Development of a STPA tool in consideration of hazard analysis for NPP I&C systems

Sung-Min Shin[a*], Seong Woo Kang[a], Jinkyun Park[a]
[a]Korea Atomic Energy Research Institute (KAERI), Daejeon, Republic of Korea
*Corresponding Author: kshpjk@kaeri.re.kr

Abstract: System-Theoretic Accident Model and Processes (STAMP) and System-Theoretic Process Analysis (STPA) have emerged as novel perspectives for analyzing hazards in control systems, revealing previously unnoticed hazards that traditional failure propagation analyses may overlook. Their contribution, therefore, holds significant implications in hazard analysis, but challenges arise in the modeling of real-world control systems in practical detail, and certain execution processes may be inefficient. Addressing these concerns to support the potential for hazard analysis for nuclear power plant (NPP) I&C system, this paper proposes supplementation of STAMP components and computerization of linkages between system components. The logic and key features of the proposed in the paper are verified through actual implementation in a window application program (TRACEIT). Based on the results, the authors believe that the proposals in this paper can practically improve the efficiency and effectiveness of STAMP/STPA and, in turn, aid in the analysis of I&C system in NPP.

Keywords: STAMP/STPA, TRACEIT, NPP I&C, Hazard Analysis

1. INTRODUCTION

System-Theoretic Process Analysis (STPA), a hazard analysis method based on System-Theoretic Accident Model and Processes (STAMP), offers a significant shift in perspective for the field of hazard analysis by providing a systematic approach to examining hazards through the lens of unsafe interactions and signals (feedbacks) in control systems [1-3]. Recent studies in various engineering fields have shown that this approach can be valuable in identifying hazard factors that may have been overlooked in other traditional analyses that focus on the impacts of failure propagation. Thus, the utility of STAMP/STPA has gained widespread recognition in the field, finding active applications in recent hazard analyses across diverse fields such as automotive [4], medical [5], aircraft [6-7], railroad [8], maritime [9-10], and especially, nuclear power plants (NPPs) [11-20].

The STAMP visually expresses the target system using the connection of many control loops, and each control loop is composed of a controller (including control algorithm and process model), controlled process, feedbacks (FBs), and control actions (CAs). Figure 1 illustrates the elements of a control loop and the Table 2 summarizes the description of key elements included in the control loop.

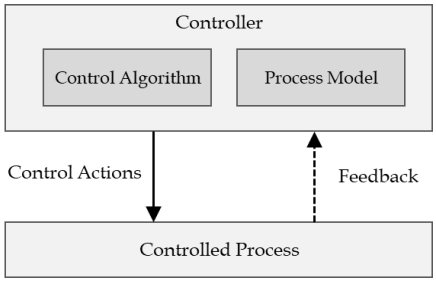


Figure 1. Typical control loop configuration of the STAMP

The STAMP is not limited to the schematic of physical and functional processes, but can also be used for the delineating interactive processes including human operators and related organizations [16]. That is, the CA in a STAMP implies not only the physical controls by engineered systems such as initiation signals or interlocks but also the managerial or operational controls that are essential for the accomplishment of a required task/function. Usually, the development of a STAMP starts at an abstract level to figure out the overall interaction, and then iteratively goes through a concretization process to express more detailed information depending on the purpose of the analysis.

Table 1. Key elements included in a control loop

Element	Description
Controlled process	Object to be controlled
Feedback (FB)	Information indicating the status of the controlled process
Controller	Subject determines whether a CA is generated or not. Control algorithm: The controller's decision-making procedures or logic Process model: Status of the controlled process understood by the controller (internal belief)
Control action (CA)	Control commands issued by the controller

Once the STAMP is developed, the STPA can be utilized as a hazard analysis technique with 4 steps. Figure 2 briefly depicts the four steps in performing the STPA. It should be noted that the second step ('Modeling the control structure') corresponds to the development of the STAMP.

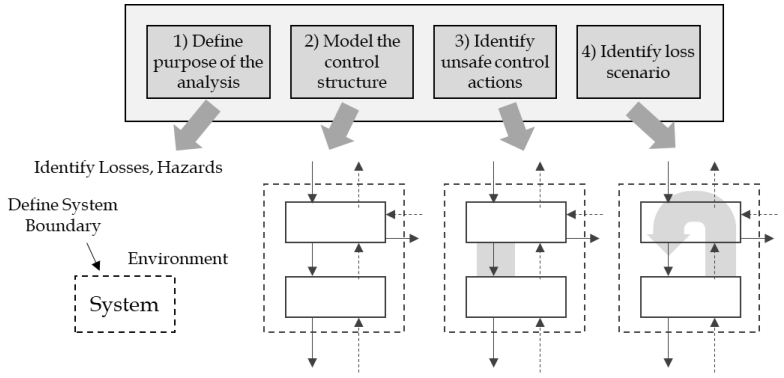


Figure 2. Four steps of the STPA

The first step of the STPA defines the catalog of undesired losses and hazards. The loss can be defined in various ways according to the analysis purpose such as human death/injury, property damage, environmental pollution, and mission failure. On the other hand, the hazard refers to a single or collective condition of the system that can lead to a predefined loss. It should be noted that the loss can be distinguished from the hazard because the former denotes a unique status that can no longer be controlled by the system.

In the second step, a control structure, in other word STAMP, is developed.

In the third step, among the CAs developed in the second step, the catalog of unsafe control actions (UCAs) is identified that could lead the status of the system to the undesired conditions (i.e., hazard). For the sake of clarity, it is recommended that the description of each UCA generally includes the following information: (1) controller, (2) control action, (3) UCA type, (4) context, and (5) relevant hazard. In other words, it is necessary to describe which control action on which controller, in what way, and in what circumstances (context), can lead to which hazard. In terms of promising UCA types, Table 2 exemplifies four kinds of UCAs that can be generally applied. The UCA can also be clearly described by considering the context of the system at the time it occurs.

Table 2. Representative UCA types

UCA type		Description format
1	Not providing causes hazard	Hazard occurs because <Controller> does not provide <Control Action>.
2	Providing causes Hazard	Hazard occurs because <Controller> provides <Control Action>.
3	Too early, too late, out of order	Hazard occurs because <Controller> provides <Control Action> too early, too late, or in the wrong order.
4	Too long or too soon	Hazard occurs because <Controller> provides <Control Action> for too long or too short.

Finally, in the fourth step, the causes of UCAs can be analyzed. Through this cause analysis, a scenario leading to one of the predefined losses (i.e., loss scenario) can be clarified by combining three pieces of

information such as 'UCA cause' – 'UCA' – 'Hazard' – 'Loss'. The causes of each UCA may contain diverse aspects including (but not limited to) incorrect feedback, inappropriate requirements, design errors and component failures.

In previous studies, the authors have applied STAMP/STPA to analyze the safety control systems of NPPs [15-17]. According to recommendations in the STPA handbook [3], a control structure is initially generated to examine the target system at an abstract level, after which it is progressively concretized with more detailed information. The development of detailed control structure is crucial for identifying all potential causes of the examined hazards. However, scrutinizing the detailed information may not be easy for complex systems, such as the safety control systems in NPPs.

When modeling the safety control systems in NPPs using STAMP, the system control structure may exhibit intricate configurations due to the incorporation of redundancy and diversity concepts as well as human factors. These concepts aim to ensure the availability of safety functions in emergency situations. Modeling and analyzing such intricate and intertwined structures pose numerous difficulties. One of the difficulties is tracking the signal linkages during the STPA process. The current STAMP, by visually representing signal flows within the control structure, requires analysts to manually recall and visually track these signal interconnections during the completeness review of the developed control structure or in a subsequent cause analysis. While this approach may be suitable for small-scale implicit analyses, it poses significant challenges in detailed large-scale analyses. Moreover, during a cause analysis to obtain comprehensive safety-related insights on the system, researchers may be required to analyze the impact of a common cause failure (CCF, an attribute shared by multiple components, such as location or platform in the safety and hazard analysis). However, the current STAMP/STPA may lack a firm foundation to analyze such an impact. Another study utilizing STPA has recognized this need as well [11].

In this paper, we propose supplementing the STAMP components and computerizing the linkages between these components. Based on the supplements, we envision a more efficient application of the STAMP/STPA technique in the detailed hazard analysis of large-scale control systems, mitigating the challenges posed by complex configurations and facilitating better identification of potential hazards representative of real-world characteristics. The validity and feasibility of the work were confirmed through their implementation in an actual window application program (TRACEIT).

2. SUPPLEMENTATION OF THE STAMP FRAMEWORK

2.1 Components and signal processing in STAMP

In order to develop and computerize a detailed large-scale control structure and comprehensively analyze the impact of potential CCFs, the authors propose to reorganize the components and connectors that compose STAMP, generalize the various signal processing performed within a control system, and assign attributes to STAMP components and connectors.

The current STAMP is built on a control loop consisting of four entities—controller, controlled process, actuator, and sensor— and two connectors—feedback and control action. In this context, analysts naturally consider how to classify each component of the target system as one of the component types of the control loop, a task that becomes confusing in cases with highly detailed modeling. As examples, is a component for transmitting audio-visual information to operators a controller, controlled process, actuator, or sensor? Is the signal generated by one controller but used as reference information in another controller a feedback or control action? To eliminate unnecessary considerations and improve the efficiency of STAMP development, the authors suggest the following.

- In addition to the four entities (controller, controlled process, actuator, sensor) of the existing STAMP, a new type of entity is defined, interface, for components that transmit signals from one entity to another entity or entities.
- All the information (feedbacks and control actions) in a control structure is provisionally defined as a signal, meaning there is no classification between feedback and control action. The signals

corresponding to control actions to be analyzed for UCAs can be selected separately before moving on to STPA step 3 after development of a control structure.

In addition to the above suggestions, we believe it would be appropriate to rename the entity previously defined as sensor to signal generator. This is because reference signals (feedback) are generated by environment and human operators, not only mechanical detectors, and it is difficult to encompass all of these sources of reference signals with the term sensor.

In some cases, UCAs, such as an incorrect order of occurrence or delayed occurrence of various control actions, are related to signal processing inside the control system. Therefore, it is necessary to model such signal processing together during development of a control structure and to analyze the effects of incorrect signal processing in STPA step 4, the cause analysis. In response to this need, as shown in Table 1, the authors generally categorized the signal processing that occurs inside a control system and defined which signal processing can be performed by each entity. The modeling method of a control structure in consideration of signal processing and computerization is described in the following sections.

Based on the authors' research experience, five types of signal processing are performed in a control system during signals' life cycle: generate – (restate or split) – synthesize – terminate. Placing restate or split in parentheses indicates that they are not essential; i.e., the life cycle of a control signal essentially involves generate – synthesize – terminate, while restate and split may occur on occasion. In addition, restate or split can occur either before and after synthesize, and synthesize can occur multiple times. The signal processing types are summarized as follows.

- Generate: The first time a signal is produced without any signal. Generate is performed in a signal generator or a controller.
- Restate: An addition or modification to a specific input signal. This is distinct from a synthesized signal in that it is done without a process model or control algorithm. An example is adding timing information to an alarm signal. Restate can be performed in any entity.
- Synthesize: The signal processing type corresponding to the existing procedure for producing a control action in current STAMP, where a signal is generated by decision-making associated with a specific process model or models and control algorithm. Here, computerization and linkage define which process model is updated by a particular signal or signals and which process model or models are referred to by a particular control algorithm. Synthesize is performed only in a controller.
- Terminate: The termination of a generated signal, after having gone through a single (or multiple) controller(s) to be synthesized and eventually executed in a controlled process. Terminate is performed only in a controlled process.
- Split: The transmittion of a signal to one or more entities. Split can be performed in any entity.

Table 3. Signal processing and related entities in the control system

Entity	Generate	Restate	Synthesize	Terminate	Split
Signal generator	X	X			X
Interface		X			X
Controller	X	X	X		X
Actuator		X			X
Controlled process		X		X	X

As shown in Table 1, the authors defined which signal processing can be performed by which entity, considering that each entity can be either machine or human, or both. The entities are summarized as follows.

Signal generator: Signal generators are the machine/human that generates an initial reference signal and may also restate or split the specific input signal received depending on the function of the control system.
Interface: Interfaces function to transmit specific inputs, sometimes modifying them such as by adding specific information. Interfaces can transmit a signal to multiple entities.

2.2 Computerization of linkages between STAMP components

The purpose of computerizing STAMP components in this study is to ensure that they are written in a specific format so that they can be used to track the flow of specific signals or signal processing. This can be implemented by assigning identities (IDs) to the STAMP components and expressing signal flows and signal processing based on the assigned IDs.

In the existing STPA, it is common to assign IDs to losses, hazards, control actions, UCAs, and loss scenarios, but in order to implement the computerization pursued in this study, IDs are assigned to all entities, connectors, signals, and process models. Regarding ID assignment, the following principles are applied.

- Basically, IDs take the form of a designator + number. The designator is given for each entity type in the case of entity, signal, and process model.
- In the case of signals, it is necessary to manage their IDs during development of a control structure, such as assigning a new signal ID according to signal processing to check the traceability later. In this regard, a new signal ID is given when an initial reference signal is generated and another signal ID is given when the existing signal is restated or synthesized, while the signal ID is retained when the same signal is transmitted as is or split (transmitted to multiple entities).

For each control, a linkage is made by connecting the beginning and end points of the signal flow; that is, the information on generate – (restate or split) – synthesize – terminate. The beginning point is defined as a signal generated by a signal generatoror a terminated signal restated in a controlled process, and the end point is defined as a terminated signal. Among the types of signal processing, generate, restate, and synthesize involve defining the information in detail through the writing of text, while split and execute are defined through the structural connection of connectors and the selection of signals to be transmitted.
For a better understanding, the computerization process of the suggested approach for a STAMP of the simple hypothetical system shown in Figure 3 is depicted in Figures 4–6. The required function of the example system is to produce an automatic trip signal by the reactor protection system (RPS) based on the temperature and pressure of the reactor, or to manually trip the reactor by a human operator if the automatic trip fails.

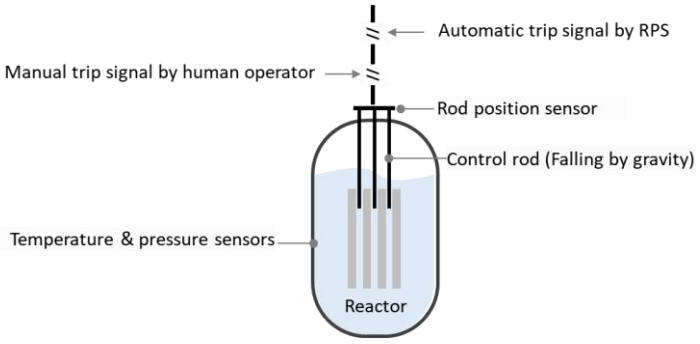
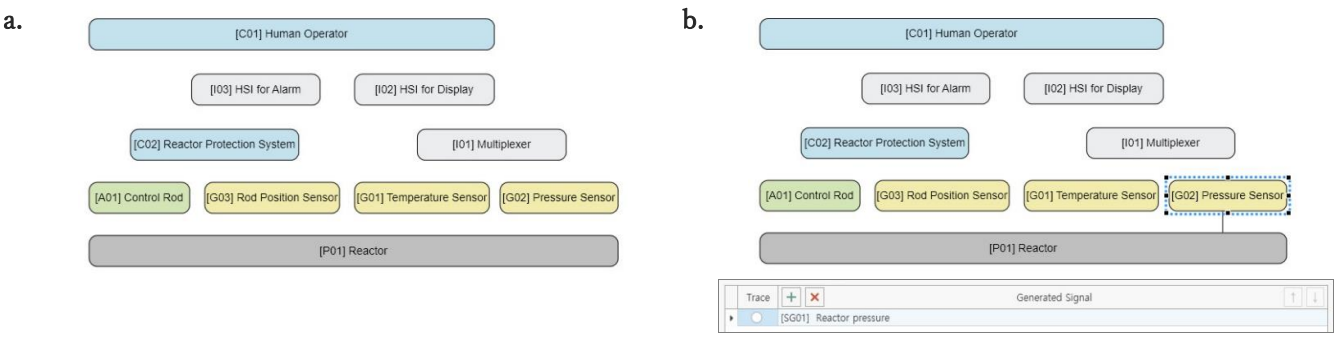


Figure 3. Simple hypothetical system to illustrate the computerization process of STAMP.



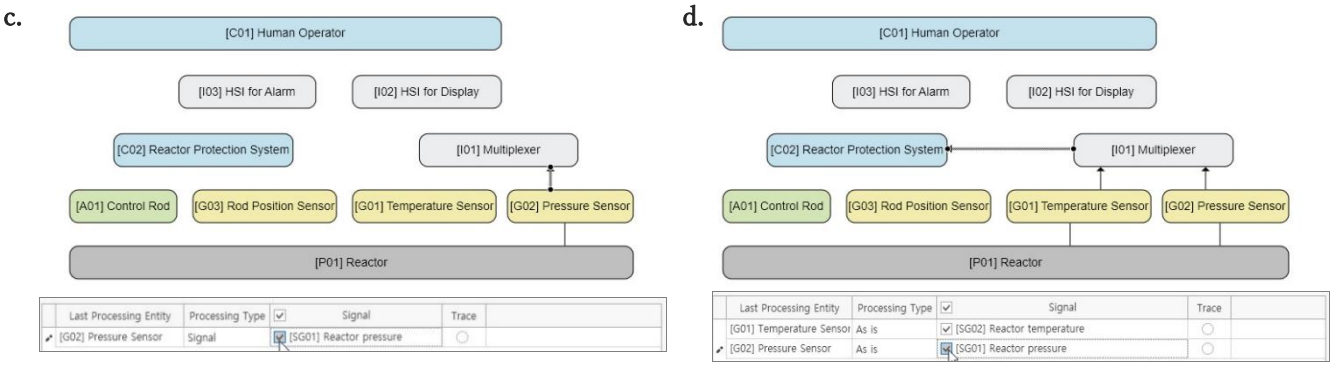
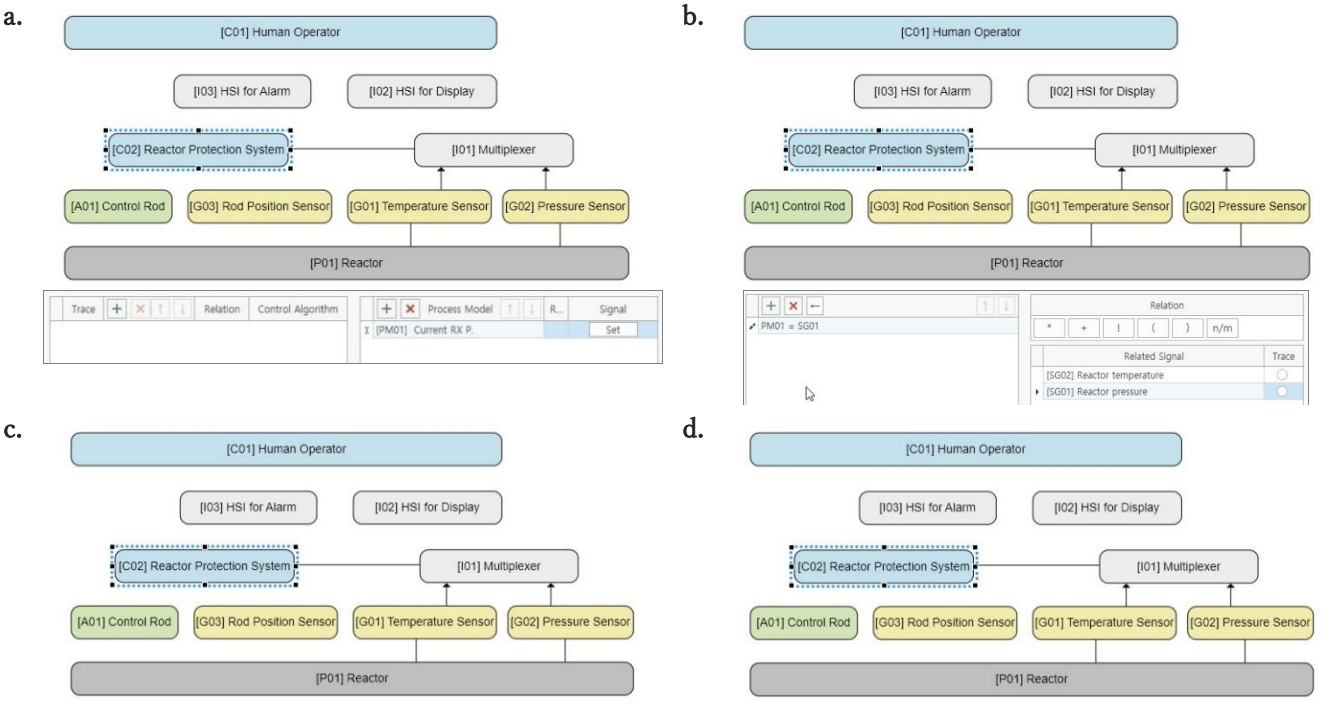


Figure 4. Automatic trip signal by the RPS: signal generation and transmission.

Figure 4 shows how signals are generated by the sensors (Pressure and Temperature Sensors) and then transmitted to one controller (RPS). Each entity has a designator in its assigned ID, with the following abbreviations: C (controller), I (interface), A (actuator), G (signal generator), and P (controlled process). Compared to the current STAMP, the difference is that the large amount of complex signal transmissions that may be involved in a detailed large-scale model is internalized instead of being written out in text on the connector line. The process shown in Figure 4a–d is as follows.

a. Referring to Figure 4, the entities that make up the example system are configured. Interfaces that aggregate and transmit signals (Multiplexer, Human–system interface (HSI) for Alarm, and HSI for Display) are additionally configured from the current STAMP, and IDs are assigned to each entity according to the entity type in the form of a designator + number.
b. A signal generator is specified, such as [G02] Pressure Sensor, and the contents of the signal generated by the sensor, such as [SG01] Reactor pressure, are defined.
c. After connecting [G02] Pressure Sensor and [I01] Multiplexer, the signal to be transmitted through the connector is selected. At this time, all the signals inside the entity where the start point of the arrow is located are displayed as a list that can be transmitted. In the case of [G02] Pressure Sensor, it has only one signal, so Figure 4c displays only [SG01] Reactor pressure.
d. An additional signal, [SG02] Reactor temperature, is generated by [G01] Temperature Sensor and transmitted to [I01] Multiplexer in a similar manner as in Figure 4b and 4c. When connecting [I01] Multiplexer to [C02] RPS, two signals ([SG02] Reactor temperature and [SG01] Reactor pressure) are shown. It is then assumed that both signals are transmitted to the RPS.



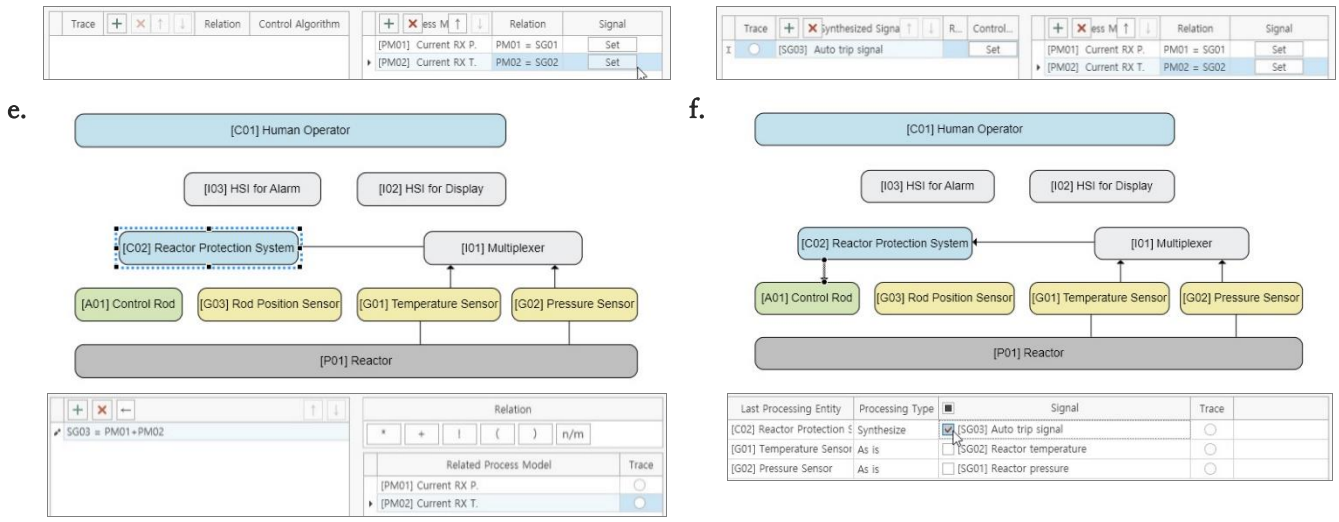


Figure 5. Automatic trip signal by the RPS: process model and control algorithm.

As a controller, [C02] RPS generates a synthesized signal through a process model and control algorithm. The computerized process of the synthesized signal is shown in Figure 5 as follows.

a. The RPS is specified and the content of the process model [PM01] is defined. For example, the RPS must be aware of the current reactor (RX) pressure to generate the automatic trip signal.
b. The process model is updated by the specific signal or signals. In the example, [PM01] Current RX P. is updated by [SG01] Reactor pressure among the input signals to the RPS. The process model in the example is formed by a single signal, but the process model can also be formed by multiple signals according to logical conditions between them. This can be represented as a logical combination of and, or, not, n out of m, etc.
c. Similar to Figure 5a and 5b, it is assumed that an additional process model in the RPS, [PM02] Current RX T., is updated by [SG02] Reactor temperature.
d. Defining the synthesized signals generated by the control algorithm is similar to defining a process model with its associated signals. Here, the RPS is specified and the contents of the synthesized signal, [SG03] Auto trip signal, are defined.
e. The control algorithm for determining whether to generate the synthesized signal, [SG03] Auto trip signal, can refer to a single or multiple process models according to the logical conditions between them. In the example, if either the temperature or the pressure of the reactor exceeds a certain predefined point, an automatic trip signal is generated, and thus the signal is defined as PM01 +(or) PM02.
f. The RPS now has a total of three signals, including one synthesized signal and two input signals. In order to drop the control rod, among the three signals, [SG03] Auto trip signal must be transmitted to [A01] Control Rod.

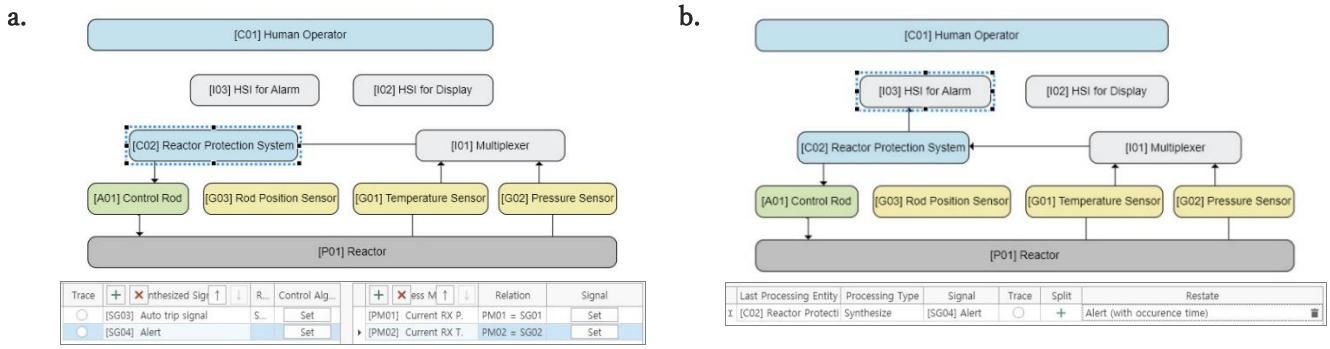


Figure 6. Manual trip signal by a human operator.

Another controller, [C01] Human Operator, should generate a manual trip signal as needed by checking the current state of [P01] Reactor and whether the control rod is inserted by [C02] RPS. The process of the reference signals and synthesized signals in this case is generally similar to the previous description for the RPS; Figure 6 is as follows.

a. In addition to generating [SG03] Auto trip signal, the RPS additionally generates [SG04] Alert signal for transmission to the human operator. It is assumed that the synthesized signal refers to the current RX temperature and pressure, that is, [PM01] +(or) [PM02], in the same way as the automatic trip signal.
b. The synthesized signal, [SG04] Alert, is transmitted to [C01] Human Operator through [I03] HSI for Alarm, which adds additional information of the timing that the alert occurred for efficient judgment by the human operator. That is, [SG04] Alert is restated in [I03] HSI for Alarm.
c. For accurate situational awareness, additional signals are transmitted to [C01] Human Operator. That is, [SG06] Control rod position generated by [G03] Rod Position Sensor and [SG02] Reactor temperature and [SG01] Reactor pressure inside [I01] Multiplexer are transmitted to [C01] Human Operator through [I02] HSI for Display. Accordingly, there are four signals inside [C01] Human Operator.
d. Two process models are required for [C01] Human Operator to generate the manual trip signal. First, [PM03] Current RX situation is assumed to be updated by checking the current RX temperature and pressure after recognizing the abnormal situation through [SG05] Alert (with occurrence time).
e. The second process model for generation of the manual trip signal, [SG04] Control rod position, is assumed to be updated by [SG06] Control rod position.
f. Next, it is assumed that the synthesized signal [SG07] Manual trip signal is generated after recognizing the current RX situation requiring a RX trip and confirming that the control rod has not been inserted. Accordingly, the control algorithm given in Figure 6f is defined as [PM03] x(and) [PM04].

2.3 Attributes of STAMP components

The current STAMP entity does not entail specification information such as its location or platform. In this paler, the authors suggest assigning attributes and values to form specifications of the entities and connectors that compose STAMP, and establishing a framework with which those attributes and values can be shared across all entities and connectors. Information on the attributes and values can later be utilized in the CCF analysis to support cause analysis during STPA step 4. Whi this purpose, the authors set up a preliminary framework for sharing attributes first by building and managing a pool of attributes and values and second by enabling each entity and connector to take a specific value for a particular attribute. For better understanding, the method of assigning and sharing attributes and values during STAMP development is briefly described based on the above examples.

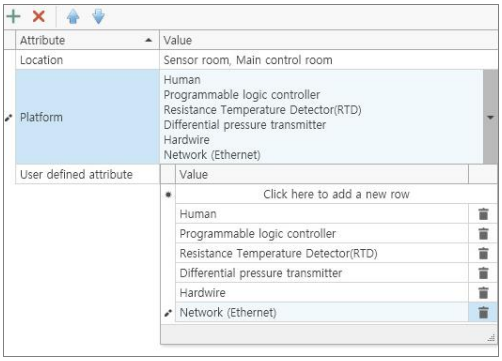


Figure 7. Pool of attributes and values.

Figure 7 shows an example of building and managing a pool of attributes and values. As mentioned earlier, typical attributes may include location and platform, but other attributes may be defined depending on the analyst developing the STAMP and the target system to be analyzed. The authors believe that it would be preferable from an analytic freedom perspective to include all the attributes and values that each STAMP component can have without distinguishing between entities and connectors or between types of entities.

a.

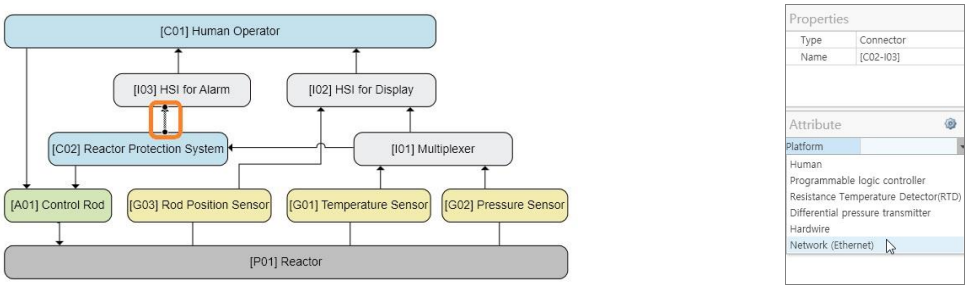


Figure 8. Selection of a specific value for a particular attribute.

Figure 8 shows an example of assigning a specific attribute value to a STAMP component. In order to efficiently handle a large amount of information, the example system assumes that a network (ethernet) is applied for signal transmission between connectors [C02–I03], and assigns the corresponding attribute value to the corresponding connector. Attribute values for entities could also be assigned in the same way.

2.4 Comparison of current and proposed STAMP

Section 2.1–2.4 described how to supplement the current STAMP framework. For a clearer understanding, Table 2 briefly compares the current STAMP with that proposed in this paper. Basically, the proposed STAMP seeks to provide a firmer foundation for performing more efficient and insightful STPA.

Table 4. Comparison of current and proposed STAMP

	Current STAMP	Proposed STAMP	Remark
STAMP components	• Entities: controller, controlled process, actuator, and sensor	• Entities: controller, controlled process, actuator, sensor, and	

	• Connectors: feedback and control action	interface • Connector: signal	
Feedback and control action	• Feedback and control actions must be distinguished	• Feedback and control actions are not distinguished, being provisionally defined as signals	• Signals corresponding to control actions can be selected separately before moving to STPA step 3
Signal notation	• Write signals to a connector	• Edit signal processing in entities • Select a signal to be transmitted by each connector	
Attributes of STAMP components	• Not considered	• Attributes and values are assigned to entities and connectors	• Attribute/value information is used as a basis for CCF (abnormal) analysis

3. CONCLUSION AND DISCUSSION

This paper suggested supplementation of the STAMP framework and the computerization of linkages between STAMP components. The computerization information can be utililzed to provide several supportive functions to enhance the efficiency of STAMP/STPA, such as highlight the views of specific signal flows in consideration of signal processing in each entity(Figure 9), review the completeness of the developed control structure, and provide an abstract view on unsafe control action (UCA) and cause analysis(Figure 10). Although not all detailed explanations were given due to the constraints of the ground, each of these functions are actually confirmed through their implementation in an actual window application program, TRACEIT.

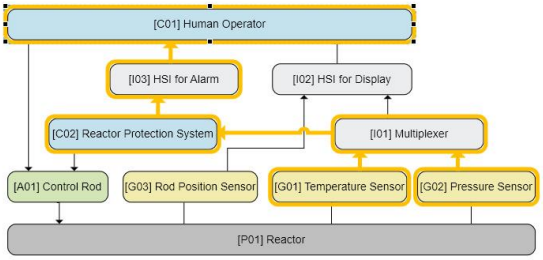


Figure 9. Example of highlight view of a specific signal flow

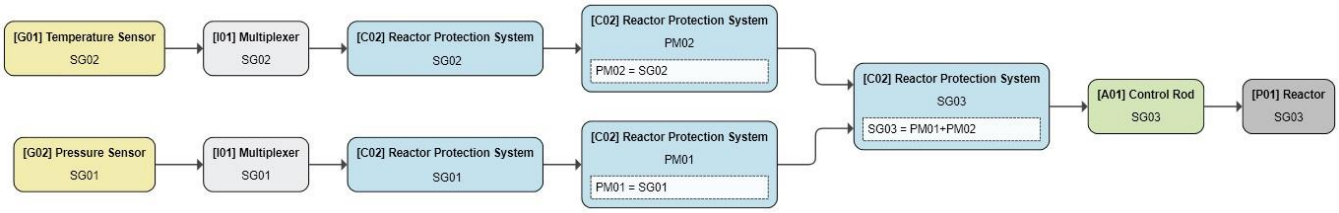


Figure 10. Exmaple of abstract view of an signal flow

This contribution aims to extend the applicability and utility of STAMP/STPA in the realm of large-scale system analysis. To do so, this work addressed specific challenges encountered in detailed large-scale analysis based on the current STAMP/STPA. Although the scope and usefulness of STAMP/STPA can be expanded through the proposed extensions, it should be noted that the supplementation of STAMP was based on the authors' experiential knowledge, particularly the content in Table 1. Signal processing and the related entities in a control system need continuous review and improvement for application to various target systems. While future refinement of this content may influence the proposed modifications, the significance of this study lies in the computerization of STAMP components and their linkages, along with the introduction of various STPA supportive functions.

Acknowledgements

This work was supported by an Innovative Small Modular Reactor Development Agency grant funded by the Korean Government (MSIT) (No. RS-2023-00258118), and by the National Research Foundation (NRF) of the Republic of Korea funded by the Ministry of Science and ICT (No. RS-2022-00144175).

References

[1] N. G. Leveson (2004) A new accident model for engineering safer systems, Safety Science, vol. 42, no . 4, pp. 237-270.
[2] N. G. Leveson (2011) Engineering a Safer World: Systems Thinking Applied to Safety, Cambridge, M A, USA: MIT Press.
[3] Leveson NG, Thomas JP. STPA handbook. MIT; 2018.
[4] Abdulkhaleq A, Wagner S, Leveson N. A comprehensive safety engineering approach for software-intensive systems based on STPA. Procedia Eng 2015;128:2–11.
[5] Faiella G, Parand A, Franklin BD, Chana P, Cesarelli M, Stanton NA, Sevdalis N. Expanding healthcare failure mode and effect analysis: A composite proactive risk analysis approach. Rel. Eng. Syst. Saf. Jan. 2018;169:117–26.
[6] Lu Y, Zhang SG, Tang P, Gong L. STAMP-based safety control approach for flight testing of a low-cost unmanned subscale blended-wing-body demonstrator. Saf. Sci. Apr. 2015;74:102–13.
[7] Allison CK, Revell KM, Sears R, Stanton NA. Systems Theoretic Accident Model and Process (STAMP) safety modelling applied to an aircraft rapid decompression event. Saf. Sci. Oct. 2017;98:159–66.
[8] Dapo Oginni, Fanny Camelia, Mikela Chatzimichailidou, Timothy L.J. Ferris, Applying System-Theoretic Process Analysis (STPA)-based methodology supported by Systems Engineering models to a UK rail project, Saf. Sci., Volume 167, 2023,
[9] Wrobel K, Montewka J, Kujala P. Towards the development of a system-theoretic model for safety assessment of autonomous merchant vessels. Rel. Eng. Syst. Saf. Oct. 2018;178:209–24.
[10] Meriam Chaal, Osiris A. Valdez Banda, Jon Arne Glomsrud, Sunil Basnet, Spyros Hirdaris, Pentti Kujala, A framework to model the STPA hierarchical control structure of an autonomous ship, Saf. Sci., Volume 132, 2020, 104939,
[11] Thomas J, Lemons FL, Leveson N. Evaluating the safety of digital instrumentation and control systems in nuclear power. Plants Res Rep 2012. Nov.NRC-HQ-11-6-04-0060.
[12] Wheeler T, Clark A, Williams A, Muna A, Dawson L, Geddes B, Blanchard D. Hazards and consequences analysis for digital systems. EPRI Technical Report; 2018. Dec.
[13] Rejzek M, Hilbes C. Use of STPA as a diverse analysis method for optimization and design verification of digital instrumentation and control systems in nuclear power plants. Nucl Eng Des 2018;331.
[14] Bao H, Shorthill T, Zhang H. Hazard analysis for identifying common cause failures of digital safety systems using a redundancy-guided systems-theoretic approach. Ann Nucl Energy 2020;148.
[15] Lee SH, Shin SM, Hwang JS, Park JK. Operational vulnerability identification procedure for nuclear facilities using STAMP/STPA. IEEE Access 2020;8(99):166034–46.
[16] Shin SM, Lee SH, Shin SK, Jang IS, Park JK. STPA-based hazard and importance analysis on NPP safety I&C systems focusing on human–system interactions. Reliab Eng Syst Saf 2021;213. 107698
[17] Shin SM, Lee SH, Shin SK. A novel approach for quantitative importance analysis of safety DI&C systems in the nuclear field. Reliab Eng Syst Saf 2022;226. 108765
[18] Shin SM, Park JW, Park JK, Kim YC, Kim JH. STAMP/STPA-based task analysis for multi-unit HRA, ASRAM2022, 2022 Nov. Daejeon Korea.
[19] EPRI, Hazard Analysis Methods for Digital Instrumentation and Control Systems, Electric Power Research Institute, EPRI 3002000509, 2013.
[20] Thomas J. Investigation of the Use of System-Theoretic Process Analysis at the NRC.