Improvement of Hardware Version Management Method for POSAFE-Q PLC

Sungwook Kim [a*], Min-Hyuk Yoon [a], Chi-Uk An [a], Kwan-Woo Yoo [a], Dong-Yeon Lee [a]
[a] *SOOSAN ENS Co., Soosan Building 2F, 13, Bamgogae-ro 5-gil, Gangnam-gu, Seoul, Korea, 06367*
[*] *Corresponding author: seong2844@soosan.co.kr*

Keywords : Nuclear Power Plants, Safety Programmable Logic Controller, Safety PLC, POSAFE-Q

1. Introduction

Nuclear power plants are composed of various hardware and software components, forming complex systems. Safety and reliability are important because complex systems must operate well as one. So, Verification and Validation (V&V) of hardware and software used in nuclear power plants are important [1,2]. Accordingly, hardware and software version must be managed well. Sometimes, POSAFE-Q encounters component discontinuation issues [3,4]. Therefore, manufacturer needs to find replacements and must change the hardware version. Existing POSAFE-Q source code contains the hardware version. Due to aforementioned issues, POSAFE-Q is necessary to perform software V&V again when component discontinuation issues occurred which needs simple component replacement. Although, there is no change to the software except for the hardware version management part in this case. However, despite this, software V&V is still required. Even a simple hardware replacement can cause significant time loss and cost issues because of software V&V again. These issues need to be resolved quickly.

In this paper, improved hardware version management method was introduced. This method can save considerable time loss and cost issues associated with performing software V&V due to simple hardware changes.

2. Methods and Results

In this section some of the methods used to improve hardware version management is introduced. In brief, describing the improvement of hardware version management part is followed by a simple example to aid understanding. Then, showing an application to POSAFE-Q.

2.1 Improved Hardware Version Management Method

Existing POSAFE-Q has the hardware version coded into the source code which needs software V&V. However, POSAFE-Q with an improved hardware version management method doesn't require software V&V for simple hardware changes because of hardware setting part. In Fig. 1 (A) shows that although the same software is used, the hardware version may be applied differently. As these things become possible, when there is a simple hardware change such as a discontinuation issue, the hardware version can manage just by changing the hardware without touching the software.

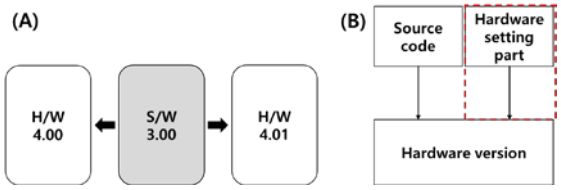

Fig. 1. Differences between before and after applying improved hardware version management method.

Fig. 1 (B) indicated that the differences between before and after applying improved hardware version management method. The red dotted box represents the hardware setting part that was added for improved hardware management. The hardware setting part includes improvement of hardware and software. The detailed explanation will be given in the next chapter.

2.2 Software Improvements for Improved Hardware Management Method

```
Major_HW_VERSION = 4.00 (constant)
Minor_HW_VERSION = 0111 (2)

highBits = Minor_HW_VERSION[3:2] << 2
lowBits = Minor_HW_VERSION[1:0]

HW_VERSION = Major_HW_VERSION + highBits + lowBits
```

Fig. 2. Example of applied source code using improved hardware management method

Set the major hardware version as constant. The major hardware version is fixed into source code which means software V&V is necessary when it changes. The minor hardware can be modified through the hardware setting part. In minor hardware version, the upper two bits are combined with the first decimal place of the hardware version, and the lower two bits, operate with the second decimal place of the hardware version. So, that, the binary number 0111 can be converted to 0.13. Hardware version are made up major and minor hardware version. In this case, hardware version shows 4.13.

2.3 Hardware Improvements for Improved Hardware Management Method

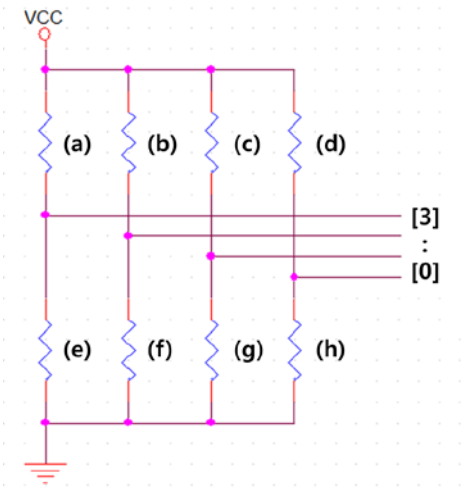

Fig. 3. Circuit diagram of the improved hardware version management part. (a) ~ (h) : The position where resistors are assembled (assemble / not assemble)

The improved hardware version management method can be possible for the setting of hardware versions using pull-up and pull-down resistors in digital circuits. This type of circuit provides a method to distinguish between various hardware versions using multiple resistors.

In this kind of circuit, there are two types of resistor configuration. Resistors could be connected to VCC it called pull-up resistor. In Fig. 3 set the input lines which are [3] ⋯ [0] to each point (a) ~ (d) to VCC and (e) ~ (h) to not assemble (N/A) turns them into pull-up resistors. Pull-up resistors represent a logic high level, typically expressed as '1', and are the same as VCC. When these inputs are connected to the pins of a microcontroller or other digital circuit, the pins will inherently have a high logic level. By applying the improved hardware management method, which uses pull-up and pull-down resistors in circuit, for hardware version management part. Therefore, we have obtained additional cases for the hardware version management. Our logic provides an additional 16 cases to manage hardware version without modifying the source code.

For example, the minor hardware version of the source code in Fig. 2 is expressed as a circuit as follows. Assembled a resistor of 1kΩ in (b), (c), (d), (e) and (a), (f), (g), (h) do not assemble. This represents input lines [3] ⋯ [0] result in a binary number 0111. The upper two bits, [3] and [2], are combined with the first decimal place of the hardware version, and the lower bits, [1] and [0], operate with the second decimal place of the hardware version. So, the hardware version of example is 0.13.

2.4 Applying The Improved Hardware Version Management Method to POSAFE-Q

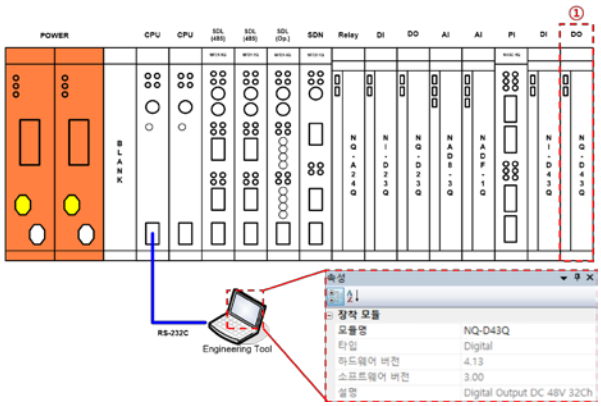

Fig. 4. Improved hardware management method applied to POSAFE-Q and its confirmation.

Here we show, improved hardware management method applied to POASE-Q. POSAFE-Q is a safety PLC that allows users to combine modules in various configuration for use. For testing purposes, various modules were configured to assemble POSAFE-Q. In Fig. 4 the hardware version information of POSAFE-Q modules can be checked by connecting a CPU with a 232 cable using the engineering tool. It was confirmed that hardware versions can be managed by changing only the hardware without touching the software.

3. Conclusions

The improved hardware version management method has been applied at POSAFE-Q which is safety PLC. This approach provides an effective solution to reduce the considerable time loss and cost issues associated with software V&V activities due to simple hardware changes. In situations where minor hardware modifications require the entire software verification process to be revisited, which can be cumbersome and costly, this improved method allows for quick and efficient identification and management of hardware versions, thereby simplifying the development process and enhancing cost efficiency.

REFERENCES

[1] IEEE Std. 1074, Standard for Developing a Software Project Life Cycle Process. 2006.
[2] IEEE Std. 1012, Standard for System, Software, and Hardware Verification and Validation. 2016.
[3] Lee, MyeongKyun, SeungWhan Song, and DongHwa Yun. Development and application of POSAFE-Q PLC platform. No. IAEA-CN--194. 2012.
[4] Ka-Ram Park et al., "Design of the FPGA based Safety PLC for the Nuclear Power Plants.", Proc. of KIEE, Summer, 2020, 459-460.