

KOROSKOR
OROSKOR
ROSKOR
OSKOR
SKOR
KOR
OR

KOROS

서비스 로봇을 위한 모듈 - 제2-4 부 :

센서모듈을 위한 정보모델

KOROS XXXX : YYYY

한국로봇산업협회

YYYY년 MM월 DD일 제정

<http://www.KOROS.or.kr>

심 의 : 지능형로봇표준포럼 운영위원회

	성	명	근 무 처	직	위
(운 영 위 원 장)	문	승	빈	세종대학교	교 수
(운 영 위 원)	이	순	걸	경희대학교	교 수
	우	종	운	한국로봇산업진흥원	센 터 장
	곽	관	웅	세종대학교	교 수
	조	영	조	한국전자통신연구원	책 임 연 구 원
	김	동	한	경희대학교	교 수
	정	영	숙	한국전자통신연구원	책 임 연 구 원
	전	진	우	한국로봇산업진흥원	실 장
	임	성	수	경희대학교	교 수
	문	인	혁	동의대학교	교 수
	권	용	관	대림대학교	교 수
	김	규	로	국가기술표준원	국 장
	김	석	중	하이젠모터	이 사
	김	승	훈	전자부품연구원	책 임
	김	혜	지	국가기술표준원	주 무 관
	단	병	주	LG전자	수 석 연 구 원
	류	영	선	라스테크	대 표
	문	전	일	한국로봇산업진흥원	원 장
	박	종	환	넬판	대 표
	박	홍	성	강원대학교	교 수
	서	준	호	한국로봇산업협회	팀 장
	서	태	원	로보테크	부 장
	심	재	홍	한국산업기술대학교	교 수
	유	원	필	한국전자통신연구원	책 임 연 구 원
	이	병	우	한국산업기술시험원	전 문 위 원
	이	용	국	현대로보틱스	수 석 연 구 원
	임	우	철	두산로보틱스	선 임
	장		민	유진로봇	이 사
	정	병	찬	한화정밀기계	수 석 연 구 원
	지	수	영	한국전자통신연구원	책 임 연 구 원
(간 사)	유	기	은	한국로봇산업협회	대 리

표준작성 기여자

	성	명	근 무 처	직	위
(과제제안자)	O	O	O	0000000000000000	O
(작성참여자)	O	O	O	0000000000000000	O
	O	O	O	0000000000000000	O
	O	O	O	0000000000000000	O
	O	O	O	0000000000000000	O
	O	O	O	0000000000000000	O

목 차

1	적용범위	4
2	인용표준	4
3	용어와 정의	4
4	모듈의 공통 정보 속성	6
4.1	개요	6
4.2	공통정보모델과 특수 정보모델과의 관계	8
4.3	센서모듈용 정보모델을 위한 클래스	8
부속서 A (규정)	센서 모듈 아이디 할당 규칙	15
부속서 B (규정)	센서모듈의 정보모델 표기법	17
B.1	개요	17
B.2	모듈의 일반정보	17
B.3	Module ID에 대한 정보	17
B.4	속성, 입력, 출력 정보	18
B.5	모듈의 서비스 정보	20
B.6	모듈의 인프라 정보	20
B.7	모듈의 안전과 보안의 정보	21
B.8	모델링 정보	23

서 문

표준의 목적

본 표준은 로봇에 사용되는 센서 모듈의 공통 특성을 명시하는 정보모델을 제공한다. 정보 모델로써 센서 모듈의 클래스 다이어그램과 관련 속성과 메소드들을 제공하는 XML 형식의 명시 방법을 제시한다. 이를 통하여 센서 모듈의 개발자 및 사용자들이 해당 모듈들을 로봇에 보다 쉽게 적용할 수 있도록 도와준다.

본 표준에서는 필수 항목은 “~하여야 한다.” 형태로, 권고사항은 “~하는 것이 좋다.” 혹은 “~하여야 할 것이다.” 형태로, 허용 항목은 “~해도 된다.” 형태로, 마지막으로 가능성 항목은 “~ 할 수 있다.” 형태로 문장을 서술한다. 이외에 대등한 문장은 KS A 001 Annex H를 참조하여 작성한다.

타 표준과의 관련성

타 국내표준과의 관련성

KOROS 1067-3:2018 개방형 로봇 소프트웨어 플랫폼 제3부: 프로파일
 KOROS 1106-2:2019 서비스 로봇용 기계적·전기적 모듈화-제2부: 하드웨어 속성
 KOROS 1149-1:2020 서비스 로봇을 위한 모듈 — 제1부: 일반요구사항
 KOROS 1148:2021 서비스 로봇을 위한 모듈 — 제2-1부 : 모듈을 위한 공통 정보 모델

타 국제표준과의 관련성

ISO 22166-1 Modularity for service robots – Part1 : General requirements
 ISO 13482:2014 Robots and robotic devices – Safety requirements for personal car robots
 ISO 13489-1:2015 Safety of machinery – Safety-related parts of control systems – Part 1 : General principles for design
 IEC 62443-4-2, Security for industrial automation and control systems – Part 4-2: Technical security requirements for IACS components
 ISO/IEC 19505-1: 2012 Information technology – Object management group unified modelling language (OMG UML) – Part 1: Infrastructure
 ISO/IEC 19516:2020 Information technology – Object management group – interface definition language (IDL) 4.2
 IEC 60529 Ed.2.2: 2013 Degrees of protection provided by enclosures (IP code)

지적재산권 인지 관련사항

해당사항 없음

적합인증 관련사항

해당사항 없음

표준의 이력

본 표준은 2020년 08월 28일에 개최된 제56차 지능형로봇표준포럼 운영위원회에서 제정되었다.

본 표준은 2021년 08월 26일에 개최된 제00차 지능형로봇표준포럼 운영위원회에서 KOROS XXXX:YYYY를 개정하였다.

서비스 로봇을 위한 모듈 - 제2-4 부 : 센서 모듈을 위한 정보모델

Modularity for service robots – Part 2-4 : Information Model for Sensor Modules

1 적용범위

본 표준은 KOROS 1148:2021 서비스 로봇을 위한 모듈 — 제2-1부 : 모듈을 위한 공통 정보 모델을 준용하여 센서 모듈에 공통적으로 적용될 정보모델을 위한 요구사항 및 가이드라인을 제시한다. 센서 모듈용 정보모델이 상호호환성과 재사용성에 어떻게 연관되는지를 제시하고 안전과 보안에 대해서 설명한다. 정보 모델로써 센서 모듈의 클래스 다이어그램과 관련 속성 및 메소드들을 제공하는 XML 형식의 명시 방법을 제시한다. 이 표준은 센서와 액추에이터가 융합된 모듈의 정보모델에는 적용되지 않을 수 있으며, 이 모듈의 센서 부분에만 적용될 수 있다.

센서 모듈은 하드웨어 및 소프트웨어 특성을 가지고 있지만, 소프트웨어 모듈을 가지고 있지 않는 경우가 많다. 이 표준은 소프트웨어 모듈을 가지고 있지 않은 하드웨어 및 소프트웨어 특성을 지닌 모듈(module with HW and SW aspects)로 제한하여 정보모델을 제공한다.

2 인용표준

다음 문서는 전체 또는 일부가 이 문서에서 규범적으로 참조되며 해당 응용 프로그램에 반드시 필요하다. 날짜가 기입된 참고 문헌에 대해서는 인용된 판만 적용된다. 기한이 지난 참조의 경우 참조 문서의 최신 버전 (모든 개정 포함)이 적용된다.

KOROS 1149-1:2020 서비스 로봇을 위한 모듈 — 제1부: 일반요구사항

KOROS 1148:2021 서비스 로봇을 위한 모듈 — 제2-1부 : 모듈을 위한 공통 정보 모델

ISO 13482:2014 Robots and robotic devices – Safety requirements for personal car robots

ISO 13489-1:2015 Safety of machinery – Safety-related parts of control systems – Part 1 : General principles for design

IEC 62443-4-2, Security for industrial automation and control systems – Part 4-2: Technical security requirements for IACS components

ISO/IEC 19505-1: 2012 Information technology – Object management group unified modelling language (OMG UML) – Part 1: Infrastructure

ISO/IEC 19516:2020 Information technology – Object management group – interface definition language (IDL) 4.2

IEC 60529 Ed.2.2: 2013 Degrees of protection provided by enclosures (IP code)

3 용어와 정의

이 표준의 목적을 위하여 다음의 용어와 정의를 적용한다.

3.1

정보모델 (Information model)

IM

동작 환경, 속성 및 함수 그리고 서로 간의 관계를 가진 객체의 추상화와 표현

3.2

공통정보모델 (Common Information model)

CIM

모든 모듈에 공통으로 적용되는 정보모델

3.3

모듈 (module)

시스템 디자인과 통합 그리고 상호운용성 및 재사용성을 도모하기 위하여 속성 프로파일에 수반되어 정의된 인터페이스로 복합 또는 결합한 컴포넌트

3.4

속성 (property)

모듈의 특성 또는 변수

3.5

소프트웨어 모듈 (software module)

SW 모듈

프로그래밍된 알고리즘으로만 구성된 모듈

[출처: KOROS 1149-1:2020 발췌]

3.6

HW-SW 특성 모듈 (module with HW and SW aspects)

기계적 부품과 전기적 회로와 같은 물리적 부품과 소프트웨어 특성을 가지고 있으면서 동시에 외부의 모듈과 데이터 교환을 할 수 있는 통신 인터페이스를 가진 모듈로 하드웨어 및 소프트웨어 특성을 모두 가진 모듈

보기 센서 모듈, 모터 모듈, 컴퓨팅 모듈

[출처: KOROS 1149-1:2020 발췌]

3.7

센서 모듈 (sensor module)

로봇 외부 환경을 인지하기 위하여 외부 환경의 물리량이나 그 변화를 감지하여 해당 신호를 알맞은 데이터로 변환하여 통신 수단을 통하여 전송하는 모듈

보기 카메라 모듈, 라이다 모듈, 초음파 모듈

3.8

하드웨어 모듈 (hardware module)

HW 모듈

기계적 부품과 전기적 회로와 같은 하드웨어 특성을 가지고 있거나 소프트웨어 특성을 동시에 가지고 있지만 외부와의 통신 인터페이스를 가지고 않고 기계적 인터페이스에 의해 연계 가능한 독립적으로 동작하는 모듈

보기 기계적 프레임, 통신기능이 없는 전원공급장치
[출처: KOROS 1149-1:2020 발췌]

3.9

인프라 모듈(infrastructure module)

다른 모듈의 작동을 지원하기 위한 시설 및 리소스를 제공하는 모듈
[출처: KOROS 1149-1:2020 발췌]

3.10

모듈 관리자 (Module manager)

모듈들을 로딩하고 두개 이상의 동일한 모듈 유형에 고유한 모듈 아이디를 할당하는 모듈

3.11

인스턴스 (Instance)

동작하거나 사용되는 프로세스 혹은 객체

3.12

작명 (naming rules)

모듈의 이름을 명하는 규칙

3.13

서비스 (service)

제공되는 인터페이스를 통하여 다른 모듈을 위한 하나 이상의 기능 또는 동작

3.14

필수 서비스 (mandatory service)

모듈에 의해서 제공되는 필수적인 서비스

3.15

선택 서비스 (optional service)

모듈에 의해서 제공되는 필수적이지는 않지만 기능을 효율적으로 활용하거나 부가적으로 제공되는 서비스

3.16

성능 수준 (performance level)

PL

예상되는 조건에서 안전관련 기능을 수행하기 위하여 안전 관련 제어부분의 기능을 명시하는데 사용되는 단계적 구분

[출처: ISO 13849-1:2015 발췌]

3.17

구성요소(component)

분리와 식별이 가능한 요소로, 다른 부분과 결합하여 더 포괄적인 기능을 만드는 요소

[출처: KOROS 1149-1 : 2020 발췌]

4 센서 모듈의 정보 속성

4.1 개요

센서 모듈의 정보모델이 지원하는 속성은 표4-1과 같아야 한다. 표4.1에 있는 속성들의 처리를 위한

센서모듈용 정보모델은 그림4-1과 같다. 표4.1의 공통정보 속성은 4.3에서 자세히 설명한다.

표4-1에 있는 ‘M’ 과 ‘O’의 기호는 필수항목과 선택항목을 의미하며 ‘-’은 해당사항 없음을 의미한다. 센서 모듈의 정보 모델은 HW-SW 특성 모듈의 정보모델과 차이가 있는 것은 다음의 항목들을 작성하지 않는다는 것이다.

- 하드웨어 특성
- 소프트웨어 특성
- 입력
- 출력

여기서 하드웨어 특성 및 소프트웨어 특성은 해당 모듈의 구성 요소로 사용하는 모듈들에 대한 집합이고, 입력 및 출력은 해당 모듈이 모듈외부에 제공하는 입력과 출력에 관련된 변수들이다. 센서 모듈은 센싱하는 HW 및 SW 컴포넌트로만 구성되어 있고, 외부로는 데이터버스와 같은 통신 수단을 통하여 센싱한 데이터를 전송한다. 이에 따라 센서 모듈은 다른 모듈들로 구성될 수가 없으며, 입력 및 출력 변수도 지원할 수가 없다.

표 4.1 — 센서 모듈의 정보 속성과 관련 그룹화

번호	속성 종류	센서 모듈	관련 그룹화/태그 이름
1	모듈이름 (Module Name)	M	GenInfo (or General Information)
2	설명 (Description)	O	
3	제조사 (Manufacturer)	M	
4	예제 (Examples)	O	
5	모듈아이디 (Module ID)	M	IDnType
6	모듈속성 (Module properties)	M	Properties
7	기능성 (Function(capabilities))	M	Services
8	인프라 (Infrastructure)	M	Infra
9	안전/보안 (Safety/Security)	M	SafeSecure
10	모델링 (Modelling)	O	Modelling
비고 1) 센서 모듈용 정보모델에는 하드웨어 특성, 소프트웨어 특성, 입력 과 출력 관련 속성은 사용하지 않는다.			

모듈이름 ‘Module Name’은 모듈의 이름을 명시하는 항목이다.

설명 ‘Description’은 모듈의 어떠한 부분인지, 무엇을 하는지, 어떻게 사용하는지 등의 모듈의 개괄적 설명을 명시한다.

제조사 ‘Manufacturer’는 모듈의 디자이너, 개발자, 제조사의 연락 정보를 제공한다.

예제 ‘Examples’는 모듈의 전형적인 유즈케이스를 명시한다.

모듈아이디 ‘Module ID’는 시스템 내부에서 유일한 아이디이어야하며, 부속서B 명시되었다.

모듈 속성 ‘Module poperties’는 모듈의 초기화에 사용되는 값들이다. 만일 명시된 값이 시스템 에러를 일으키지 않는다면 동작시에도 사용될 수 있다.

기능성 ‘Function(capabilities)’은 센서 모듈에서 생성된 데이터를 얻거나 센서 모듈을 초기화할 때 사용되는 서비스들을 인터페이스 관점에서 정의한다.

인프라 ‘Infrastructure’은 센서 모듈이 연결되는 데이터버스를 정의한다.

안전/보안 ‘Safety/Security’는 KOROS 1148:2021를 참조한다. 센서 모듈이 가능한 보안 부분은 센서 모듈이 생성한 데이터가 수정없이 필요한 모듈에 전송되는 부분이다. 매우 중요한 센서이어서 센서의 오동작이 안전에 영향을 줄 수 있다면 같은 기능을 하는 이중의 센서들을 2개 이상 사용할 수 있다.

모델링 'Modeling'은 센서 모듈의 시뮬레이션을 위해 3D 또는 유사 모델을 제공한다.

4.2 센서 모듈용 정보모델과 다른 정보모델과의 관계

센서 모듈용 정보모델과 HW-SW특성모델, 공통정보 모델간의 관계는 그림 4.1과 같다. HW모듈, SW 모듈 등 모든 유형의 정보모델에 사용되어야 한다. 그 관계는 그림4.1과 같다. HW-SW 특성 모듈은 하드웨어 특성과 소프트웨어 특성을 가지고 있다. 대부분의 HW-SW 특성 모듈은 기구적 프레임이나 커버 또는 기계적 조인트와 같은 하드웨어 모듈과 기계적으로 연결된다. 역시 센서 모듈 역시 HW 모듈과 연결되어 있어야 한다. HW모듈, SW모듈, 센서 모듈의 정보 모델간 연결 관계가 그림4.2에 있다.

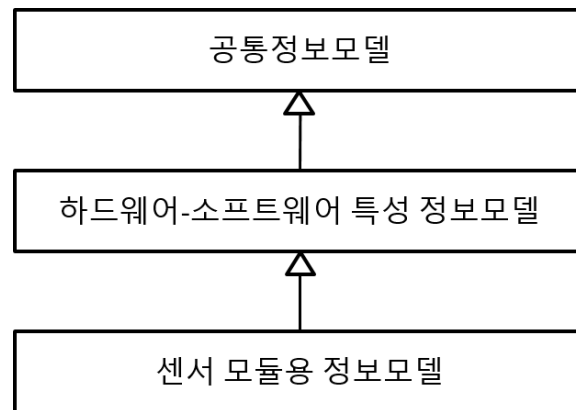


그림 4.1 — 센서 모듈과 HW-SW 특성 모듈의 정보모델 및 공통정보모델간의 관계

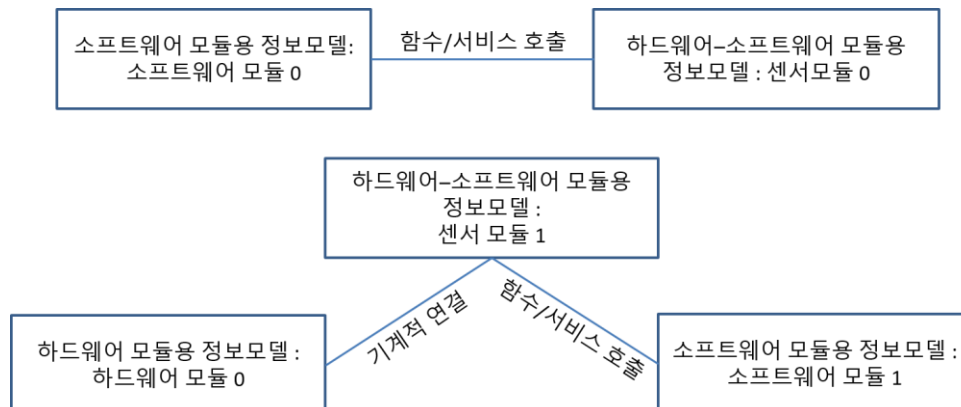


그림 4.2 — 센서 모듈의 정보 모델과 다른 종류의 정보모델간의 관계 예제

4.3 센서모듈용 정보모델을 위한 클래스

4.3.1 개요

센서모듈용 정보모델(IMSensor)은 공통정보모델을 상속받아 그림4.3과 같이 6개의 클래스로 구성되어 있다. 이 클래스는 표4.1의 그룹에 해당 되며, 일반정보그룹은 클래스 IMSensor에 포함되어 있다. 이 정보모델에는 로봇에서 센서모듈을 사용하기 위한 정보들이 포함되어 있어야 한다. 센서모듈이 장착된 위치가 다른 모듈에서 사용되면 해당 위치에 대한 정보를 로봇의 좌표 관점에서 제시하여야 한다. 센서모듈을 사용하기 위해서는 일반적으로 센서모듈용 디바이스 드라이버의 API를 사용하는데,

이에 대한 인터페이스 정보를 제공해야한다. 센서모듈을 사용하기 위해서 데이터버스(예: 필드버스)를 사용한다면 **Infrastructure** 클래스에 데이터버스 정보와 사용하는 인터페이스 정보를 제공하여야 한다. 공통정보모델에 제공하는 **Status** 클래스에는 해당 센서모듈의 상태/건전성 정보(예: 동작, 오류)를 기록하는데, 이를 위해서는 해당 센서모듈을 관리하는 소프트웨어가 별도로 있어야 한다. 그러나 일반적으로 센서모듈에는 이와 관련된 소프트웨어를 제공하지 않으므로 **Status** 클래스를 사용하지 않는 것으로 한다.

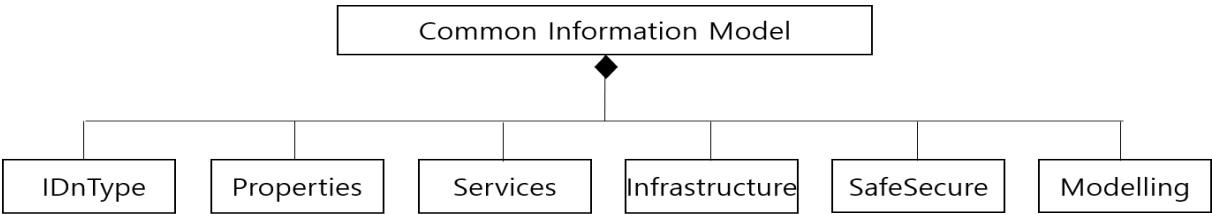


그림 4.3 — 센서모듈용 정보모델 클래스

모듈 이름, 설명, 제조사, 예제와 같은 속성의 값은 그림4.4를 통해서 제공한다. 정보모델 버전은 모듈이 명세될 때, 사용되는 정보모델의 버전 번호이다. 버전 번호는 본 문서인 정보모델 명세사항이 주요하게 수정될 때마다 업데이트된다.

비고 1 현재 버전은 1.0이다.

비고 2 속성은 다음 표식 중 하나를 사용한다: 가시성 기호 (**private (-)**, **protected (*)**, **public (+)**), 속성 이름, 데이터형의 순서로 표시한다. 속성의 이름과 속성의 데이터형의 구분자는 ‘:’ 이다. 만일 속성이 **public** 멤버로 선언 된 경우 해당 속성에 접근하기 위한 함수를 정의할 필요가 없다.

Class IMSensor
+ moduleName : String + description : String + manufacturer : String + examples : String + informationModelVersion : String + idnType : IDnType + properties : Properties + services : Services + infra : Infrastructure + safeSecure : SafeSecure + modelling : Modelling

그림 4.4 — IMSensor 클래스 다이어그램

4.3.2 모듈 아이디와 모듈 유형을 위한 클래스

모듈 아이디와 모듈 유형을 위한 모듈 정보는 **IDnType**의 클래스에서 명시해야 한다. **IDnType** 클래스는 그림4.5에 명시된 속성을 가져야 한다. 그림4.6의 속성 값은 부속서 A에 명시된 방법으로 제공한다. 로봇 내에 동일한 유형의 센서모듈이 사용된다면 개별적으로 사용된 센서모듈들을 식별하기 위하여 인스턴스 식별자(instance ID)가 필요하며, 모듈관리자 혹은 개발자가 인스턴스 식별자를 고유하게 생성하여 할당해야 한다. 사용된 속성 및 메소드의 상세한 설명은 공통정보모델의 설명을 참조한다.

다.

Class IDnType	
+ moduleID : String	// 인스턴스 아이디를 제외한 아이디
- iID : uint8	// 인스턴스 아이디, 기본값
+ getInstanceID() : uint8	
+ setInstanceID(ID: uint8) : Boolean	

그림 4.5 — IDnType 클래스 다이어그램

비고 1 모듈 아이디와 인스턴스 아이디(iID)는 모듈들에 접근하기 위해서 사용된다. iID의 기본값은 0이다.

그림 4.4의 IMSensor의 속성 'idnType'은 그림4.5의 IDnType 클래스로 정의된다.

4.3.3 모듈의 속성을 위한 클래스

센서모듈의 속성(Property) 정보는 Properties 클래스에 명시되어야 한다. Property 클래스는 공통정보모델의 Properties 클래스를 상속받아 사용하며, 센서모듈의 종류에 따라 기본적인 속성의 내용이 다를 수 있다. 센서모듈의 종류별 필수 속성과 선택적 속성이 표4.2에 제시되어있다.

표 4.2 —센서모듈의 종류별 필수 속성과 선택적 속성

센서 모듈의 종류	필수 속성	선택적 속성	센서 예
관성센서			
Proximity 센서			
시각센서			
촉각센서			
후각센서			
청각센서			
힘/토크센서			
기타 물리센서			

4.3.4 모듈의 서비스를 위한 클래스

센서모듈의 서비스는 서비스 클래스 안에 정의해야 한다. 센서모듈의 Services 클래스는 공통정보모델의 Services 클래스를 상속받아 사용하며, Services 클래스에는 센서모듈이 제공하는 기능을 얻기 위한 인터페이스를 정의하여야 한다. 이에 대한 전형적인 예는 다음과 같다.

- initialization(): 초기화
- setParameter(): 파라미터 값 설정/변경
- askSensorData(): 센서결과 요청
- respondSensorData(): 센서결과 받았다는 결과 전송

- close(): 종료

4.3.5 모듈의 인프라를 위한 클래스

센서모듈의 인프라를 위한 정보는 **Infrastructure** 클래스에 정의해야 한다. 센서모듈이 사용하는 인프라 정보는 데이터버스와 방진방수 종류이다. 센서모듈이 사용하는 데이터버스의 종류를 식별할 수 있는 정보와 방진방수 정보를 제공하면 된다. 관련 자세한 정보는 공통정보모델을 참조한다.

Class Infrastructure	
// 데이터버스	
typedef DataBus[] DataBusList;	
+ noBuses : int8	// 모듈안에서 사용되는 데이터버스갯수. 기본값 1.
+ dataBus: DataBusList	
// 방진방수 코드	
+ IPcode : Enumeration	// Ingress Protection IEC 60529; 두자리

그림 4.6 — Infra 클래스 다이어그램

4.3.6 모듈의 안전과 보안 정보를 위한 클래스

SafeSecure 클래스는 그림4.10과 같이 모듈이 제공하는 안전 레벨과 보안 레벨을 명시해야 한다. 자세한 설명은 공통정보모델을 참조한다.
센서모듈의 안전은 신뢰성있는 동작을 수행하면 되고, 별도의 안전관련 제어 동작을 할 필요는 없다. 센서모듈의 보안은 센서모듈의 기능에 따라 다를 수 있다. 센서모듈의 **Services** 클래스에서 제공하는 인터페이스를 사용하여 센서모듈의 구성을 변경하거나 센서모듈의 **Properties** 클래스에서 제공하는 좌표를 변경할 수 있다면 보안을 강화해야 한다. 따라서 센서모듈의 보안은 표4.4의 내용을 지원여부를 확인하여야 한다.

표 4.3 — 안전관련 함수와 태그유형

안전 기능(ISO 13482)	태그유형
응급정지 (Emergency stop)	ESTOP
보호정지 (Protective stop)	PSTOP
작업장 제한 (Limits to workspace (incl. forbidden area avoidance))	LIMWS
안전관련 속도제어 (Safety-related speed control)	SRSC
안전관련 힘제어 (Safety-related force control)	SRFC
위험 충돌회피 (Hazardous collision avoidance)	HCOLA
안정제어 (Stability control (incl. overload protection))	STCON

비고 6 표4.3에 명시되지 않은 안전관련 함수는 필요한 경우 추가한다.

표 4.6 — 사이버보안을 위한 측정항목과 태그유형

보안측정유형 (IEC 62443-4-2)	태그유형
SW프로세스와 장치의 식별과 인증	SD_IA

(SW process and device identification and authentication)	
식별자관리 (Identifier management)	ID_MGT
세션 잠금 (Session lock)	SESS_LOCK
동시세션 제어 (Concurrent session control)	SECC_CNTR
시간기록 (Timestamps)	TIMESTM
거절방지 (Non-repudiation)	NON_REP
통신무결성 (Communication integrity)	COMM_INTG
악성코드보호 (Protection from malicious code)	PROT_MALI_CODE
보안기능검증 (Security functionality verification)	SECUR_VERIFY
소프트웨어 및 정보 무결성 (Software and information integrity)	SW_INTGT
입력 유효성 (Input validation)	INPUT_VALD
오류처리 (Error handling)	ERR_HNDL
정보 기밀성 (Information confidentiality)	INFO_CONFI
정보 지속성 (Information persistence)	INFO_PERS
암호화 사용 (Use of cryptography)	CRYPTO

CIM의 'safeSecure'는 그림 4.10의 SafeSecure 클래스로 정의된다.

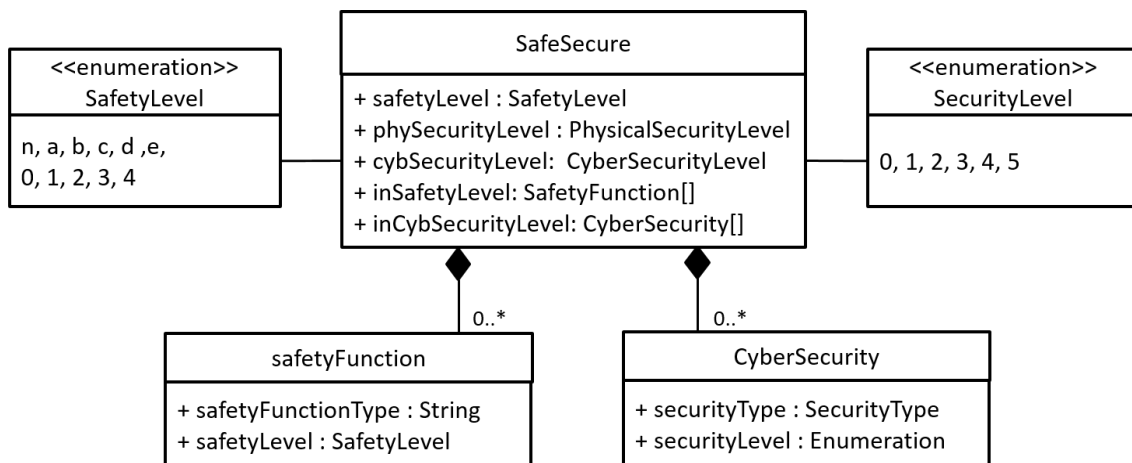


그림 4.7 — SafeSecure 클래스를 구성하는 클래스간 다이어그램

Class SafetyFunction
+ safetyFunctionType : String // 표4.5 태그이름
+ safetyLevel : SafeytLevel //그림4.9참조

그림 4.8 — SafetyFunction 클래스 다이어그램

Class CyberSecurity
+ securityType : String // 표4.6의 태그이름
+ securityLevel : SecurityLevel // 그림4.9 참조

그림 4.9 — CyberSecurity 클래스 다이어그램

Class Modelling
Typedef modelCase[] modelCaseList; + simulationModel ModelCaseList;

그림 4.13 — Modelling 클래스 다이어그램

부속서 A
(규정)

센서 모듈 아이디 할당 규칙

센서모듈용 모듈아이디는 그림A.1과 같이 6개의 요소와 표A.1과 같은 유형으로 구성해야 한다. 첫 요소는 Vendor ID or VID 이며, IETF RFC4122의 Universally Unique Identifier(UUID)를 할당되어야 한다. 나머지 5개의 요소는 판매사에 의해서 할당되어야 한다. 두번째 요소는 Product ID or PID 이며, 1바이트의 모듈의 유형 정보와 3바이트의 제품ID 이다. 전자의 정보는 그림A.1의 가장 아래에 표시된 내용을 기록해야 한다. 즉 이 내용은 센서 모듈이 복합(composite)인지 기본(basic)인지 여부, 복합모듈인 경우 관련 모듈이 무엇인지(HW 및/혹은 SW 복합인지)를 확인하고, 모듈이 안전 관련 기능 혹은 보안 관련 기능을 제공하는지를 표시한다. 후자의 정보는 문자로 표시하는 실제 제품ID 이어야 한다. 세번째 요소는 4바이트로 검토 번호 또는 Rev 이어야 한다. 네번째 요소는 4바이트 시리얼번호 또는 이어야 한다. 다섯번째 요소는 3바이트의 클래스 식별자로, 센서 모듈의 종류를 정한다. 여섯번째 요소는 0~255의 번호문자로 표시된 1바이트의 인스턴스 ID 또는 IID 이어야 한다. 인스턴스 ID의 기본 값은 0 이다. 만일 하나의 모듈에 두개 이상 동일한 유형의 모듈을 포함하고 있다면, 모듈은 다른 인스턴스 아이디들을 가지고 있어야 한다.

센서 모듈의 경우에는 PID의 첫번째 바이트의 각 필드는 다음과 같이 고정되어 있어야 한다.

Basic : 0 : SW Aspect :0, HW Aspect: 0

모듈아이디의 모든 데이터는 헥사코드나 문자-숫자 코드들이다.

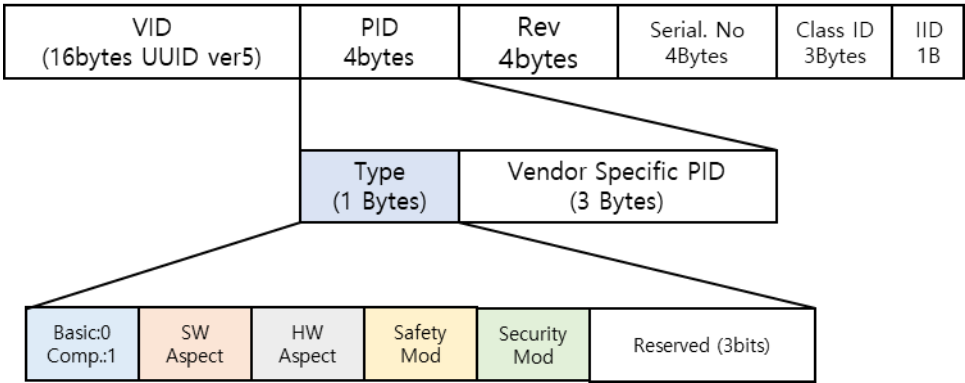


그림 A.1 모듈아이디의 관계

표 A.1 센서용 모듈아이디 요소와 데이터 유형

이름	데이터유형	길이(Byte)		표시
VID	UUID	16		Vendor ID
PID	unsigned char	4		Product ID and properties
Rev	unsigned char	4		Revision number
Serial No	unsigned long	4		Serial number
Class ID	unsigned long	3		센서모듈의 Class ID
IID	unsinged char	1		Instance ID

센서 모듈의 종류와 Class ID는 표 A.2에 있다. 센서 모듈의 종류를 정리하면, 거리 센서, 접촉 센서, 2D 이미지 센서, 3D 이미지 센서, 속도 센서, audio/sound 센서, position 센서, pose 센서, angle 센서, 속도(각/선형) 센서, 가속도 센서, 힘/토크 센서, 온도 센서, 모션 센서, 압력 센서 등이 있다.

표 A.2 센서 모듈의 종류와 Class ID

센서 모듈의 종류	Class ID	비고
관성센서		
Proximity 센서		
시각센서		
촉각센서		
후각센서		
청각센서		
힘/토크센서		
기타 물리센서		

부속서 B
(규정)

센서모듈의 정보모델 표기법

B.1 개요

부속서는 4장의 클래스의 인스턴스를 생성하는 정보를 표현한다. 정보들은 XML or JSON 으로 표시할 수 있다.본 문서에서는 정보는 XML 표시한다.

B.2 모듈의 일반정보

모듈의 일반정보를 위한 XML 요소는 표B.1과 같이 사용되어야한다..

표 B.1 GenInfo의 XML 요소

요소 이름	설명
GenInfo	모든 속성은 해당 요소안에 정의된다. 모듈의 일반적 정보를 설명한다.
ModuleName	모듈의 이름을 정의한다.
Description	모듈을 일반적으로 설명한다.
Manufactures	모듈의 제조자의 정보로써 연락처와 이름 등이다.
Examples	모듈의 전형적인 use case 예이다.
InformationModelVersion	적용한 정보모델 버전번호

보기 GenInfo 클래스에 대한 정보의 예는 다음과 같다

```
<?xml version="1.0" encoding="UTF-8" ?>    <!-- version = "1.1" OK    -->
<GenInfo>
  <ModuleName> Module Name </ModuleName>          <!-- string -->
  <Description> Description of module    </Description>    <!-- string -->
  <Manufactures> manufacturer </Manufactures>          <!-- string -->
  <Examples> list of use case </Examples>          <!-- string -->
  <InformaitonModelVersion> 1.0 </InformaitonModelVersion>    <!-- string -->
</GenInfo>
```

B.3 Module ID에 대한 정보

모듈의 아이디 정보를 위한 XML 요소는 표B.2와 같이 사용되어야한다. 'HWLIST'요소 만 존재하는 모듈은 일종의 HW 모듈이다. 'SWLIST'요소 만 있는 모듈은 일종의 SW 모듈이다. 'HWLIST'와 'SWLIST' 두 요소가 동시에 존재할 경우 복합 모듈은 일종의 HWSW 모듈이다.

표 B.2 IDnType 의 XML 요소

요소이름	설명
ID	모듈에 대한 제조업체의 고유 한 제품 참조 번호로 사용된다. 부속서 B를 참조한다.이 요소는 기본 모듈 또는 복합 모듈과 같은 모듈 유형을 제공하기위한 속성 '유형'을 가질 수 있다. 그 값이 "Bas"이거나 속성이 정의되지 않은 경우 모듈은 일종의 기본 모듈이다. 값이 "Com"이면 모듈은 일종의 복합 모듈이다.

보기 1 클래스 IDnType에 대한 정보의 기본 모듈의 예이다.

```
<?xml version="1.0" encoding="UTF-8" ?>
<IDnType>    <!-- example of basic module -->
    <ID type="Bas"> moduleId </ID> <!-- string provided by manufacturer and defined in Annex B-->
</IDnType>
```

B.4 속성, 입력, 출력 정보

대부분의 모듈 속성, 입력 및 출력은 모듈 실행과 관련이 있으며 SW특성 매개 변수로 사용된다. HW 특성의 경우 사용 사례가 제공이 필요하고 해당 예의 입력 및 출력에 대한 정보 제공된다.

4.3.3 항에 정의 된 Property 클래스에 대한 정보로써 속성의 이름과 값, 값의 데이터 유형은 필수로 제공되며, 값의 단위와 관련 설명은 명확성을 위해 포함될 수 있다.

하위 절 4.3.4에 정의 된 변수 클래스에 대한 정보에는 속성의 이름과 값, 값의 데이터 유형 및 입력 / 출력 유형이 포함된다. 또한 사용의 명확성을 위해 값의 단위와 관련 설명을 포함 할 수 있다. 특히, 동시에 입출력이 가능한 변수는 'input/output' 기호로 표시한다.

속성, 입력 및 출력 정보를위한 XML 요소는 표 B.3과 같이 사용되어야 한다.

표 B.3 속성, 입력, 출력의 XML 요소

요소이름	설명
Properties	모든 관련 속성은 이 요소 내에 정의한다. 비고 네임스페이스가 필요한 경우 서비스 로봇을 위한 모듈 – 2-2부: 소프트웨어 모듈 정보모델 및 서비스 로봇을 위한 모듈 – 2-3부: 하드웨어 모듈 정보모델 에 정의한다.
Property	'Properties'요소에서 개별 속성을 정의하는 데 사용됩니다. 이 요소는 배열 데이터 유형과 구조 및 클래스와 같은 복잡한 데이터 유형을 제공하기위한 속성 'ComplexData'를 가질 수 있다. 이 속성에 "array" 또는 "class" 가있는 경우 속성에는 배열 또는 클래스 구조가 있다. 정의되지 않은 경우 Property는 단순 데이터 유형을 갖는다.
Name	모듈에서 사용되는 속성 또는 변수의 이름 예 : 엔코더, 정격 전류, MaxCurrent, P_Coeff
Value	속성 또는 변수 초기화에 사용한다
Type	속성 또는 변수의 데이터 유형 예 : int16, float64, int32, int8, uint16, 클래스, 배열가 있다. 표 B.4를 참조한다
Unit	속성 또는 변수의 단위이다. 예 : 암페어, 미터, 섭씨, none 등
Description	관련 속성 또는 변수에 대한 설명이다

표 B.4는 이 문서에서 사용될 데이터 유형을 보여준다. 'Type'요소가 'enum'이면 새 속성 인 'enum'이 사용된다. '단위'요소는 물리량의 크기를 의미하며 예를 들면 미터, rpm, 초, bps (bit per sec), 암페어, 볼트, 와트 및 섭씨이다. 특히 'Unit' 요소에는 'none'이라는 특수 단위가있어 물리량이 없다는 의미이다.

표 B.4 변수 'Type' 의 값

유형	설명
boolean	True(1) or False(0)
int8	Signed integer of 1 byte
int16	Signed integer of 2 byte
int32	Signed integer of 4 byte
int64	Signed integer of 8 byte
uint8	unsigned integer of 1 byte
uint16	unsigned integer of 2 byte
uint32	unsigned integer of 4 byte
uint64	unsigned integer of 8 byte
float32	floating point of 4 byte
float64	floating point of 8 byte
enum	Enumeration type
array	Array type, a kind of complex data
class	Complex data type including structure and class.

보기 Property 및 IOVariable 클래스에 대한 정보의 예이다.

```
<?xml version="1.0" encoding="UTF-8" ?>
<Properties>  <-- properties -->
  <Property name="maxRatedCurrent" type="float32" unit="ampere" description =
"maximum of rated current for motor" value =15 />  <-- 속성 하나의 예제 -->
  <Property name="maxRatedVoltage" type="float32" unit="volt" description = "maximum
of rated voltage for motor" >
    <value> 5 </value>
  </Property>  <-- 속성 하나의 예제 -->
<-- following is example of array: float32 [6] initPos  -->
  <Property name="initPose" ComplexData = "array" type="float32" unit="none"
description="initial pose, x,y,z, yaw,roll,pitch, in order. Unit of yaw, roll, and pitch =
degree " >
    <value>
      0, 0, 0, 10, 20, 10
    </value>
  </Property>  <-- array 타입 속성 예제 -->
<!-- following is example of class: -->
<!--   class cylinder {           -->
<!--       float32 legnth;         -->
<!--       float32 radius         -->
<!--   }                           -->
  <Property name="cylinder" ComplexData = "class">
```

```

    <Property name="length", type="float32" value = "100.0" unit="millimetre"
description = "length of cylinder" />
    <Property name="radius", type="float32" value = "3.0" unit="millimetre" description =
"radius of cylinder" />
  </Property>    <!-- class 타입 속성 예제 -->
</Properties>

```

B.5 모듈의 서비스 정보

함수 (기능성)에 대한 정보의 예는 그림C.5 및 그림C.6과 같이 제공된다. 이 정보는 일반적으로 C / C ++, Java 및 Python과 같은 언어로 작성된 프로그램에서 사용된다. 따라서 인수의 순서를 결정하고 인수를 통해 추가 반환 값을받는 것이 중요하다. 추가 반환 값으로 사용되는 인수의 데이터 유형은 C / C ++의 포인터 또는 Java 및 Python의 참조 유형이다.

보기 1 IDL 형식으로 정의한 서비스의 예제는 다음과 같다.

```

typedef sequence<Octet> BufferType;
interface Service: CIMService
{
    // one of methods/services to be provided by module
    uint8 initialize(int64 val1, float32 val2);
    uint8 finalize(Long val1, Float val2, Long val3);
    uint8 read(Long fd, BufferType buf, UShort nbytes);
}

```

보기 2 정보모델에 나타난 IDL 형식의 서비스 정보는 다음과 같다.

```

<?xml version="1.0" encoding="UTF-8" ?>
<Services type = "IDL">
    Path of IDL file.
</Services>

```

B.6 모듈의 인프라 정보

이 정보는 인프라 지원 유형 및 전원, 데이터 버스, IP (ingress protection)와 같은 환경 보호와 관련된다. 전력선은 전력 소비를 포함하는 모듈에 공급되는 전력 유형과 모듈이 공급하는 전력 유형과 관련이 있다. 데이터 버스는 이더넷, EtherCAT, CAN, USB 및 RS422와 같은 모듈에 연결된 통신 유형과 관련된다. IP는 IEC 60529에 따라 모듈이 제공하는 IP 코드와 관련이 있다. 인프라 정보를위한 XML 요소는 표 B.5와 같이 사용되어야한다.

비고 인프라 용 데이터 bus는 시스템에서 모듈이 일반적으로 사용되는 bus를 의미한다.

데이터 버스에 대한 정보는 다음을 제공한다. 프로토콜 유형, 전송 속도 및 프로토콜 유형 CAN2.0, EtherCAT, 이더넷 및 RS485에서 지원되는 API 이 있다.

표 B.5 인프라클래스의 XML 요소

요소이름	설명
Infra	인프라의 모든 속성은이 요소 내에 정의이다. 이 요소는 모듈의 인프라에 대한 정보를 설명한다.

DataBuses	모듈이 사용하는 데이터 버스를 정의하는 데 사용된다.
Databus	모듈이 사용하는 데이터 버스 하나를 정의하는 데 사용된다.
ConnectorType	모듈이 사용하는 데이터 버스의 커넥터 유형을 정의하는 데 사용된다.
TypePhyMac	모듈이 사용하는 물리적 및 MAC 프로토콜 유형을 정의하는 데 사용된다. 예 : CAN, EtherCAT, 이더넷, RS232, RS485. 값은 ISO 22166-203에 자세히 정의된다.
TypeNetTrans	모듈이 사용하는 네트워크 및 전송 프로토콜 유형을 정의하는 데 사용된다. 예 : IPTCP, CANopen, 값은 ISO 22166-202 또는 -203에 자세히 정의된다.
TypeApp	모듈이 사용하는 물리적 및 Mac 계층 유형을 정의하는 데 사용된다. 예 : Modbus, POD, OBD, SYNC, 값은 ISO 22166-203에 자세히 정의된다.
Speed	모듈이 제공하는 전송 속도를 정의하는 데 사용된다.
IP	IEC 60529의 Ingress Protection 코드를 정의하는 데 사용된다

보기 XML로 나타낸 Infrastructure클래스를 위한 정보 예제는 다음과 같다.

```
<?xml version="1.0" encoding="UTF-8" ?>
<Infra>
  <DataBuses>    <!-- list of data buses used in Module -->
    <Databus>
      <ConnectorType> DE9 </ConnectorType>    <!-- D-Sub 9-->
      <TypePhyMac> CAN </Type>
      <TypeNetTrans> CANopen </TypeNetTrans>
      <TypeApp> OBD NMT SDO PDO SYNC </TypeApp>    <!-- CANopen -->
      <Speed value = 1000 unit="kbps" />    <!-- 1 Mbps, unit Kbps -->
    </Databus>
    <Databus>
      <ConnectorType> RJ45 </ConnectorType>    <!-- Jack for EtherNet -->
      <TypePhyMac> EtherCAT </Type>
      <TypeNetTrans> IPTCP </TypeNetTrans>
      <TypeApp> Modbus </TypeApp>
      <Speed value = 10000 unit="kbps" />    <!-- 1 Mbps, unit Kbps -->
    </Databus>
  </DataBuses>
  <IP> IP23CH    </IP>    <!--    Ingress Protection IEC 60529-->
</Infra>
```

B.7 모듈의 안전과 보안의 정보

안전과 보안 정보는 각 안전 기능에 대한 안전 성능 수준과 모듈이 지원하는 각 보안 기능에 대한 보안 관련 정보를 제공한다. Safety 와Security 정보를위한 XML 요소는 표 B.6과 같이 사용되어야 한다.

태그 PL은 [n, a, b, c, d, e] 다음과 같은 열거 값을 가져야한다. 여기서 n은 성능 수준 없음을 나타내고 a ~ e는 PLa ~ PLe (ISO 13849-1)을 나타낸다. CyberSecurity에는 다음과 같은 열거 값이 있다. [0, 1, 2, 3, 4] 여기서 0은 보안 조치 없음을 나타내고 1 ~ 4는 보안 수준 (SL) 1 ~ SL 4를 나타냅니다 (IEC 62443-4-2 참조). 물리적 보안을위한 측정은 ISO 22166-1의 하위 조항 5.6에 설명 된 방법을 사용한다. 안전 기능에 대한 안전 수준은 SIL 또는 PL을 사용하여 제공되어야한다.

표 B.6 SafeSecure 클래스 XML 요소

요소이름	설명
SafeSecure	모든 관련 속성은 이 요소 내에 정의된다. 이 요소는 모듈의 안전 및 보안에 대한 정보를 설명한다.
Safety	모듈이 제공하는 안전 기능의 성능 수준 또는 모듈이 두 개 이상의 안전 기능을 제공하는 경우 모듈의 전반적인 안전 성능 수준을 정의하는 데 사용된다..
Overall	모듈의 전반적인 성능 수준의 안전 또는 사이버 보안을 제공하는 데 사용된다.
SafetyType	지정된 개별 안전 기능의 성능 수준을 제공하는 데 사용한다. 'type'속성을 사용하여 개별 안전 기능이 지정되고 해당 값이 표 4.4에 정의 된 'Tag type' 열에 표시된다.
Security	모듈이 제공하는 보안 기능의 보안 수준 또는 모듈이 둘 이상의 보안 기능을 제공하는 경우 모듈의 전체 보안 수준을 정의하는 데 사용된다.
PhysicalSecurity	모듈의 물리적 보안 수준을 정의하는 데 사용된다. 값은 None, LatchSensor, LockwithKey, LockwithActuator이다.
CyberSecurity	사이버 보안 기능의 보안 수준 정의에 사용한다
SecType	지정된 개별 보안 기능의 보안 수준을 제공하기 위해 사용한다. 'type'속성을 사용하여 개별 보안 기능을 지정하고 표 4.4에 정의 된 'Tag type' 열에 해당 값을 표시한다.

보기 XML로 나타낸 SafeSecure클래스 정보의 예는 다음과 같다.

```
<?xml version="1.0" encoding="UTF-8" ?>
<SafeSecure>
  <Safety>
    <Overall> d </Overall>      <!-- choose one PL among a, b, c, d, e, and none. -
->
    <SafetyType type = "ESTOP "> d </SafetyType> <!-- individual safety measure. see
Table 4.5 -->
    <SafetyType type = "SRSC " value = 'e' />
  </Safety>
  <Security>
    <PhysicalSecurity>
      <!-- None, LatchSensor, LockwithKey, LockwithActuator , see sub-clause 5.6 in ISO
22166-1 -->
      None
    </PhysicalSecurity>
    <CyberSecurity> <!-- list security functions provided by module -->
      <Overall> 2 </Overall> <!-- overall security level of a module -->
      <SecType type = "HU_IA"> 2 </ SecType > <!-- individual security fct. see
Table 4.10 -->
      < SecType type = "ACNT_MGT" value = 2 /> <!-- see Table 4.10 -->
    </CyberSecurity>
  </Security>
```

</SafeSecure>

B.8 모델링 정보

모델링 정보는 모듈이 시뮬레이션을 위해 제공하는 정보를 제공한다. 예제에는 사용할 시뮬레이터 유형, 3D 모델 및 3D 모델 설명 형식 URDF (Universal Robotic Description Format) 파일이 포함된다. 안전 및 보안 정보를 위한 XML 요소는 표 B.7과 같이 사용되어야 한다. 모델링 클래스에 대한 정보의 예는 그림 C.9와 같이 제공된다.

표 B.7 Simulation 속성

요소이름	설명
Modelling	모든 속성은 이 요소 내에 정의된다. 이 요소는 모듈의 시뮬레이션 모델링에 대한 정보를 설명한다. 모듈은 하나 이상의 시뮬레이션 모델을 가질 수 있다.
SimulationModel	모듈이 제공하는 시뮬레이션 모델을 정의하는 데 사용된다. 시뮬레이션 모델은 시뮬레이터 (또는 시뮬레이션 프로그램), 모델 설명 형식 및 3D 모델 형식으로 구성된다.
Simulator	시뮬레이터 (또는 시뮬레이션 프로그램) 정의에 사용된다 예 : Gazebo, Webots, RoboDK, SimSpark, OpenRave
ModelFile	모델 설명 형식 (MDF) 및 3D 모델 형식 (3DF) 정의에 사용된다. 속성 'type'은 다음 값 중 하나를 가질 수 있다. type이 URDF 및 SDF와 같은 MDF 중 하나 인 경우 값은 모듈이 제공하는 MDF 파일의 URL 또는 경로이다. 설명 형식은 모듈 모델, 환경, 시각화 및 제어를 나타냅니다. type이 STL, OBJ, 3DS, DAE 및 FBX와 같은 3DF 중 하나 인 경우 값은 모듈이 제공하는 3DF 파일의 URL 또는 경로이다.

보기 XML로 나타낸 Modelling 클래스를 위한 정보의 예제는 다음과 같다.

```
<?xml version="1.0" encoding="UTF-8" ?>
<Modelling>
  <SimulationModel>  <!--the 1-st simulation model to be used -->
    <Simulator> Gazebo </Simulator>    <!-- Simulation programs. optional -->
    <ModelFile type="URDF">  <!-- model description file type and its file path for model
of module -->
      ../../modelDescrtionFileName <!-- the path of model description file  -->
    </ModelFile>
    <ModelFile type="STL">  <!-- 3D graphic model type and its file path for model of
module -->
      ../../module3DGraphicFileName <!--the path of 3Dgraphic file model file  -->
    </ModelFile>
  </SimulationModel>
</Modelling>
```

지능형로봇표준포럼

표준서의 서식 및 작성방법

Rules for the drafting and presentation of Korea Robot Standards

KOROS XXXX : YYYY

제 정 자 : 지능형로봇표준포럼 의장

제정 : YYYY년 MM월 DD일

지능형로봇표준포럼 사무국
서울시 용산구 한강대로 31 금영빌딩 8 층한국로봇산업협회
전화 : (02) 780-3060