# A Study on FPGA Verification and Validation Base on Schematic Design

Dongil Lee [a*]
[a]*KHNP, Central Research Institute, 70, 1312-gil, Yuseong-daero, Yuseong-gu, Daejeon, 34101, South Korea*
[*]Corresponding author: diturtle@khnp.co.kr

## 1. Introduction

Nuclear power plant (NPP) control equipment include many Field Programmable Gate Array (FPGA) and Complex Programmable Logic Devices (CPLD). In particular, FPGAs are essential for controllers with backplane-based designs.

The following two functions are difficult to implement in mass-produced electronic devices, and FPGAs are almost essential when they are required to be manufactured and designed by user's request.

First, BUS communication is mandatory on a back-plane basis, and there is case must implement BUS-master and BUS-slave function. Secondly, electronic cards (especially CPU module) may need to control memory (SRAM, FLASH, EPROM, etc.) and large amounts of electronic devices (ADC, DAC, Ethernet Controller, etc.).

FPGAs have been widely used in electronic equipment, but verification is inadequate due to the many concepts of electronic devices. Recently, a lot of research has been done on verification and the regulatory body has established regulatory guidelines.

FPGA design is designed using VHDL (VHSIC Hardware Description Language) or Verilog-HDL which is a kind of HDL (Hardware Description Language). If the FPGA is designed in HDL, the verification method is well formalized, as shown in the International Electrotechnical Commission (IEC), Nuclear Regulatory (NUREG), and the Electric Power Research Institute–Technical Report (EPRI-TR).

There are many schematic designs in facilities before the 90s, and schematic designs are often found in FPGAs that are being developed recently. The reason is that it depends on the developer's ability to implement. Of course, it would not have been possible to design a schematic if HDL was constrained in the concept or requirements phase.

Schematic design can be a very beneficial design method for existing circuit designers. However, it is very dependent on the vendor (FPGA vendor) in development and very limited in verification and testing.

In this paper, we will study the verification method of Schematic design in safety and non-safety system, not the verification problem in HDL design.

## 2. Comparison of Schematic and VHDL Designs

This section describes the differences between HDL and Schematic designs.

### *2.1 HDL Design*

HDL designs use VHDL and VerilogHDL the most.

The use of HDL enables vendor-free design and enables asset utilization (IP, Intellectual Property) of the design. Validation during design and system level verification is also possible. It has the advantage of easy setting of input and checking output result. Finally, the design period can be shortened by logic synthesis.

However, programs other than those provided by manufacturers are expensive, and there are circuits that cannot be logic synthesized by ASICs. (Use compiled cell for ROM, RAM, etc.)

### *2.2 Schematic Design*

Schematic design is a useful design approach for circuit designers in the past through Transistor Transistor Logic (TTL) and Complementary Metal Oxide Semiconductor (CMOS).

It consists of 74 series libraries provided and logical gates such as AND-Gate and OR-Gate, and designs using graphic such as CAD such as output port.

However, unlike the intuitive HDL, Schematic is very electronic and can only use the symbols provided by the manufacturer, making it highly dependent and limited in verification.

For example, if a conditional statement (case or if statement) in HDL is implemented in Schematic, it must be implemented using Symbol provided by the manufacturer or MUX and logic gate. Logic gates are also supplied by the manufacturer.

Schematic design becomes very difficult when verifying or analyzing. Many logics need to be grouped to analyze the implemented logic, and various connections are required for verification.

## 3. Schematic Verification Method

Verification of the Schematic design is similar to the verification method of the HDL design, but several options are required to justify the verification.

This section describes how to verify FPGAs in Schematic designs.

### *3.1 FPGA Design and Verification Method*

Since the HDL designing the FPGA has software characteristics, it is verified using the software Verification and Validation (V&V) verification method.

Fig. 1. shows the V-model presented by IEC62566, which is reflected by domestic and international regulators, manufacturers and power generation companies. [1]

To summarize the method proposed in IEC62566, the existing SoC verification method is performed during the detailed design and implementation phase of the V-model, which is a software verification method.

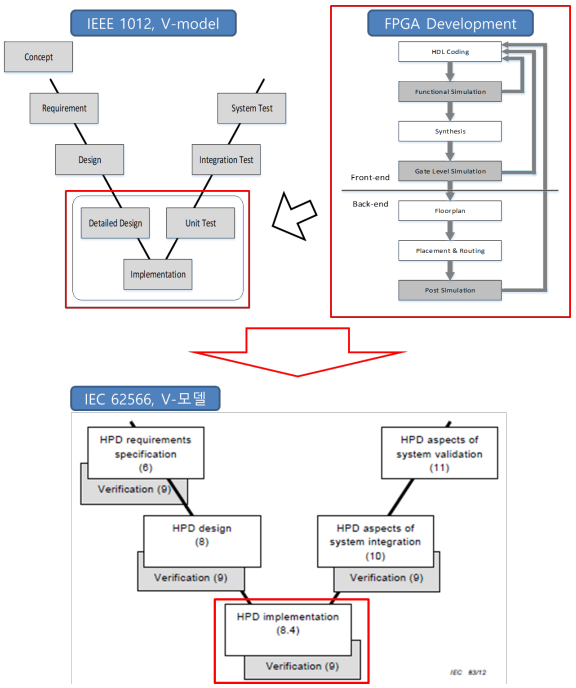


Fig. 1. V-model of IEC62566 Summary

IEC62566 is considered the most reasonable way to develop and verify an FPGA.

### *3.2 Verification Differences Between Schematic and HDL Designs*

Unlike the verification of HDL designs, the schematic design should consider the following:

a. Depends on the manufacturer
b. Symbol is Black box, not White box.
c. Verification of symbol is difficult to confirm.
d. Test Bench connection through transformation is necessary for verification.
e. There is a difference between RTL after compiling about Library HDL and Symbol of manufacturer.

Considering the above, additional verification of symbol should be added in the existing verification method. [2]

### *3.3 Additional Verification for Schematic Design*

It is important to justify schematic design by additional verification of all symbols used in Schematic design.

To justify the symbol in the schematic design, two additional verifications are needed for the symbol and the manufacturer's HDL:

a. Functional and Timing simulations to evaluate equivalence
b. Evaluation of equivalence when converted to RTL (Resister Transfer Level)

The two tests yield slightly different results. However, you can use it if it proves to be no problem with Timing and logic.

Fig. 2. shows an example in which there is a difference in the conversion of RTL, but it is not a logical problem. This implementation also makes a slight difference in timing. However, this does not affect the system.

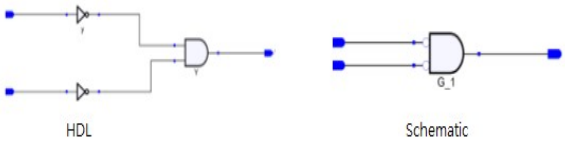


Fig. 2.RTL transformation differences between Schematic and HDL for A'& B'

## 4. Conclusions

There is a difference between HDL and Schematic design in FPGA design and verification.

The biggest difference is in the dependency of the manufacturer and the intuition of the implementation. And there are more differences in verification.

Schematic design has a different view and character than the verification method in HDL design. The verification of FPGAs is software specific, so it is accompanied by software V & V verification and hardware verification.

However, the schematic design has a small software characteristic and therefore requires further verification of the provided symbols.

Schematic design requires verification of the timing and function between the used symbol and the library of the symbol, and the evaluation of RTL equivalence between the symbol and the library.

## REFERENCES

[1] IEC, Nuclear power plants – Instrumentation and control important to safety – Development of HDL-programmed integrated circuits for systems performing category A functions, 2012.
[2] D.I. Lee, FPGA Design and Verification, KHNP-CRI Technical Report, 2018.