Development of FPGA Application Program for Diverse Protection System

Soo Yun Hwang [a*], Yoon Hee Lee [a], Se Do Sohn [a], Jae Hee Yun [a]
[a]*KEPCO E&C Company Inc., 111, Daedeok-daero 989 Beon-Gil, Yuseong-gu, Daejeon, 34057, Korea*
[*]*Corresponding author: syhwang@kepco-enc.com*

1. Introduction

The hardware platforms of the existing nuclear power plant (NPP) contain many components that are becoming obsolete at an increasing rate. Thus, various studies have been conducted to address hardware platform obsolescence [1]. Obsolete analog and digital hardware platforms in NPPs are commonly replaced with the programmable logic controller (PLC) and distributed control system (DCS). Field programmable gate arrays (FPGAs) are highlighted as an alternative to obsolete hardware platforms. FPGAs are digital integrated circuits (ICs) that contain the configurable (or programmable) blocks of logic along with configurable interconnections among these blocks. Designers can configure (or program) such devices to perform a tremendous variety of tasks. FPGAs have been evolved from the technology of programmable logic device (PLD). Nowadays, they can contain millions of logic gates by nanotechnology and can be used to implement extremely large and complex functions that previously could be realized only using the application specific integrated circuits (ASICs) [2].

In this paper, we propose the development of FPGA application program for diverse protection system (DPS) which executes the protective functions in NPP when the protective functions of the plant protection system (PPS) fail. Especially, the hardware description language (HDL) coding, pre-synthesis simulation, logic synthesis, post-synthesis simulation, placement and routing (P&R), static timing analysis (STA), post-P&R simulation, and various tests for the FPGA application program of the DPS are performed in accordance with the requirements of IEC 62566 [3].

2. Diverse Protection System

The DPS is designed, in conformance with the requirements of the 10CFR50.62 [4], to reduce the risk of an anticipated transient without scram (ATWS). The function is required to initiate protective action during an anticipated operational occurrence that is followed by a failure of the reactor trip portion of the PPS. In addition, the DPS is designed to comply with the requirements of the SECY-93-087 [5] and includes a function to mitigate the effects of a potential common cause failure of the digital computer logic within the PPS, concurrent with design basis events.

The DPS is a two channel, non-safety related system. It consists of the measurement channels, trip recognition logic, coincidence logic, and initiation circuitry. The DPS is designed to perform its function in a reliable manner. The DPS is diverse from and independent of the PPS. Satisfaction of these functions is accomplished by the reactor trip, auxiliary feedwater system (AFWS) actuation, turbine trip, and reactor trip on turbine trip.

2.1 Reactor Trip

The DPS automatically initiates a reactor trip when either pressurizer pressure or containment pressure exceeds its respective setpoint. Two pressurizer pressure channels are monitored, one in each DPS channel. The DPS initiates a reactor trip on a 2-out-of-2 coincidence of the inputs reaching a trip condition. Two containment pressure channels perform the same function as two pressurizer pressure channels. A DPS reactor trip causes interruption of power to the control element drive mechanisms (CEDMs). Power is interrupted by opening the CEDM motor generator set (MG Set) output contactors.

Controls are provided to permit the operator to manually trip the reactor from the main control room (MCR). Two controls are provided, one for each DPS channel. The DPS initiates a reactor trip on a 2-out-of-2 coincidence of the inputs reaching a manual reactor trip condition. Manual reactor trip also causes the MG Set output contactors to open.

2.2 AFWS Actuation

The DPS automatically initiates auxiliary feedwater when steam generator level falls below a pre-determined value. Four steam generator level channels are monitored, two channels from each steam generator. Each DPS channel monitors two steam generator levels, one level channel on each steam generator. The DPS initiates auxiliary feedwater actuation signals (AFASs) for a steam generator when 2-out-of-2 low level trip coincidence occurs in that steam generator.

The DPS AFASs are sent to the AFWS component controllers. The DPS AFASs cause actuation of the AFWS pumps and valves on a system level if the AFAS actuation portion by the PPS has not occurred.

2.3 Turbine Trip

The DPS turbine trip is indirectly initiated whenever the DPS reactor trip has been actuated. A turbine trip signal is generated from the control element drive mechanism control system (CEDMCS) when the MG Set's power to the CEDMCS has been interrupted.

2.4 Reactor Trip on Turbine Trip

The DPS automatically initiates a reactor trip when a turbine trip has been detected. It is expected to be used when the reactor power cutback system is out of service. The turbine trip inputs to the DPS are a digital (contact) type. Two signals are provided, one to each of the two DPS channels. The DPS initiates a reactor trip on a 2-out-of-2 coincidence of these inputs.

The reactor trip on turbine trip function has the capability of being manually enabled and disabled from the MCR.

3. Development of FPGA Application Program

The protective actuation of the DPS is performed by the application program. We applied the development process based on the requirements of IEC 62566 to develop the FPGA application program for the DPS.

3.1 Design of FPGA Application Program

Figure 1 shows the hierarchy of the FPGA application program.

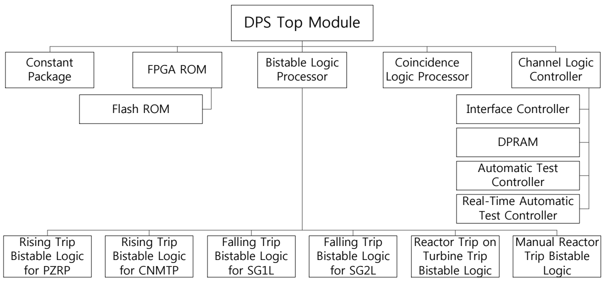
DPS Top Module
Constant Package
FPGA ROM
Bistable Logic Processor
Coincidence Logic Processor
Channel Logic Controller
Flash ROM
Interface Controller
DPRAM
Automatic Test Controller
Real-Time Automatic Test Controller
Rising Trip Bistable Logic for PZRP
Rising Trip Bistable Logic for CNMTP
Falling Trip Bistable Logic for SG1L
Falling Trip Bistable Logic for SG2L
Reactor Trip on Turbine Trip Bistable Logic
Manual Reactor Trip Bistable Logic

Fig. 1. FPGA application program hierarchy.

The FPGA application program (DPS Top Module) consists of the Constant Package, FPGA ROM, Bistable Logic Processor (BLP), Coincidence Logic Processor (CLP), and Channel Logic Controller (CLC) as shown in Figure 1.

The declarations for the constants and user-defined types of the FPGA application program are defined in the Constant Package. The constants and user-defined types defined in the Constant Package are only used for the logic synthesis of the HDL codes. Therefore, the Constant Package is not synthesized as the gate level netlist.

The setpoint and hysteresis for trip and pretrip of the parameters are stored in Flash ROM within the FPGA ROM. In the FPGA based equipment, the setpoint and hysteresis cannot be changed through the maintenance and test panel (MTP) like the PLC based existing equipment. The setpoint and hysteresis can be changed only through JTAG cable using the vendor development tool. In addition, the cyclic redundancy check codes are utilized to confirm the integrity of the Flash ROM.

The BLP includes the comparators which determine the trip and pretrip status of parameters. There is a comparator per each parameter. The comparator compares converted digital value with a pre-defined value to determine whether or not the sensed parameter has exceeded. There are six parameters in the DPS: pressurizer pressure (PZRP), containment pressure (CNMTP), steam generator 1 level (SG1L), steam generator 2 level (SG2L), reactor trip on turbine trip, and manual reactor trip. The BLP is composed of the Rising Trip Bistable Logic for PZRP, Rising Trip Bistable Logic for CNMTP, Falling Trip Bistable Logic for SG1L, Falling Trip Bistable Logic for SG2L, Reactor Trip on Turbine Trip Bistable Logic, and Manual Reactor Trip Bistable Logic as shown in Figure 1. The bistable logic for PZRP, CNMTP, SG1L, and SG2L is based on the fixed setpoint. The bistable logic for Reactor Trip on Turbine Trip and Manual Reactor Trip is based on the binary trip bistable logic.

The CLP finally determines a reactor trip signal based on the trip signals of PZRP, CNMTP, Reactor Trip on Turbine Trip, and Manual Reactor Trip and a safeguard initiation signal based on the trip signals of SG1L and SG2L, using 2-out-of-2 coincidence logic between two channels of the DPS.

The CLC performs the input and output interface controls between the FPGA application program and the other programs installed in the other FPGAs using the Interface Controller and DPRAM. It also controls the data flows for normal operation of the DPS. Address and data parity bits, and checksum are used to confirm the integrity of the data communication between the FPGAs. In addition, the FPGA application program uses the heartbeat data to check the operation status. The Automatic Test Controller and Real-Time Automatic Test Controller in CLC are operated to verify the integrity of the FPGA application program. The Automatic Test Controller is performed by the operator's request using the MTP and checks the integrity of the BLP, CLP, and a part of CLC as well as a part of input and output channels. The Real-Time Automatic Test Controller executes the self-diagnosis function and checks the integrity of the BLP, CLP, and a part of CLC. The Real-Time Automatic Test Controller is performed every 20 ms and the test results are reported as the MTP.

We categorized the FPGA application program of Figure 1 into the three modules such as a top module, sub module, and basic module for more efficient development and verification. The top module is a DPS Top Module and the sub module is a FPGA ROM, BLP, CLP, and CLC. The other modules in Figure 1 are classified as the basic module.

3.2 Development Process of FPGA Application Program

We applied IEC 62566 to develop the FPGA application program of the DPS. Figure 2 shows the development life-cycle based on IEC 62566.

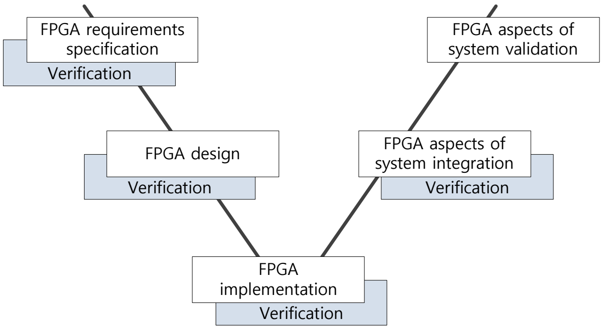
FPGA requirements specification
Verification
FPGA aspects of system validation
FPGA design
Verification
FPGA aspects of system integration
Verification
FPGA implementation
Verification

Fig. 2. Development life-cycle of FPGA application program.

A FPGA requirements specification shall document the requirements of the FPGA application program, either in the document itself or by referencing sets of requirements stated at system or subsystem level (e.g. the functional behavior to be implemented). In this stage, we derived the requirements of the FPGA application program from the system specification of the DPS and documented the software requirements specification (SRS) for the FPGA application program. The SRS is used as the source document for the FPGA design. The SRS contains the requirements of the FPGA requirements specification which are pertinent to the hardware program. The SRS is documented in accordance with IEEE Standard 830 [6].

Starting from the FPGA requirements specification, the FPGA design initially aims at defining major choices such as the decomposition into modules (application-specific or pre-developed), the operation of the defensive design, as well as the identification of needed micro-electronic technologies (including their native blocks) and pre-developed blocks. Then a register transfer level (RTL) description is built using HDLs. In this stage, the major design specifications are determined and documented as the software design description (SDD) for the FPGA application program. The SDD is documented in accordance with IEEE Standard 1016 [7]. The SDD describes how the software will be structured to satisfy the requirements identified in the SRS. It is a translation of requirements into a description of the hardware program structure, hardware program components, interfaces, and data necessary for implementation. The SDD consists of the descriptions of decomposition, dependency, interface, and detail information.

Starting from the FPGA design, the FPGA implementation synthesizes the gate level description (netlist) of the FPGA application program. Then P&R is performed and results in the physical description needed to produce the FPGA application program, such as programming file or bitstream image file. In this stage, the source files and test-bench for the FPGA application program are created according to the HDL coding standards and SDD, and then pre-synthesis simulation, logic synthesis, post-synthesis simulation, P&R, STA, and post-P&R simulation are performed sequentially.

The FPGA application program is implemented with a synthesizable RTL VHDL targeting an Actel ProASIC3 FPGA (A3P1000-256FBGA) and the Synopsys Synplify Pro is utilized for the logic synthesis. The Libero IDE vendor development tool [8] is used to perform P&R and STA. Only top module of the FPGA application program performs the logic synthesis, P&R, and STA. Through the logic synthesis, P&R, and STA, we can measure the area occupation ratio and clock period of the top module on FPGA. Table I shows the implementation results. The core means the basic unit of area in Actel FPGA. By virtue of Table I, we identified that the physical requirements for the FPGA chip are satisfied.

Table I: Implementation results of FPGA application program

Measurement Items	Results
Occupation ratio of core	63.81 %
Occupation ratio of I/O	25.99 %
Occupation ratio of RAM/FIFO	12.50 %
Occupation ratio of FlashROM	100 %
Clock period	40.725 MHz

In addition, the Modelsim SE simulator [9] is used to perform the various simulations. Figure 3 shows a part of the pre-synthesis simulation results.

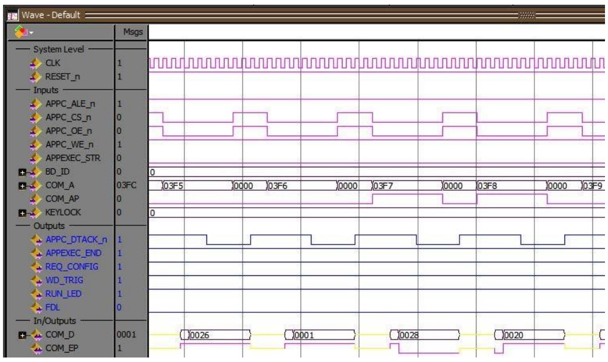

Fig. 3. The waveform of the pre-synthesis simulation results.

We performed the pre-synthesis simulation for all modules and the post-synthesis/-P&R simulations for only top module. The pre-synthesis simulation is performed to satisfy 100 % code coverages including statement, branch, condition, expression, and finite state machine coverages, and it is verified that the results of the post-synthesis simulation are the same as those of the pre-synthesis simulation and the results of the post-P&R simulation are the same as those of the post-synthesis simulation. By the simulation results, we verified that the functional requirements of the FPGA application program are satisfied.

The procedures for the simulations, logic synthesis, P&R, and STA are documented as the module test procedure. The module test procedure also includes the test cases to ensure the compliance with the SDD requirements. The test cases are documented in accordance with IEEE Standard 829 [10]. The results of the module test such as simulation results, area occupation ratio, clock period, and satisfaction of the test cases are documented as the module test report.

We performed the unit test after downloading the FPGA application program to the FPGA chip. The unit test procedure includes the test cases to ensure the compliance with the SRS requirements and the results of the unit test are documented as the unit test report.

The process of the FPGA system integration is the combining of the verified hardware and software components into subsystems and finally into the complete system. This process consists of two kinds of activities: the system integration and integrated system verification. In this stage, we performed the integration test after downloading the MTP application program to the MTP computer. The integration test procedure contains the test cases to ensure the compliance with the SRS and system design requirements as well as the interface requirements between the FPGA application program and the MTP application program. The results of the integration test are documented as the integration test report.

The verification activities undertaken as part of the development are undertaken by staff independent of those performing the design and implementation. The most appropriate way is to engage a verification team. In addition, verification activities may be undertaken as part of a third party assessment of the FPGA and of its development process in order to provide assurance that it will meet its targets. In these stages, the requirement, design, implementation, and test phase verification and validation (V&V) reports are documented by the verification team who does not participate in the development of the FPGA application program. The aforementioned V&V reports are used as the source documents for the final V&V report.

The FPGA application program is typically validated within the system validation phase. The testing shall be performed to validate the system and the FPGA application program in accordance with the system design requirements. The validation tests shall be performed on the system in its final assembly configuration including the final version of the FPGA application program. In this stage, the V&V test is performed by the verification team. The V&V test procedure includes the test cases to ensure the compliance with the requirements of the system design and SRS, and the results of the test are documented as the V&V test report. Finally, the final V&V report is documented by the verification team. The final V&V report includes the results of the V&V test as well as the requirement, design, implementation, and test phase V&V.

4. Conclusions

This paper presents the development of the FPGA application program for the DPS which executes the protective functions in NPP when the protective functions of the PPS fail. In particular, we applied IEC 62566 to develop the FPGA application program. The HDL coding, pre-synthesis simulation, logic synthesis, post-synthesis simulation, P&R, STA, post-P&R simulation, module test, unit test, integration test, and V&V test of the FPGA application program are performed in accordance with the development requirements of IEC 62566. The FPGA application program is applied to the replacement of DPS cabinet assemblies at Hanbit NPP units 3, 4, 5 & 6 and Hanul NPP units 3, 4, 5 & 6. Installation of the new DPS cabinet assemblies is completed in sequence with the approval of the Korean nuclear regulatory body, Korea Institute of Nuclear Safety. Their working results have been successful so far. In the DPS cabinet assemblies based on the FPGA, the central processing units (CPUs) and operating systems (OSs) do not exist. Therefore, the FPGA based DPS has the diverse platform compared with the PPS based on CPUs and OSs. We expect that the FPGA based platform can be applied to the safety related system or new reactors such as small modular reactor, sodium-cooled fast reactor, etc.

REFERENCES

[1] J.P. Rooney, Aging in Electronic Systems, Reliability and Maintainability Symposium, pp.293-299, 1999.

[2] Mentor Graphics Corporation and Xilinx, Inc., The Design Warrior's Gide to FPGAs: Devices, Tools and Flows, 2004.

[3] IEC 62566, Nuclear Power Plants - Instrumentation and Control Important to Safety - Development of HDL-programmed Integrated Circuits for Systems Performing Category A Functions, 2012.

[4] 10CFR50.62, Requirements for Reduction of Risk from Anticipated Transients Without Scram (ATWS) Events for Light-Water-Cooled Nuclear Power Plants, 2007.

[5] Staff Requirements Memorandum on SECY-93-087, Policy, Technical, and Licensing Issues Pertaining to Evolutionary and Advanced Light-Water Reactor (ALWR) Designs, Item II.Q, Defense Against Common-Mode Failures in Digital Instrument and Control Systems, 1993.

[6] IEEE Standard 830, IEEE Standard for Recommended Practice for Software Requirements Specifications, 1998.

[7] IEEE Standard 1016, IEEE Standard for Information Technology - Systems Design - Software Design Descriptions, 2009.

[8] Microsemi Corporation, Actel Libero IDE User's Guide, v9.1.

[9] Mentor Graphics Corporation, Modelsim SE User's Manual, v10.0d.

[10] IEEE Standard 829, IEEE Standard for Software and System Test Documentation, 2008.