*Journal of Network Intelligence* ©2021 ISSN 2414-8105 (Online)
**Taiwan Ubiquitous Information** Volume **6**, Number **4**, November **2021**

# Research on operating system middle layer platform design and testing method of smart meter

Xin Zhang, Shangli Zhou, Zi-ang He, Wenwang Xie
Digital Grid Research Institute, CSG
Guangzhou 510700, China
451928256@qq.com, xvyongj@126.com, zhanhe.wu@holley.cn

Shuya Qiao, Dewei Qiu
Chint Group R&D Center (Shanghai) Co., Ltd.
Shanghai 200063, China
zhanhe.wu@holley.cn

WeiKun Chen, Jianjun Li*
School of Coomputer Science and Technology
Hangzhou Dianzi University
No.1 of the 2nd street, XiaSha, Hangzhou, 310018, China
936177378@qq.com, 1137026854@qq.com, dingfeifei7@gmail.com
Corresponding Author: jianjun.li@hdu.edu.cn

Received May 2021; revised August 2021

Abstract. The new smart meter must not only guarantee the core functions of legal measurement, but also support the expansion of new business requirements or software upgrades under the transformation of the Energy Internet. The introduction of embedded operating system has become the development trend of new smart meters cing new challenges based on design efficiency, interactivity and complexity.The paper has studied the implantation of embedded operating system in the management core module software based on the "dual core" module design basis of a new type of the samrt meter that meets the IR46 standard requirement.Design the software layered modular architecture of BootLoader, kernel layer, middle layer and application layer to realize the decoupling and separation of application and kernel of the smart meter,which has effectively solved the shortcomings of software upgrade and expansion due to the strong coupling of the current integrated design of smart meter software,and improved the operating efficiency of hardware resources, software development efficiency and portability.It has focused on the design and interaction principle of the service modules of the middle layer according to the key application characteristics of samrt energy meter measurement, communication, upgrade and safety certification.Finally, design a software gray-box testing program for injecting a simulated test app into the operating system, that can be flexibly set and simulate the interactive module and interactive data of the tested module, which has realized the function and performance verification of each service of the middle layer, ensure the middle layer can provide a reliable and stable operation platform.

Smart meter;operating system middle layer platform;software upgrade;testing APP

1. **Introduction.** Smart meters have functions such as electricity information storage, tiered electricity prices, local communication, remote communication, prepayment, and electricity theft prevention in addition to traditional metering function compared with traditional inductive and electronic energy meters, As a substitute for traditional electricity meters, smart meters have basically realized "full coverage and full collection"

throughout the country with the rapid construction of smart grids in the past decade.In recent years, the International Organization for Legal Metrology (OIML) has promulgated the IR46 "Active Energy Meters" standard that puts forward new requirements on the metering technology and performance for smart meters, clarifying that the software for legal metering and illegal management functions needs to be separated, the software related to illegal system can be upgraded online when the management requirements change, but it can not affect the functions related to legal system measurement [1]. As an important terminal for collecting customer electricity consumption datas, it is no longer just a metering instrument for metering and charging, but also a basic data source for power supply companies to carry out fault repairs, power transactions, customer services, energy efficiency management, distribution network operation, power quality monitoring and other businesses with the new era of energy Internet and Digital grid construction transformation.Smart meter has been given more advanced business requirements to support the construction of power Internet of Things perception layers such as "wide-area interconnection", "holographic perception", and "multi-sensor collaboration" [2, 3].

How to ensure the independence and safety of legal metering and flexibly realize the online upgrade and easy expansion of other application functions based on the implementation of IR46 international recommendations and more expansion requirements for the deep application of electric energy meters,the design of the smart meters facing new challenges.

At present,most smart meter software does not have an operating system and the storage data content and format are single, the software modules are strongly coupled and nested complex and the software does not have the software upgrade function, various functions cannot be flexibly configured.In addition, there is no effective decoupling between the underlying driver and the application software, which is not convenient for development and transplantation, and the operating efficiency of hardware resources is not high [4]. The reliability verification of the smart meter software adopts black-box testing based on the whole machine,the company's R&D side has carried out the code-level white-box test, which is inefficient and complicated.

This paper proposes a kind of embedded operating system software architecture with decoupling and separation between application and kernel for smart meters based on layered and modular design ideas. The new domestic smart meters adopt "dual core" design to realize the separation of legal metering and illegal metering, that is, the metering core module and the management core modul, the modules are physically separated and the metering module operates independently. The embedded operating system is designed to run in the management core module, which is divided into BootLoader, kernel layer, middle layer and application layer. The middle layer is above the kernel layer of the operating system and below the application program, shielding the specific low-level implementation details or differentiation and providing a unified and standardized call interface for the upper-level specific application layer software [5], which can realize the decoupling and separation of the application layer and the core layer of smart meter and effectively solve the problem of software upgrades and expansion, because the current meter software is an integrated design, the software is highly coupled, it is difficult to upgrade and expand the software.It focuses on the design principles and functions of each service module of the middle layer including virtual bus service, communication management service, metering management service, platform management service and security and confidentiality service.Finally, a software gray box-based testing program was designed to verify the functionality and performance reliability of each service of the middle layer, and ensure that the middle layer can provide reliable and stable operation for upper-layer applications [6–8].

## 2. Overall architecture design of smart meter operating system software platform.

**2.1. Overall architecture design.** At present, the new generation of domestic smart meters meet the requirements of IR46 standard, which requires the separation of electronic equipment and components, the independence of metering function and other functions, and the online upgrade of non-metering part software does not affect the accuracy and stability of metering part [9]. The new domestic smart meter design has used "dual-core" modular technology program equipped with metering module and management module which are physically separated and independent of each other, the damage of a certain module does not affect the work of other module. The metering core module provides legal tasks such as power measurement and data storage, the management core module undertakes the management tasks of the entire meter, including cost control, display, external communication, event recording, data freezing, load control and other tasks, which can be expanded according to advanced application requirements, such as other modules such as non-intrusive identification, orderly charging and power quality analysis. The measurement function is separated from other management functions, the measurement function is legally certified, and software upgrades are not allowed, and some software for non-measurement functions can be upgraded online.

The smart meter operating system runs in the management core module to flexibly realize software upgrades and business function expansion, focusing on the management core software design with the operating system. The overall architecture is divided into BootLoader, kernel layer, middle layer, and application layer adopting modular and hierarchical design concept, the detailed architecture design is shown in Figure 1 below.

Realize the decoupling between the underlying hardware driver and the operating system kernel through device registration. The kernel layer of the smart meter operating system is a unified platform framework that undertakes the allocation of software and hardware resources, task scheduling, control, and coordination of concurrent activities of the embedded system of the meter and provides the corresponding kernel interfaces open to the middle and application layers for invocation, and the kernel supports dynamic loading of new smart meter technology applications [10].

The kernel layer mainly includes process management, thread management, semaphore, clock/timer management and interrupt/exception handling, memory management, file system, device management, device driver, etc..

Based on the application characteristics of the basic or key functions of the smart meter, the middle layer encapsulates various complex interfaces or differentiated details of the kernel, provides a unified and standardized calling interface for the upper-level specific application layer software, isolates the kernel layer and the application layerand.Due to the large amount of data exchange between the various modules of the smart meter, establish the middle-tier virtual bus service as the data transfer center in order to ensure the data routing and forwarding function of each interface.

The smart meter has many communication methods that have many physical interfaces for uplink and downlink communication (including PLC, Bluetooth, 485, loRa, WiFi, etc.), the types of data interaction protocols are complex, therefore, a communication management service is established to realize the link management of the communication channel, the judgment of the legality of the protocol, the frame data encryption and decryption processing, the external communication channel and the internal communication function. In order to ensure the data interaction between the management core and the metering core, establish the metering management service to realize the update and

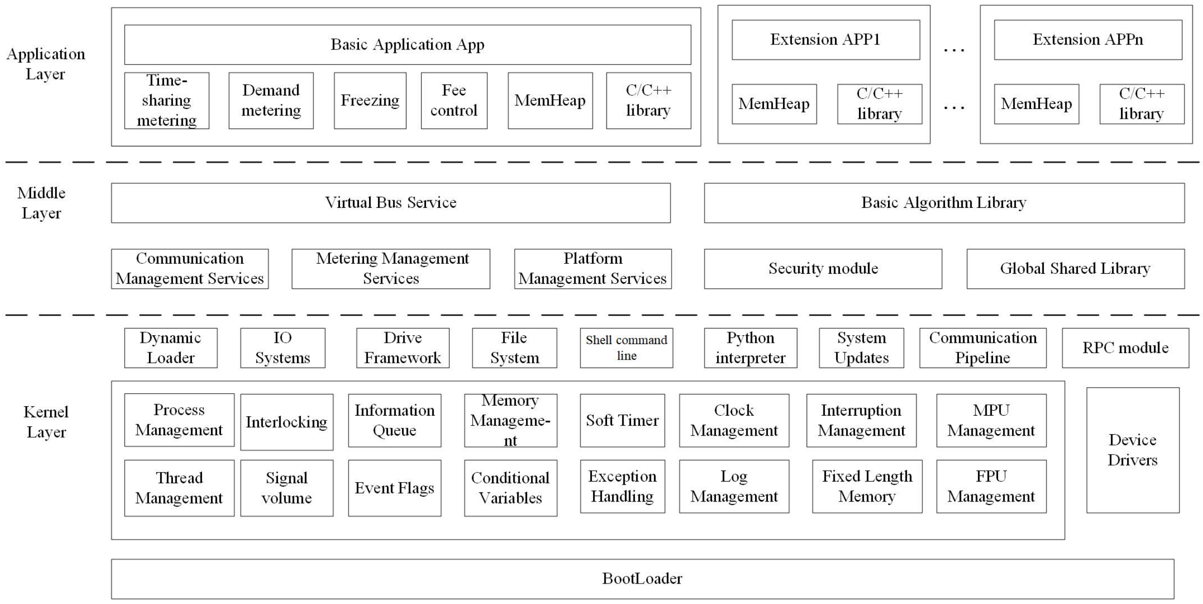


FIGURE 1. A new generation of smart meter management core (with operating system) software architecture

response service of the legal data of the smart meter. Establish the platform management service to support software upgrades due to the smart meter management function software can be upgraded online, and the reliability requirements are high. The communication and data interaction between each service of the middle layer and application APP are all through the virtual bus. The application layer adopts the form of APP, including basic application APP and all kinds of extension APP. The basic application APP realizes the basic business functions of energy meter, with functions of metering, freezing, statistics and record; other extendable application APP, which is for the advanced business application of smart meter such as power quality monitoring, load identification, orderly charging and contract power purchase, that can be expanded according to actual needs, and run on the software platform.

2.2. **Advantages of meter software architecture with operating system.** The operating system architecture designed in this paper has the following advantages.compared with the traditional integrated energy meter software architecture without operating system [11–13].

(1) Adopt modular layered architecture with the characteristics of standardization, componentization, easy expansion, portability, etc.Separate and decoupling technologies are adopted among BootLoader, kernel layer, middle layer, and application layer to realize layered and sub-module management. The operating system and the underlying hardware are deeply decoupled and easy to transplant, which improves the operating efficiency of hardware resources.

(2) The kernel supports multi-task and high-real-time scheduling management, the functional modules of each layer are configurable and can be cut, and flexibly support the application of new technology for the electric energy meter.

(3) Especially the design of the middle layer of the smart meter software architecture encapsulates the complex interfaces or differentiated details of the bottom layer and supports the standardized definition of the upper layer interface according to the key application characteristics of energy meter measurement, communication, upgrade, etc, which reduces the dependence of application development on the operating system kernel

and hardware platform and facilitates the reuse of various layers and effectively solves the shortcomings of software upgrade and expansion due to the strong coupling of the current integrated design of smart meter software and improves application development efficiency and portability at the same time.

3. **Design of key components in the middle layer of smart meter operating system.** According to the above design, the middle layer of smart meter operating system mainly includes key components such as virtual bus service, communication management service, metering management service, platform management service, security and confidentiality module, etc. The middle layer realizes the separation of operating system kernel layer and meter application layer, which is specially designed according to the characteristics of energy meter application, and its operation mechanism and main functions are designed as follows.

3.1. **Virtual bus service design.**

3.1.1. *Design principle of virtual bus service.* The virtual bus service is the middle layer service for the energy meter system, and its design principle can complete the data routing and forwarding function of each interface of the energy meter. The virtual bus is the data flow center, and each service and APP need to send data through the virtual bus to realize the communication and data interaction between each service and APP. The flow of interaction is as follows: after receiving data from each pipeline, the data is firstly parsed, mainly according to the control code and data identification in the 645 protocol, and then sent to specific services and APPs. After receiving the data, the service and APP obtain the corresponding data and construct the data frame, then return the constructed data frame to the virtual bus, which sends the data back to each channel requesting the data.

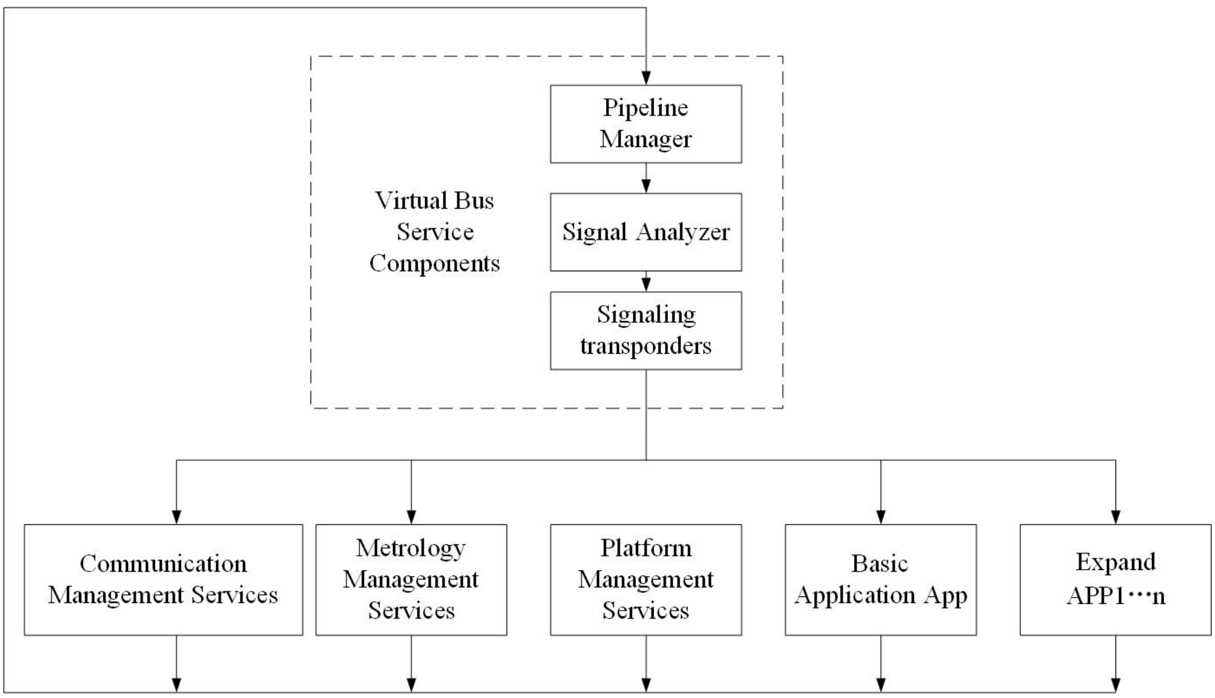


FIGURE 2. Schematic diagram of virtual bus interaction mechanism

3.1.2. *Virtual Bus Service Function Design.* After the virtual bus service component starts, it will establish a listener thread according to the APP with which it interacts with data, and open a pair of pipes to communicate with the corresponding service component and APP. Thus, the communication between processes is realized, and all the communication between processes is routed through the virtual bus service component to achieve data type resolution and distribution, so that the communication between processes can be data type independent. The specific functions are as follows.

(1) Pipeline creation function: Virtual Bus will automatically create pipelines with other APPs and services.

(2) Data interaction function with communication management service: the data from communication management service is routed to the corresponding service or APP, and the data replied to communication management service is correctly returned.

(3) Data interaction function with platform management service: forwarding upgrade commands from communication management service to platform management service, correct return transmission of reply frames from platform management service, and correct routing distribution of data requested by platform management service on its own initiative.

(4) Data interaction function with metering management service: correct routing and forwarding of data sent from each service and APP to metering management service, correct return transmission of reply data, and correct routing and distribution of data requested by metering management service on its own initiative.

(5) Interaction with the basic application APP: data routing and forwarding function for the data requested or answered by the basic application APP, and correct routing and distribution of the data sent to the basic application APP by each service.

3.2. **Communication management service design.** The communication management service is mainly responsible for completing the link management of communication channels, protocol legality judgment, frame data encryption and decryption processing, and the communication functions between the external communication pipeline and the internal virtual bus. The communication management service interacts with the external communication channel device, encryption service and virtual bus module, and its interaction schematic is shown in Figure 3. The communication management service needs to receive serial port data through the device interface of the operating system on the one hand, and application APP data through the virtual bus on the other hand, so it can be divided into serial port receiving data processing module and bus receiving data processing module by function.

3.3. **Metering management service design.** The design principle of metering management service is to be able to complete the update and response service of basic energy meter legal data on the basis of not coupling other business applications or services as much as possible. The metering management service module is the component fully responsible for the interaction with the metering unit. It is responsible for forwarding the active requests of the metering unit to the virtual bus (including the regular provision of the basic metering legal data) and for the requests of other modules or applications for obtaining the data of the metering unit through the virtual bus, whose data interaction diagram is shown in Figure 4.

The data interaction between the metering management service and the metering unit is carried out through the local UART, and the transmission process is carried out in full duplex, and the metering management service and the virtual bus communicate through two pipes. The core functions of the metering management service are both the processing

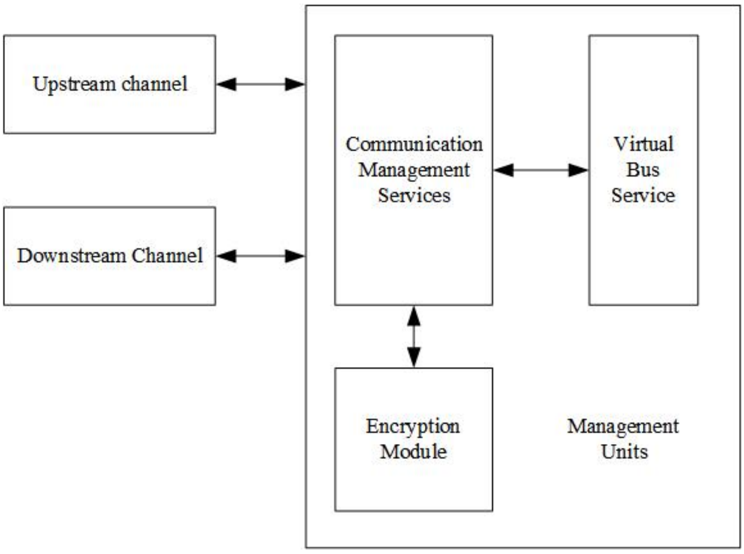


Figure 3. Schematic diagram of communication management service interaction

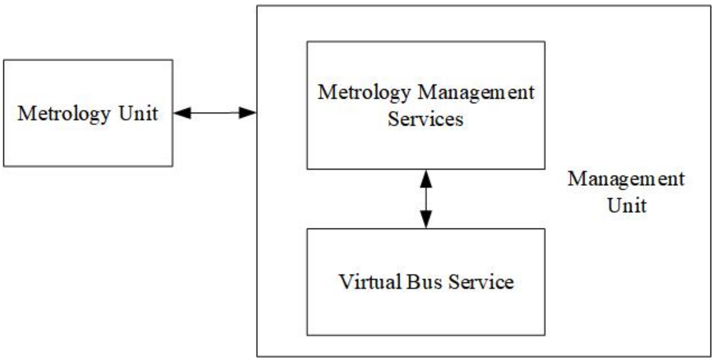


Figure 4. Schematic diagram of measurement management service interaction

function of interactive requests to metering units and the processing function of active reports from metering units.

3.4. **Platform management service design.** The principle of platform management service is to be able to complete the functions of upgrading, uninstallation and installation of meter software on the basis of not coupling other business applications or services as much as possible. The upgrade process can be divided into three major processes: upgrade preparation, file transfer and verification, and execution of upgrade, which mainly interacts with the operating system and the virtual bus.


Figure 5. Schematic diagram of platform management service interaction

3.5. **Security service design.** Security and confidentiality service is the middle layer service library of energy meter system, which is closely related to the business of communication management service module, and provides data verification and encryption and decryption service for other services through the way of interface function call. Its functions mainly include acquiring security encryption service status, data encryption/decryption processing and security authentication command processing. Based on the data interaction and communication channel of new generation energy meter, the module carries out authority classification around data and channel to achieve data confidentiality and integrity protection and effectively prevent replay attack and illegal operation.

4. **Smart meter operating system middle layer test scheme design.** The middle layer of smart meter operating system plays a role in the operation of the whole software system, and the platform is open for the meter enterprises, which can develop the upper layer applications according to the meter business requirements. A simulation test APP-based approach is designed to achieve functional and performance verification of the middle layer software.

4.1. **Overall test scheme.** The test scheme for each key component of the middle layer is shown in Figure 6 below. The management core module is loaded with the software under test, i.e., the management core software, the management core interacts with the metering core module for simulation, i.e., the metering interaction module simulation device, and the APP for testing is loaded into the management core software for interaction according to the demand of the test unit for simulation.

The simulation device of metering interaction module is used to generate some or all of the dual-core communication functions provided by the metering core, and the simulation device interacts with the management core module through the defined interface, and the upper computer controls the source of the simulation device, sends the communication data and modifies the relevant parameters to achieve deep testing of the middle layer.

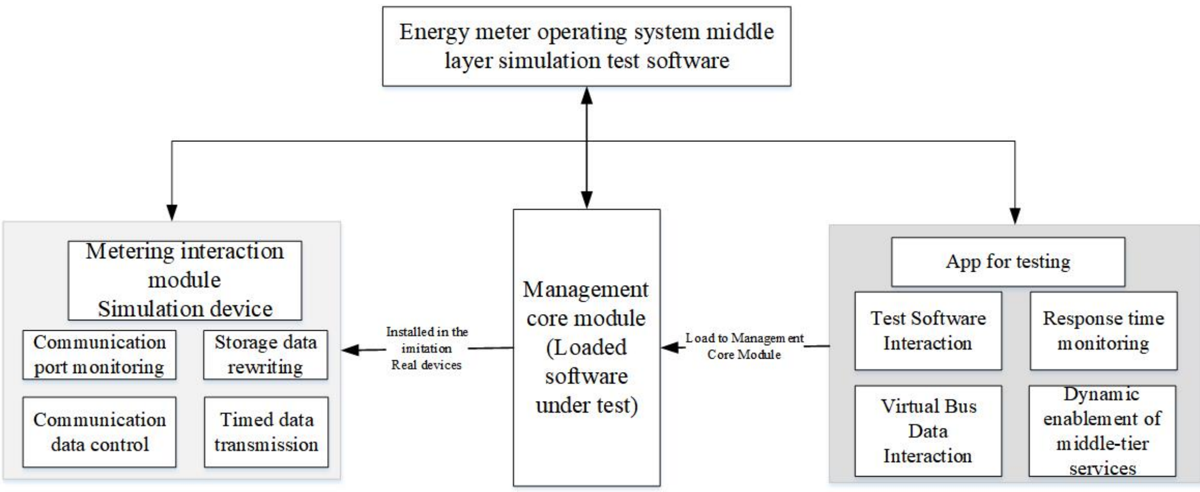


Figure 6. Test plan for the middle layer of smart meter operating system

The APP for testing is loaded into the management core module, which can send data to each module of the management core middle layer according to the testing requirements, receive the reply data of each module of the management core and transmit the data to the upper computer, so as to realize the monitoring of the tested.

The specific test contents of the middle layer can be realized by adopting the simulation of the metering interaction module device and the test APP: (1) realize the independent unit test of the virtual bus service, communication management service, metering management service and platform management service of each functional module of the middle layer, and realize the integration test of each module; (2) realize the performance test of the middle layer: including response time, concurrency test, pipeline monitoring, resource occupation etc.

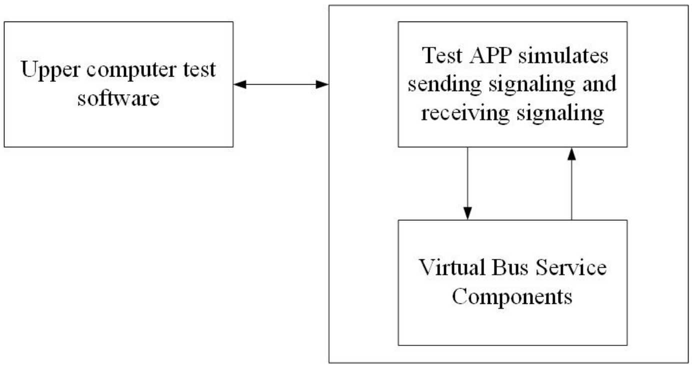


FIGURE 7. Virtual bus service testing mechanism

The test mechanism is analyzed for each service component in the middle layer, and the virtual bus is used as the data flow center, and its test mechanism is shown in Figure 7.

The test APP will send out the test signaling, and the virtual bus will receive the signaling, and forward the signaling to the corresponding target after parsing, and then the test APP will receive and answer the signaling, and then return to the virtual bus and repeat the above process to realize a complete data turnaround. The virtual bus management service components are tested comprehensively according to different test function items.

Except for the virtual bus management service module, other metering management, communication management and platform management service components have the same testing mechanism, as shown in Figure 8. After receiving the command from the host computer, the test APP simulates the virtual bus management service, establishes a good communication pipeline with the service component to be tested, then starts the service component to be tested, waits until the service component to be tested has established each communication interface, and starts to implement the test.

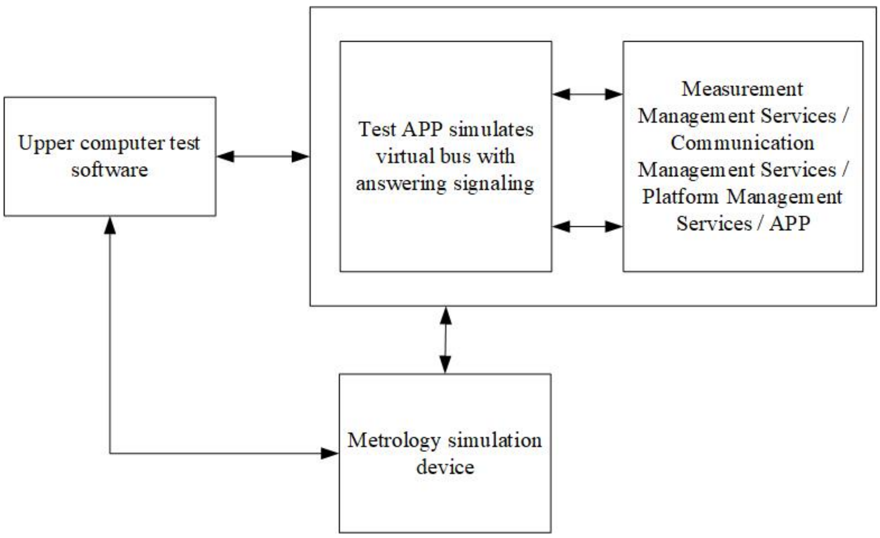


FIGURE 8. Measurement management service/communication management service/platform management service/APP testing mechanism

4.2. **Test items.** The above test plan is used to test each component of the middle layer of the operating system, and the design focuses on testing functions and performance items as shown in Table 1.

TABLE 1. Test items for the middle layer of the operating system

| Test items | Function items | Description |
|---|---|---|
| Communication management services | Channel identification function | Sending data frames to the Communication Management Service via different communication methods, the Communication Management Service should be able to correctly identify the channel |
| | Data conversion and forwarding functions | The main test is to test the correctness of the data conversion and forwarding of the Communication Management Service after it receives the data frames |
| | Pipeline data receiving and forwarding function | Mainly tests the correctness of data response from the virtual bus service to the communication management service |
| | Parameter Management | Test whether the module can be parameterized and run according to the parameters |
| | Communication reliability test | Mainly test the data concurrency and channel cascade reliability of communication management services |
| Virtual Bus Service | Pipeline creation function | Mainly test the pipeline creation function between the virtual bus and each communication service |
| | Data routing function | Stress test of virtual bus with large data |
| | Concurrent processing of multiple pipelines | Mainly testing the cotness of the virtual bus, data distribution, send-back, and abnormal data functions |
| Metrology Management Services | Data format conversion testing | Mainly tests the correctness of the data frame construction pushed and responded to by the metering management service |
| | Metering unit request test | Mainly tests the correctness of the data read by the metrology unit to the management unit |
| | Metering unit read test | Mainly testing the correctness of data read by the management unit to the metrology unit |
| | Basic legal data pushing test | Mainly testing the correctness of the data pushed from the measurement core to the management core. |
| Performance Testing | Communication response time testing | Test the response time capability of the process components under test. |
| | Communication delay test | After the data is sent between the middle layer, the data frame time of the response is controlled by the test APP to test the reliability of the middle layer service by fault injection. |
| | Data concurrency test | Test the data processing ability of the middle layer when a large amount of data processing is required at the same time. |
| | System resource consumption test | Test the resource consumption of the process group under test. |

5. **Experimental results and analysis.** Using the middle layer of smart meter operating system designed by the above scheme, the test unit under test was completed by the method of simulated test APP, and the defects of the middle layer of the operating system were found as listed in Table 2 below, mainly including the failure of pipe creation in virtual services, the error-prone concurrent processing of multiple pipes, and the poor reliability of communication management services, etc. Modifications were made according to the problems to ensure the correctness of the middle layer software.

Table 2. Records of unqualified items

| Test items | Problem Description |
|---|---|
| Virtual Service Data Routing | Virtual bus service data distribution error, which causes pipeline blockage when non-existent DIs are distributed |
| | Concurrent processing of multiple pipelines |
| Concurrent processing of multiple pipelines | Some data is not responded to when different data is processed concurrently in multiple pipelines |
| | Response error when data content differs before and after the same data frame |
| Communication management service reliability | Multiple communication trial methods send different data frames, and data abnormalities cannot be returned. |
| | A communication port sends data frames continuously and quickly, and some data is not responded |

6. **Conclusion.** Based on the idea of software layering, a design architecture of the new generation smart meter operating system platform is proposed, and the division and design of the middle layer is focused on the characteristics of smart meter applications, and the design principles and main functions of the key components virtual bus service, communication management service, metering management service, platform management service and security and confidentiality module are analyzed. The middle layer can shield the specific kernel layer implementation details or differentiation, provide a unified and standardized calling interface for the upper specific application layer software, and realize the decoupling and separation of the application and the kernel, and realize the easy upgrade, scalability, high efficiency, and high portability of the application layer software. Finally, a testing scheme for the functionality and performance of the middle layer is provided to realize the correctness verification of the middle layer software. The research content provides new ideas for the design architecture and testing methods of the new generation of intelligent introduction of embedded operating system.

**Acknowledgements.** This work was supported in part by China Southern Power Grid Limited Liability Company Science and Technology Project (670000KK52200011); National Science Fund of China no.61871170; Key Research and Development Plan of Zhejiang: No.2021C03131; The Basic Research Program of KY2017210A001; Key Laboratory of Brain Machine Collaborative Intelligence of Zhejiang Province.

**REFERENCES.**


[1] Z. Zeng, G.Y. Lin, Research on metering separation of new generation smart meters based on R46 international recommendation, *Electrical Measurement & Instrumentation*, vol. 57, no. 7, pp. 101-106, 2020.

[2] C.M. Chen, L. Chen, Y.Y. Huang, S. Kumar and J.M-T. Wu, Lightweight Authentication Protocol in Edge-based Smart Grid Environmen, *EURASIP Journal on Wireless Communications and Networking*, 2021, https://doi.org/10.1186/s13638-021-01930-6

[3] T.Y. Wu, Y.Q. Lee, C.M. Chen, Y. Tian and NA. Al-Nabhan, An enhanced pairing-based authentication scheme for smart grid communications, *Journal of Ambient Intelligence and Humanized Computing*, 2021, https://doi.org/10.1007/s12652-020-02740-2

[4] E.B. Jing, Design and Research of Smart Meter Based on ARM System, *Low Voltage Electrical Appliances*, vol. 4, no. 3, pp. 26-30, 2011.

[5] Y.L. Xiao, D.Y. Zhao, D.J. Li, Z. Li, L.X.Yang , Z.H. Bai and T. Yang, A technical scheme to realize the next generation of smart meter management core by using domestic operating system and domesticmcu, *Electronic Testing*, vol. 4, no. 23, pp. 93-96, 2020.

[6] L. Mei, N. Liu, Research on software testing method based on domestic operating system, *Aerospace Standardization*, vol. 4, no. 1, pp. 34-40, 2019.

[7] X. Wu, Y. Ma, Q.P. Dong, Research on Real-Time Operating System Performance Measurement, *Journal of System Simulation*, vol. 25, no. 2, pp. 313-316, 2013.

[8] J.M-T. Wu, Z.C. Li, Herencsar. N, Bay. Vo and J.C-W. Lin,A graph-based CNN-LSTM stock price prediction algorithm with leading indicators, *Multimedia Systems*, 2021, https://doi.org/10.1007/s00530-021-00758-w

[9] H.Z. Feng , X.Y. Lin, K. Qi and C.B. Lu,The software architecture design of the intelligent electric energy meter management core based on the IR46 concept, *Information System Engineering*, vol. 4, no. 7, pp. 135-138, 2018.

[10] X.B. He, L.H. Zhong, D.X. Lu, Design and Implementation of Embedded Operating System Encapsulated Layer , *Computer Applications*, vol. 4, no. 5, pp. 89-91, 2003.

[11] X.Y. Ou, X.H. Wang, X.Z. Hou, J.T. Liu , J. Zhang and J. Chen, Research and application of embedded RTOS used in smart meter, *Electrical Measurement and Instrumentation*, vol. 54, no. 12, pp. 106-111, 2017.

[12] M.J. Zhang, Y.A. Liu, Q.L. Zhang and Y. Zhou, Design of Memory Manage Encapsulation in Embedded Operating System Encapsulated Layer, *Modern Electronic Technology*, vol. 4, no. 16, pp. 22-23, 2005.

[13] J.M-T. Wu, Z. Li, G. Srivastava, J. Frnda, V.G. Diaz and J.C-W. Lin, A CNN-based Stock Price Trend Prediction with Futures and Historical Price, *International Conference on Pervasive Artificial Intelligence (ICPAI)*, 2020, https://doi.org/10.1109/ICPAI51961.2020.00032