

Design of logic configuration platform for large power grid security and stability simulation based on object-oriented

Yang Xuetao

China Electric Power Research Institute

Song Dunwen (✉ songdwpaper@163.com)

China Electric Power Research Institute <https://orcid.org/0000-0002-2227-2985>

Ma Shiyang

China Electric Power Research Institute

Research

Keywords: Object-oriented, large power grid simulation, logic configuration, platform design

Posted Date: May 6th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-455567/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Abstract

In response to the multi-state simulation business process requirements of large power grids in actual engineering, this paper designs a set of logic configuration platform for large power grid security and stability simulation. In order to improve the application adaptability of the existing configuration software, firstly, the overall architecture and visual operation interface design of the large power grid security and stability simulation logic configuration platform were proposed. Secondly, the key technology of realization of generalized large-scale power grid simulation logic configuration platform was discussed in depth from the aspects of logic compilation and engine implementation. Based on the configuration platform, the innovative application of the intelligent large-scale grid operation arrangement system is realized. System development replaces the engineering customization mode of conventional manual software coding with visual logic configuration mode, which greatly improves the engineering adaptability of multi-state large power grid simulation business requirements.

1. Introduction

Configuration software is a special software for data acquisition and process monitoring in the field of industrial control. It provides users with a good development interface in a flexible and diverse configuration mode. Its pre-set software modules can realize real-time collection of field device data and can monitor the equipment in real time. Due to its outstanding advantages such as cross-platform, open technology, demand customization, and diversified operating platform[1], it has become more and more widely used in various industries and fields.

First of all, the configuration software is divided into two parts from the function and use stage: the system configuration environment and the system operation environment. All configuration processes of control scheme designed by automation engineer must be completed in the system configuration environment. Various software modules provided by the system and user-expanded functions are used to configure and model the various resources of the industrial control system to form a realistic Human Machine interface (HMI) for engineering applications[2-3].

The basic characteristics of general configuration software include the following three aspects[4]:

(1) Continuity and scalability

Application programs developed based on configuration software, when the external software and hardware environment of the system changes or user needs change, there is no need to make more complex modifications, and software upgrades and secondary development can be carried out easily and simply, and maintenance is more convenient.

(2) Encapsulation

For the convenience of users, the configuration software packs the functions that can be operated in different categories. Users do not need a programmer's background, as long as they master basic

computer programming techniques. All the functions required by a complex engineering system can be easily developed.

(3) Versatility

According to the actual situation of the project, each designer can complete a project including data processing, business processing, network interaction and other functions by using the underlying tools provided by the general configuration software.

Secondly, there are two major types of application modes for configuration software. One is human-machine application, which obtains monitoring and operating data of the field controller through the network, and combines other components to complete specific functions (such as reports, historical records, etc.). The other is process control, based on the real-time controller, it realizes the control function of embedded systems, that is, logical configuration[5].

At present, the general configuration software widely used in various fields at home and abroad includes foreign *Intouch, iFix, Win CC, LabVIEW* and domestic *MCGS, FameVIEW, Kingview* and *Force Control*. With the advancement of intelligent technology and the continuous development of industry, the demand for configuration software has been improved. It has also been determined that configuration software will develop from the simple configuration monitoring function in the past to a higher and broader level[6]. The specific features of some mainstream software are as follows:

iFIX is developed from the *FIX* configuration software. It can seamlessly integrate all the components of *Intellution Dynamics* to realize real-time monitoring of the dynamic production process of complex application systems and large-scale process control systems. However, it is not widely used in China due to its high price[7]. The *Win CC* of *German Siemens* also provides a very complete development platform, which once occupied a large market share for a period of time, but its structure is relatively complicated and difficult to be mastered. It usually requires relevant professional training to be familiar with the application of *Win CC* configuration software[8]. *Kingview (Kingview)* was developed by Beijing Yakong Technology Development Co., Ltd., and it is also the earliest self-developed configuration software in China, and has a large influence in the country, but the network function of *Kingview* is weak and not suitable to be used in large-scale industrial monitoring systems[9]. *Force Control* is a configuration software of Beijing Sanwei Force Control Technology Co., Ltd. It is simple to use, convenient to develop, and has strong compatibility. It can be integrated with most third-party software and hardware systems, and can realize network communication, which has a strong network communication function[10].

Thirdly, in the field of traditional industrial automation control, *ABB, Siemens* and some domestic manufacturers have carried out research work on industrial automation control equipment with modular configuration design, and the relevant technologies are now very mature. Typical representatives are *ABB, Siemens*, etc. The programmable logic control equipment (PLC) [11]. of the iconic enterprise in the traditional industrial automation control field adopts the concept of modular configuration design in software design and hardware design. The hardware can be arbitrarily combined added, removed, plugged

and used according to the actual needs of industrial control. Software functions can also be used through ladder diagram language (LD), instruction list language (IL), function block diagram language (FBD), sequential function flowchart language (SFC) and structured text language (ST) for programming and configuration. This method has been universally recognized and accepted in the field of industrial automation control, but there is a big gap between the actual application requirements of the system and the distribution automation in the aspects of its application scenarios, actual functional requirements, external input/output interface requirements and others[12].

Especially in power system applications, **ABB**, **Siemens** and domestic related equipment manufacturers have also implemented many applications in substation automation equipment, but currently they are all based on the programmable control of protection functions, and the protection logic is implemented through a programmable language. Due to the usage scenarios, in most cases a single device only completes the measurement and control of a single line, and the type and number of external input/output interfaces are relatively fixed. Therefore, the hardware is still designed with a chassis-type plug-in board structure. And the structure does not adopt the modular configuration design, that is, the traditional soft PLC form[13]. The structure is still designed with a chassis-type plug-in board structure, and the software uses a logic programming language to realize functional configuration and user self-programming. Due to its application environment of power automation system is special, now some application scenarios require a single device to meet the measurement and control of more than 16 lines, and the types and numbers of external interfaces vary greatly according to the actual use scenarios of the project site. Therefore, higher requirements are put forward in terms of flexible expansion of hardware modules and flexible configuration of functions for the power automation terminal equipment. However, there are very few researches on the application and research of hardware equipment modularization and configuration design in the field of power automation at home and abroad.

At present, the modular design and application of power configuration software has become a frontier hot issue in ensuring the operation and maintenance safety of smart large-scale grid automation systems. Based on the data acquisition and monitoring system as the core, other advanced application functions of the power system, such as EMS, power market, advanced application functions (power consumption forecasting, state estimation, power flow calculation, etc.) are further expand. The modular power configuration software generally includes three major modules: data acquisition module (I/O module), database management module, man-machine interface module, etc.

This paper designs and develops a set of logic configuration software system for the safety and stability simulation of large power grids, replaces the complete code development model with a visual logic configuration method, separates data processing, algorithm development, and simulation business logic design, and achieves multiple the rapid development and application of generalized power grid simulation business requirements. The application of the large power grid simulation logic configuration platform can give full play to the strengths of power grid analysts, meet the engineering needs of multi-state large power grid simulation business, avoid duplication of labor, and reduce the difficulty of application.

2. Design Of The Logic Configuration Platform For The Safety And Stability Simulation Of Large Power Grids

2.1 Configuration platform structure design

In the field of power systems, the logic configuration method is applied in relay protection. In the microprocessor-based relay protection device, different protected equipment uses different algorithms and processing logics, and sets the pressure plate (function switching) and fixed values according to the specific operating environment, and performs fault identification and tripping operations. The status is diverse. For example, in the over-current protection software, when the line power supply mode or the installation position of the protection device is different, the software flow is different; when the protection model is dual power supply or with power load, the software flow will become more complicated. Although the software process is diverse, because the filtering, value determination, delay and exit operation used in the overcurrent protection software are the same software modules[14], the overcurrent protection can be constructed in the way of logic diagrams, and the same modules can be reused. The way of logical configuration adapts to different application scenarios.

Similar to the above-mentioned relay protection application scenarios, the large power grid simulation process also has the following characteristics: according to the different entry points of large power grid simulation analysis, the simulation process shows polymorphism (diversity), but the basic units used at the bottom layer are the same, so the basic simulation module can be reused, and various themes of the security and stability of the large power grid can be simulated by the logic configuration method. The concept and design of the large power grid simulation logic configuration platform has not been reported at home and abroad.

This paper designs and develops a set of logic configuration system applied to the safety and stability simulation of large power grids, changes the original complete code development model by visual logic configuration, and separates algorithm research, software writing, and power grid analysis logic configuration, etc., which can fully Give full play to the expertise of developers, avoid duplication of labor, reduce the difficulty of testing, and improve the rapid adaptability of scenario-based large power grid simulation applications.

The large power grid simulation logic configuration platform is a logic configuration platform for resource configuration and scene description for large power grid security and stability simulation. It needs to configure algorithm function modules, task logic configuration, communication information and man-machine display. Combined with networked software, the supporting environment realizes the overall configuration of simulation tasks and specific goals for large power grids.

The core part of the configuration platform is the configuration system, which includes configuration tools, compilers, scheduling engines that perform configuration logic functions, communication interfaces, etc, for software and hardware resource configuration and logical configuration operations. This large power grid simulation logic configuration platform generates the task group code from the

configuration logic, and the dispatch engine recognizes the task group code and completes the calculation.

The first is to carry out modular packaging of the simulation unit, reconstruct the existing simulation software to generate configuration components, and use automatic testing and error injection testing methods to ensure the credibility of the reconstruction work;

The second is to design object-oriented configuration data storage, to construct objects for steady-state, electromechanical transient, electromagnetic transient, and mid- and long-term dynamic model data, and to enhance the flexibility of data selection with object-oriented analyzability;

The third is to perform logic functions on the PC to strengthen the testability of logic configuration development and engineering applications.

The configuration system performs configuration and logic configuration tool maintenance, I/O and other hardware information configuration on the PC side, realizes the simulation calculation unit, fixed value and other software parameter configuration, and the component library is formed by the power grid components (models, parameters) participating in the configuration, carries out visual logic configuration.

After completing the configuration function, the configuration information data is stored in Xxx_prj.db through the configuration project management module, and the compiler is called to generate a resource code file. The code file contains software and hardware resource configuration information and efficient configuration logic codes.

The configuration system engine reads the logic code in the code file generated by the compiler, and executes the logic execution interface by the scheduled interrupt or high-priority timed task scheduling logic. The data and software parameters are read through the resource manager, and the algorithm is executed in the logical code order function and logic. After the logic processor is executed, it outputs the operation data and logic execution state, operates the peripherals according to the logic execution state, and the internal communication management module uploads the operation data and logic results to the MASTER to form HMI information and communicate with the system outside of the device.

2.2 Logical configuration platform interface

The simulation logic of the large power grid is configured in the way of logic diagrams, which can realize logical configuration such as AND, OR, and NOT. The data part can also carry out mathematical operations such as addition and subtraction, as well as complex function configuration and parameter transmission.

Power grid simulation logic configuration operation interface, including logic control unit library as shown in the upper left picture (Module), driving events are shown in the lower left picture (Event), Instantiate a symbol into a logical object in the current page by dragging and dropping from the logical control unit

library, Data transmission is realized through connection between objects. When the logical unit is selected, the corresponding attribute configuration can also be performed in the attribute window. The main window is the flow chart of configuration design.

2.3 Overall simulation function implementation

After completing the software and hardware resource configuration and logic configuration through the configuration tool, the compiler is called to generate the coding file, which contains the software and hardware resource configuration information and efficient configuration logic code.

The supporting software running in the configuration platform mainly includes configuration logic code parser, event monitoring program, scheduling service program, interrupt service program, communication service program and logic engine.

According to the logical coding sequence, the scheduling service program performs simulation calculation functions, logical operations, data transmission, data statistics, result display and other functions, logically processes the output data and status, and can also upload the internal communication management module to MASTER, to form SOCKET information and communicate with the external system of the machine.

3. Logic Diagram Compilation And Engine Implementation

The large power grid simulation configuration platform introduces the object-oriented design idea into the development of large power grid scenario simulation analysis software. Algorithm module calls and logic configuration are performed on the terminal through graphics, and the large power grid simulation process is realized by the way of visual logic configuration. Then the logic configuration information is converted into logic codes and running parameters that can be quickly identified by the platform environment through the compiler. Finally, it is scheduled and run by the scheduling unit engine program built into the configuration platform to realize the software function of the configuration.

Among them, the compiler analyzes the logic diagram files and performs correct and efficient logic calculations, generates continuous coding files according to certain rules, schedule algorithm modules in order, and correctly transmits parameters; the engine reads data and configuration information by parsing the encoding file, The driving algorithm module is scheduled according to the coding logic, and the corresponding operation is performed according to the logical running result.

3.1 logic diagrams Compilation

The large power grid simulation configuration platform uses the concept of components (classes) to design application function modules, which makes the power grid simulation module have good encapsulation, reusability and combination. The component is an object-oriented encapsulation that completes specific application functions. and its instantiated objects (logic graph nodes) perform algorithms and other operations according to the configuration parameters and input datas, and the

output data is the input of the node connected to it; the application function is encapsulated according to the concept of components with clear structure and clear objects, which is the basis of visual programming.

Logical configuration is the process of instantiating components, which is stored by object-oriented method. In the compilation work, the stored configuration logic data is encoded according to certain rules, and the coding file is formed by combining the resource definition; the engine reads the coding file and invokes the component methods (functions) of each node according to the coding rules, so that the function of the power network simulation software with logical configuration can be executed correctly.

3.1.1 Component design and instantiation

The applied basic calculation module of the large power grid simulation configuration platform is encapsulated in the form of components. Component is the basic unit for function division and code reuse, which is equivalent to the "class" of object-oriented programming, which helps to improve the reusability, readability and maintainability of the code. Each component completes a specific function, with clear input and output and parameter definitions, relevant initialization functions and timed execution functions (scheduled according to the logical configuration order), and memory parameters can also be set.

The component is equivalent to an object-oriented class, and its instance in the logical configuration view is called a node; when components have been added to the catalog, and multiple instance nodes can be generated through configuration. The input of different nodes is the node connected in the graphical configuration Output, event or data file input.

For example, in the figure below, "AND logical AND", "PFNT", "BATSWNT", and "DECISION" are examples of component classes; the input of the "AND logical AND" node includes 4 conditions, the first one which is the trigger event, The second to the fourth are three necessary input files, which are **.BSE, **.SWI, **.FLT. When the above four conditions are met at the same time, the configuration platform will drive the calculation and execution of the BATSWNT instance module. one of the inputs of the second "AND logic AND" module is the output of "BATPFNT" (Instability.scr). When Instability.scr is valid, the configuration platform will drive the DECISION instance module to calculate and execute, and finally output Strategy.res result.

The input files and event numbers get the corresponding resources or fixed values through attribute configuration.

3.1.2 Node parameter and variable management

Logic configuration is the instantiation of components, and each node is an instance object of the component, so different parameter spaces need to be allocated at compilation time. The generation of node parameters is a key to compiling. Forming a completely independent parameter space for each node is a common method of instantiating objects; However, when this method runs logically, it needs to

insert coding, first copy the node input parameters to the corresponding memory unit of the node, and then call the node logic operation method (function), resulting in memory consumption, at the same time, the increase of logic coding also reduces the execution efficiency.

For this reason, the node parameters are divided into input and output parts for distribution, that is, the node only allocates output parameter space, and its input space uses the output space of the connected node through the configured connection relationship. the input parameters allocate their own space and assign initial values only in the case of no connection. This method can reduce the allocation of node input space, and does not need special parameter copy coding, which solves the problem of space efficiency of conventional object-oriented allocation method.

In the compilation, a variable management module is designed to manage the application and memory allocation of variable uniformly. The data management of this module can be realized by using the database. The generation location is "page ID_node ID_parameter ID". When the initial value is not set, it defaults to 0. According to the unified naming rules, the variable symbol table is output, and combined with the generated position information, the parameters of the running node can be directly located and obtained, which is convenient for the implementation of auxiliary tools such as debuggers. The variable representation is shown in Table 1 as follows :

Table 1 Unified management of variables

Variable symbol	Generate location	Initial value	Data type	Variable offset	BYTE number
U32_0000_4	N1.1.1	0	Uint32	0	4
U32_0004_4	N1.2.1	0	Uint32	4	4
BOOL_0008_1	N1.3.1	0	BOOL	8	1
BOOL_0009_1	N1.3.2	0	BOOL	9	1
F32_0012_4	N1.3.3	0.0	Float32	12	4

3.1.3 Logical coding

Using the object-oriented method to record the configuration logic in units of nodes, and the logic diagram information is stored according to the page. This storage method is convenient for the compiler to use the object-oriented method for configuration logic analysis. After the configuration logic is compiled, the coding file is output. Which includes the execution order of the logic unit, the functions and parameters corresponding to the logic node, the parameter relationship between the nodes, as well as the logic task configuration, memory allocation, input and output mapping, etc. By reading the coding file, the configuration platform engine allocates memory according to the configuration information and starts the task to execute logic coding to realize the function of configuration logic reflection.

3.2 Engine Implementation

3.2.1 Engine initialization

The large power grid simulation configuration platform is divided into two parts: the scheduling service program and the application function module. Which includes event monitoring, interrupt management, task scheduling, logic execution engine and communication service; the application function module is the corresponding execution function of the components participating in the configuration.

After the design and initialization of the logic diagram is completed, the scheduling service program finishes loading the coding file, and the resource manager opens up the corresponding data and event mapping space according to the configuration in the encoding file. The input resource is read and mapped through the interrupt service program or task. The logic execution engine has been used to analysis and execute the corresponding logic codes, According to the coding order, the application function module is called to realize the simulation configuration logic function; after the logic configuration is completed, the communication service program sends the generated report information to the man-machine module for display operation.

The initialization process of the logic configuration platform engine is designed as follows: load the coding file, check the validity of the coding file, access the software and hardware resource parameters through the resource manager, and set the data mapping space for logic use; then read the variable configuration information, and allocate the variable space and initialize; finally, according to the logic linked list generated by the task, the hardware and software environment for logic execution are completed, the engine initialization process is shown in figure 4 below.

The generation of the power network simulation logic list of the logic configuration platform is the key to the correct execution of the engine. A single page of the logic configuration generally realizes complete simulation business functions such as power grid short-circuit current decision analysis, power grid static security decision analysis, power grid transient security decision-making analysis, the logic of different pages are arranged in the order of the design tree; this part constructs a logical linked list according to the task information of each page during initialization:

```

pedef struct struPAGEINFO_
{
    int      pagelD;          //Page ID
    int      pagelD;          //Task ID
    _tEncode_*  pEncode;      //Logically coded address
    tPageConfig*  pConfig;    //Configuration address in the encoding file
    BOOL      isExec;        //Has it been executed
    uint16     curSamplepoint; //Current sampling point
    struct struPAGEINFO_*pNext; //Next page in this task
}TpagelInfo;

```

Among them, pConfig points to the configuration information of this page in the encoding file; pagelD, taskID, pEncode cache the page ID, task ID and logical encoding address in the configuration information to speed up logic scheduling; isExec is the real-time status of task scheduling, and curSamplepoint caches the simulation data when the logic execution of this page enters; pNext points to the next page information, which ends when the task is empty.

3.2.2 Logical Processor

According to the task number parameter, the logic processor first reads the grid simulation data and maps it to the logic running cache, reads the first page information (tPageInfo), and calls the corresponding function according to its logic coding to execute the logic and algorithm. When the coded function is _ When ENC_END_ completes the logic processing of this page (the function code is g_FB_FUNC array subscript, If the code is outside the range of the array, an exception is triggered, fill in the exception information and end the logic execution. After completing the logic of this page, access the next page information through pNext and follow the same process; when pNext is empty, all logic execution of this task have been completed, the output data is written into the corresponding drive queue according to the mapping information through the resource manager, and the coding execution is ended and the correct state is returned.

The logic processor executes logic coding according to the tasks, which is the core of the engine. The following is the logical processor workflow:

3.2.3 Engine encapsulation

The logic execution engine of the large power grid simulation logic configuration platform initializes and generates the logic linked list of each task by reading the coding file, and the task scheduling execution completes the configuration function.

The logic execution engine is an independent and complete software module. and the engine implemented in C language has the ability of cross-platform compilation and migration. For the

perspective of maintainability, this article encapsulates the engine independently, providing initialization, task scheduling and configuration parameter access interface for initialization, interrupt and other related processing procedures; related projects encapsulate the logic processor, resource manager and internal communication management to form the kernel Engine. which partially isolates the engine from the visual design of the configuration platform and avoid their interdependence.

4. Cases And Tests Of Large Power Grid Simulation Logic Configuration Platform

4.1 Intelligent large power grid operation mode arrangement system configuration implementation

The formulation of the power grid operation mode plays a vital role in ensuring the safe and economic operation of the entire power grid. The power grid has gradually developed into a multi-source access, high-density, multi-load type network system, facing the impact of more and more variable factors. Faced with the computing needs to carry out more scenario operation methods, the scheduling process will become extremely cumbersome and complicated, and the amount of information that staff need to process will be greatly increased.

This section establishes the change rules that serve the grid arrangement and security verification of the operation mode, automatically processes the complicated power grid information, generates a more comprehensive mass operation mode, and arranges the automatic workflow based on the intelligent mode, and achieves it through the optimization indicators and high-performance parallel computing. The intelligent operation mode arrangement decision is realized. The application of the logic configuration platform can greatly reduce the development workload and computational burden, improve the comprehensiveness and effectiveness of the mode arrangement, enhance the calculation and analysis methods of the power grid, and improve the level of safe operation of power grid.

The business process of the large power grid operation mode arrangement system is shown in the following figure:

After the above system logic is decomposed, it includes the following sub-processes:

1. Extract the generator branch, load branch, and transformer branch based on the basic operation mode data respectively;
2. Calculate short-circuit current and generate short-circuit results based on basic operation mode data, transient model data and control data;
3. Based on the short-circuit result, extract the cascaded branch of the busbar with the short-circuit current exceeding the standard, and generate the neighborhood branch;
4. Based on generator branch, load branch, transformer branch, neighborhood branch and out-of-limit condition, the branch combination operation is performed to generate the combined branch.

5. Based on the combination branch and combination conditions, the execution mode combination is used to generate mode set 1;
6. Based on mode set 1, perform island detection and generate mode set 2;
7. Based on the method set 2, transient model data, control data, and out-of-limit conditions, batch short-circuit current calculation is performed to generate short-circuit results;
8. Based on the short-circuit results, execute the effect evaluation and generate mode set 3;
9. Based on the basic operating mode data, perform fixed value extraction and generate fixed value results;
10. Based on the basic operating mode data, perform breaking extraction and generate breaking results;
11. Based on mode set 3, fixed value result, interrupting result, execute batch power flow calculation to generate static and stable results;
12. Based on the static and stable results, perform out-of-limit evaluation and generate a set of mode 4;
13. Based on the mode set 4. Transient model data and fault set data, perform batch transient stability calculations to generate transient stability results;
14. Based on the results of temporary stability, perform temporary stability evaluation and generate the set of method 5;
15. The method set 5 is used as the output result and sent to the display computer- node through the network.

The logical configuration relationship of the above steps is shown in the figure 7:

4.2 System data analysis for the operation mode arrangement of large power grids

The test environment is shown in Table 2:

Table 2 Test environment

system version	Windows7 Professional Edition
processor	Intel(R) Core(TM) i5-2430M CPU 2.4GHz
RAM	12.0 GB
System type	64-bit operating system

The following is a statistical analysis of several key links datas in the operation mode arrangement system of the large power grid.

1. a) Basic data statistics

Select the BJ power grid full connection mode to arrange the mode. the data size of the method is shown in Table 3 as below:

Table 3 Case-based approach data scale

Node type	Number
+ [Communication Node Data Continuation Card]:	437
BA1 [LCCDC DC node data continuation card 1]:	42
BA2 [LCCDC DC node data continuation card 2]:	42
BA [LCCDC DC Node Data Card]:	42
BD [DC node data card at both ends]:	82
BE [AC node (PV node) card]:	373
BQ [AC node (PV node with limited reactive power) card]:	3437
BS [AC Balance Node Card]:	8
BZ+ [Flexible DC Node Data Continuation Card]:	28
BZ [Flexible DC Node Data Card]:	28
B [AC node (PQ node) card]:	56595
DC [LCCDC DC control card]:	14
L+ [Line high-resistance parameter data card]:	241
LD [DC line data card at both ends]:	41
LY [LCCDC DC line data card]:	28
LZ [Flexible DC Line Data Card]:	18
L [Line (symmetric) data card]:	35785
PZ [Modify the percentage card of power generation output load by zone]:	459
R [Load regulating transformer card for controlling node voltage (R)]:	82
T [Transformer Data Card]:	42841

b) Method combination operation

Select the BJ power grid full connection mode to arrange the method and combine the related data records:

Table 4 Mode combination test

combined branches number	Combination conditions	Method set 1 number	Average mode generation time	Way set 2 numbers	Way set 3 numbers
28	[1,6]	134596	0.2s	129521	16

c) Method set 3 related indicators

In mode set 3, according to the evaluation index, the top three modes are shown in the following table 5:

Table 5 Part of the mode collection 3 content

Serial number	Mode name	Short-circuit current evaluation index
1	SZ21230_JXX21230_2_JSZ21230_JXX21230_1	0.4866
2	QH21230_JSZ21230_2_JQH21230_JSZ21230_1	0.5376
3	JHJ21230_JQH21230_2_JHJ21230_JQH21230_1	0.5428

d) Statistical analysis of batch power flow calculation

Based on method set 3, Based on mode set 3, batch power flow calculation and statistical analysis are performed, and the results are shown in the following table:

Table 6 batch power flow calculation statistical test

open branches number	Double circuit line	Voltage out-of-limit judgment node number	Branch out-of-limit judgment Number	transformer out-of- limit judgment Number
403	156	1086	417	1034

The test datas show that the logic configuration system can meet the whole flow calculation requirements of method arrangement.

4.3 Application characteristics of configuration software

When constructing the large power grid operation mode arrangement system based-on the large power grid simulation configuration platform, the algorithm modules such as short circuit calculation and power flow calculation are relatively independent and have strong reusability. The reusable function modules and new algorithm modules that embedded in the platform are shown in Table 7 and 8 below:

Table 7 Embedded function modules

Module number	Module name
1	Short-circuit current calculation
2	Batch short-circuit current calculation
3	Batch power flow calculation
4	Batch temporary stability calculation
5	Generator branch extraction
6	Load branch extraction
7	Transformer branch extraction
8	Short circuit point cascade branch extraction
9	Branch combination operation
10	Way combination
11	Island detection
12	Limit extraction
13	Break extraction

Table 8 New algorithm modules

Module number	Module name
1	Evaluation
2	Out-of-limit evaluation
3	Temporary Stability Evaluation

The first thing for large power grid simulation analysis product is the analysis of existing software functions and the reconstruction of algorithms and function blocks, Starting from the realization of a simple input and output function and proceeding iteratively. The steps of using this platform for simulation analysis product development are:

- (1) Analyze the target function, clarify the reusable function modules and the algorithm components that need to be added;
- (2) Configure the hardware resources of the simulation system; configure the software parameters used by the simulation module;
- (3) Use the visual configuration tool to create a logic page according to the simulation function, and use the mouse to drag and drop the components for forming a node instance; realize the logic and parameter

relationship between the nodes through the connection, and use the property box to set the corresponding parameters;

(4) Invoke the compiler to generate the code file, and double-click to locate and handle when there is a compilation exception information;

(5) Logic debugging of each link;

(6) Carry on the system test of the actual product; monitor the real-time operation of each logic through the network connection tool;

(7) Man-machine modules and tools display and configure the corresponding data through encoding the resource information in the file.

The large power grid operation mode arrangement system configuration development process fully embodies the advantages of module independence and reuse: After clarifying the logical relationship, logic configuration based-on a large number of basic modules does not need to code and the problems caused by manual recoding are avoided. The debugging function of the environment greatly facilitates the simulation analysis developers and greatly improves the progress of product development.

5. Conclusion

Firstly, this paper puts forward the structure design and interface design method of large power grid configuration simulation platform; secondly, it introduces the compilation process of the logic diagram in detail from the aspects of component instantiation, node parameter variable management, logic coding methods and so on; at the same time, it is described in detail in the power grid simulation configuration platform. In the configuration platform, how to load coding file to generate the logic linked list, analyze the logic coding execution element function, and the independent encapsulation of logic execution engine to complete the realization of the configuration logic. Finally, taking the intelligent large-scale grid operation mode to arrange the system configuration implementation as an example, the configuration realization process, data testing situation and general implementation steps are introduced in detail.

Practice has proved that the object-oriented software design idea is introduced into the software development of large power grid simulation analysis platform by visual method, and the power grid simulation function is designed with the concept of components (classes), so that the power grid simulation software with visual logic configuration mode has good encapsulation, reusability, It greatly improves the engineering adaptability of polymorphic large power grid simulation business requirements.

Declarations

Acknowledgements

We thank Li de-shu, Peter Wang, Alexander , Wu hui-feng, Wang Hui and B Sari for their helpful discussions and reviews. We would also like to thank SE Bazley, OJ Sturrock, Y Ula, s, and L Bihr for their reviews.

Funding

Supported by Innovation Fund Project of China Electric Power Research Institute (No. XT83-20-001/5242001900CQ): R&D of Intelligent Operation Mode Scheduling and Assistant Decision System for Power Grid Based on Expert Knowledge and Big Data Technology

Availability of data and materials

Data sharing is not applicable to this article as no datasets were generated or analysed during the current study.

Authors' contributions

Song dun-wen contributed to design and implementation of the research, to analysis of results, and writing of the manuscript. Yang xue-tao contributed to the conception and design of this work, and contributed to the production of this manuscript. Ma shi-ying provided supervision and consultation, and contributed to the production of the manuscript. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

References

- [1] Li R S , Fan A L . Monitoring and Control System Design and Implementation of Emulational Transformer Substation Based on the MCGS[J]. Applied Mechanics & Materials, 2014, 513-517:885-888. DOI:10.4028/www.scientific.net/AMM.513-517.885. visited on 15/Apr/2021
- [2] Patel A B , Patel K R , Patel M M , et al. Introduction of Advanced Controlling System: PID Controller, PLC and DCS[J]. International Journal on Recent Trends in Engineering & Technolo, 2010. <https://searchdl.org/public/journals/2010/IJRTET/4/4/125.pdf>. visited on 15/Apr/2021
- [3] Welingkar B, Prasad S, Eisner W, et al. Intelligent migration between devices having different hardware or software configuration: US, US9037685. 2015. DOI:US20080114830 A1. visited on 15/Apr/2021
- [4] Zhang J J . The Development of Industry Ethernet and the Practical Applications[J]. China Instrumentation, 2019, (02)7, 59-62. DOI:10.3969/j.issn.1005-2852.2019.07.013. visited on 15/Apr/2021
- [5] Dai W, Xia Z-X, Chu F , Ma X-P. Configuration Software Platform for Operation Optimization and Control of Industrial Processes[J]. Control Engineering of China, 2020,27(06):919-926.

- [6] MA Xiao-meng, WANG Jing-sheng, ZHAI Jing, LEI Ting. Application of general configuration software in micro cell library[J]. Manufacturing Automation, 2019,41(02):24-26. doi:10.3969/j.issn.1009-0134.2019.02.006. visited on 15/Apr/2021
- [7] Li Yan Dong. Application of GE ProficyFIX Software in Energy Management system [J]. Metallurgical Industry Automation. 2020, v.44(S1):187-189. DOI:CNKI:SUN:YJZH.0.2020-S1-049. visited on 15/Apr/2021
- [8] Endall L . WinCC, profibus and profinet systems for teaching purposes. 2016. <http://researchrepository.murdoch.edu.au/id/eprint/35814>. visited on 15/Apr/2021
- [9] Cui Jun-Hui. Design of data acquisition and Monitoring Control system based on Kingview [J]. Automatic application, 2020,6(05):153-156. DOI:10.19769/j.zdhy.2020.05.062. visited on 15/Apr/2021
- [10] Guo Jing Tang Lian Li Anfu Cai Liangjun Chen Yi. Based on Force Control Configuration Software LoRa Gateway and Corporate LAN Design and Application of Data Mining System for Real - time and Long - distance Transmission of Energy Data[J]. Metrology Measurement Technique. 2019, 329(10):84-88. DOI:10.15988/j.cnki.1004-6941.2019.10.027. visited on 15/Apr/2021
- [11] Zhang Zhi-Hua, Zhou Jie, Cai Yue-Ming. zhoujunDistribution Automation Terminal Based-On Modular Configuration Design[J]. Automation of Electric Power Systems. 2017, 41(13):58-62. DOI:10.1109/iSPEC48194.2019.8975075. visited on 15/Apr/2021
- [12] Quan L Li L. The Study of Soft PLC Running System - ScienceDirect[J]. Procedia Engineering, 2011, 15:1234-1238. DOI:10.1016/j.proeng.2011.08.228. visited on 15/Apr/2021
- [13] LU Yu-jun, LI Cheng, CHEN Hao. Development of Logic Configuration Software for Relay Devices Based on Qt [J]. Electrical Engineering Technology, 2015, 34(5):43-46. DOI:10.19464/j.cnki.cn32-1541/tm.2015.05.013. visited on 15/Apr/2021
- [14] ZHONG Jun, ZHANG Xuan, ZHANG Xue-bin, ZHANG Guang-yu. Study on Cross-Platform Simulation of Control Logic Configuration Based on IEC61131-3 Standard [J]. Measurement & Control Technology, 2013(6):112-115. DOI:10.19708/j.ckjs.2013.06.029. visited on 15/Apr/2021

Figures

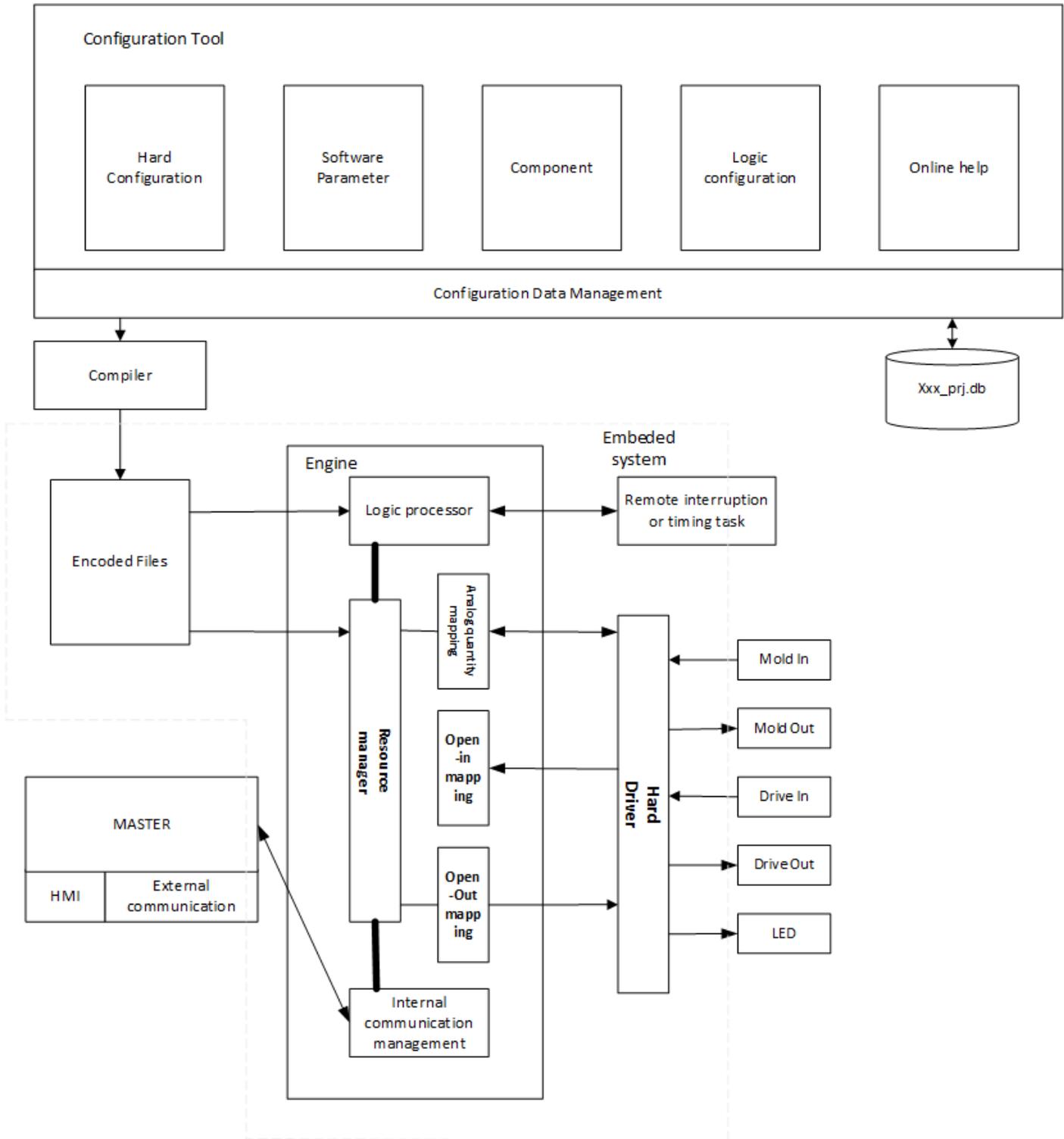


Figure 1

The configuration system composition

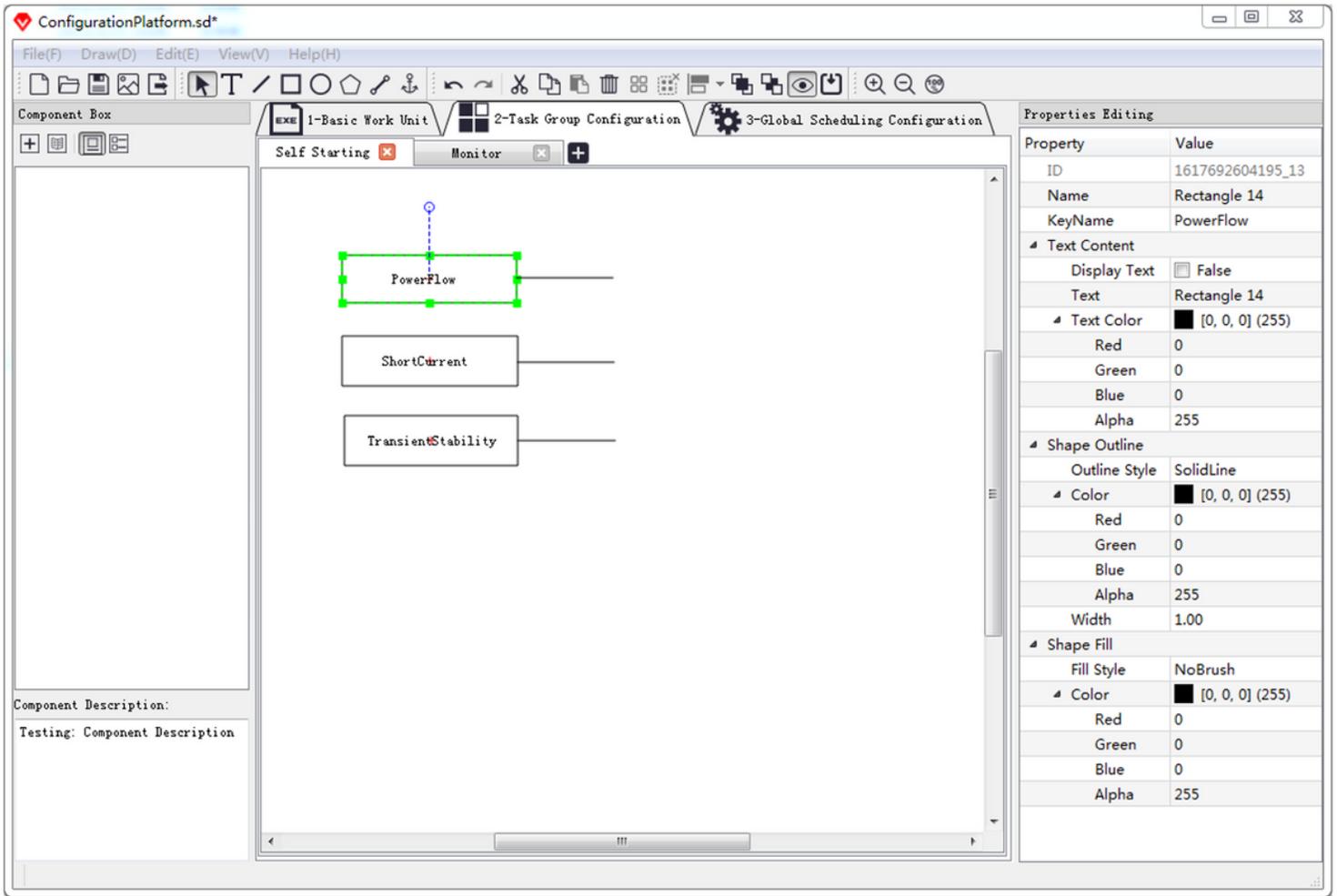


Figure 2

the large power grid simulation logic configuration interface

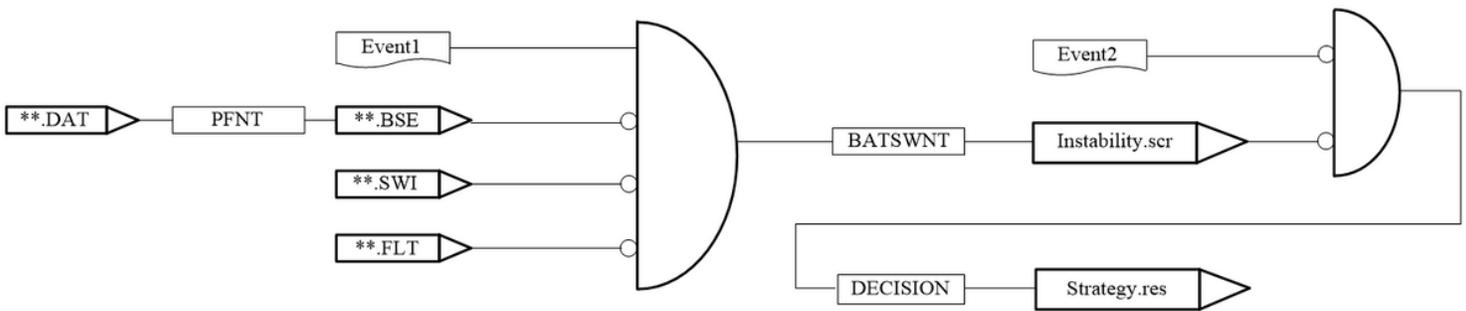


Figure 3

visual logic configuration example

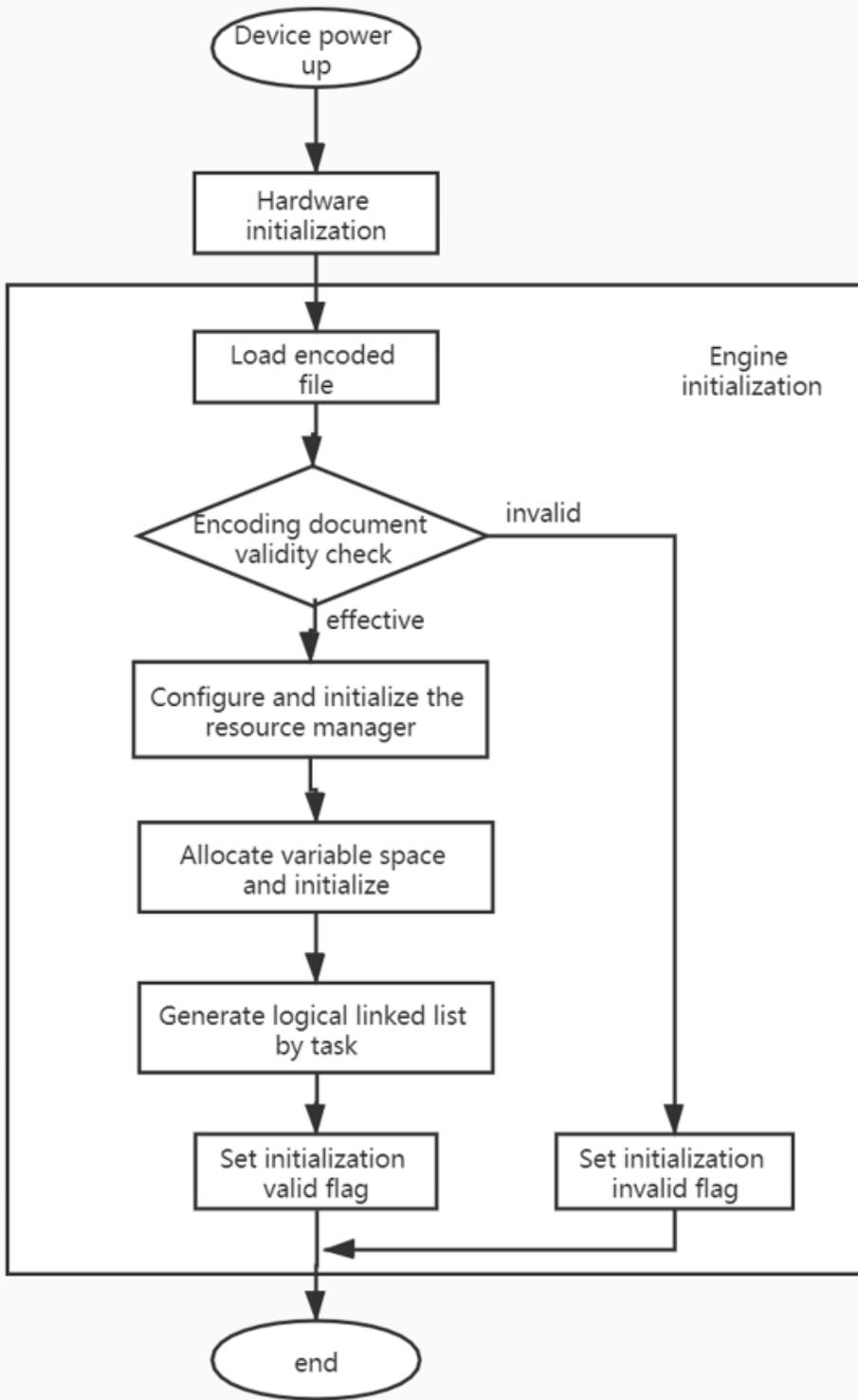


Figure 4

Engine initialization process

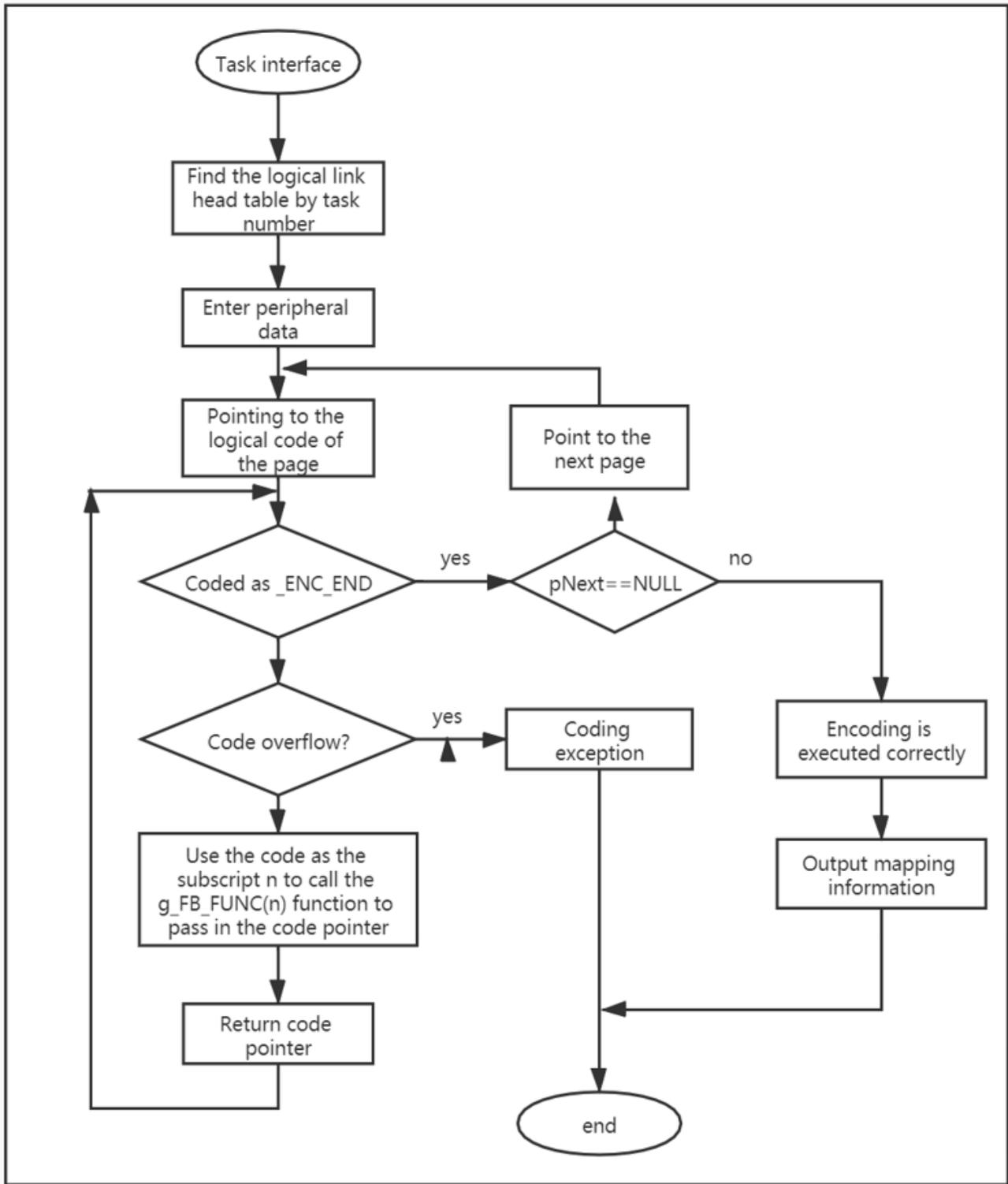


Figure 5

Logical processor workflow

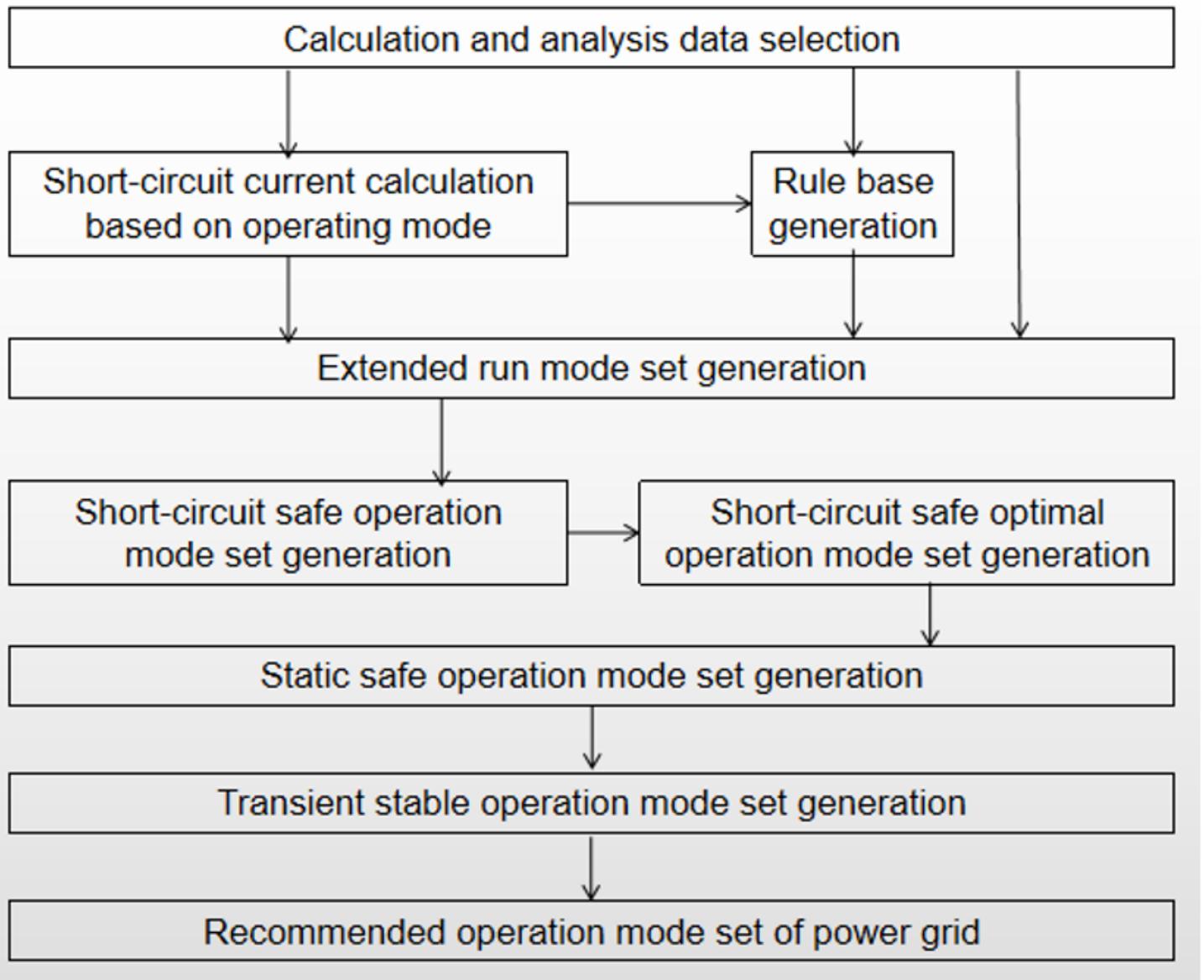


Figure 6

The large power grid operation mode arrangement system business process

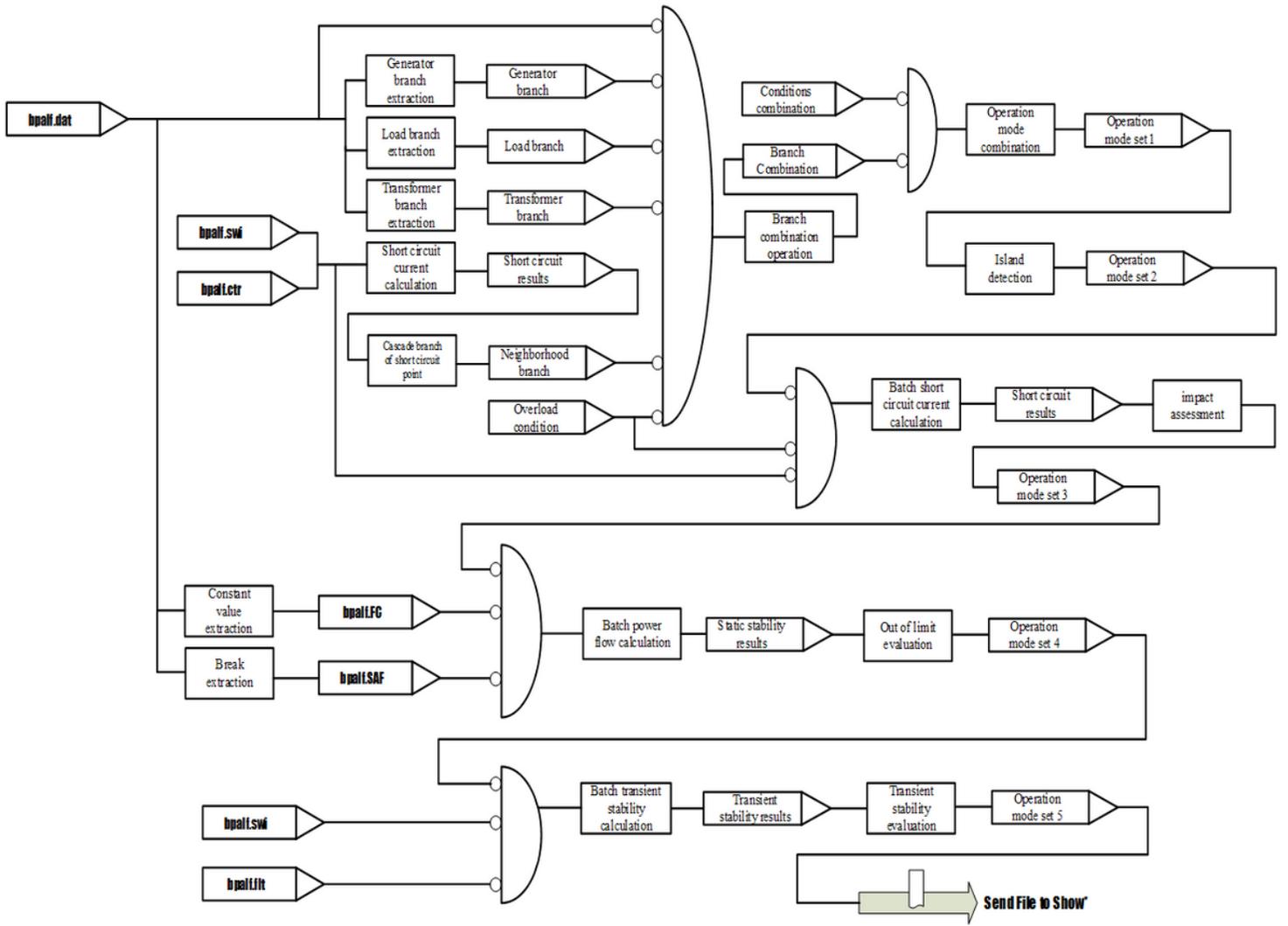


Figure 7

the large power grid operation mode arrangement system logic configuration