***Software Design Specification***

***Version***

**Project Name: *OOO***

**[*Target* / *Target version*]**

**Revision History**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Version** | **Date** | **Description** | **Author** | **Reviewer** | **Approver** |
| Ver. | YYYY/MM/DD | Draft | GD Hong | GD Hong | GD Hong |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Contents**

[1. Introduction 4](#_Toc411526900)

[**1.1** **Purpose** 4](#_Toc411526901)

[**1.2** **Scope** 4](#_Toc411526902)

[**1.3** **Terms, Abbreviations and Definitions** 4](#_Toc411526903)

[**1.4** **Reference** 4](#_Toc411526904)

[2. Software states definition 5](#_Toc411526905)

[**2.1** **Software state transition diagram** 5](#_Toc411526909)

[**2.2** **Software state description** 5](#_Toc411526910)

[**2.3** **Software state transition description** 5](#_Toc411526911)

[**2.3.1** **Condition Definition** 6](#_Toc411526917)

[**2.3.2** **Action Definition** 6](#_Toc411526918)

[3. Software static architecture 6](#_Toc411526919)

[**3.1** **Software component Diagram** 6](#_Toc411526920)

[**3.2** **Software partition information** 7](#_Toc411526921)

[**3.3** **Software interface error handling** 7](#_Toc411526922)

[**3.4** **Software component information** 7](#_Toc411526923)

[**3.4.1** **Software component name** 8](#_Toc411526924)

[**3.4.2** **Software component name** 9](#_Toc411526925)

[**3.5** **Software component organization** 10](#_Toc411526926)

[4. Software dynamic architecture 11](#_Toc411526927)

[**4.1** **Task scheduling summary** 11](#_Toc411526928)

[**4.2** **Behavior** 12](#_Toc411526929)

[**4.2.1** **xxx task behavior** 12](#_Toc411526930)

[**4.2.2** **xxx task behavior** 12](#_Toc411526931)

# **Introduction**

## **Purpose**

*이 항목은 다음 사항이 기술하여야 한다.*

1. *문서의 목적을 정확하게 기술한다.*
2. *문서를 사용하는 대상을 지정한다.*

## **Scope**

*이 항목은 다음 사항을 기술하여야 한다.*

1. *산출물의 범위 및 제약사항을 설명한다.*
2. *산출물의 적용 범위를 기술한다.*

## **Terms, Abbreviations and Definitions**

*이 항목은 문서를 이해하기 위해 필요한 모든 용어 및 약어를 기술한다.*

*예)*

|  |  |  |
| --- | --- | --- |
| *Acronym.* | *Terms* | *Definition (or Description)* |
| *HCA* | *Hydraulic Clutch Actuator* | *유압 클러치 엑츄에이터* |

## **Reference**

*이 항목은 다음 사항을 기술하여야 한다.*

1. *문서에 모든 곳에서 참조된 모든 문서의 리스트를 작성한다.*
2. *문서의 제목, 번호, 날짜, 출판 기관 등의 정보가 기술되어야 한다.*
3. *문서를 볼 수 있는 방법을 명시한다.*

|  |  |  |
| --- | --- | --- |
| *No.* | *Title* | *Note* |
| *[1]* | *Documentation\_ProjectName\_Version* | *문서화 템플릿* |

# **Software states definition**



## **Software state transition diagram**

*\* 시스템 상태 전이도를 기반으로 소프트웨어 관점의 상태 및 전이를 묘사한다. 소프트웨어 관점의 상태 전이를 묘사할 필요가 없을 경우 해당 항목은 N/A로 처리*

|  |
| --- |
| mks|16062/Text%20Attachments/mks1408602012383-0.png||  *<HCA Example>* |

## **Software state description**

*\* 소프트웨어의 상태를 정의하고 기술한다.*

|  |  |  |  |
| --- | --- | --- | --- |
| State\_ID | State Name | Safe State | Description |
|  |  |  |  |
|  |  |  |  |

*State\_ID: 소프트웨어 상태 아이디를 기술*

*State Name: 소프트웨어 상태 이름을 정의*

*Safe State: 안전 상태인지 여부를 Yes/No로 기술*

*안전 상태의 정의는 소프트웨어가 안전 목표를 위반하지 않는 상태 혹은 시스템을 위험에 빠트리지 않는 상태로 정의한다.*

*Description: 소프트웨어 상태에 대한 설명 기술*

## **Software state transition description**

*\* 소프트웨어 상태 전이 과정을 설명하기 위한 조건을 기술한다.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Transition\_ID | Current\_State | Condition | Next\_State | Action | Description |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

*Transition\_ID: 전이 ID. 식별자 부여 규칙에 따라서 식별자를 기술한다.*

*Current\_State: 현재 상태의 이름*

*Condition: 전이 조건. 이해하기 쉬운 수준으로 기술, 상세한 내용은 3.3.1에 기술*

*Next\_State: 전이가 발생하여 변경되는 상태*

*Action: 상태가 전이되면서 수행되는 행동, 이해하기 쉬운 수준으로 기술, 상세한 내용은 3.3.2에 기술*

*Description: 전이 과정에 대한 설명*



### **Condition Definition**

*\*3.3절에 정의된 condition에 대한 상세한 정의를 기술한다. 3.3 절에서의 기술이 충분할 때에는 N/A로 처리*

|  |  |
| --- | --- |
| Condition | Description |
|  |  |
|  |  |
|  |  |

### **Action Definition**

*3.3절에 정의된 Action에 대한 상세한 설명을 기술한다. 3.3에서 Action에 대한 설명이 충분하면 N/A로 처리 가능*

|  |  |
| --- | --- |
| Action | Description |
|  |  |
|  |  |

# **Software static architecture**

## **Software component Diagram**

*Software의 구조를 파악하고 Software Component의 관계를 식별할 수 있는 다이어그램 제시*

|  |
| --- |
| *예) 기존 다이어그램* |

## **Software partition information**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Partition ID | Partition Name | Mode1) | ASIL | SSR\_ID | Description |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| 1) Mode에는 User Mode와 Supervisor Mode가 있다. | | | | | |

## **Software interface error handling**

소프트웨어의 인터페이스에 대한 처리 방안을 기술한다.

함수 호출 시 데이터를 보내는 쪽에서는 보내는 데이터에 대한 범위 체크

함수 피 호출 시 데이터를 받는 쪽에서는 받는 데이터에 대한 범위 체크

만약 보내는 쪽이나 받는 쪽에서 범위 밖의 데이터가 수신이 될 경우에 대한 에러 처리 전략이 필요함.

## **Software component information**

*다이어그램에서 정의한 software component를 식별하고, 각각의 타입 및 필요 시 부여받은 ASIL을 기술*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SwC ID | SwC Name | Type | ASIL | Note |
|  |  | *New /*  *Reuse with modification/*  *Reuse without modification/*  *COTS* |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

### **Software component name**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Software component information | | | | | | | |
|  | SwC\_ID |  | | | | | |
| SwC name |  | | | | | |
| Type |  | | | | | |
| Description |  | | | | | |
| ASIL |  | | | | | |
| From SwR\_ID |  | | | | | |
| General Information | | | | | | | |
|  | File Name |  | | | | Access Right |  |
| Location |  | | | | | |
| SwC Interface  *컴포넌트의 인터페이스 함수를 정의*  *각 인터페이스를 통해 교환되는 값의 범위 및 유용한 값의 범위 등에 대한 정보 기술* | | | | | | | |
|  | Interface ID | Name | Prototype | Description | | | |
|  |  |  | 1) interface에 대한 기능적 설명을 한다.  2) parameter의 interface정보 및 return의 interface정보를 기술한다. | | | |
|  |  |  |  | | | |
|  |  |  |  |  | | | |
| SwC Data  *컴포넌트의 데이터 정의(if any)* | | | | | | | |
|  | Data ID | Name | Type | | Description | | |
|  |  |  |  | | (range, precision)을 포함하여 기술 | | |
|  |  |  |  | |  | | |
| Unit function  *컴포넌트 내 정의된 유닛 함수 기술* | | | | | | | |
|  | Unit ID | Name | Prototype | | Description | | |
|  |  |  | | 1) interface에 대한 기능적 설명을 한다.  2) parameter의 interface정보 및 return의 interface정보를 기술한다. | | |
|  |  |  | |  | | |
|  |  |  | |  | | |
|  |  |  | |  | | |
|  |  |  | |  | | |
| Interface function을 호출하는 함수 정보  *SwC의 interface function을 호출하는 외부 function에 대한 정보를 기술* | | | | | | | |
|  | | | | | | | |
| Interface function으로부터 internal unit함수간의 호출 관계 표현  (control flow및 data flow전부 포함할 수 있는 behavior를 기술함) | | | | | | | |
|  | | | | | | | |
| Verification Criteria  *요구사항으로부터 할당 받은 컴포넌트의 기능에 대한 검증 기준 기술*  *인터페이스를 시험하기 위한 검증 기준 기술* | | | | | | | |
|  | | | | | | | |

### **Software component name**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Software component information | | | | | | | |
|  | SwC\_ID |  | | | | | |
| SwC name |  | | | | | |
| Type |  | | | | | |
| Description |  | | | | | |
| ASIL |  | | | | | |
| From SwR\_ID |  | | | | | |
| General Information | | | | | | | |
|  | File Name |  | | | | Access Right |  |
| Location |  | | | | | |
| SwC Interface  *컴포넌트의 인터페이스 함수를 정의*  *각 인터페이스를 통해 교환되는 값의 범위 및 유용한 값의 범위 등에 대한 정보 기술* | | | | | | | |
|  | Interface ID | Name | Prototype | Description | | | |
|  |  |  | 1) interface에 대한 기능적 설명을 한다.  2) parameter의 interface정보 및 return의 interface정보를 기술한다. | | | |
|  |  |  |  | | | |
|  |  |  |  |  | | | |
| SwC Data  *컴포넌트의 데이터 정의* | | | | | | | |
|  | Data ID | Name | Type | | Description | | |
|  |  |  |  | | (range, precision)을 포함하여 기술 | | |
|  |  |  |  | |  | | |
| Unit function  *컴포넌트 내 정의된 유닛 함수 기술* | | | | | | | |
|  | Unit ID | Name | Prototype | | Description | | |
|  |  |  | | 1) interface에 대한 기능적 설명을 한다.  2) parameter의 interface정보 및 return의 interface정보를 기술한다. | | |
|  |  |  | |  | | |
|  |  |  | |  | | |
|  |  |  | |  | | |
|  |  |  | |  | | |
| Interface function을 호출하는 함수 정보  *SwC의 interface function을 호출하는 외부 function에 대한 정보를 기술* | | | | | | | |
|  | | | | | | | |
| Interface function으로부터 internal unit함수간의 호출 관계 표현  (control flow및 data flow전부 포함할 수 있는 behavior를 기술함) | | | | | | | |
|  | | | | | | | |
| Verification Criteria  *요구사항으로부터 할당 받은 컴포넌트의 기능에 대한 검증 기준 기술*  *인터페이스를 시험하기 위한 검증 기준 기술* | | | | | | | |
|  | | | | | | | |

## **Software component organization**

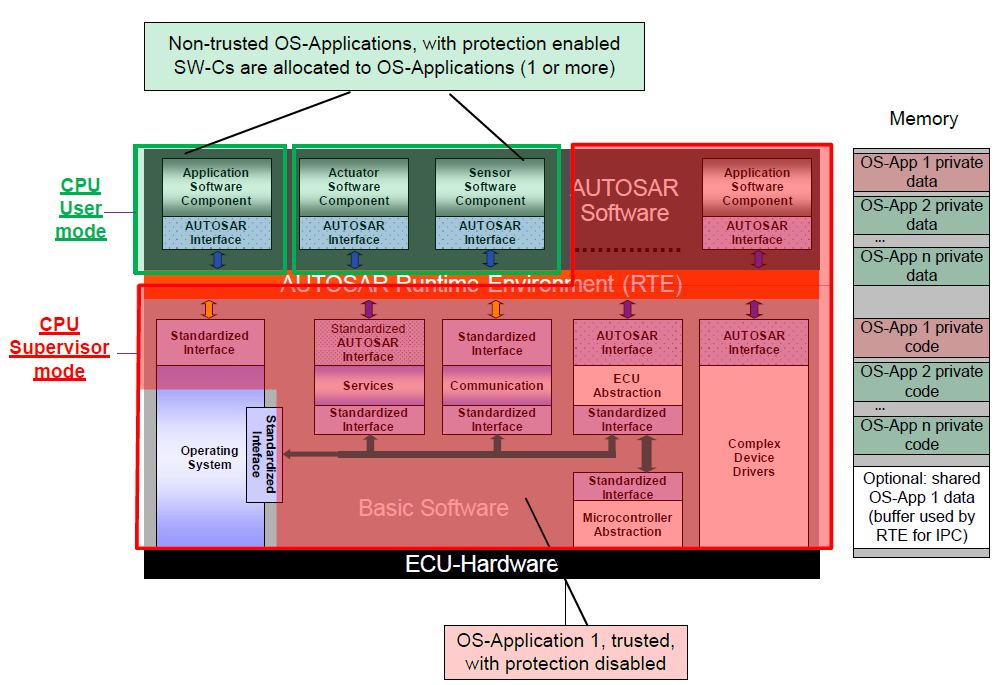
*Layered View에서는 다음의 정보를 포함하여 기술한다.*

*1) Application SwC외에 BSW SwC도 포함하여 표현한다.*

*2) 파티셔닝을 한다면 파티셔닝에 대한 정보를 표현한다.*

*3) User Mode/Supervisor Mode에 대한 정보를 표현한다.*

*예제: 아래그림과 같다*



# **Software dynamic architecture**

## **Task scheduling summary**

*Task와 SwC의 관계에 따라서 내용 구성이 많이 달라질 수 있음*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Task Scheduling Information** | | | | | | |
|  | Task Scheduling Policy | | *Rate monotonic, Cyclic Executive, Earliest Deadline First* | | | |
| Comment | |  | | | |
| **Task Information** | | | | | | |
|  | Task ID | Task Name | | Priority1) | Period(ms) | Deadline |
|  |  | |  |  |  |
|  |  | |  |  |  |
|  |  | |  |  |  |
|  |  | |  |  |  |
|  |  | |  |  |  |
|  |  | |  |  |  |
| 1) 가능하면 각 task간 서로 다른 우선순위를 갖도록 한다. | | | | | |

*periodic task들에 대한 diagram*

|  |
| --- |
| task\_B  task\_A  100ms  10ms |

## **Behavior**

*Stateflow로 표현하거나 UML의 sequence diagram으로 표현하여 기술한다.*

*소프트웨어 요구사항에 대한 동작 시나리오와 관련된 행동을 컴포넌트간 interaction으로 표현하고자 할 경우 sequence diagram으로 작성하고, 컴포넌트 자체에 대한 behavior를 표현하고자 하는 경우는 Stateflow로 표현한다.*

### **xxx task behavior**

|  |  |  |  |
| --- | --- | --- | --- |
| SwT\_ID(Task ID) |  | Task Name |  |
| Priority |  | From SwR\_ID |  |
| ASIL |  |  |  |
| Description | *Task behavior description(natural language description)* | | |
| Behavior  (semi-formal notation) | *Semi-formal notation(Flowcharts, FSM, Stateflow등의 notation)*  *주의사항. Software unit 내부의 behavior를 기술하지 않는다.* | | |
| Behavior  (Natural language) |  | | |
| Constraints, if required |  | | |
| Verification Criteria |  | | |

### **xxx task behavior**

|  |  |  |  |
| --- | --- | --- | --- |
| SwT\_ID(Task ID) |  | Task Name |  |
| Priority |  | From SwR\_ID |  |
| ASIL |  |  |  |
| Description | *Task behavior description(natural language description)* | | |
| Behavior  (semi-formal notation) | *Semi-formal notation(Flowcharts, FSM, Stateflow등의 notation)*  *주의사항. Software unit 내부의 behavior를 기술하지 않는다.* | | |
| Behavior  (Natural language) |  | | |
| Constraints, if required |  | | |
| Verification Criteria |  | | |

<End of document>