytes	4 bytes	2 bytes	2 bytes
FlowRateLogAns (0xD5)	BCD (0x00) or Multiple (0x07)	15	Date (year, month, date)	Max flow rate	Min flow rate	Time of max (hour, minute)	Time of min (hour, minute)

LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard

Part 7: Inspection Specification

Revision 1.0.1 / 2022-09

Water Supplies Department (WSD)
Hong Kong Applied Science and Technology Research Institute (ASTRI)
The University of Hong Kong (HKU)

Document Status

Document Name	LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard
Document Code	A-1
Author	• Water Supplies Department (WSD) • Hong Kong Applied Science and Technology Research Institute (ASTRI) • The University of Hong Kong (HKU)
Revision Number	1.0.1
Document Status	Released
Contributors	Water Supplies Department: • Tin Tak CHENG • Chi Lun CHAN • Ming Kwong WONG Hong Kong Applied Science and Technology Research Institute: • Dr. Abel Z. YANG • Jianxiong WANG • Carlos C. H. CHIU • Henry K. H. FONG • Hang LIU • Lei ZHU • Dr. Miao MA The University of Hong Kong: • Prof. Lawrence K. YEUNG • Dr. Vincent W.L. TAM

Document Revision History

Revision	Date issued	Reviewed by	Approved by	Date approved	Revision type
1.0.0	April 2021	WSD	WSD	April 2021	Initial Release
1.0.1	September 2022	WSD	WSD	September 2022	Minor Revision

This Hong Kong Standard was approved by WSD in September 2022.

This Hong Kong Standard exists in one official version (English).

Table of Contents

Foreword

This document (LHKS001-1:2022) has been prepared by Technical Committee formed by Water Supply Department (WSD), Hong Kong Applied Science and Technology Research Institute (ASTRI), and The University of Hong Kong (HKU).

This document shall be adopted as a Hong Kong standard, either by publication of an identical text or by endorsement, at the latest by September 2022, and conflicting Hong Kong standards shall be withdrawn at the latest by September 2022.

It is possible that some of the elements of this documents may be subject to patent rights. WSD, ASTRI, and HKU shall not be held responsible for identifying any patent rights.

This document has been prepared under a mandate given to ASTRI by WSD.

LHKS001 comprises of the following parts:

- *Part 1: General*
- *Part 2: System Specification*
- *Part 3: Communication Specification*
- *Part 4: Event Specification*
- *Part 5: Security Specification*
- *Part 6: Storage Specification*
- *Part 7: Inspection Specification*

1. Introduction

This document belongs to the LHKS001 series.

This document describes the LoRaWAN-based Wireless Smart Water Metering system structure, function, and performance requirements in a generic way. This is Part 7 of LHKS001.

Additional parts to the series of the standard LHKS001 are:

- *Part 1: General*
- *Part 2: System Specification*
- *Part 3: Communication Specification*
- *Part 4: Event Specification*
- *Part 5: Security Specification*
- *Part 6: Storage Specification*

Since the realm of water metering is going through a period of rapid change, it is likely that this and other parts of the standard will require amendments soon.

2. Normative References

The following documents, as a whole or in part, are referenced in this document and are indispensable for its applications.

[LHKS001-1] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 - Part 1: General, September 2022.

[LHKS001-2] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 - Part 2: System Specification, September 2022.

[LHKS001-3] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 - Part 3: Communication Specification, September 2022.

[LHKS001-4] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 - Part 4: Event Specification, September 2022.

[LHKS001-5] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 - Part 5: Security Specification, September 2022.

[LHKS001-6] LoRaWAN-based Hong Kong Wireless Smart Water Metering System Standard Revision 1.0.1 - Part 6: Storage Specification, September 2022.

[EN60529] EN 60529, Degrees of Protection Provided by Enclosure (IP Code).

3. Terms and Conventions

3.1 Terms

Terms used in this specification are listed in TABLE 3.1.

TABLE 3.1: Terms

Term	Definition
SWM	Smart Water Meter
LoRa™	Long Range modulation technique
LoRaWAN™	Long Range network protocol
MIU	Meter Interface Unit
CIU	Communication Interface Unit
MAC	Medium Access Control
ABP	Activation By Personalization
GW	Gateway
NS	Network Server
DCU	Date Concentration Unit
TLS	Transport Layer Security
AS	Application Server
MQTT	MQ Telemetry Transport or Message Queuing Telemetry Transport
HTTP	Hypertext Transfer Protocol
NTP	Network Time Protocol
PLC	Programmable Logic Controller
JS	Join Server
DAS	Data Analysis Server
MS	Management Server
BS	Billing Server
EIRP	Effective Isotropic Radiated Power
OTAA	Over-the-Air Activation
RX1	Receive Window 1
RX2	Receive Window 2
RF	Radio Frequency
FCnt	Frame Count
NSK	Network Session Key
ASK	Application Session Key

3.2 Conventions

The conventions used in this document are listed below.

SHALL - the use of the word ‘SHALL indicates a mandatory requirement.
SHOULD - the use of the word ‘SHOULD’ indicates a requirement for good practice, which should be implemented whenever possible.
MAY - the use of the word ‘may’ indicates a desirable requirement.

4. Inspection methods

4.1 Inspection Conditions

During the Type acceptance test, a simulation test system SHOULD be formed by the system central station and various types of terminals and channels.

At the time of System acceptance test, the hardware equipment and software system of the system master station, various types of terminals and channels SHOULD have been installed and tested. The correspondence table of inspection items and inspection phases is shown in Table 4.1.

Table 4.1: Correspondence Table of Inspection Items and Inspection Phases

No.	Inspection item	Sub item	Type acceptance test	System acceptance test
1	Structure and mechanical	Enclosure and terminal fire resistance test	✓	
2		Vibration test	✓	
3		Waterproof and dustproof	✓	
4	Environmental impact	High temperature	✓	
5		Low temperature	✓	
6		High temperature and high humidity	✓	
7	Appearance	General inspection	✓	
8	Read device information	Firmware version	✓	✓
9		Production number	✓	✓
10		Hardware version	✓	✓
11		LoRaWAN version	✓	✓
12		MIU ID	✓	✓
13	Parameter setting	Set flow leakage Setting	✓	✓
14		Ask flow leakage setting	✓	✓
15		Answer flow leakage Setting	✓	✓
16		Set reverse flow setting	✓	✓
17		Ask reverse flow setting	✓	✓
18		Answer reverse flow setting	✓	✓
19		Set tamper setting	✓	✓
20		Ask tamper setting	✓	✓
21		Answer tamper setting	✓	✓
22	Data acquisition	Instant forward volume	✓	✓
23		Instant backward volume	✓	✓
24		Half-hour volume	✓	✓
25		Instant flow rate	✓	✓
26		8 half-hour volumes of past 4 hours	✓	✓
27		Maximum and minimum flow rate of past day	✓	✓
28	Event generation and clearance	Flow Leakage	✓	✓
29		Reverse Flow	✓	✓
30		Tamper	✓	✓

4.2 Inspection Process

4.2.1 Inspection methods

The function test of MIU and the function verification test of each test item SHOULD be carried out under the test system. A data acquisition test system is composed of water meter, MIU, DCU, and LoRaWAN Server.

After the test is completed, compare the actual test results with the test requirements to determine whether the system requirements are met.

4.2.2 Environmental conditions

Except for the electrostatic discharge immunity test, where the relative humidity SHOULD be 30% to 60%, all tests are carried out under the following atmospheric conditions, namely:

a) Temperature: +15℃～+35℃;
b) Relative humidity: 25%～75%;
c) Atmospheric pressure: 86kPa～108kPa.

During the test period of each project, the atmospheric environmental conditions SHOULD be relatively stable.

4.3 Inspection Item

4.3.1 Structure and Mechanical Inspection

4.3.1.1 General Inspection

When inspecting the appearance and structure, the device under test (DUT) SHALL not have obvious bumps, scratches, cracks and burrs, the coating SHOULD not fall off, the signs and symbols SHOULD be clear and durable, and the wiring SHOULD be sturdy.

4.3.1.2 Enclosure and Terminal Fire Resistance Test

The test is carried out on a dummy unit with non-metallic enclosure, terminals, and related connections. The material used in the DUT SHOULD be the same as that of the actual unit. The heater coil temperature of the terminal test is 960℃±15℃, the heater coil temperature of the enclosure test is 650℃±10℃, and the test time is 30s. During the application of the heater coil and within 30s thereafter, observe the test point and surrounding area. The DUT SHOULD have no flame or heat. Or, if the DUT burn, the flame SHOULD extinguish within 30s after the heater coil is removed.

4.3.1.3 Enclosure and Terminal Fire Resistance Test

The DUT is not packed, turned off, and fixed in the center of the test platform.

a) Frequency range: 10Hz～150Hz;
b) Displacement amplitude: 0.075mm (frequency range ≤60Hz);
c) Acceleration amplitude: 10m/s^2 (frequency range >60Hz);
d) Number of cycles per axis: 20.

After the test, check that the DUT SHALL not have any damage and loose component, and its functions and performance SHALL meet relevant requirements.

4.3.1.4 Waterproof and Dustproof Test

MIU SHOULD meet the protection level listed in WSD specification.
DUT SHALL be tested according to BS EN 60529

4.3.2 Environmental Impact Test

4.3.2.1 High Temperature Test

Put the DUT in the center of the high-temperature test chamber, increase the temperature to the highest temperature listed in WSD MIU specification, and keep it for 6 hours. The function and performance SHALL meet the relevant regulations in the standard.

4.3.2.2 Low Temperature Test

Put the tested MIU in the center of the low-temperature test chamber, reduce the temperature to the minimum temperature specified in WSD MIU specification, and keep it for 6 hours. The function and performance SHALL meet the relevant regulations in the standard.

4.3.2.3 High Temperature and High Humidity Test

Keep the temperature (40±2) ℃ and relative humidity (93±3) % in the test chamber, and the test period is 2 days. After the test, DUT SHOULD be restored under atmospheric conditions for 1 hour to 2 hours, and the function and performance SHALL meet the relevant regulations in the standard; the DUT's metal parts SHALL be free of corrosion and rust.

4.3.3 Appearance Inspection

4.3.3.1 General Inspection

The DUT SHALL be easily and conveniently installed in the water meter of the specified model. After installation, it SHALL be mounted tightly, with no

obvious looseness and no obvious gaps in the connection part. After installation, the register reading of the mechanical meter SHALL still be read with the naked eye. The label on the mechanical meter SHALL still be able to be read normally after installation.

4.3.4 Device Information Reading Test

Install the DUT on the water meter, it SHALL connect and join the server, which SHALL be able to collect the following information by commands:

Firmware version
Production number
Hardware version
LoRaWAN version
MIU ID

The data format SHALL meets the definition of [LHKS001-3].

4.3.5 Parameter Setting Test

Install the DUT on the water meter, it SHALL connect and join the server, which SHALL be able to collect the following information by commands:

Set flow leakage setting
Ask flow leakage setting
Answer flow leakage setting

Set reverse flow setting
Ask reverse flow setting
Answer reverse flow setting

Set tamper setting
Ask tamper setting
Answer tamper setting

The data format of the DUT SHALL meets the requirements listed in [LHKS001-4] and the response time of the DUT SHALL follow LoRaWAN Class A operation, i.e., after periodic data upload from DUT.

If the DUT has a bacflow preventer, Set reverse flow setting, Ask reverset flow setting and Answer reverse flow setting tests are not requried.

4.3.6 Data Acquisition Test

The DUT SHOULD be able to correctly collect the data items specified in [LHKS001-3].

Instant forward volume
Instant backward volume
Half-hour volume

Instant flow rate
8 half-hour volumes of past 4 hours
Maximum and minimum flow rate of past day

Install the DUT on the water meter, it SHALL connect and join the server according to [LHKS001-3], and perform time synchronization. The water flow through the mechanical meter is controlled by the electronic valve, which is opened between 5th minte and 25th minute, 35th minute and 55th minute of an hour, and the value of each data volume above is recorded. Then the meter reads the data of DUT through the server. The format of the DUT SHALL meet the requirements listed in [LHKS001-3]. Instant forward volume, Instant backward volume, and 8 half-hour volumes of past 4 hours SHALL have no error. And the response time of the DUT SHALL meet the requirements listed in [LHKS001-3].

If the DUT has a backflow preventer, Instant backward volume test is not requried.

4.3.7 Flow Leakage Event Test

Install the DUT on the water meter. It SHALL connect and join the server, which SHALL be able to set the following information by commands:

min_leak_flow = Q1
leak_duration = 120 (minute)
zero_consume_duration = 120 (minute)

Turn on the electronic valve with water flow >= 3*Q1 for 150 minutes. Then, the server SHALL receive the Flow leakage event from DUT. Turn off the electronic valve.

When the water flow is turned off for 150 minutes, the server SHALL receive the Flow leakage clear event from DUT;

The format and the response time of the DUT SHALL meet the requirements listed in [LHKS001-3].

4.3.8 Reverse Flow Event Test

Install the DUT on the water meter, it SHALL connect and join the server, which SHALL be able to set the following information by commands:

*imm_reverse_flow = 0.5*Q1*
reverse_duration = 30 (minute)
*imm_forward_flow = 1*Q1*
forward_duration = 30 (minute)

Turn on the electronic valve with a negative water flow >=3*Q1 for 1 hour. Then the server SHALL receive the Reverse flow event actively from DUT. Turn off the electronic valve.

Turn on the electronic valve with a positive water flow >= 3*Q1 for 1 hour. Then, the server SHALL receive the Reverse flow clear event from DUT. Turn off the electronic valve.

The format and the response time of the DUT SHALL meet the requirements listed in [LHKS001-3].

If the DUT has a backflow preventer, this test is not required.

4.3.9 Tamper Event Test

This test is for inductive type MIU.

Install the DUT on the water meter, it SHALL connect and join the server, which SHALL be able to set the following information by commands:

tamper_sensing_period = 5 (minute)
tamper_duration = 30 (minute)
no_tamper_duration = 30 (minuts)

Electronic valve SHALL be turned on with water flow ＞ = 3 * Q1. When a magnetic field of 30-100gs is applied to the DUT perpendicularly for 1 hour, the server SHALL receive a Tamper event from the DUT.

When the magnetic field is remvoed for 1 hour, the server SHALL receive the Tamper clear event from the DUT.

The submitted data format and the response time of the DUT SHALL meets the requirements listed in [LHKS001-3].