Case-study: The regression test for the operating systems of a PLC

Young-Jun Lee, Sung-Jae Hwang
Korea Atomic Energy Research Institute (KAERI), Poscon Corp
P.O. Box 105, Yuseong, daejeon, Korea

1. Introduction

The regression test is that we retest changed software as well as interfaces with other software with relation to the software itself when a software change has happened. The performances and functions of software with interfaces to other software can be operated variously, so a regression test is a kind of test that shall be performed for critically safety-related software.

This paper describes the regression test that was performed after modifying the operating systems(OS) which were developed by KNICS(Korea Nuclear I&C System) project. Especially, it describes the experience about software integration test that is focusing on interfaces with other software.

2. The modification and interface item of OS

OS is the core software that is operated in a processor module of a safety-grade programmable logic controller. There are several tasks in OS. Also, the tasks related to a communication, a diagnosis, and a watchdog timer are modified during the development phase.

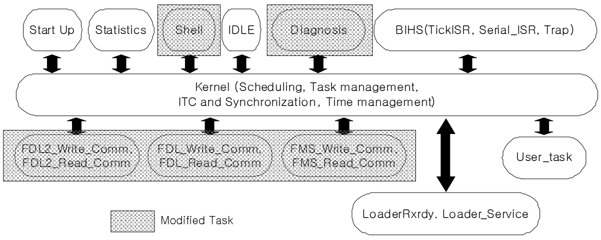
Start Up
Statistics
Shell
IDLE
Diagnosis
BIHS(TickISR, Serial_ISR, Trap)
Kernel (Scheduling, Task management, ITC and Synchronization, Time management)
FDL2_Write_Comm, FDL2_Read_Comm
FDL_Write_Comm, FDL_Read_Comm
FMS_Write_Comm, FMS_Read_Comm
User_task
LoaderRxrdy, Loader_Service
Modified Task

Figure 1. Modified Tasks of OS

As shown in Figure 1, the modified components have relations with a kernel that is a core of the operating systems. We can define the interface items for the regression test from the relations. The interface items are as follows.

- The interface items of the task related to a communication
  - Initialization parameter for communication modules
  - Interfaces with FMS communication
  - Interfaces with HR-SDL communication
  - Interfaces with HR-SDN communication

- The interface items of the tasks related to a diagnosis
  - Configuration information of the IO device
  - Error status of communication device

- The interface items of kernel in operation systems
  - Scheduling and Fail-Safe function

3. The items for regression test

The items for the regression test are defined from an interface object after classifying an interface object based on a modification of the operating systems software.

- Initialization test of the communication modules
- Function test of the tasks related to a diagnosis
- Transmission test of the data by a User Tool
- Performance test of the scheduling
- Transmission test of data from one PLC to another

- Initialization test of the communication modules

It shall be reviewed whether the variable that is set by user is normally initialized or not, because the tasks related to a communication in operating systems were deleted. The tests to confirm that initialization values by a user have sent to the inner variables of the operating systems shall be performed.

- Function test of the tasks related to a diagnosis

The tasks related to diagnosis analyze status of equipment and notify its result to user. The tests to confirm that the status of equipment is saved at another position shall be performed after changing the position about the diagnosis information.

- Transmission test of the data by a User Tool

The tests to confirm that display-configuration and sending-parameter by a user tool are normally saved shall be performed. It is similar to the initialization test of the communication modules.

- Performance test of the scheduling

It is the most important test in regression tests of operating systems. The priority of the tasks is changed and their scheduling algorithm is also changed. We shall confirm that the deterministic operation of the application is guaranteed with the correct scheduling and the diagnostic function is correctly performed according to the CPU workload. The tests for the performance of the CPU scheduling are in detail as follows.

- Execution time of the tasks by priority and to guarantee the deterministic operation of application

We tested whether a task with a higher priority than the application disturbs the deterministic operation of the application. The eight applications are running in the CPU workload of 60% for this test. As a result, the task with a higher priority than the applications was operated regularly below a 200us execution time. Therefore, we confirmed that the task has no effect on the deterministic action of the application

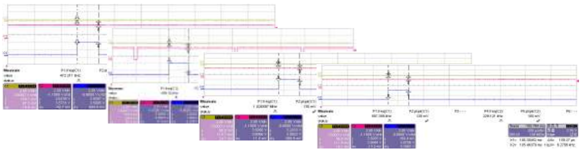

Figure 2. Execution time of a task with higher priority (159.10us ➔ 159.10us ➔ 157.48us ➔ 159.27us)

- Guarantee diagnostic function below CPU workload of 95%

We tested whether the diagnostic function below the CPU workload of 95% is guaranteed. In other words, the task related to diagnosis has a possibility to not run if an application preempts the CPU continuously in the case of the priority modification. Therefore, we have to show that the task related to diagnosis performs its functions normally.

As a result, we confirmed that the task related to a diagnosis was performed on the accumulative counts after finishing the applications.

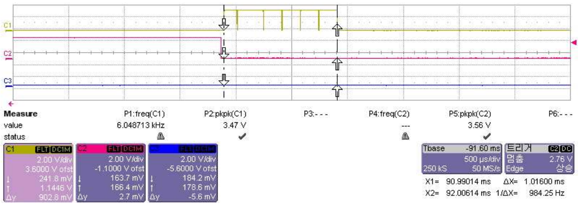

Figure 3. Execution of a task related to diagnosis

- Error information of a case persisting for five seconds in upper 95% of a CPU workload

If the CPU workload of 95% continues for 5 seconds, for guaranteeing the safe function of the equipment, the error information shall be notified to a user. The input values for the testing are as follows.

App1(exec time=24.5ms, exec period=30ms),
App2(exec time=14.0ms, exec period=20ms)

As a result of the test, the error information was set in RAM and the ERROR LED was light, so we confirmed a normal operation of the operating systems.

- Scantime Violation occurrence

When the scantime violation occurs in an application program, operating systems shall notify the information to a user. We can recognize that the problem during a scheduling is occurring. The input values for the testing are as follows.

App1(exec time=24.5ms, exec period=60ms),
App2(exec time=14.0ms, exec period=40ms)

As a result of test, the RUN LED was flickering and the ERROR LED was repeatedly changing from GREEN color to RED. We confirmed that the operating systems perceived the scantime violation.

- Transmission test of data from one PLC to another

For this test, 2 PLC was connected by communication modules. We confirmed that the data in the Dual Port Memory of one PLC was transmitted correctly to another PLC due to a change of the communication methods. We had a right result about communication.

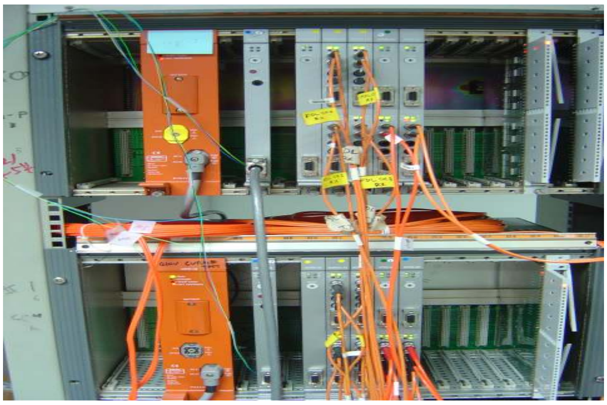

Figure 4. Test environment for data transmission

5. Conclusion

The regression tests guarantee the operation of a system through some retests in the case of changing software and hardware. At that time, the tests have to examine modified software as well as the interfaces with another software related to the software. This paper dsicribes the regression tests that were perfomed according to the modification of a operating system which was development by the KNICS project. The function and performance of the software was guaranteed through retesting the operating software itself and its relations with another software.

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] POSCON, "The Evaluation Report for performance and safety of POSAFE-Q communication module", KNICS-PLC-RR334-01