|  |  |
| --- | --- |
| Project | **Standard for Actuator Interface for Cyber and Physical World**<https://sagroups.ieee.org/2888/ **>** |
| Title | **Proposal of Generic Command Type** |
| DCN | **2888-21-0059-00-0002** |
| Date Submitted | **October 12, 2021** |
| Source(s) | Yegi Lee zxcasd312@naver.com (Konkuk University)Shin Kim new.xin22@gmail.com (Konkuk University)Kyoungro Yoon yoonk@konkuk.ac.kr (Konkuk University) |
| Re: |  |
| Abstract | This contribution proposes the generic command type that the actuators can have same command type. |
| Purpose | To start discussion on purpose of the standard |
| Notice | This document has been prepared to assist the IEEE 2888 Working Group. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein. |
| Release | The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE’s name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE’s sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that IEEE 2888 may make this contribution public. |
| Patent Policy | The contributor is familiar with IEEE patent policy, as stated in [Section 6 of the IEEE-SA Standards Board bylaws](http://standards.ieee.org/guides/opman/sect6.html#6.3) <[http://standards.ieee.org/guides/bylaws/sect6-7.html#6](http://127.0.0.1:4664/cache?event_id=757737&schema_id=1&s=5X0vID10lu_E6yrIkWkNd4Wz2H8&q=hancock)> and in *Understanding Patent Issues During IEEE Standards Development* <http://standards.ieee.org/board/pat/faq.pdf> |

# Introduction

At the last meeting, we proposed the individual actuators in the physical world in a standardized data format. And we found that many actuators have the same command types which can generate the actuator effect such as intensity and frequency. So, this contribution proposes the data format for the generic type that the actuators have in general. Accordingly, each syntax and semantic of the individual actuator is updated from the generic type that can be inherited.

2 Common class for defining individual actuator type

* 1. Data Formats for Generic Type
		1. General

This sub-clause specifies the command types for the commands that actuators have in general.

* + 1. Syntax

|  |
| --- |
| "genericCommandType":{ "intensityGenericType": { "type": "integer", "minimum": 0, "maximum": 100, "default": 100 }, "frequencyGenericType": { "type": "integer", "minimum": 0, "maximum": 100, "default": 100 } }, |

* + 1. Semantics

The semantics of the genericCommandType:

|  |  |
| --- | --- |
| *Name* | *Definition* |
| genericCommandType | Provide a data type for describing a generic command type. |
| intensityGenericType | Describes the intensity that some actuator shall emit in percentage with respect to the maximum intensity that the specific actuator can generate. If the intensity is not specified, this command shall be interpreted as turning on at the maximum intensity. |
| frequencyGenericType | Describes the frequency that some actuator shall emit in percentage with respect to the maximum intensity that the specific actuator can generate. If the frequency is not specified, this command shall be interpreted as turning on at the maximum frequency. |

* 1. Data Formats for Individual Actuators
		1. Light actuator
			1. General

This sub-clause specifies the actuator command type which can generate a light effect.

* + - 1. Syntax

|  |
| --- |
| "lightCommandData": { "type": "object", "properties": { "color": { "$ref": "#/definitions/colorType" }, "intensity": { "$ref":"#/genericCommandType/intensityGenericType" }, "additionalProperties": false}, |

* + - 1. Semantics

The semantics of the lightCommandData:

|  |  |
| --- | --- |
| *Name* | *Definition* |
| lightCommandData | Provide a structure for describing a command for a light actuator. |
| color | Describes the color that the light actuator can provide either as a reference to a term that shall be using the colorType. |
| intensity | Describes the intensity a light actuator can generate shall emit in percentage with respect to the maximum intensity using intensityGenericType. If the intensity is not specified, this command shall be interpreted as turning on at the maximum intensity. |

* + - 1. Examples

This example shows the description of the actuator command of light effect with the following semantics. This light actuator is commanded to perform the intensity of 60% of the maximum intensity with the color “blue”.

|  |
| --- |
| { "commandInfoBaseAttributes": {}, "lightCommandData": { "color": "blue", "intensity": 60 }} |

* + 1. Flash actuator
			1. General

This Subclause specifies the actuator command type which can generate a flash effect.

* + - 1. Syntax

|  |
| --- |
| "flashCommandData": { "type": "object", "properties": { "color": { "$ref": "#/definitions/colorType" }, "intensity": { "$ref":"#/genericCommandType/intensityGenericType" }, "frequency": { "$ref":"#/genericCommandType/frequencyGenericType" } }, "additionalProperties": false }, |

* + - 1. Semantics

Semantics of the flashCommandData:

|  |  |
| --- | --- |
| *Name* | *Definition* |
| flashCommandData | Provide a structure for describing a command for a flash actuator. |
| color | Describes the color that the flash actuator can provide either as a reference to a term that shall be using the colorType. |
| intensity | Describes the intensity a flash actuator can generate shall emit in percentage with respect to the maximum intensity using intensityGenericType. If the intensity is not specified, this command shall be interpreted as turning on at the maximum intensity. |
| frequency | Describes the frequency a flash actuator can generate shall emit in percentage with respect to the maximum frequency using frequencyGenericType. If the frequency is not specified, this command shall be interpreted as turning on at the maximum frequency. |

* + - 1. Example

This example shows the description of the actuator command of flash effect with the following semantics. This flash actuator is commanded to perform the light intensity is 60% of the maximum intensity and frequency is 50% of the maximum frequency.

|  |
| --- |
| { "commandInfoBaseAttributes": {}, "flashCommandData": { "intensity": 60, "frequency": 50 }} |

* + 1. Scent actuator
			1. General

This sub-clause specifies an actuator command type which can generate a scent effect.

* + - 1. Syntax

|  |
| --- |
| "scentCommandData": { "type": "object", "properties": { "scent": { "$ref": "#/definitions/scentType" }, "intensity": { "$ref":"#/genericCommandType/intensityGenericType" } }, "additionalProperties": false, "required": [ "scent" ] }, |

* + - 1. Semantics

Semantics of the scentCommandData:

| *Name* | *Definition* |
| --- | --- |
| scentCommandData | Provide a structure for describing a command for a scent actuator. |
| scent | Describes the scent that specific scent actuator can generate. The scent unit of the command value describes as a reference to a term that shall be using the scentType. |
| intensity | Describes the intensity a scent actuator can generate shall emit in percentage with respect to the maximum intensity using intensityGenericType. If the intensity is not specified, this command shall be interpreted as turning on at the maximum intensity. |

* + - 1. Examples

This example shows the description of an actuator command of scent effect with the following semantics. This scent actuator is commanded to perform the intensity 5% of the maximum intensity with the scent “rose”.

|  |
| --- |
| {"commandInfoBaseAttributes": {},"scentCommandData": { "scent": "rose", "intensity": 5 }} |

* + 1. Heating actuator
			1. General

This Subclause specifies the actuator command type which can generate a heating effect.

* + - 1. Syntax

|  |
| --- |
| "heatingCommandData": { "type": "object", "properties": { "intensity": { "$ref":"#/genericCommandType/intensityGenericType" } }, "additionalProperties": false }, |

* + - 1. Semantics

Semantics of the heatingCommandData:

| *Name* | *Definition* |
| --- | --- |
| heatingCommandData | Provide a structure for describing a command for a heating actuator. |
| intensity | Describes the intensity of the temperature effect of heating in percentage with respect to the maximum intensity using intensityGenericType. If the intensity is not specified, this command shall be interpreted as turning on at the maximum intensity.Example: If the heating actuator can control temperature between 20 and 40 degrees Celsius, intensity of 20 percent means the intensity of 24 degrees Celsius. |

* + - 1. Examples

This example shows the description of an actuator command of heating effect with the following semantics. This heating actuator is commanded to perform the intensity of 70% of the maximum intensity.

|  |
| --- |
| {"commandInfoBaseAttributes": {},"heatingCommandData": { "intensity": 70 }} |

* + 1. Cooling actuator
			1. General

This Subclause specifies the actuator command type which can generate a cooling effect.

* + - 1. Syntax

|  |
| --- |
| "coolingCommandData": { "type": "object", "properties": { "intensity": { "$ref":"#/genericCommandType/intensityGenericType" } }, "additionalProperties": false } |

* + - 1. Semantics

Semantics of the coolingCommandData:

| *Name* | *Definition* |
| --- | --- |
| coolingCommandData | Provide a structure for describing a command for a cooling actuator. |
| intensity | Describes the intensity of the temperature effect of heating in percentage with respect to the maximum intensity using intensityGenericType. If the intensity is not specified, this command shall be interpreted as turning on at the maximum intensity.Example: If the device can create cooling effect from 30 to 10 degrees Celsius, intensity of 20 percent means the intensity of 26 degrees Celsius. |

* + - 1. Example

This example shows the description of an actuator command of heating effect with the following semantics. This cooling actuator is commanded to perform the intensity of 30% of the maximum intensity.

|  |
| --- |
| {"commandInfoBaseAttributes": {},"coolingCommandData": { "intensity": 30, }} |

* + 1. Vibration actuator
			1. General

This Subclause specifies the actuator command type which can generate a vibration effect.

* + - 1. Syntax

|  |
| --- |
| "vibrationCommandData": { "type": "object", "properties": { "intensity": { "$ref":"#/genericCommandType/intensityGenericType" }, "frequency": {"$ref":"#/genericCommandType/frequencyGenericType" } }, "additionalProperties": false, "anyOf": [ { "required": [ "intensity" ] }, { "required": [ "frequency" ] } ] } |

* + - 1. Semantics

Semantics of the vibrationCommandData:

| *Name* | *Definition* |
| --- | --- |
| vibrationCommandData | Provide a structure for describing a command for a vibration actuator. |
| intensity | Describes the intensity of the vibration effect in terms of strength in percentage with respect to the maximum intensity using intensityGenericType type. If the intensity is not specified, this command shall be interpreted as turning on at the maximum intensity. |
| frequency | Describes the frequency of the vibration effect in terms of number of vibrations in percentage with respect to the maximum intensity using frequencyGenericType type. If the frequency is not specified, this command shall be interpreted as turning on at the maximum frequency. |

* + - 1. Example

This example shows the description of an actuator command of vibration effect with the following semantics. This vibration actuator is commanded to perform the frequency is 50% of the maximum frequency.

|  |
| --- |
| { "commandInfoBaseAttributes": {}, "vibrationCommandData": { "frequency": 50 }} |

* + 1. Sprayer actuator
			1. General

This sub-clause specifies a actuator command type which can generate a spraying effect.

* + - 1. Syntax

|  |
| --- |
| "sprayerCommandData": { "type": "object", "properties": { "sprayingType": { "$ref": "#/definitions/sprayingType" }, "intensity": { "$ref":"#/genericCommandType/intensityGenericType" } }, "additionalProperties": false, "required": [ "sprayingType" ] }, |

* + - 1. Semantics

Semantics of the sprayerCommandData:

| *Name* | *Definition* |
| --- | --- |
| sprayerCommandData | Provide a structure for describing a command for a sprayer actuator. |
| sprayingType | Describes the type of the sprayed material that shall be using the sprayingType. |
| intensity | Describes the intensity that the liquid is sprayed in percentage with respect to the maximum intensity using intensityGenericType. If the intensity is not specified, this command shall be interpreted as turning on at the maximum intensity. |

* + - 1. Examples

This example shows the description of an actuator command of the sprayer effect with the following semantics. This sprayer actuator is commanded to perform the material to be sprayed pure water, and the intensity shall be 45% of the maximum intensity.

|  |
| --- |
| { "commandInfoBaseAttributes": {}, "sprayerCommandData": { "sprayingType": "water", "intensity": 45 }} |

* + 1. Fog actuator
			1. General

This Subclause specifies an actuator command type which can generate a fog effect.

* + - 1. Syntax

|  |
| --- |
| "fogCommandData": { "type": "object", "properties": { "intensity": { "$ref":"#/genericCommandType/intensityGenericType" } },  "additionalProperties": false, }, |

* + 1. Semantics

Semantics of the fogCommandData:

| *Name* | *Definition* |
| --- | --- |
| fogCommandData | Provide a structure for describing a command for a fog actuator. |
| intensity | Describes the intensity of the fog effect in percentage with respect to the maximum intensity using intensityGenericType. If the intensity is not specified, this command shall be interpreted as turning on at the maximum intensity. |

* + 1. Example

This example shows the description of an actuator command of fog effect with the following semantics. This fog actuator is commanded to perform the intensity 50% of the maximum intensity.

|  |
| --- |
| { "commandInfoBaseAttributes": {}, "fogCommandData": { "intensity": 50 }} |

* + 1. Wind actuator
			1. General

This Subclause specifies an actuator command type which can generate a wind effect.

* + - 1. Syntax

|  |
| --- |
| "windCommandData": { "type": "object", "properties": { "intensity": { "$ref":"#/genericCommandType/intensityGenericType" } },  "additionalProperties": false, } |

* + - 1. Semantics

Semantics of the windCommandData:

| *Name* | *Definition* |
| --- | --- |
| windCommandData | Provide a structure for describing a command for a wind actuator. |
| intensity | Describes the intensity of the wind effect in percentage with respect to the maximum intensity using intensityGenericType. If the intensity is not specified, this command shall be interpreted as turning on at the maximum intensity. |

* + - 1. Example

This example shows the description of an actuator command of wind effect with the following semantics. This wind actuator is commanded to perform the intensity 60% of the maximum intensity.

|  |
| --- |
| { "commandInfoBaseAttributes": {}, "windCommandData": { "intensity": 60 }} |

* + 1. Bubble actuator
			1. General

This sub-clause specifies the actuator command type which can generate a bubble effect.

* + - 1. Syntax

|  |
| --- |
| "bubbleCommandData": { "type": "object", "properties": {}, "additionalProperties": false } |

* + - 1. Semantics

Semantics of the bubbleCommandData:

| *Name* | *Definition* |
| --- | --- |
| bubbleCommandData | Tool for describing a bubble actuator command. |

* + - 1. Example

This example shows the description of an actuator command of bubble effect with the following semantics. This bubble actuator is commanded to turn on to activate this actuator.

|  |
| --- |
| { "commandInfoBaseAttributes": {}, "bubbleCommandData": { }} |