Proceedings of the Korean Nuclear Society Spring Meeting
Gyeongju, Korea, 2004

Software Design Specification and Analysis (NuFDS) Approach for the Safety Critical Software based on Programmable Logic Controller (PLC)

Seo Ryong Koo and Poong Hyun Seong

Korea Advanced Institute of Science and Technology
Department of Nuclear and Quantum Engineering
373-1 Guseong-dong, Yuseong-gu, Daejeon, Korea 305-701

Jin-Yong Jung and Seong Soo Choi

Atomic Creative Technology Ltd.
1688-5 Sinil-dong Daedeok-gu, Daejon, Korea 306-230

Abstract

This paper introduces the software design specification and analysis technique for the safety-critical system based on Programmable Logic Controller (PLC). During software development phases, the design phase should perform an important role to connect between requirements phase and implementation phase as a process of translating problem requirements into software structures. In this work, the Nuclear FBD-style Design Specification and analysis (NuFDS) approach was proposed. The NuFDS approach for nuclear Instrumentation and Control (I&C) software are suggested in a straight forward manner. It consists of four major specifications as follows; Database, Software Architecture, System Behavior, and PLC Hardware Configuration. Additionally, correctness, completeness, consistency, and traceability check techniques are also suggested for the formal design analysis in NuFDS approach. In addition, for the tool supporting, we are developing NuSDS tool based on the NuFDS approach which is a tool, especially for the software design specification in nuclear fields.

1. Introduction

In the safety-critical systems such as NPP I&C systems, there is increasing use of software based systems. However, it is required the very high confidence for software quality. Recently, the concept of software verification and validation (V&V) is accepted as a way to assure the quality of new digitalized safety-critical systems [1]. And, the thorough V&V processes should be needed throughout the software development life cycle. In IEEE Standard 1012 "Software Verification and Validation"[2], minimum V&V tasks for safety-critical systems are defined along each phase. Figure 1 shows software V&V tasks during the lfie-cycle.

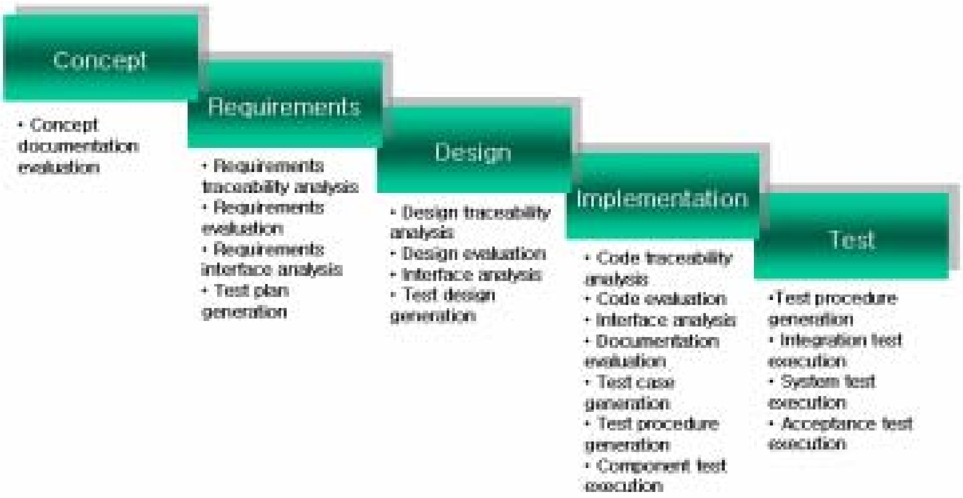
Concept
• Concept documentation evaluation
Requirements
• Requirements traceability analysis
• Requirements evaluation
• Requirements interface analysis
• Test plan generation
Design
• Design traceability analysis
• Design evaluation
• Interface analysis
• Test design generation
Implementation
• Code traceability analysis
• Code evaluation
• Interface analysis
• Documentation evaluation
• Test case generation
• Test procedure generation
• Component test execution
Test
•Test procedure generation
• Integration test execution
• System test execution
• Acceptance test execution

Figure 1. Software V&V Tasks during the Life-cycle

Among these S/W life-cycle phases, software design is a process of translating problem requirements into software structures that are to be built in the implementation phase. In general industry, a Software Design Specification (SDS) should be produced at this software design phase. SDS describes overall system architecture and contains a definition of the control structure model. SDS should be evaluated for software quality attributes such as correctness, completeness, consistency, and traceability. Therefore, it is the most important to define an effective specification method for the software design phase. The effective specification could be absolutely helpful for the design verification and validation. Also, a well-formed design specification is very useful for the coding in implementation phase. Therefore, an implementation product such as code should be easily translated from the design specification. For more smooth transition from design phase into implementation phase, it was needed to combine two phases. Since Function Block Diagram (FBD) language of PLC usually looks like design features, we can reduce the coding time and cost through the combining design phase and implementation phase, especially PLC application.

In this paper, Software FBD-style Design Specification and analysis (NuFDS) approach for the safety critical software based on PLC is proposed. Now, for the tool supporting, we are developing NuSDS tool which is the tool for the software design specification in nuclear fields based on our approach.

2. The Nuclear FBD-style Design Specification and analysis (NuFDS) Approach

In the software design phase, SDS is a description to show how to create a design which accurately and completely satisfies the behavior and constraints in the SRS. In this work, an adequate specification technique is needed for the systematic verification and easily translating into implementation phase. Therefore, in this section, we will suggest the nuclear FBD-style design specification and analysis (NuFDS) approach for generating and analyzing the SDS of nuclear I&C systems. Figure 2 shows the schematic diagram of the NuFDS approach.

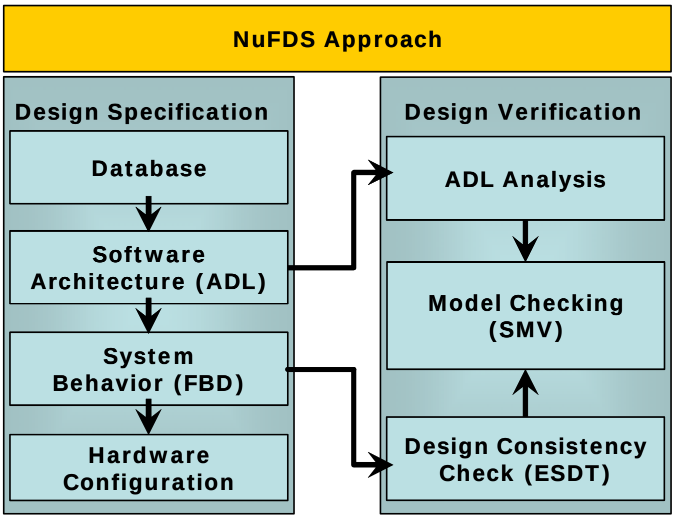

Figure 2. Schematic Diagram of the NuFDS Approach

The NuFDS approach supports two kinds of software design activities; Design specification and Design verification. Design specification of the NuFDS approach consists of four major specifications as follows; Database, Software Architecture, System Behavior, and PLC Hardware Configuration. SDS could be generated according to these four major specifications. In the design verification of the NuFDS approach, formal design analysis based on Architecture Description Language (ADL) and Symbolic Model Verifier (SMV) was proposed. In addition, for the traceability analysis between requirements and design, the design consistency check based on Extended Structured Decision Table (ESDT) was also suggested in the NuFDS approach.

2.1. Design Specification of the NuFDS Approach

In this section, the design specification of the NuFDS approach is represented according to four major specifications.

Database is an organized collection of related data for the systems and it can be a basis for the whole software design phase. In this work, we need information about input/output variables related to the digital system for PLC programming at the implementation phase. Actually, the completeness and consistency between input/output variables are the most important in PLC programming. For the database specification in NPP I&C systems, therefore, we defined eleven kinds of database fields related to input/output variables as follows:

- *Name:* the primary name of the variable
- *Description:* a notation for representing variables
- *Type:* Real, Int, Boolen, Bit and Time
- *Initial:* initial constant value needed
- *Address:* allocated memory address of PLC H/W
    - S_Address: Source address
    - D_Address: Destination address
- *Comm:* communication media
- *Source:* from where (starting position)

- *Destination:* to where (ending position)
- *Counter:* used count of variable in the specification
- *Consistency:* the user define field for DB consistency check
- I/O: Input or Output

In the first stage of the design, we should compose the input/output variables with these database fields and then we can use database table as a basis on other specifications in the design phase.

In the software design phase, architecture design represents the structure of data and program components that are required to build a computer-based system. Software architecture is also considered as the primary design artifact of a digital NPP I&C system in this work. We propose a simple notation for the software architecture design in the digital system based on PLC because users and designers prefer a simple notation like block diagram style than a complex one. Figure 3 shows a simple notation for the software architecture design proposed in the NuFDS approach.

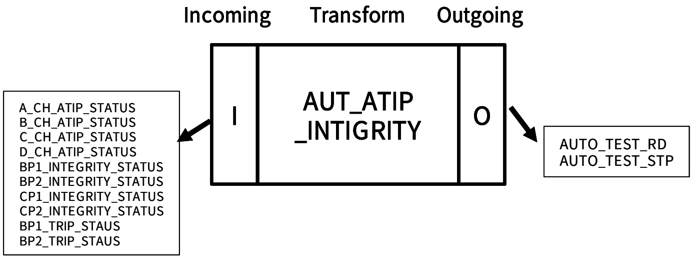

Figure 3. A Simple Notation for the S/W Architecture Design

A simple notation for the software architecture specification can represent software module decomposition for PLC programming. Proposed notation consists of 'incoming' representing the inputs, 'transform' representing the computations, and 'outgoing' representing outputs. Incoming and outgoing are related to database composed by designer in the database specification. Using this proposed notation, simple software architecture for PLC programming could be constructed in the top-down design manner. And then, this architecture notation will be refined by ADL specification for the formal design analysis.

System behavior represents the software's reaction to some external event [3]. That means the interrelation between software modules and functions. In a digital NPP I&C system based on PLC, system behavior specification can help the PLC programming. Therefore, the use of a similar notation with programming language will be very useful in the implementation phase. PLC programming supports five languages and these PLC programming languages is defined in IEC 61131-3 standard [4]. Among these languages, Function Block Diagram (FBD) is most widely used as a PLC programming language in industry fields. In order to codify more easily at the PLC implementation phase using FBD, system's behaviors would be described by FBD-style specification in this work. Figure 4 represents an example of FBD.

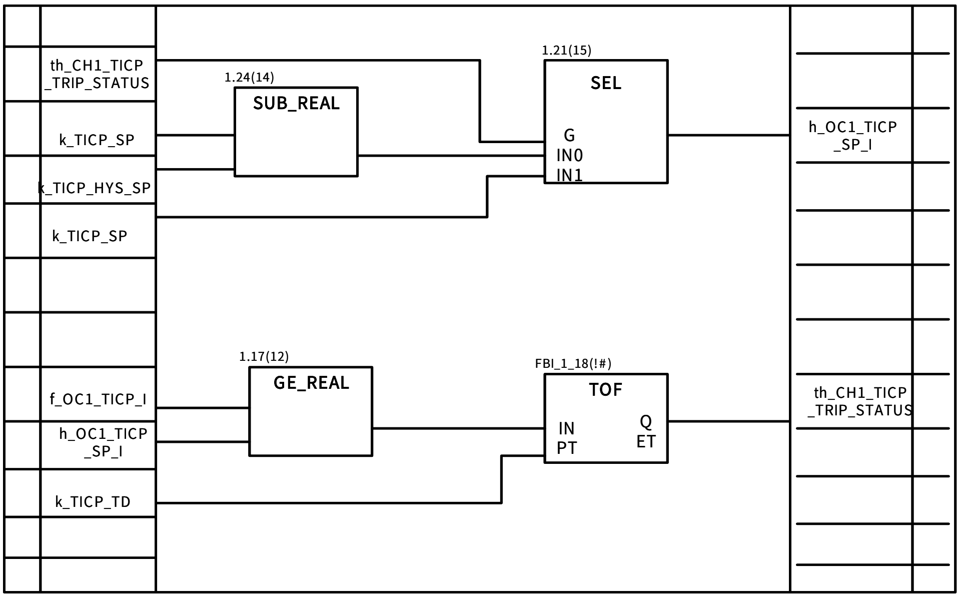

Figure 4. An Example of FBD

FBD-style specification represents the lowest processing level for each software module decomposed in software architecture specification. Also, inputs and outputs information should be from database specification for correctness and consistency of the data in the whole system. The FBD-style specifications will be helpful to implement using PLC programming language because the behavior specification in our work is almost same as PLC programming language. Consequently, programmer can reduce his implementation time and cost with combining design phase and implementation phase.

In NuFDS approach, last specification feature for generating SDS is the hardware configuration. The hardware configuration specification seems to be a minor role in the whole design phase but it can be useful to system developer. Actually, in the nuclear industry, system developer generates documents of the PLC hardware configuration and it is similar to a blueprint. Layout diagram for PLC hardware configuration provides information on the exact arrangement of the hardware components in the racks and cabinets. It will be helpful to decide the needed components and purchase PLC hardware. Figure 5 shows an example of layout diagram for PLC H/W configuration.

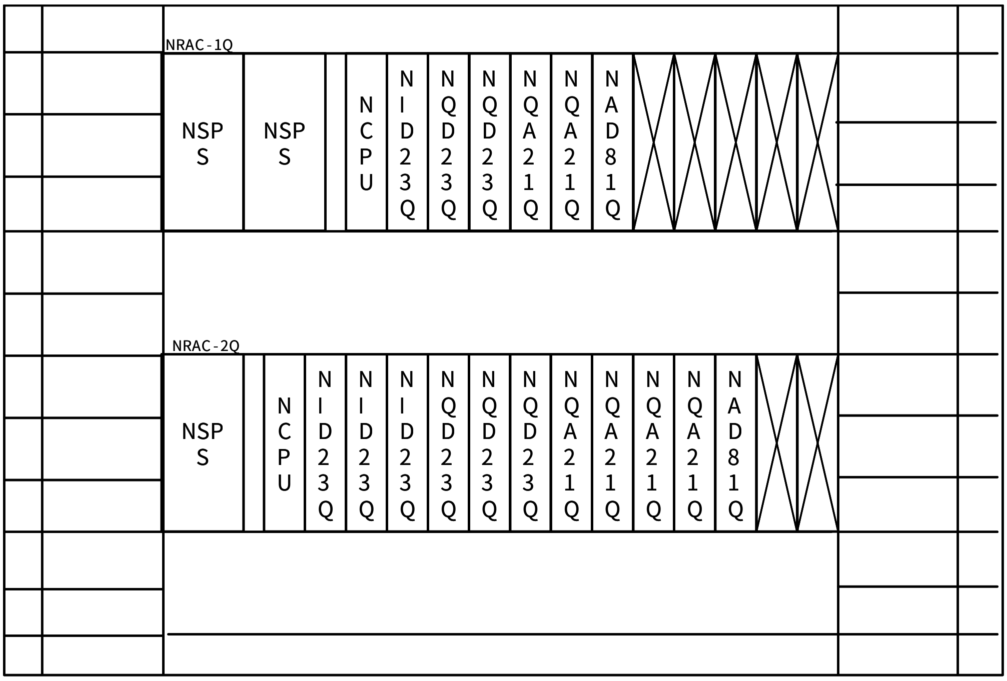

Figure 5. An Example of Layout Diagram for H/W configuration

2.2. Design Verification of the NuFDS Approach

In the NuFDS approach, we are also considering the design analysis techniques based on each design specification feature defined in section 2.1 in order to support above V&V tasks for safety critical software. Figure 6 shows the software design analysis scheme proposed in this paper. According to each specification, basic analysis techniques and formal analysis techniques based on ADL and model checker (SMV) are on considering systematically. A design analysis technique is based on safety-critical software V&V guidelines and various properties such as correctness, completeness, consistency, and traceability could be checked through the formal design analysis.

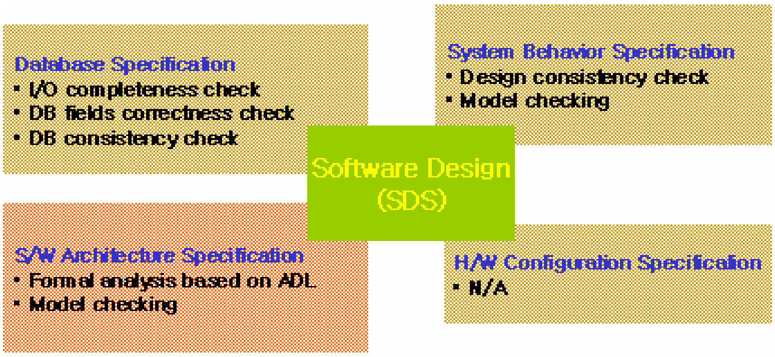
Database Specification
• I/O completeness check
• DB fields correctness check
• DB consistency check
System Behavior Specification
• Design consistency check
• Model checking
Software Design
(SDS)
S/W Architecture Specification
• Formal analysis based on ADL
• Model checking
H/W Configuration Specification
• N/A

Figure 7. Software Design Analysis Scheme

3. The Nuclear Design Specification and Verification (NuSDS) Tool

As mentioned in section 2, the NuFDS approach for the safety critical software based on PLC is proposed. Now, for the tool supporting, we are developing NuSDS tool based on the proposed approach which is the tool, especially for the software design specification in nuclear fields. SDS is a description to show how to create a design which accurately and completely satisfies the behavior and constraints in the SRS. During the coding phase of the software life cycle it is then a relatively simple matter to transform the design into sequences of executable statements written in a particular computer language. Therefore, in this work, an adequate specification technique is needed for the systematic verification and easily translating into implementation phase. Because the SDS can be considered as a blueprint when we build a house, we should have well-formed design features for the right specification. NuSDS tool supports the design specification features of the NuFDS approach for generating the SDS of nuclear systems. It consists of four major specifications; Database, Software Architecture, System Behavior, and PLC Hardware Configuration. SDS could be generated using these four major specifications in NuSDS tool.

Development of NuSDS tool can be divided into two steps. In step 1, NuSDS tool fully supports design specification along the software design specification techniques proposed in the NuFDS approach. And then, based on these design specification, NuSDS tool partially supports design analysis. It means that NuSDS tool can support translating into input language for model checking and help to connect to other V&V tools in step 2. Now, the development of NuSDS tool step 1 is finished and step 2 of NuSDS tool will be added when it will be required for design analysis. Figure 8 shows a simple scratch of NuSDS tool. NuSDS consists of tree-like information window about input/output and function decomposition, software architecture window, FBD-style specification window, layout diagram for PLC hardware configuration, and database window.

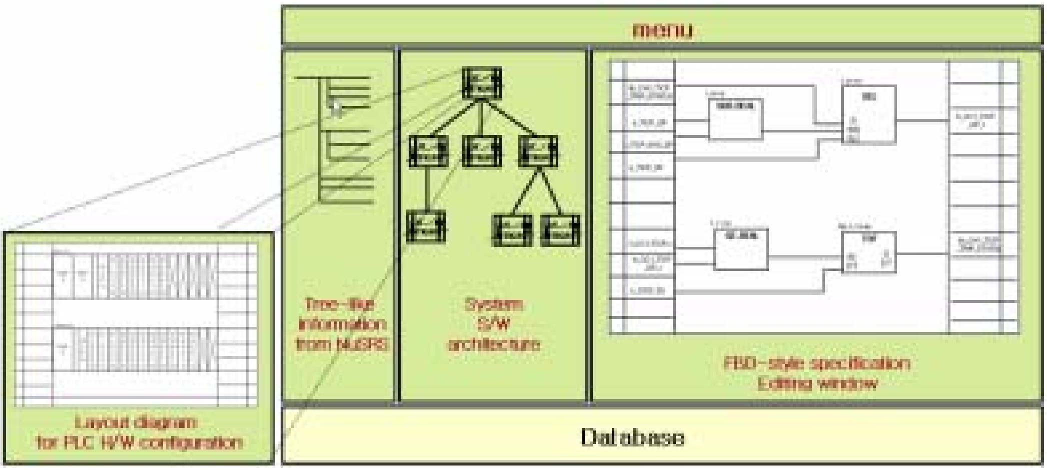
menu
Tree-like information from NuSRS
System S/W architecture
FBD-style specification Editing window
Layout diagram for PLC H/W configuration
Database

Figure 8. NuSDS tool

4. Conclusions

In this paper, a FBD-style design specification and analysis (NuFDS) approach was proposed for generating SDS in nuclear fields. A software design specification of the NuFDS approach consists of database specification, software architecture specification, system behavior specification, and hardware configuration specification. These specifications are suitable for developing SDS of a digital NPP I&C systems specifically based on PLC. It is major advantage that our specification technique is useful for representing consistency between input/output variables, module decompositions, and their interactions. Along these specification features, a design analysis technique is on considering. Consequently, our approach is believed to be a unique and promising software design specification and analysis technique.

5. Acknowledgement

This research was supported by KNICS (Korea Nuclear I&C System) R&D center and NRL project.

6. References

[1] EPRI, *Handbook for verification and validation of digital systems Vol.1: Summary*, EPRI TR-103291, Vol.1, 1994.

[2] IEEE, *IEEE Standard for Software Verification and Validation*, 1998.

[3] Roger S. Pressman, *Software Engineering: A practitioner's approach*, McGRAW-HILL Book Co, 2001.

[4] IEC, *IEC Standard 61131-3: Programmable controllers-Part 3*, IEC 61131, 1993.