

# Unittest for smart\_brain

August 29, 2025

## Contents

|          |                                                                                                             |           |
|----------|-------------------------------------------------------------------------------------------------------------|-----------|
| <b>1</b> | <b>Test Information</b>                                                                                     | <b>11</b> |
| 1.1      | Test Candidate Information . . . . .                                                                        | 11        |
| 1.2      | Unittest Information . . . . .                                                                              | 11        |
| 1.3      | Test System Information . . . . .                                                                           | 11        |
| <b>2</b> | <b>Statistic</b>                                                                                            | <b>11</b> |
| 2.1      | Test-Statistic for testrun with python3.13.5 . . . . .                                                      | 11        |
| 2.2      | Coverage Statistic . . . . .                                                                                | 12        |
| <b>3</b> | <b>Testcases with no corresponding Requirement</b>                                                          | <b>13</b> |
| 3.1      | Summary for testrun with python3.13.5 . . . . .                                                             | 13        |
| 3.1.1    | Clean-Up . . . . .                                                                                          | 13        |
| 3.1.2    | ViDevCommon.state (ffe.livingroom.main_light) → Shelly.relay/0 (ffe.livingroom.main_light) . . . . .        | 13        |
| 3.1.3    | Shelly.relay/0 (ffe.livingroom.main_light) → ViDevCommon.state (ffe.livingroom.main_light) . . . . .        | 13        |
| 3.1.4    | ViDevCommon.state (ffe.livingroom.floorlamp) → Light.state (ffe.livingroom.floor_light) . . . . .           | 14        |
| 3.1.5    | Light.state (ffe.livingroom.floor_light) → ViDevCommon.state (ffe.livingroom.floorlamp) . . . . .           | 14        |
| 3.1.6    | Shelly.relay/0 (ffe.livingroom.main_light) → Light.state (ffe.livingroom.floor_light) . . . . .             | 15        |
| 3.1.7    | ViDevCommon.state (ffe.livingroom.xmas_tree) → Powerplug1P.state (ffe.livingroom.xmas-tree) . . . . .       | 15        |
| 3.1.8    | Powerplug1P.state (ffe.livingroom.xmas-tree) → ViDevCommon.state (ffe.livingroom.xmas_tree) . . . . .       | 16        |
| 3.1.9    | ViDevCommon.brightness (ffe.livingroom.main_light) → Light.brightness (ffe.livingroom.main_light) . . . . . | 16        |
| 3.1.10   | Light.brightness (ffe.livingroom.main_light) → ViDevCommon.brightness (ffe.livingroom.main_light) . . . . . | 17        |
| 3.1.11   | ViDevCommon.color_temp (ffe.livingroom.main_light) → Light.color_temp (ffe.livingroom.main_light) . . . . . | 17        |
| 3.1.12   | Light.color_temp (ffe.livingroom.main_light) → ViDevCommon.color_temp (ffe.livingroom.main_light) . . . . . | 18        |
| 3.1.13   | ViDevCommon.brightness (ffe.livingroom.floorlamp) → Light.brightness (ffe.livingroom.floor_light) . . . . . | 19        |
| 3.1.14   | Light.brightness (ffe.livingroom.floor_light) → ViDevCommon.brightness (ffe.livingroom.floorlamp) . . . . . | 19        |

|        |                                                                                                               |    |
|--------|---------------------------------------------------------------------------------------------------------------|----|
| 3.1.15 | ViDevCommon.color_temp (ffe.livingroom.floorlamp) → Light.color_temp (ffe.livingroom.floor_light)             | 20 |
| 3.1.16 | Light.color_temp (ffe.livingroom.floor_light) → ViDevCommon.color_temp (ffe.livingroom.floorlamp)             | 21 |
| 3.1.17 | ViDevHeating.temp_setp (ffe.livingroom.heating_valve) → HeatingValve.temp_setp (ffe.livingroom.heating_valve) | 22 |
| 3.1.18 | ViDevCommon.state (ffe.sleep.main_light) → Shelly.relay/0 (ffe.sleep.main_light)                              | 22 |
| 3.1.19 | Shelly.relay/0 (ffe.sleep.main_light) → ViDevCommon.state (ffe.sleep.main_light)                              | 23 |
| 3.1.20 | ViDevCommon.state (ffe.sleep.bed_light_di) → Light.state (ffe.sleep.bed_light_di)                             | 23 |
| 3.1.21 | Light.state (ffe.sleep.bed_light_di) → ViDevCommon.state (ffe.sleep.bed_light_di)                             | 23 |
| 3.1.22 | ViDevCommon.state (ffe.sleep.bed_light_ma) → Powerplug1P.state (ffe.sleep.bed_light_ma)                       | 24 |
| 3.1.23 | Powerplug1P.state (ffe.sleep.bed_light_ma) → ViDevCommon.state (ffe.sleep.bed_light_ma)                       | 24 |
| 3.1.24 | ViDevCommon.brightness (ffe.sleep.main_light) → Light.brightness (ffe.sleep.main_light)                       | 25 |
| 3.1.25 | Light.brightness (ffe.sleep.main_light) → ViDevCommon.brightness (ffe.sleep.main_light)                       | 25 |
| 3.1.26 | ViDevCommon.color_temp (ffe.sleep.main_light) → Light.color_temp (ffe.sleep.main_light)                       | 26 |
| 3.1.27 | Light.color_temp (ffe.sleep.main_light) → ViDevCommon.color_temp (ffe.sleep.main_light)                       | 27 |
| 3.1.28 | ViDevCommon.brightness (ffe.sleep.bed_light_di) → Light.brightness (ffe.sleep.bed_light_di)                   | 28 |
| 3.1.29 | Light.brightness (ffe.sleep.bed_light_di) → ViDevCommon.brightness (ffe.sleep.bed_light_di)                   | 28 |
| 3.1.30 | ViDevHeating.temp_setp (ffe.sleep.heating_valve) → HeatingValve.temp_setp (ffe.sleep.heating_valve)           | 29 |
| 3.1.31 | ViDevCommon.state (ffe.diningroom.main_light) → Shelly.relay/0 (ffe.diningroom.main_light)                    | 30 |
| 3.1.32 | Shelly.relay/0 (ffe.diningroom.main_light) → ViDevCommon.state (ffe.diningroom.main_light)                    | 30 |
| 3.1.33 | ViDevCommon.state (ffe.diningroom.floorlamp) → Powerplug1P.state (ffe.diningroom.floor_light)                 | 30 |
| 3.1.34 | Powerplug1P.state (ffe.diningroom.floor_light) → ViDevCommon.state (ffe.diningroom.floorlamp)                 | 31 |
| 3.1.35 | Shelly.relay/0 (ffe.diningroom.main_light) → Powerplug1P.state (ffe.diningroom.floor_light)                   | 31 |
| 3.1.36 | ViDevCommon.state (ffe.diningroom.garland) → Powerplug1P.state (ffe.diningroom.garland)                       | 32 |
| 3.1.37 | Powerplug1P.state (ffe.diningroom.garland) → ViDevCommon.state (ffe.diningroom.garland)                       | 32 |
| 3.1.38 | ViDevCommon.state (ffe.kitchen.main_light) → Shelly.relay/0 (ffe.kitchen.main_light)                          | 33 |
| 3.1.39 | Shelly.relay/0 (ffe.kitchen.main_light) → ViDevCommon.state (ffe.kitchen.main_light)                          | 33 |
| 3.1.40 | ViDevCommon.state (ffe.kitchen.circulation_pump) → Shelly.relay/0 (ffe.kitchen.circulation_pump)              | 34 |

|        |                                                                                                               |    |
|--------|---------------------------------------------------------------------------------------------------------------|----|
| 3.1.41 | Shelly.relay/0 (ffe.kitchen.circulation_pump) → ViDevCommon.state (ffe.kitchen.circulation_pump)              | 34 |
| 3.1.42 | ViDevHeating.temp_setp (ffe.kitchen.heating_valve) → HeatingValve.temp_setp (ffe.kitchen.heating_valve)       | 34 |
| 3.1.43 | ViDevCommon.state (ffe.floor.main_light) → Shelly.relay/0 (ffe.floor.main_light)                              | 35 |
| 3.1.44 | Shelly.relay/0 (ffe.floor.main_light) → ViDevCommon.state (ffe.floor.main_light)                              | 35 |
| 3.1.45 | ViDevCommon.state (ffw.livingroom.main_light) → Shelly.relay/0 (ffw.livingroom.main_light)                    | 36 |
| 3.1.46 | Shelly.relay/0 (ffw.livingroom.main_light) → ViDevCommon.state (ffw.livingroom.main_light)                    | 36 |
| 3.1.47 | ViDevCommon.brightness (ffw.livingroom.main_light) → Light.brightness (ffw.livingroom.main_light)             | 37 |
| 3.1.48 | Light.brightness (ffw.livingroom.main_light) → ViDevCommon.brightness (ffw.livingroom.main_light)             | 37 |
| 3.1.49 | ViDevCommon.color_temp (ffw.livingroom.main_light) → Light.color_temp (ffw.livingroom.main_light)             | 38 |
| 3.1.50 | Light.color_temp (ffw.livingroom.main_light) → ViDevCommon.color_temp (ffw.livingroom.main_light)             | 39 |
| 3.1.51 | ViDevHeating.temp_setp (ffw.livingroom.heating_valve) → HeatingValve.temp_setp (ffw.livingroom.heating_valve) | 40 |
| 3.1.52 | ViDevCommon.state (ffw.sleep.main_light) → Shelly.relay/0 (ffw.sleep.main_light)                              | 40 |
| 3.1.53 | Shelly.relay/0 (ffw.sleep.main_light) → ViDevCommon.state (ffw.sleep.main_light)                              | 41 |
| 3.1.54 | ViDevCommon.brightness (ffw.sleep.main_light) → Light.brightness (ffw.sleep.main_light)                       | 41 |
| 3.1.55 | Light.brightness (ffw.sleep.main_light) → ViDevCommon.brightness (ffw.sleep.main_light)                       | 42 |
| 3.1.56 | ViDevHeating.temp_setp (ffw.sleep.heating_valve) → HeatingValve.temp_setp (ffw.sleep.heating_valve)           | 42 |
| 3.1.57 | ViDevCommon.state (ffw.julian.main_light) → Shelly.relay/0 (ffw.julian.main_light)                            | 43 |
| 3.1.58 | Shelly.relay/0 (ffw.julian.main_light) → ViDevCommon.state (ffw.julian.main_light)                            | 43 |
| 3.1.59 | ViDevCommon.brightness (ffw.julian.main_light) → Light.brightness (ffw.julian.main_light)                     | 44 |
| 3.1.60 | Light.brightness (ffw.julian.main_light) → ViDevCommon.brightness (ffw.julian.main_light)                     | 45 |
| 3.1.61 | ViDevCommon.color_temp (ffw.julian.main_light) → Light.color_temp (ffw.julian.main_light)                     | 45 |
| 3.1.62 | Light.color_temp (ffw.julian.main_light) → ViDevCommon.color_temp (ffw.julian.main_light)                     | 46 |
| 3.1.63 | ViDevHeating.temp_setp (ffw.julian.heating_valve) → HeatingValve.temp_setp (ffw.julian.heating_valve)         | 47 |
| 3.1.64 | ViDevCommon.state (ffw.bath.main_light) → Shelly.relay/0 (ffw.bath.main_light)                                | 47 |
| 3.1.65 | Shelly.relay/0 (ffw.bath.main_light) → ViDevCommon.state (ffw.bath.main_light)                                | 48 |

|        |                                                                                                   |    |
|--------|---------------------------------------------------------------------------------------------------|----|
| 3.1.66 | ViDevHeating.temp_setp (ffw.bath.heating_valve) → HeatingValve.temp_setp (ffw.bath.heating_valve) | 48 |
| 3.1.67 | ViDevCommon.state (ffw.floor.main_light) → Shelly.relay/0 (ffw.floor.main_light)                  | 49 |
| 3.1.68 | Shelly.relay/0 (ffw.floor.main_light) → ViDevCommon.state (ffw.floor.main_light)                  | 49 |
| 3.1.69 | ViDevCommon.state (gfw.dirk.main_light) → Shelly.relay/0 (gfw.dirk.main_light)                    | 50 |
| 3.1.70 | Shelly.relay/0 (gfw.dirk.main_light) → ViDevCommon.state (gfw.dirk.main_light)                    | 50 |
| 3.1.71 | ViDevCommon.state (gfw.dirk.desk_light) → Light.state (gfw.dirk.desk_light)                       | 50 |
| 3.1.72 | Light.state (gfw.dirk.desk_light) → ViDevCommon.state (gfw.dirk.desk_light)                       | 51 |
| 3.1.73 | ViDevCommon.state (gfw.dirk.pc_dock) → Powerplug1P.state (gfw.dirk.dock)                          | 51 |
| 3.1.74 | Powerplug1P.state (gfw.dirk.dock) → ViDevCommon.state (gfw.dirk.pc_dock)                          | 52 |
| 3.1.75 | ViDevCommon.state (gfw.dirk.amplifier) → Powerplug4P.amplifier (gfw.dirk.powerplug)               | 52 |
| 3.1.76 | Powerplug4P.amplifier (gfw.dirk.powerplug) → ViDevCommon.state (gfw.dirk.amplifier)               | 53 |
| 3.1.77 | ViDevCommon.state (gfw.dirk.phono) → Powerplug4P.phono (gfw.dirk.powerplug)                       | 53 |
| 3.1.78 | Powerplug4P.phono (gfw.dirk.powerplug) → ViDevCommon.state (gfw.dirk.phono)                       | 54 |
| 3.1.79 | ViDevCommon.state (gfw.dirk.cd_player) → Powerplug4P.cd-player (gfw.dirk.powerplug)               | 54 |
| 3.1.80 | Powerplug4P.cd-player (gfw.dirk.powerplug) → ViDevCommon.state (gfw.dirk.cd_player)               | 54 |
| 3.1.81 | ViDevCommon.state (gfw.dirk.bt) → Powerplug4P.bluetooth (gfw.dirk.powerplug)                      | 55 |
| 3.1.82 | Powerplug4P.bluetooth (gfw.dirk.powerplug) → ViDevCommon.state (gfw.dirk.bt)                      | 55 |
| 3.1.83 | Powerplug4P.phono (gfw.dirk.powerplug) → Powerplug4P.amplifier (gfw.dirk.powerplug)               | 56 |
| 3.1.84 | Powerplug4P.cd-player (gfw.dirk.powerplug) → Powerplug4P.amplifier (gfw.dirk.powerplug)           | 56 |
| 3.1.85 | Powerplug4P.bluetooth (gfw.dirk.powerplug) → Powerplug4P.amplifier (gfw.dirk.powerplug)           | 57 |
| 3.1.86 | ViDevCommon.brightness (gfw.dirk.main_light) → Light.brightness (gfw.dirk.main_light)             | 57 |
| 3.1.87 | Light.brightness (gfw.dirk.main_light) → ViDevCommon.brightness (gfw.dirk.main_light)             | 58 |
| 3.1.88 | ViDevCommon.color_temp (gfw.dirk.main_light) → Light.color_temp (gfw.dirk.main_light)             | 59 |
| 3.1.89 | Light.color_temp (gfw.dirk.main_light) → ViDevCommon.color_temp (gfw.dirk.main_light)             | 59 |
| 3.1.90 | ViDevCommon.brightness (gfw.dirk.desk_light) → Light.brightness (gfw.dirk.desk_light)             | 60 |
| 3.1.91 | Light.brightness (gfw.dirk.desk_light) → ViDevCommon.brightness (gfw.dirk.desk_light)             | 61 |
| 3.1.92 | ViDevCommon.color_temp (gfw.dirk.desk_light) → Light.color_temp (gfw.dirk.desk_light)             | 61 |
| 3.1.93 | Light.color_temp (gfw.dirk.desk_light) → ViDevCommon.color_temp (gfw.dirk.desk_light)             | 62 |
| 3.1.94 | ViDevHeating.temp_setp (gfw.dirk.heating_valve) → HeatingValve.temp_setp (gfw.dirk.heating_valve) | 63 |

|         |                                                                                                       |    |
|---------|-------------------------------------------------------------------------------------------------------|----|
| 3.1.95  | ViDevCommon.state (gfw.marion.main_light) → Shelly.relay/0 (gfw.marion.main_light)                    | 63 |
| 3.1.96  | Shelly.relay/0 (gfw.marion.main_light) → ViDevCommon.state (gfw.marion.main_light)                    | 64 |
| 3.1.97  | ViDevCommon.state (gfw.marion.window_light) → Light.state (gfw.marion.window_light)                   | 64 |
| 3.1.98  | Light.state (gfw.marion.window_light) → ViDevCommon.state (gfw.marion.window_light)                   | 65 |
| 3.1.99  | Shelly.relay/0 (gfw.marion.main_light) → Light.state (gfw.marion.window_light)                        | 65 |
| 3.1.100 | ViDevHeating.temp_setp (gfw.marion.heating_valve) → HeatingValve.temp_setp (gfw.marion.heating_valve) | 66 |
| 3.1.101 | ViDevCommon.state (gfw.floor.main_light) → Shelly.relay/0 (gfw.floor.main_light)                      | 66 |
| 3.1.102 | Shelly.relay/0 (gfw.floor.main_light) → ViDevCommon.state (gfw.floor.main_light)                      | 67 |
| 3.1.103 | ViDevCommon.brightness (gfw.floor.main_light) → Light.brightness (gfw.floor.main_light)               | 67 |
| 3.1.104 | Light.brightness (gfw.floor.main_light) → ViDevCommon.brightness (gfw.floor.main_light)               | 68 |
| 3.1.105 | ViDevCommon.color_temp (gfw.floor.main_light) → Light.color_temp (gfw.floor.main_light)               | 68 |
| 3.1.106 | Light.color_temp (gfw.floor.main_light) → ViDevCommon.color_temp (gfw.floor.main_light)               | 69 |
| 3.1.107 | ViDevCommon.state (stw.stairway.main_light) → Shelly.relay/0 (stw.firstfloor.main_light)              | 70 |
| 3.1.108 | Shelly.relay/0 (stw.firstfloor.main_light) → ViDevCommon.state (stw.stairway.main_light)              | 70 |

## A Trace for testrun with python3.13.5 72

|        |                                                                                                   |    |
|--------|---------------------------------------------------------------------------------------------------|----|
| A.1    | Tests with status Info (108)                                                                      | 72 |
| A.1.1  | Clean-Up                                                                                          | 72 |
| A.1.2  | ViDevCommon.state (ffe.livingroom.main_light) → Shelly.relay/0 (ffe.livingroom.main_light)        | 74 |
| A.1.3  | Shelly.relay/0 (ffe.livingroom.main_light) → ViDevCommon.state (ffe.livingroom.main_light)        | 76 |
| A.1.4  | ViDevCommon.state (ffe.livingroom.floorlamp) → Light.state (ffe.livingroom.floor_light)           | 80 |
| A.1.5  | Light.state (ffe.livingroom.floor_light) → ViDevCommon.state (ffe.livingroom.floorlamp)           | 82 |
| A.1.6  | Shelly.relay/0 (ffe.livingroom.main_light) → Light.state (ffe.livingroom.floor_light)             | 84 |
| A.1.7  | ViDevCommon.state (ffe.livingroom.xmas_tree) → Powerplug1P.state (ffe.livingroom.xmas-tree)       | 86 |
| A.1.8  | Powerplug1P.state (ffe.livingroom.xmas-tree) → ViDevCommon.state (ffe.livingroom.xmas_tree)       | 87 |
| A.1.9  | ViDevCommon.brightness (ffe.livingroom.main_light) → Light.brightness (ffe.livingroom.main_light) | 88 |
| A.1.10 | Light.brightness (ffe.livingroom.main_light) → ViDevCommon.brightness (ffe.livingroom.main_light) | 92 |
| A.1.11 | ViDevCommon.color_temp (ffe.livingroom.main_light) → Light.color_temp (ffe.livingroom.main_light) | 94 |

|        |                                                                                                               |     |
|--------|---------------------------------------------------------------------------------------------------------------|-----|
| A.1.12 | Light.color_temp (ffe.livingroom.main_light) → ViDevCommon.color_temp (ffe.livingroom.main_light)             | 97  |
| A.1.13 | ViDevCommon.brightness (ffe.livingroom.floorlamp) → Light.brightness (ffe.livingroom.floor_light)             | 99  |
| A.1.14 | Light.brightness (ffe.livingroom.floor_light) → ViDevCommon.brightness (ffe.livingroom.floorlamp)             | 107 |
| A.1.15 | ViDevCommon.color_temp (ffe.livingroom.floorlamp) → Light.color_temp (ffe.livingroom.floor_light)             | 112 |
| A.1.16 | Light.color_temp (ffe.livingroom.floor_light) → ViDevCommon.color_temp (ffe.livingroom.floorlamp)             | 119 |
| A.1.17 | ViDevHeating.temp_setp (ffe.livingroom.heating_valve) → HeatingValve.temp_setp (ffe.livingroom.heating_valve) | 124 |
| A.1.18 | ViDevCommon.state (ffe.sleep.main_light) → Shelly.relay/0 (ffe.sleep.main_light)                              | 127 |
| A.1.19 | Shelly.relay/0 (ffe.sleep.main_light) → ViDevCommon.state (ffe.sleep.main_light)                              | 128 |
| A.1.20 | ViDevCommon.state (ffe.sleep.bed_light_di) → Light.state (ffe.sleep.bed_light_di)                             | 129 |
| A.1.21 | Light.state (ffe.sleep.bed_light_di) → ViDevCommon.state (ffe.sleep.bed_light_di)                             | 130 |
| A.1.22 | ViDevCommon.state (ffe.sleep.bed_light_ma) → Powerplug1P.state (ffe.sleep.bed_light_ma)                       | 131 |
| A.1.23 | Powerplug1P.state (ffe.sleep.bed_light_ma) → ViDevCommon.state (ffe.sleep.bed_light_ma)                       | 132 |
| A.1.24 | ViDevCommon.brightness (ffe.sleep.main_light) → Light.brightness (ffe.sleep.main_light)                       | 133 |
| A.1.25 | Light.brightness (ffe.sleep.main_light) → ViDevCommon.brightness (ffe.sleep.main_light)                       | 136 |
| A.1.26 | ViDevCommon.color_temp (ffe.sleep.main_light) → Light.color_temp (ffe.sleep.main_light)                       | 138 |
| A.1.27 | Light.color_temp (ffe.sleep.main_light) → ViDevCommon.color_temp (ffe.sleep.main_light)                       | 141 |
| A.1.28 | ViDevCommon.brightness (ffe.sleep.bed_light_di) → Light.brightness (ffe.sleep.bed_light_di)                   | 143 |
| A.1.29 | Light.brightness (ffe.sleep.bed_light_di) → ViDevCommon.brightness (ffe.sleep.bed_light_di)                   | 146 |
| A.1.30 | ViDevHeating.temp_setp (ffe.sleep.heating_valve) → HeatingValve.temp_setp (ffe.sleep.heating_valve)           | 148 |
| A.1.31 | ViDevCommon.state (ffe.diningroom.main_light) → Shelly.relay/0 (ffe.diningroom.main_light)                    | 151 |
| A.1.32 | Shelly.relay/0 (ffe.diningroom.main_light) → ViDevCommon.state (ffe.diningroom.main_light)                    | 152 |
| A.1.33 | ViDevCommon.state (ffe.diningroom.floorlamp) → Powerplug1P.state (ffe.diningroom.floor_light)                 | 153 |
| A.1.34 | Powerplug1P.state (ffe.diningroom.floor_light) → ViDevCommon.state (ffe.diningroom.floorlamp)                 | 154 |
| A.1.35 | Shelly.relay/0 (ffe.diningroom.main_light) → Powerplug1P.state (ffe.diningroom.floor_light)                   | 155 |
| A.1.36 | ViDevCommon.state (ffe.diningroom.garland) → Powerplug1P.state (ffe.diningroom.garland)                       | 156 |

|        |                                                                                                               |     |
|--------|---------------------------------------------------------------------------------------------------------------|-----|
| A.1.37 | Powerplug1P.state (ffe.diningroom.garland) → ViDevCommon.state (ffe.diningroom.garland)                       | 157 |
| A.1.38 | ViDevCommon.state (ffe.kitchen.main_light) → Shelly.relay/0 (ffe.kitchen.main_light)                          | 158 |
| A.1.39 | Shelly.relay/0 (ffe.kitchen.main_light) → ViDevCommon.state (ffe.kitchen.main_light)                          | 159 |
| A.1.40 | ViDevCommon.state (ffe.kitchen.circulation_pump) → Shelly.relay/0 (ffe.kitchen.circulation_pump)              | 160 |
| A.1.41 | Shelly.relay/0 (ffe.kitchen.circulation_pump) → ViDevCommon.state (ffe.kitchen.circulation_pump)              | 161 |
| A.1.42 | ViDevHeating.temp_setp (ffe.kitchen.heating_valve) → HeatingValve.temp_setp (ffe.kitchen.heating_valve)       | 162 |
| A.1.43 | ViDevCommon.state (ffe.floor.main_light) → Shelly.relay/0 (ffe.floor.main_light)                              | 164 |
| A.1.44 | Shelly.relay/0 (ffe.floor.main_light) → ViDevCommon.state (ffe.floor.main_light)                              | 165 |
| A.1.45 | ViDevCommon.state (ffw.livingroom.main_light) → Shelly.relay/0 (ffw.livingroom.main_light)                    | 166 |
| A.1.46 | Shelly.relay/0 (ffw.livingroom.main_light) → ViDevCommon.state (ffw.livingroom.main_light)                    | 167 |
| A.1.47 | ViDevCommon.brightness (ffw.livingroom.main_light) → Light.brightness (ffw.livingroom.main_light)             | 168 |
| A.1.48 | Light.brightness (ffw.livingroom.main_light) → ViDevCommon.brightness (ffw.livingroom.main_light)             | 171 |
| A.1.49 | ViDevCommon.color_temp (ffw.livingroom.main_light) → Light.color_temp (ffw.livingroom.main_light)             | 174 |
| A.1.50 | Light.color_temp (ffw.livingroom.main_light) → ViDevCommon.color_temp (ffw.livingroom.main_light)             | 176 |
| A.1.51 | ViDevHeating.temp_setp (ffw.livingroom.heating_valve) → HeatingValve.temp_setp (ffw.livingroom.heating_valve) | 179 |
| A.1.52 | ViDevCommon.state (ffw.sleep.main_light) → Shelly.relay/0 (ffw.sleep.main_light)                              | 181 |
| A.1.53 | Shelly.relay/0 (ffw.sleep.main_light) → ViDevCommon.state (ffw.sleep.main_light)                              | 182 |
| A.1.54 | ViDevCommon.brightness (ffw.sleep.main_light) → Light.brightness (ffw.sleep.main_light)                       | 184 |
| A.1.55 | Light.brightness (ffw.sleep.main_light) → ViDevCommon.brightness (ffw.sleep.main_light)                       | 187 |
| A.1.56 | ViDevHeating.temp_setp (ffw.sleep.heating_valve) → HeatingValve.temp_setp (ffw.sleep.heating_valve)           | 189 |
| A.1.57 | ViDevCommon.state (ffw.julian.main_light) → Shelly.relay/0 (ffw.julian.main_light)                            | 192 |
| A.1.58 | Shelly.relay/0 (ffw.julian.main_light) → ViDevCommon.state (ffw.julian.main_light)                            | 193 |
| A.1.59 | ViDevCommon.brightness (ffw.julian.main_light) → Light.brightness (ffw.julian.main_light)                     | 194 |
| A.1.60 | Light.brightness (ffw.julian.main_light) → ViDevCommon.brightness (ffw.julian.main_light)                     | 196 |
| A.1.61 | ViDevCommon.color_temp (ffw.julian.main_light) → Light.color_temp (ffw.julian.main_light)                     | 199 |

|        |                                                                                                                |     |
|--------|----------------------------------------------------------------------------------------------------------------|-----|
| A.1.62 | Light.color_temp (ffw.julian.main_light) → ViDevCommon.color_temp (ffw.julian.main_light)                      | 201 |
| A.1.63 | ViDevHeating.temp_setp (ffw.julian.heating_valve) → HeatingValve.temp_setp (ffw.julian.heating_valve)<br>..... | 204 |
| A.1.64 | ViDevCommon.state (ffw.bath.main_light) → Shelly.relay/0 (ffw.bath.main_light)                                 | 206 |
| A.1.65 | Shelly.relay/0 (ffw.bath.main_light) → ViDevCommon.state (ffw.bath.main_light)                                 | 207 |
| A.1.66 | ViDevHeating.temp_setp (ffw.bath.heating_valve) → HeatingValve.temp_setp (ffw.bath.heating_valve)<br>.....     | 208 |
| A.1.67 | ViDevCommon.state (ffw.floor.main_light) → Shelly.relay/0 (ffw.floor.main_light)                               | 210 |
| A.1.68 | Shelly.relay/0 (ffw.floor.main_light) → ViDevCommon.state (ffw.floor.main_light)                               | 211 |
| A.1.69 | ViDevCommon.state (gfw.dirk.main_light) → Shelly.relay/0 (gfw.dirk.main_light)                                 | 212 |
| A.1.70 | Shelly.relay/0 (gfw.dirk.main_light) → ViDevCommon.state (gfw.dirk.main_light)                                 | 213 |
| A.1.71 | ViDevCommon.state (gfw.dirk.desk_light) → Light.state (gfw.dirk.desk_light)                                    | 214 |
| A.1.72 | Light.state (gfw.dirk.desk_light) → ViDevCommon.state (gfw.dirk.desk_light)                                    | 215 |
| A.1.73 | ViDevCommon.state (gfw.dirk.pc_dock) → Powerplug1P.state (gfw.dirk.dock)                                       | 216 |
| A.1.74 | Powerplug1P.state (gfw.dirk.dock) → ViDevCommon.state (gfw.dirk.pc_dock)                                       | 217 |
| A.1.75 | ViDevCommon.state (gfw.dirk.amplifier) → Powerplug4P.amplifier (gfw.dirk.powerplug)                            | 218 |
| A.1.76 | Powerplug4P.amplifier (gfw.dirk.powerplug) → ViDevCommon.state (gfw.dirk.amplifier)                            | 219 |
| A.1.77 | ViDevCommon.state (gfw.dirk.phono) → Powerplug4P.phono (gfw.dirk.powerplug)                                    | 220 |
| A.1.78 | Powerplug4P.phono (gfw.dirk.powerplug) → ViDevCommon.state (gfw.dirk.phono)                                    | 221 |
| A.1.79 | ViDevCommon.state (gfw.dirk.cd_player) → Powerplug4P.cd-player (gfw.dirk.powerplug)                            | 222 |
| A.1.80 | Powerplug4P.cd-player (gfw.dirk.powerplug) → ViDevCommon.state (gfw.dirk.cd_player)                            | 223 |
| A.1.81 | ViDevCommon.state (gfw.dirk.bt) → Powerplug4P.bluetooth (gfw.dirk.powerplug)                                   | 224 |
| A.1.82 | Powerplug4P.bluetooth (gfw.dirk.powerplug) → ViDevCommon.state (gfw.dirk.bt)                                   | 225 |
| A.1.83 | Powerplug4P.phono (gfw.dirk.powerplug) → Powerplug4P.amplifier (gfw.dirk.powerplug)                            | 226 |
| A.1.84 | Powerplug4P.cd-player (gfw.dirk.powerplug) → Powerplug4P.amplifier (gfw.dirk.powerplug)                        | 227 |
| A.1.85 | Powerplug4P.bluetooth (gfw.dirk.powerplug) → Powerplug4P.amplifier (gfw.dirk.powerplug)                        | 228 |
| A.1.86 | ViDevCommon.brightness (gfw.dirk.main_light) → Light.brightness (gfw.dirk.main_light)                          | 229 |
| A.1.87 | Light.brightness (gfw.dirk.main_light) → ViDevCommon.brightness (gfw.dirk.main_light)                          | 231 |
| A.1.88 | ViDevCommon.color_temp (gfw.dirk.main_light) → Light.color_temp (gfw.dirk.main_light)                          | 234 |
| A.1.89 | Light.color_temp (gfw.dirk.main_light) → ViDevCommon.color_temp (gfw.dirk.main_light)                          | 236 |
| A.1.90 | ViDevCommon.brightness (gfw.dirk.desk_light) → Light.brightness (gfw.dirk.desk_light)                          | 239 |

|         |                                                                                                       |     |
|---------|-------------------------------------------------------------------------------------------------------|-----|
| A.1.91  | Light.brightness (gfw.dirk.desk_light) → ViDevCommon.brightness (gfw.dirk.desk_light)                 | 242 |
| A.1.92  | ViDevCommon.color_temp (gfw.dirk.desk_light) → Light.color_temp (gfw.dirk.desk_light)                 | 244 |
| A.1.93  | Light.color_temp (gfw.dirk.desk_light) → ViDevCommon.color_temp (gfw.dirk.desk_light)                 | 247 |
| A.1.94  | ViDevHeating.temp_setp (gfw.dirk.heating_valve) → HeatingValve.temp_setp (gfw.dirk.heating_valve)     | 249 |
| A.1.95  | ViDevCommon.state (gfw.marion.main_light) → Shelly.relay/0 (gfw.marion.main_light)                    | 251 |
| A.1.96  | Shelly.relay/0 (gfw.marion.main_light) → ViDevCommon.state (gfw.marion.main_light)                    | 253 |
| A.1.97  | ViDevCommon.state (gfw.marion.window_light) → Light.state (gfw.marion.window_light)                   | 254 |
| A.1.98  | Light.state (gfw.marion.window_light) → ViDevCommon.state (gfw.marion.window_light)                   | 255 |
| A.1.99  | Shelly.relay/0 (gfw.marion.main_light) → Light.state (gfw.marion.window_light)                        | 256 |
| A.1.100 | ViDevHeating.temp_setp (gfw.marion.heating_valve) → HeatingValve.temp_setp (gfw.marion.heating_valve) | 257 |
| A.1.101 | ViDevCommon.state (gfw.floor.main_light) → Shelly.relay/0 (gfw.floor.main_light)                      | 260 |
| A.1.102 | Shelly.relay/0 (gfw.floor.main_light) → ViDevCommon.state (gfw.floor.main_light)                      | 261 |
| A.1.103 | ViDevCommon.brightness (gfw.floor.main_light) → Light.brightness (gfw.floor.main_light)               | 262 |
| A.1.104 | Light.brightness (gfw.floor.main_light) → ViDevCommon.brightness (gfw.floor.main_light)               | 266 |
| A.1.105 | ViDevCommon.color_temp (gfw.floor.main_light) → Light.color_temp (gfw.floor.main_light)               | 269 |
| A.1.106 | Light.color_temp (gfw.floor.main_light) → ViDevCommon.color_temp (gfw.floor.main_light)               | 273 |
| A.1.107 | ViDevCommon.state (stw.stairway.main_light) → Shelly.relay/0 (stw.firstfloor.main_light)              | 276 |
| A.1.108 | Shelly.relay/0 (stw.firstfloor.main_light) → ViDevCommon.state (stw.stairway.main_light)              | 277 |

## B Test-Coverage 278

|       |                         |     |
|-------|-------------------------|-----|
| B.1   | devdi                   | 278 |
| B.1.1 | devdi.__init__.py       | 278 |
| B.1.2 | devdi.rooms.py          | 278 |
| B.1.3 | devdi.topic.py          | 286 |
| B.2   | devices                 | 291 |
| B.2.1 | devices.__init__.py     | 291 |
| B.2.2 | devices.base.py         | 293 |
| B.2.3 | devices.brennenstuhl.py | 297 |
| B.2.4 | devices.hue.py          | 299 |
| B.2.5 | devices.mydevices.py    | 302 |

|        |                               |     |
|--------|-------------------------------|-----|
| B.2.6  | devices.shelly.py             | 306 |
| B.2.7  | devices.silvercrest.py        | 310 |
| B.2.8  | devices.tradfri.py            | 313 |
| B.3    | function                      | 316 |
| B.3.1  | function.__init__.py          | 316 |
| B.3.2  | function.db.py                | 318 |
| B.3.3  | function.first_floor_east.py  | 319 |
| B.3.4  | function.first_floor_west.py  | 323 |
| B.3.5  | function.garden.py            | 325 |
| B.3.6  | function.ground_floor_west.py | 326 |
| B.3.7  | function.helpers.py           | 330 |
| B.3.8  | function.modules.py           | 332 |
| B.3.9  | function.rooms.py             | 338 |
| B.3.10 | function.stairway.py          | 340 |
| B.3.11 | function.videv.py             | 340 |

## 1 Test Information

### 1.1 Test Candidate Information

| Library Information |                                                                                                                             |
|---------------------|-----------------------------------------------------------------------------------------------------------------------------|
| Name                | smart_brain                                                                                                                 |
| Version             | 1.3.2                                                                                                                       |
| Git URL             | <a href="https://git.mount-mockery.de/smarthome/smart_brain.git">https://git.mount-mockery.de/smarthome/smart_brain.git</a> |
| Git REF             | 9c0aff5244690dd8d20697652c8651cdcb805863                                                                                    |

### 1.2 Unittest Information

| Unittest Information |  |
|----------------------|--|
|                      |  |

### 1.3 Test System Information

| System Information |                                                                        |
|--------------------|------------------------------------------------------------------------|
| Architecture       | 64bit                                                                  |
| Machine            | x86_64                                                                 |
| Hostname           | erle                                                                   |
| Distribution       | Debian GNU/Linux 13 (trixie)                                           |
| System             | Linux                                                                  |
| Kernel             | 6.15.1-surface-2 (#2 SMP PREEMPT_DYNAMIC Tue Jun 24 21:02:07 UTC 2025) |
| Username           | dirk                                                                   |
| Path               | /home/dirk/work/smarthome_collection/smart_brain_test                  |

## 2 Statistic

### 2.1 Test-Statistic for testrun with python3.13.5

|                                 |                               |
|---------------------------------|-------------------------------|
| Number of tests                 | <b>108</b>                    |
| Number of successfull tests     | <b>108</b>                    |
| Number of possibly failed tests | <b>0</b>                      |
| Number of failed tests          | <b>0</b>                      |
| Executionlevel                  | Full Test (all defined tests) |
| Time consumption                | 52.866s                       |

## 2.2 Coverage Statistic

| Module- or Filename           | Line-Coverage | Branch-Coverage |
|-------------------------------|---------------|-----------------|
| devdi                         | 99.5%         | 85.7%           |
| devdi.__init__.py             | 100.0%        |                 |
| devdi.rooms.py                | 99.1%         |                 |
| devdi.topic.py                | 100.0%        |                 |
| devices                       | 72.9%         | 40.0%           |
| devices.__init__.py           | 94.7%         |                 |
| devices.base.py               | 55.0%         |                 |
| devices.brennenstuhl.py       | 93.4%         |                 |
| devices.hue.py                | 64.4%         |                 |
| devices.mydevices.py          | 66.7%         |                 |
| devices.shelly.py             | 72.3%         |                 |
| devices.silvercrest.py        | 75.8%         |                 |
| devices.tradfri.py            | 86.0%         |                 |
| function                      | 84.8%         | 46.8%           |
| function.__init__.py          | 87.3%         |                 |
| function.db.py                | 97.7%         |                 |
| function.first_floor_east.py  | 92.0%         |                 |
| function.first_floor_west.py  | 96.9%         |                 |
| function.garden.py            | 74.1%         |                 |
| function.ground_floor_west.py | 93.4%         |                 |
| function.helpers.py           | 98.5%         |                 |
| function.modules.py           | 76.0%         |                 |
| function.rooms.py             | 30.4%         |                 |
| function.stairway.py          | 90.5%         |                 |
| function.videv.py             | 94.1%         |                 |

### 3 Testcases with no corresponding Requirement

#### 3.1 Summary for testrun with python3.13.5

##### 3.1.1 Clean-Up

###### Testresult

This test was passed with the state: **Info**. See also full trace in section A.1.1!

|                     |                                                                                |
|---------------------|--------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                   |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:         | 2025-08-29 17:57:02,708                                                        |
| Finished-Time:      | 2025-08-29 17:57:03,209                                                        |
| Time-Consumption    | 0.500s                                                                         |
| <b>Testsummary:</b> |                                                                                |
| <b>Info</b>         | Collecting precondition logs...                                                |

##### 3.1.2 ViDevCommon.state (ffe.livingroom.main\_light) → Shelly.relay/0 (ffe.livingroom.main\_light)

###### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.2!

|                     |                                                                                                             |
|---------------------|-------------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                                |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                              |
| Start-Time:         | 2025-08-29 17:57:03,209                                                                                     |
| Finished-Time:      | 2025-08-29 17:57:03,512                                                                                     |
| Time-Consumption    | 0.303s                                                                                                      |
| <b>Testsummary:</b> |                                                                                                             |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                                |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                |
| <b>Info</b>         | Setting state of ViDevCommon.state (ffe.livingroom.main_light) to True                                      |
| <b>Success</b>      | Value for Shelly.relay/0 (ffe.livingroom.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.state (ffe.livingroom.main_light) to False                                     |
| <b>Success</b>      | Value for Shelly.relay/0 (ffe.livingroom.main_light) is correct (Content False and Type is <class 'bool'>). |

##### 3.1.3 Shelly.relay/0 (ffe.livingroom.main\_light) → ViDevCommon.state (ffe.livingroom.main\_light)

###### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.3!

|             |                                                                                |
|-------------|--------------------------------------------------------------------------------|
| Testrun:    | python3.13.5                                                                   |
| Caller:     | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time: | 2025-08-29 17:57:03,512                                                        |

Finished-Time: 2025-08-29 17:57:03,815

Time-Consumption 0.303s

---

**Testsummary:**

---

|                |                                                                                                                |
|----------------|----------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                   |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                   |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffe.livingroom.main_light) to True                                            |
| <b>Success</b> | Value for ViDevCommon.state (ffe.livingroom.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffe.livingroom.main_light) to False                                           |
| <b>Success</b> | Value for ViDevCommon.state (ffe.livingroom.main_light) is correct (Content False and Type is <class 'bool'>). |

---

### 3.1.4 ViDevCommon.state (ffe.livingroom.floorlamp) → Light.state (ffe.livingroom.floor\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.4!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:03,815                                                         |
| Finished-Time:   | 2025-08-29 17:57:04,117                                                         |
| Time-Consumption | 0.303s                                                                          |

---

**Testsummary:**

---

|                |                                                                                                           |
|----------------|-----------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                              |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).              |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffe.livingroom.floorlamp) to True                                     |
| <b>Success</b> | Value for Light.state (ffe.livingroom.floor_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffe.livingroom.floorlamp) to False                                    |
| <b>Success</b> | Value for Light.state (ffe.livingroom.floor_light) is correct (Content False and Type is <class 'bool'>). |

---

### 3.1.5 Light.state (ffe.livingroom.floor\_light) → ViDevCommon.state (ffe.livingroom.floorlamp)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.5!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:04,118                                                         |
| Finished-Time:   | 2025-08-29 17:57:04,422                                                         |
| Time-Consumption | 0.304s                                                                          |

---

**Testsummary:**

---

|                |                                                                                              |
|----------------|----------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                 |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>). |

---

|                |                                                                                                               |
|----------------|---------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Setting state of Light.state (ffe.livingroom.floor_light) to True                                             |
| <b>Success</b> | Value for ViDevCommon.state (ffe.livingroom.floorlamp) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Light.state (ffe.livingroom.floor_light) to False                                            |
| <b>Success</b> | Value for ViDevCommon.state (ffe.livingroom.floorlamp) is correct (Content False and Type is <class 'bool'>). |

---

### 3.1.6 Shelly.relay/0 (ffe.livingroom.main\_light) → Light.state (ffe.livingroom.floor\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.6!

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:04,422                                                         |
| Finished-Time:   | 2025-08-29 17:57:04,725                                                         |
| Time-Consumption | 0.302s                                                                          |

---

#### Testsummary:

|                |                                                                                                           |
|----------------|-----------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                              |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).              |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffe.livingroom.main_light) to True                                       |
| <b>Success</b> | Value for Light.state (ffe.livingroom.floor_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffe.livingroom.main_light) to False                                      |
| <b>Success</b> | Value for Light.state (ffe.livingroom.floor_light) is correct (Content False and Type is <class 'bool'>). |

---

### 3.1.7 ViDevCommon.state (ffe.livingroom.xmas\_tree) → Powerplug1P.state (ffe.livingroom.xmas-tree)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.7!

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:04,725                                                         |
| Finished-Time:   | 2025-08-29 17:57:05,027                                                         |
| Time-Consumption | 0.302s                                                                          |

---

#### Testsummary:

|                |                                                                                                               |
|----------------|---------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                  |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                  |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffe.livingroom.xmas_tree) to True                                         |
| <b>Success</b> | Value for Powerplug1P.state (ffe.livingroom.xmas-tree) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffe.livingroom.xmas_tree) to False                                        |
| <b>Success</b> | Value for Powerplug1P.state (ffe.livingroom.xmas-tree) is correct (Content False and Type is <class 'bool'>). |

---

### 3.1.8 Powerplug1P.state (ffe.livingroom.xmas-tree) → ViDevCommon.state (ffe.livingroom.xmas\_tree)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.8!

|                     |                                                                                                               |
|---------------------|---------------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                                  |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                                |
| Start-Time:         | 2025-08-29 17:57:05,027                                                                                       |
| Finished-Time:      | 2025-08-29 17:57:05,330                                                                                       |
| Time-Consumption    | 0.303s                                                                                                        |
| <b>Testsummary:</b> |                                                                                                               |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                                  |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                  |
| <b>Info</b>         | Setting state of Powerplug1P.state (ffe.livingroom.xmas-tree) to True                                         |
| <b>Success</b>      | Value for ViDevCommon.state (ffe.livingroom.xmas_tree) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of Powerplug1P.state (ffe.livingroom.xmas-tree) to False                                        |
| <b>Success</b>      | Value for ViDevCommon.state (ffe.livingroom.xmas_tree) is correct (Content False and Type is <class 'bool'>). |

### 3.1.9 ViDevCommon.brightness (ffe.livingroom.main\_light) → Light.brightness (ffe.livingroom.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.9!

|                     |                                                                                                           |
|---------------------|-----------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                              |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                            |
| Start-Time:         | 2025-08-29 17:57:05,330                                                                                   |
| Finished-Time:      | 2025-08-29 17:57:06,137                                                                                   |
| Time-Consumption    | 0.807s                                                                                                    |
| <b>Testsummary:</b> |                                                                                                           |
| <b>Info</b>         | Prepare: Switching on device                                                                              |
| <b>Info</b>         | Prepare: Setting devices to last state 100                                                                |
| <b>Success</b>      | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).                  |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (ffe.livingroom.main_light) to 0                                  |
| <b>Success</b>      | Value for Light.brightness (ffe.livingroom.main_light) is correct (Content 0 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (ffe.livingroom.main_light) to 20                                 |
| <b>Success</b>      | Value for Light.brightness (ffe.livingroom.main_light) is correct (Content 20 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (ffe.livingroom.main_light) to 40                                 |
| <b>Success</b>      | Value for Light.brightness (ffe.livingroom.main_light) is correct (Content 40 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (ffe.livingroom.main_light) to 60                                 |
| <b>Success</b>      | Value for Light.brightness (ffe.livingroom.main_light) is correct (Content 60 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (ffe.livingroom.main_light) to 80                                 |

**Success** Value for Light.brightness (ffe.livingroom.main\_light) is correct (Content 80 and Type is <class 'int'>).

**Info** Setting state of ViDevCommon.brightness (ffe.livingroom.main\_light) to 100

**Success** Value for Light.brightness (ffe.livingroom.main\_light) is correct (Content 100 and Type is <class 'int'>).

---

### 3.1.10 Light.brightness (ffe.livingroom.main\_light) → ViDevCommon.brightness (ffe.livingroom.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.10!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:06,137                                                         |
| Finished-Time:   | 2025-08-29 17:57:06,944                                                         |
| Time-Consumption | 0.807s                                                                          |

---

#### Testsummary:

---

|                |                                                                                                                  |
|----------------|------------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                                     |
| <b>Info</b>    | Prepare: Setting devices to last state 100                                                                       |
| <b>Success</b> | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).                         |
| <b>Info</b>    | Setting state of Light.brightness (ffe.livingroom.main_light) to 0                                               |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.livingroom.main_light) is correct (Content 0 and Type is <class 'int'>).   |
| <b>Info</b>    | Setting state of Light.brightness (ffe.livingroom.main_light) to 20                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.livingroom.main_light) is correct (Content 20 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffe.livingroom.main_light) to 40                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.livingroom.main_light) is correct (Content 40 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffe.livingroom.main_light) to 60                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.livingroom.main_light) is correct (Content 60 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffe.livingroom.main_light) to 80                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.livingroom.main_light) is correct (Content 80 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffe.livingroom.main_light) to 100                                             |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.livingroom.main_light) is correct (Content 100 and Type is <class 'int'>). |

---

### 3.1.11 ViDevCommon.color\_temp (ffe.livingroom.main\_light) → Light.color\_temp (ffe.livingroom.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.11!

---

|             |                                                                                 |
|-------------|---------------------------------------------------------------------------------|
| Testrun:    | python3.13.5                                                                    |
| Caller:     | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time: | 2025-08-29 17:57:06,944                                                         |

---

Finished-Time: 2025-08-29 17:57:07,750

Time-Consumption 0.806s

**Testsummary:**

|                |                                                                                                           |
|----------------|-----------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                              |
| <b>Info</b>    | Prepare: Setting devices to last state 10                                                                 |
| <b>Success</b> | Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).                    |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffe.livingroom.main_light) to 0                                  |
| <b>Success</b> | Value for Light.color_temp (ffe.livingroom.main_light) is correct (Content 0 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffe.livingroom.main_light) to 2                                  |
| <b>Success</b> | Value for Light.color_temp (ffe.livingroom.main_light) is correct (Content 2 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffe.livingroom.main_light) to 4                                  |
| <b>Success</b> | Value for Light.color_temp (ffe.livingroom.main_light) is correct (Content 4 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffe.livingroom.main_light) to 6                                  |
| <b>Success</b> | Value for Light.color_temp (ffe.livingroom.main_light) is correct (Content 6 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffe.livingroom.main_light) to 8                                  |
| <b>Success</b> | Value for Light.color_temp (ffe.livingroom.main_light) is correct (Content 8 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffe.livingroom.main_light) to 10                                 |
| <b>Success</b> | Value for Light.color_temp (ffe.livingroom.main_light) is correct (Content 10 and Type is <class 'int'>). |

**3.1.12 Light.color\_temp (ffe.livingroom.main\_light) → ViDevCommon.color\_temp (ffe.livingroom.main\_light)****Testresult**This test was passed with the state: **Success**. See also full trace in section A.1.12!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:07,751                                                        |
| Finished-Time:   | 2025-08-29 17:57:08,557                                                        |
| Time-Consumption | 0.807s                                                                         |

**Testsummary:**

|                |                                                                                                                |
|----------------|----------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                                   |
| <b>Info</b>    | Prepare: Setting devices to last state 10                                                                      |
| <b>Success</b> | Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).                         |
| <b>Info</b>    | Setting state of Light.color_temp (ffe.livingroom.main_light) to 0                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffe.livingroom.main_light) is correct (Content 0 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of Light.color_temp (ffe.livingroom.main_light) to 2                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffe.livingroom.main_light) is correct (Content 2 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of Light.color_temp (ffe.livingroom.main_light) to 4                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffe.livingroom.main_light) is correct (Content 4 and Type is <class 'int'>). |

|                |                                                                                                                 |
|----------------|-----------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Setting state of Light.color_temp (ffe.livingroom.main_light) to 6                                              |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffe.livingroom.main_light) is correct (Content 6 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.color_temp (ffe.livingroom.main_light) to 8                                              |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffe.livingroom.main_light) is correct (Content 8 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.color_temp (ffe.livingroom.main_light) to 10                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffe.livingroom.main_light) is correct (Content 10 and Type is <class 'int'>). |

### 3.1.13 ViDevCommon.brightness (ffe.livingroom.floorlamp) → Light.brightness (ffe.livingroom.floor\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.13!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:08,558                                                        |
| Finished-Time:   | 2025-08-29 17:57:09,365                                                        |
| Time-Consumption | 0.807s                                                                         |

#### Testsummary:

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                                |
| <b>Info</b>    | Prepare: Setting devices to last state 100                                                                  |
| <b>Success</b> | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).                    |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffe.livingroom.floorlamp) to 0                                     |
| <b>Success</b> | Value for Light.brightness (ffe.livingroom.floor_light) is correct (Content 0 and Type is <class 'int'>).   |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffe.livingroom.floorlamp) to 20                                    |
| <b>Success</b> | Value for Light.brightness (ffe.livingroom.floor_light) is correct (Content 20 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffe.livingroom.floorlamp) to 40                                    |
| <b>Success</b> | Value for Light.brightness (ffe.livingroom.floor_light) is correct (Content 40 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffe.livingroom.floorlamp) to 60                                    |
| <b>Success</b> | Value for Light.brightness (ffe.livingroom.floor_light) is correct (Content 60 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffe.livingroom.floorlamp) to 80                                    |
| <b>Success</b> | Value for Light.brightness (ffe.livingroom.floor_light) is correct (Content 80 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffe.livingroom.floorlamp) to 100                                   |
| <b>Success</b> | Value for Light.brightness (ffe.livingroom.floor_light) is correct (Content 100 and Type is <class 'int'>). |

### 3.1.14 Light.brightness (ffe.livingroom.floor\_light) → ViDevCommon.brightness (ffe.livingroom.floorlamp)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.14!

|                     |                                                                                                                 |
|---------------------|-----------------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                                    |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329)                                 |
| Start-Time:         | 2025-08-29 17:57:09,365                                                                                         |
| Finished-Time:      | 2025-08-29 17:57:10,175                                                                                         |
| Time-Consumption    | 0.810s                                                                                                          |
| <b>Testsummary:</b> |                                                                                                                 |
| <b>Info</b>         | Prepare: Switching on device                                                                                    |
| <b>Info</b>         | Prepare: Setting devices to last state 100                                                                      |
| <b>Success</b>      | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).                        |
| <b>Info</b>         | Setting state of Light.brightness (ffe.livingroom.floor_light) to 0                                             |
| <b>Success</b>      | Value for ViDevCommon.brightness (ffe.livingroom.floorlamp) is correct (Content 0 and Type is <class 'int'>).   |
| <b>Info</b>         | Setting state of Light.brightness (ffe.livingroom.floor_light) to 20                                            |
| <b>Success</b>      | Value for ViDevCommon.brightness (ffe.livingroom.floorlamp) is correct (Content 20 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of Light.brightness (ffe.livingroom.floor_light) to 40                                            |
| <b>Success</b>      | Value for ViDevCommon.brightness (ffe.livingroom.floorlamp) is correct (Content 40 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of Light.brightness (ffe.livingroom.floor_light) to 60                                            |
| <b>Success</b>      | Value for ViDevCommon.brightness (ffe.livingroom.floorlamp) is correct (Content 60 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of Light.brightness (ffe.livingroom.floor_light) to 80                                            |
| <b>Success</b>      | Value for ViDevCommon.brightness (ffe.livingroom.floorlamp) is correct (Content 80 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of Light.brightness (ffe.livingroom.floor_light) to 100                                           |
| <b>Success</b>      | Value for ViDevCommon.brightness (ffe.livingroom.floorlamp) is correct (Content 100 and Type is <class 'int'>). |

### 3.1.15 ViDevCommon.color\_temp (ffe.livingroom.floorlamp) → Light.color\_temp (ffe.livingroom.floor\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.15!

|                     |                                                                                                           |
|---------------------|-----------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                              |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329)                           |
| Start-Time:         | 2025-08-29 17:57:10,176                                                                                   |
| Finished-Time:      | 2025-08-29 17:57:10,983                                                                                   |
| Time-Consumption    | 0.808s                                                                                                    |
| <b>Testsummary:</b> |                                                                                                           |
| <b>Info</b>         | Prepare: Switching on device                                                                              |
| <b>Info</b>         | Prepare: Setting devices to last state 10                                                                 |
| <b>Success</b>      | Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).                    |
| <b>Info</b>         | Setting state of ViDevCommon.color_temp (ffe.livingroom.floorlamp) to 0                                   |
| <b>Success</b>      | Value for Light.color_temp (ffe.livingroom.floor_light) is correct (Content 0 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevCommon.color_temp (ffe.livingroom.floorlamp) to 2                                   |

|                |                                                                                                            |
|----------------|------------------------------------------------------------------------------------------------------------|
| <b>Success</b> | Value for Light.color_temp (ffe.livingroom.floor_light) is correct (Content 2 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffe.livingroom.floorlamp) to 4                                    |
| <b>Success</b> | Value for Light.color_temp (ffe.livingroom.floor_light) is correct (Content 4 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffe.livingroom.floorlamp) to 6                                    |
| <b>Success</b> | Value for Light.color_temp (ffe.livingroom.floor_light) is correct (Content 6 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffe.livingroom.floorlamp) to 8                                    |
| <b>Success</b> | Value for Light.color_temp (ffe.livingroom.floor_light) is correct (Content 8 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffe.livingroom.floorlamp) to 10                                   |
| <b>Success</b> | Value for Light.color_temp (ffe.livingroom.floor_light) is correct (Content 10 and Type is <class 'int'>). |

### 3.1.16 Light.color\_temp (ffe.livingroom.floor\_light) → ViDevCommon.color\_temp (ffe.livingroom.floorlamp)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.16!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:10,983                                                        |
| Finished-Time:   | 2025-08-29 17:57:11,794                                                        |
| Time-Consumption | 0.810s                                                                         |

#### Testsummary:

|                |                                                                                                                |
|----------------|----------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                                   |
| <b>Info</b>    | Prepare: Setting devices to last state 10                                                                      |
| <b>Success</b> | Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).                         |
| <b>Info</b>    | Setting state of Light.color_temp (ffe.livingroom.floor_light) to 0                                            |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffe.livingroom.floorlamp) is correct (Content 0 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.color_temp (ffe.livingroom.floor_light) to 2                                            |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffe.livingroom.floorlamp) is correct (Content 2 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.color_temp (ffe.livingroom.floor_light) to 4                                            |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffe.livingroom.floorlamp) is correct (Content 4 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.color_temp (ffe.livingroom.floor_light) to 6                                            |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffe.livingroom.floorlamp) is correct (Content 6 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.color_temp (ffe.livingroom.floor_light) to 8                                            |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffe.livingroom.floorlamp) is correct (Content 8 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.color_temp (ffe.livingroom.floor_light) to 10                                           |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffe.livingroom.floorlamp) is correct (Content 10 and Type is <class 'int'>). |

**3.1.17 ViDevHeating.temp\_setp (ffe.livingroom.heating\_valve) → HeatingValve.temp\_setp (ffe.livingroom.heating\_valve)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.17!

|                     |                                                                                                                    |
|---------------------|--------------------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                                       |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                                     |
| Start-Time:         | 2025-08-29 17:57:11,794                                                                                            |
| Finished-Time:      | 2025-08-29 17:57:12,298                                                                                            |
| Time-Consumption    | 0.504s                                                                                                             |
| <b>Testsummary:</b> |                                                                                                                    |
| <b>Info</b>         | Prepare: Setting devices to last state 30                                                                          |
| <b>Success</b>      | Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).                             |
| <b>Info</b>         | Setting state of ViDevHeating.temp_setp (ffe.livingroom.heating_valve) to 15                                       |
| <b>Success</b>      | Value for HeatingValve.temp_setp (ffe.livingroom.heating_valve) is correct (Content 15 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevHeating.temp_setp (ffe.livingroom.heating_valve) to 20                                       |
| <b>Success</b>      | Value for HeatingValve.temp_setp (ffe.livingroom.heating_valve) is correct (Content 20 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevHeating.temp_setp (ffe.livingroom.heating_valve) to 25                                       |
| <b>Success</b>      | Value for HeatingValve.temp_setp (ffe.livingroom.heating_valve) is correct (Content 25 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevHeating.temp_setp (ffe.livingroom.heating_valve) to 30                                       |
| <b>Success</b>      | Value for HeatingValve.temp_setp (ffe.livingroom.heating_valve) is correct (Content 30 and Type is <class 'int'>). |

**3.1.18 ViDevCommon.state (ffe.sleep.main\_light) → Shelly.relay/0 (ffe.sleep.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.18!

|                     |                                                                                                        |
|---------------------|--------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                           |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                         |
| Start-Time:         | 2025-08-29 17:57:12,298                                                                                |
| Finished-Time:      | 2025-08-29 17:57:12,600                                                                                |
| Time-Consumption    | 0.302s                                                                                                 |
| <b>Testsummary:</b> |                                                                                                        |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                           |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).           |
| <b>Info</b>         | Setting state of ViDevCommon.state (ffe.sleep.main_light) to True                                      |
| <b>Success</b>      | Value for Shelly.relay/0 (ffe.sleep.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.state (ffe.sleep.main_light) to False                                     |
| <b>Success</b>      | Value for Shelly.relay/0 (ffe.sleep.main_light) is correct (Content False and Type is <class 'bool'>). |

**3.1.19 Shelly.relay/0 (ffe.sleep.main\_light) → ViDevCommon.state (ffe.sleep.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.19!

|                     |                                                                                                           |
|---------------------|-----------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                              |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                            |
| Start-Time:         | 2025-08-29 17:57:12,601                                                                                   |
| Finished-Time:      | 2025-08-29 17:57:12,903                                                                                   |
| Time-Consumption    | 0.302s                                                                                                    |
| <b>Testsummary:</b> |                                                                                                           |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                              |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).              |
| <b>Info</b>         | Setting state of Shelly.relay/0 (ffe.sleep.main_light) to True                                            |
| <b>Success</b>      | Value for ViDevCommon.state (ffe.sleep.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of Shelly.relay/0 (ffe.sleep.main_light) to False                                           |
| <b>Success</b>      | Value for ViDevCommon.state (ffe.sleep.main_light) is correct (Content False and Type is <class 'bool'>). |

**3.1.20 ViDevCommon.state (ffe.sleep.bed\_light\_di) → Light.state (ffe.sleep.bed\_light\_di)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.20!

|                     |                                                                                                       |
|---------------------|-------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                          |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                        |
| Start-Time:         | 2025-08-29 17:57:12,903                                                                               |
| Finished-Time:      | 2025-08-29 17:57:13,205                                                                               |
| Time-Consumption    | 0.302s                                                                                                |
| <b>Testsummary:</b> |                                                                                                       |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                          |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).          |
| <b>Info</b>         | Setting state of ViDevCommon.state (ffe.sleep.bed_light_di) to True                                   |
| <b>Success</b>      | Value for Light.state (ffe.sleep.bed_light_di) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.state (ffe.sleep.bed_light_di) to False                                  |
| <b>Success</b>      | Value for Light.state (ffe.sleep.bed_light_di) is correct (Content False and Type is <class 'bool'>). |

**3.1.21 Light.state (ffe.sleep.bed\_light\_di) → ViDevCommon.state (ffe.sleep.bed\_light\_di)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.21!

---

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:13,206                                                        |
| Finished-Time:   | 2025-08-29 17:57:13,508                                                        |
| Time-Consumption | 0.302s                                                                         |

---

**Testsummary:**


---

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                |
| <b>Info</b>    | Setting state of Light.state (ffe.sleep.bed_light_di) to True                                               |
| <b>Success</b> | Value for ViDevCommon.state (ffe.sleep.bed_light_di) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Light.state (ffe.sleep.bed_light_di) to False                                              |
| <b>Success</b> | Value for ViDevCommon.state (ffe.sleep.bed_light_di) is correct (Content False and Type is <class 'bool'>). |

---

**3.1.22 ViDevCommon.state (ffe.sleep.bed\_light\_ma) → Powerplug1P.state (ffe.sleep.bed\_light\_ma)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.22!

---

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:13,508                                                        |
| Finished-Time:   | 2025-08-29 17:57:13,811                                                        |
| Time-Consumption | 0.302s                                                                         |

---

**Testsummary:**


---

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffe.sleep.bed_light_ma) to True                                         |
| <b>Success</b> | Value for Powerplug1P.state (ffe.sleep.bed_light_ma) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffe.sleep.bed_light_ma) to False                                        |
| <b>Success</b> | Value for Powerplug1P.state (ffe.sleep.bed_light_ma) is correct (Content False and Type is <class 'bool'>). |

---

**3.1.23 Powerplug1P.state (ffe.sleep.bed\_light\_ma) → ViDevCommon.state (ffe.sleep.bed\_light\_ma)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.23!

---

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:13,811                                                        |
| Finished-Time:   | 2025-08-29 17:57:14,113                                                        |
| Time-Consumption | 0.302s                                                                         |

---

**Testsummary:**


---

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                |
| <b>Info</b>    | Setting state of Powerplug1P.state (ffe.sleep.bed_light_ma) to True                                         |
| <b>Success</b> | Value for ViDevCommon.state (ffe.sleep.bed_light_ma) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Powerplug1P.state (ffe.sleep.bed_light_ma) to False                                        |
| <b>Success</b> | Value for ViDevCommon.state (ffe.sleep.bed_light_ma) is correct (Content False and Type is <class 'bool'>). |

---

**3.1.24 ViDevCommon.brightness (ffe.sleep.main\_light) → Light.brightness (ffe.sleep.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.24!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:14,113                                                         |
| Finished-Time:   | 2025-08-29 17:57:14,919                                                         |
| Time-Consumption | 0.806s                                                                          |

---

**Testsummary:**


---

|                |                                                                                                       |
|----------------|-------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                          |
| <b>Info</b>    | Prepare: Setting devices to last state 100                                                            |
| <b>Success</b> | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).              |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffe.sleep.main_light) to 0                                   |
| <b>Success</b> | Value for Light.brightness (ffe.sleep.main_light) is correct (Content 0 and Type is <class 'int'>).   |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffe.sleep.main_light) to 20                                  |
| <b>Success</b> | Value for Light.brightness (ffe.sleep.main_light) is correct (Content 20 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffe.sleep.main_light) to 40                                  |
| <b>Success</b> | Value for Light.brightness (ffe.sleep.main_light) is correct (Content 40 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffe.sleep.main_light) to 60                                  |
| <b>Success</b> | Value for Light.brightness (ffe.sleep.main_light) is correct (Content 60 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffe.sleep.main_light) to 80                                  |
| <b>Success</b> | Value for Light.brightness (ffe.sleep.main_light) is correct (Content 80 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffe.sleep.main_light) to 100                                 |
| <b>Success</b> | Value for Light.brightness (ffe.sleep.main_light) is correct (Content 100 and Type is <class 'int'>). |

---

**3.1.25 Light.brightness (ffe.sleep.main\_light) → ViDevCommon.brightness (ffe.sleep.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.25!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:14,920                                                        |
| Finished-Time:   | 2025-08-29 17:57:15,725                                                        |
| Time-Consumption | 0.806s                                                                         |

**Testsummary:**

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                                |
| <b>Info</b>    | Prepare: Setting devices to last state 100                                                                  |
| <b>Success</b> | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).                    |
| <b>Info</b>    | Setting state of Light.brightness (ffe.sleep.main_light) to 0                                               |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.sleep.main_light) is correct (Content 0 and Type is <class 'int'>).   |
| <b>Info</b>    | Setting state of Light.brightness (ffe.sleep.main_light) to 20                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.sleep.main_light) is correct (Content 20 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffe.sleep.main_light) to 40                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.sleep.main_light) is correct (Content 40 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffe.sleep.main_light) to 60                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.sleep.main_light) is correct (Content 60 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffe.sleep.main_light) to 80                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.sleep.main_light) is correct (Content 80 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffe.sleep.main_light) to 100                                             |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.sleep.main_light) is correct (Content 100 and Type is <class 'int'>). |

**3.1.26 ViDevCommon.color\_temp (ffe.sleep.main\_light) → Light.color\_temp (ffe.sleep.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.26!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:15,726                                                        |
| Finished-Time:   | 2025-08-29 17:57:16,531                                                        |
| Time-Consumption | 0.806s                                                                         |

**Testsummary:**

|                |                                                                                                     |
|----------------|-----------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                        |
| <b>Info</b>    | Prepare: Setting devices to last state 10                                                           |
| <b>Success</b> | Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).              |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffe.sleep.main_light) to 0                                 |
| <b>Success</b> | Value for Light.color_temp (ffe.sleep.main_light) is correct (Content 0 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffe.sleep.main_light) to 2                                 |

**Success** Value for Light.color\_temp (ffe.sleep.main\_light) is correct (Content 2 and Type is <class 'int'>).

**Info** Setting state of ViDevCommon.color\_temp (ffe.sleep.main\_light) to 4

**Success** Value for Light.color\_temp (ffe.sleep.main\_light) is correct (Content 4 and Type is <class 'int'>).

**Info** Setting state of ViDevCommon.color\_temp (ffe.sleep.main\_light) to 6

**Success** Value for Light.color\_temp (ffe.sleep.main\_light) is correct (Content 6 and Type is <class 'int'>).

**Info** Setting state of ViDevCommon.color\_temp (ffe.sleep.main\_light) to 8

**Success** Value for Light.color\_temp (ffe.sleep.main\_light) is correct (Content 8 and Type is <class 'int'>).

**Info** Setting state of ViDevCommon.color\_temp (ffe.sleep.main\_light) to 10

**Success** Value for Light.color\_temp (ffe.sleep.main\_light) is correct (Content 10 and Type is <class 'int'>).

---

### 3.1.27 Light.color\_temp (ffe.sleep.main\_light) → ViDevCommon.color\_temp (ffe.sleep.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.27!

---

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:16,532                                                        |
| Finished-Time:   | 2025-08-29 17:57:17,338                                                        |
| Time-Consumption | 0.806s                                                                         |

---

#### Testsummary:

---

**Info** Prepare: Switching on device

**Info** Prepare: Setting devices to last state 10

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

**Info** Setting state of Light.color\_temp (ffe.sleep.main\_light) to 0

**Success** Value for ViDevCommon.color\_temp (ffe.sleep.main\_light) is correct (Content 0 and Type is <class 'int'>).

**Info** Setting state of Light.color\_temp (ffe.sleep.main\_light) to 2

**Success** Value for ViDevCommon.color\_temp (ffe.sleep.main\_light) is correct (Content 2 and Type is <class 'int'>).

**Info** Setting state of Light.color\_temp (ffe.sleep.main\_light) to 4

**Success** Value for ViDevCommon.color\_temp (ffe.sleep.main\_light) is correct (Content 4 and Type is <class 'int'>).

**Info** Setting state of Light.color\_temp (ffe.sleep.main\_light) to 6

**Success** Value for ViDevCommon.color\_temp (ffe.sleep.main\_light) is correct (Content 6 and Type is <class 'int'>).

**Info** Setting state of Light.color\_temp (ffe.sleep.main\_light) to 8

**Success** Value for ViDevCommon.color\_temp (ffe.sleep.main\_light) is correct (Content 8 and Type is <class 'int'>).

**Info** Setting state of Light.color\_temp (ffe.sleep.main\_light) to 10

**Success** Value for ViDevCommon.color\_temp (ffe.sleep.main\_light) is correct (Content 10 and Type is <class 'int'>).

---

**3.1.28 ViDevCommon.brightness (ffe.sleep.bed\_light\_di) → Light.brightness (ffe.sleep.bed\_light\_di)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.28!

|                     |                                                                                                         |
|---------------------|---------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                            |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                          |
| Start-Time:         | 2025-08-29 17:57:17,338                                                                                 |
| Finished-Time:      | 2025-08-29 17:57:18,145                                                                                 |
| Time-Consumption    | 0.807s                                                                                                  |
| <b>Testsummary:</b> |                                                                                                         |
| <b>Info</b>         | Prepare: Switching on device                                                                            |
| <b>Info</b>         | Prepare: Setting devices to last state 100                                                              |
| <b>Success</b>      | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).                |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (ffe.sleep.bed_light_di) to 0                                   |
| <b>Success</b>      | Value for Light.brightness (ffe.sleep.bed_light_di) is correct (Content 0 and Type is <class 'int'>).   |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (ffe.sleep.bed_light_di) to 20                                  |
| <b>Success</b>      | Value for Light.brightness (ffe.sleep.bed_light_di) is correct (Content 20 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (ffe.sleep.bed_light_di) to 40                                  |
| <b>Success</b>      | Value for Light.brightness (ffe.sleep.bed_light_di) is correct (Content 40 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (ffe.sleep.bed_light_di) to 60                                  |
| <b>Success</b>      | Value for Light.brightness (ffe.sleep.bed_light_di) is correct (Content 60 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (ffe.sleep.bed_light_di) to 80                                  |
| <b>Success</b>      | Value for Light.brightness (ffe.sleep.bed_light_di) is correct (Content 80 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (ffe.sleep.bed_light_di) to 100                                 |
| <b>Success</b>      | Value for Light.brightness (ffe.sleep.bed_light_di) is correct (Content 100 and Type is <class 'int'>). |

**3.1.29 Light.brightness (ffe.sleep.bed\_light\_di) → ViDevCommon.brightness (ffe.sleep.bed\_light\_di)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.29!

|                     |                                                                                          |
|---------------------|------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                             |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)           |
| Start-Time:         | 2025-08-29 17:57:18,145                                                                  |
| Finished-Time:      | 2025-08-29 17:57:18,951                                                                  |
| Time-Consumption    | 0.806s                                                                                   |
| <b>Testsummary:</b> |                                                                                          |
| <b>Info</b>         | Prepare: Switching on device                                                             |
| <b>Info</b>         | Prepare: Setting devices to last state 100                                               |
| <b>Success</b>      | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>). |

|                |                                                                                                               |
|----------------|---------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Setting state of Light.brightness (ffe.sleep.bed_light_di) to 0                                               |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.sleep.bed_light_di) is correct (Content 0 and Type is <class 'int'>).   |
| <b>Info</b>    | Setting state of Light.brightness (ffe.sleep.bed_light_di) to 20                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.sleep.bed_light_di) is correct (Content 20 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffe.sleep.bed_light_di) to 40                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.sleep.bed_light_di) is correct (Content 40 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffe.sleep.bed_light_di) to 60                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.sleep.bed_light_di) is correct (Content 60 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffe.sleep.bed_light_di) to 80                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.sleep.bed_light_di) is correct (Content 80 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffe.sleep.bed_light_di) to 100                                             |
| <b>Success</b> | Value for ViDevCommon.brightness (ffe.sleep.bed_light_di) is correct (Content 100 and Type is <class 'int'>). |

---

### 3.1.30 ViDevHeating.temp\_setp (ffe.sleep.heating\_valve) → HeatingValve.temp\_setp (ffe.sleep.heating\_valve)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.30!

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:18,951                                                         |
| Finished-Time:   | 2025-08-29 17:57:19,455                                                         |
| Time-Consumption | 0.504s                                                                          |

---

#### Testsummary:

|                |                                                                                                               |
|----------------|---------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state 30                                                                     |
| <b>Success</b> | Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).                        |
| <b>Info</b>    | Setting state of ViDevHeating.temp_setp (ffe.sleep.heating_valve) to 15                                       |
| <b>Success</b> | Value for HeatingValve.temp_setp (ffe.sleep.heating_valve) is correct (Content 15 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevHeating.temp_setp (ffe.sleep.heating_valve) to 20                                       |
| <b>Success</b> | Value for HeatingValve.temp_setp (ffe.sleep.heating_valve) is correct (Content 20 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevHeating.temp_setp (ffe.sleep.heating_valve) to 25                                       |
| <b>Success</b> | Value for HeatingValve.temp_setp (ffe.sleep.heating_valve) is correct (Content 25 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevHeating.temp_setp (ffe.sleep.heating_valve) to 30                                       |
| <b>Success</b> | Value for HeatingValve.temp_setp (ffe.sleep.heating_valve) is correct (Content 30 and Type is <class 'int'>). |

---

**3.1.31 ViDevCommon.state (ffe.diningroom.main\_light) → Shelly.relay/0 (ffe.diningroom.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.31!

|                     |                                                                                                             |
|---------------------|-------------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                                |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                              |
| Start-Time:         | 2025-08-29 17:57:19,455                                                                                     |
| Finished-Time:      | 2025-08-29 17:57:19,757                                                                                     |
| Time-Consumption    | 0.302s                                                                                                      |
| <b>Testsummary:</b> |                                                                                                             |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                                |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                |
| <b>Info</b>         | Setting state of ViDevCommon.state (ffe.diningroom.main_light) to True                                      |
| <b>Success</b>      | Value for Shelly.relay/0 (ffe.diningroom.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.state (ffe.diningroom.main_light) to False                                     |
| <b>Success</b>      | Value for Shelly.relay/0 (ffe.diningroom.main_light) is correct (Content False and Type is <class 'bool'>). |

**3.1.32 Shelly.relay/0 (ffe.diningroom.main\_light) → ViDevCommon.state (ffe.diningroom.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.32!

|                     |                                                                                                                |
|---------------------|----------------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                                   |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                                 |
| Start-Time:         | 2025-08-29 17:57:19,758                                                                                        |
| Finished-Time:      | 2025-08-29 17:57:20,060                                                                                        |
| Time-Consumption    | 0.302s                                                                                                         |
| <b>Testsummary:</b> |                                                                                                                |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                                   |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                   |
| <b>Info</b>         | Setting state of Shelly.relay/0 (ffe.diningroom.main_light) to True                                            |
| <b>Success</b>      | Value for ViDevCommon.state (ffe.diningroom.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of Shelly.relay/0 (ffe.diningroom.main_light) to False                                           |
| <b>Success</b>      | Value for ViDevCommon.state (ffe.diningroom.main_light) is correct (Content False and Type is <class 'bool'>). |

**3.1.33 ViDevCommon.state (ffe.diningroom.floorlamp) → Powerplug1P.state (ffe.diningroom.floor\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.33!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:20,060                                                         |
| Finished-Time:   | 2025-08-29 17:57:20,362                                                         |
| Time-Consumption | 0.302s                                                                          |

---

**Testsummary:**


---

|                |                                                                                                                 |
|----------------|-----------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                    |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                    |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffe.diningroom.floorlamp) to True                                           |
| <b>Success</b> | Value for Powerplug1P.state (ffe.diningroom.floor_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffe.diningroom.floorlamp) to False                                          |
| <b>Success</b> | Value for Powerplug1P.state (ffe.diningroom.floor_light) is correct (Content False and Type is <class 'bool'>). |

---

**3.1.34 Powerplug1P.state (ffe.diningroom.floor\_light) → ViDevCommon.state (ffe.diningroom.floorlamp)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.34!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:20,362                                                         |
| Finished-Time:   | 2025-08-29 17:57:20,665                                                         |
| Time-Consumption | 0.302s                                                                          |

---

**Testsummary:**


---

|                |                                                                                                               |
|----------------|---------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                  |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                  |
| <b>Info</b>    | Setting state of Powerplug1P.state (ffe.diningroom.floor_light) to True                                       |
| <b>Success</b> | Value for ViDevCommon.state (ffe.diningroom.floorlamp) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Powerplug1P.state (ffe.diningroom.floor_light) to False                                      |
| <b>Success</b> | Value for ViDevCommon.state (ffe.diningroom.floorlamp) is correct (Content False and Type is <class 'bool'>). |

---

**3.1.35 Shelly.relay/0 (ffe.diningroom.main\_light) → Powerplug1P.state (ffe.diningroom.floor\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.35!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:20,665                                                         |
| Finished-Time:   | 2025-08-29 17:57:20,967                                                         |
| Time-Consumption | 0.302s                                                                          |

---

**Testsummary:**

|                |                                                                                                                 |
|----------------|-----------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                    |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                    |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffe.diningroom.main_light) to True                                             |
| <b>Success</b> | Value for Powerplug1P.state (ffe.diningroom.floor_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffe.diningroom.main_light) to False                                            |
| <b>Success</b> | Value for Powerplug1P.state (ffe.diningroom.floor_light) is correct (Content False and Type is <class 'bool'>). |

**3.1.36 ViDevCommon.state (ffe.diningroom.garland) → Powerplug1P.state (ffe.diningroom.garland)**
**Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.36!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:20,967                                                        |
| Finished-Time:   | 2025-08-29 17:57:21,270                                                        |
| Time-Consumption | 0.303s                                                                         |

**Testsummary:**

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffe.diningroom.garland) to True                                         |
| <b>Success</b> | Value for Powerplug1P.state (ffe.diningroom.garland) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffe.diningroom.garland) to False                                        |
| <b>Success</b> | Value for Powerplug1P.state (ffe.diningroom.garland) is correct (Content False and Type is <class 'bool'>). |

**3.1.37 Powerplug1P.state (ffe.diningroom.garland) → ViDevCommon.state (ffe.diningroom.garland)**
**Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.37!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:21,270                                                        |
| Finished-Time:   | 2025-08-29 17:57:21,573                                                        |
| Time-Consumption | 0.303s                                                                         |

**Testsummary:**

|                |                                                                                                            |
|----------------|------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                               |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).               |
| <b>Info</b>    | Setting state of Powerplug1P.state (ffe.diningroom.garland) to True                                        |
| <b>Success</b> | Value for ViDevCommon.state (ffe.diningroom.garland) is correct (Content True and Type is <class 'bool'>). |

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Setting state of Powerplug1P.state (ffe.diningroom.garland) to False                                        |
| <b>Success</b> | Value for ViDevCommon.state (ffe.diningroom.garland) is correct (Content False and Type is <class 'bool'>). |

---

### 3.1.38 ViDevCommon.state (ffe.kitchen.main\_light) → Shelly.relay/0 (ffe.kitchen.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.38!

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:21,573                                                         |
| Finished-Time:   | 2025-08-29 17:57:21,875                                                         |
| Time-Consumption | 0.302s                                                                          |

---

#### Testsummary:

|                |                                                                                                          |
|----------------|----------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                             |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).             |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffe.kitchen.main_light) to True                                      |
| <b>Success</b> | Value for Shelly.relay/0 (ffe.kitchen.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffe.kitchen.main_light) to False                                     |
| <b>Success</b> | Value for Shelly.relay/0 (ffe.kitchen.main_light) is correct (Content False and Type is <class 'bool'>). |

---

### 3.1.39 Shelly.relay/0 (ffe.kitchen.main\_light) → ViDevCommon.state (ffe.kitchen.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.39!

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:21,876                                                         |
| Finished-Time:   | 2025-08-29 17:57:22,178                                                         |
| Time-Consumption | 0.302s                                                                          |

---

#### Testsummary:

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffe.kitchen.main_light) to True                                            |
| <b>Success</b> | Value for ViDevCommon.state (ffe.kitchen.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffe.kitchen.main_light) to False                                           |
| <b>Success</b> | Value for ViDevCommon.state (ffe.kitchen.main_light) is correct (Content False and Type is <class 'bool'>). |

---

**3.1.40 ViDevCommon.state (ffe.kitchen.circulation\_pump) → Shelly.relay/0 (ffe.kitchen.circulation\_pump)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.40!

|                     |                                                                                                                |
|---------------------|----------------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                                   |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329)                                |
| Start-Time:         | 2025-08-29 17:57:22,178                                                                                        |
| Finished-Time:      | 2025-08-29 17:57:22,480                                                                                        |
| Time-Consumption    | 0.302s                                                                                                         |
| <b>Testsummary:</b> |                                                                                                                |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                                   |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                   |
| <b>Info</b>         | Setting state of ViDevCommon.state (ffe.kitchen.circulation_pump) to True                                      |
| <b>Success</b>      | Value for Shelly.relay/0 (ffe.kitchen.circulation_pump) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.state (ffe.kitchen.circulation_pump) to False                                     |
| <b>Success</b>      | Value for Shelly.relay/0 (ffe.kitchen.circulation_pump) is correct (Content False and Type is <class 'bool'>). |

**3.1.41 Shelly.relay/0 (ffe.kitchen.circulation\_pump) → ViDevCommon.state (ffe.kitchen.circulation\_pump)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.41!

|                     |                                                                                                                   |
|---------------------|-------------------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                                      |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329)                                   |
| Start-Time:         | 2025-08-29 17:57:22,481                                                                                           |
| Finished-Time:      | 2025-08-29 17:57:22,783                                                                                           |
| Time-Consumption    | 0.302s                                                                                                            |
| <b>Testsummary:</b> |                                                                                                                   |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                                      |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                      |
| <b>Info</b>         | Setting state of Shelly.relay/0 (ffe.kitchen.circulation_pump) to True                                            |
| <b>Success</b>      | Value for ViDevCommon.state (ffe.kitchen.circulation_pump) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of Shelly.relay/0 (ffe.kitchen.circulation_pump) to False                                           |
| <b>Success</b>      | Value for ViDevCommon.state (ffe.kitchen.circulation_pump) is correct (Content False and Type is <class 'bool'>). |

**3.1.42 ViDevHeating.temp\_setp (ffe.kitchen.heating\_valve) → HeatingValve.temp\_setp (ffe.kitchen.heating\_valve)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.42!

|                     |                                                                                                                 |
|---------------------|-----------------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                                    |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                                  |
| Start-Time:         | 2025-08-29 17:57:22,783                                                                                         |
| Finished-Time:      | 2025-08-29 17:57:23,288                                                                                         |
| Time-Consumption    | 0.505s                                                                                                          |
| <b>Testsummary:</b> |                                                                                                                 |
| <b>Info</b>         | Prepare: Setting devices to last state 30                                                                       |
| <b>Success</b>      | Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).                          |
| <b>Info</b>         | Setting state of ViDevHeating.temp_setp (ffe.kitchen.heating_valve) to 15                                       |
| <b>Success</b>      | Value for HeatingValve.temp_setp (ffe.kitchen.heating_valve) is correct (Content 15 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevHeating.temp_setp (ffe.kitchen.heating_valve) to 20                                       |
| <b>Success</b>      | Value for HeatingValve.temp_setp (ffe.kitchen.heating_valve) is correct (Content 20 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevHeating.temp_setp (ffe.kitchen.heating_valve) to 25                                       |
| <b>Success</b>      | Value for HeatingValve.temp_setp (ffe.kitchen.heating_valve) is correct (Content 25 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevHeating.temp_setp (ffe.kitchen.heating_valve) to 30                                       |
| <b>Success</b>      | Value for HeatingValve.temp_setp (ffe.kitchen.heating_valve) is correct (Content 30 and Type is <class 'int'>). |

### 3.1.43 ViDevCommon.state (ffe.floor.main\_light) → Shelly.relay/0 (ffe.floor.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.43!

|                     |                                                                                                        |
|---------------------|--------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                           |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                         |
| Start-Time:         | 2025-08-29 17:57:23,289                                                                                |
| Finished-Time:      | 2025-08-29 17:57:23,591                                                                                |
| Time-Consumption    | 0.302s                                                                                                 |
| <b>Testsummary:</b> |                                                                                                        |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                           |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).           |
| <b>Info</b>         | Setting state of ViDevCommon.state (ffe.floor.main_light) to True                                      |
| <b>Success</b>      | Value for Shelly.relay/0 (ffe.floor.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.state (ffe.floor.main_light) to False                                     |
| <b>Success</b>      | Value for Shelly.relay/0 (ffe.floor.main_light) is correct (Content False and Type is <class 'bool'>). |

### 3.1.44 Shelly.relay/0 (ffe.floor.main\_light) → ViDevCommon.state (ffe.floor.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.44!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:23,592                                                         |
| Finished-Time:   | 2025-08-29 17:57:23,894                                                         |
| Time-Consumption | 0.303s                                                                          |

---

**Testsummary:**


---

|                |                                                                                                           |
|----------------|-----------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                              |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).              |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffe.floor.main_light) to True                                            |
| <b>Success</b> | Value for ViDevCommon.state (ffe.floor.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffe.floor.main_light) to False                                           |
| <b>Success</b> | Value for ViDevCommon.state (ffe.floor.main_light) is correct (Content False and Type is <class 'bool'>). |

---

**3.1.45 ViDevCommon.state (ffw.livingroom.main\_light) → Shelly.relay/0 (ffw.livingroom.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.45!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:23,894                                                         |
| Finished-Time:   | 2025-08-29 17:57:24,197                                                         |
| Time-Consumption | 0.302s                                                                          |

---

**Testsummary:**


---

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffw.livingroom.main_light) to True                                      |
| <b>Success</b> | Value for Shelly.relay/0 (ffw.livingroom.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffw.livingroom.main_light) to False                                     |
| <b>Success</b> | Value for Shelly.relay/0 (ffw.livingroom.main_light) is correct (Content False and Type is <class 'bool'>). |

---

**3.1.46 Shelly.relay/0 (ffw.livingroom.main\_light) → ViDevCommon.state (ffw.livingroom.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.46!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:24,197                                                         |
| Finished-Time:   | 2025-08-29 17:57:24,500                                                         |
| Time-Consumption | 0.303s                                                                          |

---

**Testsummary:**


---

|                |                                                                                                                |
|----------------|----------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                   |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                   |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffw.livingroom.main_light) to True                                            |
| <b>Success</b> | Value for ViDevCommon.state (ffw.livingroom.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffw.livingroom.main_light) to False                                           |
| <b>Success</b> | Value for ViDevCommon.state (ffw.livingroom.main_light) is correct (Content False and Type is <class 'bool'>). |

---

**3.1.47 ViDevCommon.brightness (ffw.livingroom.main\_light) → Light.brightness (ffw.livingroom.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.47!

---

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:24,500                                                        |
| Finished-Time:   | 2025-08-29 17:57:25,307                                                        |
| Time-Consumption | 0.807s                                                                         |

---

**Testsummary:**


---

|                |                                                                                                            |
|----------------|------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                               |
| <b>Info</b>    | Prepare: Setting devices to last state 100                                                                 |
| <b>Success</b> | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).                   |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffw.livingroom.main_light) to 0                                   |
| <b>Success</b> | Value for Light.brightness (ffw.livingroom.main_light) is correct (Content 0 and Type is <class 'int'>).   |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffw.livingroom.main_light) to 20                                  |
| <b>Success</b> | Value for Light.brightness (ffw.livingroom.main_light) is correct (Content 20 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffw.livingroom.main_light) to 40                                  |
| <b>Success</b> | Value for Light.brightness (ffw.livingroom.main_light) is correct (Content 40 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffw.livingroom.main_light) to 60                                  |
| <b>Success</b> | Value for Light.brightness (ffw.livingroom.main_light) is correct (Content 60 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffw.livingroom.main_light) to 80                                  |
| <b>Success</b> | Value for Light.brightness (ffw.livingroom.main_light) is correct (Content 80 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffw.livingroom.main_light) to 100                                 |
| <b>Success</b> | Value for Light.brightness (ffw.livingroom.main_light) is correct (Content 100 and Type is <class 'int'>). |

---

**3.1.48 Light.brightness (ffw.livingroom.main\_light) → ViDevCommon.brightness (ffw.livingroom.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.48!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:25,308                                                         |
| Finished-Time:   | 2025-08-29 17:57:26,114                                                         |
| Time-Consumption | 0.807s                                                                          |

---

**Testsummary:**


---

|                |                                                                                                                  |
|----------------|------------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                                     |
| <b>Info</b>    | Prepare: Setting devices to last state 100                                                                       |
| <b>Success</b> | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).                         |
| <b>Info</b>    | Setting state of Light.brightness (ffw.livingroom.main_light) to 0                                               |
| <b>Success</b> | Value for ViDevCommon.brightness (ffw.livingroom.main_light) is correct (Content 0 and Type is <class 'int'>).   |
| <b>Info</b>    | Setting state of Light.brightness (ffw.livingroom.main_light) to 20                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffw.livingroom.main_light) is correct (Content 20 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffw.livingroom.main_light) to 40                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffw.livingroom.main_light) is correct (Content 40 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffw.livingroom.main_light) to 60                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffw.livingroom.main_light) is correct (Content 60 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffw.livingroom.main_light) to 80                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffw.livingroom.main_light) is correct (Content 80 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffw.livingroom.main_light) to 100                                             |
| <b>Success</b> | Value for ViDevCommon.brightness (ffw.livingroom.main_light) is correct (Content 100 and Type is <class 'int'>). |

---

**3.1.49 ViDevCommon.color\_temp (ffw.livingroom.main\_light) → Light.color\_temp (ffw.livingroom.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.49!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:26,114                                                         |
| Finished-Time:   | 2025-08-29 17:57:26,921                                                         |
| Time-Consumption | 0.807s                                                                          |

---

**Testsummary:**


---

|                |                                                                                                          |
|----------------|----------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                             |
| <b>Info</b>    | Prepare: Setting devices to last state 10                                                                |
| <b>Success</b> | Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).                   |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffw.livingroom.main_light) to 0                                 |
| <b>Success</b> | Value for Light.color_temp (ffw.livingroom.main_light) is correct (Content 0 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffw.livingroom.main_light) to 2                                 |

---

**Success** Value for Light.color\_temp (ffw.livingroom.main\_light) is correct (Content 2 and Type is <class 'int'>).

**Info** Setting state of ViDevCommon.color\_temp (ffw.livingroom.main\_light) to 4

**Success** Value for Light.color\_temp (ffw.livingroom.main\_light) is correct (Content 4 and Type is <class 'int'>).

**Info** Setting state of ViDevCommon.color\_temp (ffw.livingroom.main\_light) to 6

**Success** Value for Light.color\_temp (ffw.livingroom.main\_light) is correct (Content 6 and Type is <class 'int'>).

**Info** Setting state of ViDevCommon.color\_temp (ffw.livingroom.main\_light) to 8

**Success** Value for Light.color\_temp (ffw.livingroom.main\_light) is correct (Content 8 and Type is <class 'int'>).

**Info** Setting state of ViDevCommon.color\_temp (ffw.livingroom.main\_light) to 10

**Success** Value for Light.color\_temp (ffw.livingroom.main\_light) is correct (Content 10 and Type is <class 'int'>).

---

### 3.1.50 Light.color\_temp (ffw.livingroom.main\_light) → ViDevCommon.color\_temp (ffw.livingroom.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.50!

---

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:26,922                                                        |
| Finished-Time:   | 2025-08-29 17:57:27,729                                                        |
| Time-Consumption | 0.807s                                                                         |

---

#### Testsummary:

---

**Info** Prepare: Switching on device

**Info** Prepare: Setting devices to last state 10

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

**Info** Setting state of Light.color\_temp (ffw.livingroom.main\_light) to 0

**Success** Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light) is correct (Content 0 and Type is <class 'int'>).

**Info** Setting state of Light.color\_temp (ffw.livingroom.main\_light) to 2

**Success** Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light) is correct (Content 2 and Type is <class 'int'>).

**Info** Setting state of Light.color\_temp (ffw.livingroom.main\_light) to 4

**Success** Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light) is correct (Content 4 and Type is <class 'int'>).

**Info** Setting state of Light.color\_temp (ffw.livingroom.main\_light) to 6

**Success** Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light) is correct (Content 6 and Type is <class 'int'>).

**Info** Setting state of Light.color\_temp (ffw.livingroom.main\_light) to 8

**Success** Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light) is correct (Content 8 and Type is <class 'int'>).

**Info** Setting state of Light.color\_temp (ffw.livingroom.main\_light) to 10

**Success** Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light) is correct (Content 10 and Type is <class 'int'>).

---

**3.1.51 ViDevHeating.temp\_setp (ffw.livingroom.heating\_valve) → HeatingValve.temp\_setp (ffw.livingroom.heating\_valve)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.51!

|                     |                                                                                                                    |
|---------------------|--------------------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                                       |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329)                                    |
| Start-Time:         | 2025-08-29 17:57:27,729                                                                                            |
| Finished-Time:      | 2025-08-29 17:57:28,233                                                                                            |
| Time-Consumption    | 0.504s                                                                                                             |
| <b>Testsummary:</b> |                                                                                                                    |
| <b>Info</b>         | Prepare: Setting devices to last state 30                                                                          |
| <b>Success</b>      | Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).                             |
| <b>Info</b>         | Setting state of ViDevHeating.temp_setp (ffw.livingroom.heating_valve) to 15                                       |
| <b>Success</b>      | Value for HeatingValve.temp_setp (ffw.livingroom.heating_valve) is correct (Content 15 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevHeating.temp_setp (ffw.livingroom.heating_valve) to 20                                       |
| <b>Success</b>      | Value for HeatingValve.temp_setp (ffw.livingroom.heating_valve) is correct (Content 20 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevHeating.temp_setp (ffw.livingroom.heating_valve) to 25                                       |
| <b>Success</b>      | Value for HeatingValve.temp_setp (ffw.livingroom.heating_valve) is correct (Content 25 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevHeating.temp_setp (ffw.livingroom.heating_valve) to 30                                       |
| <b>Success</b>      | Value for HeatingValve.temp_setp (ffw.livingroom.heating_valve) is correct (Content 30 and Type is <class 'int'>). |

**3.1.52 ViDevCommon.state (ffw.sleep.main\_light) → Shelly.relay/0 (ffw.sleep.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.52!

|                     |                                                                                                        |
|---------------------|--------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                           |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329)                        |
| Start-Time:         | 2025-08-29 17:57:28,233                                                                                |
| Finished-Time:      | 2025-08-29 17:57:28,536                                                                                |
| Time-Consumption    | 0.303s                                                                                                 |
| <b>Testsummary:</b> |                                                                                                        |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                           |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).           |
| <b>Info</b>         | Setting state of ViDevCommon.state (ffw.sleep.main_light) to True                                      |
| <b>Success</b>      | Value for Shelly.relay/0 (ffw.sleep.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.state (ffw.sleep.main_light) to False                                     |
| <b>Success</b>      | Value for Shelly.relay/0 (ffw.sleep.main_light) is correct (Content False and Type is <class 'bool'>). |

**3.1.53 Shelly.relay/0 (ffw.sleep.main\_light) → ViDevCommon.state (ffw.sleep.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.53!

|                     |                                                                                                           |
|---------------------|-----------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                              |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329)                           |
| Start-Time:         | 2025-08-29 17:57:28,536                                                                                   |
| Finished-Time:      | 2025-08-29 17:57:28,839                                                                                   |
| Time-Consumption    | 0.303s                                                                                                    |
| <b>Testsummary:</b> |                                                                                                           |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                              |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).              |
| <b>Info</b>         | Setting state of Shelly.relay/0 (ffw.sleep.main_light) to True                                            |
| <b>Success</b>      | Value for ViDevCommon.state (ffw.sleep.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of Shelly.relay/0 (ffw.sleep.main_light) to False                                           |
| <b>Success</b>      | Value for ViDevCommon.state (ffw.sleep.main_light) is correct (Content False and Type is <class 'bool'>). |

**3.1.54 ViDevCommon.brightness (ffw.sleep.main\_light) → Light.brightness (ffw.sleep.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.54!

|                     |                                                                                                      |
|---------------------|------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                         |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329)                      |
| Start-Time:         | 2025-08-29 17:57:28,839                                                                              |
| Finished-Time:      | 2025-08-29 17:57:29,646                                                                              |
| Time-Consumption    | 0.807s                                                                                               |
| <b>Testsummary:</b> |                                                                                                      |
| <b>Info</b>         | Prepare: Switching on device                                                                         |
| <b>Info</b>         | Prepare: Setting devices to last state 100                                                           |
| <b>Success</b>      | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).             |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (ffw.sleep.main_light) to 0                                  |
| <b>Success</b>      | Value for Light.brightness (ffw.sleep.main_light) is correct (Content 0 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (ffw.sleep.main_light) to 20                                 |
| <b>Success</b>      | Value for Light.brightness (ffw.sleep.main_light) is correct (Content 20 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (ffw.sleep.main_light) to 40                                 |
| <b>Success</b>      | Value for Light.brightness (ffw.sleep.main_light) is correct (Content 40 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (ffw.sleep.main_light) to 60                                 |
| <b>Success</b>      | Value for Light.brightness (ffw.sleep.main_light) is correct (Content 60 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (ffw.sleep.main_light) to 80                                 |

**Success** Value for Light.brightness (ffw.sleep.main\_light) is correct (Content 80 and Type is <class 'int'>).

**Info** Setting state of ViDevCommon.brightness (ffw.sleep.main\_light) to 100

**Success** Value for Light.brightness (ffw.sleep.main\_light) is correct (Content 100 and Type is <class 'int'>).

---

### 3.1.55 Light.brightness (ffw.sleep.main\_light) → ViDevCommon.brightness (ffw.sleep.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.55!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:29,646                                                         |
| Finished-Time:   | 2025-08-29 17:57:30,454                                                         |
| Time-Consumption | 0.808s                                                                          |

---

#### Testsummary:

---

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                                |
| <b>Info</b>    | Prepare: Setting devices to last state 100                                                                  |
| <b>Success</b> | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).                    |
| <b>Info</b>    | Setting state of Light.brightness (ffw.sleep.main_light) to 0                                               |
| <b>Success</b> | Value for ViDevCommon.brightness (ffw.sleep.main_light) is correct (Content 0 and Type is <class 'int'>).   |
| <b>Info</b>    | Setting state of Light.brightness (ffw.sleep.main_light) to 20                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffw.sleep.main_light) is correct (Content 20 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffw.sleep.main_light) to 40                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffw.sleep.main_light) is correct (Content 40 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffw.sleep.main_light) to 60                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffw.sleep.main_light) is correct (Content 60 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffw.sleep.main_light) to 80                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (ffw.sleep.main_light) is correct (Content 80 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (ffw.sleep.main_light) to 100                                             |
| <b>Success</b> | Value for ViDevCommon.brightness (ffw.sleep.main_light) is correct (Content 100 and Type is <class 'int'>). |

---

### 3.1.56 ViDevHeating.temp\_setp (ffw.sleep.heating\_valve) → HeatingValve.temp\_setp (ffw.sleep.heating\_valve)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.56!

---

|             |                                                                                 |
|-------------|---------------------------------------------------------------------------------|
| Testrun:    | python3.13.5                                                                    |
| Caller:     | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time: | 2025-08-29 17:57:30,455                                                         |

Finished-Time: 2025-08-29 17:57:30,959

Time-Consumption 0.505s

**Testsummary:**

|                |                                                                                                               |
|----------------|---------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state 30                                                                     |
| <b>Success</b> | Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).                        |
| <b>Info</b>    | Setting state of ViDevHeating.temp_setp (ffw.sleep.heating_valve) to 15                                       |
| <b>Success</b> | Value for HeatingValve.temp_setp (ffw.sleep.heating_valve) is correct (Content 15 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevHeating.temp_setp (ffw.sleep.heating_valve) to 20                                       |
| <b>Success</b> | Value for HeatingValve.temp_setp (ffw.sleep.heating_valve) is correct (Content 20 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevHeating.temp_setp (ffw.sleep.heating_valve) to 25                                       |
| <b>Success</b> | Value for HeatingValve.temp_setp (ffw.sleep.heating_valve) is correct (Content 25 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevHeating.temp_setp (ffw.sleep.heating_valve) to 30                                       |
| <b>Success</b> | Value for HeatingValve.temp_setp (ffw.sleep.heating_valve) is correct (Content 30 and Type is <class 'int'>). |

**3.1.57 ViDevCommon.state (ffw.julian.main\_light) → Shelly.relay/0 (ffw.julian.main\_light)****Testresult**This test was passed with the state: **Success**. See also full trace in section A.1.57!

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:30,959                                                         |
| Finished-Time:   | 2025-08-29 17:57:31,262                                                         |
| Time-Consumption | 0.303s                                                                          |

**Testsummary:**

|                |                                                                                                         |
|----------------|---------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                            |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).            |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffw.julian.main_light) to True                                      |
| <b>Success</b> | Value for Shelly.relay/0 (ffw.julian.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffw.julian.main_light) to False                                     |
| <b>Success</b> | Value for Shelly.relay/0 (ffw.julian.main_light) is correct (Content False and Type is <class 'bool'>). |

**3.1.58 Shelly.relay/0 (ffw.julian.main\_light) → ViDevCommon.state (ffw.julian.main\_light)****Testresult**This test was passed with the state: **Success**. See also full trace in section A.1.58!

|             |                                                                                 |
|-------------|---------------------------------------------------------------------------------|
| Testrun:    | python3.13.5                                                                    |
| Caller:     | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time: | 2025-08-29 17:57:31,263                                                         |

Finished-Time: 2025-08-29 17:57:31,566

Time-Consumption 0.303s

#### Testsummary:

|                |                                                                                                            |
|----------------|------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                               |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).               |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffw.julian.main_light) to True                                            |
| <b>Success</b> | Value for ViDevCommon.state (ffw.julian.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffw.julian.main_light) to False                                           |
| <b>Success</b> | Value for ViDevCommon.state (ffw.julian.main_light) is correct (Content False and Type is <class 'bool'>). |

### 3.1.59 ViDevCommon.brightness (ffw.julian.main\_light) → Light.brightness (ffw.julian.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.59!

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:31,566                                                         |
| Finished-Time:   | 2025-08-29 17:57:32,374                                                         |
| Time-Consumption | 0.808s                                                                          |

#### Testsummary:

|                |                                                                                                        |
|----------------|--------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                           |
| <b>Info</b>    | Prepare: Setting devices to last state 100                                                             |
| <b>Success</b> | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).               |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffw.julian.main_light) to 0                                   |
| <b>Success</b> | Value for Light.brightness (ffw.julian.main_light) is correct (Content 0 and Type is <class 'int'>).   |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffw.julian.main_light) to 20                                  |
| <b>Success</b> | Value for Light.brightness (ffw.julian.main_light) is correct (Content 20 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffw.julian.main_light) to 40                                  |
| <b>Success</b> | Value for Light.brightness (ffw.julian.main_light) is correct (Content 40 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffw.julian.main_light) to 60                                  |
| <b>Success</b> | Value for Light.brightness (ffw.julian.main_light) is correct (Content 60 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffw.julian.main_light) to 80                                  |
| <b>Success</b> | Value for Light.brightness (ffw.julian.main_light) is correct (Content 80 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (ffw.julian.main_light) to 100                                 |
| <b>Success</b> | Value for Light.brightness (ffw.julian.main_light) is correct (Content 100 and Type is <class 'int'>). |

**3.1.60 Light.brightness (ffw.julian.main\_light) → ViDevCommon.brightness (ffw.julian.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.60!

|                     |                                                                                                              |
|---------------------|--------------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                                 |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329)                              |
| Start-Time:         | 2025-08-29 17:57:32,374                                                                                      |
| Finished-Time:      | 2025-08-29 17:57:33,180                                                                                      |
| Time-Consumption    | 0.806s                                                                                                       |
| <b>Testsummary:</b> |                                                                                                              |
| <b>Info</b>         | Prepare: Switching on device                                                                                 |
| <b>Info</b>         | Prepare: Setting devices to last state 100                                                                   |
| <b>Success</b>      | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).                     |
| <b>Info</b>         | Setting state of Light.brightness (ffw.julian.main_light) to 0                                               |
| <b>Success</b>      | Value for ViDevCommon.brightness (ffw.julian.main_light) is correct (Content 0 and Type is <class 'int'>).   |
| <b>Info</b>         | Setting state of Light.brightness (ffw.julian.main_light) to 20                                              |
| <b>Success</b>      | Value for ViDevCommon.brightness (ffw.julian.main_light) is correct (Content 20 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of Light.brightness (ffw.julian.main_light) to 40                                              |
| <b>Success</b>      | Value for ViDevCommon.brightness (ffw.julian.main_light) is correct (Content 40 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of Light.brightness (ffw.julian.main_light) to 60                                              |
| <b>Success</b>      | Value for ViDevCommon.brightness (ffw.julian.main_light) is correct (Content 60 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of Light.brightness (ffw.julian.main_light) to 80                                              |
| <b>Success</b>      | Value for ViDevCommon.brightness (ffw.julian.main_light) is correct (Content 80 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of Light.brightness (ffw.julian.main_light) to 100                                             |
| <b>Success</b>      | Value for ViDevCommon.brightness (ffw.julian.main_light) is correct (Content 100 and Type is <class 'int'>). |

**3.1.61 ViDevCommon.color\_temp (ffw.julian.main\_light) → Light.color\_temp (ffw.julian.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.61!

|                     |                                                                                        |
|---------------------|----------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                           |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329)        |
| Start-Time:         | 2025-08-29 17:57:33,180                                                                |
| Finished-Time:      | 2025-08-29 17:57:33,987                                                                |
| Time-Consumption    | 0.807s                                                                                 |
| <b>Testsummary:</b> |                                                                                        |
| <b>Info</b>         | Prepare: Switching on device                                                           |
| <b>Info</b>         | Prepare: Setting devices to last state 10                                              |
| <b>Success</b>      | Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>). |

|                |                                                                                                       |
|----------------|-------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffw.julian.main_light) to 0                                  |
| <b>Success</b> | Value for Light.color_temp (ffw.julian.main_light) is correct (Content 0 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffw.julian.main_light) to 2                                  |
| <b>Success</b> | Value for Light.color_temp (ffw.julian.main_light) is correct (Content 2 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffw.julian.main_light) to 4                                  |
| <b>Success</b> | Value for Light.color_temp (ffw.julian.main_light) is correct (Content 4 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffw.julian.main_light) to 6                                  |
| <b>Success</b> | Value for Light.color_temp (ffw.julian.main_light) is correct (Content 6 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffw.julian.main_light) to 8                                  |
| <b>Success</b> | Value for Light.color_temp (ffw.julian.main_light) is correct (Content 8 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (ffw.julian.main_light) to 10                                 |
| <b>Success</b> | Value for Light.color_temp (ffw.julian.main_light) is correct (Content 10 and Type is <class 'int'>). |

---

### 3.1.62 Light.color\_temp (ffw.julian.main\_light) → ViDevCommon.color\_temp (ffw.julian.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.62!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:33,987                                                        |
| Finished-Time:   | 2025-08-29 17:57:34,794                                                        |
| Time-Consumption | 0.806s                                                                         |

---

#### Testsummary:

|                |                                                                                                            |
|----------------|------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                               |
| <b>Info</b>    | Prepare: Setting devices to last state 10                                                                  |
| <b>Success</b> | Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).                     |
| <b>Info</b>    | Setting state of Light.color_temp (ffw.julian.main_light) to 0                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffw.julian.main_light) is correct (Content 0 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of Light.color_temp (ffw.julian.main_light) to 2                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffw.julian.main_light) is correct (Content 2 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of Light.color_temp (ffw.julian.main_light) to 4                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffw.julian.main_light) is correct (Content 4 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of Light.color_temp (ffw.julian.main_light) to 6                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffw.julian.main_light) is correct (Content 6 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of Light.color_temp (ffw.julian.main_light) to 8                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (ffw.julian.main_light) is correct (Content 8 and Type is <class 'int'>). |

**Info** Setting state of Light.color\_temp (ffw.julian.main\_light) to 10  
**Success** Value for ViDevCommon.color\_temp (ffw.julian.main\_light) is correct (Content 10 and Type is <class 'int'>).

---

### 3.1.63 ViDevHeating.temp\_setp (ffw.julian.heating\_valve) → HeatingValve.temp\_setp (ffw.julian.heating\_valve)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.63!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:34,794                                                         |
| Finished-Time:   | 2025-08-29 17:57:35,298                                                         |
| Time-Consumption | 0.504s                                                                          |

---

#### Testsummary:

---

**Info** Prepare: Setting devices to last state 30  
**Success** Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).  
**Info** Setting state of ViDevHeating.temp\_setp (ffw.julian.heating\_valve) to 15  
**Success** Value for HeatingValve.temp\_setp (ffw.julian.heating\_valve) is correct (Content 15 and Type is <class 'int'>).  
**Info** Setting state of ViDevHeating.temp\_setp (ffw.julian.heating\_valve) to 20  
**Success** Value for HeatingValve.temp\_setp (ffw.julian.heating\_valve) is correct (Content 20 and Type is <class 'int'>).  
**Info** Setting state of ViDevHeating.temp\_setp (ffw.julian.heating\_valve) to 25  
**Success** Value for HeatingValve.temp\_setp (ffw.julian.heating\_valve) is correct (Content 25 and Type is <class 'int'>).  
**Info** Setting state of ViDevHeating.temp\_setp (ffw.julian.heating\_valve) to 30  
**Success** Value for HeatingValve.temp\_setp (ffw.julian.heating\_valve) is correct (Content 30 and Type is <class 'int'>).

---

### 3.1.64 ViDevCommon.state (ffw.bath.main\_light) → Shelly.relay/0 (ffw.bath.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.64!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:35,298                                                         |
| Finished-Time:   | 2025-08-29 17:57:35,601                                                         |
| Time-Consumption | 0.303s                                                                          |

---

#### Testsummary:

---

**Info** Prepare: Setting devices to last state False  
**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).  
**Info** Setting state of ViDevCommon.state (ffw.bath.main\_light) to True  
**Success** Value for Shelly.relay/0 (ffw.bath.main\_light) is correct (Content True and Type is <class 'bool'>).

---

|                |                                                                                                       |
|----------------|-------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Setting state of ViDevCommon.state (ffw.bath.main_light) to False                                     |
| <b>Success</b> | Value for Shelly.relay/0 (ffw.bath.main_light) is correct (Content False and Type is <class 'bool'>). |

---

### 3.1.65 Shelly.relay/0 (ffw.bath.main\_light) → ViDevCommon.state (ffw.bath.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.65!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:35,601                                                        |
| Finished-Time:   | 2025-08-29 17:57:35,904                                                        |
| Time-Consumption | 0.303s                                                                         |

---

#### Testsummary:

|                |                                                                                                          |
|----------------|----------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                             |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).             |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffw.bath.main_light) to True                                            |
| <b>Success</b> | Value for ViDevCommon.state (ffw.bath.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffw.bath.main_light) to False                                           |
| <b>Success</b> | Value for ViDevCommon.state (ffw.bath.main_light) is correct (Content False and Type is <class 'bool'>). |

---

### 3.1.66 ViDevHeating.temp\_setp (ffw.bath.heating\_valve) → HeatingValve.temp\_setp (ffw.bath.heating\_valve)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.66!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:35,904                                                        |
| Finished-Time:   | 2025-08-29 17:57:36,409                                                        |
| Time-Consumption | 0.505s                                                                         |

---

#### Testsummary:

|                |                                                                                                              |
|----------------|--------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state 30                                                                    |
| <b>Success</b> | Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).                       |
| <b>Info</b>    | Setting state of ViDevHeating.temp_setp (ffw.bath.heating_valve) to 15                                       |
| <b>Success</b> | Value for HeatingValve.temp_setp (ffw.bath.heating_valve) is correct (Content 15 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevHeating.temp_setp (ffw.bath.heating_valve) to 20                                       |
| <b>Success</b> | Value for HeatingValve.temp_setp (ffw.bath.heating_valve) is correct (Content 20 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevHeating.temp_setp (ffw.bath.heating_valve) to 25                                       |
| <b>Success</b> | Value for HeatingValve.temp_setp (ffw.bath.heating_valve) is correct (Content 25 and Type is <class 'int'>). |

|                |                                                                                                              |
|----------------|--------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Setting state of ViDevHeating.temp_setp (ffw.bath.heating_valve) to 30                                       |
| <b>Success</b> | Value for HeatingValve.temp_setp (ffw.bath.heating_valve) is correct (Content 30 and Type is <class 'int'>). |

---

### 3.1.67 ViDevCommon.state (ffw.floor.main\_light) → Shelly.relay/0 (ffw.floor.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.67!

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:36,409                                                         |
| Finished-Time:   | 2025-08-29 17:57:36,712                                                         |
| Time-Consumption | 0.303s                                                                          |

---

#### Testsummary:

|                |                                                                                                        |
|----------------|--------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                           |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).           |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffw.floor.main_light) to True                                      |
| <b>Success</b> | Value for Shelly.relay/0 (ffw.floor.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.state (ffw.floor.main_light) to False                                     |
| <b>Success</b> | Value for Shelly.relay/0 (ffw.floor.main_light) is correct (Content False and Type is <class 'bool'>). |

---

### 3.1.68 Shelly.relay/0 (ffw.floor.main\_light) → ViDevCommon.state (ffw.floor.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.68!

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:36,712                                                         |
| Finished-Time:   | 2025-08-29 17:57:37,015                                                         |
| Time-Consumption | 0.303s                                                                          |

---

#### Testsummary:

|                |                                                                                                           |
|----------------|-----------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                              |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).              |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffw.floor.main_light) to True                                            |
| <b>Success</b> | Value for ViDevCommon.state (ffw.floor.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Shelly.relay/0 (ffw.floor.main_light) to False                                           |
| <b>Success</b> | Value for ViDevCommon.state (ffw.floor.main_light) is correct (Content False and Type is <class 'bool'>). |

---

**3.1.69 ViDevCommon.state (gfw.dirk.main\_light) → Shelly.relay/0 (gfw.dirk.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.69!

|                     |                                                                                                       |
|---------------------|-------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                          |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                        |
| Start-Time:         | 2025-08-29 17:57:37,015                                                                               |
| Finished-Time:      | 2025-08-29 17:57:37,318                                                                               |
| Time-Consumption    | 0.303s                                                                                                |
| <b>Testsummary:</b> |                                                                                                       |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                          |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).          |
| <b>Info</b>         | Setting state of ViDevCommon.state (gfw.dirk.main_light) to True                                      |
| <b>Success</b>      | Value for Shelly.relay/0 (gfw.dirk.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.state (gfw.dirk.main_light) to False                                     |
| <b>Success</b>      | Value for Shelly.relay/0 (gfw.dirk.main_light) is correct (Content False and Type is <class 'bool'>). |

**3.1.70 Shelly.relay/0 (gfw.dirk.main\_light) → ViDevCommon.state (gfw.dirk.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.70!

|                     |                                                                                                          |
|---------------------|----------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                             |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                           |
| Start-Time:         | 2025-08-29 17:57:37,318                                                                                  |
| Finished-Time:      | 2025-08-29 17:57:37,621                                                                                  |
| Time-Consumption    | 0.303s                                                                                                   |
| <b>Testsummary:</b> |                                                                                                          |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                             |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).             |
| <b>Info</b>         | Setting state of Shelly.relay/0 (gfw.dirk.main_light) to True                                            |
| <b>Success</b>      | Value for ViDevCommon.state (gfw.dirk.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of Shelly.relay/0 (gfw.dirk.main_light) to False                                           |
| <b>Success</b>      | Value for ViDevCommon.state (gfw.dirk.main_light) is correct (Content False and Type is <class 'bool'>). |

**3.1.71 ViDevCommon.state (gfw.dirk.desk\_light) → Light.state (gfw.dirk.desk\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.71!

---

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:37,621                                                        |
| Finished-Time:   | 2025-08-29 17:57:37,924                                                        |
| Time-Consumption | 0.303s                                                                         |

---

**Testsummary:**


---

|                |                                                                                                    |
|----------------|----------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                       |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).       |
| <b>Info</b>    | Setting state of ViDevCommon.state (gfw.dirk.desk_light) to True                                   |
| <b>Success</b> | Value for Light.state (gfw.dirk.desk_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.state (gfw.dirk.desk_light) to False                                  |
| <b>Success</b> | Value for Light.state (gfw.dirk.desk_light) is correct (Content False and Type is <class 'bool'>). |

---

**3.1.72 Light.state (gfw.dirk.desk\_light) → ViDevCommon.state (gfw.dirk.desk\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.72!

---

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:37,924                                                        |
| Finished-Time:   | 2025-08-29 17:57:38,227                                                        |
| Time-Consumption | 0.303s                                                                         |

---

**Testsummary:**


---

|                |                                                                                                          |
|----------------|----------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                             |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).             |
| <b>Info</b>    | Setting state of Light.state (gfw.dirk.desk_light) to True                                               |
| <b>Success</b> | Value for ViDevCommon.state (gfw.dirk.desk_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Light.state (gfw.dirk.desk_light) to False                                              |
| <b>Success</b> | Value for ViDevCommon.state (gfw.dirk.desk_light) is correct (Content False and Type is <class 'bool'>). |

---

**3.1.73 ViDevCommon.state (gfw.dirk.pc\_dock) → Powerplug1P.state (gfw.dirk.dock)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.73!

---

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:38,228                                                        |
| Finished-Time:   | 2025-08-29 17:57:38,531                                                        |
| Time-Consumption | 0.303s                                                                         |

---

**Testsummary:**

|                |                                                                                                    |
|----------------|----------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                       |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).       |
| <b>Info</b>    | Setting state of ViDevCommon.state (gfw.dirk.pc_dock) to True                                      |
| <b>Success</b> | Value for Powerplug1P.state (gfw.dirk.dock) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.state (gfw.dirk.pc_dock) to False                                     |
| <b>Success</b> | Value for Powerplug1P.state (gfw.dirk.dock) is correct (Content False and Type is <class 'bool'>). |

**3.1.74 Powerplug1P.state (gfw.dirk.dock) → ViDevCommon.state (gfw.dirk.pc\_dock)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.74!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:38,531                                                        |
| Finished-Time:   | 2025-08-29 17:57:38,834                                                        |
| Time-Consumption | 0.303s                                                                         |

**Testsummary:**

|                |                                                                                                       |
|----------------|-------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                          |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).          |
| <b>Info</b>    | Setting state of Powerplug1P.state (gfw.dirk.dock) to True                                            |
| <b>Success</b> | Value for ViDevCommon.state (gfw.dirk.pc_dock) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Powerplug1P.state (gfw.dirk.dock) to False                                           |
| <b>Success</b> | Value for ViDevCommon.state (gfw.dirk.pc_dock) is correct (Content False and Type is <class 'bool'>). |

**3.1.75 ViDevCommon.state (gfw.dirk.amplifier) → Powerplug4P.amplifier (gfw.dirk.powerplug)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.75!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:38,834                                                        |
| Finished-Time:   | 2025-08-29 17:57:39,137                                                        |
| Time-Consumption | 0.303s                                                                         |

**Testsummary:**

|                |                                                                                                            |
|----------------|------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                               |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).               |
| <b>Info</b>    | Setting state of ViDevCommon.state (gfw.dirk.amplifier) to True                                            |
| <b>Success</b> | Value for Powerplug4P.amplifier (gfw.dirk.powerplug) is correct (Content True and Type is <class 'bool'>). |

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Setting state of ViDevCommon.state (gfw.dirk.amplifier) to False                                            |
| <b>Success</b> | Value for Powerplug4P.amplifier (gfw.dirk.powerplug) is correct (Content False and Type is <class 'bool'>). |

---

### 3.1.76 Powerplug4P.amplifier (gfw.dirk.powerplug) → ViDevCommon.state (gfw.dirk.amplifier)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.76!

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:39,138                                                         |
| Finished-Time:   | 2025-08-29 17:57:39,441                                                         |
| Time-Consumption | 0.303s                                                                          |

---

#### Testsummary:

|                |                                                                                                         |
|----------------|---------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                            |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).            |
| <b>Info</b>    | Setting state of Powerplug4P.amplifier (gfw.dirk.powerplug) to True                                     |
| <b>Success</b> | Value for ViDevCommon.state (gfw.dirk.amplifier) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Powerplug4P.amplifier (gfw.dirk.powerplug) to False                                    |
| <b>Success</b> | Value for ViDevCommon.state (gfw.dirk.amplifier) is correct (Content False and Type is <class 'bool'>). |

---

### 3.1.77 ViDevCommon.state (gfw.dirk.phono) → Powerplug4P.phono (gfw.dirk.powerplug)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.77!

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:39,441                                                         |
| Finished-Time:   | 2025-08-29 17:57:39,744                                                         |
| Time-Consumption | 0.303s                                                                          |

---

#### Testsummary:

|                |                                                                                                         |
|----------------|---------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                            |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).            |
| <b>Info</b>    | Setting state of ViDevCommon.state (gfw.dirk.phono) to True                                             |
| <b>Success</b> | Value for Powerplug4P.phono (gfw.dirk.powerplug) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.state (gfw.dirk.phono) to False                                            |
| <b>Success</b> | Value for Powerplug4P.phono (gfw.dirk.powerplug) is correct (Content False and Type is <class 'bool'>). |

---

**3.1.78 Powerplug4P.phono (gfw.dirk.powerplug) → ViDevCommon.state (gfw.dirk.phono)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.78!

|                     |                                                                                                     |
|---------------------|-----------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                        |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                      |
| Start-Time:         | 2025-08-29 17:57:39,744                                                                             |
| Finished-Time:      | 2025-08-29 17:57:40,047                                                                             |
| Time-Consumption    | 0.303s                                                                                              |
| <b>Testsummary:</b> |                                                                                                     |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                        |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).        |
| <b>Info</b>         | Setting state of Powerplug4P.phono (gfw.dirk.powerplug) to True                                     |
| <b>Success</b>      | Value for ViDevCommon.state (gfw.dirk.phono) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of Powerplug4P.phono (gfw.dirk.powerplug) to False                                    |
| <b>Success</b>      | Value for ViDevCommon.state (gfw.dirk.phono) is correct (Content False and Type is <class 'bool'>). |

**3.1.79 ViDevCommon.state (gfw.dirk.cd\_player) → Powerplug4P.cd-player (gfw.dirk.powerplug)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.79!

|                     |                                                                                                             |
|---------------------|-------------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                                |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                              |
| Start-Time:         | 2025-08-29 17:57:40,047                                                                                     |
| Finished-Time:      | 2025-08-29 17:57:40,350                                                                                     |
| Time-Consumption    | 0.303s                                                                                                      |
| <b>Testsummary:</b> |                                                                                                             |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                                |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                |
| <b>Info</b>         | Setting state of ViDevCommon.state (gfw.dirk.cd_player) to True                                             |
| <b>Success</b>      | Value for Powerplug4P.cd-player (gfw.dirk.powerplug) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.state (gfw.dirk.cd_player) to False                                            |
| <b>Success</b>      | Value for Powerplug4P.cd-player (gfw.dirk.powerplug) is correct (Content False and Type is <class 'bool'>). |

**3.1.80 Powerplug4P.cd-player (gfw.dirk.powerplug) → ViDevCommon.state (gfw.dirk.cd\_player)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.80!

---

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:40,350                                                        |
| Finished-Time:   | 2025-08-29 17:57:40,653                                                        |
| Time-Consumption | 0.303s                                                                         |

---

**Testsummary:**


---

|                |                                                                                                         |
|----------------|---------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                            |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).            |
| <b>Info</b>    | Setting state of Powerplug4P.cd-player (gfw.dirk.powerplug) to True                                     |
| <b>Success</b> | Value for ViDevCommon.state (gfw.dirk.cd_player) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Powerplug4P.cd-player (gfw.dirk.powerplug) to False                                    |
| <b>Success</b> | Value for ViDevCommon.state (gfw.dirk.cd_player) is correct (Content False and Type is <class 'bool'>). |

---

**3.1.81 ViDevCommon.state (gfw.dirk.bt) → Powerplug4P.bluetooth (gfw.dirk.powerplug)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.81!

---

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:40,653                                                        |
| Finished-Time:   | 2025-08-29 17:57:40,956                                                        |
| Time-Consumption | 0.303s                                                                         |

---

**Testsummary:**


---

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                |
| <b>Info</b>    | Setting state of ViDevCommon.state (gfw.dirk.bt) to True                                                    |
| <b>Success</b> | Value for Powerplug4P.bluetooth (gfw.dirk.powerplug) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.state (gfw.dirk.bt) to False                                                   |
| <b>Success</b> | Value for Powerplug4P.bluetooth (gfw.dirk.powerplug) is correct (Content False and Type is <class 'bool'>). |

---

**3.1.82 Powerplug4P.bluetooth (gfw.dirk.powerplug) → ViDevCommon.state (gfw.dirk.bt)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.82!

---

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:40,956                                                        |
| Finished-Time:   | 2025-08-29 17:57:41,259                                                        |
| Time-Consumption | 0.303s                                                                         |

---

**Testsummary:**

|                |                                                                                                  |
|----------------|--------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                     |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).     |
| <b>Info</b>    | Setting state of Powerplug4P.bluetooth (gfw.dirk.powerplug) to True                              |
| <b>Success</b> | Value for ViDevCommon.state (gfw.dirk.bt) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Powerplug4P.bluetooth (gfw.dirk.powerplug) to False                             |
| <b>Success</b> | Value for ViDevCommon.state (gfw.dirk.bt) is correct (Content False and Type is <class 'bool'>). |

**3.1.83 Powerplug4P.phono (gfw.dirk.powerplug) → Powerplug4P.amplifier (gfw.dirk.powerplug)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.83!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:41,259                                                        |
| Finished-Time:   | 2025-08-29 17:57:41,562                                                        |
| Time-Consumption | 0.303s                                                                         |

**Testsummary:**

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                |
| <b>Info</b>    | Setting state of Powerplug4P.phono (gfw.dirk.powerplug) to True                                             |
| <b>Success</b> | Value for Powerplug4P.amplifier (gfw.dirk.powerplug) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Powerplug4P.phono (gfw.dirk.powerplug) to False                                            |
| <b>Success</b> | Value for Powerplug4P.amplifier (gfw.dirk.powerplug) is correct (Content False and Type is <class 'bool'>). |

**3.1.84 Powerplug4P.cd-player (gfw.dirk.powerplug) → Powerplug4P.amplifier (gfw.dirk.powerplug)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.84!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:41,562                                                        |
| Finished-Time:   | 2025-08-29 17:57:41,865                                                        |
| Time-Consumption | 0.303s                                                                         |

**Testsummary:**

|                |                                                                                                            |
|----------------|------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                               |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).               |
| <b>Info</b>    | Setting state of Powerplug4P.cd-player (gfw.dirk.powerplug) to True                                        |
| <b>Success</b> | Value for Powerplug4P.amplifier (gfw.dirk.powerplug) is correct (Content True and Type is <class 'bool'>). |

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Setting state of Powerplug4P.cd-player (gfw.dirk.powerplug) to False                                        |
| <b>Success</b> | Value for Powerplug4P.amplifier (gfw.dirk.powerplug) is correct (Content False and Type is <class 'bool'>). |

---

### 3.1.85 Powerplug4P.bluetooth (gfw.dirk.powerplug) → Powerplug4P.amplifier (gfw.dirk.powerplug)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.85!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:41,865                                                        |
| Finished-Time:   | 2025-08-29 17:57:42,168                                                        |
| Time-Consumption | 0.303s                                                                         |

---

#### Testsummary:

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                |
| <b>Info</b>    | Setting state of Powerplug4P.bluetooth (gfw.dirk.powerplug) to True                                         |
| <b>Success</b> | Value for Powerplug4P.amplifier (gfw.dirk.powerplug) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Powerplug4P.bluetooth (gfw.dirk.powerplug) to False                                        |
| <b>Success</b> | Value for Powerplug4P.amplifier (gfw.dirk.powerplug) is correct (Content False and Type is <class 'bool'>). |

---

### 3.1.86 ViDevCommon.brightness (gfw.dirk.main\_light) → Light.brightness (gfw.dirk.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.86!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:42,168                                                        |
| Finished-Time:   | 2025-08-29 17:57:42,975                                                        |
| Time-Consumption | 0.807s                                                                         |

---

#### Testsummary:

|                |                                                                                                     |
|----------------|-----------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                        |
| <b>Info</b>    | Prepare: Setting devices to last state 100                                                          |
| <b>Success</b> | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).            |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (gfw.dirk.main_light) to 0                                  |
| <b>Success</b> | Value for Light.brightness (gfw.dirk.main_light) is correct (Content 0 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (gfw.dirk.main_light) to 20                                 |
| <b>Success</b> | Value for Light.brightness (gfw.dirk.main_light) is correct (Content 20 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (gfw.dirk.main_light) to 40                                 |

**Success** Value for Light.brightness (gfw.dirk.main\_light) is correct (Content 40 and Type is <class 'int'>).

**Info** Setting state of ViDevCommon.brightness (gfw.dirk.main\_light) to 60

**Success** Value for Light.brightness (gfw.dirk.main\_light) is correct (Content 60 and Type is <class 'int'>).

**Info** Setting state of ViDevCommon.brightness (gfw.dirk.main\_light) to 80

**Success** Value for Light.brightness (gfw.dirk.main\_light) is correct (Content 80 and Type is <class 'int'>).

**Info** Setting state of ViDevCommon.brightness (gfw.dirk.main\_light) to 100

**Success** Value for Light.brightness (gfw.dirk.main\_light) is correct (Content 100 and Type is <class 'int'>).

---

### 3.1.87 Light.brightness (gfw.dirk.main\_light) → ViDevCommon.brightness (gfw.dirk.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.87!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:42,976                                                         |
| Finished-Time:   | 2025-08-29 17:57:43,783                                                         |
| Time-Consumption | 0.807s                                                                          |

---

#### Testsummary:

---

**Info** Prepare: Switching on device

**Info** Prepare: Setting devices to last state 100

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

**Info** Setting state of Light.brightness (gfw.dirk.main\_light) to 0

**Success** Value for ViDevCommon.brightness (gfw.dirk.main\_light) is correct (Content 0 and Type is <class 'int'>).

**Info** Setting state of Light.brightness (gfw.dirk.main\_light) to 20

**Success** Value for ViDevCommon.brightness (gfw.dirk.main\_light) is correct (Content 20 and Type is <class 'int'>).

**Info** Setting state of Light.brightness (gfw.dirk.main\_light) to 40

**Success** Value for ViDevCommon.brightness (gfw.dirk.main\_light) is correct (Content 40 and Type is <class 'int'>).

**Info** Setting state of Light.brightness (gfw.dirk.main\_light) to 60

**Success** Value for ViDevCommon.brightness (gfw.dirk.main\_light) is correct (Content 60 and Type is <class 'int'>).

**Info** Setting state of Light.brightness (gfw.dirk.main\_light) to 80

**Success** Value for ViDevCommon.brightness (gfw.dirk.main\_light) is correct (Content 80 and Type is <class 'int'>).

**Info** Setting state of Light.brightness (gfw.dirk.main\_light) to 100

**Success** Value for ViDevCommon.brightness (gfw.dirk.main\_light) is correct (Content 100 and Type is <class 'int'>).

---

**3.1.88 ViDevCommon.color\_temp (gfw.dirk.main\_light) → Light.color\_temp (gfw.dirk.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.88!

|                     |                                                                                                     |
|---------------------|-----------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                        |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                      |
| Start-Time:         | 2025-08-29 17:57:43,783                                                                             |
| Finished-Time:      | 2025-08-29 17:57:44,590                                                                             |
| Time-Consumption    | 0.807s                                                                                              |
| <b>Testsummary:</b> |                                                                                                     |
| <b>Info</b>         | Prepare: Switching on device                                                                        |
| <b>Info</b>         | Prepare: Setting devices to last state 10                                                           |
| <b>Success</b>      | Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).              |
| <b>Info</b>         | Setting state of ViDevCommon.color_temp (gfw.dirk.main_light) to 0                                  |
| <b>Success</b>      | Value for Light.color_temp (gfw.dirk.main_light) is correct (Content 0 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.color_temp (gfw.dirk.main_light) to 2                                  |
| <b>Success</b>      | Value for Light.color_temp (gfw.dirk.main_light) is correct (Content 2 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.color_temp (gfw.dirk.main_light) to 4                                  |
| <b>Success</b>      | Value for Light.color_temp (gfw.dirk.main_light) is correct (Content 4 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.color_temp (gfw.dirk.main_light) to 6                                  |
| <b>Success</b>      | Value for Light.color_temp (gfw.dirk.main_light) is correct (Content 6 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.color_temp (gfw.dirk.main_light) to 8                                  |
| <b>Success</b>      | Value for Light.color_temp (gfw.dirk.main_light) is correct (Content 8 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.color_temp (gfw.dirk.main_light) to 10                                 |
| <b>Success</b>      | Value for Light.color_temp (gfw.dirk.main_light) is correct (Content 10 and Type is <class 'int'>). |

**3.1.89 Light.color\_temp (gfw.dirk.main\_light) → ViDevCommon.color\_temp (gfw.dirk.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.89!

|                     |                                                                                        |
|---------------------|----------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                           |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)         |
| Start-Time:         | 2025-08-29 17:57:44,590                                                                |
| Finished-Time:      | 2025-08-29 17:57:45,398                                                                |
| Time-Consumption    | 0.807s                                                                                 |
| <b>Testsummary:</b> |                                                                                        |
| <b>Info</b>         | Prepare: Switching on device                                                           |
| <b>Info</b>         | Prepare: Setting devices to last state 10                                              |
| <b>Success</b>      | Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>). |

|                |                                                                                                           |
|----------------|-----------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Setting state of Light.color_temp (gfw.dirk.main_light) to 0                                              |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.dirk.main_light) is correct (Content 0 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.color_temp (gfw.dirk.main_light) to 2                                              |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.dirk.main_light) is correct (Content 2 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.color_temp (gfw.dirk.main_light) to 4                                              |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.dirk.main_light) is correct (Content 4 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.color_temp (gfw.dirk.main_light) to 6                                              |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.dirk.main_light) is correct (Content 6 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.color_temp (gfw.dirk.main_light) to 8                                              |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.dirk.main_light) is correct (Content 8 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.color_temp (gfw.dirk.main_light) to 10                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.dirk.main_light) is correct (Content 10 and Type is <class 'int'>). |

---

### 3.1.90 ViDevCommon.brightness (gfw.dirk.desk\_light) → Light.brightness (gfw.dirk.desk\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.90!

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:45,398                                                         |
| Finished-Time:   | 2025-08-29 17:57:46,205                                                         |
| Time-Consumption | 0.807s                                                                          |

---

#### Testsummary:

|                |                                                                                                     |
|----------------|-----------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                        |
| <b>Info</b>    | Prepare: Setting devices to last state 100                                                          |
| <b>Success</b> | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).            |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (gfw.dirk.desk_light) to 0                                  |
| <b>Success</b> | Value for Light.brightness (gfw.dirk.desk_light) is correct (Content 0 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (gfw.dirk.desk_light) to 20                                 |
| <b>Success</b> | Value for Light.brightness (gfw.dirk.desk_light) is correct (Content 20 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (gfw.dirk.desk_light) to 40                                 |
| <b>Success</b> | Value for Light.brightness (gfw.dirk.desk_light) is correct (Content 40 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (gfw.dirk.desk_light) to 60                                 |
| <b>Success</b> | Value for Light.brightness (gfw.dirk.desk_light) is correct (Content 60 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (gfw.dirk.desk_light) to 80                                 |
| <b>Success</b> | Value for Light.brightness (gfw.dirk.desk_light) is correct (Content 80 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevCommon.brightness (gfw.dirk.desk_light) to 100                                |

**Success** Value for Light.brightness (gfw.dirk.desk\_light) is correct (Content 100 and Type is <class 'int'>).

---

### 3.1.91 Light.brightness (gfw.dirk.desk\_light) → ViDevCommon.brightness (gfw.dirk.desk\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.91!

---

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:46,206                                                        |
| Finished-Time:   | 2025-08-29 17:57:47,012                                                        |
| Time-Consumption | 0.807s                                                                         |

---

#### Testsummary:

---

|                |                                                                                                            |
|----------------|------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                               |
| <b>Info</b>    | Prepare: Setting devices to last state 100                                                                 |
| <b>Success</b> | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).                   |
| <b>Info</b>    | Setting state of Light.brightness (gfw.dirk.desk_light) to 0                                               |
| <b>Success</b> | Value for ViDevCommon.brightness (gfw.dirk.desk_light) is correct (Content 0 and Type is <class 'int'>).   |
| <b>Info</b>    | Setting state of Light.brightness (gfw.dirk.desk_light) to 20                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (gfw.dirk.desk_light) is correct (Content 20 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (gfw.dirk.desk_light) to 40                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (gfw.dirk.desk_light) is correct (Content 40 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (gfw.dirk.desk_light) to 60                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (gfw.dirk.desk_light) is correct (Content 60 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (gfw.dirk.desk_light) to 80                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (gfw.dirk.desk_light) is correct (Content 80 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (gfw.dirk.desk_light) to 100                                             |
| <b>Success</b> | Value for ViDevCommon.brightness (gfw.dirk.desk_light) is correct (Content 100 and Type is <class 'int'>). |

---

### 3.1.92 ViDevCommon.color\_temp (gfw.dirk.desk\_light) → Light.color\_temp (gfw.dirk.desk\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.92!

---

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:47,013                                                        |
| Finished-Time:   | 2025-08-29 17:57:47,820                                                        |
| Time-Consumption | 0.807s                                                                         |

---

**Testsummary:**

|                |                                                                                                     |
|----------------|-----------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                        |
| <b>Info</b>    | Prepare: Setting devices to last state 10                                                           |
| <b>Success</b> | Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).              |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (gfw.dirk.desk_light) to 0                                  |
| <b>Success</b> | Value for Light.color_temp (gfw.dirk.desk_light) is correct (Content 0 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (gfw.dirk.desk_light) to 2                                  |
| <b>Success</b> | Value for Light.color_temp (gfw.dirk.desk_light) is correct (Content 2 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (gfw.dirk.desk_light) to 4                                  |
| <b>Success</b> | Value for Light.color_temp (gfw.dirk.desk_light) is correct (Content 4 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (gfw.dirk.desk_light) to 6                                  |
| <b>Success</b> | Value for Light.color_temp (gfw.dirk.desk_light) is correct (Content 6 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (gfw.dirk.desk_light) to 8                                  |
| <b>Success</b> | Value for Light.color_temp (gfw.dirk.desk_light) is correct (Content 8 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (gfw.dirk.desk_light) to 10                                 |
| <b>Success</b> | Value for Light.color_temp (gfw.dirk.desk_light) is correct (Content 10 and Type is <class 'int'>). |

**3.1.93 Light.color\_temp (gfw.dirk.desk\_light) → ViDevCommon.color\_temp (gfw.dirk.desk\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.93!

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:47,820                                                         |
| Finished-Time:   | 2025-08-29 17:57:48,627                                                         |
| Time-Consumption | 0.807s                                                                          |

**Testsummary:**

|                |                                                                                                          |
|----------------|----------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                             |
| <b>Info</b>    | Prepare: Setting devices to last state 10                                                                |
| <b>Success</b> | Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).                   |
| <b>Info</b>    | Setting state of Light.color_temp (gfw.dirk.desk_light) to 0                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.dirk.desk_light) is correct (Content 0 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of Light.color_temp (gfw.dirk.desk_light) to 2                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.dirk.desk_light) is correct (Content 2 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of Light.color_temp (gfw.dirk.desk_light) to 4                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.dirk.desk_light) is correct (Content 4 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of Light.color_temp (gfw.dirk.desk_light) to 6                                             |

|                |                                                                                                           |
|----------------|-----------------------------------------------------------------------------------------------------------|
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.dirk.desk_light) is correct (Content 6 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.color_temp (gfw.dirk.desk_light) to 8                                              |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.dirk.desk_light) is correct (Content 8 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.color_temp (gfw.dirk.desk_light) to 10                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.dirk.desk_light) is correct (Content 10 and Type is <class 'int'>). |

### 3.1.94 ViDevHeating.temp\_setp (gfw.dirk.heating\_valve) → HeatingValve.temp\_setp (gfw.dirk.heating\_valve)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.94!

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:48,627                                                         |
| Finished-Time:   | 2025-08-29 17:57:49,132                                                         |
| Time-Consumption | 0.505s                                                                          |

#### Testsummary:

|                |                                                                                                              |
|----------------|--------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state 30                                                                    |
| <b>Success</b> | Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).                       |
| <b>Info</b>    | Setting state of ViDevHeating.temp_setp (gfw.dirk.heating_valve) to 15                                       |
| <b>Success</b> | Value for HeatingValve.temp_setp (gfw.dirk.heating_valve) is correct (Content 15 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevHeating.temp_setp (gfw.dirk.heating_valve) to 20                                       |
| <b>Success</b> | Value for HeatingValve.temp_setp (gfw.dirk.heating_valve) is correct (Content 20 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevHeating.temp_setp (gfw.dirk.heating_valve) to 25                                       |
| <b>Success</b> | Value for HeatingValve.temp_setp (gfw.dirk.heating_valve) is correct (Content 25 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of ViDevHeating.temp_setp (gfw.dirk.heating_valve) to 30                                       |
| <b>Success</b> | Value for HeatingValve.temp_setp (gfw.dirk.heating_valve) is correct (Content 30 and Type is <class 'int'>). |

### 3.1.95 ViDevCommon.state (gfw.marion.main\_light) → Shelly.relay/0 (gfw.marion.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.95!

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:49,132                                                         |
| Finished-Time:   | 2025-08-29 17:57:49,435                                                         |
| Time-Consumption | 0.303s                                                                          |

#### Testsummary:

|                |                                                                                                         |
|----------------|---------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                            |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).            |
| <b>Info</b>    | Setting state of ViDevCommon.state (gfw.marion.main_light) to True                                      |
| <b>Success</b> | Value for Shelly.relay/0 (gfw.marion.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.state (gfw.marion.main_light) to False                                     |
| <b>Success</b> | Value for Shelly.relay/0 (gfw.marion.main_light) is correct (Content False and Type is <class 'bool'>). |

### 3.1.96 Shelly.relay/0 (gfw.marion.main\_light) → ViDevCommon.state (gfw.marion.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.96!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:49,436                                                        |
| Finished-Time:   | 2025-08-29 17:57:49,738                                                        |
| Time-Consumption | 0.303s                                                                         |

#### Testsummary:

|                |                                                                                                            |
|----------------|------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                               |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).               |
| <b>Info</b>    | Setting state of Shelly.relay/0 (gfw.marion.main_light) to True                                            |
| <b>Success</b> | Value for ViDevCommon.state (gfw.marion.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Shelly.relay/0 (gfw.marion.main_light) to False                                           |
| <b>Success</b> | Value for ViDevCommon.state (gfw.marion.main_light) is correct (Content False and Type is <class 'bool'>). |

### 3.1.97 ViDevCommon.state (gfw.marion.window\_light) → Light.state (gfw.marion.window\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.97!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:49,738                                                        |
| Finished-Time:   | 2025-08-29 17:57:50,041                                                        |
| Time-Consumption | 0.303s                                                                         |

#### Testsummary:

|                |                                                                                                       |
|----------------|-------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                          |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).          |
| <b>Info</b>    | Setting state of ViDevCommon.state (gfw.marion.window_light) to True                                  |
| <b>Success</b> | Value for Light.state (gfw.marion.window_light) is correct (Content True and Type is <class 'bool'>). |
| <b>Info</b>    | Setting state of ViDevCommon.state (gfw.marion.window_light) to False                                 |

**Success** Value for Light.state (gfw.marion.window\_light) is correct (Content False and Type is <class 'bool'>).

---

### 3.1.98 Light.state (gfw.marion.window\_light) → ViDevCommon.state (gfw.marion.window\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.98!

---

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:50,041                                                        |
| Finished-Time:   | 2025-08-29 17:57:50,345                                                        |
| Time-Consumption | 0.303s                                                                         |

---

#### Testsummary:

---

|                |                                                                                                              |
|----------------|--------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                 |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                 |
| <b>Info</b>    | Setting state of Light.state (gfw.marion.window_light) to True                                               |
| <b>Success</b> | Value for ViDevCommon.state (gfw.marion.window_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Light.state (gfw.marion.window_light) to False                                              |
| <b>Success</b> | Value for ViDevCommon.state (gfw.marion.window_light) is correct (Content False and Type is <class 'bool'>). |

---

### 3.1.99 Shelly.relay/0 (gfw.marion.main\_light) → Light.state (gfw.marion.window\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.99!

---

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:50,345                                                        |
| Finished-Time:   | 2025-08-29 17:57:50,647                                                        |
| Time-Consumption | 0.303s                                                                         |

---

#### Testsummary:

---

|                |                                                                                                        |
|----------------|--------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                           |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).           |
| <b>Info</b>    | Setting state of Shelly.relay/0 (gfw.marion.main_light) to True                                        |
| <b>Success</b> | Value for Light.state (gfw.marion.window_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of Shelly.relay/0 (gfw.marion.main_light) to False                                       |
| <b>Success</b> | Value for Light.state (gfw.marion.window_light) is correct (Content False and Type is <class 'bool'>). |

---

**3.1.100 ViDevHeating.temp\_setp (gfw.marion.heating\_valve) → HeatingValve.temp\_setp (gfw.marion.heating\_valve)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.100!

|                     |                                                                                                                |
|---------------------|----------------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                                   |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                                 |
| Start-Time:         | 2025-08-29 17:57:50,648                                                                                        |
| Finished-Time:      | 2025-08-29 17:57:51,153                                                                                        |
| Time-Consumption    | 0.505s                                                                                                         |
| <b>Testsummary:</b> |                                                                                                                |
| <b>Info</b>         | Prepare: Setting devices to last state 30                                                                      |
| <b>Success</b>      | Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).                         |
| <b>Info</b>         | Setting state of ViDevHeating.temp_setp (gfw.marion.heating_valve) to 15                                       |
| <b>Success</b>      | Value for HeatingValve.temp_setp (gfw.marion.heating_valve) is correct (Content 15 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevHeating.temp_setp (gfw.marion.heating_valve) to 20                                       |
| <b>Success</b>      | Value for HeatingValve.temp_setp (gfw.marion.heating_valve) is correct (Content 20 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevHeating.temp_setp (gfw.marion.heating_valve) to 25                                       |
| <b>Success</b>      | Value for HeatingValve.temp_setp (gfw.marion.heating_valve) is correct (Content 25 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevHeating.temp_setp (gfw.marion.heating_valve) to 30                                       |
| <b>Success</b>      | Value for HeatingValve.temp_setp (gfw.marion.heating_valve) is correct (Content 30 and Type is <class 'int'>). |

**3.1.101 ViDevCommon.state (gfw.floor.main\_light) → Shelly.relay/0 (gfw.floor.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.101!

|                     |                                                                                                        |
|---------------------|--------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                           |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                         |
| Start-Time:         | 2025-08-29 17:57:51,153                                                                                |
| Finished-Time:      | 2025-08-29 17:57:51,456                                                                                |
| Time-Consumption    | 0.303s                                                                                                 |
| <b>Testsummary:</b> |                                                                                                        |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                           |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).           |
| <b>Info</b>         | Setting state of ViDevCommon.state (gfw.floor.main_light) to True                                      |
| <b>Success</b>      | Value for Shelly.relay/0 (gfw.floor.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.state (gfw.floor.main_light) to False                                     |
| <b>Success</b>      | Value for Shelly.relay/0 (gfw.floor.main_light) is correct (Content False and Type is <class 'bool'>). |

**3.1.102 Shelly.relay/0 (gfw.floor.main\_light) → ViDevCommon.state (gfw.floor.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.102!

|                     |                                                                                                           |
|---------------------|-----------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                              |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                            |
| Start-Time:         | 2025-08-29 17:57:51,456                                                                                   |
| Finished-Time:      | 2025-08-29 17:57:51,759                                                                                   |
| Time-Consumption    | 0.303s                                                                                                    |
| <b>Testsummary:</b> |                                                                                                           |
| <b>Info</b>         | Prepare: Setting devices to last state False                                                              |
| <b>Success</b>      | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).              |
| <b>Info</b>         | Setting state of Shelly.relay/0 (gfw.floor.main_light) to True                                            |
| <b>Success</b>      | Value for ViDevCommon.state (gfw.floor.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>         | Setting state of Shelly.relay/0 (gfw.floor.main_light) to False                                           |
| <b>Success</b>      | Value for ViDevCommon.state (gfw.floor.main_light) is correct (Content False and Type is <class 'bool'>). |

**3.1.103 ViDevCommon.brightness (gfw.floor.main\_light) → Light.brightness (gfw.floor.main\_light)****Testresult**

This test was passed with the state: **Success**. See also full trace in section A.1.103!

|                     |                                                                                                      |
|---------------------|------------------------------------------------------------------------------------------------------|
| Testrun:            | python3.13.5                                                                                         |
| Caller:             | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329)                       |
| Start-Time:         | 2025-08-29 17:57:51,759                                                                              |
| Finished-Time:      | 2025-08-29 17:57:52,567                                                                              |
| Time-Consumption    | 0.808s                                                                                               |
| <b>Testsummary:</b> |                                                                                                      |
| <b>Info</b>         | Prepare: Switching on device                                                                         |
| <b>Info</b>         | Prepare: Setting devices to last state 100                                                           |
| <b>Success</b>      | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).             |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (gfw.floor.main_light) to 0                                  |
| <b>Success</b>      | Value for Light.brightness (gfw.floor.main_light) is correct (Content 0 and Type is <class 'int'>).  |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (gfw.floor.main_light) to 20                                 |
| <b>Success</b>      | Value for Light.brightness (gfw.floor.main_light) is correct (Content 20 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (gfw.floor.main_light) to 40                                 |
| <b>Success</b>      | Value for Light.brightness (gfw.floor.main_light) is correct (Content 40 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (gfw.floor.main_light) to 60                                 |
| <b>Success</b>      | Value for Light.brightness (gfw.floor.main_light) is correct (Content 60 and Type is <class 'int'>). |
| <b>Info</b>         | Setting state of ViDevCommon.brightness (gfw.floor.main_light) to 80                                 |

**Success** Value for Light.brightness (gfw.floor.main\_light) is correct (Content 80 and Type is <class 'int'>).

**Info** Setting state of ViDevCommon.brightness (gfw.floor.main\_light) to 100

**Success** Value for Light.brightness (gfw.floor.main\_light) is correct (Content 100 and Type is <class 'int'>).

---

### 3.1.104 Light.brightness (gfw.floor.main\_light) → ViDevCommon.brightness (gfw.floor.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.104!

---

|                  |                                                                                 |
|------------------|---------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                    |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time:      | 2025-08-29 17:57:52,568                                                         |
| Finished-Time:   | 2025-08-29 17:57:53,377                                                         |
| Time-Consumption | 0.809s                                                                          |

---

#### Testsummary:

---

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                                |
| <b>Info</b>    | Prepare: Setting devices to last state 100                                                                  |
| <b>Success</b> | Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).                    |
| <b>Info</b>    | Setting state of Light.brightness (gfw.floor.main_light) to 0                                               |
| <b>Success</b> | Value for ViDevCommon.brightness (gfw.floor.main_light) is correct (Content 0 and Type is <class 'int'>).   |
| <b>Info</b>    | Setting state of Light.brightness (gfw.floor.main_light) to 20                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (gfw.floor.main_light) is correct (Content 20 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (gfw.floor.main_light) to 40                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (gfw.floor.main_light) is correct (Content 40 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (gfw.floor.main_light) to 60                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (gfw.floor.main_light) is correct (Content 60 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (gfw.floor.main_light) to 80                                              |
| <b>Success</b> | Value for ViDevCommon.brightness (gfw.floor.main_light) is correct (Content 80 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.brightness (gfw.floor.main_light) to 100                                             |
| <b>Success</b> | Value for ViDevCommon.brightness (gfw.floor.main_light) is correct (Content 100 and Type is <class 'int'>). |

---

### 3.1.105 ViDevCommon.color\_temp (gfw.floor.main\_light) → Light.color\_temp (gfw.floor.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.105!

---

|             |                                                                                 |
|-------------|---------------------------------------------------------------------------------|
| Testrun:    | python3.13.5                                                                    |
| Caller:     | /home/dirk/work/smarthome_collection/smart_brain_test/report/___init___py (329) |
| Start-Time: | 2025-08-29 17:57:53,377                                                         |

---

Finished-Time: 2025-08-29 17:57:54,185

Time-Consumption 0.808s

**Testsummary:**

|                |                                                                                                      |
|----------------|------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                         |
| <b>Info</b>    | Prepare: Setting devices to last state 10                                                            |
| <b>Success</b> | Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).               |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (gfw.floor.main_light) to 0                                  |
| <b>Success</b> | Value for Light.color_temp (gfw.floor.main_light) is correct (Content 0 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (gfw.floor.main_light) to 2                                  |
| <b>Success</b> | Value for Light.color_temp (gfw.floor.main_light) is correct (Content 2 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (gfw.floor.main_light) to 4                                  |
| <b>Success</b> | Value for Light.color_temp (gfw.floor.main_light) is correct (Content 4 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (gfw.floor.main_light) to 6                                  |
| <b>Success</b> | Value for Light.color_temp (gfw.floor.main_light) is correct (Content 6 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (gfw.floor.main_light) to 8                                  |
| <b>Success</b> | Value for Light.color_temp (gfw.floor.main_light) is correct (Content 8 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.color_temp (gfw.floor.main_light) to 10                                 |
| <b>Success</b> | Value for Light.color_temp (gfw.floor.main_light) is correct (Content 10 and Type is <class 'int'>). |

**3.1.106 Light.color\_temp (gfw.floor.main\_light) → ViDevCommon.color\_temp (gfw.floor.main\_light)****Testresult**This test was passed with the state: **Success**. See also full trace in section A.1.106!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:54,185                                                        |
| Finished-Time:   | 2025-08-29 17:57:54,994                                                        |
| Time-Consumption | 0.809s                                                                         |

**Testsummary:**

|                |                                                                                                           |
|----------------|-----------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Switching on device                                                                              |
| <b>Info</b>    | Prepare: Setting devices to last state 10                                                                 |
| <b>Success</b> | Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).                    |
| <b>Info</b>    | Setting state of Light.color_temp (gfw.floor.main_light) to 0                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.floor.main_light) is correct (Content 0 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of Light.color_temp (gfw.floor.main_light) to 2                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.floor.main_light) is correct (Content 2 and Type is <class 'int'>). |
| <b>Info</b>    | Setting state of Light.color_temp (gfw.floor.main_light) to 4                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.floor.main_light) is correct (Content 4 and Type is <class 'int'>). |

|                |                                                                                                            |
|----------------|------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Setting state of Light.color_temp (gfw.floor.main_light) to 6                                              |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.floor.main_light) is correct (Content 6 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.color_temp (gfw.floor.main_light) to 8                                              |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.floor.main_light) is correct (Content 8 and Type is <class 'int'>).  |
| <b>Info</b>    | Setting state of Light.color_temp (gfw.floor.main_light) to 10                                             |
| <b>Success</b> | Value for ViDevCommon.color_temp (gfw.floor.main_light) is correct (Content 10 and Type is <class 'int'>). |

### 3.1.107 ViDevCommon.state (stw.stairway.main\_light) → Shelly.relay/0 (stw.firstfloor.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.107!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:54,994                                                        |
| Finished-Time:   | 2025-08-29 17:57:55,297                                                        |
| Time-Consumption | 0.303s                                                                         |

#### Testsummary:

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                |
| <b>Info</b>    | Setting state of ViDevCommon.state (stw.stairway.main_light) to True                                        |
| <b>Success</b> | Value for Shelly.relay/0 (stw.firstfloor.main_light) is correct (Content True and Type is <class 'bool'>).  |
| <b>Info</b>    | Setting state of ViDevCommon.state (stw.stairway.main_light) to False                                       |
| <b>Success</b> | Value for Shelly.relay/0 (stw.firstfloor.main_light) is correct (Content False and Type is <class 'bool'>). |

### 3.1.108 Shelly.relay/0 (stw.firstfloor.main\_light) → ViDevCommon.state (stw.stairway.main\_light)

#### Testresult

This test was passed with the state: **Success**. See also full trace in section A.1.108!

|                  |                                                                                |
|------------------|--------------------------------------------------------------------------------|
| Testrun:         | python3.13.5                                                                   |
| Caller:          | /home/dirk/work/smarthome_collection/smart_brain_test/report/__init__.py (329) |
| Start-Time:      | 2025-08-29 17:57:55,297                                                        |
| Finished-Time:   | 2025-08-29 17:57:55,600                                                        |
| Time-Consumption | 0.303s                                                                         |

#### Testsummary:

|                |                                                                                                             |
|----------------|-------------------------------------------------------------------------------------------------------------|
| <b>Info</b>    | Prepare: Setting devices to last state False                                                                |
| <b>Success</b> | Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).                |
| <b>Info</b>    | Setting state of Shelly.relay/0 (stw.firstfloor.main_light) to True                                         |
| <b>Success</b> | Value for ViDevCommon.state (stw.stairway.main_light) is correct (Content True and Type is <class 'bool'>). |

**Info**

**Success**

Setting state of Shelly.relay/0 (stw.firstfloor.main\_light) to False

Value for ViDevCommon.state (stw.stairway.main\_light) is correct (Content False and Type is <class 'bool'>).

---

## A Trace for testrun with python3.13.5

### A.1 Tests with status Info (108)

#### A.1.1 Clean-Up

##### Testresult

This test was passed with the state: **Info**.

| Info                                                                                                                         | Collecting precondition logs... |
|------------------------------------------------------------------------------------------------------------------------------|---------------------------------|
| Sending message with topic videv/all/oof and payload True                                                                    |                                 |
| Received message with topic videv/ffe/kitchen/circulation_pump/state and payload b'false'                                    |                                 |
| Received message with topic zigbee_ffe/ffe/kitchen/heating_valve/set and payload<br>↪ b'{"current_heating_setpoint": 20}'    |                                 |
| Received message with topic videv/ffe/kitchen/heating_valve/valve_temperature_setpoint and<br>↪ payload b'20'                |                                 |
| Received message with topic videv/ffe/kitchen/heating_valve/user_temperature_setpoint and<br>↪ payload b'20'                 |                                 |
| Received message with topic videv/ffe/livingroom/main_light/state and payload b'false'                                       |                                 |
| Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1/set and payload<br>↪ b'{"state": "off"}'                 |                                 |
| Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2/set and payload<br>↪ b'{"state": "off"}'                 |                                 |
| Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3/set and payload<br>↪ b'{"state": "off"}'                 |                                 |
| Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4/set and payload<br>↪ b'{"state": "off"}'                 |                                 |
| Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5/set and payload<br>↪ b'{"state": "off"}'                 |                                 |
| Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6/set and payload<br>↪ b'{"state": "off"}'                 |                                 |
| Received message with topic zigbee_ffe/ffe/livingroom/heating_valve/set and payload<br>↪ b'{"current_heating_setpoint": 20}' |                                 |
| Received message with topic videv/ffe/livingroom/heating_valve/valve_temperature_setpoint and<br>↪ payload b'20'             |                                 |
| Received message with topic videv/ffe/livingroom/heating_valve/user_temperature_setpoint and<br>↪ payload b'20'              |                                 |
| Received message with topic videv/ffe/livingroom/xmas_tree/state and payload b'false'                                        |                                 |
| Received message with topic videv/ffe/sleep/main_light/state and payload b'false'                                            |                                 |
| Received message with topic __info__ and payload b'null'                                                                     |                                 |
| Received message with topic videv/ffe/sleep/bed_light_ma/state and payload b'false'                                          |                                 |
| Received message with topic zigbee_ffe/ffe/sleep/heating_valve/set and payload<br>↪ b'{"current_heating_setpoint": 20}'      |                                 |

Received message with topic videv/ffe/sleep/heating\_valve/valve\_temperature\_setpoint and  
↪ payload b'20'

Received message with topic videv/ffe/sleep/heating\_valve/user\_temperature\_setpoint and  
↪ payload b'20'

Received message with topic videv/ffw/bath/main\_light/state and payload b'false'

Received message with topic zigbee\_ffw/ffw/bath/heating\_valve/set and payload  
↪ b'{"current\_heating\_setpoint": 20}'

Received message with topic videv/ffw/bath/heating\_valve/valve\_temperature\_setpoint and  
↪ payload b'20'

Received message with topic videv/ffw/bath/heating\_valve/user\_temperature\_setpoint and payload  
↪ b'20'

Received message with topic videv/ffw/floor/main\_light/state and payload b'false'

Received message with topic videv/ffw/julian/main\_light/state and payload b'false'

Received message with topic zigbee\_ffw/ffw/julian/heating\_valve/set and payload  
↪ b'{"current\_heating\_setpoint": 20}'

Received message with topic videv/ffw/julian/heating\_valve/valve\_temperature\_setpoint and  
↪ payload b'20'

Received message with topic videv/ffw/julian/heating\_valve/user\_temperature\_setpoint and  
↪ payload b'20'

Received message with topic videv/ffw/livingroom/main\_light/state and payload b'false'

Received message with topic zigbee\_ffw/ffw/livingroom/heating\_valve/set and payload  
↪ b'{"current\_heating\_setpoint": 20}'

Received message with topic videv/ffw/livingroom/heating\_valve/valve\_temperature\_setpoint and  
↪ payload b'20'

Received message with topic videv/ffw/livingroom/heating\_valve/user\_temperature\_setpoint and  
↪ payload b'20'

Received message with topic videv/ffw/sleep/main\_light/state and payload b'false'

Received message with topic zigbee\_ffw/ffw/sleep/window\_light/set and payload b'{"state":  
↪ "off"}'

Received message with topic zigbee\_ffw/ffw/sleep/heating\_valve/set and payload  
↪ b'{"current\_heating\_setpoint": 20}'

Received message with topic videv/ffw/sleep/heating\_valve/valve\_temperature\_setpoint and  
↪ payload b'20'

Received message with topic videv/ffw/sleep/heating\_valve/user\_temperature\_setpoint and  
↪ payload b'20'

Received message with topic videv/gfw/dirk/main\_light/state and payload b'false'

Received message with topic videv/gfw/dirk/amplifier/state and payload b'false'

Received message with topic videv/gfw/dirk/phono/state and payload b'false'

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'false'

Received message with topic videv/gfw/dirk/cd\_player/state and payload b'false'

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'false'

Received message with topic videv/gfw/dirk/bt/state and payload b'false'

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'false'

```

Received message with topic videv/gfw/dirk/pc_dock/state and payload b'false'
Received message with topic zigbee_gfw/gfw/dirk/heating_valve/set and payload
↪ b'{"current_heating_setpoint": 20}'
Received message with topic videv/gfw/dirk/heating_valve/valve_temperature_setpoint and
↪ payload b'20'
Received message with topic videv/gfw/dirk/heating_valve/user_temperature_setpoint and payload
↪ b'20'
Received message with topic videv/gfw/floor/main_light/state and payload b'false'
Received message with topic videv/gfw/marion/main_light/state and payload b'false'
Received message with topic zigbee_gfw/gfw/marion/window_light/set and payload b'{"state":
↪ "off"}'
Received message with topic zigbee_gfw/gfw/marion/heating_valve/set and payload
↪ b'{"current_heating_setpoint": 20}'
Received message with topic videv/gfw/marion/heating_valve/valve_temperature_setpoint and
↪ payload b'20'
Received message with topic videv/gfw/marion/heating_valve/user_temperature_setpoint and
↪ payload b'20'
Received message with topic videv/stw/stairway/main_light/state and payload b'false'
Received message with topic __info__ and payload b'{"app_name": "smart_brain", "version":
↪ {"readable": "1.3.2", "major": 1, "minor": 3, "patch": 2}, "git": {"url":
↪ "https://git.mount-mockery.de/smarthome/smart_brain.git", "ref":
↪ "9c0aff5244690dd8d20697652c8651cdcb805863"}}'

```

### A.1.2 ViDevCommon.state (ffe.livingroom.main\_light) → Shelly.relay/0 (ffe.livingroom.main\_light)

#### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffe/livingroom/main\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (ffe.livingroom.main\_light) to True

---

Sending message with topic videv/ffe/livingroom/main\_light/state/set and payload true

Received message with topic shellies/ffe/livingroom/main\_light/relay/0/command and payload  
↪ b'on'

Sending message with topic shellies/ffe/livingroom/main\_light/relay/0 and payload on

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

```

Received message with topic shellies/ffe/livingroom/main_light/relay/0 and payload b'on'
Received message with topic zigbee_ffe/ffe/livingroom/main_light and payload b'{"state": "on",
↪  "brightness": 127.0, "color_temp": 352.0}'
Received message with topic videv/ffe/livingroom/main_light/state and payload b'true'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1/set and payload
↪  b'{"state": "on"}'
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload {"state": "on",
↪  "brightness": 127.0, "color_temp": 352.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2/set and payload
↪  b'{"state": "on"}'
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload {"state": "on",
↪  "brightness": 127.0, "color_temp": 352.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3/set and payload
↪  b'{"state": "on"}'
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload {"state": "on",
↪  "brightness": 127.0, "color_temp": 352.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4/set and payload
↪  b'{"state": "on"}'
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload {"state": "on",
↪  "brightness": 127.0, "color_temp": 352.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5/set and payload
↪  b'{"state": "on"}'
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload {"state": "on",
↪  "brightness": 127.0, "color_temp": 352.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6/set and payload
↪  b'{"state": "on"}'
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload {"state": "on",
↪  "brightness": 127.0, "color_temp": 352.0}
Received message with topic videv/ffe/livingroom/main_light/brightness and payload b'50'
Received message with topic videv/ffe/livingroom/main_light/color_temp and payload b'5'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload b'{"state":
↪  "on", "brightness": 127.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload b'{"state":
↪  "on", "brightness": 127.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload b'{"state":
↪  "on", "brightness": 127.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload b'{"state":
↪  "on", "brightness": 127.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload b'{"state":
↪  "on", "brightness": 127.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload b'{"state":
↪  "on", "brightness": 127.0, "color_temp": 352.0}'

```

---

**Success** Value for Shelly.relay/0 (ffe.livingroom.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (ffe.livingroom.main\_light)): True (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (ffe.livingroom.main\_light)): result = True (<class 'bool'>)  
↪ 'bool'>)

---

**Info** Setting state of ViDevCommon.state (ffe.livingroom.main\_light) to False

---

Sending message with topic videv/ffe/livingroom/main\_light/state/set and payload false

Received message with topic videv/ffe/livingroom/floorlamp/state and payload b'true'

Received message with topic videv/ffe/livingroom/floorlamp/brightness and payload b'50'

Received message with topic videv/ffe/livingroom/floorlamp/color\_temp and payload b'5'

Received message with topic shellies/ffe/livingroom/main\_light/relay/0/command and payload  
↪ b'off'

Sending message with topic shellies/ffe/livingroom/main\_light/relay/0 and payload off

Received message with topic shellies/ffe/livingroom/main\_light/relay/0 and payload b'off'

---

**Success** Value for Shelly.relay/0 (ffe.livingroom.main\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (ffe.livingroom.main\_light)): False (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (ffe.livingroom.main\_light)): result = False (<class 'bool'>)  
↪ 'bool'>)

---

### A.1.3 Shelly.relay/0 (ffe.livingroom.main\_light) → ViDevCommon.state (ffe.livingroom.main\_light)

#### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffe/livingroom/main\_light/state/set and payload false

Received message with topic videv/ffe/livingroom/main\_light/state and payload b'false'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload  
↪ b'{"state": "off"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload  
↪ b'{"state": "off"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3/set and payload  
↪ b'{"state": "off"}'

---

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4/set and payload  
↪ b'{"state": "off"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5/set and payload  
↪ b'{"state": "off"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6/set and payload  
↪ b'{"state": "off"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Shelly.relay/0 (ffe.livingroom.main\_light) to True

---

Sending message with topic shellies/ffe/livingroom/main\_light/relay/0 and payload on

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic shellies/ffe/livingroom/main\_light/relay/0 and payload b'on'

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/floorlamp/state and payload b'false'

Received message with topic videv/ffe/livingroom/main\_light/state and payload b'true'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload  
↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload  
↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3/set and payload  
↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4/set and payload  
↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5/set and payload  
↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6/set and payload  
↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/floorlamp/state and payload b'true'

---

**Success** Value for ViDevCommon.state (ffe.livingroom.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffe.livingroom.main\_light)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.livingroom.main\_light)): result = True (<class  
↪ 'bool'>)

---

**Info** Setting state of Shelly.relay/0 (ffe.livingroom.main\_light) to False

---

Sending message with topic shellies/ffe/livingroom/main\_light/relay/0 and payload off

Received message with topic shellies/ffe/livingroom/main\_light/relay/0 and payload b'off'

Received message with topic videv/ffe/livingroom/main\_light/state and payload b'false'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload  
↪ b'{"state": "off"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload  
↪ b'{"state": "off"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3/set and payload  
↪ b'{"state": "off"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4/set and payload  
↪ b'{"state": "off"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5/set and payload  
↪ b'{"state": "off"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6/set and payload  
↪ b'{"state": "off"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/floorlamp/state and payload b'false'

---

**Success** Value for ViDevCommon.state (ffe.livingroom.main\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffe.livingroom.main\_light)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.livingroom.main\_light)): result = False (<class  
↪ 'bool'>)

**A.1.4 ViDevCommon.state (ffe.livingroom.floorlamp) → Light.state (ffe.livingroom.floor\_light)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffe/livingroom/floorlamp/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (ffe.livingroom.floorlamp) to True

---

Sending message with topic videv/ffe/livingroom/floorlamp/state/set and payload true

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload

↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",

↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload

↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",

↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":

↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3/set and payload

↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",

↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4/set and payload

↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",

↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5/set and payload

↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",

↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6/set and payload

↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",

↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":

↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload b'{"state":
↪ "on", "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload b'{"state":
↪ "on", "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload b'{"state":
↪ "on", "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload b'{"state":
↪ "on", "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic videv/ffe/livingroom/floorlamp/state and payload b'true'
```

---

**Success** Value for Light.state (ffe.livingroom.floor\_light) is correct (Content True and Type is <class 'bool'>).

---

```
Result (Value for Light.state (ffe.livingroom.floor_light)): True (<class 'bool'>)
```

```
Expectation (Value for Light.state (ffe.livingroom.floor_light)): result = True (<class
↪ 'bool'>)
```

---

**Info** Setting state of ViDevCommon.state (ffe.livingroom.floorlamp) to False

---

```
Sending message with topic videv/ffe/livingroom/floorlamp/state/set and payload false
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1/set and payload
↪ b'{"state": "off"}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload {"state":
↪ "off", "brightness": 127.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2/set and payload
↪ b'{"state": "off"}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload {"state":
↪ "off", "brightness": 127.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3/set and payload
↪ b'{"state": "off"}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload {"state":
↪ "off", "brightness": 127.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4/set and payload
↪ b'{"state": "off"}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload {"state":
↪ "off", "brightness": 127.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5/set and payload
↪ b'{"state": "off"}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload {"state":
↪ "off", "brightness": 127.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6/set and payload
↪ b'{"state": "off"}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload {"state":
↪ "off", "brightness": 127.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload b'{"state":
↪ "off", "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload b'{"state":
↪ "off", "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload b'{"state":
↪ "off", "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload b'{"state":
↪ "off", "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload b'{"state":
↪ "off", "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload b'{"state":
↪ "off", "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic videv/ffe/livingroom/floorlamp/state and payload b'false'
```

---

**Success** Value for Light.state (ffe.livingroom.floor\_light) is correct (Content False and Type is <class 'bool'>).

---

```
Result (Value for Light.state (ffe.livingroom.floor_light)): False (<class 'bool'>)
```

```
Expectation (Value for Light.state (ffe.livingroom.floor_light)): result = False (<class
↪ 'bool'>)
```

#### A.1.5 Light.state (ffe.livingroom.floor\_light) → ViDevCommon.state (ffe.livingroom.floorlamp)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

```
Sending message with topic videv/ffe/livingroom/floorlamp/state/set and payload false
```

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

```
Result (Start state (master, slave)): (False, False) (<class 'tuple'>)
```

```
Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)
```

---

**Info** Setting state of Light.state (ffe.livingroom.floor\_light) to True

---

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload {"state": "on",
↪ "brightness": 127.0, "color_temp": 352.0}
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload {"state": "on",
↪ "brightness": 127.0, "color_temp": 352.0}
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload {"state": "on",
↪ "brightness": 127.0, "color_temp": 352.0}
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload {"state": "on",
↪ "brightness": 127.0, "color_temp": 352.0}
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload {"state": "on",
↪ "brightness": 127.0, "color_temp": 352.0}
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload {"state": "on",
↪ "brightness": 127.0, "color_temp": 352.0}
```

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/floorlamp/state and payload b'true'

---

**Success** Value for ViDevCommon.state (ffe.livingroom.floorlamp) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffe.livingroom.floorlamp)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.livingroom.floorlamp)): result = True (<class 'bool'>)

---

**Info** Setting state of Light.state (ffe.livingroom.floor\_light) to False

---

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "off", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/floorlamp/state and payload b'false'

**Success** Value for ViDevCommon.state (ffe.livingroom.floorlamp) is correct (Content False and Type is <class 'bool'>).

Result (Value for ViDevCommon.state (ffe.livingroom.floorlamp)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.livingroom.floorlamp)): result = False (<class  
↪ 'bool'>)

### A.1.6 Shelly.relay/0 (ffe.livingroom.main\_light) → Light.state (ffe.livingroom.floor\_light)

#### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Setting devices to last state False

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

**Info** Setting state of Shelly.relay/0 (ffe.livingroom.main\_light) to True

Sending message with topic shellies/ffe/livingroom/main\_light/relay/0 and payload on

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic shellies/ffe/livingroom/main\_light/relay/0 and payload b'on'

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/main\_light/state and payload b'true'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload  
↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload  
↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3/set and payload  
↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4/set and payload

↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",

↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5/set and payload

↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",

↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6/set and payload

↪ b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",

↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":

↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":

↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":

↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":

↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":

↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":

↪ "on", "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/floorlamp/state and payload b'true'

---

**Success** Value for Light.state (ffe.livingroom.floor\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Light.state (ffe.livingroom.floor\_light)): True (<class 'bool'>)

Expectation (Value for Light.state (ffe.livingroom.floor\_light)): result = True (<class

↪ 'bool'>)

---

**Info** Setting state of Shelly.relay/0 (ffe.livingroom.main\_light) to False

---

Sending message with topic shellies/ffe/livingroom/main\_light/relay/0 and payload off

Received message with topic shellies/ffe/livingroom/main\_light/relay/0 and payload b'off'

Received message with topic videv/ffe/livingroom/main\_light/state and payload b'false'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload

↪ b'{"state": "off"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state":

↪ "off", "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload

↪ b'{"state": "off"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state":

↪ "off", "brightness": 127.0, "color\_temp": 352.0}

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3/set and payload
↳ b'{"state": "off"}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload {"state":
↳ "off", "brightness": 127.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4/set and payload
↳ b'{"state": "off"}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload {"state":
↳ "off", "brightness": 127.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5/set and payload
↳ b'{"state": "off"}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload {"state":
↳ "off", "brightness": 127.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6/set and payload
↳ b'{"state": "off"}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload {"state":
↳ "off", "brightness": 127.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload b'{"state":
↳ "off", "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload b'{"state":
↳ "off", "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload b'{"state":
↳ "off", "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload b'{"state":
↳ "off", "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload b'{"state":
↳ "off", "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload b'{"state":
↳ "off", "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic videv/ffe/livingroom/floorlamp/state and payload b'false'
```

---

**Success** Value for Light.state (ffe.livingroom.floor\_light) is correct (Content False and Type is <class 'bool'>).

---

```
Result (Value for Light.state (ffe.livingroom.floor_light)): False (<class 'bool'>)
```

```
Expectation (Value for Light.state (ffe.livingroom.floor_light)): result = False (<class
↳ 'bool'>)
```

### A.1.7 ViDevCommon.state (ffe.livingroom.xmas\_tree) → Powerplug1P.state (ffe.livingroom.xmas-tree)

#### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

```
Sending message with topic videv/ffe/livingroom/xmas_tree/state/set and payload false
```

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (ffe.livingroom.xmas\_tree) to True

---

Sending message with topic videv/ffe/livingroom/xmas\_tree/state/set and payload true

Received message with topic zigbee\_ffe/ffe/livingroom/xmas-tree/set and payload b'{"state":  
↪ "on"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/xmas-tree and payload {"state": "on"}

Received message with topic zigbee\_ffe/ffe/livingroom/xmas-tree and payload b'{"state": "on"}'

Received message with topic videv/ffe/livingroom/xmas\_tree/state and payload b'true'

---

**Success** Value for Powerplug1P.state (ffe.livingroom.xmas-tree) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Powerplug1P.state (ffe.livingroom.xmas-tree)): True (<class 'bool'>)

Expectation (Value for Powerplug1P.state (ffe.livingroom.xmas-tree)): result = True (<class  
↪ 'bool'>)

---

**Info** Setting state of ViDevCommon.state (ffe.livingroom.xmas\_tree) to False

---

Sending message with topic videv/ffe/livingroom/xmas\_tree/state/set and payload false

Received message with topic zigbee\_ffe/ffe/livingroom/xmas-tree/set and payload b'{"state":  
↪ "off"}'

Sending message with topic zigbee\_ffe/ffe/livingroom/xmas-tree and payload {"state": "off"}

Received message with topic zigbee\_ffe/ffe/livingroom/xmas-tree and payload b'{"state":  
↪ "off"}'

Received message with topic videv/ffe/livingroom/xmas\_tree/state and payload b'false'

---

**Success** Value for Powerplug1P.state (ffe.livingroom.xmas-tree) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Powerplug1P.state (ffe.livingroom.xmas-tree)): False (<class 'bool'>)

Expectation (Value for Powerplug1P.state (ffe.livingroom.xmas-tree)): result = False (<class  
↪ 'bool'>)

---

#### A.1.8 Powerplug1P.state (ffe.livingroom.xmas-tree) → ViDevCommon.state (ffe.livingroom.xmas\_tree)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffe/livingroom/xmas\_tree/state/set and payload false

---

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Powerplug1P.state (ffe.livingroom.xmas-tree) to True

---

Sending message with topic zigbee\_ffe/ffe/livingroom/xmas-tree and payload {"state": "on"}

Received message with topic zigbee\_ffe/ffe/livingroom/xmas-tree and payload b'{"state": "on"}'

Received message with topic videv/ffe/livingroom/xmas\_tree/state and payload b'true'

---

**Success** Value for ViDevCommon.state (ffe.livingroom.xmas\_tree) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffe.livingroom.xmas\_tree)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.livingroom.xmas\_tree)): result = True (<class 'bool'>)  
↪ 'bool'>)

---

**Info** Setting state of Powerplug1P.state (ffe.livingroom.xmas-tree) to False

---

Sending message with topic zigbee\_ffe/ffe/livingroom/xmas-tree and payload {"state": "off"}

Received message with topic zigbee\_ffe/ffe/livingroom/xmas-tree and payload b'{"state":  
↪ "off"}'

Received message with topic videv/ffe/livingroom/xmas\_tree/state and payload b'false'

---

**Success** Value for ViDevCommon.state (ffe.livingroom.xmas\_tree) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffe.livingroom.xmas\_tree)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.livingroom.xmas\_tree)): result = False (<class 'bool'>)  
↪ 'bool'>)

---

#### A.1.9 ViDevCommon.brightness (ffe.livingroom.main\_light) → Light.brightness (ffe.livingroom.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---

Sending message with topic shellies/ffe/livingroom/main\_light/relay/0 and payload on

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic shellies/ffe/livingroom/main\_light/relay/0 and payload b'on'

---

```

Received message with topic zigbee_ffe/ffe/livingroom/main_light and payload b'{"state": "on",
↳ "brightness": 127.0, "color_temp": 352.0}'

Received message with topic videv/ffe/livingroom/main_light/state and payload b'true'

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1/set and payload
↳ b'{"state": "on"}'

Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload {"state": "on",
↳ "brightness": 127.0, "color_temp": 352.0}

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2/set and payload
↳ b'{"state": "on"}'

Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload {"state": "on",
↳ "brightness": 127.0, "color_temp": 352.0}

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3/set and payload
↳ b'{"state": "on"}'

Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload {"state": "on",
↳ "brightness": 127.0, "color_temp": 352.0}

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4/set and payload
↳ b'{"state": "on"}'

Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload {"state": "on",
↳ "brightness": 127.0, "color_temp": 352.0}

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5/set and payload
↳ b'{"state": "on"}'

Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload {"state": "on",
↳ "brightness": 127.0, "color_temp": 352.0}

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6/set and payload
↳ b'{"state": "on"}'

Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload {"state": "on",
↳ "brightness": 127.0, "color_temp": 352.0}

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload b'{"state":
↳ "on", "brightness": 127.0, "color_temp": 352.0}'

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload b'{"state":
↳ "on", "brightness": 127.0, "color_temp": 352.0}'

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload b'{"state":
↳ "on", "brightness": 127.0, "color_temp": 352.0}'

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload b'{"state":
↳ "on", "brightness": 127.0, "color_temp": 352.0}'

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload b'{"state":
↳ "on", "brightness": 127.0, "color_temp": 352.0}'

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload b'{"state":
↳ "on", "brightness": 127.0, "color_temp": 352.0}'

Received message with topic videv/ffe/livingroom/floorlamp/state and payload b'true'

```

---

**Info** Prepare: Setting devices to last state 100

---

```

Sending message with topic videv/ffe/livingroom/main_light/brightness/set and payload 100

```

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light/set and payload  
↪ b'{"brightness": 254}'

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/main\_light/brightness and payload b'100'

---

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.brightness (ffe.livingroom.main\_light) to 0

---

Sending message with topic videv/ffe/livingroom/main\_light/brightness/set and payload 0

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light/set and payload  
↪ b'{"brightness": 1}'

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/main\_light/brightness and payload b'0'

---

**Success** Value for Light.brightness (ffe.livingroom.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffe.livingroom.main\_light)): 0 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.livingroom.main\_light)): result = 0 (<class  
↪ 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffe.livingroom.main\_light) to 20

---

Sending message with topic videv/ffe/livingroom/main\_light/brightness/set and payload 20

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light/set and payload  
↪ b'{"brightness": 52}'

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/main\_light/brightness and payload b'20'

---

**Success** Value for Light.brightness (ffe.livingroom.main\_light) is correct (Content 20 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffe.livingroom.main\_light)): 20 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.livingroom.main\_light)): result = 20 (<class  
↪ 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffe.livingroom.main\_light) to 40

---

Sending message with topic videv/ffe/livingroom/main\_light/brightness/set and payload 40

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light/set and payload

↪ b'{"brightness": 102}'

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",

↪ "brightness": 102.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",

↪ "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/main\_light/brightness and payload b'40'

---

**Success** Value for Light.brightness (ffe.livingroom.main\_light) is correct (Content 40 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffe.livingroom.main\_light)): 40 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.livingroom.main\_light)): result = 40 (<class

↪ 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffe.livingroom.main\_light) to 60

---

Sending message with topic videv/ffe/livingroom/main\_light/brightness/set and payload 60

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light/set and payload

↪ b'{"brightness": 153}'

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",

↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",

↪ "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/main\_light/brightness and payload b'60'

---

**Success** Value for Light.brightness (ffe.livingroom.main\_light) is correct (Content 60 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffe.livingroom.main\_light)): 60 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.livingroom.main\_light)): result = 60 (<class

↪ 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffe.livingroom.main\_light) to 80

---

Sending message with topic videv/ffe/livingroom/main\_light/brightness/set and payload 80

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light/set and payload

↪ b'{"brightness": 203}'

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",

↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",

↪ "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/main\_light/brightness and payload b'80'

**Success** Value for Light.brightness (ffe.livingroom.main\_light) is correct (Content 80 and Type is <class 'int'>).

Result (Value for Light.brightness (ffe.livingroom.main\_light)): 80 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.livingroom.main\_light)): result = 80 (<class 'int'>)  
↪ 'int'>)

**Info** Setting state of ViDevCommon.brightness (ffe.livingroom.main\_light) to 100

Sending message with topic videv/ffe/livingroom/main\_light/brightness/set and payload 100

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light/set and payload  
↪ b'{"brightness": 254}'

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/main\_light/brightness and payload b'100'

**Success** Value for Light.brightness (ffe.livingroom.main\_light) is correct (Content 100 and Type is <class 'int'>).

Result (Value for Light.brightness (ffe.livingroom.main\_light)): 100 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.livingroom.main\_light)): result = 100 (<class 'int'>)  
↪ 'int'>)

#### A.1.10 Light.brightness (ffe.livingroom.main\_light) → ViDevCommon.brightness (ffe.livingroom.main\_light)

##### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Switching on device

**Info** Prepare: Setting devices to last state 100

Sending message with topic videv/ffe/livingroom/main\_light/brightness/set and payload 100

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

**Info** Setting state of Light.brightness (ffe.livingroom.main\_light) to 0

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/main\_light/brightness and payload b'0'

**Success** Value for ViDevCommon.brightness (ffe.livingroom.main\_light) is correct (Content 0 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffe.livingroom.main\_light)): 0 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.livingroom.main\_light)): result = 0 (<class  
↪ 'int'>)

**Info** Setting state of Light.brightness (ffe.livingroom.main\_light) to 20

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/main\_light/brightness and payload b'20'

**Success** Value for ViDevCommon.brightness (ffe.livingroom.main\_light) is correct (Content 20 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffe.livingroom.main\_light)): 20 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.livingroom.main\_light)): result = 20  
↪ (<class 'int'>)

**Info** Setting state of Light.brightness (ffe.livingroom.main\_light) to 40

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/main\_light/brightness and payload b'40'

**Success** Value for ViDevCommon.brightness (ffe.livingroom.main\_light) is correct (Content 40 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffe.livingroom.main\_light)): 40 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.livingroom.main\_light)): result = 40  
↪ (<class 'int'>)

**Info** Setting state of Light.brightness (ffe.livingroom.main\_light) to 60

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/main\_light/brightness and payload b'60'

**Success** Value for ViDevCommon.brightness (ffe.livingroom.main\_light) is correct (Content 60 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffe.livingroom.main\_light)): 60 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.livingroom.main\_light)): result = 60  
↪ (<class 'int'>)

**Info** Setting state of Light.brightness (ffe.livingroom.main\_light) to 80

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/main\_light/brightness and payload b'80'

**Success** Value for ViDevCommon.brightness (ffe.livingroom.main\_light) is correct (Content 80 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffe.livingroom.main\_light)): 80 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.livingroom.main\_light)): result = 80  
↪ (<class 'int'>)

**Info** Setting state of Light.brightness (ffe.livingroom.main\_light) to 100

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/main\_light/brightness and payload b'100'

**Success** Value for ViDevCommon.brightness (ffe.livingroom.main\_light) is correct (Content 100 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffe.livingroom.main\_light)): 100 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.livingroom.main\_light)): result = 100  
↪ (<class 'int'>)

#### A.1.11 ViDevCommon.color\_temp (ffe.livingroom.main\_light) → Light.color\_temp (ffe.livingroom.main\_light)

##### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Switching on device

**Info** Prepare: Setting devices to last state 10

Sending message with topic videv/ffe/livingroom/main\_light/color\_temp/set and payload 10

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light/set and payload  
↪ b'{"color\_temp": 454}'

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/ffe/livingroom/main\_light/color\_temp and payload b'10'

---

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (10, 10) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (10, 10) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffe.livingroom.main\_light) to 0

---

Sending message with topic videv/ffe/livingroom/main\_light/color\_temp/set and payload 0

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light/set and payload  
↪ b'{"color\_temp": 250}'

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic videv/ffe/livingroom/main\_light/color\_temp and payload b'0'

---

**Success** Value for Light.color\_temp (ffe.livingroom.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffe.livingroom.main\_light)): 0 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffe.livingroom.main\_light)): result = 0 (<class  
↪ 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffe.livingroom.main\_light) to 2

---

Sending message with topic videv/ffe/livingroom/main\_light/color\_temp/set and payload 2

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light/set and payload  
↪ b'{"color\_temp": 291}'

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic videv/ffe/livingroom/main\_light/color\_temp and payload b'2'

---

**Success** Value for Light.color\_temp (ffe.livingroom.main\_light) is correct (Content 2 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffe.livingroom.main\_light)): 2 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffe.livingroom.main\_light)): result = 2 (<class  
↪ 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffe.livingroom.main\_light) to 4

---

Sending message with topic videv/ffe/livingroom/main\_light/color\_temp/set and payload 4

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light/set and payload

↪ b'{"color\_temp": 332}'

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",

↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",

↪ "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic videv/ffe/livingroom/main\_light/color\_temp and payload b'4'

---

**Success** Value for Light.color\_temp (ffe.livingroom.main\_light) is correct (Content 4 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffe.livingroom.main\_light)): 4 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffe.livingroom.main\_light)): result = 4 (<class

↪ 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffe.livingroom.main\_light) to 6

---

Sending message with topic videv/ffe/livingroom/main\_light/color\_temp/set and payload 6

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light/set and payload

↪ b'{"color\_temp": 372}'

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",

↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",

↪ "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic videv/ffe/livingroom/main\_light/color\_temp and payload b'6'

---

**Success** Value for Light.color\_temp (ffe.livingroom.main\_light) is correct (Content 6 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffe.livingroom.main\_light)): 6 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffe.livingroom.main\_light)): result = 6 (<class

↪ 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffe.livingroom.main\_light) to 8

---

Sending message with topic videv/ffe/livingroom/main\_light/color\_temp/set and payload 8

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light/set and payload

↪ b'{"color\_temp": 413}'

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",

↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",

↪ "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic videv/ffe/livingroom/main\_light/color\_temp and payload b'8'

**Success** Value for Light.color\_temp (ffe.livingroom.main\_light) is correct (Content 8 and Type is <class 'int'>).

Result (Value for Light.color\_temp (ffe.livingroom.main\_light)): 8 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffe.livingroom.main\_light)): result = 8 (<class 'int'>)  
↪ 'int'>)

**Info** Setting state of ViDevCommon.color\_temp (ffe.livingroom.main\_light) to 10

Sending message with topic videv/ffe/livingroom/main\_light/color\_temp/set and payload 10

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light/set and payload  
↪ b'{"color\_temp": 454}'

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/ffe/livingroom/main\_light/color\_temp and payload b'10'

**Success** Value for Light.color\_temp (ffe.livingroom.main\_light) is correct (Content 10 and Type is <class 'int'>).

Result (Value for Light.color\_temp (ffe.livingroom.main\_light)): 10 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffe.livingroom.main\_light)): result = 10 (<class 'int'>)  
↪ 'int'>)

#### A.1.12 Light.color\_temp (ffe.livingroom.main\_light) → ViDevCommon.color\_temp (ffe.livingroom.main\_light)

##### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Switching on device

**Info** Prepare: Setting devices to last state 10

Sending message with topic videv/ffe/livingroom/main\_light/color\_temp/set and payload 10

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (10, 10) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (10, 10) (<class 'tuple'>)

**Info** Setting state of Light.color\_temp (ffe.livingroom.main\_light) to 0

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic videv/ffe/livingroom/main\_light/color\_temp and payload b'0'

---

**Success** Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light)): 0 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light)): result = 0 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (ffe.livingroom.main\_light) to 2

---

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic videv/ffe/livingroom/main\_light/color\_temp and payload b'2'

---

**Success** Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light) is correct (Content 2 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light)): 2 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light)): result = 2 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (ffe.livingroom.main\_light) to 4

---

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic videv/ffe/livingroom/main\_light/color\_temp and payload b'4'

---

**Success** Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light) is correct (Content 4 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light)): 4 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light)): result = 4 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (ffe.livingroom.main\_light) to 6

---

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic videv/ffe/livingroom/main\_light/color\_temp and payload b'6'

**Success** Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light) is correct (Content 6 and Type is <class 'int'>).

Result (Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light)): 6 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light)): result = 6 (<class 'int'>)  
↪ 'int'>)

**Info** Setting state of Light.color\_temp (ffe.livingroom.main\_light) to 8

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic videv/ffe/livingroom/main\_light/color\_temp and payload b'8'

**Success** Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light) is correct (Content 8 and Type is <class 'int'>).

Result (Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light)): 8 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light)): result = 8 (<class 'int'>)  
↪ 'int'>)

**Info** Setting state of Light.color\_temp (ffe.livingroom.main\_light) to 10

Sending message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_ffe/ffe/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/ffe/livingroom/main\_light/color\_temp and payload b'10'

**Success** Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light) is correct (Content 10 and Type is <class 'int'>).

Result (Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light)): 10 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.livingroom.main\_light)): result = 10  
↪ (<class 'int'>)

### A.1.13 ViDevCommon.brightness (ffe.livingroom.floorlamp) → Light.brightness (ffe.livingroom.floor\_light)

#### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Switching on device

**Info** Prepare: Setting devices to last state 100

Sending message with topic videv/ffe/livingroom/floorlamp/brightness/set and payload 100

```

Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 352.0}
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 352.0}
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 352.0}
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 352.0}
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 352.0}
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 352.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1/set and payload
↪  b'{"brightness": 254}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2/set and payload
↪  b'{"brightness": 254}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3/set and payload
↪  b'{"brightness": 254}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload b'{"state":
↪  "on", "brightness": 254.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4/set and payload
↪  b'{"brightness": 254}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload b'{"state":
↪  "on", "brightness": 254.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5/set and payload
↪  b'{"brightness": 254}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload b'{"state":
↪  "on", "brightness": 254.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6/set and payload
↪  b'{"brightness": 254}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload b'{"state":
↪  "on", "brightness": 254.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload b'{"state":
↪  "on", "brightness": 254.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload b'{"state":
↪  "on", "brightness": 254.0, "color_temp": 352.0}'
Received message with topic videv/ffe/livingroom/floorlamp/brightness and payload b'100'

```

---

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.brightness (ffe.livingroom.floorlamp) to 0

---

Sending message with topic videv/ffe/livingroom/floorlamp/brightness/set and payload 0

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload

↪ b'{"brightness": 1}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",

↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload

↪ b'{"brightness": 1}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",

↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3/set and payload

↪ b'{"brightness": 1}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",

↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4/set and payload

↪ b'{"brightness": 1}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",

↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5/set and payload

↪ b'{"brightness": 1}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",

↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6/set and payload

↪ b'{"brightness": 1}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",

↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":

↪ "on", "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":

↪ "on", "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":

↪ "on", "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":

↪ "on", "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":

↪ "on", "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":

↪ "on", "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/floorlamp/brightness and payload b'0'

---

**Success** Value for Light.brightness (ffe.livingroom.floor\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffe.livingroom.floor\_light)): 0 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.livingroom.floor\_light)): result = 0 (<class 'int'>)

**Info** Setting state of ViDevCommon.brightness (ffe.livingroom.floorlamp) to 20

```

Sending message with topic videv/ffe/livingroom/floorlamp/brightness/set and payload 20
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1/set and payload
↪ b'{"brightness": 52}'
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload {"state": "on",
↪ "brightness": 52.0, "color_temp": 352.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2/set and payload
↪ b'{"brightness": 52}'
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload {"state": "on",
↪ "brightness": 52.0, "color_temp": 352.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3/set and payload
↪ b'{"brightness": 52}'
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload {"state": "on",
↪ "brightness": 52.0, "color_temp": 352.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4/set and payload
↪ b'{"brightness": 52}'
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload {"state": "on",
↪ "brightness": 52.0, "color_temp": 352.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5/set and payload
↪ b'{"brightness": 52}'
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload {"state": "on",
↪ "brightness": 52.0, "color_temp": 352.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6/set and payload
↪ b'{"brightness": 52}'
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload {"state": "on",
↪ "brightness": 52.0, "color_temp": 352.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload b'{"state":
↪ "on", "brightness": 52.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload b'{"state":
↪ "on", "brightness": 52.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload b'{"state":
↪ "on", "brightness": 52.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload b'{"state":
↪ "on", "brightness": 52.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload b'{"state":
↪ "on", "brightness": 52.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload b'{"state":
↪ "on", "brightness": 52.0, "color_temp": 352.0}'
Received message with topic videv/ffe/livingroom/floorlamp/brightness and payload b'20'

```

**Success** Value for Light.brightness (ffe.livingroom.floor\_light) is correct (Content 20 and Type is <class 'int'>).

Result (Value for Light.brightness (ffe.livingroom.floor\_light)): 20 (<class 'int'>)

```
Expectation (Value for Light.brightness (ffe.livingroom.floor_light)): result = 20 (<class
↳ 'int'>)
```

---

**Info**    Setting state of ViDevCommon.brightness (ffe.livingroom.floorlamp) to 40

---

```
Sending message with topic videv/ffe/livingroom/floorlamp/brightness/set and payload 40
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1/set and payload
↳ b'{"brightness": 102}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload {"state": "on",
↳ "brightness": 102.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2/set and payload
↳ b'{"brightness": 102}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload {"state": "on",
↳ "brightness": 102.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3/set and payload
↳ b'{"brightness": 102}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload {"state": "on",
↳ "brightness": 102.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4/set and payload
↳ b'{"brightness": 102}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload {"state": "on",
↳ "brightness": 102.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5/set and payload
↳ b'{"brightness": 102}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload {"state": "on",
↳ "brightness": 102.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6/set and payload
↳ b'{"brightness": 102}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload {"state": "on",
↳ "brightness": 102.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload b'{"state":
↳ "on", "brightness": 102.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload b'{"state":
↳ "on", "brightness": 102.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload b'{"state":
↳ "on", "brightness": 102.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload b'{"state":
↳ "on", "brightness": 102.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload b'{"state":
↳ "on", "brightness": 102.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload b'{"state":
↳ "on", "brightness": 102.0, "color_temp": 352.0}'
```

```
Received message with topic videv/ffe/livingroom/floorlamp/brightness and payload b'40'
```

---

**Success** Value for Light.brightness (ffe.livingroom.floor\_light) is correct (Content 40 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffe.livingroom.floor\_light)): 40 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.livingroom.floor\_light)): result = 40 (<class 'int'>)  
 ↪ 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffe.livingroom.floorlamp) to 60

---

Sending message with topic videv/ffe/livingroom/floorlamp/brightness/set and payload 60

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload  
 ↪ b'{"brightness": 153}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
 ↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload  
 ↪ b'{"brightness": 153}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
 ↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3/set and payload  
 ↪ b'{"brightness": 153}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
 ↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4/set and payload  
 ↪ b'{"brightness": 153}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
 ↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5/set and payload  
 ↪ b'{"brightness": 153}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
 ↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6/set and payload  
 ↪ b'{"brightness": 153}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
 ↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
 ↪ "on", "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
 ↪ "on", "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
 ↪ "on", "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
 ↪ "on", "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
 ↪ "on", "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/floorlamp/brightness and payload b'60'

---

**Success** Value for Light.brightness (ffe.livingroom.floor\_light) is correct (Content 60 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffe.livingroom.floor\_light)): 60 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.livingroom.floor\_light)): result = 60 (<class  
↪ 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffe.livingroom.floorlamp) to 80

---

Sending message with topic videv/ffe/livingroom/floorlamp/brightness/set and payload 80

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload  
↪ b'{"brightness": 203}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload  
↪ b'{"brightness": 203}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3/set and payload  
↪ b'{"brightness": 203}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4/set and payload  
↪ b'{"brightness": 203}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5/set and payload  
↪ b'{"brightness": 203}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6/set and payload  
↪ b'{"brightness": 203}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "on", "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "on", "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "on", "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "on", "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "on", "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/floorlamp/brightness and payload b'80'

---

**Success** Value for Light.brightness (ffe.livingroom.floor\_light) is correct (Content 80 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffe.livingroom.floor\_light)): 80 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.livingroom.floor\_light)): result = 80 (<class  
↪ 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffe.livingroom.floorlamp) to 100

---

Sending message with topic videv/ffe/livingroom/floorlamp/brightness/set and payload 100

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload  
↪ b'{"brightness": 254}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload  
↪ b'{"brightness": 254}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3/set and payload  
↪ b'{"brightness": 254}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4/set and payload  
↪ b'{"brightness": 254}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5/set and payload  
↪ b'{"brightness": 254}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6/set and payload  
↪ b'{"brightness": 254}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/floorlamp/brightness and payload b'100'

---

**Success** Value for Light.brightness (ffe.livingroom.floor\_light) is correct (Content 100 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffe.livingroom.floor\_light)): 100 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.livingroom.floor\_light)): result = 100 (<class  
↪ 'int'>)

#### A.1.14 Light.brightness (ffe.livingroom.floor\_light) → ViDevCommon.brightness (ffe.livingroom.floorlamp)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 100

---

Sending message with topic videv/ffe/livingroom/floorlamp/brightness/set and payload 100

---

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

---

**Info** Setting state of Light.brightness (ffe.livingroom.floor\_light) to 0

---

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload b'{"state":
↪ "on", "brightness": 1.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload b'{"state":
↪ "on", "brightness": 1.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload b'{"state":
↪ "on", "brightness": 1.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload b'{"state":
↪ "on", "brightness": 1.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload b'{"state":
↪ "on", "brightness": 1.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload b'{"state":
↪ "on", "brightness": 1.0, "color_temp": 352.0}'
```

```
Received message with topic videv/ffe/livingroom/floorlamp/brightness and payload b'0'
```

---

**Success** Value for ViDevCommon.brightness (ffe.livingroom.floorlamp) is correct (Content 0 and Type is <class 'int'>).

---

```
Result (Value for ViDevCommon.brightness (ffe.livingroom.floorlamp)): 0 (<class 'int'>)
```

```
Expectation (Value for ViDevCommon.brightness (ffe.livingroom.floorlamp)): result = 0 (<class
↪ 'int'>)
```

---

**Info** Setting state of Light.brightness (ffe.livingroom.floor\_light) to 20

---

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload {"state": "on",
↪ "brightness": 52.0, "color_temp": 352.0}
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload {"state": "on",
↪ "brightness": 52.0, "color_temp": 352.0}
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload {"state": "on",
↪ "brightness": 52.0, "color_temp": 352.0}
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload {"state": "on",
↪ "brightness": 52.0, "color_temp": 352.0}
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload {"state": "on",
↪ "brightness": 52.0, "color_temp": 352.0}
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload {"state": "on",
↪ "brightness": 52.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload b'{"state":
↪ "on", "brightness": 52.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload b'{"state":
↪ "on", "brightness": 52.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload b'{"state":
↪ "on", "brightness": 52.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload b'{"state":
↪ "on", "brightness": 52.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload b'{"state":
↪ "on", "brightness": 52.0, "color_temp": 352.0}'
```

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 52.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/floorlamp/brightness and payload b'20'

---

**Success** Value for ViDevCommon.brightness (ffe.livingroom.floorlamp) is correct (Content 20 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffe.livingroom.floorlamp)): 20 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.livingroom.floorlamp)): result = 20 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (ffe.livingroom.floor\_light) to 40

---

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "on", "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "on", "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "on", "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "on", "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "on", "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/floorlamp/brightness and payload b'40'

---

**Success** Value for ViDevCommon.brightness (ffe.livingroom.floorlamp) is correct (Content 40 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffe.livingroom.floorlamp)): 40 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.livingroom.floorlamp)): result = 40 (<class  
↪ 'int'>)

---

---

**Info** Setting state of Light.brightness (ffe.livingroom.floor\_light) to 60

---

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload {"state": "on",
↪ "brightness": 153.0, "color_temp": 352.0}
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload {"state": "on",
↪ "brightness": 153.0, "color_temp": 352.0}
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload {"state": "on",
↪ "brightness": 153.0, "color_temp": 352.0}
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload {"state": "on",
↪ "brightness": 153.0, "color_temp": 352.0}
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload {"state": "on",
↪ "brightness": 153.0, "color_temp": 352.0}
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload {"state": "on",
↪ "brightness": 153.0, "color_temp": 352.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload b'{"state":
↪ "on", "brightness": 153.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload b'{"state":
↪ "on", "brightness": 153.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload b'{"state":
↪ "on", "brightness": 153.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload b'{"state":
↪ "on", "brightness": 153.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload b'{"state":
↪ "on", "brightness": 153.0, "color_temp": 352.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload b'{"state":
↪ "on", "brightness": 153.0, "color_temp": 352.0}'
Received message with topic videv/ffe/livingroom/floorlamp/brightness and payload b'60'
```

---

**Success** Value for ViDevCommon.brightness (ffe.livingroom.floorlamp) is correct (Content 60 and Type is <class 'int'>).

---

```
Result (Value for ViDevCommon.brightness (ffe.livingroom.floorlamp)): 60 (<class 'int'>)
Expectation (Value for ViDevCommon.brightness (ffe.livingroom.floorlamp)): result = 60 (<class
↪ 'int'>)
```

---

**Info** Setting state of Light.brightness (ffe.livingroom.floor\_light) to 80

---

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload {"state": "on",
↪ "brightness": 203.0, "color_temp": 352.0}
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload {"state": "on",
↪ "brightness": 203.0, "color_temp": 352.0}
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload {"state": "on",
↪ "brightness": 203.0, "color_temp": 352.0}
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload {"state": "on",
↪  "brightness": 203.0, "color_temp": 352.0}

Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload {"state": "on",
↪  "brightness": 203.0, "color_temp": 352.0}

Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload {"state": "on",
↪  "brightness": 203.0, "color_temp": 352.0}

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload b'{"state":
↪  "on", "brightness": 203.0, "color_temp": 352.0}'

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload b'{"state":
↪  "on", "brightness": 203.0, "color_temp": 352.0}'

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload b'{"state":
↪  "on", "brightness": 203.0, "color_temp": 352.0}'

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload b'{"state":
↪  "on", "brightness": 203.0, "color_temp": 352.0}'

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload b'{"state":
↪  "on", "brightness": 203.0, "color_temp": 352.0}'

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload b'{"state":
↪  "on", "brightness": 203.0, "color_temp": 352.0}'

Received message with topic videv/ffe/livingroom/floorlamp/brightness and payload b'80'
```

---

**Success** Value for ViDevCommon.brightness (ffe.livingroom.floorlamp) is correct (Content 80 and Type is <class 'int'>).

---

```
Result (Value for ViDevCommon.brightness (ffe.livingroom.floorlamp)): 80 (<class 'int'>)
Expectation (Value for ViDevCommon.brightness (ffe.livingroom.floorlamp)): result = 80 (<class
↪  'int'>)
```

---

**Info** Setting state of Light.brightness (ffe.livingroom.floor\_light) to 100

---

```
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 352.0}

Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 352.0}

Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 352.0}

Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 352.0}

Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 352.0}

Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 352.0}

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload b'{"state":
↪  "on", "brightness": 254.0, "color_temp": 352.0}'

Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload b'{"state":
↪  "on", "brightness": 254.0, "color_temp": 352.0}'
```

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/livingroom/floorlamp/brightness and payload b'100'

**Success** Value for ViDevCommon.brightness (ffe.livingroom.floorlamp) is correct (Content 100 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffe.livingroom.floorlamp)): 100 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.livingroom.floorlamp)): result = 100  
↪ (<class 'int'>)

#### A.1.15 ViDevCommon.color\_temp (ffe.livingroom.floorlamp) → Light.color\_temp (ffe.livingroom.floor\_light)

##### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Switching on device

**Info** Prepare: Setting devices to last state 10

Sending message with topic videv/ffe/livingroom/floorlamp/color\_temp/set and payload 10

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload  
↪ b'{"color\_temp": 454}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload  
↪ b'{"color\_temp": 454}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3/set and payload  
↪ b'{"color\_temp": 454}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4/set and payload  
↪ b'{"color\_temp": 454}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5/set and payload  
↪ b'{"color\_temp": 454}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6/set and payload  
↪ b'{"color\_temp": 454}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/ffe/livingroom/floorlamp/color\_temp and payload b'10'

---

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (10, 10) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (10, 10) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffe.livingroom.floorlamp) to 0

---

Sending message with topic videv/ffe/livingroom/floorlamp/color\_temp/set and payload 0

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload  
↪ b'{"color\_temp": 250}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload  
↪ b'{"color\_temp": 250}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3/set and payload  
↪ b'{"color\_temp": 250}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4/set and payload  
↪ b'{"color\_temp": 250}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5/set and payload

↪ b'{"color\_temp": 250}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",

↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6/set and payload

↪ b'{"color\_temp": 250}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",

↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":

↪ "on", "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":

↪ "on", "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":

↪ "on", "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":

↪ "on", "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":

↪ "on", "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":

↪ "on", "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic videv/ffe/livingroom/floorlamp/color\_temp and payload b'0'

---

**Success** Value for Light.color\_temp (ffe.livingroom.floor\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffe.livingroom.floor\_light)): 0 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffe.livingroom.floor\_light)): result = 0 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffe.livingroom.floorlamp) to 2

---

Sending message with topic videv/ffe/livingroom/floorlamp/color\_temp/set and payload 2

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload

↪ b'{"color\_temp": 291}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",

↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload

↪ b'{"color\_temp": 291}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",

↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3/set and payload

↪ b'{"color\_temp": 291}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",

↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4/set and payload

↪ b'{"color\_temp": 291}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5/set and payload  
↪ b'{"color\_temp": 291}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6/set and payload  
↪ b'{"color\_temp": 291}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic videv/ffe/livingroom/floorlamp/color\_temp and payload b'2'

---

**Success** Value for Light.color\_temp (ffe.livingroom.floor\_light) is correct (Content 2 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffe.livingroom.floor\_light)): 2 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffe.livingroom.floor\_light)): result = 2 (<class  
↪ 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffe.livingroom.floorlamp) to 4

---

Sending message with topic videv/ffe/livingroom/floorlamp/color\_temp/set and payload 4

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload  
↪ b'{"color\_temp": 332}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload  
↪ b'{"color\_temp": 332}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3/set and payload  
↪ b'{"color\_temp": 332}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4/set and payload  
 ↪ b'{"color\_temp": 332}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
 ↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5/set and payload  
 ↪ b'{"color\_temp": 332}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
 ↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6/set and payload  
 ↪ b'{"color\_temp": 332}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
 ↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
 ↪ "on", "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
 ↪ "on", "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
 ↪ "on", "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
 ↪ "on", "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
 ↪ "on", "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
 ↪ "on", "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic videv/ffe/livingroom/floorlamp/color\_temp and payload b'4'

---

**Success** Value for Light.color\_temp (ffe.livingroom.floor\_light) is correct (Content 4 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffe.livingroom.floor\_light)): 4 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffe.livingroom.floor\_light)): result = 4 (<class  
 ↪ 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffe.livingroom.floorlamp) to 6

---

Sending message with topic videv/ffe/livingroom/floorlamp/color\_temp/set and payload 6

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload  
 ↪ b'{"color\_temp": 372}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
 ↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload  
 ↪ b'{"color\_temp": 372}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
 ↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3/set and payload  
 ↪ b'{"color\_temp": 372}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4/set and payload  
↪ b'{"color\_temp": 372}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5/set and payload  
↪ b'{"color\_temp": 372}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6/set and payload  
↪ b'{"color\_temp": 372}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic videv/ffe/livingroom/floorlamp/color\_temp and payload b'6'

---

**Success** Value for Light.color\_temp (ffe.livingroom.floor\_light) is correct (Content 6 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffe.livingroom.floor\_light)): 6 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffe.livingroom.floor\_light)): result = 6 (<class  
↪ 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffe.livingroom.floorlamp) to 8

---

Sending message with topic videv/ffe/livingroom/floorlamp/color\_temp/set and payload 8

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload  
↪ b'{"color\_temp": 413}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload  
↪ b'{"color\_temp": 413}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3/set and payload  
 ↪ b'{"color\_temp": 413}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
 ↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4/set and payload  
 ↪ b'{"color\_temp": 413}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
 ↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5/set and payload  
 ↪ b'{"color\_temp": 413}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
 ↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6/set and payload  
 ↪ b'{"color\_temp": 413}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
 ↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
 ↪ "on", "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
 ↪ "on", "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
 ↪ "on", "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
 ↪ "on", "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
 ↪ "on", "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
 ↪ "on", "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic videv/ffe/livingroom/floorlamp/color\_temp and payload b'8'

---

**Success** Value for Light.color\_temp (ffe.livingroom.floor\_light) is correct (Content 8 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffe.livingroom.floor\_light)): 8 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffe.livingroom.floor\_light)): result = 8 (<class  
 ↪ 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffe.livingroom.floorlamp) to 10

---

Sending message with topic videv/ffe/livingroom/floorlamp/color\_temp/set and payload 10

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1/set and payload  
 ↪ b'{"color\_temp": 454}'

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
 ↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2/set and payload  
 ↪ b'{"color\_temp": 454}'

```

Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 454.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3/set and payload
↪  b'{"color_temp": 454}'
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 454.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4/set and payload
↪  b'{"color_temp": 454}'
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 454.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5/set and payload
↪  b'{"color_temp": 454}'
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 454.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6/set and payload
↪  b'{"color_temp": 454}'
Sending message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 454.0}
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_1 and payload b'{"state":
↪  "on", "brightness": 254.0, "color_temp": 454.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_2 and payload b'{"state":
↪  "on", "brightness": 254.0, "color_temp": 454.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_3 and payload b'{"state":
↪  "on", "brightness": 254.0, "color_temp": 454.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_4 and payload b'{"state":
↪  "on", "brightness": 254.0, "color_temp": 454.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_5 and payload b'{"state":
↪  "on", "brightness": 254.0, "color_temp": 454.0}'
Received message with topic zigbee_ffe/ffe/livingroom/floor_light_6 and payload b'{"state":
↪  "on", "brightness": 254.0, "color_temp": 454.0}'
Received message with topic videv/ffe/livingroom/floorlamp/color_temp and payload b'10'

```

---

**Success** Value for Light.color\_temp (ffe.livingroom.floor\_light) is correct (Content 10 and Type is <class 'int'>).

---

```
Result (Value for Light.color_temp (ffe.livingroom.floor_light)): 10 (<class 'int'>)
```

```
Expectation (Value for Light.color_temp (ffe.livingroom.floor_light)): result = 10 (<class
↪  'int'>)
```

#### A.1.16 Light.color\_temp (ffe.livingroom.floor\_light) → ViDevCommon.color\_temp (ffe.livingroom.floorlamp)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 10

---

Sending message with topic videv/ffe/livingroom/floorlamp/color\_temp/set and payload 10

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (10, 10) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (10, 10) (<class 'tuple'>)

**Info** Setting state of Light.color\_temp (ffe.livingroom.floor\_light) to 0

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic videv/ffe/livingroom/floorlamp/color\_temp and payload b'0'

**Success** Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp) is correct (Content 0 and Type is <class 'int'>).

Result (Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp)): 0 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp)): result = 0 (<class  
↪ 'int'>)

**Info** Setting state of Light.color\_temp (ffe.livingroom.floor\_light) to 2

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic videv/ffe/livingroom/floorlamp/color\_temp and payload b'2'

---

**Success** Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp) is correct (Content 2 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp)): 2 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp)): result = 2 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (ffe.livingroom.floor\_light) to 4

---

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic videv/ffe/livingroom/floorlamp/color\_temp and payload b'4'

---

**Success** Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp) is correct (Content 4 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp)): 4 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp)): result = 4 (<class 'int'>)

---

**Info** Setting state of Light.color\_temp (ffe.livingroom.floor\_light) to 6

---

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic videv/ffe/livingroom/floorlamp/color\_temp and payload b'6'

---

**Success** Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp) is correct (Content 6 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp)): 6 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp)): result = 6 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (ffe.livingroom.floor\_light) to 8

---

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic videv/ffe/livingroom/floorlamp/color\_temp and payload b'8'

---

**Success** Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp) is correct (Content 8 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp)): 8 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp)): result = 8 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (ffe.livingroom.floor\_light) to 10

---

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Sending message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_1 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_2 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_3 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_4 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_5 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic zigbee\_ffe/ffe/livingroom/floor\_light\_6 and payload b'{"state":  
↪ "on", "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/ffe/livingroom/floorlamp/color\_temp and payload b'10'

---

**Success** Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp) is correct (Content 10 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp)): 10 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.livingroom.floorlamp)): result = 10 (<class  
↪ 'int'>)

**A.1.17 ViDevHeating.temp\_setp (ffe.livingroom.heating\_valve) → HeatingValve.temp\_setp (ffe.livingroom.heating\_valve)**

### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state 30

---

Sending message with topic videv/ffe/livingroom/heating\_valve/user\_temperature\_setpoint/set  
↪ and payload 30

Sending message with topic zigbee\_ffe/ffe/livingroom/heating\_valve and payload

↪ {"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}

Received message with topic zigbee\_ffe/ffe/livingroom/heating\_valve and payload

↪ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

Received message with topic zigbee\_ffe/ffe/livingroom/heating\_valve/set and payload

↪ b'{"current\_heating\_setpoint": 30}'

Received message with topic videv/ffe/livingroom/heating\_valve/valve\_temperature\_setpoint and

↪ payload b'30'

Received message with topic videv/ffe/livingroom/heating\_valve/user\_temperature\_setpoint and

↪ payload b'30'

---

**Success** Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (30, 30) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (30, 30) (<class 'tuple'>)

---

**Info** Setting state of ViDevHeating.temp\_setp (ffe.livingroom.heating\_valve) to 15

---

Sending message with topic videv/ffe/livingroom/heating\_valve/user\_temperature\_setpoint/set

↪ and payload 15

Received message with topic zigbee\_ffe/ffe/livingroom/heating\_valve/set and payload

↪ b'{"current\_heating\_setpoint": 15}'

Sending message with topic zigbee\_ffe/ffe/livingroom/heating\_valve and payload

↪ {"current\_heating\_setpoint": 15, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffe/livingroom/heating\_valve/valve\_temperature\_setpoint and

↪ payload b'15'

Received message with topic videv/ffe/livingroom/heating\_valve/user\_temperature\_setpoint and

↪ payload b'15'

Received message with topic zigbee\_ffe/ffe/livingroom/heating\_valve and payload

↪ b'{"current\_heating\_setpoint": 15, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Value for HeatingValve.temp\_setp (ffe.livingroom.heating\_valve) is correct (Content 15 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (ffe.livingroom.heating\_valve)): 15 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffe.livingroom.heating\_valve)): result = 15

↪ (<class 'int'>)

---

**Info** Setting state of ViDevHeating.temp\_setp (ffe.livingroom.heating\_valve) to 20

---

Sending message with topic videv/ffe/livingroom/heating\_valve/user\_temperature\_setpoint/set

↪ and payload 20

Received message with topic zigbee\_ffe/ffe/livingroom/heating\_valve/set and payload

↪ b'{"current\_heating\_setpoint": 20}'

Sending message with topic zigbee\_ffe/ffe/livingroom/heating\_valve and payload

↪ {"current\_heating\_setpoint": 20, "local\_temperature": 20.7, "battery": 97}

```
Received message with topic videv/ffe/livingroom/heating_valve/valve_temperature_setpoint and
↳ payload b'20'
```

```
Received message with topic videv/ffe/livingroom/heating_valve/user_temperature_setpoint and
↳ payload b'20'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/heating_valve and payload
↳ b'{"current_heating_setpoint": 20, "local_temperature": 20.7, "battery": 97}'
```

---

**Success** Value for HeatingValve.temp\_setp (ffe.livingroom.heating\_valve) is correct (Content 20 and Type is <class 'int'>).

---

```
Result (Value for HeatingValve.temp_setp (ffe.livingroom.heating_valve)): 20 (<class 'int'>)
```

```
Expectation (Value for HeatingValve.temp_setp (ffe.livingroom.heating_valve)): result = 20
↳ (<class 'int'>)
```

---

**Info** Setting state of ViDevHeating.temp\_setp (ffe.livingroom.heating\_valve) to 25

---

```
Sending message with topic videv/ffe/livingroom/heating_valve/user_temperature_setpoint/set
↳ and payload 25
```

```
Received message with topic zigbee_ffe/ffe/livingroom/heating_valve/set and payload
↳ b'{"current_heating_setpoint": 25}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/heating_valve and payload
↳ {"current_heating_setpoint": 25, "local_temperature": 20.7, "battery": 97}
```

```
Received message with topic videv/ffe/livingroom/heating_valve/valve_temperature_setpoint and
↳ payload b'25'
```

```
Received message with topic videv/ffe/livingroom/heating_valve/user_temperature_setpoint and
↳ payload b'25'
```

```
Received message with topic zigbee_ffe/ffe/livingroom/heating_valve and payload
↳ b'{"current_heating_setpoint": 25, "local_temperature": 20.7, "battery": 97}'
```

---

**Success** Value for HeatingValve.temp\_setp (ffe.livingroom.heating\_valve) is correct (Content 25 and Type is <class 'int'>).

---

```
Result (Value for HeatingValve.temp_setp (ffe.livingroom.heating_valve)): 25 (<class 'int'>)
```

```
Expectation (Value for HeatingValve.temp_setp (ffe.livingroom.heating_valve)): result = 25
↳ (<class 'int'>)
```

---

**Info** Setting state of ViDevHeating.temp\_setp (ffe.livingroom.heating\_valve) to 30

---

```
Sending message with topic videv/ffe/livingroom/heating_valve/user_temperature_setpoint/set
↳ and payload 30
```

```
Received message with topic zigbee_ffe/ffe/livingroom/heating_valve/set and payload
↳ b'{"current_heating_setpoint": 30}'
```

```
Sending message with topic zigbee_ffe/ffe/livingroom/heating_valve and payload
↳ {"current_heating_setpoint": 30, "local_temperature": 20.7, "battery": 97}
```

```
Received message with topic videv/ffe/livingroom/heating_valve/valve_temperature_setpoint and
↳ payload b'30'
```

Received message with topic videv/ffe/livingroom/heating\_valve/user\_temperature\_setpoint and  
 ↳ payload b'30'

Received message with topic zigbee\_ffe/ffe/livingroom/heating\_valve and payload  
 ↳ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

**Success** Value for HeatingValve.temp\_setp (ffe.livingroom.heating\_valve) is correct (Content 30 and Type is <class 'int'>).

Result (Value for HeatingValve.temp\_setp (ffe.livingroom.heating\_valve)): 30 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffe.livingroom.heating\_valve)): result = 30  
 ↳ (<class 'int'>)

#### A.1.18 ViDevCommon.state (ffe.sleep.main\_light) → Shelly.relay/0 (ffe.sleep.main\_light)

##### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Setting devices to last state False

Sending message with topic videv/ffe/sleep/main\_light/state/set and payload false

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

**Info** Setting state of ViDevCommon.state (ffe.sleep.main\_light) to True

Sending message with topic videv/ffe/sleep/main\_light/state/set and payload true

Received message with topic shellies/ffe/sleep/main\_light/relay/0/command and payload b'on'

Sending message with topic shellies/ffe/sleep/main\_light/relay/0 and payload on

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
 ↳ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic shellies/ffe/sleep/main\_light/relay/0 and payload b'on'

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
 ↳ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/sleep/main\_light/state and payload b'true'

Received message with topic videv/ffe/sleep/main\_light/brightness and payload b'50'

Received message with topic videv/ffe/sleep/main\_light/color\_temp and payload b'5'

**Success** Value for Shelly.relay/0 (ffe.sleep.main\_light) is correct (Content True and Type is <class 'bool'>).

Result (Value for Shelly.relay/0 (ffe.sleep.main\_light)): True (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (ffe.sleep.main\_light)): result = True (<class 'bool'>)

---

**Info** Setting state of ViDevCommon.state (ffe.sleep.main\_light) to False

---

Sending message with topic videv/ffe/sleep/main\_light/state/set and payload false

Received message with topic shellies/ffe/sleep/main\_light/relay/0/command and payload b'off'

Sending message with topic shellies/ffe/sleep/main\_light/relay/0 and payload off

Received message with topic shellies/ffe/sleep/main\_light/relay/0 and payload b'off'

Received message with topic videv/ffe/sleep/main\_light/state and payload b'false'

---

**Success** Value for Shelly.relay/0 (ffe.sleep.main\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (ffe.sleep.main\_light)): False (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (ffe.sleep.main\_light)): result = False (<class 'bool'>)

#### A.1.19 Shelly.relay/0 (ffe.sleep.main\_light) → ViDevCommon.state (ffe.sleep.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffe/sleep/main\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Shelly.relay/0 (ffe.sleep.main\_light) to True

---

Sending message with topic shellies/ffe/sleep/main\_light/relay/0 and payload on

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic shellies/ffe/sleep/main\_light/relay/0 and payload b'on'

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/sleep/main\_light/state and payload b'true'

---

**Success** Value for ViDevCommon.state (ffe.sleep.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffe.sleep.main\_light)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.sleep.main\_light)): result = True (<class  
↪ 'bool'>)

---

**Info** Setting state of Shelly.relay/0 (ffe.sleep.main\_light) to False

---

Sending message with topic shellies/ffe/sleep/main\_light/relay/0 and payload off

Received message with topic shellies/ffe/sleep/main\_light/relay/0 and payload b'off'

Received message with topic videv/ffe/sleep/main\_light/state and payload b'false'

---

**Success** Value for ViDevCommon.state (ffe.sleep.main\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffe.sleep.main\_light)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.sleep.main\_light)): result = False (<class 'bool'>)  
↪ 'bool'>)

---

#### A.1.20 ViDevCommon.state (ffe.sleep.bed\_light\_di) → Light.state (ffe.sleep.bed\_light\_di)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffe/sleep/bed\_light\_di/state/set and payload false

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di/set and payload b'{"state":  
↪ "off"}'

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (ffe.sleep.bed\_light\_di) to True

---

Sending message with topic videv/ffe/sleep/bed\_light\_di/state/set and payload true

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di/set and payload b'{"state":  
↪ "on"}'

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload {"state": "on",  
↪ "brightness": 127.0}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload b'{"state": "on",  
↪ "brightness": 127.0}'

Received message with topic videv/ffe/sleep/bed\_light\_di/state and payload b'true'

Received message with topic videv/ffe/sleep/bed\_light\_di/brightness and payload b'50'

---

**Success** Value for Light.state (ffe.sleep.bed\_light\_di) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Light.state (ffe.sleep.bed\_light\_di)): True (<class 'bool'>)

---

```
Expectation (Value for Light.state (ffe.sleep.bed_light_di)): result = True (<class 'bool'>)
```

---

**Info** Setting state of ViDevCommon.state (ffe.sleep.bed\_light\_di) to False

---

```
Sending message with topic videv/ffe/sleep/bed_light_di/state/set and payload false
```

```
Received message with topic zigbee_ffe/ffe/sleep/bed_light_di/set and payload b'{"state":  
↪ "off"}'
```

```
Sending message with topic zigbee_ffe/ffe/sleep/bed_light_di and payload {"state": "off",  
↪ "brightness": 127.0}
```

```
Received message with topic zigbee_ffe/ffe/sleep/bed_light_di and payload b'{"state": "off",  
↪ "brightness": 127.0}'
```

```
Received message with topic videv/ffe/sleep/bed_light_di/state and payload b'false'
```

---

**Success** Value for Light.state (ffe.sleep.bed\_light\_di) is correct (Content False and Type is <class 'bool'>).

---

```
Result (Value for Light.state (ffe.sleep.bed_light_di)): False (<class 'bool'>)
```

```
Expectation (Value for Light.state (ffe.sleep.bed_light_di)): result = False (<class 'bool'>)
```

#### A.1.21 Light.state (ffe.sleep.bed\_light\_di) → ViDevCommon.state (ffe.sleep.bed\_light\_di)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

```
Sending message with topic videv/ffe/sleep/bed_light_di/state/set and payload false
```

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

```
Result (Start state (master, slave)): (False, False) (<class 'tuple'>)
```

```
Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)
```

---

**Info** Setting state of Light.state (ffe.sleep.bed\_light\_di) to True

---

```
Sending message with topic zigbee_ffe/ffe/sleep/bed_light_di and payload {"state": "on",  
↪ "brightness": 127.0}
```

```
Received message with topic zigbee_ffe/ffe/sleep/bed_light_di and payload b'{"state": "on",  
↪ "brightness": 127.0}'
```

```
Received message with topic videv/ffe/sleep/bed_light_di/state and payload b'true'
```

---

**Success** Value for ViDevCommon.state (ffe.sleep.bed\_light\_di) is correct (Content True and Type is <class 'bool'>).

---

```
Result (Value for ViDevCommon.state (ffe.sleep.bed_light_di)): True (<class 'bool'>)
```

```
Expectation (Value for ViDevCommon.state (ffe.sleep.bed_light_di)): result = True (<class  
↪ 'bool'>)
```

---

**Info** Setting state of Light.state (ffe.sleep.bed\_light\_di) to False

---

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload {"state": "off",  
↪ "brightness": 127.0}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload b'{"state": "off",  
↪ "brightness": 127.0}'

Received message with topic videv/ffe/sleep/bed\_light\_di/state and payload b'false'

---

**Success** Value for ViDevCommon.state (ffe.sleep.bed\_light\_di) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffe.sleep.bed\_light\_di)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.sleep.bed\_light\_di)): result = False (<class  
↪ 'bool'>)

---

#### A.1.22 ViDevCommon.state (ffe.sleep.bed\_light\_ma) → Powerplug1P.state (ffe.sleep.bed\_light\_ma)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffe/sleep/bed\_light\_ma/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (ffe.sleep.bed\_light\_ma) to True

---

Sending message with topic videv/ffe/sleep/bed\_light\_ma/state/set and payload true

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_ma/set and payload b'{"state":  
↪ "on"}'

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_ma and payload {"state": "on"}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_ma and payload b'{"state": "on"}'

Received message with topic videv/ffe/sleep/bed\_light\_ma/state and payload b'true'

---

**Success** Value for Powerplug1P.state (ffe.sleep.bed\_light\_ma) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Powerplug1P.state (ffe.sleep.bed\_light\_ma)): True (<class 'bool'>)

Expectation (Value for Powerplug1P.state (ffe.sleep.bed\_light\_ma)): result = True (<class  
↪ 'bool'>)

---

---

**Info** Setting state of ViDevCommon.state (ffe.sleep.bed\_light\_ma) to False

---

Sending message with topic videv/ffe/sleep/bed\_light\_ma/state/set and payload false

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_ma/set and payload b'{"state":  
↪ "off"}'

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_ma and payload {"state": "off"}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_ma and payload b'{"state": "off"}'

Received message with topic videv/ffe/sleep/bed\_light\_ma/state and payload b'false'

---

**Success** Value for Powerplug1P.state (ffe.sleep.bed\_light\_ma) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Powerplug1P.state (ffe.sleep.bed\_light\_ma)): False (<class 'bool'>)

Expectation (Value for Powerplug1P.state (ffe.sleep.bed\_light\_ma)): result = False (<class  
↪ 'bool'>)

---

#### A.1.23 Powerplug1P.state (ffe.sleep.bed\_light\_ma) → ViDevCommon.state (ffe.sleep.bed\_light\_ma)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffe/sleep/bed\_light\_ma/state/set and payload false

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Powerplug1P.state (ffe.sleep.bed\_light\_ma) to True

---

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_ma and payload {"state": "on"}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_ma and payload b'{"state": "on"}'

Received message with topic videv/ffe/sleep/bed\_light\_ma/state and payload b'true'

---

**Success** Value for ViDevCommon.state (ffe.sleep.bed\_light\_ma) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffe.sleep.bed\_light\_ma)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.sleep.bed\_light\_ma)): result = True (<class  
↪ 'bool'>)

---

---

**Info** Setting state of Powerplug1P.state (ffe.sleep.bed\_light\_ma) to False

---

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_ma and payload {"state": "off"}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_ma and payload b'{"state": "off"}'

Received message with topic videv/ffe/sleep/bed\_light\_ma/state and payload b'false'

---

**Success** Value for ViDevCommon.state (ffe.sleep.bed\_light\_ma) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffe.sleep.bed\_light\_ma)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.sleep.bed\_light\_ma)): result = False (<class 'bool'>)  
↪ 'bool'>)

---

#### A.1.24 ViDevCommon.brightness (ffe.sleep.main\_light) → Light.brightness (ffe.sleep.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---

Sending message with topic shellies/ffe/sleep/main\_light/relay/0 and payload on

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic shellies/ffe/sleep/main\_light/relay/0 and payload b'on'

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/sleep/main\_light/state and payload b'true'

---

**Info** Prepare: Setting devices to last state 100

---

Sending message with topic videv/ffe/sleep/main\_light/brightness/set and payload 100

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light/set and payload b'{"brightness":  
↪ 254}'

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/sleep/main\_light/brightness and payload b'100'

---

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

---

---

**Info** Setting state of ViDevCommon.brightness (ffe.sleep.main\_light) to 0

---

Sending message with topic videv/ffe/sleep/main\_light/brightness/set and payload 0

Received message with topic zigbee\_ffe/ffe/sleep/main\_light/set and payload b'{"brightness":  
↪ 1}'

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/sleep/main\_light/brightness and payload b'0'

---

**Success** Value for Light.brightness (ffe.sleep.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffe.sleep.main\_light)): 0 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.sleep.main\_light)): result = 0 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffe.sleep.main\_light) to 20

---

Sending message with topic videv/ffe/sleep/main\_light/brightness/set and payload 20

Received message with topic zigbee\_ffe/ffe/sleep/main\_light/set and payload b'{"brightness":  
↪ 52}'

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/sleep/main\_light/brightness and payload b'20'

---

**Success** Value for Light.brightness (ffe.sleep.main\_light) is correct (Content 20 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffe.sleep.main\_light)): 20 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.sleep.main\_light)): result = 20 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffe.sleep.main\_light) to 40

---

Sending message with topic videv/ffe/sleep/main\_light/brightness/set and payload 40

Received message with topic zigbee\_ffe/ffe/sleep/main\_light/set and payload b'{"brightness":  
↪ 102}'

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/sleep/main\_light/brightness and payload b'40'

---

**Success** Value for Light.brightness (ffe.sleep.main\_light) is correct (Content 40 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffe.sleep.main\_light)): 40 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.sleep.main\_light)): result = 40 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffe.sleep.main\_light) to 60

---

Sending message with topic videv/ffe/sleep/main\_light/brightness/set and payload 60

Received message with topic zigbee\_ffe/ffe/sleep/main\_light/set and payload b'{"brightness":  
↪ 153}'

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/sleep/main\_light/brightness and payload b'60'

---

**Success** Value for Light.brightness (ffe.sleep.main\_light) is correct (Content 60 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffe.sleep.main\_light)): 60 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.sleep.main\_light)): result = 60 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffe.sleep.main\_light) to 80

---

Sending message with topic videv/ffe/sleep/main\_light/brightness/set and payload 80

Received message with topic zigbee\_ffe/ffe/sleep/main\_light/set and payload b'{"brightness":  
↪ 203}'

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/sleep/main\_light/brightness and payload b'80'

---

**Success** Value for Light.brightness (ffe.sleep.main\_light) is correct (Content 80 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffe.sleep.main\_light)): 80 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.sleep.main\_light)): result = 80 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffe.sleep.main\_light) to 100

---

Sending message with topic videv/ffe/sleep/main\_light/brightness/set and payload 100

Received message with topic zigbee\_ffe/ffe/sleep/main\_light/set and payload b'{"brightness":  
↪ 254}'

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

```
Received message with topic zigbee_ffe/ffe/sleep/main_light and payload b'{"state": "on",
↪ "brightness": 254.0, "color_temp": 352.0}'
```

```
Received message with topic videv/ffe/sleep/main_light/brightness and payload b'100'
```

---

**Success** Value for Light.brightness (ffe.sleep.main\_light) is correct (Content 100 and Type is <class 'int'>).

---

```
Result (Value for Light.brightness (ffe.sleep.main_light)): 100 (<class 'int'>)
```

```
Expectation (Value for Light.brightness (ffe.sleep.main_light)): result = 100 (<class 'int'>)
```

#### A.1.25 Light.brightness (ffe.sleep.main\_light) → ViDevCommon.brightness (ffe.sleep.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 100

---

```
Sending message with topic videv/ffe/sleep/main_light/brightness/set and payload 100
```

---

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

---

```
Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)
```

```
Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)
```

---

**Info** Setting state of Light.brightness (ffe.sleep.main\_light) to 0

---

```
Sending message with topic zigbee_ffe/ffe/sleep/main_light and payload {"state": "on",
↪ "brightness": 1.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/sleep/main_light and payload b'{"state": "on",
↪ "brightness": 1.0, "color_temp": 352.0}'
```

```
Received message with topic videv/ffe/sleep/main_light/brightness and payload b'0'
```

---

**Success** Value for ViDevCommon.brightness (ffe.sleep.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

```
Result (Value for ViDevCommon.brightness (ffe.sleep.main_light)): 0 (<class 'int'>)
```

```
Expectation (Value for ViDevCommon.brightness (ffe.sleep.main_light)): result = 0 (<class
↪ 'int'>)
```

---

**Info** Setting state of Light.brightness (ffe.sleep.main\_light) to 20

---

```
Sending message with topic zigbee_ffe/ffe/sleep/main_light and payload {"state": "on",
↪ "brightness": 52.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffe/ffe/sleep/main_light and payload b'{"state": "on",
↪ "brightness": 52.0, "color_temp": 352.0}'
```

Received message with topic videv/ffe/sleep/main\_light/brightness and payload b'20'

**Success** Value for ViDevCommon.brightness (ffe.sleep.main\_light) is correct (Content 20 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffe.sleep.main\_light)): 20 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.sleep.main\_light)): result = 20 (<class 'int'>)

**Info** Setting state of Light.brightness (ffe.sleep.main\_light) to 40

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/sleep/main\_light/brightness and payload b'40'

**Success** Value for ViDevCommon.brightness (ffe.sleep.main\_light) is correct (Content 40 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffe.sleep.main\_light)): 40 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.sleep.main\_light)): result = 40 (<class 'int'>)

**Info** Setting state of Light.brightness (ffe.sleep.main\_light) to 60

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/sleep/main\_light/brightness and payload b'60'

**Success** Value for ViDevCommon.brightness (ffe.sleep.main\_light) is correct (Content 60 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffe.sleep.main\_light)): 60 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.sleep.main\_light)): result = 60 (<class 'int'>)

**Info** Setting state of Light.brightness (ffe.sleep.main\_light) to 80

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/sleep/main\_light/brightness and payload b'80'

---

**Success** Value for ViDevCommon.brightness (ffe.sleep.main\_light) is correct (Content 80 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffe.sleep.main\_light)): 80 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.sleep.main\_light)): result = 80 (<class 'int'>)

---

**Info** Setting state of Light.brightness (ffe.sleep.main\_light) to 100

---

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/ffe/sleep/main\_light/brightness and payload b'100'

---

**Success** Value for ViDevCommon.brightness (ffe.sleep.main\_light) is correct (Content 100 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffe.sleep.main\_light)): 100 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.sleep.main\_light)): result = 100 (<class 'int'>)

---

#### A.1.26 ViDevCommon.color\_temp (ffe.sleep.main\_light) → Light.color\_temp (ffe.sleep.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 10

---

Sending message with topic videv/ffe/sleep/main\_light/color\_temp/set and payload 10

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light/set and payload b'{"color\_temp":  
↪ 454}'

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/ffe/sleep/main\_light/color\_temp and payload b'10'

---

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (10, 10) (<class 'tuple'>)

---

Expectation (Start state (master, slave)): result = (10, 10) (<class 'tuple'>)

**Info** Setting state of ViDevCommon.color\_temp (ffe.sleep.main\_light) to 0

Sending message with topic videv/ffe/sleep/main\_light/color\_temp/set and payload 0

Received message with topic zigbee\_ffe/ffe/sleep/main\_light/set and payload b'{"color\_temp":  
↪ 250}'

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic videv/ffe/sleep/main\_light/color\_temp and payload b'0'

**Success** Value for Light.color\_temp (ffe.sleep.main\_light) is correct (Content 0 and Type is <class 'int'>).

Result (Value for Light.color\_temp (ffe.sleep.main\_light)): 0 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffe.sleep.main\_light)): result = 0 (<class 'int'>)

**Info** Setting state of ViDevCommon.color\_temp (ffe.sleep.main\_light) to 2

Sending message with topic videv/ffe/sleep/main\_light/color\_temp/set and payload 2

Received message with topic zigbee\_ffe/ffe/sleep/main\_light/set and payload b'{"color\_temp":  
↪ 291}'

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic videv/ffe/sleep/main\_light/color\_temp and payload b'2'

**Success** Value for Light.color\_temp (ffe.sleep.main\_light) is correct (Content 2 and Type is <class 'int'>).

Result (Value for Light.color\_temp (ffe.sleep.main\_light)): 2 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffe.sleep.main\_light)): result = 2 (<class 'int'>)

**Info** Setting state of ViDevCommon.color\_temp (ffe.sleep.main\_light) to 4

Sending message with topic videv/ffe/sleep/main\_light/color\_temp/set and payload 4

Received message with topic zigbee\_ffe/ffe/sleep/main\_light/set and payload b'{"color\_temp":  
↪ 332}'

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic videv/ffe/sleep/main\_light/color\_temp and payload b'4'

---

**Success** Value for Light.color\_temp (ffe.sleep.main\_light) is correct (Content 4 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffe.sleep.main\_light)): 4 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffe.sleep.main\_light)): result = 4 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffe.sleep.main\_light) to 6

---

Sending message with topic videv/ffe/sleep/main\_light/color\_temp/set and payload 6

Received message with topic zigbee\_ffe/ffe/sleep/main\_light/set and payload b'{"color\_temp":  
↪ 372}'

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic videv/ffe/sleep/main\_light/color\_temp and payload b'6'

---

**Success** Value for Light.color\_temp (ffe.sleep.main\_light) is correct (Content 6 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffe.sleep.main\_light)): 6 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffe.sleep.main\_light)): result = 6 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffe.sleep.main\_light) to 8

---

Sending message with topic videv/ffe/sleep/main\_light/color\_temp/set and payload 8

Received message with topic zigbee\_ffe/ffe/sleep/main\_light/set and payload b'{"color\_temp":  
↪ 413}'

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic videv/ffe/sleep/main\_light/color\_temp and payload b'8'

---

**Success** Value for Light.color\_temp (ffe.sleep.main\_light) is correct (Content 8 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffe.sleep.main\_light)): 8 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffe.sleep.main\_light)): result = 8 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffe.sleep.main\_light) to 10

---

Sending message with topic videv/ffe/sleep/main\_light/color\_temp/set and payload 10

Received message with topic zigbee\_ffe/ffe/sleep/main\_light/set and payload b'{"color\_temp":  
↪ 454}'

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

```
Received message with topic zigbee_ffe/ffe/sleep/main_light and payload b'{"state": "on",
↪ "brightness": 254.0, "color_temp": 454.0}'
```

```
Received message with topic videv/ffe/sleep/main_light/color_temp and payload b'10'
```

---

**Success** Value for Light.color\_temp (ffe.sleep.main\_light) is correct (Content 10 and Type is <class 'int'>).

---

```
Result (Value for Light.color_temp (ffe.sleep.main_light)): 10 (<class 'int'>)
```

```
Expectation (Value for Light.color_temp (ffe.sleep.main_light)): result = 10 (<class 'int'>)
```

#### A.1.27 Light.color\_temp (ffe.sleep.main\_light) → ViDevCommon.color\_temp (ffe.sleep.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 10

---

```
Sending message with topic videv/ffe/sleep/main_light/color_temp/set and payload 10
```

---

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

---

```
Result (Start state (master, slave)): (10, 10) (<class 'tuple'>)
```

```
Expectation (Start state (master, slave)): result = (10, 10) (<class 'tuple'>)
```

---

**Info** Setting state of Light.color\_temp (ffe.sleep.main\_light) to 0

---

```
Sending message with topic zigbee_ffe/ffe/sleep/main_light and payload {"state": "on",
↪ "brightness": 254.0, "color_temp": 250.0}
```

```
Received message with topic zigbee_ffe/ffe/sleep/main_light and payload b'{"state": "on",
↪ "brightness": 254.0, "color_temp": 250.0}'
```

```
Received message with topic videv/ffe/sleep/main_light/color_temp and payload b'0'
```

---

**Success** Value for ViDevCommon.color\_temp (ffe.sleep.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

```
Result (Value for ViDevCommon.color_temp (ffe.sleep.main_light)): 0 (<class 'int'>)
```

```
Expectation (Value for ViDevCommon.color_temp (ffe.sleep.main_light)): result = 0 (<class
↪ 'int'>)
```

---

**Info** Setting state of Light.color\_temp (ffe.sleep.main\_light) to 2

---

```
Sending message with topic zigbee_ffe/ffe/sleep/main_light and payload {"state": "on",
↪ "brightness": 254.0, "color_temp": 291.0}
```

```
Received message with topic zigbee_ffe/ffe/sleep/main_light and payload b'{"state": "on",
↪ "brightness": 254.0, "color_temp": 291.0}'
```

Received message with topic videv/ffe/sleep/main\_light/color\_temp and payload b'2'

**Success** Value for ViDevCommon.color\_temp (ffe.sleep.main\_light) is correct (Content 2 and Type is <class 'int'>).

Result (Value for ViDevCommon.color\_temp (ffe.sleep.main\_light)): 2 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.sleep.main\_light)): result = 2 (<class 'int'>)

**Info** Setting state of Light.color\_temp (ffe.sleep.main\_light) to 4

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic videv/ffe/sleep/main\_light/color\_temp and payload b'4'

**Success** Value for ViDevCommon.color\_temp (ffe.sleep.main\_light) is correct (Content 4 and Type is <class 'int'>).

Result (Value for ViDevCommon.color\_temp (ffe.sleep.main\_light)): 4 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.sleep.main\_light)): result = 4 (<class 'int'>)

**Info** Setting state of Light.color\_temp (ffe.sleep.main\_light) to 6

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic videv/ffe/sleep/main\_light/color\_temp and payload b'6'

**Success** Value for ViDevCommon.color\_temp (ffe.sleep.main\_light) is correct (Content 6 and Type is <class 'int'>).

Result (Value for ViDevCommon.color\_temp (ffe.sleep.main\_light)): 6 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.sleep.main\_light)): result = 6 (<class 'int'>)

**Info** Setting state of Light.color\_temp (ffe.sleep.main\_light) to 8

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic videv/ffe/sleep/main\_light/color\_temp and payload b'8'

---

**Success** Value for ViDevCommon.color\_temp (ffe.sleep.main\_light) is correct (Content 8 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffe.sleep.main\_light)): 8 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.sleep.main\_light)): result = 8 (<class 'int'>)  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (ffe.sleep.main\_light) to 10

---

Sending message with topic zigbee\_ffe/ffe/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_ffe/ffe/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/ffe/sleep/main\_light/color\_temp and payload b'10'

---

**Success** Value for ViDevCommon.color\_temp (ffe.sleep.main\_light) is correct (Content 10 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffe.sleep.main\_light)): 10 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffe.sleep.main\_light)): result = 10 (<class 'int'>)  
↪ 'int'>)

---

#### A.1.28 ViDevCommon.brightness (ffe.sleep.bed\_light\_di) → Light.brightness (ffe.sleep.bed\_light\_di)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload {"state": "on",  
↪ "brightness": 127.0}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload b'{"state": "on",  
↪ "brightness": 127.0}'

Received message with topic videv/ffe/sleep/bed\_light\_di/state and payload b'true'

---

**Info** Prepare: Setting devices to last state 100

---

Sending message with topic videv/ffe/sleep/bed\_light\_di/brightness/set and payload 100

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload {"state": "on",  
↪ "brightness": 254.0}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di/set and payload b'{"brightness":  
↪ 254}'

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload b'{"state": "on",  
↪ "brightness": 254.0}'

---

Received message with topic videv/ffe/sleep/bed\_light\_di/brightness and payload b'100'

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

**Info** Setting state of ViDevCommon.brightness (ffe.sleep.bed\_light\_di) to 0

Sending message with topic videv/ffe/sleep/bed\_light\_di/brightness/set and payload 0

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di/set and payload b'{"brightness":  
↪ 1}'

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload {"state": "on",  
↪ "brightness": 1.0}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload b'{"state": "on",  
↪ "brightness": 1.0}'

Received message with topic videv/ffe/sleep/bed\_light\_di/brightness and payload b'0'

**Success** Value for Light.brightness (ffe.sleep.bed\_light\_di) is correct (Content 0 and Type is <class 'int'>).

Result (Value for Light.brightness (ffe.sleep.bed\_light\_di)): 0 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.sleep.bed\_light\_di)): result = 0 (<class 'int'>)

**Info** Setting state of ViDevCommon.brightness (ffe.sleep.bed\_light\_di) to 20

Sending message with topic videv/ffe/sleep/bed\_light\_di/brightness/set and payload 20

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di/set and payload b'{"brightness":  
↪ 52}'

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload {"state": "on",  
↪ "brightness": 52.0}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload b'{"state": "on",  
↪ "brightness": 52.0}'

Received message with topic videv/ffe/sleep/bed\_light\_di/brightness and payload b'20'

**Success** Value for Light.brightness (ffe.sleep.bed\_light\_di) is correct (Content 20 and Type is <class 'int'>).

Result (Value for Light.brightness (ffe.sleep.bed\_light\_di)): 20 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.sleep.bed\_light\_di)): result = 20 (<class 'int'>)

**Info** Setting state of ViDevCommon.brightness (ffe.sleep.bed\_light\_di) to 40

Sending message with topic videv/ffe/sleep/bed\_light\_di/brightness/set and payload 40

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di/set and payload b'{"brightness":  
↪ 102}'

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload {"state": "on",  
↪ "brightness": 102.0}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload b'{"state": "on",  
↪ "brightness": 102.0}'

Received message with topic videv/ffe/sleep/bed\_light\_di/brightness and payload b'40'

**Success** Value for Light.brightness (ffe.sleep.bed\_light\_di) is correct (Content 40 and Type is <class 'int'>).

Result (Value for Light.brightness (ffe.sleep.bed\_light\_di)): 40 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.sleep.bed\_light\_di)): result = 40 (<class 'int'>)

**Info** Setting state of ViDevCommon.brightness (ffe.sleep.bed\_light\_di) to 60

Sending message with topic videv/ffe/sleep/bed\_light\_di/brightness/set and payload 60

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di/set and payload b'{"brightness":  
↪ 153}'

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload {"state": "on",  
↪ "brightness": 153.0}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload b'{"state": "on",  
↪ "brightness": 153.0}'

Received message with topic videv/ffe/sleep/bed\_light\_di/brightness and payload b'60'

**Success** Value for Light.brightness (ffe.sleep.bed\_light\_di) is correct (Content 60 and Type is <class 'int'>).

Result (Value for Light.brightness (ffe.sleep.bed\_light\_di)): 60 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.sleep.bed\_light\_di)): result = 60 (<class 'int'>)

**Info** Setting state of ViDevCommon.brightness (ffe.sleep.bed\_light\_di) to 80

Sending message with topic videv/ffe/sleep/bed\_light\_di/brightness/set and payload 80

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di/set and payload b'{"brightness":  
↪ 203}'

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload {"state": "on",  
↪ "brightness": 203.0}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload b'{"state": "on",  
↪ "brightness": 203.0}'

Received message with topic videv/ffe/sleep/bed\_light\_di/brightness and payload b'80'

**Success** Value for Light.brightness (ffe.sleep.bed\_light\_di) is correct (Content 80 and Type is <class 'int'>).

Result (Value for Light.brightness (ffe.sleep.bed\_light\_di)): 80 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.sleep.bed\_light\_di)): result = 80 (<class 'int'>)

**Info** Setting state of ViDevCommon.brightness (ffe.sleep.bed\_light\_di) to 100

Sending message with topic videv/ffe/sleep/bed\_light\_di/brightness/set and payload 100

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di/set and payload b'{"brightness":  
↪ 254}'

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload {"state": "on",  
↪ "brightness": 254.0}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload b'{"state": "on",  
↪ "brightness": 254.0}'

Received message with topic videv/ffe/sleep/bed\_light\_di/brightness and payload b'100'

---

**Success** Value for Light.brightness (ffe.sleep.bed\_light\_di) is correct (Content 100 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffe.sleep.bed\_light\_di)): 100 (<class 'int'>)

Expectation (Value for Light.brightness (ffe.sleep.bed\_light\_di)): result = 100 (<class  
↪ 'int'>)

#### A.1.29 Light.brightness (ffe.sleep.bed\_light\_di) → ViDevCommon.brightness (ffe.sleep.bed\_light\_di)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 100

---

Sending message with topic videv/ffe/sleep/bed\_light\_di/brightness/set and payload 100

---

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

---

**Info** Setting state of Light.brightness (ffe.sleep.bed\_light\_di) to 0

---

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload {"state": "on",  
↪ "brightness": 1.0}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload b'{"state": "on",  
↪ "brightness": 1.0}'

Received message with topic videv/ffe/sleep/bed\_light\_di/brightness and payload b'0'

---

**Success** Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di)): 0 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di)): result = 0 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (ffe.sleep.bed\_light\_di) to 20

---

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload {"state": "on",  
↪ "brightness": 52.0}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload b'{"state": "on",  
↪ "brightness": 52.0}'

Received message with topic videv/ffe/sleep/bed\_light\_di/brightness and payload b'20'

---

**Success** Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di) is correct (Content 20 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di)): 20 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di)): result = 20 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (ffe.sleep.bed\_light\_di) to 40

---

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload {"state": "on",  
↪ "brightness": 102.0}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload b'{"state": "on",  
↪ "brightness": 102.0}'

Received message with topic videv/ffe/sleep/bed\_light\_di/brightness and payload b'40'

---

**Success** Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di) is correct (Content 40 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di)): 40 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di)): result = 40 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (ffe.sleep.bed\_light\_di) to 60

---

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload {"state": "on",  
↪ "brightness": 153.0}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload b'{"state": "on",  
↪ "brightness": 153.0}'

Received message with topic videv/ffe/sleep/bed\_light\_di/brightness and payload b'60'

---

**Success** Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di) is correct (Content 60 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di)): 60 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di)): result = 60 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (ffe.sleep.bed\_light\_di) to 80

---

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload {"state": "on",  
↪ "brightness": 203.0}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload b'{"state": "on",  
↪ "brightness": 203.0}'

Received message with topic videv/ffe/sleep/bed\_light\_di/brightness and payload b'80'

**Success** Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di) is correct (Content 80 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di)): 80 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di)): result = 80 (<class 'int'>)

**Info** Setting state of Light.brightness (ffe.sleep.bed\_light\_di) to 100

Sending message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload {"state": "on",  
↪ "brightness": 254.0}

Received message with topic zigbee\_ffe/ffe/sleep/bed\_light\_di and payload b'{"state": "on",  
↪ "brightness": 254.0}'

Received message with topic videv/ffe/sleep/bed\_light\_di/brightness and payload b'100'

**Success** Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di) is correct (Content 100 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di)): 100 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffe.sleep.bed\_light\_di)): result = 100 (<class 'int'>)

### A.1.30 ViDevHeating.temp\_setp (ffe.sleep.heating\_valve) → HeatingValve.temp\_setp (ffe.sleep.heating\_valve)

#### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Setting devices to last state 30

Sending message with topic videv/ffe/sleep/heating\_valve/user\_temperature\_setpoint/set and  
↪ payload 30

Sending message with topic zigbee\_ffe/ffe/sleep/heating\_valve and payload  
↪ {"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}

Received message with topic zigbee\_ffe/ffe/sleep/heating\_valve/set and payload  
↪ b'{"current\_heating\_setpoint": 30}'

Received message with topic videv/ffe/sleep/heating\_valve/valve\_temperature\_setpoint and  
↪ payload b'30'

Received message with topic videv/ffe/sleep/heating\_valve/user\_temperature\_setpoint and  
↪ payload b'30'

Received message with topic zigbee\_ffe/ffe/sleep/heating\_valve and payload  
↪ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

**Success** Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (30, 30) (<class 'tuple'>)

```
Expectation (Start state (master, slave)): result = (30, 30) (<class 'tuple'>)
```

---

**Info** Setting state of ViDevHeating.temp\_setp (ffe.sleep.heating\_valve) to 15

---

```
Sending message with topic videv/ffe/sleep/heating_valve/user_temperature_setpoint/set and
↳ payload 15
```

```
Received message with topic zigbee_ffe/ffe/sleep/heating_valve/set and payload
↳ b'{"current_heating_setpoint": 15}'
```

```
Sending message with topic zigbee_ffe/ffe/sleep/heating_valve and payload
↳ {"current_heating_setpoint": 15, "local_temperature": 20.7, "battery": 97}
```

```
Received message with topic videv/ffe/sleep/heating_valve/valve_temperature_setpoint and
↳ payload b'15'
```

```
Received message with topic videv/ffe/sleep/heating_valve/user_temperature_setpoint and
↳ payload b'15'
```

```
Received message with topic zigbee_ffe/ffe/sleep/heating_valve and payload
↳ b'{"current_heating_setpoint": 15, "local_temperature": 20.7, "battery": 97}'
```

---

**Success** Value for HeatingValve.temp\_setp (ffe.sleep.heating\_valve) is correct (Content 15 and Type is <class 'int'>).

---

```
Result (Value for HeatingValve.temp_setp (ffe.sleep.heating_valve)): 15 (<class 'int'>)
```

```
Expectation (Value for HeatingValve.temp_setp (ffe.sleep.heating_valve)): result = 15 (<class
↳ 'int'>)
```

---

**Info** Setting state of ViDevHeating.temp\_setp (ffe.sleep.heating\_valve) to 20

---

```
Sending message with topic videv/ffe/sleep/heating_valve/user_temperature_setpoint/set and
↳ payload 20
```

```
Received message with topic zigbee_ffe/ffe/sleep/heating_valve/set and payload
↳ b'{"current_heating_setpoint": 20}'
```

```
Sending message with topic zigbee_ffe/ffe/sleep/heating_valve and payload
↳ {"current_heating_setpoint": 20, "local_temperature": 20.7, "battery": 97}
```

```
Received message with topic videv/ffe/sleep/heating_valve/valve_temperature_setpoint and
↳ payload b'20'
```

```
Received message with topic videv/ffe/sleep/heating_valve/user_temperature_setpoint and
↳ payload b'20'
```

```
Received message with topic zigbee_ffe/ffe/sleep/heating_valve and payload
↳ b'{"current_heating_setpoint": 20, "local_temperature": 20.7, "battery": 97}'
```

---

**Success** Value for HeatingValve.temp\_setp (ffe.sleep.heating\_valve) is correct (Content 20 and Type is <class 'int'>).

---

```
Result (Value for HeatingValve.temp_setp (ffe.sleep.heating_valve)): 20 (<class 'int'>)
```

```
Expectation (Value for HeatingValve.temp_setp (ffe.sleep.heating_valve)): result = 20 (<class
↳ 'int'>)
```

---

**Info**    Setting state of ViDevHeating.temp\_setp (ffe.sleep.heating\_valve) to 25

---

Sending message with topic videv/ffe/sleep/heating\_valve/user\_temperature\_setpoint/set and  
 ↪ payload 25

Received message with topic zigbee\_ffe/ffe/sleep/heating\_valve/set and payload  
 ↪ b'{"current\_heating\_setpoint": 25}'

Sending message with topic zigbee\_ffe/ffe/sleep/heating\_valve and payload  
 ↪ {"current\_heating\_setpoint": 25, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffe/sleep/heating\_valve/valve\_temperature\_setpoint and  
 ↪ payload b'25'

Received message with topic videv/ffe/sleep/heating\_valve/user\_temperature\_setpoint and  
 ↪ payload b'25'

Received message with topic zigbee\_ffe/ffe/sleep/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 25, "local\_temperature": 20.7, "battery": 97}'

---

**Success**    Value for HeatingValve.temp\_setp (ffe.sleep.heating\_valve) is correct (Content 25 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (ffe.sleep.heating\_valve)): 25 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffe.sleep.heating\_valve)): result = 25 (<class 'int'>)  
 ↪ 'int'>)

---

**Info**    Setting state of ViDevHeating.temp\_setp (ffe.sleep.heating\_valve) to 30

---

Sending message with topic videv/ffe/sleep/heating\_valve/user\_temperature\_setpoint/set and  
 ↪ payload 30

Received message with topic zigbee\_ffe/ffe/sleep/heating\_valve/set and payload  
 ↪ b'{"current\_heating\_setpoint": 30}'

Sending message with topic zigbee\_ffe/ffe/sleep/heating\_valve and payload  
 ↪ {"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffe/sleep/heating\_valve/valve\_temperature\_setpoint and  
 ↪ payload b'30'

Received message with topic videv/ffe/sleep/heating\_valve/user\_temperature\_setpoint and  
 ↪ payload b'30'

Received message with topic zigbee\_ffe/ffe/sleep/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

---

**Success**    Value for HeatingValve.temp\_setp (ffe.sleep.heating\_valve) is correct (Content 30 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (ffe.sleep.heating\_valve)): 30 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffe.sleep.heating\_valve)): result = 30 (<class 'int'>)  
 ↪ 'int'>)

**A.1.31 ViDevCommon.state (ffe.diningroom.main\_light) → Shelly.relay/0 (ffe.diningroom.main\_light)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffe/diningroom/main\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (ffe.diningroom.main\_light) to True

---

Sending message with topic videv/ffe/diningroom/main\_light/state/set and payload true

Received message with topic shellies/ffe/diningroom/main\_light/relay/0/command and payload  
↪ b'on'

Sending message with topic shellies/ffe/diningroom/main\_light/relay/0 and payload on

Received message with topic shellies/ffe/diningroom/main\_light/relay/0 and payload b'on'

Received message with topic videv/ffe/diningroom/main\_light/state and payload b'true'

Received message with topic zigbee\_ffe/ffe/diningroom/floor\_light/set and payload b'{"state":  
↪ "on"}'

Sending message with topic zigbee\_ffe/ffe/diningroom/floor\_light and payload {"state": "on"}

Received message with topic zigbee\_ffe/ffe/diningroom/floor\_light and payload b'{"state":  
↪ "on"}'

---

**Success** Value for Shelly.relay/0 (ffe.diningroom.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (ffe.diningroom.main\_light)): True (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (ffe.diningroom.main\_light)): result = True (<class  
↪ 'bool'>)

---

**Info** Setting state of ViDevCommon.state (ffe.diningroom.main\_light) to False

---

Sending message with topic videv/ffe/diningroom/main\_light/state/set and payload false

Received message with topic videv/ffe/diningroom/floorlamp/state and payload b'true'

Received message with topic shellies/ffe/diningroom/main\_light/relay/0/command and payload  
↪ b'off'

Sending message with topic shellies/ffe/diningroom/main\_light/relay/0 and payload off

Received message with topic shellies/ffe/diningroom/main\_light/relay/0 and payload b'off'

---

**Success** Value for Shelly.relay/0 (ffe.diningroom.main\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (ffe.diningroom.main\_light)): False (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (ffe.diningroom.main\_light)): result = False (<class  
↪ 'bool'>)

---

**A.1.32 Shelly.relay/0 (ffe.diningroom.main\_light) → ViDevCommon.state (ffe.diningroom.main\_light)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffe/diningroom/main\_light/state/set and payload false

Received message with topic videv/ffe/diningroom/main\_light/state and payload b'false'

Received message with topic zigbee\_ffe/ffe/diningroom/floor\_light/set and payload b'{"state":  
↪ "off"}'

Sending message with topic zigbee\_ffe/ffe/diningroom/floor\_light and payload {"state": "off"}

Received message with topic zigbee\_ffe/ffe/diningroom/floor\_light and payload b'{"state":  
↪ "off"}'

Received message with topic videv/ffe/diningroom/floorlamp/state and payload b'false'

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Shelly.relay/0 (ffe.diningroom.main\_light) to True

---

Sending message with topic shellies/ffe/diningroom/main\_light/relay/0 and payload on

Received message with topic shellies/ffe/diningroom/main\_light/relay/0 and payload b'on'

Received message with topic videv/ffe/diningroom/main\_light/state and payload b'true'

Received message with topic zigbee\_ffe/ffe/diningroom/floor\_light/set and payload b'{"state":  
↪ "on"}'

Sending message with topic zigbee\_ffe/ffe/diningroom/floor\_light and payload {"state": "on"}

Received message with topic zigbee\_ffe/ffe/diningroom/floor\_light and payload b'{"state":  
↪ "on"}'

Received message with topic videv/ffe/diningroom/floorlamp/state and payload b'true'

---

**Success** Value for ViDevCommon.state (ffe.diningroom.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffe.diningroom.main\_light)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.diningroom.main\_light)): result = True (<class  
↪ 'bool'>)

---

**Info** Setting state of Shelly.relay/0 (ffe.diningroom.main\_light) to False

---

Sending message with topic shellies/ffe/diningroom/main\_light/relay/0 and payload off

Received message with topic shellies/ffe/diningroom/main\_light/relay/0 and payload b'off'

Received message with topic videv/ffe/diningroom/main\_light/state and payload b'false'

---

```
Received message with topic zigbee_ffe/ffe/diningroom/floor_light/set and payload b'{"state":
↪  "off"}'
```

```
Sending message with topic zigbee_ffe/ffe/diningroom/floor_light and payload {"state": "off"}
```

```
Received message with topic zigbee_ffe/ffe/diningroom/floor_light and payload b'{"state":
↪  "off"}'
```

```
Received message with topic videv/ffe/diningroom/floorlamp/state and payload b'false'
```

---

**Success** Value for ViDevCommon.state (ffe.diningroom.main\_light) is correct (Content False and Type is <class 'bool'>).

---

```
Result (Value for ViDevCommon.state (ffe.diningroom.main_light)): False (<class 'bool'>)
```

```
Expectation (Value for ViDevCommon.state (ffe.diningroom.main_light)): result = False (<class
↪  'bool'>)
```

### A.1.33 ViDevCommon.state (ffe.diningroom.floorlamp) → Powerplug1P.state (ffe.diningroom.floor\_light)

#### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

```
Sending message with topic videv/ffe/diningroom/floorlamp/state/set and payload false
```

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

```
Result (Start state (master, slave)): (False, False) (<class 'tuple'>)
```

```
Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)
```

---

**Info** Setting state of ViDevCommon.state (ffe.diningroom.floorlamp) to True

---

```
Sending message with topic videv/ffe/diningroom/floorlamp/state/set and payload true
```

```
Received message with topic zigbee_ffe/ffe/diningroom/floor_light/set and payload b'{"state":
↪  "on"}'
```

```
Sending message with topic zigbee_ffe/ffe/diningroom/floor_light and payload {"state": "on"}
```

```
Received message with topic zigbee_ffe/ffe/diningroom/floor_light and payload b'{"state":
↪  "on"}'
```

```
Received message with topic videv/ffe/diningroom/floorlamp/state and payload b'true'
```

---

**Success** Value for Powerplug1P.state (ffe.diningroom.floor\_light) is correct (Content True and Type is <class 'bool'>).

---

```
Result (Value for Powerplug1P.state (ffe.diningroom.floor_light)): True (<class 'bool'>)
```

```
Expectation (Value for Powerplug1P.state (ffe.diningroom.floor_light)): result = True (<class
↪  'bool'>)
```

---

**Info** Setting state of ViDevCommon.state (ffe.diningroom.floorlamp) to False

---

Sending message with topic videv/ffe/diningroom/floorlamp/state/set and payload false

Received message with topic zigbee\_ffe/ffe/diningroom/floor\_light/set and payload b'{"state":  
↪ "off"}'

Sending message with topic zigbee\_ffe/ffe/diningroom/floor\_light and payload {"state": "off"}

Received message with topic zigbee\_ffe/ffe/diningroom/floor\_light and payload b'{"state":  
↪ "off"}'

Received message with topic videv/ffe/diningroom/floorlamp/state and payload b'false'

---

**Success** Value for Powerplug1P.state (ffe.diningroom.floor\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Powerplug1P.state (ffe.diningroom.floor\_light)): False (<class 'bool'>)

Expectation (Value for Powerplug1P.state (ffe.diningroom.floor\_light)): result = False (<class  
↪ 'bool'>)

---

#### A.1.34 Powerplug1P.state (ffe.diningroom.floor\_light) → ViDevCommon.state (ffe.diningroom.floorlamp)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffe/diningroom/floorlamp/state/set and payload false

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Powerplug1P.state (ffe.diningroom.floor\_light) to True

---

Sending message with topic zigbee\_ffe/ffe/diningroom/floor\_light and payload {"state": "on"}

Received message with topic zigbee\_ffe/ffe/diningroom/floor\_light and payload b'{"state":  
↪ "on"}'

Received message with topic videv/ffe/diningroom/floorlamp/state and payload b'true'

---

**Success** Value for ViDevCommon.state (ffe.diningroom.floorlamp) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffe.diningroom.floorlamp)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.diningroom.floorlamp)): result = True (<class  
↪ 'bool'>)

---

---

**Info** Setting state of Powerplug1P.state (ffe.diningroom.floor\_light) to False

---

Sending message with topic zigbee\_ffe/ffe/diningroom/floor\_light and payload {"state": "off"}

Received message with topic zigbee\_ffe/ffe/diningroom/floor\_light and payload b'{"state":  
↪ "off"}'

Received message with topic videv/ffe/diningroom/floorlamp/state and payload b'false'

---

**Success** Value for ViDevCommon.state (ffe.diningroom.floorlamp) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffe.diningroom.floorlamp)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.diningroom.floorlamp)): result = False (<class  
↪ 'bool'>)

---

#### A.1.35 Shelly.relay/0 (ffe.diningroom.main\_light) → Powerplug1P.state (ffe.diningroom.floor\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Shelly.relay/0 (ffe.diningroom.main\_light) to True

---

Sending message with topic shellies/ffe/diningroom/main\_light/relay/0 and payload on

Received message with topic shellies/ffe/diningroom/main\_light/relay/0 and payload b'on'

Received message with topic videv/ffe/diningroom/main\_light/state and payload b'true'

Received message with topic zigbee\_ffe/ffe/diningroom/floor\_light/set and payload b'{"state":  
↪ "on"}'

Sending message with topic zigbee\_ffe/ffe/diningroom/floor\_light and payload {"state": "on"}

Received message with topic zigbee\_ffe/ffe/diningroom/floor\_light and payload b'{"state":  
↪ "on"}'

Received message with topic videv/ffe/diningroom/floorlamp/state and payload b'true'

---

**Success** Value for Powerplug1P.state (ffe.diningroom.floor\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Powerplug1P.state (ffe.diningroom.floor\_light)): True (<class 'bool'>)

---

Expectation (Value for Powerplug1P.state (ffe.diningroom.floor\_light)): result = True (<class 'bool'>)

**Info** Setting state of Shelly.relay/0 (ffe.diningroom.main\_light) to False

Sending message with topic shellies/ffe/diningroom/main\_light/relay/0 and payload off

Received message with topic shellies/ffe/diningroom/main\_light/relay/0 and payload b'off'

Received message with topic videv/ffe/diningroom/main\_light/state and payload b'false'

Received message with topic zigbee\_ffe/ffe/diningroom/floor\_light/set and payload b'{"state": "off"}'

Sending message with topic zigbee\_ffe/ffe/diningroom/floor\_light and payload {"state": "off"}

Received message with topic zigbee\_ffe/ffe/diningroom/floor\_light and payload b'{"state": "off"}'

Received message with topic videv/ffe/diningroom/floorlamp/state and payload b'false'

**Success** Value for Powerplug1P.state (ffe.diningroom.floor\_light) is correct (Content False and Type is <class 'bool'>).

Result (Value for Powerplug1P.state (ffe.diningroom.floor\_light)): False (<class 'bool'>)

Expectation (Value for Powerplug1P.state (ffe.diningroom.floor\_light)): result = False (<class 'bool'>)

#### A.1.36 ViDevCommon.state (ffe.diningroom.garland) → Powerplug1P.state (ffe.diningroom.garland)

##### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Setting devices to last state False

Sending message with topic videv/ffe/diningroom/garland/state/set and payload false

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

**Info** Setting state of ViDevCommon.state (ffe.diningroom.garland) to True

Sending message with topic videv/ffe/diningroom/garland/state/set and payload true

Received message with topic zigbee\_ffe/ffe/diningroom/garland/set and payload b'{"state": "on"}'

Sending message with topic zigbee\_ffe/ffe/diningroom/garland and payload {"state": "on"}

Received message with topic zigbee\_ffe/ffe/diningroom/garland and payload b'{"state": "on"}'

Received message with topic videv/ffe/diningroom/garland/state and payload b'true'

---

**Success** Value for Powerplug1P.state (ffe.diningroom.garland) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Powerplug1P.state (ffe.diningroom.garland)): True (<class 'bool'>)

Expectation (Value for Powerplug1P.state (ffe.diningroom.garland)): result = True (<class 'bool'>)

---

**Info** Setting state of ViDevCommon.state (ffe.diningroom.garland) to False

---

Sending message with topic videv/ffe/diningroom/garland/state/set and payload false

Received message with topic zigbee\_ffe/ffe/diningroom/garland/set and payload b'{"state":  
↪ "off"}'

Sending message with topic zigbee\_ffe/ffe/diningroom/garland and payload {"state": "off"}

Received message with topic zigbee\_ffe/ffe/diningroom/garland and payload b'{"state": "off"}'

Received message with topic videv/ffe/diningroom/garland/state and payload b'false'

---

**Success** Value for Powerplug1P.state (ffe.diningroom.garland) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Powerplug1P.state (ffe.diningroom.garland)): False (<class 'bool'>)

Expectation (Value for Powerplug1P.state (ffe.diningroom.garland)): result = False (<class 'bool'>)

---

#### A.1.37 Powerplug1P.state (ffe.diningroom.garland) → ViDevCommon.state (ffe.diningroom.garland)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffe/diningroom/garland/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Powerplug1P.state (ffe.diningroom.garland) to True

---

Sending message with topic zigbee\_ffe/ffe/diningroom/garland and payload {"state": "on"}

Received message with topic zigbee\_ffe/ffe/diningroom/garland and payload b'{"state": "on"}'

Received message with topic videv/ffe/diningroom/garland/state and payload b'true'

---

---

**Success** Value for ViDevCommon.state (ffe.diningroom.garland) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffe.diningroom.garland)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.diningroom.garland)): result = True (<class 'bool'>)

---

**Info** Setting state of Powerplug1P.state (ffe.diningroom.garland) to False

---

Sending message with topic zigbee\_ffe/ffe/diningroom/garland and payload {"state": "off"}

Received message with topic zigbee\_ffe/ffe/diningroom/garland and payload b'{"state": "off"}'

Received message with topic videv/ffe/diningroom/garland/state and payload b'false'

---

**Success** Value for ViDevCommon.state (ffe.diningroom.garland) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffe.diningroom.garland)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.diningroom.garland)): result = False (<class 'bool'>)

---

#### A.1.38 ViDevCommon.state (ffe.kitchen.main\_light) → Shelly.relay/0 (ffe.kitchen.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffe/kitchen/main\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (ffe.kitchen.main\_light) to True

---

Sending message with topic videv/ffe/kitchen/main\_light/state/set and payload true

Received message with topic shellies/ffe/kitchen/main\_light/relay/0/command and payload b'on'

Sending message with topic shellies/ffe/kitchen/main\_light/relay/0 and payload on

Received message with topic shellies/ffe/kitchen/main\_light/relay/0 and payload b'on'

Received message with topic videv/ffe/kitchen/main\_light/state and payload b'true'

---

**Success** Value for Shelly.relay/0 (ffe.kitchen.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (ffe.kitchen.main\_light)): True (<class 'bool'>)

---

```
Expectation (Value for Shelly.relay/0 (ffe.kitchen.main_light)): result = True (<class
↳ 'bool'>)
```

---

**Info** Setting state of ViDevCommon.state (ffe.kitchen.main\_light) to False

---

```
Sending message with topic videv/ffe/kitchen/main_light/state/set and payload false
```

```
Received message with topic shellies/ffe/kitchen/main_light/relay/0/command and payload b'off'
```

```
Sending message with topic shellies/ffe/kitchen/main_light/relay/0 and payload off
```

```
Received message with topic shellies/ffe/kitchen/main_light/relay/0 and payload b'off'
```

```
Received message with topic videv/ffe/kitchen/main_light/state and payload b'false'
```

---

**Success** Value for Shelly.relay/0 (ffe.kitchen.main\_light) is correct (Content False and Type is <class 'bool'>).

---

```
Result (Value for Shelly.relay/0 (ffe.kitchen.main_light)): False (<class 'bool'>)
```

```
Expectation (Value for Shelly.relay/0 (ffe.kitchen.main_light)): result = False (<class
↳ 'bool'>)
```

### A.1.39 Shelly.relay/0 (ffe.kitchen.main\_light) → ViDevCommon.state (ffe.kitchen.main\_light)

#### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

```
Sending message with topic videv/ffe/kitchen/main_light/state/set and payload false
```

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

```
Result (Start state (master, slave)): (False, False) (<class 'tuple'>)
```

```
Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)
```

---

**Info** Setting state of Shelly.relay/0 (ffe.kitchen.main\_light) to True

---

```
Sending message with topic shellies/ffe/kitchen/main_light/relay/0 and payload on
```

```
Received message with topic shellies/ffe/kitchen/main_light/relay/0 and payload b'on'
```

```
Received message with topic videv/ffe/kitchen/main_light/state and payload b'true'
```

---

**Success** Value for ViDevCommon.state (ffe.kitchen.main\_light) is correct (Content True and Type is <class 'bool'>).

---

```
Result (Value for ViDevCommon.state (ffe.kitchen.main_light)): True (<class 'bool'>)
```

```
Expectation (Value for ViDevCommon.state (ffe.kitchen.main_light)): result = True (<class
↳ 'bool'>)
```

---

**Info** Setting state of Shelly.relay/0 (ffe.kitchen.main\_light) to False

---

```
Sending message with topic shellies/ffe/kitchen/main_light/relay/0 and payload off
```

Received message with topic shellies/ffe/kitchen/main\_light/relay/0 and payload b'off'

Received message with topic videv/ffe/kitchen/main\_light/state and payload b'false'

**Success** Value for ViDevCommon.state (ffe.kitchen.main\_light) is correct (Content False and Type is <class 'bool'>).

Result (Value for ViDevCommon.state (ffe.kitchen.main\_light)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.kitchen.main\_light)): result = False (<class 'bool'>)

#### A.1.40 ViDevCommon.state (ffe.kitchen.circulation\_pump) → Shelly.relay/0 (ffe.kitchen.circulation\_pump)

##### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Setting devices to last state False

Sending message with topic videv/ffe/kitchen/circulation\_pump/state/set and payload false

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

**Info** Setting state of ViDevCommon.state (ffe.kitchen.circulation\_pump) to True

Sending message with topic videv/ffe/kitchen/circulation\_pump/state/set and payload true

Received message with topic shellies/ffe/kitchen/circulation\_pump/relay/0/command and payload b'on'

Sending message with topic shellies/ffe/kitchen/circulation\_pump/relay/0 and payload on

Received message with topic shellies/ffe/kitchen/circulation\_pump/relay/0 and payload b'on'

Received message with topic videv/ffe/kitchen/circulation\_pump/timer and payload b'600'

Received message with topic shellies/ffe/kitchen/main\_light/relay/0/command and payload b'on'

Sending message with topic shellies/ffe/kitchen/main\_light/relay/0 and payload on

Received message with topic videv/ffe/kitchen/circulation\_pump/state and payload b'true'

Received message with topic shellies/ffe/kitchen/main\_light/relay/0 and payload b'on'

**Success** Value for Shelly.relay/0 (ffe.kitchen.circulation\_pump) is correct (Content True and Type is <class 'bool'>).

Result (Value for Shelly.relay/0 (ffe.kitchen.circulation\_pump)): True (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (ffe.kitchen.circulation\_pump)): result = True (<class 'bool'>)

**Info** Setting state of ViDevCommon.state (ffe.kitchen.circulation\_pump) to False

Sending message with topic videv/ffe/kitchen/circulation\_pump/state/set and payload false

```
Received message with topic videv/ffe/kitchen/main_light/state and payload b'true'
Received message with topic shellies/ffe/kitchen/circulation_pump/relay/0/command and payload
↪ b'off'
Sending message with topic shellies/ffe/kitchen/circulation_pump/relay/0 and payload off
Received message with topic shellies/ffe/kitchen/circulation_pump/relay/0 and payload b'off'
```

---

**Success** Value for Shelly.relay/0 (ffe.kitchen.circulation\_pump) is correct (Content False and Type is <class 'bool'>).

---

```
Result (Value for Shelly.relay/0 (ffe.kitchen.circulation_pump)): False (<class 'bool'>)
Expectation (Value for Shelly.relay/0 (ffe.kitchen.circulation_pump)): result = False (<class
↪ 'bool'>)
```

#### A.1.41 Shelly.relay/0 (ffe.kitchen.circulation\_pump) → ViDevCommon.state (ffe.kitchen.circulation\_pump)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

```
Sending message with topic videv/ffe/kitchen/circulation_pump/state/set and payload false
Received message with topic videv/ffe/kitchen/circulation_pump/timer and payload b'0'
Received message with topic videv/ffe/kitchen/circulation_pump/state and payload b'false'
```

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

```
Result (Start state (master, slave)): (False, False) (<class 'tuple'>)
Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)
```

---

**Info** Setting state of Shelly.relay/0 (ffe.kitchen.circulation\_pump) to True

---

```
Sending message with topic shellies/ffe/kitchen/circulation_pump/relay/0 and payload on
Received message with topic shellies/ffe/kitchen/circulation_pump/relay/0 and payload b'on'
Received message with topic videv/ffe/kitchen/circulation_pump/timer and payload b'600'
Received message with topic shellies/ffe/kitchen/main_light/relay/0/command and payload b'off'
Sending message with topic shellies/ffe/kitchen/main_light/relay/0 and payload off
Received message with topic videv/ffe/kitchen/circulation_pump/state and payload b'true'
Received message with topic shellies/ffe/kitchen/main_light/relay/0 and payload b'off'
Received message with topic videv/ffe/kitchen/main_light/state and payload b'false'
```

---

**Success** Value for ViDevCommon.state (ffe.kitchen.circulation\_pump) is correct (Content True and Type is <class 'bool'>).

---

```
Result (Value for ViDevCommon.state (ffe.kitchen.circulation_pump)): True (<class 'bool'>)
Expectation (Value for ViDevCommon.state (ffe.kitchen.circulation_pump)): result = True
↪ (<class 'bool'>)
```

---

**Info** Setting state of Shelly.relay/0 (ffe.kitchen.circulation\_pump) to False

---

Sending message with topic shellies/ffe/kitchen/circulation\_pump/relay/0 and payload off

Received message with topic shellies/ffe/kitchen/circulation\_pump/relay/0 and payload b'off'

Received message with topic videv/ffe/kitchen/circulation\_pump/timer and payload b'0'

Received message with topic videv/ffe/kitchen/circulation\_pump/state and payload b'false'

---

**Success** Value for ViDevCommon.state (ffe.kitchen.circulation\_pump) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffe.kitchen.circulation\_pump)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffe.kitchen.circulation\_pump)): result = False  
↪ (<class 'bool'>)

---

**A.1.42 ViDevHeating.temp\_setp (ffe.kitchen.heating\_valve) → HeatingValve.temp\_setp (ffe.kitchen.heating\_valve)**

#### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state 30

---

Sending message with topic videv/ffe/kitchen/heating\_valve/user\_temperature\_setpoint/set and  
↪ payload 30

Sending message with topic zigbee\_ffe/ffe/kitchen/heating\_valve and payload  
↪ {"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}

Received message with topic zigbee\_ffe/ffe/kitchen/heating\_valve and payload  
↪ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

Received message with topic zigbee\_ffe/ffe/kitchen/heating\_valve/set and payload  
↪ b'{"current\_heating\_setpoint": 30}'

Received message with topic videv/ffe/kitchen/heating\_valve/valve\_temperature\_setpoint and  
↪ payload b'30'

Received message with topic videv/ffe/kitchen/heating\_valve/user\_temperature\_setpoint and  
↪ payload b'30'

---

**Success** Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (30, 30) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (30, 30) (<class 'tuple'>)

---

**Info** Setting state of ViDevHeating.temp\_setp (ffe.kitchen.heating\_valve) to 15

---

Sending message with topic videv/ffe/kitchen/heating\_valve/user\_temperature\_setpoint/set and  
↪ payload 15

Received message with topic zigbee\_ffe/ffe/kitchen/heating\_valve/set and payload

↪ b'{"current\_heating\_setpoint": 15}'

Sending message with topic zigbee\_ffe/ffe/kitchen/heating\_valve and payload

↪ {"current\_heating\_setpoint": 15, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffe/kitchen/heating\_valve/valve\_temperature\_setpoint and

↪ payload b'15'

Received message with topic videv/ffe/kitchen/heating\_valve/user\_temperature\_setpoint and

↪ payload b'15'

Received message with topic zigbee\_ffe/ffe/kitchen/heating\_valve and payload

↪ b'{"current\_heating\_setpoint": 15, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Value for HeatingValve.temp\_setp (ffe.kitchen.heating\_valve) is correct (Content 15 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (ffe.kitchen.heating\_valve)): 15 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffe.kitchen.heating\_valve)): result = 15

↪ (<class 'int'>)

---

**Info** Setting state of ViDevHeating.temp\_setp (ffe.kitchen.heating\_valve) to 20

---

Sending message with topic videv/ffe/kitchen/heating\_valve/user\_temperature\_setpoint/set and

↪ payload 20

Received message with topic zigbee\_ffe/ffe/kitchen/heating\_valve/set and payload

↪ b'{"current\_heating\_setpoint": 20}'

Sending message with topic zigbee\_ffe/ffe/kitchen/heating\_valve and payload

↪ {"current\_heating\_setpoint": 20, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffe/kitchen/heating\_valve/valve\_temperature\_setpoint and

↪ payload b'20'

Received message with topic videv/ffe/kitchen/heating\_valve/user\_temperature\_setpoint and

↪ payload b'20'

Received message with topic zigbee\_ffe/ffe/kitchen/heating\_valve and payload

↪ b'{"current\_heating\_setpoint": 20, "local\_temperature": 20.7, "battery": 97}'

Received message with topic shellies/ffe/kitchen/main\_light/relay/0/command and payload b'on'

Sending message with topic shellies/ffe/kitchen/main\_light/relay/0 and payload on

Received message with topic shellies/ffe/kitchen/main\_light/relay/0 and payload b'on'

---

**Success** Value for HeatingValve.temp\_setp (ffe.kitchen.heating\_valve) is correct (Content 20 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (ffe.kitchen.heating\_valve)): 20 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffe.kitchen.heating\_valve)): result = 20

↪ (<class 'int'>)

---

**Info** Setting state of ViDevHeating.temp\_setp (ffe.kitchen.heating\_valve) to 25

---

Sending message with topic videv/ffe/kitchen/heating\_valve/user\_temperature\_setpoint/set and

↪ payload 25

Received message with topic videv/ffe/kitchen/main\_light/state and payload b'true'

Received message with topic zigbee\_ffe/ffe/kitchen/heating\_valve/set and payload

↪ b'{"current\_heating\_setpoint": 25}'

Sending message with topic zigbee\_ffe/ffe/kitchen/heating\_valve and payload

↪ {"current\_heating\_setpoint": 25, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffe/kitchen/heating\_valve/valve\_temperature\_setpoint and

↪ payload b'25'

Received message with topic videv/ffe/kitchen/heating\_valve/user\_temperature\_setpoint and

↪ payload b'25'

Received message with topic zigbee\_ffe/ffe/kitchen/heating\_valve and payload

↪ b'{"current\_heating\_setpoint": 25, "local\_temperature": 20.7, "battery": 97}'

**Success** Value for HeatingValve.temp\_setp (ffe.kitchen.heating\_valve) is correct (Content 25 and Type is <class 'int'>).

Result (Value for HeatingValve.temp\_setp (ffe.kitchen.heating\_valve)): 25 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffe.kitchen.heating\_valve)): result = 25

↪ (<class 'int'>)

**Info** Setting state of ViDevHeating.temp\_setp (ffe.kitchen.heating\_valve) to 30

Sending message with topic videv/ffe/kitchen/heating\_valve/user\_temperature\_setpoint/set and

↪ payload 30

Received message with topic zigbee\_ffe/ffe/kitchen/heating\_valve/set and payload

↪ b'{"current\_heating\_setpoint": 30}'

Sending message with topic zigbee\_ffe/ffe/kitchen/heating\_valve and payload

↪ {"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffe/kitchen/heating\_valve/valve\_temperature\_setpoint and

↪ payload b'30'

Received message with topic videv/ffe/kitchen/heating\_valve/user\_temperature\_setpoint and

↪ payload b'30'

**Success** Value for HeatingValve.temp\_setp (ffe.kitchen.heating\_valve) is correct (Content 30 and Type is <class 'int'>).

Result (Value for HeatingValve.temp\_setp (ffe.kitchen.heating\_valve)): 30 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffe.kitchen.heating\_valve)): result = 30

↪ (<class 'int'>)

#### A.1.43 ViDevCommon.state (ffe.floor.main\_light) → Shelly.relay/0 (ffe.floor.main\_light)

##### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Setting devices to last state False

Sending message with topic videv/ffe/floor/main\_light/state/set and payload false

Received message with topic zigbee\_ffe/ffe/kitchen/heating\_valve and payload  
 ↳ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

**Info** Setting state of ViDevCommon.state (ffe.floor.main\_light) to True

Sending message with topic videv/ffe/floor/main\_light/state/set and payload true

Received message with topic shellies/ffe/floor/main\_light/relay/0/command and payload b'on'

Sending message with topic shellies/ffe/floor/main\_light/relay/0 and payload on

Received message with topic shellies/ffe/floor/main\_light/relay/0 and payload b'on'

Received message with topic videv/ffe/floor/main\_light/state and payload b'true'

**Success** Value for Shelly.relay/0 (ffe.floor.main\_light) is correct (Content True and Type is <class 'bool'>).

Result (Value for Shelly.relay/0 (ffe.floor.main\_light)): True (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (ffe.floor.main\_light)): result = True (<class 'bool'>)

**Info** Setting state of ViDevCommon.state (ffe.floor.main\_light) to False

Sending message with topic videv/ffe/floor/main\_light/state/set and payload false

Received message with topic shellies/ffe/floor/main\_light/relay/0/command and payload b'off'

Sending message with topic shellies/ffe/floor/main\_light/relay/0 and payload off

Received message with topic shellies/ffe/floor/main\_light/relay/0 and payload b'off'

Received message with topic videv/ffe/floor/main\_light/state and payload b'false'

**Success** Value for Shelly.relay/0 (ffe.floor.main\_light) is correct (Content False and Type is <class 'bool'>).

Result (Value for Shelly.relay/0 (ffe.floor.main\_light)): False (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (ffe.floor.main\_light)): result = False (<class 'bool'>)

#### A.1.44 Shelly.relay/0 (ffe.floor.main\_light) → ViDevCommon.state (ffe.floor.main\_light)

##### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Setting devices to last state False

Sending message with topic videv/ffe/floor/main\_light/state/set and payload false

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

```
Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)
```

---

**Info** Setting state of Shelly.relay/0 (ffe.floor.main\_light) to True

---

```
Sending message with topic shellies/ffe/floor/main_light/relay/0 and payload on
```

```
Received message with topic shellies/ffe/floor/main_light/relay/0 and payload b'on'
```

```
Received message with topic videv/ffe/floor/main_light/state and payload b'true'
```

---

**Success** Value for ViDevCommon.state (ffe.floor.main\_light) is correct (Content True and Type is <class 'bool'>).

---

```
Result (Value for ViDevCommon.state (ffe.floor.main_light)): True (<class 'bool'>)
```

```
Expectation (Value for ViDevCommon.state (ffe.floor.main_light)): result = True (<class  
↪ 'bool'>)
```

---

**Info** Setting state of Shelly.relay/0 (ffe.floor.main\_light) to False

---

```
Sending message with topic shellies/ffe/floor/main_light/relay/0 and payload off
```

```
Received message with topic shellies/ffe/floor/main_light/relay/0 and payload b'off'
```

```
Received message with topic videv/ffe/floor/main_light/state and payload b'false'
```

---

**Success** Value for ViDevCommon.state (ffe.floor.main\_light) is correct (Content False and Type is <class 'bool'>).

---

```
Result (Value for ViDevCommon.state (ffe.floor.main_light)): False (<class 'bool'>)
```

```
Expectation (Value for ViDevCommon.state (ffe.floor.main_light)): result = False (<class  
↪ 'bool'>)
```

#### A.1.45 ViDevCommon.state (ffw.livingroom.main\_light) → Shelly.relay/0 (ffw.livingroom.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

```
Sending message with topic videv/ffw/livingroom/main_light/state/set and payload false
```

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

```
Result (Start state (master, slave)): (False, False) (<class 'tuple'>)
```

```
Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)
```

---

**Info** Setting state of ViDevCommon.state (ffw.livingroom.main\_light) to True

---

```
Sending message with topic videv/ffw/livingroom/main_light/state/set and payload true
```

```
Received message with topic shellies/ffw/livingroom/main_light/relay/0/command and payload  
↪ b'on'
```

Sending message with topic shellies/ffw/livingroom/main\_light/relay/0 and payload on

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic shellies/ffw/livingroom/main\_light/relay/0 and payload b'on'

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/livingroom/main\_light/state and payload b'true'

Received message with topic videv/ffw/livingroom/main\_light/brightness and payload b'50'

Received message with topic videv/ffw/livingroom/main\_light/color\_temp and payload b'5'

---

**Success** Value for Shelly.relay/0 (ffw.livingroom.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (ffw.livingroom.main\_light)): True (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (ffw.livingroom.main\_light)): result = True (<class  
↪ 'bool'>)

---

**Info** Setting state of ViDevCommon.state (ffw.livingroom.main\_light) to False

---

Sending message with topic videv/ffw/livingroom/main\_light/state/set and payload false

Received message with topic shellies/ffw/livingroom/main\_light/relay/0/command and payload  
↪ b'off'

Sending message with topic shellies/ffw/livingroom/main\_light/relay/0 and payload off

Received message with topic shellies/ffw/livingroom/main\_light/relay/0 and payload b'off'

Received message with topic videv/ffw/livingroom/main\_light/state and payload b'false'

---

**Success** Value for Shelly.relay/0 (ffw.livingroom.main\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (ffw.livingroom.main\_light)): False (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (ffw.livingroom.main\_light)): result = False (<class  
↪ 'bool'>)

---

#### A.1.46 Shelly.relay/0 (ffw.livingroom.main\_light) → ViDevCommon.state (ffw.livingroom.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffw/livingroom/main\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

---

**Info** Setting state of Shelly.relay/0 (ffw.livingroom.main\_light) to True

---

Sending message with topic shellies/ffw/livingroom/main\_light/relay/0 and payload on

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic shellies/ffw/livingroom/main\_light/relay/0 and payload b'on'

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/livingroom/main\_light/state and payload b'true'

---

**Success** Value for ViDevCommon.state (ffw.livingroom.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffw.livingroom.main\_light)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffw.livingroom.main\_light)): result = True (<class  
↪ 'bool'>)

---

**Info** Setting state of Shelly.relay/0 (ffw.livingroom.main\_light) to False

---

Sending message with topic shellies/ffw/livingroom/main\_light/relay/0 and payload off

Received message with topic shellies/ffw/livingroom/main\_light/relay/0 and payload b'off'

Received message with topic videv/ffw/livingroom/main\_light/state and payload b'false'

---

**Success** Value for ViDevCommon.state (ffw.livingroom.main\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffw.livingroom.main\_light)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffw.livingroom.main\_light)): result = False (<class  
↪ 'bool'>)

---

#### A.1.47 ViDevCommon.brightness (ffw.livingroom.main\_light) → Light.brightness (ffw.livingroom.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---

Sending message with topic shellies/ffw/livingroom/main\_light/relay/0 and payload on

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic shellies/ffw/livingroom/main\_light/relay/0 and payload b'on'

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/livingroom/main\_light/state and payload b'true'

---

---

**Info** Prepare: Setting devices to last state 100

---

Sending message with topic videv/ffw/livingroom/main\_light/brightness/set and payload 100

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light/set and payload  
↪ b'{"brightness": 254}'

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/livingroom/main\_light/brightness and payload b'100'

---

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.brightness (ffw.livingroom.main\_light) to 0

---

Sending message with topic videv/ffw/livingroom/main\_light/brightness/set and payload 0

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light/set and payload  
↪ b'{"brightness": 1}'

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/livingroom/main\_light/brightness and payload b'0'

---

**Success** Value for Light.brightness (ffw.livingroom.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffw.livingroom.main\_light)): 0 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.livingroom.main\_light)): result = 0 (<class  
↪ 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffw.livingroom.main\_light) to 20

---

Sending message with topic videv/ffw/livingroom/main\_light/brightness/set and payload 20

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light/set and payload  
↪ b'{"brightness": 52}'

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/livingroom/main\_light/brightness and payload b'20'

---

---

**Success** Value for Light.brightness (ffw.livingroom.main\_light) is correct (Content 20 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffw.livingroom.main\_light)): 20 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.livingroom.main\_light)): result = 20 (<class 'int'>)  
↪ 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffw.livingroom.main\_light) to 40

---

Sending message with topic videv/ffw/livingroom/main\_light/brightness/set and payload 40

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light/set and payload  
↪ b'{"brightness": 102}'

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/livingroom/main\_light/brightness and payload b'40'

---

**Success** Value for Light.brightness (ffw.livingroom.main\_light) is correct (Content 40 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffw.livingroom.main\_light)): 40 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.livingroom.main\_light)): result = 40 (<class 'int'>)  
↪ 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffw.livingroom.main\_light) to 60

---

Sending message with topic videv/ffw/livingroom/main\_light/brightness/set and payload 60

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light/set and payload  
↪ b'{"brightness": 153}'

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/livingroom/main\_light/brightness and payload b'60'

---

**Success** Value for Light.brightness (ffw.livingroom.main\_light) is correct (Content 60 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffw.livingroom.main\_light)): 60 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.livingroom.main\_light)): result = 60 (<class 'int'>)  
↪ 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffw.livingroom.main\_light) to 80

---

Sending message with topic videv/ffw/livingroom/main\_light/brightness/set and payload 80

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light/set and payload  
 ↪ b'{"brightness": 203}'

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
 ↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
 ↪ "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/livingroom/main\_light/brightness and payload b'80'

---

**Success** Value for Light.brightness (ffw.livingroom.main\_light) is correct (Content 80 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffw.livingroom.main\_light)): 80 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.livingroom.main\_light)): result = 80 (<class  
 ↪ 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffw.livingroom.main\_light) to 100

---

Sending message with topic videv/ffw/livingroom/main\_light/brightness/set and payload 100

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light/set and payload  
 ↪ b'{"brightness": 254}'

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
 ↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
 ↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/livingroom/main\_light/brightness and payload b'100'

---

**Success** Value for Light.brightness (ffw.livingroom.main\_light) is correct (Content 100 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffw.livingroom.main\_light)): 100 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.livingroom.main\_light)): result = 100 (<class  
 ↪ 'int'>)

#### A.1.48 Light.brightness (ffw.livingroom.main\_light) → ViDevCommon.brightness (ffw.livingroom.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 100

---

Sending message with topic videv/ffw/livingroom/main\_light/brightness/set and payload 100

---

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

**Info** Setting state of Light.brightness (ffw.livingroom.main\_light) to 0

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/livingroom/main\_light/brightness and payload b'0'

**Success** Value for ViDevCommon.brightness (ffw.livingroom.main\_light) is correct (Content 0 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffw.livingroom.main\_light)): 0 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.livingroom.main\_light)): result = 0 (<class 'int'>)  
↪ ('int'>)

**Info** Setting state of Light.brightness (ffw.livingroom.main\_light) to 20

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/livingroom/main\_light/brightness and payload b'20'

**Success** Value for ViDevCommon.brightness (ffw.livingroom.main\_light) is correct (Content 20 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffw.livingroom.main\_light)): 20 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.livingroom.main\_light)): result = 20  
↪ (<class 'int'>)

**Info** Setting state of Light.brightness (ffw.livingroom.main\_light) to 40

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/livingroom/main\_light/brightness and payload b'40'

**Success** Value for ViDevCommon.brightness (ffw.livingroom.main\_light) is correct (Content 40 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffw.livingroom.main\_light)): 40 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.livingroom.main\_light)): result = 40  
↪ (<class 'int'>)

---

**Info** Setting state of Light.brightness (ffw.livingroom.main\_light) to 60

---

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/livingroom/main\_light/brightness and payload b'60'

---

**Success** Value for ViDevCommon.brightness (ffw.livingroom.main\_light) is correct (Content 60 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffw.livingroom.main\_light)): 60 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.livingroom.main\_light)): result = 60  
↪ (<class 'int'>)

---

**Info** Setting state of Light.brightness (ffw.livingroom.main\_light) to 80

---

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/livingroom/main\_light/brightness and payload b'80'

---

**Success** Value for ViDevCommon.brightness (ffw.livingroom.main\_light) is correct (Content 80 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffw.livingroom.main\_light)): 80 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.livingroom.main\_light)): result = 80  
↪ (<class 'int'>)

---

**Info** Setting state of Light.brightness (ffw.livingroom.main\_light) to 100

---

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/livingroom/main\_light/brightness and payload b'100'

---

**Success** Value for ViDevCommon.brightness (ffw.livingroom.main\_light) is correct (Content 100 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffw.livingroom.main\_light)): 100 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.livingroom.main\_light)): result = 100  
↪ (<class 'int'>)

**A.1.49 ViDevCommon.color\_temp (ffw.livingroom.main\_light) → Light.color\_temp (ffw.livingroom.main\_light)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 10

---

Sending message with topic videv/ffw/livingroom/main\_light/color\_temp/set and payload 10

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light/set and payload  
↪ b'{"color\_temp": 454}'

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/ffw/livingroom/main\_light/color\_temp and payload b'10'

---

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (10, 10) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (10, 10) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffw.livingroom.main\_light) to 0

---

Sending message with topic videv/ffw/livingroom/main\_light/color\_temp/set and payload 0

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light/set and payload  
↪ b'{"color\_temp": 250}'

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic videv/ffw/livingroom/main\_light/color\_temp and payload b'0'

---

**Success** Value for Light.color\_temp (ffw.livingroom.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffw.livingroom.main\_light)): 0 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffw.livingroom.main\_light)): result = 0 (<class  
↪ 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffw.livingroom.main\_light) to 2

---

Sending message with topic videv/ffw/livingroom/main\_light/color\_temp/set and payload 2

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light/set and payload  
↪ b'{"color\_temp": 291}'

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic videv/ffw/livingroom/main\_light/color\_temp and payload b'2'

---

**Success** Value for Light.color\_temp (ffw.livingroom.main\_light) is correct (Content 2 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffw.livingroom.main\_light)): 2 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffw.livingroom.main\_light)): result = 2 (<class  
↪ 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffw.livingroom.main\_light) to 4

---

Sending message with topic videv/ffw/livingroom/main\_light/color\_temp/set and payload 4

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light/set and payload  
↪ b'{"color\_temp": 332}'

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic videv/ffw/livingroom/main\_light/color\_temp and payload b'4'

---

**Success** Value for Light.color\_temp (ffw.livingroom.main\_light) is correct (Content 4 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffw.livingroom.main\_light)): 4 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffw.livingroom.main\_light)): result = 4 (<class  
↪ 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffw.livingroom.main\_light) to 6

---

Sending message with topic videv/ffw/livingroom/main\_light/color\_temp/set and payload 6

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light/set and payload  
↪ b'{"color\_temp": 372}'

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic videv/ffw/livingroom/main\_light/color\_temp and payload b'6'

---

**Success** Value for Light.color\_temp (ffw.livingroom.main\_light) is correct (Content 6 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffw.livingroom.main\_light)): 6 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffw.livingroom.main\_light)): result = 6 (<class  
↪ 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffw.livingroom.main\_light) to 8

---

Sending message with topic videv/ffw/livingroom/main\_light/color\_temp/set and payload 8

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light/set and payload

↪ b'{"color\_temp": 413}'

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",

↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",

↪ "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic videv/ffw/livingroom/main\_light/color\_temp and payload b'8'

---

**Success** Value for Light.color\_temp (ffw.livingroom.main\_light) is correct (Content 8 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffw.livingroom.main\_light)): 8 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffw.livingroom.main\_light)): result = 8 (<class

↪ 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffw.livingroom.main\_light) to 10

---

Sending message with topic videv/ffw/livingroom/main\_light/color\_temp/set and payload 10

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light/set and payload

↪ b'{"color\_temp": 454}'

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",

↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",

↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/ffw/livingroom/main\_light/color\_temp and payload b'10'

---

**Success** Value for Light.color\_temp (ffw.livingroom.main\_light) is correct (Content 10 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffw.livingroom.main\_light)): 10 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffw.livingroom.main\_light)): result = 10 (<class

↪ 'int'>)

**A.1.50** Light.color\_temp (ffw.livingroom.main\_light) → ViDevCommon.color\_temp (ffw.livingroom.main\_light)

#### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 10

---

Sending message with topic videv/ffw/livingroom/main\_light/color\_temp/set and payload 10

---

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (10, 10) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (10, 10) (<class 'tuple'>)

---

**Info** Setting state of Light.color\_temp (ffw.livingroom.main\_light) to 0

---

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic videv/ffw/livingroom/main\_light/color\_temp and payload b'0'

---

**Success** Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light)): 0 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light)): result = 0 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (ffw.livingroom.main\_light) to 2

---

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic videv/ffw/livingroom/main\_light/color\_temp and payload b'2'

---

**Success** Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light) is correct (Content 2 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light)): 2 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light)): result = 2 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (ffw.livingroom.main\_light) to 4

---

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic videv/ffw/livingroom/main\_light/color\_temp and payload b'4'

---

**Success** Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light) is correct (Content 4 and Type is <class 'int'>).

Result (Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light)): 4 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light)): result = 4 (<class 'int'>)

**Info** Setting state of Light.color\_temp (ffw.livingroom.main\_light) to 6

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic videv/ffw/livingroom/main\_light/color\_temp and payload b'6'

**Success** Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light) is correct (Content 6 and Type is <class 'int'>).

Result (Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light)): 6 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light)): result = 6 (<class 'int'>)

**Info** Setting state of Light.color\_temp (ffw.livingroom.main\_light) to 8

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic videv/ffw/livingroom/main\_light/color\_temp and payload b'8'

**Success** Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light) is correct (Content 8 and Type is <class 'int'>).

Result (Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light)): 8 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light)): result = 8 (<class 'int'>)

**Info** Setting state of Light.color\_temp (ffw.livingroom.main\_light) to 10

Sending message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_ffw/ffw/livingroom/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/ffw/livingroom/main\_light/color\_temp and payload b'10'

---

**Success** Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light) is correct (Content 10 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light)): 10 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffw.livingroom.main\_light)): result = 10  
 ↪ (<class 'int'>)

#### A.1.51 ViDevHeating.temp\_setp (ffw.livingroom.heating\_valve) → HeatingValve.temp\_setp (ffw.livingroom.heating\_valve)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state 30

---

Sending message with topic videv/ffw/livingroom/heating\_valve/user\_temperature\_setpoint/set  
 ↪ and payload 30

Sending message with topic zigbee\_ffw/ffw/livingroom/heating\_valve and payload  
 ↪ {"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}

Received message with topic zigbee\_ffw/ffw/livingroom/heating\_valve/set and payload  
 ↪ b'{"current\_heating\_setpoint": 30}'

Received message with topic videv/ffw/livingroom/heating\_valve/valve\_temperature\_setpoint and  
 ↪ payload b'30'

Received message with topic videv/ffw/livingroom/heating\_valve/user\_temperature\_setpoint and  
 ↪ payload b'30'

Received message with topic zigbee\_ffw/ffw/livingroom/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (30, 30) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (30, 30) (<class 'tuple'>)

---

**Info** Setting state of ViDevHeating.temp\_setp (ffw.livingroom.heating\_valve) to 15

---

Sending message with topic videv/ffw/livingroom/heating\_valve/user\_temperature\_setpoint/set  
 ↪ and payload 15

Received message with topic zigbee\_ffw/ffw/livingroom/heating\_valve/set and payload  
 ↪ b'{"current\_heating\_setpoint": 15}'

Sending message with topic zigbee\_ffw/ffw/livingroom/heating\_valve and payload  
 ↪ {"current\_heating\_setpoint": 15, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffw/livingroom/heating\_valve/valve\_temperature\_setpoint and  
 ↪ payload b'15'

Received message with topic videv/ffw/livingroom/heating\_valve/user\_temperature\_setpoint and  
 ↪ payload b'15'

Received message with topic zigbee\_ffw/ffw/livingroom/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 15, "local\_temperature": 20.7, "battery": 97}'

**Success** Value for HeatingValve.temp\_setp (ffw.livingroom.heating\_valve) is correct (Content 15 and Type is <class 'int'>).

Result (Value for HeatingValve.temp\_setp (ffw.livingroom.heating\_valve)): 15 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffw.livingroom.heating\_valve)): result = 15  
 ↪ (<class 'int'>)

**Info** Setting state of ViDevHeating.temp\_setp (ffw.livingroom.heating\_valve) to 20

Sending message with topic videv/ffw/livingroom/heating\_valve/user\_temperature\_setpoint/set  
 ↪ and payload 20

Received message with topic zigbee\_ffw/ffw/livingroom/heating\_valve/set and payload  
 ↪ b'{"current\_heating\_setpoint": 20}'

Sending message with topic zigbee\_ffw/ffw/livingroom/heating\_valve and payload  
 ↪ {"current\_heating\_setpoint": 20, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffw/livingroom/heating\_valve/valve\_temperature\_setpoint and  
 ↪ payload b'20'

Received message with topic videv/ffw/livingroom/heating\_valve/user\_temperature\_setpoint and  
 ↪ payload b'20'

Received message with topic zigbee\_ffw/ffw/livingroom/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 20, "local\_temperature": 20.7, "battery": 97}'

**Success** Value for HeatingValve.temp\_setp (ffw.livingroom.heating\_valve) is correct (Content 20 and Type is <class 'int'>).

Result (Value for HeatingValve.temp\_setp (ffw.livingroom.heating\_valve)): 20 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffw.livingroom.heating\_valve)): result = 20  
 ↪ (<class 'int'>)

**Info** Setting state of ViDevHeating.temp\_setp (ffw.livingroom.heating\_valve) to 25

Sending message with topic videv/ffw/livingroom/heating\_valve/user\_temperature\_setpoint/set  
 ↪ and payload 25

Received message with topic zigbee\_ffw/ffw/livingroom/heating\_valve/set and payload  
 ↪ b'{"current\_heating\_setpoint": 25}'

Sending message with topic zigbee\_ffw/ffw/livingroom/heating\_valve and payload  
 ↪ {"current\_heating\_setpoint": 25, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffw/livingroom/heating\_valve/valve\_temperature\_setpoint and  
 ↪ payload b'25'

Received message with topic videv/ffw/livingroom/heating\_valve/user\_temperature\_setpoint and  
 ↪ payload b'25'

Received message with topic zigbee\_ffw/ffw/livingroom/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 25, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Value for HeatingValve.temp\_setp (ffw.livingroom.heating\_valve) is correct (Content 25 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (ffw.livingroom.heating\_valve)): 25 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffw.livingroom.heating\_valve)): result = 25  
 ↪ (<class 'int'>)

---

**Info** Setting state of ViDevHeating.temp\_setp (ffw.livingroom.heating\_valve) to 30

---

Sending message with topic videv/ffw/livingroom/heating\_valve/user\_temperature\_setpoint/set  
 ↪ and payload 30

Received message with topic zigbee\_ffw/ffw/livingroom/heating\_valve/set and payload  
 ↪ b'{"current\_heating\_setpoint": 30}'

Sending message with topic zigbee\_ffw/ffw/livingroom/heating\_valve and payload  
 ↪ {"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffw/livingroom/heating\_valve/valve\_temperature\_setpoint and  
 ↪ payload b'30'

Received message with topic videv/ffw/livingroom/heating\_valve/user\_temperature\_setpoint and  
 ↪ payload b'30'

Received message with topic zigbee\_ffw/ffw/livingroom/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Value for HeatingValve.temp\_setp (ffw.livingroom.heating\_valve) is correct (Content 30 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (ffw.livingroom.heating\_valve)): 30 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffw.livingroom.heating\_valve)): result = 30  
 ↪ (<class 'int'>)

---

#### A.1.52 ViDevCommon.state (ffw.sleep.main\_light) → Shelly.relay/0 (ffw.sleep.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffw/sleep/main\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (ffw.sleep.main\_light) to True

---

Sending message with topic videv/ffw/sleep/main\_light/state/set and payload true

---

```
Received message with topic shellies/ffw/sleep/main_light/relay/0/command and payload b'on'
Sending message with topic shellies/ffw/sleep/main_light/relay/0 and payload on
Sending message with topic zigbee_ffw/ffw/sleep/main_light and payload {"state": "on",
↪  "brightness": 127.0}
Received message with topic shellies/ffw/sleep/main_light/relay/0 and payload b'on'
Received message with topic zigbee_ffw/ffw/sleep/main_light and payload b'{"state": "on",
↪  "brightness": 127.0}'
Received message with topic videv/ffw/sleep/main_light/state and payload b'true'
Received message with topic zigbee_ffw/ffw/sleep/window_light/set and payload b'{"state":
↪  "on"}'
Sending message with topic zigbee_ffw/ffw/sleep/window_light and payload {"state": "on",
↪  "brightness": 127.0, "color_temp": 352.0}
Received message with topic videv/ffw/sleep/main_light/brightness and payload b'50'
Received message with topic zigbee_ffw/ffw/sleep/window_light and payload b'{"state": "on",
↪  "brightness": 127.0, "color_temp": 352.0}'
```

---

**Success** Value for Shelly.relay/0 (ffw.sleep.main\_light) is correct (Content True and Type is <class 'bool'>).

---

```
Result (Value for Shelly.relay/0 (ffw.sleep.main_light)): True (<class 'bool'>)
Expectation (Value for Shelly.relay/0 (ffw.sleep.main_light)): result = True (<class 'bool'>)
```

---

**Info** Setting state of ViDevCommon.state (ffw.sleep.main\_light) to False

---

```
Sending message with topic videv/ffw/sleep/main_light/state/set and payload false
Received message with topic videv/ffw/sleep/window_light/state and payload b'true'
Received message with topic videv/ffw/sleep/window_light/brightness and payload b'50'
Received message with topic videv/ffw/sleep/window_light/color_temp and payload b'5'
Received message with topic shellies/ffw/sleep/main_light/relay/0/command and payload b'off'
Sending message with topic shellies/ffw/sleep/main_light/relay/0 and payload off
Received message with topic shellies/ffw/sleep/main_light/relay/0 and payload b'off'
```

---

**Success** Value for Shelly.relay/0 (ffw.sleep.main\_light) is correct (Content False and Type is <class 'bool'>).

---

```
Result (Value for Shelly.relay/0 (ffw.sleep.main_light)): False (<class 'bool'>)
Expectation (Value for Shelly.relay/0 (ffw.sleep.main_light)): result = False (<class 'bool'>)
```

#### A.1.53 Shelly.relay/0 (ffw.sleep.main\_light) → ViDevCommon.state (ffw.sleep.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

```
Sending message with topic videv/ffw/sleep/main_light/state/set and payload false
Received message with topic videv/ffw/sleep/main_light/state and payload b'false'
```

Received message with topic zigbee\_ffw/ffw/sleep/window\_light/set and payload b'{"state":  
↪ "off"}'

Sending message with topic zigbee\_ffw/ffw/sleep/window\_light and payload {"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/sleep/window\_light and payload b'{"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/sleep/window\_light/state and payload b'false'

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Shelly.relay/0 (ffw.sleep.main\_light) to True

---

Sending message with topic shellies/ffw/sleep/main\_light/relay/0 and payload on

Sending message with topic zigbee\_ffw/ffw/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 127.0}

Received message with topic shellies/ffw/sleep/main\_light/relay/0 and payload b'on'

Received message with topic zigbee\_ffw/ffw/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 127.0}'

Received message with topic videv/ffw/sleep/main\_light/state and payload b'true'

Received message with topic zigbee\_ffw/ffw/sleep/window\_light/set and payload b'{"state":  
↪ "on"}'

Sending message with topic zigbee\_ffw/ffw/sleep/window\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/sleep/window\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/sleep/window\_light/state and payload b'true'

---

**Success** Value for ViDevCommon.state (ffw.sleep.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffw.sleep.main\_light)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffw.sleep.main\_light)): result = True (<class  
↪ 'bool'>)

---

**Info** Setting state of Shelly.relay/0 (ffw.sleep.main\_light) to False

---

Sending message with topic shellies/ffw/sleep/main\_light/relay/0 and payload off

Received message with topic shellies/ffw/sleep/main\_light/relay/0 and payload b'off'

Received message with topic videv/ffw/sleep/main\_light/state and payload b'false'

Received message with topic zigbee\_ffw/ffw/sleep/window\_light/set and payload b'{"state":  
↪ "off"}'

Sending message with topic zigbee\_ffw/ffw/sleep/window\_light and payload {"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}

```
Received message with topic zigbee_ffw/ffw/sleep/window_light and payload b'{"state": "off",
↪ "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic videv/ffw/sleep/window_light/state and payload b'false'
```

---

**Success** Value for ViDevCommon.state (ffw.sleep.main\_light) is correct (Content False and Type is <class 'bool'>).

---

```
Result (Value for ViDevCommon.state (ffw.sleep.main_light)): False (<class 'bool'>)
```

```
Expectation (Value for ViDevCommon.state (ffw.sleep.main_light)): result = False (<class
↪ 'bool'>)
```

#### A.1.54 ViDevCommon.brightness (ffw.sleep.main\_light) → Light.brightness (ffw.sleep.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---

```
Sending message with topic shellies/ffw/sleep/main_light/relay/0 and payload on
```

```
Sending message with topic zigbee_ffw/ffw/sleep/main_light and payload {"state": "on",
↪ "brightness": 127.0}
```

```
Received message with topic shellies/ffw/sleep/main_light/relay/0 and payload b'on'
```

```
Received message with topic zigbee_ffw/ffw/sleep/main_light and payload b'{"state": "on",
↪ "brightness": 127.0}'
```

```
Received message with topic videv/ffw/sleep/main_light/state and payload b'true'
```

```
Received message with topic zigbee_ffw/ffw/sleep/window_light/set and payload b'{"state":
↪ "on"}'
```

```
Sending message with topic zigbee_ffw/ffw/sleep/window_light and payload {"state": "on",
↪ "brightness": 127.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_ffw/ffw/sleep/window_light and payload b'{"state": "on",
↪ "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic videv/ffw/sleep/window_light/state and payload b'true'
```

---

**Info** Prepare: Setting devices to last state 100

---

```
Sending message with topic videv/ffw/sleep/main_light/brightness/set and payload 100
```

```
Sending message with topic zigbee_ffw/ffw/sleep/main_light and payload {"state": "on",
↪ "brightness": 254.0}
```

```
Received message with topic zigbee_ffw/ffw/sleep/main_light/set and payload b'{"brightness":
↪ 254}'
```

```
Received message with topic zigbee_ffw/ffw/sleep/main_light and payload b'{"state": "on",
↪ "brightness": 254.0}'
```

```
Received message with topic videv/ffw/sleep/main_light/brightness and payload b'100'
```

---

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.brightness (ffw.sleep.main\_light) to 0

---

Sending message with topic videv/ffw/sleep/main\_light/brightness/set and payload 0

Received message with topic zigbee\_ffw/ffw/sleep/main\_light/set and payload b'{"brightness":  
↪ 1}'

Sending message with topic zigbee\_ffw/ffw/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 1.0}

Received message with topic zigbee\_ffw/ffw/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 1.0}'

Received message with topic videv/ffw/sleep/main\_light/brightness and payload b'0'

---

**Success** Value for Light.brightness (ffw.sleep.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffw.sleep.main\_light)): 0 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.sleep.main\_light)): result = 0 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffw.sleep.main\_light) to 20

---

Sending message with topic videv/ffw/sleep/main\_light/brightness/set and payload 20

Received message with topic zigbee\_ffw/ffw/sleep/main\_light/set and payload b'{"brightness":  
↪ 52}'

Sending message with topic zigbee\_ffw/ffw/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 52.0}

Received message with topic zigbee\_ffw/ffw/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 52.0}'

Received message with topic videv/ffw/sleep/main\_light/brightness and payload b'20'

---

**Success** Value for Light.brightness (ffw.sleep.main\_light) is correct (Content 20 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffw.sleep.main\_light)): 20 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.sleep.main\_light)): result = 20 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffw.sleep.main\_light) to 40

---

Sending message with topic videv/ffw/sleep/main\_light/brightness/set and payload 40

Received message with topic zigbee\_ffw/ffw/sleep/main\_light/set and payload b'{"brightness":  
↪ 102}'

Sending message with topic zigbee\_ffw/ffw/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 102.0}

---

Received message with topic zigbee\_ffw/ffw/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 102.0}'

Received message with topic videv/ffw/sleep/main\_light/brightness and payload b'40'

**Success** Value for Light.brightness (ffw.sleep.main\_light) is correct (Content 40 and Type is <class 'int'>).

Result (Value for Light.brightness (ffw.sleep.main\_light)): 40 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.sleep.main\_light)): result = 40 (<class 'int'>)

**Info** Setting state of ViDevCommon.brightness (ffw.sleep.main\_light) to 60

Sending message with topic videv/ffw/sleep/main\_light/brightness/set and payload 60

Received message with topic zigbee\_ffw/ffw/sleep/main\_light/set and payload b'{"brightness":  
↪ 153}'

Sending message with topic zigbee\_ffw/ffw/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 153.0}

Received message with topic zigbee\_ffw/ffw/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 153.0}'

Received message with topic videv/ffw/sleep/main\_light/brightness and payload b'60'

**Success** Value for Light.brightness (ffw.sleep.main\_light) is correct (Content 60 and Type is <class 'int'>).

Result (Value for Light.brightness (ffw.sleep.main\_light)): 60 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.sleep.main\_light)): result = 60 (<class 'int'>)

**Info** Setting state of ViDevCommon.brightness (ffw.sleep.main\_light) to 80

Sending message with topic videv/ffw/sleep/main\_light/brightness/set and payload 80

Received message with topic zigbee\_ffw/ffw/sleep/main\_light/set and payload b'{"brightness":  
↪ 203}'

Sending message with topic zigbee\_ffw/ffw/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 203.0}

Received message with topic zigbee\_ffw/ffw/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 203.0}'

Received message with topic videv/ffw/sleep/main\_light/brightness and payload b'80'

**Success** Value for Light.brightness (ffw.sleep.main\_light) is correct (Content 80 and Type is <class 'int'>).

Result (Value for Light.brightness (ffw.sleep.main\_light)): 80 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.sleep.main\_light)): result = 80 (<class 'int'>)

**Info** Setting state of ViDevCommon.brightness (ffw.sleep.main\_light) to 100

Sending message with topic videv/ffw/sleep/main\_light/brightness/set and payload 100

Received message with topic zigbee\_ffw/ffw/sleep/main\_light/set and payload b'{"brightness":  
↪ 254}'

Sending message with topic zigbee\_ffw/ffw/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 254.0}

Received message with topic zigbee\_ffw/ffw/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0}'

Received message with topic videv/ffw/sleep/main\_light/brightness and payload b'100'

---

**Success** Value for Light.brightness (ffw.sleep.main\_light) is correct (Content 100 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffw.sleep.main\_light)): 100 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.sleep.main\_light)): result = 100 (<class 'int'>)

---

#### A.1.55 Light.brightness (ffw.sleep.main\_light) → ViDevCommon.brightness (ffw.sleep.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 100

---

Sending message with topic videv/ffw/sleep/main\_light/brightness/set and payload 100

---

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

---

**Info** Setting state of Light.brightness (ffw.sleep.main\_light) to 0

---

Sending message with topic zigbee\_ffw/ffw/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 1.0}

Received message with topic zigbee\_ffw/ffw/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 1.0}'

Received message with topic videv/ffw/sleep/main\_light/brightness and payload b'0'

---

**Success** Value for ViDevCommon.brightness (ffw.sleep.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffw.sleep.main\_light)): 0 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.sleep.main\_light)): result = 0 (<class 'int'>)  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (ffw.sleep.main\_light) to 20

---

Sending message with topic zigbee\_ffw/ffw/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 52.0}

Received message with topic zigbee\_ffw/ffw/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 52.0}'

Received message with topic videv/ffw/sleep/main\_light/brightness and payload b'20'

---

**Success** Value for ViDevCommon.brightness (ffw.sleep.main\_light) is correct (Content 20 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffw.sleep.main\_light)): 20 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.sleep.main\_light)): result = 20 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (ffw.sleep.main\_light) to 40

---

Sending message with topic zigbee\_ffw/ffw/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 102.0}

Received message with topic zigbee\_ffw/ffw/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 102.0}'

Received message with topic videv/ffw/sleep/main\_light/brightness and payload b'40'

---

**Success** Value for ViDevCommon.brightness (ffw.sleep.main\_light) is correct (Content 40 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffw.sleep.main\_light)): 40 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.sleep.main\_light)): result = 40 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (ffw.sleep.main\_light) to 60

---

Sending message with topic zigbee\_ffw/ffw/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 153.0}

Received message with topic zigbee\_ffw/ffw/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 153.0}'

Received message with topic videv/ffw/sleep/main\_light/brightness and payload b'60'

---

**Success** Value for ViDevCommon.brightness (ffw.sleep.main\_light) is correct (Content 60 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffw.sleep.main\_light)): 60 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.sleep.main\_light)): result = 60 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (ffw.sleep.main\_light) to 80

---

Sending message with topic zigbee\_ffw/ffw/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 203.0}

Received message with topic zigbee\_ffw/ffw/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 203.0}'

Received message with topic videv/ffw/sleep/main\_light/brightness and payload b'80'

**Success** Value for ViDevCommon.brightness (ffw.sleep.main\_light) is correct (Content 80 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffw.sleep.main\_light)): 80 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.sleep.main\_light)): result = 80 (<class 'int'>)

**Info** Setting state of Light.brightness (ffw.sleep.main\_light) to 100

Sending message with topic zigbee\_ffw/ffw/sleep/main\_light and payload {"state": "on",  
↪ "brightness": 254.0}

Received message with topic zigbee\_ffw/ffw/sleep/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0}'

Received message with topic videv/ffw/sleep/main\_light/brightness and payload b'100'

**Success** Value for ViDevCommon.brightness (ffw.sleep.main\_light) is correct (Content 100 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (ffw.sleep.main\_light)): 100 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.sleep.main\_light)): result = 100 (<class 'int'>)

#### A.1.56 ViDevHeating.temp\_setp (ffw.sleep.heating\_valve) → HeatingValve.temp\_setp (ffw.sleep.heating\_valve)

##### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Setting devices to last state 30

Sending message with topic videv/ffw/sleep/heating\_valve/user\_temperature\_setpoint/set and  
↪ payload 30

Sending message with topic zigbee\_ffw/ffw/sleep/heating\_valve and payload  
↪ {"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}

Received message with topic zigbee\_ffw/ffw/sleep/heating\_valve/set and payload  
↪ b'{"current\_heating\_setpoint": 30}'

Received message with topic videv/ffw/sleep/heating\_valve/valve\_temperature\_setpoint and  
↪ payload b'30'

Received message with topic videv/ffw/sleep/heating\_valve/user\_temperature\_setpoint and  
↪ payload b'30'

Received message with topic zigbee\_ffw/ffw/sleep/heating\_valve and payload  
↪ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

**Success** Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (30, 30) (<class 'tuple'>)

```
Expectation (Start state (master, slave)): result = (30, 30) (<class 'tuple'>)
```

---

**Info** Setting state of ViDevHeating.temp\_setp (ffw.sleep.heating\_valve) to 15

---

```
Sending message with topic videv/ffw/sleep/heating_valve/user_temperature_setpoint/set and
↳ payload 15
```

```
Received message with topic zigbee_ffw/ffw/sleep/heating_valve/set and payload
↳ b'{"current_heating_setpoint": 15}'
```

```
Sending message with topic zigbee_ffw/ffw/sleep/heating_valve and payload
↳ {"current_heating_setpoint": 15, "local_temperature": 20.7, "battery": 97}
```

```
Received message with topic videv/ffw/sleep/heating_valve/valve_temperature_setpoint and
↳ payload b'15'
```

```
Received message with topic videv/ffw/sleep/heating_valve/user_temperature_setpoint and
↳ payload b'15'
```

```
Received message with topic zigbee_ffw/ffw/sleep/heating_valve and payload
↳ b'{"current_heating_setpoint": 15, "local_temperature": 20.7, "battery": 97}'
```

---

**Success** Value for HeatingValve.temp\_setp (ffw.sleep.heating\_valve) is correct (Content 15 and Type is <class 'int'>).

---

```
Result (Value for HeatingValve.temp_setp (ffw.sleep.heating_valve)): 15 (<class 'int'>)
```

```
Expectation (Value for HeatingValve.temp_setp (ffw.sleep.heating_valve)): result = 15 (<class
↳ 'int'>)
```

---

**Info** Setting state of ViDevHeating.temp\_setp (ffw.sleep.heating\_valve) to 20

---

```
Sending message with topic videv/ffw/sleep/heating_valve/user_temperature_setpoint/set and
↳ payload 20
```

```
Received message with topic zigbee_ffw/ffw/sleep/heating_valve/set and payload
↳ b'{"current_heating_setpoint": 20}'
```

```
Sending message with topic zigbee_ffw/ffw/sleep/heating_valve and payload
↳ {"current_heating_setpoint": 20, "local_temperature": 20.7, "battery": 97}
```

```
Received message with topic videv/ffw/sleep/heating_valve/valve_temperature_setpoint and
↳ payload b'20'
```

```
Received message with topic videv/ffw/sleep/heating_valve/user_temperature_setpoint and
↳ payload b'20'
```

```
Received message with topic zigbee_ffw/ffw/sleep/heating_valve and payload
↳ b'{"current_heating_setpoint": 20, "local_temperature": 20.7, "battery": 97}'
```

---

**Success** Value for HeatingValve.temp\_setp (ffw.sleep.heating\_valve) is correct (Content 20 and Type is <class 'int'>).

---

```
Result (Value for HeatingValve.temp_setp (ffw.sleep.heating_valve)): 20 (<class 'int'>)
```

```
Expectation (Value for HeatingValve.temp_setp (ffw.sleep.heating_valve)): result = 20 (<class
↳ 'int'>)
```

---

**Info**    Setting state of ViDevHeating.temp\_setp (ffw.sleep.heating\_valve) to 25

---

Sending message with topic videv/ffw/sleep/heating\_valve/user\_temperature\_setpoint/set and  
 ↪ payload 25

Received message with topic zigbee\_ffw/ffw/sleep/heating\_valve/set and payload  
 ↪ b'{"current\_heating\_setpoint": 25}'

Sending message with topic zigbee\_ffw/ffw/sleep/heating\_valve and payload  
 ↪ {"current\_heating\_setpoint": 25, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffw/sleep/heating\_valve/valve\_temperature\_setpoint and  
 ↪ payload b'25'

Received message with topic videv/ffw/sleep/heating\_valve/user\_temperature\_setpoint and  
 ↪ payload b'25'

Received message with topic zigbee\_ffw/ffw/sleep/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 25, "local\_temperature": 20.7, "battery": 97}'

---

**Success**    Value for HeatingValve.temp\_setp (ffw.sleep.heating\_valve) is correct (Content 25 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (ffw.sleep.heating\_valve)): 25 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffw.sleep.heating\_valve)): result = 25 (<class 'int'>)  
 ↪ 'int'>)

---

**Info**    Setting state of ViDevHeating.temp\_setp (ffw.sleep.heating\_valve) to 30

---

Sending message with topic videv/ffw/sleep/heating\_valve/user\_temperature\_setpoint/set and  
 ↪ payload 30

Received message with topic zigbee\_ffw/ffw/sleep/heating\_valve/set and payload  
 ↪ b'{"current\_heating\_setpoint": 30}'

Sending message with topic zigbee\_ffw/ffw/sleep/heating\_valve and payload  
 ↪ {"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffw/sleep/heating\_valve/valve\_temperature\_setpoint and  
 ↪ payload b'30'

Received message with topic videv/ffw/sleep/heating\_valve/user\_temperature\_setpoint and  
 ↪ payload b'30'

Received message with topic zigbee\_ffw/ffw/sleep/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

---

**Success**    Value for HeatingValve.temp\_setp (ffw.sleep.heating\_valve) is correct (Content 30 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (ffw.sleep.heating\_valve)): 30 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffw.sleep.heating\_valve)): result = 30 (<class 'int'>)  
 ↪ 'int'>)

### A.1.57 ViDevCommon.state (ffw.julian.main\_light) → Shelly.relay/0 (ffw.julian.main\_light)

#### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffw/julian/main\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (ffw.julian.main\_light) to True

---

Sending message with topic videv/ffw/julian/main\_light/state/set and payload true

Received message with topic shellies/ffw/julian/main\_light/relay/0/command and payload b'on'

Sending message with topic shellies/ffw/julian/main\_light/relay/0 and payload on

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic shellies/ffw/julian/main\_light/relay/0 and payload b'on'

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/julian/main\_light/state and payload b'true'

Received message with topic videv/ffw/julian/main\_light/brightness and payload b'50'

Received message with topic videv/ffw/julian/main\_light/color\_temp and payload b'5'

---

**Success** Value for Shelly.relay/0 (ffw.julian.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (ffw.julian.main\_light)): True (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (ffw.julian.main\_light)): result = True (<class 'bool'>)

---

**Info** Setting state of ViDevCommon.state (ffw.julian.main\_light) to False

---

Sending message with topic videv/ffw/julian/main\_light/state/set and payload false

Received message with topic shellies/ffw/julian/main\_light/relay/0/command and payload b'off'

Sending message with topic shellies/ffw/julian/main\_light/relay/0 and payload off

Received message with topic shellies/ffw/julian/main\_light/relay/0 and payload b'off'

Received message with topic videv/ffw/julian/main\_light/state and payload b'false'

---

**Success** Value for Shelly.relay/0 (ffw.julian.main\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (ffw.julian.main\_light)): False (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (ffw.julian.main\_light)): result = False (<class  
↪ 'bool'>)

---

**A.1.58 Shelly.relay/0 (ffw.julian.main\_light) → ViDevCommon.state (ffw.julian.main\_light)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffw/julian/main\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Shelly.relay/0 (ffw.julian.main\_light) to True

---

Sending message with topic shellies/ffw/julian/main\_light/relay/0 and payload on

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic shellies/ffw/julian/main\_light/relay/0 and payload b'on'

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/julian/main\_light/state and payload b'true'

---

**Success** Value for ViDevCommon.state (ffw.julian.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffw.julian.main\_light)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffw.julian.main\_light)): result = True (<class  
↪ 'bool'>)

---

**Info** Setting state of Shelly.relay/0 (ffw.julian.main\_light) to False

---

Sending message with topic shellies/ffw/julian/main\_light/relay/0 and payload off

Received message with topic shellies/ffw/julian/main\_light/relay/0 and payload b'off'

Received message with topic videv/ffw/julian/main\_light/state and payload b'false'

---

**Success** Value for ViDevCommon.state (ffw.julian.main\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffw.julian.main\_light)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffw.julian.main\_light)): result = False (<class  
↪ 'bool'>)

---

**A.1.59 ViDevCommon.brightness (ffw.julian.main\_light) → Light.brightness (ffw.julian.main\_light)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---

Sending message with topic shellies/ffw/julian/main\_light/relay/0 and payload on

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic shellies/ffw/julian/main\_light/relay/0 and payload b'on'

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/julian/main\_light/state and payload b'true'

---

**Info** Prepare: Setting devices to last state 100

---

Sending message with topic videv/ffw/julian/main\_light/brightness/set and payload 100

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light/set and payload b'{"brightness":  
↪ 254}'

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/julian/main\_light/brightness and payload b'100'

---

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.brightness (ffw.julian.main\_light) to 0

---

Sending message with topic videv/ffw/julian/main\_light/brightness/set and payload 0

Received message with topic zigbee\_ffw/ffw/julian/main\_light/set and payload b'{"brightness":  
↪ 1}'

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/julian/main\_light/brightness and payload b'0'

---

**Success** Value for Light.brightness (ffw.julian.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffw.julian.main\_light)): 0 (<class 'int'>)

---

Expectation (Value for Light.brightness (ffw.julian.main\_light)): result = 0 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffw.julian.main\_light) to 20

---

Sending message with topic videv/ffw/julian/main\_light/brightness/set and payload 20

Received message with topic zigbee\_ffw/ffw/julian/main\_light/set and payload b'{"brightness":  
↪ 52}'

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/julian/main\_light/brightness and payload b'20'

---

**Success** Value for Light.brightness (ffw.julian.main\_light) is correct (Content 20 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffw.julian.main\_light)): 20 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.julian.main\_light)): result = 20 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffw.julian.main\_light) to 40

---

Sending message with topic videv/ffw/julian/main\_light/brightness/set and payload 40

Received message with topic zigbee\_ffw/ffw/julian/main\_light/set and payload b'{"brightness":  
↪ 102}'

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/julian/main\_light/brightness and payload b'40'

---

**Success** Value for Light.brightness (ffw.julian.main\_light) is correct (Content 40 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffw.julian.main\_light)): 40 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.julian.main\_light)): result = 40 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffw.julian.main\_light) to 60

---

Sending message with topic videv/ffw/julian/main\_light/brightness/set and payload 60

Received message with topic zigbee\_ffw/ffw/julian/main\_light/set and payload b'{"brightness":  
↪ 153}'

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/julian/main\_light/brightness and payload b'60'

---

**Success** Value for Light.brightness (ffw.julian.main\_light) is correct (Content 60 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffw.julian.main\_light)): 60 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.julian.main\_light)): result = 60 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffw.julian.main\_light) to 80

---

Sending message with topic videv/ffw/julian/main\_light/brightness/set and payload 80

Received message with topic zigbee\_ffw/ffw/julian/main\_light/set and payload b'{"brightness":  
↪ 203}'

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/julian/main\_light/brightness and payload b'80'

---

**Success** Value for Light.brightness (ffw.julian.main\_light) is correct (Content 80 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffw.julian.main\_light)): 80 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.julian.main\_light)): result = 80 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (ffw.julian.main\_light) to 100

---

Sending message with topic videv/ffw/julian/main\_light/brightness/set and payload 100

Received message with topic zigbee\_ffw/ffw/julian/main\_light/set and payload b'{"brightness":  
↪ 254}'

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/julian/main\_light/brightness and payload b'100'

---

**Success** Value for Light.brightness (ffw.julian.main\_light) is correct (Content 100 and Type is <class 'int'>).

---

Result (Value for Light.brightness (ffw.julian.main\_light)): 100 (<class 'int'>)

Expectation (Value for Light.brightness (ffw.julian.main\_light)): result = 100 (<class 'int'>)

---

#### A.1.60 Light.brightness (ffw.julian.main\_light) → ViDevCommon.brightness (ffw.julian.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 100

---

Sending message with topic videv/ffw/julian/main\_light/brightness/set and payload 100

---

---

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

---

**Info** Setting state of Light.brightness (ffw.julian.main\_light) to 0

---

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/julian/main\_light/brightness and payload b'0'

---

**Success** Value for ViDevCommon.brightness (ffw.julian.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffw.julian.main\_light)): 0 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.julian.main\_light)): result = 0 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (ffw.julian.main\_light) to 20

---

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/julian/main\_light/brightness and payload b'20'

---

**Success** Value for ViDevCommon.brightness (ffw.julian.main\_light) is correct (Content 20 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffw.julian.main\_light)): 20 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.julian.main\_light)): result = 20 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (ffw.julian.main\_light) to 40

---

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/julian/main\_light/brightness and payload b'40'

---

---

**Success** Value for ViDevCommon.brightness (ffw.julian.main\_light) is correct (Content 40 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffw.julian.main\_light)): 40 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.julian.main\_light)): result = 40 (<class 'int'>)  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (ffw.julian.main\_light) to 60

---

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/julian/main\_light/brightness and payload b'60'

---

**Success** Value for ViDevCommon.brightness (ffw.julian.main\_light) is correct (Content 60 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffw.julian.main\_light)): 60 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.julian.main\_light)): result = 60 (<class 'int'>)  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (ffw.julian.main\_light) to 80

---

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/julian/main\_light/brightness and payload b'80'

---

**Success** Value for ViDevCommon.brightness (ffw.julian.main\_light) is correct (Content 80 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffw.julian.main\_light)): 80 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.julian.main\_light)): result = 80 (<class 'int'>)  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (ffw.julian.main\_light) to 100

---

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/ffw/julian/main\_light/brightness and payload b'100'

---

**Success** Value for ViDevCommon.brightness (ffw.julian.main\_light) is correct (Content 100 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (ffw.julian.main\_light)): 100 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (ffw.julian.main\_light)): result = 100 (<class 'int'>)

---

#### A.1.61 ViDevCommon.color\_temp (ffw.julian.main\_light) → Light.color\_temp (ffw.julian.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 10

---

Sending message with topic videv/ffw/julian/main\_light/color\_temp/set and payload 10

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light/set and payload b'{"color\_temp":  
↪ 454}'

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/ffw/julian/main\_light/color\_temp and payload b'10'

---

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (10, 10) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (10, 10) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.color\_temp (ffw.julian.main\_light) to 0

---

Sending message with topic videv/ffw/julian/main\_light/color\_temp/set and payload 0

Received message with topic zigbee\_ffw/ffw/julian/main\_light/set and payload b'{"color\_temp":  
↪ 250}'

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic videv/ffw/julian/main\_light/color\_temp and payload b'0'

---

**Success** Value for Light.color\_temp (ffw.julian.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (ffw.julian.main\_light)): 0 (<class 'int'>)

---

```
Expectation (Value for Light.color_temp (ffw.julian.main_light)): result = 0 (<class 'int'>)
```

---

**Info** Setting state of ViDevCommon.color\_temp (ffw.julian.main\_light) to 2

---

```
Sending message with topic videv/ffw/julian/main_light/color_temp/set and payload 2
```

```
Received message with topic zigbee_ffw/ffw/julian/main_light/set and payload b'{"color_temp":  
↪ 291}'
```

```
Sending message with topic zigbee_ffw/ffw/julian/main_light and payload {"state": "on",  
↪ "brightness": 254.0, "color_temp": 291.0}
```

```
Received message with topic zigbee_ffw/ffw/julian/main_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color_temp": 291.0}'
```

```
Received message with topic videv/ffw/julian/main_light/color_temp and payload b'2'
```

---

**Success** Value for Light.color\_temp (ffw.julian.main\_light) is correct (Content 2 and Type is <class 'int'>).

---

```
Result (Value for Light.color_temp (ffw.julian.main_light)): 2 (<class 'int'>)
```

```
Expectation (Value for Light.color_temp (ffw.julian.main_light)): result = 2 (<class 'int'>)
```

---

**Info** Setting state of ViDevCommon.color\_temp (ffw.julian.main\_light) to 4

---

```
Sending message with topic videv/ffw/julian/main_light/color_temp/set and payload 4
```

```
Received message with topic zigbee_ffw/ffw/julian/main_light/set and payload b'{"color_temp":  
↪ 332}'
```

```
Sending message with topic zigbee_ffw/ffw/julian/main_light and payload {"state": "on",  
↪ "brightness": 254.0, "color_temp": 332.0}
```

```
Received message with topic zigbee_ffw/ffw/julian/main_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color_temp": 332.0}'
```

```
Received message with topic videv/ffw/julian/main_light/color_temp and payload b'4'
```

---

**Success** Value for Light.color\_temp (ffw.julian.main\_light) is correct (Content 4 and Type is <class 'int'>).

---

```
Result (Value for Light.color_temp (ffw.julian.main_light)): 4 (<class 'int'>)
```

```
Expectation (Value for Light.color_temp (ffw.julian.main_light)): result = 4 (<class 'int'>)
```

---

**Info** Setting state of ViDevCommon.color\_temp (ffw.julian.main\_light) to 6

---

```
Sending message with topic videv/ffw/julian/main_light/color_temp/set and payload 6
```

```
Received message with topic zigbee_ffw/ffw/julian/main_light/set and payload b'{"color_temp":  
↪ 372}'
```

```
Sending message with topic zigbee_ffw/ffw/julian/main_light and payload {"state": "on",  
↪ "brightness": 254.0, "color_temp": 372.0}
```

```
Received message with topic zigbee_ffw/ffw/julian/main_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color_temp": 372.0}'
```

```
Received message with topic videv/ffw/julian/main_light/color_temp and payload b'6'
```

**Success** Value for Light.color\_temp (ffw.julian.main\_light) is correct (Content 6 and Type is <class 'int'>).

Result (Value for Light.color\_temp (ffw.julian.main\_light)): 6 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffw.julian.main\_light)): result = 6 (<class 'int'>)

**Info** Setting state of ViDevCommon.color\_temp (ffw.julian.main\_light) to 8

Sending message with topic videv/ffw/julian/main\_light/color\_temp/set and payload 8

Received message with topic zigbee\_ffw/ffw/julian/main\_light/set and payload b'{"color\_temp": 413}'

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on", "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on", "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic videv/ffw/julian/main\_light/color\_temp and payload b'8'

**Success** Value for Light.color\_temp (ffw.julian.main\_light) is correct (Content 8 and Type is <class 'int'>).

Result (Value for Light.color\_temp (ffw.julian.main\_light)): 8 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffw.julian.main\_light)): result = 8 (<class 'int'>)

**Info** Setting state of ViDevCommon.color\_temp (ffw.julian.main\_light) to 10

Sending message with topic videv/ffw/julian/main\_light/color\_temp/set and payload 10

Received message with topic zigbee\_ffw/ffw/julian/main\_light/set and payload b'{"color\_temp": 454}'

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on", "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on", "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/ffw/julian/main\_light/color\_temp and payload b'10'

**Success** Value for Light.color\_temp (ffw.julian.main\_light) is correct (Content 10 and Type is <class 'int'>).

Result (Value for Light.color\_temp (ffw.julian.main\_light)): 10 (<class 'int'>)

Expectation (Value for Light.color\_temp (ffw.julian.main\_light)): result = 10 (<class 'int'>)

#### A.1.62 Light.color\_temp (ffw.julian.main\_light) → ViDevCommon.color\_temp (ffw.julian.main\_light)

##### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Switching on device

**Info** Prepare: Setting devices to last state 10

Sending message with topic videv/ffw/julian/main\_light/color\_temp/set and payload 10

---

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (10, 10) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (10, 10) (<class 'tuple'>)

---

**Info** Setting state of Light.color\_temp (ffw.julian.main\_light) to 0

---

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic videv/ffw/julian/main\_light/color\_temp and payload b'0'

---

**Success** Value for ViDevCommon.color\_temp (ffw.julian.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffw.julian.main\_light)): 0 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffw.julian.main\_light)): result = 0 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (ffw.julian.main\_light) to 2

---

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic videv/ffw/julian/main\_light/color\_temp and payload b'2'

---

**Success** Value for ViDevCommon.color\_temp (ffw.julian.main\_light) is correct (Content 2 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffw.julian.main\_light)): 2 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffw.julian.main\_light)): result = 2 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (ffw.julian.main\_light) to 4

---

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic videv/ffw/julian/main\_light/color\_temp and payload b'4'

---

---

**Success** Value for ViDevCommon.color\_temp (ffw.julian.main\_light) is correct (Content 4 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffw.julian.main\_light)): 4 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffw.julian.main\_light)): result = 4 (<class 'int'>)

---

**Info** Setting state of Light.color\_temp (ffw.julian.main\_light) to 6

---

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic videv/ffw/julian/main\_light/color\_temp and payload b'6'

---

**Success** Value for ViDevCommon.color\_temp (ffw.julian.main\_light) is correct (Content 6 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffw.julian.main\_light)): 6 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffw.julian.main\_light)): result = 6 (<class 'int'>)

---

**Info** Setting state of Light.color\_temp (ffw.julian.main\_light) to 8

---

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic videv/ffw/julian/main\_light/color\_temp and payload b'8'

---

**Success** Value for ViDevCommon.color\_temp (ffw.julian.main\_light) is correct (Content 8 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffw.julian.main\_light)): 8 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffw.julian.main\_light)): result = 8 (<class 'int'>)

---

**Info** Setting state of Light.color\_temp (ffw.julian.main\_light) to 10

---

Sending message with topic zigbee\_ffw/ffw/julian/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_ffw/ffw/julian/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/ffw/julian/main\_light/color\_temp and payload b'10'

---

---

**Success** Value for ViDevCommon.color\_temp (ffw.julian.main\_light) is correct (Content 10 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (ffw.julian.main\_light)): 10 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (ffw.julian.main\_light)): result = 10 (<class 'int'>)

---

#### A.1.63 ViDevHeating.temp\_setp (ffw.julian.heating\_valve) → HeatingValve.temp\_setp (ffw.julian.heating\_valve)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state 30

---

Sending message with topic videv/ffw/julian/heating\_valve/user\_temperature\_setpoint/set and  
↪ payload 30

Sending message with topic zigbee\_ffw/ffw/julian/heating\_valve and payload  
↪ {"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}

Received message with topic zigbee\_ffw/ffw/julian/heating\_valve and payload  
↪ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

Received message with topic zigbee\_ffw/ffw/julian/heating\_valve/set and payload  
↪ b'{"current\_heating\_setpoint": 30}'

Received message with topic videv/ffw/julian/heating\_valve/valve\_temperature\_setpoint and  
↪ payload b'30'

Received message with topic videv/ffw/julian/heating\_valve/user\_temperature\_setpoint and  
↪ payload b'30'

---

**Success** Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (30, 30) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (30, 30) (<class 'tuple'>)

---

**Info** Setting state of ViDevHeating.temp\_setp (ffw.julian.heating\_valve) to 15

---

Sending message with topic videv/ffw/julian/heating\_valve/user\_temperature\_setpoint/set and  
↪ payload 15

Received message with topic zigbee\_ffw/ffw/julian/heating\_valve/set and payload  
↪ b'{"current\_heating\_setpoint": 15}'

Sending message with topic zigbee\_ffw/ffw/julian/heating\_valve and payload  
↪ {"current\_heating\_setpoint": 15, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffw/julian/heating\_valve/valve\_temperature\_setpoint and  
↪ payload b'15'

Received message with topic videv/ffw/julian/heating\_valve/user\_temperature\_setpoint and  
↪ payload b'15'

---

Received message with topic zigbee\_ffw/ffw/julian/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 15, "local\_temperature": 20.7, "battery": 97}'

**Success** Value for HeatingValve.temp\_setp (ffw.julian.heating\_valve) is correct (Content 15 and Type is <class 'int'>).

Result (Value for HeatingValve.temp\_setp (ffw.julian.heating\_valve)): 15 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffw.julian.heating\_valve)): result = 15 (<class 'int'>)

**Info** Setting state of ViDevHeating.temp\_setp (ffw.julian.heating\_valve) to 20

Sending message with topic videv/ffw/julian/heating\_valve/user\_temperature\_setpoint/set and  
 ↪ payload 20

Received message with topic zigbee\_ffw/ffw/julian/heating\_valve/set and payload  
 ↪ b'{"current\_heating\_setpoint": 20}'

Sending message with topic zigbee\_ffw/ffw/julian/heating\_valve and payload  
 ↪ {"current\_heating\_setpoint": 20, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffw/julian/heating\_valve/valve\_temperature\_setpoint and  
 ↪ payload b'20'

Received message with topic videv/ffw/julian/heating\_valve/user\_temperature\_setpoint and  
 ↪ payload b'20'

Received message with topic zigbee\_ffw/ffw/julian/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 20, "local\_temperature": 20.7, "battery": 97}'

**Success** Value for HeatingValve.temp\_setp (ffw.julian.heating\_valve) is correct (Content 20 and Type is <class 'int'>).

Result (Value for HeatingValve.temp\_setp (ffw.julian.heating\_valve)): 20 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffw.julian.heating\_valve)): result = 20 (<class 'int'>)

**Info** Setting state of ViDevHeating.temp\_setp (ffw.julian.heating\_valve) to 25

Sending message with topic videv/ffw/julian/heating\_valve/user\_temperature\_setpoint/set and  
 ↪ payload 25

Received message with topic zigbee\_ffw/ffw/julian/heating\_valve/set and payload  
 ↪ b'{"current\_heating\_setpoint": 25}'

Sending message with topic zigbee\_ffw/ffw/julian/heating\_valve and payload  
 ↪ {"current\_heating\_setpoint": 25, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffw/julian/heating\_valve/valve\_temperature\_setpoint and  
 ↪ payload b'25'

Received message with topic videv/ffw/julian/heating\_valve/user\_temperature\_setpoint and  
 ↪ payload b'25'

Received message with topic zigbee\_ffw/ffw/julian/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 25, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Value for HeatingValve.temp\_setp (ffw.julian.heating\_valve) is correct (Content 25 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (ffw.julian.heating\_valve)): 25 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffw.julian.heating\_valve)): result = 25 (<class 'int'>)  
 ↪ 'int'>)

---

**Info** Setting state of ViDevHeating.temp\_setp (ffw.julian.heating\_valve) to 30

---

Sending message with topic videv/ffw/julian/heating\_valve/user\_temperature\_setpoint/set and  
 ↪ payload 30

Received message with topic zigbee\_ffw/ffw/julian/heating\_valve/set and payload  
 ↪ b'{"current\_heating\_setpoint": 30}'

Sending message with topic zigbee\_ffw/ffw/julian/heating\_valve and payload  
 ↪ {"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffw/julian/heating\_valve/valve\_temperature\_setpoint and  
 ↪ payload b'30'

Received message with topic videv/ffw/julian/heating\_valve/user\_temperature\_setpoint and  
 ↪ payload b'30'

Received message with topic zigbee\_ffw/ffw/julian/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Value for HeatingValve.temp\_setp (ffw.julian.heating\_valve) is correct (Content 30 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (ffw.julian.heating\_valve)): 30 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffw.julian.heating\_valve)): result = 30 (<class 'int'>)  
 ↪ 'int'>)

---

#### A.1.64 ViDevCommon.state (ffw.bath.main\_light) → Shelly.relay/0 (ffw.bath.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffw/bath/main\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (ffw.bath.main\_light) to True

---

Sending message with topic videv/ffw/bath/main\_light/state/set and payload true

---

```
Received message with topic shellies/ffw/bath/main_light/relay/0/command and payload b'on'
Sending message with topic shellies/ffw/bath/main_light/relay/0 and payload on
Received message with topic shellies/ffw/bath/main_light/relay/0 and payload b'on'
Received message with topic videv/ffw/bath/main_light/state and payload b'true'
```

---

**Success** Value for Shelly.relay/0 (ffw.bath.main\_light) is correct (Content True and Type is <class 'bool'>).

---

```
Result (Value for Shelly.relay/0 (ffw.bath.main_light)): True (<class 'bool'>)
Expectation (Value for Shelly.relay/0 (ffw.bath.main_light)): result = True (<class 'bool'>)
```

---

**Info** Setting state of ViDevCommon.state (ffw.bath.main\_light) to False

---

```
Sending message with topic videv/ffw/bath/main_light/state/set and payload false
Received message with topic shellies/ffw/bath/main_light/relay/0/command and payload b'off'
Sending message with topic shellies/ffw/bath/main_light/relay/0 and payload off
Received message with topic shellies/ffw/bath/main_light/relay/0 and payload b'off'
Received message with topic videv/ffw/bath/main_light/state and payload b'false'
```

---

**Success** Value for Shelly.relay/0 (ffw.bath.main\_light) is correct (Content False and Type is <class 'bool'>).

---

```
Result (Value for Shelly.relay/0 (ffw.bath.main_light)): False (<class 'bool'>)
Expectation (Value for Shelly.relay/0 (ffw.bath.main_light)): result = False (<class 'bool'>)
```

---

#### A.1.65 Shelly.relay/0 (ffw.bath.main\_light) → ViDevCommon.state (ffw.bath.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

```
Sending message with topic videv/ffw/bath/main_light/state/set and payload false
```

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

```
Result (Start state (master, slave)): (False, False) (<class 'tuple'>)
Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)
```

---

**Info** Setting state of Shelly.relay/0 (ffw.bath.main\_light) to True

---

```
Sending message with topic shellies/ffw/bath/main_light/relay/0 and payload on
Received message with topic shellies/ffw/bath/main_light/relay/0 and payload b'on'
Received message with topic videv/ffw/bath/main_light/state and payload b'true'
```

---

**Success** Value for ViDevCommon.state (ffw.bath.main\_light) is correct (Content True and Type is <class 'bool'>).

---

```
Result (Value for ViDevCommon.state (ffw.bath.main_light)): True (<class 'bool'>)
```

---

```
Expectation (Value for ViDevCommon.state (ffw.bath.main_light)): result = True (<class
↪ 'bool'>)
```

---

**Info** Setting state of Shelly.relay/0 (ffw.bath.main\_light) to False

---

```
Sending message with topic shellies/ffw/bath/main_light/relay/0 and payload off
```

```
Received message with topic shellies/ffw/bath/main_light/relay/0 and payload b'off'
```

```
Received message with topic videv/ffw/bath/main_light/state and payload b'false'
```

---

**Success** Value for ViDevCommon.state (ffw.bath.main\_light) is correct (Content False and Type is <class 'bool'>).

---

```
Result (Value for ViDevCommon.state (ffw.bath.main_light)): False (<class 'bool'>)
```

```
Expectation (Value for ViDevCommon.state (ffw.bath.main_light)): result = False (<class
↪ 'bool'>)
```

#### A.1.66 ViDevHeating.temp\_setp (ffw.bath.heating\_valve) → HeatingValve.temp\_setp (ffw.bath.heating\_valve)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state 30

---

```
Sending message with topic videv/ffw/bath/heating_valve/user_temperature_setpoint/set and
↪ payload 30
```

```
Sending message with topic zigbee_ffw/ffw/bath/heating_valve and payload
↪ {"current_heating_setpoint": 30, "local_temperature": 20.7, "battery": 97}
```

```
Received message with topic zigbee_ffw/ffw/bath/heating_valve/set and payload
↪ b'{"current_heating_setpoint": 30}'
```

```
Received message with topic videv/ffw/bath/heating_valve/valve_temperature_setpoint and
↪ payload b'30'
```

```
Received message with topic videv/ffw/bath/heating_valve/user_temperature_setpoint and payload
↪ b'30'
```

```
Received message with topic zigbee_ffw/ffw/bath/heating_valve and payload
↪ b'{"current_heating_setpoint": 30, "local_temperature": 20.7, "battery": 97}'
```

---

**Success** Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).

---

```
Result (Start state (master, slave)): (30, 30) (<class 'tuple'>)
```

```
Expectation (Start state (master, slave)): result = (30, 30) (<class 'tuple'>)
```

---

**Info** Setting state of ViDevHeating.temp\_setp (ffw.bath.heating\_valve) to 15

---

```
Sending message with topic videv/ffw/bath/heating_valve/user_temperature_setpoint/set and
↪ payload 15
```

Received message with topic zigbee\_ffw/ffw/bath/heating\_valve/set and payload

↪ b'{"current\_heating\_setpoint": 15}'

Sending message with topic zigbee\_ffw/ffw/bath/heating\_valve and payload

↪ {"current\_heating\_setpoint": 15, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffw/bath/heating\_valve/valve\_temperature\_setpoint and

↪ payload b'15'

Received message with topic videv/ffw/bath/heating\_valve/user\_temperature\_setpoint and payload

↪ b'15'

Received message with topic zigbee\_ffw/ffw/bath/heating\_valve and payload

↪ b'{"current\_heating\_setpoint": 15, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Value for HeatingValve.temp\_setp (ffw.bath.heating\_valve) is correct (Content 15 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (ffw.bath.heating\_valve)): 15 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffw.bath.heating\_valve)): result = 15 (<class 'int'>)

---

**Info** Setting state of ViDevHeating.temp\_setp (ffw.bath.heating\_valve) to 20

---

Sending message with topic videv/ffw/bath/heating\_valve/user\_temperature\_setpoint/set and

↪ payload 20

Received message with topic zigbee\_ffw/ffw/bath/heating\_valve/set and payload

↪ b'{"current\_heating\_setpoint": 20}'

Sending message with topic zigbee\_ffw/ffw/bath/heating\_valve and payload

↪ {"current\_heating\_setpoint": 20, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffw/bath/heating\_valve/valve\_temperature\_setpoint and

↪ payload b'20'

Received message with topic videv/ffw/bath/heating\_valve/user\_temperature\_setpoint and payload

↪ b'20'

Received message with topic zigbee\_ffw/ffw/bath/heating\_valve and payload

↪ b'{"current\_heating\_setpoint": 20, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Value for HeatingValve.temp\_setp (ffw.bath.heating\_valve) is correct (Content 20 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (ffw.bath.heating\_valve)): 20 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffw.bath.heating\_valve)): result = 20 (<class 'int'>)

---

**Info** Setting state of ViDevHeating.temp\_setp (ffw.bath.heating\_valve) to 25

---

Sending message with topic videv/ffw/bath/heating\_valve/user\_temperature\_setpoint/set and

↪ payload 25

Received message with topic zigbee\_ffw/ffw/bath/heating\_valve/set and payload

↪ b'{"current\_heating\_setpoint": 25}'

Sending message with topic zigbee\_ffw/ffw/bath/heating\_valve and payload

↪ {"current\_heating\_setpoint": 25, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffw/bath/heating\_valve/valve\_temperature\_setpoint and

↪ payload b'25'

Received message with topic videv/ffw/bath/heating\_valve/user\_temperature\_setpoint and payload

↪ b'25'

Received message with topic zigbee\_ffw/ffw/bath/heating\_valve and payload

↪ b'{"current\_heating\_setpoint": 25, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Value for HeatingValve.temp\_setp (ffw.bath.heating\_valve) is correct (Content 25 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (ffw.bath.heating\_valve)): 25 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffw.bath.heating\_valve)): result = 25 (<class 'int'>)

---

**Info** Setting state of ViDevHeating.temp\_setp (ffw.bath.heating\_valve) to 30

---

Sending message with topic videv/ffw/bath/heating\_valve/user\_temperature\_setpoint/set and

↪ payload 30

Received message with topic zigbee\_ffw/ffw/bath/heating\_valve/set and payload

↪ b'{"current\_heating\_setpoint": 30}'

Sending message with topic zigbee\_ffw/ffw/bath/heating\_valve and payload

↪ {"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/ffw/bath/heating\_valve/valve\_temperature\_setpoint and

↪ payload b'30'

Received message with topic videv/ffw/bath/heating\_valve/user\_temperature\_setpoint and payload

↪ b'30'

Received message with topic zigbee\_ffw/ffw/bath/heating\_valve and payload

↪ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Value for HeatingValve.temp\_setp (ffw.bath.heating\_valve) is correct (Content 30 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (ffw.bath.heating\_valve)): 30 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (ffw.bath.heating\_valve)): result = 30 (<class 'int'>)

---

#### A.1.67 ViDevCommon.state (ffw.floor.main\_light) → Shelly.relay/0 (ffw.floor.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffw/floor/main\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (ffw.floor.main\_light) to True

---

Sending message with topic videv/ffw/floor/main\_light/state/set and payload true

Received message with topic shellies/ffw/floor/main\_light/relay/0/command and payload b'on'

Sending message with topic shellies/ffw/floor/main\_light/relay/0 and payload on

Received message with topic shellies/ffw/floor/main\_light/relay/0 and payload b'on'

Received message with topic videv/ffw/floor/main\_light/state and payload b'true'

---

**Success** Value for Shelly.relay/0 (ffw.floor.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (ffw.floor.main\_light)): True (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (ffw.floor.main\_light)): result = True (<class 'bool'>)

---

**Info** Setting state of ViDevCommon.state (ffw.floor.main\_light) to False

---

Sending message with topic videv/ffw/floor/main\_light/state/set and payload false

Received message with topic shellies/ffw/floor/main\_light/relay/0/command and payload b'off'

Sending message with topic shellies/ffw/floor/main\_light/relay/0 and payload off

Received message with topic shellies/ffw/floor/main\_light/relay/0 and payload b'off'

Received message with topic videv/ffw/floor/main\_light/state and payload b'false'

---

**Success** Value for Shelly.relay/0 (ffw.floor.main\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (ffw.floor.main\_light)): False (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (ffw.floor.main\_light)): result = False (<class 'bool'>)

---

#### A.1.68 Shelly.relay/0 (ffw.floor.main\_light) → ViDevCommon.state (ffw.floor.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/ffw/floor/main\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

---

**Info** Setting state of Shelly.relay/0 (ffw.floor.main\_light) to True

---

Sending message with topic shellies/ffw/floor/main\_light/relay/0 and payload on

Received message with topic shellies/ffw/floor/main\_light/relay/0 and payload b'on'

Received message with topic videv/ffw/floor/main\_light/state and payload b'true'

---

**Success** Value for ViDevCommon.state (ffw.floor.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffw.floor.main\_light)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffw.floor.main\_light)): result = True (<class 'bool'>)  
↪ 'bool'>)

---



---

**Info** Setting state of Shelly.relay/0 (ffw.floor.main\_light) to False

---

Sending message with topic shellies/ffw/floor/main\_light/relay/0 and payload off

Received message with topic shellies/ffw/floor/main\_light/relay/0 and payload b'off'

Received message with topic videv/ffw/floor/main\_light/state and payload b'false'

---

**Success** Value for ViDevCommon.state (ffw.floor.main\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (ffw.floor.main\_light)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (ffw.floor.main\_light)): result = False (<class 'bool'>)  
↪ 'bool'>)

---

#### A.1.69 ViDevCommon.state (gfw.dirk.main\_light) → Shelly.relay/0 (gfw.dirk.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/gfw/dirk/main\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---



---

**Info** Setting state of ViDevCommon.state (gfw.dirk.main\_light) to True

---

Sending message with topic videv/gfw/dirk/main\_light/state/set and payload true

Received message with topic shellies/gfw/dirk/main\_light/relay/0/command and payload b'on'

---

```

Sending message with topic shellies/gfw/dirk/main_light/relay/0 and payload on
Sending message with topic zigbee_gfw/gfw/dirk/main_light and payload {"state": "on",
↪  "brightness": 127.0, "color_temp": 352.0}
Received message with topic shellies/gfw/dirk/main_light/relay/0 and payload b'on'
Received message with topic zigbee_gfw/gfw/dirk/main_light and payload b'{"state": "on",
↪  "brightness": 127.0, "color_temp": 352.0}'
Received message with topic videv/gfw/dirk/main_light/state and payload b'true'
Received message with topic videv/gfw/dirk/main_light/brightness and payload b'50'
Received message with topic videv/gfw/dirk/main_light/color_temp and payload b'5'

```

---

**Success** Value for Shelly.relay/0 (gfw.dirk.main\_light) is correct (Content True and Type is <class 'bool'>).

---

```

Result (Value for Shelly.relay/0 (gfw.dirk.main_light)): True (<class 'bool'>)
Expectation (Value for Shelly.relay/0 (gfw.dirk.main_light)): result = True (<class 'bool'>)

```

---

**Info** Setting state of ViDevCommon.state (gfw.dirk.main\_light) to False

---

```

Sending message with topic videv/gfw/dirk/main_light/state/set and payload false
Received message with topic shellies/gfw/dirk/main_light/relay/0/command and payload b'off'
Sending message with topic shellies/gfw/dirk/main_light/relay/0 and payload off
Received message with topic shellies/gfw/dirk/main_light/relay/0 and payload b'off'
Received message with topic videv/gfw/dirk/main_light/state and payload b'false'

```

---

**Success** Value for Shelly.relay/0 (gfw.dirk.main\_light) is correct (Content False and Type is <class 'bool'>).

---

```

Result (Value for Shelly.relay/0 (gfw.dirk.main_light)): False (<class 'bool'>)
Expectation (Value for Shelly.relay/0 (gfw.dirk.main_light)): result = False (<class 'bool'>)

```

#### A.1.70 Shelly.relay/0 (gfw.dirk.main\_light) → ViDevCommon.state (gfw.dirk.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

```

Sending message with topic videv/gfw/dirk/main_light/state/set and payload false

```

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

```

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)
Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

```

---

**Info** Setting state of Shelly.relay/0 (gfw.dirk.main\_light) to True

---

```

Sending message with topic shellies/gfw/dirk/main_light/relay/0 and payload on

```

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic shellies/gfw/dirk/main\_light/relay/0 and payload b'on'

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/main\_light/state and payload b'true'

**Success** Value for ViDevCommon.state (gfw.dirk.main\_light) is correct (Content True and Type is <class 'bool'>).

Result (Value for ViDevCommon.state (gfw.dirk.main\_light)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (gfw.dirk.main\_light)): result = True (<class  
↪ 'bool'>)

**Info** Setting state of Shelly.relay/0 (gfw.dirk.main\_light) to False

Sending message with topic shellies/gfw/dirk/main\_light/relay/0 and payload off

Received message with topic shellies/gfw/dirk/main\_light/relay/0 and payload b'off'

Received message with topic videv/gfw/dirk/main\_light/state and payload b'false'

**Success** Value for ViDevCommon.state (gfw.dirk.main\_light) is correct (Content False and Type is <class 'bool'>).

Result (Value for ViDevCommon.state (gfw.dirk.main\_light)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (gfw.dirk.main\_light)): result = False (<class  
↪ 'bool'>)

#### A.1.71 ViDevCommon.state (gfw.dirk.desk\_light) → Light.state (gfw.dirk.desk\_light)

##### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Setting devices to last state False

Sending message with topic videv/gfw/dirk/desk\_light/state/set and payload false

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"state": "off"}'

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

**Info** Setting state of ViDevCommon.state (gfw.dirk.desk\_light) to True

Sending message with topic videv/gfw/dirk/desk\_light/state/set and payload true

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"state": "on"}'

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/state and payload b'true'

Received message with topic videv/gfw/dirk/desk\_light/brightness and payload b'50'

Received message with topic videv/gfw/dirk/desk\_light/color\_temp and payload b'5'

---

**Success** Value for Light.state (gfw.dirk.desk\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Light.state (gfw.dirk.desk\_light)): True (<class 'bool'>)

Expectation (Value for Light.state (gfw.dirk.desk\_light)): result = True (<class 'bool'>)

---

**Info** Setting state of ViDevCommon.state (gfw.dirk.desk\_light) to False

---

Sending message with topic videv/gfw/dirk/desk\_light/state/set and payload false

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"state": "off"}'

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/state and payload b'false'

---

**Success** Value for Light.state (gfw.dirk.desk\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Light.state (gfw.dirk.desk\_light)): False (<class 'bool'>)

Expectation (Value for Light.state (gfw.dirk.desk\_light)): result = False (<class 'bool'>)

#### A.1.72 Light.state (gfw.dirk.desk\_light) → ViDevCommon.state (gfw.dirk.desk\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/gfw/dirk/desk\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Light.state (gfw.dirk.desk\_light) to True

---

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/state and payload b'true'

---

**Success** Value for ViDevCommon.state (gfw.dirk.desk\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (gfw.dirk.desk\_light)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (gfw.dirk.desk\_light)): result = True (<class  
↪ 'bool'>)

---

**Info** Setting state of Light.state (gfw.dirk.desk\_light) to False

---

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/state and payload b'false'

---

**Success** Value for ViDevCommon.state (gfw.dirk.desk\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (gfw.dirk.desk\_light)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (gfw.dirk.desk\_light)): result = False (<class  
↪ 'bool'>)

### A.1.73 ViDevCommon.state (gfw.dirk.pc\_dock) → Powerplug1P.state (gfw.dirk.dock)

#### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/gfw/dirk/pc\_dock/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (gfw.dirk.pc\_dock) to True

---

Sending message with topic videv/gfw/dirk/pc\_dock/state/set and payload true

Received message with topic zigbee\_gfw/gfw/dirk/dock/set and payload b'{"state": "on"}'

Sending message with topic zigbee\_gfw/gfw/dirk/dock and payload {"state": "on"}

Received message with topic zigbee\_gfw/gfw/dirk/dock and payload b'{"state": "on"}'

Received message with topic videv/gfw/dirk/pc\_dock/state and payload b'true'

---

**Success** Value for Powerplug1P.state (gfw.dirk.dock) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Powerplug1P.state (gfw.dirk.dock)): True (<class 'bool'>)

Expectation (Value for Powerplug1P.state (gfw.dirk.dock)): result = True (<class 'bool'>)

---

**Info** Setting state of ViDevCommon.state (gfw.dirk.pc\_dock) to False

---

Sending message with topic videv/gfw/dirk/pc\_dock/state/set and payload false

Received message with topic zigbee\_gfw/gfw/dirk/dock/set and payload b'{"state": "off"}'

Sending message with topic zigbee\_gfw/gfw/dirk/dock and payload {"state": "off"}

Received message with topic zigbee\_gfw/gfw/dirk/dock and payload b'{"state": "off"}'

Received message with topic videv/gfw/dirk/pc\_dock/state and payload b'false'

---

**Success** Value for Powerplug1P.state (gfw.dirk.dock) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Powerplug1P.state (gfw.dirk.dock)): False (<class 'bool'>)

Expectation (Value for Powerplug1P.state (gfw.dirk.dock)): result = False (<class 'bool'>)

---

#### A.1.74 Powerplug1P.state (gfw.dirk.dock) → ViDevCommon.state (gfw.dirk.pc\_dock)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/gfw/dirk/pc\_dock/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Powerplug1P.state (gfw.dirk.dock) to True

---

Sending message with topic zigbee\_gfw/gfw/dirk/dock and payload {"state": "on"}

Received message with topic zigbee\_gfw/gfw/dirk/dock and payload b'{"state": "on"}'

Received message with topic videv/gfw/dirk/pc\_dock/state and payload b'true'

---

**Success** Value for ViDevCommon.state (gfw.dirk.pc\_dock) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (gfw.dirk.pc\_dock)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (gfw.dirk.pc\_dock)): result = True (<class 'bool'>)

---

---

**Info** Setting state of Powerplug1P.state (gfw.dirk.dock) to False

---

Sending message with topic zigbee\_gfw/gfw/dirk/dock and payload {"state": "off"}

Received message with topic zigbee\_gfw/gfw/dirk/dock and payload b'{"state": "off"}'

Received message with topic videv/gfw/dirk/pc\_dock/state and payload b'false'

---

**Success** Value for ViDevCommon.state (gfw.dirk.pc\_dock) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (gfw.dirk.pc\_dock)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (gfw.dirk.pc\_dock)): result = False (<class 'bool'>)

---

#### A.1.75 ViDevCommon.state (gfw.dirk.amplifier) → Powerplug4P.amplifier (gfw.dirk.powerplug)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/gfw/dirk/amplifier/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (gfw.dirk.amplifier) to True

---

Sending message with topic videv/gfw/dirk/amplifier/state/set and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'true'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'true'

Received message with topic videv/gfw/dirk/amplifier/state and payload b'true'

---

**Success** Value for Powerplug4P.amplifier (gfw.dirk.powerplug) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Powerplug4P.amplifier (gfw.dirk.powerplug)): True (<class 'bool'>)

Expectation (Value for Powerplug4P.amplifier (gfw.dirk.powerplug)): result = True (<class 'bool'>)

---

**Info** Setting state of ViDevCommon.state (gfw.dirk.amplifier) to False

---

Sending message with topic videv/gfw/dirk/amplifier/state/set and payload false

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'false'

---

Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload false

Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'false'

Received message with topic videv/gfw/dirk/amplifier/state and payload b'false'

**Success** Value for Powerplug4P.amplifier (gfw.dirk.powerplug) is correct (Content False and Type is <class 'bool'>).

Result (Value for Powerplug4P.amplifier (gfw.dirk.powerplug)): False (<class 'bool'>)

Expectation (Value for Powerplug4P.amplifier (gfw.dirk.powerplug)): result = False (<class 'bool'>)

#### A.1.76 Powerplug4P.amplifier (gfw.dirk.powerplug) → ViDevCommon.state (gfw.dirk.amplifier)

##### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Setting devices to last state False

Sending message with topic videv/gfw/dirk/amplifier/state/set and payload false

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

**Info** Setting state of Powerplug4P.amplifier (gfw.dirk.powerplug) to True

Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'true'

Received message with topic videv/gfw/dirk/amplifier/state and payload b'true'

**Success** Value for ViDevCommon.state (gfw.dirk.amplifier) is correct (Content True and Type is <class 'bool'>).

Result (Value for ViDevCommon.state (gfw.dirk.amplifier)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (gfw.dirk.amplifier)): result = True (<class 'bool'>)

**Info** Setting state of Powerplug4P.amplifier (gfw.dirk.powerplug) to False

Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload false

Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'false'

Received message with topic videv/gfw/dirk/amplifier/state and payload b'false'

**Success** Value for ViDevCommon.state (gfw.dirk.amplifier) is correct (Content False and Type is <class 'bool'>).

Result (Value for ViDevCommon.state (gfw.dirk.amplifier)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (gfw.dirk.amplifier)): result = False (<class 'bool'>)

**A.1.77 ViDevCommon.state (gfw.dirk.phono) → Powerplug4P.phono (gfw.dirk.powerplug)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/gfw/dirk/phono/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (gfw.dirk.phono) to True

---

Sending message with topic videv/gfw/dirk/phono/state/set and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/2/set and payload b'true'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/2 and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/2 and payload b'true'

Received message with topic videv/gfw/dirk/phono/state and payload b'true'

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'true'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'true'

---

**Success** Value for Powerplug4P.phono (gfw.dirk.powerplug) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Powerplug4P.phono (gfw.dirk.powerplug)): True (<class 'bool'>)

Expectation (Value for Powerplug4P.phono (gfw.dirk.powerplug)): result = True (<class 'bool'>)

---

**Info** Setting state of ViDevCommon.state (gfw.dirk.phono) to False

---

Sending message with topic videv/gfw/dirk/phono/state/set and payload false

Received message with topic videv/gfw/dirk/amplifier/state and payload b'true'

Received message with topic my\_apps/gfw/dirk/powerplug/output/2/set and payload b'false'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/2 and payload false

Received message with topic my\_apps/gfw/dirk/powerplug/output/2 and payload b'false'

---

**Success** Value for Powerplug4P.phono (gfw.dirk.powerplug) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Powerplug4P.phono (gfw.dirk.powerplug)): False (<class 'bool'>)

Expectation (Value for Powerplug4P.phono (gfw.dirk.powerplug)): result = False (<class 'bool'>)

---

**A.1.78 Powerplug4P.phono (gfw.dirk.powerplug) → ViDevCommon.state (gfw.dirk.phono)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/gfw/dirk/phono/state/set and payload false  
 Received message with topic videv/gfw/dirk/phono/state and payload b'false'  
 Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'false'  
 Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload false  
 Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'false'  
 Received message with topic videv/gfw/dirk/amplifier/state and payload b'false'

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)  
 Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Powerplug4P.phono (gfw.dirk.powerplug) to True

---

Sending message with topic my\_apps/gfw/dirk/powerplug/output/2 and payload true  
 Received message with topic my\_apps/gfw/dirk/powerplug/output/2 and payload b'true'  
 Received message with topic videv/gfw/dirk/phono/state and payload b'true'  
 Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'true'  
 Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload true  
 Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'true'  
 Received message with topic videv/gfw/dirk/amplifier/state and payload b'true'

---

**Success** Value for ViDevCommon.state (gfw.dirk.phono) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (gfw.dirk.phono)): True (<class 'bool'>)  
 Expectation (Value for ViDevCommon.state (gfw.dirk.phono)): result = True (<class 'bool'>)

---

**Info** Setting state of Powerplug4P.phono (gfw.dirk.powerplug) to False

---

Sending message with topic my\_apps/gfw/dirk/powerplug/output/2 and payload false  
 Received message with topic my\_apps/gfw/dirk/powerplug/output/2 and payload b'false'  
 Received message with topic videv/gfw/dirk/phono/state and payload b'false'  
 Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'false'  
 Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload false  
 Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'false'  
 Received message with topic videv/gfw/dirk/amplifier/state and payload b'false'

---

**Success** Value for ViDevCommon.state (gfw.dirk.phono) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (gfw.dirk.phono)): False (<class 'bool'>)  
 Expectation (Value for ViDevCommon.state (gfw.dirk.phono)): result = False (<class 'bool'>)

---

**A.1.79 ViDevCommon.state (gfw.dirk.cd\_player) → Powerplug4P.cd-player (gfw.dirk.powerplug)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/gfw/dirk/cd\_player/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (gfw.dirk.cd\_player) to True

---

Sending message with topic videv/gfw/dirk/cd\_player/state/set and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/3/set and payload b'true'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/3 and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/3 and payload b'true'

Received message with topic videv/gfw/dirk/cd\_player/state and payload b'true'

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'true'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'true'

---

**Success** Value for Powerplug4P.cd-player (gfw.dirk.powerplug) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Powerplug4P.cd-player (gfw.dirk.powerplug)): True (<class 'bool'>)

Expectation (Value for Powerplug4P.cd-player (gfw.dirk.powerplug)): result = True (<class 'bool'>)

---

**Info** Setting state of ViDevCommon.state (gfw.dirk.cd\_player) to False

---

Sending message with topic videv/gfw/dirk/cd\_player/state/set and payload false

Received message with topic videv/gfw/dirk/amplifier/state and payload b'true'

Received message with topic my\_apps/gfw/dirk/powerplug/output/3/set and payload b'false'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/3 and payload false

Received message with topic my\_apps/gfw/dirk/powerplug/output/3 and payload b'false'

Received message with topic videv/gfw/dirk/cd\_player/state and payload b'false'

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'false'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload false

Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'false'

---

**Success** Value for Powerplug4P.cd-player (gfw.dirk.powerplug) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Powerplug4P.cd-player (gfw.dirk.powerplug)): False (<class 'bool'>)

Expectation (Value for Powerplug4P.cd-player (gfw.dirk.powerplug)): result = False (<class 'bool'>)

---

**A.1.80 Powerplug4P.cd-player (gfw.dirk.powerplug) → ViDevCommon.state (gfw.dirk.cd\_player)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/gfw/dirk/cd\_player/state/set and payload false

Received message with topic videv/gfw/dirk/amplifier/state and payload b'false'

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Powerplug4P.cd-player (gfw.dirk.powerplug) to True

---

Sending message with topic my\_apps/gfw/dirk/powerplug/output/3 and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/3 and payload b'true'

Received message with topic videv/gfw/dirk/cd\_player/state and payload b'true'

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'true'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'true'

Received message with topic videv/gfw/dirk/amplifier/state and payload b'true'

---

**Success** Value for ViDevCommon.state (gfw.dirk.cd\_player) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (gfw.dirk.cd\_player)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (gfw.dirk.cd\_player)): result = True (<class 'bool'>)

---

**Info** Setting state of Powerplug4P.cd-player (gfw.dirk.powerplug) to False

---

Sending message with topic my\_apps/gfw/dirk/powerplug/output/3 and payload false

Received message with topic my\_apps/gfw/dirk/powerplug/output/3 and payload b'false'

Received message with topic videv/gfw/dirk/cd\_player/state and payload b'false'

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'false'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload false

Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'false'

Received message with topic videv/gfw/dirk/amplifier/state and payload b'false'

---

**Success** Value for ViDevCommon.state (gfw.dirk.cd\_player) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (gfw.dirk.cd\_player)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (gfw.dirk.cd\_player)): result = False (<class  
↪ 'bool'>)

---

**A.1.81 ViDevCommon.state (gfw.dirk.bt) → Powerplug4P.bluetooth (gfw.dirk.powerplug)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/gfw/dirk/bt/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (gfw.dirk.bt) to True

---

Sending message with topic videv/gfw/dirk/bt/state/set and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/4/set and payload b'true'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/4 and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/4 and payload b'true'

Received message with topic videv/gfw/dirk/bt/state and payload b'true'

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'true'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'true'

---

**Success** Value for Powerplug4P.bluetooth (gfw.dirk.powerplug) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Powerplug4P.bluetooth (gfw.dirk.powerplug)): True (<class 'bool'>)

Expectation (Value for Powerplug4P.bluetooth (gfw.dirk.powerplug)): result = True (<class 'bool'>)

---

**Info** Setting state of ViDevCommon.state (gfw.dirk.bt) to False

---

Sending message with topic videv/gfw/dirk/bt/state/set and payload false

Received message with topic videv/gfw/dirk/amplifier/state and payload b'true'

Received message with topic my\_apps/gfw/dirk/powerplug/output/4/set and payload b'false'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/4 and payload false

Received message with topic my\_apps/gfw/dirk/powerplug/output/4 and payload b'false'

---

**Success** Value for Powerplug4P.bluetooth (gfw.dirk.powerplug) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Powerplug4P.bluetooth (gfw.dirk.powerplug)): False (<class 'bool'>)

Expectation (Value for Powerplug4P.bluetooth (gfw.dirk.powerplug)): result = False (<class 'bool'>)

---

**A.1.82 Powerplug4P.bluetooth (gfw.dirk.powerplug) → ViDevCommon.state (gfw.dirk.bt)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

```
Sending message with topic videv/gfw/dirk/bt/state/set and payload false
Received message with topic videv/gfw/dirk/bt/state and payload b'false'
Received message with topic my_apps/gfw/dirk/powerplug/output/1/set and payload b'false'
Sending message with topic my_apps/gfw/dirk/powerplug/output/1 and payload false
Received message with topic my_apps/gfw/dirk/powerplug/output/1 and payload b'false'
Received message with topic videv/gfw/dirk/amplifier/state and payload b'false'
```

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

```
Result (Start state (master, slave)): (False, False) (<class 'tuple'>)
Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)
```

---

**Info** Setting state of Powerplug4P.bluetooth (gfw.dirk.powerplug) to True

---

```
Sending message with topic my_apps/gfw/dirk/powerplug/output/4 and payload true
Received message with topic my_apps/gfw/dirk/powerplug/output/4 and payload b'true'
Received message with topic videv/gfw/dirk/bt/state and payload b'true'
Received message with topic my_apps/gfw/dirk/powerplug/output/1/set and payload b'true'
Sending message with topic my_apps/gfw/dirk/powerplug/output/1 and payload true
Received message with topic my_apps/gfw/dirk/powerplug/output/1 and payload b'true'
Received message with topic videv/gfw/dirk/amplifier/state and payload b'true'
```

---

**Success** Value for ViDevCommon.state (gfw.dirk.bt) is correct (Content True and Type is <class 'bool'>).

---

```
Result (Value for ViDevCommon.state (gfw.dirk.bt)): True (<class 'bool'>)
Expectation (Value for ViDevCommon.state (gfw.dirk.bt)): result = True (<class 'bool'>)
```

---

**Info** Setting state of Powerplug4P.bluetooth (gfw.dirk.powerplug) to False

---

```
Sending message with topic my_apps/gfw/dirk/powerplug/output/4 and payload false
Received message with topic my_apps/gfw/dirk/powerplug/output/4 and payload b'false'
Received message with topic videv/gfw/dirk/bt/state and payload b'false'
Received message with topic my_apps/gfw/dirk/powerplug/output/1/set and payload b'false'
Sending message with topic my_apps/gfw/dirk/powerplug/output/1 and payload false
Received message with topic my_apps/gfw/dirk/powerplug/output/1 and payload b'false'
Received message with topic videv/gfw/dirk/amplifier/state and payload b'false'
```

---

**Success** Value for ViDevCommon.state (gfw.dirk.bt) is correct (Content False and Type is <class 'bool'>).

---

```
Result (Value for ViDevCommon.state (gfw.dirk.bt)): False (<class 'bool'>)
Expectation (Value for ViDevCommon.state (gfw.dirk.bt)): result = False (<class 'bool'>)
```

---

**A.1.83 Powerplug4P.phono (gfw.dirk.powerplug) → Powerplug4P.amplifier (gfw.dirk.powerplug)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---



---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Powerplug4P.phono (gfw.dirk.powerplug) to True

---

Sending message with topic my\_apps/gfw/dirk/powerplug/output/2 and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/2 and payload b'true'

Received message with topic videv/gfw/dirk/phono/state and payload b'true'

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'true'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'true'

Received message with topic videv/gfw/dirk/amplifier/state and payload b'true'

---

**Success** Value for Powerplug4P.amplifier (gfw.dirk.powerplug) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Powerplug4P.amplifier (gfw.dirk.powerplug)): True (<class 'bool'>)

Expectation (Value for Powerplug4P.amplifier (gfw.dirk.powerplug)): result = True (<class 'bool'>)

---

**Info** Setting state of Powerplug4P.phono (gfw.dirk.powerplug) to False

---

Sending message with topic my\_apps/gfw/dirk/powerplug/output/2 and payload false

Received message with topic my\_apps/gfw/dirk/powerplug/output/2 and payload b'false'

Received message with topic videv/gfw/dirk/phono/state and payload b'false'

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'false'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload false

Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'false'

Received message with topic videv/gfw/dirk/amplifier/state and payload b'false'

---

**Success** Value for Powerplug4P.amplifier (gfw.dirk.powerplug) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Powerplug4P.amplifier (gfw.dirk.powerplug)): False (<class 'bool'>)

Expectation (Value for Powerplug4P.amplifier (gfw.dirk.powerplug)): result = False (<class 'bool'>)

---

**A.1.84 Powerplug4P.cd-player (gfw.dirk.powerplug) → Powerplug4P.amplifier (gfw.dirk.powerplug)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---



---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Powerplug4P.cd-player (gfw.dirk.powerplug) to True

---

Sending message with topic my\_apps/gfw/dirk/powerplug/output/3 and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/3 and payload b'true'

Received message with topic videv/gfw/dirk/cd\_player/state and payload b'true'

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'true'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'true'

Received message with topic videv/gfw/dirk/amplifier/state and payload b'true'

---

**Success** Value for Powerplug4P.amplifier (gfw.dirk.powerplug) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Powerplug4P.amplifier (gfw.dirk.powerplug)): True (<class 'bool'>)

Expectation (Value for Powerplug4P.amplifier (gfw.dirk.powerplug)): result = True (<class 'bool'>)

---

**Info** Setting state of Powerplug4P.cd-player (gfw.dirk.powerplug) to False

---

Sending message with topic my\_apps/gfw/dirk/powerplug/output/3 and payload false

Received message with topic my\_apps/gfw/dirk/powerplug/output/3 and payload b'false'

Received message with topic videv/gfw/dirk/cd\_player/state and payload b'false'

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'false'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload false

Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'false'

Received message with topic videv/gfw/dirk/amplifier/state and payload b'false'

---

**Success** Value for Powerplug4P.amplifier (gfw.dirk.powerplug) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Powerplug4P.amplifier (gfw.dirk.powerplug)): False (<class 'bool'>)

Expectation (Value for Powerplug4P.amplifier (gfw.dirk.powerplug)): result = False (<class 'bool'>)

---

**A.1.85 Powerplug4P.bluetooth (gfw.dirk.powerplug) → Powerplug4P.amplifier (gfw.dirk.powerplug)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---



---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Powerplug4P.bluetooth (gfw.dirk.powerplug) to True

---

Sending message with topic my\_apps/gfw/dirk/powerplug/output/4 and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/4 and payload b'true'

Received message with topic videv/gfw/dirk/bt/state and payload b'true'

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'true'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload true

Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'true'

Received message with topic videv/gfw/dirk/amplifier/state and payload b'true'

---

**Success** Value for Powerplug4P.amplifier (gfw.dirk.powerplug) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Powerplug4P.amplifier (gfw.dirk.powerplug)): True (<class 'bool'>)

Expectation (Value for Powerplug4P.amplifier (gfw.dirk.powerplug)): result = True (<class 'bool'>)

---

**Info** Setting state of Powerplug4P.bluetooth (gfw.dirk.powerplug) to False

---

Sending message with topic my\_apps/gfw/dirk/powerplug/output/4 and payload false

Received message with topic my\_apps/gfw/dirk/powerplug/output/4 and payload b'false'

Received message with topic videv/gfw/dirk/bt/state and payload b'false'

Received message with topic my\_apps/gfw/dirk/powerplug/output/1/set and payload b'false'

Sending message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload false

Received message with topic my\_apps/gfw/dirk/powerplug/output/1 and payload b'false'

Received message with topic videv/gfw/dirk/amplifier/state and payload b'false'

---

**Success** Value for Powerplug4P.amplifier (gfw.dirk.powerplug) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Powerplug4P.amplifier (gfw.dirk.powerplug)): False (<class 'bool'>)

Expectation (Value for Powerplug4P.amplifier (gfw.dirk.powerplug)): result = False (<class 'bool'>)

---

**A.1.86 ViDevCommon.brightness (gfw.dirk.main\_light) → Light.brightness (gfw.dirk.main\_light)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---

```
Sending message with topic shellies/gfw/dirk/main_light/relay/0 and payload on
Sending message with topic zigbee_gfw/gfw/dirk/main_light and payload {"state": "on",
↪ "brightness": 127.0, "color_temp": 352.0}
Received message with topic shellies/gfw/dirk/main_light/relay/0 and payload b'on'
Received message with topic zigbee_gfw/gfw/dirk/main_light and payload b'{"state": "on",
↪ "brightness": 127.0, "color_temp": 352.0}'
Received message with topic videv/gfw/dirk/main_light/state and payload b'true'
```

---

**Info** Prepare: Setting devices to last state 100

---

```
Sending message with topic videv/gfw/dirk/main_light/brightness/set and payload 100
Sending message with topic zigbee_gfw/gfw/dirk/main_light and payload {"state": "on",
↪ "brightness": 254.0, "color_temp": 352.0}
Received message with topic zigbee_gfw/gfw/dirk/main_light/set and payload b'{"brightness":
↪ 254}'
Received message with topic zigbee_gfw/gfw/dirk/main_light and payload b'{"state": "on",
↪ "brightness": 254.0, "color_temp": 352.0}'
Received message with topic videv/gfw/dirk/main_light/brightness and payload b'100'
```

---

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

---

```
Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)
Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)
```

---

**Info** Setting state of ViDevCommon.brightness (gfw.dirk.main\_light) to 0

---

```
Sending message with topic videv/gfw/dirk/main_light/brightness/set and payload 0
Received message with topic zigbee_gfw/gfw/dirk/main_light/set and payload b'{"brightness":
↪ 1}'
Sending message with topic zigbee_gfw/gfw/dirk/main_light and payload {"state": "on",
↪ "brightness": 1.0, "color_temp": 352.0}
Received message with topic zigbee_gfw/gfw/dirk/main_light and payload b'{"state": "on",
↪ "brightness": 1.0, "color_temp": 352.0}'
Received message with topic videv/gfw/dirk/main_light/brightness and payload b'0'
```

---

**Success** Value for Light.brightness (gfw.dirk.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

```
Result (Value for Light.brightness (gfw.dirk.main_light)): 0 (<class 'int'>)
```

---

```
Expectation (Value for Light.brightness (gfw.dirk.main_light)): result = 0 (<class 'int'>)
```

---

**Info** Setting state of ViDevCommon.brightness (gfw.dirk.main\_light) to 20

---

```
Sending message with topic videv/gfw/dirk/main_light/brightness/set and payload 20
```

```
Received message with topic zigbee_gfw/gfw/dirk/main_light/set and payload b'{"brightness":
↪ 52}'
```

```
Sending message with topic zigbee_gfw/gfw/dirk/main_light and payload {"state": "on",
↪ "brightness": 52.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_gfw/gfw/dirk/main_light and payload b'{"state": "on",
↪ "brightness": 52.0, "color_temp": 352.0}'
```

```
Received message with topic videv/gfw/dirk/main_light/brightness and payload b'20'
```

---

**Success** Value for Light.brightness (gfw.dirk.main\_light) is correct (Content 20 and Type is <class 'int'>).

---

```
Result (Value for Light.brightness (gfw.dirk.main_light)): 20 (<class 'int'>)
```

```
Expectation (Value for Light.brightness (gfw.dirk.main_light)): result = 20 (<class 'int'>)
```

---

**Info** Setting state of ViDevCommon.brightness (gfw.dirk.main\_light) to 40

---

```
Sending message with topic videv/gfw/dirk/main_light/brightness/set and payload 40
```

```
Received message with topic zigbee_gfw/gfw/dirk/main_light/set and payload b'{"brightness":
↪ 102}'
```

```
Sending message with topic zigbee_gfw/gfw/dirk/main_light and payload {"state": "on",
↪ "brightness": 102.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_gfw/gfw/dirk/main_light and payload b'{"state": "on",
↪ "brightness": 102.0, "color_temp": 352.0}'
```

```
Received message with topic videv/gfw/dirk/main_light/brightness and payload b'40'
```

---

**Success** Value for Light.brightness (gfw.dirk.main\_light) is correct (Content 40 and Type is <class 'int'>).

---

```
Result (Value for Light.brightness (gfw.dirk.main_light)): 40 (<class 'int'>)
```

```
Expectation (Value for Light.brightness (gfw.dirk.main_light)): result = 40 (<class 'int'>)
```

---

**Info** Setting state of ViDevCommon.brightness (gfw.dirk.main\_light) to 60

---

```
Sending message with topic videv/gfw/dirk/main_light/brightness/set and payload 60
```

```
Received message with topic zigbee_gfw/gfw/dirk/main_light/set and payload b'{"brightness":
↪ 153}'
```

```
Sending message with topic zigbee_gfw/gfw/dirk/main_light and payload {"state": "on",
↪ "brightness": 153.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_gfw/gfw/dirk/main_light and payload b'{"state": "on",
↪ "brightness": 153.0, "color_temp": 352.0}'
```

```
Received message with topic videv/gfw/dirk/main_light/brightness and payload b'60'
```

---

**Success** Value for Light.brightness (gfw.dirk.main\_light) is correct (Content 60 and Type is <class 'int'>).

---

Result (Value for Light.brightness (gfw.dirk.main\_light)): 60 (<class 'int'>)

Expectation (Value for Light.brightness (gfw.dirk.main\_light)): result = 60 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (gfw.dirk.main\_light) to 80

---

Sending message with topic videv/gfw/dirk/main\_light/brightness/set and payload 80

Received message with topic zigbee\_gfw/gfw/dirk/main\_light/set and payload b'{"brightness":  
↪ 203}'

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/main\_light/brightness and payload b'80'

---

**Success** Value for Light.brightness (gfw.dirk.main\_light) is correct (Content 80 and Type is <class 'int'>).

---

Result (Value for Light.brightness (gfw.dirk.main\_light)): 80 (<class 'int'>)

Expectation (Value for Light.brightness (gfw.dirk.main\_light)): result = 80 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (gfw.dirk.main\_light) to 100

---

Sending message with topic videv/gfw/dirk/main\_light/brightness/set and payload 100

Received message with topic zigbee\_gfw/gfw/dirk/main\_light/set and payload b'{"brightness":  
↪ 254}'

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/main\_light/brightness and payload b'100'

---

**Success** Value for Light.brightness (gfw.dirk.main\_light) is correct (Content 100 and Type is <class 'int'>).

---

Result (Value for Light.brightness (gfw.dirk.main\_light)): 100 (<class 'int'>)

Expectation (Value for Light.brightness (gfw.dirk.main\_light)): result = 100 (<class 'int'>)

---

#### A.1.87 Light.brightness (gfw.dirk.main\_light) → ViDevCommon.brightness (gfw.dirk.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 100

---

Sending message with topic videv/gfw/dirk/main\_light/brightness/set and payload 100

---

---

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

---

**Info** Setting state of Light.brightness (gfw.dirk.main\_light) to 0

---

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/main\_light/brightness and payload b'0'

---

**Success** Value for ViDevCommon.brightness (gfw.dirk.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (gfw.dirk.main\_light)): 0 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (gfw.dirk.main\_light)): result = 0 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (gfw.dirk.main\_light) to 20

---

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/main\_light/brightness and payload b'20'

---

**Success** Value for ViDevCommon.brightness (gfw.dirk.main\_light) is correct (Content 20 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (gfw.dirk.main\_light)): 20 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (gfw.dirk.main\_light)): result = 20 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (gfw.dirk.main\_light) to 40

---

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/main\_light/brightness and payload b'40'

---

---

**Success** Value for ViDevCommon.brightness (gfw.dirk.main\_light) is correct (Content 40 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (gfw.dirk.main\_light)): 40 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (gfw.dirk.main\_light)): result = 40 (<class 'int'>)  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (gfw.dirk.main\_light) to 60

---

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/main\_light/brightness and payload b'60'

---

**Success** Value for ViDevCommon.brightness (gfw.dirk.main\_light) is correct (Content 60 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (gfw.dirk.main\_light)): 60 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (gfw.dirk.main\_light)): result = 60 (<class 'int'>)  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (gfw.dirk.main\_light) to 80

---

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/main\_light/brightness and payload b'80'

---

**Success** Value for ViDevCommon.brightness (gfw.dirk.main\_light) is correct (Content 80 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (gfw.dirk.main\_light)): 80 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (gfw.dirk.main\_light)): result = 80 (<class 'int'>)  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (gfw.dirk.main\_light) to 100

---

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/main\_light/brightness and payload b'100'

---

**Success** Value for ViDevCommon.brightness (gfw.dirk.main\_light) is correct (Content 100 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (gfw.dirk.main\_light)): 100 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (gfw.dirk.main\_light)): result = 100 (<class 'int'>)

---

#### A.1.88 ViDevCommon.color\_temp (gfw.dirk.main\_light) → Light.color\_temp (gfw.dirk.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 10

---

Sending message with topic videv/gfw/dirk/main\_light/color\_temp/set and payload 10

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light/set and payload b'{"color\_temp":  
↪ 454}'

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/gfw/dirk/main\_light/color\_temp and payload b'10'

---

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (10, 10) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (10, 10) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.dirk.main\_light) to 0

---

Sending message with topic videv/gfw/dirk/main\_light/color\_temp/set and payload 0

Received message with topic zigbee\_gfw/gfw/dirk/main\_light/set and payload b'{"color\_temp":  
↪ 250}'

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic videv/gfw/dirk/main\_light/color\_temp and payload b'0'

---

**Success** Value for Light.color\_temp (gfw.dirk.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (gfw.dirk.main\_light)): 0 (<class 'int'>)

---

```
Expectation (Value for Light.color_temp (gfw.dirk.main_light)): result = 0 (<class 'int'>)
```

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.dirk.main\_light) to 2

---

```
Sending message with topic videv/gfw/dirk/main_light/color_temp/set and payload 2
```

```
Received message with topic zigbee_gfw/gfw/dirk/main_light/set and payload b'{"color_temp":  
↪ 291}'
```

```
Sending message with topic zigbee_gfw/gfw/dirk/main_light and payload {"state": "on",  
↪ "brightness": 254.0, "color_temp": 291.0}
```

```
Received message with topic zigbee_gfw/gfw/dirk/main_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color_temp": 291.0}'
```

```
Received message with topic videv/gfw/dirk/main_light/color_temp and payload b'2'
```

---

**Success** Value for Light.color\_temp (gfw.dirk.main\_light) is correct (Content 2 and Type is <class 'int'>).

---

```
Result (Value for Light.color_temp (gfw.dirk.main_light)): 2 (<class 'int'>)
```

```
Expectation (Value for Light.color_temp (gfw.dirk.main_light)): result = 2 (<class 'int'>)
```

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.dirk.main\_light) to 4

---

```
Sending message with topic videv/gfw/dirk/main_light/color_temp/set and payload 4
```

```
Received message with topic zigbee_gfw/gfw/dirk/main_light/set and payload b'{"color_temp":  
↪ 332}'
```

```
Sending message with topic zigbee_gfw/gfw/dirk/main_light and payload {"state": "on",  
↪ "brightness": 254.0, "color_temp": 332.0}
```

```
Received message with topic zigbee_gfw/gfw/dirk/main_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color_temp": 332.0}'
```

```
Received message with topic videv/gfw/dirk/main_light/color_temp and payload b'4'
```

---

**Success** Value for Light.color\_temp (gfw.dirk.main\_light) is correct (Content 4 and Type is <class 'int'>).

---

```
Result (Value for Light.color_temp (gfw.dirk.main_light)): 4 (<class 'int'>)
```

```
Expectation (Value for Light.color_temp (gfw.dirk.main_light)): result = 4 (<class 'int'>)
```

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.dirk.main\_light) to 6

---

```
Sending message with topic videv/gfw/dirk/main_light/color_temp/set and payload 6
```

```
Received message with topic zigbee_gfw/gfw/dirk/main_light/set and payload b'{"color_temp":  
↪ 372}'
```

```
Sending message with topic zigbee_gfw/gfw/dirk/main_light and payload {"state": "on",  
↪ "brightness": 254.0, "color_temp": 372.0}
```

```
Received message with topic zigbee_gfw/gfw/dirk/main_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color_temp": 372.0}'
```

```
Received message with topic videv/gfw/dirk/main_light/color_temp and payload b'6'
```

---

**Success** Value for Light.color\_temp (gfw.dirk.main\_light) is correct (Content 6 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (gfw.dirk.main\_light)): 6 (<class 'int'>)

Expectation (Value for Light.color\_temp (gfw.dirk.main\_light)): result = 6 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.dirk.main\_light) to 8

---

Sending message with topic videv/gfw/dirk/main\_light/color\_temp/set and payload 8

Received message with topic zigbee\_gfw/gfw/dirk/main\_light/set and payload b'{"color\_temp":  
↪ 413}'

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic videv/gfw/dirk/main\_light/color\_temp and payload b'8'

---

**Success** Value for Light.color\_temp (gfw.dirk.main\_light) is correct (Content 8 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (gfw.dirk.main\_light)): 8 (<class 'int'>)

Expectation (Value for Light.color\_temp (gfw.dirk.main\_light)): result = 8 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.dirk.main\_light) to 10

---

Sending message with topic videv/gfw/dirk/main\_light/color\_temp/set and payload 10

Received message with topic zigbee\_gfw/gfw/dirk/main\_light/set and payload b'{"color\_temp":  
↪ 454}'

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/gfw/dirk/main\_light/color\_temp and payload b'10'

---

**Success** Value for Light.color\_temp (gfw.dirk.main\_light) is correct (Content 10 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (gfw.dirk.main\_light)): 10 (<class 'int'>)

Expectation (Value for Light.color\_temp (gfw.dirk.main\_light)): result = 10 (<class 'int'>)

---

#### A.1.89 Light.color\_temp (gfw.dirk.main\_light) → ViDevCommon.color\_temp (gfw.dirk.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 10

---

Sending message with topic videv/gfw/dirk/main\_light/color\_temp/set and payload 10

---

---

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (10, 10) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (10, 10) (<class 'tuple'>)

---

**Info** Setting state of Light.color\_temp (gfw.dirk.main\_light) to 0

---

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic videv/gfw/dirk/main\_light/color\_temp and payload b'0'

---

**Success** Value for ViDevCommon.color\_temp (gfw.dirk.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.dirk.main\_light)): 0 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.dirk.main\_light)): result = 0 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (gfw.dirk.main\_light) to 2

---

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic videv/gfw/dirk/main\_light/color\_temp and payload b'2'

---

**Success** Value for ViDevCommon.color\_temp (gfw.dirk.main\_light) is correct (Content 2 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.dirk.main\_light)): 2 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.dirk.main\_light)): result = 2 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (gfw.dirk.main\_light) to 4

---

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic videv/gfw/dirk/main\_light/color\_temp and payload b'4'

---

---

**Success** Value for ViDevCommon.color\_temp (gfw.dirk.main\_light) is correct (Content 4 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.dirk.main\_light)): 4 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.dirk.main\_light)): result = 4 (<class 'int'>)

---

**Info** Setting state of Light.color\_temp (gfw.dirk.main\_light) to 6

---

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic videv/gfw/dirk/main\_light/color\_temp and payload b'6'

---

**Success** Value for ViDevCommon.color\_temp (gfw.dirk.main\_light) is correct (Content 6 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.dirk.main\_light)): 6 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.dirk.main\_light)): result = 6 (<class 'int'>)

---

**Info** Setting state of Light.color\_temp (gfw.dirk.main\_light) to 8

---

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic videv/gfw/dirk/main\_light/color\_temp and payload b'8'

---

**Success** Value for ViDevCommon.color\_temp (gfw.dirk.main\_light) is correct (Content 8 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.dirk.main\_light)): 8 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.dirk.main\_light)): result = 8 (<class 'int'>)

---

**Info** Setting state of Light.color\_temp (gfw.dirk.main\_light) to 10

---

Sending message with topic zigbee\_gfw/gfw/dirk/main\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_gfw/gfw/dirk/main\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/gfw/dirk/main\_light/color\_temp and payload b'10'

---

**Success** Value for ViDevCommon.color\_temp (gfw.dirk.main\_light) is correct (Content 10 and Type is <class 'int'>).

Result (Value for ViDevCommon.color\_temp (gfw.dirk.main\_light)): 10 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.dirk.main\_light)): result = 10 (<class 'int'>)

#### A.1.90 ViDevCommon.brightness (gfw.dirk.desk\_light) → Light.brightness (gfw.dirk.desk\_light)

##### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Switching on device

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/state and payload b'true'

**Info** Prepare: Setting devices to last state 100

Sending message with topic videv/gfw/dirk/desk\_light/brightness/set and payload 100

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"brightness":  
↪ 254}'

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/brightness and payload b'100'

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

**Info** Setting state of ViDevCommon.brightness (gfw.dirk.desk\_light) to 0

Sending message with topic videv/gfw/dirk/desk\_light/brightness/set and payload 0

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"brightness":  
↪ 1}'

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/brightness and payload b'0'

**Success** Value for Light.brightness (gfw.dirk.desk\_light) is correct (Content 0 and Type is <class 'int'>).

Result (Value for Light.brightness (gfw.dirk.desk\_light)): 0 (<class 'int'>)

Expectation (Value for Light.brightness (gfw.dirk.desk\_light)): result = 0 (<class 'int'>)

**Info** Setting state of ViDevCommon.brightness (gfw.dirk.desk\_light) to 20

Sending message with topic videv/gfw/dirk/desk\_light/brightness/set and payload 20

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"brightness":  
↪ 52}'

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/brightness and payload b'20'

**Success** Value for Light.brightness (gfw.dirk.desk\_light) is correct (Content 20 and Type is <class 'int'>).

Result (Value for Light.brightness (gfw.dirk.desk\_light)): 20 (<class 'int'>)

Expectation (Value for Light.brightness (gfw.dirk.desk\_light)): result = 20 (<class 'int'>)

**Info** Setting state of ViDevCommon.brightness (gfw.dirk.desk\_light) to 40

Sending message with topic videv/gfw/dirk/desk\_light/brightness/set and payload 40

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"brightness":  
↪ 102}'

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/brightness and payload b'40'

**Success** Value for Light.brightness (gfw.dirk.desk\_light) is correct (Content 40 and Type is <class 'int'>).

Result (Value for Light.brightness (gfw.dirk.desk\_light)): 40 (<class 'int'>)

Expectation (Value for Light.brightness (gfw.dirk.desk\_light)): result = 40 (<class 'int'>)

**Info** Setting state of ViDevCommon.brightness (gfw.dirk.desk\_light) to 60

Sending message with topic videv/gfw/dirk/desk\_light/brightness/set and payload 60

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"brightness":  
↪ 153}'

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/brightness and payload b'60'

---

**Success** Value for Light.brightness (gfw.dirk.desk\_light) is correct (Content 60 and Type is <class 'int'>).

---

Result (Value for Light.brightness (gfw.dirk.desk\_light)): 60 (<class 'int'>)

Expectation (Value for Light.brightness (gfw.dirk.desk\_light)): result = 60 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (gfw.dirk.desk\_light) to 80

---

Sending message with topic videv/gfw/dirk/desk\_light/brightness/set and payload 80

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"brightness":  
↪ 203}'

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/brightness and payload b'80'

---

**Success** Value for Light.brightness (gfw.dirk.desk\_light) is correct (Content 80 and Type is <class 'int'>).

---

Result (Value for Light.brightness (gfw.dirk.desk\_light)): 80 (<class 'int'>)

Expectation (Value for Light.brightness (gfw.dirk.desk\_light)): result = 80 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (gfw.dirk.desk\_light) to 100

---

Sending message with topic videv/gfw/dirk/desk\_light/brightness/set and payload 100

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"brightness":  
↪ 254}'

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/brightness and payload b'100'

---

**Success** Value for Light.brightness (gfw.dirk.desk\_light) is correct (Content 100 and Type is <class 'int'>).

---

Result (Value for Light.brightness (gfw.dirk.desk\_light)): 100 (<class 'int'>)

Expectation (Value for Light.brightness (gfw.dirk.desk\_light)): result = 100 (<class 'int'>)

---

**A.1.91 Light.brightness (gfw.dirk.desk\_light) → ViDevCommon.brightness (gfw.dirk.desk\_light)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 100

---

Sending message with topic videv/gfw/dirk/desk\_light/brightness/set and payload 100

---

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

---

**Info** Setting state of Light.brightness (gfw.dirk.desk\_light) to 0

---

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/brightness and payload b'0'

---

**Success** Value for ViDevCommon.brightness (gfw.dirk.desk\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (gfw.dirk.desk\_light)): 0 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (gfw.dirk.desk\_light)): result = 0 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (gfw.dirk.desk\_light) to 20

---

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/brightness and payload b'20'

---

**Success** Value for ViDevCommon.brightness (gfw.dirk.desk\_light) is correct (Content 20 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (gfw.dirk.desk\_light)): 20 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (gfw.dirk.desk\_light)): result = 20 (<class  
↪ 'int'>)

---

---

**Info** Setting state of Light.brightness (gfw.dirk.desk\_light) to 40

---

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/brightness and payload b'40'

---

**Success** Value for ViDevCommon.brightness (gfw.dirk.desk\_light) is correct (Content 40 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (gfw.dirk.desk\_light)): 40 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (gfw.dirk.desk\_light)): result = 40 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (gfw.dirk.desk\_light) to 60

---

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/brightness and payload b'60'

---

**Success** Value for ViDevCommon.brightness (gfw.dirk.desk\_light) is correct (Content 60 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (gfw.dirk.desk\_light)): 60 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (gfw.dirk.desk\_light)): result = 60 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (gfw.dirk.desk\_light) to 80

---

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/brightness and payload b'80'

---

**Success** Value for ViDevCommon.brightness (gfw.dirk.desk\_light) is correct (Content 80 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (gfw.dirk.desk\_light)): 80 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (gfw.dirk.desk\_light)): result = 80 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (gfw.dirk.desk\_light) to 100

---

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/dirk/desk\_light/brightness and payload b'100'

---

**Success** Value for ViDevCommon.brightness (gfw.dirk.desk\_light) is correct (Content 100 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (gfw.dirk.desk\_light)): 100 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (gfw.dirk.desk\_light)): result = 100 (<class  
↪ 'int'>)

---

#### A.1.92 ViDevCommon.color\_temp (gfw.dirk.desk\_light) → Light.color\_temp (gfw.dirk.desk\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 10

---

Sending message with topic videv/gfw/dirk/desk\_light/color\_temp/set and payload 10

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"color\_temp":  
↪ 454}'

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/gfw/dirk/desk\_light/color\_temp and payload b'10'

---

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (10, 10) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (10, 10) (<class 'tuple'>)

---



---

**Info** Setting state of ViDevCommon.color\_temp (gfw.dirk.desk\_light) to 0

---

Sending message with topic videv/gfw/dirk/desk\_light/color\_temp/set and payload 0

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"color\_temp":  
↪ 250}'

---

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic videv/gfw/dirk/desk\_light/color\_temp and payload b'0'

---

**Success** Value for Light.color\_temp (gfw.dirk.desk\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (gfw.dirk.desk\_light)): 0 (<class 'int'>)

Expectation (Value for Light.color\_temp (gfw.dirk.desk\_light)): result = 0 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.dirk.desk\_light) to 2

---

Sending message with topic videv/gfw/dirk/desk\_light/color\_temp/set and payload 2

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"color\_temp":  
↪ 291}'

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic videv/gfw/dirk/desk\_light/color\_temp and payload b'2'

---

**Success** Value for Light.color\_temp (gfw.dirk.desk\_light) is correct (Content 2 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (gfw.dirk.desk\_light)): 2 (<class 'int'>)

Expectation (Value for Light.color\_temp (gfw.dirk.desk\_light)): result = 2 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.dirk.desk\_light) to 4

---

Sending message with topic videv/gfw/dirk/desk\_light/color\_temp/set and payload 4

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"color\_temp":  
↪ 332}'

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic videv/gfw/dirk/desk\_light/color\_temp and payload b'4'

---

**Success** Value for Light.color\_temp (gfw.dirk.desk\_light) is correct (Content 4 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (gfw.dirk.desk\_light)): 4 (<class 'int'>)

Expectation (Value for Light.color\_temp (gfw.dirk.desk\_light)): result = 4 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.dirk.desk\_light) to 6

---

Sending message with topic videv/gfw/dirk/desk\_light/color\_temp/set and payload 6

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"color\_temp":  
↪ 372}'

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic videv/gfw/dirk/desk\_light/color\_temp and payload b'6'

---

**Success** Value for Light.color\_temp (gfw.dirk.desk\_light) is correct (Content 6 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (gfw.dirk.desk\_light)): 6 (<class 'int'>)

Expectation (Value for Light.color\_temp (gfw.dirk.desk\_light)): result = 6 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.dirk.desk\_light) to 8

---

Sending message with topic videv/gfw/dirk/desk\_light/color\_temp/set and payload 8

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"color\_temp":  
↪ 413}'

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic videv/gfw/dirk/desk\_light/color\_temp and payload b'8'

---

**Success** Value for Light.color\_temp (gfw.dirk.desk\_light) is correct (Content 8 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (gfw.dirk.desk\_light)): 8 (<class 'int'>)

Expectation (Value for Light.color\_temp (gfw.dirk.desk\_light)): result = 8 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.dirk.desk\_light) to 10

---

Sending message with topic videv/gfw/dirk/desk\_light/color\_temp/set and payload 10

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light/set and payload b'{"color\_temp":  
↪ 454}'

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/gfw/dirk/desk\_light/color\_temp and payload b'10'

---

**Success** Value for Light.color\_temp (gfw.dirk.desk\_light) is correct (Content 10 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (gfw.dirk.desk\_light)): 10 (<class 'int'>)

Expectation (Value for Light.color\_temp (gfw.dirk.desk\_light)): result = 10 (<class 'int'>)

**A.1.93 Light.color\_temp (gfw.dirk.desk\_light) → ViDevCommon.color\_temp (gfw.dirk.desk\_light)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 10

---

Sending message with topic videv/gfw/dirk/desk\_light/color\_temp/set and payload 10

---

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (10, 10) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (10, 10) (<class 'tuple'>)

---

**Info** Setting state of Light.color\_temp (gfw.dirk.desk\_light) to 0

---

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic videv/gfw/dirk/desk\_light/color\_temp and payload b'0'

---

**Success** Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light)): 0 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light)): result = 0 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (gfw.dirk.desk\_light) to 2

---

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic videv/gfw/dirk/desk\_light/color\_temp and payload b'2'

---

**Success** Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light) is correct (Content 2 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light)): 2 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light)): result = 2 (<class  
↪ 'int'>)

---

---

**Info** Setting state of Light.color\_temp (gfw.dirk.desk\_light) to 4

---

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic videv/gfw/dirk/desk\_light/color\_temp and payload b'4'

---

**Success** Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light) is correct (Content 4 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light)): 4 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light)): result = 4 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (gfw.dirk.desk\_light) to 6

---

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic videv/gfw/dirk/desk\_light/color\_temp and payload b'6'

---

**Success** Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light) is correct (Content 6 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light)): 6 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light)): result = 6 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (gfw.dirk.desk\_light) to 8

---

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic videv/gfw/dirk/desk\_light/color\_temp and payload b'8'

---

**Success** Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light) is correct (Content 8 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light)): 8 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light)): result = 8 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (gfw.dirk.desk\_light) to 10

---

Sending message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_gfw/gfw/dirk/desk\_light and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/gfw/dirk/desk\_light/color\_temp and payload b'10'

---

**Success** Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light) is correct (Content 10 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light)): 10 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.dirk.desk\_light)): result = 10 (<class  
↪ 'int'>)

---

#### A.1.94 ViDevHeating.temp\_setp (gfw.dirk.heating\_valve) → HeatingValve.temp\_setp (gfw.dirk.heating\_valve)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state 30

---

Sending message with topic videv/gfw/dirk/heating\_valve/user\_temperature\_setpoint/set and  
↪ payload 30

Sending message with topic zigbee\_gfw/gfw/dirk/heating\_valve and payload  
↪ {"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}

Received message with topic zigbee\_gfw/gfw/dirk/heating\_valve/set and payload  
↪ b'{"current\_heating\_setpoint": 30}'

Received message with topic videv/gfw/dirk/heating\_valve/valve\_temperature\_setpoint and  
↪ payload b'30'

Received message with topic videv/gfw/dirk/heating\_valve/user\_temperature\_setpoint and payload  
↪ b'30'

Received message with topic zigbee\_gfw/gfw/dirk/heating\_valve and payload  
↪ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (30, 30) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (30, 30) (<class 'tuple'>)

---

**Info** Setting state of ViDevHeating.temp\_setp (gfw.dirk.heating\_valve) to 15

---

Sending message with topic videv/gfw/dirk/heating\_valve/user\_temperature\_setpoint/set and  
↪ payload 15

Received message with topic zigbee\_gfw/gfw/dirk/heating\_valve/set and payload

↪ b'{"current\_heating\_setpoint": 15}'

Sending message with topic zigbee\_gfw/gfw/dirk/heating\_valve and payload

↪ {"current\_heating\_setpoint": 15, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/gfw/dirk/heating\_valve/valve\_temperature\_setpoint and

↪ payload b'15'

Received message with topic videv/gfw/dirk/heating\_valve/user\_temperature\_setpoint and payload

↪ b'15'

Received message with topic zigbee\_gfw/gfw/dirk/heating\_valve and payload

↪ b'{"current\_heating\_setpoint": 15, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Value for HeatingValve.temp\_setp (gfw.dirk.heating\_valve) is correct (Content 15 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (gfw.dirk.heating\_valve)): 15 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (gfw.dirk.heating\_valve)): result = 15 (<class 'int'>)

---

**Info** Setting state of ViDevHeating.temp\_setp (gfw.dirk.heating\_valve) to 20

---

Sending message with topic videv/gfw/dirk/heating\_valve/user\_temperature\_setpoint/set and

↪ payload 20

Received message with topic zigbee\_gfw/gfw/dirk/heating\_valve/set and payload

↪ b'{"current\_heating\_setpoint": 20}'

Sending message with topic zigbee\_gfw/gfw/dirk/heating\_valve and payload

↪ {"current\_heating\_setpoint": 20, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/gfw/dirk/heating\_valve/valve\_temperature\_setpoint and

↪ payload b'20'

Received message with topic videv/gfw/dirk/heating\_valve/user\_temperature\_setpoint and payload

↪ b'20'

Received message with topic zigbee\_gfw/gfw/dirk/heating\_valve and payload

↪ b'{"current\_heating\_setpoint": 20, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Value for HeatingValve.temp\_setp (gfw.dirk.heating\_valve) is correct (Content 20 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (gfw.dirk.heating\_valve)): 20 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (gfw.dirk.heating\_valve)): result = 20 (<class 'int'>)

---

**Info** Setting state of ViDevHeating.temp\_setp (gfw.dirk.heating\_valve) to 25

---

Sending message with topic videv/gfw/dirk/heating\_valve/user\_temperature\_setpoint/set and

↪ payload 25

Received message with topic zigbee\_gfw/gfw/dirk/heating\_valve/set and payload

↪ b'{"current\_heating\_setpoint": 25}'

Sending message with topic zigbee\_gfw/gfw/dirk/heating\_valve and payload

↪ {"current\_heating\_setpoint": 25, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/gfw/dirk/heating\_valve/valve\_temperature\_setpoint and

↪ payload b'25'

Received message with topic videv/gfw/dirk/heating\_valve/user\_temperature\_setpoint and payload

↪ b'25'

Received message with topic zigbee\_gfw/gfw/dirk/heating\_valve and payload

↪ b'{"current\_heating\_setpoint": 25, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Value for HeatingValve.temp\_setp (gfw.dirk.heating\_valve) is correct (Content 25 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (gfw.dirk.heating\_valve)): 25 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (gfw.dirk.heating\_valve)): result = 25 (<class 'int'>)

---

**Info** Setting state of ViDevHeating.temp\_setp (gfw.dirk.heating\_valve) to 30

---

Sending message with topic videv/gfw/dirk/heating\_valve/user\_temperature\_setpoint/set and

↪ payload 30

Received message with topic zigbee\_gfw/gfw/dirk/heating\_valve/set and payload

↪ b'{"current\_heating\_setpoint": 30}'

Sending message with topic zigbee\_gfw/gfw/dirk/heating\_valve and payload

↪ {"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/gfw/dirk/heating\_valve/valve\_temperature\_setpoint and

↪ payload b'30'

Received message with topic videv/gfw/dirk/heating\_valve/user\_temperature\_setpoint and payload

↪ b'30'

Received message with topic zigbee\_gfw/gfw/dirk/heating\_valve and payload

↪ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Value for HeatingValve.temp\_setp (gfw.dirk.heating\_valve) is correct (Content 30 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (gfw.dirk.heating\_valve)): 30 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (gfw.dirk.heating\_valve)): result = 30 (<class 'int'>)

---

#### A.1.95 ViDevCommon.state (gfw.marion.main\_light) → Shelly.relay/0 (gfw.marion.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/gfw/marion/main\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (gfw.marion.main\_light) to True

---

Sending message with topic videv/gfw/marion/main\_light/state/set and payload true

Received message with topic shellies/gfw/marion/main\_light/relay/0/command and payload b'on'

Sending message with topic shellies/gfw/marion/main\_light/relay/0 and payload on

Received message with topic shellies/gfw/marion/main\_light/relay/0 and payload b'on'

Received message with topic videv/gfw/marion/main\_light/state and payload b'true'

Received message with topic zigbee\_gfw/gfw/marion/window\_light/set and payload b'{"state":  
↪ "on"}'

Sending message with topic zigbee\_gfw/gfw/marion/window\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/marion/window\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

---

**Success** Value for Shelly.relay/0 (gfw.marion.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (gfw.marion.main\_light)): True (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (gfw.marion.main\_light)): result = True (<class 'bool'>)

---

**Info** Setting state of ViDevCommon.state (gfw.marion.main\_light) to False

---

Sending message with topic videv/gfw/marion/main\_light/state/set and payload false

Received message with topic videv/gfw/marion/window\_light/state and payload b'true'

Received message with topic videv/gfw/marion/window\_light/brightness and payload b'50'

Received message with topic videv/gfw/marion/window\_light/color\_temp and payload b'5'

Received message with topic shellies/gfw/marion/main\_light/relay/0/command and payload b'off'

Sending message with topic shellies/gfw/marion/main\_light/relay/0 and payload off

Received message with topic shellies/gfw/marion/main\_light/relay/0 and payload b'off'

---

**Success** Value for Shelly.relay/0 (gfw.marion.main\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (gfw.marion.main\_light)): False (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (gfw.marion.main\_light)): result = False (<class  
↪ 'bool'>)

---

**A.1.96 Shelly.relay/0 (gfw.marion.main\_light) → ViDevCommon.state (gfw.marion.main\_light)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/gfw/marion/main\_light/state/set and payload false

Received message with topic videv/gfw/marion/main\_light/state and payload b'false'

Received message with topic zigbee\_gfw/gfw/marion/window\_light/set and payload b'{"state":  
↪ "off"}'

Sending message with topic zigbee\_gfw/gfw/marion/window\_light and payload {"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/marion/window\_light and payload b'{"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Shelly.relay/0 (gfw.marion.main\_light) to True

---

Sending message with topic shellies/gfw/marion/main\_light/relay/0 and payload on

Received message with topic videv/gfw/marion/window\_light/state and payload b'false'

Received message with topic shellies/gfw/marion/main\_light/relay/0 and payload b'on'

Received message with topic videv/gfw/marion/main\_light/state and payload b'true'

Received message with topic zigbee\_gfw/gfw/marion/window\_light/set and payload b'{"state":  
↪ "on"}'

Sending message with topic zigbee\_gfw/gfw/marion/window\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/marion/window\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/marion/window\_light/state and payload b'true'

---

**Success** Value for ViDevCommon.state (gfw.marion.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (gfw.marion.main\_light)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (gfw.marion.main\_light)): result = True (<class  
↪ 'bool'>)

---

**Info** Setting state of Shelly.relay/0 (gfw.marion.main\_light) to False

---

Sending message with topic shellies/gfw/marion/main\_light/relay/0 and payload off

Received message with topic shellies/gfw/marion/main\_light/relay/0 and payload b'off'

Received message with topic videv/gfw/marion/main\_light/state and payload b'false'

Received message with topic zigbee\_gfw/gfw/marion/window\_light/set and payload b'{"state":  
↪ "off"}'

Sending message with topic zigbee\_gfw/gfw/marion/window\_light and payload {"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/marion/window\_light and payload b'{"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/marion/window\_light/state and payload b'false'

**Success** Value for ViDevCommon.state (gfw.marion.main\_light) is correct (Content False and Type is <class 'bool'>).

Result (Value for ViDevCommon.state (gfw.marion.main\_light)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (gfw.marion.main\_light)): result = False (<class  
↪ 'bool'>)

#### A.1.97 ViDevCommon.state (gfw.marion.window\_light) → Light.state (gfw.marion.window\_light)

##### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Setting devices to last state False

Sending message with topic videv/gfw/marion/window\_light/state/set and payload false

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

**Info** Setting state of ViDevCommon.state (gfw.marion.window\_light) to True

Sending message with topic videv/gfw/marion/window\_light/state/set and payload true

Received message with topic zigbee\_gfw/gfw/marion/window\_light/set and payload b'{"state":  
↪ "on"}'

Sending message with topic zigbee\_gfw/gfw/marion/window\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/marion/window\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/marion/window\_light/state and payload b'true'

**Success** Value for Light.state (gfw.marion.window\_light) is correct (Content True and Type is <class 'bool'>).

Result (Value for Light.state (gfw.marion.window\_light)): True (<class 'bool'>)

Expectation (Value for Light.state (gfw.marion.window\_light)): result = True (<class 'bool'>)

---

**Info** Setting state of ViDevCommon.state (gfw.marion.window\_light) to False

---

Sending message with topic videv/gfw/marion/window\_light/state/set and payload false

Received message with topic zigbee\_gfw/gfw/marion/window\_light/set and payload b'{"state":  
↪ "off"}'

Sending message with topic zigbee\_gfw/gfw/marion/window\_light and payload {"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/marion/window\_light and payload b'{"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/marion/window\_light/state and payload b'false'

---

**Success** Value for Light.state (gfw.marion.window\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Light.state (gfw.marion.window\_light)): False (<class 'bool'>)

Expectation (Value for Light.state (gfw.marion.window\_light)): result = False (<class 'bool'>)

---

#### A.1.98 Light.state (gfw.marion.window\_light) → ViDevCommon.state (gfw.marion.window\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/gfw/marion/window\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Light.state (gfw.marion.window\_light) to True

---

Sending message with topic zigbee\_gfw/gfw/marion/window\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/marion/window\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/marion/window\_light/state and payload b'true'

---

**Success** Value for ViDevCommon.state (gfw.marion.window\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (gfw.marion.window\_light)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (gfw.marion.window\_light)): result = True (<class  
↪ 'bool'>)

---

---

**Info** Setting state of Light.state (gfw.marion.window\_light) to False

---

Sending message with topic zigbee\_gfw/gfw/marion/window\_light and payload {"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/marion/window\_light and payload b'{"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/marion/window\_light/state and payload b'false'

---

**Success** Value for ViDevCommon.state (gfw.marion.window\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (gfw.marion.window\_light)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (gfw.marion.window\_light)): result = False (<class  
↪ 'bool'>)

---

#### A.1.99 Shelly.relay/0 (gfw.marion.main\_light) → Light.state (gfw.marion.window\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Shelly.relay/0 (gfw.marion.main\_light) to True

---

Sending message with topic shellies/gfw/marion/main\_light/relay/0 and payload on

Received message with topic shellies/gfw/marion/main\_light/relay/0 and payload b'on'

Received message with topic videv/gfw/marion/main\_light/state and payload b'true'

Received message with topic zigbee\_gfw/gfw/marion/window\_light/set and payload b'{"state":  
↪ "on"}'

Sending message with topic zigbee\_gfw/gfw/marion/window\_light and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/marion/window\_light and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/marion/window\_light/state and payload b'true'

---

**Success** Value for Light.state (gfw.marion.window\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Light.state (gfw.marion.window\_light)): True (<class 'bool'>)

---

Expectation (Value for Light.state (gfw.marion.window\_light)): result = True (<class 'bool'>)

---

**Info** Setting state of Shelly.relay/0 (gfw.marion.main\_light) to False

---

Sending message with topic shellies/gfw/marion/main\_light/relay/0 and payload off

Received message with topic shellies/gfw/marion/main\_light/relay/0 and payload b'off'

Received message with topic videv/gfw/marion/main\_light/state and payload b'false'

Received message with topic zigbee\_gfw/gfw/marion/window\_light/set and payload b'{"state":  
↪ "off"}'

Sending message with topic zigbee\_gfw/gfw/marion/window\_light and payload {"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/marion/window\_light and payload b'{"state": "off",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/marion/window\_light/state and payload b'false'

---

**Success** Value for Light.state (gfw.marion.window\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Light.state (gfw.marion.window\_light)): False (<class 'bool'>)

Expectation (Value for Light.state (gfw.marion.window\_light)): result = False (<class 'bool'>)

**A.1.100 ViDevHeating.temp\_setp (gfw.marion.heating\_valve) → HeatingValve.temp\_setp (gfw.marion.heating\_valve)**

### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state 30

---

Sending message with topic videv/gfw/marion/heating\_valve/user\_temperature\_setpoint/set and  
↪ payload 30

Sending message with topic zigbee\_gfw/gfw/marion/heating\_valve and payload  
↪ {"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}

Received message with topic zigbee\_gfw/gfw/marion/heating\_valve/set and payload  
↪ b'{"current\_heating\_setpoint": 30}'

Received message with topic videv/gfw/marion/heating\_valve/valve\_temperature\_setpoint and  
↪ payload b'30'

Received message with topic videv/gfw/marion/heating\_valve/user\_temperature\_setpoint and  
↪ payload b'30'

Received message with topic zigbee\_gfw/gfw/marion/heating\_valve and payload  
↪ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

---

**Success** Start state (master, slave) is correct (Content (30, 30) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (30, 30) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (30, 30) (<class 'tuple'>)

---

**Info**    Setting state of ViDevHeating.temp\_setp (gfw.marion.heating\_valve) to 15

---

Sending message with topic videv/gfw/marion/heating\_valve/user\_temperature\_setpoint/set and  
 ↪ payload 15

Received message with topic zigbee\_gfw/gfw/marion/heating\_valve/set and payload  
 ↪ b'{"current\_heating\_setpoint": 15}'

Sending message with topic zigbee\_gfw/gfw/marion/heating\_valve and payload  
 ↪ {"current\_heating\_setpoint": 15, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/gfw/marion/heating\_valve/valve\_temperature\_setpoint and  
 ↪ payload b'15'

Received message with topic videv/gfw/marion/heating\_valve/user\_temperature\_setpoint and  
 ↪ payload b'15'

Received message with topic zigbee\_gfw/gfw/marion/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 15, "local\_temperature": 20.7, "battery": 97}'

---

**Success**    Value for HeatingValve.temp\_setp (gfw.marion.heating\_valve) is correct (Content 15 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (gfw.marion.heating\_valve)): 15 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (gfw.marion.heating\_valve)): result = 15 (<class  
 ↪ 'int'>)

---

**Info**    Setting state of ViDevHeating.temp\_setp (gfw.marion.heating\_valve) to 20

---

Sending message with topic videv/gfw/marion/heating\_valve/user\_temperature\_setpoint/set and  
 ↪ payload 20

Received message with topic zigbee\_gfw/gfw/marion/heating\_valve/set and payload  
 ↪ b'{"current\_heating\_setpoint": 20}'

Sending message with topic zigbee\_gfw/gfw/marion/heating\_valve and payload  
 ↪ {"current\_heating\_setpoint": 20, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/gfw/marion/heating\_valve/valve\_temperature\_setpoint and  
 ↪ payload b'20'

Received message with topic videv/gfw/marion/heating\_valve/user\_temperature\_setpoint and  
 ↪ payload b'20'

Received message with topic zigbee\_gfw/gfw/marion/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 20, "local\_temperature": 20.7, "battery": 97}'

---

**Success**    Value for HeatingValve.temp\_setp (gfw.marion.heating\_valve) is correct (Content 20 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (gfw.marion.heating\_valve)): 20 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (gfw.marion.heating\_valve)): result = 20 (<class  
 ↪ 'int'>)

---

**Info**    Setting state of ViDevHeating.temp\_setp (gfw.marion.heating\_valve) to 25

---

Sending message with topic videv/gfw/marion/heating\_valve/user\_temperature\_setpoint/set and  
 ↪ payload 25

Received message with topic zigbee\_gfw/gfw/marion/heating\_valve/set and payload  
 ↪ b'{"current\_heating\_setpoint": 25}'

Sending message with topic zigbee\_gfw/gfw/marion/heating\_valve and payload  
 ↪ {"current\_heating\_setpoint": 25, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/gfw/marion/heating\_valve/valve\_temperature\_setpoint and  
 ↪ payload b'25'

Received message with topic videv/gfw/marion/heating\_valve/user\_temperature\_setpoint and  
 ↪ payload b'25'

Received message with topic zigbee\_gfw/gfw/marion/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 25, "local\_temperature": 20.7, "battery": 97}'

---

**Success**    Value for HeatingValve.temp\_setp (gfw.marion.heating\_valve) is correct (Content 25 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (gfw.marion.heating\_valve)): 25 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (gfw.marion.heating\_valve)): result = 25 (<class  
 ↪ 'int'>)

---

**Info**    Setting state of ViDevHeating.temp\_setp (gfw.marion.heating\_valve) to 30

---

Sending message with topic videv/gfw/marion/heating\_valve/user\_temperature\_setpoint/set and  
 ↪ payload 30

Received message with topic zigbee\_gfw/gfw/marion/heating\_valve/set and payload  
 ↪ b'{"current\_heating\_setpoint": 30}'

Sending message with topic zigbee\_gfw/gfw/marion/heating\_valve and payload  
 ↪ {"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}

Received message with topic videv/gfw/marion/heating\_valve/valve\_temperature\_setpoint and  
 ↪ payload b'30'

Received message with topic videv/gfw/marion/heating\_valve/user\_temperature\_setpoint and  
 ↪ payload b'30'

Received message with topic zigbee\_gfw/gfw/marion/heating\_valve and payload  
 ↪ b'{"current\_heating\_setpoint": 30, "local\_temperature": 20.7, "battery": 97}'

---

**Success**    Value for HeatingValve.temp\_setp (gfw.marion.heating\_valve) is correct (Content 30 and Type is <class 'int'>).

---

Result (Value for HeatingValve.temp\_setp (gfw.marion.heating\_valve)): 30 (<class 'int'>)

Expectation (Value for HeatingValve.temp\_setp (gfw.marion.heating\_valve)): result = 30 (<class  
 ↪ 'int'>)

**A.1.101 ViDevCommon.state (gfw.floor.main\_light) → Shelly.relay/0 (gfw.floor.main\_light)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/gfw/floor/main\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (gfw.floor.main\_light) to True

---

Sending message with topic videv/gfw/floor/main\_light/state/set and payload true

Received message with topic shellies/gfw/floor/main\_light/relay/0/command and payload b'on'

Sending message with topic shellies/gfw/floor/main\_light/relay/0 and payload on

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic shellies/gfw/floor/main\_light/relay/0 and payload b'on'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1/get and payload b'{"state": ""}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2/get and payload b'{"state": ""}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic videv/gfw/floor/main\_light/state and payload b'true'

Received message with topic videv/gfw/floor/main\_light/brightness and payload b'50'

Received message with topic videv/gfw/floor/main\_light/color\_temp and payload b'5'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

---

**Success** Value for Shelly.relay/0 (gfw.floor.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (gfw.floor.main\_light)): True (<class 'bool'>)

---

```
Expectation (Value for Shelly.relay/0 (gfw.floor.main_light)): result = True (<class 'bool'>)
```

---

**Info** Setting state of ViDevCommon.state (gfw.floor.main\_light) to False

---

```
Sending message with topic videv/gfw/floor/main_light/state/set and payload false
```

```
Received message with topic shellies/gfw/floor/main_light/relay/0/command and payload b'off'
```

```
Sending message with topic shellies/gfw/floor/main_light/relay/0 and payload off
```

```
Received message with topic shellies/gfw/floor/main_light/relay/0 and payload b'off'
```

```
Received message with topic videv/gfw/floor/main_light/state and payload b'false'
```

---

**Success** Value for Shelly.relay/0 (gfw.floor.main\_light) is correct (Content False and Type is <class 'bool'>).

---

```
Result (Value for Shelly.relay/0 (gfw.floor.main_light)): False (<class 'bool'>)
```

```
Expectation (Value for Shelly.relay/0 (gfw.floor.main_light)): result = False (<class 'bool'>)
```

#### A.1.102 Shelly.relay/0 (gfw.floor.main\_light) → ViDevCommon.state (gfw.floor.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

```
Sending message with topic videv/gfw/floor/main_light/state/set and payload false
```

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

```
Result (Start state (master, slave)): (False, False) (<class 'tuple'>)
```

```
Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)
```

---

**Info** Setting state of Shelly.relay/0 (gfw.floor.main\_light) to True

---

```
Sending message with topic shellies/gfw/floor/main_light/relay/0 and payload on
```

```
Sending message with topic zigbee_gfw/gfw/floor/main_light_1 and payload {"state": "on",  
↪ "brightness": 127.0, "color_temp": 352.0}
```

```
Sending message with topic zigbee_gfw/gfw/floor/main_light_2 and payload {"state": "on",  
↪ "brightness": 127.0, "color_temp": 352.0}
```

```
Received message with topic shellies/gfw/floor/main_light/relay/0 and payload b'on'
```

```
Received message with topic zigbee_gfw/gfw/floor/main_light_1 and payload b'{"state": "on",  
↪ "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_gfw/gfw/floor/main_light_2 and payload b'{"state": "on",  
↪ "brightness": 127.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_gfw/gfw/floor/main_light_1/get and payload b'{"state": ""}'
```

```
Sending message with topic zigbee_gfw/gfw/floor/main_light_1 and payload {"state": "on",  
↪ "brightness": 127.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_gfw/gfw/floor/main_light_2/get and payload b'{"state": ""}'
```

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic videv/gfw/floor/main\_light/state and payload b'true'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

---

**Success** Value for ViDevCommon.state (gfw.floor.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (gfw.floor.main\_light)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (gfw.floor.main\_light)): result = True (<class  
↪ 'bool'>)

---

**Info** Setting state of Shelly.relay/0 (gfw.floor.main\_light) to False

---

Sending message with topic shellies/gfw/floor/main\_light/relay/0 and payload off

Received message with topic shellies/gfw/floor/main\_light/relay/0 and payload b'off'

Received message with topic videv/gfw/floor/main\_light/state and payload b'false'

---

**Success** Value for ViDevCommon.state (gfw.floor.main\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (gfw.floor.main\_light)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (gfw.floor.main\_light)): result = False (<class  
↪ 'bool'>)

### A.1.103 ViDevCommon.brightness (gfw.floor.main\_light) → Light.brightness (gfw.floor.main\_light)

#### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---

Sending message with topic shellies/gfw/floor/main\_light/relay/0 and payload on

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}

Received message with topic shellies/gfw/floor/main\_light/relay/0 and payload b'on'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 127.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1/get and payload b'{"state": ""}'

```

Sending message with topic zigbee_gfw/gfw/floor/main_light_1 and payload {"state": "on",
↪  "brightness": 127.0, "color_temp": 352.0}
Received message with topic zigbee_gfw/gfw/floor/main_light_2/get and payload b'{"state": ""}'
Sending message with topic zigbee_gfw/gfw/floor/main_light_2 and payload {"state": "on",
↪  "brightness": 127.0, "color_temp": 352.0}
Received message with topic videv/gfw/floor/main_light/state and payload b'true'
Received message with topic zigbee_gfw/gfw/floor/main_light_1 and payload b'{"state": "on",
↪  "brightness": 127.0, "color_temp": 352.0}'
Received message with topic zigbee_gfw/gfw/floor/main_light_2 and payload b'{"state": "on",
↪  "brightness": 127.0, "color_temp": 352.0}'

```

---

**Info** Prepare: Setting devices to last state 100

---

```

Sending message with topic videv/gfw/floor/main_light/brightness/set and payload 100
Sending message with topic zigbee_gfw/gfw/floor/main_light_1 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 352.0}
Sending message with topic zigbee_gfw/gfw/floor/main_light_2 and payload {"state": "on",
↪  "brightness": 254.0, "color_temp": 352.0}
Received message with topic zigbee_gfw/gfw/floor/main_light_1/set and payload b'{"brightness":
↪  254}'
Received message with topic zigbee_gfw/gfw/floor/main_light_2/set and payload b'{"brightness":
↪  254}'
Received message with topic zigbee_gfw/gfw/floor/main_light_1 and payload b'{"state": "on",
↪  "brightness": 254.0, "color_temp": 352.0}'
Received message with topic zigbee_gfw/gfw/floor/main_light_2 and payload b'{"state": "on",
↪  "brightness": 254.0, "color_temp": 352.0}'
Received message with topic videv/gfw/floor/main_light/brightness and payload b'100'

```

---

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

---

```

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)
Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

```

---

**Info** Setting state of ViDevCommon.brightness (gfw.floor.main\_light) to 0

---

```

Sending message with topic videv/gfw/floor/main_light/brightness/set and payload 0
Received message with topic zigbee_gfw/gfw/floor/main_light_1/set and payload b'{"brightness":
↪  1}'
Sending message with topic zigbee_gfw/gfw/floor/main_light_1 and payload {"state": "on",
↪  "brightness": 1.0, "color_temp": 352.0}
Received message with topic zigbee_gfw/gfw/floor/main_light_2/set and payload b'{"brightness":
↪  1}'
Sending message with topic zigbee_gfw/gfw/floor/main_light_2 and payload {"state": "on",
↪  "brightness": 1.0, "color_temp": 352.0}

```

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/floor/main\_light/brightness and payload b'0'

**Success** Value for Light.brightness (gfw.floor.main\_light) is correct (Content 0 and Type is <class 'int'>).

Result (Value for Light.brightness (gfw.floor.main\_light)): 0 (<class 'int'>)

Expectation (Value for Light.brightness (gfw.floor.main\_light)): result = 0 (<class 'int'>)

**Info** Setting state of ViDevCommon.brightness (gfw.floor.main\_light) to 20

Sending message with topic videv/gfw/floor/main\_light/brightness/set and payload 20

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1/set and payload b'{"brightness":  
↪ 52}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2/set and payload b'{"brightness":  
↪ 52}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/floor/main\_light/brightness and payload b'20'

**Success** Value for Light.brightness (gfw.floor.main\_light) is correct (Content 20 and Type is <class 'int'>).

Result (Value for Light.brightness (gfw.floor.main\_light)): 20 (<class 'int'>)

Expectation (Value for Light.brightness (gfw.floor.main\_light)): result = 20 (<class 'int'>)

**Info** Setting state of ViDevCommon.brightness (gfw.floor.main\_light) to 40

Sending message with topic videv/gfw/floor/main\_light/brightness/set and payload 40

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1/set and payload b'{"brightness":  
↪ 102}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2/set and payload b'{"brightness":  
↪ 102}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/floor/main\_light/brightness and payload b'40'

---

**Success** Value for Light.brightness (gfw.floor.main\_light) is correct (Content 40 and Type is <class 'int'>).

---

Result (Value for Light.brightness (gfw.floor.main\_light)): 40 (<class 'int'>)

Expectation (Value for Light.brightness (gfw.floor.main\_light)): result = 40 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (gfw.floor.main\_light) to 60

---

Sending message with topic videv/gfw/floor/main\_light/brightness/set and payload 60

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1/set and payload b'{"brightness":  
↪ 153}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2/set and payload b'{"brightness":  
↪ 153}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/floor/main\_light/brightness and payload b'60'

---

**Success** Value for Light.brightness (gfw.floor.main\_light) is correct (Content 60 and Type is <class 'int'>).

---

Result (Value for Light.brightness (gfw.floor.main\_light)): 60 (<class 'int'>)

Expectation (Value for Light.brightness (gfw.floor.main\_light)): result = 60 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.brightness (gfw.floor.main\_light) to 80

---

Sending message with topic videv/gfw/floor/main\_light/brightness/set and payload 80

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1/set and payload b'{"brightness":  
↪ 203}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2/set and payload b'{"brightness":  
↪ 203}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/floor/main\_light/brightness and payload b'80'

**Success** Value for Light.brightness (gfw.floor.main\_light) is correct (Content 80 and Type is <class 'int'>).

Result (Value for Light.brightness (gfw.floor.main\_light)): 80 (<class 'int'>)

Expectation (Value for Light.brightness (gfw.floor.main\_light)): result = 80 (<class 'int'>)

**Info** Setting state of ViDevCommon.brightness (gfw.floor.main\_light) to 100

Sending message with topic videv/gfw/floor/main\_light/brightness/set and payload 100

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1/set and payload b'{"brightness":  
↪ 254}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2/set and payload b'{"brightness":  
↪ 254}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/floor/main\_light/brightness and payload b'100'

**Success** Value for Light.brightness (gfw.floor.main\_light) is correct (Content 100 and Type is <class 'int'>).

Result (Value for Light.brightness (gfw.floor.main\_light)): 100 (<class 'int'>)

Expectation (Value for Light.brightness (gfw.floor.main\_light)): result = 100 (<class 'int'>)

#### A.1.104 Light.brightness (gfw.floor.main\_light) → ViDevCommon.brightness (gfw.floor.main\_light)

##### Testresult

This test was passed with the state: **Success**.

**Info** Prepare: Switching on device

**Info** Prepare: Setting devices to last state 100

Sending message with topic videv/gfw/floor/main\_light/brightness/set and payload 100

**Success** Start state (master, slave) is correct (Content (100, 100) and Type is <class 'tuple'>).

Result (Start state (master, slave)): (100, 100) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (100, 100) (<class 'tuple'>)

---

**Info** Setting state of Light.brightness (gfw.floor.main\_light) to 0

---

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 1.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/floor/main\_light/brightness and payload b'0'

---

**Success** Value for ViDevCommon.brightness (gfw.floor.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (gfw.floor.main\_light)): 0 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (gfw.floor.main\_light)): result = 0 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (gfw.floor.main\_light) to 20

---

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 52.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/floor/main\_light/brightness and payload b'20'

---

**Success** Value for ViDevCommon.brightness (gfw.floor.main\_light) is correct (Content 20 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.brightness (gfw.floor.main\_light)): 20 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (gfw.floor.main\_light)): result = 20 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.brightness (gfw.floor.main\_light) to 40

---

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 102.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/floor/main\_light/brightness and payload b'40'

**Success** Value for ViDevCommon.brightness (gfw.floor.main\_light) is correct (Content 40 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (gfw.floor.main\_light)): 40 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (gfw.floor.main\_light)): result = 40 (<class  
↪ 'int'>)

**Info** Setting state of Light.brightness (gfw.floor.main\_light) to 60

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 153.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/floor/main\_light/brightness and payload b'60'

**Success** Value for ViDevCommon.brightness (gfw.floor.main\_light) is correct (Content 60 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (gfw.floor.main\_light)): 60 (<class 'int'>)

Expectation (Value for ViDevCommon.brightness (gfw.floor.main\_light)): result = 60 (<class  
↪ 'int'>)

**Info** Setting state of Light.brightness (gfw.floor.main\_light) to 80

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 203.0, "color\_temp": 352.0}'

Received message with topic videv/gfw/floor/main\_light/brightness and payload b'80'

**Success** Value for ViDevCommon.brightness (gfw.floor.main\_light) is correct (Content 80 and Type is <class 'int'>).

Result (Value for ViDevCommon.brightness (gfw.floor.main\_light)): 80 (<class 'int'>)

```
Expectation (Value for ViDevCommon.brightness (gfw.floor.main_light)): result = 80 (<class
↳ 'int'>)
```

---

**Info** Setting state of Light.brightness (gfw.floor.main\_light) to 100

---

```
Sending message with topic zigbee_gfw/gfw/floor/main_light_1 and payload {"state": "on",
↳ "brightness": 254.0, "color_temp": 352.0}
```

```
Sending message with topic zigbee_gfw/gfw/floor/main_light_2 and payload {"state": "on",
↳ "brightness": 254.0, "color_temp": 352.0}
```

```
Received message with topic zigbee_gfw/gfw/floor/main_light_1 and payload b'{"state": "on",
↳ "brightness": 254.0, "color_temp": 352.0}'
```

```
Received message with topic zigbee_gfw/gfw/floor/main_light_2 and payload b'{"state": "on",
↳ "brightness": 254.0, "color_temp": 352.0}'
```

```
Received message with topic videv/gfw/floor/main_light/brightness and payload b'100'
```

---

**Success** Value for ViDevCommon.brightness (gfw.floor.main\_light) is correct (Content 100 and Type is <class 'int'>).

---

```
Result (Value for ViDevCommon.brightness (gfw.floor.main_light)): 100 (<class 'int'>)
```

```
Expectation (Value for ViDevCommon.brightness (gfw.floor.main_light)): result = 100 (<class
↳ 'int'>)
```

#### A.1.105 ViDevCommon.color\_temp (gfw.floor.main\_light) → Light.color\_temp (gfw.floor.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 10

---

```
Sending message with topic videv/gfw/floor/main_light/color_temp/set and payload 10
```

```
Sending message with topic zigbee_gfw/gfw/floor/main_light_1 and payload {"state": "on",
↳ "brightness": 254.0, "color_temp": 454.0}
```

```
Sending message with topic zigbee_gfw/gfw/floor/main_light_2 and payload {"state": "on",
↳ "brightness": 254.0, "color_temp": 454.0}
```

```
Received message with topic zigbee_gfw/gfw/floor/main_light_1/set and payload b'{"color_temp":
↳ 454}'
```

```
Received message with topic zigbee_gfw/gfw/floor/main_light_2/set and payload b'{"color_temp":
↳ 454}'
```

```
Received message with topic zigbee_gfw/gfw/floor/main_light_1 and payload b'{"state": "on",
↳ "brightness": 254.0, "color_temp": 454.0}'
```

```
Received message with topic zigbee_gfw/gfw/floor/main_light_2 and payload b'{"state": "on",
↳ "brightness": 254.0, "color_temp": 454.0}'
```

```
Received message with topic videv/gfw/floor/main_light/color_temp and payload b'10'
```

---

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (10, 10) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (10, 10) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.floor.main\_light) to 0

---

Sending message with topic videv/gfw/floor/main\_light/color\_temp/set and payload 0

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1/set and payload b'{"color\_temp":  
↪ 250}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2/set and payload b'{"color\_temp":  
↪ 250}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic videv/gfw/floor/main\_light/color\_temp and payload b'0'

---

**Success** Value for Light.color\_temp (gfw.floor.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (gfw.floor.main\_light)): 0 (<class 'int'>)

Expectation (Value for Light.color\_temp (gfw.floor.main\_light)): result = 0 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.floor.main\_light) to 2

---

Sending message with topic videv/gfw/floor/main\_light/color\_temp/set and payload 2

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1/set and payload b'{"color\_temp":  
↪ 291}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2/set and payload b'{"color\_temp":  
↪ 291}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic videv/gfw/floor/main\_light/color\_temp and payload b'2'

---

---

**Success** Value for Light.color\_temp (gfw.floor.main\_light) is correct (Content 2 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (gfw.floor.main\_light)): 2 (<class 'int'>)

Expectation (Value for Light.color\_temp (gfw.floor.main\_light)): result = 2 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.floor.main\_light) to 4

---

Sending message with topic videv/gfw/floor/main\_light/color\_temp/set and payload 4

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1/set and payload b'{"color\_temp":  
↪ 332}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2/set and payload b'{"color\_temp":  
↪ 332}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic videv/gfw/floor/main\_light/color\_temp and payload b'4'

---

**Success** Value for Light.color\_temp (gfw.floor.main\_light) is correct (Content 4 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (gfw.floor.main\_light)): 4 (<class 'int'>)

Expectation (Value for Light.color\_temp (gfw.floor.main\_light)): result = 4 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.floor.main\_light) to 6

---

Sending message with topic videv/gfw/floor/main\_light/color\_temp/set and payload 6

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1/set and payload b'{"color\_temp":  
↪ 372}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2/set and payload b'{"color\_temp":  
↪ 372}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic videv/gfw/floor/main\_light/color\_temp and payload b'6'

---

---

**Success** Value for Light.color\_temp (gfw.floor.main\_light) is correct (Content 6 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (gfw.floor.main\_light)): 6 (<class 'int'>)

Expectation (Value for Light.color\_temp (gfw.floor.main\_light)): result = 6 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.floor.main\_light) to 8

---

Sending message with topic videv/gfw/floor/main\_light/color\_temp/set and payload 8

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1/set and payload b'{"color\_temp": 413}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on", "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2/set and payload b'{"color\_temp": 413}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on", "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on", "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on", "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic videv/gfw/floor/main\_light/color\_temp and payload b'8'

---

**Success** Value for Light.color\_temp (gfw.floor.main\_light) is correct (Content 8 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (gfw.floor.main\_light)): 8 (<class 'int'>)

Expectation (Value for Light.color\_temp (gfw.floor.main\_light)): result = 8 (<class 'int'>)

---

**Info** Setting state of ViDevCommon.color\_temp (gfw.floor.main\_light) to 10

---

Sending message with topic videv/gfw/floor/main\_light/color\_temp/set and payload 10

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1/set and payload b'{"color\_temp": 454}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on", "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2/set and payload b'{"color\_temp": 454}'

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on", "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on", "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on", "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/gfw/floor/main\_light/color\_temp and payload b'10'

---

---

**Success** Value for Light.color\_temp (gfw.floor.main\_light) is correct (Content 10 and Type is <class 'int'>).

---

Result (Value for Light.color\_temp (gfw.floor.main\_light)): 10 (<class 'int'>)

Expectation (Value for Light.color\_temp (gfw.floor.main\_light)): result = 10 (<class 'int'>)

---

#### A.1.106 Light.color\_temp (gfw.floor.main\_light) → ViDevCommon.color\_temp (gfw.floor.main\_light)

##### Testresult

This test was passed with the state: **Success**.

---

**Info** Prepare: Switching on device

---



---

**Info** Prepare: Setting devices to last state 10

---

Sending message with topic videv/gfw/floor/main\_light/color\_temp/set and payload 10

---

**Success** Start state (master, slave) is correct (Content (10, 10) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (10, 10) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (10, 10) (<class 'tuple'>)

---

**Info** Setting state of Light.color\_temp (gfw.floor.main\_light) to 0

---

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 250.0}'

Received message with topic videv/gfw/floor/main\_light/color\_temp and payload b'0'

---

**Success** Value for ViDevCommon.color\_temp (gfw.floor.main\_light) is correct (Content 0 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.floor.main\_light)): 0 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.floor.main\_light)): result = 0 (<class  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (gfw.floor.main\_light) to 2

---

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 291.0}'

Received message with topic videv/gfw/floor/main\_light/color\_temp and payload b'2'

---

**Success** Value for ViDevCommon.color\_temp (gfw.floor.main\_light) is correct (Content 2 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.floor.main\_light)): 2 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.floor.main\_light)): result = 2 (<class 'int'>)  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (gfw.floor.main\_light) to 4

---

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 332.0}'

Received message with topic videv/gfw/floor/main\_light/color\_temp and payload b'4'

---

**Success** Value for ViDevCommon.color\_temp (gfw.floor.main\_light) is correct (Content 4 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.floor.main\_light)): 4 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.floor.main\_light)): result = 4 (<class 'int'>)  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (gfw.floor.main\_light) to 6

---

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 372.0}'

Received message with topic videv/gfw/floor/main\_light/color\_temp and payload b'6'

---

**Success** Value for ViDevCommon.color\_temp (gfw.floor.main\_light) is correct (Content 6 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.floor.main\_light)): 6 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.floor.main\_light)): result = 6 (<class 'int'>)  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (gfw.floor.main\_light) to 8

---

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 413.0}'

Received message with topic videv/gfw/floor/main\_light/color\_temp and payload b'8'

---

**Success** Value for ViDevCommon.color\_temp (gfw.floor.main\_light) is correct (Content 8 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.floor.main\_light)): 8 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.floor.main\_light)): result = 8 (<class 'int'>)  
↪ 'int'>)

---

**Info** Setting state of Light.color\_temp (gfw.floor.main\_light) to 10

---

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Sending message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload {"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_1 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic zigbee\_gfw/gfw/floor/main\_light\_2 and payload b'{"state": "on",  
↪ "brightness": 254.0, "color\_temp": 454.0}'

Received message with topic videv/gfw/floor/main\_light/color\_temp and payload b'10'

---

**Success** Value for ViDevCommon.color\_temp (gfw.floor.main\_light) is correct (Content 10 and Type is <class 'int'>).

---

Result (Value for ViDevCommon.color\_temp (gfw.floor.main\_light)): 10 (<class 'int'>)

Expectation (Value for ViDevCommon.color\_temp (gfw.floor.main\_light)): result = 10 (<class 'int'>)  
↪ 'int'>)

**A.1.107 ViDevCommon.state (stw.stairway.main\_light) → Shelly.relay/0 (stw.firstfloor.main\_light)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/stw/stairway/main\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of ViDevCommon.state (stw.stairway.main\_light) to True

---

Sending message with topic videv/stw/stairway/main\_light/state/set and payload true

Received message with topic shellies/stw/firstfloor/main\_light/relay/0/command and payload  
↔ b'on'

Sending message with topic shellies/stw/firstfloor/main\_light/relay/0 and payload on

Received message with topic shellies/stw/firstfloor/main\_light/relay/0 and payload b'on'

Received message with topic videv/stw/stairway/main\_light/state and payload b'true'

Received message with topic videv/stw/stairway/main\_light/timer and payload b'100'

---

**Success** Value for Shelly.relay/0 (stw.firstfloor.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (stw.firstfloor.main\_light)): True (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (stw.firstfloor.main\_light)): result = True (<class  
↔ 'bool'>)

---

**Info** Setting state of ViDevCommon.state (stw.stairway.main\_light) to False

---

Sending message with topic videv/stw/stairway/main\_light/state/set and payload false

Received message with topic shellies/stw/firstfloor/main\_light/relay/0/command and payload  
↔ b'off'

Sending message with topic shellies/stw/firstfloor/main\_light/relay/0 and payload off

Received message with topic shellies/stw/firstfloor/main\_light/relay/0 and payload b'off'

Received message with topic videv/stw/stairway/main\_light/timer and payload b'99'

Received message with topic videv/stw/stairway/main\_light/state and payload b'false'

Received message with topic shellies/stw/firstfloor/main\_light/relay/0/command and payload  
↔ b'off'

Received message with topic videv/stw/stairway/main\_light/timer and payload b'0'

---

**Success** Value for Shelly.relay/0 (stw.firstfloor.main\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for Shelly.relay/0 (stw.firstfloor.main\_light)): False (<class 'bool'>)

Expectation (Value for Shelly.relay/0 (stw.firstfloor.main\_light)): result = False (<class  
↔ 'bool'>)

---

**A.1.108 Shelly.relay/0 (stw.firstfloor.main\_light) → ViDevCommon.state (stw.stairway.main\_light)****Testresult**

This test was passed with the state: **Success**.

---

**Info** Prepare: Setting devices to last state False

---

Sending message with topic videv/stw/stairway/main\_light/state/set and payload false

---

**Success** Start state (master, slave) is correct (Content (False, False) and Type is <class 'tuple'>).

---

Result (Start state (master, slave)): (False, False) (<class 'tuple'>)

Expectation (Start state (master, slave)): result = (False, False) (<class 'tuple'>)

---

**Info** Setting state of Shelly.relay/0 (stw.firstfloor.main\_light) to True

---

Sending message with topic shellies/stw/firstfloor/main\_light/relay/0 and payload on

Received message with topic shellies/stw/firstfloor/main\_light/relay/0 and payload b'on'

Received message with topic videv/stw/stairway/main\_light/state and payload b'true'

Received message with topic videv/stw/stairway/main\_light/timer and payload b'100'

---

**Success** Value for ViDevCommon.state (stw.stairway.main\_light) is correct (Content True and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (stw.stairway.main\_light)): True (<class 'bool'>)

Expectation (Value for ViDevCommon.state (stw.stairway.main\_light)): result = True (<class 'bool'>)

---

**Info** Setting state of Shelly.relay/0 (stw.firstfloor.main\_light) to False

---

Sending message with topic shellies/stw/firstfloor/main\_light/relay/0 and payload off

Received message with topic shellies/stw/firstfloor/main\_light/relay/0 and payload b'off'

Received message with topic videv/stw/stairway/main\_light/state and payload b'false'

Received message with topic shellies/stw/firstfloor/main\_light/relay/0/command and payload  
↪ b'off'

Received message with topic videv/stw/stairway/main\_light/timer and payload b'0'

---

**Success** Value for ViDevCommon.state (stw.stairway.main\_light) is correct (Content False and Type is <class 'bool'>).

---

Result (Value for ViDevCommon.state (stw.stairway.main\_light)): False (<class 'bool'>)

Expectation (Value for ViDevCommon.state (stw.stairway.main\_light)): result = False (<class 'bool'>)

---

## B Test-Coverage

### B.1 devdi

The line coverage for devdi was 99.5%

The branch coverage for devdi was 85.7%

#### B.1.1 devdi.\_\_init\_\_.py

The line coverage for devdi.\_\_init\_\_.py was 100.0%

The branch coverage for devdi.\_\_init\_\_.py was 85.7%

1

#### B.1.2 devdi.rooms.py

The line coverage for devdi.rooms.py was 99.1%

The branch coverage for devdi.rooms.py was 85.7%

```

1 import config
2 from .topic import get_topic
3 import logging
4 #
5 from devdi import topic as props
6 from mqtt import mqtt_client
7 """
8 In this module we initialise the smartzhome devices for all rooms.
9 These rooms can be used in the different project for smarthome.
10
11 The device names in the room classes follow this definition:
12     switch_main_light
13     light_main_light
14     motion_main_light_xx (xx: gf, ff)
15     videv_main_light
16
17     switch_desk_light
18     light_desk_light
19     videv_desk_light
20
21     switch_floor_light
22     light_floor_light
23     videv_floor_light
24
25     switch_window_light
26     light_window_light
27     videv_window_light
28
29     switch_wardrobe_light
30     light_wardrobe_light
31     videv_wardrobe_light
32
33     switch_bed_dirk_light
34     light_bed_dirk_light
35     videv_bed_dirk_light
36

```

```

37     switch_bed_marion_light
38     light_bed_marion_light
39     videv_bed_marion_light
40
41     switch_window_light
42     light_window_light
43     videv_window_light
44
45     switch_garland_light
46     videv_garland_light
47
48     switch_repeater
49     videv_repeater
50
51     switch_xmas_tree_light
52     videv_xmas_tree_light
53
54     switch_xmas_star_light
55     videv_xmas_star_light
56
57     switch_circulation_pump
58     videv_circulation_pump
59
60     switch_powerplug_4
61     videv_amplifier
62     videv_cd_player
63     videv_bluetooth
64     videv_phono
65
66     switch_pc_dock
67     videv_pc_dock
68
69     remote_ctrl
70     audio_status_spotify
71     audio_status_mpd
72     audio_status_bluetooth
73
74
75     valve_heating
76     ambient_info
77     videv_heating
78
79     videv_multistate
80     videv_mode
81
82     input_device

```

```

85 The following devices are already in use and have to be defined in devices.xxx

```

```

86 """

```

```

87 from devices import group
88
89 from devices import shelly_sw1
90 from devices import hue_sw_br_ct
91 from devices import tradfri_sw
92 from devices import tradfri_sw_br
93 from devices import tradfri_sw_br_ct
94 from devices import tradfri_button
95 from devices import livarno_sw_br_ct
96 from devices import brennenstuhl_heatingvalve
97 from devices import silvercrest_powerplug

```

```

98 from devices import silvercrest_motion_sensor
99 from devices import my_powerplug
100 from devices import audio_status
101 from devices import remote
102 from devices import my_ambient
103 #
104 from devices import videv_sw
105 from devices import videv_sw_br
106 from devices import videv_sw_br_ct
107 from devices import videv_sw_tm
108 from devices import videv_sw_mo
109 from devices import videv_heu
110 from devices import videv_pure_switch
111 from devices import videv_multistate
112 from devices import videv_audio_player
113 #
114 #
115 try:
116     from config import APP_NAME as ROOT_LOGGER_NAME
117 except ImportError:
118     ROOT_LOGGER_NAME = 'root'
119 logger = logging.getLogger(ROOT_LOGGER_NAME).getChild(__name__)
120
121
122 class base_room(object):
123     def __get_group__(self, class_type, mqtt_client, stg, loc, roo, fun, num):
124         dg = []
125         topic = get_topic(stg, loc, roo, fun)
126         for i in range(1, num + 1):
127             device_topic = topic + '%d' % i
128             dg.append(class_type(mqtt_client, device_topic))
129         this_device = group(*dg)
130         return this_device
131
132
133 #
134 # FFE
135 #####
136 #
137 class ffe_floor(base_room):
138     def __init__(self, mqtt_client: mqtt_client):
139         loc = props.LOC_FFE
140         roo = props.ROO_FLO
141         #
142         # http://shelly1l-3C6105E4E629
143         self.switch_main_light = shelly_sw1(mqtt_client, get_topic(props.STG_SHE, loc, roo, props.FUN_MAL))
144         self.videv_main_light = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.FUN_MAL))
145
146
147 class ffe_diningroom(base_room):
148     def __init__(self, mqtt_client: mqtt_client):
149         loc = props.LOC_FFE
150         roo = props.ROO_DIN
151         #
152         # http://shelly1l-84CCA8ADD055
153         self.switch_main_light = shelly_sw1(mqtt_client, get_topic(props.STG_SHE, loc, roo, props.FUN_MAL))
154         self.videv_main_light = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.FUN_MAL))

```

```

155         self.switch_floor_light = silvercrest_powerplug(mqtt_client, get_topic(props.STG_ZFE, loc
, roo, props.FUN_FLL))
156         self.videv_floor_light = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.
FUN_FLL))
157
158         self.valve_heating = brennenstuhl_heatingvalve(mqtt_client, get_topic(props.STG_ZFE, loc,
, roo, props.FUN_HEA))
159         self.videv_heating = videv_heu(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.
FUN_HEA))
160
161         if config.CHRISTMAS:
162             self.switch_garland_light = silvercrest_powerplug(mqtt_client, get_topic(props.
STG_ZFE, loc, roo, props.FUN_GAR))
163             self.videv_garland_light = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo,
, props.FUN_GAR))
164
165
166 class ffe_kitchen(base_room):
167     def __init__(self, mqtt_client: mqtt_client):
168         loc = props.LOC_FFE
169         roo = props.ROO_KIT
170         #
171         # http://shelly1l-8CAAB5616C01
172         self.switch_main_light = shelly_sw1(mqtt_client, get_topic(props.STG_SHE, loc, roo, props
.FUN_MAL))
173         self.light_main_light: hue_sw_br_ct = self.__get_group__(hue_sw_br_ct, mqtt_client, props
.STG_ZFE, loc, roo, props.FUN_MAL, 2)
174         self.videv_main_light = videv_sw_br_ct(mqtt_client, get_topic(props.STG_VDE, loc, roo,
, props.FUN_MAL))
175
176         # http://shelly1-e89f6d85a466
177         self.switch_circulation_pump = shelly_sw1(mqtt_client, get_topic(props.STG_SHE, loc, roo,
, props.FUN_CIR))
178         self.videv_circulation_pump = videv_sw_tm(mqtt_client, get_topic(props.STG_VDE, loc, roo,
, props.FUN_CIR))
179
180         self.valve_heating = brennenstuhl_heatingvalve(mqtt_client, get_topic(props.STG_ZFE, loc,
, roo, props.FUN_HEA))
181         self.videv_heating = videv_heu(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.
FUN_HEA))
182
183
184 class ffe_livingroom(base_room):
185     def __init__(self, mqtt_client: mqtt_client):
186         loc = props.LOC_FFE
187         roo = props.ROO_LIV
188         #
189         # http://shelly1l-3C6105E3F910
190         self.switch_main_light = shelly_sw1(mqtt_client, get_topic(props.STG_SHE, loc, roo, props
.FUN_MAL))
191         self.light_main_light = tradfri_sw_br_ct(mqtt_client, get_topic(props.STG_ZFE, loc, roo,
, props.FUN_MAL))
192         self.videv_main_light = videv_sw_br_ct(mqtt_client, get_topic(props.STG_VDE, loc, roo,
, props.FUN_MAL))
193
194         self.light_floor_light: tradfri_sw_br_ct = self.__get_group__(tradfri_sw_br_ct,
mqtt_client, props.STG_ZFE, loc, roo, props.FUN_FLL, 6)
195         self.videv_floor_light = videv_sw_br_ct(mqtt_client, get_topic(props.STG_VDE, loc, roo,
, props.FUN_FLL))
196

```

## Unittest for smart\_brain

```

197     self.valve_heating = brennenstuhl_heatingvalve(mqtt_client, get_topic(props.STG_ZFE, loc,
198         roo, props.FUN_HEA))
199     self.videv_heating = videv_heating(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.
200         FUN_HEA))
201
202     self.ambient_info = my_ambient(mqtt_client, get_topic(props.STG_MYA, loc, roo, props.
203         FUN_AMB))
204
205     if config.CHRISTMAS:
206         self.switch_xmas_tree_light = silvercrest_powerplug(mqtt_client, get_topic(props.
207             STG_ZFE, loc, roo, props.FUN_XTR))
208         self.videv_xmas_tree_light = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo,
209             props.FUN_XTR))
210
211         self.switch_xmas_star_light = silvercrest_powerplug(mqtt_client, get_topic(props.
212             STG_ZFE, loc, roo, props.FUN_XST))
213         self.videv_xmas_star_light = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo,
214             props.FUN_XST))
215
216 class ffe_sleep(base_room):
217     def __init__(self, mqtt_client: mqtt_client):
218         loc = props.LOC_FFE
219         roo = props.ROO_SLP
220         #
221         # http://shelly1l-E8DB84A254C7
222         self.switch_main_light = shelly_sw1(mqtt_client, get_topic(props.STG_SHE, loc, roo, props.
223             FUN_MAL))
224         self.light_main_light = tradfri_sw_br_ct(mqtt_client, get_topic(props.STG_ZFE, loc, roo,
225             props.FUN_MAL))
226         self.videv_main_light = videv_sw_br_ct(mqtt_client, get_topic(props.STG_VDE, loc, roo,
227             props.FUN_MAL))
228
229         self.input_device = tradfri_button(mqtt_client, get_topic(props.STG_ZFE, loc, roo, props.
230             FUN_INP))
231
232         self.light_bed_dirk_light = tradfri_sw_br(mqtt_client, get_topic(props.STG_ZFE, loc, roo,
233             props.FUN_BLD))
234         self.videv_bed_dirk_light = videv_sw_br(mqtt_client, get_topic(props.STG_VDE, loc, roo,
235             props.FUN_BLD))
236
237         self.switch_bed_marion_light = silvercrest_powerplug(mqtt_client, get_topic(props.STG_ZFE
238             , loc, roo, props.FUN_BLM))
239         self.videv_bed_marion_light = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo,
240             props.FUN_BLM))
241
242         self.light_wardrobe_light = tradfri_sw_br(mqtt_client, get_topic(props.STG_ZFE, loc, roo,
243             props.FUN_WLI))
244         self.videv_wardrobe_light = videv_sw_br(mqtt_client, get_topic(props.STG_VDE, loc, roo,
245             props.FUN_WLI))
246
247         self.valve_heating = brennenstuhl_heatingvalve(mqtt_client, get_topic(props.STG_ZFE, loc,
248             roo, props.FUN_HEA))
249         self.videv_heating = videv_heating(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.
250             FUN_HEA))
251
252         self.videv_multistate = videv_multistate(mqtt_client, get_topic(props.STG_VDE, loc, roo,
253             props.FUN_VMS))
254
255     #
256     # FFW
257     #####
258     #

```

```

240 class ffw_bath(base_room):
241     def __init__(self, mqtt_client: mqtt_client):
242         loc = props.LOC_FFW
243         roo = props.ROO_BAT
244         #
245         # http://shelly1-58BF25D84219
246         self.switch_main_light = shelly_sw1(mqtt_client, get_topic(props.STG_SHE, loc, roo, props
.FUN_MAL))
247         self.videv_main_light = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo, props
.FUN_MAL))
248
249         self.valve_heating = brennenstuhl_heatingvalve(mqtt_client, get_topic(props.STG_ZFW, loc,
roo, props.FUN_HEA))
250         self.videv_heating = videv_he(mqtt_client, get_topic(props.STG_VDE, loc, roo, props
.FUN_HEA))
251
252
253 class ffw_floor(base_room):
254     def __init__(self, mqtt_client: mqtt_client):
255         loc = props.LOC_FFW
256         roo = props.ROO_FLO
257         #
258         # http://shelly1-58BF25D848EA
259         self.switch_main_light = shelly_sw1(mqtt_client, get_topic(props.STG_SHE, loc, roo, props
.FUN_MAL))
260         self.videv_main_light = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo, props
.FUN_MAL))
261
262
263 class ffw_julian(base_room):
264     def __init__(self, mqtt_client: mqtt_client):
265         loc = props.LOC_FFW
266         roo = props.ROO_JUL
267         #
268         # http://shelly1-3C6105E43452
269         self.switch_main_light = shelly_sw1(mqtt_client, get_topic(props.STG_SHE, loc, roo, props
.FUN_MAL))
270         self.light_main_light = tradfri_sw_br_ct(mqtt_client, get_topic(props.STG_ZFW, loc, roo,
props.FUN_MAL))
271         self.videv_main_light = videv_sw_br_ct(mqtt_client, get_topic(props.STG_VDE, loc, roo,
props.FUN_MAL))
272
273         self.valve_heating = brennenstuhl_heatingvalve(mqtt_client, get_topic(props.STG_ZFW, loc,
roo, props.FUN_HEA))
274         self.videv_heating = videv_he(mqtt_client, get_topic(props.STG_VDE, loc, roo, props
.FUN_HEA))
275
276
277 class ffw_livingroom(base_room):
278     def __init__(self, mqtt_client: mqtt_client):
279         loc = props.LOC_FFW
280         roo = props.ROO_LIV
281         #
282         # http://shelly1-84CCA8ACE6A1
283         self.switch_main_light = shelly_sw1(mqtt_client, get_topic(props.STG_SHE, loc, roo, props
.FUN_MAL))
284         self.light_main_light = tradfri_sw_br_ct(mqtt_client, get_topic(props.STG_ZFW, loc, roo,
props.FUN_MAL))
285         self.videv_main_light = videv_sw_br_ct(mqtt_client, get_topic(props.STG_VDE, loc, roo,
props.FUN_MAL))
286

```

```

287         self.valve_heating = brennenstuhl_heatingvalve(mqtt_client, get_topic(props.STG_ZFW, loc,
288         roo, props.FUN_HEA))
289         self.videv_heating = videv_heating(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.
290         FUN_HEA))
291
292     class ffw_sleep(base_room):
293     def __init__(self, mqtt_client: mqtt_client):
294         loc = props.LOC_FFW
295         roo = props.ROO_SLP
296         #
297         # http://shelly1-3494546A51F2
298         self.switch_main_light = shelly_sw1(mqtt_client, get_topic(props.STG_SHE, loc, roo, props
299         .FUN_MAL))
300         self.light_main_light = tradfri_sw_br(mqtt_client, get_topic(props.STG_ZFW, loc, roo,
301         props.FUN_MAL))
302         self.videv_main_light = videv_sw_br(mqtt_client, get_topic(props.STG_VDE, loc, roo, props
303         .FUN_MAL))
304
305         self.light_window_light = tradfri_sw_br_ct(mqtt_client, get_topic(props.STG_ZFW, loc, roo
306         , props.FUN_WIL))
307         self.videv_window_light = videv_sw_br_ct(mqtt_client, get_topic(props.STG_VDE, loc, roo,
308         props.FUN_WIL))
309
310         self.valve_heating = brennenstuhl_heatingvalve(mqtt_client, get_topic(props.STG_ZFW, loc,
311         roo, props.FUN_HEA))
312         self.videv_heating = videv_heating(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.
313         FUN_HEA))
314
315     #
316     # GAR
317     #####
318     #
319     class gar_garden(base_room):
320     def __init__(self, mqtt_client: mqtt_client):
321         loc = props.LOC_GAR
322         roo = props.ROO_GAR
323         #
324         self.switch_garland_light = silvercrest_powerplug(mqtt_client, get_topic(props.STG_ZGW,
325         loc, roo, props.FUN_GAR))
326         self.videv_garland_light = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo, props
327         .FUN_GAR))
328
329         self.switch_repeater = silvercrest_powerplug(mqtt_client, get_topic(props.STG_ZGW, loc,
330         roo, props.FUN_REP))
331         self.videv_repeater = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.
332         FUN_REP))
333
334         self.videv_mode = videv_pure_switch(mqtt_client, get_topic(props.STG_VDE, loc, roo, props
335         .FUN_MOD))
336
337     #
338     # GFW
339     #####
340     #
341     class gfw_dirk(base_room):
342     def __init__(self, mqtt_client: mqtt_client):
343         loc = props.LOC_GFW
344         roo = props.ROO_DIR

```

```

332     #
333     # http://shelly1l-3C6105E44F27
334     self.switch_main_light = shelly_sw1(mqtt_client, get_topic(props.STG_SHE, loc, roo, props
335     .FUN_MAL))
336     self.light_main_light = tradfri_sw_br_ct(mqtt_client, get_topic(props.STG_ZGW, loc, roo,
337     props.FUN_MAL))
338     self.videv_main_light = videv_sw_br_ct(mqtt_client, get_topic(props.STG_VDE, loc, roo,
339     props.FUN_MAL))
340
341     self.input_device = tradfri_button(mqtt_client, get_topic(props.STG_ZGW, loc, roo, props.
342     FUN_INP))
343     self.videv_multistate = videv_multistate(mqtt_client, get_topic(props.STG_VDE, loc, roo,
344     props.FUN_VMS))
345
346     self.switch_powerplug_4 = my_powerplug(mqtt_client, get_topic(props.STG_MYA, loc, roo,
347     props.FUN_MPP))
348     self.KEY_POWERPLUG_AMPLIFIER = self.switch_powerplug_4.KEY_OUTPUT_0
349     self.KEY_POWERPLUG_PHONO = self.switch_powerplug_4.KEY_OUTPUT_1
350     self.KEY_POWERPLUG_CD_PLAYER = self.switch_powerplug_4.KEY_OUTPUT_2
351     self.KEY_POWERPLUG_BT = self.switch_powerplug_4.KEY_OUTPUT_3
352     self.switch_powerplug_4.set_ch_name(self.KEY_POWERPLUG_AMPLIFIER, "amplifier")
353     self.switch_powerplug_4.set_ch_name(self.KEY_POWERPLUG_PHONO, "phono")
354     self.switch_powerplug_4.set_ch_name(self.KEY_POWERPLUG_CD_PLAYER, "cd-player")
355     self.switch_powerplug_4.set_ch_name(self.KEY_POWERPLUG_BT, "bluetooth")
356
357     self.videv_amplifier = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.
358     FUN_AMP))
359     self.videv_cd_player = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.
360     FUN_CDP))
361     self.videv_bluetooth = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.
362     FUN_BTP))
363     self.videv_phono = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.FUN_PHO
364     ))
365
366     self.light_desk_light = tradfri_sw_br_ct(mqtt_client, get_topic(props.STG_ZGW, loc, roo,
367     props.FUN_DEL))
368     self.videv_desk_light = videv_sw_br_ct(mqtt_client, get_topic(props.STG_VDE, loc, roo,
369     props.FUN_DEL))
370
371     self.switch_pc_dock = silvercrest_powerplug(mqtt_client, get_topic(props.STG_ZGW, loc,
372     roo, props.FUN_DCK))
373     self.videv_pc_dock = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.
374     FUN_DCK))
375
376     self.remote_ctrl = remote(mqtt_client, get_topic(props.STG_MYA, loc, roo, props.FUN_RCA))
377     self.audio_status_spotify = audio_status(mqtt_client, get_topic(props.STG_MYA, loc, roo,
378     props.FUN_ASS))
379     self.audio_status_mpd = audio_status(mqtt_client, get_topic(props.STG_MYA, loc, roo,
380     props.FUN_ASM))
381     self.audio_status_bluetooth = audio_status(mqtt_client, get_topic(props.STG_MYA, loc, roo
382     , props.FUN_ASB))
383     self.videv_audio_player = videv_audio_player(mqtt_client, get_topic(props.STG_VDE, loc,
384     roo, props.FUN_VAU))
385
386     self.valve_heating = brennenstuhl_heatingvalve(mqtt_client, get_topic(props.STG_ZGW, loc,
387     roo, props.FUN_HEA))
388     self.ambient_info = my_ambient(mqtt_client, get_topic(props.STG_MYA, loc, roo, props.
389     FUN_AMB))
390     self.videv_heating = videv_heating(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.
391     FUN_HEA))

```

```

373 class gfw_floor(base_room):
374     def __init__(self, mqtt_client: mqtt_client):
375         loc = props.LOC_GFW
376         roo = props.ROO_FLO
377         #
378         # http://shelly1l-84CCA8AD1148
379         self.switch_main_light = shelly_sw1(mqtt_client, get_topic(props.STG_SHE, loc, roo, props
.FUN_MAL))
380         self.light_main_light: tradfri_sw_br_ct = self.__get_group__(tradfri_sw_br_ct,
mqtt_client, props.STG_ZGW, loc, roo, props.FUN_MAL, 2)
381         self.videv_main_light = videv_sw_br_ct(mqtt_client, get_topic(props.STG_VDE, loc, roo,
props.FUN_MAL))
382
383
384 class gfw_marion(base_room):
385     def __init__(self, mqtt_client: mqtt_client):
386         loc = props.LOC_GFW
387         roo = props.ROO_MAR
388         # http://shelly1l-E8DB84A1E067
389         self.switch_main_light = shelly_sw1(mqtt_client, get_topic(props.STG_SHE, loc, roo, props
.FUN_MAL))
390         self.videv_main_light = videv_sw(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.
FUN_MAL))
391
392         self.light_window_light = tradfri_sw_br_ct(mqtt_client, get_topic(props.STG_ZGW, loc, roo
, props.FUN_WIL))
393         self.videv_window_light = videv_sw_br_ct(mqtt_client, get_topic(props.STG_VDE, loc, roo,
props.FUN_WIL))
394
395         self.valve_heating = brennenstuhl_heatingvalve(mqtt_client, get_topic(props.STG_ZGW, loc,
roo, props.FUN_HEA))
396         self.videv_heating = videv_heating(mqtt_client, get_topic(props.STG_VDE, loc, roo, props.
FUN_HEA))
397
398
399 #
400 # STW
401 #####
402 #
403 class stairway(base_room):
404     def __init__(self, mqtt_client: mqtt_client):
405         loc = props.LOC_STW
406         #
407         # http://shelly1-3494546A9364
408         self.switch_main_light = shelly_sw1(mqtt_client, get_topic(props.STG_SHE, loc, props.
ROO_STF, props.FUN_MAL))
409         self.motion_main_light_gf = silvercrest_motion_sensor(mqtt_client, get_topic(props.
STG_ZGW, loc, props.ROO_STG, props.FUN_MSE))
410         self.motion_main_light_ff = silvercrest_motion_sensor(mqtt_client, get_topic(props.
STG_ZFE, loc, props.ROO_STF, props.FUN_MSE))
411         self.videv_main_light = videv_sw_mo(mqtt_client, get_topic(props.STG_VDE, loc, props.
ROO_STF, props.FUN_MAL))

```

### B.1.3 devdi.topic.py

The line coverage for devdi.topic.py was 100.0%

The branch coverage for devdi.topic.py was 85.7%

```
1 from collections import UserString
```

```
2
```

```

3 STOP_EXECUTION_TOPIC = "TESTRUN_WHILE_DEBUG_ON/STOP_EXECUTION"
4
5 #
6 # Device TYpe definitions
7 #
8 DTY_SHY_SW1 = 1
9 """ Shelly """
10 DTY_TLI_Sxx = 2
11 """ Tradfri Light (Switching only) """
12 DTY_TLI_SBx = 3
13 """ Tradfri Light (Switching and Brightnes) """
14 DTY_TLI_SBT = 4
15 """ Tradfri Light (Switching, Brightnes and Colortemperature) """
16 DTY_TIN_5xx = 5
17 """ Tradfri Input Device (5 Buttons) """
18 DTY_LLI_SBT = 6
19 """ Livarno Light (Switching, Brightnes and Colortemperature) """
20 DTY_BVL_xxx = 7
21 """ Brennenstuhl Heatingvalve """
22 DTY_SPP_SW1 = 8
23 """ Silvercrest Powerplug """
24 DTY_SMS_xxx = 9
25 """ Silvercrest Motion Sensor """
26 DTY_MPP_4xx = 10
27 """ My Powerplug (4 plugs) """
28 DTY_MAS_xxx = 11
29 """ My Audio status (MPD) """
30 DTY_MRE_xxx = 12
31 """ My Remote control """
32 DTY_MAM_THP = 13
33 """ My Ambient Information (Temperature, Humidity, Pressure)"""
34 DTY_HLI_SBT = 14
35 """ Hue Light (Switching, Brightnes and Colortemperature) """
36
37 #
38 # Source Transmission Group
39 #
40 STG_ZGW = 1
41 """ Zigbee ground floor west """
42 STG_ZFW = 2
43 """ Zigbee first floor west """
44 STG_ZFE = 3
45 """ Zigbee first floor east """
46 STG_SHE = 4
47 """ Shellies """
48 STG_MYA = 5
49 """ My Applications """
50 STG_VDE = 6
51 """ Videv Devices """
52
53
54 #
55 # LOCation
56 #
57 LOC_GFW = 1
58 """ Ground floor west """
59 LOC_GFE = 2
60 """ Ground floor east """
61 LOC_STW = 3
62 """ Stairway """
63 LOC_FFW = 4

```

```

64  """ First floor west """
65  LOC_FFE = 5
66  """ First floor east """
67  LOC_STW = 6
68  """ Stairways """
69  LOC_GAR = 7
70
71
72  #
73  # ROOMs
74  #
75  ROO_DIN = 1
76  """ Diningroom """
77  ROO_KIT = 2
78  """ Kitchen """
79  ROO_LIV = 3
80  """ Livingroom """
81  ROO_FLO = 4
82  """ Floor """
83  ROO_SLP = 5
84  """ Sleep """
85  ROO_BAT = 6
86  """ Bath """
87  ROO_DIR = 7
88  """ Dirk """
89  ROO_MAR = 8
90  """ Marion """
91  ROO_JUL = 9
92  """ Julian """
93  ROO_STG = 10
94  """ ground floor """
95  ROO_STF = 11
96  """ first floor """
97  ROO_GAR = 12
98  """ garden """
99
100
101  #
102  # FUNctions
103  #
104  FUN_MAL = 1
105  """ Main Light """
106  FUN_DEL = 2
107  """ Desk Light """
108  FUN_FLL = 3
109  """ Floor Light """
110  FUN_BLD = 4
111  """ Bed Light Dirk """
112  FUN_BLM = 5
113  """ Bed Light Marion """
114  FUN_HEA = 6
115  """ Heating """
116  FUN_MPP = 7
117  """ Multiple Powerplugs """
118  FUN_INP = 8
119  """ Input Device """
120  FUN_CIR = 9
121  """ Circulation Pump """
122  FUN_GAR = 10
123  """ Garland """
124  FUN_XTR = 11

```

```

125 """ X-Mas Tree """
126 FUN_XST = 12
127 """ X-Mas Star """
128 FUN_MSE = 13
129 """ Motion Sensor """
130 FUN_RCA = 14
131 """ Remote Control Amplifier """
132 FUN_RCC = 15
133 """ Remote Control CD-Player """
134 FUN_ASS = 16
135 """ Audio status spotify """
136 FUN_ASM = 17
137 """ Audio status mpd """
138 FUN_ASB = 18
139 """ Audio status bluetooth """
140 FUN_DCK = 19
141 """ Docking Station """
142 FUN_AMB = 20
143 """ Ambient information """
144 FUN_REP = 21
145 """ Repeater suppla """
146 FUN_WLI = 22
147 """ Warddrobe light """
148 FUN_WIL = 23
149 """ Window light """
150 FUN_AMP = 24
151 """ Amplifier """
152 FUN_CDP = 25
153 """ CD Player """
154 FUN_BTP = 26
155 """ Bluetooth """
156 FUN_PHO = 27
157 """ Phono """
158 FUN_VMS = 28
159 """ Virtual Multi State"""
160 FUN_MOD = 29
161 """ Mode """
162 FUN_VAU = 30
163 """ Virtual Audio player status """
164
165
166 STG_TOPIC = {
167     STG_ZGW: 'zigbee_gfw',
168     STG_ZFW: 'zigbee_ffw',
169     STG_ZFE: 'zigbee_ffe',
170     STG_SHE: 'shellies',
171     STG_MYA: 'my_apps',
172     STG_VDE: 'videv',
173 }
174
175 LOC_TOPIC = {
176     LOC_GFE: 'gfe',
177     LOC_GFW: 'gfw',
178     LOC_FFE: 'ffe',
179     LOC_FFW: 'ffw',
180     LOC_GAR: 'gar',
181     LOC_STW: 'stw',
182 }
183
184 ROO_TOPIC = {

```

```

185     ROO_DIN: 'diningroom',
186     ROO_KIT: 'kitchen',
187     ROO_LIV: 'livingroom',
188     ROO_FLO: 'floor',
189     ROO_SLP: 'sleep',
190     ROO_BAT: 'bath',
191     ROO_DIR: 'dirk',
192     ROO_MAR: 'marion',
193     ROO_JUL: 'julian',
194     ROO_STG: 'groundfloor',
195     ROO_STF: 'firstfloor',
196     ROO_GAR: 'garden',
197 }
198
199 FUN_TOPIC = {
200     FUN_MAL: 'main_light',
201     FUN_DEL: 'desk_light',
202     FUN_FLL: 'floor_light',
203     FUN_BLD: 'bed_light_di',
204     FUN_BLM: 'bed_light_ma',
205     FUN_HEA: 'heating_valve',
206     FUN_MPP: 'powerplug',
207     FUN_INP: 'input_device',
208     FUN_DCK: 'dock',
209     FUN_CIR: 'circulation_pump',
210     FUN_GAR: 'garland',
211     FUN_XTR: 'xmas-tree',
212     FUN_XST: 'xmas-star',
213     FUN_MSE: 'motion_sensor',
214     FUN_RCA: 'remote_ctrl/RAS5',
215     FUN_RCC: 'remote_ctrl/EUR642100',
216     FUN_ASS: 'audio_status_spotify',
217     FUN_ASM: 'audio_status_mpd',
218     FUN_ASB: 'audio_status_bt',
219     FUN_AMB: 'ambient',
220     FUN_REP: 'repeater',
221     FUN_WLI: 'wardrobe_light',
222     FUN_WIL: 'window_light',
223     FUN_AMP: 'amplifier',
224     FUN_CDP: 'cd_player',
225     FUN_BTP: 'bt',
226     FUN_PHO: 'phono',
227     FUN_VMS: 'active_brightness_device',
228     FUN_MOD: 'mode',
229     FUN_VAU: 'audio_player'
230 }
231
232
233 def get_topic(stg, loc, roo, fun):
234     stg_topic = STG_TOPIC[stg]
235     loc_topic = LOC_TOPIC[loc]
236     roo_topic = ROO_TOPIC[roo]
237     fun_topic = FUN_TOPIC[fun]
238     s = '/'.join([stg_topic, loc_topic, roo_topic, fun_topic])
239     # TODO: /\ Changed TOPIC in VIDEV /\ - Remove this line after changing nodered
240     TOPIC_STW_STAIRWAY_MAIN_LIGHT_VIDEV = "videv/stw/stairway/main_light"
241     if stg == STG_VDE and fun == FUN_DCK:
242         s = '/'.join([stg_topic, loc_topic, roo_topic, 'pc_dock'])
243     if stg == STG_VDE and fun == FUN_FLL:
244         s = '/'.join([stg_topic, loc_topic, roo_topic, 'floorlamp'])
245     if stg == STG_VDE and roo == ROO_STF and fun == FUN_MAL:

```

```

246     s = TOPIC_STW_STAIRWAY_MAIN_LIGHT_VIDEV
247     if stg == STG_VDE and fun == FUN_XTR:
248         s = '/'.join([stg_topic, loc_topic, roo_topic, 'xmas_tree'])
249     # TODO: /\ Changed TOPIC in VIDEV /\ - Remove this line after changing nodered
250     return s

```

## B.2 devices

The line coverage for devices was 72.9%

The branch coverage for devices was 40.0%

### B.2.1 devices.\_\_init\_\_.py

The line coverage for devices.\_\_init\_\_.py was 94.7%

The branch coverage for devices.\_\_init\_\_.py was 40.0%

```

1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3  #
4  """
5  devices (DEVICES)
6  =====
7
8  **Author:**
9
10 * Dirk Alders <sudo-dirk@mount-mockery.de>
11
12 **Description:**
13
14     This Module supports smarthome devices
15
16 **Submodules:**
17
18 * :mod:`shelly`
19 * :mod:`silvercrest_powerplug`
20
21 **Unittest:**
22
23     See also the :download:`unittest <devices/_testresults_/unittest.pdf>` documentation.
24
25 **Module Documentation:**
26
27 """
28 import logging
29
30 from devices.shelly import shelly as shelly_sw1
31 from devices.shelly import shelly_rpc as shelly_pro3
32 from devices.hue import hue_light as hue_sw_br_ct
33 from devices.tradfri import tradfri_light as tradfri_sw
34 from devices.tradfri import tradfri_light as tradfri_sw_br
35 from devices.tradfri import tradfri_light as tradfri_sw_br_ct
36 from devices.tradfri import tradfri_button as tradfri_button
37 from devices.tradfri import tradfri_light as livarno_sw_br_ct
38 from devices.brennenstuhl import brennenstuhl_heatingvalve
39 from devices.silvercrest import silvercrest_button
40 from devices.silvercrest import silvercrest_powerplug
41 from devices.silvercrest import silvercrest_motion_sensor

```

```

42 from devices.mydevices import powerplug as my_powerplug
43 from devices.mydevices import audio_status
44 from devices.mydevices import remote
45
46 from function.videv import videv_switching as videv_sw
47 from function.videv import videv_switch_brightness as videv_sw_br
48 from function.videv import videv_switch_brightness_color_temp as videv_sw_br_ct
49 from function.videv import videv_switching_timer as videv_sw_tm
50 from function.videv import videv_switching_motion as videv_sw_mo
51 from function.videv import videv_heating as videv_he
52 from function.videv import videv_pure_switch
53 from function.videv import videv_multistate
54 from function.videv import videv_audio_player
55
56 try:
57     from config import APP_NAME as ROOT_LOGGER_NAME
58 except ImportError:
59     ROOT_LOGGER_NAME = 'root'
60 logger = logging.getLogger(ROOT_LOGGER_NAME).getChild(__name__)
61
62
63 def my_ambient(mqtt_client, topic):
64     logger.warning("Device type my_ambient is not yet implemented. Topic %s will not be supported", topic)
65     return None
66
67
68 class group(object):
69     def __init__(self, *args):
70         super().__init__()
71         self._members = args
72         self._iter_counter = 0
73         #
74         self.methods = []
75         self.variables = []
76         for name in [m for m in args[0].__class__.__dict__.keys()]:
77             if not name.startswith('_') and callable(getattr(args[0], name)): # add all public
78                 callable attributes to the list
79                 self.methods.append(name)
80                 if not name.startswith('_') and not callable(getattr(args[0], name)): # add all
81                     public callable attributes to the list
82                     self.variables.append(name)
83         #
84         for member in self:
85             methods = [m for m in member.__class__.__dict__.keys() if not m.startswith(
86                 '_') if not m.startswith('_') and callable(getattr(args[0], m))]
87             if self.methods != methods:
88                 raise ValueError("All given instances needs to have same methods:", self.methods,
89                                     methods)
90         #
91         variables = [v for v in member.__class__.__dict__.keys() if not v.startswith(
92             '_') if not v.startswith('_') and not callable(getattr(args[0], v))]
93         if self.variables != variables:
94             raise ValueError("All given instances needs to have same variables:", self.
95                                 variables, variables)
96
97     def __iter__(self):
98         return self
99

```

```

96     def __next__(self):
97         if self._iter_counter < len(self):
98             self._iter_counter += 1
99             return self._members[self._iter_counter - 1]
100         self._iter_counter = 0
101         raise StopIteration
102
103     def __getitem__(self, i):
104         return self._members[i]
105
106     def __len__(self):
107         return len(self._members)
108
109     def __getattr__(self, name):
110         def group_execution(*args, **kwargs):
111             for member in self[:]:
112                 m = getattr(member, name)
113                 m(*args, **kwargs)
114         try:
115             rv = super().__getattr__(name)
116         except AttributeError:
117             if callable(getattr(self[0], name)):
118                 return group_execution
119             else:
120                 return getattr(self[0], name)
121         else:
122             return rv

```

## B.2.2 devices.base.py

The line coverage for devices.base.py was 55.0%

The branch coverage for devices.base.py was 40.0%

```

1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3  #
4  from base import mqtt_base
5  from base import videv_base
6  import json
7
8
9  def is_json(data):
10     try:
11         json.loads(data)
12     except json.decoder.JSONDecodeError:
13         return False
14     else:
15         return True
16
17
18  class base(mqtt_base):
19     TX_TOPIC = "set"
20     TX_VALUE = 0
21     TX_DICT = 1
22     TX_TYPE = -1
23     TX_FILTER_DATA_KEYS = []
24     #
25     RX_KEYS = []
26     RX_IGNORE_TOPICS = []
27     RX_IGNORE_KEYS = []
28     RX_FILTER_DATA_KEYS = []

```

```

29 #
30 CFG_DATA = {}
31
32 def __init__(self, mqtt_client, topic):
33     super().__init__(mqtt_client, topic, default_values=dict.fromkeys(self.RX_KEYS))
34     # data storage
35     self.__cfg_by_mid__ = None
36     # initialisations
37     mqtt_client.add_callback(topic=self.topic, callback=self.receive_callback)
38     mqtt_client.add_callback(topic=self.topic+"/#", callback=self.receive_callback)
39     #
40     self.add_callback(None, None, self.__state_logging__, on_change_only=True)
41
42 def __cfg_callback__(self, key, value, mid):
43     if self.CFG_DATA.get(key) != value and self.__cfg_by_mid__ != mid and mid is not None:
44         self.__cfg_by_mid__ = mid
45         self.logger.warning("Differing configuration identified: Sending default
configuration to device: %s", repr(self.CFG_DATA))
46         if self.TX_TYPE == self.TX_DICT:
47             self.mqtt_client.send(self.topic + '/' + self.TX_TOPIC, json.dumps(self.CFG_DATA)
)
48         else:
49             for key in self.CFG_DATA:
50                 self.send_command(key, self.CFG_DATA.get(key))
51
52 def set(self, key, data, mid=None, block_callback=[]):
53     if key in self.CFG_DATA:
54         self.__cfg_callback__(key, data, mid)
55     if key in self.RX_IGNORE_KEYS:
56         pass # ignore these keys
57     elif key in self.RX_KEYS:
58         return super().set(key, data, block_callback)
59     else:
60         self.logger.warning("Unexpected key %s", key)
61
62 def receive_callback(self, client, userdata, message):
63     if message.topic != self.topic + '/' + videv_base.KEY_INFO:
64         content_key = message.topic[len(self.topic) + 1:]
65         if content_key not in self.RX_IGNORE_TOPICS and (not message.topic.endswith(self.
TX_TOPIC) or len(self.TX_TOPIC) == 0):
66             self.logger.debug("Unpacking content_key \"%s\" from message.", content_key)
67             if is_json(message.payload):
68                 data = json.loads(message.payload)
69                 if type(data) is dict:
70                     for key in data:
71                         self.set(key, self.__device_to_instance_filter__(key, data[key]),
message.mid)
72             else:
73                 self.set(content_key, self.__device_to_instance_filter__(content_key,
data), message.mid)
74             # String
75             else:
76                 self.set(content_key, self.__device_to_instance_filter__(content_key, message
.payload.decode('utf-8')), message.mid)
77             else:
78                 self.logger.debug("Ignoring topic %s", content_key)
79
80 def __device_to_instance_filter__(self, key, data):
81     if key in self.RX_FILTER_DATA_KEYS:
82         if data in [1, 'on', 'ON']:
83             return True
84         elif data in [0, 'off', 'OFF']:
85             return False
86     return data

```

```

87
88     def __instance_to_device_filter__(self, key, data):
89         if key in self.TX_FILTER_DATA_KEYS:
90             if data is True:
91                 return "on"
92             elif data is False:
93                 return "off"
94             return data
95
96     def send_command(self, key, data):
97         data = self.__instance_to_device_filter__(key, data)
98         if self.TX_TOPIC is not None:
99             if self.TX_TYPE < 0:
100                 self.logger.error("Unknown tx type. Set TX_TYPE of class to a known value")
101             else:
102                 self.logger.debug("Sending data for %s - %s", key, str(data))
103                 if self.TX_TYPE == self.TX_DICT:
104                     try:
105                         self.mqtt_client.send('/'.join([self.topic, self.TX_TOPIC]), json.dumps({
106                             key: data}))
107                     except TypeError:
108                         print(self.topic)
109                         print(key.__dict__)
110                         print(key)
111                         print(data)
112                         raise TypeError
113                 else:
114                     if type(data) not in [str, bytes]:
115                         data = json.dumps(data)
116                     self.mqtt_client.send('/'.join([self.topic, key, self.TX_TOPIC] if len(self.
117 TX_TOPIC) > 0 else [self.topic, key]), data)
118                 else:
119                     self.logger.error("Unknown tx topic. Set TX_TOPIC of class to a known value")
120
121 class base_rpc(mqtt_base):
122     SRC_RESPONSE = "/response"
123     SRC_NULL = "/null"
124     #
125     EVENTS_TOPIC = "/events/rpc"
126     TX_TOPIC = "/rpc"
127     RESPONSE_TOPIC = SRC_RESPONSE + "/rpc"
128     NULL_TOPIC = SRC_NULL + "/rpc"
129     #
130     RPC_ID_GET_STATUS = 1
131     RPC_ID_SET = 1734
132     #
133     def __init__(self, mqtt_client, topic):
134         super().__init__(mqtt_client, topic, default_values=dict.fromkeys(self.RX_KEYS))
135         # data storage
136         self.__cfg_by_mid__ = None
137         # initialisations
138         mqtt_client.add_callback(topic=self.topic, callback=self.receive_callback)
139         mqtt_client.add_callback(topic=self.topic+"/#", callback=self.receive_callback)
140         #
141         self.add_callback(None, None, self.__state_logging__, on_change_only=False)
142         #
143         self.rpc_get_status()
144
145     def receive_callback(self, client, userdata, message):
146         data = json.loads(message.payload)
147         #

```

```

148         if message.topic == self.topic + self.EVENTS_TOPIC:
149             self.events(data)
150         elif message.topic == self.topic + self.RESPONSE_TOPIC:
151             self.response(data)
152         elif message.topic == self.topic + self.NULL_TOPIC or message.topic == self.topic + self.
TX_TOPIC or message.topic == self.topic + "/online":
153             pass # ignore response
154         else:
155             self.logger.warning("Unexpected message received: %s::%s", message.topic, json.dumps(
data, sort_keys=True, indent=4))
156
157     def events(self, data):
158         for rx_key in data["params"]:
159             if rx_key == "events":
160                 for evt in data["params"]["events"]:
161                     key = evt["component"]
162                     event = evt["event"]
163                     if key in self.RX_KEYS:
164                         if event == "btn_down":
165                             self.set(key, True)
166                         elif event == "btn_up":
167                             self.set(key, False)
168                     else:
169                         key = key + ":" + event
170                         if key in self.RX_KEYS:
171                             self.set(key, True)
172                     else:
173                         self.logger.warning("Unexpected event with data=%s", json.dumps(
data, sort_keys=True, indent=4))
174                 elif rx_key in self.RX_KEYS:
175                     state = data["params"][rx_key].get("output")
176                     if state is not None:
177                         self.set(rx_key, state)
178
179     def response(self, data):
180         try:
181             rpc_id = data.get("id")
182         except AttributeError:
183             rpc_id = None
184         try:
185             rpc_method = data.get("method")
186         except AttributeError:
187             rpc_method = None
188         if rpc_id == self.RPC_ID_GET_STATUS:
189             #
190             # Shelly.GetStatus
191             #
192             for rx_key in data.get("result", []):
193                 if rx_key in self.RX_KEYS:
194                     key_data = data["result"][rx_key]
195                     state = key_data.get("output", key_data.get("state"))
196                     if state is not None:
197                         self.set(rx_key, state)
198             else:
199                 self.logger.warning("Unexpected response with data=%s", json.dumps(data, sort_keys=
True, indent=4))
200
201     def rpc_tx(self, **kwargs):
202         if not "id" in kwargs:
203             raise AttributeError("'id' is missing in keyword arguments")
204         self.mqtt_client.send(self.topic + self.TX_TOPIC, json.dumps(kwargs))
205

```

```

206 def rpc_get_status(self):
207     self.rpc_tx(
208         id=self.RPC_ID_GET_STATUS,
209         src=self.topic + self.SRC_RESPONSE,
210         method="Shelly.GetStatus"
211     )
212
213 def rpc_switch_set(self, key, state: bool):
214     self.rpc_tx(
215         id=self.RPC_ID_SET,
216         src=self.topic + self.SRC_NULL,
217         method="Switch.Set",
218         params={"id": int(key[-1]), "on": state}
219     )
220
221
222 class base_output(base):
223     def __init__(self, mqtt_client, topic):
224         super().__init__(mqtt_client, topic)
225         self.__all_off_enabled__ = True
226
227     def disable_all_off(self, state=True):
228         self.__all_off_enabled__ = not state
229
230     def all_off(self):
231         if self.__all_off_enabled__:
232             try:
233                 self.__all_off__()
234             except (AttributeError, TypeError) as e:
235                 self.logger.warning("Method all_off was used, but __all_off__ method wasn't
                callable: %s", repr(e))

```

### B.2.3 devices.brennenstuhl.py

The line coverage for devices.brennenstuhl.py was 93.4%

The branch coverage for devices.brennenstuhl.py was 40.0%

```

1 #!/usr/bin/env python
2 # -*- coding: utf-8 -*-
3 #
4 from devices.base import base
5 import task
6 import time
7
8
9 class brennenstuhl_heatingvalve(base):
10     """ Communication (MQTT)
11
12     brennenstuhl_heatingvalve {
13         |         "away_mode": ["ON", "OFF"]
14         |         "battery": [0...100] %
15         |         "child_lock": ["LOCK", "UNLOCK"]
16         |         "current_heating_setpoint": [5...30] °C
17         |         "linkquality": [0...255] lqi
18         |         "local_temperature": [numeric] °C
19         |         "preset": ["manual", ...]
20         |         "system_mode": ["heat", ...]
21         |         "valve_detection": ["ON", "OFF"]
22         |         "window_detection": ["ON", "OFF"]
23         |     }

```

```

24         +- set {
25             "away_mode": ["ON", "OFF", "TOGGLE"]
26             "child_lock": ["LOCK", "UNLOCK"]
27             "current_heating_setpoint": [5...30] °C
28             "preset": ["manual", ...]
29             "system_mode": ["heat", ...]
30             "valve_detection": ["ON", "OFF", "TOGGLE"]
31             "window_detection": ["ON", "OFF", "TOGGLE"]
32         }
33
34     """
35     KEY_LINKQUALITY = "linkquality"
36     KEY_BATTERY = "battery"
37     KEY_HEATING_SETPOINT = "current_heating_setpoint"
38     KEY_TEMPERATURE = "local_temperature"
39     #
40     KEY_AWAY_MODE = "away_mode"
41     KEY_CHILD_LOCK = "child_lock"
42     KEY_PRESET = "preset"
43     KEY_SYSTEM_MODE = "system_mode"
44     KEY_VALVE_DETECTION = "valve_detection"
45     KEY_WINDOW_DETECTION = "window_detection"
46     #
47     RETRY_CYCLE_TIME = 2.5
48     MAX_TX_RETRIES = 20
49     RETRY_TIMEOUT = RETRY_CYCLE_TIME * MAX_TX_RETRIES
50     #
51     TX_TYPE = base.TX_DICT
52     #
53     RX_KEYS = [KEY_LINKQUALITY, KEY_BATTERY, KEY_HEATING_SETPOINT, KEY_TEMPERATURE]
54     RX_IGNORE_KEYS = [KEY_AWAY_MODE, KEY_CHILD_LOCK, KEY_PRESET, KEY_SYSTEM_MODE,
55                       KEY_VALVE_DETECTION, KEY_WINDOW_DETECTION]
56     #
57     CFG_DATA = {
58         KEY_WINDOW_DETECTION: "ON",
59         KEY_VALVE_DETECTION: "ON",
60         KEY_SYSTEM_MODE: "heat",
61         KEY_PRESET: "manual"
62     }
63
64     def __init__(self, mqtt_client, topic):
65         super().__init__(mqtt_client, topic)
66         self.add_callback(self.KEY_HEATING_SETPOINT, None, self.__valave_temp_rx__)
67         self.__tx_temperature__ = None
68         self.__rx_temperature__ = None
69         self.__tx_timestamp__ = 0
70         #
71         self.task = task.periodic(self.RETRY_CYCLE_TIME, self.__task__)
72         self.task.run()
73
74     def __state_logging__(self, inst, key, data):
75         if key in [self.KEY_HEATING_SETPOINT, self.KEY_CHILD_LOCK, self.KEY_WINDOW_DETECTION,
76                  self.KEY_VALVE_DETECTION]:
77             self.logger.info("State change of '%s' to '%s'", key, repr(data))
78
79     def send_command(self, key, data):
80         if key == self.KEY_HEATING_SETPOINT:
81             self.__tx_temperature__ = data
82             self.__tx_timestamp__ = time.time()
83             base.send_command(self, key, data)
84
85     def __valave_temp_rx__(self, inst, key, data):
86         if key == self.KEY_HEATING_SETPOINT:
87             self.__rx_temperature__ = data

```

## Unittest for smart\_brain

```
85
86 def __task__(self, rt):
87     if self.__tx_temperature__ is not None and self.__tx_timestamp__ is not None: # Already
88         send a setpoint
89         if self.__tx_temperature__ != self.__rx_temperature__: #
90             Setpoint and valve feedback are unequal
91             if time.time() - self.__tx_timestamp__ < self.RETRY_TIMEOUT: # Timeout
92                 condition allows resend of setpoint
93                 self.logger.warning("Setpoint not yet acknowledged by device. Sending setpoint
94                 again")
95                 self.set_heating_setpoint(self.__tx_temperature__)
96                 return
97             else:
98                 self.__tx_timestamp__ = None # Disable
99                 resend logic, if setpoint and valve setpoint are equal
100
101 #
102 # RX
103 #
104 @property
105 def linkquality(self):
106     return self.get(self.KEY_LINKQUALITY)
107
108 @property
109 def heating_setpoint(self):
110     return self.get(self.KEY_HEATING_SETPOINT)
111
112 @property
113 def temperature(self):
114     return self.get(self.KEY_TEMPERATURE)
115
116 #
117 # TX
118 #
119 def set_heating_setpoint(self, setpoint):
120     self.send_command(self.KEY_HEATING_SETPOINT, setpoint)
121
122 def set_heating_setpoint_mcb(self, device, key, data):
123     self.set_heating_setpoint(data)
```

### B.2.4 devices.hue.py

The line coverage for devices.hue.py was 64.4%

The branch coverage for devices.hue.py was 40.0%

```
1 #!/usr/bin/env python
2 # -*- coding: utf-8 -*-
3 #
4 from devices.base import base, base_output
5 import logging
6
7
8 class hue_light(base_output):
```

```

9     """ Communication (MQTT)
10
11     hue_light {
12         |      "state": ["ON" / "OFF" / "TOGGLE"]
13         |      "linkquality": [0...255] |qi
14         |      "brightness": [0...254]
15         |      "color_mode": ["color_temp"]
16         |      "color_temp": ["coolest", "cool", "neutral", "warm", "warmest", 250...454]
17         |    }
18     +- get {
19         |      "state": ""
20         |    }
21     +- set {
22         |      "state": ["ON" / "OFF"]
23         |      "brightness": [0...256]
24         |      "color_temp": [250...454]
25         |      "transition": [0...] seconds
26         |      "brightness_move": [-X...0...X] X/s
27         |      "brightness_step": [-X...0...X]
28         |      "color_temp_move": [-X...0...X] X/s
29         |      "color_temp_step": [-X...0...X]
30     }
31
32     """
33     KEY_LINKQUALITY = "linkquality"
34     KEY_OUTPUT_0 = "state"
35     KEY_BRIGHTNESS = "brightness"
36     KEY_COLOR_TEMP = "color_temp"
37
38     #
39     TX_TYPE = base.TX_DICT
40     TX_FILTER_DATA_KEYS = [KEY_OUTPUT_0, KEY_BRIGHTNESS, KEY_COLOR_TEMP]
41     STATE_KEYS = TX_FILTER_DATA_KEYS
42
43     #
44     RX_KEYS = [KEY_LINKQUALITY, KEY_OUTPUT_0, KEY_BRIGHTNESS, KEY_COLOR_TEMP]
45     RX_IGNORE_KEYS = ['update', 'color_mode']
46     RX_FILTER_DATA_KEYS = [KEY_OUTPUT_0, KEY_BRIGHTNESS, KEY_COLOR_TEMP]
47
48     def __state_logging__(self, inst, key, data):
49         if key in [self.KEY_OUTPUT_0, self.KEY_BRIGHTNESS, self.KEY_COLOR_TEMP]:
50             self.logger.info("State change of '%s' to '%s'", key, repr(data))
51
52     def __device_to_instance_filter__(self, key, data):
53         if key == self.KEY_BRIGHTNESS:
54             return int(round((data - 1) * 100 / 253, 0))
55         elif key == self.KEY_COLOR_TEMP:
56             return int(round((data - 250) * 10 / 204, 0))
57         return super().__device_to_instance_filter__(key, data)
58
59     def __instance_to_device_filter__(self, key, data):
60         if key == self.KEY_BRIGHTNESS:
61             return int(round(data * 253 / 100 + 1, 0))
62         elif key == self.KEY_COLOR_TEMP:
63             return int(round(data * 204 / 10 + 250, 0))
64         return super().__instance_to_device_filter__(key, data)
65
66     #
67     # RX
68     #
69     @property
70     def output_0(self):
71         """rv: [True, False]"""
72         return self.get(self.KEY_OUTPUT_0, False)

```

```

71     @property
72     def linkquality(self):
73         """rv: numeric value"""
74         return self.get(self.KEY_LINKQUALITY, 0)
75
76     @property
77     def brightness(self):
78         """rv: numeric value [0%, ..., 100%]"""
79         return self.get(self.KEY_BRIGHTNESS, 0)
80
81     @property
82     def color_temp(self):
83         """rv: numeric value [0, ..., 10]"""
84         return self.get(self.KEY_COLOR_TEMP, 0)
85
86     #
87     # TX
88     #
89     def request_data(self, device=None, key=None, data=None):
90         self.mqtt_client.send(self.topic + "/set", '{"hue_power_on_behavior": "recover"}')
91
92     def set_output_0(self, state):
93         """state: [True, False]"""
94         self.send_command(self.KEY_OUTPUT_0, state)
95
96     def set_output_0_mcb(self, device, key, data):
97         self.set_output_0(data)
98
99     def toggle_output_0_mcb(self, device, key, data):
100         self.set_output_0(not self.output_0)
101
102     def set_brightness(self, brightness):
103         """brightness: [0, ..., 100]"""
104         self.send_command(self.KEY_BRIGHTNESS, brightness)
105
106     def set_brightness_mcb(self, device, key, data):
107         self.set_brightness(data)
108
109     def set_color_temp(self, color_temp):
110         """color_temp: [0, ..., 10]"""
111         self.send_command(self.KEY_COLOR_TEMP, color_temp)
112
113     def set_color_temp_mcb(self, device, key, data):
114         self.set_color_temp(data)
115
116     def __all_off__(self):
117         if self.output_0:
118             self.set_output_0(False)

```

**B.2.5** devices.mydevices.py

The line coverage for devices.mydevices.py was 66.7%

The branch coverage for devices.mydevices.py was 40.0%

```

1 #!/usr/bin/env python
2 # -*- coding: utf-8 -*-
3 #
4 from devices.base import base, base_output
5 import logging
6
7
8 class powerplug(base_output):
9     """ Communication (MQTT)
10
11     my_powerplug
12         +- output
13             +- 1 [True, False] <- status
14             | +- set [True, False, "toggle"] <- command
15             +- 2 [True, False] <- status
16             | +- set [True, False, "toggle"] <- command
17             +- 3 [True, False] <- status
18             | +- set [True, False, "toggle"] <- command
19             +- 4 [True, False] <- status
20             | +- set [True, False, "toggle"] <- command
21             +- all
22             +- set [True, False, "toggle"] <- command
23
24     KEY_OUTPUT_0 = "output/1"
25     KEY_OUTPUT_1 = "output/2"
26     KEY_OUTPUT_2 = "output/3"
27     KEY_OUTPUT_3 = "output/4"
28     KEY_OUTPUT_ALL = "output/all"
29     KEY_OUTPUT_LIST = [KEY_OUTPUT_0, KEY_OUTPUT_1, KEY_OUTPUT_2, KEY_OUTPUT_3]
30     #
31     TX_TYPE = base.TX_VALUE
32     #
33     RX_KEYS = [KEY_OUTPUT_0, KEY_OUTPUT_1, KEY_OUTPUT_2, KEY_OUTPUT_3]
34
35     def __state_logging__(self, inst, key, data):
36         if key in self.KEY_OUTPUT_LIST:
37             self.logger.info("State change of '%s' to '%s'", key, repr(data))
38
39     #
40     # RX
41     #
42     @property
43     def output_0(self):
44         """rv: [True, False]"""
45         return self.get(self.KEY_OUTPUT_0)
46
47     @property
48     def output_1(self):
49         """rv: [True, False]"""
50         return self.get(self.KEY_OUTPUT_1)
51
52     @property
53     def output_2(self):
54         """rv: [True, False]"""
55         return self.get(self.KEY_OUTPUT_2)
56

```

```

57     @property
58     def output_3(self):
59         """rv: [True, False]"""
60         return self.get(self.KEY_OUTPUT_3)
61
62     #
63     # TX
64     #
65     def set_output(self, key, state):
66         if key in self.KEY_OUTPUT_LIST:
67             self.send_command(key, state)
68         else:
69             logging.error("Unknown key to set the output!")
70
71     def set_output_0(self, state):
72         """state: [True, False]"""
73         self.send_command(self.KEY_OUTPUT_0, state)
74
75     def set_output_0_mcb(self, device, key, data):
76         self.set_output_0(data)
77
78     def toggle_output_0_mcb(self, device, key, data):
79         self.set_output_0(not self.output_0)
80
81     def set_output_1(self, state):
82         """state: [True, False]"""
83         self.send_command(self.KEY_OUTPUT_1, state)
84
85     def set_output_1_mcb(self, device, key, data):
86         self.set_output_1(data)
87
88     def toggle_output_1_mcb(self, device, key, data):
89         self.set_output_1(not self.output_1)
90
91     def set_output_2(self, state):
92         """state: [True, False]"""
93         self.send_command(self.KEY_OUTPUT_2, state)
94
95     def set_output_2_mcb(self, device, key, data):
96         self.set_output_2(data)
97
98     def toggle_output_2_mcb(self, device, key, data):
99         self.set_output_2(not self.output_2)
100
101     def set_output_3(self, state):
102         """state: [True, False]"""
103         self.send_command(self.KEY_OUTPUT_3, state)
104
105     def set_output_3_mcb(self, device, key, data):
106         self.set_output_3(data)
107
108     def toggle_output_3_mcb(self, device, key, data):
109         self.set_output_3(not self.output_3)
110
111     def set_output_all(self, state):
112         """state: [True, False, 'toggle']"""
113         self.send_command(self.KEY_OUTPUT_ALL, state)
114
115     def set_output_all_mcb(self, device, key, data):
116         self.set_output_all(data)
117
118     def __all_off__(self):
119         self.set_output_all(False)

```

```

120
121
122 class remote(base):
123     """ Communication (MQTT)
124
125     remote (RAS5)                                     ← command
126         + CD [dc]
127         + LINE1 [dc]
128         + LINE2 [dc]
129         + LINE3 [dc]
130         + MUTE [dc]
131         + POWER [dc]
132         + VOLDDOWN [dc]
133         + VOLUP [dc]
134         + PHONO [dc]
135         + DOCK [dc]
136
137     remote (EUR642100)                                ← command
138         + OPEN_CLOSE [dc]
139         + VOLDDOWN [dc]
140         + VOLUP [dc]
141         + ONE [dc]
142         + TWO [dc]
143         + THREE [dc]
144         + FOUR [dc]
145         + FIVE [dc]
146         + SIX [dc]
147         + SEVEN [dc]
148         + EIGHT [dc]
149         + NINE [dc]
150         + ZERO [dc]
151         + TEN [dc]
152         + TEN_PLUS [dc]
153         + PROGRAM [dc]
154         + CLEAR [dc]
155         + RECALL [dc]
156         + TIME_MODE [dc]
157         + A_B_REPEAT [dc]
158         + REPEAT [dc]
159         + RANDOM [dc]
160         + AUTO_CUE [dc]
161         + TAPE_LENGTH [dc]
162         + SIDE_A_B [dc]
163         + TIME_FADE [dc]
164         + PEAK_SEARCH [dc]
165         + SEARCH_BACK [dc]
166         + SEARCH_FOR [dc]
167         + TRACK_NEXT [dc]
168         + TRACK_PREV [dc]
169         + STOP [dc]
170         + PAUSE [dc]
171         + PLAY [dc]
172     """
173     KEY_CD = "CD"
174     KEY_LINE1 = "LINE1"
175     KEY_LINE2 = "LINE2"
176     KEY_LINE3 = "LINE3"
177     KEY_PHONO = "PHONO"
178     KEY_MUTE = "MUTE"
179     KEY_POWER = "POWER"
180     KEY_VOLDOWN = "VOLDOWN"
181     KEY_VOLUP = "VOLUP"

```

```

182     #
183     TX_TOPIC = ''
184     TX_TYPE = base.TX_VALUE
185     #
186     RX_IGNORE_TOPICS = [KEY_CD, KEY_LINE1, KEY_LINE2, KEY_LINE3, KEY_PHONO, KEY_MUTE, KEY_POWER,
187                          KEY_VOLUP, KEY_VOLDOWN]
188
189     def __state_logging__(self, inst, key, data):
190         pass # This is just a TX device using self.set_*
191
192     def set_cd(self, device=None, key=None, data=None):
193         self.logger.info("Changing amplifier source to CD")
194         self.send_command(self.KEY_CD, None)
195
196     def set_line1(self, device=None, key=None, data=None):
197         self.logger.info("Changing amplifier source to LINE1")
198         self.send_command(self.KEY_LINE1, None)
199
200     def set_line2(self, device=None, key=None, data=None):
201         self.logger.info("Changing amplifier source to LINE2")
202         self.send_command(self.KEY_LINE2, None)
203
204     def set_line3(self, device=None, key=None, data=None):
205         self.logger.info("Changing amplifier source to LINE3")
206         self.send_command(self.KEY_LINE3, None)
207
208     def set_phono(self, device=None, key=None, data=None):
209         self.logger.info("Changing amplifier source to PHONO")
210         self.send_command(self.KEY_PHONO, None)
211
212     def set_mute(self, device=None, key=None, data=None):
213         self.logger.info("Muting / Unmuting amplifier")
214         self.send_command(self.KEY_MUTE, None)
215
216     def set_power(self, device=None, key=None, data=None):
217         self.logger.info("Power on/off amplifier")
218         self.send_command(self.KEY_POWER, None)
219
220     def set_volume_up(self, data=False):
221         """data: [True, False]"""
222         self.logger.info("Increasing amplifier volume")
223         self.send_command(self.KEY_VOLUP, data)
224
225     def set_volume_down(self, data=False):
226         """data: [True, False]"""
227         self.logger.info("Decreasing amplifier volume")
228         self.send_command(self.KEY_VOLDOWN, data)
229
230     def default_inc(self, device=None, key=None, data=None):
231         self.set_volume_up(True)
232
233     def default_dec(self, device=None, key=None, data=None):
234         self.set_volume_down(True)
235
236     def default_stop(self, device=None, key=None, data=None):
237         self.set_volume_up(False)
238
239 class audio_status(base):

```

```

240     """ Communication (MQTT)
241
242     audio_status
243         +- state [True, False]                <- status
244         +- title [text]                        <- status
245     """
246     KEY_STATE = "state"
247     KEY_TITLE = "title"
248     #
249     TX_TYPE = base.TX_VALUE
250     #
251     RX_KEYS = [KEY_STATE, KEY_TITLE]
252
253     def __state_logging__(self, inst, key, data):
254         if key in [self.KEY_STATE, self.KEY_TITLE]:
255             self.logger.info("State change of '%s' to '%s'", key, repr(data))
256
257     def set_state(self, num, data):
258         """data: [True, False]"""
259         self.send_command(self.KEY_STATE + "/" + str(num), data)
260
261     def set_state_mcb(self, device, key, data):
262         self.set_state(data)

```

## B.2.6 devices.shelly.py

The line coverage for devices.shelly.py was 72.3%

The branch coverage for devices.shelly.py was 40.0%

```

1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3  #
4  from devices.base import base_output
5  from devices.base import base_rpc
6  import task
7
8
9  class shelly(base_output):
10     """ Communication (MQTT)
11
12     shelly
13         +- relay
14         |     +- 0 ["on" / "off"]                <- status
15         |     | +- command ["on"/ "off"]          <- command
16         |     | +- energy [numeric]               <- status
17         |     +- 1 ["on" / "off"]                <- status
18         |     | +- command ["on"/ "off"]          <- command
19         |     | +- energy [numeric]               <- status
20         +- input
21         |     +- 0 [0 / 1]                        <- status
22         |     +- 1 [0 / 1]                        <- status
23         +- input_event
24         |     +- 0                                <- status
25         |     +- 1                                <- status
26         +- logpush
27         |     +- 0 [0 / 1]                        <- status
28         |     +- 1 [0 / 1]                        <- status
29         +- temperature [numeric] °C               <- status
30         +- temperature_f [numeric] F              <- status
31         +- overtemperature [0 / 1]                <- status

```

```

32         +- id                                     <- status
33         +- model                                  <- status
34         +- mac                                    <- status
35         +- ip                                     <- status
36         +- new_fw                                <- status
37         +- fw_ver                                <- status
38     """
39     KEY_OUTPUT_0 = "relay/0"
40     KEY_OUTPUT_1 = "relay/1"
41     KEY_INPUT_0 = "input/0"
42     KEY_INPUT_1 = "input/1"
43     KEY_LONGPUSH_0 = "longpush/0"
44     KEY_LONGPUSH_1 = "longpush/1"
45     KEY_TEMPERATURE = "temperature"
46     KEY_OVERTEMPERATURE = "overtemperature"
47     KEY_ID = "id"
48     KEY_MODEL = "model"
49     KEY_MAC = "mac"
50     KEY_IP = "ip"
51     KEY_NEW_FIRMWARE = "new_fw"
52     KEY_FIRMWARE_VERSION = "fw_ver"
53     #
54     TX_TOPIC = "command"
55     TX_TYPE = base_output.TX_VALUE
56     TX_FILTER_DATA_KEYS = [KEY_OUTPUT_0, KEY_OUTPUT_1]
57     #
58     RX_KEYS = [KEY_OUTPUT_0, KEY_OUTPUT_1, KEY_INPUT_0, KEY_INPUT_1, KEY_LONGPUSH_0,
59                KEY_LONGPUSH_1, KEY_OVERTEMPERATURE, KEY_TEMPERATURE,
60                KEY_ID, KEY_MODEL, KEY_MAC, KEY_IP, KEY_NEW_FIRMWARE, KEY_FIRMWARE_VERSION]
61     RX_IGNORE_TOPICS = [KEY_OUTPUT_0 + '/' + "energy", KEY_OUTPUT_1 + '/' + "energy", '
62                        input_event/0', 'input_event/1']
63     RX_IGNORE_KEYS = ['temperature_f']
64     RX_FILTER_DATA_KEYS = [KEY_INPUT_0, KEY_INPUT_1, KEY_LONGPUSH_0, KEY_LONGPUSH_1, KEY_OUTPUT_0
65                            , KEY_OUTPUT_1, KEY_OVERTEMPERATURE]
66
67     def __init__(self, mqtt_client, topic):
68         super().__init__(mqtt_client, topic)
69         #
70         self.output_key_delayed = None
71         self.delayed_flash_task = task.delayed(0.75, self.flash_task)
72         self.delayed_off_task = task.delayed(0.75, self.off_task)
73         #
74         self.all_off_requested = False
75
76     def __state_logging__(self, inst, key, data):
77         if key in [self.KEY_OUTPUT_0, self.KEY_OUTPUT_1]:
78             self.logger.info("State change of '%s' to '%s'", key, repr(data))
79         elif key in [self.KEY_INPUT_0, self.KEY_INPUT_1, self.KEY_LONGPUSH_0, self.KEY_LONGPUSH_1
80                     ]:
81             self.logger.info("Input action '%s' with '%s'", key, repr(data))
82
83     def flash_task(self, *args):
84         if self.flash_active:
85             self.send_command(self.output_key_delayed, not self.get(self.output_key_delayed))
86             self.output_key_delayed = None
87         if self.all_off_requested:
88             self.delayed_off_task.run()
89
90     def off_task(self, *args):
91         self.all_off()

```

```

89     @property
90     def flash_active(self):
91         return self.output_key_delayed is not None
92
93     #
94     # RX
95     #
96     @property
97     def output_0(self):
98         """rv: [True, False]"""
99         return self.get(self.KEY_OUTPUT_0)
100
101     @property
102     def output_1(self):
103         """rv: [True, False]"""
104         return self.get(self.KEY_OUTPUT_1)
105
106     @property
107     def input_0(self):
108         """rv: [True, False]"""
109         return self.get(self.KEY_INPUT_0)
110
111     @property
112     def input_1(self):
113         """rv: [True, False]"""
114         return self.get(self.KEY_INPUT_1)
115
116     @property
117     def longpush_0(self):
118         """rv: [True, False]"""
119         return self.get(self.KEY_LONGPUSH_0)
120
121     @property
122     def longpush_1(self):
123         """rv: [True, False]"""
124         return self.get(self.KEY_LONGPUSH_1)
125
126     @property
127     def temperature(self):
128         """rv: numeric value"""
129         return self.get(self.KEY_TEMPERATURE)
130
131     #
132     # TX
133     #
134     def set_output_0(self, state):
135         """state: [True, False]"""
136         self.send_command(self.KEY_OUTPUT_0, state)
137
138     def set_output_0_mcb(self, device, key, data):
139         self.set_output_0(data)
140
141     def toggle_output_0_mcb(self, device, key, data):
142         self.set_output_0(not self.output_0)
143
144     def set_output_1(self, state):
145         """state: [True, False]"""
146         self.send_command(self.KEY_OUTPUT_1, state)
147
148     def set_output_1_mcb(self, device, key, data):
149         self.set_output_1(data)
150
151     def toggle_output_1_mcb(self, device, key, data):
152         self.set_output_1(not self.output_1)

```

```

153
154     def flash_0_mcb(self, device, key, data):
155         self.output_key_delayed = self.KEY_OUTPUT_0
156         self.toggle_output_0_mcb(device, key, data)
157         self.delayed_flash_task.run()
158
159     def flash_1_mcb(self, device, key, data):
160         self.output_key_delayed = self.KEY_OUTPUT_1
161         self.toggle_output_1_mcb(device, key, data)
162         self.delayed_flash_task.run()
163
164     def __all_off__(self):
165         if self.flash_active:
166             self.all_off_requested = True
167         else:
168             if self.output_0:
169                 self.set_output_0(False)
170             if self.output_1:
171                 self.set_output_1(False)
172
173
174 class shelly_rpc(base_rpc):
175     KEY_OUTPUT_0 = "switch:0"
176     KEY_OUTPUT_1 = "switch:1"
177     KEY_OUTPUT_2 = "switch:2"
178     KEY_INPUT_0 = "input:0"
179     KEY_INPUT_1 = "input:1"
180     KEY_INPUT_2 = "input:2"
181     KEY_LONGPUSH_0 = "input:0:long_push"
182     KEY_LONGPUSH_1 = "input:1:long_push"
183     KEY_LONGPUSH_2 = "input:2:long_push"
184     KEY_SINGLEPUSH_0 = "input:0:single_push"
185     KEY_SINGLEPUSH_1 = "input:1:single_push"
186     KEY_SINGLEPUSH_2 = "input:2:single_push"
187     KEY_DOUBLEPUSH_0 = "input:0:double_push"
188     KEY_DOUBLEPUSH_1 = "input:1:double_push"
189     KEY_DOUBLEPUSH_2 = "input:2:double_push"
190     KEY_TRIPLEPUSH_0 = "input:0:triple_push"
191     KEY_TRIPLEPUSH_1 = "input:1:triple_push"
192     KEY_TRIPLEPUSH_2 = "input:2:triple_push"
193
194     RX_KEYS = [KEY_OUTPUT_0, KEY_OUTPUT_1, KEY_OUTPUT_2, KEY_INPUT_0, KEY_INPUT_1, KEY_INPUT_2,
195                KEY_LONGPUSH_0, KEY_LONGPUSH_1, KEY_LONGPUSH_2, KEY_SINGLEPUSH_0, KEY_SINGLEPUSH_1,
196                KEY_SINGLEPUSH_2,
197                KEY_DOUBLEPUSH_0, KEY_DOUBLEPUSH_1, KEY_DOUBLEPUSH_2, KEY_TRIPLEPUSH_0,
198                KEY_TRIPLEPUSH_1, KEY_TRIPLEPUSH_2]
199
200     def __state_logging__(self, inst, key, data):
201         if key in [self.KEY_OUTPUT_0, self.KEY_OUTPUT_1, self.KEY_OUTPUT_2]:
202             self.logger.info("State change of '%s' to '%s'", key, repr(data))
203         elif key in [self.KEY_INPUT_0, self.KEY_INPUT_1, self.KEY_INPUT_2]:
204             self.logger.info("Input action '%s' with '%s'", key, repr(data))
205         elif key in [self.KEY_LONGPUSH_0, self.KEY_LONGPUSH_1, self.KEY_LONGPUSH_2,
206                    self.KEY_SINGLEPUSH_0, self.KEY_SINGLEPUSH_1, self.KEY_SINGLEPUSH_2,
207                    self.KEY_DOUBLEPUSH_0, self.KEY_DOUBLEPUSH_1, self.KEY_DOUBLEPUSH_2,
208                    self.KEY_TRIPLEPUSH_0, self.KEY_TRIPLEPUSH_1, self.KEY_TRIPLEPUSH_2]:
209             self.logger.info("Input action '%s'", key)
210
211     def set_output_0(self, state):
212         """state: [True, False]"""
213         self.rpc_switch_set(self.KEY_OUTPUT_0, state)

```

## Unittest for smart\_brain

```

213 def set_output_0_mcb(self, device, key, data):
214     self.set_output_0(data)
215
216 def toggle_output_0_mcb(self, device, key, data):
217     self.set_output_0(not self.get(self.KEY_OUTPUT_0))
218
219 def set_output_1(self, state):
220     """state: [True, False]"""
221     self.rpc_switch_set(self.KEY_OUTPUT_1, state)
222
223 def set_output_1_mcb(self, device, key, data):
224     self.set_output_1(data)
225
226 def toggle_output_1_mcb(self, device, key, data):
227     print(self.get(self.KEY_OUTPUT_1))
228     self.set_output_1(not self.get(self.KEY_OUTPUT_1))
229
230 def set_output_2(self, state):
231     """state: [True, False]"""
232     self.rpc_switch_set(self.KEY_OUTPUT_2, state)
233
234 def set_output_2_mcb(self, device, key, data):
235     self.set_output_2(data)
236
237 def toggle_output_2_mcb(self, device, key, data):
238     self.set_output_2(not self.get(self.KEY_OUTPUT_2))

```

### B.2.7 devices.silvercrest.py

The line coverage for devices.silvercrest.py was 75.8%

The branch coverage for devices.silvercrest.py was 40.0%

```

1 #!/usr/bin/env python
2 # -*- coding: utf-8 -*-
3 #
4 from devices.base import base, base_output
5 import logging
6
7
8 class silvercrest_button(base):
9     """ Communication (MQTT)
10
11     tradfri_button {
12         "action": ["pressed"]
13         "battery": [0...100] %
14         "battery_low": [true | false]
15         "tamper": [true | false]
16         "linkquality": [0...255] lqi
17         "update": []
18     }
19
20     """
21     ACTION_PRESSED = "pressed"
22     #
23     KEY_LINKQUALITY = "linkquality"
24     KEY_BATTERY = "battery"
25     KEY_BATTERY_LOW = "battery_low"
26     KEY_TAMPER = "tamper"
27     KEY_ACTION = "action"
28     #
29     RX_KEYS = [KEY_LINKQUALITY, KEY_BATTERY, KEY_ACTION, KEY_BATTERY_LOW, KEY_TAMPER]

```

```

30 def __init__(self, mqtt_client, topic):
31     super().__init__(mqtt_client, topic)
32
33 def __state_logging__(self, inst, key, data):
34     if key == self.KEY_ACTION:
35         self.logger.info("Input '%s' with '%s'", key, repr(data))
36         self[self.KEY_ACTION] = None
37     elif key in [self.KEY_BATTERY_LOW, self.KEY_TAMPER]:
38         self.logger.info("Input '%s' with '%s'", key, repr(data))
39
40 #
41 # RX
42 #
43 @property
44 def action(self):
45     """rv: action_txt"""
46     return self.get(self.KEY_ACTION)
47
48
49 class silvercrest_powerplug(base_output):
50     """ Communication (MQTT)
51
52     silvercrest_powerplug {
53         |           "state": ["ON" / "OFF"]
54         |           "linkquality": [0..255] |qi
55         |       }
56         +- get {
57         |           "state": ""
58         |       }
59         +- set {
60         |           "state": ["ON" / "OFF"]
61         |       }
62     """
63     KEY_LINKQUALITY = "linkquality"
64     KEY_OUTPUT_0 = "state"
65     #
66     TX_TYPE = base.TX_DICT
67     TX_FILTER_DATA_KEYS = [KEY_OUTPUT_0]
68     #
69     RX_KEYS = [KEY_LINKQUALITY, KEY_OUTPUT_0]
70     RX_FILTER_DATA_KEYS = [KEY_OUTPUT_0]
71
72     def __state_logging__(self, inst, key, data):
73         if key in [self.KEY_OUTPUT_0]:
74             self.logger.info("State change of '%s' to '%s'", key, repr(data))
75
76 #
77 # RX
78 #
79 @property
80 def output_0(self):
81     """rv: [True, False]"""
82     return self.get(self.KEY_OUTPUT_0)
83
84 @property
85 def linkquality(self):
86     """rv: numeric value"""
87     return self.get(self.KEY_LINKQUALITY)
88
89 #
90 # TX
91 #
92 def set_output_0(self, state):

```

```

93         """state: [True, False]"""
94         self.send_command(self.KEY_OUTPUT_0, state)
95
96     def set_output_0_mcb(self, device, key, data):
97         self.set_output_0(data)
98
99     def toggle_output_0_mcb(self, device, key, data):
100         self.set_output_0(not self.output_0)
101
102     def __all_off__(self):
103         if self.output_0:
104             self.set_output_0(False)
105
106
107 class silvercrest_motion_sensor(base):
108     """ Communication (MQTT)
109
110     silvercrest_motion_sensor {
111
112         battery: [0...100] %
113         battery_low: [True, False]
114         linkquality: [0...255] lqi
115         occupancy: [True, False]
116         tamper: [True, False]
117         voltage: [0...] mV
118     }
119
120     """
121     KEY_BATTERY = "battery"
122     KEY_BATTERY_LOW = "battery_low"
123     KEY_LINKQUALITY = "linkquality"
124     KEY_OCCUPANCY = "occupancy"
125     KEY_UNMOUNTED = "tamper"
126     KEY_VOLTAGE = "voltage"
127
128     #
129     TX_TYPE = base.TX_DICT
130     #
131     RX_KEYS = [KEY_BATTERY, KEY_BATTERY_LOW, KEY_LINKQUALITY, KEY_OCCUPANCY, KEY_UNMOUNTED,
132                KEY_VOLTAGE]
133
134     def __init__(self, mqtt_client, topic):
135         super().__init__(mqtt_client, topic)
136
137     def __state_logging__(self, inst, key, data):
138         if key in [self.KEY_OCCUPANCY, self.KEY_UNMOUNTED]:
139             self.logger.info("State change of '%s' to '%s'", key, repr(data))
140
141     #
142     # RX
143     #
144     @property
145     def linkquality(self):
146         """rv: numeric value"""
147         return self.get(self.KEY_LINKQUALITY)
148
149     @property
150     def battery(self):
151         """rv: numeric value"""
152         return self.get(self.KEY_BATTERY)

```

**B.2.8** devices.tradfri.py

The line coverage for devices.tradfri.py was 86.0%

The branch coverage for devices.tradfri.py was 40.0%

```

1 #!/usr/bin/env python
2 # -*- coding: utf-8 -*-
3 #
4 from devices.base import base, base_output
5 import logging
6
7
8 class tradfri_light(base_output):
9     """ Communication (MQTT)
10
11     tradfri_light {
12         |         "state": ["ON" / "OFF" / "TOGGLE"]
13         |         "linkquality": [0...255] lqi
14         |         "brightness": [0...254]
15         |         "color_mode": ["color_temp"]
16         |         "color_temp": ["coolest", "cool", "neutral", "warm", "warmest",
17         |         250...454]
18         |         "color_temp_startup": ["coolest", "cool", "neutral", "warm", "warmest",
19         |         , "previous", 250...454]
20         |         "update": []
21         |     }
22         + get {
23         |         "state": ""
24         |     }
25         + set {
26         |         "state": ["ON" / "OFF"]
27         |         "brightness": [0...256]
28         |         "color_temp": [250...454]
29         |         "transition": [0...] seconds
30         |         "brightness_move": [-X...0...X] X/s
31         |         "brightness_step": [-X...0...X]
32         |         "color_temp_move": [-X...0...X] X/s
33         |         "color_temp_step": [-X...0...X]
34         |     }
35     """
36     KEY_LINKQUALITY = "linkquality"
37     KEY_OUTPUT_0 = "state"
38     KEY_BRIGHTNESS = "brightness"
39     KEY_COLOR_TEMP = "color_temp"
40     KEY_BRIGHTNESS_FADE = "brightness_move"
41     #
42     TX_TYPE = base.TX_DICT
43     TX_FILTER_DATA_KEYS = [KEY_OUTPUT_0, KEY_BRIGHTNESS, KEY_COLOR_TEMP, KEY_BRIGHTNESS_FADE]
44     #
45     RX_KEYS = [KEY_LINKQUALITY, KEY_OUTPUT_0, KEY_BRIGHTNESS, KEY_COLOR_TEMP]
46     RX_IGNORE_KEYS = ['update', 'color_mode', 'color_temp_startup']
47     RX_FILTER_DATA_KEYS = [KEY_OUTPUT_0, KEY_BRIGHTNESS, KEY_COLOR_TEMP]
48
49     def state_logging(self, inst, key, data):
50         if key in [self.KEY_OUTPUT_0, self.KEY_BRIGHTNESS, self.KEY_COLOR_TEMP, self.
51         KEY_BRIGHTNESS_FADE]:
52             self.logger.info("State change of '%s' to '%s'", key, repr(data))
53
54     def __device_to_instance_filter__(self, key, data):
55         if key == self.KEY_BRIGHTNESS:
56             return int(round((data - 1) * 100 / 253, 0))
57         elif key == self.KEY_COLOR_TEMP:
58             return int(round((data - 250) * 10 / 204, 0))
59         return super().__device_to_instance_filter__(key, data)

```

```

57
58     def __instance_to_device_filter__(self, key, data):
59         if key == self.KEY_BRIGHTNESS:
60             return int(round(data * 253 / 100 + 1, 0))
61         elif key == self.KEY_COLOR_TEMP:
62             return int(round(data * 204 / 10 + 250, 0))
63         return super().__instance_to_device_filter__(key, data)
64
65     #
66     # RX
67     #
68     @property
69     def output_0(self):
70         """rv: [True, False]"""
71         return self.get(self.KEY_OUTPUT_0, False)
72
73     @property
74     def linkquality(self):
75         """rv: numeric value"""
76         return self.get(self.KEY_LINKQUALITY, 0)
77
78     @property
79     def brightness(self):
80         """rv: numeric value [0%, ..., 100%]"""
81         return self.get(self.KEY_BRIGHTNESS, 0)
82
83     @property
84     def color_temp(self):
85         """rv: numeric value [0, ..., 10]"""
86         return self.get(self.KEY_COLOR_TEMP, 0)
87
88     #
89     # TX
90     #
91     def request_data(self, device=None, key=None, data=None):
92         self.mqtt_client.send(self.topic + "/get", '{"%s": ""}' % self.KEY_OUTPUT_0)
93
94     def set_output_0(self, state):
95         """state: [True, False]"""
96         self.send_command(self.KEY_OUTPUT_0, state)
97
98     def set_output_0_mcb(self, device, key, data):
99         self.set_output_0(data)
100
101     def toggle_output_0_mcb(self, device, key, data):
102         self.set_output_0(not self.output_0)
103
104     def set_brightness(self, brightness):
105         """brightness: [0, ..., 100]"""
106         self.send_command(self.KEY_BRIGHTNESS, brightness)
107
108     def set_brightness_mcb(self, device, key, data):
109         self.set_brightness(data)
110
111     def default_inc(self, speed=40):
112         self.send_command(self.KEY_BRIGHTNESS_FADE, speed)
113
114     def default_dec(self, speed=-40):
115         self.default_inc(speed)
116
117     def default_stop(self):
118         self.default_inc(0)
119
120     def set_color_temp(self, color_temp):

```

```

121         """color_temp: [0, ..., 10]"""
122         self.send_command(self.KEY_COLOR_TEMP, color_temp)
123
124     def set_color_temp_mcb(self, device, key, data):
125         self.set_color_temp(data)
126
127     def __all_off__(self):
128         if self.output_0:
129             self.set_output_0(False)
130
131
132 class tradfri_button(base):
133     """ Communication (MQTT)
134
135     tradfri_button {
136         "action": [
137             "arrow_left_click",
138             "arrow_left_hold",
139             "arrow_left_release",
140             "arrow_right_click",
141             "arrow_right_hold",
142             "arrow_right_release",
143             "brightness_down_click",
144             "brightness_down_hold",
145             "brightness_down_release",
146             "brightness_up_click",
147             "brightness_up_hold",
148             "brightness_up_release",
149             "toggle"
150         ]
151         "action_duration": [0...] s
152         "battery": [0...100] %
153         "linkquality": [0...255] lqi
154         "update": []
155     }
156 """
157 ACTION_TOGGLE = "toggle"
158 ACTION_BRIGHTNESS_UP = "brightness_up_click"
159 ACTION_BRIGHTNESS_DOWN = "brightness_down_click"
160 ACTION_RIGHT = "arrow_right_click"
161 ACTION_LEFT = "arrow_left_click"
162 ACTION_BRIGHTNESS_UP_LONG = "brightness_up_hold"
163 ACTION_BRIGHTNESS_UP_RELEASE = "brightness_up_release"
164 ACTION_BRIGHTNESS_DOWN_LONG = "brightness_down_hold"
165 ACTION_BRIGHTNESS_DOWN_RELEASE = "brightness_down_release"
166 ACTION_RIGHT_LONG = "arrow_right_hold"
167 ACTION_RIGHT_RELEASE = "arrow_right_release"
168 ACTION_LEFT_LONG = "arrow_left_hold"
169 ACTION_LEFT_RELEASE = "arrow_left_release"
170 #
171 KEY_LINKQUALITY = "linkquality"
172 KEY_BATTERY = "battery"
173 KEY_ACTION = "action"
174 KEY_ACTION_DURATION = "action_duration"
175 #
176 RX_KEYS = [KEY_LINKQUALITY, KEY_BATTERY, KEY_ACTION]
177 RX_IGNORE_KEYS = ['update', KEY_ACTION_DURATION]
178
179     def __init__(self, mqtt_client, topic):
180         super().__init__(mqtt_client, topic)
181
182     def __state_logging__(self, inst, key, data):

```

```

183         if key in [self.KEY_ACTION]:
184             self.logger.info("Input '%s' with '%s'", key, repr(data))
185
186         #
187         # RX
188         #
189         @property
190         def action(self):
191             """rv: action_txt"""
192             return self.get(self.KEY_ACTION)

```

### B.3 function

The line coverage for function was 84.8%

The branch coverage for function was 46.8%

#### B.3.1 function.\_\_init\_\_.py

The line coverage for function.\_\_init\_\_.py was 87.3%

The branch coverage for function.\_\_init\_\_.py was 46.8%

```

1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3  #
4  import config
5  from devdi.topic import STOP_EXECUTION_TOPIC
6  import devices
7  from function.garden import garden
8  from function.stairway import stairway
9  from function.ground_floor_west import ground_floor_west
10 from function.first_floor_west import first_floor_west
11 from function.first_floor_east import first_floor_east
12 from function.rooms import room_collection
13 from function.videv import all_off, videv_pure_switch
14 import json
15 import logging
16 import mqtt
17
18 try:
19     from config import APP_NAME as ROOT_LOGGER_NAME
20 except ImportError:
21     ROOT_LOGGER_NAME = 'root'
22 logger = logging.getLogger(ROOT_LOGGER_NAME).getChild(__name__)
23
24
25 class all_functions(room_collection):
26     def __init__(self, mqtt_client: mqtt.mqtt_client):
27         super().__init__(mqtt_client)
28         #
29         self.run = True
30         if config.DEBUG:
31             mqtt_client.add_callback(STOP_EXECUTION_TOPIC, self.__stop_execution__)
32         #
33         # Rooms
34         #
35         # garden
36         self.gar = garden(self.mqtt_client)
37         # stairway

```

```

38     self.stw = stairway(self.mqtt_client)
39     # ground floor west
40     self.gfw = ground_floor_west(self.mqtt_client)
41     # first floor west
42     self.ffw = first_floor_west(self.mqtt_client)
43     # first floor east
44     self.ffe = first_floor_east(self.mqtt_client)
45     #
46     # Interactions
47     #
48     # cross_room_interactions
49     self.init_cross_room_interactions()
50     # Off Buttons
51     self.init_off_functionality()
52     # Summer / Winter mode
53     self.init_sumer_winter_mode()
54
55     def __stop_execution__(self, client, userdata, message):
56         if config.DEBUG:
57             try:
58                 data = json.loads(message.payload)
59             except:
60                 logger.error("Error while receiving mqtt message: topic=%s - payload=%s", repr(
61                     message.topic), repr(message.payload))
62             else:
63                 if data is True:
64                     self.run = False
65
66     def init_cross_room_interactions(self):
67         # shelly dirk input 1
68         self.last_gfw_dirk_input_1 = None
69         self.gfw_dirk.switch_main_light.add_callback(self.gfw_dirk.switch_main_light.KEY_INPUT_1,
70             None, self.gfw_dirk_input_1)
71         # tradfri button ffe_sleep_right_click
72         self.ffe_sleep.input_device.add_callback(self.ffe_sleep.input_device.KEY_ACTION,
73             self.ffe_sleep.input_device.ACTION_RIGHT, self.
74             ffe.floor.switch_main_light.toggle_output_0_mcb)
75
76     def init_off_functionality(self):
77         # ALL OFF - Virtual device
78         self.videv_all_off = all_off(self.mqtt_client, config.TOPIC_ALL_OFF_VIDEV, self)
79
80         # ALL OFF - Long push stairway
81         self.stw.stairway.switch_main_light.add_callback(self.stw.stairway.switch_main_light.
82             KEY_LONGPUSH_0,
83             True, self.stw.stairway.
84             switch_main_light.flash_0_mcb)
85         self.stw.stairway.switch_main_light.add_callback(self.stw.stairway.switch_main_light.
86             KEY_LONGPUSH_0, True, self.all_off)
87
88         # FFE ALL OFF - Long push ffe_floor
89         self.ffe.floor.switch_main_light.add_callback(self.ffe.floor.switch_main_light.
90             KEY_LONGPUSH_0,
91             True, self.ffe.floor.switch_main_light.
92             flash_0_mcb)
93         self.ffe.floor.switch_main_light.add_callback(self.ffe.floor.switch_main_light.
94             KEY_LONGPUSH_0, True, self.ffe.all_off)
95
96         # FFE ALL OFF - Long push input device
97         self.ffe_sleep.input_device.add_callback(devices.tradfri_button.KEY_ACTION, devices.
98             tradfri_button.ACTION_RIGHT_LONG, self.ffe.all_off)
99
100         # FFW ALL OFF - Long push ffw_floor

```

```

91     self.ffw.floor.switch_main_light.add_callback(self.ffw.floor.switch_main_light.
KEY_LONGPUSH_0,
92                                                     True, self.ffw.floor.switch_main_light.
flash_0_mcb)
93     self.ffw.floor.switch_main_light.add_callback(self.ffw.floor.switch_main_light.
KEY_LONGPUSH_0, True, self.ffw.all_off)
94
95     def init_sumer_winter_mode(self):
96         # ALL summer/winter mode
97         self.videv_summer_mode = videv_pure_switch(self.mqtt_client, config.
TOPIC_ALL_SUMMER_WINTER_MODE)
98
99         self.videv_summer_mode.add_callback(self.videv_summer_mode.KEY_STATE, None, self.gfw.
summer_mode)
100         self.videv_summer_mode.add_callback(self.videv_summer_mode.KEY_STATE, None, self.ffw.
summer_mode)
101         self.videv_summer_mode.add_callback(self.videv_summer_mode.KEY_STATE, None, self.ffe.
summer_mode)
102
103     def gfw_dirk_input_1(self, device, key, data):
104         if self.last_gfw_dirk_input_1 is not None:
105             if self.last_gfw_dirk_input_1 != data:
106                 self.gfw.floor.switch_main_light.toggle_output_0_mcb(device, key, data)
107                 self.last_gfw_dirk_input_1 = data

```

### B.3.2 function.db.py

The line coverage for function.db.py was 97.7%

The branch coverage for function.db.py was 46.8%

```

1 from function.modules import heating_function
2 import os
3 import sqlite3
4
5 db_file = os.path.join(os.path.dirname(__file__), '..', 'database.db')
6
7 db_mapping_radiator = {
8     0: heating_function.KEY_AWAY_MODE,
9     1: heating_function.KEY_SUMMER_MODE,
10    2: heating_function.KEY_USER_TEMPERATURE_SETPPOINT,
11    3: heating_function.KEY_TEMPERATURE_SETPPOINT
12 }
13
14
15 def get_radiator_data(topic):
16     db_data = __storage__.get_radiator_data(topic)
17     rv = {}
18     for index in db_mapping_radiator:
19         rv[db_mapping_radiator[index]] = db_data[index]
20     return rv
21
22
23 def set_radiator_data(device, key, data):
24     if key in db_mapping_radiator.values():
25         db_data = []
26         for index in range(0, len(db_mapping_radiator)):
27             db_data.append(device.get(db_mapping_radiator[index]))
28         return __storage__.store_radiator_data(device.heating_valve.topic, db_data)
29
30

```

```

31 class __storage__(object):
32     def __init__(self):
33         self.conn = sqlite3.connect(db_file)
34         self.c = self.conn.cursor()
35         with self.conn:
36             self.c.execute("""CREATE TABLE IF NOT EXISTS radiator (
37                 topic text PRIMARY KEY,
38                 away_mode integer,
39                 summer_mode integer,
40                 user_temperatur_setpoint real,
41                 temperatur_setpoint real
42             )""")
43
44     def store_radiator_data(self, topic, target_data):
45         try:
46             with self.conn:
47                 self.c.execute(
48                     'INSERT INTO radiator VALUES (?, ?, ?, ?, ?)', [topic] + target_data)
49         except sqlite3.IntegrityError:
50             db_data = self.get_radiator_data(topic)
51             if db_data != target_data:
52                 with self.conn:
53                     self.c.execute(
54                         'UPDATE radiator SET away_mode = ?, summer_mode = ?,
55                         user_temperatur_setpoint = ?, temperatur_setpoint = ? WHERE topic = ?', target_data + [topic
56 ])
57
58     def get_radiator_data(self, topic):
59         """ returns a list [away_mode, summer_mode, user_temperatur_setpoint, temperatur_setpoint
60         ] or [None, None, None, None]"""
61         self.c.execute("SELECT * FROM radiator WHERE topic=?", (topic, ))
62         data = self.c.fetchone()
63         if data is not None:
64             data = list(data)
65             data[1] = data[1] == 1
66             data[2] = data[2] == 1
67             return data[1:]
68         else:
69             return [None, None, None, None]
70
71     def __del__(self):
72         self.conn.close()

```

### B.3.3 function.first\_floor\_east.py

The line coverage for function.first\_floor\_east.py was 92.0%

The branch coverage for function.first\_floor\_east.py was 46.8%

```

1 #!/usr/bin/env python
2 # -*- coding: utf-8 -*-
3 #
4
5 import config
6 from devdi import rooms
7 from function.db import get_radiator_data, set_radiator_data
8 from function.helpers import day_event
9 from function.modules import brightness_choose_n_action, timer_on_activation, heating_function,
10 switched_light
11 from function.rooms import room, room_collection
12 import logging

```

```

12
13 try:
14     from config import APP_NAME as ROOT_LOGGER_NAME
15 except ImportError:
16     ROOT_LOGGER_NAME = 'root'
17 logger = logging.getLogger(ROOT_LOGGER_NAME).getChild(__name__)
18
19
20 class first_floor_east(room_collection):
21     def __init__(self, mqtt_client):
22         super().__init__(mqtt_client)
23         self.dining = first_floor_east_dining(mqtt_client)
24         self.floor = first_floor_east_floor(mqtt_client)
25         self.kitchen = first_floor_east_kitchen(mqtt_client)
26         self.livingroom = first_floor_east_living(mqtt_client)
27         self.sleep = first_floor_east_sleep(mqtt_client)
28
29
30 class first_floor_east_floor(rooms.ffe_floor, room):
31     def __init__(self, mqtt_client):
32         super().__init__(mqtt_client)
33         room.__init__(self, mqtt_client)
34         #
35         # connect videv and switch
36         self.videv_main_light.connect_sw_device(self.switch_main_light, self.switch_main_light.
KEY_OUTPUT_0)
37
38
39 class first_floor_east_kitchen(rooms.ffe_kitchen, room):
40     def __init__(self, mqtt_client):
41         super().__init__(mqtt_client)
42         room.__init__(self, mqtt_client)
43         #
44         # light <-> videv
45         self.videv_main_light.connect_sw_device(self.switch_main_light, self.switch_main_light.
KEY_OUTPUT_0)
46         self.videv_main_light.connect_br_device(self.light_main_light, self.light_main_light.
KEY_BRIGHTNESS)
47         self.videv_main_light.connect_ct_device(self.light_main_light, self.light_main_light.
KEY_COLOR_TEMP)
48
49         # Request hue data of lead light after power on
50         switched_light(self.switch_main_light, self.switch_main_light.KEY_OUTPUT_0, self.
light_main_light)
51
52         # circulation pump
53         self.circulation_pump = timer_on_activation(self.switch_circulation_pump, self.
switch_circulation_pump.KEY_OUTPUT_0, 10*60)
54         self.switch_circulation_pump.add_callback(self.switch_circulation_pump.KEY_OUTPUT_0, True
, self.switch_main_light.flash_0_mcb, True)
55         self.videv_circulation_pump.connect_sw_device(self.switch_circulation_pump, self.
switch_circulation_pump.KEY_OUTPUT_0)
56         self.videv_circulation_pump.connect_tm_device(self.circulation_pump, timer_on_activation.
KEY_TIMER)
57
58         # heating function
59         self.heating_function = heating_function(
60             self.valve_heating,
61             config.DEFAULT_TEMPERATURE,
62             **get_radiator_data(self.valve_heating.topic)
63         )

```

```

64     self.heating_function.add_callback(None, None, set_radiator_data, True)
65     self.videv_heating.connect_heating_function(self.heating_function)
66
67
68 class first_floor_east_dining(rooms.ffe_diningroom, room):
69     def __init__(self, mqtt_client):
70         super().__init__(mqtt_client)
71         room.__init__(self, mqtt_client)
72         #
73         self.day_events = day_event((6, 0), (22, 0), 30, -30)
74         self.day_events.add_callback(None, True, self.__day_events__, True)
75
76         # light <=> videv
77         self.videv_main_light.connect_sw_device(self.switch_main_light, self.switch_main_light.
KEY_OUTPUT_0)
78         self.videv_floor_light.connect_sw_device(self.switch_floor_light, self.switch_floor_light.
KEY_OUTPUT_0)
79         if config.CHRISTMAS:
80             self.videv_garland_light.connect_sw_device(self.switch_garland_light, self.
switch_garland_light.KEY_OUTPUT_0)
81
82         # main light -> floor_light
83         self.switch_main_light.add_callback(self.switch_main_light.KEY_OUTPUT_0, None, self.
switch_floor_light.set_output_0_mcb, True)
84
85         # heating function
86         self.heating_function = heating_function(
87             self.valve_heating,
88             config.DEFAULT_TEMPERATURE,
89             **get_radiator_data(self.valve_heating.topic)
90         )
91         self.heating_function.add_callback(None, None, set_radiator_data, True)
92         # heating function <=> videv
93         self.videv_heating.connect_heating_function(self.heating_function)
94
95     def __day_events__(self, device, key, data):
96         if key in (self.day_events.KEY_SUNSET, self.day_events.KEY_START_OF_DAY):
97             if config.CHRISTMAS:
98                 self.switch_garland_light.set_output_0(True)
99         elif key in (self.day_events.KEY_START_OF_NIGHT, self.day_events.KEY_SUNRISE):
100             if config.CHRISTMAS:
101                 self.switch_garland_light.set_output_0(False)
102
103
104 class first_floor_east_sleep(rooms.ffe_sleep, room):
105     def __init__(self, mqtt_client):
106         super().__init__(mqtt_client)
107         room.__init__(self, mqtt_client)
108         #
109         self.light_wardrobe_light.disable_all_off() # Always on - Off by light sensor
110         # light <=> videv
111         self.videv_main_light.connect_sw_device(self.switch_main_light, self.switch_main_light.
KEY_OUTPUT_0)
112         self.videv_main_light.connect_br_device(self.light_main_light, self.light_main_light.
KEY_BRIGHTNESS)
113         self.videv_main_light.connect_ct_device(self.light_main_light, self.light_main_light.
KEY_COLOR_TEMP)
114         #
115         self.videv_bed_dirk_light.connect_sw_device(self.light_bed_dirk_light, self.
light_bed_dirk_light.KEY_OUTPUT_0)
116         self.videv_bed_dirk_light.connect_br_device(self.light_bed_dirk_light, self.
light_bed_dirk_light.KEY_BRIGHTNESS)
117         #

```

```

118     self.videv_bed_marion_light.connect_sw_device(self.switch_bed_marion_light, self.
        switch_bed_marion_light.KEY_OUTPUT_0)
119     #
120     self.videv_wardrobe_light.connect_sw_device(self.light_wardrobe_light, self.
        light_wardrobe_light.KEY_OUTPUT_0)
121     self.videv_wardrobe_light.connect_br_device(self.light_wardrobe_light, self.
        light_wardrobe_light.KEY_BRIGHTNESS)
122
123     # button / brightness function
124     self.brightness_functions = brightness_choose_n_action(self.input_device)
125     self.brightness_functions.add(self.light_main_light, self.switch_main_light, self.
        switch_main_light.KEY_OUTPUT_0)
126     self.brightness_functions.add(self.light_bed_dirk_light, self.light_bed_dirk_light, self.
        light_bed_dirk_light.KEY_OUTPUT_0)
127     # button / main light
128     self.input_device.add_callback(self.input_device.KEY_ACTION, self.input_device.
        ACTION_TOGGLE, self.switch_main_light.toggle_output_0_mcb)
129     # button / bed light
130     self.input_device.add_callback(self.input_device.KEY_ACTION, self.input_device.
        ACTION_LEFT, self.light_bed_dirk_light.toggle_output_0_mcb)
131     self.input_device.add_callback(self.input_device.KEY_ACTION, self.input_device.
        ACTION_LEFT_LONG,
132                                     self.switch_bed_marion_light.toggle_output_0_mcb)
133     # button
134     self.videv_multistate.connect_br_function(self.brightness_functions,
        brightness_choose_n_action.KEY_ACTIVE_DEVICE, 2)
135
136     # heating function
137     self.heating_function = heating_function(
138         self.valve_heating,
139         config.DEFAULT_TEMPERATURE,
140         **get_radiator_data(self.valve_heating.topic)
141     )
142     self.heating_function.add_callback(None, None, set_radiator_data, True)
143     self.videv_heating.connect_heating_function(self.heating_function)
144
145
146 class first_floor_east_living(rooms.ffe_livingroom, room):
147     def __init__(self, mqtt_client):
148         super().__init__(mqtt_client)
149         room.__init__(self, mqtt_client)
150         #
151         # light <-> videv
152         self.videv_main_light.connect_sw_device(self.switch_main_light, self.switch_main_light.
            KEY_OUTPUT_0)
153         self.videv_main_light.connect_br_device(self.light_main_light, self.light_main_light.
            KEY_BRIGHTNESS)
154         self.videv_main_light.connect_ct_device(self.light_main_light, self.light_main_light.
            KEY_COLOR_TEMP)
155         #
156         self.videv_floor_light.connect_sw_device(self.light_floor_light, self.light_floor_light.
            KEY_OUTPUT_0)
157         self.videv_floor_light.connect_br_device(self.light_floor_light, self.light_floor_light.
            KEY_BRIGHTNESS)
158         self.videv_floor_light.connect_ct_device(self.light_floor_light, self.light_floor_light.
            KEY_COLOR_TEMP)
159         #
160         if config.CHRISTMAS:
161             self.videv_xmas_tree_light.connect_sw_device(self.switch_xmas_tree_light, self.
                switch_xmas_tree_light.KEY_OUTPUT_0)
162
163         # main light -> floor_light

```

```

164     self.switch_main_light.add_callback(self.switch_main_light.KEY_OUTPUT_0, None, self.
        light_floor_light.set_output_0_mcb, True)
165
166     # heating function
167     self.heating_function = heating_function(
168         self.valve_heating,
169         config.DEFAULT_TEMPERATURE,
170         **get_radiator_data(self.valve_heating.topic)
171     )
172     self.heating_function.add_callback(None, None, set_radiator_data, True)
173     self.videv_heating.connect_heating_function(self.heating_function)

```

### B.3.4 function.first\_floor\_west.py

The line coverage for function.first\_floor\_west.py was 96.9%

The branch coverage for function.first\_floor\_west.py was 46.8%

```

1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3  #
4
5  import config
6  from devdi import rooms
7  from function.db import get_radiator_data, set_radiator_data
8  from function.modules import heating_function
9  from function.rooms import room, room_collection
10 import logging
11
12
13 try:
14     from config import APP_NAME as ROOT_LOGGER_NAME
15 except ImportError:
16     ROOT_LOGGER_NAME = 'root'
17 logger = logging.getLogger(ROOT_LOGGER_NAME).getChild(__name__)
18
19
20 class first_floor_west(room_collection):
21     def __init__(self, mqtt_client):
22         super().__init__(mqtt_client)
23         self.floor = first_floor_west_floor(mqtt_client)
24         self.bath = first_floor_west_bath(mqtt_client)
25         self.julian = first_floor_west_julian(mqtt_client)
26         self.livingroom = first_floor_west_living(mqtt_client)
27         self.sleep = first_floor_west_sleep(mqtt_client)
28
29
30 class first_floor_west_floor(rooms.ffw_floor, room):
31     def __init__(self, mqtt_client):
32         super().__init__(mqtt_client)
33         room.__init__(self, mqtt_client)
34         #
35         # connect videv and switch
36         self.videv_main_light.connect_sw_device(self.switch_main_light, self.switch_main_light.
            KEY_OUTPUT_0)
37
38
39 class first_floor_west_julian(rooms.ffw_julian, room):
40     def __init__(self, mqtt_client):
41         super().__init__(mqtt_client)
42         room.__init__(self, mqtt_client)

```

```

43     #
44     # light <=> videv
45     self.videv_main_light.connect_sw_device(self.switch_main_light, self.switch_main_light.
KEY_OUTPUT_0)
46     self.videv_main_light.connect_br_device(self.light_main_light, self.light_main_light.
KEY_BRIGHTNESS)
47     self.videv_main_light.connect_ct_device(self.light_main_light, self.light_main_light.
KEY_COLOR_TEMP)
48
49     # heating function
50     self.heating_function = heating_function(
51         self.valve_heating,
52         config.DEFAULT_TEMPERATURE,
53         **get_radiator_data(self.valve_heating.topic)
54     )
55     self.heating_function.add_callback(None, None, set_radiator_data, True)
56     self.videv_heating.connect_heating_function(self.heating_function)
57
58
59 class first_floor_west_bath(rooms.ffw_bath, room):
60     def __init__(self, mqtt_client):
61         super().__init__(mqtt_client)
62         room.__init__(self, mqtt_client)
63     #
64     # light <=> videv
65     self.videv_main_light.connect_sw_device(self.switch_main_light, self.switch_main_light.
KEY_OUTPUT_0)
66
67     # heating function
68     self.heating_function = heating_function(
69         self.valve_heating,
70         config.DEFAULT_TEMPERATURE,
71         **get_radiator_data(self.valve_heating.topic)
72     )
73     self.heating_function.add_callback(None, None, set_radiator_data, True)
74     self.videv_heating.connect_heating_function(self.heating_function)
75
76
77 class first_floor_west_living(rooms.ffw_livingroom, room):
78     def __init__(self, mqtt_client):
79         super().__init__(mqtt_client)
80         room.__init__(self, mqtt_client)
81     #
82     # light <=> videv
83     self.videv_main_light.connect_sw_device(self.switch_main_light, self.switch_main_light.
KEY_OUTPUT_0)
84     self.videv_main_light.connect_br_device(self.light_main_light, self.light_main_light.
KEY_BRIGHTNESS)
85     self.videv_main_light.connect_ct_device(self.light_main_light, self.light_main_light.
KEY_COLOR_TEMP)
86
87     # heating function
88     self.heating_function = heating_function(
89         self.valve_heating,
90         config.DEFAULT_TEMPERATURE,
91         **get_radiator_data(self.valve_heating.topic)
92     )
93     self.heating_function.add_callback(None, None, set_radiator_data, True)
94     self.videv_heating.connect_heating_function(self.heating_function)
95
96

```

```

97 class first_floor_west_sleep(rooms.ffw_sleep, room):
98     def __init__(self, mqtt_client):
99         super().__init__(mqtt_client)
100         room.__init__(self, mqtt_client)
101         #
102         # light <=> videv
103         self.videv_main_light.connect_sw_device(self.switch_main_light, self.switch_main_light.
KEY_OUTPUT_0)
104         self.videv_main_light.connect_br_device(self.light_main_light, self.light_main_light.
KEY_BRIGHTNESS)
105         #
106         self.videv_window_light.connect_sw_device(self.light_window_light, self.
light_window_light.KEY_OUTPUT_0)
107         self.videv_window_light.connect_br_device(self.light_window_light, self.
light_window_light.KEY_BRIGHTNESS)
108         self.videv_window_light.connect_ct_device(self.light_window_light, self.
light_window_light.KEY_COLOR_TEMP)
109
110         # main light -> window light
111         self.switch_main_light.add_callback(self.switch_main_light.KEY_OUTPUT_0, None, self.
light_window_light.set_output_0_mcb, True)
112
113         # heating function
114         self.heating_function = heating_function(
115             self.valve_heating,
116             config.DEFAULT_TEMPERATURE,
117             **get_radiator_data(self.valve_heating.topic)
118         )
119         self.heating_function.add_callback(None, None, set_radiator_data, True)
120         self.videv_heating.connect_heating_function(self.heating_function)

```

### B.3.5 function.garden.py

The line coverage for function.garden.py was 74.1%

The branch coverage for function.garden.py was 46.8%

```

1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3  #
4
5  from devdi import rooms
6  from function.helpers import day_event
7  from function.rooms import room, room_collection
8  import logging
9
10 try:
11     from config import APP_NAME as ROOT_LOGGER_NAME
12 except ImportError:
13     ROOT_LOGGER_NAME = 'root'
14 logger = logging.getLogger(ROOT_LOGGER_NAME).getChild(__name__)
15
16
17 class garden(room_collection):
18     def __init__(self, mqtt_client):
19         super().__init__(mqtt_client)
20         self.garden = garden_garden(mqtt_client)
21
22

```

```

23 class garden_garden(rooms.gar_garden, room):
24     def __init__(self, mqtt_client):
25         super().__init__(mqtt_client)
26         room.__init__(self, mqtt_client)
27         #
28         self.day_events = day_event((6, 0), (22, 0), 30, -30)
29         self.day_events.add_callback(None, True, self.__day_events__, True)
30
31         # xxx <=> videv
32         self.videv_garland_light.connect_sw_device(self.switch_garland_light, self.
switch_garland_light.KEY_OUTPUT_0)
33         self.videv_repeater.connect_sw_device(self.switch_repeater, self.switch_repeater.
KEY_OUTPUT_0)
34
35     def __day_events__(self, device, key, data):
36         if self.videv_mode.get(self.videv_mode.KEY_STATE):
37             if key in (self.day_events.KEY_SUNSET, self.day_events.KEY_START_OF_DAY):
38                 self.switch_garland_light.set_output_0(True)
39             elif key in (self.day_events.KEY_START_OF_NIGHT, self.day_events.KEY_SUNRISE):
40                 self.switch_garland_light.set_output_0(False)

```

### B.3.6 function.ground\_floor\_west.py

The line coverage for function.ground\_floor\_west.py was 93.4%

The branch coverage for function.ground\_floor\_west.py was 46.8%

```

1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3  #
4
5  import config
6  from devdi import rooms
7  from function.db import get_radiator_data, set_radiator_data
8  from function.modules import brightness_choose_n_action, heating_function, switched_light
9  from function.rooms import room, room_collection
10 import logging
11 import task
12
13 try:
14     from config import APP_NAME as ROOT_LOGGER_NAME
15 except ImportError:
16     ROOT_LOGGER_NAME = 'root'
17 logger = logging.getLogger(ROOT_LOGGER_NAME).getChild(__name__)
18
19
20 class ground_floor_west(room_collection):
21     def __init__(self, mqtt_client):
22         super().__init__(mqtt_client)
23         self.dirk = ground_floor_west_dirk(mqtt_client)
24         self.floor = ground_floor_west_floor(mqtt_client)
25         self.marion = ground_floor_west_marion(mqtt_client)
26
27
28 class ground_floor_west_dirk(rooms.gfw_dirk, room):
29     STATE_ACTIVE_DEVICE_MAIN_LIGHT = 0
30     STATE_ACTIVE_DEVICE_DESK_LIGHT = 1
31     STATE_ACTIVE_DEVICE_AMPLIFIER = 2
32     STATE_ACTIVE_DEVICE_MAX_VALUE = STATE_ACTIVE_DEVICE_AMPLIFIER
33     #

```

```

34     AUDIO_SOURCE_PC = 0
35     AUDIO_SOURCE_CD = 1
36     AUDIO_SOURCE_RASPI = 2
37     AUDIO_SOURCE_BT = 3
38     AUDIO_SOURCE_PHONO = 4
39
40     def __init__(self, mqtt_client):
41         super().__init__(mqtt_client)
42         room.__init__(self, mqtt_client)
43         #
44         # light <-> videv
45         self.videv_main_light.connect_sw_device(self.switch_main_light, self.switch_main_light,
46 KEY_OUTPUT_0)
47         self.videv_main_light.connect_br_device(self.light_main_light, self.light_main_light,
48 KEY_BRIGHTNESS)
49         self.videv_main_light.connect_ct_device(self.light_main_light, self.light_main_light,
50 KEY_COLOR_TEMP)
51         #
52         self.videv_desk_light.connect_sw_device(self.light_desk_light, self.light_desk_light,
53 KEY_OUTPUT_0)
54         self.videv_desk_light.connect_br_device(self.light_desk_light, self.light_desk_light,
55 KEY_BRIGHTNESS)
56         self.videv_desk_light.connect_ct_device(self.light_desk_light, self.light_desk_light,
57 KEY_COLOR_TEMP)
58         #
59         self.videv_amplifier.connect_sw_device(self.switch_powerplug_4, self,
60 KEY_POWERPLUG_AMPLIFIER)
61         self.videv_bluetooth.connect_sw_device(self.switch_powerplug_4, self.KEY_POWERPLUG_BT)
62         self.videv_cd_player.connect_sw_device(self.switch_powerplug_4, self,
63 KEY_POWERPLUG_CD_PLAYER)
64         self.videv_phono.connect_sw_device(self.switch_powerplug_4, self.KEY_POWERPLUG_PHONO)
65         #
66         self.videv_pc_dock.connect_sw_device(self.switch_pc_dock, self.switch_pc_dock,
67 KEY_OUTPUT_0)
68
69         # amplifier on, if playing device on
70         self.switch_powerplug_4.add_callback(self.KEY_POWERPLUG_PHONO, None, self,
71 switch_powerplug_4.set_output_0_mcb, True)
72         self.switch_powerplug_4.add_callback(self.KEY_POWERPLUG_CD_PLAYER, None, self,
73 switch_powerplug_4.set_output_0_mcb, True)
74         self.switch_powerplug_4.add_callback(self.KEY_POWERPLUG_BT, None, self.switch_powerplug_4,
75 .set_output_0_mcb, True)
76         # amplifier on, if player on
77         self.audio_status_bluetooth.add_callback(self.audio_status_bluetooth.KEY_STATE, None,
78 self.switch_powerplug_4.set_output_0_mcb, True)
79         self.audio_status_mpd.add_callback(self.audio_status_mpd.KEY_STATE, None, self,
80 switch_powerplug_4.set_output_0_mcb, True)
81         self.audio_status_spotify.add_callback(self.audio_status_spotify.KEY_STATE, None, self,
82 switch_powerplug_4.set_output_0_mcb, True)
83
84         # Audio source selection
85         self.switch_powerplug_4.add_callback(self.KEY_POWERPLUG_AMPLIFIER, True, self,
86 audio_source_selector, True)
87         self.switch_powerplug_4.add_callback(self.KEY_POWERPLUG_CD_PLAYER, True, self,
88 audio_source_selector, True)
89         self.switch_powerplug_4.add_callback(self.KEY_POWERPLUG_BT, True, self,
90 audio_source_selector, True)
91         self.switch_powerplug_4.add_callback(self.KEY_POWERPLUG_PHONO, True, self,
92 audio_source_selector, True)
93         self.audio_status_bluetooth.add_callback(self.audio_status_bluetooth.KEY_STATE, True,
94 self.audio_source_selector, True)

```

```

75     self.audio_status_mpd.add_callback(self.audio_status_mpd.KEY_STATE, True, self.
audio_source_selector, True)
76     self.audio_status_spotify.add_callback(self.audio_status_spotify.KEY_STATE, True, self.
audio_source_selector, True)
77     self.audio_source = self.AUDIO_SOURCE_PC
78     self.delayed_task_remote = task.delayed(1.0, self.send_audio_source)
79
80     # input device functions
81     # Brightness functionality
82     self.brightness_functions = brightness_choose_n_action(self.input_device)
83     self.brightness_functions.add(self.light_main_light, self.switch_main_light, self.
switch_main_light.KEY_OUTPUT_0)
84     self.brightness_functions.add(self.light_desk_light, self.light_desk_light, self.
light_desk_light.KEY_OUTPUT_0)
85     self.brightness_functions.add(self.remote_ctrl, self.switch_powerplug_4, self.
KEY_POWERPLUG_AMPLIFIER)
86     # Button - Main light
87     self.input_device.add_callback(self.input_device.KEY_ACTION, self.input_device.
ACTION_TOGGLE,
88                                     self.switch_main_light.toggle_output_0_mcb)
89     # Button - Desk light
90     self.input_device.add_callback(self.input_device.KEY_ACTION, self.input_device.
ACTION_RIGHT,
91                                     self.light_desk_light.toggle_output_0_mcb)
92     # Button - Amplifier
93     self.input_device.add_callback(self.input_device.KEY_ACTION, self.input_device.
ACTION_LEFT_LONG,
94                                     self.switch_powerplug_4.toggle_output_0_mcb)
95     # Button - CD player
96     self.input_device.add_callback(self.input_device.KEY_ACTION, self.input_device.
ACTION_RIGHT_LONG,
97                                     self.switch_powerplug_4.toggle_output_2_mcb)
98     # Button - PC dock
99     self.input_device.add_callback(self.input_device.KEY_ACTION, self.input_device.
ACTION_LEFT,
100                                     self.switch_pc_dock.toggle_output_0_mcb)
101
102     # additional videv connections
103     self.videv_multistate.connect_br_function(self.brightness_functions,
brightness_choose_n_action.KEY_ACTIVE_DEVICE, 3)
104     #
105     self.videv_audio_player.connect_audio_device(self.audio_status_bluetooth)
106     self.videv_audio_player.connect_audio_device(self.audio_status_mpd)
107     self.videv_audio_player.connect_audio_device(self.audio_status_spotify)
108
109     # heating function
110     self.heating_function = heating_function(
111         self.valve_heating,
112         config.DEFAULT_TEMPERATURE,
113         **get_radiator_data(self.valve_heating.topic)
114     )
115     self.heating_function.add_callback(None, None, set_radiator_data, True)
116     # heating function <--> videv
117     self.videv_heating.connect_heating_function(self.heating_function)
118
119     def audio_source_selector(self, device, key, data):
120         if device == self.switch_powerplug_4 and key == self.KEY_POWERPLUG_CD_PLAYER:
121             # switch on of cd player
122             self.audio_source = self.AUDIO_SOURCE_CD
123         elif device == self.switch_powerplug_4 and key == self.KEY_POWERPLUG_BT:
124             # switch on of bluetooth
125             self.audio_source = self.AUDIO_SOURCE_BT
126         elif device == self.switch_powerplug_4 and key == self.KEY_POWERPLUG_PHONO:

```

```

127         # switch on of bluetooth
128         self.audio_source = self.AUDIO_SOURCE_PHONO
129     elif device in [self.audio_status_spotify, self.audio_status_mpd, self.
audio_status_bluetooth]:
130         # switch on raspi-source
131         self.audio_source = self.AUDIO_SOURCE_RASPI
132     elif device == self.switch_powerplug_4 and key == self.KEY_POWERPLUG_AMPLIFIER:
133         # switch on of amplifier -> select source and reset stored source value
134         self.delayed_task_remote.run()
135
136     def send_audio_source(self):
137         if self.audio_source == self.AUDIO_SOURCE_PC:
138             logger.info("Sending IR command to change audio source to pc")
139             self.remote_ctrl.set_line3()
140         elif self.audio_source == self.AUDIO_SOURCE_CD:
141             logger.info("Sending IR command to change audio source to cd")
142             self.remote_ctrl.set_cd()
143         elif self.audio_source == self.AUDIO_SOURCE_BT:
144             logger.info("Sending IR command to change audio source to bluetooth")
145             self.remote_ctrl.set_line2()
146         elif self.audio_source == self.AUDIO_SOURCE_PHONO:
147             logger.info("Sending IR command to change audio source to phono")
148             self.remote_ctrl.set_phono()
149         elif self.audio_source == self.AUDIO_SOURCE_RASPI:
150             logger.info("Sending IR command to change audio source to raspi")
151             self.remote_ctrl.set_line1()
152         self.audio_source = self.AUDIO_SOURCE_PC
153
154
155 class ground_floor_west_floor(rooms.gfw_floor, room):
156     def __init__(self, mqtt_client):
157         super().__init__(mqtt_client)
158         room.__init__(self, mqtt_client)
159         #
160         # Request silvercrest data of lead light after power on
161         switched_light(self.switch_main_light, self.switch_main_light.KEY_OUTPUT_0, self.
light_main_light)
162         # light <-> videv
163         self.videv_main_light.connect_sw_device(self.switch_main_light, self.switch_main_light.
KEY_OUTPUT_0)
164         self.videv_main_light.connect_br_device(self.light_main_light, self.light_main_light.
KEY_BRIGHTNESS)
165         self.videv_main_light.connect_ct_device(self.light_main_light, self.light_main_light.
KEY_COLOR_TEMP)
166
167
168 class ground_floor_west_marion(rooms.gfw_marion, room):
169     def __init__(self, mqtt_client):
170         super().__init__(mqtt_client)
171         room.__init__(self, mqtt_client)
172         #
173         # light <-> videv
174         self.videv_main_light.connect_sw_device(self.switch_main_light, self.switch_main_light.
KEY_OUTPUT_0)
175         #
176         self.videv_window_light.connect_sw_device(self.light_window_light, self.
light_window_light.KEY_OUTPUT_0)
177         self.videv_window_light.connect_br_device(self.light_window_light, self.
light_window_light.KEY_BRIGHTNESS)
178         self.videv_window_light.connect_ct_device(self.light_window_light, self.
light_window_light.KEY_COLOR_TEMP)
179
180         # main light -> window_light

```

```

181     self.switch_main_light.add_callback(self.switch_main_light.KEY_OUTPUT_0, None, self.
light_window_light.set_output_0_mcb, True)
182
183     # heating function
184     self.heating_function = heating_function(
185         self.valve_heating,
186         config.DEFAULT_TEMPERATURE,
187         **get_radiator_data(self.valve_heating.topic)
188     )
189     self.heating_function.add_callback(None, None, set_radiator_data, True)
190     # heating function <=> videv
191     self.videv_heating.connect_heating_function(self.heating_function)

```

### B.3.7 function.helpers.py

The line coverage for function.helpers.py was 98.5%

The branch coverage for function.helpers.py was 46.8%

```

1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3  #
4  from base import common_base
5  import config
6  import geo
7  import task
8  import time
9
10
11 def now():
12     return time.mktime(time.localtime())
13
14
15 def next_sunrise_time(time_offs_min=30):
16     tm = now()
17     rv = time.mktime(geo.sun.sunrise(config.GEO_POSITION)) + time_offs_min * 60
18     if tm > rv:
19         rv = time.mktime(geo.sun.sunrise(config.GEO_POSITION, date=time.localtime(tm + 24 * 60 *
60))) + time_offs_min * 60
20     return rv
21
22
23 def next_sunset_time(time_offs_min=-30):
24     tm = now()
25     rv = time.mktime(geo.sun.sunset(config.GEO_POSITION)) + time_offs_min * 60
26     if tm > rv:
27         rv = time.mktime(geo.sun.sunset(config.GEO_POSITION, date=time.localtime(tm + 24 * 60 *
60))) + time_offs_min * 60
28     return rv
29
30
31 def next_user_time(hh, mm):
32     ts = time.localtime()
33     tm = time.mktime(ts)
34     ut_ts = list(ts)
35     ut_ts[3] = hh
36     ut_ts[4] = mm
37     ut = time.mktime(time.struct_time(list(ts[:3]) + [hh, mm, 0] + list(ts[6:])))
38     if ts[3] > hh or (ts[3] == hh and ts[4] >= mm):
39         ut += 24 * 60 * 60
40     #

```

```

41     return ut
42
43
44 class day_state(common_base):
45     """
46     Class to subscribe day events as a callback (see add_callback)
47
48     :param time_start_of_day: Time of a day (tuple including hour and minute) for start of day or
49         None for no start of day state.
50     :type time_start_of_day: tuple
51     :param time_start_of_night: Time of a day (tuple including hour and minute) for start of
52         night or None for no end of day state.
53     :type time_start_of_night: tuple
54     :param time_offset_sunrise: time offset for sunrise in minutes (negative values lead to
55         earlier sunrise state) or None for no sunrise state.
56     :type time_start_of_day: int
57     :param time_offset_sunset: time offset for sunset in minutes (negative values lead to earlier
58         sunset state) or None for no sunrise state.
59     :type time_start_of_day: int
60     """
61     KEY_SUNRISE = 'sunrise'
62     KEY_SUNSET = 'sunset'
63     KEY_START_OF_NIGHT = 'start_of_night'
64     KEY_START_OF_DAY = 'start_of_day'
65     #
66     STATES = (KEY_START_OF_DAY, KEY_SUNRISE, KEY_SUNSET, KEY_START_OF_NIGHT)
67
68     def __init__(self, time_start_of_day, time_start_of_night, time_offset_sunrise,
69         time_offset_sunset):
70         self.__time_start_of_day__ = time_start_of_day
71         self.__time_start_of_night__ = time_start_of_night
72         self.__time_offset_sunrise__ = time_offset_sunrise
73         self.__time_offset_sunset__ = time_offset_sunset
74         super().__init__()
75         #
76
77     def get_state(self):
78         tm = {}
79         if self.__time_offset_sunrise__ is not None:
80             tm[next_sunrise_time(self.__time_offset_sunrise__)] = self.KEY_SUNRISE
81         if self.__time_start_of_day__ is not None:
82             tm[next_user_time(*(self.__time_start_of_day__))] = self.KEY_START_OF_DAY
83         if self.__time_offset_sunset__ is not None:
84             tm[next_sunset_time(self.__time_offset_sunset__)] = self.KEY_SUNSET
85         if self.__time_start_of_night__ is not None:
86             tm[next_user_time(*(self.__time_start_of_night__))] = self.KEY_START_OF_NIGHT
87         #
88         tms = list(tm.keys())
89         tms.sort()
90         return tm[tms[-1]]
91
92 class day_event(day_state):
93     """
94     Class to subscribe day events as a callback (see add_callback)
95
96     :param time_start_of_day: Time of a day (tuple including hour and minute) for start of day or
97         None for no start of day state.
98     :type time_start_of_day: tuple
99     :param time_start_of_night: Time of a day (tuple including hour and minute) for start of
100         night or None for no end of day state.
101     :type time_start_of_night: tuple

```

```

96     :param time_offset_sunrise: time offset for sunrise in seconds (negative values lead to
97     earlier sunrise state) or None for no sunrise state.
98     :type time_start_of_day: int
99     :param time_offset_sunset: time offset for sunset in seconds (negative values lead to earlier
100     sunset state) or None for no sunrise state.
101     :type time_start_of_day: int
102     """
103
104     def __init__(self, time_start_of_day=(5, 0), time_start_of_night=(22, 0), time_offset_sunrise
105     =30, time_offset_sunset=-30):
106         super().__init__(time_start_of_day, time_start_of_night, time_offset_sunrise,
107         time_offset_sunset)
108         #
109         current_day_state = self.get_state()
110         for key in self.STATES:
111             self[key] = current_day_state == key
112         #
113         cyclic = task.periodic(30, self.__cyclic__)
114         cyclic.run()
115
116     def __cyclic__(self, a):
117         current_day_state = self.get_state()
118         for key in self.STATES:
119             self.set(key, current_day_state == key)

```

### B.3.8 function.modules.py

The line coverage for function.modules.py was 76.0%

The branch coverage for function.modules.py was 46.8%

```

1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3  #
4  """
5  Functional Modules
6
7  Targets:
8  * Device like structure to be compatible with videv
9    - KEY_* as part of the class for all parameters which needs to be accessed from videv
10   - Method *.set(key, data) to pass data from videv to Module
11   - Method .add_callback(key, data, callback, on_change_only=False) to register videv
12   actualisation on changes
13   """
14
15  from base import common_base
16  import config
17  import devices
18  from function.helpers import day_state
19  import logging
20  import task
21  import time
22
23  try:
24      from config import APP_NAME as ROOT_LOGGER_NAME
25  except ImportError:
26      ROOT_LOGGER_NAME = 'root'
27
28  logger = logging.getLogger(ROOT_LOGGER_NAME).getChild(__name__)

```

```

29 class switched_light(object):
30     def __init__(self, sw_device, sw_key, li_device):
31         sw_device.add_callback(sw_device.KEY_OUTPUT 0, True, li_device.request_data, True)
32
33
34 class brightness_choose_n_action(common_base):
35     KEY_ACTIVE_DEVICE = 'active_device'
36     #
37     DEFAULT_VALUES = {KEY_ACTIVE_DEVICE: None}
38
39     def __init__(self, button_tradfri):
40         super().init()
41         # brightness change
42         button_tradfri.add_callback(button_tradfri.KEY_ACTION, button_tradfri.
43             ACTION_BRIGHTNESS_DOWN_LONG, self.brightness_action)
44         button_tradfri.add_callback(button_tradfri.KEY_ACTION, button_tradfri.
45             ACTION_BRIGHTNESS_UP_LONG, self.brightness_action)
46         button_tradfri.add_callback(button_tradfri.KEY_ACTION, button_tradfri.
47             ACTION_BRIGHTNESS_DOWN_RELEASE, self.brightness_action)
48         button_tradfri.add_callback(button_tradfri.KEY_ACTION, button_tradfri.
49             ACTION_BRIGHTNESS_UP_RELEASE, self.brightness_action)
50         # device change
51         button_tradfri.add_callback(button_tradfri.KEY_ACTION, button_tradfri.
52             ACTION_BRIGHTNESS_UP, self.choose_next_device)
53         button_tradfri.add_callback(button_tradfri.KEY_ACTION, button_tradfri.
54             ACTION_BRIGHTNESS_DOWN, self.choose_prev_device)
55         #
56         self.brightness_device_list = []
57         self.callback_device_list = []
58         self.device_states = []
59
60     def add(self, brightness_device, callback_device, callback_key):
61         """
62         brightness_device: A device for brightness function needs to have the following methods:
63         * .default_inc()
64         * .default_dec()
65         * .default_stop()
66         callback_device: A device for installing callback which are executed, when the device is
67         switched on or off. It needs the following method:
68         * .add_callback(key, data or None, callback, on_changes_only)
69         """
70         self.brightness_device_list.append(brightness_device)
71         self.callback_device_list.append((callback_device, callback_key))
72         self.device_states.append(False)
73         callback_device.add_callback(callback_key, True, self.device_state_action, True)
74         callback_device.add_callback(callback_key, False, self.device_state_action, True)
75
76     def device_state_action(self, device, key, data):
77         self.device_states[self.callback_device_list.index((device, key))] = data
78         if data is True:
79             self.set(self.KEY_ACTIVE_DEVICE, self.callback_device_list.index((device, key)))
80         else:
81             if self[self.KEY_ACTIVE_DEVICE] is not None:
82                 if self.callback_device_list[self[self.KEY_ACTIVE_DEVICE]][0] == device:
83                     self.choose_next_device()
84
85     def choose_prev_device(self, device=None, key=None, data=None):
86         if self[self.KEY_ACTIVE_DEVICE] is not None:
87             start_value = self[self.KEY_ACTIVE_DEVICE]
88             for i in range(0, len(self.brightness_device_list)):
89                 target_state = (start_value - i - 1) % (len(self.brightness_device_list))
90                 if self.device_states[target_state]:
91                     self.set(self.KEY_ACTIVE_DEVICE, target_state)
92                     return
93             self.set(self.KEY_ACTIVE_DEVICE, None)

```

```

87
88     def choose_next_device(self, device=None, key=None, data=None):
89         if self[self.KEY_ACTIVE_DEVICE] is not None:
90             start_value = self[self.KEY_ACTIVE_DEVICE]
91             for i in range(0, len(self.brightness_device_list)):
92                 target_state = (start_value + i + 1) % (len(self.brightness_device_list))
93                 if self.device_states[target_state]:
94                     self.set(self.KEY_ACTIVE_DEVICE, target_state)
95                     return
96         self.set(self.KEY_ACTIVE_DEVICE, None)
97
98     def brightness_action(self, device, key, data):
99         if self[self.KEY_ACTIVE_DEVICE] is not None:
100             target = self.brightness_device_list[self[self.KEY_ACTIVE_DEVICE]]
101             if data == devices.tradfri_button.ACTION_BRIGHTNESS_UP_LONG:
102                 logger.info("Increasing \"%s\" - %s", type(self).__name__, target.topic)
103                 target.default_inc()
104             elif data == devices.tradfri_button.ACTION_BRIGHTNESS_DOWN_LONG:
105                 logger.info("Decreasing \"%s\" - %s", type(self).__name__, target.topic)
106                 target.default_dec()
107             elif data in [devices.tradfri_button.ACTION_BRIGHTNESS_UP_RELEASE, devices.
108 tradfri_button.ACTION_BRIGHTNESS_DOWN_RELEASE]:
109                 target.default_stop()
110
111 class timer_on_activation(common_base):
112     KEY_TIMER = 'timer'
113     #
114     DEFAULT_VALUES = {
115         KEY_TIMER: 0
116     }
117
118     def __init__(self, sw_device, sw_key, timer_reload_value):
119         super().init()
120         #
121         self.timer_reload_value = timer_reload_value
122         #
123         sw_device.add_callback(sw_key, None, self.circ_pump_actions, True)
124         #
125         self.ct = task.periodic(6, self.cyclic_task)
126         self.ct.run()
127
128     def circ_pump_actions(self, device, key, data):
129         if data is True:
130             self.set(self.KEY_TIMER, self.timer_reload_value)
131         else:
132             self.set(self.KEY_TIMER, 0)
133
134     def cyclic_task(self, rt):
135         timer_value = self[self.KEY_TIMER] - self.ct.cycle_time
136         if timer_value <= 0:
137             self.set(self.KEY_TIMER, 0)
138         else:
139             self.set(self.KEY_TIMER, timer_value)
140
141
142 class heating_function(common_base):
143     KEY_USER_TEMPERATURE_SETPOINT = 'user_temperature_setpoint'
144     KEY_TEMPERATURE_SETPOINT = 'temperature_setpoint'
145     KEY_TEMPERATURE_CURRENT = 'temperature_current'
146     KEY_AWAY_MODE = 'away_mode'
147     KEY_SUMMER_MODE = 'summer_mode'
148     KEY_START_BOOST = 'start_boost'
149     KEY_SET_DEFAULT_TEMPERATURE = 'set_default_temperature'
150     KEY_BOOST_TIMER = 'boost_timer'

```

```

151 #
152 BOOST_TEMPERATURE = 30
153 AWAY_REDUCTION = 5
154 SUMMER_TEMPERATURE = 5
155
156 class value_timeout_list(object):
157     MAX_DELAY = 10
158
159     def __init__(self):
160         self.__data__ = []
161         self.__time__ = []
162
163     def __cleanup__(self):
164         now = time.time()
165         for i, tm in enumerate(self.__time__):
166             if tm + self.MAX_DELAY < now:
167                 del (self.__data__[i])
168                 del (self.__time__[i])
169
170     def new(self, item):
171         self.__cleanup__()
172         self.__data__.append(item)
173         self.__time__.append(time.time())
174
175     def is_valid_value(self, data):
176         self.__cleanup__()
177         return data not in self.__data__
178
179     def __init__(self, heating_valve, default_temperature, **kwargs):
180         self.heating_valve = heating_valve
181         self.default_temperature = default_temperature
182         #
183         self.valve_value = self.value_timeout_list()
184         #
185         super().__init__({
186             self.KEY_USER_TEMPERATURE_SETPOINT: kwargs.get(self.KEY_USER_TEMPERATURE_SETPOINT,
187                 self.default_temperature),
188             self.KEY_TEMPERATURE_SETPOINT: kwargs.get(self.KEY_TEMPERATURE_SETPOINT, self.
189                 default_temperature),
190             self.KEY_TEMPERATURE_CURRENT: kwargs.get(self.KEY_TEMPERATURE_CURRENT, None),
191             self.KEY_AWAY_MODE: kwargs.get(self.KEY_AWAY_MODE, False),
192             self.KEY_SUMMER_MODE: kwargs.get(self.KEY_SUMMER_MODE, False),
193             self.KEY_START_BOOST: kwargs.get(self.KEY_START_BOOST, True),
194             self.KEY_SET_DEFAULT_TEMPERATURE: kwargs.get(self.KEY_SET_DEFAULT_TEMPERATURE, False)
195         },
196             self.KEY_BOOST_TIMER: kwargs.get(self.KEY_BOOST_TIMER, 0)
197         })
198         #
199         self.heating_valve.set_heating_setpoint(self[self.KEY_TEMPERATURE_SETPOINT])
200         #
201         self.heating_valve.add_callback(self.heating_valve.KEY_HEATING_SETPOINT, None, self.
202             get_radiator_setpoint)
203         self.heating_valve.add_callback(self.heating_valve.KEY_TEMPERATURE, None, self.
204             get_radiator_temperature)
205         #
206         self.add_callback(self.KEY_USER_TEMPERATURE_SETPOINT, None, self.
207             user_temperature_setpoint, False)
208         self.add_callback(self.KEY_TEMPERATURE_SETPOINT, None, self.set_heating_setpoint, True)
209         self.add_callback(self.KEY_AWAY_MODE, None, self.away_mode, True)
210         self.add_callback(self.KEY_SUMMER_MODE, None, self.summer_mode, True)
211         self.add_callback(self.KEY_SET_DEFAULT_TEMPERATURE, None, self.setpoint_to_default)
212         self.add_callback(self.KEY_START_BOOST, True, self.boost, False)
213         self.add_callback(self.KEY_BOOST_TIMER, 0, self.timer_expired, True)

```

```

208     # cyclic task initialisation
209     self.ct = task.periodic(1, self.cyclic_task)
210     self.ct.run()
211
212     def timer_expired(self, device, data, key):
213         self.set(self.KEY_TEMPERATURE_SETPOINT, self[self.KEY_USER_TEMPERATURE_SETPOINT])
214         self.heating_valve.logger.info('Timer expired. returning to regular temperature setpoint
215                                     %.1f°C.',
216                                     self[self.KEY_TEMPERATURE_SETPOINT])
217
218     def cyclic_task(self, rt):
219         timer_value = self[self.KEY_BOOST_TIMER] - self.ct.cycle_time
220         if self[self.KEY_BOOST_TIMER] <= 0:
221             self.set(self.KEY_BOOST_TIMER, 0)
222         else:
223             self.set(self.KEY_BOOST_TIMER, timer_value)
224
225     def cancel_boost(self):
226         self.set(self.KEY_BOOST_TIMER, 0, block_callback=[self.timer_expired])
227
228     def send_command(self, key, data, block_callback=[]):
229         return super().set(key, data, block_callback)
230
231     def away_mode(self, device, key, value):
232         if value is True:
233             self.cancel_boost()
234             self.set(self.KEY_SUMMER_MODE, False, [self.summer_mode])
235             self.set(self.KEY_TEMPERATURE_SETPOINT, self[self.KEY_USER_TEMPERATURE_SETPOINT] -
236                     self.AWAY_REDUCTION)
237         else:
238             self.set(self.KEY_TEMPERATURE_SETPOINT, self[self.KEY_USER_TEMPERATURE_SETPOINT])
239
240     def summer_mode(self, device, key, value):
241         if value is True:
242             self.cancel_boost()
243             self.set(self.KEY_AWAY_MODE, False, [self.away_mode])
244             self.set(self.KEY_TEMPERATURE_SETPOINT, self.SUMMER_TEMPERATURE)
245         else:
246             self.set(self.KEY_TEMPERATURE_SETPOINT, self[self.KEY_USER_TEMPERATURE_SETPOINT])
247
248     def boost(self, device, key, data):
249         if self[self.KEY_BOOST_TIMER] == 0:
250             self.heating_valve.logger.info('Starting boost mode with setpoint %.1f°C.', self.
251             BOOST_TEMPERATURE)
252             self.set(self.KEY_BOOST_TIMER, 15*60)
253             self.set(self.KEY_TEMPERATURE_SETPOINT, self.BOOST_TEMPERATURE)
254         else:
255             self.set(self.KEY_BOOST_TIMER, min(self[self.KEY_BOOST_TIMER] + 15 * 60, 60 * 60))
256             self.set(self.KEY_AWAY_MODE, False, [self.away_mode])
257             self.set(self.KEY_SUMMER_MODE, False, [self.summer_mode])
258
259     def setpoint_to_default(self, device, key, data):
260         self.cancel_boost()
261         self.set(self.KEY_AWAY_MODE, False, [self.away_mode])
262         self.set(self.KEY_SUMMER_MODE, False, [self.summer_mode])
263         self.set(self.KEY_USER_TEMPERATURE_SETPOINT, self.default_temperature, [self.
264         user_temperature_setpoint])
265         self.set(self.KEY_TEMPERATURE_SETPOINT, self.default_temperature)
266
267     def user_temperature_setpoint(self, device, key, data):
268         self.cancel_boost()
269         self.set(self.KEY_AWAY_MODE, False, [self.away_mode])
270         self.set(self.KEY_SUMMER_MODE, False, [self.summer_mode])
271         self.set(self.KEY_TEMPERATURE_SETPOINT, data)

```

```

268
269     def set_heating_setpoint(self, device, key, data):
270         self.valve_value.new(data)
271         self.heating_valve.set_heating_setpoint(data)
272
273     def get_radiator_setpoint(self, device, key, data):
274         if self.valve_value.is_valid_value(data):
275             if self[self.KEY_BOOST_TIMER] == 0 and not self[self.KEY_AWAY_MODE] and not self[self
276                 .KEY_SUMMER_MODE]:
277                 self.set(self.KEY_USER_TEMPERATURE_SETPOINT, data, block_callback=[self.
278                     set_heating_setpoint])
279
280
281     def get_radiator_temperature(self, device, key, data):
282         self.set(self.KEY_TEMPERATURE_CURRENT, data)
283
284
285 class motion_sensor_light(common_base):
286     KEY_TIMER = 'timer'
287     KEY_MOTION_SENSOR = 'motion_%d'
288     KEY_MOTION_SENSOR_0 = 'motion_%d' % 0
289     KEY_MOTION_SENSOR_1 = 'motion_%d' % 1
290     KEY_MOTION_SENSOR_2 = 'motion_%d' % 2
291     KEY_MOTION_SENSOR_3 = 'motion_%d' % 3
292     KEY_MOTION_SENSOR_4 = 'motion_%d' % 4
293
294     def __init__(self, sw_device, sw_method, *args, timer_value=30):
295         """
296         sw_device is the device switching the light, args are 0–n motion sensors
297         """
298         dv = dict.fromkeys([self.KEY_MOTION_SENSOR % i for i in range(0, len(args))])
299         for key in dv:
300             dv[key] = False
301         dv[self.KEY_TIMER] = 0
302         super().__init__(default_values=dv)
303         #
304         self.sw_device = sw_device
305         self.sw_method = sw_method
306         self.motion_sensors = args
307         self.timer_reload_value = timer_value
308         #
309         sw_device.add_callback(devices.shelly_sw1.KEY_OUTPUT_0, True, self.reload_timer, True)
310         sw_device.add_callback(devices.shelly_sw1.KEY_OUTPUT_0, False, self.reset_timer, True)
311         for motion_sensor in args:
312             motion_sensor.add_callback(motion_sensor.KEY_OCCUPANCY, None, self.
313                 set_motion_detected, True)
314         #
315         self.add_callback(self.KEY_TIMER, 0, self.timer_expired, True)
316         #
317         cyclic_task = task.periodic(1, self.cyclic_task)
318         cyclic_task.run()
319
320     def reload_timer(self, device, key, data):
321         self.set(self.KEY_TIMER, self.timer_reload_value)
322
323     def reset_timer(self, device=None, key=None, data=None):
324         self.set(self.KEY_TIMER, 0)
325
326     def set_motion_detected(self, device, key, data):
327         for sensor_index, arg_device in enumerate(self.motion_sensors):
328             if arg_device.topic == device.topic:
329                 break
330         self.set(self.KEY_MOTION_SENSOR % sensor_index, data)

```

```

327         # auto light on with state sunset -> time_offset_sunrise=60 (longer sunset) and
time_offset_sunset=-60 (longer sunset)
328         if day_state(None, None, 60, -60).get_state() == day_state.KEY_SUNSET:
329             if data is True:
330                 logger.info("%s: Motion detected - Switching on main light %s", device.topic,
self.sw_device.topic)
331                 self.sw_method(True)
332
333     def motion_detected(self):
334         for i in range(0, len(self.motion_sensors)):
335             if self[self.KEY_MOTION_SENSOR % i]:
336                 return True
337         return False
338
339     def timer_expired(self, device, key, data):
340         logger.info("No motion and time ran out - Switching off main light %s", self.sw_device.
topic)
341         self.sw_method(False)
342
343     def cyclic_task(self, cyclic_task):
344         min_value = 10 if self.motion_detected() else 0
345         if self[self.KEY_TIMER] != 0:
346             self.set(self.KEY_TIMER, max(min_value, self[self.KEY_TIMER] - cyclic_task.cycle_time
))

```

### B.3.9 function.rooms.py

The line coverage for function.rooms.py was 30.4%

The branch coverage for function.rooms.py was 46.8%

```

1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3  #
4
5  import logging
6  import inspect
7
8  try:
9      from config import APP_NAME as ROOT_LOGGER_NAME
10 except ImportError:
11     ROOT_LOGGER_NAME = 'root'
12 logger = logging.getLogger(ROOT_LOGGER_NAME).getChild(__name__)
13
14
15 class room(object):
16     def __init__(self, mqtt_client):
17         self.mqtt_client = mqtt_client
18
19     def all_off(self, device=None, key=None, data=None):
20         logger.info("Switching all off \"%s\"", type(self).__name__)
21         for name, obj in inspect.getmembers(self):
22             try:
23                 if obj.__module__.startswith('devices'):
24                     obj.all_off()
25             except AttributeError:
26                 pass # not a module or has no method all_off
27
28     def summer_mode(self, enable):

```

## Unittest for smart\_brain

```

29         for name, obj in inspect.getmembers(self):
30             if obj.__class__.__name__ == 'heating_function':
31                 if obj.__module__ == 'function_modules':
32                     obj.set(obj.KEY_SUMMER_MODE, enable)
33
34
35 class room_collection(object):
36     ALLOWED_CLASSES = ("room", "room_collection")
37
38     def __init__(self, mqtt_client):
39         self.mqtt_client = mqtt_client
40
41     def all_off(self, device=None, key=None, data=None):
42         logger.info("Switching all off \"%s\"", type(self).__name__)
43         for sub_name in dir(self):
44             # attribute name is not private
45             if not sub_name.startswith("__"):
46                 sub = getattr(self, sub_name)
47                 # try to call all_off
48                 try:
49                     sub.all_off()
50                 except AttributeError:
51                     pass # don't mind, if sub has no method all_off
52
53     def summer_mode(self, device=None, key=None, data=None):
54         logger.info("Changing to %s \"%s\"", "summer mode" if data else "winter_mode", type(self)
55             .__name__)
56         for sub_name in dir(self):
57             # attribute name is not private
58             if not sub_name.startswith("__"):
59                 sub = getattr(self, sub_name)
60                 if sub.__class__.__bases__[0].__name__ in self.ALLOWED_CLASSES:
61                     sub.summer_mode(data)
62
63     def all_devices(self, object_to_analyse=None, depth=0):
64         target = object_to_analyse or self
65         #
66         devices = []
67         for name, obj in inspect.getmembers(target):
68             if not callable(obj): # sort out methods
69                 try:
70                     if obj.__module__.startswith('function.') and not obj.__module__.endswith('.
71 videv'):
72                         devices.extend(self.all_devices(obj, depth+1)) # rekurse in
73 function instances
74                     elif obj.__module__ == "devices":
75                         devices.append(obj)
76                 except AttributeError:
77                     pass # sort out non modules
78         return devices

```

**B.3.10** function.stairway.py

The line coverage for function.stairway.py was 90.5%

The branch coverage for function.stairway.py was 46.8%

```

1 #!/usr/bin/env python
2 # -*- coding: utf-8 -*-
3 #
4
5 import config
6 from devdi import rooms
7 import logging
8 from function.modules import motion_sensor_light
9 from function.rooms import room, room_collection
10
11 try:
12     from config import APP_NAME as ROOT_LOGGER_NAME
13 except ImportError:
14     ROOT_LOGGER_NAME = 'root'
15 logger = logging.getLogger(ROOT_LOGGER_NAME).getChild( name )
16
17
18 class stairway(room_collection):
19     def __init__(self, mqtt_client):
20         super().__init__(mqtt_client)
21         self.stairway = stairway_stairway(mqtt_client)
22
23
24 class stairway_stairway(rooms.stairway, room):
25     def __init__(self, mqtt_client):
26         super().__init__(mqtt_client)
27         room.__init__(self, mqtt_client)
28         #
29         # connect videv and switch
30         self.videv_main_light.connect_sw_device(self.switch_main_light, self.switch_main_light.
KEY OUTPUT 0)
31
32         self.motion_sensor_light = motion_sensor_light(
33             self.switch_main_light, self.switch_main_light.set_output_0,
34             self.motion_main_light_gf, self.motion_main_light_ff,
35             timer_value=config.USER_ON_TIME_STAIRWAYS
36         )
37         self.videv_main_light.connect_mo_function(self.motion_sensor_light)

```

**B.3.11** function.videv.py

The line coverage for function.videv.py was 94.1%

The branch coverage for function.videv.py was 46.8%

```

1 #!/usr/bin/env python
2 # -*- coding: utf-8 -*-
3 #
4 """
5 Virtual Device(s)
6
7 Targets:
8 * MQTT-Interface to control joined devices as one virtual device
9 * Primary signal routing
10 * No functionality should be implemented here
11 """
12

```

```

13 from base import videv_base
14 from function.rooms import room, room_collection
15
16
17 class videv_pure_switch(videv_base):
18     KEY_STATE = 'state'
19
20     def __init__(self, mqtt_client, topic):
21         super().__init__(mqtt_client, topic)
22         self[self.KEY_STATE] = False
23         #
24         self.mqtt_client.add_callback(self.topic + '/state/set', self.__state__)
25
26     def __state__(self, mqtt_client, userdata, message):
27         self.set(self.KEY_STATE, message.payload == b'true')
28         self.__tx__(self.KEY_STATE, message.payload == b'true')
29
30
31 class videv_switching(videv_base):
32     KEY_STATE = 'state'
33
34     def __init__(self, mqtt_client, topic):
35         super().__init__(mqtt_client, topic)
36
37     def connect_sw_device(self, sw_device, sw_key):
38         self.add_routing(self.KEY_STATE, sw_device, sw_key)
39
40
41 class videv_switching_timer(videv_switching):
42     KEY_TIMER = 'timer'
43
44     def __init__(self, mqtt_client, topic):
45         super().__init__(mqtt_client, topic)
46
47     def connect_tm_device(self, tm_device, tm_key):
48         self.add_display(self.KEY_TIMER, tm_device, tm_key)
49
50
51 class videv_switching_motion(videv_switching):
52     KEY_STATE = 'state'
53     #
54     KEY_TIMER = 'timer'
55     KEY_MOTION_SENSOR = 'motion_%d'
56
57     def __init__(self, mqtt_client, topic):
58         super().__init__(mqtt_client, topic)
59
60     def connect_mo_function(self, mo_function):
61         self.add_display(self.KEY_TIMER, mo_function, mo_function.KEY_TIMER)
62         # motion sensor state
63         for index, motion_sensor in enumerate(mo_function.motion_sensors):
64             self.add_display(self.KEY_MOTION_SENSOR % index, motion_sensor, motion_sensor.
KEY_OCCUPANCY)
65
66
67 class videv_switch_brightness(videv_switching):
68     KEY_BRIGHTNESS = 'brightness'
69
70     def __init__(self, mqtt_client, topic):
71         super().__init__(mqtt_client, topic)
72
73     def connect_br_device(self, br_device, br_key):
74         self.add_routing(self.KEY_BRIGHTNESS, br_device, br_key)

```

```

75
76
77 class videv_switch_brightness_color_temp(videv_switch_brightness):
78     KEY_COLOR_TEMP = 'color_temp'
79
80     def __init__(self, mqtt_client, topic):
81         super().__init__(mqtt_client, topic)
82
83     def connect_ct_device(self, ct_device, ct_key):
84         self.add_routing(self.KEY_COLOR_TEMP, ct_device, ct_key)
85
86
87 class videv_heating(videv_base):
88     KEY_USER_TEMPERATURE_SETPOINT = 'user_temperature_setpoint'
89     KEY_VALVE_TEMPERATURE_SETPOINT = 'valve_temperature_setpoint'
90     KEY_AWAY_MODE = 'away_mode'
91     KEY_SUMMER_MODE = 'summer_mode'
92     KEY_START_BOOST = 'start_boost'
93     KEY_SET_DEFAULT_TEMPERATURE = 'set_default_temperature'
94     KEY_BOOST_TIMER = 'boost_timer'
95     #
96     KEY_TEMPERATURE = 'temperature'
97
98     def __init__(self, mqtt_client, topic):
99         super().__init__(mqtt_client, topic)
100
101     def connect_heating_function(self, heating_function):
102         #
103         self.add_routing(self.KEY_USER_TEMPERATURE_SETPOINT, heating_function, heating_function.
104             KEY_USER_TEMPERATURE_SETPOINT)
105         self.add_routing(self.KEY_AWAY_MODE, heating_function, heating_function.KEY_AWAY_MODE)
106         self.add_routing(self.KEY_SUMMER_MODE, heating_function, heating_function.KEY_SUMMER_MODE
107         )
108         #
109         self.add_control(self.KEY_START_BOOST, heating_function, heating_function.KEY_START_BOOST
110             , False)
111         self.add_control(self.KEY_SET_DEFAULT_TEMPERATURE, heating_function, heating_function.
112             KEY_SET_DEFAULT_TEMPERATURE, False)
113         #
114         self.add_display(self.KEY_VALVE_TEMPERATURE_SETPOINT, heating_function, heating_function.
115             KEY_TEMPERATURE_SETPOINT)
116         self.add_display(self.KEY_BOOST_TIMER, heating_function, heating_function.KEY_BOOST_TIMER
117         )
118         self.add_display(self.KEY_TEMPERATURE, heating_function, heating_function.
119             KEY_TEMPERATURE_CURRENT, False)
120
121
122 class videv_multistate(videv_base):
123     KEY_STATE = 'state_%d'
124
125     def __init__(self, mqtt_client, topic):
126         super().__init__(mqtt_client, topic)
127
128     def connect_br_function(self, device, key_for_device, num_states):
129         self.num_states = num_states
130         # send default values
131         for i in range(0, num_states):
132             self.__tx__(self.KEY_STATE % i, False)
133         #
134         device.add_callback(key_for_device, None, self.__index_rx__, True)

```

```

129     def __index_rx__(self, device, key, data):
130         for i in range(0, self.num_states):
131             self.__tx__(self.KEY_STATE % i, i == data)
132
133
134 class videv_audio_player(videv_base):
135     KEY_ACTIVE_PLAYER = 'player_%d'
136     KEY_TITLE = 'title'
137     NO_TITLE = '___'
138
139     def __init__(self, mqtt_client, topic):
140         super().__init__(mqtt_client, topic)
141         self.__device_cnt__ = 0
142
143     def connect_audio_device(self, device):
144         self.add_display(self.KEY_ACTIVE_PLAYER % self.__device_cnt__, device, device.KEY_STATE)
145         device.add_callback(device.KEY_TITLE, None, self.__title_rx__, True)
146         self.__device_cnt__ += 1
147
148     def __title_rx__(self, device, key, data):
149         self.__tx__(self.KEY_TITLE, data or self.NO_TITLE)
150
151
152 class all_off(videv_base):
153     ALLOWED_CLASSES = (room, room_collection, )
154
155     def __init__(self, mqtt_client, topic, room_collection):
156         super().__init__(mqtt_client, topic)
157         self.__room_collection__ = room_collection
158         # init inst dict
159         self.__inst_dict__ = {}
160         self.__add_instances__("all", self.__room_collection__)
161         # register mqtt callbacks for all my keys
162         for key in self.__inst_dict__:
163             mqtt_client.add_callback(topic + "/" + key, self.all_off)
164
165     def __check_inst_capabilities__(self, name, inst):
166         # fits to specified classes
167         if isinstance(inst, self.ALLOWED_CLASSES):
168             try:
169                 # all_off method is callable
170                 return callable(inst.all_off)
171             except AttributeError:
172                 # all_off method does not exist
173                 return False
174         return False
175
176     def __add_instances__(self, name, inst, level=0):
177         if self.__check_inst_capabilities__(name, inst):
178             # add given instance to my __inst_dict__
179             self.__inst_dict__[name] = inst
180             # iterate over all attribute names of instance
181             for sub_name in dir(inst):
182                 # attribute name is not private
183                 if not sub_name.startswith("__"):
184                     sub = getattr(inst, sub_name)
185                     # recurse with this object
186                     if level == 0:
187                         self.__add_instances__(sub_name, sub, level=level+1)
188                     else:
189                         self.__add_instances__(name + "/" + sub_name, sub, level=level+1)
190
191     def all_off(self, client, userdata, message):
192         key = message.topic[len(self.topic) + 1:]
193         self.__inst_dict__[key].all_off()

```