

---

# SCDL introduction

@ASAM WS in Munich  
20180905

SCN-SG  
Shuhei YAMASHITA (DNV GL)

# Plot

---

- What is SCDL ?
- Why we need SCDL ?
- Grammar example of SCDL

---

# What is SCDL?

# What is SCDL ?

---

- Safety Concept Description Language
- Modeling language for SR / SC
  - following ISO 26262 original intention
  - considering context and semantics of the standard
- Especially focusing on key factors such as SR, SC, Element, ASIL, dependency and interaction between SRs, decomposition, FFI and so on.
- Characterized with function block diagram base graphical expression

SR : Safety Requirement  
SC : Safety Concept  
FFI : Freedom From Interference

# What is SCDL ?

---

- SCDL specification is open to public on the SCN-SG web site.
- SCN-SG has been studying grammar of SCDL, it's use-cases, meta-models and other topics on a voluntary basis since 2015.

SCN-SG : Safety Concept Notation Study Group

---

# Why we need SCDL ?

# Why we need SCDL ?

- ISO 26262 requires Semi formal notation for SR specification especially for higher ASIL : C & D with ++. (e.g. Part8-6)

**Table 1 – Specifying safety requirements Methods**

	ASIL			
	A	B	C	D
1a Informal notations for requirements specification	++	++	+	+
<b>1b Semi-formal notations for requirements specification</b>	+	+	++	++
1c Formal notations for requirements specification	+	+	+	+

SR : Safety Requirement

# Why we need SCDL ?

- What is semi formal notation in the context of the standard?

## From ISO 26262 Part 1 :

### 1.117 semi-formal notation

description technique whose syntax is completely defined but whose semantics definition can be incomplete

EXAMPLE System Analysis and Design Techniques (SADT); Unified Modeling Language (UML).

### 1.118 semi-formal verification

**verification** (1.137) that is based on a description given in **semi-formal notation** (1.117)

EXAMPLE Use of test vectors generated from a semi-formal model to test that the **system** (1.129) behaviour matches the model.



# Why we need SCDL ?

- In short;

method	syntax	semantics
Informal notation [1.63]	definition can be incomplete	definition can be incomplete
Semiformal notation [1.117]	completely defined	definition can be incomplete
Formal notation [1.47]	completely defined	completely defined

## Why we need SCDL ?

---

That's why many safety engineers have been trying to use SysML or other existing general purpose languages.


- Consequences were unfortunate : usage of general purpose languages creates various expression even for very simple concept.
- This forces many safety engineers to tolerate unnecessary additional work load.
- → Came up with an idea of safety concept oriented language which can help effective and efficient functional safety development.

SysML : System Modeling Language

# Why we need SCDL ?

- SCDL made semantics complete;

method	syntax	semantics
Informal notation [1.63]	definition can be incomplete	definition can be incomplete
<b>'General purpose'</b> Semiformal notation [1.117]	completely defined	definition can be incomplete
Formal notation [1.47]	completely defined	completely defined
<b>'SC oriented'</b> <b>SCDL as semiformal notation</b>	<b>completely defined</b> <b>(with a metamodel)</b>	<b>completely defined</b>

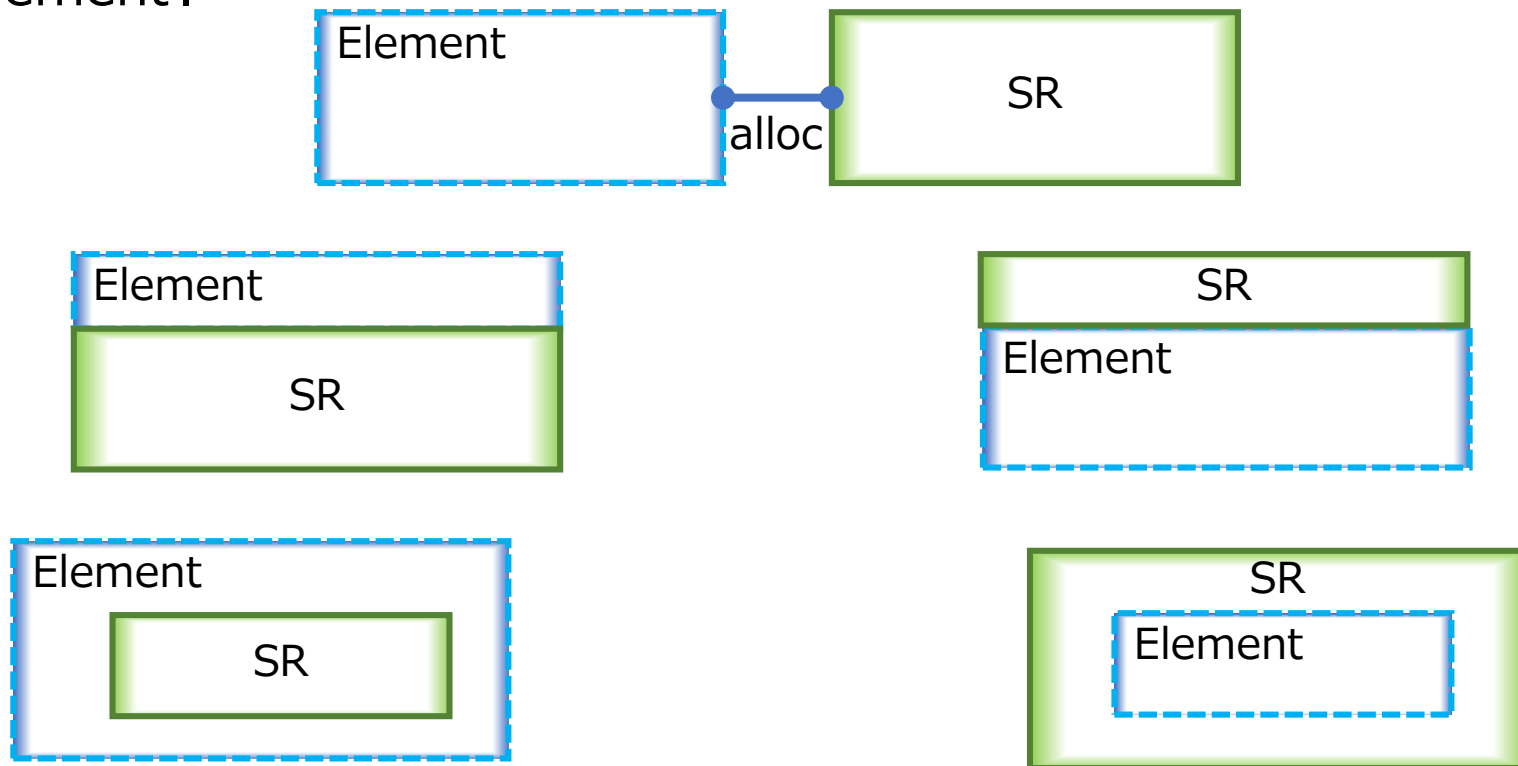


---

# Grammar example of SCDL

# Grammar Example of SCDL

SysML allows following all combinations for 'SR allocation on Element'.

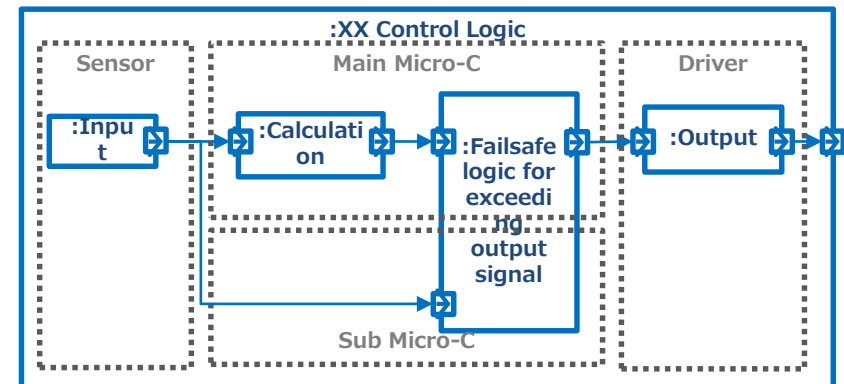
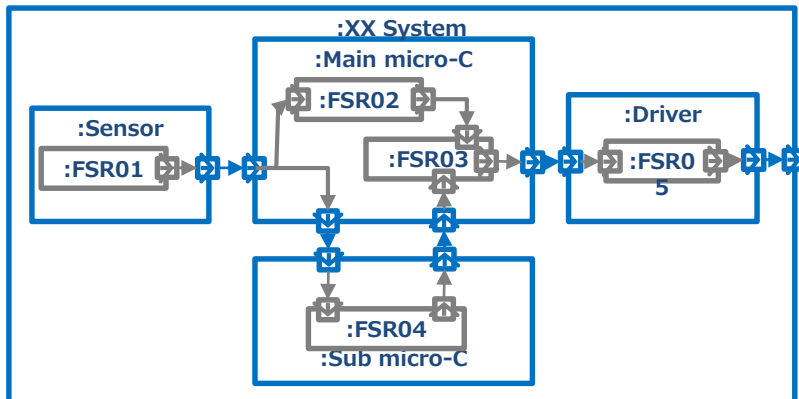
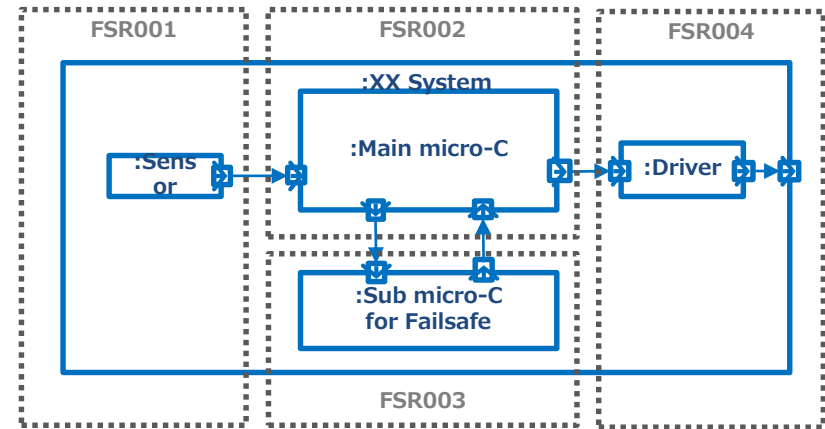


SysML : System Modeling Language  
 SR : Safety Requirement

# Grammar Example of SCDL

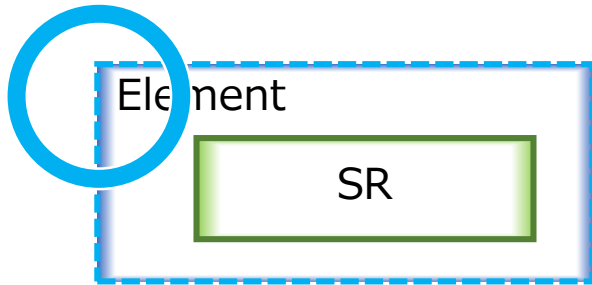
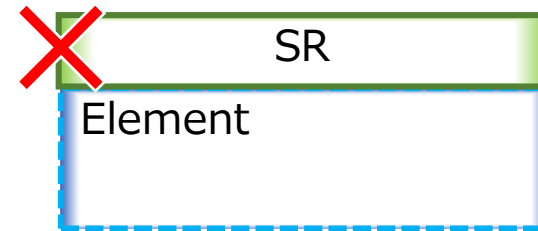
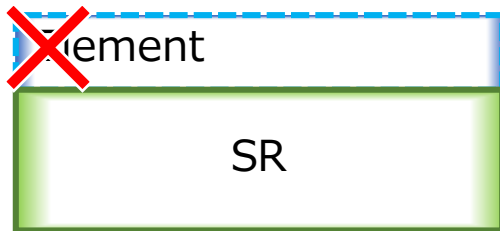
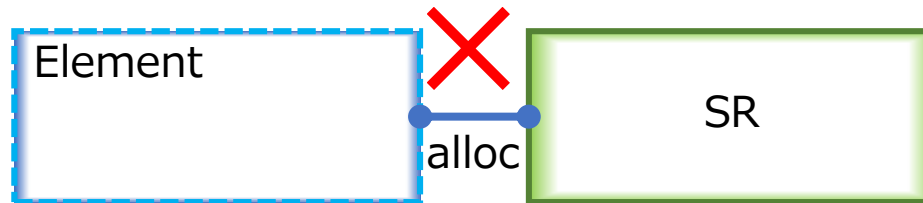
A variety of ways for SC expression can happen easily.

SC : Safety Concept



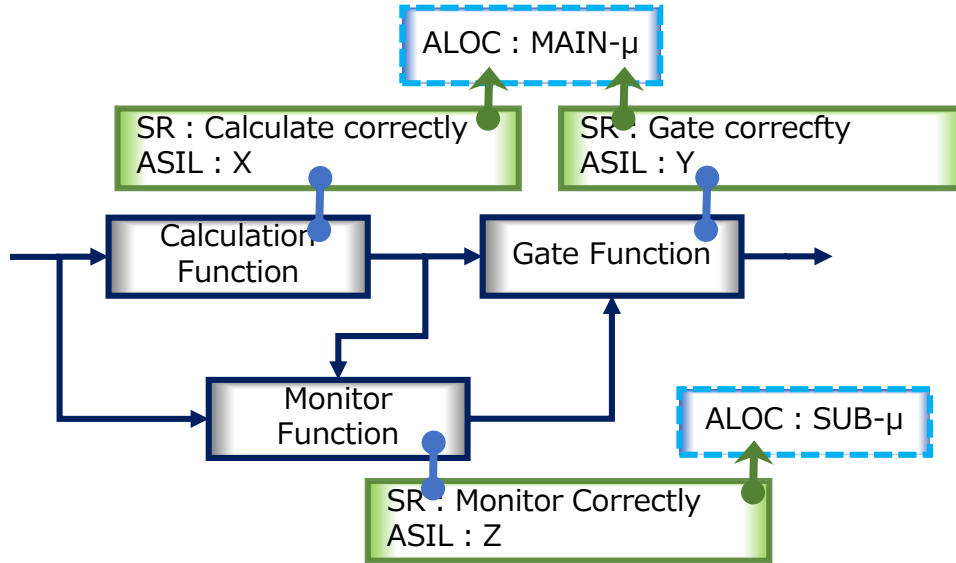
# Grammar Example of SCDL

SCDL chose only one expression for 'SR allocation on Element'.



SR : Safety Requirement

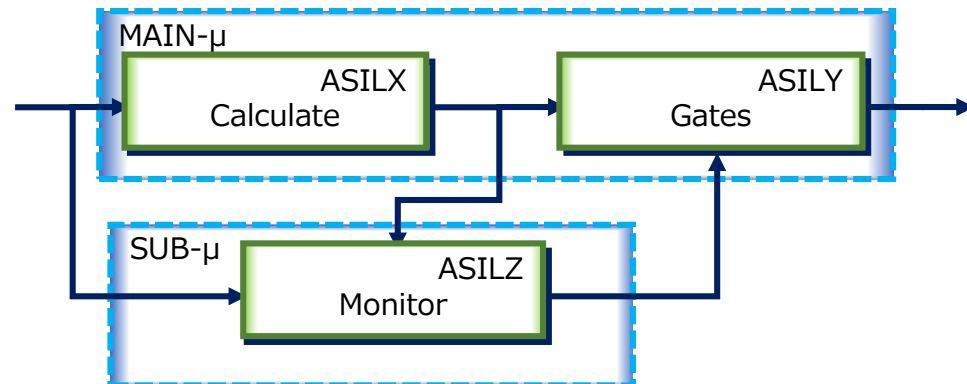
# Grammar Example of SCDL



Typical expression for SC based on Function Block Diagram before SCDL

SCDL could simplify SC expression.

SCDL expression for SC

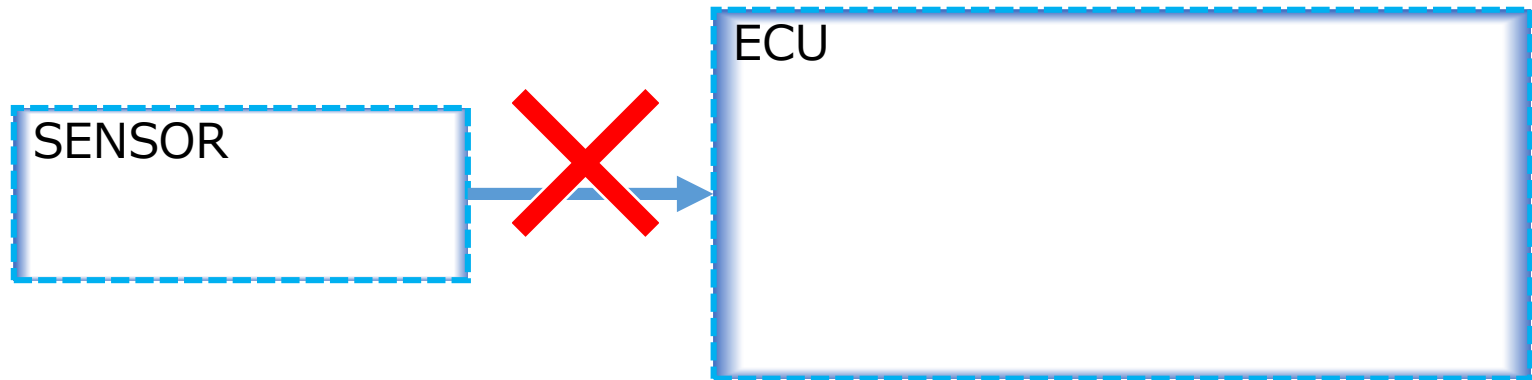


SC : Safety Concept  
SR : Safety Requirement

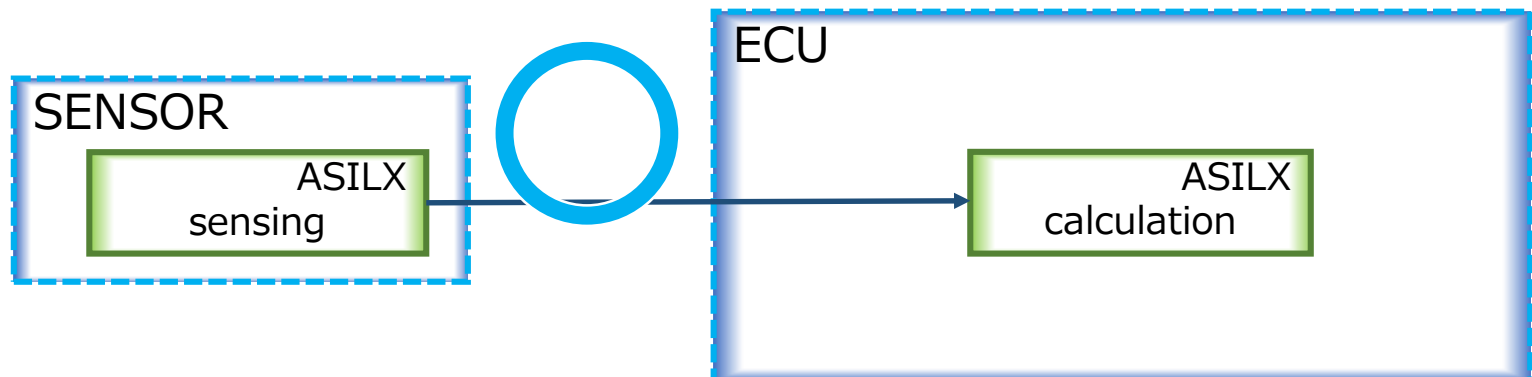


# Grammar Example of SCDL

Do not connect elements each other with an arrow.

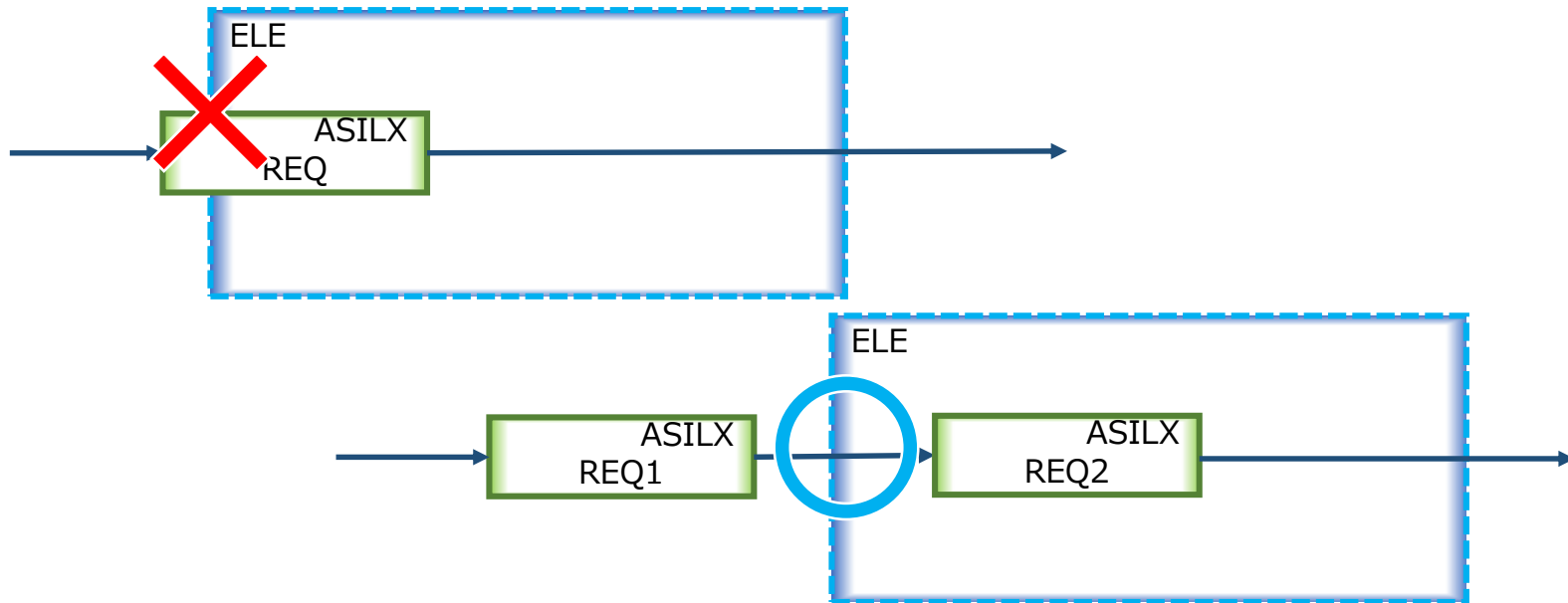


Connect their requirements.



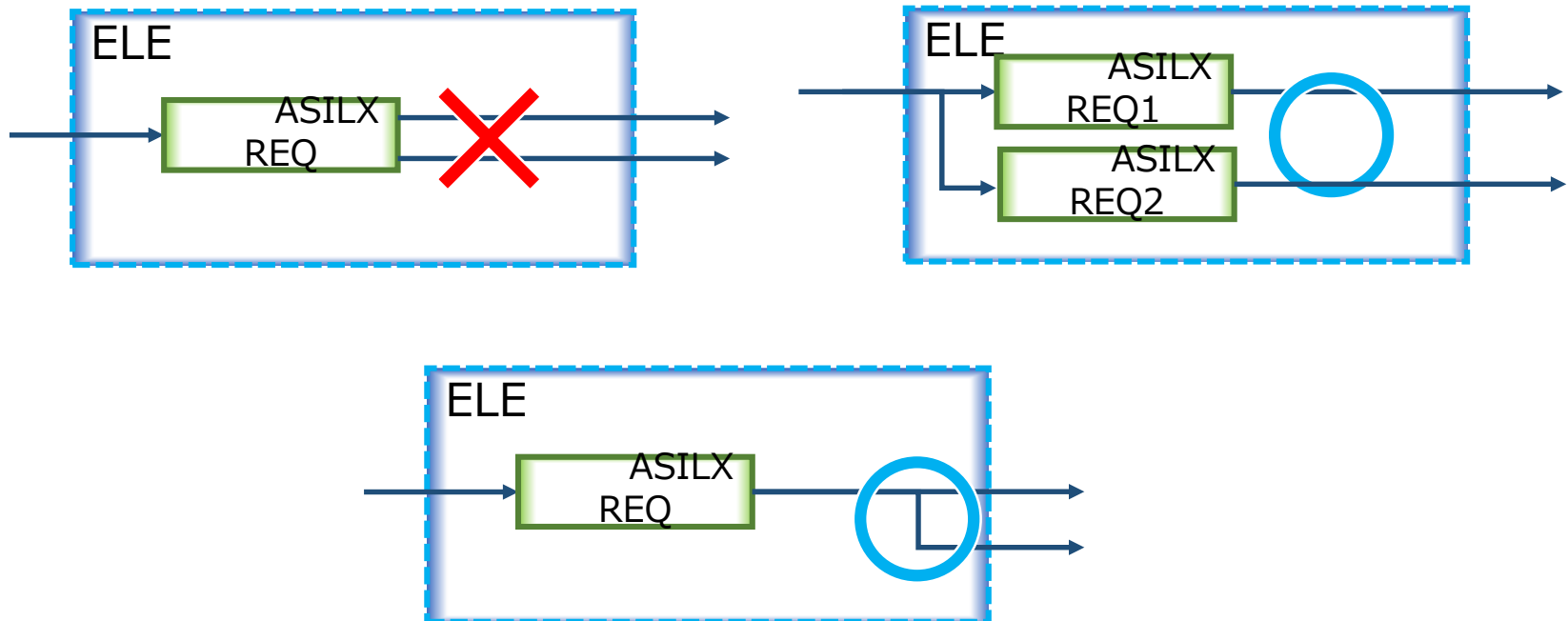
# Grammar Example of SCDL

Do not cross the element boundary with the requirement.  
(to keep the requirements' granularity appropriate)



# Grammar Example of SCDL

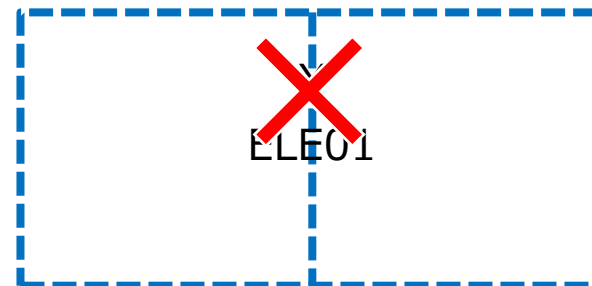
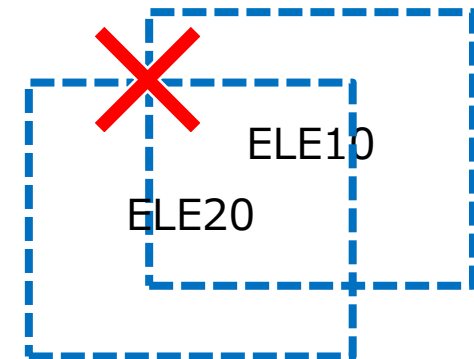
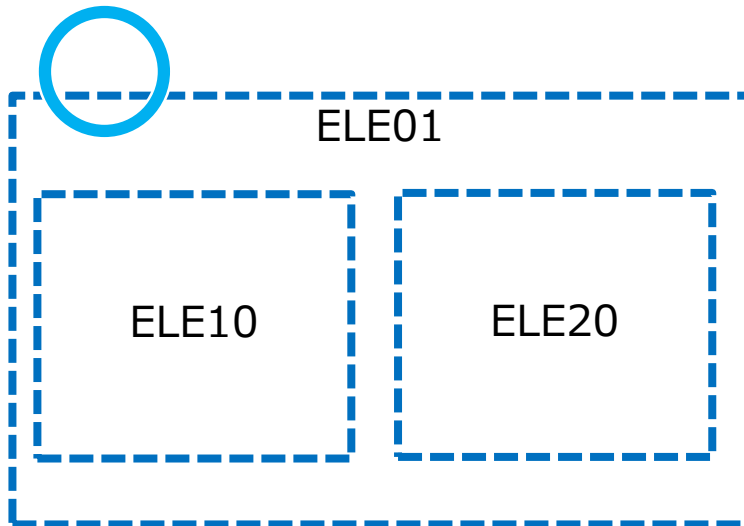
Do not give multiple output to a single requirement.  
 (to keep the requirements appropriately atomic)



# Grammar Example of SCDL

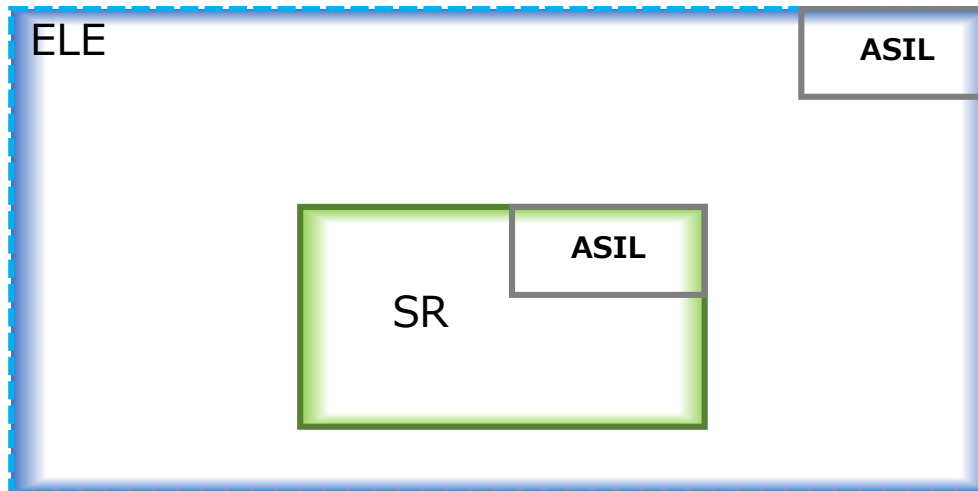
Do not cross any element boundary line with another Element.

Do not divide one element into multiple elements.



# Grammar Example of SCDL

A place holder for ASIL is defined for both Element and SR.

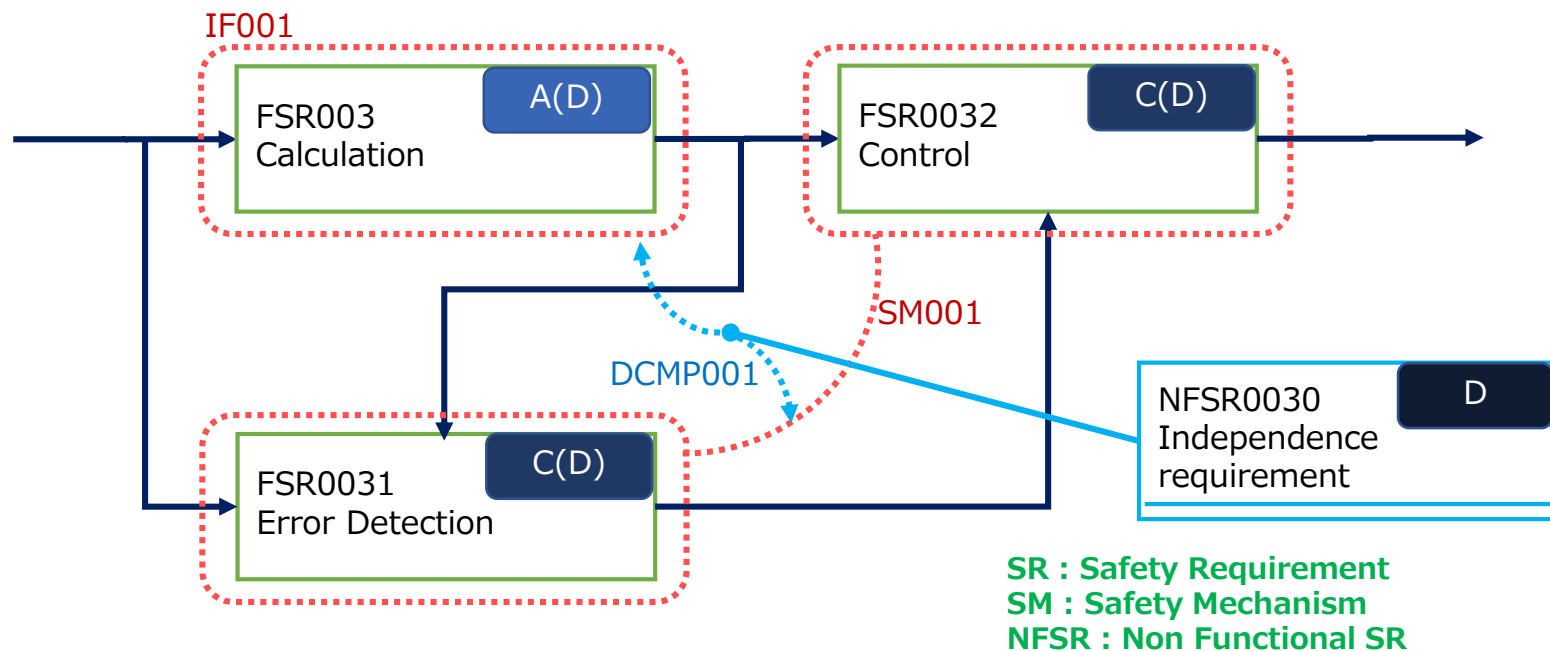


SR : Safety Requirement

# Grammar Example of SCDL

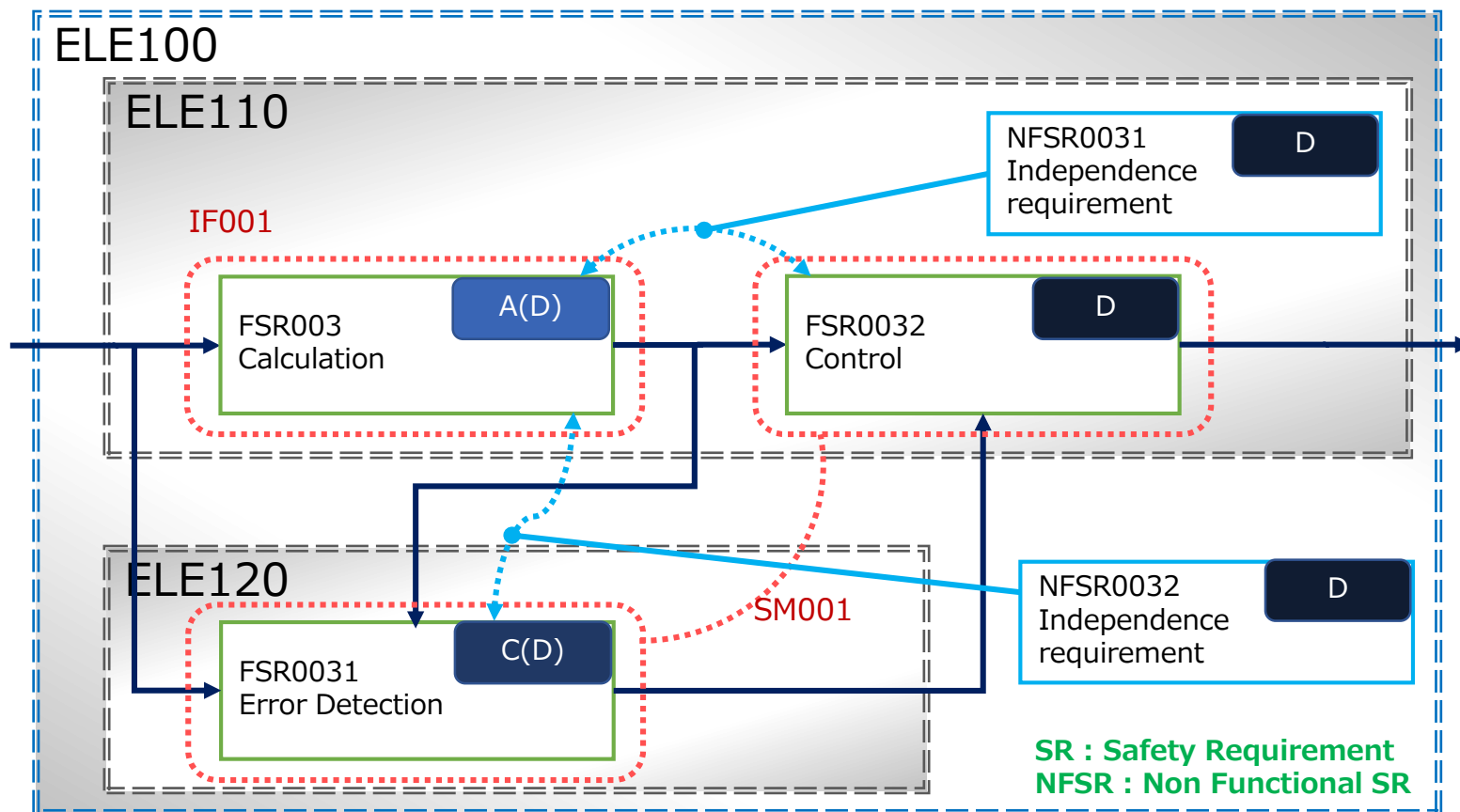
Expression for SR structure of SM is following decomposition logic perfectly :

- Interactions among SRs (e.g.; Detection & Control)
- Pairs of redundant SR groups (DCMP001 = IF001 + SM001)
- Independence requirements (NFSR0030; floating)



# Grammar Example of SCDL

Example of results of decomposition : The diagram indicates SRs allocation on Elements including independence requirement.

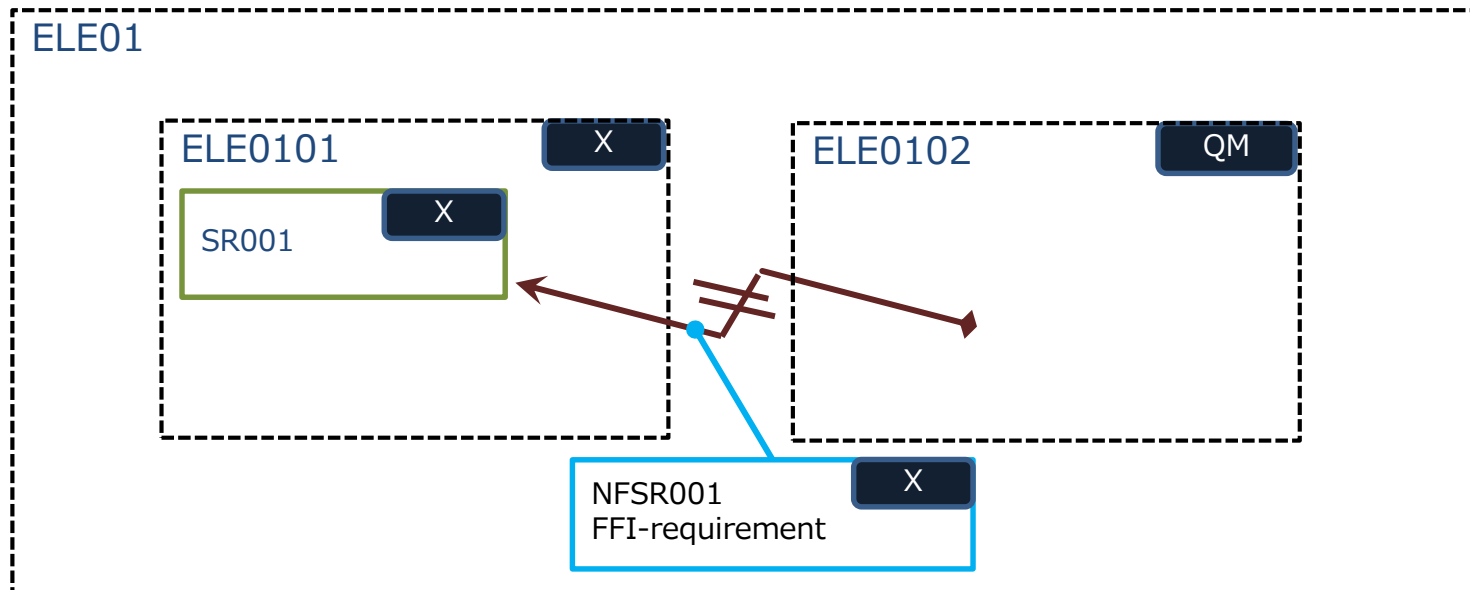


SR : Safety Requirement  
NFSR : Non Functional SR

# Grammar Example of SCDL

FFI requirement expression following 'Criteria for Coexistence'.

FFI : Freedom From Interference

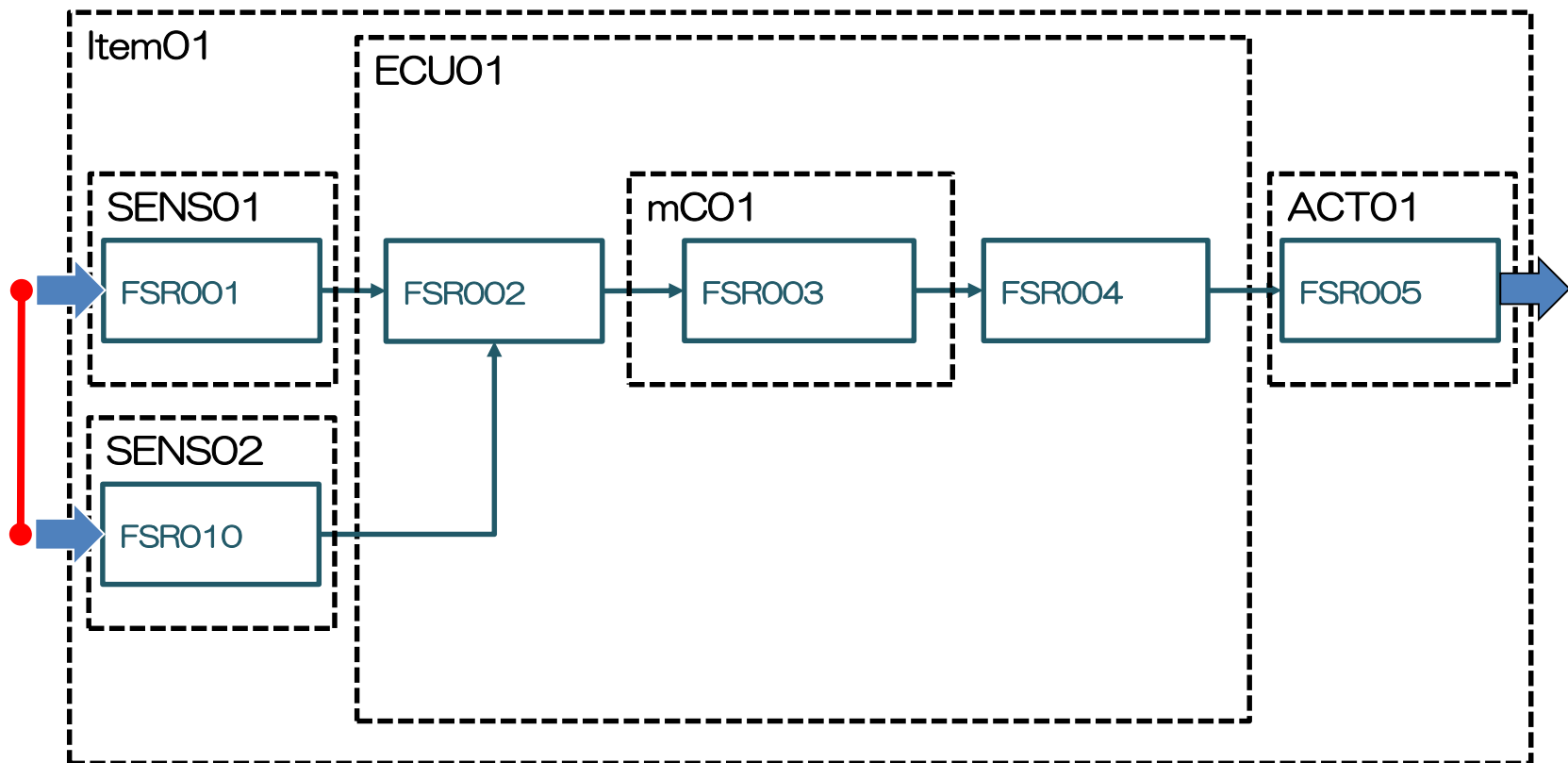




# Grammar Example of SCDL

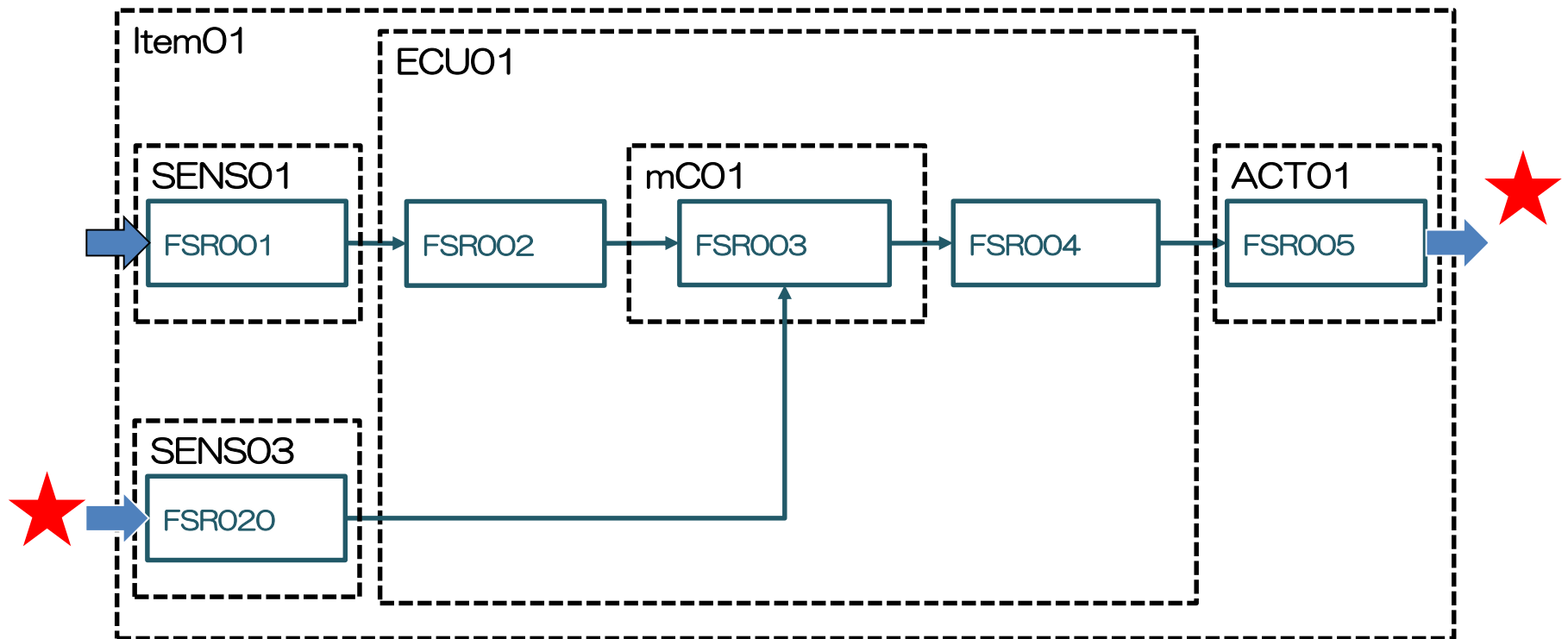
A bridge symbol between two input block arrow stands for OT link.  
e.g. a mechanical link for two redundant sensors

OT : Other Technology



# Grammar Example of SCDL

A pair of star symbols is used to stand for 'External Plant',  
e.g. Vehicle, Engine, Motor



---

## Q & A