

## Component Testing Methodology for Safety Grade PLC(POSAFE-Q) Operating Systems

Young Jun Lee, Sung-Jae Hwang, Ah-Young Sung, Han-sung Son  
Korea Atomic Energy Research Institute (KAERI), Poscon Corp, Ewha Womans University, EnEsys Corp.  
P.O. Box 105, Yuseong, daejeon, Korea

### 1. Introduction

A Safety-Grade Programmable Logic Controller (POSAFE-Q) can be applied to a Reactor Protection System in nuclear power plants. This POSAFE-Q is being developed in the Korea Nuclear Instrumentation and Control System (KNICS).

A Processor Module Operating System is developed to operate a POSAFE-Q. A variety of testing is executing to guarantee perfectly functionalities and performances.

This Paper addresses the method, procedure and testing environment that are needed for component testing.

### 2. The target of component testing

The POSAFE-Q processor module operating system consists of CRK and systems tasks. CRK is the real time kernel.

Testing targets of CRK and system tasks are divided in detail. Each testing item has identification to achieve relationship with Design Requirements

#### 2.1 CRK Testing Items

The CRK is core of operating system. It operates system task and user task, allocates resource to system, controls the operating system. The major abilities of CRK are as shown below

- Scheduler
- Time Management
- ITC and Concurrence control
- Task Management
- BIHS(Basic Interrupt Handler System)

#### 2.2 System Task testing items

System Task is the task to perform various functions and interfaces with internal, external devices. Testing items of system task are as shown below.

- StartUp
- Shell task
- Diagnosis task
- LoaderRxdy task
- Loader\_Servie task
- FMS\_Write\_Comm task
- FMS\_Read\_Comm task

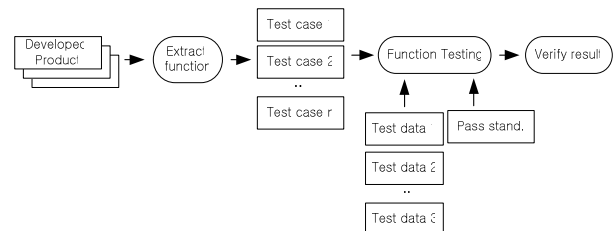
Shell task takes charge of signing and displaying,

Diagnosis task performs diagnosis of system and related jobs, LoaderRxdy task and Loader\_Service task performs service related to communication with pSET(POSAFE-Q Software Engineering Tool), FMS\_Write\_Comm task and FMS\_Read\_Comm task interfaces with Profibus-FMS Communication Devices.

### 3. The Methodology of component testing

#### 3.1 CRK testing

To test The CRK of POSAFE-Q processor module operating system, The tester shall verify exactly the CRK functions that describe in design specification. As Figure 1, CRK functions are caught from developed products. Each Input/Output for test items is defined. After designing CRK functions and test items, we validate that functions of the operating system satisfy design specification.



[figure1] CRK Testing Approach

#### 3.2 System task testing

Systems tasks of The POSAFE-Q processor module operating system are tasks that interface with other devices. These components perform a conditional operation in accordance with input value or global variable and transmit computational results to upper components. For this reason, the components with these features perform whitebox testing.

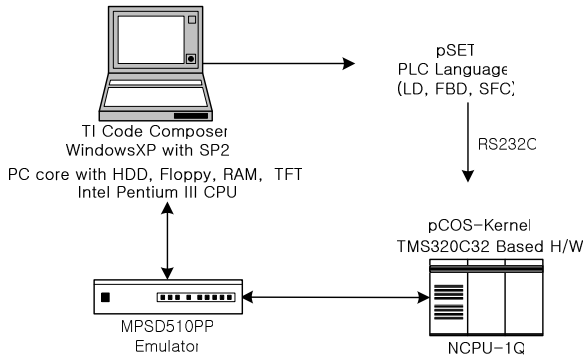
The whitebox testing criteria applicable to systems tasks is Branch Coverage. It calculates a condition statement. Its formula is below

- Branch Coverage = the number of execute condition statement / the number of total conditional statement

The testing approach of system tasks is similar to Figure 1's approach. But the Inline Assembly language area that is used to implement system tasks is not considered. We analyze that these test datum satisfies perfect Coverage

#### 4. Testing Environment

The POSAFE-Q processor module operating system uses component testing environment as Figure2. It supply testing environment with emulator between Host and Target system



[Figure2] component testing environment

Generally, software and hardware resources are needed to test software of target system. The POSAFE-Q processor module operating system needs Windows XP, Code Composer, MPD510PP Driver and pSET to test component. Code Composer downloads executing files and supplies debugging ability to verify variables in the middle of operation. Hardware resources need bus module, power module, emulator to operate processor module. Host PC used in host area is also needed.

#### 5. Component Testing procedure

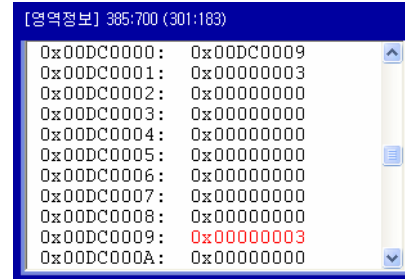
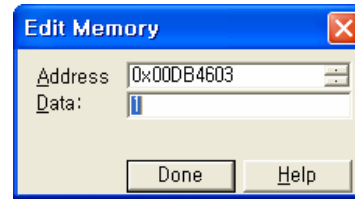
Table1 is shown test data chart to design for testing component of POSAFE-Q processor module operating systems. It describes predicting results about input value. If testing values are assigned in component input variables, expected output values have to be described.

[Table1] Test Criteria Chart

n	In variable	Input value	Out variable	Expect out value	result	P/F	etc
1	In variable1	Input value 1	Out variable 1	Expect out value 1	result1		
2	In variable2	Input value2	Out variable2	Expect out value2	result2		
	.	.	.	.	.		
	.	.	.	.	.		

Watch Window screen in Code Composer is used to assign testing value in input variable. In detail, Watch Window screen displays variable list and Edit Variable screen allocates variable to values.

Memory Edit screen in Code Composer is also need to assign values to Main Memory



[Figure3] variable and memory value verification

To verify test result, after pausing operating system at special point of implementation code, we inspect the variable for testing.

Finally, we determine whether it is success or fail through comparison of real result and expecting result.

#### 5. Conclusion

The testing of POSAFE-Q processor module operating system is performed to satisfy Branch Coverage at source code level.

First, we divided operating system into CRK and system tasks. Second, we created test data from test criteria that is needed to test operating system. Third, we assign input value to variables through Tool, Finally we acquire a result value after running operating system in testing environment.

We confirmed that the components of POSAFE-Q processor module operating system are operated correctly.

Also, we verified the consistency of components and design specification through comparison of predicting results and test results

#### REFERENCES

- [1] USNRC Reg. Guide 1.152, Rev.01, Jan. 1996, "Criteria for Programmable Digital Computers System Software in Safety Related Systems of Nuclear Power Plants"
- [2] IEEE Std. 7-4.3.2, 1993, "Standard Criteria for Digital Computers in Safety System of Nuclear Power Generating Stations"
- [3] IEEE Std. 829-1998, "IEEE Standard for S/W Test Documentation"
- [4] IEEE Std. 1008-1987, "IEEE Standard for Software Unit Testing"
- [5] KAERI, "A Safety Grade PLC(POSAFE-Q) Processor Module Operating system Component Test Plan" KNICS-PLC-STG101-01
- [6] KAERI, "A Safety Grade PLC(POSAFE-Q) Processor Module Operating System Component Test Procedure" KNICS-PLC-STP151-01