Transactions of the Korean Nuclear Society Autumn Meeting
Yeosu, Korea, October 25-26, 2018

In-Core Protection System (ICOPS) Software Development and Verification for Korean Nuclear Power Plants

Kim Chang Ho [a*], Ko Do Young [b], Cho Joo Hyun [a], Kim Tae Jin [a], Yun Jae Hee [a]
[a] *I&C System Engineering Department, KEPCO-E&C, Daejeon, Korea*
[b] *Central Research Institute, Korea Hydro & Nuclear Power Corporation, Daejeon, Korea*
[*]*Corresponding author: kimch@kepco-enc.com*

1. Introduction

The In-core Protection System (ICOPS) is a part of the Plant Protection System that monitors the major variables of the power plant, calculates the departure from nucleate Boiling Ratio (DNBR)/Local Power Density (LPD) and then generates a trip signal when calculated values exceed the trip setpoints. This is the same function as the Core Protection Calculation System (CPCS) of the Optimized Power Reactor (OPR) 1000/Advanced Power Reactor (APR) 1400 Nuclear Power Plants (NPP) in Korea.

The Reactor COre Protection System (RCOPS) developed for Shin-hanul NPP units 1 and 2 also adopted the WEC algorithm (CETOP-D), which has been analyzed as an obstacle to the export of APR1400 NPPs. To resolve this issue, KEPCO-NF has developed a new algorithm, which consists of HEATUP, ORCA and CPEN. Doosan Heavy Industries & Construction supplied POSAFE-Q based ICOPS development equipment and KEPCO E&C is performing the ICOPS software development, implementation and verification/validation. The software lifecycle processes based on the IEEE std. 1074 [1] have been incorporated for software implementation and test phases accordingly.

2. Implementation and Testing of ICOPS Software

In this section, software implementation, software implementation issues to resolve, testing and results of testing are described.

2.1 Model-based Software Development

The software of the legacy CPCS and the RCOPS has been manually implemented for all complex algorithms. However, a computer-aided software engineering tool, widely used in major industries, has been adopted in order to increase the software reliability and to effectively develop and implement the software. In other words, complex algorithms are modelled using a model-based software tool, and the verified source code is automatically generated using a qualified code generator after verifying the functions. At the same time, software documents are automatically generated and published using the function of tool. In addition, the developed source codes are imported as user function of pSET II program using POSAFE-Q Software Engineering Tool (pSET) Wrapper developed by KEPCO E&C. The unit software are being implemented using the imported user functions. The main development process is shown in Fig. 1.

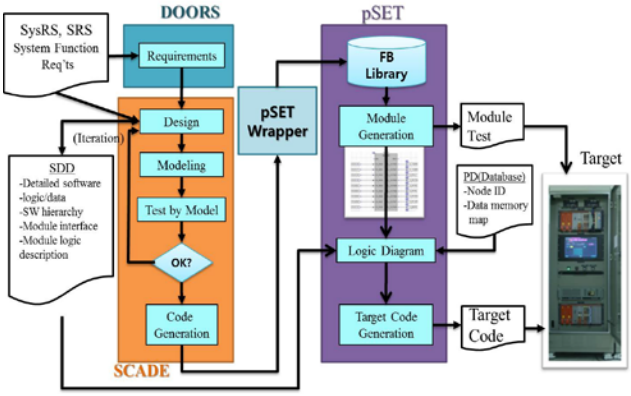

Fig. 1. Software Development Processes

2.2 Software Architecture Design

The software architecture design for newly developed ORCA algorithm to calculate DNBR is performed at the first stage. The safety system software is divided into module and unit software, and related tests are performed respectively. An important aspect of the software design is to establish a scope of the module. If the range of the module is too large, the test cases of the coverage test are increased.

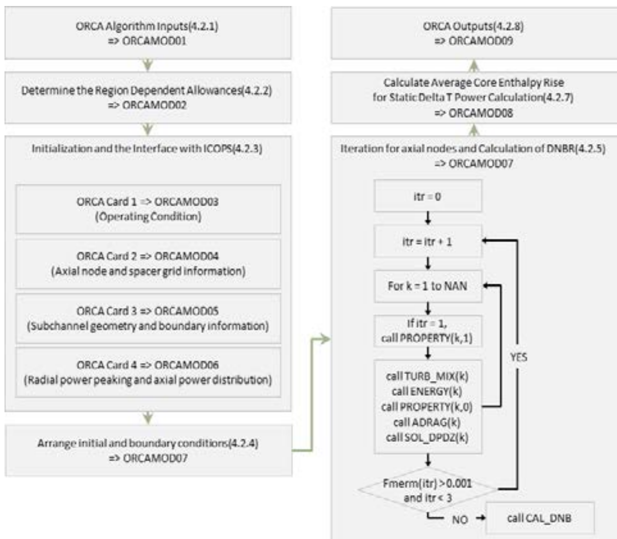

Fig. 2. The Architecture Design of ORCA modules

Therefore, an appropriate range of module software should be selected. The ORCA algorithm is optimized with nine modules considering the inputs/outputs limitation, the complexity of the algorithm, and the module test. The final architecture design of the module is shown in Fig. 2.

2.3 Resolution of Software Implementation Issues

Several issues have been raised during software design and implementation phases. This section explains the major issues and the resolutions for them.

2.3.1 Limitation of pSET II Tool

pSET II is an engineering tool for developing software for the POSAFE-Q Platform. Most algorithms in the ICOPS are implemented by developing user functions of pSET II tool and inserting them into Function Block Diagram(FBD). Huge number of inputs/outputs of ORCA modules has emerged as a major issue in the development of user functions. After searching for various solutions, this has been resolved by reallocating inputs/outputs to arrays of user defined data-type besides existing input/output types such as bool, integer or real.

In addition, another issue of the input/output limitation of the internal function called by the user function has been resolved by declaring the input/output variables as structure type.

2.3.2 CPU Load

For the OPR1000/APR1400 CPCS using the Common Q Platform, the CPU load limit is set at 75%. However, the CPU load limit of the POSAFE-Q platform is 60%. The ICOPS must satisfy this CPU load limitation to ensure deterministic performance required by Branch Technical Position 7-21 [2]. When the newly developed ORCA algorithm is implemented and loaded into the PLC, the CPU load exceeds 60%. The following three methods have been introduced to solve this problem.

In the Functional Design Requirements (FDR) [3] of ICOPS, several enhancements are introduced differently from existing algorithms. One of them is a consolidation of UPDATE (50 msec cycle time) and POWER (250 msec cycle time) algorithms into the 50 msec cycle time HEATUP algorithm. According to the analysis, this has been identified as a factor to increase the CPU load. In order to solve the CPU load problem, POWER task is separated into 250 msec cycle time same as before.

The algorithm of the ICOPS was produced automatically using commercial tools. However, various safety features and defensive codes have been added to the automatically generated source code, which is another factor increasing the CPU load. Therefore, the 7th module of ORCA, which is the largest module, is manually implemented in the same way as the existing one to reduce the CPU load.

2.4 Verification & Validation

For testing of PLC modules, test cases were developed based on the ICOPS FDR. The module test consists of two types. One is a coverage test that examines all the branches of each module. The modules of the ICOPS basically are satisfied 100% coverage. Another is function test according to the requirements. Function test is performed for each module using the ICOPS development equipment and I/O simulator. At this time, using the coverage test cases, it is confirmed that the ICOPS modules meet the acceptance criteria in the FDR.

Unit software for ICOPS is currently being developed and the unit test will be conducted. In addition, One Channel Software Testing (OCST) will be performed after the unit test is completed.

2.5 Test Results

The major modules of the ICOPS are HEATUP, POWER and ORCA. The ORCA module is the most complex one, and this is a challenging task to implement and verify software. Test results of ORCA algorithm performed in the ICOPS PLC shows that the functions of the ICOPS modules are confirmed to meet the acceptance criteria. The test results of the ORCA algorithm are shown in Table 1.

Table 1: Test Results of ORCA algorithm

Major Parameters	Expected value	Actual Results	Deviation (%)
CPIN1/CPIN2	0.97563	0.97563	0.0002%
CPOUT1/CPOUT2	0.98425	0.98425	0.0001%
GPRFST	1.30994	1.30947	0.0366%
XY	1.27773	1.27768	0.0039%
XXY	1.00023	1.00230	0.2065%
LNX	0.26998	0.26961	0.1369%
FKST	1.12168	1.12169	0.0006%
QLOCST	100.603	100.604	0.0001%
DNBR	1.22058	1.21773	0.2337%

3. Conclusion

At the initial phase, the ICOPS was developed to verify algorithm performance for the Korean NPPs.

Based on this former work, the system with unique algorithm is being promoted to overcome the obstacle for exporting Korean NPPs to overseas. The main algorithms are being developed in the software implementation phase and testing phase. If all tests are carried out, the ICOPS system will be successfully developed and verified.

4. ACKNOWLEDGMENTS

This development work was one of essential NSSS design technology development for Korean NPPs, which was supported by the Central Research Institute of the Korea Hydra & Nuclear Power Co. Ltd (KHNP-CRI), Republic of Korea.

REFERENCES

[1] IEEE std. 1074, Developing a Software Project Life Cycle Processes, July, 2006

[2] SRP Branch Technical Position 7-21, Guidance on Digital Computer Real-Time Performance, March, 2007

[3] Ko Seok, Lee Jae Kyu, Park Young Ho, Kim Young Baek APR+KD-TR-CCD-17002, Functional Design Requirements for an In-Core Protection System for APR+, June, 2017