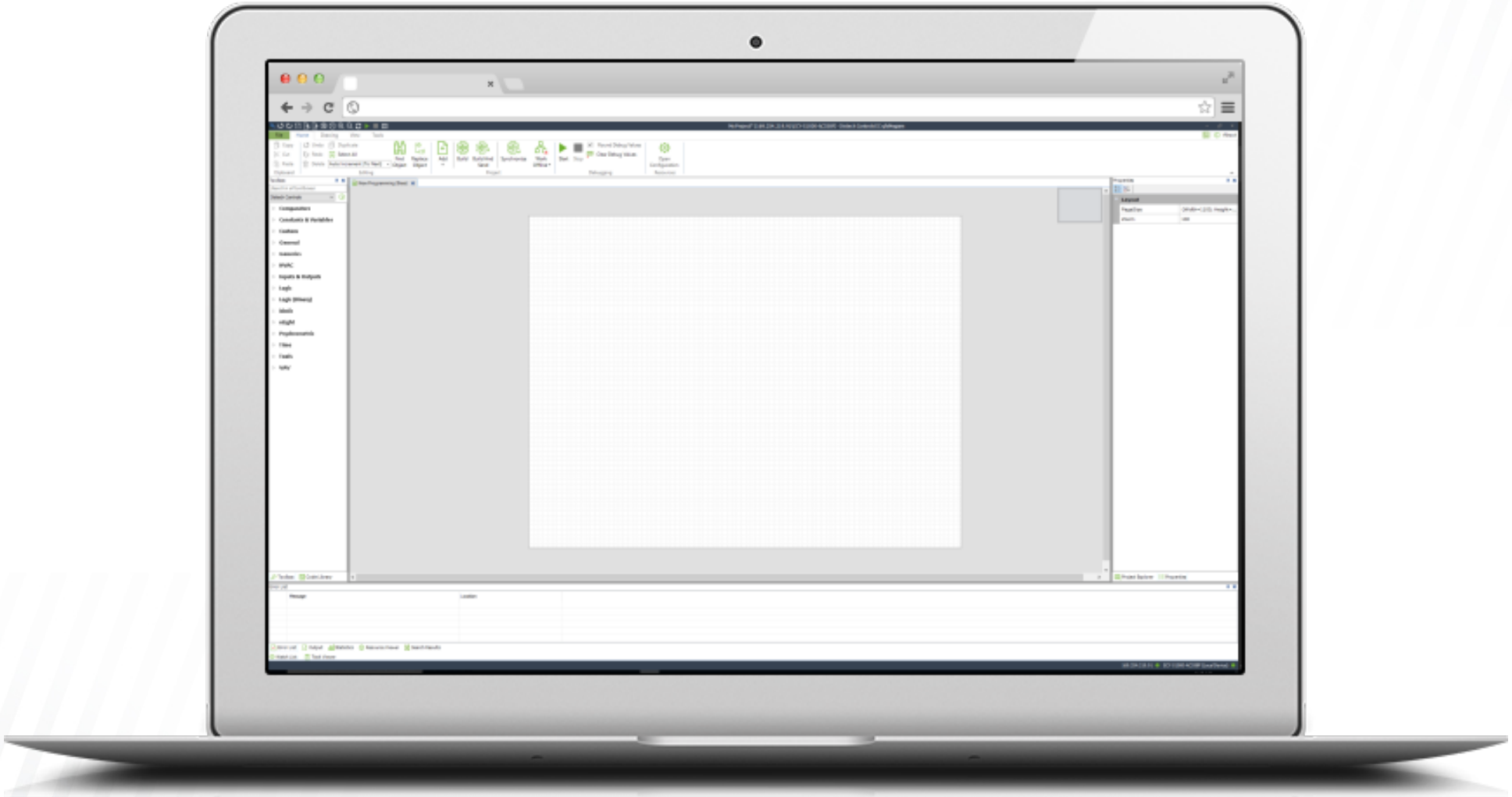
ATRIUS™

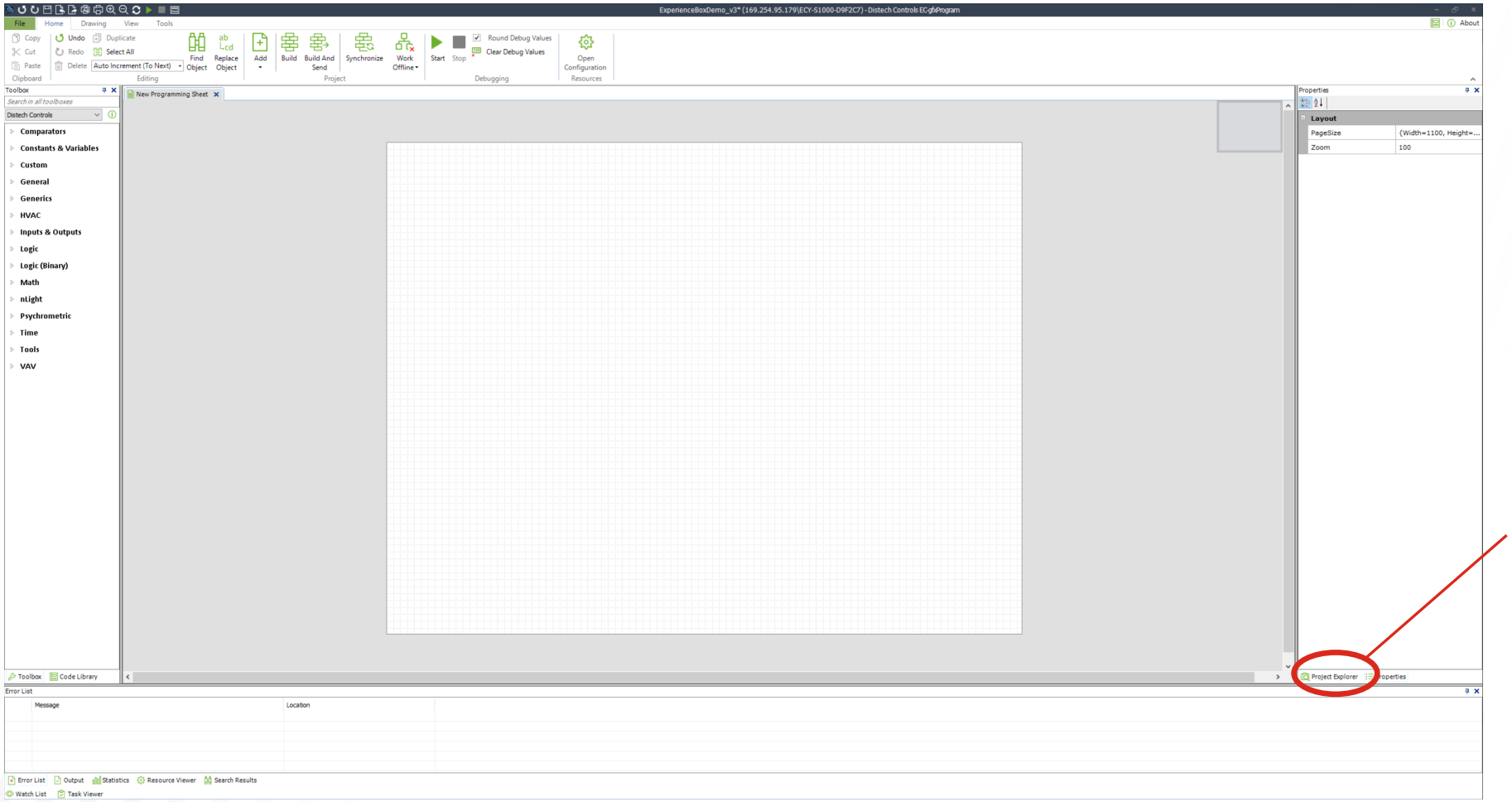
SETUP – OPEN EC-GFX

SETUP – OPEN EC-GFX



ATRIUS™

SETUP – NAME YOUR PROGRAMMING SHEET



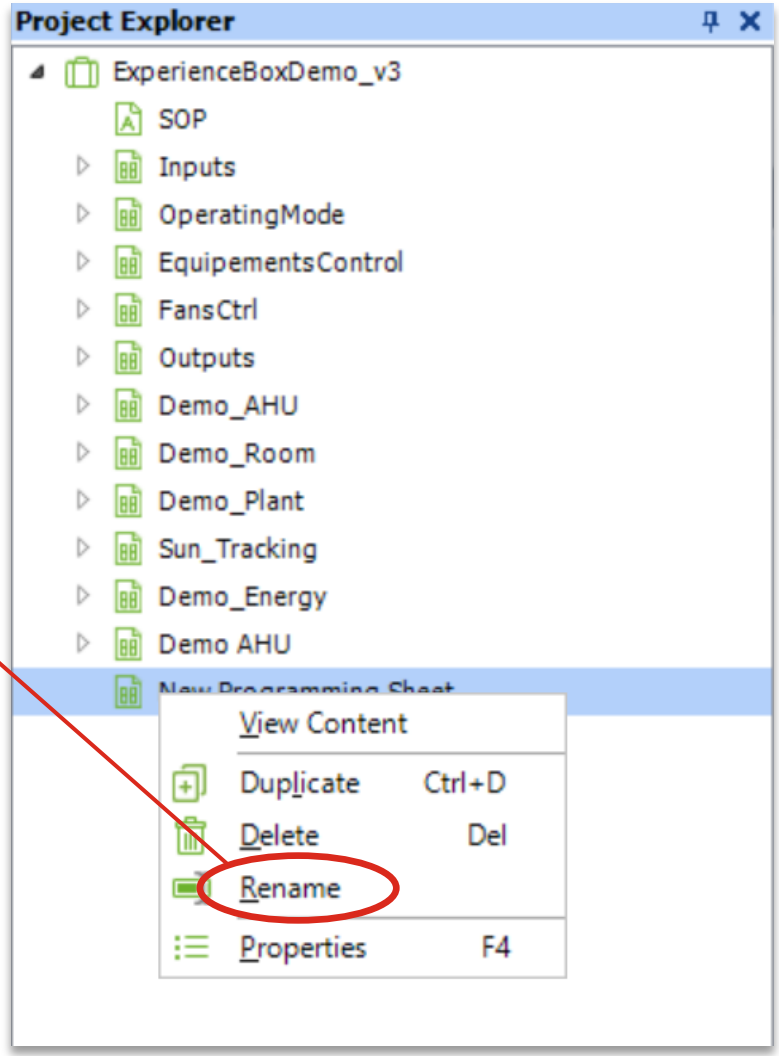
Click **Project Explorer**.

ATRIUS™

SETUP – NAME YOUR PROGRAMMING SHEET

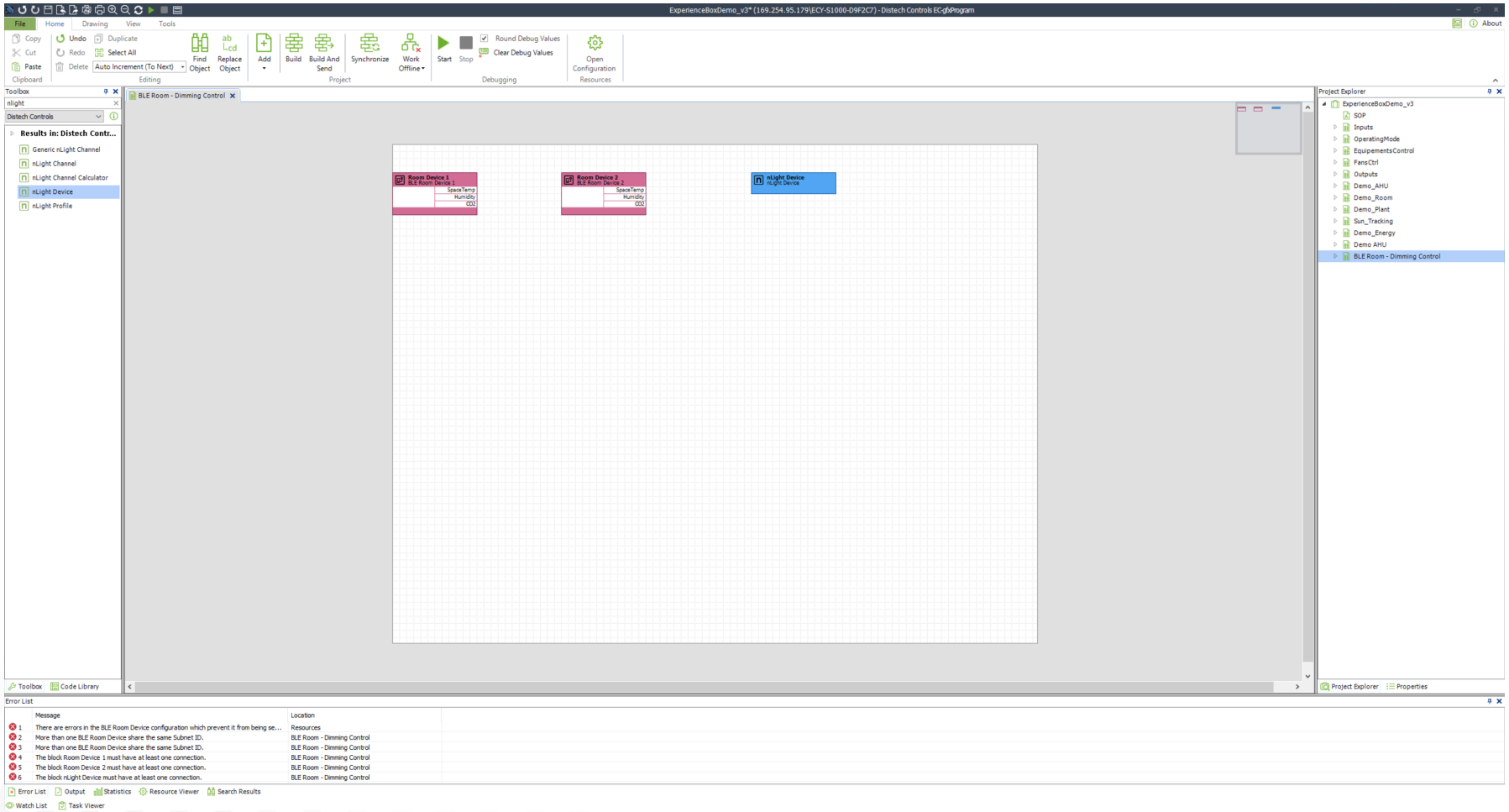
Right click your sheet.

Click **Rename**.



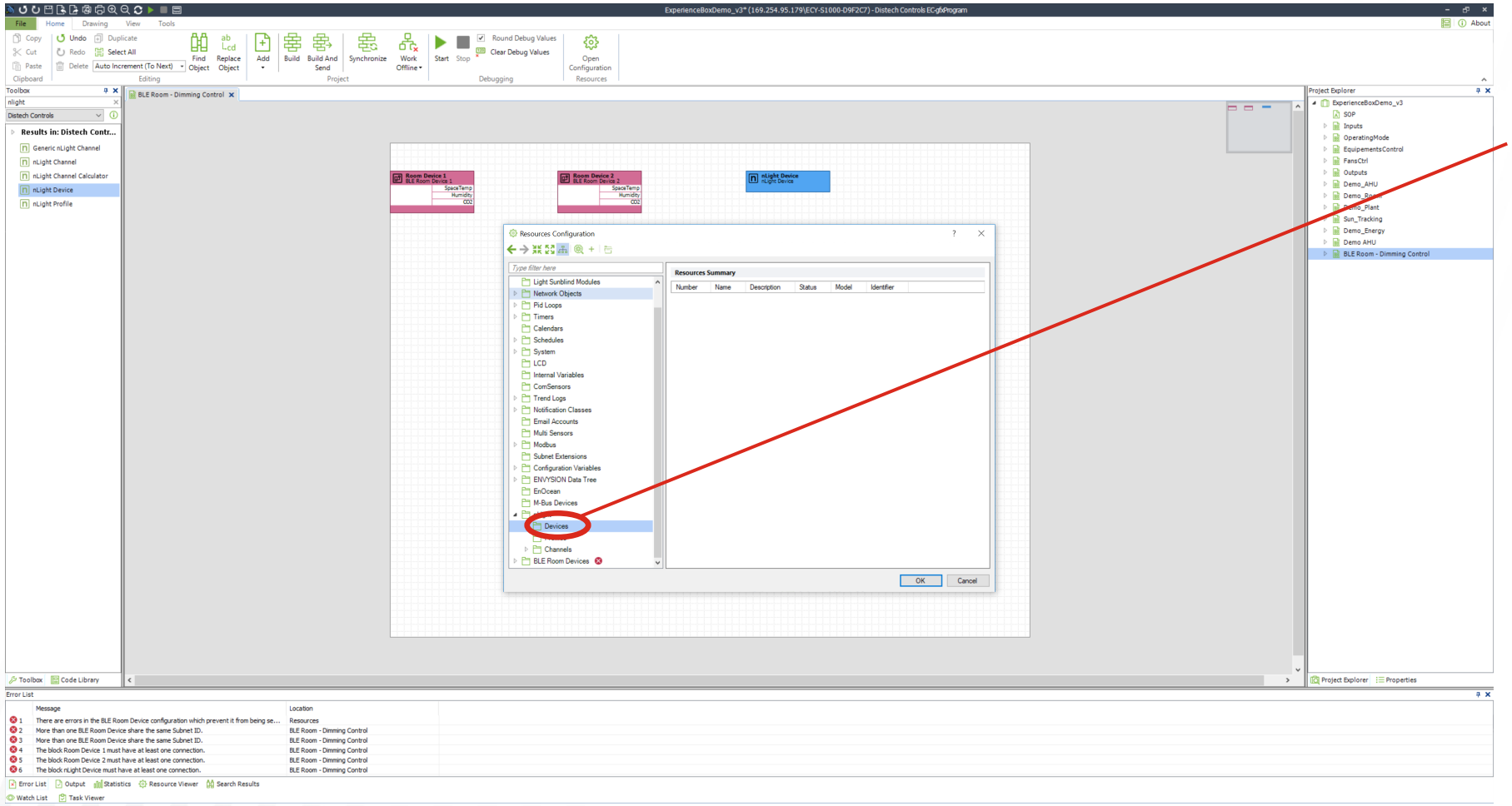
ATRIUS™

SETUP – INSERT BLOCKS FOR CORRESPONDING DEVICES



ATRIUS™

SETUP – IMPORT DEVICES FOR CONFIGURATION USING OPEN CONFIGURATION



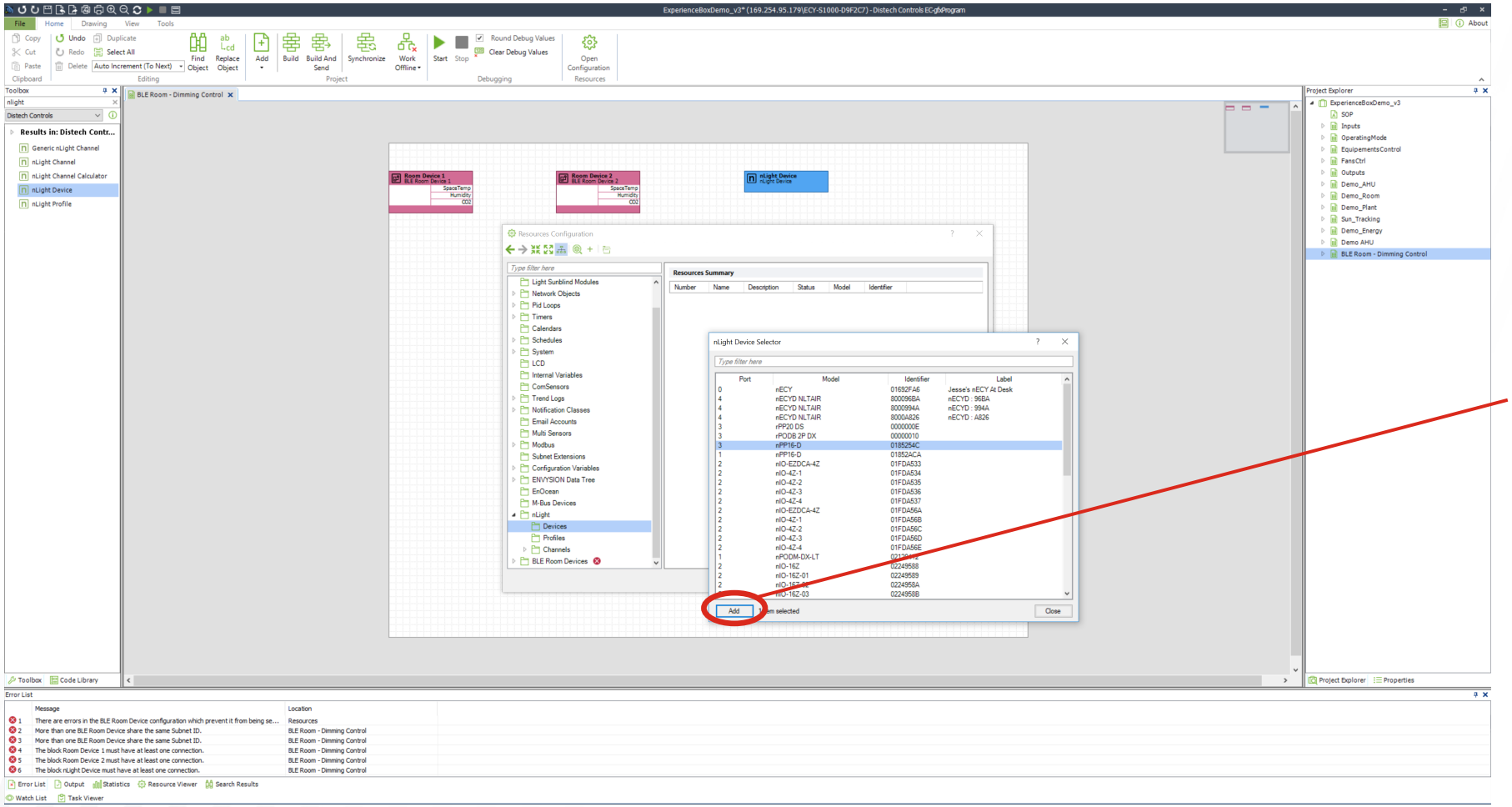
Expand the network tree in the respective area.

Right click **Devices**.

Click **Import From Device**.

ATRIUS™

SETUP – IMPORT DEVICES FOR CONFIGURATION USING OPEN CONFIGURATION



Select the device(s) you wish to import.

Click **Add**.

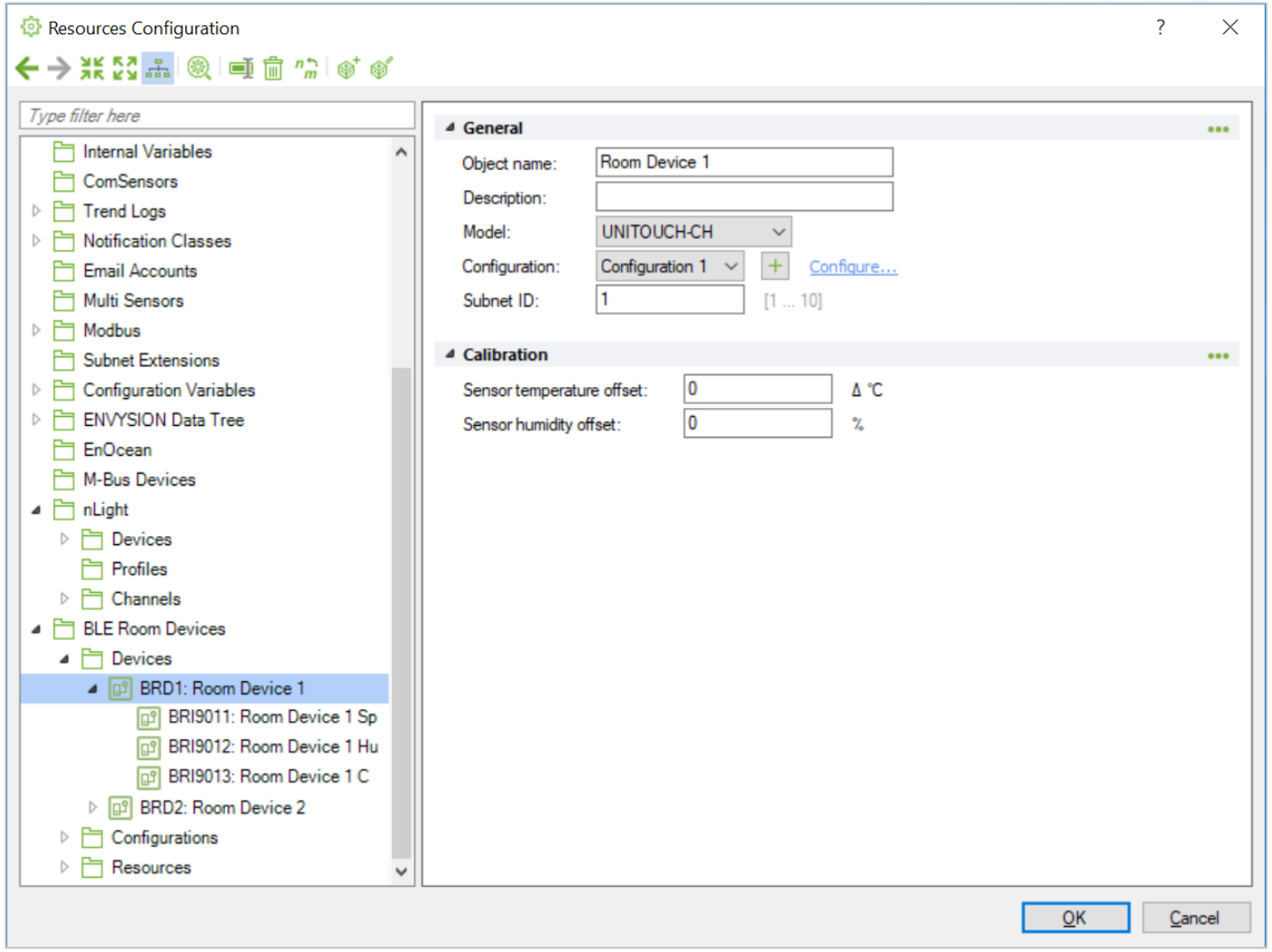
ATRIUS™

SETUP – CONFIRM DEVICES ARE NOT ON THE SAME SUBNET ADDRESS

From open configuration, select a device.

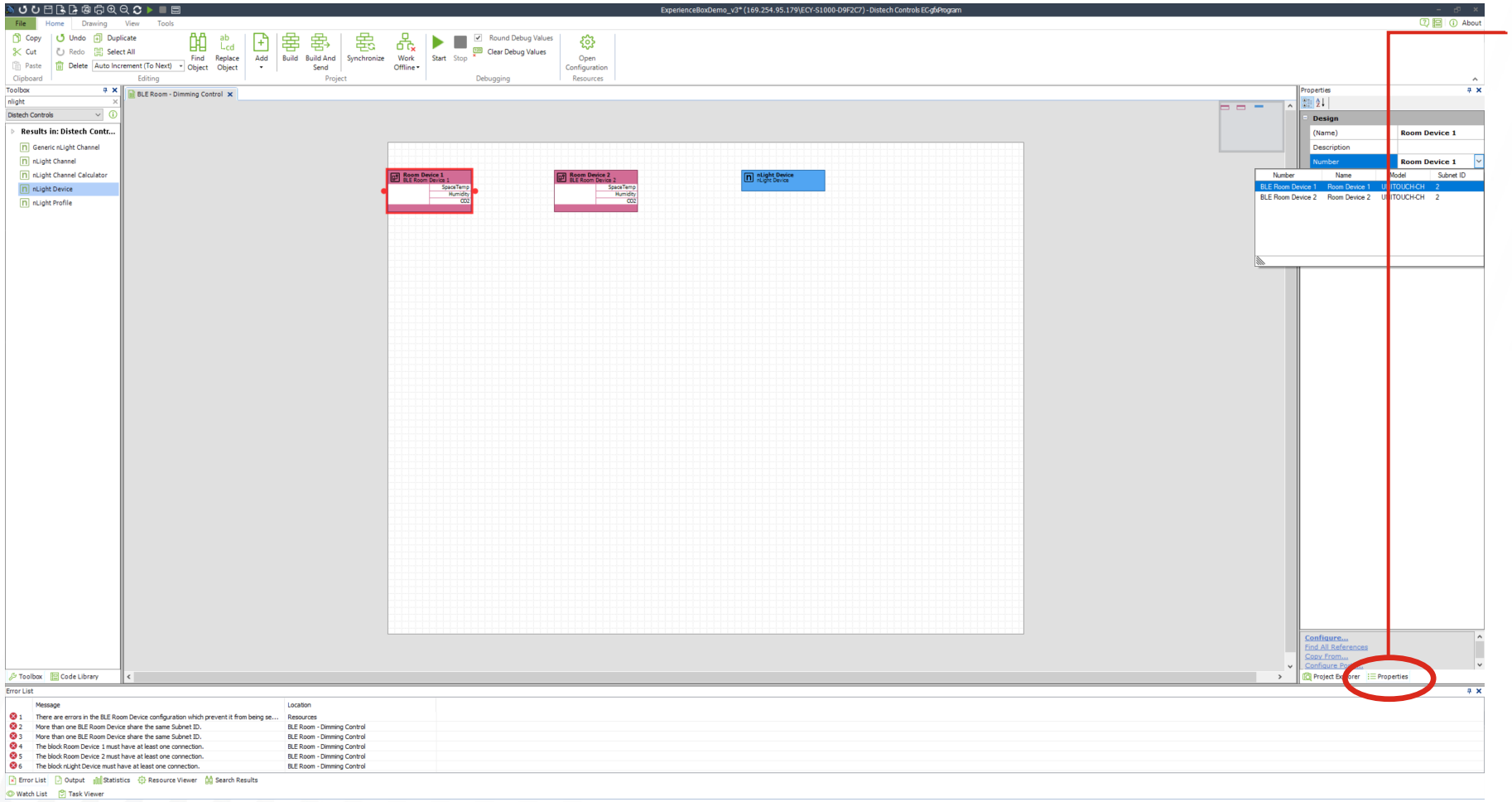
Update the subnet field for each device, removing any conflicts.

Note: two devices cannot have the same subnet address.

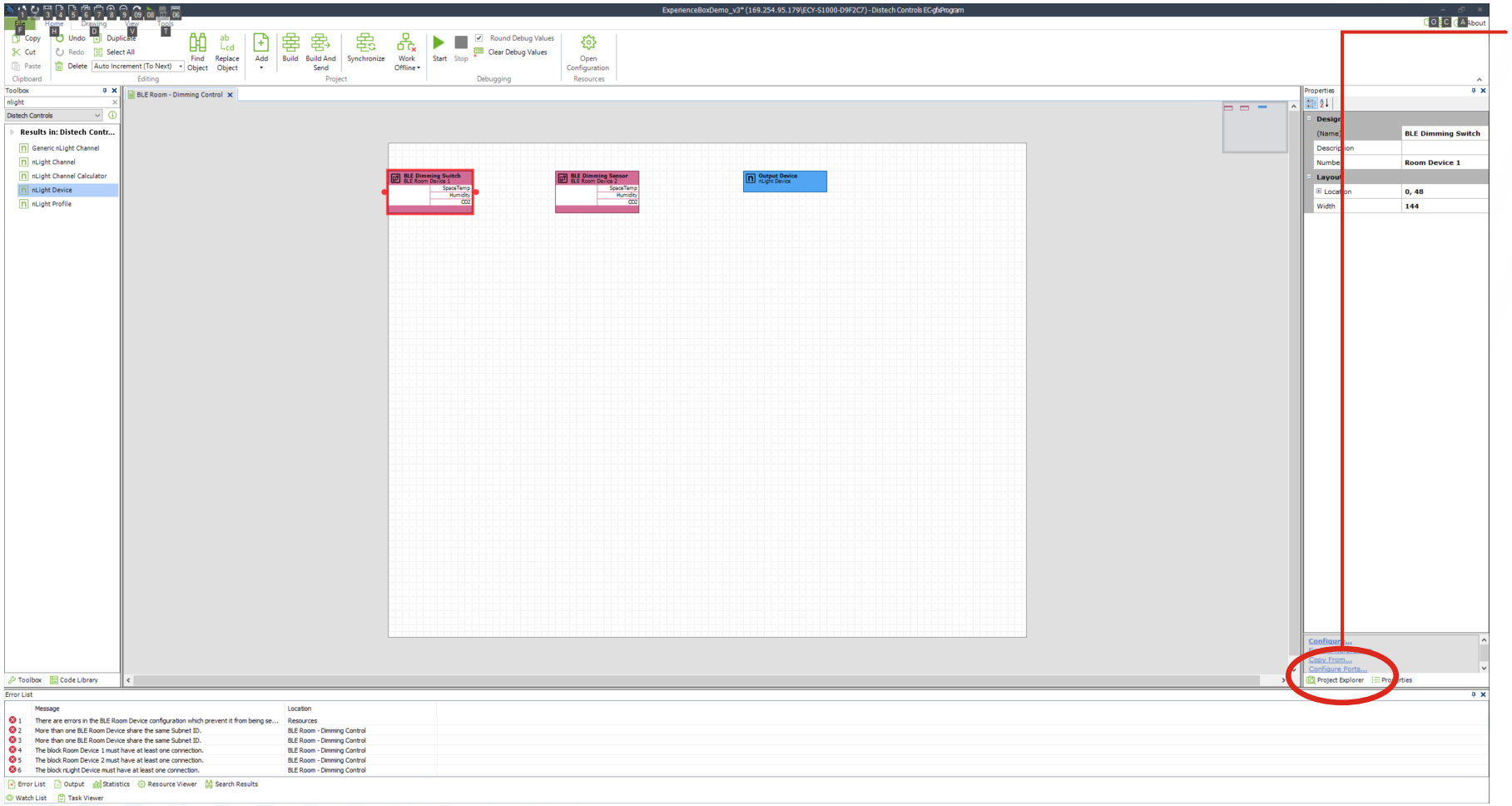


ATRIUS™

SETUP – ASSIGN DEVICES TO BLOCKS



SETUP – UPDATE BLOCK NAMES.



Click **Properties**.

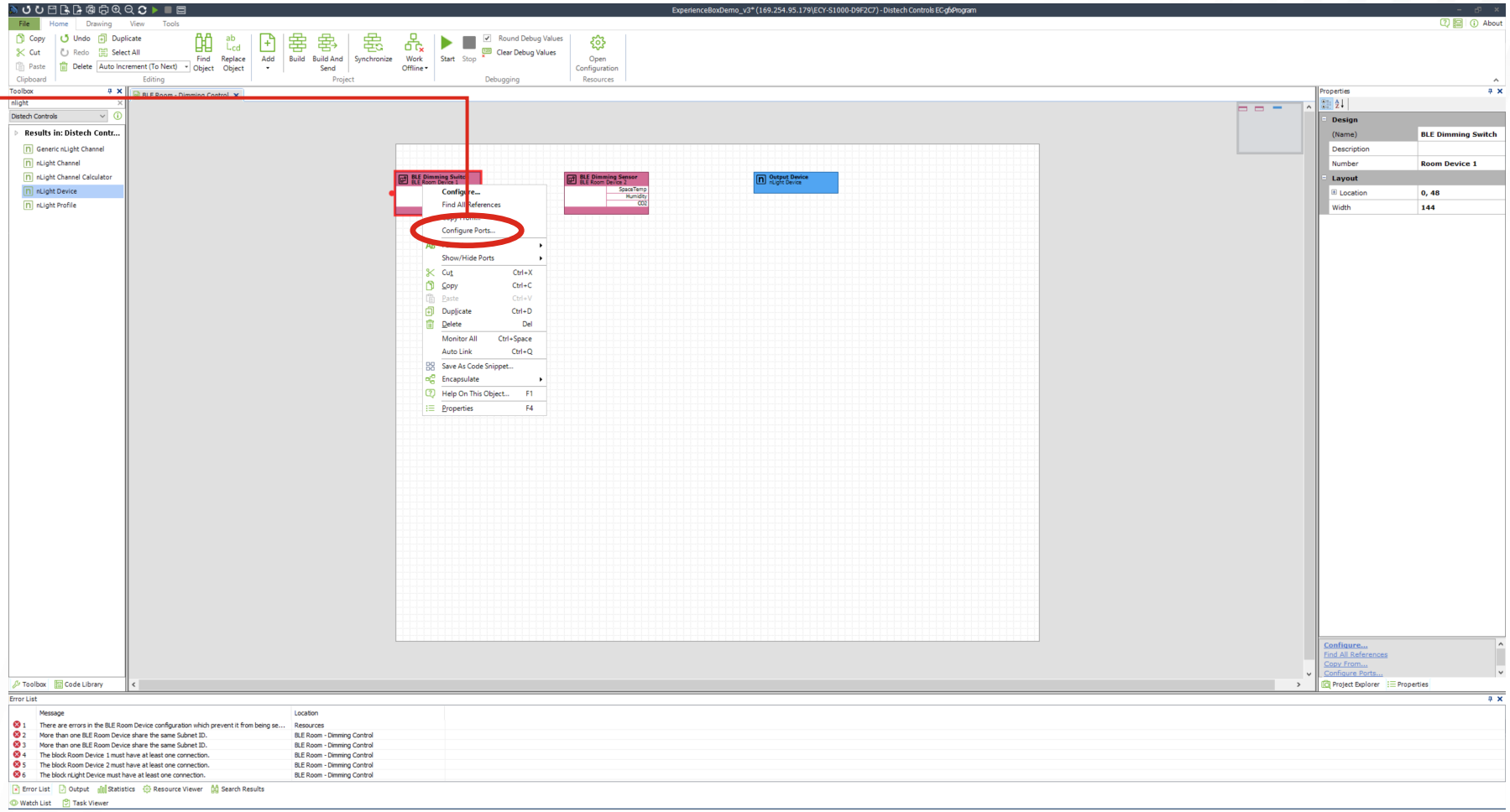
Adjust the block's **Name** field.

SETUP – CONFIGURE PORTS

Right click a block.

Click **Configure Ports**.

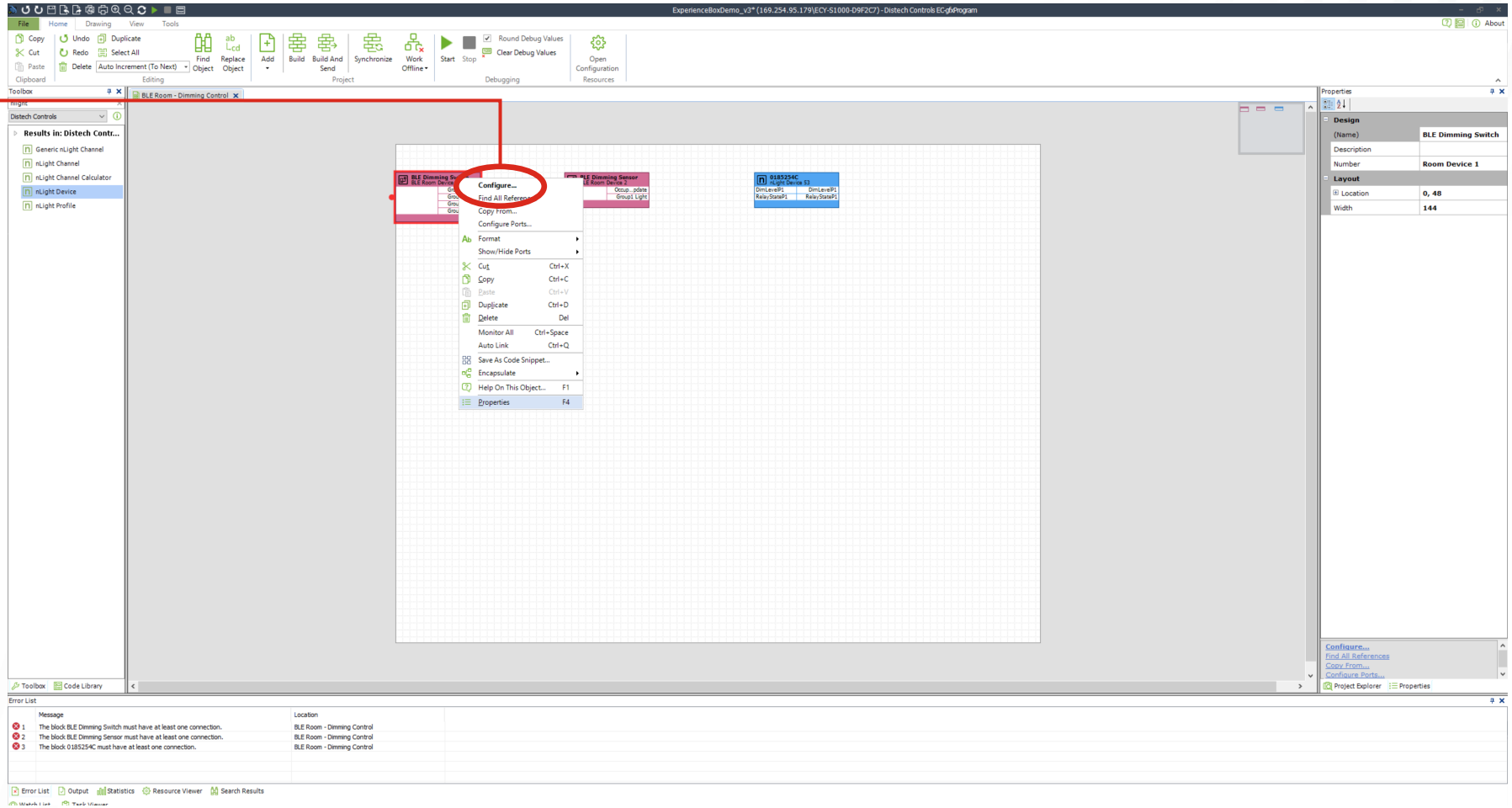
Update input and output ports for any points you want to utilize.



SETUP – CONFIGURE YOUR BLOCK

Right click the block.

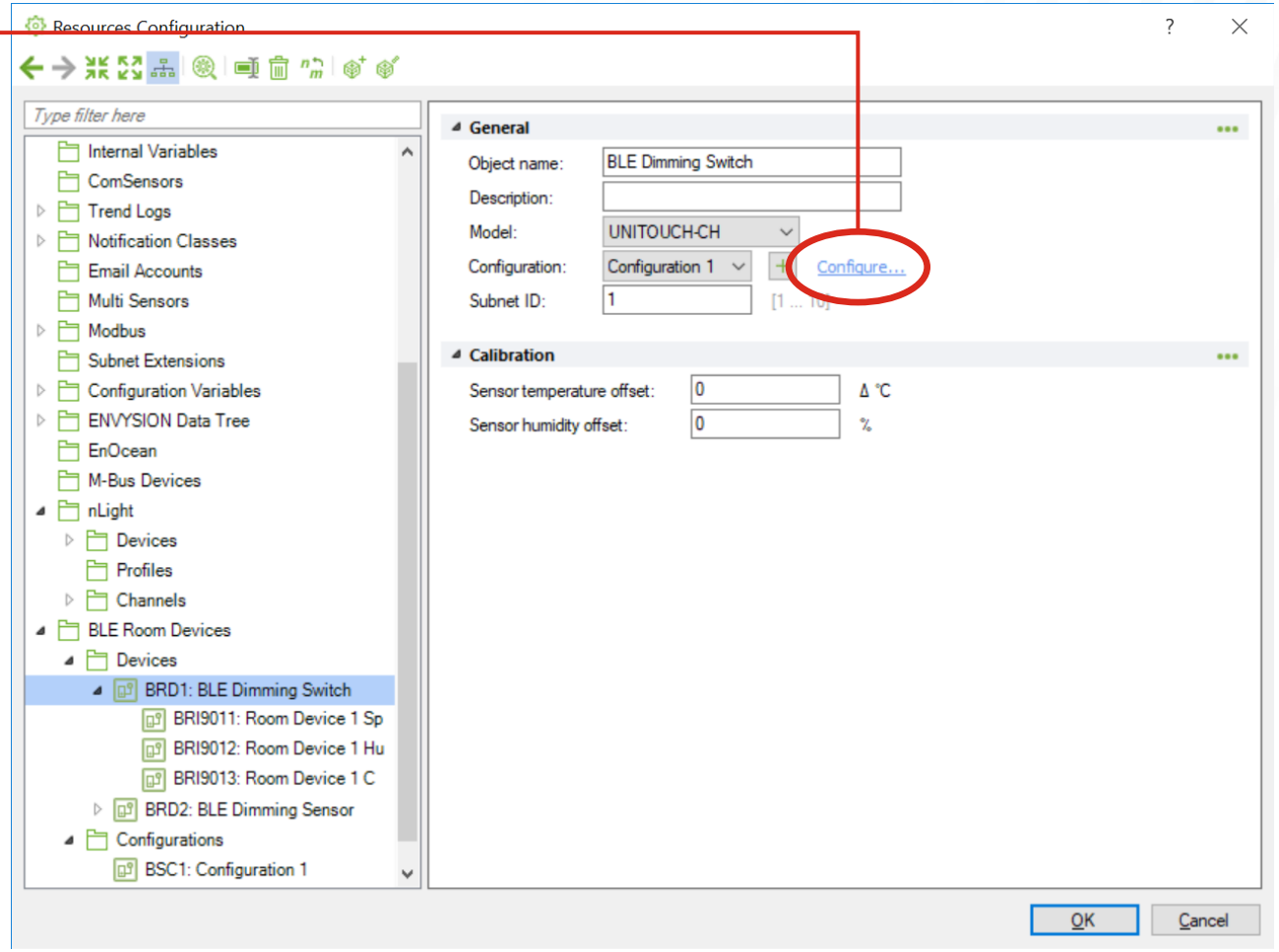
Click **Configure**.



ATRIUS™

SETUP – CONFIGURE YOUR BLOCK (CONTINUED)

Click **Configure** beside **Configuration**.

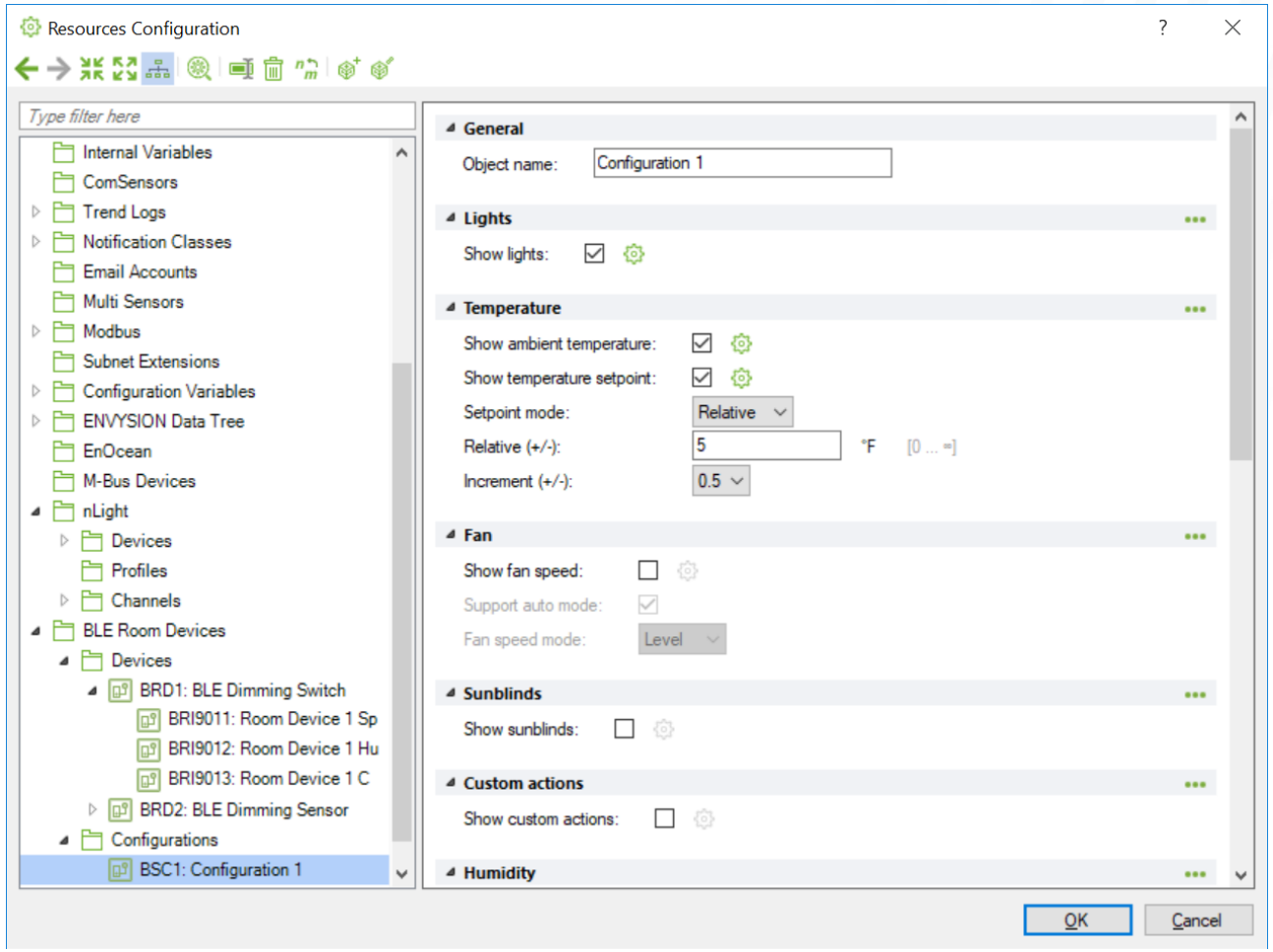


ATRIUS™

SETUP – CONFIGURE YOUR BLOCK (CONTINUED)

Adjust your block's preferred settings that are shown.

Note: Recommended settings for a switch controlling lights would be to expose the "Show lights" option and to configure the corresponding modes, enabling the number of control outputs as the device allows or application requires.



ATRIUS™

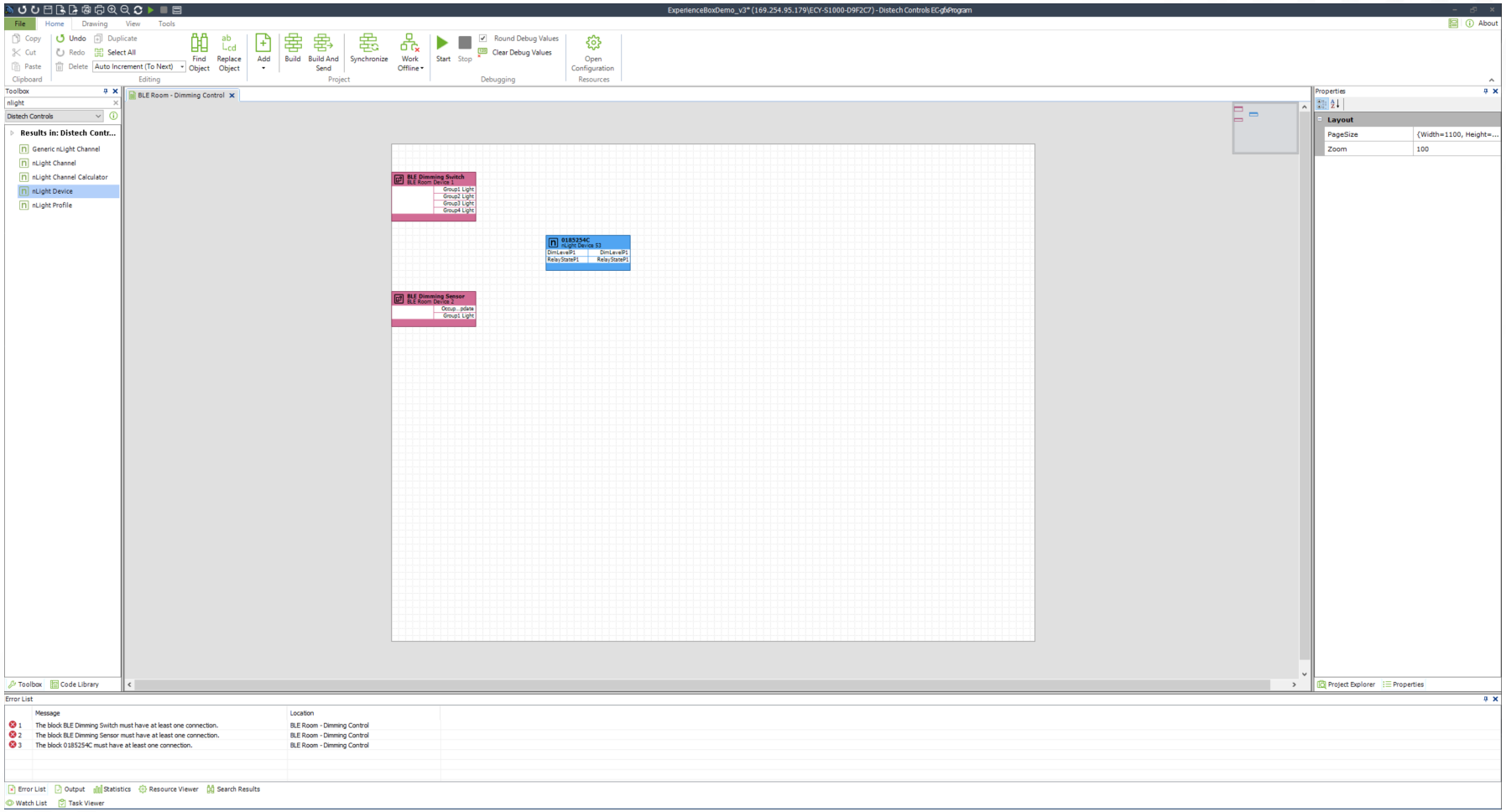
SETUP – ADJUST BLOCKS LOGICALLY ON YOUR PROGRAMMING SHEET

Input devices are best arranged on the left

Output devices are best arranged on the right.

For complex programming, **encapsulation** may help to simplify your programming.

To encapsulate, select a set of blocks, right click, and click **Encapsulate**. This creates a single block representative of all blocks selected. Double clicking the block exposes the previous blocks.



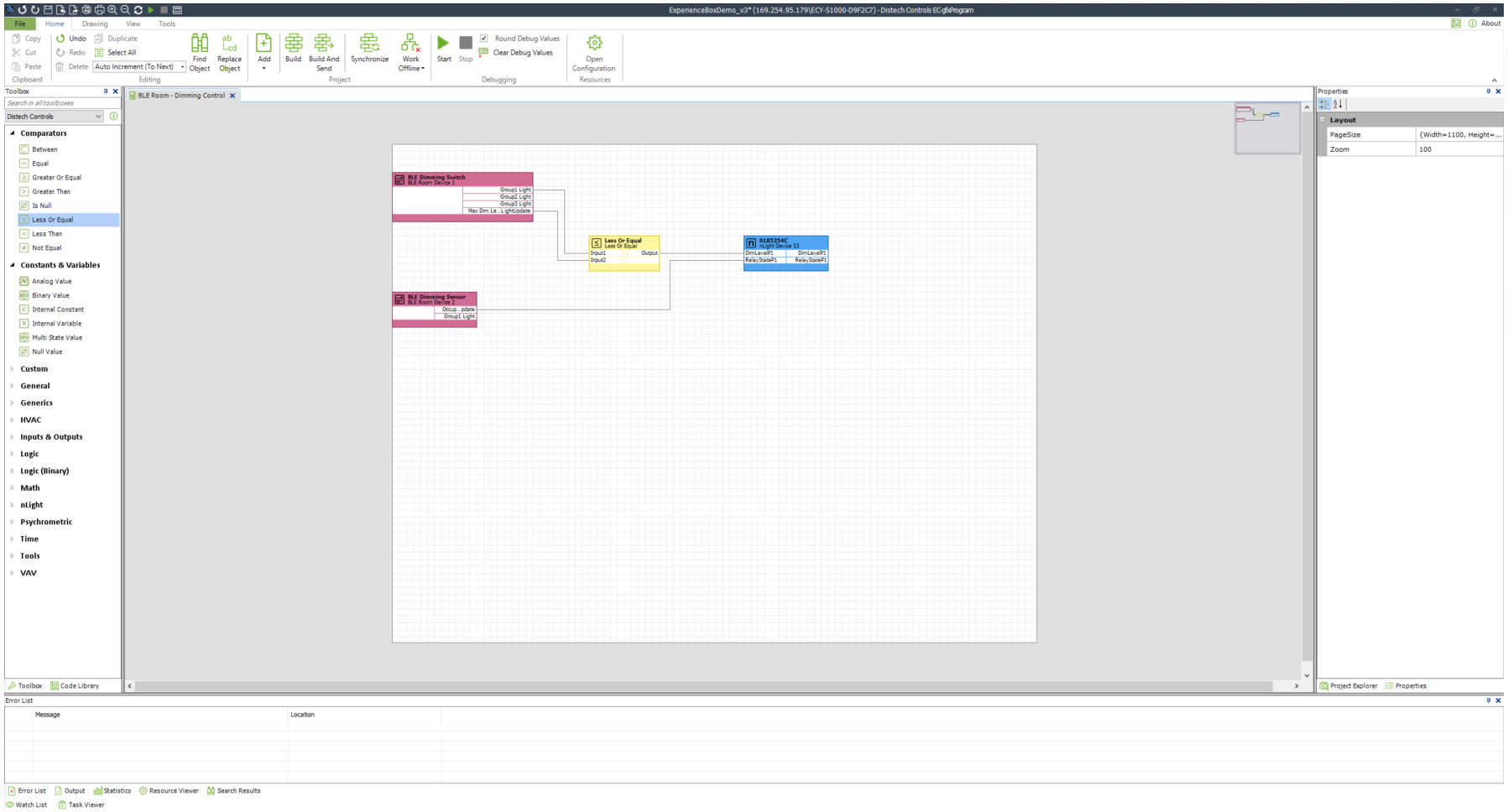
ATRIUS™

CONDITIONAL BLOCKS. LINK INPUTS AND OUTPUTS.

Conditional blocks are used to compare values and to automate decisions.

In the example, we are using a comparator to compare a user-defined max dim level and an output defined by a slider switch. If the slider's value is less than the max dim, it passes to the output block. If it is greater than, it is not passed to the output block.

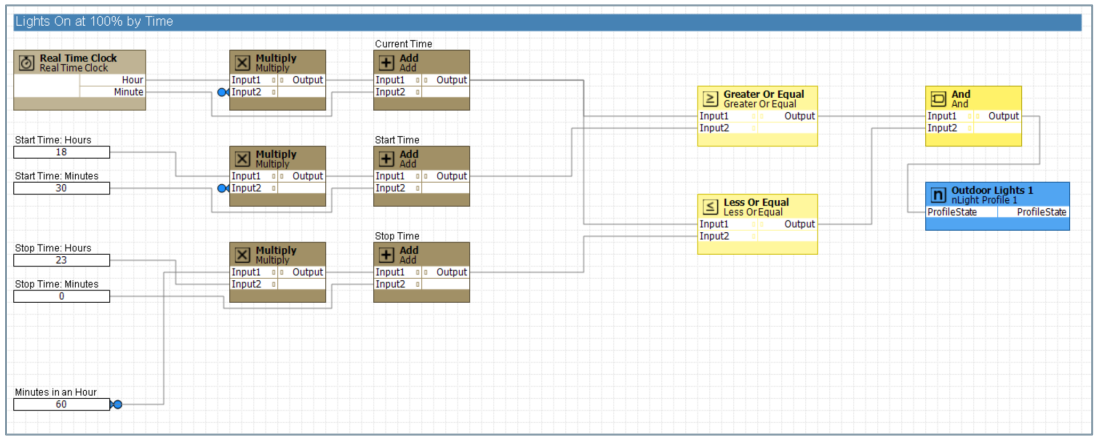
Not affected by the comparator is an output from a sensor block controlling the on/off relay state of the output.



ATRIUS™

LIGHTS ON BY TIMECLOCK AND ASTRONOMICAL TIMECLOCK

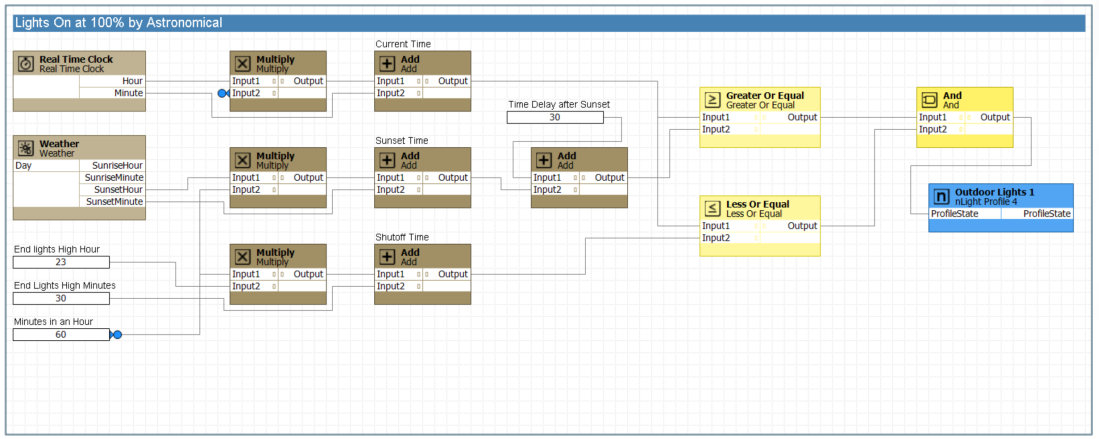
Using the Real Time Clock block and conditional statements, your program can automatically make decisions based on time of day.



To configure Time Control (Scheduling)

- Convert the time clock value to a numeric value using a Multiply block. By multiplying the output of “Real Time Clock” by 60 and using an Add block to add the minutes of the day, the output is a single number that can be associated with time of day.
- The same conversion steps are applied to an on time and off time.
- Use comparator blocks to compare the numeric value to set on and off times. In the example, we configured a schedule to have lights on between 6:30PM and 11PM. As long as a time is before 11PM AND after 6:30PM, the output is on.

Using the Weather block and conditional statements, your program can automatically make decisions based on astronomical information.



To configure Astronomical control

- Convert the astronomical value for SunsetHour or SunriseHour to a numeric value using a Multiply block. By multiplying the output of “Real Time Clock” by 60 and using an Add block to add the minutes outputs, the output is a single number that can be associated with time of day.
- Use comparator blocks to compare the numeric value to set on and off times. In the example, we use automatically determined astronomic on and astronomic off times. As long as we are after the set astronomic time AND before the off time (11PM), the output is on.

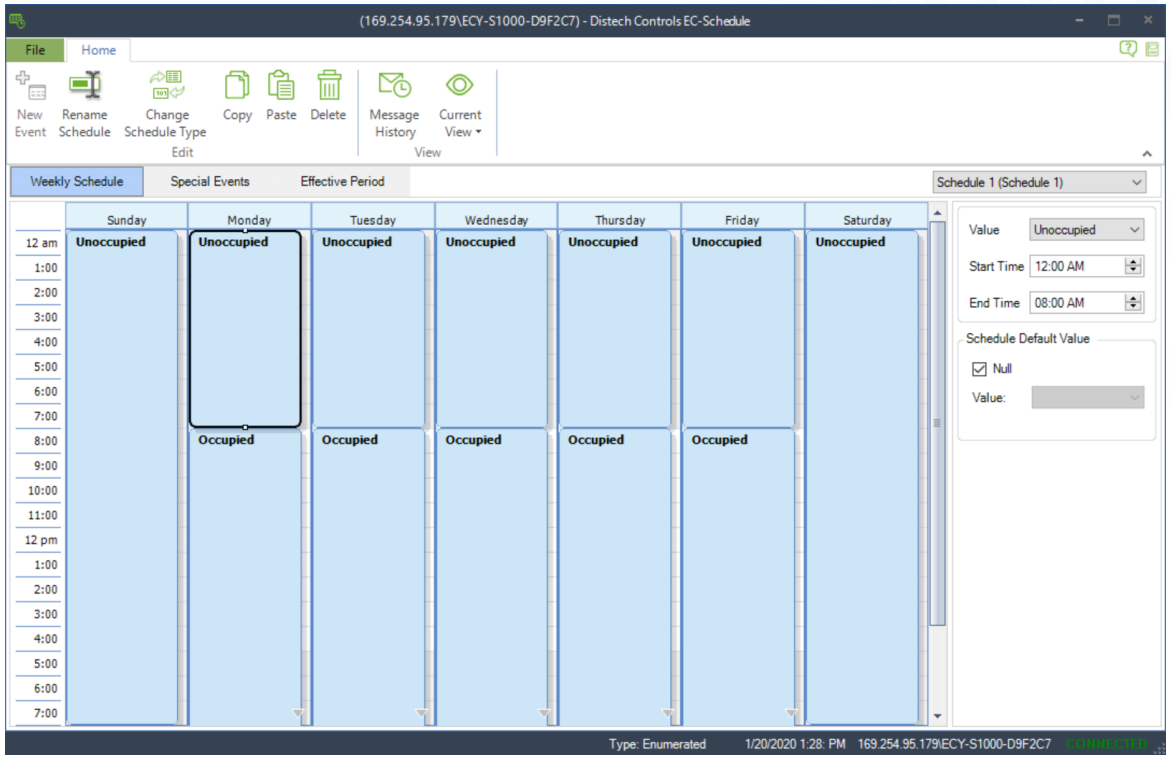
ATRIUS™

ADDITIONAL SCHEDULING CAPABILITIES

To aid in schedule creation, **Schedule blocks** can be used.

Schedules can be created inside of schedule blocks. Using Comparator blocks and Logic blocks, days can be excluded.

The Current State output of a schedule block can be connected with an “Equal” comparator block to set conditions for when areas should be on.



ATRIUS™

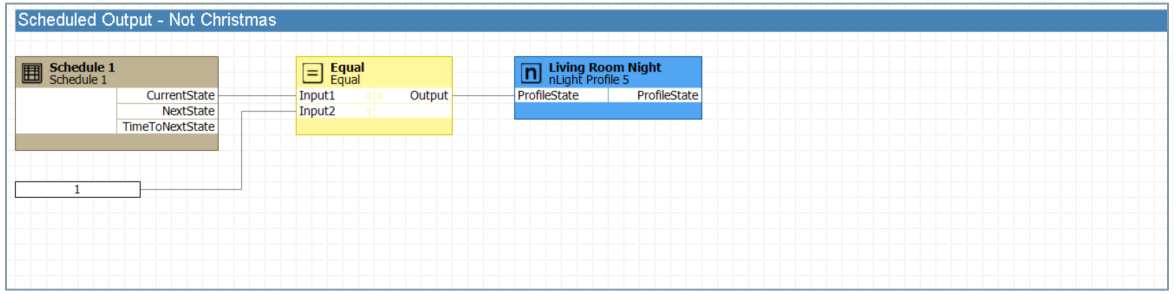
SCHEDULE BLOCK – CONTINUED

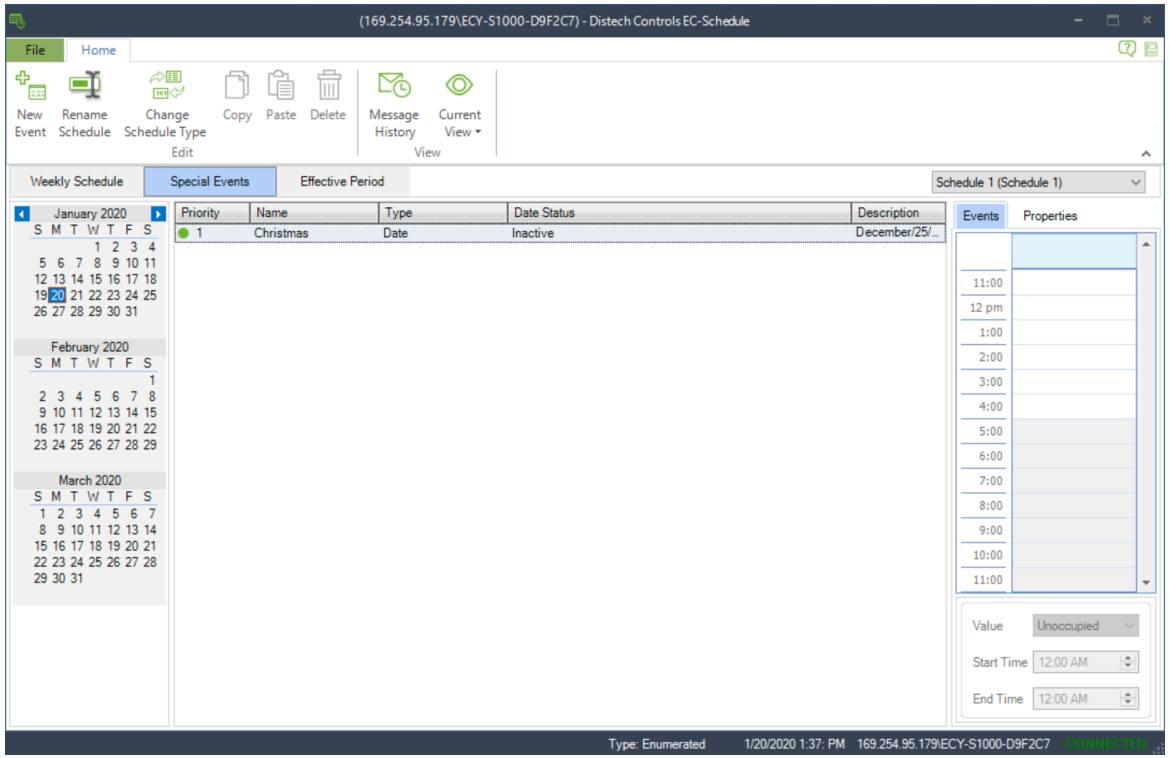
The Current State output of a schedule block can be connected with an “Equal” comparator block to set conditions for when areas should be on.

In the example, we are using a Schedule block to identify when the space is occupied.

Additionally, we have used the “Special Events” to exclude Christmas yearly from the scheduled occupied periods. “Special Events” can be used to exclude any dates or range of dates from a schedule.

Schedules can be created to occur infinitely into the future or set to occur any time before the year 2154.





ATRIUS™

PARTIAL ON

Partial On programs are used to meet code requirements where auto-on is acceptable.

In the example, we are using an occupancy sensor's Motion output port (triggered when motion is seen) to send a command to a conditional block.

The conditional block turns the lights on to a set percentage (50%) only if the relay is off.

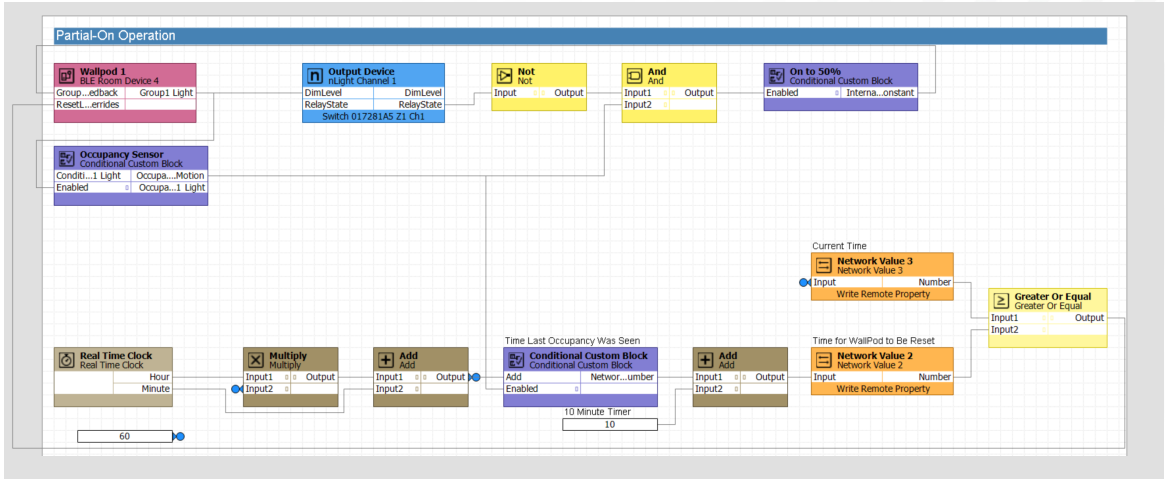
Additionally, we have used the Update port of the motion sensor to make sure that the command is sent only once upon the occ sensor's state updating.

Lights can be overridden by the switch once automatically turned on.

We further make use of a timer so that lights turn off after a configured set of time.

When occupancy is seen, a conditional block is enabled that writes the current time + 10 minutes to Network Value 2. Network Value 2 is then compared to Network Value 3, which represents the current time. If the current time equals or exceeds the time since occupancy was last seen + 10 minutes, the wallpod will be reset, and the lights will go off.

Users can adjust the "10 Minute Timer" field to adjust timeout as desired.



ATRIUS™

SCHEDULED OVERRIDE

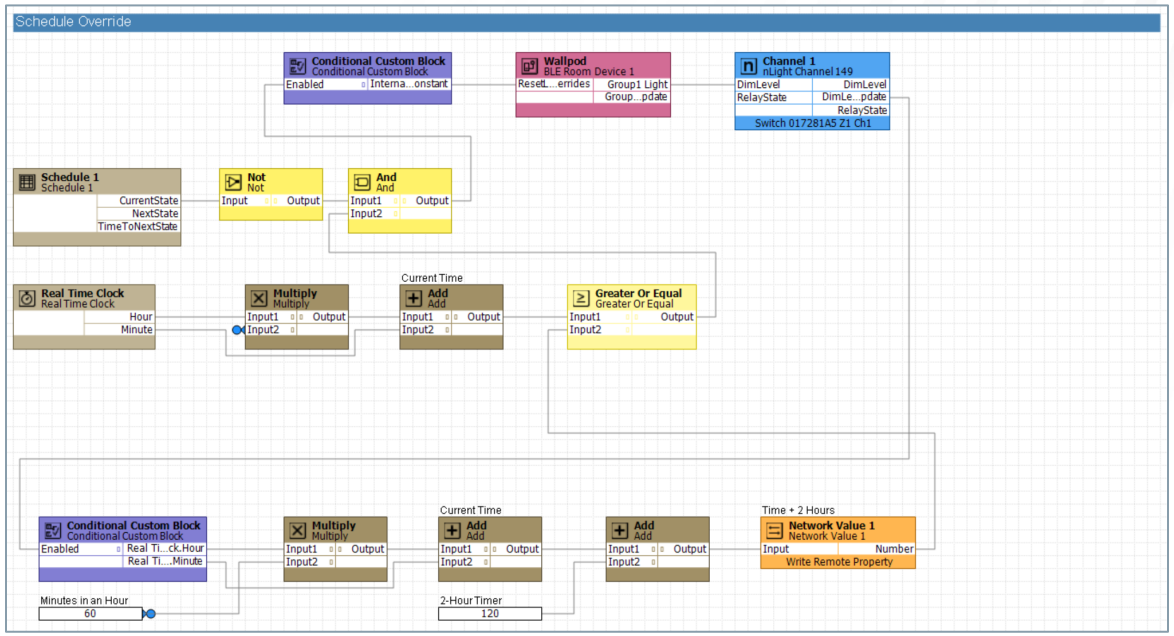
Code often requires that manual control of a space be capable of turning lights on no longer than 2 hours after occupied hours.

When a switch is pressed, the output will change.
If – per the schedule – the area is unoccupied, a snapshot of the time will be taken based on when the switch was pressed.

Two hours (a variable time) will be added to the time and compared to the present time.

When the current time reaches or exceeds the snapshot time + 2 hours, a command will be sent to set the output to 0% or OFF.

Users can adjust the “2-Hour Timer” field to update the timeout as desired.



ATRIUS™

VACANCY OPERATION

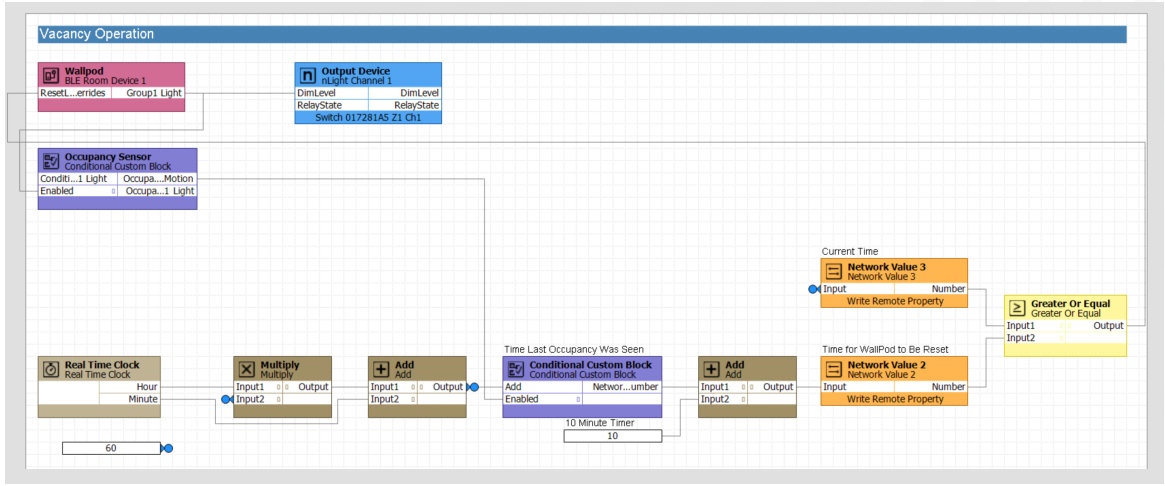
Code often dictates that areas must turn on through manual operation (the push of a button).

In the example shown, we use the output of a Wallpod block to both control the Output Device block via its DimLevel port and to enable the Occupancy Sensor block through the Enabled port.

By enabling the occupancy sensor using the output of the wallpod, the occupancy sensor will never be active before the wallpod is pressed, which accomplishes vacancy mode.

We further make use of a timer so that lights turn off after a configured set of time.

When occupancy is seen, a conditional block is enabled that writes the current time + 10 minutes to Network Value 2. Network Value 2 is then compared to Network Value 3, which represents the current time. If the current time equals or exceeds the time since occupancy was last seen + 10 minutes, the wallpod will be reset, and the lights will go off.



Users can adjust the “10 Minute Timer” field to update the timeout as desired.

ATRIUS™

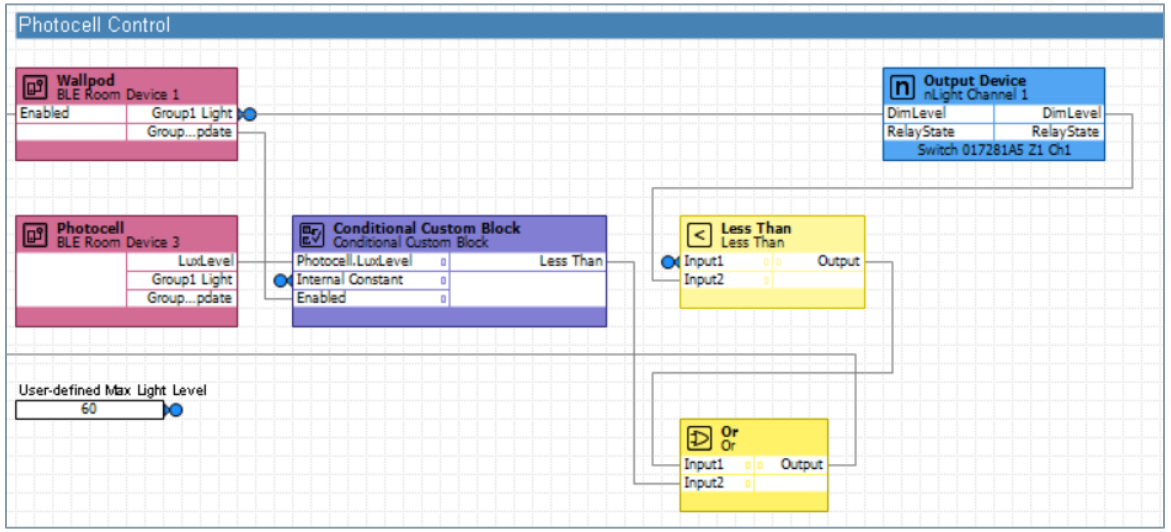
PHOTOCELL OPERATION

Code most often requires that areas that are exposed to natural light have photocell capability to limit users from adjusting light level above user-configurable, photocell-verified light levels.

In the example, we compare the lux level seen from a photocell to a user-defined maximum light level. When the wallpod is pressed, its output is compared to the DimLevel on the Output Device.

Simultaneously, the photocell's sensed lux level is compared to the user-defined max light level. If the output is less than the DimLevel OR the photocell's seen level is still below the user-defined max light level, it is enabled to adjust the output device's light level.

Users can adjust the user-defined max light level to configure how the space performs.



ATRIUS™