

homeputer CL

Studio-Version

für HomeMatic

Version 3.00

copyright (c)

contronics GmbH 2007

www.contronics.de

Die **homeputer-Software** bietet die Möglichkeit komplexe logische Funktionen zur Steuerung des Systems auf einfache Weise zu realisieren. Dies geschieht über eine einfach zu erlernende Makrosprache mit einem grossen Funktionsumfang. Das homeputer-Programm erlaubt zudem umfangreiche individuelle Visualisierungen. In der Studio-Version und CLX-Version können beliebig viele individuelle Ansichten erstellt werden, in der Standard-Version ist die Anzahl auf drei beschränkt, in der Starter-Version auf eine individuelle Ansicht.

Für jedes Objekt können unterschiedliche Darstellungsformen für die einzelnen Ansichten definiert werden. Die Visualisierungen können optional mit Hintergrundbildern, z.B. Grundrissen eines Hauses, versehen werden.

Aktuelle Informationen zur homeputer-Software für das HomeMatic-System finden Sie im Internet unter:
www.contronics.de/html/homematic.html

Zum Anlernen und zur Konfiguration von Modulen und auch zur direkten Verknüpfung von Modulen wird die WEB-UI der CCU bzw. bei anderen Steuerungszentralen wie der CL-Box die mitgelieferte WEB-UI für den PC benutzt.

Die homeputer-Software wird zur Erstellung von Zeittabellen, intelligenten Funktionen und Visualisierungen benutzt und erlaubt die Erstellung sehr komplexer Anwendungen.

Falls eine CCU benutzt wird, können die implementierten Programmierungsoptionen der CCU unabhängig von der homeputer-Software weiter benutzt werden. Dabei sollten Sie aber beachten, dass es dann schwierig werden kann den Überblick über die programmierten Funktionen zu behalten wenn beide Möglichkeiten der Programmierung gleichzeitig benutzt werden.

Falls eine HomeMatic-CCU als Zentrale benutzt und über eine USB-Verbindung an den PC angeschlossen wird, ist standardmässig eine IP-Adresse für die CCU vorgegeben. Zum Zeitpunkt der Erstellung dieser Beschreibung ist das die IP-Adresse 10.101.81.51.

Sollte diese Adresse nicht mehr aktuell sein können Sie die korrekte Adresse im entsprechenden Kapitel über die USB-Verbindung in der Bedienungsanleitung der CCU finden.

In den Teilen, in denen die verschiedenen Software-Versionen identisch sind, sind auch die Hilfetexte identisch. An einigen Stellen der Hilfetexte kann es daher vorkommen, dass Funktionen beschrieben sind, die in der verwendeten Software-Version nicht zur Verfügung stehen. An diesen Stellen befindet sich dann ein entsprechender Hinweis.

Sollte dieser Hinweis im Einzelfall fehlen oder missverständlich sein, bitten wir um einen entsprechenden Hinweis um dies richtig zu stellen.

Die homeputer-Software besteht aus mehreren Programmen:

1. Das **homeputer CL-Konfigurationsprogramm**

Mit diesem Programm werden die Anwendungen auf einem PC mit Betriebssystem Windows erstellt. Diese Konfigurationen und Programmierungen umfassen Zeittabellen zur Anwesenheitssimulation und zum Schalten einzelner Objekte, Makros zur Ablaufsteuerung und logischen Verknüpfung von Objekten und Funktionen und Definitionen zur Visualisierung der Anwendung.

Wenn die Anwendung mit allen Funktionen definiert und programmiert ist, wird mit einem Compiler (der aus dem Konfigurationsprogramm aufgerufen wird) eine Datei mit optimiertem binärem Ausführungscode erstellt. Dieser Ausführungscode wird auf die HomeMatic-Zentrale übertragen und wird dort selbständig, ohne dass ein PC eingeschaltet sein muss von der

2. **Exec-Engine** verarbeitet.

Die Exec-Engine ist ein Programm, das auf der Hardware der Zentrale läuft und aufgrund der erstellten Makros, Zeittabellen und Abhängigkeiten die logische Steuerung übernimmt. Die Exec-Engine stellt auch die Daten für Visualisierungen zur Verfügung. Die eigentlichen Visualisierungen werden vom Programm

3. **VisuWin** angezeigt.

Das Programm homeputer CL VisuWin kann auf einem oder mehreren PCs im Netzwerk installiert werden und erlaubt die Visualisierung, Kontrolle und Steuerung der Anwendung am PC-Bildschirm. Dabei können bis zu drei PC's gleichzeitig benutzt werden (in erweiterten Versionen der Studio-Software auch mehr, auf Anfrage). Alle definierten Ansichten können in unterschiedlichen eigenen Fenstern auf dem PC-Bildschirm angezeigt werden, wobei es auch möglich ist per Mausclick Zustände zu verändern und so Steuerungen direkt am PC-Bildschirm vorzunehmen.

4. ExecEngineWin

ist ein weiteres Programm, das mit der auf der Zentrale laufenden Exec-Engine kommuniziert, den Status der Exec-Engine anzeigt, die Ausführung starten und anhalten kann und für die Übertragung des erstellten Ausführungscodes vom PC auf die HomeMatic-Zentrale zuständig ist. Dies passiert z.B. nach einer Änderung des Ausführungscodes mit einem Knopfdruck.

Das Funktions-Prinzip ist einfach:

Der Ausführungsteil der Software (die Exec-Engine) empfängt Meldungen von den Sensoren (=Sendern des HomeMatic-Systems). Für diese kann ein Makro (=eine Folge von Anweisungen) erstellt werden. Dieses Makro wird ausgeführt, sobald eine Meldung vom zugehörigen Sensor empfangen wird. In dem Makro können die Aktoren (Empfänger) geschaltet werden, aber auch andere Aktionen ausgeführt werden. So ist es z.B. möglich Windows-Programme zu starten oder als Hinweis für bestimmte Situationen Wave-Dateien auf einem Visualisierungs-PC abzuspielen. Weiterhin können Schaltungen über Zeittabellen definiert werden, in der Studio-Version können Makros zudem in bestimmten Zeitintervallen (von Sekundenbruchteilen bis stündlich und täglich) aktiviert werden. Wenn Zustände von Objekten durch Makros verändert werden, so wird der neue Zustand nicht bei Ausführung der Makro-Anweisung, sondern erst nach Beendigung des Makros oder bei Ausführung einer WARTEN-Anweisung in dem Makro an den Empfänger gesendet.

Die umfangreichen Möglichkeiten der Software erlauben die Realisierung sehr grosser und komplexer Anwendungen.

Software-Installation

Systemvoraussetzungen des PC

- Standard-PC mit mindestens einem Pentium-Prozessor
- CD-Laufwerk
- Festplatte mit mindestens 10 MB freiem Speicher
- Grafikkarte mit einer Auflösung von mindestens 800x600 Pixel
- Betriebssystem Windows 2000, XP oder höher
- Maus oder ein kompatibles Zeigegerät

Installation der Software

Installation der Software auf dem PC:

Das Installationsprogramm kopiert alle Dateien und richtet die Anwendung auf Ihrem Computer ein. Sie müssen die Installation nur einmal durchführen.

Das PC-Programm ist ein Windows-Programm. Sie benötigen das Betriebssystem Windows 2000, XP oder höher auf Ihrem Rechner.

Eventuell laufende Programme sollten Sie beenden, da sie den Installationsvorgang behindern können.

1. Legen Sie die CD in das CD-Laufwerk ein.
2. Das Installationsprogramm wird je nach Einstellung Ihres Rechners automatisch gestartet.
3. Wenn das Installationsprogramm automatisch startet weiter mit Schritt 7.
4. Starten Sie den Windows-Explorer, wenn sie diesen nicht als Programm in Ihrer Liste haben doppelklicken Sie einfach auf "Eigene Dateien", klicken dann auf Ordner, wählen im linken Fensterteil Arbeitsplatz.
5. Wählen Sie das Laufwerk aus, in dem die CD eingelegt ist.
6. In der rechten Fensterhälfte sehen Sie einige Dateien. Doppelklicken Sie auf die Datei **CDInst** oder wenn diese nicht vorhanden ist auf STARTSETUP bzw. STARTSETUP.EXE
7. Folgen Sie den Hinweisen des Installationsprogramms.

Installation der Software auf der HomeMatic-Zentrale:

1. Starten Sie die WEB-Oberfläche der Zentrale durch Eingabe der IP-Adresse der Zentrale in einem Internet-Browser.
2. Wählen Sie den Menüpunkt **Systemkonfiguration** und dort den Unterpunkt **Systemsteuerung**.
3. Klicken Sie auf die Box **Zusatzsoftware**
4. Es öffnet sich ein Fenster in dem der Speicherort für die Software angegeben werden muss. Wählen Sie das CD-Laufwerk und dort das Unterverzeichnis HOMEMATIC-SOFTWARE Wählen Sie in diesem Unterverzeichnis die Datei **HPInst.IMG**
5. Klicken Sie dann auf den Knopf **[Installieren]**.

6. Es erscheint ein neues Fenster, klicken Sie in diesem Fenster auf den Knopf **[Installation starten]**
7. Die Software für die Zentrale wird nun automatisch installiert, nach der Installation wird die Zentrale automatisch neu gestartet.

PLN / Freigabecode eingeben

Die Software muss über die PLN (Persönlichen Lizenz-Nummer) oder einen Freigabecode aktiviert werden, um ohne zeitliche Einschränkung benutzt werden zu können.

Je nach Version wird bei ersten Starten des PC-Programms nach der PLN oder nach einem Freigabecode gefragt.

Wenn nach der PLN gefragt wird geben Sie diese einfach ein, wird nach einem Freigabecode gefragt holen Sie mit Ihrer PLN einen Freigabecode für die angezeigte Lizenznummer auf den Internetseiten von contronics unter <http://www.contronics.online.de/cgi-bin/RelCodeHM.cgi>

Diese Freigabe ist nur beim ersten Programmstart und manchmal nach einem Update erforderlich.

Bewahren Sie Ihre PLN gut auf, es kann sein, dass Sie diese benötigen um künftige Updates zu erhalten bzw. freizuschalten.

Der Schutz der Software über PLN und Freigabecode dient dazu die illegale Nutzung des Programms zu unterbinden.

Damit wird auch Ihre Investition geschützt. Denn nur wenn Weiterentwicklung, Pflege und Erweiterungen der Software aus neuen Lizenzgebühren finanziert werden können, sind wir in der Lage Updates für registrierte Benutzer in Zukunft kostenfrei bzw. kostengünstig zur Verfügung stellen zu können.

Einstellungen nach der Installation

Nach der Installation der Software müssen einige Einstellungen vorgenommen werden. Diese betreffen hauptsächlich die Kommunikation mit der HomeMatic-Zentrale.

Auf der HomeMatic-CCU muss die ExecEngine installiert werden, bevor eine Verbindung zur Zentrale hergestellt werden kann. Wie das geschieht ist unter [Software-Installation](#) beschrieben. Ob die Software ExecEngine auf der Zentrale installiert ist, können Sie in der WEB-UI überprüfen mit dem Menüpunkt: Systemsteuerung->Systemkonfiguration->Zusatzsoftware

In dem Fenster Zusatzsoftware muss "homeputer CL" angezeigt werden.

Auf der Zentrale muss eine feste IP-Adresse eingestellt werden, bevor die Verbindung konfiguriert wird. Wie das gemacht wird ist in der Bedienungsanleitung beschrieben.

Im Konfigurationsprogramm muss im [Fenster Hardware](#) die IP-Adresse der Zentrale und der Port für die [Exec-Engine](#) eingegeben werden. Standardmässig wird Port 2110 benutzt, diesen sollten Sie nicht ohne Grund ändern. Weiterhin muss in diesem Fenster auf der Seite *eigene IP* die eigene IP-Adresse ausgewählt werden, falls der PC über mehr als eine Netzwerkkarte verfügt.

Nachdem das passiert ist können Module aus der Zentrale importiert werden.

Bitte stellen Sie auf der Seite "Verzeichnisse" des Einstellungsfensters die Verzeichnisse für Projektdateien und Bitmaps ein. Im Normalfall sind diese Verzeichnisse für Projektdateien und Bitmaps bei der Installation schon eingerichtet und eingetragen worden.

Diese Einstellungen müssen vor der ersten Erstellung eines eigenen Projekts vorgenommen werden, damit dieses in dem Verzeichnis gespeichert werden kann und dort auch die Ausführungsdatei für die Zentrale abgelegt werden kann.

Bevor die Datei mit dem Ausführungscode eines erstellten Projektes zur Zentrale übertragen werden kann muss die Verbindung zur Zentrale konfiguriert werden und es muss eine Konfigurationsdatei zur Zentrale übertragen werden. Die Einrichtung der Verbindung kann mit dem Menüpunkt *Zentralenverbindung einrichten* im Menü *Konfigurieren* durchgeführt werden.

Das Erstellen und Übertragen des Ausführungscode zur Zentrale kann mit dem "Blitz"-Button oder dem Menüpunkt *Compilieren und übertragen* im Menü *Ausführung* erfolgen. Dabei werden alle notwendigen Einstellungen und Aktionen automatisch durchgeführt. Sollte nach einer Übertragung die Statusanzeige in *ExecEngineWin* nach einiger Zeit nicht die grüne Aktiv-Meldung bringen, kann es helfen die Zentrale einmal zu reseten. Bei der HomeMatic-CCU kann man das mit dem kleinen schwarzen Knopf über dem Netzwerkanschluss tun.

Je nach eventuell notwendigen Einstellungen bezüglich der Netzwerkverbindungen (Firewall, Portfreigabe im Router) kann es vorkommen, dass über den Menüpunkt *Zentralenverbindung einrichten* die Verbindung nicht hergestellt werden kann oder die automatische Übertragung des Ausführungscode nicht funktioniert.

Dann kann die Einrichtung der Verbindung und die Übertragung manuell vorgenommen werden. Wie das funktioniert ist im Folgenden beschrieben:

Vor der Übertragung einer Ausführungsdatei an die Exec-Engine müssen im Programm [ExecEngineWin](#) die folgenden Schritte durchgeführt werden. Das Programm *ExecEngineWin* wird aufgerufen mit dem Button **[Kontrollprogramm für Zentrale]** im Fenster *Hardware*.

1. Auf der Seite Einstellungen müssen IP-Adresse der Zentrale und Port der Exec-Engine (2110) eingetragen werden. Danach muss die Konfigurationsdatei durch Mausclick auf den Knopf **[Konfigurationsdatei übertragen]** zur Exec-Engine übertragen werden.

2. Auf der Seite Dateien die *aktuelle Ausführungsdatei* eintragen und die Startdatei mit dem Namen dieser Datei durch Mausclick auf den Knopf **[Startdatei einstellen]** zur Exec-Engine übertragen.

3. Auf der Seite Dateien den Pfad für die Ausführungsdatei angeben die aktuelle Ausführungsdatei als *Zu übertragende Datei* angeben und mit dem Knopf **[Übertragen]** zur Exec-Engine übertragen. Im Eingabefeld "Pfad für Ausführungsdateien" muss der Pfad für die zu übertragende Ausführungsdatei angegeben werden, im Feld "zu übertragende Ausführungsdatei" darf nur der Dateiname stehen, ohne den Pfad zur Datei. Das ist erforderlich, weil die Datei im Dateisystem der Zentrale unabhängig vom Pfad der Datei auf dem PC

abgespeichert wird.

4. Auf der Seite Steuerung/Status die Exec-Engine mit dem Knopf **[Exec-Engine starten]** starten. Bitte beachten Sie, dass bei jedem Start der Exec-Engine als erstes die Zustände und Werte aller angeschlossenen Module ausgelesen werden, daher kann es in Abhängigkeit von der Anzahl der Module einige Sekunden (bei sehr vielen Modulen auch länger als 10 Sekunden) dauern bis diese im normalen Betrieb läuft.

Nachdem diese Schritte erledigt sind, sind alle erforderlichen Daten auf der Zentrale gespeichert. Bei einem Neustart der Zentrale startet die Exec-Engine automatisch mit der aktuell eingestellten Ausführungsdatei.

Wenn die aktuelle Ausführungsdatei verändert wird, kann diese durch Anklicken des Knopfes **[Aktuelle Datei übernehmen und ausführen]** übertragen werden.

Dabei wird automatisch die aktuelle Ausführung beendet, die eingestellte aktuelle Datei übertragen und die Ausführung wieder gestartet.

Wenn sich der Name der aktuellen Ausführungsdatei ändert ist etwas mehr zu tun:

1. Exec-Engine beenden mit dem entsprechenden Knopf auf der Seite Steuerung/Status,
2. Den neuen Dateinamen unter "aktuelle Ausführungsdatei" eintragen und die Startdatei mit "Startdatei einstellen" übertragen,
3. den neuen Namen unter "Zu übertragende Ausführungsdatei" eintragen und mit dem Knopf "Übertragen" zur Zentrale übertragen.
4. Exec-Engine wieder starten mit dem entsprechenden Knopf auf der Seite Steuerung/Status.

Installationshinweise Hardware

Funkmodule sollten nicht in der Nähe grosser Metallflächen oder direkt neben oder auf elektronischen Geräte aufgestellt werden.

Die Reichweite von Funkmodulen im Gebäude ist abhängig von den individuellen Örtlichkeiten und wird durch Decken, Wände und elektromagnetische Strahlungen vermindert.

Beachten Sie unbedingt die Sicherheitshinweise der jeweiligen Bedienungsanleitungen bei der Installation von elektrischen Komponenten.

Die wichtigsten Begriffe des Systems

Ein **Projekt** ist die komplette individuell erstellte Anwendung.

Makro ist die Bezeichnung für eine oder mehrere Anweisungen, die bestimmte Aktionen bewirken. Das System verfügt über leistungsfähige Anweisungen, mit denen auch sehr komplexe Funktionen realisiert werden können. In Makros können Aktionen über wenn-Anweisungen von bestimmten Bedingungen abhängig gemacht werden. Das können Zustände bzw. Werte von Objekten, aber auch z.B. Wochentag und Uhrzeit sein.

Aktoren sind Module, die Endgeräte wie Beleuchtungen, Jalousien usw. schalten und steuern.

Sensoren sind Eingabelemente wie z.B. Taster, Fernbedienungen, Bewegungsmelder und Temperatursensoren.

Objekt ist der Oberbegriff für Sensoren und Aktoren bei der Bearbeitung innerhalb des PC-Programms, wobei der Begriff „Objekt“ auch die Definitionen und Makros des Sensors bzw. Aktors mit einschließt. Wenn das Steuerungsprogramm Objektzustände von Aktor-Objekten verändert, wird automatisch eine entsprechende Meldung an den Aktor gesendet. Wenn ein Sensor eine Meldung sendet, wird das Makro des Sensor-Objekts ausgeführt. In der Studio-Version gibt es die Möglichkeit Objekte ohne zugeordnete Hardware zu definieren.

Ansichten zeigen die Zustände und Werte von Objekten und ermöglichen je nach Objekttyp die Änderung von Zuständen und Werten am Bildschirm.

In der Standard-Version können bis zu drei, in der Studio-Version beliebig viele individuelle Ansichten erstellt werden. Diesen Ansichten kann optional ein Hintergrundbitmap zugewiesen werden.

Welche Darstellungsart verwendet werden soll, kann für jede Ansicht in der jeweiligen Definition eines Objekts definiert werden.

Erste Schritte

Einige allgemeine Hinweise:

bitte lesen Sie auch die Seiten [Allgemein](#) und [Die wichtigsten Begriffe](#)

bitte nehmen Sie nach der Installation unbedingt die auf der Seite [Einstellungen nach der Installation](#) beschriebenen Einstellungen vor.

Um zu prüfen ob alle Einstellungen korrekt sind (und um ein schnelles Erfolgserlebnis zu haben) wäre der erste Schritt

Eine einfache Visualisierung

ohne logische Funktionen und eigene Ansichten ist relativ einfach machbar. Als erstes muss eine Verbindung zur Zentrale konfiguriert werden. Die Einrichtung der Verbindung kann mit dem Menüpunkt *Zentralenverbindung einrichten* im Menü *Konfigurieren* durchgeführt werden. Falls diese automatische Verbindungseinrichtung Fehlermeldungen anzeigt und nicht funktioniert nehmen Sie bitte die einzelnen Schritte der Einrichtung manuelle vor wie unter [Einstellungen nach der Installation](#) beschrieben.

- Lernen Sie einige Module in der Zentrale an. Übernehmen Sie die Module mit dem Knopf "Module importieren" im Fenster Modulauswahl.

- Klicken Sie auf den "Blitz"-Button oder wählen im Menü *Ausführung* den Punkt *Compilieren und übertragen*

- Starten Sie das Programm [ExecEngineWin](#) um zu kontrollieren ob die ExecEngine auf der Zentrale aktiv ist. Bitte beachten Sie, dass diese nur aktiv sein kann nachdem ein Projekt zur Zentrale übertragen wurde.

Dieses Programm wird wie folgt gestartet: Menü Konfigurieren -> Hardware-Schnittstelle dann auf den Knopf "Kontrollprogramm für Zentrale" klicken.

Auf der Seite muss nach der Übertragung eines Projekts der Status Aktiv angezeigt werden (Knopf oben links ist grün). Falls

- Starten Sie das Programm *VisuWin* (Ausführung->Visualisierung oder Klick auf den "Auge"-Button) und wählen die Ansicht "Alle Objekte".

Nun sehen Sie alle Objekte in einer Ansicht und können diese mit der Maustaste bedienen.

Wenn das alles funktioniert hat, sind alle Grundeinstellungen bezüglich der Kommunikation mit der Zentrale in Ordnung und wir können uns grösseren Aufgaben zuwenden.

Das wichtigste Werkzeug zur Erstellung leistungsfähiger Anwendungen sind die Makros.

Grundlagen der Makro-Programmierung

Mit den Anweisungen in den [Makros](#) werden die Aktionen, logischen Verknüpfungen, also alle wichtigen Funktionen einer Anwendung programmiert.

Makros werden aktiviert durch Empfang einer Meldung von einem Sensor (das kann eine z.B. Fernbedienung oder ein Temperatursensor sein).

Um Aktionen also z.B. bei Betätigung einer Fernbedienungstaste auszulösen, werden entsprechende Anweisungen im Makro der Fernbedienungstaste programmiert.

In der Studio-Version können Makros zusätzlich in einem individuell einstellbarem Zeitintervall ausgeführt werden. Die Aktivierung von Makros im Zeitintervall wird normalerweise bei Aktionen verwendet, z.B. um eine Lampe nach einer bestimmten Zeit auszuschalten.

Um ein wenig in die Praxis einzusteigen sollten Sie mit kleinen Testprojekten üben.

Ein kleines Beispielprojekt Teil 1

In diesem Teil des kleinen Beispielprojekt erstellen wir ein Makro für eine Fernbedienungstaste zum Schalten

einer Lampe.

Falls Sie schon Module in der Zentrale angelernt haben, können Sie diese im Fenster Modulauswahl mit dem Knopf "Module aus Zentrale importieren" übernehmen und die vorhandenen Module benutzen, soweit diese benutzt werden können (also eine Fernbedienungstaste und einen Schaltaktor wie z.B. eine Schaltsteckdose).

Als erstes schreiben wir ein kleines Makro für eine Fernbedienungstaste, mit der eine Lampe geschaltet wird. Falls in der Zentrale noch keine Fernbedienung angelernt und übernommen worden ist, wählen Sie einfach im Fenster Modulauswahl eine Fernbedienung aus und bringen diese durch Anklicken des Knopfes mit dem roten Pfeil in das rechte Fenster. Nun wird noch eine Lampe benötigt. Wenn es noch von der Zentrale übernommenes Modul zum Schalten einer Lampe gibt (z.B. eine Schaltsteckdose) wählen Sie in der Modulauswahl eine Schaltsteckdose aus und bringen diese durch Anklicken des Knopfes mit dem roten Pfeil in das rechte Fenster. Gehen Sie in die Modulbearbeitung und nennen das Objekt "Lampe1". Wenn es schon ein geeignetes Modul gibt benennen Sie dieses um auf "Lampe1".

Wenn Sie eine Fernbedienung an der Zentrale angelernt haben beachten Sie bitte, dass die einzelnen Tasten vor der Verwendung "registriert werden müssen. Infos dazu finden Sie auf der Seite [Schaltsensoren](#) am Ende der Seite.

Wählen Sie dann die neu angelegte Fernbedienung (oder eine beliebige schon in der Zentrale vorhandene Fernbedienung) aus und klicken auf den Knopf "Bearbeiten".

Wählen Sie in dem Fenster der Fernbedienung eine Taste aus und klicken dann auf "Makro". In dem Fenster zur Makrobearbeitung erstellen Sie dann folgende Anweisung:

```
Lampe1 einschalten für 10 Sekunden
```

Diese Anweisung bewirkt, dass das Objekt "Lampe1" für 5 Minuten eingeschaltet wird. In der Standardversion werden Makros von Fernbedienungstasten grundsätzlich ausgeführt, wenn eine Meldung von der Fernbedienungstaste empfangen wird. In der Studio-Version kann die Aktivierung des Makros aufgrund eines Empfangs von der Hardware auf der Seite Makro der Objektdefinitionen eingestellt werden. Um das Makro direkt in der Visualisierung ohne Fernbedienung testen zu können, aktivieren Sie auf der Seite Allgemein des Objektfensters die Option "Ausführen bei Eingabe". Wenn diese Option aktiviert ist, wird das Makro ausgeführt, wenn das Icon mit der Maus angeklickt wird. (Das ist normalerweise standardmäßig nicht aktiviert, da ein Anklicken des Icons in der Visualisierung auch eine Sendung verursachen kann, je nach Programmierung und Konstellation könnte das dann zu unbeabsichtigten Reaktionen führen.)

Klicken Sie auf den "Blitz"-Button oder wählen im Menü *Ausführung* den Punkt *Compilieren und übertragen*. Falls es bei der Übertragung Probleme gibt überprüfen Sie bitte die Einstellungen wie unter [Einstellungen nach der Installation](#) beschrieben.

Starten Sie dann das Programm VisuWin (Ausführung->Visualisierung oder Klick auf den Auge-Knopf) und wählen die Ansicht "Alle Objekte".

Klicken Sie auf das Icon der Fernbedienungstaste oder auf die Fernbedienungstaste und prüfen ob alles funktioniert.

Nun ändern wir das Makro der Fernbedienung etwas ab, wir löschen die vorhandene Anweisung und erstellen eine Anweisung:

```
Lampe1 umschalten
```

Diese Anweisung bewirkt, dass das Objekt "Lampe1" umgeschaltet wird (d.h. wenn es ausgeschaltet war wird es eingeschaltet, war es eingeschaltet, wird es ausgeschaltet). Mit dieser Anweisung ist es möglich eine Lampe mit beliebig vielen Tastern mit jeweils nur einer Tastenbetätigung zu schalten (in der konventionellen Elektroinstallation wird diese Funktion "Wechselschaltung" genannt).

Klicken Sie auf den "Blitz"-Button oder wählen im Menü *Ausführung* den Punkt *Compilieren und übertragen*.

Ein kleines Beispielprojekt Teil 2 (nur Studio-Version)

In der Studio-Version können auch für Aktoren Makros erstellt werden. Wir erweitern nun das Beispielprojekt um eine zusätzliche Funktion, indem wir auch für die Lampe ein Makro erstellen. Die Lampe soll nach 5

Minuten ausgeschaltet werden, unabhängig davon wie sie eingeschaltet wurde-
Dazu wählen wir das Modul "Lampe1" im Fenster Modulauswahl aus und klicken auf "Bearbeiten" und im Fenster der Lampe auf "Makro".

Im Makro erstellen Sie nun folgende Anweisungen:

```
wenn Lampe1 eingeschaltet und Schaltdauer(Lampe1) > "00:05:00" dann  
    Lampe1 ausschalten  
endewenn
```

Dieses Makro soll bewirken, dass die Lampe1 immer 5 Minuten nachdem sie eingeschaltet wurde wieder ausgeschaltet wird.

Damit das auch wie gewünscht funktioniert muss das Makro im Zeitintervall aktiviert werden.

Da die Uhrzeit in der Uhrzeitabfrage volle Minuten hat, können wir hier das Zeitintervall "jede Minute" einstellen. Würde die abgefragte Uhrzeit Sekunden beinhalten (z.B. "00:05:15") müsste das Zeitintervall "alle 5 Sekunden" eingestellt werden.

Bitte beachten Sie dabei:

Uhrzeitabfragen müssen immer durch 5 Sekunden teilbar sein. Wenn ein Makro Uhrzeitabfragen beinhaltet, ist es wichtig darauf zu achten, dass das Zeitintervall entsprechend eingestellt wird. Wenn in einem Makro Sonnenaufgang oder Sonnenuntergang abgefragt werden, muss dieses mindestens jede Minute ausgeführt werden, weil Sonnenauf- und -untergang immer volle Minuten haben.

Übertragen Sie das Projekt mit dem "Blitz-Button auf die Zentrale und spielen ein wenig damit.

Erweitern Sie dieses erste kleine Projekt und spielen ein wenig mit den Möglichkeiten. Sehen Sie sich die Anweisungen und Funktionen der Makrosprache an und lassen Sie Ihrer Phantasie freien Lauf bezüglich der Realisierung Ihrer individuellen Projekte.

Nochmal eine Checkliste zu Projekterstellung:

Schritte zur Erstellung von Projekten

1. Lernen Sie die Module in der Zentrale an.
2. Übernehmen Sie die Module aus der Zentrale durch Anklicken des Knopfes "Module aus Zentrale importieren"
3. Erstellen Sie Ihre individuellen Ansichten auf der Seite Ansichten des Einstellungsfensters. Dabei können Sie auch individuelle Hintergrundbilder (z.B. Grundrisse) festlegen.
4. Auf der Seite Visualisierung der Objektdefinitionen können Sie für jedes Objekt mehrere unterschiedliche Darstellungen festlegen. Durch Auswahl einer Visualisierung und einer Ansicht und anschließendes Anklicken des Zuordnungsknopfes zwischen der Darstellungs- und Visualisierungsliste können die verschiedenen Objektdarstellungen den einzelnen Ansichten zugeordnet werden.
5. Erstellen Sie die Makros für die einzelnen Objekte.
6. Erstellen und übertragen Sie den Ausführungscode auf die Zentrale. Klicken Sie dazu auf den "Blitz"-Button oder wählen im Menü *Ausführung* den Punkt *Compilieren und übertragen* oder führen Sie die einzelnen Schritte manuell aus.
Dazu sind folgende Schritte erforderlich:
 - Erstellen Sie den Ausführungscode (Menü *Ausführung*->*Compilieren* oder auf den "Zahrad-Button" klicken)
 - Übertragen Sie den Ausführungscode auf die Zentrale. Infos dazu finden Sie in der Beschreibung zu [ExecEngineWin](#)
7. Starten Sie das Programm VisuWin und prüfen Sie ob in den Ansichten alles so dargestellt wird, wie Sie es wollten.

Hauptfenster

Das Hauptfenster des Programms hat neben der Toolbar nur eine Statuszeile, in der wichtige Systemmeldungen angezeigt werden.

Von diesem Hauptfenster aus werden alle anderen Fenster aufgerufen.

Auf der Seite [Allgemein](#) im Einstellungsfenster können Optionen zum Verhalten des Hauptfensters eingestellt werden.

Unter anderem ist es möglich, das Hauptfenster immer im Vordergrund vor allen anderen Fenstern anzeigen zu lassen.

Wenn grössere Fenster der Anwendung geöffnet werden und das Hauptfenster diese verdecken würde oder diese Fenster das Hauptfenster verdecken würden, wird das Hauptfenster automatisch am oberen oder unteren Rand dieser Fenster positioniert.

Mit dem "Blitz"-Button oder dem Menüpunkt *Compilieren und übertragen* wird der Ausführungscode für die Zentrale erstellt und zur Zentrale übertragen. Vor der ersten Übertragung auf die Zentrale muss die Verbindung zur Zentrale konfiguriert und eine Konfigurationsdatei zur Zentrale übertragen werden. Das geschieht mit dem Menüpunkt *Konfigurieren->Assistent Zentralenverbindung*. Bitte lesen Sie dazu auch das Kapitel [Einstellungen nach der Installation](#).

Visualisierungsfenster

Die Visualisierung wird vom Programm [VisuWin](#) bereitgestellt. Jede definierte Ansicht hat ein eigenes Visualisierungsfenster, in dem die Zustände und Werte der einzelnen Objekte angezeigt werden.

Mit der rechten Maustaste kann ein Pop-Up-Menü aufgerufen werden, mit dem in den Vollbildschirm-Modus geschaltet werden kann, dabei wird die aktuelle Ansicht auf dem ganzen Bildschirm ohne Menüs und Fensterrahmen angezeigt.

Dieser Modus kann mit der ESC-Taste beendet werden.

Das Fenster Modulauswahl

Im linken Teil des Fensters wird eine Liste aller verfügbaren Module angezeigt. Im Auswahlfeld links unter der Liste kann der Typ der anzuzeigenden Module ausgewählt werden.

Ein Modul, das für ein [Projekt](#) benutzt werden soll, wird im linken Listfenster markiert und mit dem Pfeil-Knopf in das rechte Listfenster *verwendete Module* kopiert, wobei der im Projekt verwendete Name des Moduls vergeben wird.

Bevor Module aus der Zentrale übernommen werden können muss die IP-Adresse der Zentrale im [Fenster HomeMatic-Zentrale](#) angegeben werden.

Mit dem Knopf **[Module importieren]** werden alle in der Zentrale angelegten Module übernommen. Module, die schon verwendet werden bleiben erhalten. Die Erkennung ob Module schon existieren wird über die Adresse der Module durchgeführt. Bei schon vorhandenen Modulen werden keine Änderungen vorgenommen. Nach der Übernahme haben alle Module als Name ihre Adresse mit vorangestelltem Unterstrich.

Ein Modul kann mit dem Knopf **[Modul löschen]** gelöscht werden, das Löschen mit der Entfernen-Taste ist aus Sicherheitsgründen nicht möglich. Bevor ein Modul tatsächlich gelöscht wird, erscheint eine Bestätigungsabfrage.

Wenn ein Modul in Makros verwendet wird, kann es nicht gelöscht werden (denn dann würde das Makro ja nicht mehr funktionieren). Es erscheint dann ein Fenster, in dem angezeigt wird in welchen Makros das Modul verwendet wird. Erst wenn alle Makroanweisungen die das Modul betreffen in den Makros entfernt wurden, kann es gelöscht werden.

Mit dem Knopf **[Bearbeiten]** oder durch Doppelklick auf ein Modul im rechten Listfenster wird das Definitionsfenster für das markierte Modul geöffnet.

Die Art des Fenster ist abhängig vom Typ des Moduls:

[Aktoren](#) zur Bearbeitung von Aktoren (Empfängern)

[Schaltsensoren](#) zur Bearbeitung von Schaltsensoren (Sendern)

[Sensoren](#) zur Bearbeitung von Sensoren, die Zahlenwerte senden.

Das Fenster Schaltsensoren

Im Fenster Schaltsensoren werden alle erforderlichen Informationen für einen Schaltsensor eingestellt.

Im rechten oberen Teil können Name und Bezeichnung für das [Objekt](#) verändert werden.

Der Name wird bei Definitionen und in Makros verwendet, die Bezeichnung wird z.B in [Visualisierungen](#) angezeigt.

Mit dem Knopf **[Visualisierung]** wird die [Seite Visualisierung der Objekdefinition](#) aufgerufen.

Für jede Taste eines Senders (also jeden Kanal) kann ein separates Makro erstellt oder aus der Liste vorhandener Makros ausgewählt werden.

Durch Klick auf den Knopf **[Makro bearbeiten]** wird das [Fenster Makro bearbeiten](#) geöffnet, in dem Sie die einzelnen Makroanweisungen erstellen und editieren können.

Durch Klick auf den Knopf **[Makro auswählen]** wird ein Fenster geöffnet, in dem eine Liste aller vorhandenen Makros angezeigt wird. Es erscheinen auch Makros, für die noch keine Anweisungen hinterlegt wurden, damit können auch Makros ausgewählt, die später erst erstellt werden sollen.

Wenn sie hier ein Makro auswählen, so wird dieses ausgeführt, wenn während der [Ausführung](#) die zugehörige Taste betätigt wird. Diese Methode ermöglicht es vorhandene Makros mehrfach zu nutzen ohne die Anweisungen mehrmals eingeben zu müssen. Tatsächlich wird in diesem Fall in das Makro der Taste nur eine Anweisung mit dem Aufruf des ausgewählten Makros erstellt. Wenn Sie mit dem Knopf **[Makro bearbeiten]** in das Makro gehen, sehen Sie diese Anweisung und können bei Bedarf weitere Anweisungen zufügen.

Mit dem Knopf **[Makro löschen]** können Sie das Makro für die aktuelle Taste löschen, wenn das Makro mehrere Anweisungen hat, erscheint vor dem Löschen eine Bestätigungsabfrage.

In diesem Fenster gbt es zwei spezielle Knöpfe, die normalerweise nicht aktiviert sind. Um diese zu aktivieren muss eine Verbindung zur Zentrale hergestellt werden. Das geschieht im [Fenster HomeMatic-Zentrale](#) mit dem Knopf **[Verbindung zur Zentrale herstellen]**.

Mit dem Knopf **[Adresse empfangen]** können die Adressen einzelner Tasten überprüft werden. Gehen Sie dazu auf die Zeile einer Taste und klicken dann auf den Knopf **[Adresse empfangen]**. Es erscheint ein Hinweisfenster. Betätigen Sie jetzt die entsprechende Taste der Fernbedienung bzw. des Schaltsensors. Wenn die empfangene Adresse identisch mit der aktuell ausgewählten Zeile der Taste ist, ertönt ein Bestätigungston und das "Warte-Fenster" verschwindet. Ist die empfangene Adresse nicht identisch erscheint eine Abfrage ob die empfangene Adresse in die aktuelle Zeile eingesetzt werden soll. Beantworten Sie die Abfrage nur mit Ja, wenn Sie sicher sind, dass die Adressen ausgetauscht werden sollen, normalerweise ist das nicht erforderlich oder sinnvoll. Es kann jedoch nötig werden, wenn eine z.B. eine Fernbedienung gegen eine andere getauscht werden soll um nicht alle Definitionen neu eingeben zu müssen.

Bitte bei Funkmodulen unbedingt beachten:

Bevor Tasten von Fernbedierungen und anderen Funk-Schaltsensoren benutzt werden können, müssen diese in der Zentrale registriert werden, damit diese Meldungen von diesen Tasten zulässt.

Das passiert mit dem Knopf

[Taste registrieren] Wenn dieser Knopf nicht aktiviert ist, schliessen Sie dieses Fenster und stellen die Verbindung zur Zentrale her. Das geschieht im [Fenster HomeMatic-Zentrale](#) mit dem Knopf **[Verbindung zur Zentrale herstellen]**.Klicken Sie dann auf den Knopf **[Taste registrieren]** und befolgen die Anweisungen am Bildschirm. Je nach Modul und Konfigurationsstatus muss vor der zu registrierenden Taste die Konfigurationstaste gedrückt werden.

Für leitungsgebundene Sensoren (HMW...-Module) ist eine Registrierung von Tasten nicht erforderlich.

Das Fenster Aktoren

Im Fenster Aktoren werden alle erforderlichen Informationen zur Steuerung eines Aktors (Empfängers) hinterlegt.

Im rechten oberen Teil können Name und Bezeichnung für das [Objekt](#) verändert werden.

Der Name wird bei Definitionen und in Makros verwendet, die Bezeichnung wird z.B in [Visualisierungen](#) angezeigt.

Mit dem Knopf **[Visualisierung]** wird die [Seite Visualisierung der Objekdefinition](#) aufgerufen.

Durch die Auswahl eines Bild-Symbols kann sich auch der „Typ“ des Objekts ändern.

Wenn Sie beispielsweise für eine Rollladen- und Markisensteuerung das Bild-Symbol "Markise" auswählen ändern sich die zur Verfügung stehenden Zustände auf "oben", "halb", "unten", "viertel" und "dreiviertel" stehen nicht mehr zur Verfügung.

In der Studio-Version können Sie jedoch durch das Anlegen eigener Typen auch einen Markisentyp mit mehr als drei Zuständen erzeugen (oder Sie ändern einfach die Bitmaps für den Typ Rollladen).

In der Tabelle im rechten Teil des Fensters können Schaltzeiten für das Objekt festgelegt werden. Dabei ist es auch möglich den Zeitpunkt des Sonnenaufgangs oder Sonnenuntergangs für das aktuelle Datum auszuwählen. Dies geschieht mit einem Pop-Up-Menü, das durch Klick auf die rechte Maustaste geöffnet wird wenn der Cursor sich im Zeiteingabefeld befindet. Es kann auch direkt **SA** für Sonnenaufgang oder **SU** für Sonnenuntergang in die Stundenstellen des Zeiteingabefensters eingetragen werden.

Bitte beachten Sie:

Alle Zeitbearbeitungen erfolgen im 5-Sekunden-Rhythmus, bei Zeitangaben müssen die Sekunden immer durch 5 teilbar sein (xx:xx:00, xx:xx:05, xx:xx:10 ... xx:xx:55).

Bitte beachten Sie folgende Hinweise:

[Besonderheiten bei Dimmern](#)

Das Fenster Sensoren

Im Fenster Sensoren werden alle Sensoren verwaltet, die Zahlenwerte wie Temperatur, Luftfeuchtigkeit, Helligkeit usw. schicken.

Im rechten oberen Teil können Name und Bezeichnung für das [Objekt](#) verändert werden.

Der Name wird bei Definitionen und in Makros verwendet, die Bezeichnung wird z.B in [Visualisierungen](#) am Bildschirm angezeigt.

Mit dem Knopf **[Visualisierung]** wird die [Seite Visualisierung der Objekdefinition](#) aufgerufen.

Im rechten Teil des Fensters können Makros für verschiedene Werte-Bereiche des Sensors definiert werden. Geben Sie in den beiden Eingabefeldern oberhalb des Knopfes [Neuer Bereich] den gewünschten Wertebereich an und klicken dann auf den Knopf **[Neuer Bereich]**. Das Fenster Makro bearbeiten wird geöffnet und Sie können die Anweisungen eingeben, die ausgeführt werden sollen, wenn der Sensor einen Wert sendet, der innerhalb des Bereichs liegt. Die Bereiche können sich beliebig überschneiden, es werden dann alle Teil-Makros ausgeführt. Das bietet eine Vielzahl von Möglichkeiten, **beachten Sie aber dabei:**

Wenn beispielsweise der Bereich von 10,0 bis 20,0 für ein Teilmakro und der Bereich von 20,0 bis 30,0 für ein anderes Teilmakro angegeben wird werden beide Teilmakros ausgeführt (weil der Wert 20,0 für beide gültig ist). Soll nur ein Makro ausgeführt werden, muss in diesem fall z.B. der zweite Bereich bei 20,1 beginnen.

Mit dem Knopf **[Bereich bearbeiten]** können vorhandene Teilmakros zu den verschiedenen Bereichen bearbeitet werden. Wählen Sie in der Tabelle die Zeile aus und klicken Sie auf den Knopf **[Bereich bearbeiten]**. Einfacher ist ein Doppelklick auf die Zeile, auch dann wird das Fenster zur Bearbeitung des Teilmakros geöffnet.

Mit dem Knopf **[Bereich löschen]** wird die ausgewählte Bereichszeile inklusive der Makroanweisungen gelöscht. Die ausgewählte Zeile ist die farblich hinterlegte Zeile.

Das Fenster Einstellungen

In diesem Fenster können auf mehreren Seiten Einstellungen vorgenommen werden.

Bitte beachten Sie:

Je nach Seite und Art der Daten werden die neuen Werte erst nach dem Schliessen des Fensters übernommen. Schliessen Sie also dieses Fenster, nachdem Sie Änderungen vorgenommen haben um diese zu übernehmen.

Die Seite Objekte

Auf dieser Seite wird eine Übersicht über alle [Objekte](#) angezeigt.

Durch Anklicken der Schaltfläche **[Modulwahl]** wird das Fenster zur Modulwahl aufgerufen.

Durch Betätigung des Knopfes **[Bezüge anzeigen]** wird angezeigt, in welchen Makros das ausgewählte Objekt verwendet wird. Diese Funktion ist in komplexen Projekten wichtig um Zuordnungen schnell erkennen zu können.

Mit den weiteren Schaltflächen werden die Objekte verwaltet. Das Löschen ist nur für Objekte möglich, die nicht mit einem Hardwaremodul verbunden sind.

Mit dem Button **[Typdefinitionen]** wird ein Fenster geöffnet, mit dem selbst definierte Typen und Zustände erstellt und verwaltet werden können.

Mit dem Button [\[Spezielle Objekte\]](#) wird ein Fenster geöffnet, in dem Objekte für die Bearbeitung von Batteriewarnungen und Kommunikationsstörungen festgelegt werden können.

Die Seite Ansichten

Auf der Seite werden unterschiedliche Ansichten zur Visualisierung eingerichtet.

Für jede Ansicht kann optional ein [Bitmap](#) als Hintergrundbild festgelegt werden. Wenn für eine Ansicht ein Hintergrundbild festgelegt wurde, werden die [Objekte](#) als Bild-Symbol dargestellt – um welches Objekt es sich handelt, geht dann aus der Positionierung im Hintergrundbild hervor.

Wenn kein Hintergrundbild ausgewählt wird, werden die Objekte je nach Typ und Definition als [Objektrahmen](#) oder [Textrahmen](#) dargestellt.

Für jede Ansicht kann mit dem Knopf **[Hintergrundfarbe]** eine Hintergrundfarbe ausgewählt werden. Diese Farbe füllt das ganze Fenster aus, mit Ausnahme des Bereichs, in dem das Hintergrundbild steht.

Bei der Standardversion ist die Anzahl der möglichen individuellen Ansichten auf drei beschränkt, bei der Starter-Version kann nur eine individuelle Ansicht erstellt werden.

Die Seite Makros

Auf dieser Seite können Makros definiert werden. Diese können zu bestimmten Ausführungszeiten gestartet werden, aber auch durch andere Makros aufgerufen werden.

Mit dem Button **[Ausführungszeiten]** wird ein Fenster geöffnet, in dem Makros zu bestimmten Zeiten aktiviert werden können.

Bitte beachten Sie:

Alle Zeitbearbeitungen erfolgen im 5-Sekunden-Rhythmus, bei Zeitangaben müssen die Sekunden immer durch 5 teilbar sein (xx:xx:00, xx:xx:05, xx:xx:10 ... xx:xx:55).

Die Seite Allgemein

Auf dieser Seite können einige allgemeine Einstellungen vorgenommen werden.

Bei Auswahl der Option *Letztes Programm automatisch starten*, wird das letzte verwendete Programm bei einem Programmstart automatisch geladen und ausgeführt.

Bei Auswahl der Option *Hinweise am Mauszeiger anzeigen* werden zu vielen Bildelementen erläuternde Hinweise angezeigt, wenn der Mauszeiger auf diesen Elementen steht.

Im Auswahlfenster *Postleitzahl zur Berechnung Sonnenzeiten* wird die eigene Postleitzahl angegeben. Aufgrund dieser Postleitzahl werden jeweils die tagesaktuellen Zeiten für [Sonnenaufgang und Sonnenuntergang](#) ermittelt.

Mit dem Button **[weitere Einstellungen]** wird das Fenster [Einstellungen zu Sonnenzeiten](#) geöffnet, in dem detaillierte Angaben zum geografischen Standort gemacht werden können. Dort können auch Grenzwerte und Veränderungen der Sonnezeiten festgelegt werden.

Mit der Schaltfläche **[Anwesenheitssimulation]** wird das Fenster zur Verwaltung der Daten für die [Anwesenheitssimulation](#) geöffnet.

nur Studio-Version:

Mit der Schaltfläche **[e-mail-Einstellungen]** wird das [Fenster der e-mail-Konfiguration](#) geöffnet.

Einstellungen zu Sonnenzeiten

In diesem Fenster können individuelle Anpassungen bezüglich der Zeiten zum [Sonnenaufgang und Sonnenuntergang](#) gemacht werden.

Da je nach Jahreszeit die Zeiten für Sonnenaufgang und Sonnenuntergang sehr früh bzw sehr spät sein können, kann es sinnvoll sein diese Werte zu begrenzen. Es wird dann bei Benutzung dieser Funktionen der angegebene Grenzwert nicht unter- bzw. überschritten.

Weiterhin können individuelle Anpassungen bezüglich der Zeitpunkte festgelegt werden. Dabei werden die Minuten angegeben, um die der berechnete Zeitpunkt verändert wird.

e-mail Konfiguration

In diesem Fenster werden die Einstellungen zum e-mail-Versand festgelegt..

Im Eingabefeld "SMTP-Server" wird der Name des Posteingangsserver Ihres Providers eingetragen. Es muss sich dabei um einen SMTP-Server handeln.

Wenn Sie den Namen des SMTP-Servers Ihres Providers nicht kennen, können Sie diesen beim Provider erfragen, einige gebräuchliche Provider und zugehörige SMTP-Server sind am Ende dieser Seite aufgeführt.

Im Eingabefeld "Port" wird der benutzte Port des SMTP-Servers eingetragen, das ist normalerweise immer der Port 25, nur in speziellen aussergewöhnlichen Fällen wird ein anderer Port als 25 benutzt.

Im Eingabefeld "e-mail-Adresse" wird die Adresse eingetragen, die als Absender der e-mail benutzt werden soll. Bei manchen Providern muss das zwingend exakt die Adresse sein, unter der man als Benutzer eingetragen ist - andernfalls wird die mail nicht versendet.

Im Eingabefeld "Benutzer" wird der Benutzername eingetragen, im Feld "Kennwort" das Kennwort. Benutzername und Kennwort werden vom Provider vergeben.

In seltenen Fällen, kann es vorkommen, dass Benutzername und Kennwort nicht angegeben werden müssen oder sogar nicht benutzt werden können. In diesen Fällen muss das Optionsfeld "Authentifizierung" deaktiviert werden. Benutzer und Kennwort haben Sie normalerweise schon bei der Einrichtung Ihres e-mail-Programms auf dem PC angegeben.

Sollte der e-mail-Versand bei Ihrem Provider aus irgendeinem Grund nicht korrekt funktionieren, so können Sie sich einfach im Internet z.B. bei "www.web.de" oder "www.gmx.net" ein kostenloses e-mail-Konto einrichten und dieses benutzen.

Hier die SMTP-Server Provider:

```
t-online : smtpmail.t-online.de
lundl    : smtp.lundl.de
web.de   : smtp.web.de
gmx.net  : mail.gmx.net
```

Die Seite Verzeichnisse

Auf dieser Seite werden die benutzten Standardverzeichnisse für [Bitmaps](#) und die eigenen Projektdateien eingestellt.

Diese Verzeichnisse werden vom Programm automatisch benutzt, es können natürlich im Einzelfall auch Dateien aus anderen Verzeichnissen ausgewählt werden.

Das Fenster Hardware

In diesem Fenster werden Einstellungen zur Verbindung mit der Zentrale verwaltet.

Auf der Seite *Zentrale* des Formuläreiters werden die IP-Adresse der Zentrale und der Port der Exec-Engine auf der Zentrale eingegeben. Als Port sollte 2110 angegeben werden. Dieser Port wird auch in den Verbindungsdaten des Programms [VisuWin](#) angegeben und dient der Verbindung mit der *ExecEngine* auf der Zentrale.

Auf der Seite *eigene IP* werden im Listfenster die auf dem PC zur Verfügung stehenden IP-Adressen angezeigt. Hier muss die IP-Adresse durch Ankreuzen ausgewählt werden, die zur Herstellung der Verbindung mit der Zentrale benutzt werden soll. Als Port wird hier 3101 vorgeschlagen, es können aber auch andere Ports benutzt werden, falls dieser schon anderweitig belegt ist.

Auf der Seite *Zentrale* kann mit dem Knopf **[Verbinden mit Zentrale]** eine Verbindung zur Zentrale hergestellt werden, nachdem alle Verbindungsdaten korrekt eingegeben wurden. Eine Verbindung zur Zentrale ist erforderlich um Tasten von Schaltsensoren (z.B. Fernbedienungen) zu registrieren. Nähere Infos dazu finden Sie in der Hilfe zu den [Schaltsensoren](#).

Mit dem Knopf **[Kontrollprogramm für Zentrale]** wird das Programm ExecEngineWin gestartet, das zur Steuerung und zum Dateiaustausch mit der Exec-Engine auf der Zentrale dient.

Objektdefinition

Das Fenster zur Definition der Objekte wird auf der [Seite Objekte](#) des Einstellungsfensters durch Doppelklick auf ein Objekt oder den Button **[Bearbeiten]** gestartet.

[Die Seite Allgemein](#)

[Die Seite Makro](#)

[Die Seite Zeit-Tabelle](#)

[Die Seite Visualisierung](#)

Die Seite Allgemein der Objektdefinition

Auf dieser Seite wird der Objekttyp und die Eigenschaften des Objekts definiert.

Auswahlbox Objekttyp

In der Auswahlbox Objekttyp oben links im Fenster kann der Typ des Objekts ausgewählt werden. Wenn es sich um ein Objekt zu einem Hardwaremodul handelt wurde der für dieses Modul korrekte Standard-Typ bei der Erstellung des Moduls eingetragen. Dieser kann verändert werden, z.B. von Rolllade zu Markise oder es kann einem Dimmer oder Rolllalldenaktor der Typ Zahl zugewiesen werden um die prozentuale Einstellung des Moduls zu steuern. Da je nach Programmversion auch benutzerdefinierte Objekttypen erstellt werden können kann das Programm nicht prüfen ob ein Objekttyp kompatibel zum zugeordneten Hardwaremodul ist. Bei einer Veränderung des Typs eines Objekts zu einem Hardwaremodul liegt es in der Verantwortung des Benutzers darauf zu achten, dass der ausgewählte Typ kompatibel zum Hardwaremodul ist. Eine Veränderung des Objekttyps verändert nicht den Typ der zugeordneten Hardware!

Optionsfelder

Deaktivieren beim Start

Das Objekt ist beim Start des Steuerungsprogramms [deaktiviert](#). Es kann mit der [Programmanweisung AKTIVIEREN](#) aktiviert werden.

Unsichtbar beim Start

Diese Auswahl bewirkt, daß das Objekt bei der Ansicht nach dem Programmstart [unsichtbar](#) ist. Während der Programmausführung kann das Objekt [sichtbar](#) gemacht werden.

Ausführen bei Eingabe

Diese Option ist standardmäßig ausgewählt und bewirkt, dass das Objektprogramm ausgeführt wird wenn dieses Objekt mit der Maus angeklickt wird.

Ton beim Umschalten

Diese Auswahl bewirkt einen Bestätigungston bei einer Änderung des Objektzustands.

nicht benutzen

Wenn diese Option aktiviert ist, steht dieses Objekt für die Programmierung nicht zur Verfügung und wird in Auswahllisten für Objekte nicht angezeigt und darf auch anderweitig nicht benutzt werden.

Bitte beachten Sie:

Wird dieses Objekt z.B. in einem manuell erstelltem Makro verwendet, so führt das zu einem Fehler bei der Erstellung des Ausführungscodes.

Diese Option sollte nur ausnahmsweise mit Vorsicht benutzt werden

Bezeichnung

Hier kann eine Bezeichnung für das Objekt angegeben werden. Diese Bezeichnung wird im [Ausführungsfenster](#) verwendet.

Startwert

Der hier angegebene Wert wird beim Start des Programms für das Objekt eingestellt.

Weitere Eingabefelder in diesem Fenster können sein:

Resetzeit in Sekunden (nur bei bestimmten Sensoren)

Bei Tasten von Fernbedienungen, bei Bewegungsmeldern und anderen Binärsensoren erscheint ein zusätzliches Eingabefeld *Resetzeit in Sekunden*. In dieses Feld kann die Zeit in Sekunden angegeben werden, nach der z.B. eine gedrückte Taste wieder in den Normalzustand (nicht gedrückt) versetzt wird oder ein Bewegungsmelder wieder in seinen Normalzustand gesetzt wird. Dieses automatische Rücksetzen dient dazu den Normalzustand eines Objektes wieder herzustellen, das Makro des Objektes wird dadurch nicht gestartet.

Visualisierungs-Aktion (nur bei Objekten vom Typ Knopf)

Bei Objekten vom Typ Knopf erscheint ein zusätzliches Eingabefeld *Visualisierungs-Aktion*. In diesem Feld kann eingetragen werden was bei Betätigung des Knopfes im Visualisierungsprogramm passieren soll. Es gibt folgende Eingabemöglichkeiten in diesem Feld:

Vollbild

Das Wort Vollbild bewirkt, dass die aktuelle Ansicht als Vollbild angezeigt wird und den ganzen Bildschirm ausfüllt. Mit der ESC-Taste kann zurück in den Fenstermodus geschaltet werden.

Fenster

Das Wort Fenster bewirkt, dass die aktuelle Ansicht als Fenster dargestellt wird. Diese Funktion dient dazu aus den Vollbild-Modus zu beenden ohne die Taste ESC benutzen zu müssen (z.B. für Touch-Screen-Monitore).

Name einer Ansicht

Wenn Name einer Ansicht im Feld Visualisierungs-Aktion eingetragen wird, wird das Fenster für diese Ansicht bei Betätigung des Knopfes angezeigt.

Diese Visualisierungs-Aktionen werden ausschliesslich im Visualisierungsprogramm [VisuWin](#) durchgeführt und haben keinen weiteren Einfluss auf die Programmausführung.

Die Seite Makro der Objektdefinition

Diese Seite steht nur in der Studio-Version zur Verfügung.

Für jedes Objekt kann ein eigenes Makro erstellt werden. In diesem Makro wird mit den [Makroanweisung](#) festgelegt, welche Aktionen und Funktionen mit diesem Makro aktiviert bzw. ausgeführt werden sollen. Makros können in bestimmten Zeitintervallen oder durch eine Eingabemeldung des zugeordneten [Sensors](#) aktiviert werden. Das Zeitintervall wird in der Auswahlbox über dem Fenster mit dem Programmtext eingestellt.

In den meisten Fällen ist das Zeitintervall "jede Minute" oder "alle 5 Sekunden" die richtige Wahl. Ausführungsintervalle kleiner 5 Sekunden oder gar permanent sollte nur gewählt werden wenn diese kurzen Intervalle tatsächlich erforderlich sind. Wenn zu viele Makros unnötigerweise in sehr kurzen Intervallen ausgeführt werden, kann das dazu führen, dass die Anwendung insgesamt langsamer wird und einzelne Aktionen grössere Reaktionszeiten haben, da Makros die unnötigerweise zu oft ausgeführt werden die Gesamtperformance behindern.

Wenn in dem Objektprogramm Zeiten abgefragt werden, bei denen Sekunden angegeben sind, muss das Zeitintervall auf *alle 5 Sekunden* eingestellt werden, da die Funktion "Uhrzeit" die Zeit im 5-Sekunden-Raster zurückgibt.

Für Sensoren gibt es den Auswahlpunkt **Ausführung bei Empfang**.

Wenn dieser aktiviert ist, wird das Objektprogramm ausgeführt, sobald eine Meldung vom zugeordneten Sensor empfangen wird. Die Ausführung erfolgt auch wenn sich der Sensorwert nicht geändert hat.

Für Aktoren gibt es den Auswahlpunkt **Ausführung bei Änderung**.

Achtung : Die unsachgemässe Verwendung dieser Option kann schwerwiegende Fehler zur Folge haben.

Wenn diese Option aktiviert ist, wird das Makro immer dann ausgeführt, wenn sich der Wert des Objekts ändert - egal an welcher Stelle diese Änderung erfolgt (also auch innerhalb des eigenen Objektmakros).

Diese Option bietet viele Möglichkeiten, die Benutzung ist jedoch auch sehr gefährlich - es können Endlosschleifen entstehen, die das ganze System blockieren.

Diese Option sollte nur mit grosser Vorsicht benutzt werden, und auch nur wenn man sich wirklich über die Folgen im Klaren ist.

Die meisten schweren Fehler, bei denen im ersten Moment scheinbar unerklärliche Dinge passieren und es auch sein kann, dass die ganze Steuerung nicht mehr funktioniert, ist auf die unsachgemässe Verwendung dieser Funktion zurückzuführen.

Makros können die Zustände des eigenen und aller anderen [Objekte](#) abfragen und ändern. Zustände von Objekten und den diesen zugeordneten Hardwaremodulen werden automatisch abgeglichen, d.h. wenn ein zugeordneter Sensor eine Zustandsänderung meldet, ändert sich der Zustand des Objekts. Wenn das Programm einen Objektzustand ändert wird eine Meldung an den zugeordneten [Aktor](#) (Empfänger) geschickt.

Mit dem Button **[Menü-Makro bearbeiten]** kann das Fenster zur Menügeführten Erstellung von Makro-Anweisungen aufgerufen werden. Dieses Fenster ist nur aus Kompatibilitätsgründen zur Standardversion des Programms vorhanden und verfügt nur über eine eingeschränkte Auswahl an Makroanweisungen.

Mit der Schaltfläche **[Objekte/Var.]** kann ein Element einer Liste der Objekte und Variablen ausgewählt werden, das an der Cursorposition in das Makro eingesetzt wird.

Mit der Schaltfläche **[Anweisungen]** kann eine Anweisung aus der Liste der gebräuchlichsten Anweisungen ausgewählt werden. Diese wird direkt der Cursorposition in das Makro eingesetzt. Die Anweisungen sind nicht alphabetisch sondern nach Anwendungshäufigkeit und Gruppen sortiert.

Mit der Schaltfläche **[Editor]** wird ein separater Editor aufgerufen, der eine bessere Übersichtlichkeit und Optionen zum Suchen/Ersetzen bietet.

Dieser Editor kann auch durch Doppelklick auf das Makrofenster gestartet werden.

Im unteren Teil der Seite können [Variablen](#) für das Objektprogramm definiert werden.

Variablen sind dem jeweiligen Objekt zugeordnet, können jedoch auch in anderen Objektprogrammen verwendet werden, wenn der Name der Variablen qualifiziert, d.h. mit vorangestelltem zugehörigem Objektnamen, von der Variablen durch einen Punkt getrennt, angegeben wird.

Wenn Beispielsweise im Programm **Gartenlicht** die Variable **Autoschalter** definiert ist, so kann diese in anderen Programmen unter dem Namen **Gartenlicht.Autoschalter** verwendet werden.

Die Seite Zeit-Tabelle der Objektdefinition

Auf dieser Seite kann festgelegt werden wann ein Aktor-Objekt geschaltet wird bzw. wann ein Makro gestartet werden soll.

Bei Sensor-Objekten ist diese Seite nicht sichtbar.

Die Einschaltdauer steht nur bestimmten Objekten in Abhängigkeit des Modultyps zur Verfügung.

Im Normalfall ist diese Tabelle immer während der Ausführung aktiviert.

Nur Studio-Version:

In der Studio-Version ist es möglich die Aktivierung der Tabelle über Makros zu steuern, indem die Variable des Typs Schalter mit dem Namen *Zeittabelle* entsprechend geschaltet wird. Diese Variable wird automatisch angelegt wenn Einträge in der Zeittabelle vorgenommen werden. Die in der Tabelle festgelegten Zeit-Schaltungen werden nur vorgenommen, wenn dieser (Variablen-) Schalter eingeschaltet ist.

Die Aktivierung der Tabelle kann also z.B. mit den Makroanweisungen

```
Objektname.Zeittabelle einschalten und
```

```
Objektname.Zeittabelle ausschalten
```

gesteuert werden.

Die Seite Visualisierung

Auf dieser Seite wird definiert, wie ein Objekt in der Visualisierung im Programm [VisuWin](#) dargestellt werden soll.

Dazu können im Listenfeld *Darstellung* eine oder mehrere unterschiedliche Darstellungsformen für das Objekt angelegt werden.

Im Listenfeld Ansichten werden die Ansichten angezeigt, in denen das Objekt dargestellt wird. Es können zusätzliche Ansichten ausgewählt werden, in diesen wird das Objekt dann in der linken oberen Ecke positioniert und kann später im [Entwurfsfenster](#) für Ansichten an die gewünschte Stelle positioniert werden.

Mit dem Knopf zwischen den beiden Listenfeldern wird die aktuell ausgewählte Darstellungsform der aktuell ausgewählten Ansicht zugeordnet. So kann das Objekt in jeder Ansicht unterschiedlich dargestellt werden, es kann allerdings auch eine Darstellungsform für alle Ansichten benutzt werden.

Die Darstellungsform *Standard* wird automatisch erstellt, wenn ein Objekt in einer Ansicht positioniert wird und noch keine Darstellungsform definiert wurde.

Je nach Objekt und ausgewählter Darstellungsart erscheint der Knopf [Rahmendarstellung], mit dem vielfältige Einstellungen wie Farbe, Ränder, Schriftarten und -größen zur Darstellung in einem Rahmen vorgenommen werden können.

Wenn das Optionsfeld keine Eingabe aktiviert wird, kann der Wert des Objekts bei der Visualisierung in der aktuell ausgewählten Darstellung nicht verändert werden.

Das Fenster Anwesenheitssimulation

Das Fenster Anwesenheitssimulation wird aufgerufen aus der [Seite Allgemein](#) des Einstellungsfensters.

In diesem Fenster kann angegeben werden, innerhalb welcher Zeiten Empfänger geschaltet werden sollen. Der Empfänger wird dann an den ausgewählten Tagen zu einem zufälligen Zeitpunkt innerhalb des angegebenen Zeitraums geschaltet. Wenn keine obere Grenze des Zeitrahmens angegeben wird (Spalte *und*) wird innerhalb von 1-5 Sekunden nach dem in der Spalte *zwischen* festgelegten Zeitpunkt geschaltet.

Wenn keine Dauer angegeben wird bleibt der Schaltzustand erhalten.

Die Anwesenheitssimulation dient dazu, bei Abwesenheit der Bewohner Geräte so zu schalten, dass ein Beobachter den Eindruck bekommt das Haus oder die Wohnung sei nicht verlassen.

Im Normalfall ist es jedoch nicht gewünscht, dass die Anwesenheitssimulation dauernd aktiv ist. Sobald Zeiten in der Anwesenheitssimulationstabelle eingetragen sind, wird bei der Ausführung ein spezielles [Objekt](#) generiert, das Objekt **Anwesenheitssimulation**. Dieses Objekt ist ein Schalter-Objekt, das am Bildschirm oder im Menü ein- und ausgeschaltet werden kann.

Wenn es ein Objekt mit dem Namen *Anwesenheitssimulation* (z.B. als Taster) schon gibt wird, so wird dieses benutzt um die Anwesenheitssimulation ein- bzw. auszuschalten. Es wird dann kein neues Objekt generiert.

Die Anwesenheitssimulation kann nur über einen tatsächlich vorhandenen oder „virtuellen“ Schalter mit dem Namen „Anwesenheitssimulation“ ein- und ausgeschaltet werden.

Bitte beachten Sie:

Wenn Sie ein eigenes Objekt Anwesenheitssimulation erstellen, müssen Sie darauf achten, dass dieses ein Objekt ist, dass ein- und ausgeschaltet werden kann.

Das Entwurfswfenster

In diesem Fenster können individuelle eigene Ansichten erstellt werden.

Das Fenster wird aufgerufen durch Anklicken des Buttons **[Ansichten bearbeiten]** auf der Seite *Ansichten* des Einstellungsfensters.

Wenn ein Hintergrundbild ausgewählt wurde, werden die [Objekte](#), für die nicht [Objektrahmen oder Textrahmen](#) als Darstellungsart gewählt wurde als Bild-Symbol dargestellt.

Wenn kein Hintergrundbild gewählt wurde, werden alle Objekte mit einem Rechteck dargestellt.

Die Grösse dieser Rechtecke kann verändert werden, indem man mit der Maus auf den Rand des Rechtecks geht. Es erscheint ein Doppelpfeil für die Richtung und man kann die Grösse des Objektes mit gedrückter Maustaste verändern.

Die Mindestgrösse des Objekts richtet sich nach seiner Textlänge im Visualisierungsfenster (nicht in diesem Entwurfswfenster!). Es kann nicht so klein gemacht werden, dass die aktuelle Bezeichnung des Objekts abgeschnitten würde. (Ausser der Bezeichnungstext würde nach der Positionierung in diesem Fenster verändert).

Normalerweise entspricht eine Ansicht einem Geschoss des Gebäudes.

Zur Darstellung einer Ansicht kann auf der [Seite Ansichten](#) ein [Bitmap](#) ausgewählt werden. Vor dem Hintergrund dieses Bitmaps werden die dieser Ansicht zugeordneten Objekte dargestellt. Jedes Objekt kann in bis zu drei verschiedenen Ansichten mit unterschiedlicher Grösse und Position dargestellt werden.

Um ein Objekt auszuwählen, wählen Sie im Menü Objekte den Menüpunkt *Objektliste* aus. Es erscheint ein Fenster mit allen zur Verfügung stehenden Objekten. Wählen Sie das gewünschte Objekt aus diesem Fenster aus und ziehen es mit gedrückter linker Maustaste auf die gewünschte Position innerhalb des Entwurfswfensters und lassen die Maustaste los (Drag&Drop). Das Objekt lässt sich nur an eine freie Stelle positionieren, ob das Objekt an eine Stelle gesetzt werden kann, sehen Sie an der Form des Mauszeigers.

Eine weitere Möglichkeit ein Objekt auszuwählen ist ein Doppelklick auf das Objekt oder das Anklicken des Knopfes **[Übertragen]**. Auf diese Weise ausgewählte Objekte werden in der linken oberen Ecke des Entwurfswfensters platziert – von dort müssen Sie diese an die gewünschte Position bringen.

Um die Position eines Objektes zu verändern, gehen Sie mit der Maus auf das Objekt (als Mauszeiger erscheint eine Hand), drücken die linke Maustaste und ziehen das Objekt an die gewünschte Stelle.

Wenn der Mauszeiger sich nicht über einem Objekt befindet und die rechte Maustaste betätigt wird, wird ein Popup-Menü aufgerufen, mit dem markierte Objekte behandelt werden können. Objekte werden markiert durch Drücken der STRG-Taste und gleichzeitigem Drücken der linken Maustaste, wenn der Mauszeiger sich in einem Objekt befindet, oder wenn die linke obere Ecke des Objekts im Markierungsrechteck liegt. Das Markierungsrechteck wird erzeugt, indem der Mauszeiger bei gedrückter linker Maustaste von links oben nach rechts unten gezogen wird.

Nach der Markierung der Objekte kann mit dem Popup-Menü die gewünschte Aktion für die markierten Objekte ausgewählt werden. Die Markierung bleibt erhalten, bis sie mit dem Menüpunkt **Markierung entfernen** zurückgesetzt wird. Dieser Menüpunkt setzt auch den aktuellen Positionierungs-Modus zurück.

Um Objekte gruppenweise auszurichten oder zu positionieren, markieren Sie diese indem Sie die STRG-Taste drücken und mit er Maus anklicken, oder die linke Maustaste ausserhalb eines Objektes drücken und mit gedrückter Maustaste ein Rechteck ziehen, das die linke obere Ecke der zu markierenden Objekte einschliesst.

Alles markieren markiert alle Objekte im Entwurfswfenster.

Ausrichtung vertikal bedeutet, dass die markierten Objekte bei Betätigung der linken Maustaste auf die aktuelle vertikale Position (x-Position) des Mauszeigers gesetzt werden.

Ausrichtung horizontal bedeutet, dass die markierten Objekte bei Betätigung der linken Maustaste auf die

aktuelle horizontale Position (y-Position) des Mauszeigers gesetzt werden.

Gleiche Breite bedeutet, dass alle markierten Objekte auf die Breite des breitesten markierten Objekts gebracht werden.

Gleiche Höhe bedeutet, dass alle markierten Objekte auf die Höhe des höchsten markierten Objekts gebracht werden.

Verschieben verschiebt alle markierten Objekte synchron zum Mauszeiger, wenn dieser sich nicht in einem Objekt befindet und der linke Mausknopf gedrückt wird.

Die jeweilig aktivierte Aktion wird durch den Mauszeiger dargestellt, wenn dieser sich nicht auf einem Objekt befindet.

Diese Möglichkeiten der Positions- und Grössenmanipulation sind hilfreich bei der Positionierung vieler [Objektrahmen](#) oder [Textrahmen](#) in einer Ansicht, wenn kein Hintergrundbild angegeben wurde.

Das Fenster Makro-Anweisungen über Menü erstellen

In diesem Fenster werden die Anweisungen für Makros menügeführt erstellt.

Dieses Fenster ist in der Standard-Version und in der Studio-Version vorhanden.

In diesem Fenster können die erweiterten Anweisungen und Funktionen der Studio-Version nicht erstellt oder bearbeitet werden.

Wenn Sie einen Aktor schalten wollen, wählen Sie diesen aus dem Auswahlfenster *Objekt schalten/steuern* aus. Je nach Typ öffnet sich ein Fenster in dem Sie den gewünschten Zustand für das Objekt angeben können.

Wenn Schaltmodule eingeschaltet werden, kann die Einschaltdauer angegeben werden.

Mit dem Auswahlfenster *Anweisungen* können Anweisungen erstellt werden, die nicht das direkte Schalten von Empfängern betreffen.

Mit dem Knopf [**wenn-Bedingung**] wird ein Fenster geöffnet, mit dem Sie einen wenn-Block erstellen können. Anweisungen in einem wenn-Block werden nur ausgeführt wenn die Bedingung in der wenn-Zeile wahr ist.

Diese Anweisung ermöglicht komplexe logische Verknüpfungen aller [Objekte](#).

Bitte beachten Sie folgende Hinweise:

[Besonderheiten bei Dimmern](#)

Bedenken Sie bei allen Zeitangaben:

Zeitbearbeitungen erfolgen im 5-Sekunden-Rhythmus, bei Zeitangaben müssen die Sekunden immer durch 5 teilbar sein (xx:xx:00, xx:xx:05, xx:xx:10 ... xx:xx:55).

Fenster Makro-Anweisungen erstellen

warten

Mit dieser Anweisung kann das Makro an der Stelle der Anweisung für die angegebene Zeit angehalten werden.

Audio-Datei abspielen

Mit dieser Anweisung kann eine beliebige Wave-Datei abgespielt werden. Die Datei kann in einem Datei-Auswahlfenster ausgewählt werden.

Makro aufrufen

Mit dieser Anweisung wird innerhalb der Ausführung des aktuellen Makros ein anderes Makro ausgeführt.

Makro starten

Mit dieser Anweisung wird innerhalb der Ausführung des aktuellen Makros ein anderes Makro gestartet. Die Ausführung beginnt erst **nach** der Beendigung des aktuellen Makros.

Windows-Programm starten

Mit dieser Anweisung können Sie aus einem Makro heraus ein beliebiges Windows-Programm starten. Das Programm kann mit einem Dateiauswahlfenster ausgewählt werden.

Anweisung warten

Mit dieser Anweisung kann das Makro an der Stelle der Anweisung für die angegebene Zeit angehalten werden.

Diese Anweisung hat keine Auswirkungen auf andere Makros oder Funktionen.

Wenn das Makro innerhalb der Wartezeit erneut aufgerufen wird (nach Zeit, durch Tasterbetätigung oder durch ein anderes Makro) wird die alte Ausführung (und damit auch die Warte-Anweisung) abgebrochen, die Ausführung des Makros wird mit der ersten Zeile des Makros neu gestartet.

Anweisung Audio-Datei abspielen

Mit dieser Anweisung kann eine beliebige Wave-Datei im Visualisierungsprogramm [VisuWin](#) abgespielt werden. Die Datei kann in einem Datei-Auswahlfenster ausgewählt werden.

Diese Option dient dazu aktuelle Informationen und gegebenenfalls akustische Warnungen auszugeben.

Wenn eine solche Anweisung ausgeführt wird, so beginnt die Ausgabe sofort, eine eventuell noch aktive Ausgabe einer anderen Wave-Datei wird abgebrochen.

Damit ist sichergestellt, dass aktuelle Meldungen und eventuell wichtige Warnhinweise ohne Verzögerungen direkt ausgegeben werden!

Bitte beachten Sie:

Wenn Sie diesen Befehl mehrmals direkt hintereinander verwenden, werden Sie nur die letzte Datei hören, da die vorigen durch die jeweils nächste Anweisung unterbrochen werden.

Wenn Sie mehrere Dateien hintereinander ausgeben wollen, müssen Sie zwischen den Dateien [warte-Anweisungen](#) mit entsprechender Wartezeit einfügen.

Anweisung Makro aufrufen

Diese Anweisung führt ein Makro direkt aus, achten Sie unbedingt darauf, dass dadurch mehrmaligen verschachtelten Aufruf von Makros ein Makro nicht nochmals aufgerufen wird, dadurch würde eine Endlosschleife entstehen, die alles blockiert und letztlich zum Abbruch des aktuellen Makros aufgrund einer Zeitüberschreitung führt.

Das aufrufende Makro setzt seine Ausführung in der Zeile hinter dem Aufruf fort, wenn das aufgerufene Makro beendet wird.

Bitte beachten Sie dabei:

Wenn in dem aufgerufenen Makro eine warte-Anweisung steht, wird in dem Moment in das aufrufende Makro zurückgesprungen, in dem die warte-Anweisung beginnt, die Verarbeitung des aufrufenden Makros wird fortgesetzt. Das aufgerufene Makro bleibt auch aktiv, dort wird die Ausführung nach Ablauf der Wartezeit wieder fortgesetzt.

Anweisung Makro starten

Mit dieser Anweisung wird innerhalb der Ausführung des aktuellen Makros ein anderes Makro zur Ausführung vorbereitet, jedoch nicht direkt ausgeführt. Die Ausführung beginnt erst nach Beendigung des aktuellen Makros.

Die Anweisung bewirkt, dass das zu startende Makro in eine Ausführungswarteschlange gestellt wird. Es können mehrere Makros gleichzeitig in dieser Warteschlange stehen, es wird immer das Makro als nächstes ausgeführt, das am längsten in der Warteschlange steht (FIFO-Prinzip).

Diese Anweisung hat einige Vorteile gegenüber der Anweisung *Makro aufrufen* und sollte - wenn der gleiche Zweck erfüllt wird – anstelle der Anweisung *Makro aufrufen* benutzt werden.

Anweisung Windows-Programm starten

Mit dieser Anweisung können beliebige Windows-Programme gestartet werden.
Das Programm kann mit einem Dateiauswahlfenster ausgewählt werden.

Das angegebene Windowsprogramm wird aus dem dem Visualisierungsprogramm [VisuWin](#) gestartet, dieses muss dazu aktiv sein.

Programmiersprache

Die Programmiersprache des Systems ist leicht zu erlernen und zur effektiven Programmierung des speziellen Anwendungsbereichs dieses Systems entwickelt. Diese Sprache ermöglicht es jedem Anwender ohne spezielle Programmierkenntnisse komplexe Steuerungsfunktionen zu realisieren. Für jedes [Objekt](#) kann ein [Makro](#) erstellt werden, welches durch die Änderung von Zuständen bzw. Werten die mit den Objekten verbundenen [Aktoren](#) steuert. Zustände und Werte von Objekten und [Variablen](#) sowie Zeiten können mit vielfältigen [Bedingungen](#) abgefragt und durch [Zuweisungen](#) verändert werden.

Anstelle der in der Beschreibung der einzelnen Anweisungen verwendeten Syntax kann auch die bei der automatischen Erstellung von Makro-Anweisungen verwendete Syntax verwendet werden.

In jeder Zeile eines Programms darf nur eine Anweisung stehen, Kommentarzeilen müssen mit 2 Schrägstrichen(//) beginnen, diese werden dann bei der Generierung des Codes ignoriert.

In den Beispielen sind Programmanweisungen teilweise eingerückt, dies ist nicht erforderlich, aber aufgrund besserer Übersichtlichkeit empfehlenswert.

Ein wichtiger Hinweis:

Bitte denken Sie daran Ihre Programm-Dateien (Endung SPG) vor Änderungen unter einem anderen Namen abzuspeichern oder zu kopieren. Änderungen können nicht rückgängig gemacht werden, wenn diese sich als fehlerhaft herausstellen, oder beim Speichern bzw. Beenden des Programms ein Fehler auftritt kann das vorhandene funktionierende Programm verloren sein. Bei jedem Speichern wird automatisch eine Sicherheitskopie der SPG-Datei mit der Endung "SPB" erstellt, diese wird bei jedem erneuten Speichern wieder überschrieben, wenn also ein fehlerhaftes Programm zweimal abgespeichert wird, ist das Original verloren. Grundsätzlich sollten unbedingt auch Sicherungskopien der SPG-Datei auf einem separaten Datenträger gespeichert werden, so dass diese im Falle eines Festplattendefekts nicht verloren geht.

Wenn während der Code-Erstellung Fehler auftreten, wird das fehlerhafte Objektprogramm mit einer entsprechenden Fehlermeldung angezeigt. Die fehlerhafte Zeile wird soweit möglich markiert, hier kann es jedoch zu Verschiebungen kommen, da Fehler im Quell-Code nicht immer einer Zeile zugeordnet werden können. Wenn in der markierten Zeile kein Fehler erkennbar ist, müssen die vorangegangenen Zeilen geprüft werden.

Anweisungsteile, die bei der Erklärung der Syntax in < spitzen Klammern > stehen sind optional und können weggelassen werden.

Die Uhrzeit wird nicht jede Sekunde, sondern alle 5 Sekunden aktualisiert. Bei Zeitvergleichen ist also zu beachten, das die Sekunden immer durch 5 teilbar sein müssen.

Wenn Schaltmodule eingeschaltet werden, kann die Einschaltdauer angegeben werden.

Grundsätzlich kann man zwischen Anweisungen und Funktionen einer Programmiersprache unterscheiden.

[Anweisungen](#) beschreiben in der Programmzeile was getan werden soll.

[Funktionen](#) liefern in der Programmzeile Werte, die zur Ausführung erforderlich sind.

Bitte beachten Sie folgende Hinweise:

[Besonderheiten bei Dimmern](#)

Variablen

Variablen dienen der Zwischenspeicherung von Werten. Variablen werden wie die Objektprogramme auf der [Seite Programmierung](#) der [Objektdefinition](#) definiert. Einer Variablen kann im [Objektprogramm](#) ein Wert zugewiesen werden. Dieser Wert kann der Inhalt einer anderen Variablen, ein Objektzustand oder eine Konstante sein. Die in einem Objektprogramm definierten Variablen können von allen anderen Objektprogrammen benutzt werden, wenn der Name des Objekts, in dem die Variable definiert wurde der Variablen getrennt durch einen Punkt vorangestellt wird.

Der Name einer Variablen kann die Buchstaben A-z, Ziffern und den Unterstrich beinhalten. Er muss mit einem Buchstaben beginnen, landesspezifische Buchstaben und Sonderzeichen sind nicht erlaubt (Beispiel: Var_99).

Für jede Variable muß ein Typ festgelegt werden. Es können die gleiche Typen wie für Objekte verwendet werden. Zusätzlich zu diesen Typen können für Variablen auch Zeittypen (Uhrzeit, Datum) benutzt werden.

Bei Zuweisungen zwischen nicht gleichen Typen erfolgt die [Zuweisung](#) entsprechend der Reihenfolge der für diesen Typ definierten Zustände, unabhängig von der Bezeichnung des Zustands. Intern werden Zustände als Zahl entsprechend der Reihenfolge in der Definition des Typs behandelt. Bei Zuweisungen wird diese Zahl zugewiesen.

Es gibt eine vordefinierte Variable des Typs Uhrzeit für jedes Objekt. Der Name der Variablen ist **CT** (für **C**hange**T**ime). In dieser Variablen wird die jeweils letzte Änderungszeit eines Objekts gespeichert. Mit der Funktion [STOPPUHR](#)(Objektname.CT) kann ermittelt werden wie lange ein Objekt sich in seinem aktuellen Zustand befindet. Die aktuelle Schaltdauer eines Objekts kann auch mit der Funktion [SCHALTDAUER](#)(Objektname) ermittelt werden.

Bitte beachten Sie:

Wenn Sie eine Variable vom Typ Zahl definieren, gibt es zwei unterschiedliche Typen von Zahlen:

Zahlen mit Komma (Gleitkommazahlen) und Zahlen ohne Komma. Von welchem Typ die Zahl ist, wird durch den Startwert festgelegt. Wird als Startwert eine Zahl mit Komma angegeben (z.B. 1,0) so wird die Variable als Zahl mit Kommastellen behandelt, andernfalls als Zahl ohne Kommastelle.

Es macht bei Rechenoperationen und Zuweisungen einen grossen Unterschied wie eine Zahl definiert ist. Wenn einer Zahl ohne Kommastellen eine Zahl mit Kommastellen zugewiesen wird, werden die Kommastellen einfach abgeschnitten.

Wenn eine Rechenoperation nicht das gewünschte Ergebnis liefert, prüfen Sie wie die in der Anweisung verwendeten Zahlen definiert sind.

Es gibt vordefinierte Variablen, die verwendet werden können. Jedes Objekt hat beispielsweise eine Variable vom Typ Zeit mit dem Namen CT, in der die letzte Schaltzeit des Objekt abgespeichert ist. Um also die aktuelle Schaltdauer eines Objekts zu ermitteln kann man folgende Anweisung verwenden:
Anzeige:=Schaltdauer(Lampe.CT)

Ein Beispiel zur Verwendung von Variablen :

Die minimal und maximal-Temperatur sollen jede Stunde auf einer Anzeige ausgegeben werden. Im Objektprogramm des Objekts TempWG sind die Variablen MIN und MAX als Zahl definiert. Das Makro des Objekts TempWG:

```
wenn TempWG > MAX dann
  MAX:=TempWG
endewenn
wenn TempWG < MIN dann
  MIN:=TempWG
endewenn
```

Das Makro des Objekts Anzeige, das zu jeder vollen Stunde aktiviert wird:

Anzeige:="Min: "+TempWG.MIN+" Max: "+TempWG.MAX

Zuweisungen

Zustände bzw. Werte von [Objekten](#) oder [Variablen](#) können durch Zuweisungen geändert werden.

Die grundsätzliche Form einer Zuweisung im Programm ist

Ziel := Quelle

wobei Ziel ein Objekt oder eine Variable sein kann. Quelle kann ein Objekt, eine Variable oder eine Konstante sein kann. Bei Zuweisungen von Werten unterschiedlichen Typs wird soweit möglich automatisch eine Konvertierung entsprechend des Typs des Ziels vorgenommen.

Bei Zuweisungen zwischen nicht gleichen selbstdefinierten Typen erfolgt die Zuweisung entsprechend der Reihenfolge der für diesen Typ definierten Zustände, unabhängig von der Bezeichnung des Zustands. Zum Beispiel muß bei der Definition neuer Typen mit den Zuständen AUS/AN darauf geachtet werden, dass der Zustand AUS immer zuerst definiert wird, damit bei Zuweisung an andere AUS/AN-Objekte der richtige Zustand übertragen wird.

Bei der Verwendung von Schlüsselwörtern wie *einschalten* bei Zuweisungen oder *ausgeschaltet* in wenn-Bedingungen ist es erforderlich, dass AUS immer als erster Zustand definiert wird, da intern mit der numerischen Reihenfolge der Zustände gearbeitet wird. Durch diese Verfahrensweise ist es möglich, auch Zuweisungen zwischen Typen mit unterschiedlicher Bezeichnung für gleiche Zustände durchführen zu können. Bei der Definition der Zustände von Fenstern und Türen muss *offen* als erster Zustand und *zu* als zweiter Zustand definiert werden, damit die entsprechenden Schlüsselwörter bei Zuweisungen und Bedingungen benutzt werden können.

Beispiele für Zuweisungen:

LichtBad:=SchalterBad hat die gleiche Wirkung wie

LichtBad wie SchalterBad

Licht1Garten einschalten hat die gleiche Wirkung wie

Licht1Garten:=1

Wird einem Objekt ein anderer Zustand als Text zugewiesen, so muss dieser in Hochkommas gesetzt werden und genau der Schreibweise der Typdefinition entsprechen, auch Gross/Kleinschreibung muss berücksichtigt werden.

Beispiel:

Anstatt

Rolllade runterfahren

Könnte man auch die Anweisung

RollladeWohnen:="unten"

verwenden, das Wort **unten** muss genau wie in der Typdefinition geschrieben sein und in Hochkommas gesetzt werden.

Eine besondere Anweisung zum Einschalten von Lichtern oder Geräten ist die Anweisung

Objekt einschalten für Zeitdauer

als Zeitdauer kann die Zeit in Hochkommas oder als Variable angegeben werden.

Beispiel:

Stehlampe einschalten für "00:30:00"

oder

Dauer1:="00:10:00"

Gartenlicht einschalten für Dauer1

wobei Dauer1 eine Variable vom Typ *Uhr* oder *Zeichen* sein muss

Weitere besondere Schlüsselwörter für Zuweisungen sind:

AUSSCHALTEN

ÖFFNEN

SCHLIESSEN

Und für Rollladen:

RAUFFAHREN

RUNTERFAHREN

Rechenfunktion bei Zuweisungen numerischer Werte

Bei Zuweisungen an ein Objekt bzw. eine Variable des Typs *Zahl* können die vier Grundrechenarten benutzt werden. Damit ist es z.B. möglich Eingabewerte von Sensoren für die weitere Verarbeitung zu verändern.

Beispiel:

NeuTemp:=AltTemp-5

Rechenfunktion bei Zuweisungen von Uhrzeiten

Uhrzeiten können mit Hilfe der Operatoren **+** und **-** addiert bzw. subtrahiert werden.

Beispiel:

Sie wollen, dass ein Makro alle morgendlichen Aktivitäten um eine Viertelstunde verzögert. Dies könnte so aussehen:

RollRaufZeit:= RollRaufZeit + "00:15:00"

KaffeeEinZeit:= KaffeeEinZeit + "00:15:00"

Anstatt von Uhrzeiten in Hochkommata kann auch die Zeit in Sekunden angegeben werden. Um also 2 Minuten zu einer Uhrzeit zu addieren könnte man schreiben:

ZeitLicht1:=ZeitLicht1 + 129

Rechenfunktion bei Zuweisungen von Datumswerten

Es ist auch möglich ein Datum durch eine Rechenoperation zu ermitteln.

Das Datum 11 Tage nach dem aktuellen Tag kann man z.B. einfach ermitteln mit der Anweisung:

Zukunft:=Datum+11

wobei die Variable *Zukunft* natürlich vom Typ *Datum* sein muss.

Verknüpfungsfunktion bei Zuweisungen von Texten

Mit Hilfe von Zuweisungen können zwei oder mehr Zeichenketten verbunden werden. Wenn in der Anweisung Operanden eines anderen Typs als Zeichenkette benutzt wird, werden diese soweit möglich automatisch konvertiert.

Beispiel:

AnzeigeKueche:="Garage ist "+Garagentor

Wenn das Objekt Garagentor die Typen *auf* und *zuhat*, und das Garagentor auf ist erscheint in der Anzeige ***Garage ist auf***

AnzeigeBad:="Uhrzeit+" Uhr"

Der Text in der Anzeige sieht dann so aus: ***22:30:00 Uhr***

Bedingungen

Im Folgenden werden die Formen der Bedingungen in wenn-Anweisungen beschrieben, nicht die Wenn-Anweisung selber.

Details zu dieser Anweisung finden Sie im Kapitel [wenn-Anweisungen](#).

In wenn-Anweisungen wird aufgrund von Bedingungen entschieden welche weiteren Anweisungen ausgeführt werden. Eine Bedingung ist entweder WAHR oder FALSCH. Bedingungen können mit UND bzw. ODER verknüpft werden.

Bedingungen sind folgendermaßen aufgebaut:

wenn *Operand1* *Vergleichsoperator* *Operand2*

Beispiel:

wenn Temperatur kleiner 21,5

Wenn die Operanden unterschiedlichen Typs sind, findet soweit möglich eine automatische Konvertierung statt.

Wenn einer der Operanden eine Konstante ist, muß diese als *Operand2* stehen, damit eine korrekte Konvertierung durchgeführt werden kann.

Beispiel:

Richtig: wenn Uhrzeit = "15:15:00" dann

Falsch: wenn "15:15:00" = Uhrzeit dann

Standard-Vergleichsoperatoren

= oder **gleich**

<> oder **ungleich**

< oder **kleiner**

> oder **groesser**

<= (kleiner oder gleich)

>= (grösser oder gleich)

Besondere Vergleichsbedingungen

wenn Objekt eingeschaltet

hat die gleiche Wirkung wie

wenn Objekt = 1 bzw. wenn Objekt = "an"

wenn Objekt ausgeschaltet

hat die gleiche Wirkung wie

wenn Objekt = 0 bzw. wenn Objekt = "aus"

weitere besondere Schlüsselwörter für Bedingungen sind

GEÖFFNET und **GESCHLOSSEN** (für Türen, Fenster, Tore),

wenn Fenster **geoeffnet** dann

Zeit-Vergleiche

Bei Zeitvergleichen ist zu beachten, daß die Uhrzeit im 5-Sekunden-Takt aktualisiert wird, die Sekunden der Uhrzeit bei der Prüfung auf Zeitgleichheit also immer durch 5 teilbar sein müssen.

wenn **Uhrzeit = "HH:MM:SS"**

bei dieser Bedingung ist zu beachten, daß die Uhrzeit in Hochkommas gesetzt wird. Weiterhin müssen die Sekunden durch 5 teilbar sein, da die Bedingung sonst nie zutrifft. Stunden.Minuten und Tage

wenn **Datum** = "TT.MM.JJJJ"

bei dieser Bedingung ist zu beachten, daß das Datum in Hochkommas gesetzt wird und das Jahr vollständig, also 4-stellig mit Jahrtausend angegeben werden muss.

wenn **Wochentag** = "Wochentag"

Prüfung auf einen Wochentag. Der Wochentag wird in Hochkommas gesetzt und muß mit einem Großbuchstaben und nachfolgenden Kleinbuchstaben geschrieben werden.

Beispiel : wenn **Wochentag** = "Freitag" dann

wenn **Monat** Vergleichsoperator *Zahl*

Beispiel :

wenn **Monat** **groesser 4** und **Monat** **kleiner 10** dann

bewirkt, daß die folgenden Anweisungen nur zwischen Mai und September ausgeführt werden.

Vergleichsoperator =* für Vergleiche mit Jokerzeichen

Beim Vergleich von Uhrzeit und Datum mit einer Konstanten können auch Jokerzeichen verwendet werden. Als Vergleichsoperator muss dann =* verwendet werden. Operand2 muß eine Konstante in Hochkommas sein.

Beispiele:

Um Anweisungen jede volle und halbe Stunde auszuführen:

wenn **Uhrzeit** =* "***:00:00" oder **Uhrzeit** =* "***:30:00" dann

.....

Um Anweisungen immer am Ersten eines Monats auszuführen:

wenn **Datum** =* "01.**.****" dann

.....

Bedingung für Zeiträume

Manchmal ist es erforderlich Aktionen nur innerhalb bestimmter Zeiträume auszuführen.

Dies ist möglich mit der Bedingungsanweisung :

wenn **Uhrzeit/Datum** **zwischen** "*Zeitkonstante*" und "*Zeitkonstante*" dann

.....

Beispiel:

wenn **Uhrzeit** **zwischen** "23:00:00" und "05:00:00" dann

.....

Vergleichsoperator =+ für Vergleiche mit Wochenmaske

Mit diesem Vergleichsoperator ist es möglich Anweisungen nur an bestimmten Wochentagen ausführen zu lassen. Als Operand2 muß eine 7-stellige Konstante bestehend aus Nullen und Einsen in Hochkommas verwendet werden. Jede Stelle der Konstanten steht für einen Wochentag, beginnend mit Sonntag. Die Bedingung ist wahr wenn an der Stelle des aktuellen Wochentags eine Eins steht.

Beispiel : Es soll geprüft werden, ob der aktuelle Tag ein Freitag, Samstag oder Sonntag ist.

wenn Wochentag =+ "1000011" dann

ABFRAGE

Syntax :

ABFRAGE(*Objektname*)

Mit dieser Anweisung kann der aktuelle Zustand/Wert der Hardware eines Objekts abgefragt werden. Normalerweise ist das nicht nötig, da Änderungen automatisch aktualisiert werden. In speziellen Anwendungen kann es sinnvoll sein diese Anweisung zu verwenden, sie sollte jedoch nicht zu häufig und möglichst nur bei sicherheitsrelevanten Objekten benutzt werden, da sie relativ zeitintensiv ist und es bei zu häufiger Verwendung zu Verzögerungen bei der normalen Kommunikation kommen kann,

Beispiel:

ABFRAGE (Tiefkuehltruhe)

AKTIVIEREN

Syntax :

AKTIVIEREN(OBJEKT)

Das angegebene Objekt (und somit auch das Makro des Objekts) wird aktiviert, d.h. das Makro des Objekts wird entsprechend seines Aktivierungsintervalls oder bei Betätigung eines zugeordneten Sensors ausgeführt und das Objekt kann am Bildschirm wieder angeklickt werden.

siehe auch [DEAKTIVIEREN](#)

ANZEIGEN

Syntax :

ANZEIGEN(OBJEKT)

Mit dieser Anweisung wird der Text und die über Variablen eingestellten Zusatzinformationen zu einem speziellen Anzeige-Hardwaremodul (z.B. [19-Tasten-Fernbedienung](#)) gesendet.

Für die 19-Tasten-Fernbedienung ist das Objekt, das als Parameter eingetragen werden muss das letzte Objekt der Fernbedienung,
also zum Beispiel:

```
Anzeigen(FernbedienungW_18)
```

AUFRUFEN

Syntax :

AUFRUFEN (OBJEKT)

Diese Anweisung führt das angegebene Makro sofort aus. Nach Beendigung des aufgerufenen Makros wird die Ausführung mit der nächsten Anweisung fortgesetzt.

DEAKTIVIEREN

Syntax :

DEAKTIVIEREN(OBJEKT)

Das angegebene Objekt wird deaktiviert. Die Programme deaktivierter Objekte werden nur ausgeführt durch einen direkten Aufruf mit dem Befehl [STARTE](#) oder [AUFRUFEN](#). Alle anderen Aktivierungen aufgrund des angegebenen Ausführungsintervalls oder des Empfangs von mit dem Objekt verbundenen Steuerungselementen sind gesperrt.

Das Objekt kann am Bildschirm nicht mehr angeklickt werden.

siehe auch [AKTIVIEREN](#)

ERLEDIGT

Syntax :

ERLEDIGT(OBJEKT)

Das angegebene Objekt wird als erledigt gekennzeichnet, d.h. die nächste [GESCHALTET](#) - Abfrage gibt den Wert FALSCH zurück, falls sich der Zustand des Objekts nach der ERLEDIGT - Anweisung nicht wieder geändert hat.

siehe auch GESCHALTET

PLAY

Syntax :

PLAY("Pfad\Datei")

Die in Klammern angegebene Audio-Datei wird abgespielt.

Bitte beachten Sie:

Diese Funktion wird im Programm VisuWin ausgeführt, wenn kein Programm [VisuWin](#) aktiv ist wird diese Anweisung ignoriert.

Wenn die Datei nicht existiert oder es sich nicht um eine Wave-Datei handelt, so wird diese Anweisung ignoriert. Es wird keine Fehlermeldung ausgegeben, um die weitere Ausführung nicht zu behindern.

Beispiel:

```
wenn Klingelknopf gedruickt dann
  wenn Uhrzeit zwischen "10:00:00" und "18:00:00" dann
    PLAY ("C:\PROGRAMME\HOMEPUTER\WAV\TROMPETE.WAV")
  sonst
    PLAY ("C:\PROGRAMME\HOMEPUTER\WAV\FLOETE.WAV")
  endewenn
endewenn
```

SCHLIESSEDATEIEN

Syntax :

SCHLIESSEDATEIEN

Mit dieser Anweisung werden alle geöffneten Dateien, die mit der Anweisung `SCHREIBEDATEI` erstellt worden sind geschlossen.

Diese Anweisung muss verwendet werden damit keine Daten verloren gehen bevor die Zentrale ausgeschaltet wird um den USB-Stick zu entfernen.

Beispiel:

```
SCHLIESSEDATEIEN
```

SCHREIBEDATEI

Syntax :

SCHREIBEDATEI("Dateiname","Text",Zeilenvorschubcode)

Mit dieser Anweisung können Textdateien erstellt werden. Die Textdateien werden auf einem USB-Stick, der in der Zentrale eingesteckt sein muss geschrieben.

Als erster Parameter in der Klammer wird der Dateiname angegeben, als zweiter Parameter die Zeile, die in die Datei geschrieben werden soll. Beide Parameter sind vom Typ Zeichen und können Konstanten oder Variablen sein. Der Dateiname kann auch einen gültigen Pfad beinhalten, dieser muss dann aber auf dem USB-Stick vorhanden sein erstellt werden.

Als dritter Parameter wird angegeben welcher Zeilenvorschub für die ausgegebene Zeile verwendet werden soll. Dieser Parameter ist vom Typ Zahl und kann als Konstante oder Variable angegeben werden. Es gibt folgende Codes für den Zeilenvorschub:

0 = kein Zeilenvorschub, die nächste Ausgabe beginnt unmittelbar hinter dieser Zeile

1 = Linux Zeilenvorschub (LF), es wird nur ein Linefeed ausgegeben, mit einem Editor unter Windows wird normalerweise kein Zeilenvorschub dargestellt.

2 = Windows-Zeilenvorschub (CR/LF), es werden die für Windows üblichen zwei Codes für den Zeilenvorschub ausgegeben, so dass die Datei nach einem Download unter Windows mit einem Texteditor betrachtet werden kann. Im Normalfall sollte diese Option benutzt werden.

Bevor der USB-Stick entfernt wird muss die Anwendung **SCHLIESSEDATEIEN** ausgeführt werden, sonst gehen die zuletzt geschriebenen Daten bzw. Dateien verloren.

Bitte beachten Sie unbedingt:

Der USB-Stick muss formatiert sein und **vor** dem Start der Zentrale eingesteckt werden. Nach dem Einstecken eines USB-Sticks muss die Zentrale ggfs. neu gebootet werden oder der USB-Stick muss über den entsprechenden Punkt des WEB-Servers eingebunden werden.

Das passiert mit dem untersten Button auf der Seite **<http://<IP der Zentrale>/addons/contronics/files/workfiles.cgi>**

Über dies WEB-Serverseite können die Dateien auch bearbeitet werden und es kann ein Download der Dateien durchgeführt werden.

Wenn der USB-Stick nicht eingebunden worden ist, wird auf die RAMDISK der Zentrale geschrieben.

Wenn die Dateien dort zuviel Platz beanspruchen wird die Funktion der Zentrale beeinträchtigt!

Die Benutzung dieser Anweisung ohne eingestecktem USB-Stick ist daher sehr riskant. Wenn diese Anweisung ohne USB-Stick benutzt wird muss auf jeden Fall dafür gesorgt werden, dass die erstellten Dateien regelmässig mit der Anweisung **LÖSCHEDATEIEN** gelöscht werden bevor diese zu gross werden. Wieviel Platz zur Verfügung steht hängt letztlich im Einzelfall vom Typ der Zentrale und von der individuellen Konfiguration und Anwendung ab.

Beispiel:

(Zeile ist im Maro als Variable vom Typ Zeichen definiert)

```
Zeile:="Uhrzeit " + Uhrzeit + ": Bewegungsmelder Haustür"  
SCHREIBEDATEI ("BEWEGUNGEN.TXT",Zeile,2)
```

GEHEZU

Syntax :

GEHEZU ADRESSE

.....

ADRESSE:

.....

Achtung: Bei falscher Anwendung dieser Anweisung können Endlosschleifen entstehen, durch die das aktuelle Makro ständig läuft ohne beendet zu werden.

Eine gefährliche Anweisung, die im Normalfall nicht gebraucht wird und die man vermeiden sollte.

Die GEHEZU-Anweisung bewirkt, daß die Programmausführung an der durch *ADRESSE* festgelegten Stelle fortgesetzt wird. Als Adresse kann ein beliebiger Name aus Buchstaben und Ziffern gewählt werden, das erste Zeichen muss ein Buchstabe sein. Die Adresse muß immer am Zeilenanfang beginnen und mit einem Doppelpunkt abgeschlossen werden. Innerhalb eines Objektprogramms muß eine Adresse eindeutig sein.

Diese Anweisung sollte möglichst vermieden werden, da bei falscher Anwendung Endlosschleifen entstehen können und die Ausführung des aktuellen [Objektprogramms](#) nicht beendet wird. Wenn die Sprungadresse vor der GEHEZU-Anweisung liegt, wird die Ausführung des Makros für einen Prozesstakt unterbrochen, damit andere Makros und Ein/Ausgaben ausgeführt werden können, und eine eventuelle Endlosschleife nicht das komplette System blockiert.

Beispiel:

WENN Uhrzeit > "22:00:00" DANN

GEHEZU ABEND

ENDEWENN

Anweisungen

.....

GEHEZU ENDE

ABEND:

Anweisungen

.....

ENDE:

Anweisungen

LÖSCHEANZEIGE

Syntax :

LÖSCHEANZEIGE(OBJEKT)

Mit dieser Anweisung werden die Zusatzvariablen eines speziellen Anzeigemoduls (z.B. [19-Tasten-Fernbedienung](#)) gelöscht.

Diese Anweisung bewirkt keine Aktualisierung des Hardwaremoduls, falls dieses aktualisiert werden soll, muss zusätzlich die Anweisung [ANZEIGEN](#) benutzt werden.

LÖSCHEDATEI

Syntax :

LÖSCHEDATEI("Dateiname")

Mit dieser Anweisung können Dateien gelöscht werden. Das können selbst erstellte Textdateien, aber auch beliebige andere Dateien wie die Historydatei **history.txt** oder die Systemlog-Datei **syslog.txt** sein.

Bitte beachten Sie unbedingt:

Durch das beabsichtigte oder unbeabsichtigte Löschen von Dateien, die Sie nicht selbst erstellt haben kann die Systemintegrität zerstört und die Funktionen der Zentrale beeinträchtigt werden.

**Löschen Sie daher keinesfalls Dateien, von denen Sie nicht wissen wofür diese benutzt werden .
Achten Sie unbedingt darauf, dass das auch nicht versehentlich, etwa durch Angabe eines falschen Dateinamens geschieht.**

Beispiel:

```
LÖSCHEDATEI ("BEWEGUNGEN.TXT")
```

SENDEMAIL

Syntax :

SENDEMAIL("Zieladresse", "Betrefftext", "Erste Zeile<<zweite Zeile<<dritte Zeile")

Mit dieser Anweisung können e-mails versendet werden. Voraussetzung ist, dass im Netzwerk, in dem die Zentrale angeschlossen ist, eine Verbindung zum Internet besteht oder automatisch aufgebaut wird sobald die Zentrale eine Verbindungsanforderung zu einem mail-Server schickt.

Als erster Parameter der Anweisung wird die Zieladresse angegeben, als zweiter Parameter folgt der Betrefftext, als dritter Parameter der Text der email. Wenn mehrere Zeilen Text gesendet werden sollen, so können die <<-Zeichen als Zeilenvorschub-Zeichen verwendet werden.

Mit der Funktion [MAILLOG](#), kann der Status der jeweils letzten fünf versendeten emails überprüft werden.

Bitte beachten Sie, dass jedesmal wenn die Anweisung ausgeführt wird eine mail gesendet wird. Die Programmierung muss so aufgebaut sein, dass nicht ungewollt viele mails versendet werden. Werden mails z.B. als SMS auf das Handy gesendet, können Fehler in der Programmierung, die ungewollt viele mails erzeugen, zu erheblichen Kosten führen. Es müssen also Vorkehrungen getroffen werden, um ein zu häufiges Versenden von emails zu verhindern.

Im folgenden Beispiel verhindert die Abfrage nach `MaxTemp`, dass bei konstanter Temperatur z.B. über einem Grenzwert dauernd gesendet würde.

Beispiel:

```
wenn TempASH550 > MaxTemp dann
  Mailtext:="Info Treibhaus:<<Temperatur ist " + TempASH550 + " Grad"
  sendemail("Meine.BueroAdresse@Mein-Provider.com", "Temperaturinfo", Mailtext)
  MaxTemp:=MaxTemp + 0.5
endewenn
```

Würde die Abfrage dagegen nur z.B. lauten

```
wenn TempASH550 > 25 dann
so würde bei jeder Makroausführung, bei der eine höhere Temperatur als 25 Grad erreicht ist eine mail
gesendet, also z.B. alle paar Minuten wenn ein Temperaturwert empfangen wird, oder gar alle 5 Sekunden,
wenn das Makro in diesem Zeitintervall ausgeführt würde.
```

Eine andere Möglichkeit ist, das Versenden von emails z.B. über "virtuelle" Objekte zu steuern. Im folgenden Beispiel würde z.B. nur jede halbe Stunde eine mail verschickt:

```
wenn mailSchalter eingeschaltet und Schaltdauer(mailSchalter) > "00:30:00"
dann
  mailSchalter ausschalten
endewenn
wenn mailSchalter ausgeschaltet dann
  sendemail("Meine.BueroAdresse@Mein-Provider.com", "Wasseralarm", "Wasser im
Keller !")
  mailSchalter einschalten
endewenn
```

Beachten Sie bitte, dass die Funktion `SCHALTDAUER` nur für Objekte, nicht aber für Variablen verwendet werden kann.

Eine weitere Möglichkeit ist z.B. die Zeiten zwischen zwei mails über `STARTUHR` und `STOPPUHR` zu begrenzen.

```
wenn STOPPUHR(LetzteMail) > "00:30:00" dann
  sendemail("Meine.BueroAdresse@Mein-Provider.com", "Wasseralarm", "Wasser im
Keller !")
  STARTUHR(LetzteMail)
endewenn
```

Bitte unbedingt beachten:

Testen Sie Ihre Programmierung und überprüfen ob nicht unbeabsichtigt viele mails versendet werden.
Beachten Sie, dass mails Kosten verursachen können, z.B. durch Verbindungsentgelte oder SMS auf ein Handy.

SETZEWERT

Syntax :

SETZEWERT(Objektname, Wertecode, Wert)

Mit dieser Anweisung können verschiedene Einstellungen in den Hardwaremodulen vorgenommen werden. Diese Anweisung erzeugt immer einen Sendebefehl und sollte nur selten verwendet werden, Sie darf keinesfalls in Makros erwendet werden, die in kurzen Zeitintervallen ausgeführt werden, das könnte das ganze System blockieren.

Es muss auch unbedingt beachtet werden, dass bei Einstellungen über diese Anweisung nur ein Sendebefehl erzeugt wird. Die Werte und Zustände von Objekten und Variablen werden nicht direkt beeinflusst, sondern erst durch eventuelle Rückmeldungen von den Hardwaremodulen aktualisiert. Das kann zur Folge haben, dass Hardware und Visualisierung nicht oder nicht zeitgleich synchron sind und Zustandsabfragen in wenn-Bedingungen nicht aktuell sind. Daher sollte diese Möglichkeit den Zustand von Hardwaremodulen zu ändern nur in Ausnahmefällen benutzt werden.

Die Anweisung sollte nur von erfahrenen Benutzern verwendet werden, die unsachgemässe Verwendung dieser Anweisung kann zu schweren Fehlfunktionen führen.

Benutzen Sie diese Anweisung nur wenn Sie genau wissen was Sie tun !!

Die Wertecodes können Konstanten oder auch Variablen vom Typ Zeichen sein.

Die Wertecodes müssen in Grossbuchstaben geschrieben sein, es kann auch die englische Bezeichnung (in Klammern) verwendet werden.

Folgende Wertecodes werden momentan unterstützt:

EINSCHALTZEIT (ON_TIME), Einheit ist Sekunden

Beispiel:

```
Setzewert (Stehlampe, "EINSCHALTZEIT", 30)  
Stehlampe einschalten
```

Mit diesem Schlüsselwert kann z.B. für Schaltaktoren und Dimmer die Einschaltzeit eingestellt werden werden. Es ist keine Dauer-Einstellung, sie ist nur gültig für den nächsten Einschaltbefehl an das Modul. Bei dieser Möglichkeit ein Gerät für eine bestimmte Zeit einzuschalten, wird der interne Timer des Moduls benutzt, d.h. das Gerät schaltet nach der angegebenen Anzahl von Sekunden aus, auch wenn die Zentrale nicht mehr aktiv ist. Die Verwendung dieser Methode kann bei sicherheitskritischen Funktionen vorteilhaft sein.

VERZÖGERUNG (RAMP_TIME), Einheit ist Sekunden

Beispiel:

```
Setzewert (Dimmer, "VERZÖGERUNG", 8)
```

Mit diesem Schlüsselwert kann z.B. die Dimmggeschwindigkeit von Dimmern gesteuert werden.

ALTE STUFE (OLD_LEVEL)

Beispiel:

```
Setzewert (Dimmer, "ALTE STUFE", 1)
```

Mit diesem Schlüsselwert kann ein Dimmer auf seine letzte Dimmstufe vor dem Ausschalten gesetzt werden. Die 1 ist als fester Wert erforderlich.

STUFE (LEVEL)

Beispiel:

```
Setzewert (Dimmer, "STUFE", "0.6")
```

Mit diesem Schlüsselwert kann die Helligkeit eines Dimmers eingestellt werden. Der Wert ist ein Dezimalwert zwischen 0 und 1, als Parameter in Form einer Konstante oder Variable vom Typ Zeichen.

Diese Möglichkeit der Dimmerssteuerung sollte nur in Ausnahmefällen benutzt werden, besser ist es den Dimmer über eine Zuweisung in einem Makro (z.B. Dimmer:=60) einzustellen.

ZUSTAND (STATE)

Beispiel:

Setzwert(Stehlampe,"ZUSTAND",1)

Mit diesem Schlüsselwert kann ein Binäraktor geschaltet werden. Bei Wert 1 wird eingeschaltet, bei Wert 0 wird ausgeschaltet. Diese Steuerungsmöglichkeit sollte nur in Ausnahmefällen verwendet werden, besser ist es Binäraktoren über die Anweisungen "einschalten" und "ausschalten" zu steuern.

SICHTBAR

Syntax :

SICHTBAR(*Objekt*)

Das angegebene Objekt wird auf dem Bildschirm sichtbar gemacht. Falls es schon sichtbar war hat diese Anweisung keine Auswirkung. Ein Objekt kann mit der Anweisung [UNSICHTBAR](#) unsichtbar gemacht werden.

siehe auch [UNSICHTBAR](#)

SPRACHAUSGABE

Syntax :

SPRACHAUSGABE(Variable)

Diese Funktion gibt den Text-Inhalt der Variablen als Sprache auf die Soundkarte des PC's aus. Dazu muss auf einem PC das Programm [VisuWin](#) aktiv sein. Ist kein Programm VisuWin aktiv, so wird diese Anweisung ignoriert.

Anstelle einer Variablen kann auch ein Text in Hochkommas verwendet werden.

Um die standardmässig in Windows vorhandenen englischen Stimmen der Sprachausgabefunktion SAPI 5 mit besserer landesprachlicher Aussprache auszugeben, kann ein Sternchen vor den auszugebenden Text gesetzt werden. Die beste Qualität erreicht man jedoch mit der Installation der landesprachlichen Stimmen. Diese stehen zusammen mit einigen kostenfreien Microsoft-Programmen wie z.B. Microsoft Reader oder MS-Autoroute 2007 als Download zur Verfügung.

Weitere Informationen zur [Sprachausgabe](#)

Beispiel:

```
Sprachausgabe("Das Licht wurde eingeschaltet")
```

Wenn zusammengesetzter Text ausgegeben werden soll, muss dieser vorher in einer Variablen erstellt werden:

```
Text:="Die Temperatur im Terrarium ist mit "+TempTerrarium+" zu niedrig, bitte prüfen"
```

```
Sprachausgabe(Text)
```

STARTE

Syntax :

STARTE(OBJEKT)

Diese Anweisung stellt das angegebene Makro in die Ausführungswarteschlange. Die Programme der Objekte in der Ausführungswarteschlange werden nach dem FIFO (First In First Out) -Prinzip abgearbeitet. Nach jeder Ausführung eines Programms innerhalb des normalen Ausführungsintervalls werden alle Programme in der Ausführungswarteschlange ausgeführt.

Die Ausführung des Makros wird also erst gestartet nachdem das aktuelle Makro beendet worden ist.

STARTUHR

Syntax :

STARTUHR(*Zeitvariable*)

Die aktuelle Zeit wird in der Zeitvariablen gespeichert. Dies ermöglicht eine Stoppuhrfunktion indem mit der Funktion [STOPPUHR](#)(Zeitvariable) die Differenz zwischen gespeicherter und aktueller Zeit ermittelt wird.

Beispiel:

Die Variable Uhr1 ist als Variable vom Typ Uhrzeit definiert.

STARTUHR(Uhr1)

Die seit der Ausführung der obigen Anweisung vergangene Zeit kann mit der Anweisung [STOPPUHR](#)(Uhr1) ermittelt werden.

siehe auch [STOPPUHR](#)

STARTWIN

Syntax :

STARTWIN("Pfad\Programm,<Parameter>,<Arbeitsverzeichnis>")

Startet ein Windows-Programm.

Die Parameter <Parameter> und <Arbeitsverzeichnis> sind optional, bitte beachten Sie, dass das nicht automatisch als Pfad für das Programm benutzt wird, der Pfad für das Programm muss im ersten Parameter angegeben werden.

Beispiel:

Wenn eine bestimmte Temperatur unterschritten wird, soll ein vordefiniertes Fax geschickt werden.

WENN Temp1 KLEINER 15 DANN

STARTWIN("C:\FAX\SENDEFAX.EXE,HEIZUNGSSTOERUNG")

ENDEWENN

STOPPE

Syntax :

STOPPE(*Objektname*)

Mit dieser Anweisung kann der aktive Verstellvorgang von Rollladen- und Markisensteuerungen unterbrochen werden.

Beispiel:

STOPPE (RolloWohnen)

TEMPERATURABSENKUNG

Syntax :

TEMPERATURABSENKUNG(*Raumreglername,Absenktemperatur*)

Diese Anweisung ist nur für das Makro von Tür/Fensterkontakten vorgesehen. Damit kann ein Raumregler direkt in Abhängigkeit von Tür-Fensterkontakten über Makros sehr einfach gesteuert werden, ohne dass man bei der Erstellung der Anwendung in den Makros die unterschiedlichen Zustände von mehreren Fensterkontakten für einen Raumregler berücksichtigen muss. Mit dieser Anweisung können bis zu 8 Fensterkontakte für einen Raumregler verwendet werden. Beim Öffnen eines Fensters wird die in der Anweisung angegebene Absenktemperatur für den Raumregler eingestellt. Erst wenn alle Fenster wieder geschlossen sind, wird die zuletzt am Raumregler eingestellt Temperatur wieder aktiviert.

Wenn diese Anweisung benutzt wird, darf der Fensterkontakt nicht durch eine direkte Verknüpfung mit dem Raumregler verbunden sein und der Modus des Raumreglers muss auf "manuell" stehen. Andernfalls ist eine Steuerung des Raumreglers über die Makros der Fensterkontakte nicht möglich.

Es ist zu beachten, dass der Abgleich von Temperaturen einige Minuten dauern kann.

Wenn die Absenktemperatur eine Kommastelle hat, muss die Temperatur in Hochkommata gesetzt werden.

Beispiel:

```
TEMPERATURABSENKUNG (Raumregler1, "17,5")
```

UNSICHTBAR

Syntax :

UNSICHTBAR(*Objekt*)

Das angegebene Objekt wird auf dem Bildschirm nicht mehr angezeigt. Es kann mit der Anweisung [SICHTBAR](#) wieder zur Anzeige gebracht werden.

siehe auch [SICHTBAR](#)

VERLASSEN

Syntax :

VERLASSEN

Das aktuelle Makro wird sofort beendet. Die Ausführung wird mit dem nächsten anstehenden Makro fortgesetzt.

Beispiel:

WENN Uhrzeit groesser "23:00:00" DANN

VERLASSEN

ENDEWENN

WARTE

Syntax :

WARTE ("00:00:10")

Als Parameter für diese Anweisung kann die Zeit in Hochkommas oder eine Zeichen- oder Uhrzeit-Variable verwendet werden. Falls eine Variable verwendet wird darf diese natürlich nicht in Hochkommas gesetzt werden. Beispiel : *WARTE(Wartezeit)*

Mit der Anweisung WARTE kann die Ausführung eines Makros für die angegebene Zeit unterbrochen werden. Diese Anweisung hat keine Auswirkung auf andere Makros und Funktionen, diese laufen weiter. Nach der angegebene Zeit wird die Ausführung mit der nächsten Anweisung hinter der WARTE-Anweisung fortgesetzt.

Eine WARTE-Anweisung wird abgebrochen, wenn das Makro vor beenden der Wartezeit erneut aufgerufen wird.

WENN

Syntax :

WENN <NICHT> *Bedingung* <UND/ODER> *Bedingung* **DANN**

Anweisungen

<**SONST**>

Anweisungen

ENDEWENN

Die wenn-Anweisung ist die einzige Anweisung, die sich über mehrere Zeilen erstrecken kann.

Der **wenn**-Teil kann in mehreren Zeilen stehen während **sonst** und **endewenn** –Zeilen jeweils einzelne Zeilen sein müssen.

Mit der WENN-Anweisung ist es möglich den weiteren Programmablauf von einer oder mehreren [Bedingungen](#) abhängig zu machen. Wenn-Anweisungen können auch verschachtelt werden, d.h. zwischen dem **wenn** und dem **endewenn** (bzw. sonst) können weitere wenn-Anweisungen stehen. Jede wenn-Anweisung muß mit einer endewenn-Anweisung beendet werden, ansonsten wird bei der Code-Generierung eine entsprechende Fehlermeldung ausgegeben.

Beispiele:

```
WENN LichtBad ausgeschaltet UND
    SCHALTDAUER(LichtBad) groesser "00:05:00" DANN
    Ventilator ausschalten
ENDEWENN
```

```
WENN Temperatur < 20.5 dann
    Heizluefter einschalten
SONST
    Heizluefter ausschalten
ENDEWENN
```

Vor jeder Bedingung kann ein **NICHT** gesetzt werden, dann wird die Anweisung hinter dann ausgeführt wenn die Bedingung **nicht** zutrifft.

Bitte beachten Sie:

Das Wort **NICHT** muss **vor** der eigentlichen Bedingung stehen, es darf **nicht in der Bedingung** stehen.

Beispiel:

Falsch wäre die umgangssprachliche Formulierung:

wenn LichtBad **NICHT** ausgeschaltet oder Wochentag **NICHT**="Montag" dann
es würde ein Syntaxfehler angezeigt.

Richtig ist:

wenn **NICHT** LichtBad ausgeschaltet oder **NICHT** Wochentag="Montag" dann

Bitte beachten Sie:

Häufige Syntaxfehler in wenn-Anweisungen sind, dass das Wort **dann** vergessen wird und dass die Anweisung nicht mit einem **endewenn** abgeschlossen wird.

Da wenn-Anweisungen sich immer über mehrere Zeilen erstrecken, kann die Zeile für einen Syntaxfehler oft nicht bestimmt werden. Wenn in einem Makro mit einer wenn-Anweisung ein Syntaxfehler ohne Fehlerbeschreibung auftritt, prüfen Sie alle Elemente der wenn-Anweisung um den Fehler zu finden.

AKTIVIERT

Syntax :

AKTIVIERT(*Objekt*)

Mit dieser Funktion kann festgestellt werden, ob ein Objekt aktiviert ist. Diese Funktion ist z.B. nützlich wenn ein Programm für besondere Aktionen nur zeitweise aktiviert wird.

Beispiel:

```
wenn AKTIVIERT (Alarmanlage) dann
  Starte (AlleLichterAn)
endewenn
```

BATTERIELEER

Syntax :

BATTERIELEER(*Objekt*)

Mit dieser Funktion kann in Bedingungen geprüft werden, ob ein Modul eine schwache Batterie gemeldet hat.

Die Funktion ist WAHR, wenn eine Batteriewarnung von dem Modul empfangen wurde.

Beispiel:

```
wenn BATTERIELEER(TEMPERATURSENSOR) dann  
  Anzeige:="Batterie Temperatursensor austauschen"  
endwenn
```

Hinweis: Um auf Batteriewarnungen zu reagieren kann ein Objekt vom Typ Zeichen definiert werden, nähere Infos dazu unter [Spezielle Objekte](#). Jedesmal wenn eine Batteriewarnung empfangen wird, enthält das Objekt den Namen des Moduls, für das die Meldung gesendet wurde und das Makro des Objekts wird ausgeführt.

DATEIVORHANDEN

Syntax :

DATEIVORHANDEN

Mit dieser Funktion kann geprüft werden ob eine Datei existiert.

Beispiel:

```
wenn DATEIVORHANDEN("BEWEGUNGEN.TXT") dann
  Anzeige:="Meldungen vom Bewegungsmelder vorhanden"
endwenn
```

DATUM

Syntax :

DATUM

Diese Funktion gibt das aktuelle Datum im Format TT.MM.JJJJ zurück.

Beispiel:

```
WENN DATUM =* "15.02.****" UND UHRZEIT="08:00:00" DANN  
PLAY("c:\AudioDaten\homeputer\HappyBirthday.wav")  
ENDEWENN
```

GESCHALTET

Syntax :

GESCHALTET(Objekt)

Diese Funktion wird ausschließlich in Bedingungen verwendet, um zu prüfen ob ein Objektzustand sich geändert hat.

Beispiel:

```
WENN GESCHALTET(Alarmsensor) DANN
  SENDEMAIL(. . . . .)
  ERLEDIGT(Alarmsensor)
ENDEWENN
```

Durch die Anweisung [ERLEDIGT](#) werden die weiteren GESCHALTET-Abfragen unwahr bis der Zustand des Objekts LichtBad sich z.B. durch Schalterbetätigung ändert. Die folgenden Anweisungen würden Beispielsweise verhindern, dass eine Alarm-mail mehrmals gesendet wird, da die GESCHALTET-Bedingung nur bei der ersten Ausführung der Anweisung zutrifft.

siehe auch [ERLEDIGT](#)

JAHR

Syntax :

JAHR

Diese Funktion gibt das aktuelle Jahr als Zahl zurück.

MAILLOG

Syntax :

MAILLOG(Index)

Mit dieser Funktion können die Logs der letzten 5 e-mail-Sendungen abgerufen werden. Der Index 1 der Tabelle enthält immer den Status der zuletzt gesendeten e-mail, der Index 5 den Status der ältesten der letzten fünf mails.

Die Status-Infos für eine mail werden als Text mit drei Zeilen gespeichert. Die Zeilen werden jeweils durch << getrennt und können daher direkt auf ein Anzeigeobjekt vom Typ Zeichen ausgegeben werden.

In der ersten Zeile stehen Datum, Uhrzeit und der Status der Sendung OK=korrekt versandt, bei einem Fehler das Wort Error und eine dreistellige Fehlerkennung.

In der zweiten Zeile steht die Zieladresse der mail.

In der dritten Zeile steht der Betrefftext der mail.

Die Fehlerkennungen haben folgende Bedeutungen:

104 = Mailserver antwortet nicht. Mögliche Gründe: z.B. keine Verbindung zum Internet, Server nicht aktiv oder eventuell falscher Servername

106 = Verbindungsaufbau mit Server fehlgeschlagen. Möglicher Grund: z.B. falscher Port

108 = Servername nicht gefunden. Möglicher Grund: falscher Servername

109 = Verbindungsaufbau des Netzwerks fehlgeschlagen, Client wurde nicht akzeptiert.

128 = Anmeldung fehlgeschlagen. Möglicher Grund: z.B. Benutzername oder Kennwort falsch

134 = Fehler der Zugangs- oder Sendedaten, Benutzername / Kennwort falsch oder Absendeadresse entspricht nicht der korrekten Mailadresse

Beispiel:

```
Anzeige1 := MAILLOG(1)
```

Anzeige1 hätte danach folgenden z.B. Textinhalt:

02.06.2008 12:23:20 OK<<Meine.BueroAdresse@Mein-Provider.com<<Temperaturinfo

Auf dem Anzeigepanel des Objekt Anzeige1 würde dann Meldung erscheinen:

02.06.2008 12:23:20 OK

Meine.Bueroadresse@Mein-Provider.com

Temperaturinfo

MONAT

Syntax :

MONAT

Diese Funktion gibt den aktuellen Monat als Zahl zurück.

Beispiel:

```
WENN Monat = 5 DANN  
    Anzeig1:="Der Mai ist gekommen.."  
ENDEWENN
```

MONATSTAG

Syntax :

MONATSTAG

Diese Funktion gibt den aktuellen Tag des Monats als Zahl zurück.

Beispiel:

```
WENN MONATSTAG=1 DANN  
  Anzeige1:="Ein neuer Monat hat begonnen"  
ENDEWENN
```

SCHALTDAUER

Syntax :

SCHALTDAUER(*Objekt*)

Diese Funktion gibt die Zeit zurück, in der das Objekt sich in seinem aktuellen Zustand befindet.

Es wird die reine Uhrzeitdifferenz zwischen letzter Schaltzeit und aktueller Uhrzeit ermittelt ermittelt, Tage werden nicht berücksichtigt, d.h. als Ergebnis kann es keinen Wert grösser als 23:59:59 geben.

SONNENAUFGANG

Syntax :

SONNENAUFGANG

Mit dieser Funktion kann die Uhrzeit des Sonnenaufgangs für den aktuellen Tag ermittelt werden.

Beispiel:

```
wenn Uhrzeit = SONNENAUFGANG dann  
RollladenSchlafzimmer hochfahren  
endewenn
```

siehe auch [SONNENUNTERGANG](#)

SONNENUNTERGANG

Syntax :

SONNENUNTERGANG

Mit dieser Funktion kann die Uhrzeit des Sonnenuntergangs für den aktuellen Tag ermittelt werden.

Da diese Zeit vom geografischen Standort abhängig ist, kann auf der Seite *Einstellungen* das Postleitzahlgebiet angegeben werden. Durch Mausklick auf die Schaltfläche [Eingabe Koordinaten] können auch die exakten Koordinaten des Standorts eingegeben werden. Die Berechnung erfolgt durch astronomische Näherungsformeln, es wird die sogenannte „bürgerliche Dämmerung“ berechnet. Je nach Jahreszeit und geographischem Standort kann es zu einer Abweichung von einigen Minuten kommen.

Beispiel:

```
wenn Uhrzeit = SONNENUNTERGANG dann  
  RollladenSchlafzimmer runterfahren  
endewenn
```

STOPPUHR

Syntax:

STOPPUHR(Zeitvariable)

Diese Funktion gibt die Zeitdauerdifferenz zwischen der aktuellen Uhrzeit und der in der Zeitvariablen hinterlegten Zeit im Format HH:MM:SS zurück. Es wird die reine Uhrzeitdifferenz ermittelt, Tage werden nicht berücksichtigt, d.h. als Ergebnis kann es keinen Wert grösser als 23:59:59 geben.

Beispiel:

Das Treppenhauslicht soll ausgeschaltet werden, wenn es länger als 2 Minuten an ist und der Taster zum Einschalten länger als 1 Minute nicht mehr betätigt wurde. Beachten Sie, daß als Parameter bei der Ermittlung der Stopzeiten die Variable CT des Objekts und nicht das Objekt selbst verwendet wird.

```
wenn LiTreppenh eingeschaltet dann
  wenn Stoppuhr (LiTreppenh.ct) groesser "00:02:00" und
    Stoppuhr (TastTreppenh.ct) groesser "00:01:00" dann
      LiTreppenh aussschalten
    endewenn
  endewenn
```

Bitte beachten Sie:

Als Parameter darf nicht der Objektname, sondern es muss eine Zeitvariable oder die für jedes Objekt vorhandenen Zeitvariable CT (also *Objektname.CT*) angegeben werden. Es gibt auch noch eine andere Funktion um festzustellen wie lange ein Objekt sich in seinem aktuellen Zustand befindet, das ist die Funktion SCHALTDAUER(Objekt), bei dieser Funktion wird das Objekt (ohne den Zusatz.CT) als Parameter angegeben.

WOCHENTAG

Syntax :

WOCHENTAG

Diese Funktion gibt den aktuellen Wochentag als Text zurück. Bei Vergleichsbedingungen mit dieser Funktion muss die Groß/Kleinschreibung beachtet werden.

Beispiel:

WENN **WOCHENTAG** = "Montag" DANN

AKTTAG:=**WOCHENTAG**

ENDEWENN

UHRZEIT

Syntax :

UHRZEIT

Diese Funktion gibt die aktuelle Uhrzeit im Format HH:MM:SS zurück. Die Uhrzeit wird im 5-Sekunden-Takt aktualisiert, die Sekunden der Uhrzeit sind also immer durch 5 teilbar.

Beispiel:

WENN **UHRZEIT** groesser "23:00:00" DANN
AKTZEIT:=**UHRZEIT**
ENDEWENN

UHRZEITSEKUNDE

Syntax :

UHRZEITSEKUNDE

Diese Funktion gibt die aktuelle Uhrzeit im Format HH:MM:SS zurück. Die Sekunden sind aktuell und nicht wie in der Funktion UHRZEIT im 5-Sekundenintervall.

ZEIT

Syntax :

ZEIT

Diese Funktion ergibt die Zeit in einem internen Format. Dieses Format wird z.B. verwendet um die Stoppzeit zu ermitteln.

Beispiel:

Eine Zeitvariable kann mit der Funktion Zeit gefüllt werden. Die [Variable](#) UHR1 muss vom Typ *Zeit* sein.

Nachdem die Anweisung

UHR1:=ZEIT

ausgeführt wurde kann später mit der Funktion

STOPPUHR(UHR1)

die seit der Ausführung der ersten Anweisung vergangene Zeit ermittelt werden.

Die Anweisung UHR1:=ZEIT hat die gleiche Wirkung

wie die Anweisung STARTUHR(UHR1).

Weiterhin ist der Typ *Zeit* nützlich um die komplette Zeit (also Datum und Uhrzeit zu merken und auszugeben. Wenn eine Variable dieses Typs einem Zeichen-Objekt zugewiesen wird, werden Datum und Uhrzeit als Text ausgegeben.

ZUFALLSZEIT

Syntax :

ZUFALLSZEIT(Zeitvariable)

Generiert mit einem Zufallsgenerator eine Zeit bis maximal zum in der Zeitvariablen angegebenen Wert. Die Zeitvariable muss den Typ Uhrzeit haben (nicht den Typ Zeit, da dieser auch das Datum beinhaltet). Der maximal zulässige Wert ist "23:59:59". Es wird immer eine durch 5 teilbare Sekundenzahl zurückgegeben.

Beispiel:

Ein Programm zur Anwesenheitssimulation während des Urlaub soll das Licht zwischen 20 und 21 Uhr einschalten und zwischen 22 und 23:30 Uhr ausschalten.

Die Ein- und Ausschaltzeiten werden am besten von einem Programm, welches bei Tageswechsel läuft festgelegt.

Dies könnte so aussehen:

Einschaltzeit:="20:00:00"+**Zufallszeit("01:00:00")**

** (*generiert eine Zeit zwischen 20 und 21 Uhr*)

Ausschaltzeit:= "22:00:00"+**Zufallszeit("01:30:00")**

** (*generiert eine Zeit zwischen 22 und 23:30 Uhr*)

Verwendet werden können diese Zeiten dann in einem Makro um z.B. Beleuchtungen zu schalten. Ein solches Makro könnte so aussehen:

```
wenn Uhrzeit=Einschaltzeit dann
  Licht einschalten
endewenn
wenn Uhrzeit=Ausschaltzeit dann
  Licht ausschalten
endewenn
```

ERSETZEN

Syntax :

ERSETZEN(*Zeile,Alt,Neu*)

Diese Funktion ersetzt den Text des Parameters "Alt" in der Zeile durch den Text des Parameters "Neu". "Zeile" ist ein Objekt/eine Variable vom Typ Zeichen, "Alt" und "Neu" sind Variablen/Objekte vom Typ Zeichen oder Konstanten in Hochkommas.

Beispiel:

ZEILE:="Es ist jetzt dunkel im Zimmer"

ZEILE:=**ERSETZEN**(ZEILE,"dunkel","hell")

Nach Ausführung hat sich die Zeile geändert auf : "Es ist jetzt hell im Zimmer".

GROSSBUCHSTABEN

Syntax :

GROSSBUCHSTABEN(*Text*)

Diese Funktion wandelt alle Kleinbuchstaben des Parameters Text in Grossbuchstaben um. Text kann eine Variable oder Konstante vom Typ Zeichen sein.

Beispiel:

GROSSETEXT:=GROSSBUCHSTABEN(ZEILE)

KLEINBUCHSTABEN

Syntax :

KLEINBUCHSTABEN(*Text*)

Diese Funktion wandelt alle Grossbuchstaben des Parameters Text in Kleinbuchstaben um. Text kann eine Variable oder Konstante vom Typ Zeichen sein.

Beispiel:

KLEINERTEXT:=KLEINBUCHSTABEN(ZEILE)

LINKERTEIL

Syntax :

LINKERTEIL(*Text,Anzahl*)

Diese Funktion gibt die angegebene Anzahl Buchstaben der linken Seite des Textes zurück.
Text ist eine Variable des Typs Zeichen, Anzahl ist eine Variable des Typs Zahl ohne Kommastellen oder eine Konstante.

Beispiel:

ZEILE:="Das ist eine Zeile"

LINKS:=**LINKERTEIL**(ZEILE,3)

In der Textvariablen LINKS würde nach der Ausführung nur das Wort "Das" stehen.

RECHTERTEIL

Syntax :

RECHTERTEIL(*Text,Anzahl*)

Diese Funktion gibt die angegebene Anzahl Buchstaben der rechten Seite des Textes zurück.
Text ist eine Variable des Typs Zeichen, Anzahl ist eine Variable des Typs Zahl ohne Kommastellen oder eine Konstante.

Beispiel:

ZEILE:="Das ist eine Zeile"

RECHTS:=**RECHTERTEIL**(ZEILE,5)

In der Textvariablen RECHTS würde nach der Ausführung nur das Wort "Zeile" stehen.

TEXTPOSITION

Syntax :

TEXTPOSITION(*Textteil, Text*)

Diese Funktion gibt die Position eines Textteils innerhalb einer Textzeile zurück. Das Ergebnis ist vom Typ Zahl ohne Kommastellen.

Textteil ist ein Objekt/eine Variable vom Typ Zeichen oder eine Konstante in Hochkommas, Text eine Variable oder ein Objekt vom Typ Zeichen.

Beispiel:

ZEILE:="Wo ist denn das XY in dieser Zeile?"

XYPOS:=**TEXTPOSITION**(ZEILE,"XY")

In der Variablen XYPOS würde nach der Ausführung die Zahl 16 stehen, weil das "X" von "XY" an der 16. Stelle der Zeile steht.

TEXTTEIL

Syntax :

TEXTTEIL(Zeile,Pos,Anzahl)

Diese Funktion ermittelt einen Textabschnitt aus der Zeile, beginnend an der Stelle "Pos" mit der Anzahl im Parameter "Anzahl" angegebenen Buchstaben.

Zeile ein Objekt/eine Variable vom Typ Zeichen, "Pos" und "Anzahl" sind Variablen/Objekte vom Typ Zahl ohne Kommastellen oder Konstanten.

Beispiel:

```
ZEILE:="Das xyz wir herausgeholt"  
NEUERTEXT:=TEXTTEIL(ZEILE,5,3)
```

In der Variablen NEUERTEXT würde nach der Ausführung "xyz" stehen.

Hinweise zu Rollladen- und Markisensteuerungen

Die Laufzeiten für Rollladen- und Markisensteuerungen werden in der Konfiguration über die WEB-Oberfläche der Zentrale angegeben. Es sollte niemals eine kleinere Motorumschaltzeit als 0,5 Sekunden angegeben werden, da sonst die Gefahr besteht, dass Relais beschädigt werden.

Im Folgenden wird für Rollladen- und Markisensteuerungen nur der Begriff Rolllade verwendet.

Das Verstellen von Rollladen kann im Makro mit den Schlüsselwörtern **runterfahren** und **rauffahren** geschehen.

also z.B.

```
RollladeWohnen runterfahren
```

Um in Zwischenstellungen zu fahren, wird die jeweilige Zwischenstellung direkt als Wert zugewiesen.

also z.B.

```
RollladeWohnen:="halb"
```

Beachten Sie dabei bitte, dass die gewünschten Zwischenstellung in Hochkomma gesetzt werden und exakt dem Wort des Zustands entsprechen muss (auch Gross-/Kleinschreibung beachten). Die genauen Wörter für die jeweiligen Positionen werden auf der Seite "Visualisierung" der Objektdefinitionen angezeigt.

Die Abfrage ob eine Rolllade geöffnet oder geschlossen ist kann mit den Schlüsselwörtern **geoeffnet** und **geschlossen** vorgenommen werden.

also z.B.

```
wenn RollladeKueche geoeffnet dann
  AnzeigeRK:="offen"
endewenn
```

oder

```
wenn RollladeKueche geschlossen dann
  AnzeigeRK:="geschlossen"
endewenn
```

Die Abfrage von Zwischenpositionen wird wie die Zuweisungen von Zwischenpositionen mit der Bezeichnung der Zwischenstellung vorgenommen.

also z.B.

```
wenn RollladeKueche = "viertel" dann
  RollladeKueche:="dreiviertel"
endewenn
```

Für Rollladen gibt es eine zusätzliche spezielle Anweisung zum Stoppen eines Verstellvorgangs. Dies passiert mit der Anweisung

```
STOPPE (Rolladexx) .
```

Bitte beachten Sie:

Nach einem Fahrbefehl an eine Rolllade wird der neue Zustand in der Visualisierung für einen Sekundenbruchteil angezeigt, danach wird jeweils der Zustand angezeigt, der von dem Steuerungsmodul aktuell gesendet wird. Das ist normalerweise erst einmal der ursprüngliche Zustand, erst nach Abschluss der Verstellvorgangs wird der neue Zustand empfangen und angezeigt.

Hinweise zum Raumthermostat

Über eine kurze Betätigung der Menü-Taste kann der Modus eingestellt werden. Empfehlenswert ist der Modus "Manu". In diesem Modus kann der Raumthermostat über die Zentrale eingestellt werden, aber es sind auch amuelle Veränderungen der Temperatur am Raumthermostat möglich.

Damit ein Raumthermostat ausschliesslich über die Zentrale gesteuert wird, kann der Modus auf "Cent". Weitere Infos dazu finden Sie in der Bedienungsanleitung des Raumthermostats.

Zur Visualisierung des Raumthermostats wird ein spezielles Panel benutzt.

Die Temperatur auf der linken Seite des Panels ist die Solltemperatur, die über die Visualisierung verändert werden kann. Die Temperatur auf der rechten Seite des Panels ist die aktuelle Ist-Temperatur, die vom Raumthermostat empfangen wurde. In der Mitte wird die aktuelle gemessene Luftfeuchtigkeit angezeigt.

Die Darstellung von Luftfeuchtigkeit und Ist-Temperatur kann in der Definition des Raumthermostat-Panels ausgeschaltet werden.

Die Solltemperatur wird in den Makros direkt über den Namen des Reglers eingestellt bzw. abgefragt.

Beispiel:

RaumreglerWohnen setzen auf 21,5

oder

RaumreglerWohnen:=21,5

Es können auch die Werte **Off** und **On** über Makros im Raumthermostat gesetzt werden, indem für Off der Wert 0, für On der Wert 100 zugewiesen wird.

also z.B.

RaumreglerWohnen:=100

Der Modus **Off** schliesst das Ventil vollständig, der Modus **On** öffnet das Ventil vollständig. Bei beiden Modi wird die interne Zeittabelle des Raumtherostats deaktiviert.

Die Ist-Temperatur steht in Makros als Variable *Temperatur* des Raumthermostats zur Verfügung, die Luftfeuchtigkeit als Variable *Luftfeuchtigkeit*,

also z.B.

```
wenn Raumregler.Temperatur > 22.5 dann
```

```
.....
```

Wenn das Raumthermostat-Panel angeklickt wird, öffnet sich ein Fenster in dem die Solltemperatur verändert werden kann.

Bitte beachten Sie:

Die Raumthermostate sind nur im Abstand von einigen Minuten kommunikationsbereit, daher gibt es eine Verzögerung, bis geänderte Werte im Raumthermostat angezeigt werden.

Hinweise zur 19-Tasten-Fernbedienung

Die zwei oberen Tasten des Dreier-Blocks auf der rechten Seite können nicht durch das Programm benutzt werden, diese sind für interne Funktionen der Fernbedienung vorgesehen (siehe Bedienungsanleitung der Fernbedienung).

Das letzte Objekt der Fernbedienung (Standardname nach Modulimport .._18) ist ein Objekt vom Typ Zeichen, über das Text auf der Fernbedienung ausgegeben werden kann. Zur Ausgabe von Text wird dieser einfach dem Objekt zugewiesen, also z.B.:

```
AnzeigeRC19:="Alarm"
```

Weiterhin können die einzelnen Icons der Fernbedienung über Variablen dieses Objekts aktiviert werden.

Folgende Variablen für die einzelnen Icons stehen zur Verfügung:

PfeilUnten, PfeilOben, Glocke, Rollo, Licht, Uhr, Tür, Telefon, Szene, Schalter, Fenster.

Um die entsprechenden Icons anzuzeigen werden die Variablen "eingeschaltet". Also z.B.:

```
AnzeigeRC19.Glocke einschalten
```

```
AnzeigeRC19.Licht einschalten
```

bewirken, dass die Icons "Glocke" und "Glühbirne" angezeigt werden.

Ausser den Variablen für die Icons gibt es noch weitere Variablen um Informationen auf der Fernbedienung auszugeben.

Die Variable **Einheit** kann mit Werten von 0 bis 4 belegt werden und bewirkt die Anzeige einer der folgenden Einheiten:

0 = Keine Einheit

1 = Prozent

2 = Watt

3 = Celsius

4 = Fahrenheit

Mit der Variablen **Beleuchtung** kann die Hintergrundbeleuchtung gesteuert werden.

0 = Keine Hintergrundbeleuchtung

1 = Hintergrundbeleuchtung einschalten

2 = Langsames Blinken

3 = Schnelles Blinken

Mit der Variablen **Beep** kann die Ausgabe eines akustischen Signals aktiviert werden.

0 = Keine Ton

1 = Ton 1

2 = Ton 2

3 = Ton 3

Die Ausgabe auf die Fernbedienung erfolgt nicht automatisch nach Zuweisung von Werten sondern mit einem speziellen Ausgabebefehl nachdem alle gewünschten Werte gesetzt wurden. Dieser Befehl lautet:

Anzeigen(Objektname)

Die Ausgabe wird dann einige Zeit auf der Fernbedienung angezeigt. Soll die Ausgabe nochmals angezeigt werden, so muss die Ausgabeanweisung erneut ausgeführt werden.

Alle Variablen für die Icon-Darstellung auf der Fernbedienung können gelöscht werden mit der Anweisung

LöscheAnzeige(Objektname)

Dies erspart das Löschen der einzelnen Variablen, wenn die Ausgabe an die Fernbedienung geändert werden soll. Diese Anweisung bewirkt keine Aktualisierung der Anzeige des Moduls, die Aktualisierung muss jeweils mit der Anweisung **Anzeigen(...)** vorgenommen werden.

Bildsymbole auswählen und bearbeiten

In diesem Fenster kann ein Bild-Symbol für das aktuelle Objekt ausgewählt werden.

Beim Öffnen des Fensters wird eine Auswahl von Bild-Symbolen angezeigt, die am ehesten auf das Objekt zutreffen. Mit der Schaltfläche **[Mehr anzeigen]** werden weitere im Programm zur Verfügung stehende Bild-Symbole angezeigt, unabhängig davon ob für das aktuelle Objekt zutreffen oder sinnvoll sind.

Mit der Schaltfläche **[Bild ändern]** können beliebige [Bitmaps](#) zur Darstellung des Objekts bzw. des Objektzustands eingebunden werden.

Solche Bitmaps können auch selbst erstellt werden. Das kann mit z.B. mit dem standardmässig in jeder Windows-Version vorhandenem Programm MS-Paint/Paintbrush geschehen (unter Zubehör).

Mit der Schaltfläche **[Neuer Darstellungstyp]** können neue Darstellungstypen erstellt werden. Dazu wird die aktuell ausgewählte Objektdarstellung inklusive der Zustandsarten, aber ohne die zugehörigen Bilder kopiert. Als Bild wird für jeden Zustand ein Testbild eingesetzt. Dieses kann dann durch eigene Bitmaps ersetzt werden.

Ein solcher selbst definierter Darstellungstyp steht auch für andere Objekte des Projekts zur Verfügung.

Mit der Schaltfläche **[Darstellungstyp löschen]** können selbst erstellte Darstellungstypen wieder gelöscht werden, jedoch nicht die standardmässig im Programm angebotenen Darstellungstypen.

Objektrahmen und Textrahmen

Objektrahmen und Textrahmen dienen dazu ein Objekt und seinen Zustand oder Wert auf dem Bildschirm darzustellen. In einem Objektrahmen wird neben der Ausgabe der Objektbezeichnung und dem aktuelle Zustand als Text im rechten Teil auch ein Symbol angezeigt, wenn für dieses Objekt ein Symbol ausgewählt wurde. Durch Anklicken eines Objektrahmens oder Textrahmens kann der Zustand des Objekt verändert werden.

Die Grösse und Position für Objektrahmen und Textrahmen der einzelnen [Objekte](#) kann im [Entwurfswindow](#) festgelegt werden. Dieses Entwurfswindow wird mit dem Knopf **[Ansicht bearbeiten]** auf der [Seite Ansichten](#) des [Einstellungsfensters](#) geöffnet.

Balkendarstellung

Die Balkendarstellung eignet sich für Objekte des Typs Zahl, also z.B. Temperatur, Luftfeuchtigkeit, Windgeschwindigkeit usw.

Die Balkendarstellung kann horizontal (Standard-Einstellung) oder vertikal erfolgen.

Bei vertikaler Darstellung wird die Bezeichnung und der Wert nur dann mit ausgegeben wenn er nicht breiter ist als die Balkendarstellung, ansonsten werden Wert und Bezeichnung nicht angezeigt.

Sprachausgabe

Mit der Sprachausgabe-Funktion kann Text als Sprache ausgegeben werden. Es ist damit möglich in einem Makro eine Zeichen-Variable oder einen Text in Hochkommas als Sprache auszugeben.

Die Ausgabe erfolgt über das Visualisierungsprogramm [VisuWin](#), dieses muss auf einem PC aktiv sein, damit die Sprache auf diesem PC ausgegeben werden kann.

Hardware-Voraussetzung ist natürlich eine Soundkarte.

Zur Sprachausgabe werden die Microsoft-Sprachfunktionen der SAPI 5.1 (SAPI=Speech Application Programm Interface) benutzt.

Damit die Sprachfunktionen genutzt werden können, muss dieses Feature auf Ihrem Betriebssystem installiert sein.

Bei Windows XP ist das normalerweise standardmässig der Fall, bei anderen Windows-Versionen muss die SAPI in der Regel zusätzlich installiert werden.

Für XP ist normalerweise nur eine englische Stimme (Microsoft Sam) installiert (Schauen Sie einfach nach unter: Systemsteuerung->Sprachein-/ausgabe).

Um die englischen Stimmen der Sprachausgabefunktion mit besserer landesprachlicher Aussprache auszugeben, kann ein Sternchen vor den auszugebenden Text gesetzt werden, die beste Qualität erreicht man aber mit den landespezifischen Stimmen. Diese stehen leider nicht separat, sondern nur als Zusatz zu anderen Microsoft Programmen (z.B. Microsoft Reader, MS Autoroute 2007) zur Verfügung.

Informationen bzw. entsprechende Links zu Installationsdateien für weitere (englische) XP-Stimmen und die SAPI 5 für andere Windows-Versionen als XP sind auf den Internetseiten von Microsoft zu finden.

Aktuelle Informationen und Updates zum Programm finden Sie im Internet unter:
www.contronics.de/html/homematic.html