Design Verification Enhancement of FPGA-based Plant Protection System Trip Logics for Nuclear Power Plant

Ibrahim Ahmed [a,b*], Jaecheon Jung [a], Gyunyoung Heo [b]
[a]*Deparment of Nuclear Power Plant Engineering, KEPCO International Nuclear Graduate School, Ulsan, Korea*
[b]*Department of Nuclear Engineering, Kyung Hee Uninersity, Giheung-gu, Yongin-si, Gyeonggi-do 446-701, Korea*
[*]*Corresponding author: ibroah777@yahoo.com*

1. Introduction

The challenges such as fast obsolescence, the vulnerability to cyber-attack, and other related issues of microprocessor-based software systems have currently led the nuclear stakeholders to consider field programmable gate array (FPGA) as an alternative to software systems because of its hardware related benefits. However, the FPGA needs to be configured which involves the development and the verification and validation (V&V) of HDL codes. As part of strengthening the application of FPGA technology and find solution to its challenges in NPPs, international atomic energy agency (IAEA) [1] has indicated interest by joining sponsorship of Topical Group on FPGA Applications in NPPs (TG-FAN) that hold meetings up to 7th times until now, in form of workshop (International workshop on the application of FPGAs in NPPs) annually since 2008. The workshops attracted a significant interest and had a broad representation of stakeholders such as regulators, utilities, research organizations, system designers, and vendors, from various countries that converge to discuss the current issues regarding instrumentation and control (I&C) systems as well as FPGA applications. Two out of many technical issues identified by the group are lifecycle of FPGA-based platforms, systems, and applications; and methods and tools for V&V. Therefore, in this work, several design steps that involved the use of model-based systems engineering process as well as MATLAB/SIMULINK model which lead to the enhancement of design verification are employed.

Three general types of the bistable setpoint algorithms in nuclear power plant – fixed, variable manual reset, and variable automatic rate limiting setpoints, are designed in this work. During the design, variable overpower trip (VOPT) and low pressurizer pressure trip (LPPT) setpoint parameters are respectively chosen for the design of variable automatic rate limiting and variable manual reset setpoints algorithms. Also, to test for the design of the fixed setpoint algorithm, the high steam generator water level is used.

2. Methodology

In this section the approaches used for the design which enhanced the design verification are briefly described. In this paper, it is assumed that the needs and requirements analysis has been performed, which are not discussed here in this paper.

2.1 EFFBD Model

Before going into the detailed algorithm design and development, the block diagram of the PPS bistable trip logic functions are first modelled using the Vitech's CORE® Model Based Systems engineering software in order to verify the flow of the functional execution of the system. To do the modelling, enhanced functional flow block diagram (EFFBD) model is employed. EFFBD provides a basis for establishing a process model which represents sequential, parallel, and decision logic used to depict system behaviours. Fig. 1 and fig. 2 shows the EFFBD model of VOPT and LPPT respectively.

2.2 FSMD Model

After the EFFBD models, the design of algorithms are then performed using finite state machine with data path (FSMD) modelling techniques. FSMD is an architectural model that supports a structured design approach. Following such a structured approach can leads to design simplification and speeds up the design procedure as well as enhances the design verification. This can also make the final design easy to be verified and validated, hence reducing the V&V load.

There are two major design blocks for each of the algorithms – Data path and FSM (finite state machine) controller. In designing each of the trip function using the FSMD approach, the following design steps are adopted:

- ✓ *Interface definition of the overall FSMD for the trip function design*
- ✓ *Data path design*
  - Data path structural design
  - Interface definition of the data path
- ✓ *FSM controller design*
  - Interface definition of the FSM controller
  - FSM structural design and its transition diagram.

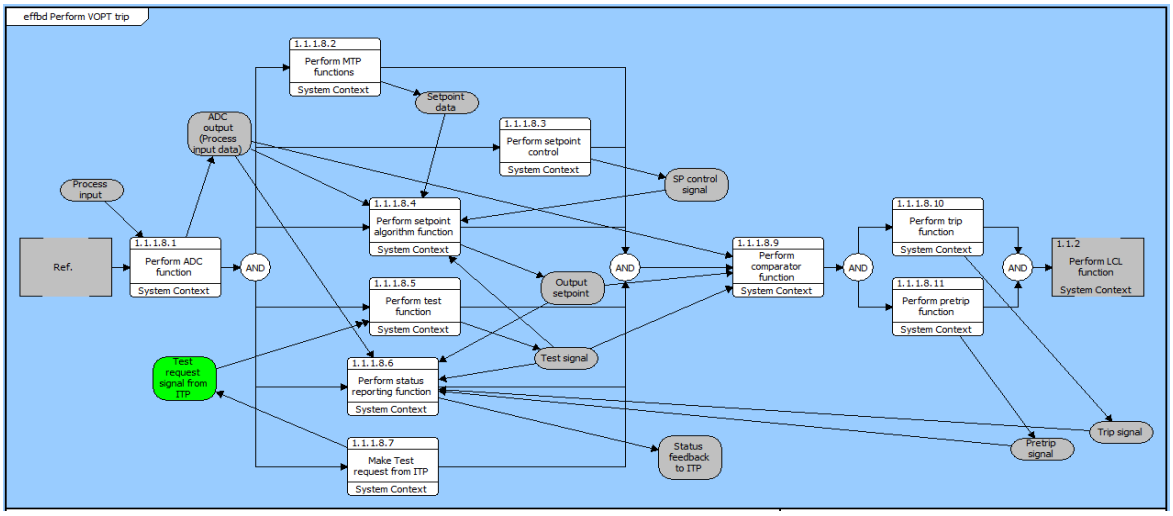

Fig. 1. PPS bistable VOPT EFFBD model from Vitech CORE tool.

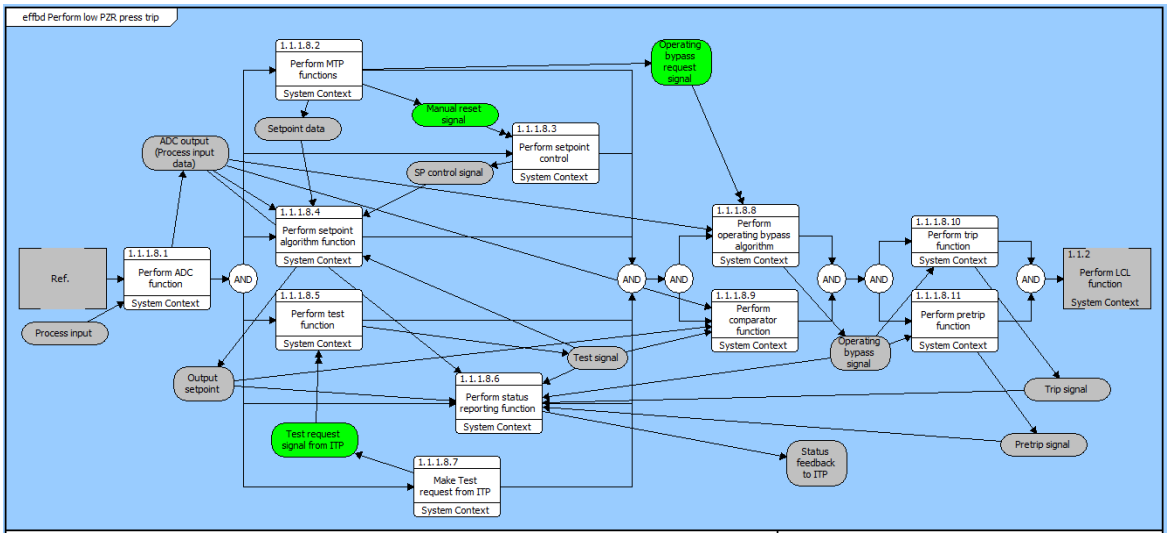

Fig. 2. PPS bistable LPPT EFFBD model from Vitech CORE tool.

2.3 VHDL coding

Having developed the design using FSMD technique, the next stage is to implement the developed FSMDs by developing HDL code for all the three types of algorithms. This involves writing the register transfer logic (RTL) that will be implemented on FPGA using any of the hardware description languages (HDLs). The widely use HDL languages are Very high speed integrated Hardware Description Language (VHDL) and Verilog. In this work, VHDL is chosen and used because of its flexibility and unique features.

VHDL is a hardware description language that is used to describe the behaviour of the digital system in electronic design. It is a language widely used to model and design digital hardware. Among its unique feature is design reusability [2] which allows procedures and functions to be placed in a package so that they are available to any design unit that uses them. This is impossible in Verilog because there is no concept of packages in Verilog. VHDL also has some features such as configuration, generate and package statements, together with the generic clause, that help the designer to manage large designs; whereas in Verilog, there are no such statements.

The Active-HDL software developed by Aldec is used for writing, simulating, and synthesizing of the VHDL code. Active-HDL's Integrated Design Environment (IDE) includes a full HDL and graphical design tool suite and RTL/gate-level mixed-language simulator for rapid deployment and verification of FPGA designs [3].

2.4 Design Verification and Validation

This section discussed V&V that are performed in this work. The developed VHDL code for the trip algorithms are verified and validated. Firstly, the functional correctness of the design and timing response of the system is verified using VHDL simulator from Active-HDL tool. The test bench approach is used to perform the functional verification.

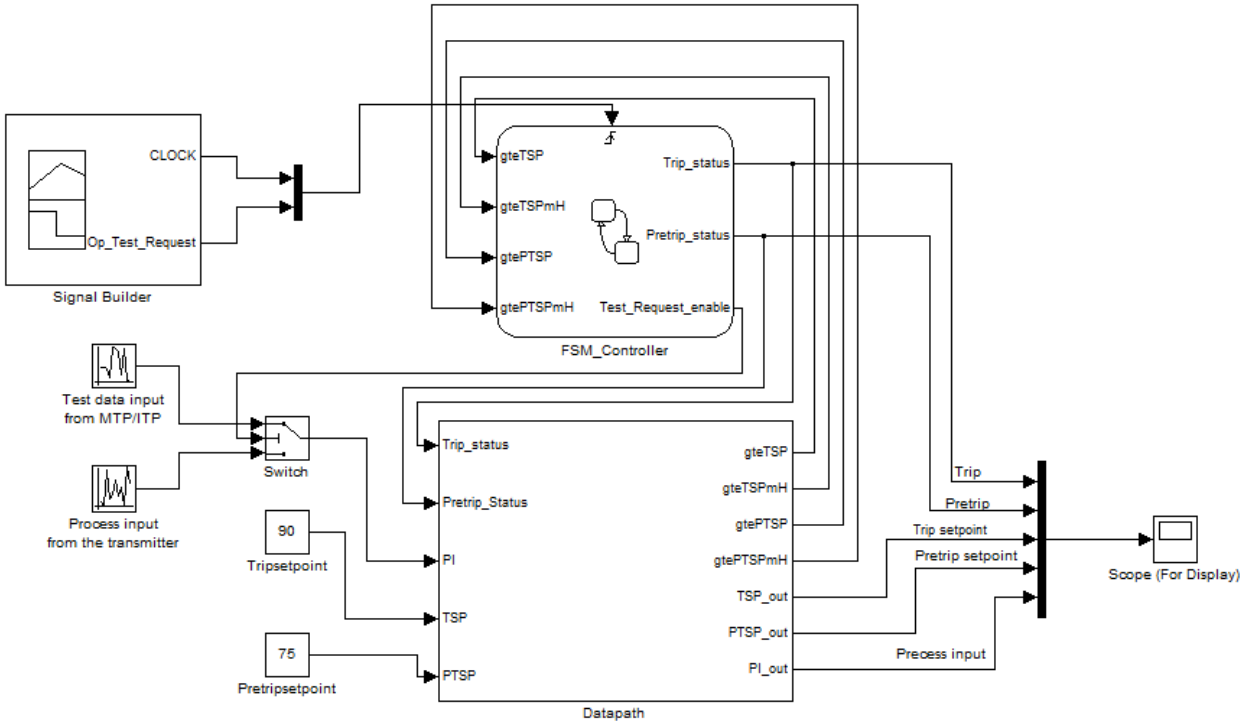

Fig. 3. Simulink model of FSMD for fixed setpoint algorithm.

To further enhance the design verification process, MATLAB/SIMULINK and StateFlow are used to verify the FSMD models in order to ensure that the FSMD models are properly developed. Fig. 3 shows the SIMULINK model with the embedded StateFlow chart for the fixed setpoint algorithm. Lastly after design verification, the design is validated practically using Xilinx Spartan3E-100 CP132 FPGA on Digilent Basys2 Board and the code is downloaded into the FPGA for testing and validation of the final design.

There are several effective design languages to verify the VHDL code using test bench. One of them is SystemVerilog [4]. To use the SystemVerilog language, the verification engineer has to use the verification tool that supports the SystemVerilog. The tool that most verification engineers use to write the test bench code for design verification is QuestaSim™ software verification developed by Mentor Graphics because it supports SystemVerilog. However, the test bench can also be written in VHDL to test for the logic correctness of the design using VHDL simulators. VHDL simulators normally offer some interactive stimuli capture feature. In this work, the Active-HDL tool simulator environment is as well used to write the test bench code for design verification.

3. Conclusions

The design of FPGA-based plant protection system (PPS) in nuclear power plant requires guidelines and methodology to achieve a satisfactory design results. While the regulators are interested in the effective and efficient design verification and validation. In this work, to enhance the design verification, several design strategies that involved the use of model-based systems engineering process are discussed. The verified and validated design output works correctly and effectively.

Conclusively, the model-based systems engineering approach and the structural step-by-step design modeling techniques including SIMULINK model utilized in this work have shown how FPGA PPS trip logics design verification can be enhanced. If these design approaches are employ in the design of FPGA-based I&C systems, the design can be easily verified and validated.

REFERENCES

[1] IAEA, Application of Field Programmable Gate Arrays (FPGAs) in Instrumentation and Control Systems of NPPs, (2013). https://www.iaea.org/NuclearPower/News/2013/2013-02-15-npe.html (accessed February 20, 2016).

[2] D.J. Smith, HDLChip Design - A practical guide for designing, synthesizing and simulating ASICs and FPGAs using VHDL or Verilog, 1st ed., Doone Publications, Madison, Ali, USA, 1996.

[3] Aldec, Active-HDL - FPGA Simulation - Products - Aldec, (n.d.). https://www.aldec.com/en/products/fpga_simulation/active-hdl (accessed February 29, 2016).

[4] C. Spear, SystemVerilog for Verification: A Guide to Learning the Testbench Language Features, 2nd ed., Springer, New York, USA, 2008.