

ioB Scripte

Automatiken

```

/*****
* LICHT
*****/

// RGB Lampe Wohnzimmer Nachts automatisch ausschalten WZ
> 23:00 Uhr WZ_Sofalicht_Aus
    if (getState("Global.0.Automatik.WZ_Sofalicht_Aus").val == true) {
        if (hour == 23 && min == 0) {
            setState("Global.0.Licht.State.WZ_Sofa", 0); Logger("RGB Lampe Wohnzimmer
            aus"); }
        }

// Sideboard & Fenster automatisch aus WZ
> "" WZ_Sideboard_Fenster_Aus
    if (getState("Global.0.Automatik.WZ_Sideboard_Fenster_Aus").val == true)
    {
        if (hour == 23 && min == 0) { setState("hm-
        rpc.0.LEQ0900960.3.STATE", false); setState("hm-rpc.0.LEQ0900960.4.STATE",
        false); Logger("Sideboard & Fenster aus"); }
    }

// Tannenbaum Automatik (Licht) WZ
Abends an ab 16:00 - 22:00 WZ_Tannenbaum_Auto
    if (getState("Global.0.Automatik.WZ_Tannenbaum_Auto").val == true) {
        if (hour == 16 && min == 0) { setState("sonoff.0.IOT-SD11-WZ-
        Weihnachtsbaum.POWER", true); Logger("Tannenbaum an"); }
        if (hour == 22 && min == 0) { setState("sonoff.0.IOT-SD11-WZ-
        Weihnachtsbaum.POWER", false); Logger("Tannenbaum aus"); }
    }

// Küche Bewegungsmelder An / Aus ? KU
KU_Bewegungsmelder_AnAus
// siehe Küche -> LichtSpüle

// Küche Deko an / aus KU
Wie Tannenbaum KU_Deko_Auto
    if (getState("Global.0.Automatik.KU_Deko_Auto").val == true) {
        if (hour == 16 && min == 0) { setState("hm-
        rpc.0.NEQ0951347.2.STATE", true); Logger("Küche Deko an"); }
        if (hour == 22 && min == 0) { setState("hm-
        rpc.0.NEQ0951347.2.STATE", false); Logger("Küche Deko aus"); }
    }

// Beleuchtung Bild Automatik FL
ab SonUn bis 22:30 FL_Fotowand_Auto

```

```

    if (getState("Global.0.Automatik.FL_Fotowand_Auto").val == true) {
        if (act == typeSunsetStart) {
            setState("sonoff.0.IOT-SD09-FL-LichtBild.POWER", true ); Logger("Beleuchtung
            Bild an"); } // AN
            if (act == typeSchedule && hour == 22 && min == 30) {
                setState("sonoff.0.IOT-SD09-FL-LichtBild.POWER", false); Logger("Beleuchtung
                Bild aus"); } // AUS
        }

        // Beleuchtung Handlauf Automatik
        ""
        FL_TreppeBeleuchtung_Auto
        if (getState("Global.0.Automatik.FL_TreppeBeleuchtung_Auto").val ==
        true) {
            if (act == typeSunsetStart) {
                setState("sonoff.0.IOT-SD10-FL-LichtTreppe.POWER", true );
                Logger("Beleuchtung Handlauf an"); } // AN
                if (act == typeSchedule && hour == 22 && min == 30) {
                    setState("sonoff.0.IOT-SD10-FL-LichtTreppe.POWER", false);
                    Logger("Beleuchtung Handlauf aus"); } // AUS
            }

            // Stern oben an / aus
            ""
            FL_SternOben_Auto
            if (getState("Global.0.Automatik.FL_SternOben_Auto").val == true) {
                if (act == typeSunsetStart) {
                    setState("sonoff.0.IOT-SD08-FL-FensterOben.POWER", true );
                    Logger("Beleuchtung Handlauf an"); } // AN
                    if (act == typeSchedule && hour == 22 && min == 30) {
                        setState("sonoff.0.IOT-SD08-FL-FensterOben.POWER", false);
                        Logger("Beleuchtung Handlauf aus"); } // AUS
                }

                // Lennard Zimmerlicht morgens an und gegen 8 aus
                LN
                6 - 7:30 Uhr , 22 Uhr aus
                LN_Zimmerlicht_Auto
                if (getState("Global.0.Automatik.LN_Zimmerlicht_Auto").val == true) {
                    // 6 Uhr ein Mo-Th, 6:40 Fr -> Montag = 1, Freitag = 5 !
                    if ((hour == 6 && min == 0 && [1, 2, 3, 4].includes(day)) ||
                        (hour == 6 && min == 40 && day == 5)) { setState("hm-
                    rpc.0.MEQ0987276.1.STATE", true); Logger("Lennard Zimmerlicht Ein"); }
                    // 7:30 aus
                    if (hour == 7 && min == 30) { setState("hm-
                    rpc.0.MEQ0987276.1.STATE", false); Logger("Lennard Zimmerlicht Aus"); }
                    // Licht Abends ausschalten wenn noch an
                    if (hour == 22 && min == 0) { setState("hm-
                    rpc.0.MEQ0987276.1.STATE", false); Logger("Lennard Zimmerlicht Abends Aus");
                    }
                }

                // Haustürlicht Automatik
                SonUn bis 23:00 + Bewegung wenn da
                OD
                OD_Haustuer_Auto
                //

```

```

6 Uhr -> Sonnenuntergang
    if (getState("Global.0.Automatik.OD_Haustuer_Auto").val == true) {
        if (hour == 6 && min == 0) { setState("hm-
rpc.0.NEQ1504413.1.STATE", true); Logger("Haustürlicht An"); } //
/*OD_SW_Haustue_Licht:1.STATE*/
        if (act == typeSunriseEndPlus30) { setState("hm-
rpc.0.NEQ1504413.1.STATE", false); Logger("Haustürlicht Aus"); }
        if (act == typeSunsetStartMinus30) { setState("hm-
rpc.0.NEQ1504413.1.STATE", true); Logger("Haustürlicht An"); }
        if (hour == 23 && min == 0) { setState("hm-
rpc.0.NEQ1504413.1.STATE", false); Logger("Haustürlicht Aus"); }
        // TBD : Hier muss sauber der Sonnenaufgang genutzt werden ...
        if (hour == 8 && min == 0) { setState("hm-
rpc.0.NEQ1504413.1.STATE", false); Logger("Haustürlicht Aus Workaround für
Sommer"); }
    }

    // Garagenlicht mit Haustür an OD
    "" OD_HaustuerMitGarage_Auto
    // TBD

    /*****
    * HEIZUNG
    *****/
    // Alle Heizungen im Haus auf Nachtbetrieb absenken ALL
-> "Heizung aus" Script um 21:00 ALL_HZ_Nachtbetrieb
    // Heizung Morgens hochfahren ? ALL
-> um 6 Warm KU KU_HZ_Morgens
    // Badezimmer Heizung auf 21°C von 5:30 bis 9:30 BD
siehe Text BD_HZ_Morgens
    // Wohnzimmer Temp halten WZ
WZ_HZ_Tagbetrieb
    // Heizung Lennards Zimmer kontrollieren LN
mal sehen LN_HZ_Überwachen
    /*****
    * ROLLOS
    *****/

    // Morgens alle Rollos hoch ALL
um 8 Uhr ALL_RL_Öffnen
    if (getState("Global.0.Automatik.ALL_RL_Öffnen").val == true) {
        if (act == typeSchedule && hour == 8 && min == 0) {
setState("Global.0.Rollos.Control", "allup"); Logger("Alle Rollos
hochfahren"); }
    }

    // Rollos automatisch runter fahren (auf Terrassentür achten) ALL
um SonUn Ende + 30Min ALL_RL_Schließen
    // Überwachung Terrassentür in Rollos -> Steuerung
    if (getState("Global.0.Automatik.ALL_RL_Schließen").val == true) {

```

```

        if (act == typeSunsetStartPlus30) {
setState("Global.0.Rollos.Control", "alldown"); Logger("Alle Rollos
schließen"); }
    }

/*****
* STECKDOSEN
*****/

    // Hifi abschalten WZ                                     WZ
Nachts um 0:00                                                WZ_Hifi_Aus
    // Steckdose AZ                                           AZ
""                                                            AZ_Steckdose_Auto
    // TV abschalten GZ                                       GZ
""                                                            GZ_TV_Aus
    // PS4 abschalten                                         GZ
Ab 21:30 an Werktagen                                         GZ_PS_Aus
    // Waschmaschine aus wenn fertig                           KL
Abschalten der SD bei Fertigmeldung                         KL_Waschmaschine_Aus
    // -> Wird in Weisse Ware -> Waschmaschine behandelt !
    // -> Weil dort wird direkt auf Waschm. Fertig reagiert ...

    // Trockner aus wenn fertig                               KL
""                                                            KL_Trockner_Aus

    // WZ Hifi Aus um 1 Uhr
    if (getState("Global.0.Automatik.WZ_Hifi_Aus").val == true) {
        if (act == typeSchedule && hour == 1 && min == 0) { setState("hm-
rpc.0.MEQ1810633.3.STATE", false); Logger("WZ Hifi aus"); } // AUS
    }

    // GZ TV Aus um 1 Uhr
    if (getState("Global.0.Automatik.GZ_TV_Aus").val == true) {
        if (act == typeSchedule && hour == 1 && min == 0) {
setState("sonoff.0.IOT-SD05-GZ-Tv0ben.POWER", false); Logger("GZ TV aus"); }
// AUS
    }

/*****
* DRAUSSEN
*****/

    // Markiese um 0:00 automatisch einziehen                 OD
0:00 Uhr                                                       OD_Markiese_Rein
    // Markiese Sommerbetrieb Ausfahren möglich             OD
on / off                                                       OD_Markiese_Sommerbetrieb
    // Schuppen an / aus -> Mäher aus                         OD
Ab 22 Uhr bis 4 Uhr                                           OD_Schuppen_Auto
    // Garage automatisch nach x Min schließen              OD
nach 5 Minuten

```

Bewegung flur

[download](#)

```

/*****
* Licht bei Bewegung im Flur einschalten
*
* Geräte
* hm-rpc.0.OEQ0536226.3.MOTION  FL_SB_Motion_Oben_Motion
Bewegungsmelder Oben
* hm-rpc.0.OEQ0538240.3.MOTION  FL_SB_Motion_Unten_Motion
Bewegungsmelder Unten
* hm-rpc.0.OEQ0213729.1.STATE    FL_AS_LichtOben:1          Licht Oben
(einfach)
* hm-rpc.0.OEQ0213729.2.STATE    FL_AS_LichtOben:2          Licht Oben
(doppelt)
* hm-rpc.0.OEQ0378804.1.STATE    FL_AS_LichtUnten:1         Licht Unten
kmplett
*****/

//console.log("sunriseEnd" + getAstroDate("sunriseEnd", undefined,
+10)); // BSP 07:13
//console.log("sunriseEnd" + getAstroDate("sunriseEnd", undefined,
0)); // BSP 07:03
//console.log("sunsetStart" + getAstroDate("sunsetStart", undefined,
-10)); // BSP 19:30
//console.log("sunsetStart" + getAstroDate("sunsetStart", undefined,
0)); // BSP 19:40

// Bewegung Flur Unten
// - unten das Licht einschalten
// - oben das doppelte Licht einschalten
// - nach x Sekunden alles wieder ausschalten
on({id: 'hm-rpc.0.OEQ0538240.3.MOTION', val: true}, function (obj) {
    // Licht NICHT einschalten zwischen Sonnenaufgang (+30 Min) und
    Sonnenuntergang (-30 Min)
    if (compareTime(getAstroDate("sunriseEnd", undefined, +30),
getAstroDate("sunsetStart", undefined, -30), "not between", null)) {
        // Ist das Licht schon an ?
        if (getState("hm-rpc.0.OEQ0378804.1.STATE").val == false) {
            console.log('Flur unten : Licht ein');
            setState("hm-rpc.0.OEQ0378804.1.STATE", true);
        }
        // Licht Unten ein
        setStateDelayed("hm-rpc.0.OEQ0378804.1.STATE", false,
180000, false); // Licht Unten aus nach 180 Sekunden
        setState("hm-rpc.0.OEQ0213729.2.STATE", true);
        // 2er Licht Oben ein
        setStateDelayed("hm-rpc.0.OEQ0213729.2.STATE", false,
180000, false); // 2er Licht Oben aus nach 180 Sekunden
    }
}

```

```

    }
    // else {
    //     console.log("Licht bleibt am Tag aus ...");
    // }
    setState("hm-rpc.0.0EQ0538240.3.MOTION", false);
// Motion wieder zurück setzen - wird nicht immer resettet
});

// Bewegung Flur Oben
// - oben das einfache Licht einschalten
// - nach x Sekunden alles wieder ausschalten
on({id: 'hm-rpc.0.0EQ0536226.3.MOTION', val: true}, function (obj) {
    // Licht NICHT einschalten zwischen Sonnenaufgang (+30 Min) und
    Sonnenuntergang (-30 Min)
    if (compareTime(getAstroDate("sunriseEnd", undefined, +30),
getAstroDate("sunsetStart", undefined, -30), "not between", null)) {
        // Ist das Licht schon an ?
        if (getState("hm-rpc.0.0EQ0213729.1.STATE").val == false) {
            console.log('Flur oben : Licht ein');
            setState("hm-rpc.0.0EQ0213729.1.STATE", true);
// 1er Licht Oben ein
            setStateDelayed("hm-rpc.0.0EQ0213729.1.STATE", false,
180000, false); // 1er Licht Oben aus nach 180 Sekunden
        }
    }
    setState("hm-rpc.0.0EQ0536226.3.MOTION", false);
// Motion wieder zurück setzen - wird nicht immer resettet
});

```

Schalter

[download](#)

```

/*****
* Schalter im Flur
*
* Geräte
* hm-rpc.0.0EQ0536226.3.MOTION      FL_SB_Motion_Oben_Motion
Bewegungsmelder Oben
* hm-rpc.0.0EQ0538240.3.MOTION      FL_SB_Motion_Unten_Motion
Bewegungsmelder Unten
* hm-rpc.0.0EQ0538240.1.PRESS_SHORT
Bewegungsmelder Taste Hoch
* hm-rpc.0.0EQ0538240.2.PRESS_SHORT
Bewegungsmelder Taste Runter
* hm-rpc.0.0EQ0213729.1.STATE      FL_AS_LichtOben:1      Licht
Oben (einfach)
* hm-rpc.0.0EQ0213729.2.STATE      FL_AS_LichtOben:2      Licht

```

```
Oben (doppelt)
* hm-rpc.0.OEQ0378804.1.STATE          FL_AS_LichtUnten:1          Licht
Unten kmplett
*****/

// Bewegungsmelder Flur unten Taster hoch
on({id: 'hm-rpc.0.OEQ0538240.1.PRESS_SHORT', change: "any"}, function
(obj) {
    console.log('Licht unten eingeschaltet (10 Minuten)');
    setState("hm-rpc.0.OEQ0378804.1.STATE", true);
// Licht Unten ein
    setStateDelayed("hm-rpc.0.OEQ0378804.1.STATE", false, 600000,
false);    // Licht Unten aus nach 10 Minuten
});

// Bewegungsmelder Flur unten Taster runter
on({id: 'hm-rpc.0.OEQ0538240.2.PRESS_SHORT', change: "any"}, function
(obj) {
    setState("hm-rpc.0.OEQ0378804.1.STATE", false);
// Licht unten aus
});

// 6fach Taster (1) : Garage öffnen
on({id: 'hm-rpc.0.NEQ1068837.1.PRESS_SHORT', change: "any"}, function
(obj) {
    setState("Global.0.Garage.Open", true);
});

// 6fach Taster (2) : Garage schließen
on({id: 'hm-rpc.0.NEQ1068837.2.PRESS_SHORT', change: "any"}, function
(obj) {
    setState("Global.0.Garage.Close", true);
});

// 6fach Taster (3) : Licht Haustür
on({id: 'hm-rpc.0.NEQ1068837.3.PRESS_SHORT', change: "any"}, function
(obj) {
    setState("hm-rpc.0.NEQ1504413.1.STATE", !getState("hm-
rpc.0.NEQ1504413.1.STATE").val);    /*OD_SW_Haustue_Licht:1.STATE*/
});

// 6fach Taster (4) : Licht Garten
on({id: 'hm-rpc.0.NEQ1068837.4.PRESS_SHORT', change: "any"}, function
(obj) {
    setState("hm-rpc.0.MEQ1809943.1.STATE", !getState("hm-
rpc.0.MEQ1809943.1.STATE").val);    /*OD_AS_4Fach_Licht:1.STATE*/
});

// 6fach Taster (5) : Licht 2er Garage
on({id: 'hm-rpc.0.NEQ1068837.5.PRESS_SHORT', change: "any"}, function
(obj) {
```

```

        setState("hm-rpc.0.NEQ1222700.1.STATE", !getState("hm-
rpc.0.NEQ1222700.1.STATE").val);
/*GR_AS_LichtAussen_2erLicht:1.STATE*/
});

// 6fach Taster (6) : Licht 4er Garage
on({id: 'hm-rpc.0.NEQ1068837.6.PRESS_SHORT', change: "any"}, function
(obj) {
    setState("hm-rpc.0.NEQ1222700.2.STATE", !getState("hm-
rpc.0.NEQ1222700.2.STATE").val);
/*GR_AS_LichtAussen_4erLicht:2.STATE*/
});

```

Garage

[download](#)

```

/*****
* Beleutung Garage wenn das Tor geöffnet / geschlossen wird
* > Tagsüber Licht aus lassen
* > Haustür mit an wenn Nachts und Haustür aus
* > Licht im Flur unten auch mit einschalten ??
*
* Geräte
* hm-rpc.0.OEQ0430315.1.STATE    GR_ST_Garage           Garagen
Kontakt
* hm-rpc.0.NEQ1222700.1.STATE    GR_AS_LichtAussen_2erLicht  2er Licht
Garage
* hm-rpc.0.NEQ1222700.2.STATE    GR_AS_LichtAussen_4erLicht  4er Licht
Garage
* hm-rpc.0.OEQ0378804.1.STATE    FL_AS_LichtUnten:1.STATE    Flur Licht
unten
*****/

on({id: 'hm-rpc.0.OEQ0430315.1.STATE', change: "ne"}, function (obj) {
    if (getState("hm-rpc.0.OEQ0430315.1.STATE").val == true) {
        // Licht NICHT einschalten zwischen Sonnenaufgang (+30 Min) und
        Sonnenuntergang (-30 Min)
        if (compareTime(getAstroDate("sunriseEnd", undefined, +30),
getAstroDate("sunsetStart", undefined, -30), "not between", null)) {
            // Licht bei Garage auf einschalten
            console.log('Tor Offen -> Licht an');
            setState("hm-rpc.0.NEQ1222700.1.STATE", true);
            setState("hm-rpc.0.NEQ1222700.2.STATE", true);
        } else {
            console.log("Licht bleibt am Tag aus ...");
        }
    } else {

```



```
        // Licht bei Garage zu verzögert ausschalten
        console.log('Tor Schließen -> Licht aus');
        setStateDelayed("hm-rpc.0.NEQ1222700.1.STATE", false, 300000,
false);    // 5 Min
        setStateDelayed("hm-rpc.0.NEQ1222700.2.STATE", false, 240000,
false);    // 4 Min
    }
});
```

Briefkasten

[download](#)

```
on({id: 'zigbee.0.00158d000572c991.contact', change: "ne"}, function
(obj) {
    var value = obj.state.val;
    var oldValue = obj.oldState.val;
    // Wenn der Briefkasten zu ist, dann ist der Schalter auf "True"
    if ((obj.state ? obj.state.val : "") == false) {
        console.log('Briefkasten : geöffnet');
        setState("Global.0.Post.Eingeworfen"/*Eingeworfen*/, true, true);
        setState("alexa2.0.Echo-
Devices.G090XG08945606VC.Commands.speak"/*speak*/, 'Es liegt Post im
Briefkasten', true);
        sendTo("telegram.0", "send", {
            text: 'Posteinwurf',
            disable_notification: true
        });
        console.log("telegram: " + 'Posteinwurf');
    }
});

// Nachts den Befüllstatus resettten
schedule('{"time":{"exactTime":true,"start":"00:00"},"period":{"days":1
}}', function () {
    console.log('Flag Reset');
    setState("Global.0.Post.Eingeworfen"/*Eingeworfen*/, false, true);
});
```

Licht nach Sonne

[download](#)

```
// Licht um 6 Uhr ein
```

```
schedule("0 6 * * *", function () {
  if (compareTime('06:00', null, ">", getAstroDate("sunrise",
undefined, 0))) {
    console.log('Haustür : Schalte das Licht ein (06:00)');
    setState("hm-
rpc.0.NEQ1504413.1.STATE"/*OD_SW_Haustue_Licht:1.STATE*/, true);
  } else {
    console.log('Haustür : Licht bleibt aus (Sonnenaufgang < 06:00)');
  }
});

// Licht um X Uhr aus nach Sonnenstand
schedule({astro: "sunriseEnd", shift: 30}, function () {
  console.log(['Haustür : Schalte das Licht aus um : ',formatDate(new
Date(), "hh:mm:ss"),' (Sunrise End : ',getAstroDate("sunriseEnd",
undefined, 0),')'].join('')));
  setState("hm-
rpc.0.NEQ1504413.1.STATE"/*OD_SW_Haustue_Licht:1.STATE*/, false);
});

// Licht Abends um X Uhr an nach Sonnenstand
schedule({astro: "sunsetStart", shift: -20}, function () {
  if (compareTime('23:00', null, "<", getAstroDate("sunsetStart",
undefined, -20))) {
    console.log(['Haustür : Schalte das Licht ein um :
',formatDate(new Date(), "hh:mm:ss"),' (Sunset Start :
',getAstroDate("sunsetStart", undefined, 0),')'].join('')));
    setState("hm-
rpc.0.NEQ1504413.1.STATE"/*OD_SW_Haustue_Licht:1.STATE*/, true);
  } else {
    console.log('Haustür : Licht bleibt aus (Sonnenuntergang >
23:00)');
  }
});

// Licht um 23 Uhr aus
schedule("0 23 * * *", function () {
  console.log('Haustür : Schalte das Licht aus (22:00)');
  setState("hm-
rpc.0.NEQ1504413.1.STATE"/*OD_SW_Haustue_Licht:1.STATE*/, false);
});

schedule({astro: "sunrise", shift: 0}, function () {
  console.debug(['Astro : Sunrise : ' + String(formatDate(new Date(),
"hh:mm:ss"))]);
});

schedule({astro: "sunsetStart", shift: 0}, function () {
  console.debug(['Astro : Sunset Start : ' + String(formatDate(new
Date(), "hh:mm:ss"))]);
});
```

```
schedule({astro: "sunriseEnd", shift: 0}, function () {
    console.debug(('Astro : Sunrise End : ' + String(formatDate(new
Date(), "hh:mm:ss"))));
});

schedule({astro: "sunset", shift: 0}, function () {
    console.debug(('Astro : Sunset : ' + String(formatDate(new Date(),
"hh:mm:ss"))));
});
```

Licht bei Bewegung Garage

[download](#)

```
// Licht Haustür bei Bewegung an und wieder aus machen
on({id: 'hm-rpc.0.NEQ0119442.1.MOTION', val: true}, function (obj) {
    var value = obj.state.val;
    var oldValue = obj.oldState.val;
    console.log('Bewegung Haustür erkannt');
    if (getState("hm-rpc.0.NEQ1222700.1.STATE").val == false) {
        console.log('Schalte das Licht Garage EIN für 4 Minuten');
        // Motion wieder zurück setzen - wird nicht immer resettet
        setState("hm-rpc.0.NEQ0119442.1.MOTION"/*OD_SB_Haustuer:1.MOTION*/,
false);
        setState("hm-
rpc.0.NEQ1222700.1.STATE"/*GR_AS_LichtAussen_2erLicht:1.STATE*/, true);
        setStateDelayed("hm-
rpc.0.NEQ1222700.1.STATE"/*GR_AS_LichtAussen_2erLicht:1.STATE*/, false,
420000, false);
        setState("hm-
rpc.0.NEQ1222700.2.STATE"/*GR_AS_LichtAussen_4erLicht:2.STATE*/, true);
        setStateDelayed("hm-
rpc.0.NEQ1222700.2.STATE"/*GR_AS_LichtAussen_4erLicht:2.STATE*/, false,
360000, false);
    } else {
        console.log('Licht draussen ist schon eingeschaltet');
    }
});
```

Heizung

[download](#)

```
// Heizung alle ausschalten / auf Default setzen
```

```
on({id: "Global.0.Heizung.AlleAus"/*AlleAus*/, val: true}, function
(obj) {
  var value = obj.state.val;
  var oldValue = obj.oldState.val;
  if (getState("Global.0.Heizung.AlleAus").val == true) {
    console.log(('Heizung : Schalte alle Heizungen auf Default /
Aus'));
    setState("Global.0.Heizung.AlleAus"/*AlleAus*/, false);
    // Erdgeschoss
    setState("hm-
rpc.0.LEQ0423081.4.SET_TEMPERATURE"/*AZ_HZ:4.SET_TEMPERATURE*/, 16);
    setState("hm-
rpc.0.MEQ0248967.4.SET_TEMPERATURE"/*WZ_HZ_Gross:4.SET_TEMPERATURE*/,
18);
    setState("hm-
rpc.0.MEQ0249461.4.SET_TEMPERATURE"/*WZ_HZ_Sofa:4.SET_TEMPERATURE*/,
19);
    setState("hm-
rpc.0.MEQ0249467.4.SET_TEMPERATURE"/*WZ_HZ_Klein:4.SET_TEMPERATURE*/,
18.5);
    setState("hm-
rpc.0.MEQ0249356.4.SET_TEMPERATURE"/*KU_HZ:4.SET_TEMPERATURE*/, 19);
    setState("hm-
rpc.0.MEQ1552083.4.SET_TEMPERATURE"/*WC_HZ:4.SET_TEMPERATURE*/, 14);
    // Keller
    setState("hm-
rpc.0.MEQ1554281.4.SET_TEMPERATURE"/*KL_HZ_Hobby:4.SET_TEMPERATURE*/,
8);
    setState("hm-
rpc.0.MEQ1883925.4.SET_TEMPERATURE"/*KL_HZ_Bastel:4.SET_TEMPERATURE*/,
0);
    setState("hm-
rpc.0.MEQ1885072.4.SET_TEMPERATURE"/*KL_HZ_Sauna:4.SET_TEMPERATURE*/,
0);
    // Erste Etage
    setState("hm-
rpc.0.MEQ1554264.4.SET_TEMPERATURE"/*LN_HZ:4.SET_TEMPERATURE*/, 17);
    setState("hm-
rpc.0.MEQ0249466.4.SET_TEMPERATURE"/*BD_HZ:4.SET_TEMPERATURE*/, 18);
    setState("hm-
rpc.0.MEQ1550232.4.SET_TEMPERATURE"/*SZ_HZ:4.SET_TEMPERATURE*/, 15);
    setState("hm-
rpc.0.MEQ1550206.4.SET_TEMPERATURE"/*GZ_HZ:4.SET_TEMPERATURE*/, 15);
  }
});
```

[download](#)

```
// Heizung um 5:30 Uhr einschalten wenn Automatik eingeschaltet
schedule("30 5 * * *", function () {
```

```

    if (getState("Global.0.Heizung.BD_Automatik").val == true) {
        console.log('Bad : Schalte Heizung ein (06:00)');
        setState("hm-
rpc.0.MEQ0249466.4.SET_TEMPERATURE"/*BD_HZ:4.SET_TEMPERATURE*/, 21);
    }
});

// Heizung um 9:30 Uhr ausschalten
schedule("30 9 * * *", function () {
    console.log('Bad : Schalte Heizung aus (09:30)');
    setState("hm-
rpc.0.MEQ0249466.4.SET_TEMPERATURE"/*BD_HZ:4.SET_TEMPERATURE*/, 18);
});

```

[download](#)

```

// Heizungen im WZ gemeinsam steuern
on({id: "Global.0.Heizung.WZ_SetTempAlle"/*WZ_SetTempAlle*/, change:
"ne"}, function (obj) {
    var value = obj.state.val;
    var oldValue = obj.oldState.val;
    console.log(['WZ Heizung : Setzte Temperatur auf
',getState("Global.0.Heizung.WZ_SetTempAlle").val,' °C'].join(''));
    setState("hm-
rpc.0.MEQ0248967.4.SET_TEMPERATURE"/*WZ_HZ_Gross:4.SET_TEMPERATURE*/,
getState("Global.0.Heizung.WZ_SetTempAlle").val);
    setState("hm-
rpc.0.MEQ0249461.4.SET_TEMPERATURE"/*WZ_HZ_Sofa:4.SET_TEMPERATURE*/,
(getState("Global.0.Heizung.WZ_SetTempAlle").val + 1));
    setState("hm-
rpc.0.MEQ0249467.4.SET_TEMPERATURE"/*WZ_HZ_Klein:4.SET_TEMPERATURE*/,
(getState("Global.0.Heizung.WZ_SetTempAlle").val + 0.5));
});

```

Licht Bewegung Küche

[download](#)

```

/*****
* Licht bei Bewegung in der Küche einschalten
*
* Geräte
* zigbee.0.14b457fffe7c749a.occupancy    KU_Bewegung
Bewegungssensor
* hm-rpc.0.NEQ0951347.1.STATE            KU_AS_Regal_Deko:1.STATE
Schaltkontakt
*****/

```

```
var Lampe_Aus;

// Bewegungsflag setzen
on({id: 'zigbee.0.14b457fffe7c749a.occupancy', change: "ne"}, function
(obj) {
    if (obj.state.val == true) {
        // Nur ausführen wenn dunkel
        if (compareTime(getAstroDate("sunriseEnd", undefined, 10),
getAstroDate("sunsetStart", undefined, -10), "not between", null)) {
            // Soll der Bewegungsmelder genutzt werden ?
            if
(getState("Global.0.Automatik.KU_Bewegungsmelder_AnAus").val == true) {
                setState("Global.0.Räume.Küche.KU_BewegungErkannt",
true);
                if (Lampe_Aus) {clearTimeout(Lampe_Aus); Lampe_Aus =
null;};
                Lampe_Aus = setTimeout(function () {
                    setState("Global.0.Räume.Küche.KU_BewegungErkannt",
false);
                }, 360000); // 6 Minuten
            }
        }
    }
});

// Tastendruck auswerten und Flag setzen
on({id: 'hm-rpc.0.MEQ1844894.1.INSTALL_TEST', change: "any"}, function
(obj) {
    var motion =
getState("Global.0.Räume.Küche.KU_BewegungErkannt").val;
    var press = getState("Global.0.Räume.Küche.KU_LichtSpüleAn").val;
    // Wenn das Licht aus ist -> an
    if (!motion && !press){
        setState("Global.0.Räume.Küche.KU_LichtSpüleAn", true);
        console.log("Tastendruck - Licht an");
    }

    // Wenn das Licht an ist -> aus
    if (motion || press) {
        setState("Global.0.Räume.Küche.KU_LichtSpüleAn", false);
        setState("Global.0.Räume.Küche.KU_BewegungErkannt", false);
        console.log("Tastendruck - Licht AUS");
    }
});

// Licht einschalten wenn sich ein Status ändert
on({id: new RegExp('Global\\.0\\.Räume\\.Küche\\.KU_BewegungErkannt' +
'|' + 'Global\\.0\\.Räume\\.Küche\\.KU_LichtSpüleAn'), change: "ne"},
function (obj) {
    var motion =
getState("Global.0.Räume.Küche.KU_BewegungErkannt").val;
```

```

    var press = getState("Global.0.Räume.Küche.KU_LichtSpüleAn").val;
    console.log("Licht schalten - Motion : " + motion + ", Switch : " +
    press + ", State : " + motion || press);
    setState("hm-rpc.0.NEQ0951347.1.STATE", motion || press);
  });

// Licht ausschalten wenn lange gedrückt
// on({id: 'hm-rpc.0.MEQ1844894.1.PRESS_LONG_RELEASE', change: "any"},
function (obj) {
//   console.log("Langer Tastendruck - Licht aus");
//   setState("Global.0.Räume.Küche.KU_LichtSpüleAn", false);
//   setState("Global.0.Räume.Küche.KU_BewegungErkannt", false);
// });

```

Rollos

[download](#)

```

/*****
* Rollos
* Level 0 (unten) bis 100 (oben) Mitte ist eher 80
*
* hm-rpc.0.NEQ1214347.1.LEVEL    WZ_RL_ErkerLinks:1.LEVEL
* hm-rpc.0.LEQ1025898.1.LEVEL    WZ_RL_ErkerMitte:1.LEVEL
* hm-rpc.0.QEQ1928719.1.LEVEL    WZ_RL_ErkerRechts:1.LEVEL
* hm-rpc.0.LEQ1026100.1.LEVEL    WZ_RL_SeiteGross:1.LEVEL
* hm-rpc.0.NEQ1562563.1.LEVEL    WZ_RL_SeiteKlein:1.LEVEL
* hm-rpc.0.LEQ1025905.1.LEVEL    WZ_RL_Strasse:1.LEVEL
* hm-rpc.0.LEQ1025913.1.LEVEL    KU_RL_Einfahrt:1.LEVEL
* hm-rpc.0.PEQ0183133.1.LEVEL    KU_RL_Feld:1.LEVEL
* hm-rpc.0.LEQ1032151.1.LEVEL    AZ_RL_Strasse:1.LEVEL
* hm-rpc.0.LEQ1032236.1.LEVEL    AZ_RL_Einfahrt:1.LEVEL
*/

on({id: 'Global.0.Rollos.Control', change: "ne"}, function (obj) {
  var value = obj.state.val;
  var oldValue = obj.oldState.val;

  switch (value) {
    // Für Tags alle hochfahren
    case "allup":
      setStateDelayed("hm-rpc.0.NEQ1214347.1.LEVEL", 100, 0,
false); // WZ_RL_ErkerLinks
      setStateDelayed("hm-rpc.0.LEQ1025898.1.LEVEL", 100, 1000,
false); // WZ_RL_ErkerMitte
      setStateDelayed("hm-rpc.0.QEQ1928719.1.LEVEL", 100, 2000,
false); // WZ_RL_ErkerRechts
      setStateDelayed("hm-rpc.0.LEQ1026100.1.LEVEL", 100, 3000,

```

```
false); // WZ_RL_SeiteGross
    setStateDelayed("hm-rpc.0.NEQ1562563.1.LEVEL", 50, 4000,
false); // WZ_RL_SeiteKlein
    setStateDelayed("hm-rpc.0.LEQ1025905.1.LEVEL", 100, 5000,
false); // WZ_RL_Strasse
    setStateDelayed("hm-rpc.0.LEQ1025913.1.LEVEL", 100, 6000,
false); // KU_RL_Einfahrt
    setStateDelayed("hm-rpc.0.PEQ0183133.1.LEVEL", 100, 7000,
false); // KU_RL_Feld
    setStateDelayed("hm-rpc.0.LEQ1032151.1.LEVEL", 40, 8000,
false); // AZ_RL_Strasse
    setStateDelayed("hm-rpc.0.LEQ1032236.1.LEVEL", 70, 9000,
false); // AZ_RL_Einfahrt
    break;
// Für Nachts alle absenken
case "alldown":
    setStateDelayed("hm-rpc.0.NEQ1214347.1.LEVEL", 0, 0,
false); // WZ_RL_ErkerLinks
    setStateDelayed("hm-rpc.0.LEQ1025898.1.LEVEL", 0, 1000,
false); // WZ_RL_ErkerMitte
    // Nicht schließen wenn Terrassentür offen
    if (getState("hm-rpc.0.NEQ1481485.1.STATE").val == false) {
        setStateDelayed("hm-rpc.0.QEQ1928719.1.LEVEL", 0,
2000, false); // WZ_RL_ErkerRechts
    } else {
        console.log("WZ_RL_ErkerRechts wird nicht geschlossen
-> Terrassentür auf");
    }
    setStateDelayed("hm-rpc.0.LEQ1026100.1.LEVEL", 0, 3000,
false); // WZ_RL_SeiteGross
    setStateDelayed("hm-rpc.0.NEQ1562563.1.LEVEL", 0, 4000,
false); // WZ_RL_SeiteKlein
    setStateDelayed("hm-rpc.0.LEQ1025905.1.LEVEL", 0, 5000,
false); // WZ_RL_Strasse
    setStateDelayed("hm-rpc.0.LEQ1025913.1.LEVEL", 0, 6000,
false); // KU_RL_Einfahrt
    setStateDelayed("hm-rpc.0.PEQ0183133.1.LEVEL", 0, 7000,
false); // KU_RL_Feld
    setStateDelayed("hm-rpc.0.LEQ1032151.1.LEVEL", 0, 8000,
false); // AZ_RL_Strasse
    setStateDelayed("hm-rpc.0.LEQ1032236.1.LEVEL", 0, 9000,
false); // AZ_RL_Einfahrt
    break;
// Arbeitszimmer abdunkeln
case "azdark":
    setStateDelayed("hm-rpc.0.LEQ1032151.1.LEVEL", 15, 0,
false); // AZ_RL_Strasse
    setStateDelayed("hm-rpc.0.LEQ1032236.1.LEVEL", 25, 1000,
false); // AZ_RL_Einfahrt
    break;
// Arbeitszimmer aufhellen
```



```
        case "azlight":
            setStateDelayed("hm-rpc.0.LEQ1032151.1.LEVEL", 40, 0,
false); // AZ_RL_Strasse
            setStateDelayed("hm-rpc.0.LEQ1032236.1.LEVEL", 70, 1000,
false); // AZ_RL_Einfahrt
            break;
        // Rollos im Erker auf 40% runter wegen Sonne
        case "erkersun":
            setStateDelayed("hm-rpc.0.NEQ1214347.1.LEVEL", 70, 0,
false); // WZ_RL_ErkerLinks
            setStateDelayed("hm-rpc.0.LEQ1025898.1.LEVEL", 70, 1000,
false); // WZ_RL_ErkerMitte
            if (getState("hm-rpc.0.NEQ1481485.1.STATE").val == false) {
                setStateDelayed("hm-rpc.0.QEQ1928719.1.LEVEL", 70,
2000, false); // WZ_RL_ErkerRechts
            } else {
                console.log("WZ_RL_ErkerRechts wird nicht geschlossen
-> Terrassentür auf");
            }
            break;
        // Rolles um 30% Runter im Wohnzimmer für Vögel
        case "wz_birds":
            setStateDelayed("hm-rpc.0.NEQ1214347.1.LEVEL", 82, 0,
false); // WZ_RL_ErkerLinks
            setStateDelayed("hm-rpc.0.LEQ1025898.1.LEVEL", 82, 1000,
false); // WZ_RL_ErkerMitte
            if (getState("hm-rpc.0.NEQ1481485.1.STATE").val == false) {
                setStateDelayed("hm-rpc.0.QEQ1928719.1.LEVEL", 90,
2000, false); // WZ_RL_ErkerRechts
            } else {
                console.log("WZ_RL_ErkerRechts wird nicht geschlossen
-> Terrassentür auf");
            }
            setStateDelayed("hm-rpc.0.LEQ1026100.1.LEVEL", 82, 3000,
false); // WZ_RL_SeiteGross
            setStateDelayed("hm-rpc.0.NEQ1562563.1.LEVEL", 50, 4000,
false); // WZ_RL_SeiteKlein
            setStateDelayed("hm-rpc.0.LEQ1025905.1.LEVEL", 70, 5000,
false); // WZ_RL_Strasse
            break;
        // alle anhalten
        case "stop":
            setState("hm-rpc.0.NEQ1214347.1.STOP", true); //
WZ_RL_ErkerLinks
            setState("hm-rpc.0.LEQ1025898.1.STOP", true); //
WZ_RL_ErkerMitte
            setState("hm-rpc.0.QEQ1928719.1.STOP", true); //
WZ_RL_ErkerRechts
            setState("hm-rpc.0.LEQ1026100.1.STOP", true); //
WZ_RL_SeiteGross
            setState("hm-rpc.0.NEQ1562563.1.STOP", true); //
```

```
WZ_RL_SeiteKlein
    setState("hm-rpc.0.LEQ1025905.1.STOP", true); //
WZ_RL_Strasse
    setState("hm-rpc.0.LEQ1025913.1.STOP", true); //
KU_RL_Einfahrt
    setState("hm-rpc.0.PEQ0183133.1.STOP", true); //
KU_RL_Feld
    setState("hm-rpc.0.LEQ1032151.1.STOP", true); //
AZ_RL_Strasse
    setState("hm-rpc.0.LEQ1032236.1.STOP", true); //
AZ_RL_Einfahrt
    break;
}

setState("Global.0.Rollos.Control", "done");
});
```

Terasse

[download](#)

```
on({id: 'Global.0.Markiese.MoveTopOut', change: "any"}, async function
(obj) {
    var value = obj.state.val;
    var oldValue = obj.oldState.val;
    if (getState("Global.0.Markiese.MoveTopOut").val == true) {
        if (getState("Global.0.Automatik.OD_Markiese_Sommerbetrieb").val ==
true) {
            console.log('Markiese Oben : Ausfahren aktiviert ...');
            setState("Global.0.Markiese.MoveTopOut/*MoveTopOut*/, false);
            setState("mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Oben_Rein/*mqtt.0
.Schaltschrank.Switch_Bank_2.Markise_Oben_Rein*/", '0');
            setState("mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Oben_Raus/*mqtt.0
.Schaltschrank.Switch_Bank_2.Markise_Oben_Raus*/", '1');
            // Nach 60 Sekunden den Ausgang wieder abschalten !
            setStateDelayed("mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Oben_Raus"/
*mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Oben_Raus*/", '0', 60000,
false);
        } else {
            console.warn('Markiese Oben : Winterbetrieb aktiviert - kein
Ausfahren !');
        }
    }
});

on({id: 'Global.0.Markiese.MoveTopIn', change: "any"}, async function
(obj) {
    var value = obj.state.val;
```

```
var oldValue = obj.oldState.val;
if (getState("Global.0.Markiese.MoveTopIn").val == true) {
    console.log('Markiese Oben : Einfahren aktiviert ...');
    setState("Global.0.Markiese.MoveTopIn"/*MoveTopIn*/, false);
setState("mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Oben_Raus"/*mqtt.0
.Schaltschrank.Switch_Bank_2.Markise_Oben_Raus*/, '0');
setState("mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Oben_Rein"/*mqtt.0
.Schaltschrank.Switch_Bank_2.Markise_Oben_Rein*/, '1');
    // Nach 60 Sekunden den Ausgang wieder abschalten !
setStateDelayed("mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Oben_Rein"/*
mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Oben_Rein*/, '0', 60000,
false);
}
});

on({id: 'Global.0.Markiese.MoveTopStop', change: "any"}, async function
(obj) {
    var value = obj.state.val;
    var oldValue = obj.oldState.val;
    if (getState("Global.0.Markiese.MoveTopStop").val == true) {
        console.log('Markiese Oben : NOTSTOP ...');
setState("mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Oben_Rein"/*mqtt.0
.Schaltschrank.Switch_Bank_2.Markise_Oben_Rein*/, '0');
setState("mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Oben_Raus"/*mqtt.0
.Schaltschrank.Switch_Bank_2.Markise_Oben_Raus*/, '0');
        setState("Global.0.Markiese.MoveTopOut"/*MoveTopOut*/, false);
        setState("Global.0.Markiese.MoveTopIn"/*MoveTopIn*/, false);
        setState("Global.0.Markiese.MoveTopStop"/*MoveTopStop*/, false);
    }
});

on({id: 'Global.0.Markiese.MoveFrontOut', change: "any"}, async
function (obj) {
    var value = obj.state.val;
    var oldValue = obj.oldState.val;
    if (getState("Global.0.Markiese.MoveFrontOut").val == true) {
        if (getState("Global.0.Automatik.0D_Markiese_Sommerbetrieb").val ==
true) {
            console.log('Markiese Vorne : Ausfahren aktiviert ...');
            setState("Global.0.Markiese.MoveFrontOut"/*MoveFrontOut*/,
false);
setState("mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Vorne_up"/*mqtt.0.
Schaltschrank.Switch_Bank_2.Markise_Vorne_up*/, '0');
setState("mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Vorne_down"/*mqtt.
0.Schaltschrank.Switch_Bank_2.Markise_Vorne_down*/, '1');
            // Nach 60 Sekunden den Ausgang wieder abschalten !
setStateDelayed("mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Vorne_down"
/*mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Vorne_down*/, '0', 60000,
false);
        } else {
            console.warn('Markiese Vorne : Winterbetrieb aktiviert - kein
```

```
Ausfahren !');
    }
  }
});

on({id: 'Global.0.Markiese.MoveFrontIn', change: "any"}, async function
(obj) {
  var value = obj.state.val;
  var oldValue = obj.oldState.val;
  if (getState("Global.0.Markiese.MoveFrontIn").val == true) {
    console.log('Markiese Vorne : Einfahren aktiviert ...');
    setState("Global.0.Markiese.MoveFrontIn"/*MoveFrontIn*/, false);
    setState("mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Vorne_down"/*mqtt.
0.Schaltschrank.Switch_Bank_2.Markise_Vorne_down*/, '0');
    setState("mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Vorne_up"/*mqtt.0.
Schaltschrank.Switch_Bank_2.Markise_Vorne_up*/, '1');
    // Nach 60 Sekunden den Ausgang wieder abschalten !
    setStateDelayed("mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Vorne_up"/*
mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Vorne_up*/, '0', 60000,
false);
  }
});

on({id: 'Global.0.Markiese.MoveFrontStop', change: "any"}, async
function (obj) {
  var value = obj.state.val;
  var oldValue = obj.oldState.val;
  if (getState("Global.0.Markiese.MoveFrontStop").val == true) {
    console.log('Markiese Vorne : NOTSTOP ...');
    setState("mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Vorne_down"/*mqtt.
0.Schaltschrank.Switch_Bank_2.Markise_Vorne_down*/, '0');
    setState("mqtt.0.Schaltschrank.Switch_Bank_2.Markise_Vorne_up"/*mqtt.0.
Schaltschrank.Switch_Bank_2.Markise_Vorne_up*/, '0');
    setState("Global.0.Markiese.MoveFrontOut"/*MoveFrontOut*/, false);
    setState("Global.0.Markiese.MoveFrontIn"/*MoveFrontIn*/, false);
    setState("Global.0.Markiese.MoveFrontStop"/*MoveFrontStop*/,
false);
  }
});
```

From:

<https://drklipper.de/> - Dr. Klipper Wiki

Permanent link:

<https://drklipper.de/doku.php?id=haussteuerung:automatiken>Last update: **2025/10/22 06:44**