

OCF Resource to AllJoyn Interface Mapping Specification

VERSION 2.0.5 | September 2019



OPEN CONNECTIVITY
FOUNDATION™

CONTACT admin@openconnectivity.org

Copyright Open Connectivity Foundation, Inc. © 2019.
All Rights Reserved.

Legal Disclaimer

3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

NOTHING CONTAINED IN THIS DOCUMENT SHALL BE DEEMED AS GRANTING YOU ANY KIND OF LICENSE IN ITS CONTENT, EITHER EXPRESSLY OR IMPLIEDLY, OR TO ANY INTELLECTUAL PROPERTY OWNED OR CONTROLLED BY ANY OF THE AUTHORS OR DEVELOPERS OF THIS DOCUMENT. THE INFORMATION CONTAINED HEREIN IS PROVIDED ON AN "AS IS" BASIS, AND TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, THE AUTHORS AND DEVELOPERS OF THIS SPECIFICATION HEREBY DISCLAIM ALL OTHER WARRANTIES AND CONDITIONS, EITHER EXPRESS OR IMPLIED, STATUTORY OR AT COMMON LAW, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. OPEN CONNECTIVITY FOUNDATION, INC. FURTHER DISCLAIMS ANY AND ALL WARRANTIES OF NON-INFRINGEMENT, ACCURACY OR LACK OF VIRUSES.

The OCF logo is a trademark of Open Connectivity Foundation, Inc. in the United States or other countries. *Other names and brands may be claimed as the property of others.

Copyright © 2017-2019 Open Connectivity Foundation, Inc. All rights reserved.

Copying or other form of reproduction and/or distribution of these works are strictly prohibited.

21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63

CONTENTS

1	Scope	1
2	Normative references	1
3	Terms and definitions	1
4	Document conventions and organization	1
4.1	Conventions	1
4.2	Notation	2
5	Theory of operation	2
5.1	Interworking approach	2
5.2	Mapping syntax	3
5.2.1	Introduction	3
5.2.2	General	3
5.2.3	Value assignment	3
5.2.4	Property naming	3
5.2.5	Arrays	3
5.2.6	Default mapping	3
5.2.7	Conditional mapping	3
5.2.8	Loops	3
5.2.9	Method invocation	4
6	Device type mapping	4
6.1	AllJoyn device types to OCF device types	4
6.2	OCF device types with no AllJoyn equivalent	5
7	Resource to interface equivalence	5
7.1	Introduction	5
7.2	Environment.CurrentAirQuality mapping	7
7.3	Environment.CurrentAirQualityLevel mapping	7
7.4	Operation.ClimateControlMode mapping	7
7.5	Operation.FanSpeedLevel mapping	7
7.6	Operation.HeatingZone mapping	7
7.7	Operation.OnOffStatus, Operation.OnControl, and Operation.OffControl mapping	7
7.8	Operation.OvenCyclePhase	8
8	Detailed mapping APIs	8
8.1	Introduction	8
8.2	Current Air Quality	8
8.2.1	Derived model	8
8.2.2	Property definition	8
8.2.3	Derived model definition	9
8.3	Current Air Quality Level	11
8.3.1	Derived model	11
8.3.2	Property definition	11
8.3.3	Derived model definition	12
8.4	Current Humidity	13

64	8.4.1	Derived model	13
65	8.4.2	Property definition	13
66	8.4.3	Derived model definition	13
67	8.5	Current Temperature.....	14
68	8.5.1	Derived model	14
69	8.5.2	Property definition	14
70	8.5.3	Derived model definition	14
71	8.6	Target Humidity	16
72	8.6.1	Derived model	16
73	8.6.2	Property definition	16
74	8.6.3	Derived model definition	16
75	8.7	Target Temperature	18
76	8.7.1	Derived model	18
77	8.7.2	Property definition	18
78	8.7.3	Derived model definition	18
79	8.8	Audio Volume	20
80	8.8.1	Derived model	20
81	8.8.2	Property definition	20
82	8.8.3	Derived model definition	20
83	8.9	Climate Control Mode	21
84	8.9.1	Derived model	21
85	8.9.2	Property definition	21
86	8.9.3	Derived model definition	22
87	8.10	Closed Status	23
88	8.10.1	Derived model	23
89	8.10.2	Property definition	23
90	8.10.3	Derived model definition	23
91	8.11	Cycle Control.....	24
92	8.11.1	Derived model	24
93	8.11.2	Property definition	24
94	8.11.3	Derived model definition	25
95	8.12	Fan Speed Level.....	26
96	8.12.1	Derived model	26
97	8.12.2	Property definition	26
98	8.12.3	Derived model definition	26
99	8.13	Heating Zone	27
100	8.13.1	Derived model	27
101	8.13.2	Property definition	27
102	8.13.3	Derived model definition	28
103	8.14	HVAC Fan Mode	29
104	8.14.1	Derived model	29
105	8.14.2	Property definition	29
106	8.14.3	Derived model definition	30
107	8.15	On/Off Control	31

108	8.15.1	Derived model	31
109	8.15.2	Property definition	31
110	8.15.3	Derived model definition	31
111	8.16	On Off Mapping	32
112	8.16.1	Derived model	32
113	8.16.2	Property definition	32
114	8.16.3	Derived model definition	32
115	8.17	Oven Cycle Phase	33
116	8.17.1	Derived model	33
117	8.17.2	Property definition	33
118	8.17.3	Derived model definition	34
119			

120

121

No table of figures entries found.

Figures

Tables

122	
123	Table 1 – AllJoyn to OCF device type mapping4
124	Table 2 – OCF device types with no AllJoyn equivalent.....5
125	Table 3 – AllJoyn interface to OCF resource type mapping – minimum interface set6
126	Table 4 – AllJoyn interface to OCF resource type mapping – optional interface set.....6
127	Table 5 – Interface to resource summary8
128	Table 6 – The property mapping for "asa.environment.currentairquality"9
129	Table 7 – The properties of "asa.environment.currentairquality"9
130	Table 8 – The property mapping for "asa.environment.currentairqualitylevel" 11
131	Table 9 – The properties of "asa.environment.currentairqualitylevel". 11
132	Table 10 – The property mapping for "asa.environment.currenthumidity" 13
133	Table 11 – The properties of "asa.environment.currenthumidity" 13
134	Table 12 – The property mapping for "asa.environment.currenttemperature" 14
135	Table 13 – The properties of "asa.environment.currenttemperature" 14
136	Table 14 – The property mapping for "asa.environment.targethumidity" 16
137	Table 15 – The properties of "asa.environment.targethumidity" 16
138	Table 16 – The property mapping for "asa.environment.targettemperature" 18
139	Table 17 – The properties of "asa.environment.targettemperature" 18
140	Table 18 – The property mapping for "asa.operation.audiovolume"20
141	Table 19 – The properties of "asa.operation.audiovolume"20
142	Table 20 – The property mapping for "asa.operation.climatecontrolmode"21
143	Table 21 – The properties of "asa.operation.climatecontrolmode"22
144	Table 22 – The property mapping for "asa.operation.closedstatus"23
145	Table 23 – The properties of "asa.operation.closedstatus"23
146	Table 24 – The property mapping for "asa.operation.cyclecontrol"24
147	Table 25 – The properties of "asa.operation.cyclecontrol"24
148	Table 26 – The property mapping for "asa.operation.fanspeedlevel"26
149	Table 27 – The properties of "asa.operation.fanspeedlevel"26
150	Table 28 – The property mapping for "asa.operation.heatingzone"27
151	Table 29 – The properties of "asa.operation.heatingzone"28
152	Table 30 – The property mapping for "asa.operation.hvacfanmode"29
153	Table 31 – The properties of "asa.operation.hvacfanmode"30
154	Table 32 – The property mapping for "asa.operation.offcontrol"31
155	Table 33 – The properties of "asa.operation.offcontrol"31
156	Table 34 – The property mapping for "asa.operation.oncontrol"31
157	Table 35 – The properties of "asa.operation.oncontrol"31
158	Table 36 – The property mapping for "asa.operation.onoffstatus"32
159	Table 37 – The properties of "asa.operation.onoffstatus"32
160	Table 38 – The property mapping for "asa.operation.ovencyclephase"33

161 Table 39 – The properties of "asa.operation.ovencyclephase".33
162

163 **1 Scope**

164 This document provides detailed mapping information to provide equivalency between AllJoyn
165 defined Interfaces and OCF defined Resources.

166 This document provides mapping for Device Types (AllJoyn to/from OCF), identifies equivalent
167 OCF Resources for both mandatory and optional AllJoyn interfaces and for each interface defines
168 the detailed Property by Property mapping using OCF defined extensions to JSON schema to
169 programmatically define the mappings.

170 **1 Normative references**

171 The following documents are referred to in the text in such a way that some or all of their content
172 constitutes requirements of this document. For dated references, only the edition cited applies. For
173 undated references, the latest edition of the referenced document (including any amendments)
174 applies.

175 ISO/IEC 30118-1:2018 Information technology -- Open Connectivity Foundation (OCF)
176 Specification -- Part 1: Core specification
177 <https://www.iso.org/standard/53238.html>
178 Latest version available at: https://openconnectivity.org/specs/OCF_Core_Specification.pdf

179 ISO/IEC 30118-4:2018 Information technology -- Open Connectivity Foundation
(OCF) 180 Specification -- Part 4: Resource Type specification
181 <https://www.iso.org/standard/74241.html>
182 Latest version available at:
183 https://openconnectivity.org/specs/OCF_Resource_Type_Specification.pdf

184 ISO/IEC 30118-5:2019, Information technology – Open Connectivity Foundation (OCF)
185 Specification – Part 5: Device specification
186 <https://www.iso.org/standard/74242.html>
187 Latest version available at: https://openconnectivity.org/specs/OCF_Device_Specification.pdf

188 Derived Models for Interoperability between IoT Ecosystems, Stevens & Merriam, March 2016
189 [https://www.iab.org/wp-content/IAB-uploads/2016/03/OCF-Derived-Models-for-Interoperability-](https://www.iab.org/wp-content/IAB-uploads/2016/03/OCF-Derived-Models-for-Interoperability-Between-IoT-Ecosystems_v2-examples.pdf)
190 [Between-IoT-Ecosystems_v2-examples.pdf](https://www.iab.org/wp-content/IAB-uploads/2016/03/OCF-Derived-Models-for-Interoperability-Between-IoT-Ecosystems_v2-examples.pdf)

191 AllJoyn Common Data Model Interface Definitions
192 <https://github.com/alljoyn/cdm>

193 **2 Terms and definitions**

194 For the purposes of this document, the terms and definitions given in ISO/IEC 30118-1:2018 and
195 the following apply.

196 ISO and IEC maintain terminological databases for use in standardization at the following
197 addresses:

- 198 – ISO Online browsing platform: available at <https://www.iso.org/obp>
- 199 – IEC Electropedia: available at <http://www.electropedia.org/>

200 **3 Document conventions and organization**

201 **3.1 Conventions**

202 In this document a number of terms, conditions, mechanisms, sequences, parameters, events,
203 states, or similar terms are printed with the first letter of each word in uppercase and the rest

204 lowercase (e.g., Network Architecture). Any lowercase uses of these words have the normal
205 technical English meaning.

206 **3.2 Notation**

207 In this document, features are described as required, recommended, allowed or DEPRECATED as
208 follows:

209 Required (or shall or mandatory).

210 These basic features shall be implemented to comply with the Mapping Specification. The
211 phrases "shall not", and "PROHIBITED" indicate behaviour that is prohibited, i.e. that if
212 performed means the implementation is not in compliance.

213 Recommended (or should).

214 These features add functionality supported by the Mapping Specification and should be
215 implemented. Recommended features take advantage of the capabilities the Mapping
216 Specification, usually without imposing major increase of complexity. Notice that for compliance
217 testing, if a recommended feature is implemented, it shall meet the specified requirements to
218 be in compliance with these guidelines. Some recommended features could become
219 requirements in the future. The phrase "should not" indicates behaviour that is permitted but
220 not recommended.

221 Allowed (or allowed).

222 These features are neither required nor recommended by the Mapping Specification, but if the
223 feature is implemented, it shall meet the specified requirements to be in compliance with these
224 guidelines.

225 Conditionally allowed (CA)

226 The definition or behaviour depends on a condition. If the specified condition is met, then the
227 definition or behaviour is allowed, otherwise it is not allowed.

228 Conditionally required (CR)

229 The definition or behaviour depends on a condition. If the specified condition is met, then the
230 definition or behaviour is required. Otherwise the definition or behaviour is allowed as default
231 unless specifically defined as not allowed.

232 DEPRECATED

233 Although these features are still described in this document, they should not be implemented
234 except for backward compatibility. The occurrence of a deprecated feature during operation of
235 an implementation compliant with the current document has no effect on the implementation's
236 operation and does not produce any error conditions. Backward compatibility may require that
237 a feature is implemented and functions as specified but it shall never be used by
238 implementations compliant with this document.

239 Strings that are to be taken literally are enclosed in "double quotes".

240 Words that are emphasized are printed in *italic*.

241 **4 Theory of operation**

242 **4.1 Interworking approach**

243 The interworking between AllJoyn defined interfaces and OCF defined Resource Types is modelled
244 using the derived model syntax described in Derived Models for Interoperability between IoT
245 Ecosystems. Determination of the minimum set of AllJoyn interfaces for which equivalency is
246 required within the OCF data model was done by listing the set of interfaces required for each of

247 the device types defined by the CDM Project inside of AllJoyn. Where the AllJoyn interface supports
248 methods then an actuation design pattern is applied.

249 **4.2 Mapping syntax**

250 **4.2.1 Introduction**

251 Within the defined syntax for derived modelling used by this document there are two blocks that
252 define the actual Property-Property equivalence or mapping. These blocks are identified by the
253 keywords "x-to-ocf" and "x-from-ocf". Derived Models for Interoperability between IoT Ecosystems
254 does not define a rigid syntax for these blocks; they are free form string arrays that contain pseudo-
255 coded mapping logic.

256 Within this document we apply the rules defined in clause 5.2 to these blocks to ensure consistency
257 and re-usability and extensibility of the mapping logic that is defined.

258 **4.2.2 General**

259 All statements are terminated with a carriage return.

260 **4.2.3 Value assignment**

261 The equals sign (=) is used to assign one value to another. The assignee is on the left of the
262 operator; the value being assigned on the right.

263 **4.2.4 Property naming**

264 All Property names are identical to the name used by the original model; for example, from the
265 OCF Temperature Resource the Property name "temperature" is used whereas when referred to
266 the derived ecosystem then the semantically equivalent Property name is used.

267 When the same name is used by both OCF and the derived ecosystem for semantically equivalent
268 values then the name of the OCF defined Property is prepended by the ecosystem designator "ocf"
269 to avoid ambiguity (e.g. "ocf.step").

270 **4.2.5 Arrays**

271 An array element is indicated by the use of square brackets "[]" with the index of the element
272 contained therein, e.g. range[1]. All arrays start at an index of 0. If an entire array is being
273 referenced then no index is included, e.g. selectablehumiditylevels[].

274 **4.2.6 Default mapping**

275 There are cases where the specified mapping is not possible as one or more of the Properties
276 being mapped is optional in the source model. In all such instances a default mapping is provided.
277 The default map is indicated by the prepending of an "otherwise:" modifier to the assignment. (e.g.
278 "otherwise: step = 1").

279 **4.2.7 Conditional mapping**

280 When a mapping is dependent on the meeting of other conditions then the syntax:

281 if "condition", "mapping".

282 Is applied.

283 E.g. if step >0, ocf.step = step.

284 **4.2.8 Loops**

285 When a mapping can be represented by a repeated loop governed by some condition then the
286 syntax:

287 for "initialize", "condition", "increment": "mapping"

288 Where:

289 "initialize" is an initial local loop control variable setting.

290 "condition" is the loop controller, the loop repeats until the condition evaluates to "false".

291 "increment" allows for update of the control variable, if omitted an increment of "1" is assumed.

292 Is applied.

293 E.g. for $x=0, x < \text{sizeof}(\text{supportedmodes})$: `ocf.supportedmodes[x] = modearray[supportedmodes[x]]`

294 **4.2.9 Method invocation**

295 The invocation of a method or remote procedure call (RPC) from the derived ecosystem as part of
296 the mapping from an OCF Resource is indicated by the use of a double colon "::" delimiter between
297 the applicable resource, service, interface or other construct identifier and the method or RPC
298 name. The method name always includes trailing parentheses which would include any parameters
299 should they be passed.

300 For example, when dealing with the `switchon()` method from AllJoyn this gives a complete method
301 invocation as: `operation.oncontrol::switchon()`.

302 **5 Device type mapping**

303 **5.1 AllJoyn device types to OCF device types**

304 Table 1 captures the equivalency mapping between AllJoyn defined Device Types (see AllJoyn
305 Common Data Model Interface Definitions) and OCF defined Device Types (see Table 10-1 in
306 ISO/IEC 30118-5:2019). The minimum interface set for the AllJoyn definitions is provided in the
307 HAE Theory of Operation; the minimum Resource sets for each OCF Device is provided in
308 ISO/IEC 30118-5:2019.

309 **Table 1 – AllJoyn to OCF device type mapping**

Classification	AllJoyn Device Type	AllJoyn ID	OCF Device Type
Air Care	Air Conditioner	5	oic.d.airconditioner
	Air Purifier	9	oic.d.airpurifier
	Air Quality Monitor	11	oic.d.airqualitymonitor
	Dehumidifier	8	oic.d.dehumidifier
	Humidifier	7	oic.d.humidifier
	Electric Fan	10	oic.d.fan
	Thermostat	6	oic.d.thermostat
Fabric Care	Clothes Washer	12	oic.d.washer
	Clothes Dryer	13	oic.d.dryer
	Clothes Washer-Dryer	14	oic.d.washerdryer
Food Preservation	Refrigerator	2	oic.d.refrigerator
	Ice-Maker	4	oic.r.icemaker (maps to Resource)
	Freezer	3	oic.d.freezer
Food Preparation	Oven	17	oic.d.oven

	Cooktop	18	oic.d.cooktop
	Cookerhood	19	oic.d.cookerhood
	Food probe	20	oic.d.foodprobe
Dish Care	Dishwasher	15	oic.d.dishwasher
Floor Care	Robot Cleaner	16	oic.d.robotcleaner
Entertainment	Television	21	oic.d.tv
	Set Top Box (STB)	22	oic.d.stb

310 **5.2 OCF device types with no AllJoyn equivalent**

311 Table 2 captures the Device Types defined by OCF have no direct equivalent in AllJoyn, they shall
312 all be mapped to an AllJoyn Device Type of "Other" (Id of "1").

313 **Table 2 – OCF device types with no AllJoyn equivalent**

OCF Device Name	OCF Device Type
Receiver	oic.d.receiver
Blind	oic.d.blind
Door	oic.d.door
Garage Door	oic.d.garagedoor
Generic Sensor	oic.d.sensor
Light	oic.d.light
Smart Plug	oic.d.smartplug
Switch	oic.d.switch
Water Valve	oic.d.watervalve
Printer	oic.d.printer
Multi-Function Printer	oic.d.multifunctionprinter
Scanner	oic.r.scanner
Camera	oic.d.camera
Security Panel	oic.d.securitypanel
Smart Lock	oic.d.smartlock

314 **6 Resource to interface equivalence**

315 **6.1 Introduction**

316 Clause 7 captures the equivalency mapping between AllJoyn defined Interfaces (see AllJoyn
317 Common Data Model Interface Definitions) and OCF defined Resource Types (see ISO/IEC 30118-
318 4:2018). Detailed Property by Property mappings are provided in clause 8.

319 Table 3 captures the mappings for Interfaces that are part of the minimum set for an AllJoyn Device.

320 Table 4 captures the mappings for Interfaces that are optional for an AllJoyn Device; deep
321 translation for these interfaces via derived modelling is not within the scope of this release of the
322 document.

Table 3 – AllJoyn interface to OCF resource type mapping – minimum interface set

AllJoyn Interface	OCF Resource Type Name	OCF Resource Type ID	OCF Interface(s)
Environment.CurrentAirQuality	Air Quality Collection	oic.r.airqualitycollection	oic.if.s
Environment.CurrentAirQualityLevel	Air Quality Collection	oic.r.airqualitycollection	oic.if.s
Environment.CurrentHumidity	Humidity	oic.r.humidity	oic.if.s
Environment.CurrentTemperature	Temperature	oic.r.temperature	oic.if.s
Environment.TargetHumidity	Humidity	oic.r.humidity, oic.r.selectablelevels	oic.if.a
Environment.TargetTemperature	Temperature	oic.r.temperature	oic.if.a
Operation.AudioVolume	Audio Controls	oic.r.audio	oic.if.a
Operation.Channel	Not mapped		
Operation.ClimateControlMode	Mode	oic.r.mode	oic.if.a
	Operational State	oic.r.operational.state	oic.if.s
Operation.ClosedStatus	Door	oic.r.door	oic.if.s
Operation.CycleControl	Operational State	oic.r.operational.state	oic.if.s
Operation.FanSpeedLevel	Air Flow	oic.r.airflow	oic.if.a
Operation.HeatingZone	Heating Zone Collection	oic.r.heatingzonecollection	oic.if.s
Operation.HvacFanMode	Mode	oic.r.mode	oic.if.a
Operation.OnOffStatus	Binary Switch	oic.r.switch.binary	oic.if.s
Operation.OvenCyclePhase	Operational State	oic.r.operationalstate	oic.if.s

Table 4 – AllJoyn interface to OCF resource type mapping – optional interface set

AllJoyn Interface	OCF Resource Type Name	OCF Resource Type ID	OCF Interface(s)
Environment.TargetTemperatureLevel	Mode	oic.r.mode	oic.if.a
Environment.WaterLevel	TBD	TBD	oic.if.s
Environment.WindDirection	Air Flow	oic.r.airflow	oic.if.a
Operation.AirRecirculationMode	Mode	oic.r.mode	oic.if.a
Operation.Alerts	TBD	TBD	TBD
Operation.AudioVideoInput	Media Source List	oic.r.media.input	oic.if.a
Operation.BatteryStatus	Battery	oic.r.energy.battery	oic.if.s
Operation.CurrentPower	Energy Usage	oic.r.energy.usage	oic.if.s
Operation.DishWashingCyclePhase	Operational State	oic.r.operationalstate	oic.if.s
Operation.EnergyUsage	Energy Usage	oic.r.energy.usage	oic.if.s
Operation.FilterStatus	TBD	TBD	TBD
Operation.LaundryCyclePhase	Mode	oic.r.mode	oic.if.s
Operation.MoistureOutputLevel	Mode	oic.r.mode	oic.if.a
Operation.PlugInUnits	TBD	TBD	TBD

Operation.RapidMode	Refrigeration	oic.r.refrigeration	oic.if.a
Operation.RemoteControllability	TBD	TBD	TBD
Operation.RepeatMode	Ecomode	oic.r.ecomode	oic.if.a
Operation.ResourceSaving	TBD	TBD	TBD
Operation.RobotCleaningCyclePhase	Mode	oic.r.mode	oic.if.s
Operation.SoilLevel	Mode	oic.r.mode	oic.if.a
Operation.SpinSpeedLevel	Mode	oic.r.mode	oic.if.a
Operation.Timer	Time Period	oic.r.time.period	oic.if.s

326 **6.2 Environment.CurrentAirQuality mapping**

327 If more than one instance of the AirQuality interface is exposed, then each instance maps to an
328 instance of the OCF AirQuality Resource. The mapping defined in clause 8.2 describes the
329 population of the OCF AirQuality Resource. Even if there is only a single instance of an OCF
330 AirQuality Resource this shall be included in an instance of an OCF AirQualityCollection. The
331 number of links in the collection equates to the number of instances of the AllJoyn CurrentAirQuality
332 interface that are exposed. When mapping from OCF the valueType of the Resource shall be
333 introspected, this API is invoked only if this is set to "Measured".

334 **6.3 Environment.CurrentAirQualityLevel mapping**

335 If more than one instance of the AirQualityLevel interface is exposed, then each instance maps to
336 an instance of the OCF AirQuality Resource. The mapping defined in clause 8.2 describes the
337 population of the OCF AirQuality Resource. Even if there is only a single instance of an OCF
338 AirQuality Resource then this shall be included in an instance of an OCF AirQualityCollection. The
339 number of links in the collection equates to the number of instances of the AllJoyn CurrentAirQuality
340 interface that are exposed. When mapping from OCF the valueType of the Resource shall be
341 introspected, this API is invoked only if this is set to "Qualitative".

342 **6.4 Operation.ClimateControlMode mapping**

343 ClimateControlMode has three Properties; these map as follows: mode and supportedmodes maps
344 to the Mode Resource, operationalstate maps to the OperationalState Resource This can be
345 represented in OCF either as two distinct Resource instances or a single instance with two
346 Resource Types (oic.r.mode, oic.r.operationalstate).

347 **6.5 Operation.FanSpeedLevel mapping**

348 The setting of the FanSpeedLevel to "0x00" (off) is handled via the "OffControl" interface rather
349 than writing directly to this interface. In such a case an instance of Binary Switch shall be exposed
350 on the OCF side; this can be modelled as AirFlowControl which is then a collection of Binary Switch
351 and AirFlow.

352 **6.6 Operation.HeatingZone mapping**

353 Each element in the array of heating zones within the AllJoyn HeatingZone interface maps to an
354 instance of OCF HeatingZone, itself a link in an instance of an OCF HeatingZoneCollection. The
355 mapping defined clause 8.13 describes the population of the OCF HeatingZone Resource that
356 constitutes the Resources that are contained in the collection.

357 **6.7 Operation.OnOffStatus, Operation.OnControl, and Operation.OffControl mapping**

358 A discovered instance of a Binary Switch is always mapped to an Operation.OnOffStatus interface.
359 A RETRIEVE on a Binary Switch maps to an action on an instance of an Operation.OnOffStatus
360 Interface. An UPDATE on a Binary Switch maps to a method invocation on either
361 Operation.OnControl or OffControl. value = true maps to Operation.OnControl value = false maps
362 to Operation.OffControl.

363 **6.8 Operation.OvenCyclePhase**

364 OvenCyclePhase cyclephase Property pre-defines values 0x00-0x7F, 0x80-0xFF is for vendor
365 specific values. The mapping defined in clause 8 covers only specification defined values. Any
366 vendor defined value shall be represented in OCF using the x.<organization> syntax for a vendor
367 defined Property.

368 **7 Detailed mapping APIs**

369 **7.1 Introduction**

370 This clause provides a mapping description (using JSON that aligns with the Derived Modelling
371 syntax described in Derived Models for Interoperability between IoT Ecosystems for all Interfaces
372 and Resources that are within scope.

373 The derived model definitions presented in clause 8 are formatted for readability, and so may
374 appear to have extra line breaks.

375 Table 5 provides a reference and link to the per Interface clauses.

376

Table 5 – Interface to resource summary

AllJoyn Interface Name	Equivalent Resource(s)	Clause
Environment.CurrentAirQuality	oic.r.airqualitycollection	8.2
Environment.CurrentAirQualityLevel	oic.r.airqualitycollection	8.3
Environment.CurrentHumidity	oic.r.humidity	8.4
Environment.CurrentTemperature	oic.r.temperature	8.5
Environment.TargetHumidity	oic.r.humidity, oic.r.selectablelevels	8.6
Environment.TargetTemperature	oic.r.temperature	8.7
Operation.AudioVolume	oic.r.audio	8.8
Operation.ClimateControlMode	oic.r.mode, oic.r.operationalstate	8.9
Operation.ClosedStatus	oic.r.door	8.10
Operation.CycleControl	oic.r.operational.state	8.11
Operation.FanSpeedLevel	oic.r.airflow	8.12
Operation.HeatingZone	oic.r.heatingzonecollection	8.13
Operation.HvacFanMode	oic.r.mode	8.14
Operation.OnControl, Operation.OffControl	oic.r.switch.binary	8.15
Operation.OnOffStatus,	oic.r.switch.binary	8.16
Operation.OvenCyclePhase	oic.r.operationalstate	8.17

377 **7.2 Current Air Quality**

378 **7.2.1 Derived model**

379 The derived model: "asa.environment.currentairquality".

380 **7.2.2 Property definition**

381 Table 6 provides the detailed per Property mapping for "asa.environment.currentairquality".

Table 6 – The property mapping for "asa.environment.currentairquality".

AllJoyn Property name	OCF Resource	To OCF	From OCF
minvalue	oic.r.airquality	range[0] = minvalue	minvalue = range[0]
maxvalue	oic.r.airquality	range[1] = maxvalue	maxvalue = range[1]
contaminanttype	oic.r.airquality	valuetype = Measuredcontaminanttypearray [CH2O,CO2,CO,PM2_5,PM10,VOC] ocf.contaminanttype = contaminanttypearray[contaminanttype]	contaminanttype = indexof contaminanttypearray[ocf.contaminanttype]
currentvalue	oic.r.airquality	contaminantvalue = currentvalue	currentvalue = contaminantvalue
updatemintime	oic.r.value.conditional	ocf.minnotifyperiod = updatemintime	updatemintime = ocf.minnotifyperiod
precision	oic.r.airquality	ocf.precision = precision	precision = ocf.precision

383 Table 7 provides the details of the Properties that are part of "asa.environment.currentairquality".

384

Table 7 – The properties of "asa.environment.currentairquality".

AllJoyn name	Property	Type	Required	Description
minvalue		number	yes	
maxvalue		number	yes	
contaminanttype		integer	yes	The contaminant type
currentvalue		number	yes	
updatemintime		integer	yes	
precision		number	yes	

385 7.2.3 Derived model definition

```

386 {
387   "id": "http://openinterconnect.org/asamapping/schemas/asa.environment.currentairquality.json#",
388   "$schema": "http://json-schema.org/draft-04/schema#",
389   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
390   "title": "Current Air Quality",
391   "definitions": {
392     "asa.environment.currentairquality": {
393       "type": "object",
394       "properties": {
395         "contaminanttype": {
396           "type": "integer",
397           "description": "The contaminant type",
398           "x-ocf-conversion": {
399             "x-ocf-alias": "oic.r.airquality",
400             "x-to-ocf": [
401               "valuetype = Measured",
402               "contaminanttypearray = [CH2O,CO2,CO,PM2_5,PM10,VOC]",
403               "ocf.contaminanttype = contaminanttypearray[contaminanttype]"
404             ],
405             "x-from-ocf": [
406               "contaminanttype = indexof contaminanttypearray[ocf.contaminanttype]"
407             ]
408           }
409         },
410         "currentvalue": {
411           "type": "number",

```

```

412         "x-ocf-conversion": {
413             "x-ocf-alias": "oic.r.airquality",
414             "x-to-ocf": [
415                 "contaminantvalue = currentvalue"
416             ],
417             "x-from-ocf": [
418                 "currentvalue = contaminantvalue"
419             ]
420         },
421     },
422     "minvalue": {
423         "type": "number",
424         "x-ocf-conversion": {
425             "x-ocf-alias": "oic.r.airquality",
426             "x-to-ocf": [
427                 "range[0] = minvalue"
428             ],
429             "x-from-ocf": [
430                 "minvalue = range[0]"
431             ]
432         },
433     },
434     "maxvalue": {
435         "type": "number",
436         "x-ocf-conversion": {
437             "x-ocf-alias": "oic.r.airquality",
438             "x-to-ocf": [
439                 "range[1] = maxvalue"
440             ],
441             "x-from-ocf": [
442                 "maxvalue = range[1]"
443             ]
444         },
445     },
446     "precision": {
447         "type": "number",
448         "x-ocf-conversion": {
449             "x-ocf-alias": "oic.r.airquality",
450             "x-to-ocf": [
451                 "ocf.precision = precision"
452             ],
453             "x-from-ocf": [
454                 "precision = ocf.precision"
455             ]
456         },
457     },
458     "updatemintime": {
459         "type": "integer",
460         "x-ocf-conversion": {
461             "x-ocf-alias": "oic.r.value.conditional",
462             "x-to-ocf": [
463                 "ocf.minnotifyperiod = updatemintime"
464             ],
465             "x-from-ocf": [
466                 "updatemintime = ocf.minnotifyperiod"
467             ]
468         },
469     },
470 },
471 },
472 },
473 "type": "object",
474 "allOf": [
475     {"$ref": "#/definitions/asa.environment.currentairquality"}
476 ],
477 "required": ["contaminanttype", "currentvalue", "minvalue", "maxvalue", "precision", "updatemintime"]
478 }
479

```

480 **7.3 Current Air Quality Level**

481 **7.3.1 Derived model**

482 The derived model: "asa.environment.currentairqualitylevel".

483 **7.3.2 Property definition**

484 Table 8 provides the detailed per Property mapping for "asa.environment.currentairqualitylevel".

485 **Table 8 – The property mapping for "asa.environment.currentairqualitylevel".**

AllJoyn name	Property	OCF Resource	To OCF	From OCF
contaminanttype		oic.r.airquality	valuetype = Qualitativeif contaminanttype = 0, ocf.contaminanttype = CH2Oif contaminanttype = 1, ocf.contaminanttype = CO2if contaminanttype = 2, ocf.contaminanttype = COif contaminanttype = 3, ocf.contaminanttype = PM2_5if contaminanttype = 4, ocf.contaminanttype = PM10if contaminanttype = 5, ocf.contaminanttype = VOCif contaminanttype = 253, ocf.contaminanttype = Smokeif contaminanttype = 254, ocf.contaminanttype = Odorif contaminanttype = 255, ocf.contaminanttype = AirPollution	if ocf.contaminanttype = CH2O, contaminanttype = 0if ocf.contaminanttype = CO2, contaminanttype = 1if ocf.contaminanttype = CO, contaminanttype = 2if ocf.contaminanttype = PM2_5, contaminanttype = 3if ocf.contaminanttype = PM10, contaminanttype = 4if ocf.contaminanttype = VOC, contaminanttype = 5if ocf.contaminanttype = Smoke, contaminanttype = 253if ocf.contaminanttype = Odor, contaminanttype = 254if ocf.contaminanttype = AirPollution, contaminanttype = 255
maxlevel		oic.r.airquality	range[0] = 0range[1] = maxvalue	maxvalue = range[1]
currentlevel		oic.r.airquality	contaminantvalue = currentlevel	currentlevel = contaminantvalue

486 Table 9 provides the details of the Properties that are part of
487 "asa.environment.currentairqualitylevel".

488 **Table 9 – The properties of "asa.environment.currentairqualitylevel".**

AllJoyn name	Property	Type	Required	Description
contaminanttype		integer	yes	The contaminant type

maxlevel	integer	yes	
currentlevel	integer	yes	

489 7.3.3 Derived model definition

```

490 {
491   "id":
492   "http://openinterconnect.org/asamapping/schemas/asa.environment.currentairqualitylevel.json#",
493   "$schema": "http://json-schema.org/draft-04/schema#",
494   "description" : "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
495   "title": "Current Air Quality Level",
496   "definitions": {
497     "asa.environment.currentairqualitylevel": {
498       "type": "object",
499       "properties": {
500         "contaminanttype": {
501           "type": "integer",
502           "description": "The contaminant type",
503           "x-ocf-conversion": {
504             "x-ocf-alias": "oic.r.airquality",
505             "x-to-ocf": [
506               "valuetype = Qualitative",
507               "if contaminanttype = 0, ocf.contaminanttype = CH2O",
508               "if contaminanttype = 1, ocf.contaminanttype = CO2",
509               "if contaminanttype = 2, ocf.contaminanttype = CO",
510               "if contaminanttype = 3, ocf.contaminanttype = PM2_5",
511               "if contaminanttype = 4, ocf.contaminanttype = PM10",
512               "if contaminanttype = 5, ocf.contaminanttype = VOC",
513               "if contaminanttype = 253, ocf.contaminanttype = Smoke",
514               "if contaminanttype = 254, ocf.contaminanttype = Odor",
515               "if contaminanttype = 255, ocf.contaminanttype = AirPollution"
516             ],
517             "x-from-ocf": [
518               "if ocf.contaminanttype = CH2O, contaminanttype = 0",
519               "if ocf.contaminanttype = CO2, contaminanttype = 1",
520               "if ocf.contaminanttype = CO, contaminanttype = 2",
521               "if ocf.contaminanttype = PM2_5, contaminanttype = 3",
522               "if ocf.contaminanttype = PM10, contaminanttype = 4",
523               "if ocf.contaminanttype = VOC, contaminanttype = 5",
524               "if ocf.contaminanttype = Smoke, contaminanttype = 253",
525               "if ocf.contaminanttype = Odor, contaminanttype = 254",
526               "if ocf.contaminanttype = AirPollution, contaminanttype = 255"
527             ]
528           }
529         },
530         "currentlevel": {
531           "type": "integer",
532           "x-ocf-conversion": {
533             "x-ocf-alias": "oic.r.airquality",
534             "x-to-ocf": [
535               "contaminantvalue = currentlevel"
536             ],
537             "x-from-ocf": [
538               "currentlevel = contaminantvalue"
539             ]
540           }
541         },
542         "maxlevel": {
543           "type": "integer",
544           "x-ocf-conversion": {
545             "x-ocf-alias": "oic.r.airquality",
546             "x-to-ocf": [
547               "range[0] = 0",
548               "range[1] = maxvalue"
549             ],
550             "x-from-ocf": [
551               "maxvalue = range[1]"
552             ]
553           }
554       }
555     }
556   }

```

```

556     }
557   },
558   "type": "object",
559   "allOf": [
560     { "$ref": "#/definitions/asa.environment.currentairqualitylevel" }
561   ],
562   "required": ["contaminanttype", "currentlevel", "maxlevel"]
563 }
564

```

565 7.4 Current Humidity

566 7.4.1 Derived model

567 The derived model: "asa.environment.currenthumidity".

568 7.4.2 Property definition

569 Table 10 provides the detailed per Property mapping for "asa.environment.currenthumidity".

570 **Table 10 – The property mapping for "asa.environment.currenthumidity".**

AllJoyn name	Property	OCF Resource	To OCF	From OCF
maxvalue		oic.r.humidity	range[0] = Orange[1] = maxvalue	maxvalue = range[1]
currentvalue		oic.r.humidity	humidity = currentValue	currentvalue = humidity

571 Table 11 provides the details of the Properties that are part of "asa.environment.currenthumidity".

572 **Table 11 – The properties of "asa.environment.currenthumidity".**

AllJoyn name	Property	Type	Required	Description
maxvalue		number	yes	Max measured value for humidty
currentvalue		number	yes	Measured value

573 7.4.3 Derived model definition

```

574 {
575   "id": "http://openinterconnect.org/asamapping/schemas/asa.environment.currenthumidity.json#",
576   "$schema": "http://json-schema.org/draft-04/schema#",
577   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
578   "title": "Current Humidity",
579   "definitions": {
580     "asa.environment.currenthumidity": {
581       "type": "object",
582       "properties": {
583         "currentvalue": {
584           "type": "number",
585           "description": "Measured value",
586           "x-ocf-conversion": {
587             "x-ocf-alias": "oic.r.humidity",
588             "x-to-ocf": [
589               "humidity = currentValue"
590             ],
591             "x-from-ocf": [
592               "currentvalue = humidity"
593             ]
594           }
595         },
596         "maxvalue": {
597           "type": "number",
598           "description": "Max measured value for humidty",
599           "x-ocf-conversion": {
600             "x-ocf-alias": "oic.r.humidity",

```

```

601         "x-to-ocf": [
602             "range[0] = 0",
603             "range[1] = maxvalue"
604         ],
605         "x-from-ocf": [
606             "maxvalue = range[1]"
607         ]
608     }
609 }
610 }
611 }
612 },
613 "type": "object",
614 "allOf": [
615     {"$ref": "#/definitions/asa.environment.currenthumidity"}
616 ],
617 "required": [ "currentvalue", "maxvalue" ]
618 }
619

```

620 7.5 Current Temperature

621 7.5.1 Derived model

622 The derived model: "asa.environment.currenttemperature".

623 7.5.2 Property definition

624 Table 12 provides the detailed per Property mapping for "asa.environment.currenttemperature".

625 **Table 12 – The property mapping for "asa.environment.currenttemperature".**

AllJoyn name	Property	OCF Resource	To OCF	From OCF
precision		oic.r.temperature	ocf.precision = precision	precision = ocf.precision
currentvalue		oic.r.temperature	temperature = currentValueunits = C	oneOf
updatemintime		oic.r.value.conditional	ocf.minnotifyperiod = updatemintime	updatemintime = ocf.minnotifyperiod

626 Table 13 provides the details of the Properties that are part of
627 "asa.environment.currenttemperature".

628 **Table 13 – The properties of "asa.environment.currenttemperature".**

AllJoyn name	Property	Type	Required	Description
precision		number	yes	
currentvalue		number	yes	Measured value
updatemintime		integer	yes	

629 7.5.3 Derived model definition

```

630 {
631     "id": "http://openinterconnect.org/asamapping/schemas/asa.environment.currenttemperature.json#",
632     "$schema": "http://json-schema.org/draft-04/schema#",
633     "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
634     "title": "Current Temperature",
635     "definitions": {
636         "asa.environment.currenttemperature": {
637             "type": "object",
638             "properties": {
639                 "currentvalue": {
640                     "type": "number",
641                     "description": "Measured value",
642                     "x-ocf-conversion": {

```

```

643     "x-ocf-alias": "oic.r.temperature",
644     "x-to-ocf": [
645         "temperature = currentValue",
646         "units = C"
647     ],
648     "x-from-ocf": {
649         "oneOf": [
650             {
651                 "properties": {
652                     "units": "string",
653                     "enum": ["C"]
654                 },
655                 "x-from-ocf": [
656                     "currentvalue = temperature"
657                 ]
658             },
659             {
660                 "properties": {
661                     "units": "string",
662                     "enum": ["F"]
663                 },
664                 "x-from-ocf": [
665                     "currentvalue = (temperature-32)*5/9"
666                 ]
667             },
668             {
669                 "properties": {
670                     "units": "string",
671                     "enum": ["K"]
672                 },
673                 "x-from-ocf": [
674                     "currentvalue = temperature-273.15"
675                 ]
676             }
677         ]
678     }
679 },
680 "precision": {
681     "type": "number",
682     "x-ocf-conversion": {
683         "x-ocf-alias": "oic.r.temperature",
684         "x-to-ocf": [
685             "ocf.precision = precision"
686         ],
687         "x-from-ocf": [
688             "precision = ocf.precision"
689         ]
690     }
691 },
692 "updatemintime": {
693     "type": "integer",
694     "x-ocf-conversion": {
695         "x-ocf-alias": "oic.r.value.conditional",
696         "x-to-ocf": [
697             "ocf.minnotifyperiod = updatemintime"
698         ],
699         "x-from-ocf": [
700             "updatemintime = ocf.minnotifyperiod"
701         ]
702     }
703 }
704 }
705 }
706 }
707 },
708 "type": "object",
709 "allOf": [
710     {"$ref": "#/definitions/asa.environment.currenttemperature"}
711 ],
712 "required": [ "currentvalue", "precision", "updatemintime" ]

```

713 }
714

715 **7.6 Target Humidity**

716 **7.6.1 Derived model**

717 The derived model: "asa.environment.targethumidity".

718 **7.6.2 Property definition**

719 Table 14 provides the detailed per Property mapping for "asa.environment.targethumidity".

720 **Table 14 – The property mapping for "asa.environment.targethumidity".**

AllJoyn Property name	OCF Resource	To OCF	From OCF
minvalue	oic.r.humidity	range[0] = minvalue	minvalue = range[0]otherwise: minvalue = 0
targetvalue	oic.r.humidity,oic.r.selectablelevels	if minvalue != maxvalue, ocf.desiredhumidity = targetvalue;ocf.targetlevel = selectablehumiditylevels[0].if minvalue == maxvalue, ocf.targetlevel = targetvalue.	if x-ocf-alias == oic.r.humidity, targetvalue = desiredhumidity.if x-ocf-alias == oic.r.selectablelevels, targetvalue = targetlevel.
maxvalue	oic.r.humidity	range[1] = maxvalue	maxvalue = range[1]otherwise: maxvalue = 100
stepvalue	oic.r.humidity	step = stepvalue	stepvalue = stepotherwise: stepvalue = 1
selectablehumiditylevels	oic.r.selectablelevels	availablelevels[] = selectablehumiditylevels[]	selectablehumiditylevels[] = availablelevels[]

721 Table 15 provides the details of the Properties that are part of "asa.environment.targethumidity".

722 **Table 15 – The properties of "asa.environment.targethumidity".**

AllJoyn Property name	Type	Required	Description
minvalue	number	yes	
targetvalue	number	yes	Measured value
maxvalue	number	yes	
stepvalue	number	yes	
selectablehumiditylevels	array	yes	

723 **7.6.3 Derived model definition**

```
724 {
725   "id": "http://openinterconnect.org/asamapping/schemas/asa.environment.targethumidity.json#",
726   "$schema": "http://json-schema.org/draft-04/schema#",
727   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
728   "title": "Target Humidity",
729   "definitions": {
730     "asa.environment.targethumidity": {
731       "type": "object",
732       "properties": {
733         "targetvalue": {
```



```

734         "type": "number",
735         "description": "Measured value",
736         "x-ocf-conversion": {
737             "x-ocf-alias": "oic.r.humidity,oic.r.selectablelevels",
738             "x-to-ocf": [
739                 "if minvalue != maxvalue, ocf.desiredhumidity = targetvalue;ocf.targetlevel =
740 selectablehumiditylevels[0].",
741                 "if minvalue == maxvalue, ocf.targetlevel = targetvalue."
742             ],
743             "x-from-ocf": [
744                 "if x-ocf-alias == oic.r.humidity, targetvalue = desiredhumidity.",
745                 "if x-ocf-alias == oic.r.selectablelevels, targetvalue = targetlevel."
746             ]
747         }
748     },
749     "minvalue": {
750         "type": "number",
751         "x-ocf-conversion": {
752             "x-ocf-alias": "oic.r.humidity",
753             "x-to-ocf": [
754                 "range[0] = minvalue"
755             ],
756             "x-from-ocf": [
757                 "minvalue = range[0]",
758                 "otherwise: minvalue = 0"
759             ]
760         }
761     },
762     "maxvalue": {
763         "type": "number",
764         "x-ocf-conversion": {
765             "x-ocf-alias": "oic.r.humidity",
766             "x-to-ocf": [
767                 "range[1] = maxvalue"
768             ],
769             "x-from-ocf": [
770                 "maxvalue = range[1]",
771                 "otherwise: maxvalue = 100"
772             ]
773         }
774     },
775     "stepvalue": {
776         "type": "number",
777         "x-ocf-conversion": {
778             "x-ocf-alias": "oic.r.humidity",
779             "x-to-ocf": [
780                 "step = stepvalue"
781             ],
782             "x-from-ocf": [
783                 "stepvalue = step",
784                 "otherwise: stepvalue = 1"
785             ]
786         }
787     },
788     "selectablehumiditylevels": {
789         "type": "array",
790         "items": {
791             "type": "number"
792         },
793         "x-ocf-conversion": {
794             "x-ocf-alias": "oic.r.selectablelevels",
795             "x-to-ocf": [
796                 "availablelevels[] = selectablehumiditylevels[]"
797             ],
798             "x-from-ocf": [
799                 "selectablehumiditylevels[] = availablelevels[]"
800             ]
801         }
802     }
803 }
804 }

```

```

805     },
806     "type": "object",
807     "allOf": [
808       { "$ref": "#/definitions/asa.environment.targethumidity" }
809     ],
810     "required": [ "targetvalue", "minvalue", "maxvalue", "stepvalue", "selectablehumiditylevels" ]
811   }
812

```

813 7.7 Target Temperature

814 7.7.1 Derived model

815 The derived model: "asa.environment.targettemperature".

816 7.7.2 Property definition

817 Table 16 provides the detailed per Property mapping for "asa.environment.targettemperature".

818 **Table 16 – The property mapping for "asa.environment.targettemperature".**

AllJoyn name	Property	OCF Resource	To OCF	From OCF
minvalue		oic.r.temperature	range[0] = minvalue	minvalue = range[0] otherwise: minvalue = -MAXINT
targetvalue		oic.r.temperature	temperature = targetvalue units = C	oneOf
maxvalue		oic.r.temperature	range[1] = maxvalue	maxvalue = range[1] otherwise: maxvalue = MAXINT
step		oic.r.temperature	ocf.step = step	step = ocf.step otherwise: step = undefined (0x00)

819 Table 17 provides the details of the Properties that are part of "asa.environment.targettemperature".

820 **Table 17 – The properties of "asa.environment.targettemperature".**

AllJoyn name	Property	Type	Required	Description
minvalue		number	yes	
targetvalue		number	yes	Measured value
maxvalue		number	yes	
step		number	yes	

821 7.7.3 Derived model definition

```

822 {
823   "id": "http://openinterconnect.org/asamapping/schemas/asa.environment.targettemperature.json#",
824   "$schema": "http://json-schema.org/draft-04/schema#",
825   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
826   "title": "Target Temperature",
827   "definitions": {
828     "asa.environment.targettemperature": {
829       "type": "object",
830       "properties": {
831         "targetvalue": {
832           "type": "number",
833           "description": "Measured value",
834           "x-ocf-conversion": {
835             "x-ocf-alias": "oic.r.temperature",
836             "x-to-ocf": [
837               "temperature = targetvalue",
838               "units = C"

```

```

839     ],
840     "x-from-ocf": {
841         "oneOf": [
842             {
843                 "properties": {
844                     "units": "string",
845                     "enum": ["C"]
846                 },
847                 "x-from-ocf": [
848                     "targetvalue = temperature"
849                 ]
850             },
851             {
852                 "properties": {
853                     "units": "string",
854                     "enum": ["F"]
855                 },
856                 "x-from-ocf": [
857                     "targetvalue = (temperature-32)*5/9"
858                 ]
859             },
860             {
861                 "properties": {
862                     "units": "string",
863                     "enum": ["K"]
864                 },
865                 "x-from-ocf": [
866                     "targetvalue = temperature-273.15"
867                 ]
868             }
869         ]
870     }
871 },
872 "minvalue": {
873     "type": "number",
874     "x-ocf-conversion": {
875         "x-ocf-alias": "oic.r.temperature",
876         "x-to-ocf": [
877             "range[0] = minvalue"
878         ],
879     },
880     "x-from-ocf": [
881         "minvalue = range[0]",
882         "otherwise: minvalue = -MAXINT"
883     ]
884 },
885 },
886 "maxvalue": {
887     "type": "number",
888     "x-ocf-conversion": {
889         "x-ocf-alias": "oic.r.temperature",
890         "x-to-ocf": [
891             "range[1] = maxvalue"
892         ],
893     },
894     "x-from-ocf": [
895         "maxvalue = range[1]",
896         "otherwise: maxvalue = MAXINT"
897     ]
898 },
899 "step": {
900     "type": "number",
901     "x-ocf-conversion": {
902         "x-ocf-alias": "oic.r.temperature",
903         "x-to-ocf": [
904             "ocf.step = step"
905         ],
906     },
907     "x-from-ocf": [
908         "step = ocf.step",
909         "otherwise: step = undefined (0x00)"
910     ]
911 }

```

```

910     }
911   }
912 }
913 },
914 },
915 "type": "object",
916 "allOf": [
917   {"$ref": "#/definitions/asa.environment.targettemperature"}
918 ],
919 "required": [ "targetvalue", "minvalue", "maxvalue", "step" ]
920 }
921 }

```

922 7.8 Audio Volume

923 7.8.1 Derived model

924 The derived model: "asa.operation.audiovolume".

925 7.8.2 Property definition

926 Table 18 provides the detailed per Property mapping for "asa.operation.audiovolume".

927 **Table 18 – The property mapping for "asa.operation.audiovolume".**

AllJoyn name	Property	OCF Resource	To OCF	From OCF
mute		oic.r.audio	ocf.mute = mute	mute = ocf.mute
maxvolume		oic.r.audio	range[0] = 0 range[1] = maxvolume	maxvolume = range[1] otherwise: maxvalue = 100
volume		oic.r.audio	ocf.volume = volume	volume = ocf.volume

928 Table 19 provides the details of the Properties that are part of "asa.operation.audiovolume".

929 **Table 19 – The properties of "asa.operation.audiovolume".**

AllJoyn name	Property	Type	Required	Description
mute		boolean	yes	
maxvolume		integer	yes	
volume		integer	yes	Speaker volume index

930 7.8.3 Derived model definition

```

931 {
932   "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.audiovolume.json#",
933   "$schema": "http://json-schema.org/draft-04/schema#",
934   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
935   "title": "Audio Volume",
936   "definitions": {
937     "asa.operation.audiovolume": {
938       "type": "object",
939       "properties": {
940         "volume": {
941           "type": "integer",
942           "description": "Speaker volume index",
943           "x-ocf-conversion": {
944             "x-ocf-alias": "oic.r.audio",
945             "x-to-ocf": [
946               "ocf.volume = volume"
947             ],
948             "x-from-ocf": [
949               "volume = ocf.volume"
950             ]
951           }
952         },
953         "maxvolume": {

```

```

954     "type": "integer",
955     "x-ocf-conversion": {
956       "x-ocf-alias": "oic.r.audio",
957       "x-to-ocf": [
958         "range[0] = 0",
959         "range[1] = maxvolume"
960       ],
961       "x-from-ocf": [
962         "maxvolume = range[1]",
963         "otherwise: maxvalue = 100"
964       ]
965     }
966   },
967   "mute": {
968     "type": "boolean",
969     "x-ocf-conversion": {
970       "x-ocf-alias": "oic.r.audio",
971       "x-to-ocf": [
972         "ocf.mute = mute"
973       ],
974       "x-from-ocf": [
975         "mute = ocf.mute"
976       ]
977     }
978   }
979 }
980 }
981 },
982 "type": "object",
983 "allOf": [
984   {"$ref": "#/definitions/asa.operation.audiovolume"}
985 ],
986 "required": [ "volume", "maxvolume", "mute" ]
987 }
988

```

989 7.9 Climate Control Mode

990 7.9.1 Derived model

991 The derived model: "asa.operation.climatecontrolmode".

992 7.9.2 Property definition

993 Table 20 provides the detailed per Property mapping for "asa.operation.climatecontrolmode".

994 **Table 20 – The property mapping for "asa.operation.climatecontrolmode".**

AllJoy n Proper ty name	OCF Resourc e	To OCF	From OCF
operati onalsta te	oic.r.ope rationalst ate	machinestates = [Idle,Heating,Cooling,PendingHeat,P endingCool,AuxilliaryHeat]currentma chinestate = machinestates[operationalstate]	statearray = [Idle,Heating,Cooling,PendingHeat ,PendingCool,AuxilliaryHeat]opera tionalstate = indexof statearray[currentmachinestate[0]]
suppor tedmo des	oic.r.mod e	modearray = [Off,Heat,Cool,Auto,AuxilliaryHeat,D ry,ContinuousDry]for x=0, x < sizeof(supportedmodes): ocf.supportedmodes[x] = modearray[supportedmodes[x]]	modearray = [Off,Heat,Cool,Auto,AuxilliaryHeat, Dry,ContinuousDry]for x=0, x < sizeof(supportedmodes): supportedmodes[x] = indexof modearray[ocf.supportedmodes[x]]
mode	oic.r.mod e	modearray = [Off,Heat,Cool,Auto,AuxilliaryHeat,D	modearray = [Off,Heat,Cool,Auto,AuxilliaryHeat,

		ry,ContinuousDry]ocf.mode[0] = modearray[mode]	=	Dry,ContinuousDry]mode = indexof modeArray[ocf.mode[0]]
--	--	--	---	---

995 Table 21 provides the details of the Properties that are part of "asa.operation.climatecontrolmode".

996 **Table 21 – The properties of "asa.operation.climatecontrolmode".**

AllJoyn name	Property	Type	Required	Description
operationalstate		integer	yes	Current status of device
supportedmodes		array	yes	Array of supported modes
mode		integer	yes	Current mode of device.

997 **7.9.3 Derived model definition**

```

998 {
999   "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.climatecontrolmode.json#",
1000   "$schema": "http://json-schema.org/draft-04/schema#",
1001   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1002   "title": "Climate Control Mode",
1003   "definitions": {
1004     "asa.operation.climatecontrolmode": {
1005       "type": "object",
1006       "properties": {
1007         "mode": {
1008           "type": "integer",
1009           "description": "Current mode of device.",
1010           "x-ocf-conversion": {
1011             "x-ocf-alias": "oic.r.mode",
1012             "x-to-ocf": [
1013               "modearray = [Off,Heat,Cool,Auto,AuxilliaryHeat,Dry,ContinuousDry]",
1014               "ocf.mode[0] = modearray[mode]"
1015             ],
1016             "x-from-ocf": [
1017               "modearray = [Off,Heat,Cool,Auto,AuxilliaryHeat,Dry,ContinuousDry]",
1018               "mode = indexof modeArray[ocf.mode[0]]"
1019             ]
1020           }
1021         },
1022         "supportedmodes": {
1023           "type": "array",
1024           "items": {
1025             "type": "integer"
1026           },
1027           "description": "Array of supported modes",
1028           "x-ocf-conversion": {
1029             "x-ocf-alias": "oic.r.mode",
1030             "x-to-ocf": [
1031               "modearray = [Off,Heat,Cool,Auto,AuxilliaryHeat,Dry,ContinuousDry]",
1032               "for x=0, x < sizeof(supportedmodes): ocf.supportedmodes[x] =
1033 modearray[supportedmodes[x]]"
1034             ],
1035             "x-from-ocf": [
1036               "modearray = [Off,Heat,Cool,Auto,AuxilliaryHeat,Dry,ContinuousDry]",
1037               "for x=0, x < sizeof(supportedmodes): supportedmodes[x] = indexof
1038 modearray[ocf.supportedmodes[x]]"
1039             ]
1040           }
1041         },
1042         "operationalstate": {
1043           "type": "integer",
1044           "description": "Current status of device",
1045           "x-ocf-conversion": {
1046             "x-ocf-alias": "oic.r.operationalstate",
1047             "x-to-ocf": [

```

```

1048         "machinestates = [Idle,Heating,Cooling,PendingHeat,PendingCool,AuxilliaryHeat]",
1049         "currentmachinestate = machinestates[operationalstate]"
1050     ],
1051     "x-from-ocf": [
1052         "statearray = [Idle,Heating,Cooling,PendingHeat,PendingCool,AuxilliaryHeat]",
1053         "operationalstate = indexof statearray[currentmachinestate[0]]"
1054     ]
1055 }
1056 }
1057 }
1058 }
1059 },
1060 "type": "object",
1061 "allOf": [
1062     {"$ref": "#/definitions/asa.operation.climatecontrolmode"}
1063 ],
1064 "required": [ "mode","supportedmodes","operationalstate" ]
1065 }
1066

```

1067 **7.10 Closed Status**

1068 **7.10.1 Derived model**

1069 The derived model: "asa.operation.closedstatus".

1070 **7.10.2 Property definition**

1071 Table 22 provides the detailed per Property mapping for "asa.operation.closedstatus".

1072 **Table 22 – The property mapping for "asa.operation.closedstatus".**

AllJoyn name	Property	OCF Resource	To OCF	From OCF
isclosed		oic.r.door	if isClosed ocf.openState = Closed.if !isClosed ocf.openState = Open.	isClosed = (openState == Closed)

1073 Table 23 provides the details of the Properties that are part of "asa.operation.closedstatus".

1074 **Table 23 – The properties of "asa.operation.closedstatus".**

AllJoyn name	Property	Type	Required	Description
isclosed		boolean	yes	Open/Closed status Indicator

1075 **7.10.3 Derived model definition**

```

1076 {
1077     "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.closedstatus.json#",
1078     "$schema": "http://json-schema.org/draft-04/schema#",
1079     "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1080     "title": "Closed Status",
1081     "definitions": {
1082         "asa.operation.closedstatus": {
1083             "type": "object",
1084             "properties": {
1085                 "isclosed": {
1086                     "type": "boolean",
1087                     "description": "Open/Closed status Indicator",
1088                     "x-ocf-conversion": {
1089                         "x-ocf-alias": "oic.r.door",
1090                         "x-to-ocf": [
1091                             "if isClosed ocf.openState = Closed.",
1092                             "if !isClosed ocf.openState = Open."

```

```

1093     ],
1094     "x-from-ocf": [
1095         "isClosed = (openState == Closed)"
1096     ]
1097 }
1098 }
1099 }
1100 }
1101 },
1102 "type": "object",
1103 "allOf": [
1104     {"$ref": "#/definitions/asa.operation.closedstatus"}
1105 ],
1106 "required": [ "isclosed" ]
1107 }
1108

```

1109 **7.11 Cycle Control**

1110 **7.11.1 Derived model**

1111 The derived model: "asa.operation.cyclecontrol".

1112 **7.11.2 Property definition**

1113 Table 24 provides the detailed per Property mapping for "asa.operation.cyclecontrol".

1114 **Table 24 – The property mapping for "asa.operation.cyclecontrol".**

AllJoyn Property name	OCF Resource	To OCF	From OCF
operationalstate	oic.r.operationalstate	statearray [Idle,Working,ReadyToStart,DelayedStart,Pause,EndOfCycle]currentmachinestate = statearray[operationalstate]	statearray [Idle,Working,ReadyToStart,DelayedStart,Pause,EndOfCycle]operationalstate = indexof statearray[currentmachinestate[0]]
executeoperationalcomand	oic.r.action		
SupportedOperationalcomands	oic.r.action		
supportedoperationalstates	oic.r.operationalstate	statearray [Idle,Working,ReadyToStart,DelayedStart,Pause,EndOfCycle]for x=0, x < sizeof(supportedoperationalstates): machinestates[x] = statearray[supportedoperationalstates[x]]	statearray [Idle,Working,ReadyToStart,DelayedStart,Pause,EndOfCycle]for x=0, x < sizeof(machinestates): supportedoperationalstates[x] = indexof statearray[machinestates[x]]

1115 Table 25 provides the details of the Properties that are part of "asa.operation.cyclecontrol".

1116 **Table 25 – The properties of "asa.operation.cyclecontrol".**

AllJoyn Property name	Type	Required	Description
operationalstate	integer	yes	Current operational state of the appliance
executeoperationalcomand		no	Execute an operational command

SupportedOperationalcommands	array	no	Array of operatinal commands supported by the appliance
supportedoperationalstates	array	yes	Array of operational states supported by the Appliance.

1117 7.11.3 Derived model definition

```

1118 {
1119   "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.cyclecontrol.json#",
1120   "$schema": "http://json-schema.org/draft-04/schema#",
1121   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1122   "title": "Cycle Control",
1123   "definitions": {
1124     "asa.operation.cyclecontrol": {
1125       "type": "object",
1126       "properties": {
1127         "operationalstate": {
1128           "type": "integer",
1129           "description": "Current operational state of the appliance",
1130           "x-ocf-conversion": {
1131             "x-ocf-alias": "oic.r.operationalstate",
1132             "x-to-ocf": [
1133               "statearray = [Idle,Working,ReadyToStart,DelayedStart,Pause,EndOfCycle]",
1134               "currentmachinestate = statearray[operationalstate]"
1135             ],
1136             "x-from-ocf": [
1137               "statearray = [Idle,Working,ReadyToStart,DelayedStart,Pause,EndOfCycle]",
1138               "operationalstate = indexof statearray[currentmachinestate[0]]"
1139             ]
1140           }
1141         },
1142         "supportedoperationalstates": {
1143           "type": "array",
1144           "items": {
1145             "type": "integer"
1146           },
1147           "description": "Array of operational states supported by the Appliance.",
1148           "x-ocf-conversion": {
1149             "x-ocf-alias": "oic.r.operationalstate",
1150             "x-to-ocf": [
1151               "statearray = [Idle,Working,ReadyToStart,DelayedStart,Pause,EndOfCycle]",
1152               "for x=0, x < sizeof(supportedoperationalstates): machinestates[x] =
1153 statearray[supportedoperationalstates[x]]"
1154             ],
1155             "x-from-ocf": [
1156               "statearray = [Idle,Working,ReadyToStart,DelayedStart,Pause,EndOfCycle]",
1157               "for x=0, x < sizeof(machinestates): supportedoperationalstates[x] = indexof
1158 statearray[machinestates[x]]"
1159             ]
1160           }
1161         },
1162         "SupportedOperationalcommands": {
1163           "type": "array",
1164           "items": {
1165             "type": "integer"
1166           },
1167           "description": "Array of operatinal commands supported by the appliance",
1168           "x-ocf-conversion": {
1169             "x-ocf-alias": "oic.r.action"
1170           }
1171         },
1172         "executeoperationalcomand": {
1173           "x-ocf-type": "method",
1174           "description": "Execute an operational command",
1175           "x-ocf-conversion": {

```

```

1176         "x-ocf-alias": "oic.r.action"
1177     }
1178 }
1179 }
1180 },
1181 },
1182 "type": "object",
1183 "allOf": [
1184     {"$ref": "#/definitions/asa.operation.cyclecontrol"}
1185 ],
1186 "required": [ "operationalstate","supportedoperationalstates" ]
1187 }
1188

```

1189 **7.12 Fan Speed Level**

1190 **7.12.1 Derived model**

1191 The derived model: "asa.operation.fanspeedlevel".

1192 **7.12.2 Property definition**

1193 Table 26 provides the detailed per Property mapping for "asa.operation.fanspeedlevel".

1194 **Table 26 – The property mapping for "asa.operation.fanspeedlevel".**

AllJoyn name	Property	OCF Resource	To OCF	From OCF
fanspeedlevel		oic.r.airflow	speed = fanspeedlevel	fanspeedlevel = speed
maxfanspeedlevel		oic.r.airflow	range[0] = 0range[1] = maxfanspeedlevel	maxfanspeedlevel = range[1]otherwise: maxfanspeedlevel = 100
automode		oic.r.airflow	if automode != NotSupported(0xFF) ocf.automode = automodeelse no mapping	automode = ocf.automodeotherwise: automode = NotSupported(0xFF)

1195 Table 27 provides the details of the Properties that are part of "asa.operation.fanspeedlevel".

1196 **Table 27 – The properties of "asa.operation.fanspeedlevel".**

AllJoyn name	Property	Type	Required	Description
fanspeedlevel		integer	yes	Fan speed level. 0 = off.
maxfanspeedlevel		integer	yes	Max level allowed for fan speed
automode		integer	yes	Auto mode status.

1197 **7.12.3 Derived model definition**

```

1198 {
1199     "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.fanspeedlevel.json#",
1200     "$schema": "http://json-schema.org/draft-04/schema#",
1201     "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1202     "title": "Fan Speed Level",
1203     "definitions": {
1204         "asa.operation.fanspeedlevel": {
1205             "type": "object",
1206             "properties": {
1207                 "fanspeedlevel": {
1208                     "type": "integer",
1209                     "description": "Fan speed level. 0 = off.",

```

```

1210     "x-ocf-conversion": {
1211         "x-ocf-alias": "oic.r.airflow",
1212         "x-to-ocf": [
1213             "speed = fanspeedlevel"
1214         ],
1215         "x-from-ocf": [
1216             "fanspeedlevel = speed"
1217         ]
1218     },
1219 },
1220 "maxfanspeedlevel": {
1221     "type": "integer",
1222     "description": "Max level allowed for fan speed",
1223     "x-ocf-conversion": {
1224         "x-ocf-alias": "oic.r.airflow",
1225         "x-to-ocf": [
1226             "range[0] = 0",
1227             "range[1] = maxfanspeedlevel"
1228         ],
1229         "x-from-ocf": [
1230             "maxfanspeedlevel = range[1]",
1231             "otherwise: maxfanspeedlevel = 100"
1232         ]
1233     },
1234 },
1235 "automode": {
1236     "type": "integer",
1237     "description": "Auto mode status.",
1238     "x-ocf-conversion": {
1239         "x-ocf-alias": "oic.r.airflow",
1240         "x-to-ocf": [
1241             "if automode != NotSupported(0xFF)",
1242             " ocf.automode = automode",
1243             "else no mapping"
1244         ],
1245         "x-from-ocf": [
1246             "automode = ocf.automode",
1247             "otherwise: automode = NotSupported(0xFF)"
1248         ]
1249     },
1250 }
1251 }
1252 }
1253 },
1254 "type": "object",
1255 "allOf": [
1256     {"$ref": "#/definitions/asa.operation.fanspeedlevel"}
1257 ],
1258 "required": [ "fanspeedlevel", "maxfanspeedlevel", "automode" ]
1259 }
1260

```

1261 7.13 Heating Zone

1262 7.13.1 Derived model

1263 The derived model: "asa.operation.heatingzone".

1264 7.13.2 Property definition

1265 Table 28 provides the detailed per Property mapping for "asa.operation.heatingzone".

1266 **Table 28 – The property mapping for "asa.operation.heatingzone".**

AllJoyn Property name	OCF Resource	To OCF	From OCF
numberofheatingzones	oic.r.heatingzonecollection	number of links in the collection = numberofheatingzones	numberofheatingzones = number of links in the collection

heatinglevels	oic.r.heatingzone	Instance of oic.r.heatingzone per array item for x=0, x<sizeof(heatinglevels): ocf.heatinglevel = maxheatinglevels[x]	for x=0;x<numlinks(oic.r.heatingzonecollection): heatinglevels[x] = ocf.heatinglevel
maxheatinglevels	oic.r.heatingzone	Instance of oic.r.heatingzone per array item for x=0, x<sizeof(maxheatinglevels): ocf.maxheatinglevel = maxheatinglevels[x]	for x=0;x<numlinks(oic.r.heatingzonecollection): maxheatinglevels[x] = ocf.maxheatinglevel

1267 Table 29 provides the details of the Properties that are part of "asa.operation.heatingzone".

1268 **Table 29 – The properties of "asa.operation.heatingzone".**

AllJoyn name	Property	Type	Required	Description
numberofheatingzones		integer	yes	Number of heating zones.
heatinglevels		array	yes	Current heating levels for each zone.
maxheatinglevels		array	yes	Max heating levels for each zone

1269 **7.13.3 Derived model definition**

```

1270 {
1271   "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.heatingzone.json#",
1272   "$schema": "http://json-schema.org/draft-04/schema#",
1273   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1274   "title": "Heating Zone",
1275   "definitions": {
1276     "asa.operation.heatingzone": {
1277       "type": "object",
1278       "properties": {
1279         "numberofheatingzones": {
1280           "type": "integer",
1281           "description": "Number of heating zones.",
1282           "x-ocf-conversion": {
1283             "x-ocf-alias": "oic.r.heatingzonecollection",
1284             "x-to-ocf": [
1285               "number of links in the collection = numberofheatingzones"
1286             ],
1287             "x-from-ocf": [
1288               "numberofheatingzones = number of links in the collection"
1289             ]
1290           }
1291         },
1292         "maxheatinglevels": {
1293           "type": "array",
1294           "items": {
1295             "type": "integer"
1296           },
1297           "description": "Max heating levels for each zone",
1298           "x-ocf-conversion": {
1299             "x-ocf-alias": "oic.r.heatingzone",
1300             "x-to-ocf": [

```

```

1301         "Instance of oic.r.heatingzone per array item ",
1302         "for x=0, x<sizeof(maxheatinglevels): ocf.maxheatinglevel = maxheatinglevels[x]"
1303     ],
1304     "x-from-ocf": [
1305         "for x=0;x<numlinks(oic.r.heatingzonecollection): maxheatinglevels[x] =
1306 ocf.maxheatinglevel"
1307     ]
1308 }
1309 },
1310 "heatinglevels": {
1311     "type": "array",
1312     "items": {
1313         "type": "integer"
1314     },
1315     "description": "Current heating levels for each zone.",
1316     "x-ocf-conversion": {
1317         "x-ocf-alias": "oic.r.heatingzone",
1318         "x-to-ocf": [
1319             "Instance of oic.r.heatingzone per array item ",
1320             "for x=0, x<sizeof(heatinglevels): ocf.heatinglevel = maxheatinglevels[x]"
1321         ],
1322         "x-from-ocf": [
1323             "for x=0;x<numlinks(oic.r.heatingzonecollection): heatinglevels[x] = ocf.heatinglevel"
1324         ]
1325     }
1326 }
1327 }
1328 }
1329 },
1330 "type": "object",
1331 "allOf": [
1332     {"$ref": "#/definitions/asa.operation.heatingzone"}
1333 ],
1334 "required": [ "numberofheatingzones", "maxheatinglevels", "heatinglevels" ]
1335 }
1336

```

1337 **7.14 HVAC Fan Mode**

1338 **7.14.1 Derived model**

1339 The derived model: "asa.operation.hvacfanmode".

1340 **7.14.2 Property definition**

1341 Table 30 provides the detailed per Property mapping for "asa.operation.hvacfanmode".

1342 **Table 30 – The property mapping for "asa.operation.hvacfanmode".**

AllJoyn Property name	OCF Resource	To OCF	From OCF
mode	oic.r.mode	modearray = [Auto,Circulation,Continuous]ocf.mode[0] = modearray[mode]	modearray = [Auto,Circulation,Continuous]mode = indexof modeArray[ocf.mode[0]]
supportedmodes	oic.r.mode	modearray = [Auto,Circulation,Continuous]for x=0, x < sizeof(supportedmodes): ocf.supportedmodes[x] = modearray[supportedmodes[x]]	modearray = [Auto,Circulation,Continuous]for x=0, x < sizeof(supportedmodes): supportedmodes[x] = indexof modearray[ocf.supportedmodes[x]]

1343 Table 31 provides the details of the Properties that are part of "asa.operation.hvacfanmode".

Table 31 – The properties of "asa.operation.hvacfanmode".

AllJoyn name	Property	Type	Required	Description
mode		integer	yes	Current mode of device.
supportedmodes		array	yes	Array of supported modes

1345 **7.14.3 Derived model definition**

```

1346 {
1347   "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.hvacfanmode.json#",
1348   "$schema": "http://json-schema.org/draft-04/schema#",
1349   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1350   "title": "HVAC Fan Mode",
1351   "definitions": {
1352     "asa.operation.hvacfanmode": {
1353       "type": "object",
1354       "properties": {
1355         "mode": {
1356           "type": "integer",
1357           "description": "Current mode of device.",
1358           "x-ocf-conversion": {
1359             "x-ocf-alias": "oic.r.mode",
1360             "x-to-ocf": [
1361               "modearray = [Auto,Circulation,Continuous]",
1362               "ocf.mode[0] = modearray[mode]"
1363             ],
1364             "x-from-ocf": [
1365               "modearray = [Auto,Circulation,Continuous]",
1366               "mode = indexof modeArray[ocf.mode[0]]"
1367             ]
1368           }
1369         },
1370         "supportedmodes": {
1371           "type": "array",
1372           "items": {
1373             "type": "integer"
1374           },
1375           "description": "Array of supported modes",
1376           "x-ocf-conversion": {
1377             "x-ocf-alias": "oic.r.mode",
1378             "x-to-ocf": [
1379               "modearray = [Auto,Circulation,Continuous]",
1380               "for x=0, x < sizeof(supportedmodes): ocf.supportedmodes[x] =
1381 modearray[supportedmodes[x]]"
1382             ],
1383             "x-from-ocf": [
1384               "modearray = [Auto,Circulation,Continuous]",
1385               "for x=0, x < sizeof(supportedmodes): supportedmodes[x] = indexof
1386 modearray[ocf.supportedmodes[x]]"
1387             ]
1388           }
1389         }
1390       }
1391     }
1392   },
1393   "type": "object",
1394   "allOf": [
1395     { "$ref": "#/definitions/asa.operation.hvacfanmode" }
1396   ],
1397   "required": [ "mode", "supportedmodes" ]
1398 }
1399

```

1400 **7.15 On/Off Control**

1401 **7.15.1 Derived model**

1402 The derived model: "asa.operation.offcontrol".

1403 The derived model: "asa.operation.oncontrol".

1404 **7.15.2 Property definition**

1405 Table 32 provides the detailed per Property mapping for "asa.operation.offcontrol".

1406 **Table 32 – The property mapping for "asa.operation.offcontrol".**

AllJoyn Property name	OCF Resource	To OCF	From OCF
switchon	oic.r.switch.binary	value = false	if ocf.value = false, asa.operation.offcontrol::switchoff().

1407 Table 33 provides the details of the Properties that are part of "asa.operation.offcontrol".

1408 **Table 33 – The properties of "asa.operation.offcontrol".**

AllJoyn Property name	Property	Type	Required	Description
switchon		string	no	Turn off the device

1409 Table 34 provides the detailed per Property mapping for "asa.operation.oncontrol".

1410 **Table 34 – The property mapping for "asa.operation.oncontrol".**

AllJoyn Property name	OCF Resource	To OCF	From OCF
switchon	oic.r.switch.binary	value = true	if ocf.value = true, asa.operation.oncontrol::switchon().

1411 Table 35 provides the details of the Properties that are part of "asa.operation.oncontrol".

1412 **Table 35 – The properties of "asa.operation.oncontrol".**

AllJoyn Property name	Property	Type	Required	Description
switchon		string	no	Turn on the device

1413 **7.15.3 Derived model definition**

```

1414 {
1415   "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.oncontrol.json#",
1416   "$schema": "http://json-schema.org/draft-04/schema#",
1417   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1418   "title": "On/Off Control",
1419   "definitions": {
1420     "asa.operation.oncontrol": {
1421       "type": "object",
1422       "properties": {
1423         "switchon": {
1424           "type": "string",
1425           "format": "method",
1426           "description": "Turn on the device",
1427           "x-ocf-conversion": {
1428             "x-ocf-alias": "oic.r.switch.binary",
1429             "x-to-ocf": [
1430               "value = true"
1431             ],
1432             "x-from-ocf": [
1433               "if ocf.value = true, asa.operation.oncontrol::switchon()."
1434             ]
1435           }
1436         }
1437       }
1438     }
1439   }

```

```

1435     }
1436   }
1437 }
1438 },
1439 "asa.operation.offcontrol": {
1440   "type": "object",
1441   "properties": {
1442     "switchon": {
1443       "type": "string",
1444       "format": "method",
1445       "description": "Turn off the device",
1446       "x-ocf-conversion": {
1447         "x-ocf-alias": "oic.r.switch.binary",
1448         "x-to-ocf": [
1449           "value = false"
1450         ],
1451         "x-from-ocf": [
1452           "if ocf.value = false, asa.operation.offcontrol::switchoff()."
1453         ]
1454       }
1455     }
1456   }
1457 },
1458 },
1459 "type": "object",
1460 "oneOf": [
1461   {"$ref": "#/definitions/asa.operation.oncontrol"},
1462   {"$ref": "#/definitions/asa.operation.offcontrol"}
1463 ]
1464 }
1465

```

1466 7.16 On Off Mapping

1467 7.16.1 Derived model

1468 The derived model: "asa.operation.onoffstatus".

1469 7.16.2 Property definition

1470 Table 36 provides the detailed per Property mapping for "asa.operation.onoffstatus".

1471 **Table 36 – The property mapping for "asa.operation.onoffstatus".**

AllJoyn name	Property	OCF Resource	To OCF	From OCF
onoff		oic.r.switch.binary	value = onoff	onoff = value

1472 Table 37 provides the details of the Properties that are part of "asa.operation.onoffstatus".

1473 **Table 37 – The properties of "asa.operation.onoffstatus".**

AllJoyn name	Property	Type	Required	Description
onoff		boolean	yes	On/Off status of the device

1474 7.16.3 Derived model definition

```

1475 {
1476   "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.onoffstatus.json#",
1477   "$schema": "http://json-schema.org/draft-04/schema#",
1478   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1479   "title": "On Off Mapping",
1480   "definitions": {
1481     "asa.operation.onoffstatus": {
1482       "type": "object",
1483       "properties": {
1484         "onoff": {
1485           "type": "boolean",

```



```

1486     "description": "On/Off status of the device",
1487     "x-ocf-conversion": {
1488       "x-ocf-alias": "oic.r.switch.binary",
1489       "x-to-ocf": [
1490         "value = onoff"
1491       ],
1492       "x-from-ocf": [
1493         "onoff = value"
1494       ]
1495     }
1496   }
1497 }
1498 }
1499 },
1500 "type": "object",
1501 "allOf": [
1502   {"$ref": "#/definitions/asa.operation.onoffstatus"}
1503 ],
1504 "required": [ "onoff" ]
1505 }
1506

```

1507 **7.17 Oven Cycle Phase**

1508 **7.17.1 Derived model**

1509 The derived model: "asa.operation.ovencyclephase".

1510 **7.17.2 Property definition**

1511 Table 38 provides the detailed per Property mapping for "asa.operation.ovencyclephase".

1512 **Table 38 – The property mapping for "asa.operation.ovencyclephase".**

AllJoyn Property name	OCF Resource	To OCF	From OCF
getvendorphasesdescription	oic.r.action		
supportedcyclephases	oic.r.operationalstate	phasearray [Unavailable,Preheating,Cooking,Cleaning]for x=0, x < sizeof(supportedcyclephases): machinestates[x] = phasearray[supportedcyclephases[x]]	phasearray [Unavailable,Preheating,Cooking,Cleaning]for x=0, x < sizeof(machinestates): supportedcyclephases[x] = indexof phasearray[machinestates[x]]
cyclephase	oic.r.operationalstate	phasearray [Unavailable,Preheating,Cooking,Cleaning]currentmachinestate = phasearray[cyclephase]	phasearray [Unavailable,Preheating,Cooking,Cleaning]cyclephase = indexof statearray[currentmachinestate[0]]

1513 Table 39 provides the details of the Properties that are part of "asa.operation.ovencyclephase".

1514 **Table 39 – The properties of "asa.operation.ovencyclephase".**

AllJoyn Property name	Type	Required	Description
getvendorphasesdescription		no	Get cycle phases description
supportedcyclephases	array	yes	Array of cycle phases supported by the Appliance.

cyclephase	integer	yes	Current phase of the operational cycle
------------	---------	-----	--

1515 **7.17.3 Derived model definition**

```

1516 {
1517   "id": "http://openinterconnect.org/asamapping/schemas/asa.operation.ovencyclephase.json#",
1518   "$schema": "http://json-schema.org/draft-04/schema#",
1519   "description": "Copyright (c) 2017 Open Connectivity Foundation, Inc. All rights reserved.",
1520   "title": "Oven Cycle Phase",
1521   "definitions": {
1522     "asa.operation.ovencyclephase": {
1523       "type": "object",
1524       "properties": {
1525         "cyclephase": {
1526           "type": "integer",
1527           "description": "Current phase of the operational cycle",
1528           "x-ocf-conversion": {
1529             "x-ocf-alias": "oic.r.operationalstate",
1530             "x-to-ocf": [
1531               "phasearray = [Unavailable,Preheating,Cooking,Cleaning]",
1532               "currentmachinestate = phasearray[cyclephase]"
1533             ],
1534             "x-from-ocf": [
1535               "phasearray = [Unavailable,Preheating,Cooking,Cleaning]",
1536               "cyclephase = indexOf statearray[currentmachinestate[0]]"
1537             ]
1538           },
1539         },
1540         "supportedcyclephases": {
1541           "type": "array",
1542           "items": {
1543             "type": "integer"
1544           },
1545           "description": "Array of cycle phases supported by the Appliance.",
1546           "x-ocf-conversion": {
1547             "x-ocf-alias": "oic.r.operationalstate",
1548             "x-to-ocf": [
1549               "phasearray = [Unavailable,Preheating,Cooking,Cleaning]",
1550               "for x=0, x < sizeof(supportedcyclephases): machinestates[x] =
1551 phasearray[supportedcyclephases[x]]"
1552             ],
1553             "x-from-ocf": [
1554               "phasearray = [Unavailable,Preheating,Cooking,Cleaning]",
1555               "for x=0, x < sizeof(machinestates): supportedcyclephases[x] = indexOf
1556 phasearray[machinestates[x]]"
1557             ]
1558           },
1559         },
1560         "getvendorphasesdescription": {
1561           "x-ocf-type": "method",
1562           "description": "Get cycle phases description",
1563           "x-ocf-conversion": {
1564             "x-ocf-alias": "oic.r.action"
1565           }
1566         }
1567       }
1568     },
1569   },
1570   "type": "object",
1571   "allOf": [
1572     {"$ref": "#/definitions/asa.operation.ovencyclephase"}
1573   ],
1574   "required": [ "cyclephase", "supportedcyclephases" ]
1575 }
1576

```