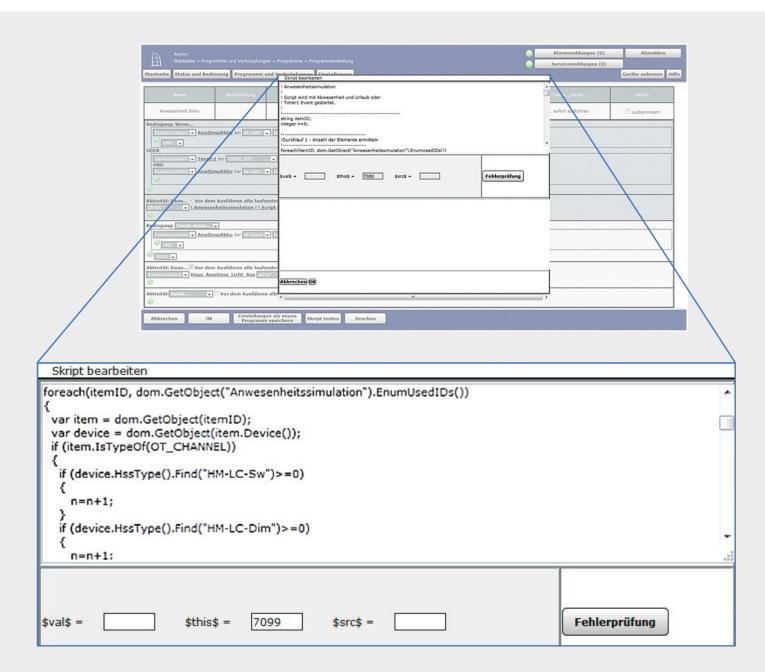
Homematic Scriptprogrammierung

Teil 7 - Kommunikation mit mediola und weitere Scripte



Im siebten Teil der Artikelserie beschäftigen wir uns mit der Scriptprogrammierung im Zusammenhang mit dem mediola Gateway und dem mediola CREATOR und wir schreiben weitere Scripte.

Der mediola CREATOR

Mit dem mediola AIO CREATOR bzw. dem neuen mediola AIO CREATOR NEO kann man sich eine ganz persönliche Steuerungs-App erstellen, mit welcher Homematic Komponenten wie auch Komponenten anderer Hersteller mit dem Smartphone oder dem Tablet bedient werden können.

Die AIO CREATOR (NEO) Software ist dabei die zentrale Komponente, mit der individuelle Fernbedienungsvisualisierungen für Smartphones und Tablets erstellt, administriert und angepasst werden können.

Im Teil 6 der Artikelserie haben wir ein Script vorgestellt, mit dem Heizkörperventilen die Daten für den Urlaubsbetrieb übermittelt werden können. Mit einer Fernbedienungsseite auf einem Tablet oder einem Smartphone können wir nun mithilfe des AIO CREATOR eine komfortable Eingabemöglichkeit für diese Daten schaffen.

Auf die Bedienung des AIO CREATOR soll hier nicht näher eingegangen werden, hierzu geben die Anleitungen zu diesem System eine qute Hilfestellung.

Zunächst wollen wir die Eingabeseite erstellen (es muss jeweils für den Start des Urlaubs als auch für das Urlaubsende ein Datum in der Form Tag, Monat und Jahr definiert werden).

Hier ein Beispiel für eine Datums-Eingabeseite auf einem Pad, die mit dem AIO CREATOR erstellt wurde:



Im Teil 6 der Artikelserie wurden folgende Variablen zum Setzen von Start- und Endedatum des Urlaubs verwendet:

Variable	Тур	Bedeutung
Urlaub_S_T	Zahl	Startdatum Urlaub Tag (031)
Urlaub_S_M	Zahl	Startdatum Urlaub Monat (012)
Urlaub_S_J	Zahl	Startdatum Urlaub Jahr (4-stellig)
Urlaub_E_T	Zahl	Endedatum Urlaub Tag (031)
Urlaub_E_M	Zahl	Endedatum Urlaub Monat (012)
Urlaub_E_J	Zahl	Endedatum Urlaub Jahr (4-stellig)
Temp_Urlaub	Zahl	Temperatur im Haus für die Urlaubszeit [°C]

Mit der Eingabeseite des AIO CREATOR werden nun die System-(Zentralen-)Variablen *Urlaub_S_T*, *Urlaub_S_M* sowie *Urlaub_S_J* mit den gewünschten Werten beschrieben und zwar:

Urlaub_S_T durch Drücken auf eine der Tasten 1 ... 31

Urlaub_S_M durch Verändern des Monats mit den UP/DN-Tasten (Up/Down)

Urlaub S J durch Verändern des Jahrs mit den UP/DN-Tasten (Up/Down)

Die Variablen *Urlaub_E_T*, *Urlaub_E_M* sowie *Urlaub_E_J* für das Endedatum des Urlaubs werden über eine zweite Eingabeseite beschrieben.

Auf diese Art müsste für jede Anwendung, die ein Datum benötigt, eine weitere Eingabeseite erzeugt werden, die dann mit den entsprechenden Systemvariablen verbunden wird. Dieses "Erzeugen" geschieht zwar einfach durch Kopieren der ganzen Eingabeseite, allerdings müssen dann alle 35 Tasten mit neuen zugehörigen Systemvariablen verbunden werden.

Es gibt aber auch eine einfache Möglichkeit, mit Kopien der Seite zu arbeiten, ohne für jede neue Datums-Auswahlseite den 35 Tasten der Mediola-Seiten neue Systemvariablen zuordnen zu müssen:

Die Eingabeseite beschreibt grundsätzlich immer die gleichen (temporären) Systemvariablen. Die Rücktaste der Eingabeseite wird nun aber nicht alleine mit der Funktion "zurück zur aufrufenden Seite" belegt, sondern mit einem Makro.



Ein Makro ist z. B. eine Liste (Zusammenstellung) von mehreren Befehlen, die sequenziell abgearbeitet werden. Und in unserem Beispiel wird beim Drücken der "Zurück"-Taste zunächst eine weitere Systemvariable mit Zahlen 1,2,3 ... belegt, wobei die Zahl einen festen Bezug zur Seite und damit zur Funktion der Seite hat. Damit kann ein eindeutiger Bezug zu unterschiedlichen (Zentralen-)Variablensätzen in der CCU hergestellt werden.

Danach wird mit einem zweiten Befehl im Makro der Rücksprung ausgeführt.

Die Zahlen haben in unserem Beispiel dann die Werte:

- 1 = Startdatum Urlaub setzen
- 2 = Endedatum Urlaub setzen

In den folgenden Scripten wurden diese Zentralenvariablen verwendet:

Zentralenvariable	Verbindung zur mediola Oberfläche	Funktion	Bemerkung
Kal_Ausw_Tag	Taste 1–31	Beinhaltet den Tag des gewünschten Datums	
Kal_Ausw_Monat	Über die Tasten Up und Down, die Zen- tralenvariablen Kal_Ausw_Mon_Up und Kal_Ausw_Mon_Dn und dann das Script	Beinhaltet den Monat des gewünschten Datums	
Kal_Ausw_Jahr	Über die Tasten Up und Down, die Zen- tralenvariablen Kal_Ausw_Jahr_Up und Kal_Ausw_Jahr_Dn und dann das Script	Beinhaltet das Jahr des gewünschten Datums	
Kal_Ausw_Mon_Up		Wird true, wenn die "Monat_Up"-Taste auf der Auswahlseite gedrückt wurde	Wird vom Script in der CCU wieder false gesetzt!
Kal_Ausw_Mon_Dn		Wird true, wenn die "Monat_Dn"-Taste auf der Auswahlseite gedrückt wurde	Wird vom Script in der CCU wieder false gesetzt!
Kal_Ausw_Jahr_Up		Wird true, wenn die "Jahr_Up"-Taste auf der Auswahlseite gedrückt wurde	Wird vom Script in der CCU wieder false gesetzt!
Kal_Ausw_Jahr_Dn		Wird true, wenn die "Jahr_Dn"-Taste auf der Auswahlseite gedrückt wurde	Wird vom Script in der CCU wieder false gesetzt!
Kal_Uebernahme		Bestimmt, welchen Zentralenvariablen der Datumswert zugeordnet wird	1 = Startdatum 2 = Endedatum

Auf der Seite der CCU sind nun die folgenden Scripte und ein aufrufendes Zentralenprogramm notwendig:

Das erste Script setzt den Übernahmewert für den Monat.

Sobald die "Monat_Up"-Taste oder die "Monat_Dn"-Taste auf der Auswahlseite der Fernbedienung gedrückt wurde, wird das folgende Script aufgerufen. Es erhöht bzw. vermindert den Wert der Zentralenvariablen Kal_Ausw_Monat bei jedem Tastendruck um den Wert 1 (Zeile 7 bzw. 19). Wenn der Wert von Kal_Ausw_Monat 12 beträgt, wird beim Drücken der Up-Taste der Wert auf 1 gesetzt, beträgt er 1, wird er beim Drücken der Dn-Taste auf 12 gesetzt (Zeile 11 bzw. 23).

01	!Kalendermonat einstellen
02	!
03	<pre>if (dom.GetObject("Kal_Ausw_Mon_Up").State() == true)</pre>
04	{
0.5	<pre>if (dom.GetObject("Kal_Ausw_Monat").State()<12)</pre>
06	{
07	dom.GetObject("Kal_Ausw_Monat").State(dom.GetObject("Kal_Ausw_Monat").Stat
	e()+1);
08	}
09	else
10	{
11	<pre>dom.GetObject("Kal_Ausw_Monat").State(1);</pre>
12	}
13	<pre>dom.GetObject("Kal_Ausw_Mon_Up").State(false);</pre>
14	}
15	<pre>if (dom.GetObject("Kal_Ausw_Mon_Dn").State() == true)</pre>
16	{
17	<pre>if (dom.GetObject("Kal_Ausw_Monat").State()>1)</pre>
18	{
19	dom.GetObject("Kal_Ausw_Monat").State(dom.GetObject("Kal_Ausw_Monat").Stat
	e()-1);
20	}



```
else
22
23
            dom.GetObject("Kal_Ausw_Monat").State(12);
24
        dom.GetObject("Kal_Ausw_Mon_Dn").State(false);
25
26
27
    if (dom.GetObject("Kal_Ausw_Monat").State()==1)
     {dom.GetObject("Kal_Ausw_Monat_Text").State("Januar")};
28
    if (dom.GetObject("Kal_Ausw_Monat").State()==2)
    {dom.GetObject("Kal_Ausw_Monat_Text").State("Februar")};
    if (dom.GetObject("Kal_Ausw_Monat").State()==3)
29
    {dom.GetObject("Kal_Ausw_Monat_Text").State("März")};
    if (dom.GetObject("Kal_Ausw_Monat").State()==4)
30
    {dom.GetObject("Kal_Ausw_Monat_Text").State("April")};
31
    if (dom.GetObject("Kal_Ausw_Monat").State() == 5)
    {dom.GetObject("Kal_Ausw_Monat_Text").State("Juni")};
32
    if (dom.GetObject("Kal_Ausw_Monat").State()==7)
    {dom.GetObject("Kal_Ausw_Monat_Text").State("Juli")};
33
    if (dom.GetObject("Kal_Ausw_Monat").State() == 8)
    {dom.GetObject("Kal_Ausw_Monat_Text").State("August")};
34
    if (dom.GetObject("Kal_Ausw_Monat").State()==9)
    {dom.GetObject("Kal_Ausw_Monat_Text").State("September")};
35
    if (dom.GetObject("Kal_Ausw_Monat").State() == 10)
    {dom.GetObject("Kal_Ausw_Monat_Text").State("Oktober")};
36
    if (dom.GetObject("Kal_Ausw_Monat").State()==11)
    {dom.GetObject("Kal_Ausw_Monat_Text").State("November")};
37
    if (dom.GetObject("Kal_Ausw_Monat").State()==12)
    {dom.GetObject("Kal_Ausw_Monat_Text").State("Dezember")};
```

Die Zeilen 27 bis 37 beschreiben die Zentralenvariable *Kal_Ausw_Monat_Text* mit dem Monatsnamen, entsprechend dem Wert von *Kal_Ausw_Monat* (1...12) zur freien Verwendung.

```
Die Zeile 7 (bzw. 19) inkrementiert (bzw. dekrementiert) den Zählerstand: dom.GetObject("Kal_Ausw_Monat").State(dom.GetObject("Kal_Ausw_Monat").State()+1);
```

```
Zum besseren Verständnis hier die Rechnung in einer anderen Schreibweise:
var Wert_Aktuell = dom.GetObject("Kal_Ausw_Monat").State();
var Wert_Neu = Wert_Aktuell + 1;
dom.GetObject("Kal Ausw Monat").State(Wert Neu);
```

Wir benötigen in dieser Schreibweise zwei zusätzliche (Programm-)Variablen, es werden statt einer Zeile drei benötigt.

Script Nummer 2 überträgt – je nach Wert der Zentralenvariable *Kal_Uebernahme* – die gewählten Werte in die Variablen für den Start Urlaub oder das Ende Urlaub.

01	!Kalenderuebernahme Start Urlaub		
02	!		
03	if (dom.GetObject("Kal_Uebernahme").State()==1)		
04	{		
05	dom.GetObject("Urlaub_S_T").State(dom.GetObject("Kal_Ausw_Tag").Sta		
	te());		
06	<pre>dom.GetObject("Urlaub_S_M").State(dom.GetObject("Kal_Ausw_Monat").S</pre>		
	tate());		
07	<pre>dom.GetObject("Urlaub_S_J").State(dom.GetObject("Kal_Ausw_Jahr").St</pre>		
	ate());		
08	}		
09			
10	!Kalenderuebernahme EndeUrlaub		
11	!		
12	<pre>if (dom.GetObject("Kal_Uebernahme").State() == 2)</pre>		
13	{		
14	<pre>dom.GetObject("Urlaub_E_T").State(dom.GetObject("Kal_Ausw_Tag").Sta</pre>		
	te());		



15	dom.GetObject("Urlaub_E_M").State(dom.GetObject("Kal_Ausw_Monat").S
	tate());
16	dom.GetObject("Urlaub_E_J").State(dom.GetObject("Kal_Ausw_Jahr").St
	ate());
17	}
18	<pre>dom.GetObject("Kal Uebernahme").State(0);</pre>

In Zeile 18 wird dann abschließend die Zentralenvariable *Kal_Uebernahme* wieder zurückgesetzt → (auf den Wert 0 gesetzt).

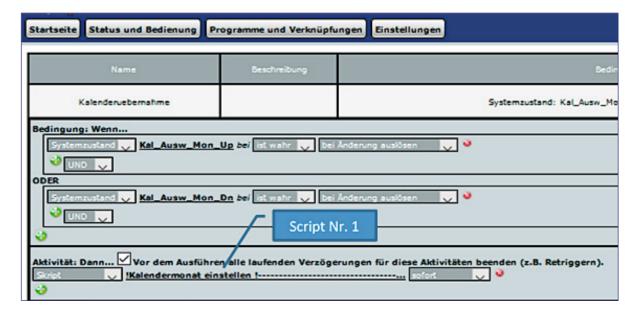
Das dritte Script setzt den Übernahmewert für das Jahr.

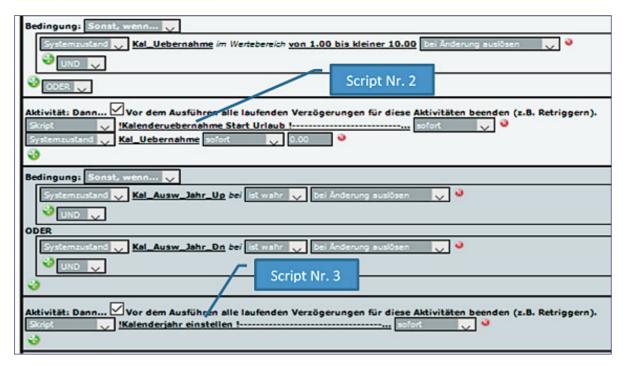
Sobald die "Jahr_Up"-Taste oder die "Jahr_Dn"-Taste auf der Auswahlseite der Fernbedienung gedrückt wurde, wird das folgende Script aufgerufen. Es erhöht bzw. vermindert den Wert der Zentralenvariablen *Kal_Ausw_Jahr* bei jedem Tastendruck um den Wert 1 (Zeile 7 bzw. 19). Wenn der Wert von Kal_Ausw_Jahr 2200 beträgt, wird beim Drücken der Up-Taste der Wert auf 1 gesetzt, beträgt er 1, wird er beim Drücken der Dn-Taste auf 2200 gesetzt (Zeile 11 bzw. 24).

```
!Kalenderjahr einstellen
0.2
     if (dom.GetObject("Kal_Ausw_Jahr_Up").State()==true)
03
04
        if (dom.GetObject("Kal_Ausw_Jahr").State()<5555)
05
06
07
          dom.GetObject("Kal Ausw Jahr").State(dom.GetObject("Kal Ausw Jahr").State()+1);
80
       else
09
10
          dom.GetObject("Kal_Ausw_Jahr").State(1);
12
        dom.GetObject("Kal_Ausw_Jahr_Up").State(false);
13
14
15
     if (dom.GetObject("Kal_Ausw_Jahr_Dn").State()==true)
16
17
18
        if (dom.GetObject("Kal_Ausw_Jahr").State()>1)
19
20
          dom.GetObject("Kal_Ausw_Jahr").State(dom.GetObject("Kal_Ausw_Jahr").State()-1);
21
22
       else
23
          dom.GetObject("Kal_Ausw_Jahr").State(2015);
25
26
        dom.GetObject("Kal_Ausw_Jahr_Dn").State(false);
```

Die Zentralenvariablen Kal_Ausw_Jahr_Dn und Kal_Ausw_Jahr_Up werden am Ende des Scripts jeweils zurückgesetzt (auf den Wert false gesetzt).

Das aufrufende Zentralenprogramm sieht folgendermaßen aus:



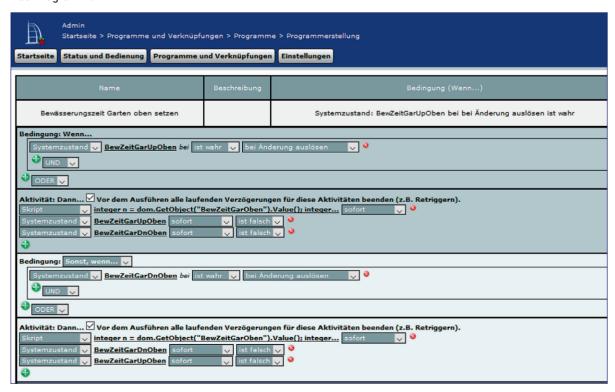


Einschaltzeit verändern.

In diesem Zusammenhang gleich noch ein ähnliches Beispiel:

Die Einschaltdauer eines Aktors soll mit einer Up- und einer Down-Taste einer mediola Fernbedienungsoberfläche verändert werden. Nehmen wir an, diese Einschaltdauer ist die Einschaltdauer einer Gartenbewässerung und ist in der Zentralenvariablen BewZeitGarOben gespeichert. Für die beiden Taster (Up und Down) benötigen wir noch zwei boolesche Variablen: BewZeitGarUpOben und BewZeitGarDnOben. Mit diesen beiden Variablen wird ein Script (Up-/Down-Behandlung in einem Script, wie im Beispiel vorher beschrieben) oder es werden zwei Scripte (wie in diesem Beispiel) getriggert.

Das Programm:





Folgende Variablen (Zentralenvariablen) werden verwendet:

Zentralenvariablenname	Funktion	Einheit	Тур
BewZeitGarOben	Bewässerungszeit Garten oberer Teil	[min]	Wert
BewZeitGarObenMax	Bewässerungszeit Garten oberer Teil Maximalwert	[min]	Wert
BewZeitGarUpOben	Von der mediola Taste Up		Boolean
BewZeitGarDnOben	Von der mediola Taste Down		Boolean

Weiterhin wird ein Aktor verwendet, der die Bewässerung einschalten soll, er hat beispielsweise den Namen "AktorGarBewOben".

Das Script (Wert vermindern):

01	<pre>integer n = dom.GetObject("BewZeitGarOben").Value();</pre>
02	<pre>integer max = dom.GetObject("BewZeitGarObenMax").Value();</pre>
03	!
04	!Zeit incrementieren
0.5	!
06	n = n + 1;
07	if $(n > max) \{n = 1; \}$
8.0	!
09	!Variable aktualisieren
10	!
11	<pre>dom.GetObject("BewZeitGarOben").State(n);</pre>
12	
13	!
14	!Zeit setzen
15	!
16	<pre>object aktor = dom.GetObject("AktorGarBewOben:1");</pre>
17	<pre>aktor.DPByHssDP("ON_TIME").State(60.0 * n);</pre>

Das zweite Script zum Dekrementieren des Werts entspricht obigem Script, bis auf die Zeilen 06 und 07:

06	n = n - 1;	
07	if $(n < 1) \{n = max; \}$	

In der nächsten Folge beschäftigen wir uns mit Informationen, die über eine mediola Visualisierung sehr komfortabel dargestellt werden können, und natürlich mit den dafür notwendigen Scripten.

Sehr geehrter Leser,

bei diesem Artikel zur Scriptprogrammierung handelt es sich um einen Fachbeitrag eines erfahrenen Homematic Users und Autors. Die ELV-/eQ-3-Unternehmensgruppe selbst nutzt die Möglichkeiten dieser Schnittstelle nicht, möchte aber den Anwendern der CCU2 den Zugang zu dieser Schnittstelle nicht verwehren.

Sollten Sie Schwierigkeiten bei der Verwendung dieser zusätzlichen Programmiermöglichkeit der CCU2 haben, so haben Sie bitte Verständnis dafür, dass wir Ihnen hierzu leider keinen Support geben können. In den entsprechenden Foren und Internet-Plattformen rund um das Thema "Programmierung Homematic CCU" finden Sie jedoch sicherlich im Bedarfsfall die notwendigen Anregungen und Hilfestellungen für Ihr Projekt.

Mögliche Quellen im Internet:

https://www.homematic-inside.de/software/download/item/homematic-skript https://homematic-forum.de/forum/viewtopic.php?f=19&t=18692