

# KNX APP-Control Server

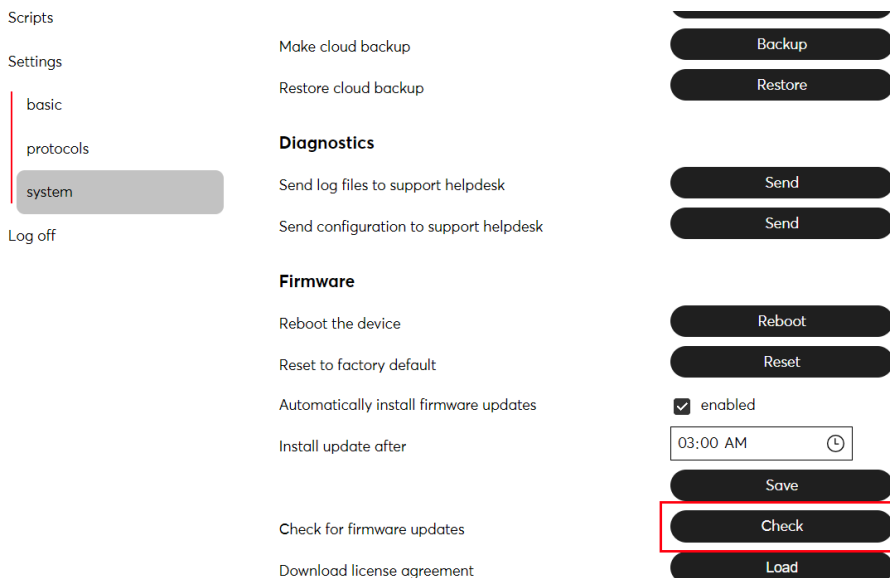
## New Features

### Contents

<b>1. Purpose of this document .....</b>	<b>2</b>
<b>2. New App UI Design.....</b>	<b>2</b>
<b>3. Smart Energy Management .....</b>	<b>3</b>
3.1. Energy Provider .....	3
3.2. Available Energy Sources .....	4
3.2.1. Solar panels .....	4
3.2.2. Wind turbines.....	4
3.2.3. Other sources.....	4
3.3. Energy Storage.....	4
3.4. Power Consumers.....	5
3.5. Scheduling Options.....	5
3.6. Shutdown Options .....	5
3.7. Priority Options.....	6
3.8. Putting the Smart Energy Management into Operation .....	6
3.9. Smart Energy Management on the KNX APP-Control Server Controller .....	6
3.10. Smart Energy Management in the KNX APP-Control Server App .....	7
<b>4. KNX programming via Remote Access Service.....</b>	<b>7</b>
<b>5. Modbus Client .....</b>	<b>8</b>
<b>6. BACnet Server .....</b>	<b>9</b>
<b>7. KNX Recorder .....</b>	<b>9</b>
<b>8. Support for Dahua and Hikvision IP Cameras, including PTZ Control .....</b>	<b>10</b>
<b>9. Weather Forecast Information .....</b>	<b>10</b>
<b>10. New Trigger Options .....</b>	<b>10</b>
<b>11. Include LUA Scripts as Library in other LUA Scripts .....</b>	<b>10</b>
<b>12. KNX IP Router .....</b>	<b>12</b>
<b>13. KNX IP Secure Routing.....</b>	<b>12</b>
<b>14. KNX IP Secure .....</b>	<b>13</b>
<b>15. Philips Hue API Extension .....</b>	<b>13</b>
<b>16. New LUS Functions .....</b>	<b>13</b>
<b>17. RTSP Support for Cameras .....</b>	<b>14</b>
<b>18. New Firmware Core.....</b>	<b>14</b>
<b>19. OCPP Car Charger .....</b>	<b>14</b>
<b>20. Logic Module .....</b>	<b>16</b>
<b>21. Miscellaneous .....</b>	<b>18</b>

# 1. Purpose of this document

This document describes the new features of the KNX APP-Control Server (CT/S2.1). The new features are provided free-of-charge with firmware V2.0. The firmware update can be done directly in the web browser of the KNX APP-Control Server.

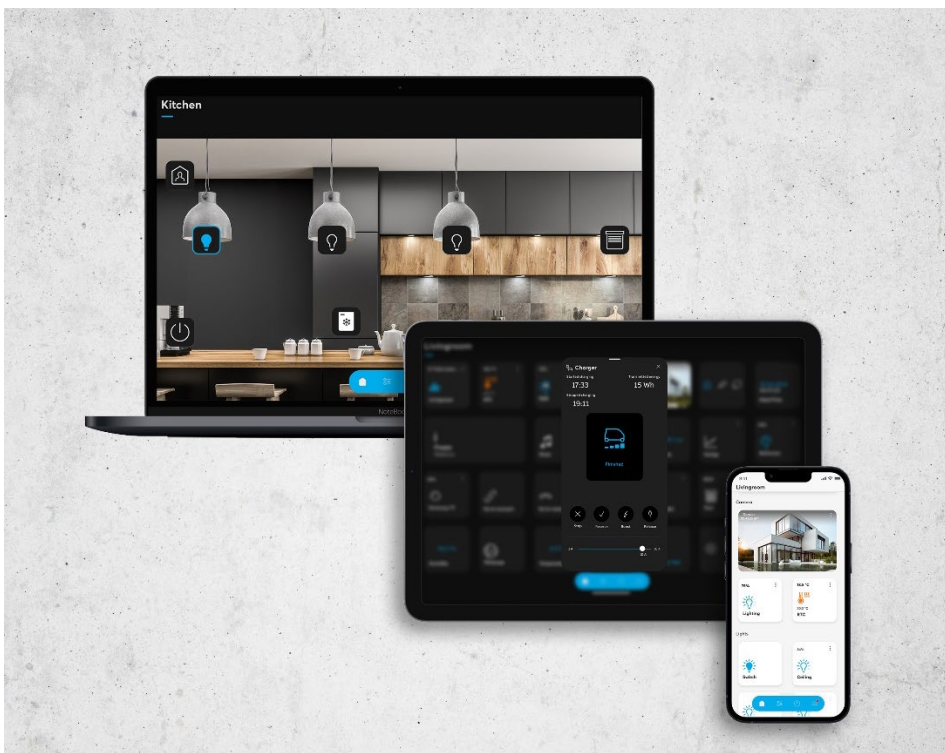


Further product details and documents are available on the product detail page:

<https://new.abb.com/products/2CKA006136A0218/ct-s2-1>

# 2. New App UI Design

The KNX APP-Control Server app has a new modern design which is in line with ABB's touch panels (SmartTouch®, RoomTouch®). Background images or floor plans can be uploaded. Possibility to switch between light and dark mode.



## 3. Smart Energy Management

With Smart Energy Management in the KNX APP-Control Server, you can optimize your power usage to be more self-sufficient and lower your energy bills. You can use the APP-Control Server to schedule energy consuming equipment in a balanced way based on available energy, current pricing and energy demand. To make this possible, the APP-Control Server assumes the capacity and current energy consumption of your main connection, along with the pricing of your energy supplier. To determine your own energy production with solar panels or a wind turbine, for example, the optimal capacity is corrected with the weather forecast to predict a realistic yield. It is additionally possible to include a battery, in which surplus energy can be stored to be used again when you need it. To plan efficiently, each energy user's energy consumption, run time, schedule and priority is determined.

The end user can dynamically add or remove consumers from the schedule at any time, after which the APP-Control Server will find the optimal time of day while guaranteeing that they are available when you need them. Of course, through the APP-Control Server app, you always have a complete and clear overview of how your energy is managed.

For example, when you get home from work, you can schedule the charging of your car where it has to be fully charged no later than 08:00 AM the next morning. During the evening, you add the washing machine and dishwasher to the program, with the requirement that they must be running by 5:00 PM the following evening. The APP-Control Server will then determine the optimal time for charging and running the appliances. This is preferably done when your solar panels are generating enough energy, or else when energy prices are most favorable. Depending on the available energy, other energy use and the power requirement for the scheduled equipment, the APP-Control Server will run multiple consumers at the same time or schedule them one after the other.

The following sections will guide you through all the steps to set up and use Smart Energy Management in the APP-Control Server.

### 3.1. Energy Provider

There should always be one energy provider for the APP-Control Server, with which the characteristics of your supplier's power grid are determined.

You have the following settings for this:

- Tariff, where you have the options of a single tariff or double tariff (with fixed schedule) or a variable tariff that uses spot prices, which can vary hourly. For using spot prices, your region is important, and any price corrections from your supplier. Here you can also specify whether the tariff should be displayed in charts. For use in actions, you can specify what the limits are when the rate should be considered "high" or "low," as a percentage.
- Type of power connection, namely 1 phase or 3 phases, and the voltage.
- For each phase, the maximum power of the network, the maximum power that the APP-Control Server may allocate to energy consumers and the expected average continuous consumption (not managed by the APP-Control Server). In addition, for each phase an object must be linked with the current consumption. Here it is important to know whether this is the full consumption or the net consumption, from which any produced power has already been removed. It is also possible to set whether consumers managed by Smart Energy Management (also known as "end-points") may be switched off when the total consumption becomes too high.

## 3.2. Available Energy Sources

For each energy source, the type must be specified. Here you have a choice between solar panels, wind turbines and other.

### 3.2.1. Solar panels

For solar panels the maximum (realistic) production must be specified on December 21 and June 21. In addition, you can specify whether the APP-Control Server should correct the yield on the east and west side (for example, when the panels are not exactly facing south) and whether the APP-Control Server should consider the weather forecast to determine the energy yield. Furthermore, an object should be linked that indicates the current energy production and for which phase (if any).

For each day of the year, the APP-Control Server will calculate how much energy would be generated if it were cloudless. The expected production is then corrected based on the weather forecast. A cloud factor of 100% means that if it is completely cloudy, the expected energy production is reduced to 0.

### 3.2.2. Wind turbines

For a wind turbine, the maximum (realistic) production at the optimal wind speed must be specified. A reduction can then be specified for each wind direction, such as when there is a tree or building in a direction that affects the maximum output of the turbine. Furthermore, an object should be linked that indicates the current energy production and for which phase (if any).

Based on the predicted wind speed and direction, the APP-Control Server will calculate how much energy should be generated.

### 3.2.3. Other sources

For other energy sources, you can specify how much energy is produced and link an object that indicates the current energy production and for which phase (if applicable).

## 3.3. Energy Storage

Energy storage (a battery) allows you to store energy surplus and use it later when you need it. Each battery has a name, a maximum capacity, a maximum charge current and a maximum discharge current. In addition, the following parameters are required:

- Recalculation interval (between 5 and 30 minutes). This period is used to recalculate whether to start or stop charging or discharging. This prevents the battery from having to switch over in short succession. Note that if you have multiple batteries, the shortest specified interval is used for all batteries.
- Battery charging, where you can determine whether the battery should be charged only when there is excess power, or also when there is a low rate (in the 30% cheapest bandwidth)
- Use battery, where you can determine when the battery can be used. This can be always when power is needed, only when the tariff is "not low" (not in the 30% cheapest bandwidth) or only when the tariff is high (in the 30% most expensive bandwidth).
- If the installation has 3 phases, it should be specified which phase the battery is connected to or whether it is a 3-phase connection.

In addition to these configuration options, objects must be linked for status information on current capacity, charge current and discharge current. To control the battery, objects must

be linked to turn charging and discharging on/off and the request amount of current at which to charge or discharge the battery.

### 3.4. Power Consumers

There are many different types of consumers (or "endpoints") to set.

- Each consumer has a name and type so that the user can easily identify them. This allows the user to schedule and prioritize them, and of these, current and historical information can be viewed.
- In addition, the following parameters must be specified:
- The average usage (in kW) of the endpoint. This can be a static value, or adjustable via a component.
- If the installation has 3 phases, it must be specified which phase the endpoint is connected to or whether it is a 3-phase connection.
- The average duration how long the endpoint is active. For example, the time it takes to fully charge the car, or the time the washing machine runs. This can be a static value, or adjustable via a component.
- The component that allows the endpoint to be turned on and off by the Smart Energy Management.
- If the endpoint should always be ready at a certain time, for example if the car should always be charged before 08:00, or the dishwasher should be ready at 17:00, this can be set here. If no end time is specified, then Smart Energy Management ensures that the endpoint is always scheduled within 24 hours.

### 3.5. Scheduling Options

When an endpoint is scheduled, it is added to the Smart Energy Management's scheduler. That means it will look for the optimal time to run the endpoint within the next 24 hours, or before the specified end time (if applicable).

- It is possible to schedule an endpoint through the app, schedule the endpoint automatically every day from a moment, or use a component to schedule it.
- It is also possible to mutually exclude endpoints. This ensures that certain endpoints will never be activated in parallel. This can be useful if, for example, you want to schedule one pump, heater or filter at multiple times for a shorter period of time. So you can schedule them multiple times, and then Smart Energy Management will look for the best times in the day, but always one after the other.

### 3.6. Shutdown Options

Some endpoints have their own run time, while others need to be actively turned off.

For example, a washer or dryer has its own program, and should never be forced to be turned off. In contrast, a pump, which must run for several hours a day, should be explicitly turned off after its run time.

- You can specify whether an endpoint may be turned off by Smart Energy Management, for example when there is insufficient power available.
- You can also specify whether Smart Energy Management should always turn off the endpoint after a certain amount of time. This can be either after the average run time, or some other time.

### 3.7. Priority Options

When multiple endpoints are scheduled simultaneously, Smart Energy Management will schedule them depending on their priority. Endpoints with a higher priority will be executed earlier.

When an endpoint needs to be ready at a certain time, the priority is automatically increased at that time to make sure it succeeds.

- You can specify the default priority as a number between 1 and 100.
- It is possible to request priority through the app, or through a component. When priority is requested for a (scheduled) endpoint, Smart Energy Management adjusts the schedule to start it as soon as possible.

The same parameters apply to OCPP car chargers, except that average usage does not need to be specified, as this information is available in the OCPP configuration.

### 3.8. Putting the Smart Energy Management into Operation

Once all energy sources and endpoints for the Smart Energy Management have been set up, they must be made available on the APP-Control Server controller.

By loading the relevant APP-Control Server project with the Smart Energy Management on the APP-Control Server controller, it becomes active.

Loading is done by logging on to the APP-Control Server controller and pressing "Load configuration". It is also possible to click "Request device to retrieve configuration" from the project's settings page in My Buildings.

### 3.9. Smart Energy Management on the KNX APP-Control Server Controller

On the Smart Energy Management page of the APP-Control Server controller you can see the configuration and graphs of the last 24 hours and the next 24 hours. If the installation has 3 phases it is possible to show the graphs per phase, or in one combined graph.

On this page it is also possible to manually schedule endpoints, remove them from the schedule or prioritize them.

In the graph of the past 24 hours you can see:

- The rate for this period
- The predicted production
- The actual production
- The actual consumption
- The endpoints that have been active during this period (if any)

For the next 24 hours you will see:

- The tariff for this period
- The predicted production
- The predicted usage (based on the average usage plus the scheduled endpoints)
- The endpoints scheduled for this period (if any)

Below this is a table showing all configured endpoints with their current status and schedule. From this table an endpoint can be scheduled and/or priority requested.

When an endpoint is scheduled, Smart Energy Management will look for the optimal time to activate it. When priority is requested for a scheduled endpoint, Smart Energy Management will reschedule to enable the endpoint as soon as possible.

### 3.10. Smart Energy Management in the KNX APP-Control Server App

When the Smart Energy Management is configured, a button is shown in the configuration menu of the APP-Control Server app to access the Smart Energy Management.

Here the same graphs are available as on the APP-Control Server controller (see previous chapter) and it is also possible to schedule endpoints for Smart Energy Management or request priority for scheduled endpoints.

It is also possible to include Smart Energy Management directly in the visualization with the "Smart Energy Control" element.

For example, you can choose specific endpoints to display on a page. This will show the current status, it can be scheduled, you can see the scheduled time and you can request priority.

It is also possible with the "Smart Energy Control" element to include an overview of all energy, with consumption, production and any energy storage. This gives a practical overview briefly of the energy performance of your home or building.

Thanks to all these functionalities, it is possible to let the APP-Control Server manage your energy consumption efficiently, while always maintaining full control and insight into how this is done.

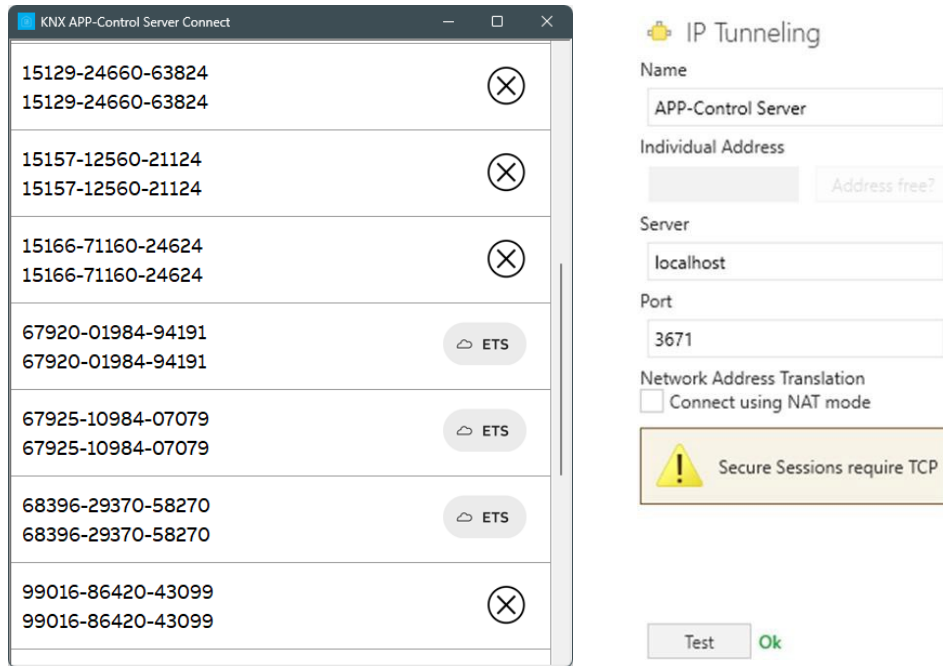
## 4. KNX programming via Remote Access Service

Program the KNX bus with the APP-Control Server via the remote access service. No port forwarding or VPN is required.

What is needed?

- Ensure that KNX programming is allowed via the remote access service (basic settings on the device) and that the remote access service is activated and connected.
- We strongly recommend using KNX IP secure; you can change this in the protocol settings.
- Download the "APP-Control Server Connect" Windows app and install the APP-Control Server Connect app for Windows.
- Start the program and log in with your ProService account.
- Use the "ETS" button of the device you want to connect to. A flashing icon means a connection is being made; a static blue icon means the application is connected.
- In ETS, use localhost to connect instead of the IP address or hostname of the device.

In addition to this new feature, we have also made it possible to access your APP-Control Server via a remote web connector from the remote access service. This is possible via the ProService Portal or use the APP-Control Server Connect program and connect. Through your internet browser on your PC, you can access the web browser of the APP-Control Server by entering <https://localhost/>.



## 5. Modbus Client

The APP-Control can now as a Modbus client, read Modbus server data, this is currently only possible via Modbus TCP.

On the My Building portal under the project, a new option has been added (under Components), namely Modbus.

First, you need to add a device here for the respective Modbus device. Various settings can be entered here that depend on the Modbus server:

- Device ID, also known as unitID, is the ID of the Modbus device on the Modbus line. (For Modbus TCP often 255)
- Address is for Modbus TCP the IP address of the Modbus server.
- Interval indicates how often the registers to be read
- Timeout indicates how long the APP-Control maximum to wait for a response.
- Endian is a setting that says something about the byte order in the data packets, this depends on your Modbus server

After adding the device, you can add registers, coils and inputs per device, click on the icon with the + symbol to the right of the IP address in the list of Modbus devices.

Here you can specify the required settings for each Modbus address:

- Type of Modbus address (input, coil etc)
- A name for recognition
- The start address (the prefix is added automatically)
- The data format
- The resolution. For example, you can enter 0.001 if you want to read kilowatts but the Modbus register returns the data in watts. This is usually also mentioned with the Modbus server.



After this, the Modbus addresses are usable in the component list, by using MB:XX/YYYYYY where XX is the number of the Modbus device (listed on the Modbus register) and YYYYYY is the Modbus address.

Edit Modbus device < II

Name

Parameters **Group objects**

420480	Total energy accu. - Active import	Holding register 64 bits unsigned int	0.01	
423296	Instantaneous values - Voltage L1-N	Holding register 32 bits unsigned int	0.1	
423308	Instantaneous values - Current L1	Holding register 32 bits unsigned int	0.01	
423314	Instantaneous values - Current N	Holding register 32 bits unsigned int	0.01	
423316	Instantaneous values - Active power Total	Holding register 32 bits signed int	0.01	
423318	Instantaneous values - Active power L1	Holding register 32 bits signed int	0.01	
423340	Instantaneous values - Frequency	Holding register 16 bits unsigned int	0.01	
423354	Instantaneous values - Power factor Total	Holding register 16 bits signed int	0.001	

Modbus

ABB - Terra AC Wallbox

4 byte value (DPT 14.x ...)	Max rated / settable current (A)	8/2/0	MB:17/416390, 8/2/0	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4 byte value (DPT 12.x)	Error Code	8/2/1	MB:17/416392, 8/2/1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4 byte value (DPT 12.x)	Socket lock state	8/2/2	MB:17/416394, 8/2/2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4 byte value (DPT 12.x)	Charging state	8/2/3	MB:17/416396, 8/2/3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4 byte value (DPT 14.x ...)	Current charging current limit (A)	8/2/4	MB:17/416398, 8/2/4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4 byte value (DPT 14.x ...)	Charging current phase 1 (A)	8/2/5	MB:17/416400, 8/2/5	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4 byte value (DPT 14.x ...)	Charging current phase 2 (A)	8/2/6	MB:17/416402, 8/2/6	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4 byte value (DPT 14.x ...)	Charging current phase 3 (A)	8/2/7	MB:17/416404, 8/2/7	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

## 6. BACnet Server

The APP-Control can now act as a BACnet server to access KNX (and other protocols) to BACnet. This requires an upgrade license.

The APP-Control supports binary inputs, outputs and values, and analog inputs, outputs and values. From all 6 types it is possible to make up to 1000 objects available on BACnet.

After applying the license, two additional columns can be used in the project components table: BACnet type and BACnet id. In addition, a number of BACnet parameters can be set on the device under protocols.

## 7. KNX Recorder

The KNX recorder allows the APP-Control controller to monitor and store KNX bus traffic for analysis. The KNX recorder has two options:

- 1) **Offline operation.** Here all data remains on the device, max. 1000 telegrams. A filter can be set. The telegrams can be downloaded via a Download button to your computer, in an XML format which can be read into ETS.
- 2) **Offline + online monitoring.** Same as above, but the telegrams are also automatically sent to the cloud at least every 10 minutes. Or when the internal buffer is full (at high bus load) and when you press the "Synchronize" button.

The telegrams can then be found online under "Applications -> KNX recorder". You can directly view them, filter, search, sort etc. But also here you can make an XML export for reading in ETS.

The online functionality will soon not be available by default for every device, details will follow later.

In both cases you can choose continuous monitoring, or 24h, where after 24h recording is automatically stopped.

In the example below, only group addresses with main group 1 or with main group 3 and middle group 3 are recorded.

**KNX GA telegram recorder**

Enabled  Offline + online

Record GA filter

Recorded telegrams

Online recorder

## 8. Support for Dahua and Hikvision IP Cameras, including PTZ Control

Use the My Building portal website to set up these types of cameras directly now. You can specify for each camera whether it supports PTZ or not.

## 9. Weather Forecast Information

The APP-Control now supports various functions to use weather forecast information from your location. (The coordinates on the system settings determine the location)

The information can be used in LUA scripts and in triggers/conditions. There are 6 new LUA functions available, see below, to query this weather information.

It is also possible to display this weather information directly in the app.

## 10. New Trigger Options

Time triggers and conditions can be used directly with the conditions without the need to work with a scheduler. For example: Check at 7:00 AM whether the sun is already up or not, and turn on the outdoor light, if not.

Weather forecast conditions: For example: check at 8:00 AM if it will be above 28 degrees for the next 20 hours and turn on an extra ventilation setting if so.

Or check at 4:00 PM whether or not it will rain in the coming day, and then turn on the garden sprinkler setting or not.

## 11. Include LUA Scripts as Library in other LUA Scripts

It is now possible to put frequently used proprietary functions into a separate LUA script and use them in other scripts via the include function.

Explanation of the new LUA functions:

`ct.include(script_id)`

`script_id` = the id of the LUA script to be used as a library.

`ct.getmeteostatus()`

This function returns the number of hours the weather information is out of date. For example 1, means the weather information is 1 hour old. The weather information is updated automatically, this function is available to query the status if necessary.

`ct.getmeteoinfo(what[, offset[, unit]])`

This function returns the requested value of the requested property. If the data is not available, -1000 is returned.

`what(string)` is the property being requested. (See below for what properties are available)

`offset` (optional, default = 0) in hours, how many hours in the future you want to know this information. For example, 4 returns the information about the forecast in 4 hours.

`unit` (optional, default = 'C') for temperatures, the value can be requested in various units: Celcius (C), Kelvin

(K) and Fahrenheit (F). For example:

`ct.getmeteoinfo("temperature", 8, "F")` returns the expected temperature in Fahrenheit in 8 hours. `ct.getmeteoinfo("rain", 24)` gives the expected precipitation in mm over 24 hours.

`ct.getmeteoinfomin(what, from_offset, to_offset[, unit])`

This function returns the minimum value of the property you query from the period between the `from_offset` to the `to_offset`. When no data is available, or in case of an error, -1000 is returned. `what(string)` is the property being queried. (See below for what properties are available) `from_offset` in hours, from when to be queried.

`from_to` in hours, up to when should be queried.

`unit` (optional, default = 'C') for temperatures, the value can be requested in various units: Celcius (C), Kelvin

(K) and Fahrenheit (F). For example:

`ct.getmeteoinfomin("temperature", 0, 8, "F")` returns the expected minimum temperature in Fahrenheit between now and 8 hours from now.

`ct.getmeteoinfo("rain", 24, 48)` returns the expected minimum precipitation in mm between in 24 hours and in 48 hours.

`ct.getmeteoinfomax(what, from_offset, to_offset[, unit])`

This function returns the maximum value of the property you query from the period between the `from_offset` to the `to_offset`. If no data is available, or if there is an error, -1000 is returned.

For the properties "precipitation", "rain" and "snow", the total forecast is returned, also taking into account the probability.

what (string) is the property being queried. (See below for what properties are available)  
 from\_offset in hours, from when to be queried.

from\_to in hours, up to when should be queried.

unit (optional, default = 'C') for temperatures, the value can be requested in various units:  
 Celcius (C), Kelvin

(K) and Fahrenheit (F)

Table of properties to be queried (weather forecast) "temperature" temperature (in C, F or K)

"feels like" wind chill (in C, F or K) "pressure" air pressure at sea level (in hPa) "humidity" relative humidity (in %)

"dew point" temperature at which water droplets begin to form upon condensation. (Combines air pressure and humidity, in C, F or K)

"uv index" ultra violet radiation risk index, "clouds (in %)

"visibility" average visibility (in meters) Not available for min/max functions "wind speed" wind speed (in m/s)

"wind gust" wind gust, where available (in m/s)

"wind degrees" wind direction, 0 = N, 90 = E, 180 = S, 270 = W

Not available for min/max functions "rain" expected amount of rain (in mm) "snow" expected amount of snow (in mm) "precipitation" expected (in mm) "precipitation chance" (rain and snow)

## 12. KNX IP Router

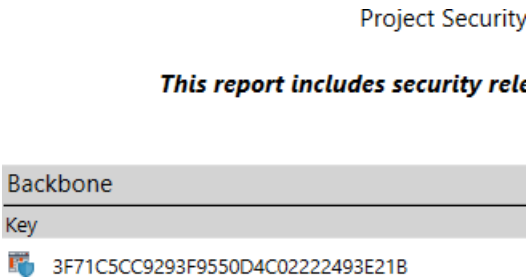
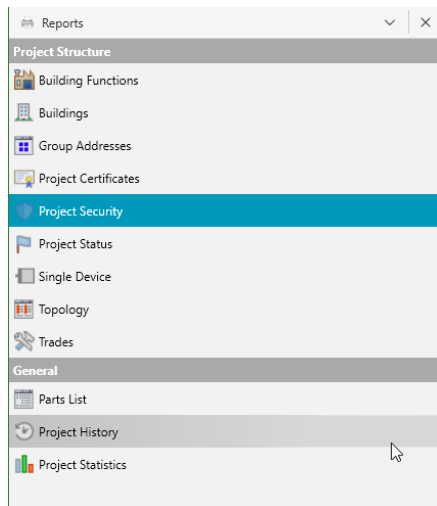
The APP-Control Server can now also be used as a KNX IP router. The router settings are under the protocol settings page of the device. Several default filter options are available, but you can also set them manually:

For example, for IP -> TP (twister pair) you can choose "let through except" and manually specify a number of group addresses to block (or visa versa): 1/0/\*, 4/\*/\* will block all group addresses starting with 1/0/... and 4/... in this example.

## 13. KNX IP Secure Routing

You can enable it under protocols -> KNX -> Router. To use KNX IP Secure for the IP router function, the backbone key must be entered.

This can be viewed in ETS (for a secure backbone) via: Reports -> Project Security and then scroll down until Backbone is displayed.



## 14. KNX IP Secure

It is now possible to set up 10 KNX tunnel connections to the APP-Control Server via KNX IP Secure and program KNX devices via this secure connection or e.g. start group monitors. This also works for the new KNX router functionality.

In addition, several new KNX settings have been added to the protocol settings page. Among which a possibility to set a password for the KNX tunnel. (Only when secure is enabled)

Also, the various tunnels now use their own physical addresses, instead of sharing the physical address of the APP-Control Server.

## 15. Philips Hue API Extension

Now possible to link and use up to 5 bridges with the APP-Control Server. The first HUE bridge now has the notation HUE1:xxxx, the second HUE2:xxxx, the third HUE3:xxxx etc.

It is now possible to switch HUE groups, dim them, etc. When using a lamp, use the number of the lamp, as before. For example: HUE1:1/s switches the first lamp of the first HUE bridge.

When using a group, use the number of the group with a "G" in front of it, for example: HUE1:G1/s switches the first group of the first HUE bridge.

The status page of the APP-Control Server shows the lamps with names and numbers, but now also the groups with names and numbers.

The old notation of the HUE bridge (HUE:xxxx) of course continues to work, when saving in a project it is automatically converted to HUE1:xxxx.

## 16. New LUS Functions

`ct.getSunAzimuth()`

Retrieves the current sun angle relative to north. (Rotation Angle)

`ct.getSunAltitude()` Allows the current sun angle relative to the horizon to be queried. (Altitude Angle) Both can be useful for automatic louver controls for example.

`ct.getusername()` -> this function returns the username of the user who started the script (if available)

## 17. RTSP Support for Cameras

The APP-Control app now supports RTSP camera streams. For this purpose, the camera type "RTSP" has been added in the project. Snapshots can also be taken of these cameras.

## 18. New Firmware Core

The heart of the firmware (core) has been completely rewritten and, in addition, the data model has been modified:

In the past, the application consisted of several modules, which were built together to eventually form one complete application that ran on the device.

In this new firmware, this has changed. All the different modules are now built into different applications, with two additions: 1 for monitoring (controlling, starting and stopping the other modules) and 1 as a communication layer between the different applications. This new approach has several advantages and also a few limitations.

Advantages:

More secure: the different modules can only access the information they actually need, in addition they run at a lower user level.

More flexible: it is easier to add new modules in the future.

More reliable: when a problem occurs in one of the modules it usually does not affect the other modules. Easy to find and solve problems: because the modules are now separate applications, any problems in them are easier to find.

Limitations:

Sometimes slightly slower: because some operations require multiple communications between different modules, this can sometimes be slightly slower. This is mainly noticeable in the web interface.

## 19. OCPP Car Charger

It is now possible to connect a wallbox car charger to the APP-Control Server. This is done via the OCPP (version 1.6j) standard, so the wallbox must support it.

An app element for operation is also available. Currently, only the KNX <-> OCPP function is available for testing.

How to set this up?

1. On my Buildings under the application OCPP auto chargers, set the parameters and the KNX groups. (Two chargers are currently supported per APP-Control Server). The OCPP wallbox will log in with a unique ID, this is often the serial number of the device. If this is not known and cannot be found in the wallbox documentation, after step 3 when the wallbox tries to connect, this ID can also be read on the status page of your APP-Control Server. We support both the ws (websocket) and wss (secure websocket) standard, for this.
2. Load the configuration in the APP-Control Server and turn on OCPP under protocols (also choose here whether the wallbox will connect via ws or wss). For the wss default, you may need the ROOT certificate.
3. Set the wallbox to connect to the APP-Control Server, at wss://ip-address:8016/ where the IP address, is the IP address of your APP-Control Server. How this is done depends on the wallbox and brand. In some cases, contacting the manufacturer's support department is necessary.

4. On the status page you should now be able to see that the OCPP is going to connect, this may take a while.

The following objects are available:

- Status (enum).
  - 0 - Available
  - 1 - Preparing
  - 2 - Charging
  - 3 - Temporarily interrupted by vehicle
  - 4 - Temporarily interrupted by wallbox
  - 5 - Completed
  - 6 - Reserved (wallbox is reserved for a future charging action)
  - 7 - Not available
  - 8 - Fault
- Error code (enum)
  - 0 - No error
  - 1 - Error with lock
  - 2 - Error in communication with vehicle
  - 3 - Ground fault
  - 4 - Temperature too high
  - 5 - Internal error
  - 6 - Local list error (for authorization of RFID)
  - 7 - Other error
  - 8 - Current too high
  - 9 - Voltage too high
  - 10 - Meter reading error
  - 11 - Power switch error
  - 12 - RFID read error
  - 13 - Reset fault
  - 14 - Voltage too low
  - 15 - Signal weak
- Start / stop charging
 

Both status and ability to request this. Please note some cars will not accept a second charging session without re-plugging!
- Reserve now / release
 

This can be requested, not all wall-boxes support this.
- Make available / not available
 

Both status, and ability to request this
- Release plug

Some wallboxes can lock the plug, by sending a 1 to this object, it is possible to unlock the plug. (If supported!)

- Transaction consumption (Wh)  
After stopping a charging action (transaction), the consumption of this action will be sent on this.
- Start transaction time  
When starting a charging action, the time is controlled on this.
- End of transaction time  
When stopping a charging action, time is controlled on this.
- Charge current (A)  
On this, the current charge current is controlled.
- Maximum charge current (A)
- Charger reset  
Allows a soft reset to be sent to the charger, for example after a malfunction.

## 20. Logic Module

In addition to the current triggers, actions and scripts, we have now added a graphical logic module.

It is possible to create 20 logic diagrams each with up to 20 logic blocks on it. Inputs and outputs do not count here.

In applications, under Logic, on My Buildings, logic modules can be created. Remember to press "Save" when you finish customizing your logic module.

It is also possible to simulate the different logic blocks. It is also possible to copy logic blocks. All blocks have inputs and/or outputs, which have a color. Bit inputs/outputs are black, numbers pink and texts green.

Inputs can only be linked to outputs of the same type.

The following inputs are now available:

- BIT input (component or constant value)
- NUMBER input (component, all kinds of DPTs are included, or constant value)
- TEXT input (component, or constant value)
- TRIGGER input (e.g. HTTP triggers, page open etc)
- TIME input (allows sending a 1 at the beginning of a period and a 0 at the end)

The following outputs are now available:

- BIT output (component, but also all kinds of commands, scenes etc.)
- NUMBER output (component, but also a few commands)
- TEXT output (component)

Then various LOGIC blocks are now available:

- AND port (2v, 4v and 8v)
- OR port (2v, 4v and 8v)



- XOF port (outputs a 1 when either input is 1, and 0 when either 0 or both are 1)
- BIT invert, turns a 1 into a 0 and vice versa
- BIT delay and NUMBER delay:
  - The top input determines the delay time. (Can also have a constant value of an input)
  - The lower input is output delayed. You can use the parameters (on the right in the editor) to determine whether the value should always be delayed.
- NUMBER compare: is number A (input 1) larger, smaller, equal etc.? To number B (input 2)
- BLOCK (bit, number, text): The first input determines whether or not the second input is passed to the output.
- BIT to NUMBER: This allows two numbers to be entered through the first two inputs. (These can also be constant values). Then, when a BIT arrives on the last input, the corresponding value from either the first or the second input is transmitted on the output.
- BIT splitter to 3 outputs. To link from one output of another component to multiple inputs of other components.
- NUMBER splitter to 3 outputs. To link from an output of another component to multiple inputs of other components.
- NUMBER min: gives the minimum of the two input values on the output.
- NUMBER max: gives the maximum of the two input values on the output.
- NUMBER average: gives the average of the two input values on the output.
- BIT to TEXT: allows a text to be sent on an output depending on a BIT value.
- NUMBER to TEXT: allows to send a text on an output depending on a NUMBER value.
- NUMBER format: allows to convert a number to text with a leading text, a text after the number and to specify the number of decimals.
- Pushbutton detection. Allow a push button to send a 1 when pressed and a 0 when released. This module can use this to determine whether the push button is pressed 1x, 2x or long.
- Counter: Via the R input the counter is set to 0. Via the + the value 1 is increased, and via the - the value 1 is decreased. The new value is issued on the output.
- Staircase lighting automaton, you can use this to simply turn off the lights after an adjustable time (which is given via input 1).
- Watchdog: it expects an input trigger every x seconds. Does it not get it? Then the alarm output is activated
- BIT-BYTE converter: with this it is possible to convert 8 bits to a byte and vice versa.
- Threshold monitor: The first input makes it possible to (temporarily) disable the monitor. The second and third inputs, determine the upper and lower limit to be used. The fourth input is the value to be monitored. If the upper limit or lower limit is exceeded, the corresponding alarm outputs are activated.
- Calculation function: this allows two numbers to be calculated, for example, select + to add two numbers.
- Absolute value: the output gives the absolute value of the input, so when the input is negative, it gives the positive value as the output.
- Toggle bit: allows a bit to be toggled between 0 (OFF) and 1 (ON)

- Time counter: this can be used to create an operating hour counter, for example. The time may or may not also be cyclically controlled. The time output is always in seconds. Through the request input, the current time value can also be requested. With the reset input, the value can be reset to 0.
- Virtual dimmer: With this it is possible to simulate a dimmer, for example to be able to dim a HUE lamp up / down with a pulse button. The speed is the time in seconds it takes to go from 0 to 100. So, the larger the value, the slower the dimmer dims up/down. The bit inputs can be used to dim up, down or alternately up and down separately. A 0 on one of these inputs, stops the up/down dimming. There is also an option to dim with a number. A positive number dims up, a negative number dims down and a 0 stops the dimming. Through the status input any feedback is possible, to set the dimmer to that value. (When it is also controllable via other elements for example).
- Multiplexer (can send out numbers via a selection choice)
- Curve adjustment: this makes it possible to adjust a dim curve, for example.
- Human Centric Lighting curve: to generate a dynamic brightness and color temperature for lighting, matching natural lighting conditions. When activated, it will gradually change the light brightness and color during the day.
- Color temperature to whites: to control two lights, one warm white, one cool white, to create one balanced colored light, based on a brightness and color temperature input.
- Bits to number: this can take a part of a telegram (e.g. Modbus) and make a readable telegram out of this.

## 21. Miscellaneous

- It is now possible to change the name of your phone or tablet in the app's settings so that they are recognizable in alarm functions.
- Internal group addresses can now be used. These are structured as follows:
  - Starts with I: (capital i and a :)
  - Followed by a number (1, 2, 3, )
- Updates to the firmware are now displayed in this screen.
- Updating the firmware is now possible from within the app (setup -> firmware update)
- Improved local configuration backup: when the current configuration cannot be loaded properly (e.g. due to a power failure while saving the configuration), a previous version of the configuration is automatically loaded.
- KNXnet/IP tunnel improvements: support for extended frames (up to an APSU length of 55 bytes), and initial connection is faster.
- New element: album cover image. Display an album cover on a page. (Works with certain uPnP devices and Sonos)
- New component type available: byte value -128..127 (DPT 6.010) that also accepts relatively dim group addresses as status (DPT 3.007)
- New component type available: 8-byte value (DPT 29.x) for use in the Smart Energy Manager
- This component can be used, for example, to easily control a virtual dimmer with KNX dimmer buttons and use it to dim a HUE lamp

- The apps now also support color temperature when set at the dimmer
- The apps now also support blinds with position control and slat control
- Increased the amount of Master RTCs: from 16 to 32 Masters. The total is still 100 RTCs
- HTTP commands have been extended. It is now possible to send requests other than GET requests, and (for PUT and POST) to send a body along with the command.
- In addition, it is possible to view/use the result, when the command is started via an LUA script, the function returns the contents of the HTTP command.
- The status page of the web interface now shows whether there have been incorrect login attempts and from where (and when) the previous login was correct.
- The "variable text" element can now also display meta data from Sonos devices, without using the uPnP module.
- New HVAC options for the thermostat component in the project. It now also supports fan control, as well as HVAC control modes, for air conditioning devices. When the HVAC control mode is used, the default HVAC modes are disabled.
- New image element for on image pages.
- Support for animated GIF images, for the above image element, among others.
- New features for most element types:
  - Element visible (show from) depending on another element. This works on both list pages and image pages.
  - Element enabled (operable) depending on another element. This works on both list pages and image pages. For some elements, this replaces the "read only" option.
- The combination of the above new options makes it possible to create overlays over, for example, a floor plan, such as a glow that certain lights are on.
- But it is also possible to either disable or not show certain option if, for example, alarm is on or outside office hours.
- New bootloader, you now see 2x fast flashing of the OK LED to indicate that the device is booting (first stage)
- New reset options:
  - R1/RESET short press (< 0.5 sec) = rebooting
  - R1/RESET medium press (1 to 3 sec) = network reset only
  - R1/RESET long press (5+ sec) = factory reset