Architecture Framework for Spacecraft Computer Control Safety System

Seiko Shirasaka
Keio University, Graduate School of System Design and Management
4-1-1 Hiyoshi, Kohoku-ku, Yokohama 223-8526, Japan
shirasaka@z3.keio.jp

Abstract. We have a strong demand to design computer control safety system because crew stats to stay the international space station continuously, but there is no design framework for spacecraft computer control safety system. In this paper, we propose the architecture framework for spacecraft computer control safety system to guide a safety systems engineer to design computer controlled safety system. This framework is based on the general safety design process and NASA's computer safety requirements. The architecture framework can be proposed in accordance with the coming new IEEE1471 standard. This architecture framework can cover all design processes which are mandatory to be conducted to design system safety architecture. From this result, we suppose that this architecture framework reaches at least mandatory level.

Background

Overview. In the space mission, a hazard was not controlled by a computer because the on–orbit crew could execute safing action at the space shuttle era. However, in the international space station, a computer has to control hazards without crew support because all crew go to bed at the same time. During crew night, a computer has to control hazards. Because of this situation, we have a strong demand to design computer control safety system, but there is no design framework for spacecraft computer control safety system. And NASA introduced the new requirements when a computer is used for a hazard control. Because this requirement document brought new idea, it is not easy to design the system architecture to meet the requirements. From the point of view of architecture, the IEEE 1471 standard defines the architecture description. And the next IEEE1471 is trying to incorporate the idea of the architecture framework.

Safety Design Process

General Process. Safety Design is conducted as follows. (Figure 1) Safety design starts from hazard identification. The safety requirements define what is a hazard. And then a cause to realize the hazard is identified. The Fault Tree Analysis (FTA) is usually used for the cause identification process. After the causes are identified, the controls to prevent hazardous conditions are designed. When the control is designed, failure tolerance requirements and mission operability requirements have to be considered. The failure tolerance requirement is that the number of failure counts for a hazard to be controlled. For example, in the international space station program, the requirements are like follows:

- Two fail safe for catastrophic hazard : Two failure or two operator error or combination of one failure and one operator error should not cause catastrophic hazard

- One fail safe for critical hazard : One failure or one operator error should not cause critical hazard

The mission operability requirement is that he number of failure counts for a mission to be continued. For example, one fail operative requirement is that mission should be able to continue after one failure or one operator error. When we design controls to a hazard, we also take these requirements into count because all of them affect system architecture.

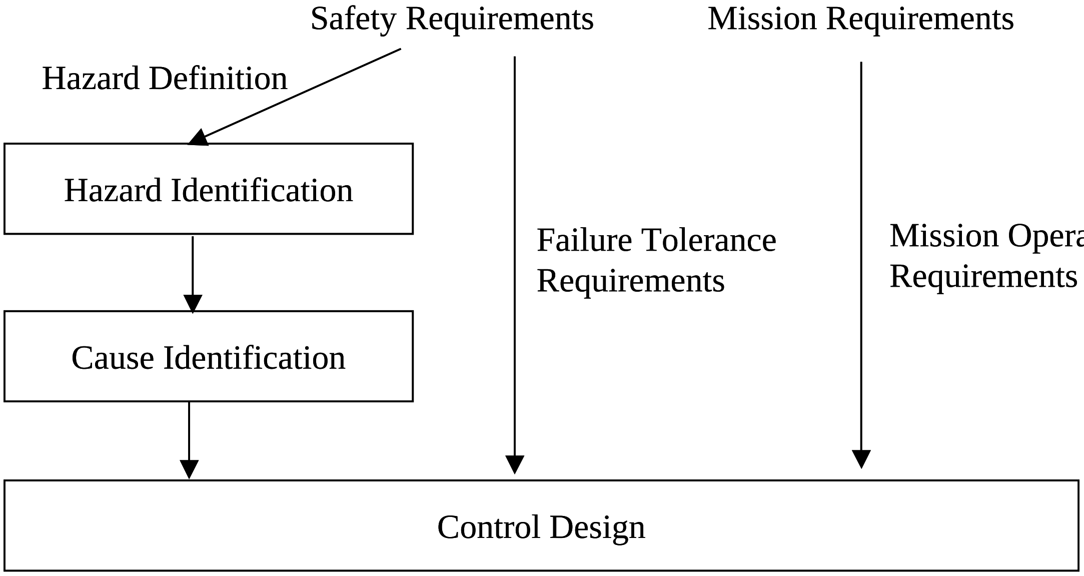

Figure 1: The Overview of Safety Design Process

The freedom of the control design is pretty large. Especially, if a computer controls a cause, there is no standard design in general. To guide an engineer to design computer control safety system, a special requirements are generated in the international space station program. That is “Computer Based Control System Safety Requirements”. This is applied to a space system which belongs to the international space station programme. However, there are just two requirements are special to the space systems. All the other requirements can be applied to other technical system.

Computer Based Control System Safety Requirements

Background. Computer Based Control System (CBCS) Safety Requirements are one of NASA International Space Station (ISS) Safety Requirements. For space shuttle program, an on-board crew could be the final control to prevent hazards. However, all on-orbit crew go to bed at the same time in the international space station. During crew night time, hazards have to be controlled automatically without crew support. It means that a computer will control a hazard. To ensure this situation, NASA introduced new safety requirements for computer based control system in 1995.

Introduction. NASA CBCS safety requirements are applied to a system which uses one or more computers to control hazards. It consists of following unique three technical requirements.

- General requirement
- Must work function (MWF) requirement
- Must not work function (MNWF) requirement

General requirements are always applied as far as a computer is used as a hazard control. MWF requirements and MNWF requirements are applied if a computer controls more than one control. (Table 1)

Table 1 : Application category of CBCS requirements

	One control	More than one control	
General	x	x	x
MWF	-	x	-
MNWF	-	-	x

General Requirements. General requirements are applied to all computers which control a hazard. They are component level requirements. They are a kind of basic requirements which should be satisfied if the computer is used for hazard control. One example of these requirements is "A processor shall continue to operate safely during off-nominal power conditions, or contain design features which safe the processor during off-nominal power conditions." There are 13 requirements as CBCS general requirements.

MWF Requirements. These requirements are applicable to the design of CBCS functions whose inadvertent shutdown would cause a hazard. The design approach to meet these requirements is fault tolerant approach. Even if a failure inadvertently shuts down a function which control a hazard, other function should control the hazard. When all controls are shutdown, a hazard could be caused..

MNWF Requirements. These requirements are applicable to the design of CBCS functions whose inadvertent operation would cause a hazard. To prevent inadvertent operation of a function, inhibits are used as controls. According to the NASA's CBCS requirement document, an inhibit is defined as follows: "A design feature that provides a physical interruption between an energy source and a function (e.g., a relay or transistor between a battery and a pyrotechnic initiator, a latch valve between a propellant tank and a thruster, etc.). Note: Software inhibits are not counted in meeting safety requirements for multiple inhibits." Inhibits have to be hardware and to interrupt energy source. There are two design approaches for MNWF implementation. One is "Fault Containment Approach." And the other one is "Control Path Separation Approach." The fault containment approach uses a unique computer for each inhibit-control respectively. The control path separation approach uses one computer to control all inhibits. However, software is carefully designed not to remove more than one inhibits by one failure.

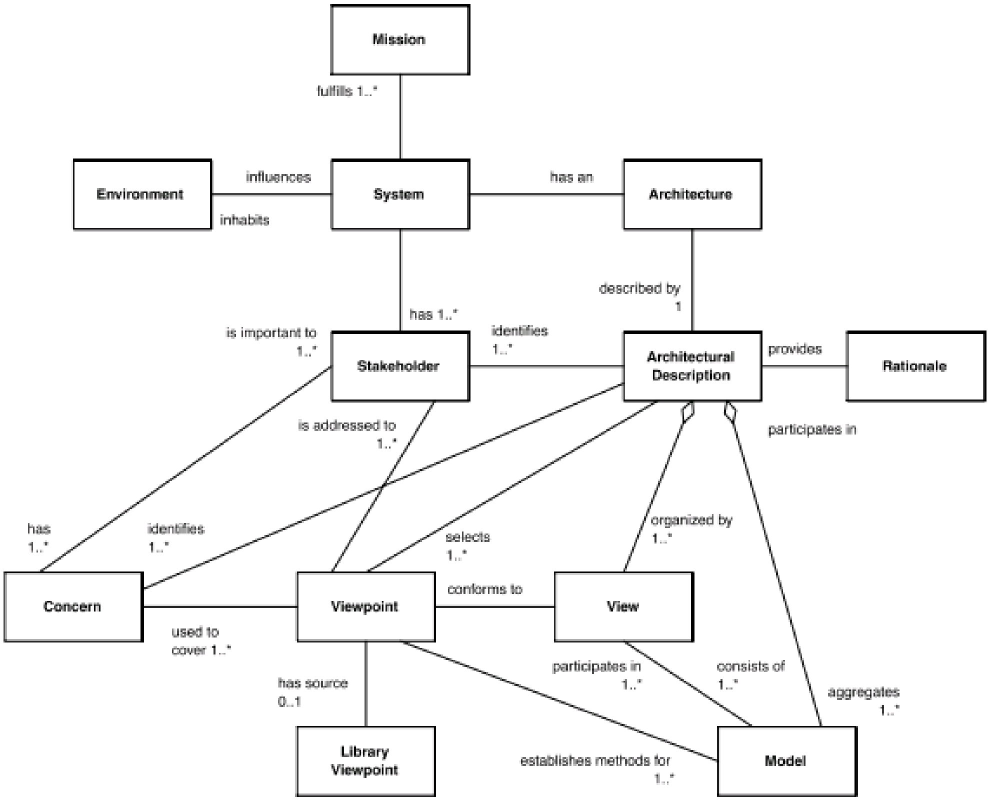

Figure 2 : Conceptual model of architecture description in IEEE1471-2000

Architecture Framework and Architecture Description

IEEE1471-2000. The IEEE 1471-2000 “Recommended practice for architectural description of software-intensive systems” was released in 2000. This standard brought the idea of view and viewpoint to describe system architecture. (Figure 2) The IEEE1471-2000 proposes the general idea of viewpoint, but it does NOT propose any specific viewpoint. To use this standard to design a system architecture, an engineer has to start from the identification of a stakeholder, his or her concern and then define the viewpoints. However, if the application scope is limited, the stakeholder can be specified. That means the concern and the viewpoint can be specified, too. There are some standard or architecture framework specify viewpoint.

Standard. Architectural design is one of the key activities in systems engineering. System architecture is usually described in several view points because a description from one viewpoint is not enough to describe it correctly. Some of the systems engineering standards such as IEEE 1220 and ANSI/EIA632 specify the viewpoints. For example IEEE1220 specifies three view points to describe system architecture. (Figure 3) Those are operational view, functional view and physical view. ANSI/EIA632 specifies two view points: logical view and physical view. We can just follow these standards as far as they are appropriate to be applied.

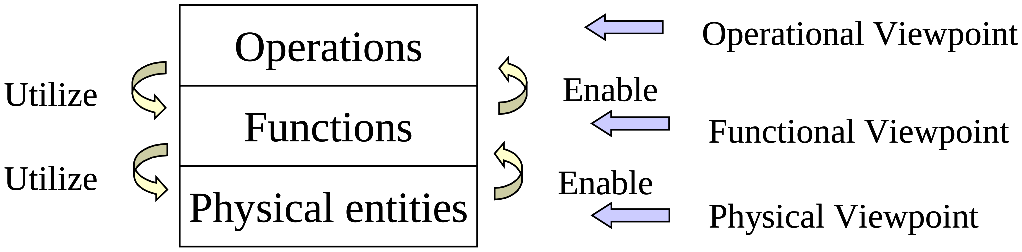

Figure 3: The viewpoint structure of IEEE1220

Architecture Framework. There are several architecture frameworks and they also specify multiple viewpoints to describe system architecture. One of the most famous frameworks is DoD architecture framework (DoDAF). DoDAF specifies three view points: operational view, system view and technical view. (Figure 4) Federal enterprise architecture framework (FEAF) specifies four view points: business view, data view, application view and technology view. These views are clearly specific to information systems. It means that FEAF can be used only for information system. The Zachman's framework specifies six views: scope view, business view, system view, technology view, detailed representation view and functioning enterprise view. These views are not specific to information systems but to technical systems. There are other frameworks such as The Open Group Architecture Framework (TOGAF) and Ministry of Defense Architecture Framework (MODAF). Their view points are also limited to technical systems architecture description. These architecture frameworks are widely used for architectural design of technical systems.

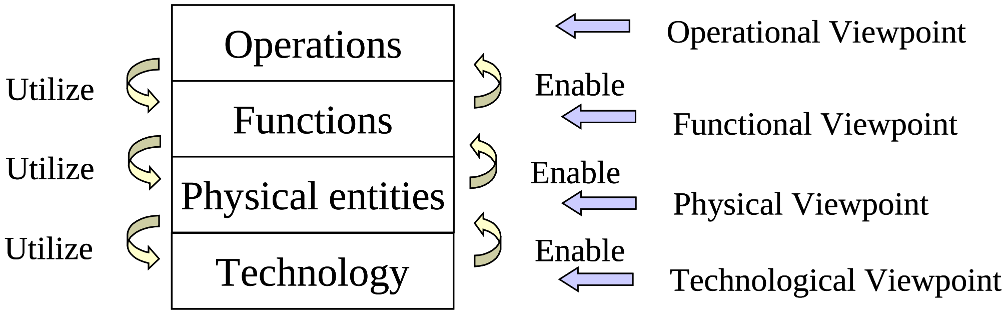

Figure 4: The viewpoint structure of DODAF

Updated IEEE1471. Currently updated IEEE1471 is in work. The updated standard has not yet approved. However, it is trying to incorporate new idea into the current IEEE1741-2000. That is the architecture framework. (Figure 5) It defines the usage of the architecture framework and the requirements of the architecture framework.

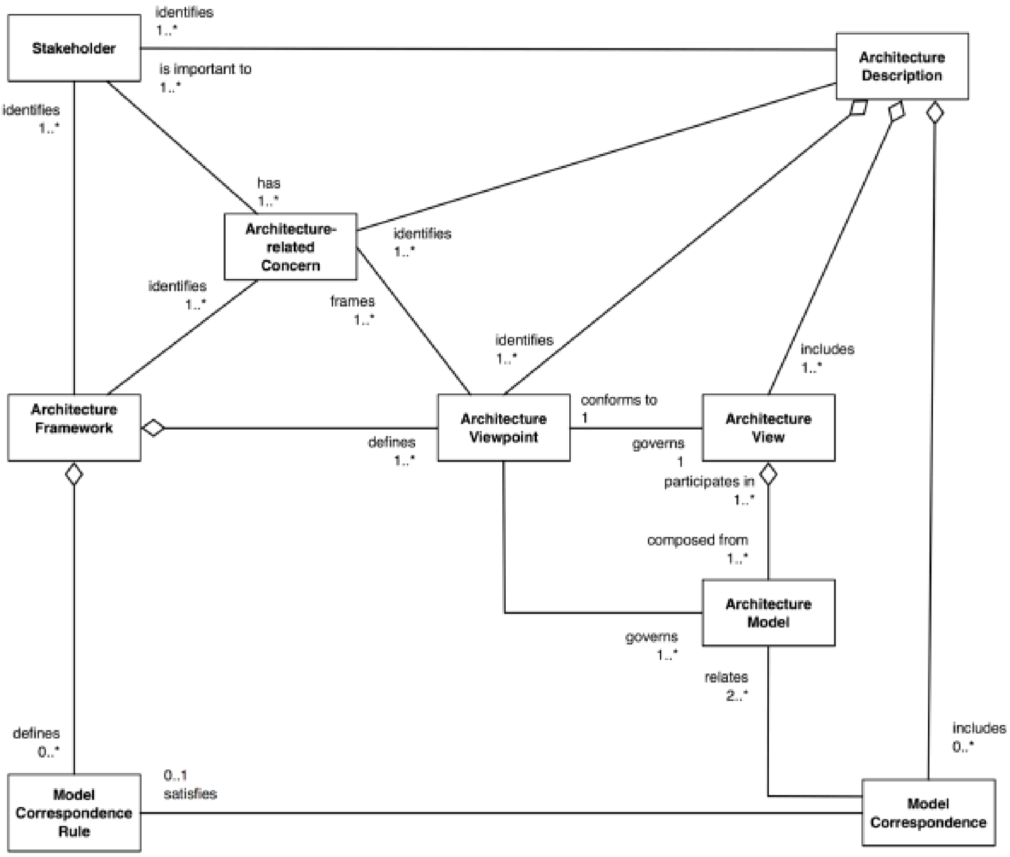

Figure 5: Architecture framework in next IEEE1471

According to the next IEEE1471, followings are required for the architecture framework.

- the identification of one or more concerns
- the identification of one or more stakeholders having those concerns
- one or more architecture viewpoints which frame those concerns
- zero or more model correspondence rules

Architecture Framework for Spacecraft Computer Control Safety System

Overview. The general safety design process is described in the first section. (Figure 1) When we take CBCS approach into count, the CBCS control concept has to be considered. (Figure 6) And the architecture views which are developed are also described in Figure 6. The hazard/control architecture view is developed at the hazard / cause identification process. And the CBCS architecture view is developed at the CBCS control concept design process. The safety failure tolerance architecture view is developed at the failure tolerance design process. The operational failure tolerance architecture view is developed at the mission operability design process. Finally functional and physical control architecture views are developed to realize the CBCS architecture view, the safety failure tolerance architecture view and operational failure tolerance architecture view. Each architecture view has one architecture viewpoint respectively. The architecture of viewpoints is described in Figure 7. Most of the relationship between the viewpoints is "enabler". (Shirasaka, S 2008) However, the relationship between the hazard/cause viewpoint and CBCS viewpoint is not "enabler", because CBCS does NOT enable hazard/cause.

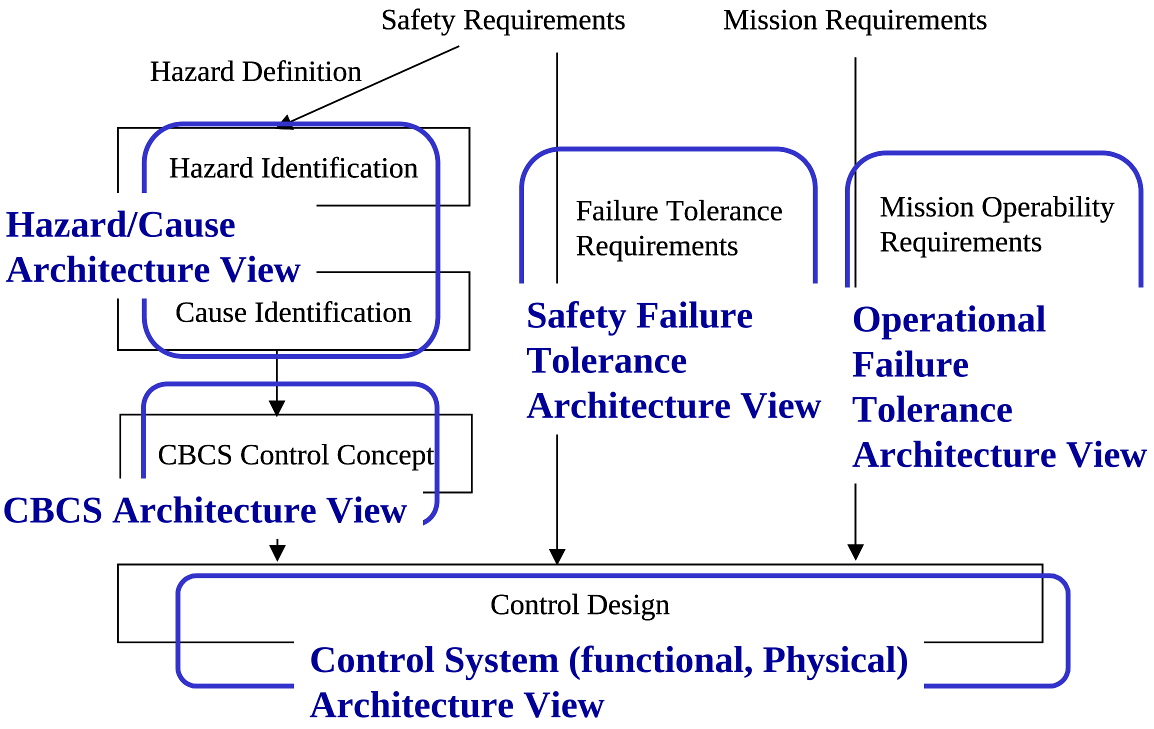
Safety Requirements
Mission Requirements
Hazard Definition
Hazard Identification
Hazard/Cause Architecture View
Cause Identification
Failure Tolerance Requirements
Mission Operability Requirements
Safety Failure Tolerance Architecture View
Operational Failure Tolerance Architecture View
CBCS Control Concept
CBCS Architecture View
Control Design
Control System (functional, Physical) Architecture View

Figure 6: CBCS Safety Design Process & Architecture

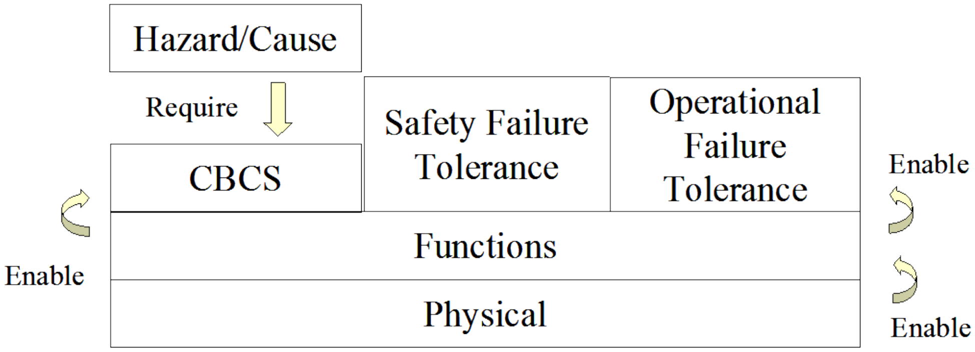

Figure 7: Viewpoint architecture

According to the next IEEE1471 draft version, an architecture framework shall have following four items:

- the identification of one or more concerns;
- the identification of one or more stakeholders having those concerns;

- one or more architecture viewpoints which frame those concerns;
- zero or more model correspondence rules.

Three of the four items for the architecture framework for spacecraft computer control safety system will be described in following section. (There is no model correspondence rule.)

Concern and Stakeholder. Some of the biggest stakeholders for safety critical system are safety engineers who have to design the control for hazards and system architect who have to incorporate safety design into the system architecture. There are several important concerns for safety design. Safety engineers have several concerns which related to the architecture. The first concern is that what is a hazard to the system of interest and what causes the hazard. Off course, other stakeholders like customers or users concern about the hazard. The second safety engineer's architecture related concern is controls. In this case, CBCS is the direct control method, so CBCS control concept is the second concern. The third safety engineer's concern is safety failure tolerance requirements. For example, safety architecture to meet one failure safe requirement is very different from that to meet two failure safe requirements. This may be also acquirer concern. There are several system architect concerns. The first system architect's architecture related concern is operational failure tolerance requirements. They tremendously affect the system architecture. Off course, they are also user concern and acquirer concern. The second system architect's concern is control system functions which are designed by system architect. The third system architect's concern is control system physical architecture which is also designed by system architect.

Architecture viewpoints. There are five viewpoints corresponding to stakeholder's concerns. The five viewpoints are described in accordance with next IEEE1471 format which include concerns, model types and sources.

- Hazard / Cause viewpoint

Concerns framed by the viewpoint	Hazard and its cause identification
Model types used in this viewpoint	Fault Tree Analysis (FTA)
Notation;	Node, and / or connector and lines
Source	Not applicable

- CBCS viewpoint

Concerns framed by the viewpoint	CBCS concept to control hazard cause
Model types used in this viewpoint	MWF: Functional flow block diagram (FFBD) Or system function diagram (SV-4) MNWF: Inhibit allocation diagram
Notation;	MWF : follow FFBD or SV-4 MNWF : Ad hoc
Source	SV-4 : derived from DoDAF format

	Others : Not applicable

- Safety failure tolerance viewpoint

Concerns framed by the viewpoint	Number of failure to be controlled safely
Model types used in this viewpoint	Table of hazard level and failure number
Notation;	Ad hoc
Source	Not applicable

- Operational failure tolerance viewpoint

Concerns framed by the viewpoint	Number of failure to be controlled operatively
Model types used in this viewpoint	Failure number
Notation;	Ad hoc
Source	Not applicable

- Control system viewpoint

Concerns framed by the viewpoint	Functional and physical architecture
Model types used in this viewpoint	System architecture diagram
Notation;	Node, and / or connector and lines
Source	Not applicable

Architecture Views for Spacecraft Computer Control System

Overview. Each viewpoint has one architecture view. The architecture view includes following five items in accordance with the next IEEE1471:

a) a version identifier;

b) overview information as specified by the organization or project;

c) configuration control information as specified by the organization or project;

d) architecture models addressing all of the concerns framed by its governing viewpoint and covering the whole system from that viewpoint;

e) recording of any known issues within a view with respect to its governing viewpoint.

The examples of each architecture views are described in this section.

Hazard / Cause view. This is the first view (V-1) for the spacecraft computer control system.

Unique identifier: V-1

Overview: This view shows the hazards and their causes.

Configuration information: Version 1.0

Model name (identifier): The hazards and the causes identification diagram model (MV-1) is shown in Figure 8. Model type: Fault Tree Analysis.

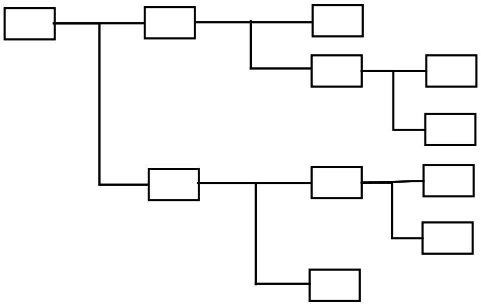

Figure 8: Model MV-1

CBCS view. This is the second view (V-2) for the spacecraft computer control system.

Unique identifier: V-2

Overview: This view shows the CBCS control concept (MWF or MNWF). In case of MWF, the functional structure to realize MWF is shown. And in case of MNWF, inhibit allocation and the inhibit control path are shown.

Configuration information: Version 1.0

Model name (identifier): The CBCS control concept model (MV-2) is shown in Figure 9. Model type: Functional flow diagram (MWF) or data flow diagram (MNWF)

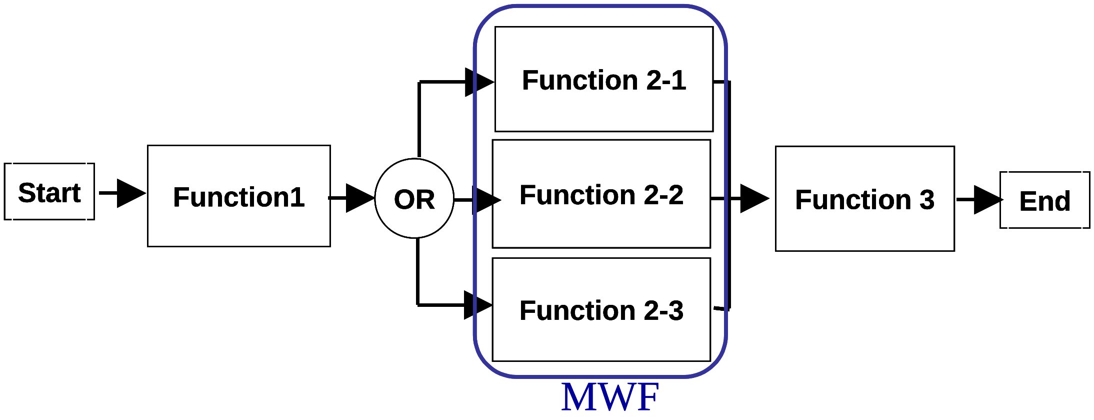

Figure 9: Model MV-2

Safety failure tolerance view. This is the third view (V-3) for the spacecraft computer control system.

Unique identifier: V-3

Overview: This view shows how many failure shall be controlled with respect to hazard level.

Configuration information: Version 1.0

Model name (identifier): The safety failure tolerance model (MV-3) is shown in Figure 10.

Model type: hazard level and failure tolerance requirement spreadsheet.

Hazard Level	**Failure Tolerance Req.**
Catastrophic Hazard	Two failure tolerance
Critical Hazard	One failure tolerance

Figure 10: Model MV-3

Operational failure tolerance view. This is the fourth view (V-4) for the spacecraft computer control system.

Unique identifier: V-4

Overview: This view shows how many failures shall be considered to continue a mission.

Model name (identifier): The operational failure tolerance model (MV-4) is shown in Figure 11. Model type: operational failure tolerance requirement spreadsheet.

Operation	**Failure Tolerance Req.**
Data collection	Two failure tolerance
Auto targeting	One failure tolerance

Figure 11: Model MV-4

Control System view. This is the fifth view (V-5) for the spacecraft computer control system.

Unique identifier: V-5

Overview: This view shows that what function is allocated to which subsystem.

Configuration information: Version 1.0

Model name (identifier): The control system model (MV-5) is shown in Figure 12. Model type: system flow diagram.

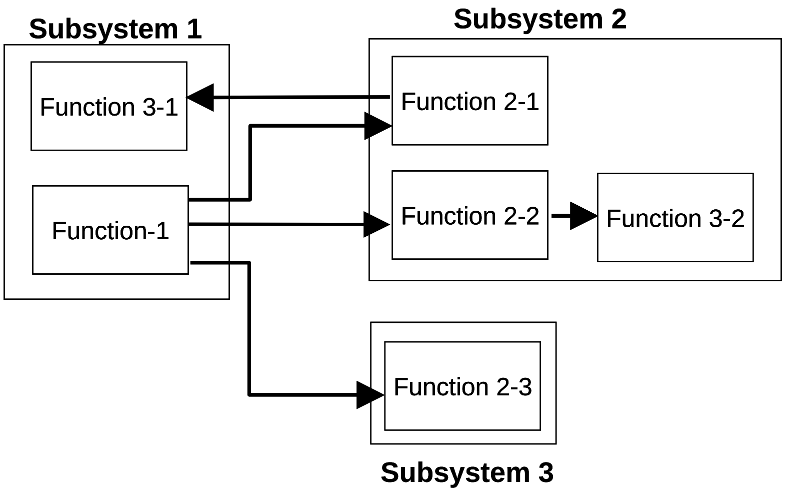

Figure 12: Model MV-5

Conclusion

We develop the architecture framework for spacecraft computer control safety system to guide a safety systems engineer to design computer controlled safety system. This framework is based on the general safety design process and NASA’s computer safety requirements. The architecture framework can be proposed in accordance with the coming new IEEE1471 standard. This architecture framework can cover all design processes which are mandatory to be conducted to design system safety architecture. From this result, we suppose that this architecture framework reaches at least mandatory level. However, we have to evaluate this architecture framework by using it for designing.

Future Work

We are applying this architecture framework to a virtual manned visiting vehicle safety system architecture design. After the trial application is completed, the results will be evaluated as the next step. We will improve the architecture framework and also evaluate the quality of the architecture description which is generated from this architecture framework.

References

IEEE. 2000. IEEE1471-2000 Systems and software engineering — Recommended practice for architectural description of software-intensive systems

IEEE. 2009. IEEE P42010/D6 Systems and software engineering — Architecture description

NASA. 2000. International Space Station Program 30559 Safety Review Process Revision B

NASA. 2000. International Space Station Program 30309 Safety Analysis and Risk Assessment Requirements

NASA. 1995. International Space Station Program 50038 Computer Based Control System Safety Requirements Revision B

Rechtin, E. and Maier, M. W.1996. The Art of Systems Architecting. CRC Press.

Shirasaka, S. A. 2007. A Standard Approach To Find Out Multiple View Points To Describe An Architecture Of Social Systems - Designing Better Payment Architecture To Solve Claim-Payment Failues Of Japan’s Insurance Companies -. In Proceedings of the 19th Annual International Symposium of the International Council on Systems Engineering (Singapore). Seattle: INCOSE.

U.S. Department of Defense. 2007. DoD Architecture Framework v.1.5

U.S. Department of Defense. 1997. Federal Enterprise Architecture Framework v.1.1

Rechtin, E. and Maier, M. W.1996. The Art of Systems Architecting. CRC Press.

Biography

Seiko Shirasaka is Visiting Associate Professor of Graduate School of System Design Management, Keio University. In 1994 he earned a Master’s degree in Astronautics from University of Tokyo and immediately joined in Mitsubishi Electric Corporation. Since then he worked for several space system development projects as a systems engineer. He was a leading systems engineer of HTV (H-II Transfer vehicle) which was launched in 2009 September and completed the mission successfully.