
Instrukcja użytkownika

Grenton 2

Wersja dokumentu: 1.0.1

Data: 2019.08.27

Spis

Instrukcja użytkownika

Spis

Ważne informacje

I. Struktura systemu

II. Fundamenty - interfejs logiczny GRENTON

1. Wprowadzenie

2. Cechy

2.1. Cechy wbudowane

2.2. Cechy użytkownika

3. Metody

4. Zdarzenia

5. Adresy cech i metod

III. Przygotowanie projektu

1. Przygotowanie instalacji elektrycznej

A. Topologia instalacji elektrycznej

B. Magistrala

C. Przydatne wskazówki

2. Wybór architektury systemu

A. Konfiguracja podstawowa – scentralizowany System z jednym CLU

B. Konfiguracja rozbudowana – system rozproszony z wieloma CLU i sterowaniem z tabletu

C. Integracja wielu budynków w jeden system

3. Zasilanie modułów

IV. Instalacja komponentów

1. Montaż modułów na rozdzielni

2. Montaż modułów podtynkowych kablowych

3. Montaż modułów podtynkowych Z-Wave

V. Object Manager

1. Instalacja OM

2. Struktura menu

3. Pliki projektów

3.1. Katalog zapisanych projektów

3.2. Kopia zapasowa projektu (backup)

4. Podstawowe elementy

- 4.1. Konfigurator obiektów
- 4.2. Script builder
- 4.3. Diagram połączeń
- 4.4. Visual Builder
- 4.5. myGrenton
- 4.6. Kosz

VI. Podstawowa konfiguracja systemu

1. Połączenie OM z CLU
2. Adresy IP
3. Otwarcie nowego projektu
4. Funkcja CLU DISCOVERY
5. Statusy CLU
6. Obsługa modułów Z-Wave
 - 6.1. Dodawanie modułów Z-Wave
 - 6.2. Usuwanie modułów Z-Wave
 - 6.3. Brak komunikacji z modułem Z-Wave - mechanizm zliczania niepowodzeń komunikacji i blokowania komunikacji urządzenia w sieci Z-Wave
 - 6.4. Wskazówki konfiguracyjne sieci Z-Wave
 - 6.5. Czyszczenie informacji o węzłach
7. Wysyłanie konfiguracji do CLU
8. Wartości początkowe cech
9. Tworzenie podstawowych powiązań
10. Przeprowadzanie aktualizacji
 - 10.1. Proces aktualizacji bazy interfejsów
 - 10.2. Proces aktualizacji firmware'u CLU
 - 10.3. Proces aktualizacji firmware'u modułów z serii 2.0
11. Inne operacje na systemie

VII. Zaawansowane funkcje konfiguracyjne

1. Kontenery
2. Skrypty
 - A. Tworzenie skryptów w trybie graficznym
 - B. Tworzenie skryptów w trybie tekstowym
 - C. Przekazywanie parametrów do skryptu
 - D. Wywoływanie skryptów
3. Data i czas

VIII. Visual Builder

1. Sterowanie systemem z poziomu smartfonów
2. Struktura interfejsu
3. Aplikacja na smartfony – GRENTON HOME MANAGER
4. Tworzenie nowego interfejsu
 - 4.1. Wybór skórki graficznej
 - 4.2. Tworzenie stron interfejsu
 - 4.3. Komponenty
 - 4.4. Panele
 - 4.5. Kontenery
 - 4.6. Wstawianie komponentów i tworzenie powiązań z obiektami systemu
 - 4.7. Wysyłanie interfejsu do urządzenia mobilnego
5. Automatyczne tworzenie interfejsu - generator GUI
 - 5.1. Tworzenie interfejsu z dostępną rozdzielczością
 - A. Konfigurator prosty
 - B. Konfigurator zaawansowany
 - 5.2. Tworzenie interfejsu z własną rozdzielczością
 - 5.3. Zmiana orientacji interfejsu z własną rozdzielczością
6. Konfiguracja wideodomofonu
 - 6.1. Podłączenie i konfiguracja wideodomofonu
 - A. Podłączenie wideodomofonu
 - B. Konfiguracja kamery
 - C. Konfiguracja SIP:

- 6.2. Stworzenie i konfiguracja interfejsu aplikacji
 - A. Dodawanie wiadomości do interfejsu aplikacji w programie Object Manager
 - B. Konfiguracja aplikacji Home Manager
- 6.3. Wykonanie połączenia z domofonu
- 7. Obsługa obrazu z kamer
 - A. Dodawanie komponentu *Kamera*
 - B. Dodanie panelu *Kamera*
- 8. Zdalny dostęp aplikacji mobilnej do systemu
 - 8.1. Konfiguracja systemu
 - 8.2. Ustawienie przekierowania portów w routerze sieci lokalnej
 - 8.3. Konfiguracja aplikacji mobilnej Home Manager
 - 9.4. Uruchomienie zdalnego dostępu
- IX. Obiekty CLU
 - 1. Timery
 - A. Tworzenie timerów
 - B. Parametry konfiguracyjne obiektu Timer
 - 2. Kalendarz
 - A. Tworzenie kalendarza
 - B. Właściwości kalendarza
 - C. Reguły kalendarza
 - D. Tworzenie reguły kalendarza za pomocą interfejsu graficznego
 - E. Tworzenie reguły kalendarza zgodnie z formatem CRON
 - F. Parametry konfiguracyjne obiektu Kalendarz
 - 3. Harmonogram
 - A. Tworzenie harmonogramu
 - B. Formułowanie wartości dla harmonogramu
 - C. Ustawianie wartości wyjścia za pomocą harmonogramu
 - D. Parametry konfiguracyjne obiektu Harmonogram
 - 4. Regulator PID
 - A. Tworzenie regulatora
 - B. Sterowanie za pomocą regulatora
 - C. Tryby pracy
 - D. Zasada działania regulatora PID
 - E. Parametry konfiguracyjne obiektu Regulator PID
 - 5. Termostat
 - A. Tworzenie termostatu
 - B. Formułowanie wartości dla termostatu
 - C. Parametry konfiguracyjne obiektu Termostat
- X. Pomiar mediów
 - 1. Uruchomienie pomiaru mediów po stronie Object Managera
 - A. Stworzenie konfiguracji
 - B. Odczyt pomiaru mediów w Object Managerze
 - C. Skonfigurowanie pomiaru mediów dla interfejsu aplikacji
 - 2. Używanie pomiaru mediów po stronie aplikacji Home Manager
 - A. Pobieranie pomiarów:
 - B. Opcje widoku panelu pomiaru mediów:
 - C. Synchronizacja oraz pobieranie pomiarów:
- XI. Funkcje serwisowe CLU
 - 1. Przywracanie ustawień fabrycznych CLU - *Hard Reset*
 - 2. Autodiagnostyka systemu - *Debugging CLU*
- XII. SMART PANEL
 - 1. Wyposażenie Smart Panelu
 - 2. Podłączenie Smart Panelu do CLU
 - 3. Informacje pomocne przy tworzeniu konfiguracji
 - 4. Konfiguracja modułu Smart Panel w wersji v3
 - 4.1. Parametry konfiguracyjne
 - A. Panel
 - B. Przyciski

- C. Czujniki temperatury i oświetlenia
- 4.2. Tworzenie konfiguracji przycisków i wyświetlacza
- 4.3. Tworzenie konfiguracji sensora gestów
- 4.4. Konfiguracja czujnika zbliżenia
- 4.5. Tworzenie konfiguracji wielu stron panelu dotykowego
- 5. Konfiguracja modułu Smart Panel w wersji v4
 - 5.1. Parametry konfiguracyjne
 - A. Panel
 - B. Przyciski
 - C. Konfiguracja stron (Panel_Page)
 - D. Czujniki temperatury i oświetlenia
 - 5.2. Tworzenie konfiguracji sensora gestów
 - 5.3. Konfiguracja czujnika zbliżenia
 - 5.4. Obiekt Panel - nowa funkcjonalność
 - 5.5. Obiekt Panel - mechanizm zarządzania stronami
 - 5.6. Kompatybilność wstecz
 - 5.7. Tworzenie konfiguracji z wykorzystaniem obiektu strony Buttons
 - 5.8. Tworzenie konfiguracji z wykorzystaniem obiektu strony FreeDraw
 - A. Ogólne zasady tworzenia konfiguracji
 - B. Konfiguracja strony jako zegarka
 - 5.9. Tworzenie konfiguracji z wykorzystaniem obiektu strony Thermostats
 - A. Tworzenie konfiguracji z termostatem lokalnym
 - B. Tworzenie konfiguracji z termostatem zdalnym
 - C. Predefiniowane zachowanie przycisków
- XIII. Moduł GATE Alarm
 - 1. Integracja z centralą Satel
 - 1.1. Informacje ogólne
 - 1.2. Konfiguracja modułu GATE Alarm
 - 1.3. Obiekty wirtualne
 - A. Satel
 - B. Strefa
 - C. Wyjście
 - D. Wejście
 - 2. Przywracanie ustawień fabrycznych - *Hard Reset*
 - 3. Parametry konfiguracyjne
- XIV. Moduł GATE Modbus
 - 1. Informacje ogólne
 - 2. Konfiguracja modułu GATE Modbus
 - 3. Parametry rejestrów
 - A. Rejestry 16-bitowe
 - B. Pola w rejestrach 16-bitowych
 - C. 32-bitowe wartości całkowite rejestrów
 - D. 32-bitowe wartości zmiennoprzecinkowe rejestrów
 - E. Dyskretne wejścia/wyjścia
 - 4. Przywracanie ustawień fabrycznych - *Hard Reset*
 - 5. Parametry konfiguracyjne
- XV. Moduł GATE HTTP
 - 1. Informacje ogólne
 - 2. Konfiguracja modułu GATE HTTP
 - 2.1 Obiekty wirtualne
 - 2.1.1. HTTPRequest
 - 2.1.2. Pobieranie określonych wartości z otrzymanej odpowiedzi (XML,JSON)
 - 2.2.1. HttpListener
 - 2.2.2. Przygotowanie odpowiedzi wysyłanej do serwera
 - 2.2.3. Odczyt wartości kluczy z parametru querystringparams
 - 3. Możliwość połączenia z Gate za pomocą TELNET
 - 4. Przywracanie ustawień fabrycznych - *Hard Reset*
 - 5. Parametry konfiguracyjne

- A. Obiekt GATE
- B. Obiekt HttpRequest
- C. Obiekt HttpListener

XVI. Moduły Z-Wave

1. Fibaro UBS
 - 1.1. Informacje ogólne
 - 1.2. Obiekty
 - A. ZWAVE_DIN
 - B. ZWAVE_1W_SENSOR
 - C. ZWAVE_CONFIG
3. NEO Coolcam Motion Sensor (PIR)
 - 3.1. Informacje ogólne
 - 3.2. Obiekty
 - A. BINARY_SENSOR
 - B. ANALOG_SENSOR
 - C. ZWAVE_BATTERY
 - D. ZWAVE_WAKEUP
 - E. ZWAVE_CONFIG
4. NEO Coolcam Door / Window Sensor
 - 4.1. Informacje ogólne
 - 4.2. Obiekty
 - A. BINARY_SENSOR
 - B. ZWAVE_BATTERY
 - C. ZWAVE_WAKEUP
 - D. ZWAVE_CONFIG
5. INFIBITY Motion Sensor (PIR) [NEO Coolcam]
 - 5.1. Informacje ogólne
 - 5.2. Obiekty
 - A. BINARY_SENSOR
 - B. ANALOG_SENSOR
 - C. ZWAVE_BATTERY
 - D. ZWAVE_WAKEUP
 - E. ZWAVE_CONFIG
6. INFIBITY Door/Window Sensor [NEO Coolcam]
 - 6.1. Informacje ogólne
 - 6.2. Obiekty
 - A. BINARY_SENSOR
 - B. ZWAVE_BATTERY
 - C. ZWAVE_WAKEUP
 - D. ZWAVE_CONFIG
7. INFIBITY Water Sensor [NEO Coolcam]
 - 7.1. Informacje ogólne
 - 7.2. Obiekty
 - A. BINARY_SENSOR
 - B. ZWAVE_BATTERY
 - C. ZWAVE_WAKEUP
 - D. ZWAVE_CONFIG
8. Heiman Smart Smoke Sensor
 - 8.1. Informacje ogólne
 - 8.2. Obiekty
 - A. BINARY_SENSOR
 - B. ZWAVE_BATTERY
 - C. ZWAVE_WAKEUP
 - D. ZWAVE_CONFIG
9. INFIBITY Siren Alarm [NEO Coolcam]
 - 9.1. Informacje ogólne
 - 9.2. Obiekty
 - A. ZWAVE_DOUT

- B. ZWAVE_BATTERY
- C. ZWAVE_WAKEUP
- D. ZWAVE_CONFIG
- 10. Danfoss Living Connect
 - 10.1. Informacje ogólne
 - 10.2. Obiekty
 - A. ZWAVE_THERMOSTAT
 - B. ZWAVE_BATTERY
 - D. ZWAVE_CONFIG
- 11. POPP Z-Weather
 - 11.1. Informacje ogólne
 - 11.2. Obiekty
 - A. ZWAVE_WEATHER
 - B. ZWAVE_BATTERY
 - C. ZWAVE_WAKEUP
 - D. ZWAVE_CONFIG
- 12. FAKRO AMZ Solar
 - 12.1. Informacje ogólne
 - 12.2. Obiekty
 - ZWAVE_FAKRO
 - ZWAVE_CONFIG
- 13. FAKRO ARF
 - 13.1. Informacje ogólne
 - 13.2. Obiekty
 - A. ZWAVE_FAKRO
 - B. ZWAVE_CONFIG
- 14. FAKRO FTP_V
 - 14.1. Informacje ogólne
 - 14.2. Obiekty
 - A. ZWAVE_FAKRO
 - B. ZWAVE_CONFIG
- XVII. Aplikacja mobilna myGrenton
 - 1. Instalacja myGrenton
 - 2. Tworzenie interfejsu
 - 2.1. Dodawanie strony do interfejsu
 - 2.2. Usuwanie strony z interfejsu
 - 3. Widżety
 - 3.1. Nagłówek (HEADER)
 - 3.2. Wartość (VALUE)
 - 3.3. Włącz/Wyłącz (ON_OFF)
 - 3.4. Scena (SCENE)
 - 3.5. Otwórz/Zamknij (SCENE_DOUBLE)
 - 3.6. Ściemniacz (DIMMER)
 - 3.7. Oświetlenie LED (LED)
 - 3.8. Termostat (THERMOSTAT)
 - 3.9. Rolety (ROLLER_SHUTTER)
 - 3.10. Usuwanie widżetu
 - 4. Personalizacja interfejsu
 - 5. Wysłanie interfejsu na urządzenie
 - 5.1. Wysłanie na telefon interfejs myGrenton
 - 5.2. Udostępnienie w chmurze interfejsu myGrenton a następnie wysłanie na telefon
 - 6. Ustawienia aplikacji oraz interfejsu
 - 6.1. Ustawienia aplikacji
 - 6.2. Ustawienia interfejsu
- XVIII. Safe Mode

Ważne informacje

UWAGA!

Niniejsza dokumentacja obejmuje funkcjonalności oraz zasady działania dla modułów Grenton z serii 2.0. Zachowany zostaje opis funkcjonalny dla Object Managera a także Home Managera. Aplikacja myGrenton jest całkowicie kompatybilna z systemami Grenton 2.0 - w przypadku systemów 1.0 dostęp do poszczególnych funkcji może być ograniczony lub całkowicie niedostępny.

I. Struktura systemu

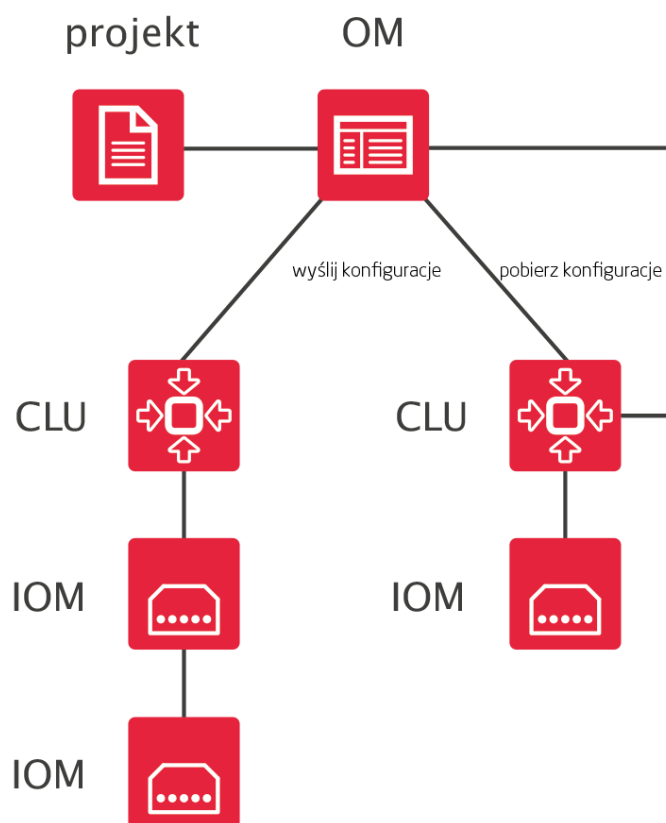
System Inteligentnych budynków GRENTON został zaprojektowany z myślą o obsłudze małych, średnich i dużych obiektów. Instalacje stworzone na bazie systemu można w łatwy sposób modyfikować, rozbudowywać oraz integrować z innymi systemami.

W skład systemu wchodzi: moduły CLU, moduły IOM, Object Manager, sensory oraz aplikacje na smartfony:

- Moduły CLU (*Common Logic Unit*). Realizują funkcję przetwarzania logiki oraz przechowywania konfiguracji. CLU stanowią podstawę każdego systemu. CLU komunikują się ze sobą za pomocą magistrali systemowej, która działa w oparciu o standard Ethernet 100 Mbps. Moduły CLU zapewniają również komunikację z modułami IOM przy pomocy magistrali lokalnej.
- Moduły IOM są modułami realizującymi funkcje wejścia/wyjścia, które podłączane są do CLU poprzez magistralę lokalną TF-Bus lub bezprzewodowo z wykorzystaniem standardu Z-Wave. Moduły IOM mogą zawierać różnego typu wejścia/wyjścia (np. przekaźniki, włączniki, sensory światła, temperatury) oraz ich kombinacje.
- Object Manager – jest oprogramowaniem umożliwiającym konfigurację systemu, funkcji logicznych itd.
- Aplikacje sterujące – pozwalają na uruchamianie zaprojektowanych w OM graficznych interfejsów użytkownika, które dają możliwość sterowania funkcjami systemu przy użyciu smartfonów, tabletów, komputerów PC, TV itp.

Konfiguracja systemu przechowywana jest w postaci pliku projektu i ustawiana przy pomocy programu Object Manager (OM). Utworzona konfiguracja jest następnie przesyłana do modułów CLU, które przechowują ją w swojej pamięci. Moduły IOM nie przechowują konfiguracji i są sterowane bezpośrednio z CLU, do którego zostały podłączone.

W razie utraty pliku projektu OM posiada funkcję ściągnięcia danych z CLU. Pobranie danych z CLU wiąże się jednak z utratą: widoku graficznego utworzonych skryptów, kontenerów, interfejsów mobilnych oraz typów obiektów (źródło/odbiornik).



II. Fundamenty - interfejs logiczny GRENTON

1. Wprowadzenie

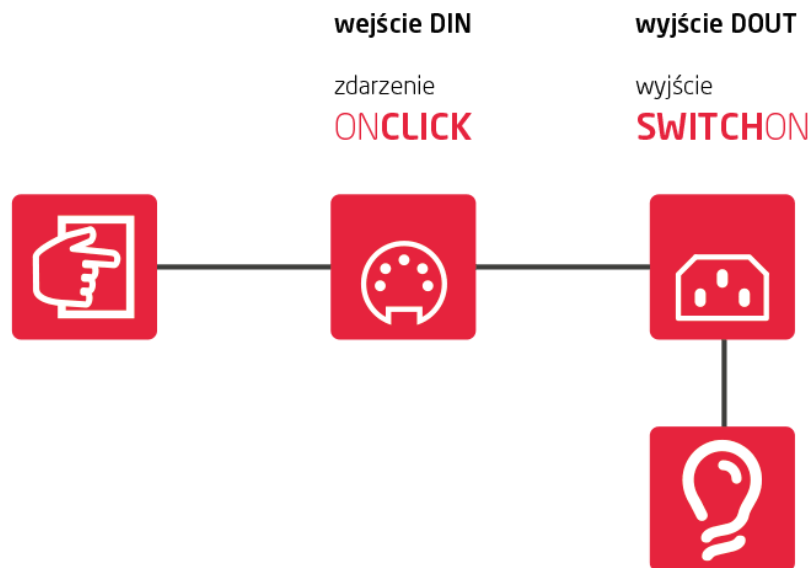
System GRENTON działa w oparciu o tzw. model wywoływany zdarzeniami (event driven model). Domownicy i otoczenie powodują generowanie w systemie zdarzeń, z którymi powiązane są reakcje systemu (np. w odpowiedzi na naciśnięcie wyłącznika system włącza lampę).

Podstawą interfejsu logicznego są obiekty. W systemie GRENTON, każdy obiekt jest traktowany oraz zachowuje się w taki sposób jak każdy fizyczny przedmiot, np. piłka. Każdy obiekt posiada swoje cechy, możemy wykonać na nim pewne czynności, może również wywoływać jakieś zdarzenia. Odnosząc to do naszej piłki: możemy nią kopnąć, czyli sterować, tocząca się piłka może przewrócić stojącą butelkę, a więc wywoła zdarzenie i na koniec możemy powiedzieć że jest czerwona, czyli posiada pewne swoje cechy.

W systemie każde wejście lub wyjście posiada własny zbiór cech, metod i zdarzeń, które nazywane są jego interfejsem logicznym.

Unikalnym rozwiązaniem systemu GRENTON jest to, iż każda cecha lub metoda są dostępne w dowolnym miejscu systemu, na każdym CLU, niezależnie od tego, na którym CLU, wejściu lub wyjściu fizycznie się one znajdują. Możliwe jest więc wywołanie metody z wyjścia podłączonego do CLU A w następstwie zdarzenia, które zaszło w obrębie CLU B.

Również wyjścia posiadają specyficzne dla siebie zdarzenia, co umożliwia np. zaświecenie jednej lampy w skutek włączenia innej. Pełną listę zdarzeń metod i cech każdego wejścia/wyjścia można znaleźć w karcie katalogowej modułu.



2. Cechy

2.1. Cechy wbudowane

Cechy wbudowane to zbiór parametrów/informacji opisujących dany obiekt (wejście, wyjście itd.). Niektóre z cech mogą być ustawiane w trakcie działania systemu i służą do określenia sposobu działania obiektu (tryb pracy przycisku). Inne natomiast mogą być jedynie odczytywane, gdyż np. pokazują fizyczne wartości (np. cecha `value` dla czujnika temperatury).

2.2. Cechy użytkownika

W CLU można zdefiniować cechy, które mogą być następnie użyte jako zmienne do przechowywania parametrów podczas działania systemu (np. liczniki, znaczniki). Cechy użytkownika mogą być używane w identyczny sposób jak cechy wbudowane, z tą różnicą, że wszystkie cechy użytkownika mogą być zarówno zapisywane do pamięci ulotnej, jak i z niej odczytywane.

3. Metody

Metody są rozkazami, które mogą zostać wydane danemu obiektowi. Każdy obiekt posiada charakterystyczne dla siebie metody. Dla wyjścia przekaźnikowego mogą być to metody `switchon` (załącz) lub `switchoff` (wyłącz). Dodatkowo metody mogą zawierać parametry wymagane lub opcjonalne, które precyzują sposób ich wywołania (np. czas włączenia).

4. Zdarzenia

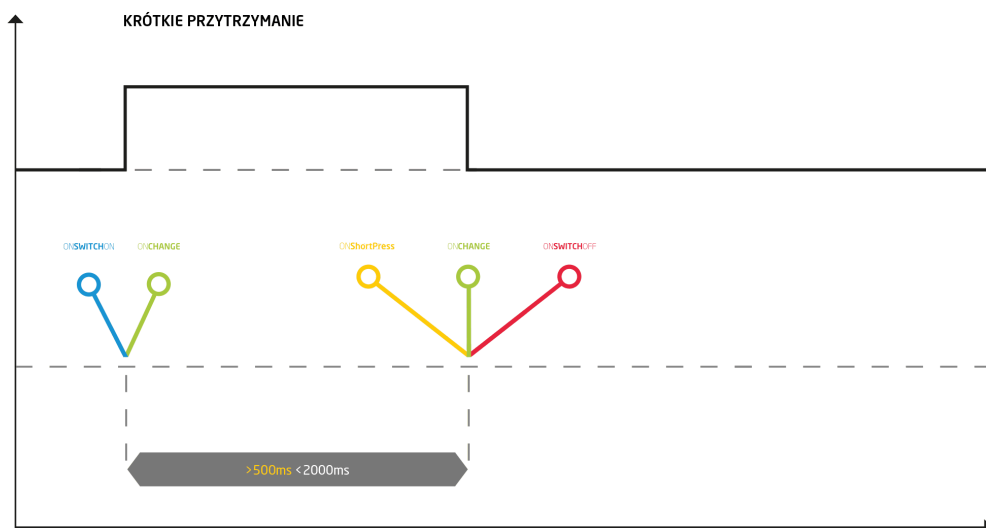
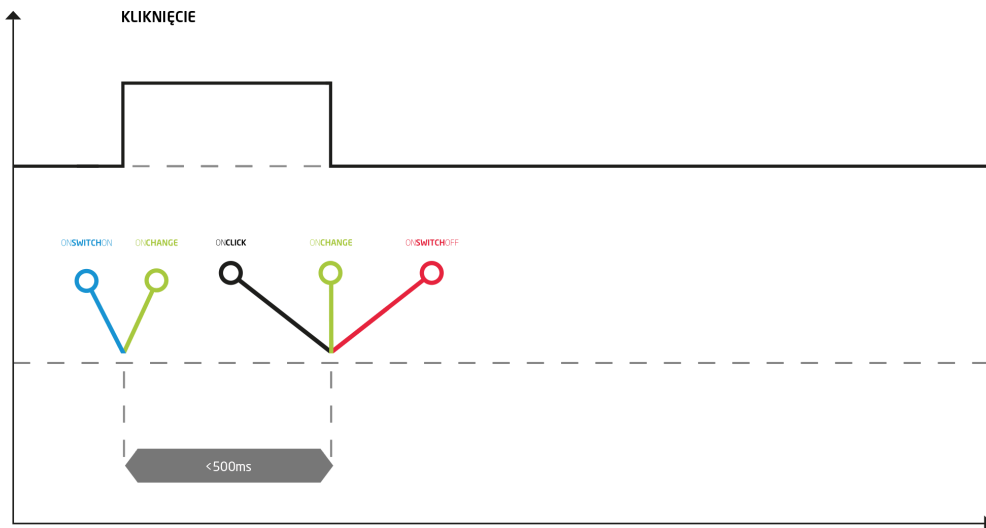
Zdarzenia są elementami interfejsu logicznego wywoływanymi w reakcji na zmiany zachodzące w związku z obiektem (np. włączenie przycisku, zmiana temperatury itp.). Z każdym zdarzeniem możemy powiązać nawet kilka metod, które zostaną wykonane w momencie zaistnienia zdarzenia, np. w momencie naciśnięcia przycisku zostaną włączone lampy. Wiążąc zdarzenia

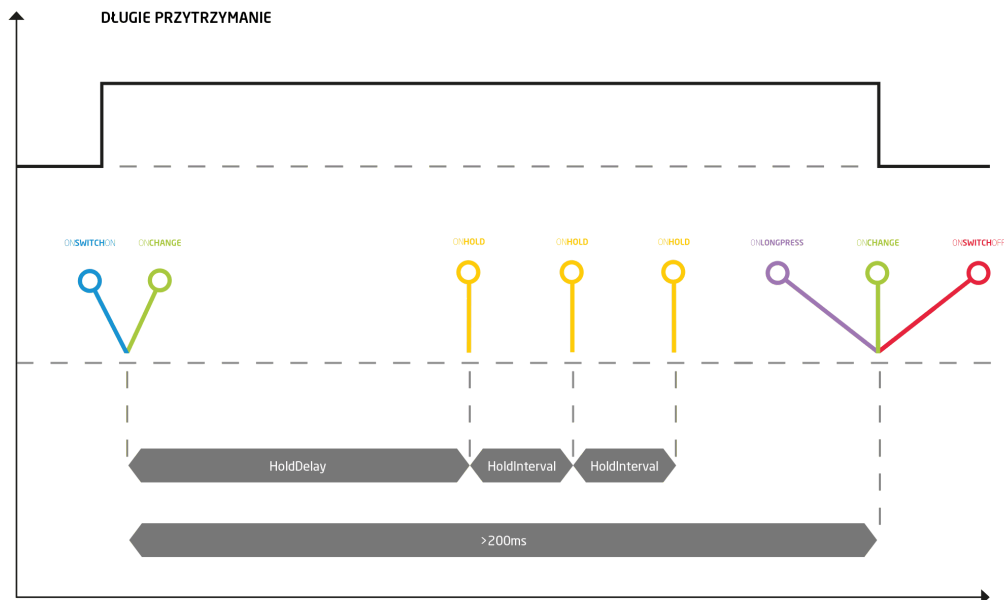
jednych obiektów (głównie z wejść, ale czasem również wyjść) z metodami innych obiektów, tworzymy konfigurację logiczną systemu.

Każdy typ obiektu (rodzaj wejścia/wyjścia) posiada własną listę zdarzeń, które są wywoływane w ściśle określony sposób, w zależności od akcji podejmowanych przez użytkownika. Przykładowo wejście binarne posiada następująca listę zdarzeń:

- `OnChange`
- `OnSwitchOn`
- `OnSwitchOff`
- `OnShortPress`
- `OnLongPress`
- `OnClick`
- `OnHold`

które wywoływane są zgodnie z poniższymi schematami:





5. Adresy cech i metod

Każda cecha i metoda posiada w systemie adres dzięki któremu można odwoływać się do niej w skryptach oraz podczas powiązywania ze zdarzeniami. Adres składa się z 3 części połączonych znakami `->`:

- identyfikator CLU lub kontenera;
- nazwa obiektu (wejścia, wyjścia, CLU);
- nazwa cechy/metody wraz z parametrami (o ile występują).

Przykładowo: `CLU1->Lampa1->switchon()` – metoda powodująca włączenie wyjścia `Lampa1`

`oświetlenie->Lampa1->value()` – cecha pokazująca, czy lampa jest włączona czy wyłączona, dla lampy znajdującej się w kontenerze `Oświetlenie`.

III. Przygotowanie projektu

1. Przygotowanie instalacji elektrycznej

UWAGA! Instalacje elektryczne w domach mieszkalnych oraz budynkach użyteczności publicznej mogą być wykonywane tylko zgodnie z obowiązującymi przepisami i normami elektrycznymi oraz tylko przez wykwalifikowanych specjalistów posiadających odpowiednią wiedzę oraz wymagane uprawnienia wynikające z przepisów prawa obowiązującego w danym kraju.

A. Topologia instalacji elektrycznej

System GRENTON umożliwia tworzenie instalacji zarówno scentralizowanych jak i rozproszonych. Dla nowo projektowanych budynków sugerujemy sprowadzenie wszystkich obwodów do jednej rozdzielni elektrycznej, co pozwoli na bardziej elastyczne podejście w projektowaniu instalacji i bardziej ekonomiczne zarządzanie zasobami.

Do każdego urządzenia, które ma zostać połączone z systemem, powinien zostać wydzielony osobny obwód elektryczny zakończony w rozdzielni elektrycznej. Dobór przekroju przewodów powinien zostać przeprowadzony w oparciu o obowiązujące normy. Jeśli nie ma możliwości ułożenia przewodu bezpośrednio od rozdzielni do sterowanego urządzenia, możliwe są trzy ścieżki:

1. Wyniesienie modułu CLU wraz z modułami IOM. W takim przypadku moduły CLU należy połączyć w rozdzielni z modułem wyniesionym za pomocą magistrali systemowej – takie rozwiązanie ma sens, gdy w ramach jednego systemu integrujemy co najmniej dwa budynki.
2. Wyniesienie jednego lub więcej modułów IOM. Łączenie modułów odbywa się przy pomocy magistrali lokalnej – rozwiązanie polecane, gdy wynoszona jest niewielka liczba modułów.
3. Wykorzystanie radiowych modułów IOM działających w oparciu o Z-Wave. Rozwiązanie sugerowane w sytuacji, gdy nie ma możliwości wykonania instalacji kablowej (istniejące budynki itp.).

B. Magistrala

W systemie znajdują się 2 magistrale :

1. **Systemowa**, służąca do połączenia pomiędzy modułami CLU-CLU oraz CLU-SMARTFONY itp. Magistrala systemowa – Ethernet. Moduły mogą być ze sobą łączone szeregowo. Maksymalna długość kabla pomiędzy dwoma modułami CLU wynosi 90 m. Zalecany kabel UTP (minimum kat. 5e). Długość magistrali systemowej może zostać zwiększona przez wykorzystanie urządzeń sieciowych - np. switch, router, itp.
2. **Lokalna**, służąca do połączenia pomiędzy modułami CLU-IOM. Magistrala lokalna – moduły IOM mogą być łączone ze sobą szeregowo, a także dołączane do magistrali przy pomocy odczepów. Maksymalna długość magistrali pomiędzy dwoma jej końcami wynosi 300 m.

UWAGA! Może być konieczne dodatkowe zasilanie magistrali.

Zalecany kabel o stałej impedancji falowej i przekroju minimalnym 0,5 mm², np. kabel UTP (opcjonalnie ekranowane kable: FTP lub E-BUS). Przy większej ilości modułów lub bardziej rozbudowanej magistrali należy uwzględnić spadki napięć przy doborze przekroju kabla magistralnego.

C. Przydatne wskazówki

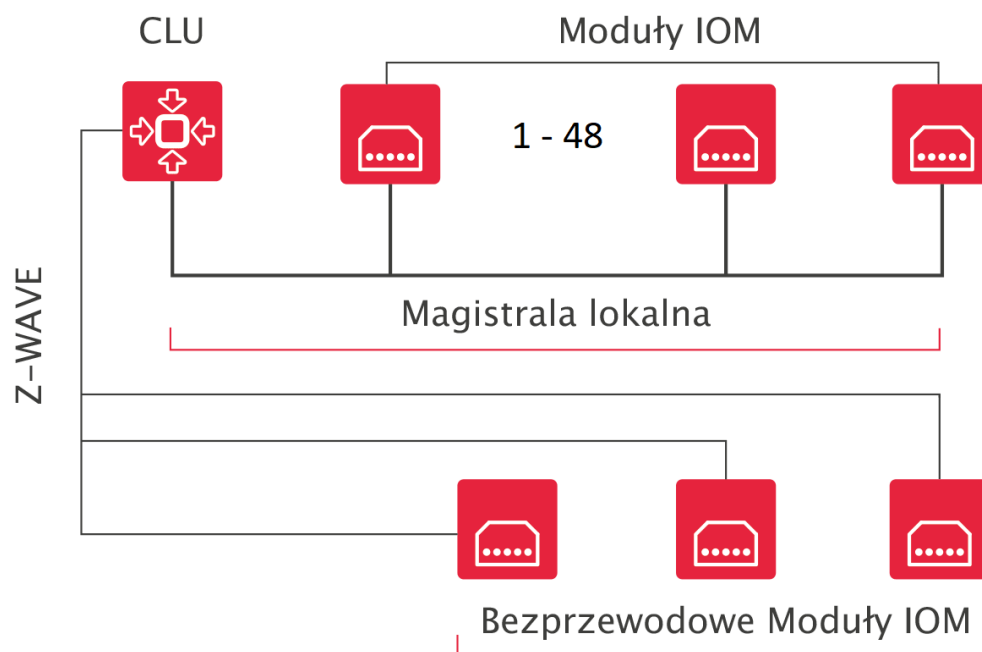
- Przed rozpoczęciem wykonania instalacji elektrycznej należy przygotować projekt instalacji systemu inteligentnego.
- Jeśli jeszcze nie wiadomo, którymi urządzeniami będzie odbywało się sterowanie, zaleca się dociągnięcie okablowania do wszystkich możliwych miejsc.
- Do włączników oświetlenia można stosować dowolny cienki kabel, np. YTDY – pozwoli to zmniejszyć koszty instalacji.
- Należy pamiętać o przygotowaniu instalacji pod czujniki temperatury i stację pogodową.
- Umieszczenie gniazda zasilania na tarasie i poprowadzenie do niego osobnego zasilania da możliwość sterowania zasilaniem w tym gnieździe z systemu.

2. Wybór architektury systemu

W zależności od rodzaju obiektów, wymagań oraz wielkości można stosować różne konfiguracje – system jest w pełni skalowalny. W zależności od skali i potrzeb możliwych jest kilka konfiguracji:

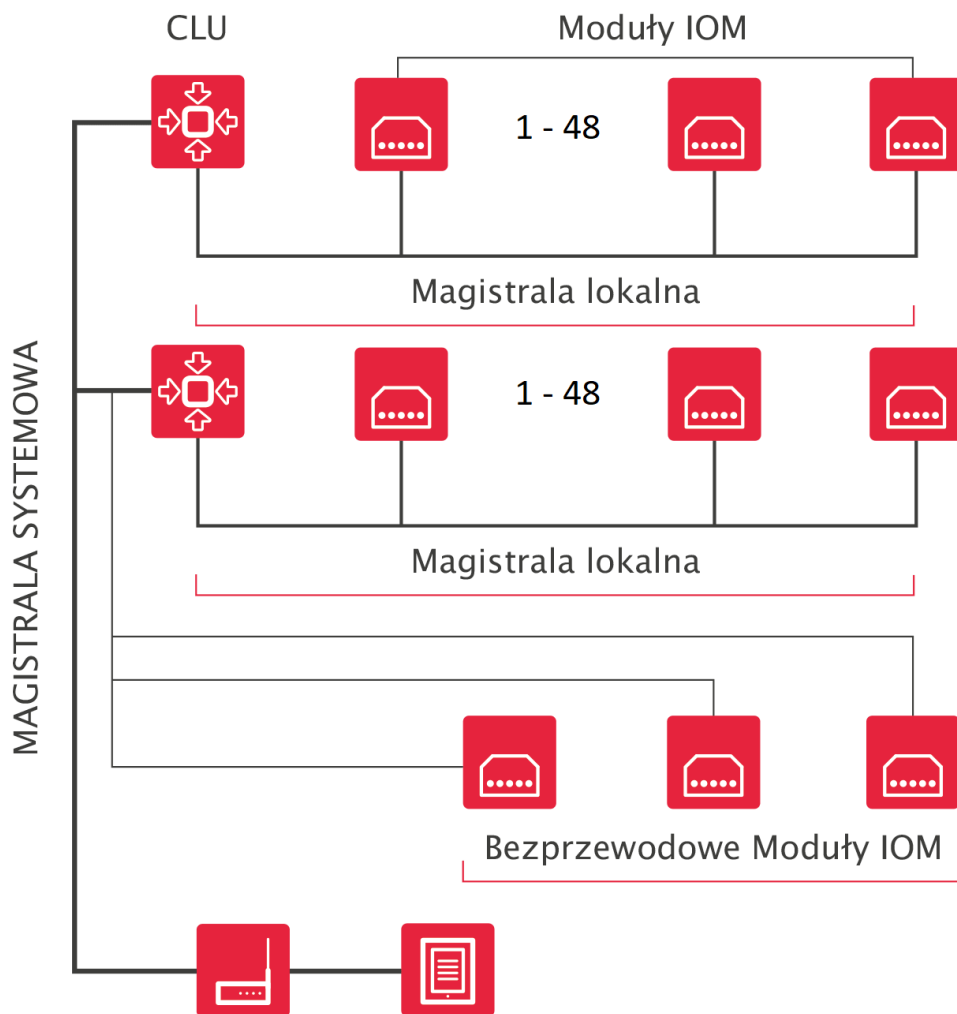
A. Konfiguracja podstawowa – scentralizowany System z jednym CLU

Na schemacie został przedstawiony system zbudowany w oparciu o jedno CLU. W systemie skonfigurowanym w taki sposób maksymalnie może znajdować się do 48 modułów IOM bez względu na ich rodzaj (lub do 128 wejść/wyjść). Należy pamiętać o zapewnieniu magistrali zasilania odpowiedniego do obciążenia.



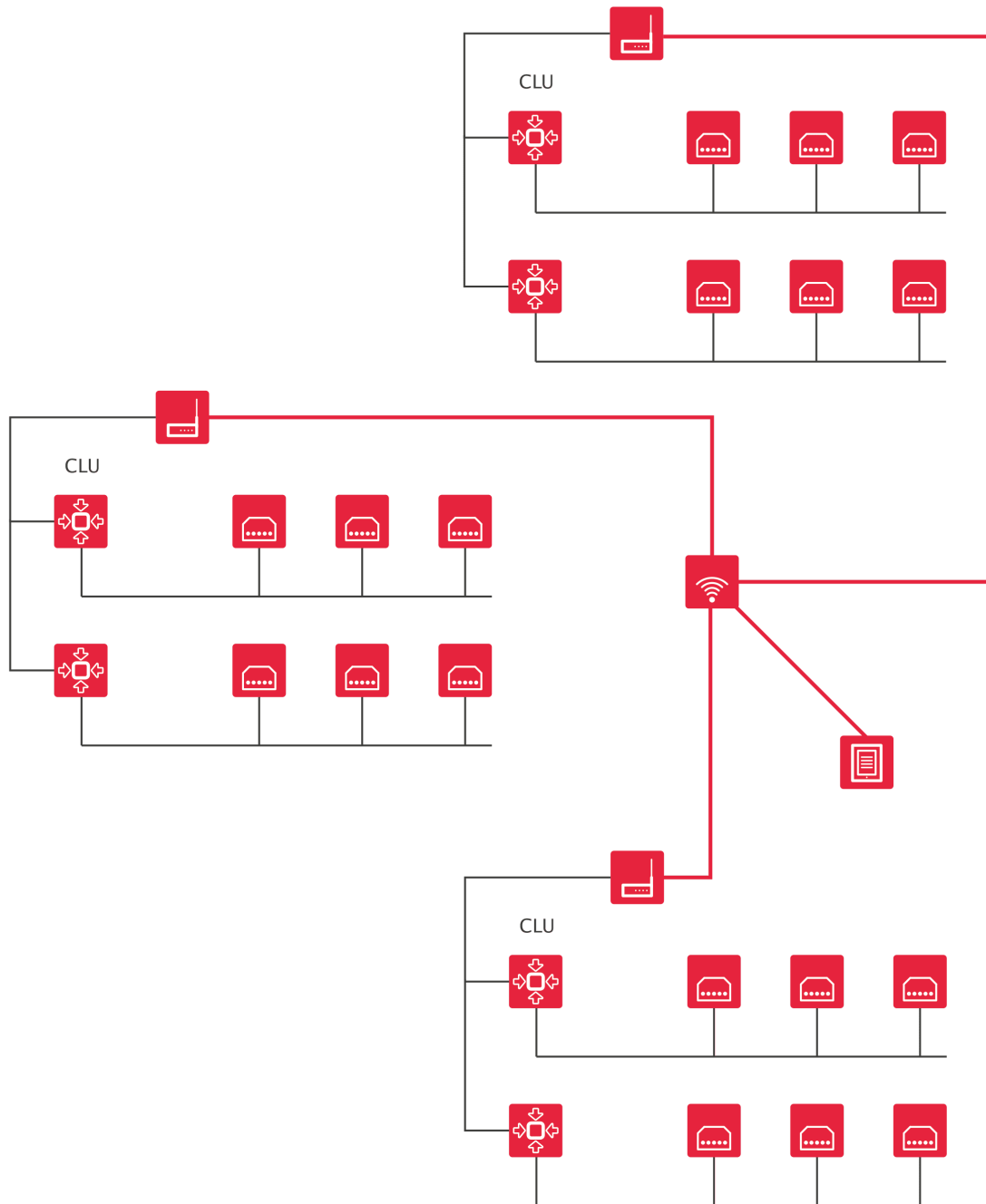
B. Konfiguracja rozbudowana – system rozproszony z wieloma CLU i sterowaniem z tabletu

Pojemność systemu można zwiększyć poprzez dołączanie kolejnych modułów CLU. Jednostki CLU łączymy pomiędzy sobą za pomocą magistrali systemowej. Instalacja dodatkowo może być rozszerzona o smartfony, tablety itp.



C. Integracja wielu budynków w jeden system

System może być rozbudowywany praktycznie w nieograniczony sposób. Do jednego systemu może zostać wpięte kilka obiektów. Dzięki czemu będziemy mieć możliwość centralnego sterowania za pomocą jednego systemu.

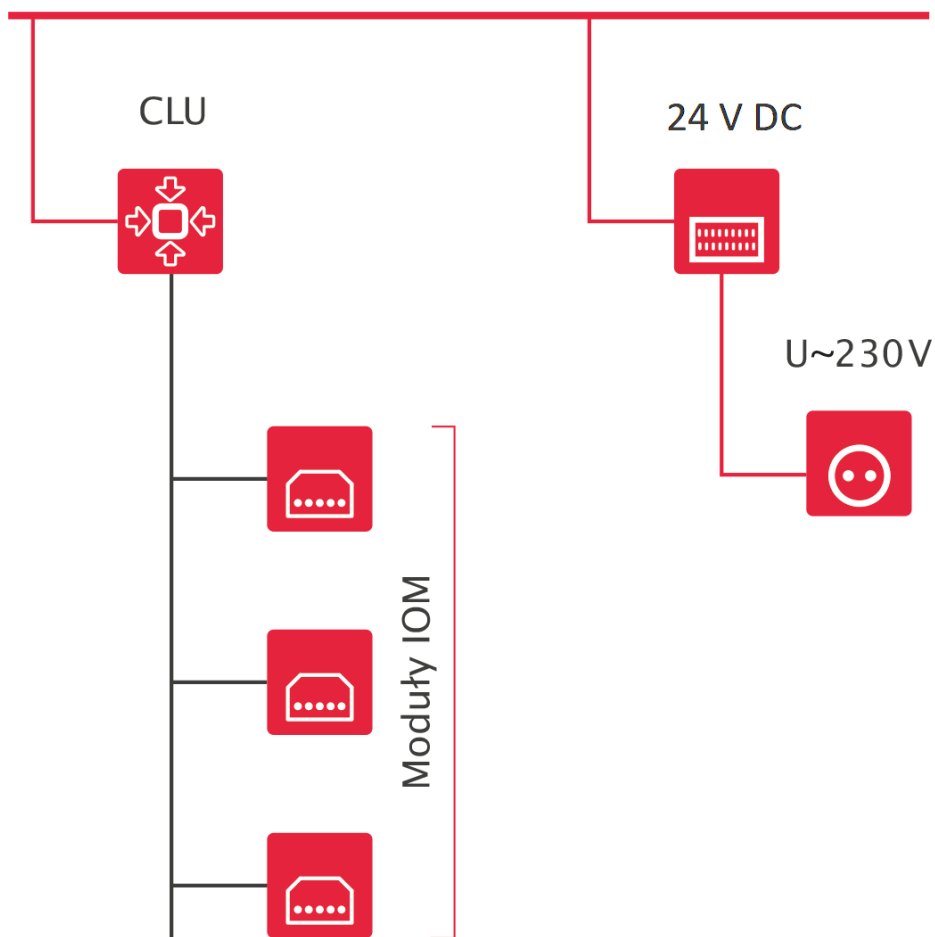


3. Zasilanie modułów

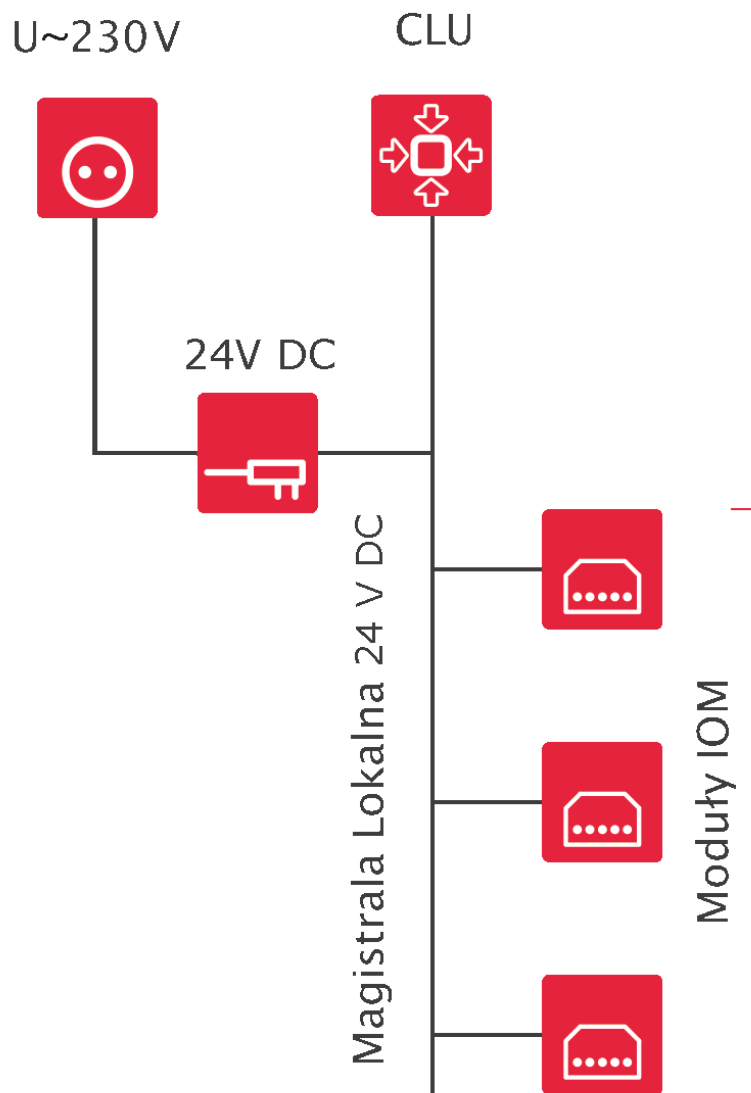
Zasilanie modułów CLU oraz IOM może być zrealizowane na dwa sposoby

1. Poprzez podłączenie zasilania do magistrali systemowej 24V DC – w takiej sytuacji moduł CLU będzie zasilał podłączone do niego moduły IOM po magistrali lokalnej. Przez magistralę lokalną (TFbus) może przepływać prąd o maksymalnym natężeniu 1000mA.

Magistrala Systemowa

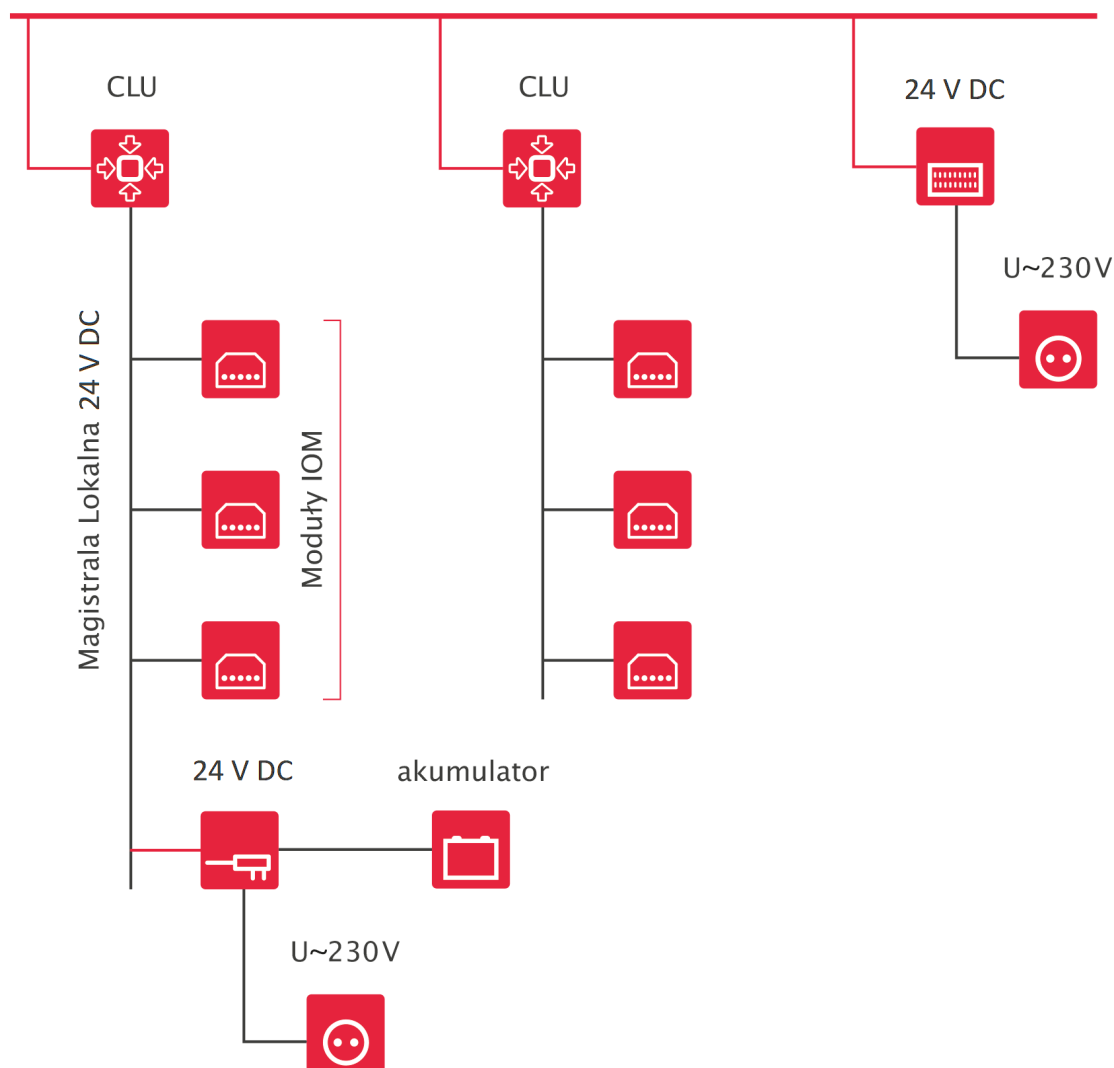


2. Poprzez podłączenie zasilania 24 V DC do magistrali lokalnej. W takiej sytuacji CLU będzie zasilane z magistrali lokalnej.



W przypadku modułów podtyrkowych możliwe jest opcjonalne wykorzystanie zasilacza podtyrkowego 24 V DC.

Magistrala Systemowa



UWAGA! CLU może być jednocześnie podłączone do zasilania z magistrali systemowej oraz magistrali lokalnej!

IV. Instalacja komponentów

Większość modułów dostarczanych jest w dwóch rodzajach wykonania: na szynę DIN (do montażu w rozdzielni) i w wersji podtynkowej. Dodatkowo dostępne są moduły Z-Wave: Relay, Roller Shutter oraz Digital IN.

1. Montaż modułów na rozdzielni

Moduły oferowane przez GRENTON dostarczane są w obudowach przystosowanych do montażu w rozdzielniach na szynie DIN. Aby zamontować moduł, należy założyć moduł na szynę oraz zablokować zatrzask znajdujący się od dołu modułu. Następnie podłączyć moduły do magistrali systemowej za pomocą specjalnych złącz magistralnych, oraz podpiąć przewody przyłączeniowe zgodnie z dołączoną do modułów instrukcją instalacji.

UWAGA! Moduły w OM identyfikowane są za pomocą nr seryjnego. Po zainstalowaniu modułu należy zapisać jego nr seryjny oraz fizycznie podpięte wejścia/wyjścia - ułatwi to identyfikację konkretnych obiektów.

2. Montaż modułów podtynkowych kablowych

Moduły przeznaczone do montażu podtynkowego przystosowane są do montażu w puszkach o średnicy 70 mm oraz większości puszek 60 mm. W przypadku modułów podtynkowych zaleca się stosowanie puszek z kieszenią boczną. Dla puszek 60 mm należy dokonać sprawdzenia, czy moduły mieszczą się w tym konkretnym typie puszek.

Do montażu większej liczby modułów, należy stosować puszki pogłębiane.

3. Montaż modułów podtynkowych Z-Wave

Moduły bezprzewodowe przystosowane są do montażu w puszkach instalacyjnych o średnicy minimum 60 mm. Dla modułów podtynkowych zaleca się stosowanie puszek z kieszenią boczną.

V. Object Manager

1. Instalacja OM

Minimalne wymagania sprzętowe dla komputera oraz szczegółowa instrukcja instalacji programu konfiguracyjnego Object Manager dołączona jest do plików instalacyjnych oprogramowania.

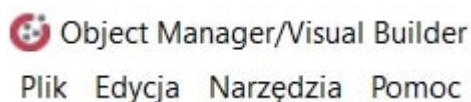
Aktualną wersję Object Managera można pobrać ze strony: <https://www.grenton.pl/wsparcie/materialy-do-pobrania.html>.

UWAGA! Folder, w którym będzie instalowany Object Manager nie może zawierać w nazwie znaków specjalnych tj. %, !, # itd.

2. Struktura menu

Obsługa Object Manager'a dokonywana jest za pomocą trzech dostępnych dla użytkownika paneli menu:

- **główne**



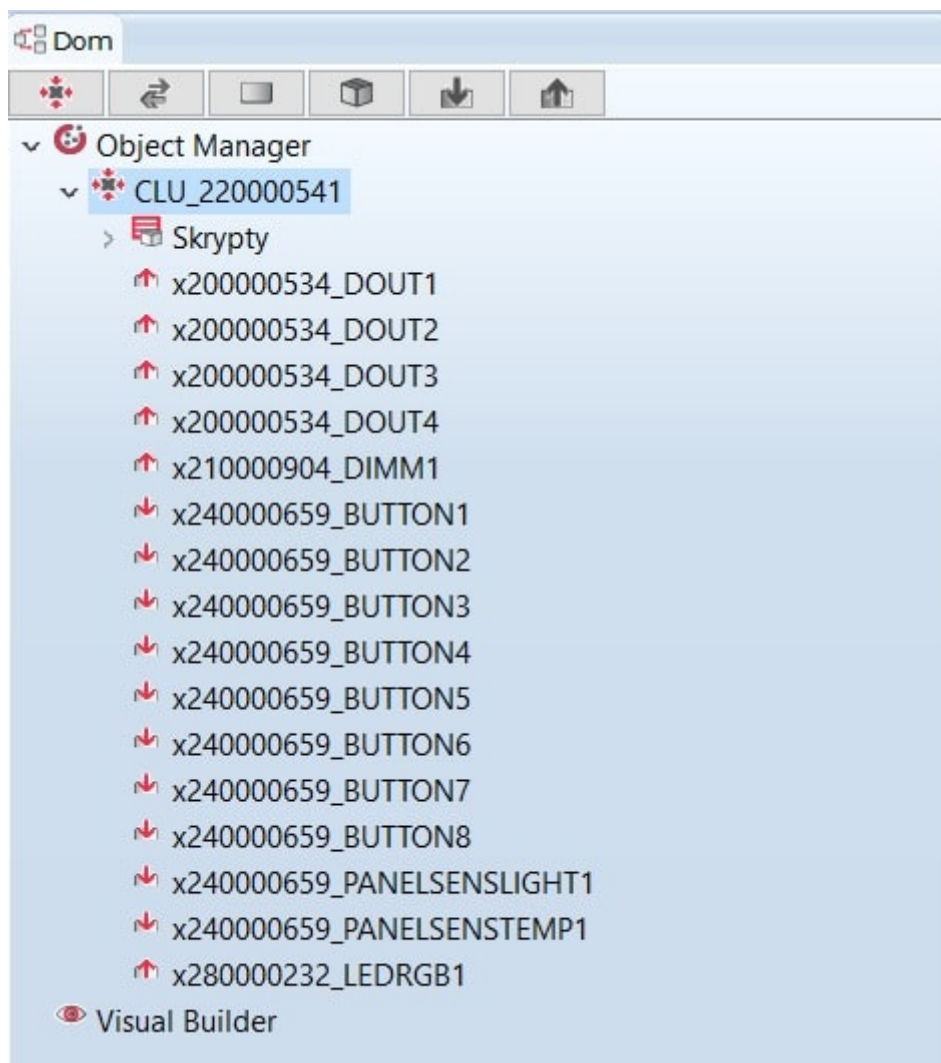
Zawiera podstawowe komendy służące do obsługi projektu.

- **akcji**



Ikony zawarte w tym menu wykorzystywane są przy programowaniu i konfigurowaniu urządzeń. Podświetlone są tylko te ikony, które w danej chwili mogą zostać użyte. Wynika to z kontekstu, w którym w danej chwili się znajduje (np. jeżeli w bocznym drzewku wybrano CLU, to aktywne stają się ikony związane z CLU).

- **obiektów**



Składa się z dwóch części: listy obiektów (CLU, wejść, wyjść) oraz Visual Buildera.

Wszystkie dane o konfiguracji systemu przechowywane są w pliku projektu. W OM można przechowywać dowolną liczbę projektów, z których każdy jest związany z inną instalacją/budynkiem/apartamentem.

3. Pliki projektów

3.1. Katalog zapisanych projektów

Po zainstalowaniu Object Managera należy wskazać katalog, w którym będą przechowywane zapisane projekty.

Domyślna ścieżka dostępu do katalogu: `C : \ \OM\projects`

Wszystkie pliki utworzonych i zapisanych projektów zapisywane są w tym katalogu z rozszerzeniem `*.omp` (np. projekt.omp).

3.2. Kopia zapasowa projektu (backup)

W trakcie pracy nad projektem, istnieje możliwość wykonania kopii zapasowej projektu, która nie ulega modyfikacji, mimo wykonania zmian w projekcie. Dzięki temu istnieje możliwość odzyskania wcześniejszej wersji projektu, gdyby użytkownik wykonał niepożądane zmiany w konfiguracji. Do każdego projektu można utworzyć dowolną liczbę kopii zapasowych.

UWAGA! Zaleca się możliwe najczęstsze wykonywanie kopii zapasowych projektu - zwłaszcza przed wykonaniem istotnych zmian w konfiguracji systemu.

By utworzyć kopię zapasową projektu, należy z menu głównego wybrać pozycję **Plik->wykonaj kopię zapasową projektu** (kopię można również wykonać skrótem klawiszowym **CTRL+Shift+B**).




Zapisane kopie dostępne są na liście otwieranej po kliknięciu w **Dostępne kopie zapasowe** lub w oknie otwierania projektu w zakładce **Kopie zapasowe**.

UWAGA! Po wybraniu kopii zapasowej z listy zostanie ona załadowana, a zmiany obecnie wprowadzone w projekcie, jeśli nie były zapisane, zostaną utracone.

4. Podstawowe elementy

4.1. Konfigurator obiektów






Każde wejście, wyjście, sensor czy inne urządzenie fizycznie podpięte do systemu ma swoje odzwierciedlenie w OM w postaci obiektów. Obiekty nie odzwierciedlają fizycznych modułów, ale poszczególne wejścia i wyjścia. Każdy obiekt posiada swoje wartości początkowe, cechy wbudowane oraz zdarzenia, zobrazowane w konfiguratorze obiektów. Oto formatka, która otwiera się po kliknięciu na danym obiekcie.







CLU_220000541->x200000534_DOUT1

Nazwa: Źródło/Odbiornik:

Identyfikacja: Typ:

 Sterowanie
  Schematy konfiguracji
  Zdarzenia
  Cechy wbudowane
  Statystyki

Metoda	Nazwa parametru	Wartość	Wywołaj
SetValue	Value	<input type="text" value="Off"/>	
Switch	Time	<input checked="" type="radio"/> Unlimited <input type="radio"/> Time <input type="text"/> ms	
SwitchOn	Time	<input checked="" type="radio"/> Unlimited <input type="radio"/> Time <input type="text"/> ms	
SwitchOff	Time	<input checked="" type="radio"/> Unlimited <input type="radio"/> Time <input type="text"/> ms	

Powyższa formatka zawiera następujące sekcje:






1. Informacje podstawowe





Nazwa: Źródło/Odbiornik:

Identyfikacja: Typ:

Sekcja mieści się w górnej części formatki i zawiera podstawowe informacje właściwe dla każdego obiektu, np: adres IP, nazwę, typ modułu, numer seryjny oraz numer wejścia/wyjścia w ramach danego modułu. W sekcji tej użytkownik może również zdefiniować rodzaj źródła lub odbiornika fizycznie podłączonego do tego obiektu.

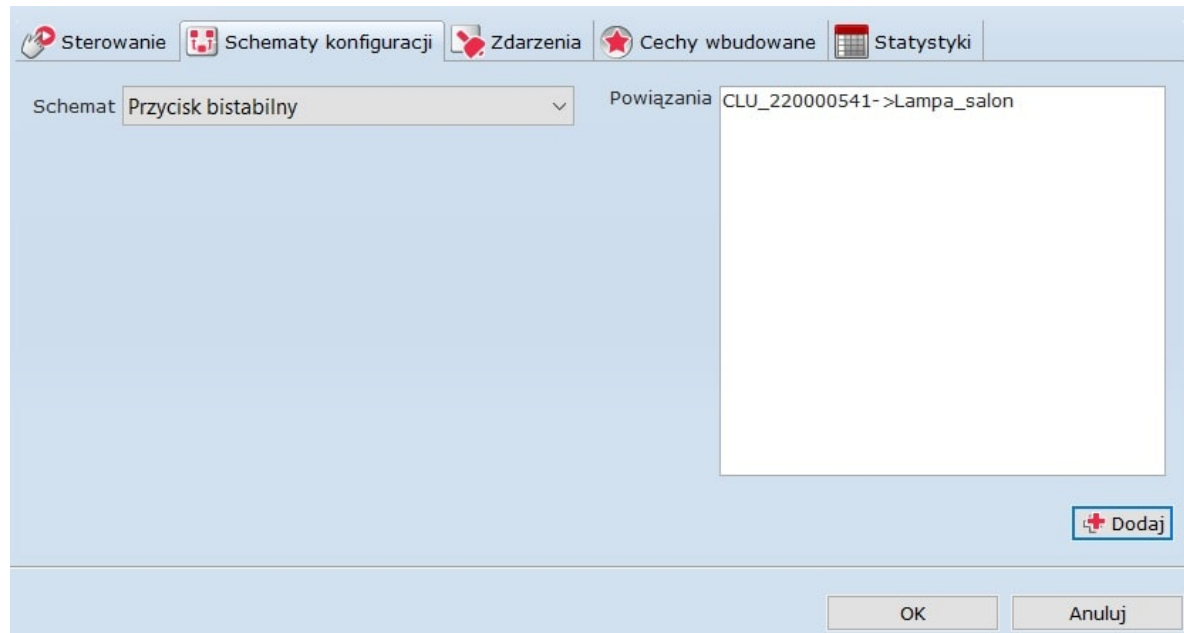
2. Zakładka sterowanie

 Sterowanie
  Schematy konfiguracji
  Zdarzenia
  Cechy wbudowane
  Statystyki

Metoda	Nazwa parametru	Wartość	Wywołaj
SetValue	Value	<input type="text" value="Off"/>	
Switch	Time	<input checked="" type="radio"/> Unlimited <input type="radio"/> Time <input type="text"/> ms	
SwitchOn	Time	<input checked="" type="radio"/> Unlimited <input type="radio"/> Time <input type="text"/> ms	
SwitchOff	Time	<input checked="" type="radio"/> Unlimited <input type="radio"/> Time <input type="text"/> ms	

Zawiera metody (wraz ze wszystkimi parametrami), właściwe dla przeglądanej obiektu. Umożliwia wywołanie danej metody z poziomu OM. Przykładowo dla wyjścia przekaźnikowego możemy wywołać metodę `SwitchOn` z parametrem `Time` (np. 30 s), co spowoduje załączenie tego wyjścia na czas 30 s. W celu wywołania metody z poziomu OM należy - w zakładce *Sterowanie* - wprowadzić wartości parametrów wywoływanej metody (o ile są konieczne) i wcisnąć przycisk *Wywołaj*.

3. Schemat konfiguracji

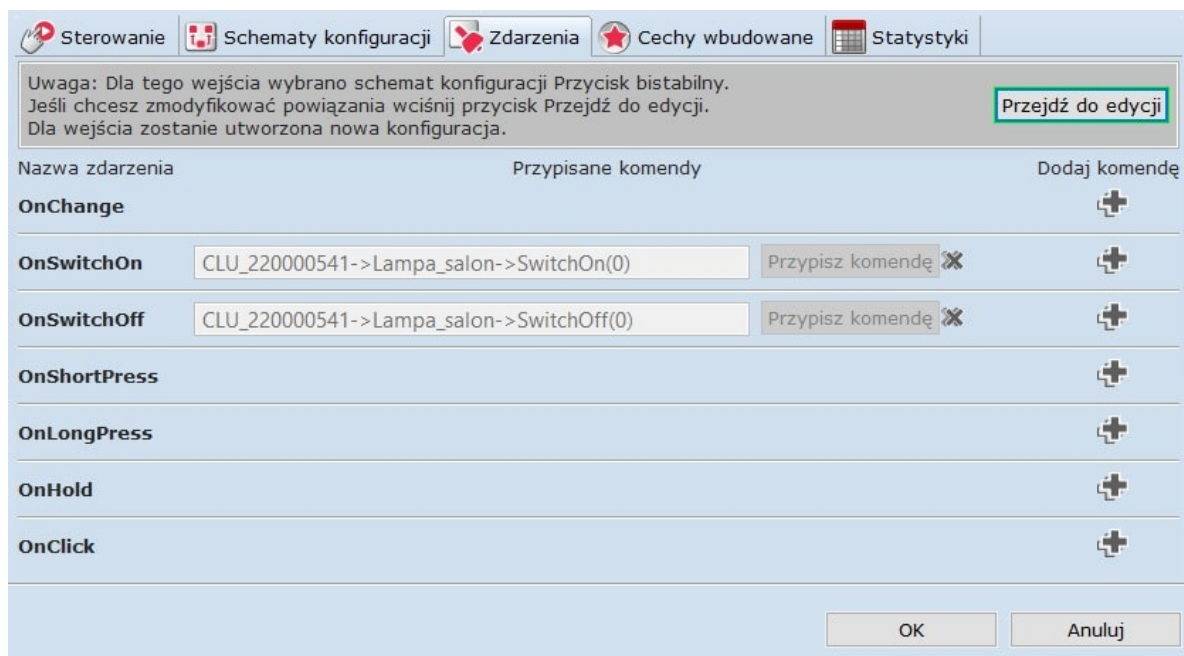


Schematy konfiguracji definiują sposób zachowania obiektu i pozwalają na uproszczoną konfigurację logiki. Po wybraniu schematu konfiguracji dla danego wejścia i dodając powiązania obiektów, Object Manager automatycznie utworzy powiązania odpowiednich zdarzeń z metodami powiązanych obiektów.

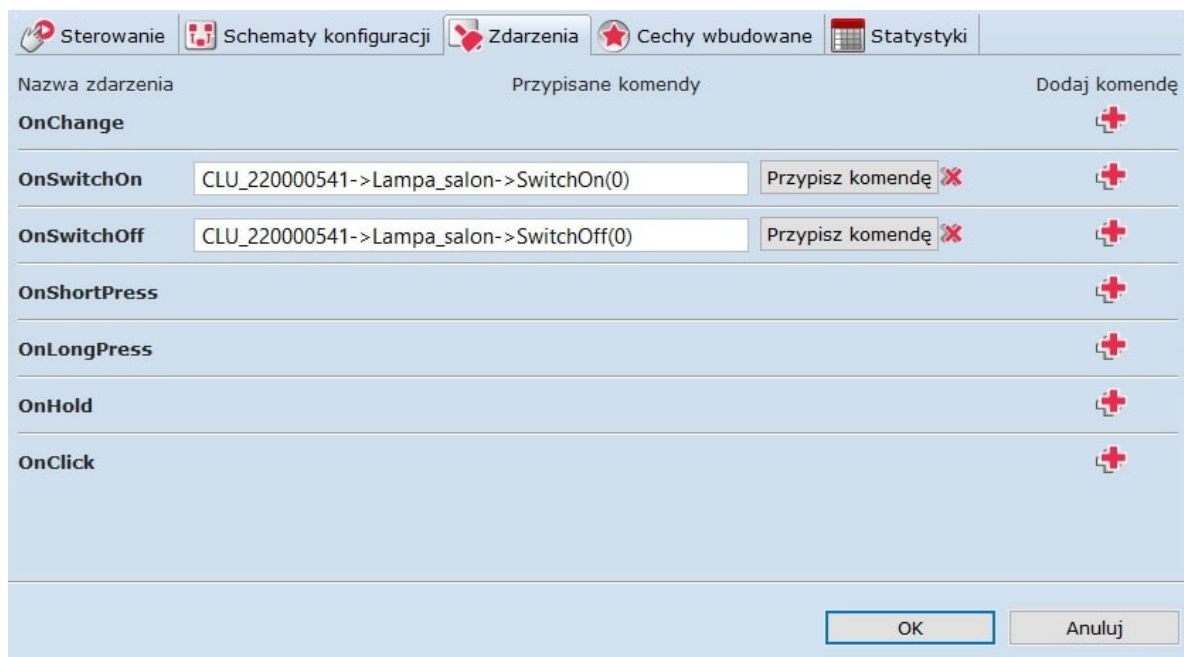
W przypadku gdy użytkownik stworzył własne powiązania zdarzenie-metoda, korzystając z zakładki *Zdarzenia* widoczne są one na liście jako *Schemat użytkownika*.

4. Zdarzenia - opis zakładki

Zakładka ta zawiera listę zdarzeń odpowiednią dla danego typu obiektu oraz przypisanych im metod, które są wywoływane po zaistnieniu zdarzenia (jeśli użytkownik je zdefiniował). W przypadku jeśli wybrany został schemat konfiguracji, zakładka jest w trybie tylko do odczytu i pokazuje jedynie powiązania stworzone w ramach wybranego schematu.



W każdej chwili można przejść do edycji powiązań zdarzenie-metoda, klikając *Przejdź do edycji*. W takiej sytuacji utworzony zostanie *Schemat użytkownika*, który pojawi się na liście w zakładce schematy konfiguracji.





Po dodaniu komendy do wybranego zdarzenia, otwiera się lista obiektów. Następnie po zaznaczeniu interesującego nas obiektu pojawia się lista metod, jakie można na nim wywołać. Dodanie wybranej metody powoduje utworzenie nowej zależności pomiędzy obiektami.

5. Cechy wbudowane

W tej części przedstawione są wartości, jakie aktualnie posiada wybrany obiekt oraz wartości początkowe, które zostały w nim zapisane (wartości początkowe ustawiane w przypadku restartu systemu, np. po zaniku zasilania). Wpisanie wartości w polu *Wartości początkowe*, spowoduje ustawienie jej podczas startu CLU.


Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
Intertion	0	<input type="text" value="0"/>	ms	[0-2000]
HoldDelay	500	<input type="text" value="500"/>	ms	[0-5000]
HoldInterval	100	<input type="text" value="50"/>	ms	[0-2000]
Value	0		bool	[0-1]
StatisticState	0	<input type="text" value="Off"/>	number	0,1,2

Auto odświeżanie 

 **Odśwież**


6. Cechy użytkownika (tylko CLU)


Zakładka ta pozwala użytkownikowi zdefiniować na CLU własną listę cech, które mogą być następnie używane do przechowywania różnego typu danych (liczniki, znaczniki). Dodanie cechy użytkownika następuje po kliknięciu przycisku *Dodaj* i podaniu nazwy cechy. Następnie należy zdefiniować wartość początkową oraz typ cechy (tekstowa, numeryczna lub boolowska).



CLU
×


Nazwa: ID:

IP: FW:

 Sterowanie

 Zdarzenia

 Cechy wbudowane

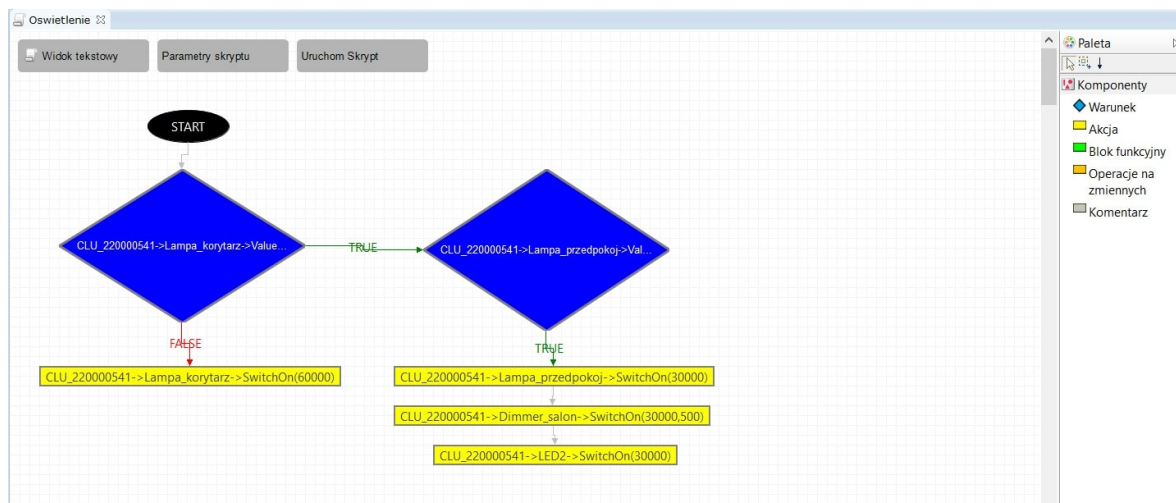
 **Cechy użytkownika**

Nazwa cechy	Aktualna wartość	Wartość początkowa	Typ	
Strona	1	<input type="text" value="1"/>	<input type="text" value="number"/> <ul style="list-style-type: none"> number string <li style="background-color: #007bff; color: white;">number boolean 	✘

4.2. Script builder

To narzędzie służące do tworzenia skryptów, które może działać w dwóch trybach:

1. **Graficznym** (uproszczonym), w którym w prosty sposób możemy stworzyć schemat, poprzez przeciąganie oraz łączenie elementów.



Tryb graficzny umożliwia tworzenie skomplikowanych skryptów składających się z wielu warunków oraz metod. Możliwe jest również wykorzystanie zmiennych i parametrów. Jedynym ograniczeniem jest brak możliwości tworzenia pętli, które wymagają użycia trybu tekstowego.

2. **Tekstowym** (pełnym), w którym użytkownik ma możliwość tworzenia logiki z wykorzystaniem rozszerzonego języka LUA. Dzięki temu możliwe jest tworzenie bardzo zaawansowanych skryptów z wykorzystaniem wszelkich elementów języka LUA (w tym pętli, tablic itp.).

```
1 if(not (CLU_220000541->Lampa_korytarz->Value==1)) then
2 CLU_220000541->Lampa_korytarz->SwitchOn(60000)
3 else
4 if(CLU_220000541->Lampa_przedpokoj->Value==0) then
5 CLU_220000541->Lampa_przedpokoj->SwitchOn(30000)
6 CLU_220000541->Dimmer_salon->SwitchOn(30000,500)
7 CLU_220000541->LED2->SwitchOn(30000)
8 end
9 end
10
```

W stosunku do standardowego języka LUA, język został rozszerzony o możliwość bezpośredniego odwoływania się do adresów, metod i cech, które traktowane są tak samo jak inne funkcje LUA.

4.3. Diagram połączeń

Narzędzie obrazujące zależności oraz powiązania pomiędzy wszystkim obiektami znajdującymi się w systemie. Dzięki niemu można w prosty i czytelny sposób znaleźć interesującą zależność lub bez konieczności przeszukiwania konfiguracji sprawdzić zależności dla danego modułu.











Diagram połączeń można uruchomić z menu górnego: *Narzędzia->Diagram połączeń* lub skrótem klawiszowym [ALT+Q].

Każdy obiekt w systemie prezentowany jest na diagramie jako koło z wyświetlonym obok adresem. Kolor koła uzależniony jest od typu obiektu:

- CLU – kolor czerwony;
- Wejście/wyjście – kolor wiśniowy;
- Zdarzenia wejść lub wyjść – kolor jasnoniebieski;
- Zdarzenia generowane przez timery – kolor ciemnoniebieski;
- Metody wbudowane – kolor ciemnozielony;

- Metody skryptowe – kolor jasnozielony;
- Cechy wbudowane – kolor żółty;
- Cechy zdefiniowane – kolor pomarańczowy;

Powiązania między obiektami wyświetlane są w postaci strzałek, których grot wskazuje obiekt wyzwalany.

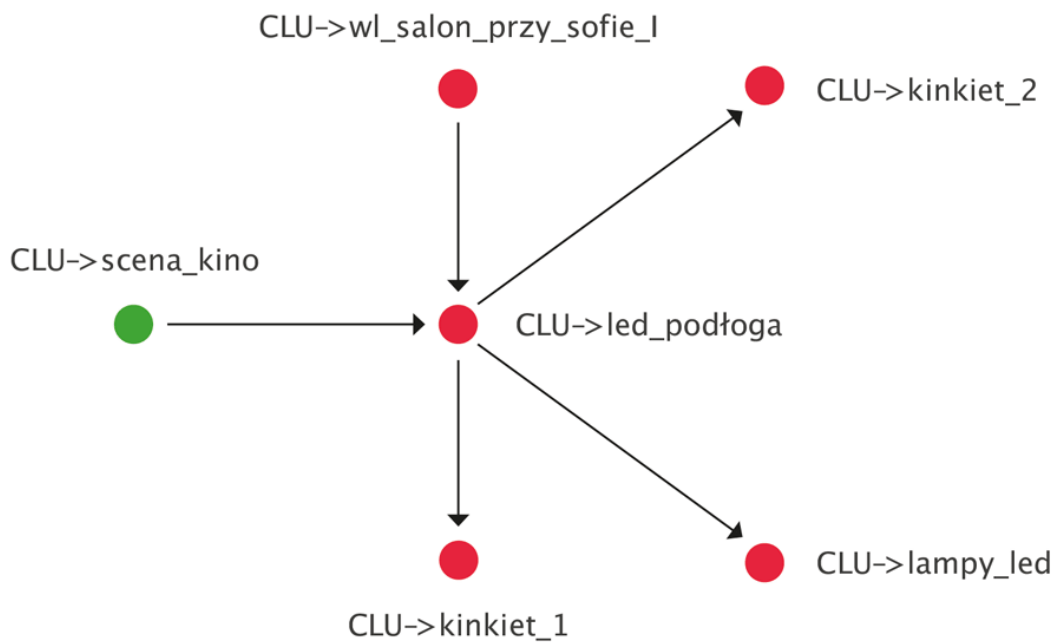
RODZAJ POWIĄZANIA	ZAKOŃCZENIE OD STRONY OBIEKTU AKTYWNEGO	ZAKOŃCZENIE OD STRONY OBIEKTU PASYWNEGO
Bezwarunkowe wywołanie metody		
Wywołanie metody objęte warunkiem		
Odczyt wartości cech		
Zapis wartości cech		
Powiązanie obiektów dwukierunkowe		

Powiązania obrazowane są na trzech poziomach:

1. CLU-CLU – wyświetla powiązania pomiędzy dwoma CLU, jeżeli jakkolwiek obiekt jednego CLU (wejścia/wyjścia) jest powiązany z drugim CLU.
2. Powiązania pomiędzy obiektami – wyświetla powiązania pomiędzy poszczególnymi obiektami (wejściami/wyjściami) bez wskazywania na konkretne zdarzenia, cechy, metody.
3. Powiązania zdarzeń, metod i cech – wyświetla najbardziej szczegółowy widok obrazujący co wywołują konkretne zdarzenia itd.

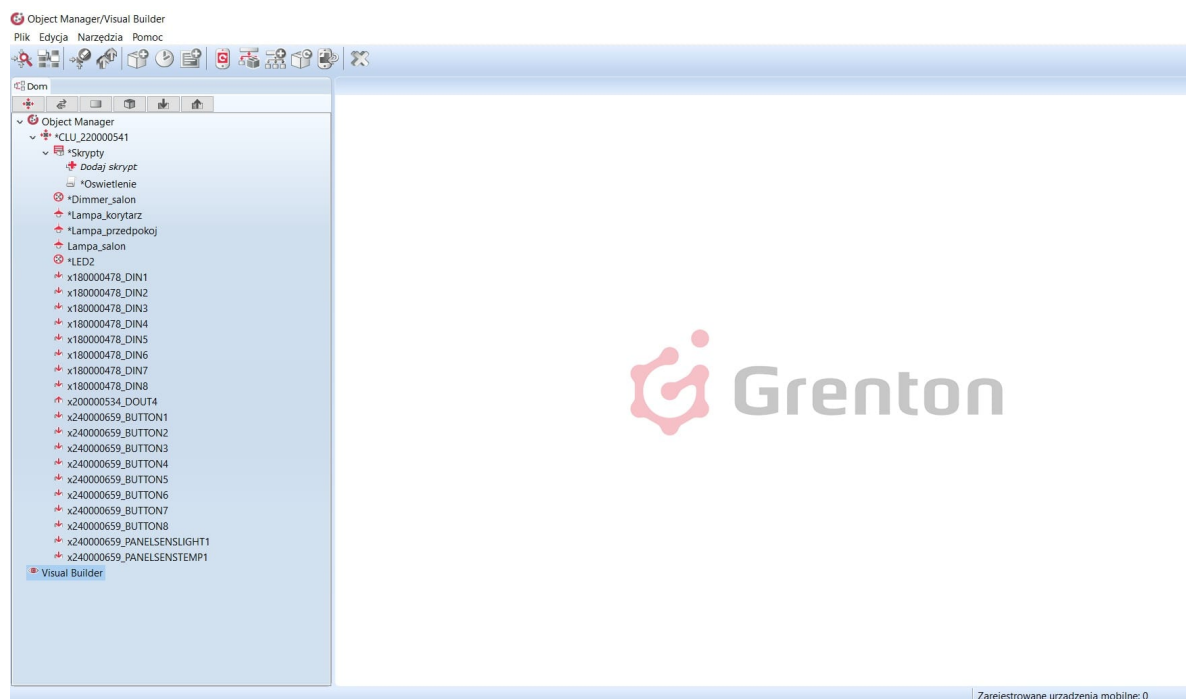
Również nawigacja odbywa się w dwóch płaszczyznach:

1. W płaszczyźnie poziomej – umożliwia przechodzenie pomiędzy obiektami na tym samym poziomie poprzez kliknięcie na dowolny obiekt na diagramie (z wyjątkiem centralnego).
2. W płaszczyźnie pionowej – umożliwia przechodzenie góra-dół, pomiędzy kolejnymi poziomami poprzez kliknięcie na obiekt centralny i wybór obiektu z listy, która się pojawia (przejdźcie w dół) lub poprzez naciśnięcie przycisku „w górę” znajdującego się w górnej części diagramu (przejdźcie w górę).



4.4. Visual Builder

Visual Builder jest narzędziem służącym do tworzenia interfejsu użytkownika na urządzenia mobilne dla aplikacji Home Manager. Interfejs może być tworzony automatycznie na podstawie projektu instalacji lub może zostać zaprojektowany i utworzony przez użytkownika wg osobistych preferencji. Użytkownik ma możliwość stosowania własnych grafik. Tworzenie interfejsu odbywa się poprzez drag&drop komponentów Visual Buildera i umożliwia utworzenie interfejsu dla wszystkich popularnych rozdzielczości. Ikona uruchamiająca VB znajduje się na końcu rozwijanego drzewka obiektów.



4.5. myGreenton

Narzędzie służącym do tworzenia interfejsu użytkownika na urządzenia mobilne dla aplikacji myGrenton. Tworzenie interfejsu odbywa się poprzez drag&drop elementów z drzewa obiektów dostępnych w danym projekcie (modułów podłączonych do CLU) - utworzony widget będzie posiadał uprzednio zdefiniowany szablon.



4.6. Kosz

Wzorowany jest na rozwiązaniu znanym z systemów operacyjnych. Usunięty obiekt, skrypt czy aplikacja w projekcie nie jest usuwana bezpowrotnie, ale trafia do kosza dając możliwość użytkownikowi na odzyskanie skasowanych danych w przypadku zmiany koncepcji.

Kosz ma postać zakładki znajdującej się w drzewku obiektów i pojawia się gdy jakiś obiekt zostanie usunięty. Obiekty z kosza można w dowolnej chwili przywrócić klikając na nie prawym klawiszem i wybierając *Przywróć* z menu kontekstowego.

Obiekt można bezpowrotnie usunąć z kosza wybierając *Usuń* z menu kontekstowego. Przywrócenie do projektu usuniętego w taki sposób modułu jest możliwe jedynie poprzez wykonanie *CLU Discovery*.

Kosz jest świetnym rozwiązaniem do przechowywania obiektów, które nie są w tej chwili wykorzystywane, ale mogą być przydatne w przyszłości.

VI. Podstawowa konfiguracja systemu

1. Połączenie OM z CLU

W celu skonfigurowania urządzenia w systemie, komputer musi być podłączony do modułów CLU. W trakcie przeprowadzania prac, wszystkie moduły CLU muszą być połączone ze sobą przy pomocy kabla Ethernet.

Istnieją dwa sposoby połączenia:

1. Bezpośrednie połączenie z komputerem Kabel sieciowy podłączamy do karty sieciowej w komputerze i łączymy z gniazdem sieciowym w module CLU.
2. Połączenie przez sieć lokalną Można połączyć się z systemem GRENTON wykorzystując sieć lokalną. W tym celu zarówno moduł CLU jak i komputer, z którego będziemy się łączyć, muszą znajdować się w tej samej podsieci.

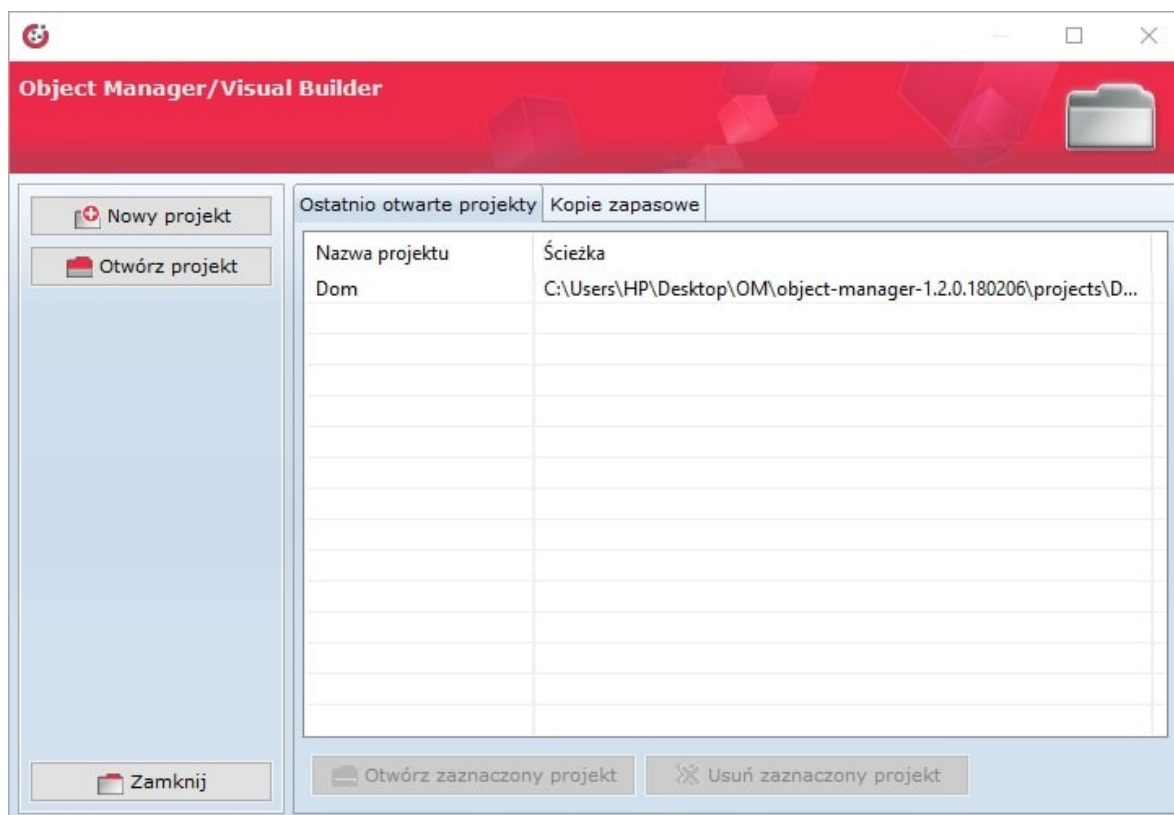
2. Adresy IP

Moduł CLU, jak każde urządzenie sieciowe, otrzymuje swój własny adres IP. Każdy z modułów zainstalowanych w danym systemie musi posiadać swój unikalny adres IP, jednakże wszystkie moduły CLU w systemie muszą pracować w tej samej podsieci, by mogły się ze sobą komunikować. Adres IP dla danego CLU może zostać w dowolnym momencie zmieniony przez użytkownika. Zmiany adresu można dokonać poprzez formatkę konfiguratora urządzeń dla wybranego CLU przez wpisanie w polu zawierającego stary adres, nowego adresu.

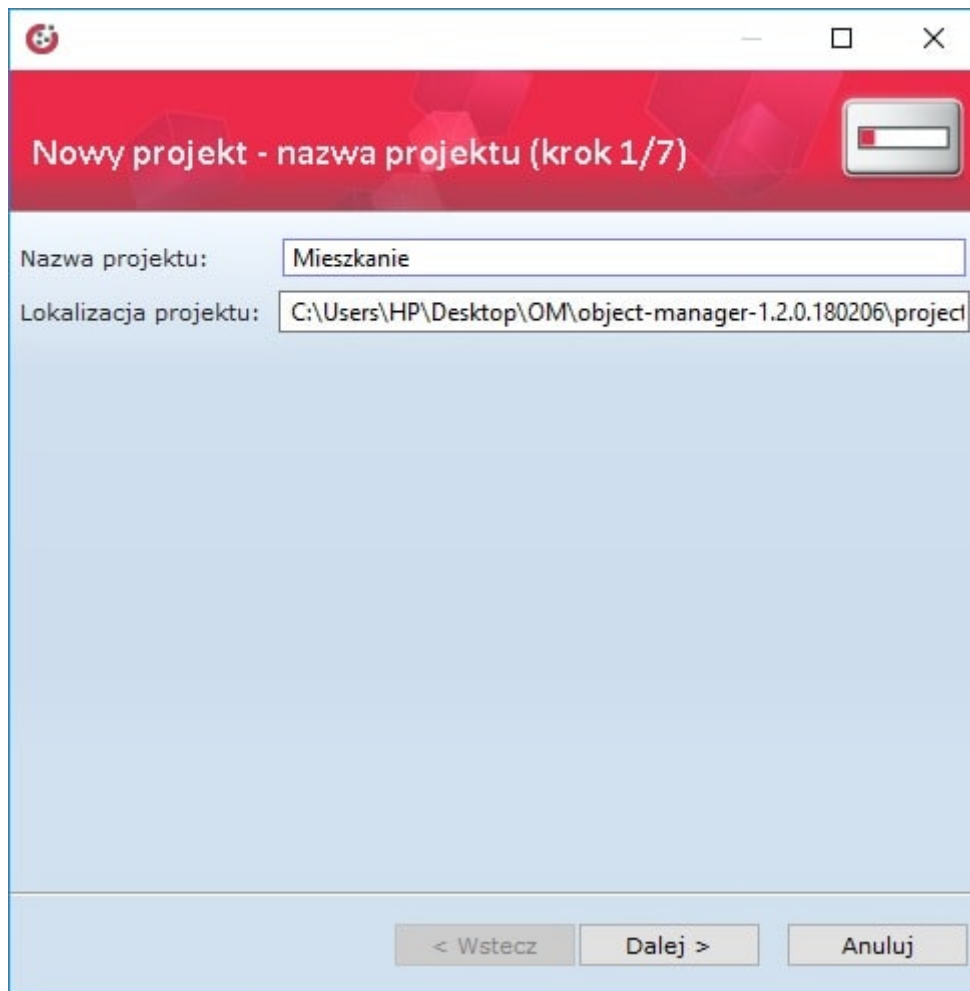
UWAGA! Po podpięciu CLU (lub kilku CLU) do karty sieciowej komputera, zostanie mu nadany nowy adres IP - zgodny z pulą adresów karty sieciowej komputera.

3. Otwarcie nowego projektu

Po otwarciu Object Managera pojawia się okno z możliwością wyboru dwóch opcji: otwarcie zapisanego projektu oraz utworzenie nowego projektu.



1. Należy wybrać utworzenie nowego projektu, a następnie nadać nazwę tworzonemu projektowi.



2. Program Object Manager wyświetli okno konfiguracji sieciowej, w którym można wskazać zakres dostępnych adresów IP. Możliwe jest zezwolenie, aby system automatycznie wybrał CLU adres IP.

Nowy projekt - konfiguracja sieciowa (krok 2/7)

Podaj parametry sieci:

Maska sieciowa: 255.255.255.0

Brama: 192.168.0.1


Podaj zakres adresów IP, jakie mają być przydzielane modułom CLU:

Pozwól, aby system nadał adresy IP dla znalezionych CLU

Wskaż zakres adresów IP

Początek zakresu IP: 192.168.0.2

Koniec zakresu IP: 192.168.0.255


 Uwaga: Jeśli w Twojej sieci adres IP nadawany jest przez serwer DHCP, zapoznaj się z instrukcją obsługi jak prawidłowo ustawić zakres IP w takim przypadku.

< Wstecz Dalej > Anuluj

UWAGA!

Dla Object Manager w wersji 1.2.1 ustawienia dla dostępnego interfejsu sieciowego są automatycznie zaczytywane. Pozostałe dostępne interfejsy są dostępne po rozwinięciu listy.

Nowy projekt
Konfiguracja sieciowa (krok 2/6)


Interfejs sieciowy: wlan4 (192.168.4.223) 

Maska sieciowa: 255.255.255.0

Brama: 192.168.4.1

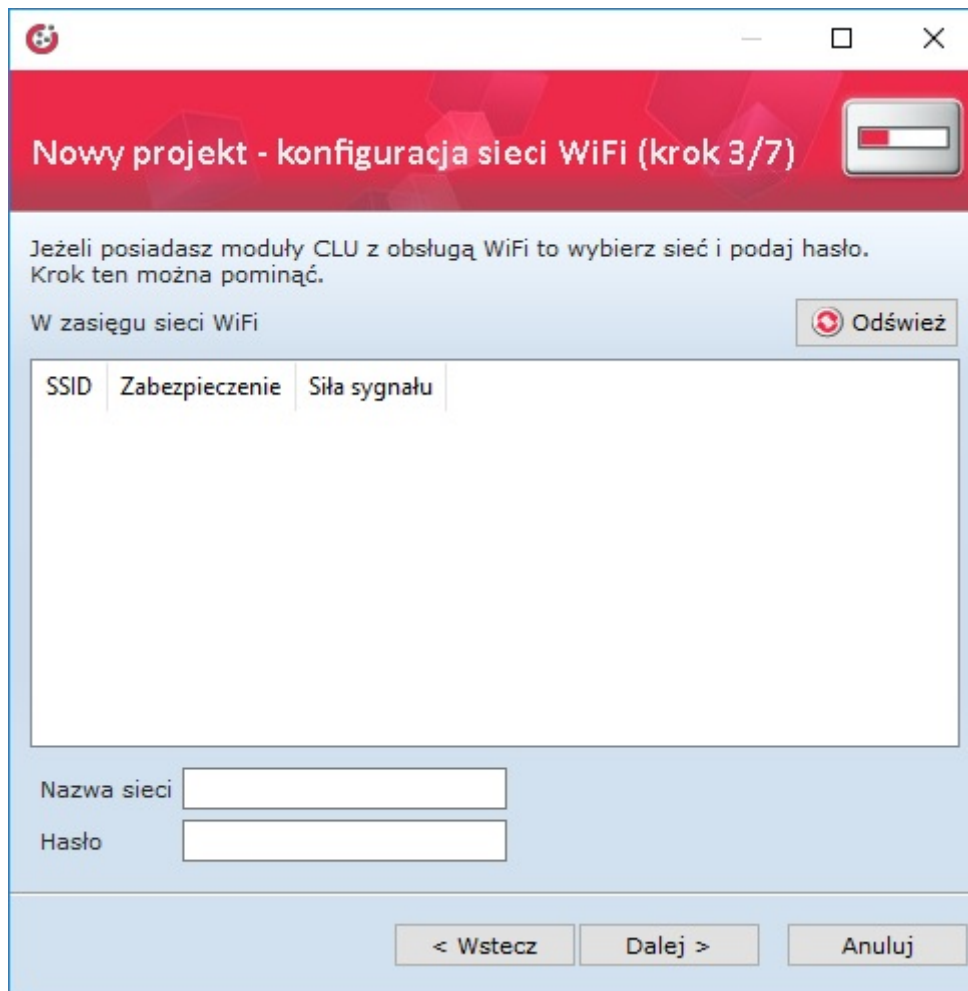
Początek zakresu IP: 192.168.4.2

Koniec zakresu IP: 192.168.4.255

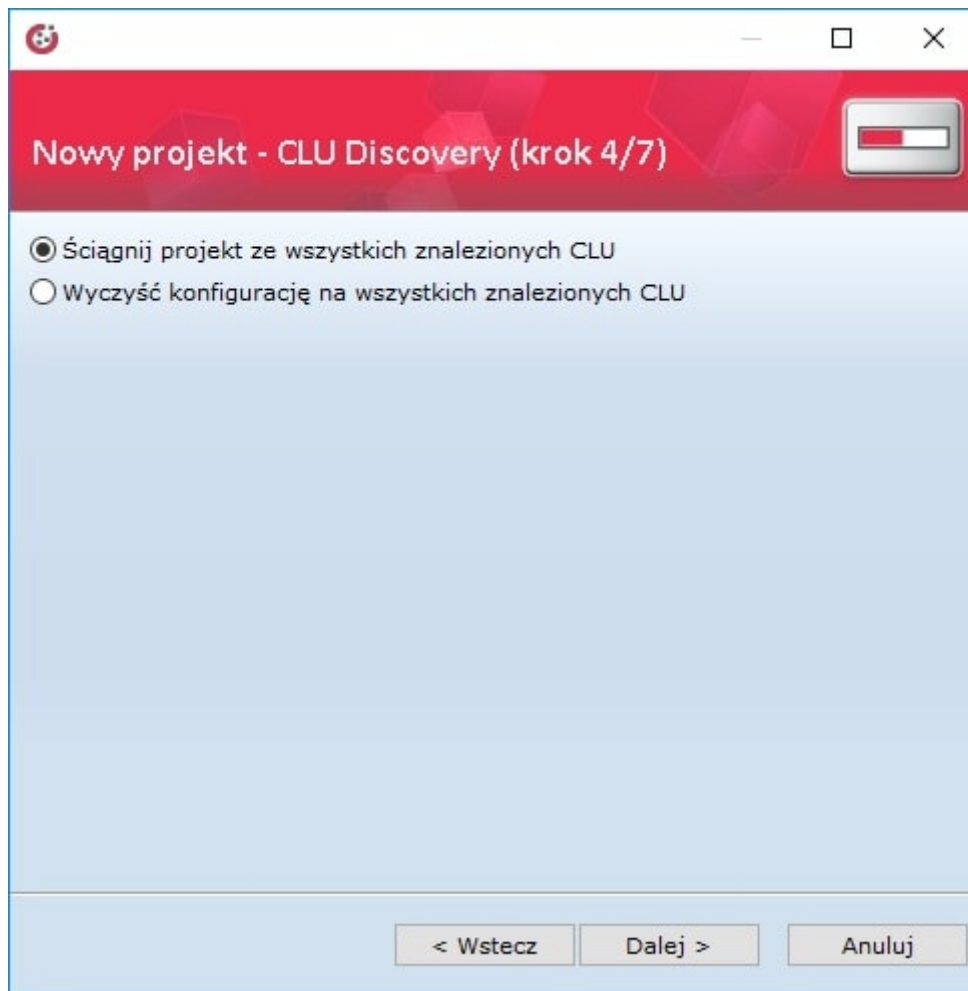
 Uwaga: Jeśli w Twojej sieci adres IP nadawany jest przez serwer DHCP, zapoznaj się z instrukcją obsługi jak prawidłowo ustawić zakres IP w takim przypadku.

< Wstecz **Dalej >** Anuluj

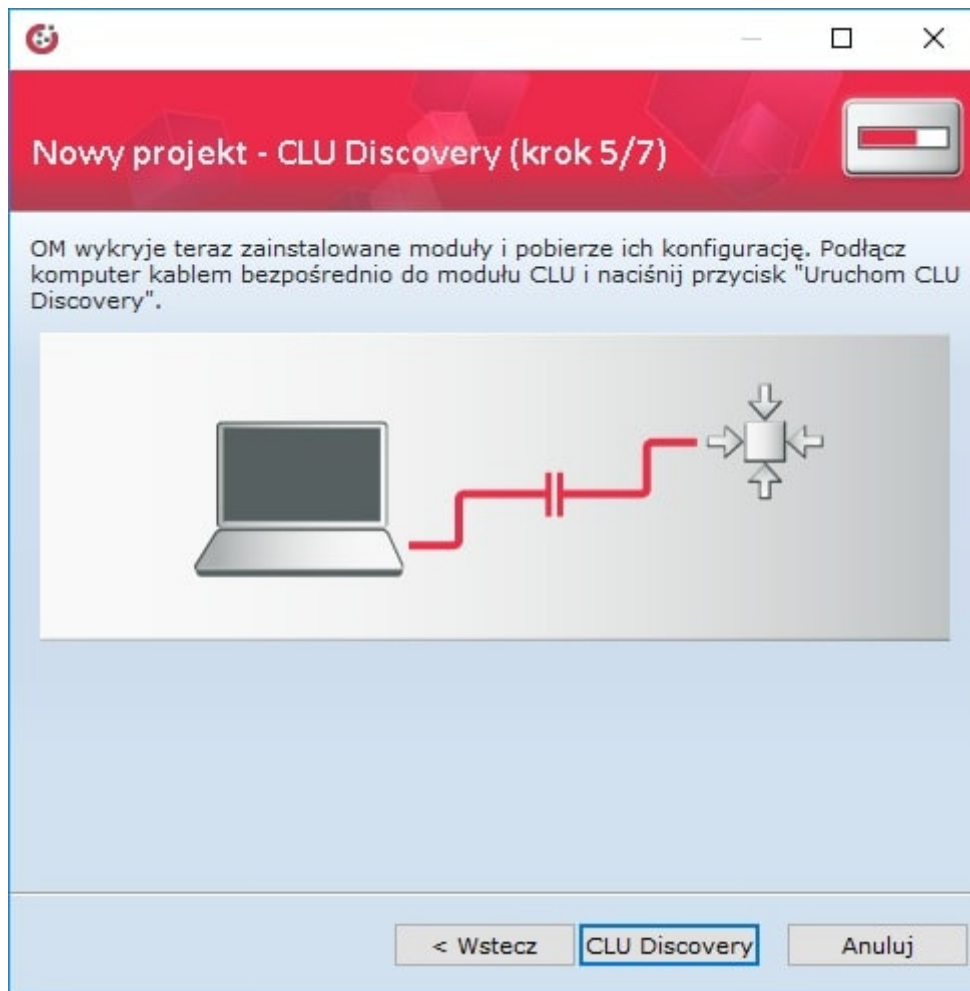
3. W kolejnym oknie pojawi się krok dotyczący konfiguracji sieci WiFi, który należy pominąć.



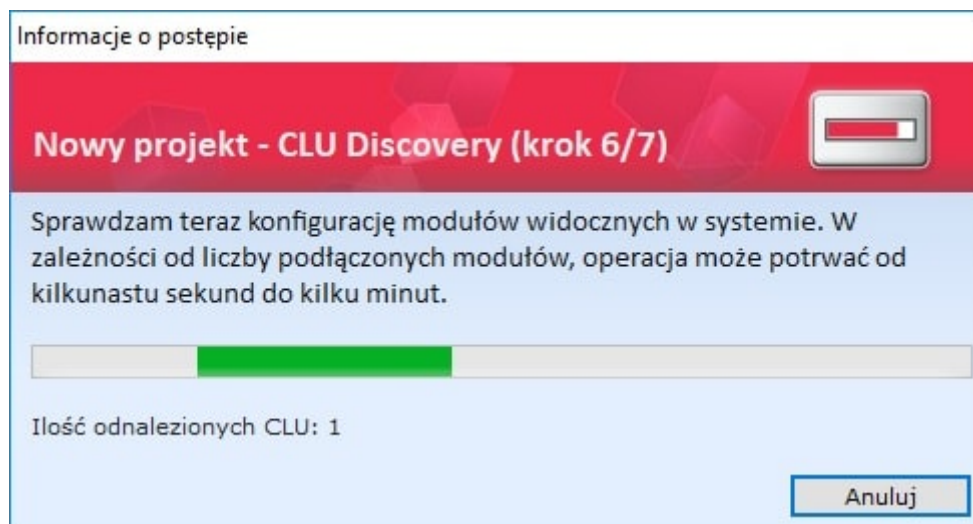
4. W czwartym kroku można dokonać wyboru pomiędzy pobraniem istniejącej konfiguracji systemu do nowo tworzonego projektu, a całkowitym wyczyszczeniem konfiguracji i rozpoczęciem tworzenia projektu od podstaw. Pierwsza opcja jest przydatna w sytuacji konieczności odtworzenia konfiguracji po utracie pliku projektu.



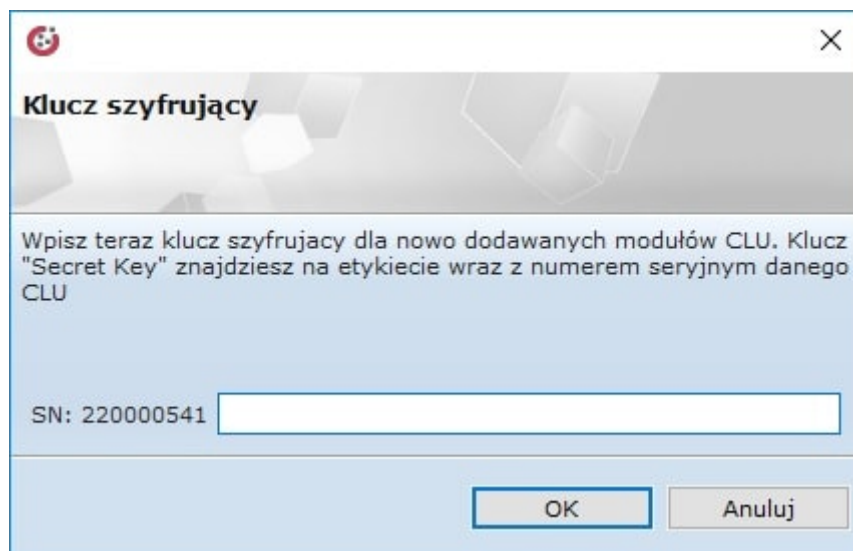
5. W kolejnym kroku należy uruchomić procedurę wyszukiwania dostępnych modułów – CLU DISCOVERY.



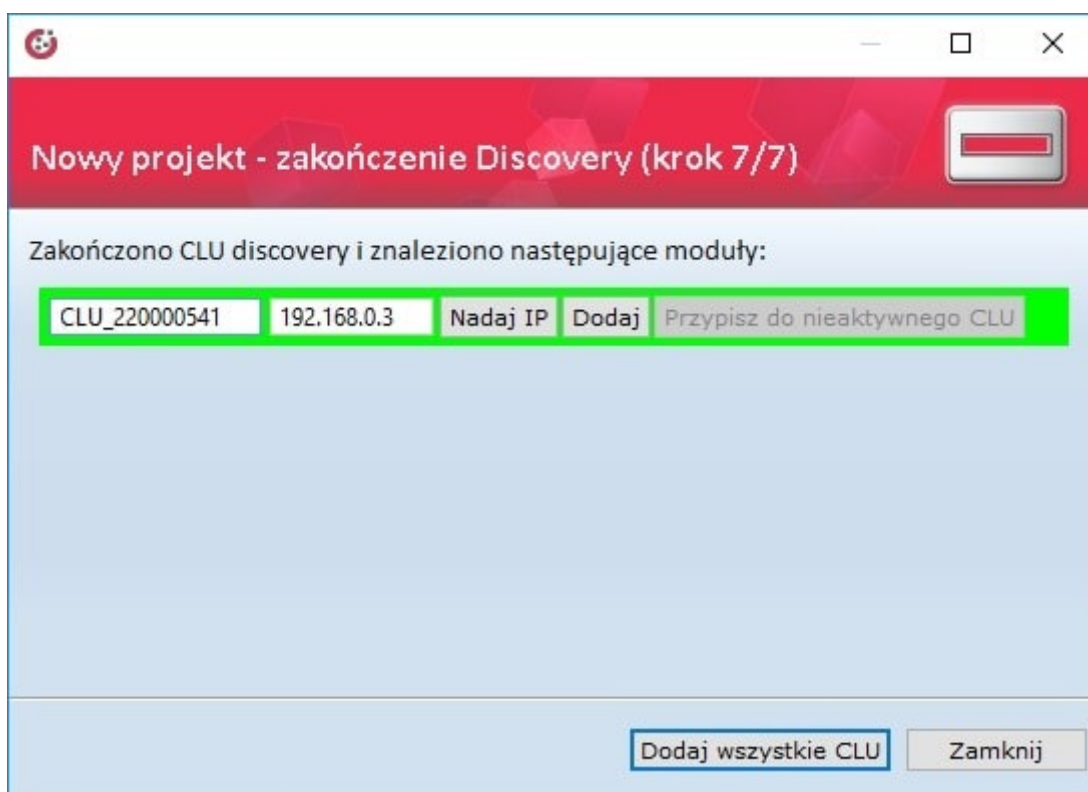
6. W szóstym kroku OM rozpoczyna wyszukiwanie dostępnych modułów CLU.



By dokończyć tworzenie nowego projektu - po wyszukaniu dostępnych CLU - w wyświetlonym oknie należy podać *Secret Key* danego CLU, który znajduje się na obudowie modułu.



7. Po zakończeniu OM wyświetli listę znalezionych modułów CLU. W tym oknie można dodać wszystkie lub wybrane moduły do tworzonego projektu.



4. Funkcja CLU DISCOVERY

Funkcja *CLU DISCOVERY* w sposób całkowicie automatyczny wynajduje moduły CLU oraz podłączone do nich moduły IOM. Jest uruchamiana obligatoryjnie podczas otwierania nowego projektu, lecz może być również uruchomiona w każdej chwili ręcznie z menu akcji.



Z funkcji *CLU DISCOVERY* należy korzystać gdy:

- Podłączono do systemu nowy moduł CLU lub IOM;

- Dokonano wymiany modułu CLU lub IOM na inny;
- Przełączono moduł IOM z jednego CLU do innego;
- Istnieje potrzeba odzyskania całkowicie usuniętego obiektu IOM.

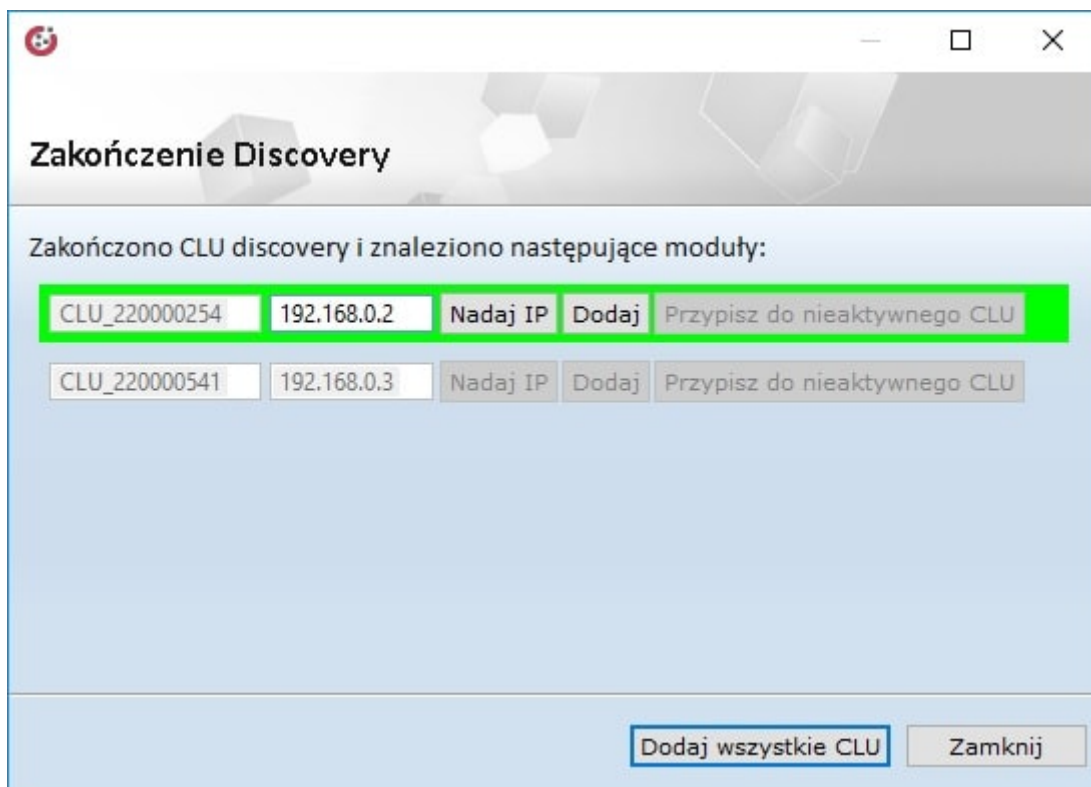
Po prawidłowym przeprowadzeniu CLU DISCOVERY wszystkie zmiany zostaną wykryte i wprowadzone do projektu.

Przed rozpoczęciem CLU DISCOVERY konieczne jest sprawdzenie czy:

- Wszystkie moduły są prawidłowo podłączone i zasilone;
- Moduły CLU są ze sobą połączone;
- Komputer, na którym uruchomiony jest OM, jest podłączony do tej samej sieci co CLU.

UWAGA! Jeżeli w sieci znajduje się router, zalecamy, by podczas procesu CLU DISCOVERY komputer był podłączony kablem sieciowym bezpośrednio z CLU. W większości przypadków CLU DISCOVERY zakończy się sukcesem również w przypadku połączenia za pośrednictwem routera, jednak w przypadku specyficznej konfiguracji routera CLU DISCOVERY może nie wykryć modułów CLU.

Wykryte moduły zostaną wyświetlone w postaci listy:



Kolor pozycji oznacza:

- **Zielony** – nowo wykryte CLU, który można dodać do projektu;
- **Czerwony** – CLU, które z różnych przyczyn nie może zostać dodane do projektu (wersja nieobsługiwana przez OM itp.);
- **Niebieski** – CLU dodane wcześniej do projektu (dotyczy sytuacji, w której CLU DISCOVERY uruchomiono na istniejącym projekcie).

Moduły można dodać pojedynczo wciskając przycisk *Dodaj* lub za jednym razem wszystkie przy pomocy przycisku *Dodaj wszystkie*.

Po wykonaniu powyższych czynności, w projekcie znajduje się lista obiektów obecnych w systemie – jest możliwa ich konfiguracja.

5. Statusy CLU

Na podstawie wyglądu ikony modułu CLU - znajdującego się w menu obiektów otwartego projektu - użytkownik zostaje poinformowany, jaki jest aktualny status zarówno konfiguracji, jak i połączenia pomiędzy OM i CLU. Dla każdego CLU znajdującego się w projekcie można wyróżnić cztery tryby pracy: normalny, niepodłączony, błąd konfiguracji oraz tryb emergency.

Tryb normalny

CLU znajdujące się w normalnym trybie pracy nie zawiera błędów konfiguracyjnych, a połączenie pomiędzy OM i CLU jest aktywne. Nazwa tego modułu jest wyświetlana w kolorze czarnym, natomiast ikona obrazująca ten stan wygląda następująco:



Jeżeli nazwa danego CLU poprzedzona jest symbolem * oznacza to, iż w projekcie dokonano zmiany w konfiguracji, która nie została jeszcze wysłana do tego CLU.

Niepodłączony

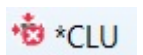
Jeśli pomiędzy modulem CLU i OM nie ma połączenia (brak fizycznego podłączenia lub błąd w konfiguracji LAN), to nazwa CLU wyświetlona będzie w kolorze czerwonym, a ikona tego modułu będzie wyglądać następująco:



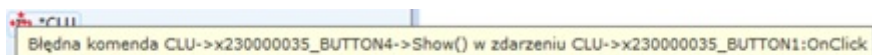
Jeśli CLU jest w stanie niepodłączonym, użytkownika ma możliwość dokonywania oraz zapisywania zmian w projekcie, natomiast nowa konfiguracja nie zostanie wysłana do CLU – jest to możliwe tylko w trybie normalnym.

Błąd konfiguracji

Jeżeli podczas pracy nad projektem zostaną wprowadzone zmiany zawierające błędy konfiguracyjne (np. zostanie utworzone powiązanie z nieistniejącym obiektem lub zostaną wprowadzone nieobsługiwane komendy) to CLU, w którym został wykryty błąd, zostanie przełączone w tryb pracy **Błąd konfiguracji**. Nazwa tego CLU wyświetlana będzie w kolorze czarnym, ale przy ikonie zostanie wyświetlony symbol błędu:



Po najechnięciu myszką na to CLU rozwinie się pole z listą wykrytych błędów.



UWAGA! OM nie pozwala na wysłanie do CLU konfiguracji zawierającej błędy!

Tryb emergency

Jeżeli do CLU zostanie wysłana konfiguracja zawierająca błędy w składni (np. po wysłaniu skryptu w edycji tekstowej) lub w wyniku działania skryptu dojdzie do zawieszenia interpretera LUA, to wtedy CLU zmieni tryb pracy na TRYB EMERGENCY. Nazwa CLU zmieni kolor na pomarańczowy, a przy ikonie pojawi się symbol awarii:



Jeżeli CLU zostało wprowadzone w tryb emergency, należy sprawdzić poprawność ostatnio wprowadzonych zmian i wysłać ponownie konfigurację do CLU.

UWAGA! Moduły CLU wyjęte z pudełka (w stanie fabrycznym) znajdują się w trybie *Emergency!*

6. Obsługa modułów Z-Wave

Bezprzewodowe moduły IOM komunikują się z pozostałymi elementami systemu z wykorzystaniem protokołu Z-Wave. Zarówno z poziomu OM jak i z punktu widzenia sterowania są rozpoznawane i działają dokładnie w taki sam sposób, jak pozostałe moduły w systemie GRENTON.

Korzystanie z modułów Z-Wave w systemie jest możliwe, gdy znajduje się w nim co najmniej jeden moduł CLU wyposażony w kontroler Z-Wave.

UWAGA! Dodanie modułu Z-Wave do systemu powinno odbywać się po umieszczeniu go **w miejscu docelowym instalacji** - wynika to z wymagań dotyczących tworzenia sieci *mesh*, zasięgu działania urządzenia oraz występujących zakłóceń sieci Z-Wave.

6.1. Dodawanie modułów Z-Wave

By moduły IOM Z-Wave były obecne w systemie konieczne jest ich dodanie do CLU. Można to zrobić na dwa sposoby:

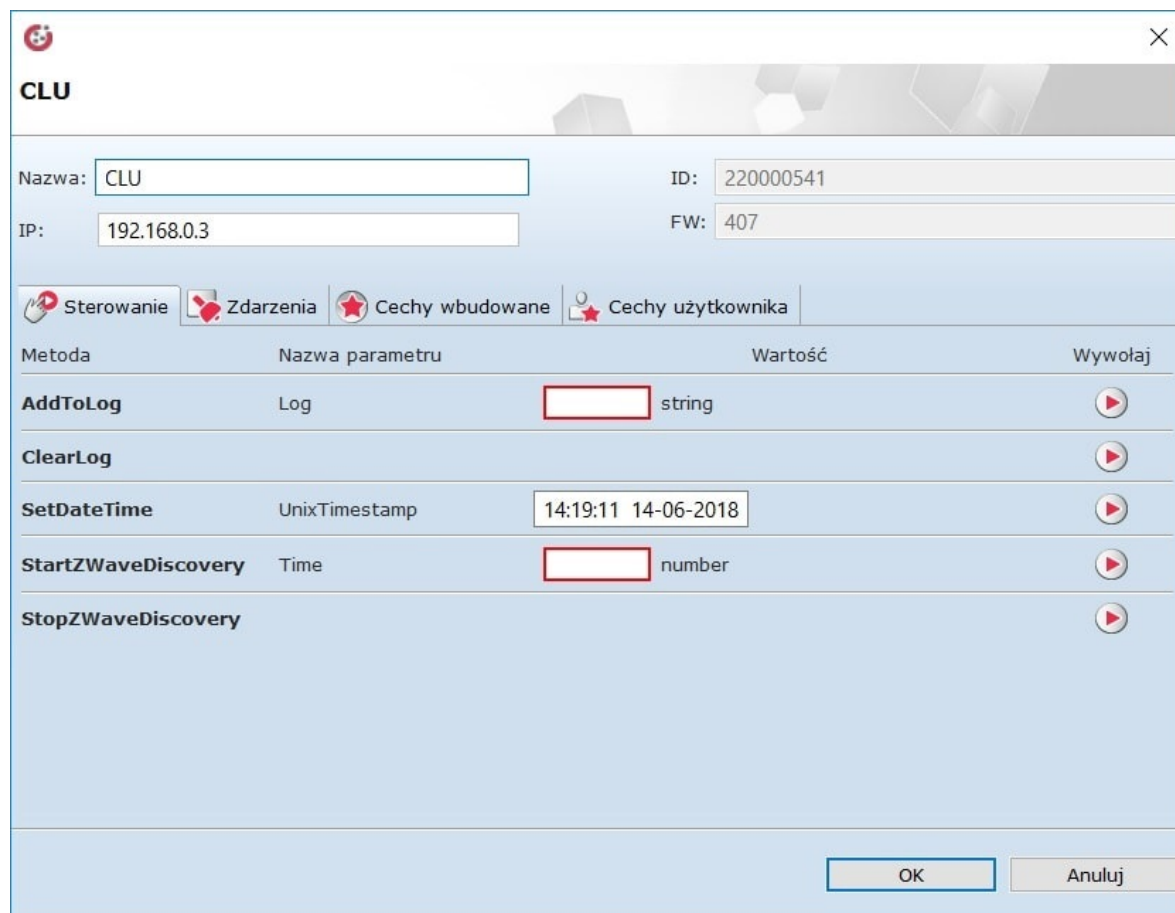
1. **Poprzez przycisk *LINK* znajdujący się na module CLU.** W tym celu konieczne jest wciśnięcie przycisku **Link**, znajdującego się na module CLU z kontrolerem Z-Wave.

Po jego naciśnięciu CLU przechodzi do trybu dodawania modułów – dioda ON mruga cały czas w odstępach 200ms.

Później należy nacisnąć przycisk znajdujący się na dodawanym module Z-Wave. Prawidłowe dodanie modułu zostanie zasygnalizowane przez zapalenie na 1 sekundę diody ON, a następnie przez trzykrotne mrugnięcie diod ON oraz ERR w odstępach 200ms. Po zakończeniu dodawania modułu Z-Wave dioda ON zacznie migać z częstotliwością 500ms. Po zakończeniu dodawania modułów Z-Wave należy wykonać *CLU Discovery* - nowe moduły Z-Wave zostaną dodane do projektu.

2. **Za pomocą oprogramowania *Object Manager*.** Taki sposób dodawania pozwala na zdefiniowanie czasu przez jaki CLU będzie oczekiwało na „przedstawienie się” modułów bezprzewodowych. Rozwiązanie to jest bardzo przydatne, gdy dodawane moduły znajdują się w większej odległości od CLU i potrzeba więcej czasu na naciśnięcie znajdującego się na nich przycisku.

W celu dodania modułów bezprzewodowych przy pomocy OM należy otworzyć konfigurator obiektu dla modułu CLU Z-Wave, do którego będą dodawane moduły bezprzewodowe (dwukrotne kliknięcie w ikonę CLU na liście obiektów). Następnie w zakładce **Sterowanie** konieczne jest podanie czasu (jako parametr) dla metody **StartZWaveDiscovery** i wywołanie tej metody:



The screenshot shows a window titled 'CLU' with a close button in the top right. Below the title bar, there are input fields for 'Nazwa: CLU', 'ID: 220000541', 'IP: 192.168.0.3', and 'FW: 407'. A navigation bar contains four tabs: 'Sterowanie' (selected), 'Zdarzenia', 'Cechy wbudowane', and 'Cechy użytkownika'. Below the tabs is a table with the following columns: 'Metoda', 'Nazwa parametru', 'Wartość', and 'Wywołaj'.

Metoda	Nazwa parametru	Wartość	Wywołaj
AddToLog	Log	<input type="text" value=""/> string	
ClearLog			
SetDateTime	UnixTimestamp	14:19:11 14-06-2018	
StartZWaveDiscovery	Time	<input type="text" value=""/> number	
StopZWaveDiscovery			

At the bottom right of the window are 'OK' and 'Anuluj' buttons.

Podany czas, będzie czasem jaki CLU oczekuje na zgłoszenie się nowych modułów Z-Wave. Po jego upływie wyszukiwanie zostaje zakończone, nawet jeśli żadne moduły nie zostały znalezione. Wpisanie cyfry 0 spowoduje, że wyszukiwanie zostanie zakończone automatycznie po znalezieniu pierwszego nowego modułu.

Po wywołaniu metody **StartZWaveDiscovery** należy nacisnąć przycisk znajdujący się na dodawanym module Z-Wave. Poprawne dodanie modułu zostanie zasygnalizowane przez zapalenie na 1 sekundę diody ON, a następnie przez trzykrotne mrugnięcie diod ON oraz ERR w odstępach 200ms. Po poprawnym dodaniu modułów Z-Wave dioda ON zacznie migać z częstotliwością 500ms. Po zakończeniu dodawania modułów Z-Wave należy wykonać proces *CLU Discovery* - nowe moduły Z-Wave zostaną dodane do projektu.

UWAGA! Wywołanie metody **StopZWaveDiscovery** przerywa wyszukiwanie modułów Z-Wave.

UWAGA! Nie należy dodawać do systemu modułów, które już wcześniej zostały do niego podłączone. W przypadku braku pewności czy dany moduł został wcześniej dodany, w pierwszej kolejności należy wykonać procedurę usunięcia dla tego modułu.

Sytuacja ma się podobnie, gdy moduł Z-Wave był podłączony i nie został usunięty z innego kontrolera - najpierw należy wykonać procedurę usunięcia takiego modułu.

6.2. Usuwanie modułów Z-Wave

By moduł bezprzewodowy przestał się pojawiać w konfiguracji systemu musi on zostać usunięty z CLU Z-Wave.

W tym celu konieczne jest wciśnięcie przycisku `unLink`, znajdującego się na module CLU z kontrolerem.

Po jego naciśnięciu CLU przechodzi do trybu usuwania modułów – dioda ERR mruga cały czas w odstępach 200ms.

Następnie należy nacisnąć przycisk znajdujący się na module bezprzewodowym, który ma zostać usunięty. Prawidłowe usunięcie modułu zostanie zasygnalizowane przez trzykrotne mrugnięcie diod ON oraz ERR w odstępach 200ms. Po zakończeniu usuwania modułu Z-Wave dioda ERR zgaśnie, a ON zacznie migać z częstotliwością 500ms. Ostatnim krokiem będzie przeprowadzenie *CLU Discovery* - usunięte moduły zostaną wyszarzone.

6.3. Brak komunikacji z modułem Z-Wave - mechanizm zliczania niepowodzeń komunikacji i blokowania komunikacji urządzenia w sieci Z-Wave

UWAGA! Przedstawiony mechanizm dostępny jest dla CLU od wersji **04.07.41(183201)**

Niepowodzenia w komunikacji z urządzeniem Z-Wave mogą nastąpić w sytuacji, gdy:

- moduł Z-Wave jest uszkodzony,
- brak zasilania (230V) na module / wyczerpania baterii zasilającej moduł,
- urządzenie pracuje na pograniczu zasięgu z kontrolerem / nie jest w zasięgu kontrolera,
- kontroler (CLU) po wysłaniu rozkazu nie otrzyma potwierdzenia od urządzenia (ACK).

Informacje o statusie urządzenia w sieci Z-Wave można odczytać z pozycji Object Managera przy pomocy obiektu ZWAVE_CONFIG danego modułu Z-Wave

UWAGA! Obiekty ZWAVE_CONFIG nie są dostępne dla wszystkich modułów Z-Wave - posiadają je moduły Grenton Z-Wave oraz wybrane moduły, które są wspierane przez system Grenton.

Dla danego obiektu dostępne są następujące cechy:

CLU_221000005->x390110083_ZWAVE_CONFIG1

Nazwa: x390110083_ZWAVE_CONFIG1 Źródło/Odbiornik:
 Identyfikacja: 390110083 5 Typ: ZWAVE_CONFIG

Sterowanie
 Schematy konfiguracji
 Zdarzenia
 Cechy wbudowane
 Statystyki

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
NodeID	8			[0-232]
Banned	0			[0-1]
FailCount	0			[0-65536]

Auto odświeżanie

- **NodeID** - Numer modułu (węzła) w sieci Z-Wave (nadawany dla każdego modułu Z-Wave po dodaniu go do kontrolera)
- **Banned** - Informacja o zablokowaniu komunikacji Z-Wave z modułem
- **FailCount** - Liczba nieudanych prób komunikacji z modułem Z-Wave

Mechanizm zliczania niepowodzeń w komunikacji:

- W przypadku wystąpienia niepowodzenia komunikacji z modułem (brak odpowiedzi, potwierdzenia itp.) następuje inkrementacja cechy **FailCount** obiektu ZWAVE_CONFIG danego urządzenia Z-Wave.

CLU_221000005->x390110083_ZWAVE_CONFIG1

Nazwa: Źródło/Odbiornik:

Identyfikacja: 5 Typ:

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
NodeID	8			[0-232]
Banned	0			[0-1]
FailCount	1			[0-65536]

Auto odświeżanie

- Ponowna próba wysłania rozkazu do urządzenia ponawiana jest co 15 sekund - wykonywane są 3 próby komunikacji z urządzeniem.
- W przypadku 3 prób niepowodzenia komunikacji z modułem cecha **Banned** zostaje ustawiona na wartość 1 a cała komunikacja z modułem zostaje zablokowana.

CLU_221000005->x390110083_ZWAVE_CONFIG1

Nazwa: Źródło/Odbiornik:

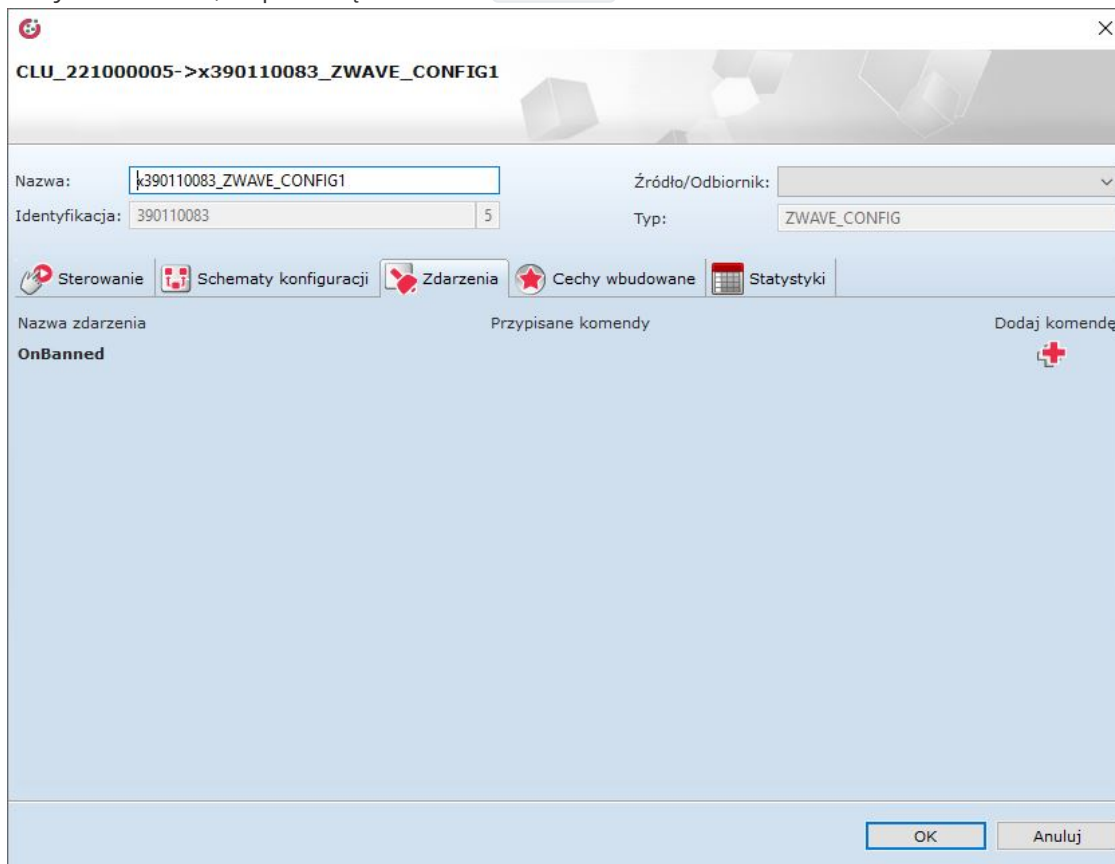
Identyfikacja: 5 Typ:

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
NodeID	8			[0-232]
Banned	1			[0-1]
FailCount	4			[0-65536]

Auto odświeżanie

Mechanizm blokowania komunikacji z modułem

- W momencie ustawienia cechy **Banned** na wartość 1, następuje zablokowanie komunikacji z urządzeniem Z-Wave - oznacza to, że wszystkie wywołania akcji na urządzeniu (tj. zmiana stanu wyjścia, zapytanie o parametry) nie są wysyłane przez CLU do zablokowanego modułu.
- Istnieje możliwość przypisania dowolnej akcji w momencie zablokowania komunikacji z danym modułem, za pomocą zdarzenia **OnBanned**



- Do zbanowanego modułu, co 1,5 minuty wysyłane jest krótkie zapytanie (NOP):
 - jeżeli moduł nie potwierdzi otrzymanie zapytania, cecha **Banned** nadal przyjmuje wartość 1, a kolejne zapytanie zostaje ponawiane co 1,5 minuty,

UWAGA! W przypadku gdy więcej niż jeden moduł zostanie zbanowany, wtedy wysyłanie NOP odbywa się co 1,5 minuty do **kolejnego zabowanego modułu**.
Przykład:

zbanowane 3 moduły A,B,C

CLU - NOP -> moduł A

1,5 minuty przerwy

CLU - NOP -> moduł B

1,5 minuty przerwy

CLU - NOP -> moduł C

1,5 minuty przerwy

CLU - NOP -> moduł A

itd.

- jeżeli moduł potwierdzi otrzymanie zapytania (ACK), cecha **Banned** zmienia wartość na 0 - oznacza to, że możliwe jest ponowne wysyłanie rozkazów do danego urządzenia.

CLU_221000005->x390110083_ZWAVE_CONFIG1

Nazwa: x390110083_ZWAVE_CONFIG1 Źródło/Odbiornik:
 Identyfikacja: 390110083 5 Typ: ZWAVE_CONFIG

Sterowanie
 Schematy konfiguracji
 Zdarzenia
 Cechy wbudowane
 Statystyki

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
NodeID	8			[0-232]
Banned	0			[0-1]
FailCount	4			[0-65536]

Auto odświeżanie

- Istnieje możliwość ręcznego usunięcia blokady - za pomocą metody **RemoveBan**.

CLU_221000005->x390110083_ZWAVE_CONFIG1

Nazwa: x390110083_ZWAVE_CONFIG1 Źródło/Odbiornik:
 Identyfikacja: 390110083 5 Typ: ZWAVE_CONFIG

Sterowanie
 Schematy konfiguracji
 Zdarzenia
 Cechy wbudowane
 Statystyki

Metoda	Nazwa parametru	Wartość	Wywołaj
RemoveBan			<input type="button" value="▶"/>
ClearFailCount			<input type="button" value="▶"/>

- Po wywołaniu tej metody cecha **Banned** zmienia wartość na 0 - oznacza to, że możliwe jest ponowne wysyłanie rozkazów do danego urządzenia.

UWAGA! `RemoveBan` nie jest jednoznaczne z ponowną poprawną komunikacją z modułem – umożliwia ponowne wysłanie rozkazu/zapytania do modułu! W przypadku niepowodzenia cały proces blokowania jest rozpoczynany na nowo!

- W przypadku wystąpienia niepowodzenia komunikacji z modułem, cały mechanizm (zliczania niepowodzeń w komunikacji oraz blokowania) rozpoczyna się od początku.

Należy pamiętać, że w przypadku odblokowania komunikacji z modułem cecha `FailCount` nie jest zerowana - można to zrobić przy pomocy metody **ClearFailCount**.

6.4. Wskazówki konfiguracyjne sieci Z-Wave

Podczas tworzenia sieci Z-Wave istotne jest by:

- Konfiguracja sieci Z-Wave odbywała się po montażu urządzeń.

Z uwagi na fakt, że sieć Z-Wave definiowana jest statycznie, urządzenia Z-Wave należy linkować, gdy znajdują się w swoich docelowych lokalizacjach. Zmiana położenia urządzeń Z-Wave po ich dodaniu może powodować nieprzewidziane problemy z komunikacją w sieci Z-Wave - z wszystkimi urządzeniami!

- Antena (w modułach, które ją posiadają) nie była zwinięta ani zawinięta wokół modułu.

Antena powinna być umieszczona możliwie w kierunku „od modułu”

- Wybudzanie modułów bateryjnych nie następowało w tym samym czasie.

Wybudzanie modułów w tym samym czasie doprowadza do pojawienia się opóźnień w działaniu. W celu uniknięcia opisaney sytuacji, należy dla wszystkich urządzeń stosować inne czasy wybudzania (w obiekcie `ZWAVE_WAKEUP` dla modułów bateryjnych) i dobierać je w taki sposób, by ustawione czasy miały możliwie jak największą „najmniejszą wspólną wielokrotność”, np.: 57min, 58min, 59min, 60min, 61min, itd...

- W sieci Z-Wave nie znajdowały się nieaktywne moduły (uszkodzone lub niepoprawnie usunięte).

Podlinkowany moduł, którego brakuje w systemie powoduje ciągłe próby odnowienia komunikacji z nim, co z kolei może wprowadzać chwilowe opóźnienia i braki komunikacji także z innymi urządzeniami.

UWAGA! Dla CLU Z-Wave umieszczonego w skrzynce/szafie zaleca się wykorzystanie dłuższych anten oraz wyniesienie ich poza rozdzielnię.

6.5. Czyszczenie informacji o węzłach

Istnieje możliwość jednoczesnego usunięcia wszystkich modułów Z-Wave z CLU. Do tego celu służy funkcja `HardReset` - [patrz pkt XI.1.](#)

7. Wysyłanie konfiguracji do CLU



Konfiguracja przechowywana jest w OM i dopóki nie zostanie przesłana do CLU, nie jest brana pod uwagę w działaniu systemu. Żeby wysłać konfigurację do CLU należy nacisnąć przycisk **wyślij konfigurację** w menu .

Object Manager wykryje, na których CLU została dokonana zmiana i wyśle konfigurację.

UWAGA! Po wysłaniu konfiguracji CLU zostaną zrestartowane, przez co lampy podłączone do systemu mogą zgasnąć, a system może przez kilka sekund nie reagować na naciśnięcia wyłączników itp.

8. Wartości początkowe cech

Każdy obiekt w systemie posiada swoją własną listę cech, z których część może być ustawiana. Cechy mogą być ustawiane w trakcie startu systemu (restart CLU), dzięki czemu możliwa jest jednorazowa konfiguracja zachowania obiektów (np. ustawienie przycisków panelu dotykowego jako bistabilne, monostabilne). Wartości początkowe cech ustawiane są w zakładce: *Cechy wbudowane* w formacie obiektu (CLU, wejścia, wyjścia):

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
Mode	0	Monostabilny		0,1,2
HoldDelay	1000	1000	ms	[0-5000]
HoldInterval	100	50	ms	[0-2000]
Value	0		bool	0,1

By ustawić wybraną cechę, należy w odpowiednim polu, w kolumnie **wartość początkowa** wpisać żadaną wartość, a następnie wysłać konfigurację do CLU.

9. Tworzenie podstawowych powiązań

Wywoływanie reakcji w systemie (np. załączenie oświetlenia po wciśnięciu klawisza) jest realizowane przez tworzenie powiązań pomiędzy obiektami. Z reguły są to powiązania pomiędzy wejściem (np. wyłącznik) a wyjściem (lampa). System nie ogranicza jednak w żaden sposób tworzenia powiązań i umożliwia ich tworzenie pomiędzy zdarzeniami wyjść ze zdarzeniami

dowolnych innych obiektów, przez co możliwe jest, np. włączenie oświetlenia LED-owego w momencie wyłączenia lampy głównej.

Powiązania mogą być tworzone na dwa sposoby:

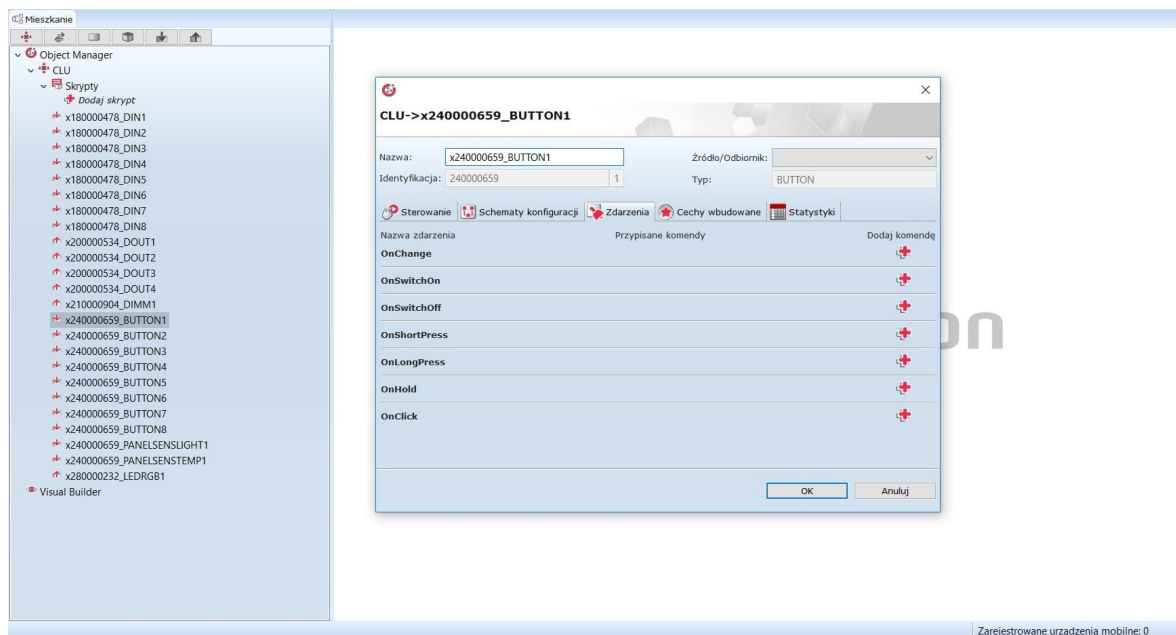
- Poprzez użycie schematów konfiguracji – pozwala to na szybkie tworzenie typowych powiązań typu wyłącznik-lampa;
- Poprzez ręczne tworzenie powiązań zdarzenie-metoda – co zapewni ogromną elastyczność w tworzeniu logiki systemu.

W celu utworzenia powiązania przy pomocy schematu konfiguracji należy wykonać następujące czynności:

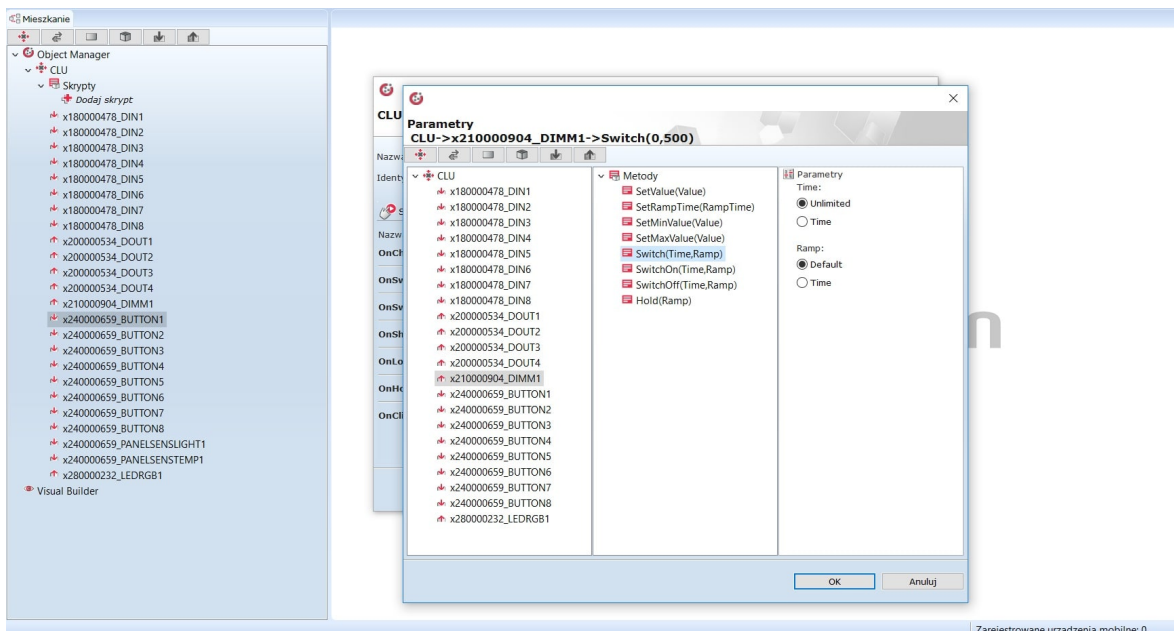
- Kliknąć na wejście, które ma zostać powiązać z wyjściem;
- Przejść do zakładki schematu konfiguracji, z listy wybrać interesujący schemat;
- Klikając **Dodaj powiązania** wybrać wyjścia, które mają być wyzwalone;
- Skonfigurować pozostałe wejścia i wysłać konfigurację do CLU.

By ręcznie utworzyć powiązanie zdarzenie-metoda:

- Z listy obiektów w systemie wybrać interesujący obiekt kliknąć na niego dwukrotnie;
- Przejść do zakładki **zdarzenia**:

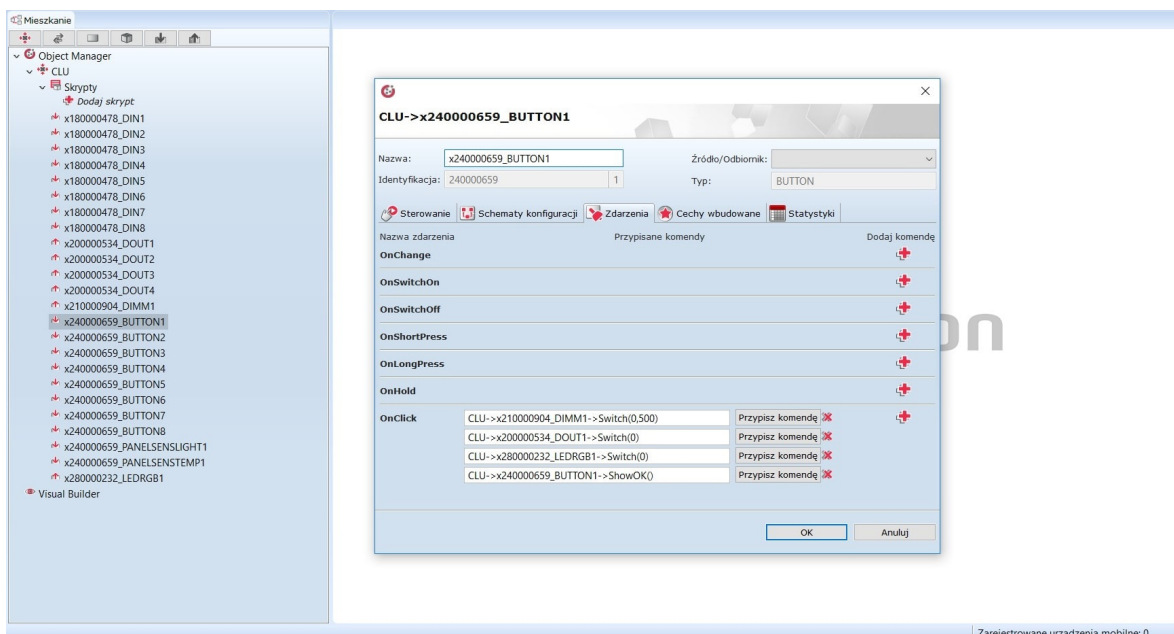


- Znaleźć na liście zdarzenie, które ma zostać powiązane i kliknąć **+**;
- W formatce wybierania metody wybrać kolejno obiekt, metodę i parametry:



- Skonfigurować pozostałe zdarzenia i wysłać konfigurację do CLU.

Do każdego zdarzenia można dodać do 4 metod wyjść. Jeśli konieczne jest dodanie większej liczby metod lub warunków, proponuje się utworzenie skryptu.



10. Przeprowadzanie aktualizacji

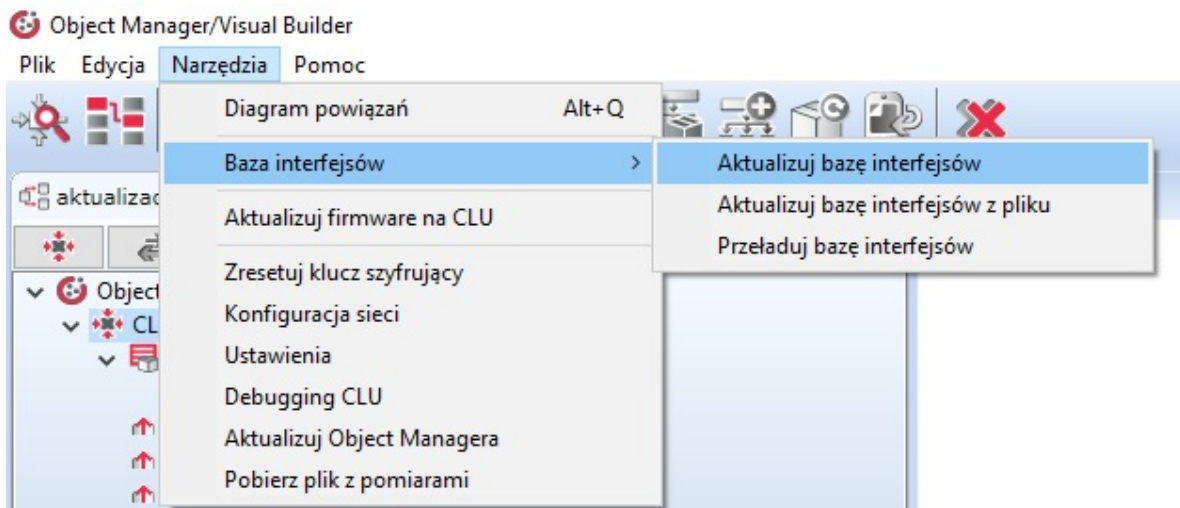
10.1. Proces aktualizacji bazy interfejsów

Jeśli przy pierwszym uruchomieniu Object Managera zaznaczona została opcja *automatycznego aktualizowania bazy interfejsów*, nie ma konieczności jej dodatkowego uruchamiania. W przeciwnym wypadku, należy pamiętać o regularnym przeprowadzaniu aktualizacji. Aktualizacja bazy interfejsów powinna zostać wykonana zawsze przed aktualizacją oprogramowania danego modułu Grenton, a do jej przeprowadzenia niezbędne jest połączenie z internetem (aktualizacja odbywa się z serwera).

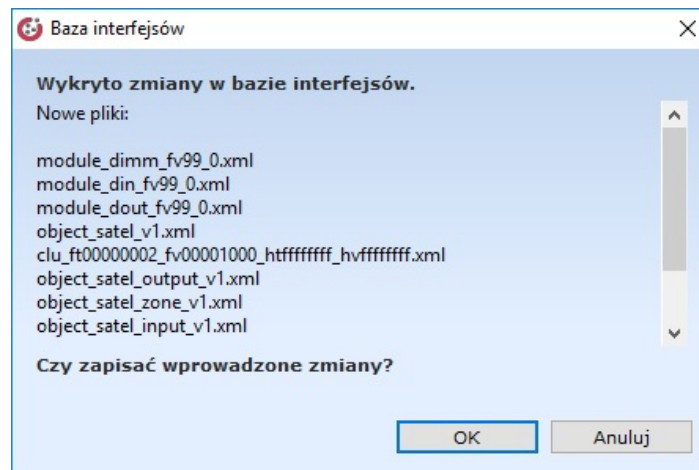
W celu zaktualizowania bazy interfejsów w Object Managerze należy:

- Z paska menu wybrać *Narzędzia*;

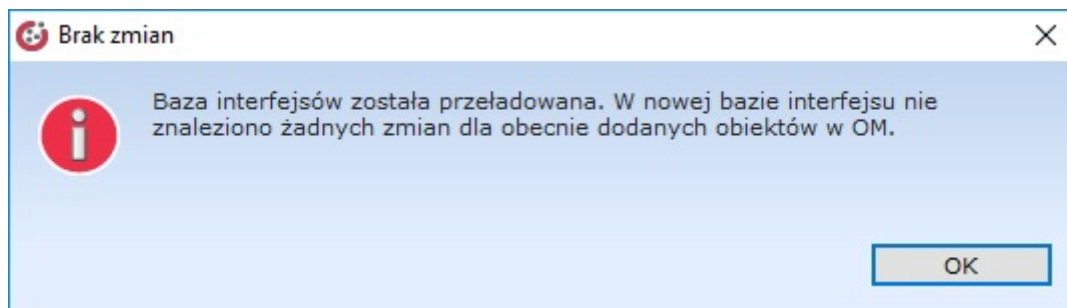
- Zaznaczyć pozycję *Baza interfejsów*;
- Z wyświetlonej listy wybrać *Aktualizuj bazę interfejsów*:



- Po chwili pojawi się okno z wykrytymi zmianami w bazie interfejsów, które należy zaakceptować, klikając przycisk *OK*:

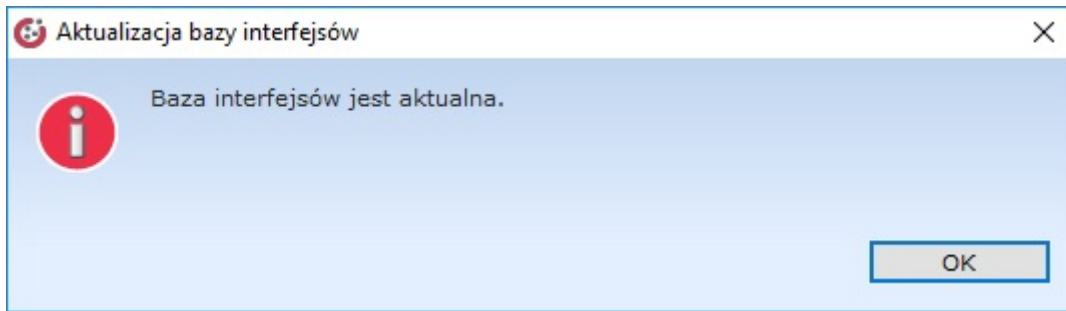


- Następnie wyświetlone zostanie okno informujące o zakończeniu przeładowywania bazy interfejsów:



- Ostatnim etapem jest przesłanie konfiguracji do jednostki centralnej, które następuje automatycznie.

UWAGA! Jeżeli konfiguracja jest aktualna, to po wyborze opcji: *Aktualizuj bazę interfejsów* wyświetlony zostanie komunikat:



10.2. Proces aktualizacji firmware'u CLU

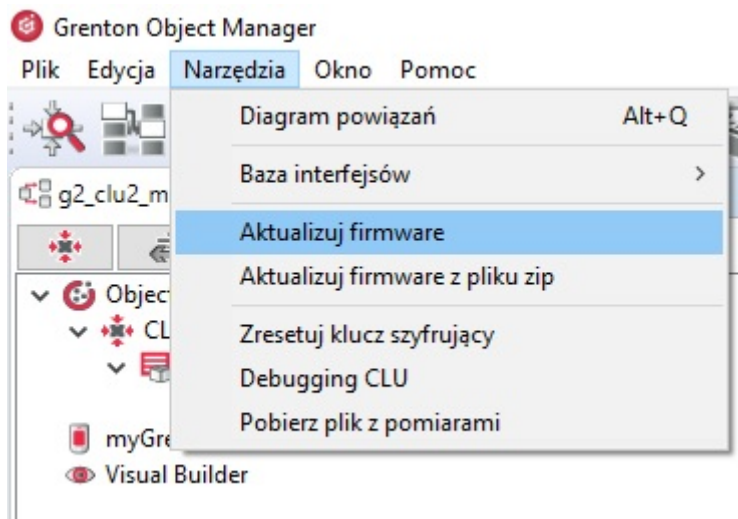
Aktualizację firmware'u przeprowadza się w celu dodania obsługi nowych urządzeń oraz zwiększenia możliwości systemu. Więcej szczegółów można znaleźć w Release Notes.

UWAGA!

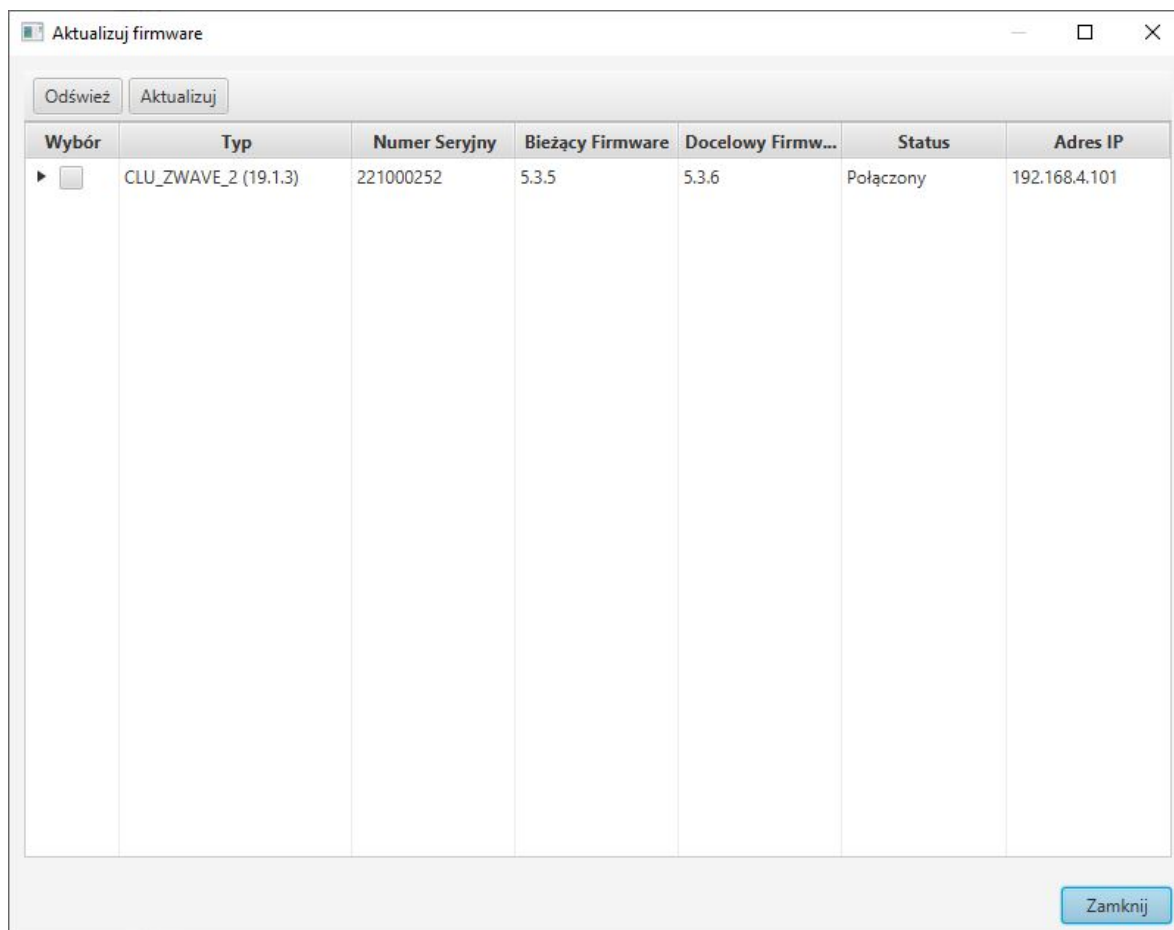
Aktualizacja firmware CLU 2.0 możliwa jest tylko w Object Managerze w wersji 1.3.0.1927 lub wyższej!

Chcąc zaktualizować firmware na CLU należy:

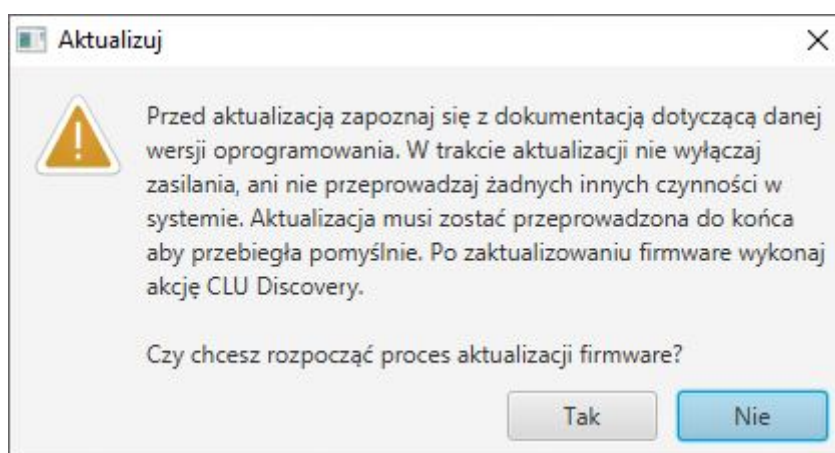
- Z paska menu wybierz **Narzędzia**
- Wybierz opcję **Aktualizuj firmware**



- Zaznacz obiekt typu `CLU_ZWAVE_2`. Zaznaczenia pola wyboru jest możliwe tylko w przypadku, gdy bieżący firmware na CLU jest nieaktualny.



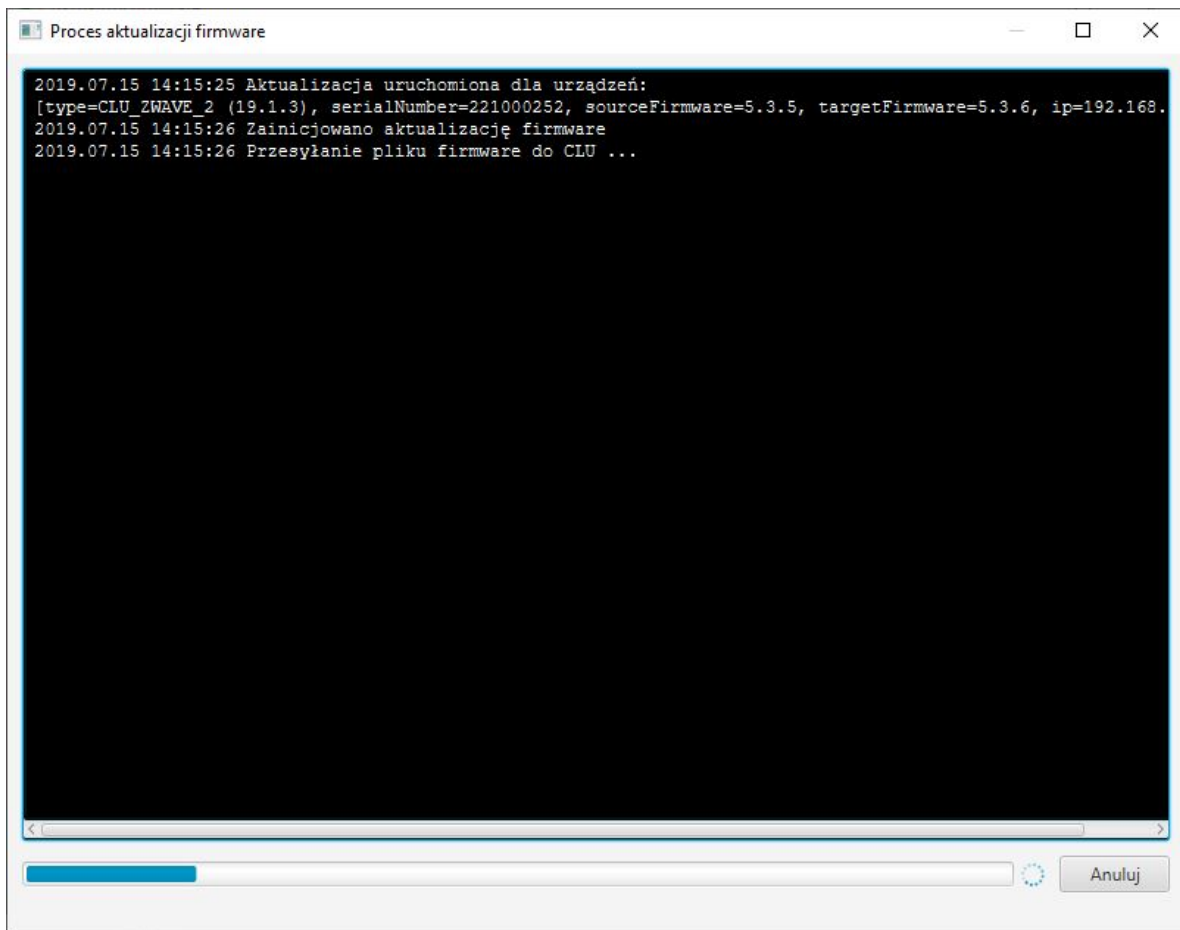
- Wybierz opcję **Aktualizuj**. Po zapoznaniu się z komunikatem, wybierz „Tak” aby zaakceptować.



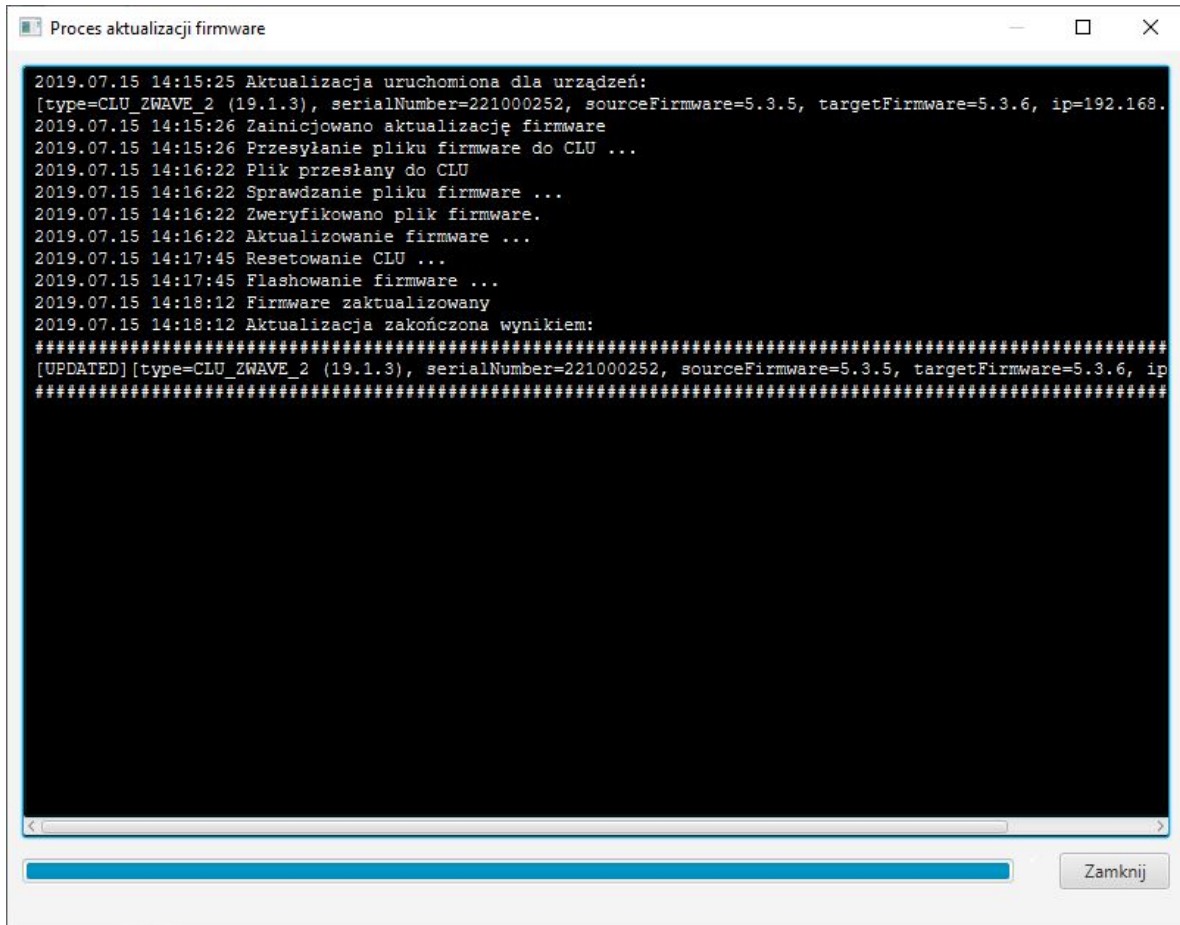
- Po zaakceptowaniu rozpocznie się proces aktualizacji.

UWAGA!

Podczas procesu aktualizacji nie wolno wyłączać zasilania ani przeprowadzać innych czynności w systemie. Aby aktualizacja przebiegła pomyślnie należy przeprowadzić ją do końca.



- W przypadku poprawnej aktualizacji, przy zaktualizowanym CLU pojawia się informacja [UPDATED].



- Aby zakończyć proces aktualizacji należy zamknąć okno - klikając przycisk **Zamknij**

Jeżeli aktualizacja przebiegła pomyślnie, wersja firmware'u po odświeżeniu powinna być zgodna z 05.03.06.

UWAGA !!

Po zakończeniu aktualizacji należy wykonać CLU Discovery.

Istnieje możliwość aktualizacji większej ilości CLU w ramach jednego procesu. W tym celu na liście wyboru należy zaznaczyć wszystkie CLU, które mają zostać zaktualizowane.

10.3. Proces aktualizacji firmware'u modułów z serii 2.0

UWAGA!

Proces aktualizacji firmware urządzeń możliwy jest tylko dla modułów z serii 2.0 !

Aktualizacja modułów z serii 2.0 przebiega podobnie jak w przypadku aktualizacji firmware CLU. Przed rozpoczęciem aktualizacji należy pamiętać o następujących kwestiach:

- Aktualizacja firmware danego modułu jest tylko możliwa w przypadku, gdy firmware na CLU jest aktualny. W innym przypadku należy również zaznaczyć CLU, które zostanie zaktualizowane w pierwszej kolejności.
- Aktualizacja odbywa się dla wszystkich modułów tego samego typu. Zaznaczając dany moduł, zostają zaznaczone wszystkie moduły tego samego typu znajdujące się na liście (o ile występują).
- W momencie rozpoczęcia procesu, nie ma możliwości zatrzymania aktualizacji dla urządzenia które w danym momencie jest aktualizowane. Przerwanie aktualizacji nastąpi po zakończeniu procesu dla danej grupy urządzeń (anulowana dla kolejnej grupy modułów).
- W niektórych przypadkach, aktualizacja danego urządzenia może przebiegać wieloetapowo. W takim przypadku po zakończeniu procesu aktualizacji należy sprawdzić, czy nie jest dostępna kolejna nowa wersja firmware'u dla danego modułu.

UWAGA !!

Po zakończeniu aktualizacji należy wykonać CLU Discovery.

11. Inne operacje na systemie

Czyszczenie konfiguracji

Użytkownik zawsze ma możliwość wyczyszczenia konfiguracji dowolnego CLU znajdującego się w systemie. W celu wyczyszczenia konfiguracji na wybranym CLU, najpierw musimy je zaznaczyć, a następnie kliknąć w ikonę czyszczenia.

Czyszczenie konfiguracji kasuje wszystkie dokonane zmiany i ustawienia oraz ustawia wartości domyślne (default value).

UWAGA! Po wyczyszczeniu konfiguracji na danym CLU, powiązania pomiędzy obiektami innych CLU, a obiektami CLU czyszczonego, zostaną utracone!

Pobieranie konfiguracji z istniejącego obiektu

Object Manager pozwala na pobranie konfiguracji znajdującej się w istniejącym i działającym systemie. Konfigurację można pobrać tylko i wyłącznie podczas tworzenia nowego czystego projektu – nie jest możliwe pobranie konfiguracji dla projektu, który posiada już jakieś dane.

Dodawanie nowego modułu CLU lub IOM

Po zainstalowaniu nowego modułu należy dodać go do systemu. Moduł musi zostać wpięty do magistrali systemowej (przed wpięciem nowego modułu należy odłączyć zasilanie magistrali). W przypadku modułów Z-Wave należy dodać je do kontrolera - [patrz pkt VI.6.1.](#) Po poprawnym zainstalowaniu modułu, należy uruchomić CLU DISCOVERY, spowoduje to automatyczne wyszukanie i dodanie nowego modułu. Jeśli w systemie istnieją nieużywane wejścia/wyjścia, system uruchomi listę umożliwiającą przypisanie nieaktywnych wejść/wyjść do wejść/wyjść z nowego modułu. Po wykonaniu powyższej procedury moduł pojawi się na liście obiektów.

Wymiana modułu IOM (wejść/wyjść)

Jeśli dany moduł zostanie wymieniony na inny, ale o takich samych parametrach (taki sam typ i taka sama ilość wejść/wyjść), należy dokonać również wymiany modułu w projekcie w programie Object Manager. Po poprawnym zainstalowaniu i podpięciu modułu należy uruchomić funkcję CLU DISCOVERY. System automatycznie wyszuka i rozpozna nowy moduł, oraz automatycznie przypisze do niego wejścia/wyjścia ze „starego” modułu. Po wyszukaniu zostanie wyświetlona lista z przypisaniem wejść/wyjść pomiędzy wymienionymi modułami oraz opcja potwierdzenia i zaakceptowania zmiany. W przypadku zaakceptowania zmian, na liście obiektów nic się nie zmieni, a wszystkie przypisania dokonają się automatycznie. Brak akceptacji spowoduje pojawienie się na liście obiektów nowych pozycji, przy czym jednocześnie również wyświetlane będą nieaktywne wejścia/wyjścia (oznaczone kolorem szarym).

Przełączenie modułu z jednego CLU do innego znajdującego się w tym samym systemie.

W sytuacji gdy zachodzi konieczność przełączenia modułu IOM z jednego CLU do drugiego, należy przełączyć fizycznie moduł (przełączyć kable), a następnie wykonać funkcję CLU DISCOVERY, która uaktualni listę modułów we wszystkich CLU.

VII. Zaawansowane funkcje konfiguracyjne

1. Kontenery

W celu łatwiejszego zarządzania dostępnymi wejściami/wyjściami OM posiada funkcjonalność kontenerów, która pozwala grupować wejścia/wyjścia zgodnie z potrzebami użytkownika.

Kontener może być wykorzystany na przykład do posortowania wejść/wyjść zgodnie z ich funkcjami (oświetlenie, ogrzewanie itp.) lub też umiejscowieniem w budynku (salon, kuchnia itp.).

Nowy kontener tworzy się poprzez: kliknięcie w menu górnym ikony kontenera lub kliknięcie prawym przyciskiem myszy na drzewku obiektów i wybranie opcji *Nowy kontener*. W drzewie, na poziomie kontenera głównego, pojawi się ikona nowego kontenera. Należy nadać mu nazwę, ale trzeba pamiętać, że nie może zawierać polskich liter.

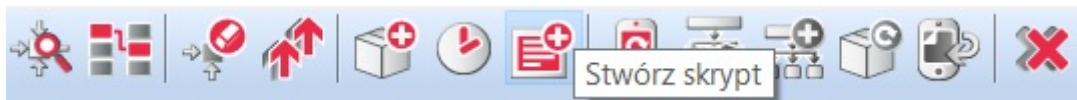
Wejścia/wyjścia do kontenerów przypisuje się przez: przeciągnięcie z CLU lub po kliknięciu na nim prawym przyciskiem myszy i wybraniu opcji *Przenieś do kontenera*.

2. Skrypty

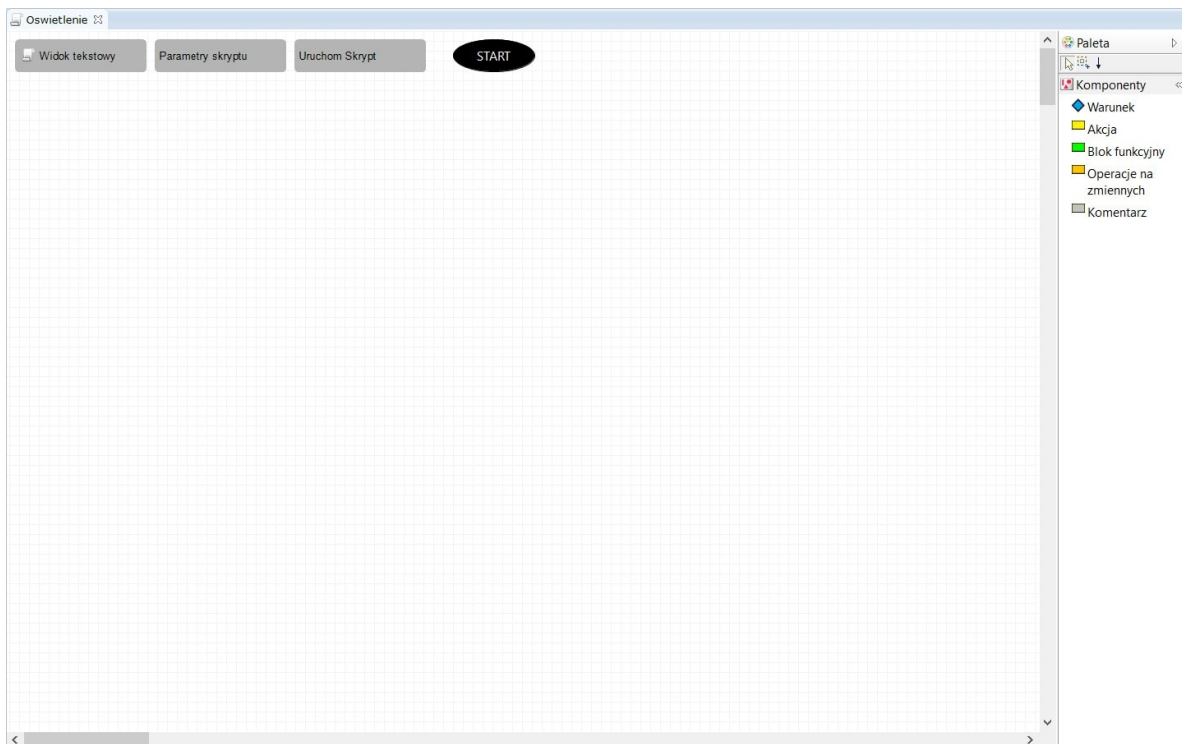
Skrypty umożliwiają tworzenie bardzo rozbudowanej logiki z wykorzystaniem instrukcji warunkowych, pętli czy zmiennych. Dają tym samym możliwość tworzenia zaawansowanych scen, które modyfikują działanie w zależności od warunków zewnętrznych.

Utworzone skrypty widoczne są w systemie jako metody CLU i mogą być wywoływane poprzez przypisanie do zdarzeń dowolnego obiektu, a także wywoływane z poziomu innych skryptów.

Skrypty tworzy się klikając na CLU, na którym skrypt ma być przechowywany. W tym celu należy uruchomić, a następnie wybrać opcję: **Nowy skrypt** w menu akcji, zgodnie z poniższym rysunkiem:



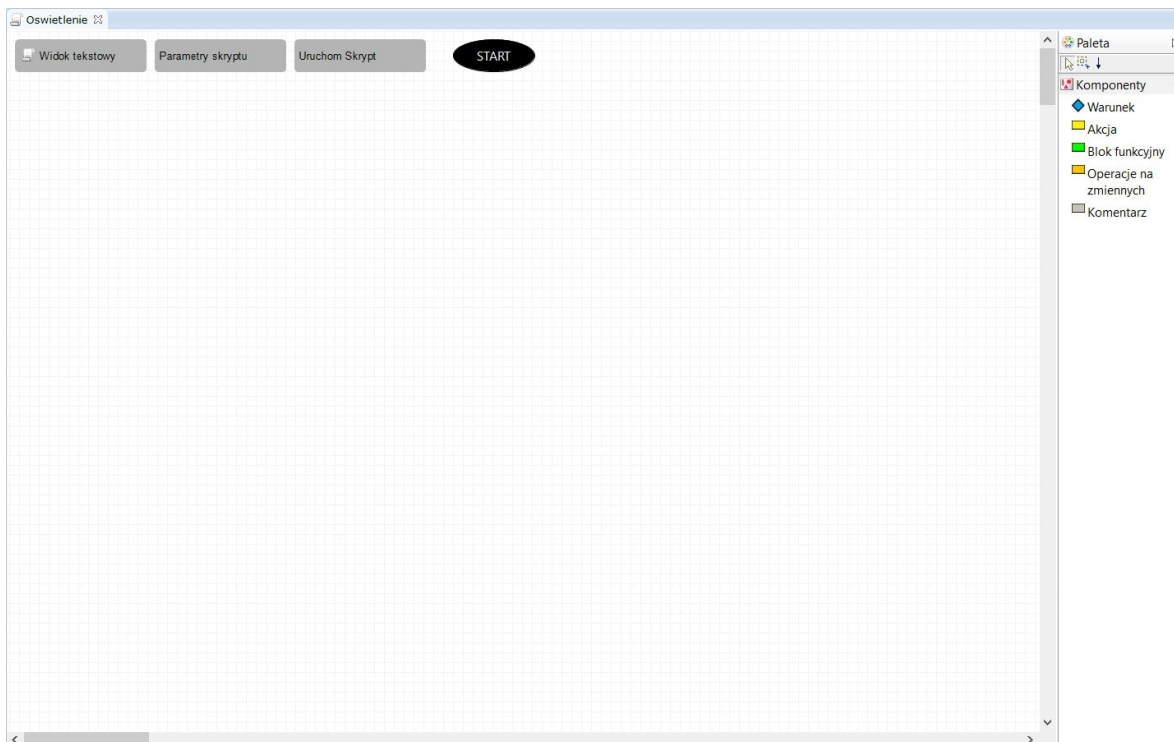
Po podaniu nazwy skryptu (bez polskich znaków) w zakładce otworzy się Script Builder przy pomocy którego można stworzyć skrypt. Script Builder może pracować w dwóch trybach: graficznym i tekstowym. Po utworzeniu nowego skryptu, Script Builder uruchamia się w trybie graficznym, z którego możemy przejść do trybu tekstowego klikając przycisk **Edycja** zgodnie z poniższym rysunkiem.



UWAGA! Przejście z trybu graficznego w tryb tekstowy jest nieodwracalne. Gdy w edytorze graficznym stworzony został jakiś skrypt, zostanie on przekonwertowany na postać tekstową, jednak po dokonaniu edycji w edytorze tekstowym powrót do edytora graficznego nie będzie możliwy.

A. Tworzenie skryptów w trybie graficznym

Po otwarciu pojawia się czysty arkusz.



Z prawej strony arkusza znajduje się lista komponentów. Rozkazy można wstawiać z listy poprzez przeciągnięcie do arkusza. Po upuszczeniu rozkazu na arkuszu otwiera się okno dialogowe, które umożliwia podanie parametrów rozkazu lub instrukcji warunkowych. Po dodaniu do arkusza nowego komponentu, automatycznie tworzone jest połączenie pomiędzy ostatnim wstawionym komponentem (lub polem `Start` jeśli jest to pierwszy komponent), a obecnie wstawianym. Rozkazy wykonywane są w kolejności zgodnej z połączeniami – rozpoczynając od startu. Zmiana kolejności wykonywania rozkazów jest możliwa po usunięciu istniejącego połączenia i poprowadzeniu nowego (zgodnie z żądaną kolejnością), za pomocą narzędzia `Connection`.



UWAGA! Pozostawienie na arkuszu komponentu, który nie jest połączony z innymi komponentami, traktowane jest jako błąd i zostanie wyświetlone jako błąd konfiguracji dla CLU, na którym skrypt został utworzony.

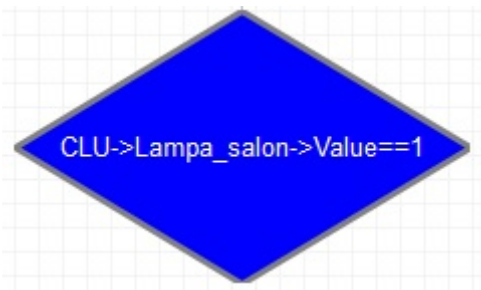
W Script Builderze wykorzystywane są następujące komponenty:

Akcja

`CLU_1->Lampa_salon->SwitchOn(0)`

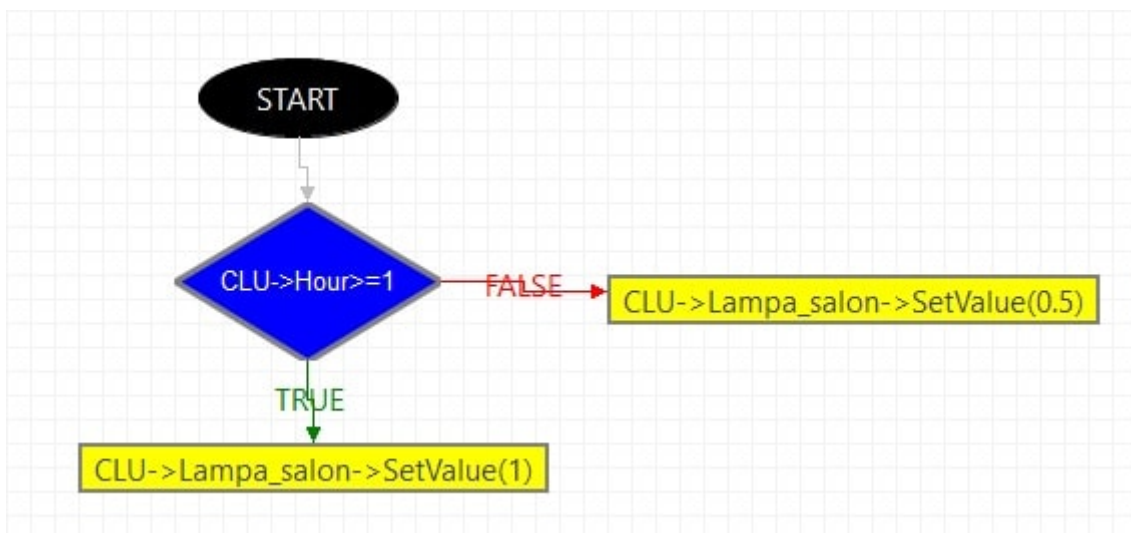
Blok, w którym wpisywany jest rozkaz, jaki ma zostać wykonany. Rozkazem może być wywołanie metody, ale również zmiana wartości oraz wywołanie skryptu. Po przeciągnięciu ikony akcji do arkusza otwiera się okno z listą obiektów oraz ich metodami. Skrypty są dostępne na liście po kliknięciu na CLU, na którym są umieszczone jako metody CLU.

Warunek

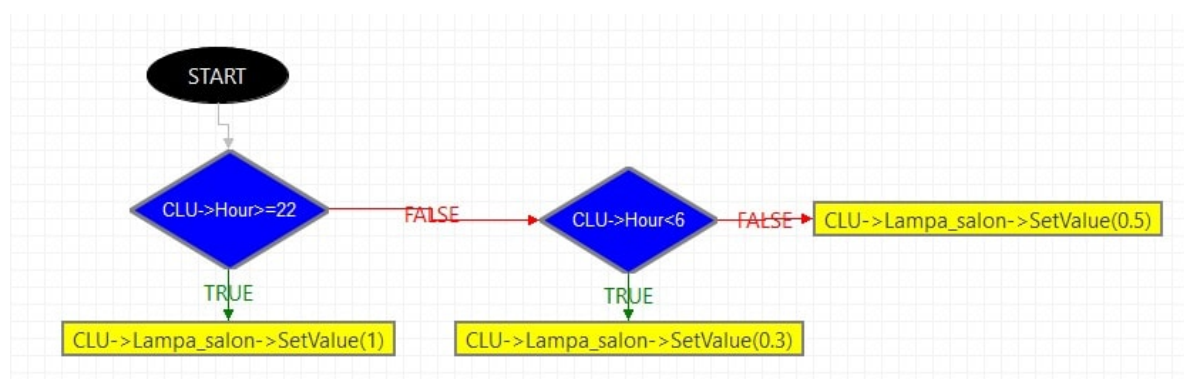


Blok logiczny realizujący funkcję `IF then ELSE`. Użycie tego bloku umożliwia uzależnienie wykonywanej akcji od warunków, np. jeśli jest ciemno, to zapal światło, jeśli nie, to wyłącz. Po przeciągnięciu do arkusza, w parametrach bloku należy wpisać warunek, jaki ma być spełniony. Po dodaniu komponentu *Warunek* należy wstawić komponent *Akcja* lub *Operacje na zmiennych*. OM zapyta, czy polecenie ma być wykonane, gdy warunek zostanie spełniony (`true`) lub odwrotnie - gdy nie zostanie spełniony (`false`). Do warunku można dołączyć dwie akcje, z których jedna zostanie wykonana, gdy warunek będzie spełniony, a w innym przypadku zostanie wykonana druga akcja. Przypisanie `true` / `false` można zmienić klikając dwukrotnie na strzałkę.

Poniższy rysunek pokazuje prostą instrukcję warunkową, która uzależnia natężenie oświetlenia w zależności od godziny.



Warunki mogą być łączone kaskadowo, dzięki czemu istnieje możliwość implementacji operatora `and`, czyli wykonywania akcji, gdy kilka warunków zostanie spełnionych. Poniższy diagram pokazuje przykład zastosowania takiego łączenia:



Warunki mogą porównywać dowolną cechę obiektu lub parametr skryptu z liczbą, tekstem, inną cechą lub parametrem skryptu.

Blok funkcyjny

delay(500)

Zawiera instrukcje wywoływane wewnątrz skryptu, które mogą być wykorzystane przy tworzeniu bardziej zaawansowanych scen (funkcja *Delay*) oraz debugingu (funkcja *Print*). Po przeciągnięciu ikony bloku do arkusza otwiera się okno wyboru. Okno zawiera listę bloków funkcyjnych, na której znajdują się:

- **DELAY** Umożliwia wstawienie zwłoki czasowej pomiędzy kolejnymi instrukcjami w trakcie realizowania skryptu.
- **PRINT** Komenda wywołująca wyświetlenie z góry zadeklarowanego tekstu na liście rozkazów.

Operacja na zmiennych

CLU_1->Strona++

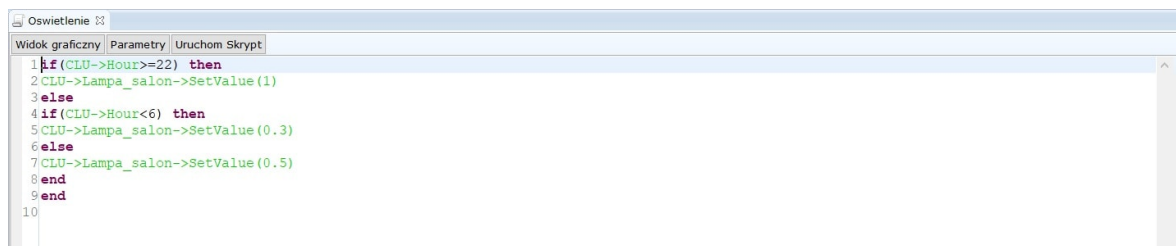
Blok ten umożliwia tworzenie skomplikowanych funkcji logicznych z wykorzystaniem zmiennych. By wykorzystać zmienne w skrypcie, muszą na początku zostać zadeklarowane. Zmienne można zadeklarować w parametrach skryptu oraz w cechach użytkownika CLU. Zmienna, zadeklarowana jako parametr skryptu, może być wykorzystywana wewnątrz skryptu - na potrzeby dokonywania obliczeń w czasie wykonywania tego skryptu. Dane przechowywane wewnątrz tej zmiennej nie są dostępne na zewnątrz. By dane zawarte w zmiennej mogły być przechowywane lub wykorzystywane poza skrypcem, należy skorzystać z cechy użytkownika CLU.

B. Tworzenie skryptów w trybie tekstowym

Drugą metodą tworzenia skryptów jest użycie edytora tekstowego, który daje praktycznie nieograniczone możliwości tworzenia skryptów z wykorzystaniem instrukcji LUA, rozszerzonych o możliwość używania adresów obiektów interfejsu logicznego.

Adresy interfejsu logicznego traktowane są jako funkcje i mogą być wywoływane i używane jako parametry w instrukcjach warunkowych, pętlach itp.

Poniższy skrypt pokazuje sposób użycia adresów interfejsu logicznego w skryptach:



```
Oświetlenie
Widok graficzny Parametry Uruchom Skrypt
1 if (CLU->Hour>=22) then
2 CLU->Lampa_salon->SetValue(1)
3 else
4 if (CLU->Hour<6) then
5 CLU->Lampa_salon->SetValue(0.3)
6 else
7 CLU->Lampa_salon->SetValue(0.5)
8 end
9 end
10
```

C. Przekazywanie parametrów do skryptu

Skrypty mogą posiadać parametry startowe, które są przekazywane podczas ich wywołania (np. w zdarzeniu), a następnie można z nich korzystać wewnątrz skryptu (np. w instrukcjach warunkowych). Parametry skryptu tworzone są poprzez kliknięcie w opcję **Parametry skryptu**. W następstwie czego otworzy się formatka, w której można zdefiniować parametry - podając nazwę, wartość domyślną, typ oraz ograniczenia.

Wartość domyślna to wartość parametru, która zostanie przyjęta, jeśli podczas wywołania skryptu parametr nie zostanie określony.

Typ pozwala zdefiniować rodzaj danych, które będą przechowywane w parametrze:

- **string** – dla danych tekstowych;
- **num** – dla danych liczbowych;
- **boolean** – dla zmiennych `true` / `false`.

Ograniczenia

Dla parametrów liczbowych można wprowadzić ograniczenie maksymalnej i minimalnej wartości, jaką może przyjąć parametr. W przypadku próby wywołania skryptu z parametrem spoza tego zakresu, skrypt zostanie wywołany w wartością domyślną parametru.

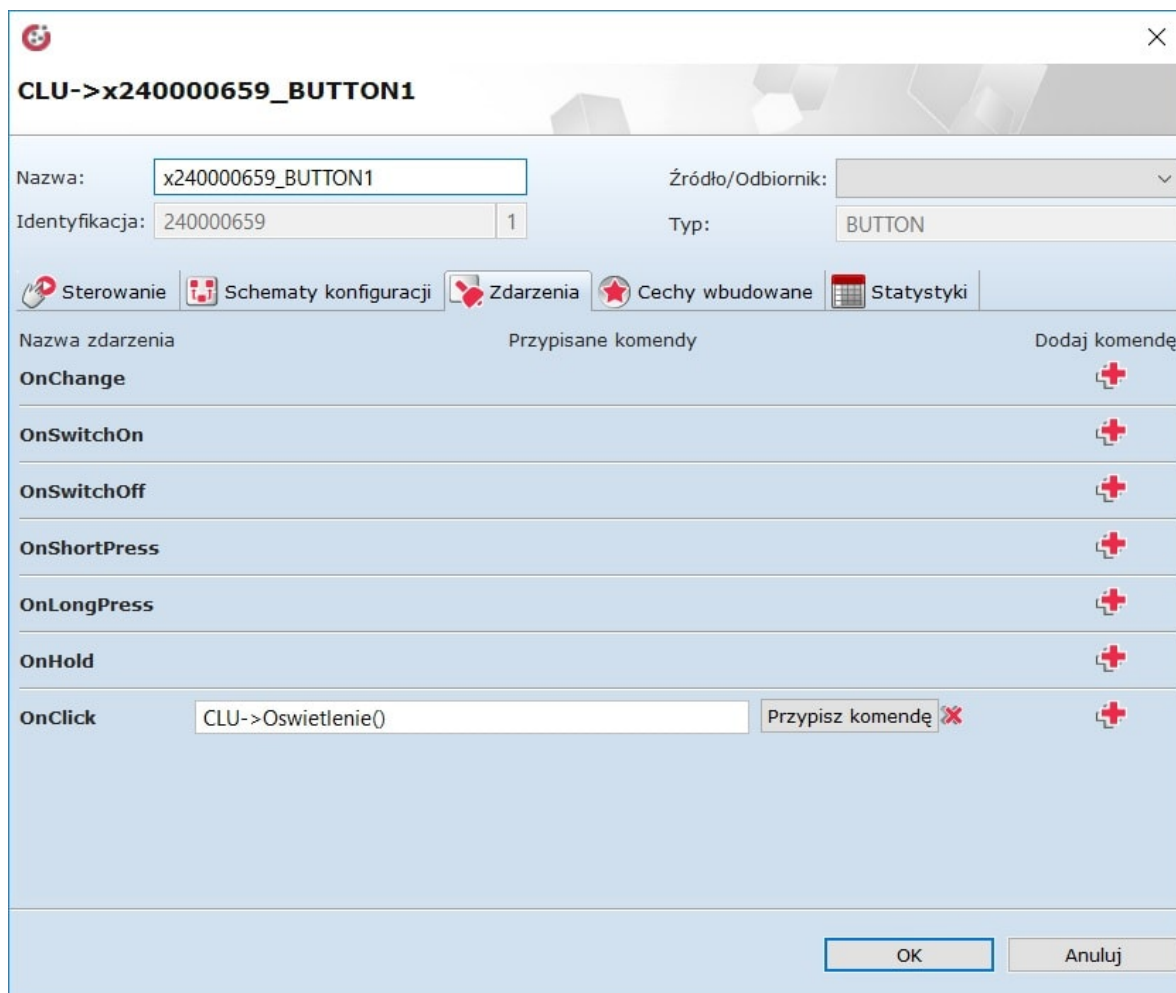
Zdefiniowane parametry dostępne są w skrypcie w formacie wyboru rozkazu lub w bloku operacje na zmiennych.

UWAGA! Parametr skryptu przechowuje wartości, które mogą być wykorzystywane tylko wewnątrz niego (lokalnie). Wartości te nie są dostępne w innych skryptach. Jeśli konieczne jest zapisanie wartości lub ciągu znaków do wykorzystania w innych obszarach, należy wykorzystać cechy użytkownika dostępne w CLU lub też przekazać wartość do drugiego skryptu przy pomocy jego parametru.

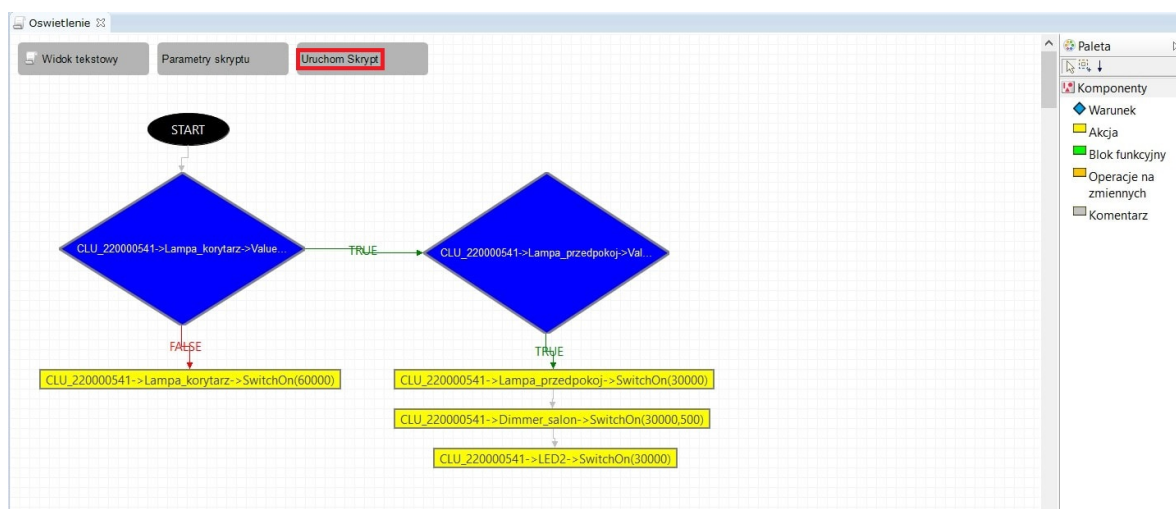
D. Wywoływanie skryptów

Skrypty są widoczne i traktowane jako metody CLU. Mogą być wywoływane ze zdarzeń z dowolnego obiektu oraz z bloku akcji w innym skrypcie (na identycznych zasadach jak inne metody).

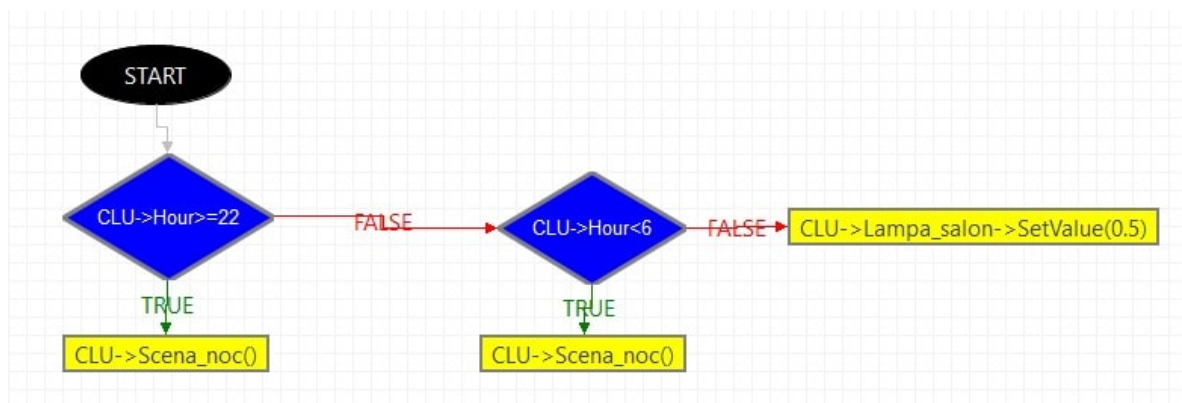
- **Wywołanie poprzez zdarzenie** Poniższy rysunek przedstawia przypisanie skryptu do włącznika, skrypt zostanie uruchomiony po jego wciśnięciu.



- **Wywołanie z poziomu skryptu** Poniższy rysunek ukazuje sposób wywołania z poziomu skryptu za pomocą przycisku *Uruchom skrypt*.



- **Wywołanie z innego skryptu** Poniższy rysunek pokazuje fragment diagramu, w którym wywołano inny skrypt przy pomocy bloku akcji.



3. Data i czas

CLU wyposażone jest w zegar czasu rzeczywistego (RTC) podtrzymywany wbudowaną baterią. CLU udostępnia kilkanaście cech, które mogą być wykorzystywane w skryptach. Pełna lista cech związanych z czasem jest następująca:

Nazwa	Opis
<code>Uptime</code>	Czas pracy urządzenia od ostatniego resetu (w sekundach)
<code>Log</code>	Wewnętrzny log urządzenia
<code>State</code>	Stan urządzenia (lista stanów)
<code>IsLocalPower</code>	<code>true</code> - jeżeli zasilany lokalnie, <code>false</code> - jeżeli zasilany z magistrali systemowej
<code>Date</code>	Zwraca aktualną datę
<code>Time</code>	Zwraca aktualny czas (hh:mm:dd)
<code>Day</code>	Zwraca numer bieżącego dnia miesiąca
<code>Month</code>	Zwraca numer bieżącego miesiąca
<code>Year</code>	Zwraca numer bieżącego roku
<code>DayOfWeek</code>	Zwraca numer bieżącego dnia tygodnia (0=niedziela)
<code>Hour</code>	Zwraca aktualną godzinę (bez minut i sekund)
<code>Minute</code>	Zwraca aktualną liczbę minut od ostatniej pełnej godziny
<code>UnixTime</code>	Zwraca aktualny uniksowy znacznik czasu
<code>LibraryVersion</code>	Informacja na temat bieżącego oprogramowania CLU

Warto zwrócić uwagę na cechę `unixTime`, która podaje liczbę sekund od 1970 r. w postaci jednej liczby. Może być ona przydatna do sprawdzenia ile czasu minęło od poprzedniego wykonania skryptu lub wywołania zdarzenia.

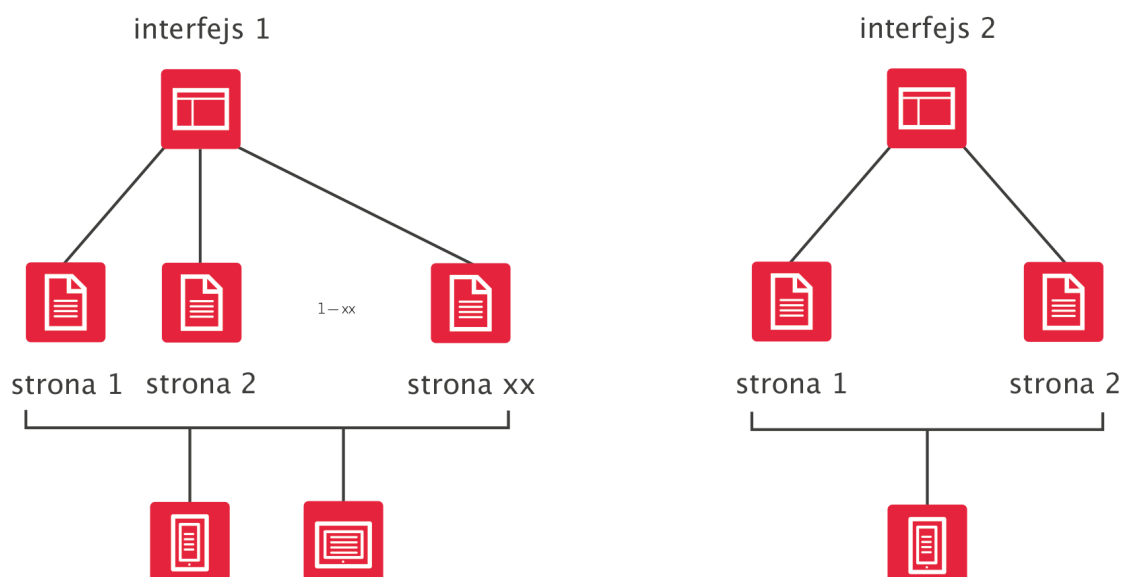
Czas ustawiany jest przy pomocy metody `setDateTime`. Aktualny czas dla CLU można wpisać „ręcznie” lub skorzystać z opcji automatycznego ustawiania czasu. Po zaznaczeniu pola `Auto-odświeżanie`, aktualna data i czas będzie pobierana z systemu operacyjnego.

VIII. Visual Builder

1. Sterowanie systemem z poziomu smartfonów

System umożliwia sterowanie przy pomocy dowolnych urządzeń pracujących w oparciu o system operacyjny Android oraz iOS (tablety, telefony komórkowe, media playery). Dla każdego systemu można przygotować wiele interfejsów użytkownika, z których każdy może zawierać sporą liczbę podstron. Umożliwia to stworzenie różnych interfejsów dla różnych użytkowników, zgodnie z ich potrzebami i upodobaniami, a także logiczne posortowanie funkcji sterujących w ramach każdego interfejsu (np. każde pomieszczenie na osobnej podstronie lub też podział wg funkcji – ogrzewanie, oświetlenie itp.).

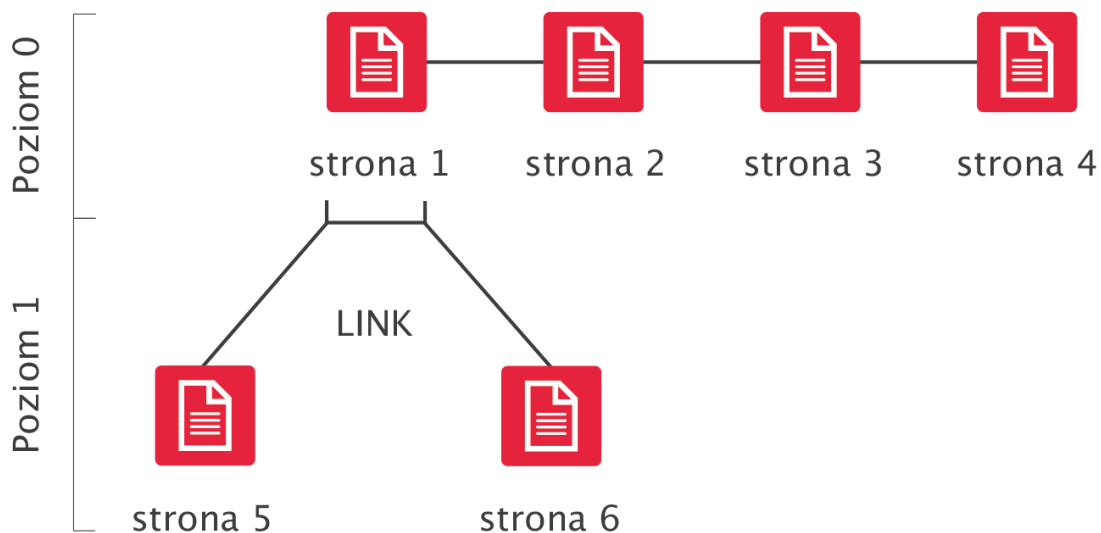
Interfejsy są tworzone przy pomocy narzędzia Visual Builder będącego częścią Object Managera, a następnie przesyłane do aplikacji zainstalowanej na urządzeniu Android lub iOS.



2. Struktura interfejsu

Każdy interfejs składa się z jednej lub większej liczby podstron, na których rozmieszczone są elementy sterujące (przyciski, suwaki). Projektant może w pełni kontrolować układ stron, rozmieszczenie elementów graficznych oraz wygląd interfejsu, który ustalany jest poprzez zmianę skórki graficznej.

Strony w interfejsie mogą występować na dwóch poziomach: poziom zero oraz poziom jeden. Strony znajdujące się na poziomie zero dostępne są jako podstawowe strony interfejsu, po których nawigacja odbywa się poprzez przewijanie stron lewo/prawo. Do stron poziomu jeden, użytkownik może dostać się poprzez komponent *Link*.



3. Aplikacja na smartfony – GRENTON HOME MANAGER

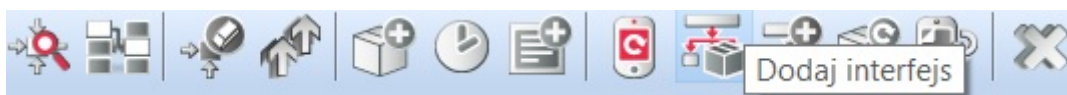
Aplikacja GRENTON HOME MANAGER pozwala na uruchamianie interfejsów użytkownika zaprojektowanych w Visual Builderze na urządzeniach Android oraz iOS. Do aplikacji przesyłany jest gotowy pakiet przygotowany w Visual Builderze, zawierający opis interfejsu, wszystkie pliki z nim związane i dane konfiguracyjne.

W zależności od stworzonego interfejsu, aplikacja umożliwi przegląd bieżącego stanu systemu oraz sterowanie wszystkimi funkcjami dostępnymi w systemie.

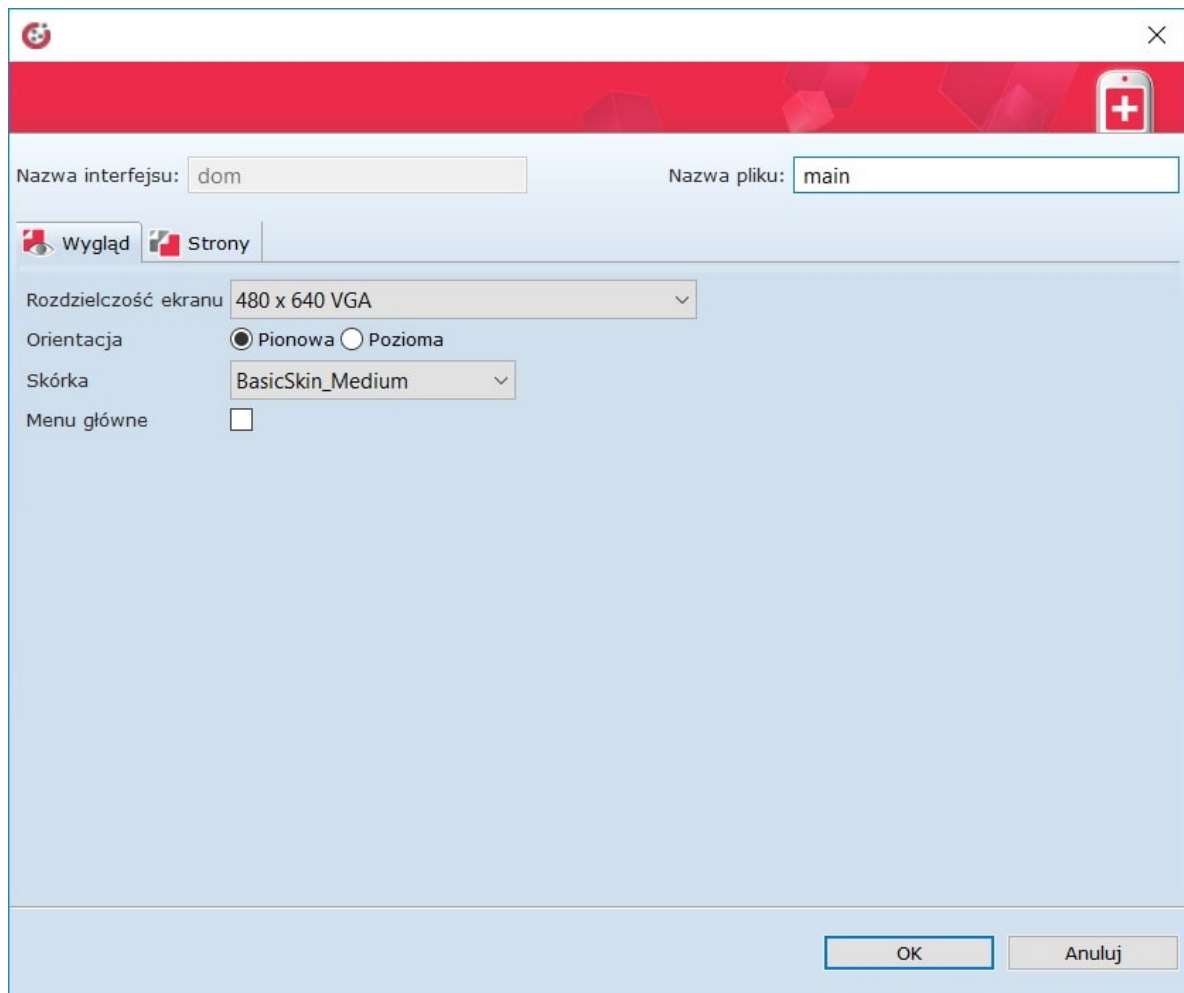
W celu sterowania systemem GRENTON ze smartfona należy zainstalować w nim ww. aplikację, a następnie przesłać do niej interfejs stworzony przy pomocy Visual Buildera. Aplikację można bezpłatnie pobrać ze sklepu GOOGLE PLAY na urządzenia Android oraz z APP STORE na urządzenia iOS. Do poprawnego działania aplikacji, urządzenie, na którym została zainstalowana, musi być podłączone do tej samej sieci LAN, co system GRENTON lub musi zostać skonfigurowane połączenie zdalne.

4. Tworzenie nowego interfejsu

Utworzenie nowego interfejsu odbywa się poprzez wybranie ikony **Dodaj interfejs** z menu akcji.



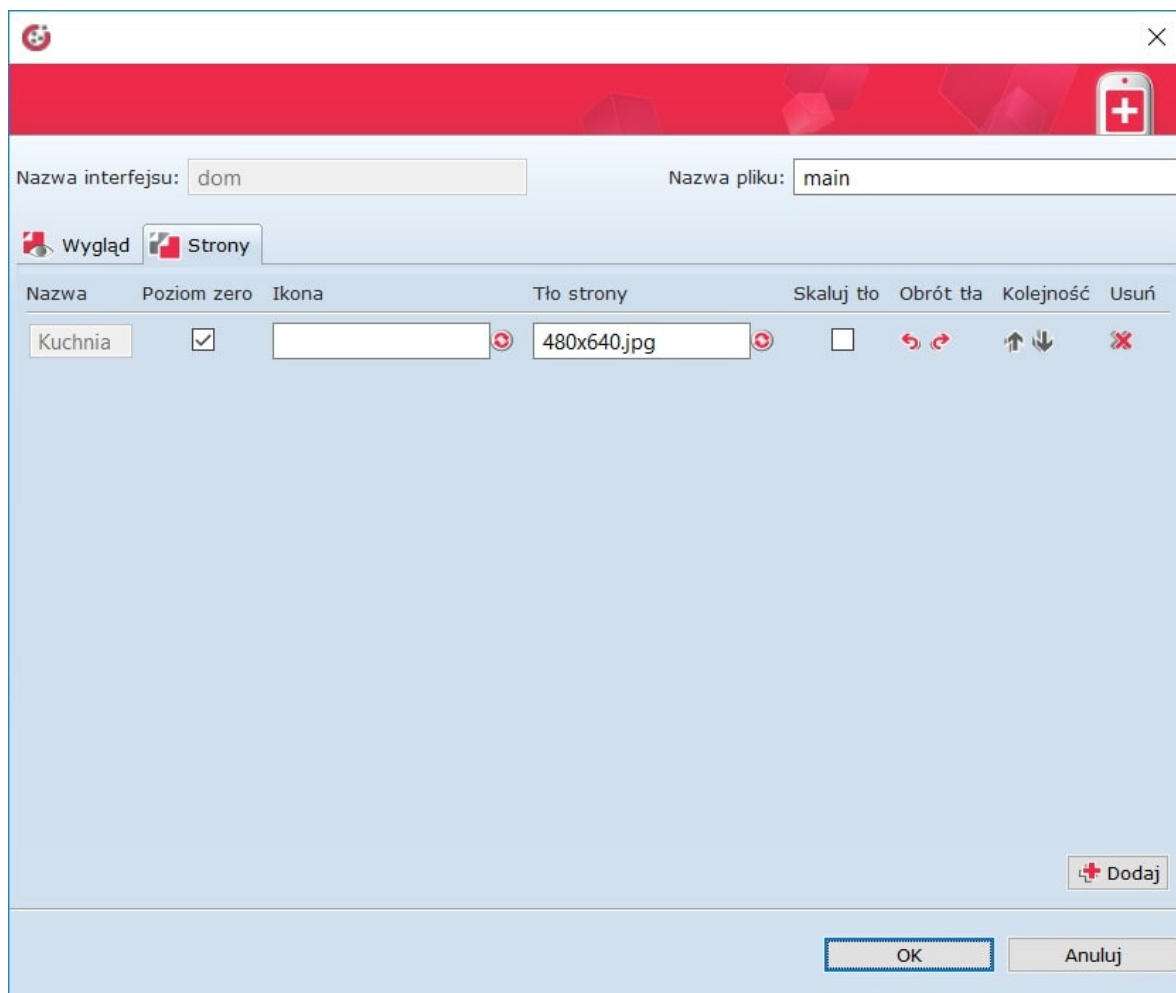
Po wpisaniu nazwy otworzy się okno utworzonego interfejsu, zawierające dwie zakładki: **wygląd** i **strony** (okno interfejsu jest również dostępne po dwukrotnym kliknięciu na ikonę utworzonego interfejsu w menu obiektów). Zakładka **wygląd**:



Zawiera informacje dotyczące sposobu wyświetlania interfejsu, takie jak: rozdzielczość, orientacja, lista dostępnych skórek oraz pole, po zaznaczeniu którego tworzone jest menu główne.

W prawym górnym rogu znajduje się pole `Nazwa pliku`. Nazwa ta, po wysłaniu interfejsu do urządzenia mobilnego, jest wyświetlana na jego liście interfejsów. W przypadku, gdy do jednego urządzenia wysyłany jest więcej niż jeden interfejs, należy pamiętać o przypisaniu każdemu z nich innej nazwy.

W zakładce `Strony` zawarta jest lista wszystkich utworzonych stron.



W tej zakładce można dokonać zmiany kolejności wyświetlania się stron oraz usunąć wcześniej utworzone. Po zaznaczeniu opcji **Poziom zero** strona będzie widoczna w menu głównym. Dodatkowo, w zakładce *Strony*, możliwa jest zmiana ikony, wyświetlanej u dołu strony w menu oraz jej tła.

Jeśli wybrane tło ma orientację inną niż używana w interfejsie, wolno dokonać jego obrotu za pomocą przycisków **Obrót tła**.

Dodatkowo istnieje możliwość skalowania tła. Wybór tej opcji powoduje dopasowanie dowolnej rozdzielczości tła do rozdzielczości tworzonego interfejsu.

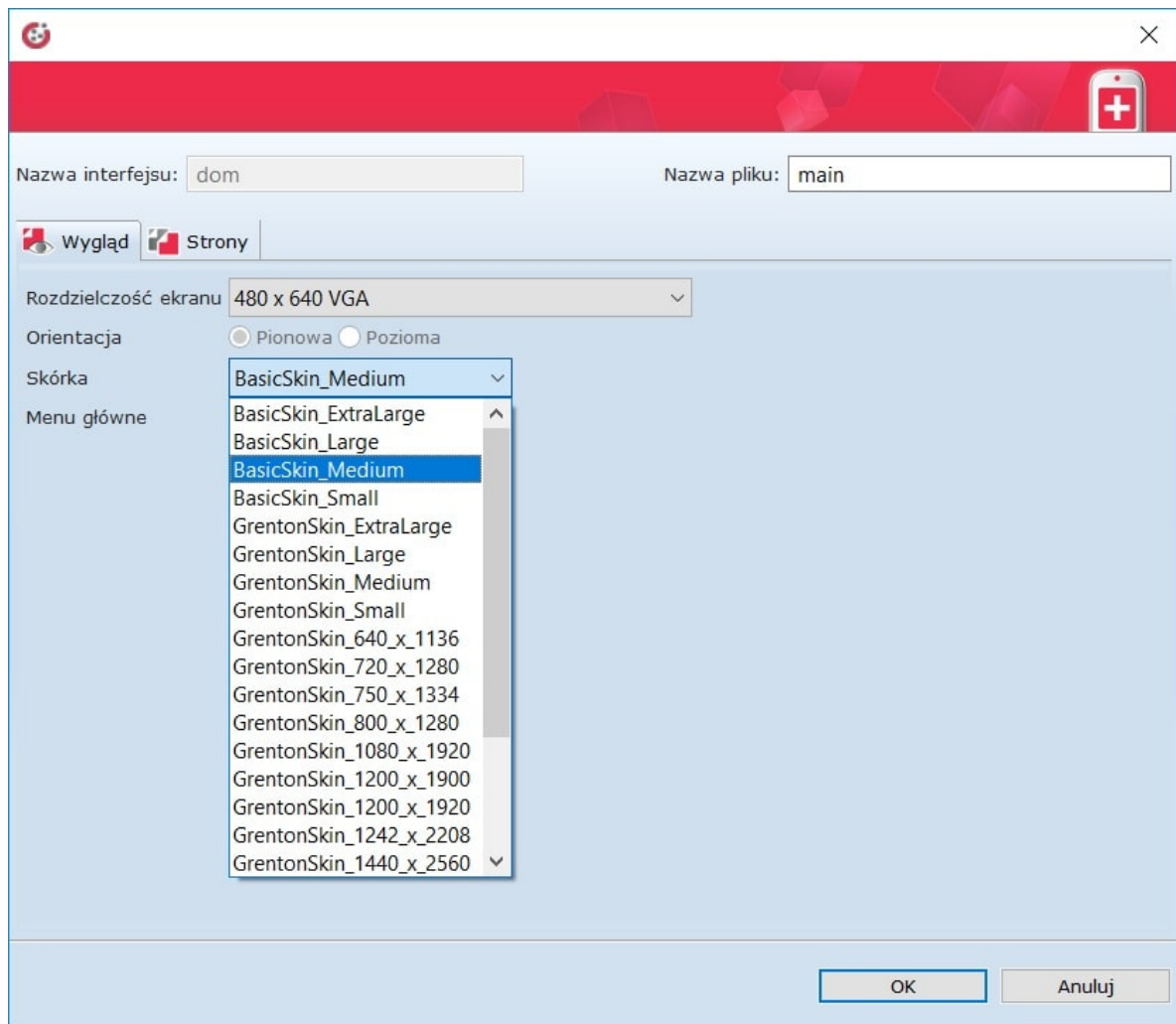
UWAGA! Nowo utworzony projekt w zakładce *Strony* nie posiada żadnych informacji. Informacje pojawiają się tam w momencie utworzenia stron interfejsu.

4.1. Wybór skóry graficznej

Skórki to zestaw ustawień graficznych dla interfejsu na urządzenie mobilne.

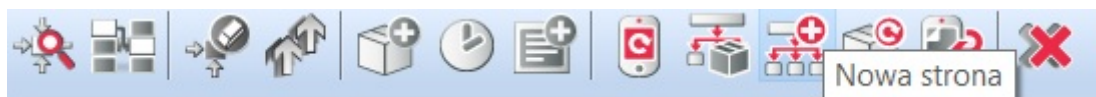
Skórki GRENTON

Użytkownik, w tworzonych interfejsach, może wykorzystać dostarczone wraz z OM skóry. Lista dostępnych skórek znajduje się w parametrach interfejsu mobilnego.



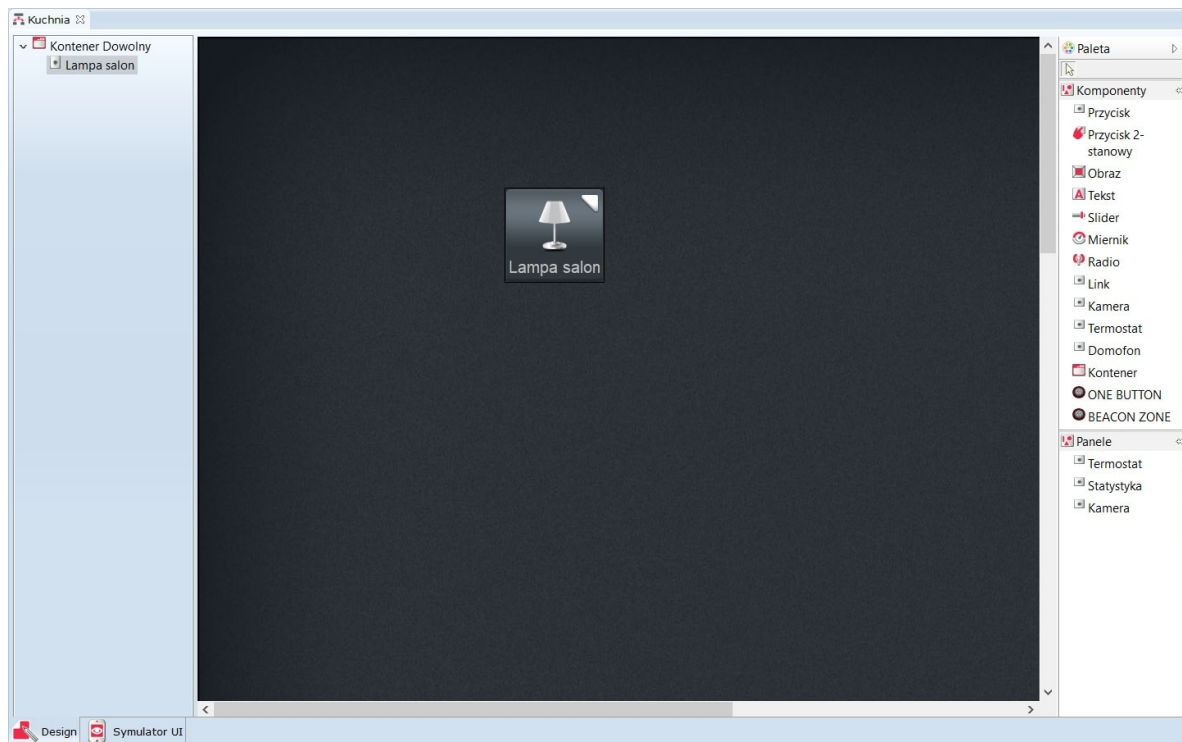
4.2. Tworzenie stron interfejsu

Po utworzeniu interfejsu, należy dodać do niego nowe strony. Tworzenie strony odbywa się z menu akcji:



Po utworzeniu nowej strony oraz nadaniu jej nazwy zostaje otwarty arkusz edycji. Znajdują się w nim dwie zakładki: *Design* oraz *Symulator UI* (zakładki znajdują się u dołu strony).






W zakładce *Design* znajdują się: lista obiektów, kontener główny, a także lista komponentów i paneli.



Lista obiektów wyświetla wszystkie obiekty wykorzystane w bieżącym arkuszu roboczym.

4.3. Komponenty

Komponenty – lista obiektów, które mogą być wykorzystane podczas tworzenia interfejsu. Lista komponentów obejmuje:

- **Przycisk** – działa jak przycisk monostabilny.
 Przycisk
- **Przycisk** - działa jak przycisk bistabilny.
 Przycisk 2-standowy
- **Obraz** – umożliwia wstawienie obrazka z pliku zewnętrznego.
 Obraz
- **Tekst** – umożliwia wstawienie pola tekstowego.
 Tekst
- **Slider** – suwak umożliwiający wykonanie płynnej regulacji.
 Slider
- **Miernik** – odwzorowuje w sposób analogowy wartość obiektu.

- **Radio** – odwzorowuje w sposób cyfrowy (on/off) stan danego obiektu.



- **Link** – umożliwia tworzenie linków do innych stron w ramach tego samego interfejsu.



- **Kontener** – porządkuje układ komponentów na obszarze roboczym w określony sposób.



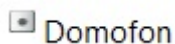
- **Kamera** – umożliwia wyświetlanie obrazu z kamery IP w aplikacji Home Manager.



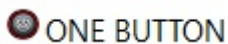
- **Termostat** – umożliwia wyświetlenie obiektu wirtualnego Termostat w aplikacji Home Manager.



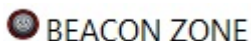
- **Domofon** – umożliwia skonfigurowanie domofonu (skonfigurowanie połączenia z serwerem SIP, przypisanie metod do konkretnych zdarzeń oraz wyświetlanie obrazu z kamery IP podczas połączenia).



- **ONE BUTTON** – umożliwia przypisanie BEACONOWI metody do zdarzenia w trybie ONE BUTTON.




- **BEACON ZONE** – umożliwia skonfigurowanie BEACONA w trybie BEACON ZONE oraz przypisanie konkretnych metod do zdarzeń (po dodaniu do strony widoczny u jej dołu pasek BEACON ZONE).



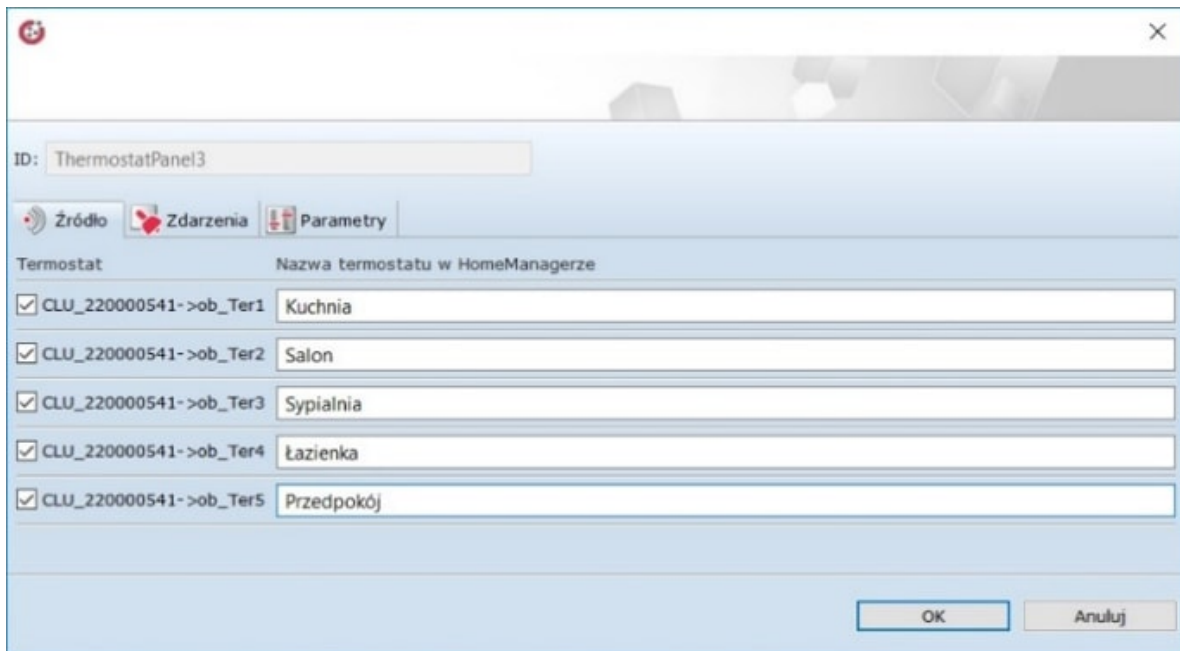
Wybrane obiekty wstawiane są do kontenera z listy komponentów poprzez przeciągnięcie, a ich rozmieszczenie jest uzależnione od typu zastosowanego kontenera głównego.

4.4. Panele

Panele – lista obiektów, które mogą zostać wykorzystane podczas tworzenia interfejsu na urządzenie mobilne. Panele w odróżnieniu do komponentów zajmują całą stronę interfejsu mobilnego. Lista paneli obejmuje:

- **Termostat** – tworzy panel dla termostatu na całej stronie interfejsu w HM.  **Termostat**

Jako źródło panelu termostatu ustawia się wcześniej utworzony obiekt wirtualny **Termostat**.



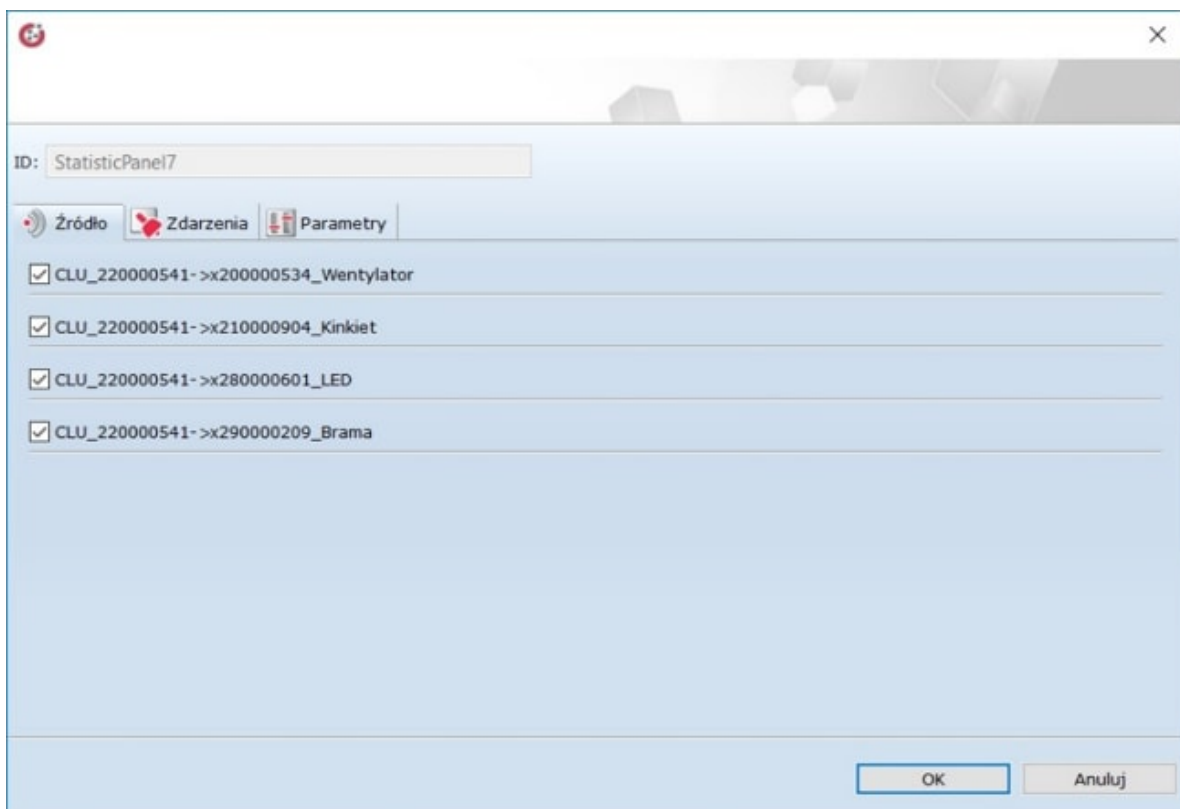
The screenshot shows a configuration window for a thermostat panel. At the top, there is a close button (X) and a header area with a red icon. Below the header, the ID field is set to "TermostatPanel3". There are three tabs: "Źródło" (selected), "Zdarzenia", and "Parametry". The main area is titled "Termostat" and "Nazwa termostatu w HomeManagerze". It contains a list of five entries, each with a checked checkbox and a text field:

Termostat	Nazwa termostatu w HomeManagerze
<input checked="" type="checkbox"/> CLU_220000541->ob_Ter1	Kuchnia
<input checked="" type="checkbox"/> CLU_220000541->ob_Ter2	Salon
<input checked="" type="checkbox"/> CLU_220000541->ob_Ter3	Sypialnia
<input checked="" type="checkbox"/> CLU_220000541->ob_Ter4	Łazienka
<input checked="" type="checkbox"/> CLU_220000541->ob_Ter5	Przedpokój

At the bottom right, there are "OK" and "Anuluj" buttons.

- **Statystyka** – tworzy panel do pomiaru mediów na stronie interfejsu w HM.  **Statystyka**


Po przeciągnięciu panelu na stronę interfejsu należy wybrać obiekty, dla których pomiar mediów będzie prezentowany w HM. W oknie wyświetlone będą jedynie obiekty, dla których wcześniej załączono *Pomiar Mediów*.



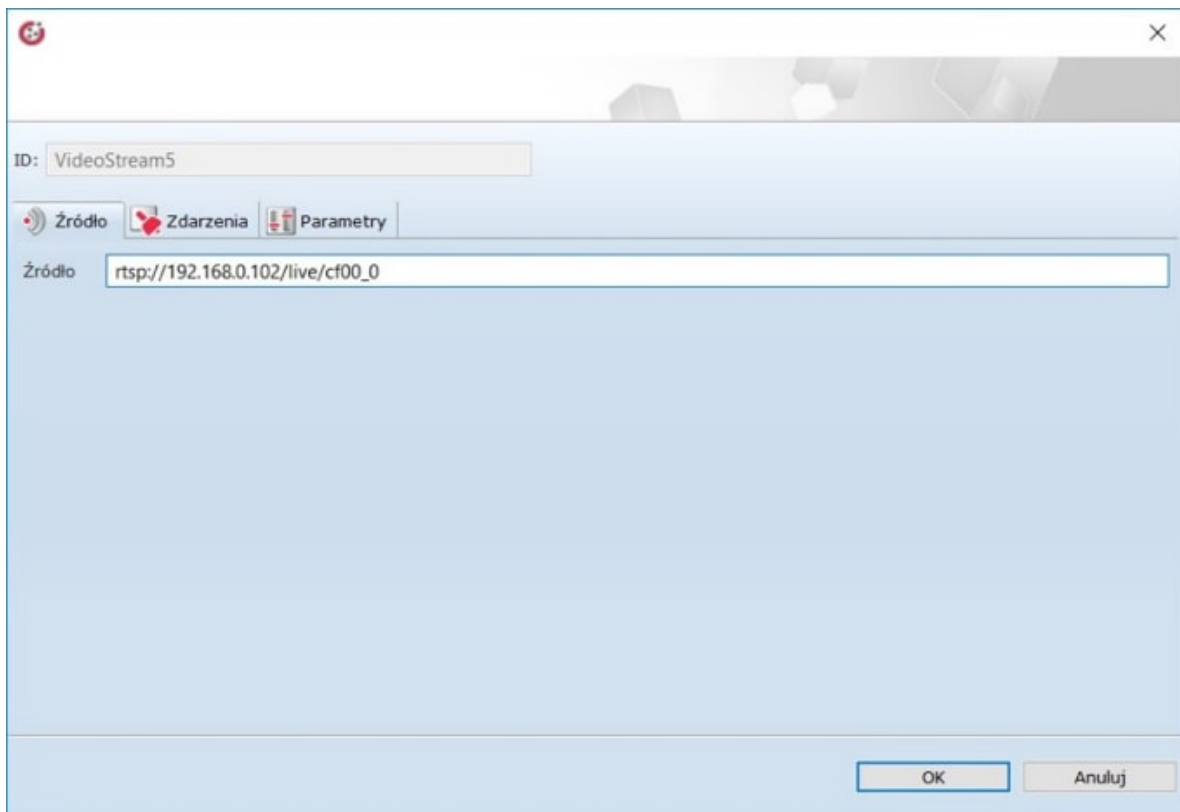
The screenshot shows a configuration window for a statistics panel. At the top, there is a close button (X) and a header area with a red icon. Below the header, the ID field is set to "StatisticPanel7". There are three tabs: "Źródło" (selected), "Zdarzenia", and "Parametry". The main area contains a list of four entries, each with a checked checkbox and a text field:

<input checked="" type="checkbox"/> CLU_220000541->x200000534_Wentylator
<input checked="" type="checkbox"/> CLU_220000541->x210000904_Kinkiet
<input checked="" type="checkbox"/> CLU_220000541->x280000601_LED
<input checked="" type="checkbox"/> CLU_220000541->x290000209_Brama

At the bottom right, there are "OK" and "Anuluj" buttons.

- **Kamera** – tworzy panel do wyświetlania obrazu z kamery IP na zdefiniowanej przestrzeni strony interfejsu w HM.  Kamera

Jako źródło panelu kamery należy podać strumień RTSP danej kamery IP.

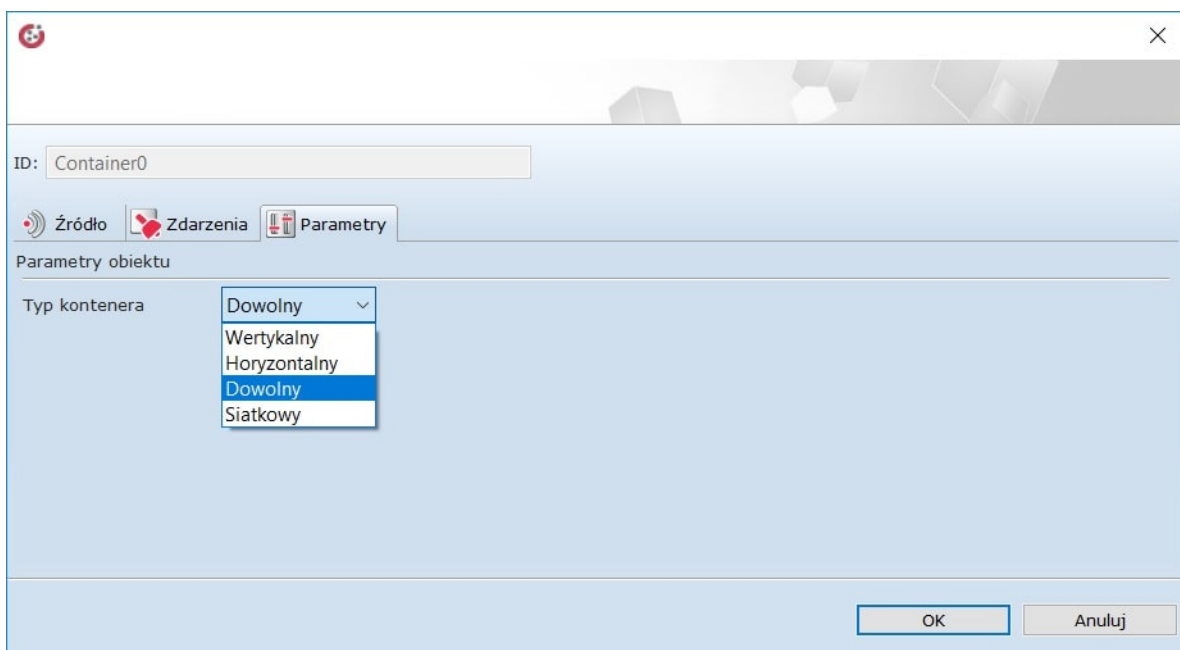


4.5. Kontenery

Kontener to zasobnik obiektów, określający ich sposób rozmieszczenia w obszarze roboczym.

Obiekty wewnątrz obszaru roboczego rozmieszczane są zgodnie z typem wybranego kontenera.

Typ kontenera można zmienić w parametrach obiektu dla tego kontenera. Okno z parametrami otwiera się po dwukrotnym kliknięciu na obiekt kontenera, znajdujący się na pierwszym miejscu listy obiektów.

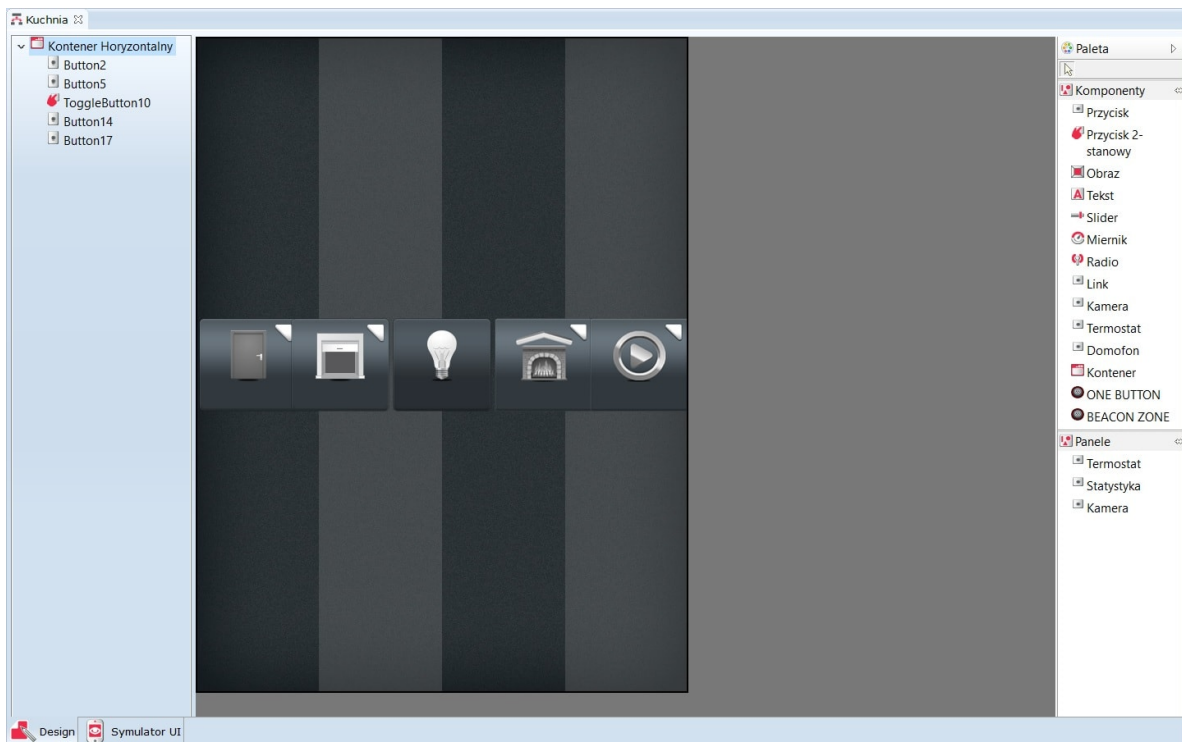


Istnieją cztery typy dostępnych kontenerów:

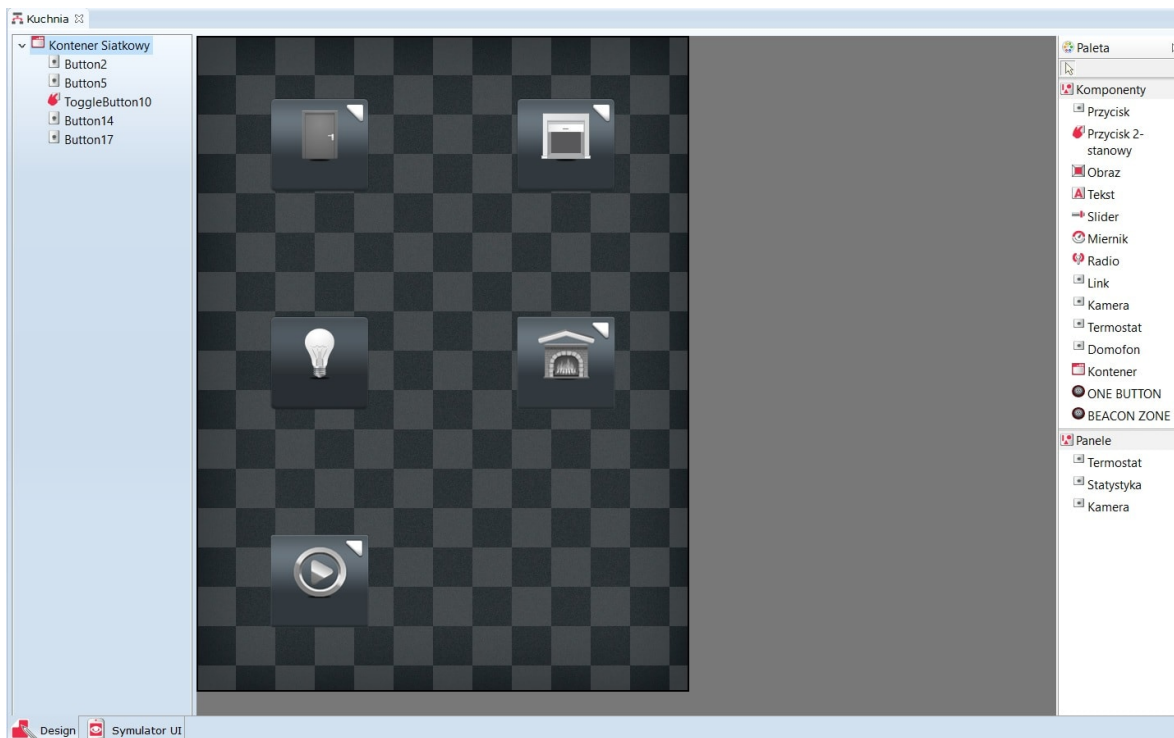
1. **Wertykalny** – rozmieszcza wstawiane elementy w sposób pionowy w równych, automatycznie się tworzących sekcjach.



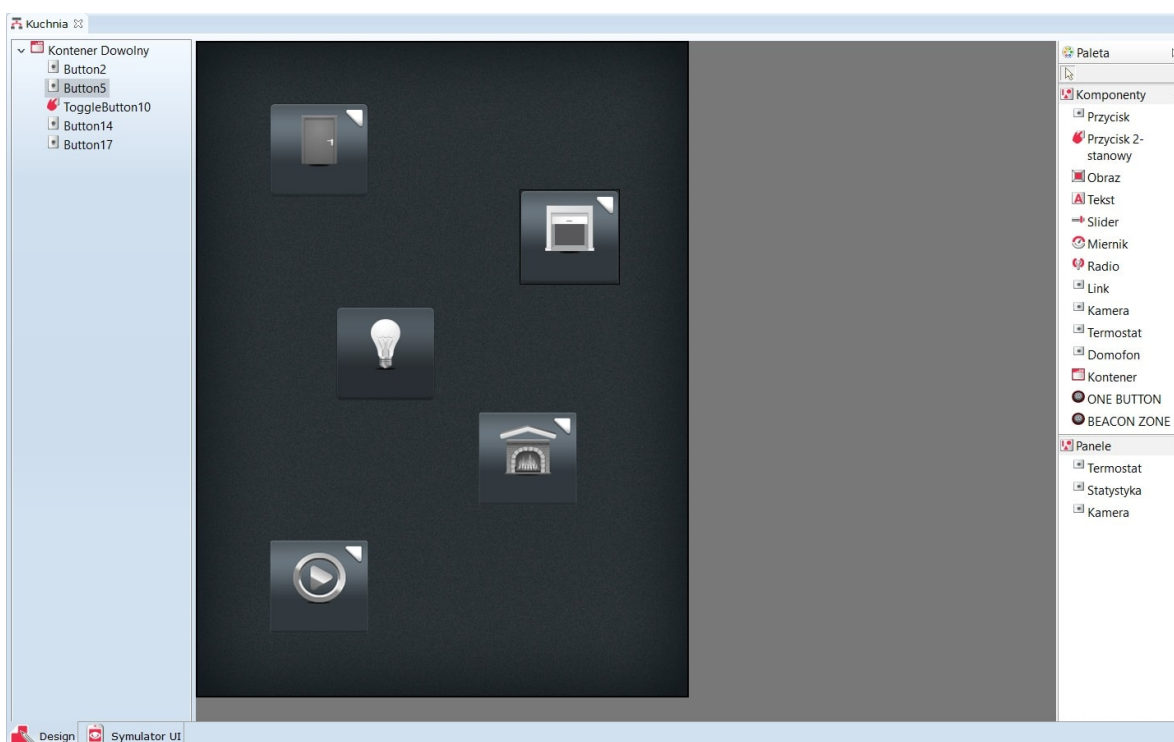
2. **Horyzontalny** – wstawiane elementy rozmieszczane są w poziomych sekcjach.



3. **Siatkowy** – wstawiane komponenty układane są wg symetrycznej siatki.



4. **Dowolny** – umożliwia dowolny rozkład wstawionych komponentów na całym obszarze kontenera.

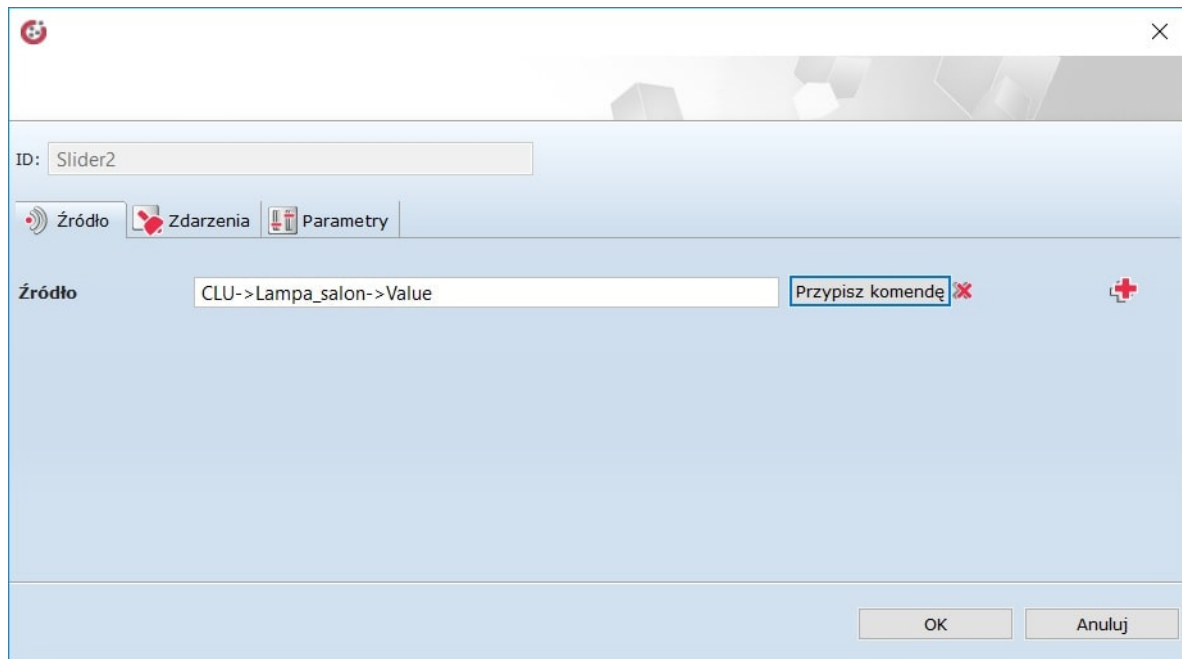


4.6. Wstawianie komponentów i tworzenie powiązań z obiektami systemu

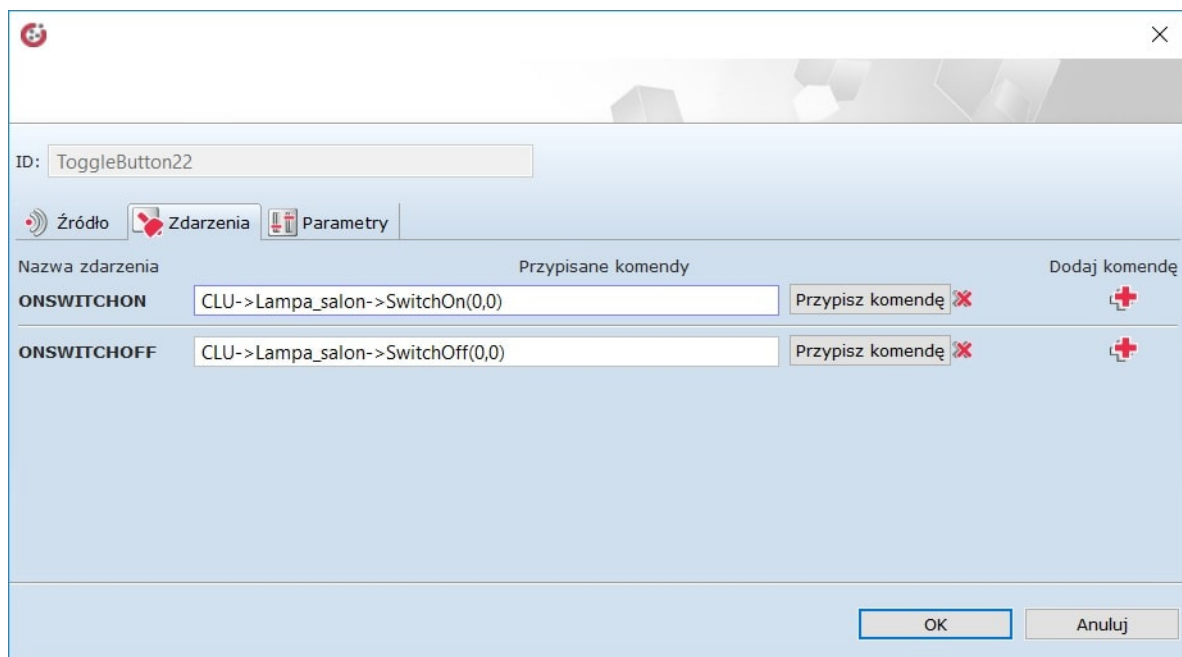
Po wybraniu komponentu z listy znajdującej się po prawej stronie i wstawieniu go do kontenera głównego, automatycznie otwiera się okno właściwości dla tego obiektu. W oknie tym znajdują się trzy zakładki (**Źródło** , **Zdarzenia** oraz **Parametry**), w których należy ustawić odpowiednio:

1. W zakładce **Źródło** należy wskazać obiekt, którego wartość ma być odwzorowana oraz czas odświeżania dla tej wartości, np. jeżeli wstawimy do interfejsu slider, który ma sterować

dimmerem, to w celu wyświetlania na smartfonie aktualnej wartości załączonego oświetlenia, jako źródło musi zostać wskazany sterowany dimmer.



2. **Zdarzenia** wykorzystywane są dla obiektów, za pomocą których odbywa się sterowanie, np. przycisk lub slider. W tej zakładce znajdują się zdarzenia odpowiednie dla danego typu obiektów, do których należy przypisać metody sterowanych obiektów.



3. W zakładce **Parametry** znajdują się dane związane z wyświetlaniem danego obiektu w interfejsie. Użytkownik ma możliwość: zmiany skórki obiektu, ustawienia położenia komponentu na stronie, a także edycji wyświetlanego podpisu obiektu.

ID: ToggleButton22

Źródło Zdarzenia Parametry

Skórka Lampa_Toggle

Parametry obiektu

X	315
Y	65
Text	Lampa wł
Text in not selected st	Lampa wył

OK Anuluj

UWAGA! Jeśli w polu `Text` wprowadzona zostanie komenda `$value$`, to będzie w nim wyświetlana aktualna wartość cechy `value` obiektu wybranego w zakładce `Źródło`.

Po ukończeniu komponowania interfejsu lub w trakcie jego tworzenia, użytkownik ma możliwość sprawdzania działania oraz wyglądu przygotowanego interfejsu. W tym celu należy uruchomić Symulator UI (druga zakładka na dole strony).

4.7. Wysyłanie interfejsu do urządzenia mobilnego

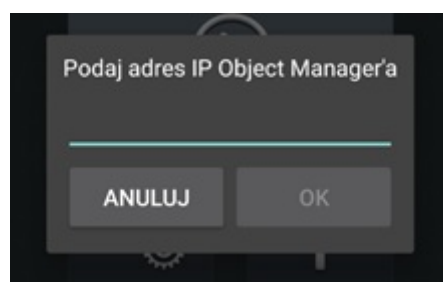
Chcąc sterować systemem za pomocą urządzenia mobilnego, utworzony interfejs musi zostać wysłany do aplikacji GRENTON HOME MANAGER zainstalowanej na wybranym urządzeniu.

W tym celu należy:

- Wybrać z listy utworzonych interfejsów Visual Buildera ten, który ma zostać wysłany - ikona narzędzia do wysyłania interfejsu znajduje się w menu głównym:



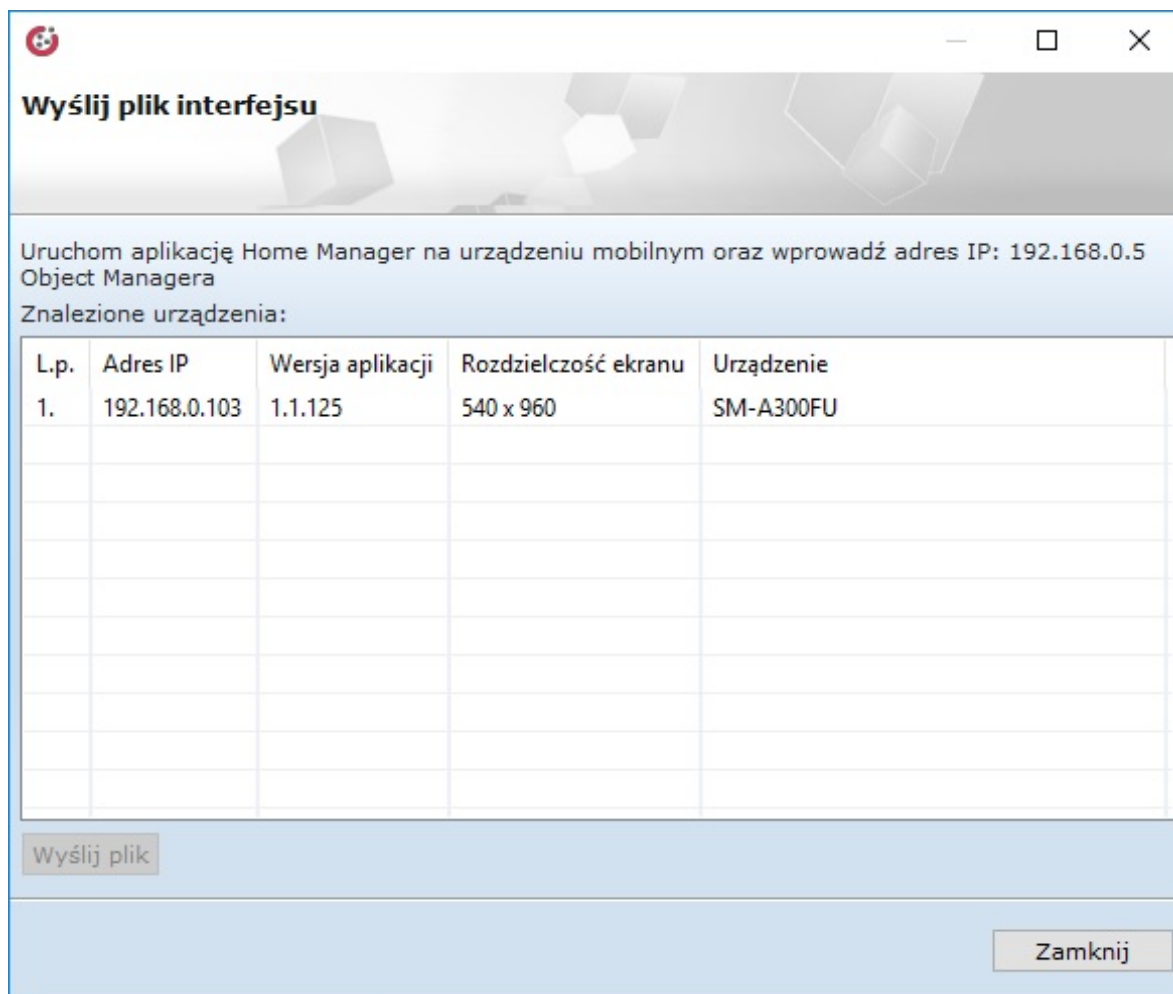
- W urządzeniu mobilnym połączyć się z siecią, w której znajduje się CLU (po wyświetleniu w Object Managerze okna wysyłania);
- W otwartej aplikacji Home Manager wybrać z menu głównego *Połącz z OM*;
- Podać adres IP Object Managera i wybrać *OK*:



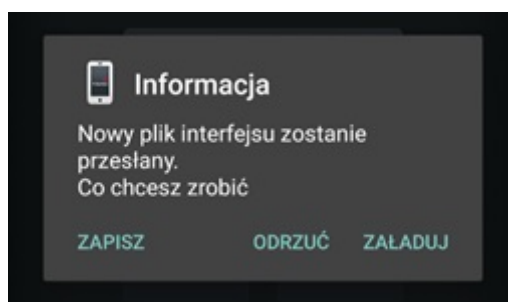
- Urządzenie mobilne pojawi się w oknie wysyłania, które zostało wyświetlone w programie Object Manager;

UWAGA! Na liście wyświetlane są urządzenia, na których uruchomiona jest aplikacja GRENTON HOME MANAGER i została w niej włączona opcja **Połącz z OM** w ustawieniach aplikacji.

- Kliknąć myszką dwukrotnie na jego nazwę lub zaznaczyć i wybrać *Wyślij plik*:



- W aplikacji mobilnej pojawi się okno przyjmowania interfejsu. Wybrać *Zapisz*:

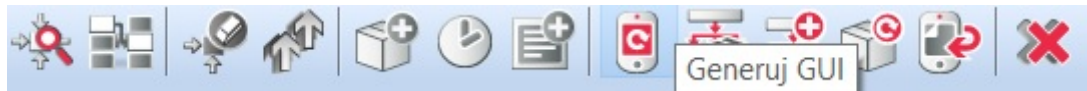


- Na ekranie pojawi się pasek stanu przesyłania. Po zakończeniu, na górnej belce programu, wyświetli się informacja o poprawnym zakończeniu przesyłania.
- Po wysłaniu pliku z interfejsem do urządzenia mobilnego, by zdalne sterowanie było możliwe, należy załadować przesłany interfejs.

5. Automatyczne tworzenie interfejsu - generator GUI

Funkcja ta umożliwia szybkie stworzenie interfejsu poprzez wybór obiektów, którymi chcemy sterować spośród wszystkich obiektów dostępnych w systemie.

Automatyczne tworzenie interfejsu użytkownika rozpoczynamy od uruchomienia Generатора GUI. Ikona generatora znajduje się w menu obiektów:

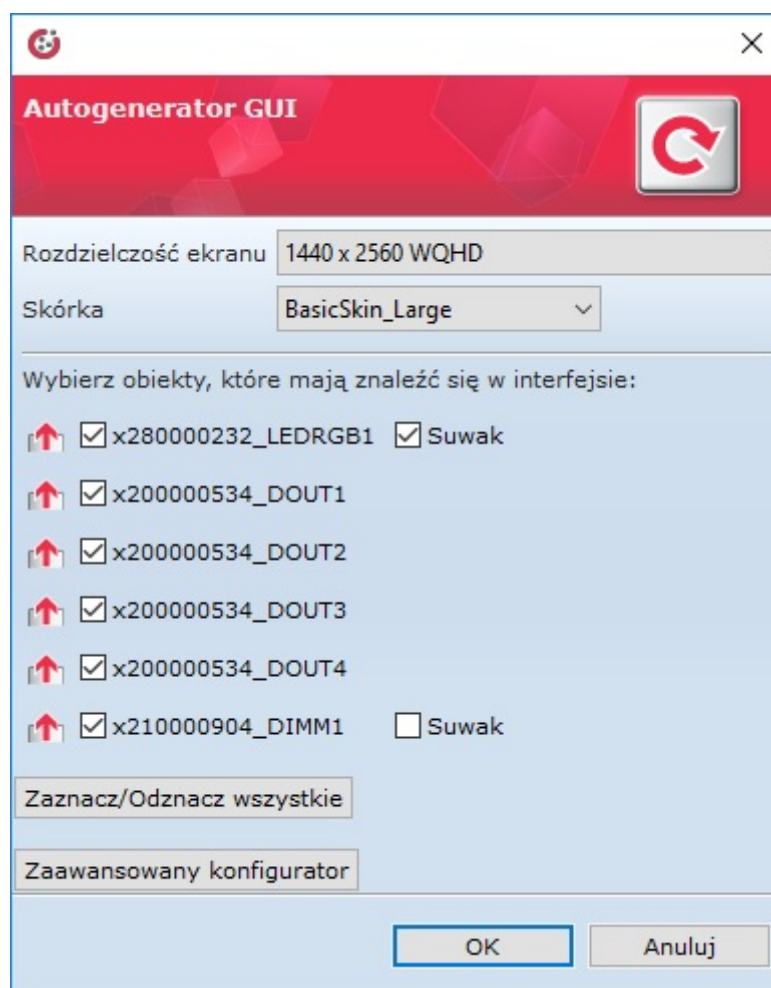


5.1. Tworzenie interfejsu z dostępną rozdzielczością

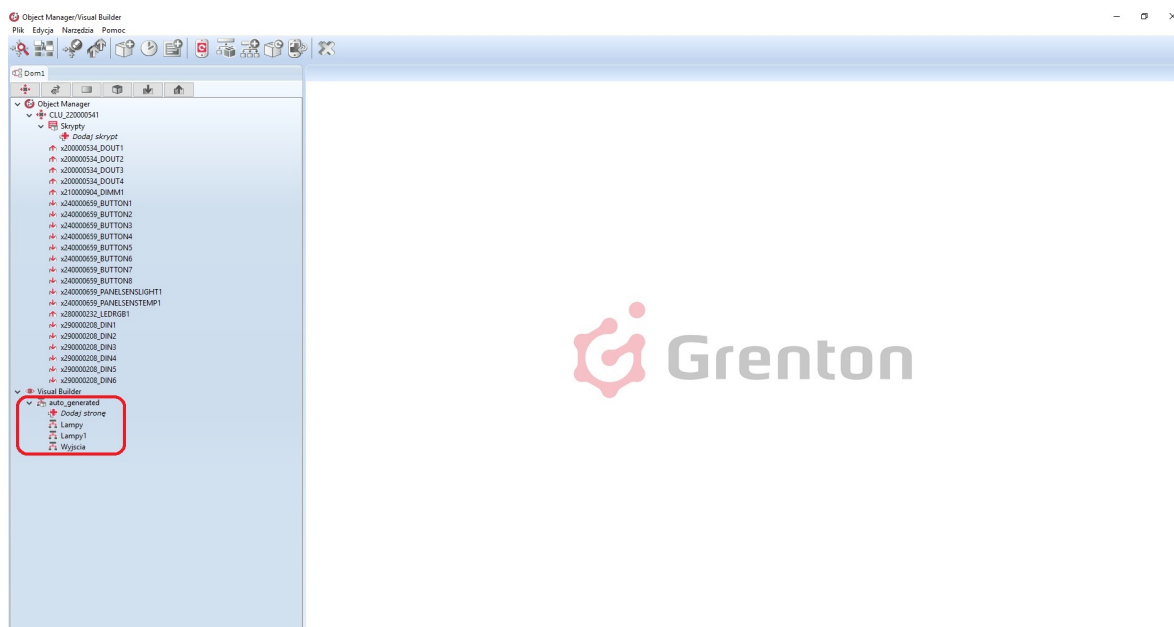
A. Konfigurator prosty

Po kliknięciu na wskazaną ikonę otwiera się okno **Autogeneratora GUI**. Jest to prosty konfigurator, w którym należy wybrać:

- rozdzielczość, z jaką pracuje urządzenie mobilne;
- skórę określającą wygląd ikon w interfejsie;
- obiekty (z listy obiektów), które mają zostać zawarte w utworzonym interfejsie.

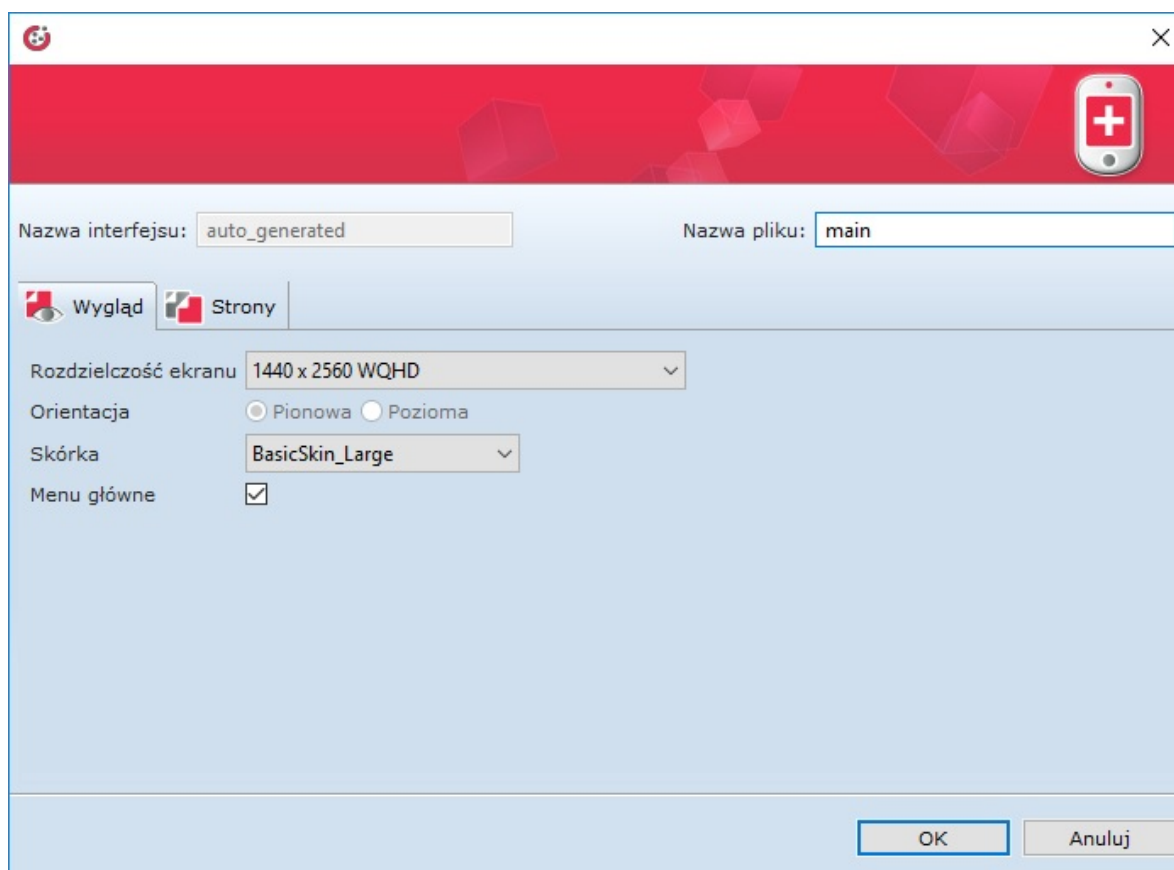


Po zaznaczeniu interesujących obiektów należy kliknąć **OK**. W rezultacie na liście obiektów (pod ikoną utworzonego interfejsu) pojawiają się nowo utworzone strony zgodnie z poniższym rysunkiem:



W każdej chwili istnieje możliwość zmiany ustawień interfejsu - wystarczy dwukrotnie kliknąć na jego nazwę, a otworzy się okno z dwoma zakładkami: **wygląd** i **Strony**.

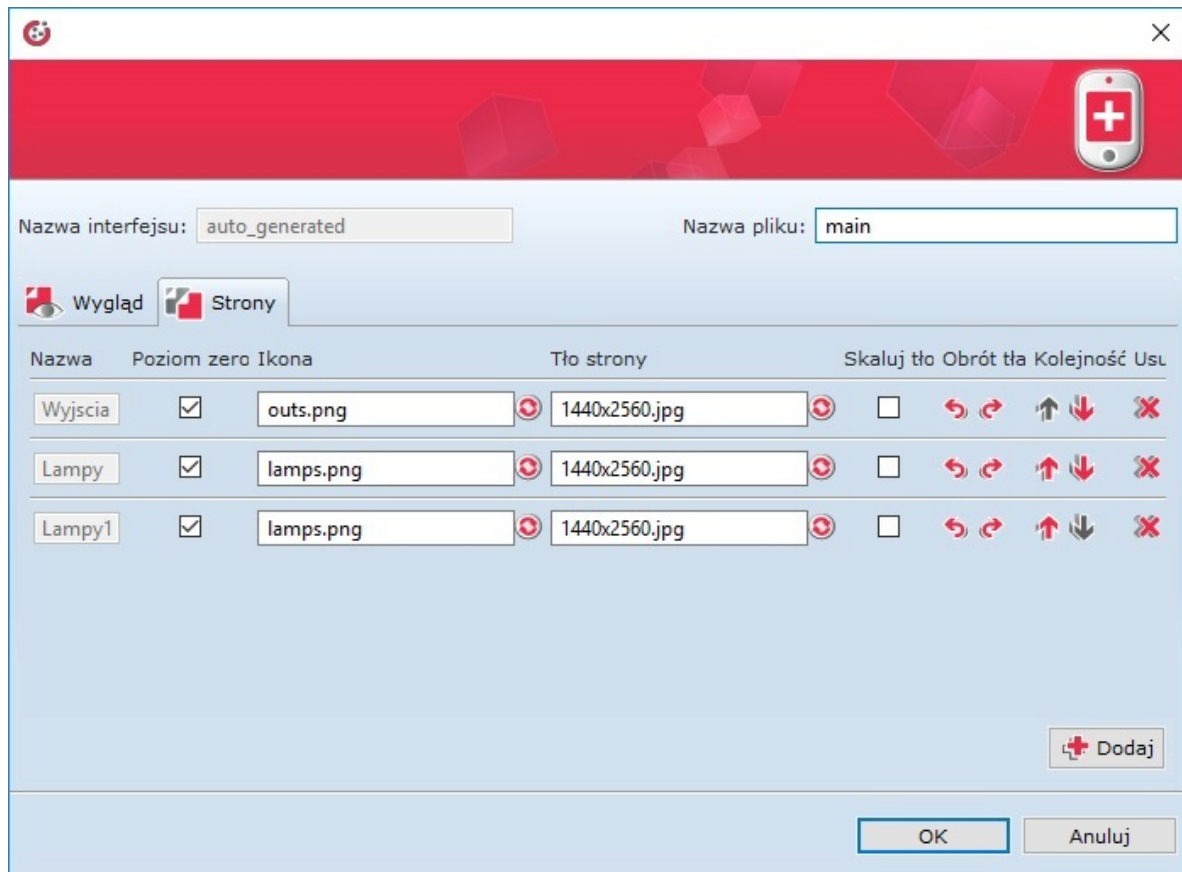
W zakładce **wygląd** użytkownik ma możliwość wybrania skórek widocznych w interfejsie. W tym widoku znajduje się również pole *Menu główne*. Po jego zaznaczeniu zostanie utworzone menu, zawierające wszystkie dostępne i zaznaczone strony.



Zakładka **strony** zawiera listę utworzonych stron oraz umożliwia zmianę ich parametrów, takich jak:

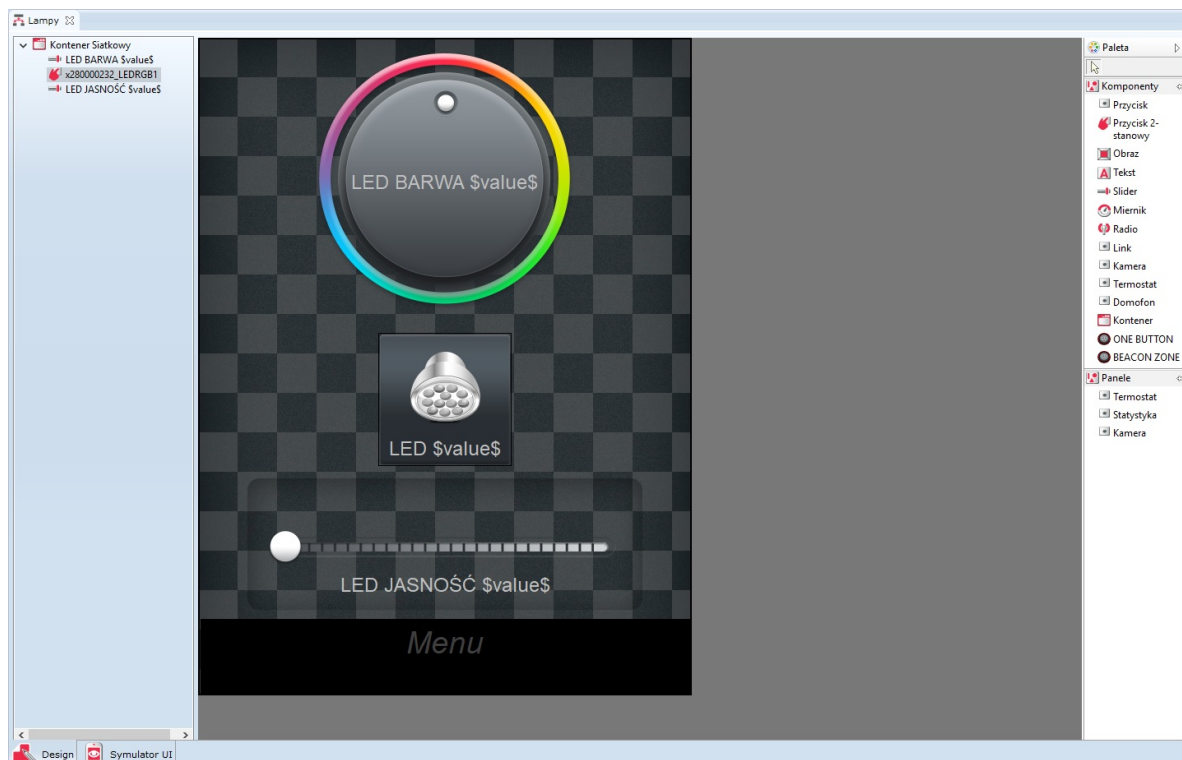
- **poziom zero** – określenie, czy strona ma być wyświetlana w menu;
- **ikona** – ikona wyświetlana w menu (domyślnie jest to ikona zgodna z wybraną skórką);
- **tło strony** – tło wyświetlanej strony (domyślnie wyświetlane jest tło ze skórki, jednakże użytkownik może zdefiniować własne);
- **skaluj tło** - dopasowanie wybranej rozdzielczości do rozdzielczości urządzenia mobilnego;

- **obrót tła** - zmiana orientacji tła;
- **kolejność** – ustawienie kolejności wyświetlania stron w menu;
- **usuń** – oznacza całkowite usunięcie strony z interfejsu.

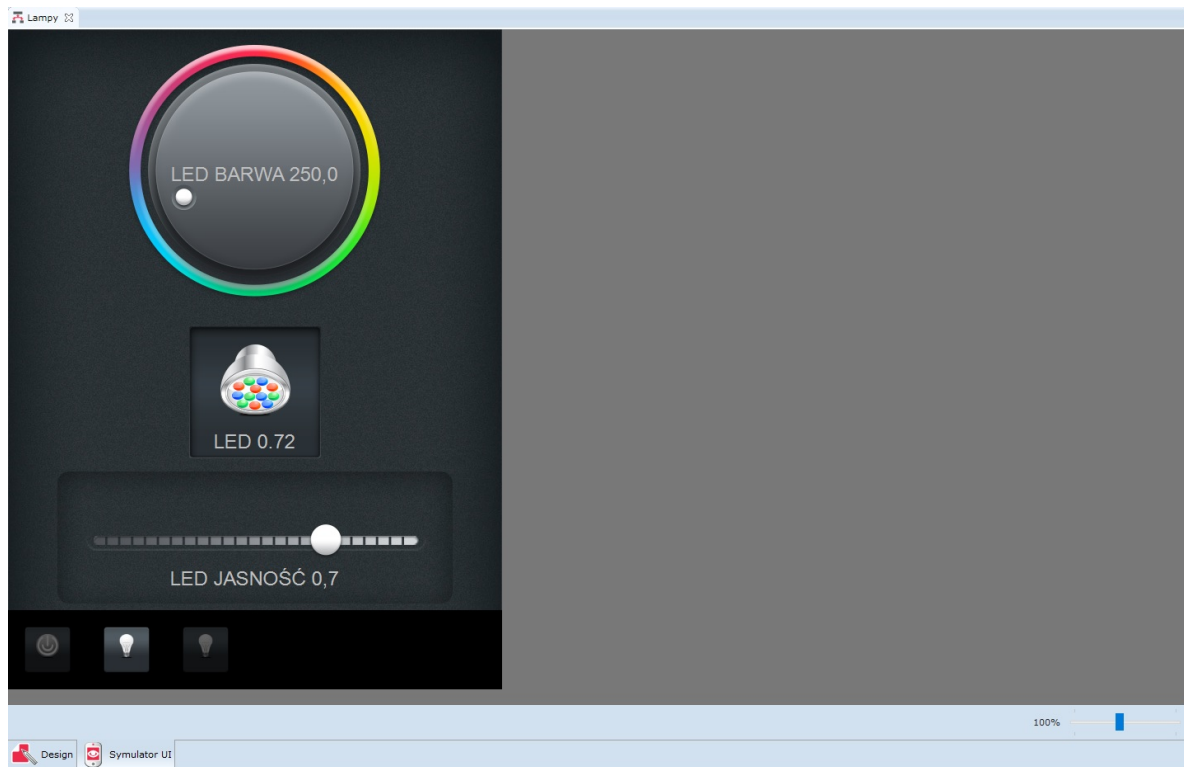


Użytkownik ma również możliwość dokonywania zmian w wygenerowanych stronach. Dwukrotne kliknięcie na ikonie strony otworzy arkusz edycji, zawierający dwie zakładki **Design** oraz **symulator**.

Zakładka **Design** – wyświetla obszar roboczy zawarty w kontenerze oraz umożliwia edycję utworzonej strony.



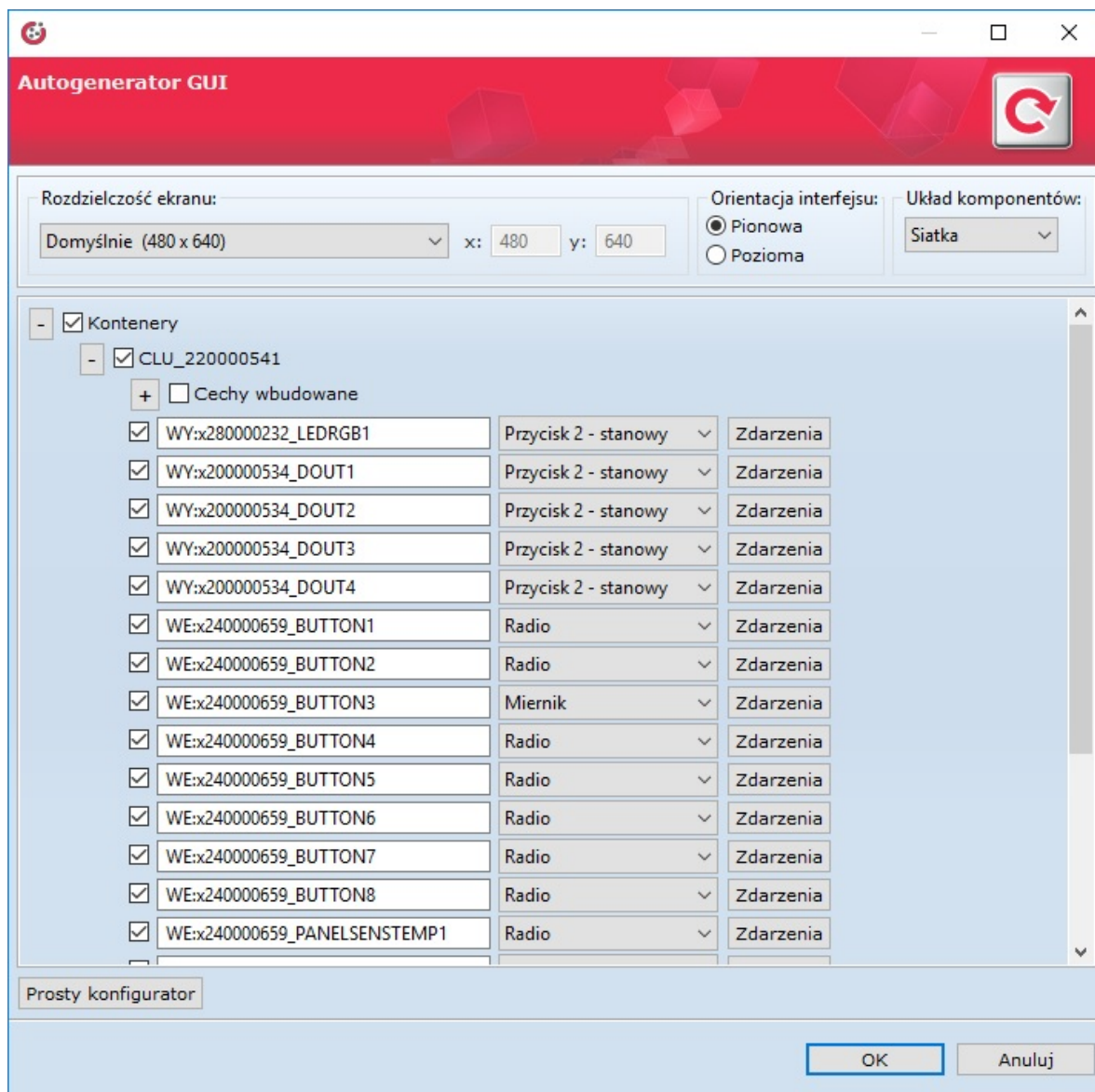
Zakładka **symulator** – daje użytkownikowi możliwość sprawdzenia wyglądu oraz sposobu działania utworzonego interfejsu z ekranu komputera (zanim zostanie wysłane do urządzenia mobilnego).



B. Konfigurator zaawansowany

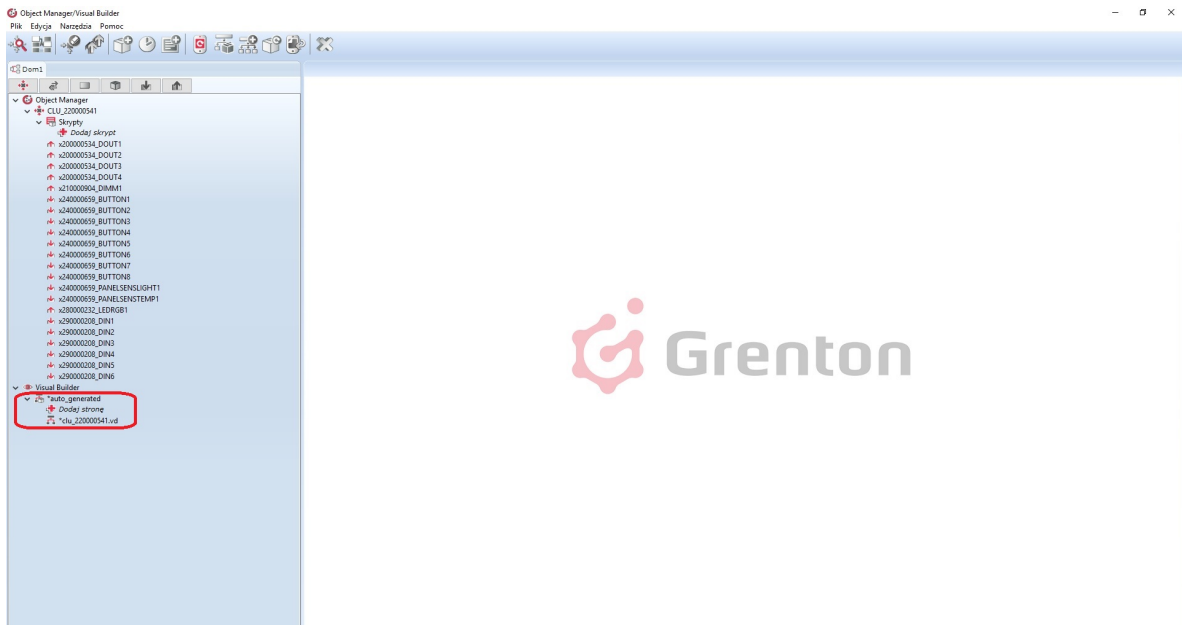
Po kliknięciu ikony **Generuj GUI** w oknie **Autogeneratora** istnieje możliwość wybrania opcji **Zaawansowany konfigurator**. Wybór tej opcji spowoduje otwarcie nowego okna, w którym należy wybrać:

- rozdzielczość, z jaką pracuje rządzenie mobilne;
- orientację interfejsu (pionową lub poziomą);
- układ komponentów (siatka lub lista);
- obiekty i cechy (z listy obiektów), które mają zostać zawarte w utworzonym interfejsie;
- wyświetlaną ikonę oraz zdarzenia dla każdego obiektu.



Następnie, po ustawieniu wszystkich parametrów i naciśnięciu **OK**, otwiera się okno utworzonego interfejsu. Okno to, poza polem nazwy utworzonego interfejsu, zawiera dwie zakładki: **wygląd** i **strony**. Ich funkcjonalności są dokładnie takie same, jak w przypadku konfiguratora prostego.

Po ustawieniu wszystkich parametrów w oknie utworzonego interfejsu i kliknięciu **OK**, na liście obiektów (pod ikoną utworzonego interfejsu) pojawiają się nowo utworzone strony zgodnie z poniższym rysunkiem:

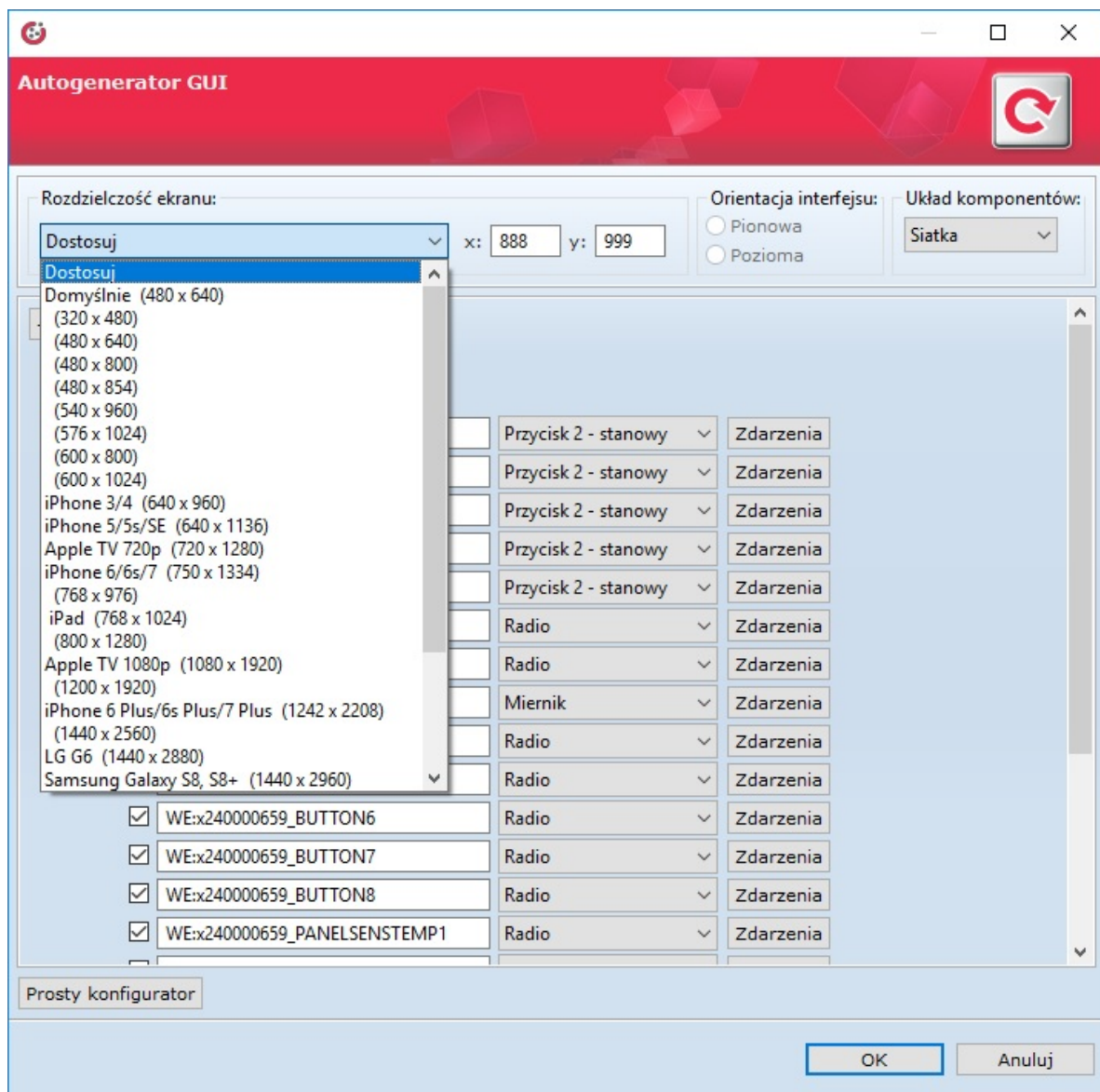


Podobnie jak w przypadku konfiguratora prostego - użytkownik ma możliwość dokonywania zmian w wygenerowanych stronach. Dwukrotne kliknięcie na ikonie strony otworzy arkusz edycji, zawierający dwie zakładki **Design** oraz **Symulator**.

5.2. Tworzenie interfejsu z własną rozdzielczością

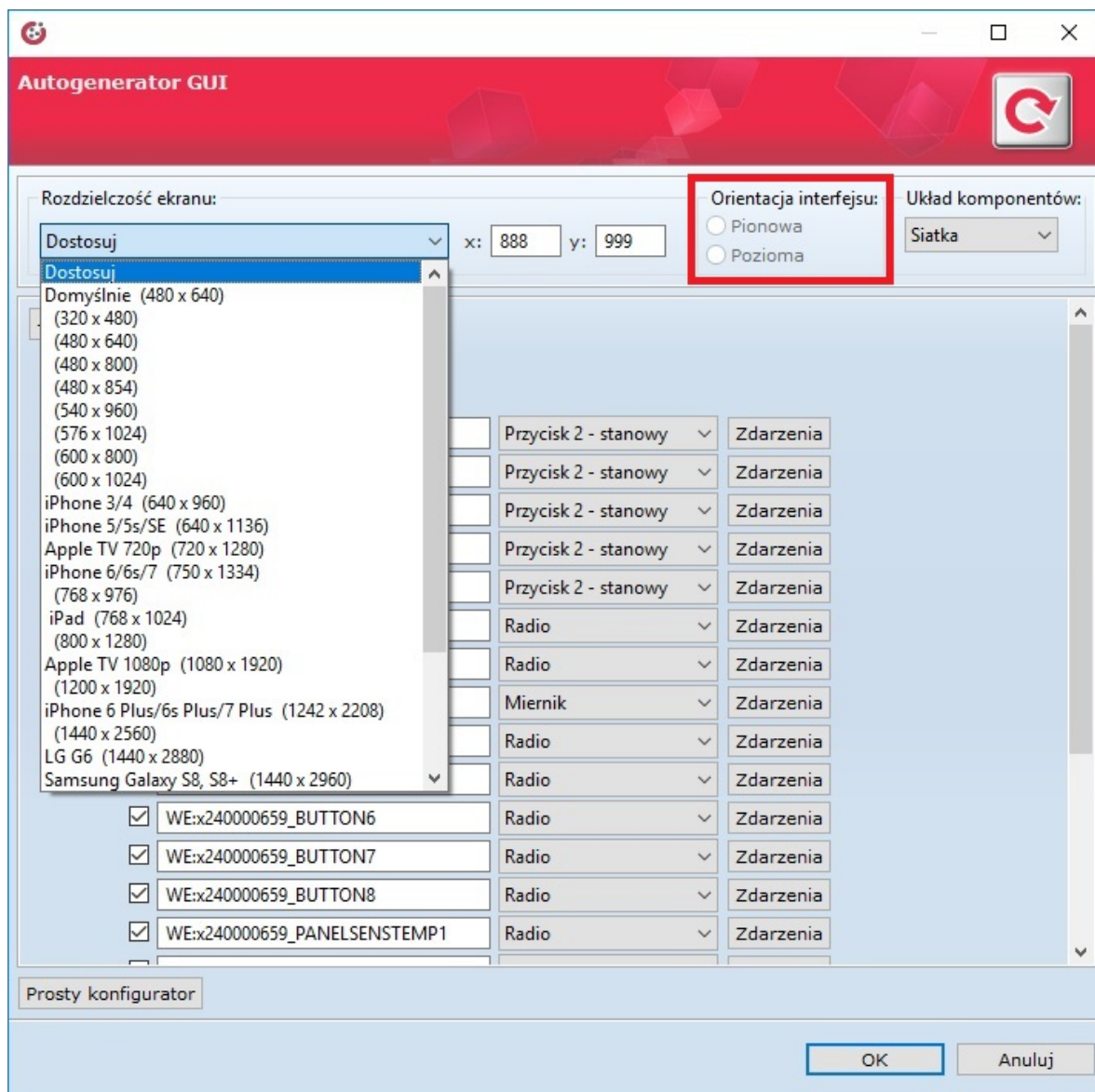
W przypadku konfiguratora zaawansowanego istnieje możliwość stworzenia interfejsu o własnej, wybranej rozdzielczości. W tym celu należy:

- Kliknąć w ikonę **Generuj GUI** w górnym oknie obiektów;
- Wybrać konfigurator zaawansowany;
- W oknie wyboru rozdzielczości zaznaczyć opcję *Dostosuj* oraz wprowadzić wymiary interfejsu;
- Wybrać pozostałe parametry interfejsu;
- Zaakceptować wprowadzone ustawienia.



5.3. Zmiana orientacji interfejsu z własną rozdzielczością

Korzystając z zaawansowanego konfiguratora zmiana orientacji interfejsu nie odbywa się w oknie Autogenerators GUI.



Chcąc dokonać zmiany orientacji interfejsu z własną rozdzielczością, po jego utworzeniu należy kolejno:

- Kliknąć dwukrotnie na jego nazwę;
- Przejść do zakładki **strony**;
- Usunąć wszystkie widoczne strony;
- Przejść do zakładki **wygląd**;
- Wybrać orientację - poziomą lub pionową;
- Ponownie przejść do zakładki **strony**;
- Dodać strony do interfejsu;
- Zaakceptować zmiany klikając OK;
- Wysłać interfejs do urządzenia mobilnego.

6. Konfiguracja wideodomofonu

6.1. Podłączenie i konfiguracja wideodomofonu

Konfiguracja wideodomofonu z systemem Grenton jest możliwa dla urządzeń podłączonych do sieci wspólnej (LAN) lub korzystających ze zdalnego dostępu do danej sieci, umożliwiających użycie strumienia *rtsp* kamery IP wbudowanej w urządzenie. Do poprawnej konfiguracji wideodomofonu potrzebne są co najmniej dwa konta na serwerze SIP.

Przykładowa konfiguracja została wykonana na domofonie *Akuvox R26*.

UWAGA! Panel Domofon dostępny jest dla Object Managera w wersji 1.2.0.180202 i wyższej.

A. Podłączenie wideodomofonu

Należy:

- Podłączyć wideodomofon do zasilania;
- Podpiąć wideodomofon przy pomocy kabla sieciowego RJ45 do routera.

B. Konfiguracja kamery

Panel wideodomofonu w aplikacji Grenton Home Manager wykorzystuje wizualizację kamery wbudowanej w urządzenie - chcąc mieć dostęp do obrazu kamery, należy wystawić odpowiedni port w ustawieniach sieci.

W celu dokonania konfiguracji portu, należy zalogować się do ustawień routera przy pomocy jego adresu IP w przeglądarce internetowej, dokonać odpowiednich zmian, po czym zapisać ustawienia:

- Należy wejść w ustawienia przekierowań ¹;
- Odnaleźć ustawienia portów;
- Ustawić port wyzwalania i przekazywania na **554** ² oraz protokół wyzwalania i przekazywania na **TCP**;
- Zapisać ustawienia;

UWAGA! Należy pamiętać, że w celu umożliwienia zdalnego połączenia aplikacji, konieczne jest ustawienie portu **1234** w protokole **UDP**.

- Na koniec należy przejść do listy aktualnie podłączonych urządzeń do sieci i zapisać adres IP wideodomofonu – będzie on potrzebny przy konfiguracji serwera *SIP*.

C. Konfiguracja SIP:

- Do stworzenia konfiguracji wideodomofonu potrzeba co najmniej dwa konta *SIP*;
- Przy pomocy przeglądarki należy zalogować się do wideodomofonu ³;
- Konieczne jest odszukanie ustawień konta *SIP* ⁴;
- Następnie należy wybrać jedno z dostępnych kont (np. **Konto_1**) i ustawić jego stan na aktywowany (`enabled`);
- W kolejnym kroku należy ustawić numer/nazwę konta *SIP* oraz jego hasło;
- Później konieczne jest wprowadzenie ustawień serwera *SIP* (`server IP` , `Port` , `Registration Period`) – ustawienia te powinny się pojawić przy zakładaniu kont;
- Następnie należy odnaleźć ustawienia kodeków wykorzystywanych w działaniu i aktywować kodeki typu *PCMU*;
- Na końcu konieczne jest odnalezienie ustawień Intercomu, gdzie należy skonfigurować numer/nazwę klienta, na który ma dzwonić wideodomofon (drugie założone konto *SIP*) i ustawić (jeśli istnieje taka możliwość) zachowanie urządzenia w momencie nieodebrania połączenia.

UWAGA! Jeśli przy ustawianiu Intercomu, konieczny jest wybór jednego konta z kilku konfigurowanych, należy wybrać wcześniej wybrane - w przykładzie **Konto_1!**

6.2. Stworzenie i konfiguracja interfejsu aplikacji

A. Dodawanie wideodomofonu do interfejsu aplikacji w programie Object Manager

W celu dodania wideodomofonu do interfejsu należy:

- Z menu głównego kliknąć przycisk *Dodaj interfejs*:



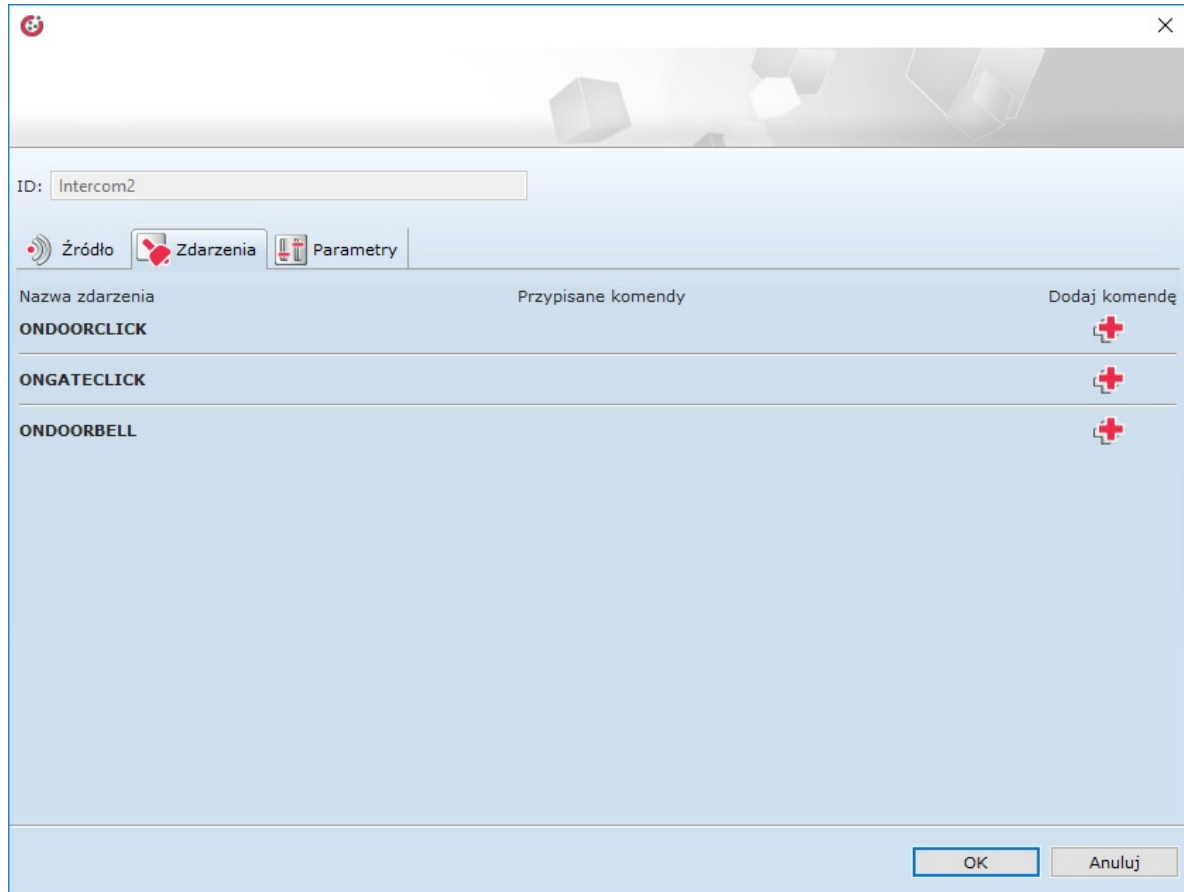
- Skonfigurować ustawienia interfejsu – wybrać: rozdzielczość, nazwę, skórę, dodać przynajmniej jedną stronę;
- Do utworzonej strony - z palety komponentów - dodać przycisk *Domofon*:

 **Domofon**

- W oknie, które otworzy się po dodaniu przycisku, ustawić parametry wideodomofonu:
 - **Źródło** – strumień *rtsp* odnaleziony w ustawieniach wideodomofonu lub jego dokumentacji;
 - **Adres IP** – adres IP wideodomofonu (zapisany wcześniej przy dokonywaniu jego konfiguracji);
 - **Konto** – numer/nazwa konta *SIP* wprowadzonego jako pierwsze w ustawieniach wideodomofonu – konto, z którego będą wykonywane połączenia (wybrane w punkcie 3 rozdziału „Podłączenie i konfiguracja wideodomofonu”):

A screenshot of the 'Domofon' configuration dialog box. The window has a title bar with a close button. Below the title bar is a decorative header with a 3D cube graphic. The main area contains an 'ID:' field with the value 'Intercom2'. Below this are three tabs: 'Źródło' (selected), 'Zdarzenia', and 'Parametry'. Under the 'Źródło' tab, there are three input fields labeled 'Źródło(Stream)', 'Adres IP', and 'Konto'. At the bottom right, there are 'OK' and 'Anuluj' buttons.

- Przejść do zakładki *Zdarzenia*:
 - Do zdarzenia `onDoorClick` przypisać metodę, która ma być wywoływana po naciśnięciu przycisku otwierania furtki w panelu domofonu w aplikacji Home Manager;
 - Zdarzenie `onGateClick` powiązać z metodą, która ma być wywoływana po naciśnięciu przycisku otwierania bramy wjazdowej w panelu domofonu w aplikacji Home Manager;
 - Do zdarzenia `onDoorBell` przypisać metodę lub skrypt, które mają być wykonywane w momencie wykonania połączenia – w chwili naciśnięcia dzwonka na domofonie:



- Kliknąć *OK*;
- Wysłać interfejs na urządzenie mobilne - [patrz pkt VIII.4.7.](#)

B. Konfiguracja aplikacji Home Manager

W celu przeprowadzenia konfiguracji należy:

- Otworzyć aplikację Home Manager;
- Z menu głównego wybrać *Ustawienia* (piktogram koła zębatego);
- Z sekcji *Domofon* wybrać *Konfiguracja SIP* ⁵;
- W ustawieniach podać:
 - **Adres serwera** – adres IP serwera *SIP*, na którym zostały założone konta;
 - **Nazwę użytkownika** – numer/nazwę konta *SIP*, na które będą wykonywane połączenia – podanego w ustawieniach domofonu, jako konto docelowe do odbierania połączeń (wybrane w punkcie 3 rozdziału „Podłączenie i konfiguracja wideodomofonu”);
 - **Hasło** – hasło do powyższego konta *SIP*, na które będą wykonywane połączenia z domofonu:



Adres serwera

178.183.115.179

Nazwa użytkownika

20031

Hasło

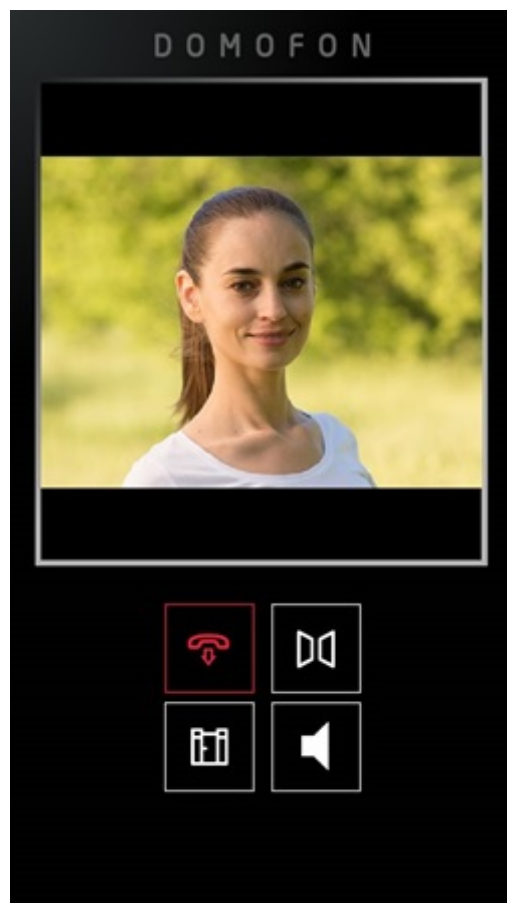
.....

ZAPISZ ANULUJ

- Zatwierdzić wprowadzone ustawienia przyciskiem *Zapisz*;
- Poprawnie przeprowadzona konfiguracja spowoduje, że na ekranie urządzenia mobilnego - w jego pasku powiadomień - pojawi się informacja o połączeniu z serwerem *SIP*;
- Wyjść z ustawień aplikacji.

6.3. Wykonanie połączenia z domofonu

1. Na domofonie nacisnąć przycisk dzwonienia.
2. Niezależnie od tego czy aplikacja Home Manager w urządzeniu mobilnym jest otwarta, zostanie nawiązane połączenie – na ekranie pojawi się panel wideodomofonu.
3. Przycisk po lewej stronie u góry służy do odbierania połączenia – dopóki nie zostanie użyty - rozmówca nic nie usłyszy, a domofon ciągle będzie dzwonił.
4. Z pozycji panelu wideodomofonu można wyzwolić zdarzenie `onDoorClick` oraz `onGateClick`, które zadziałają w zależności od ustawienia dokonanego w programie Object Manager.
5. W panelu domofonu znajduje się również przycisk, służący do załączania/wyłączania trybu głośnomówiącego.



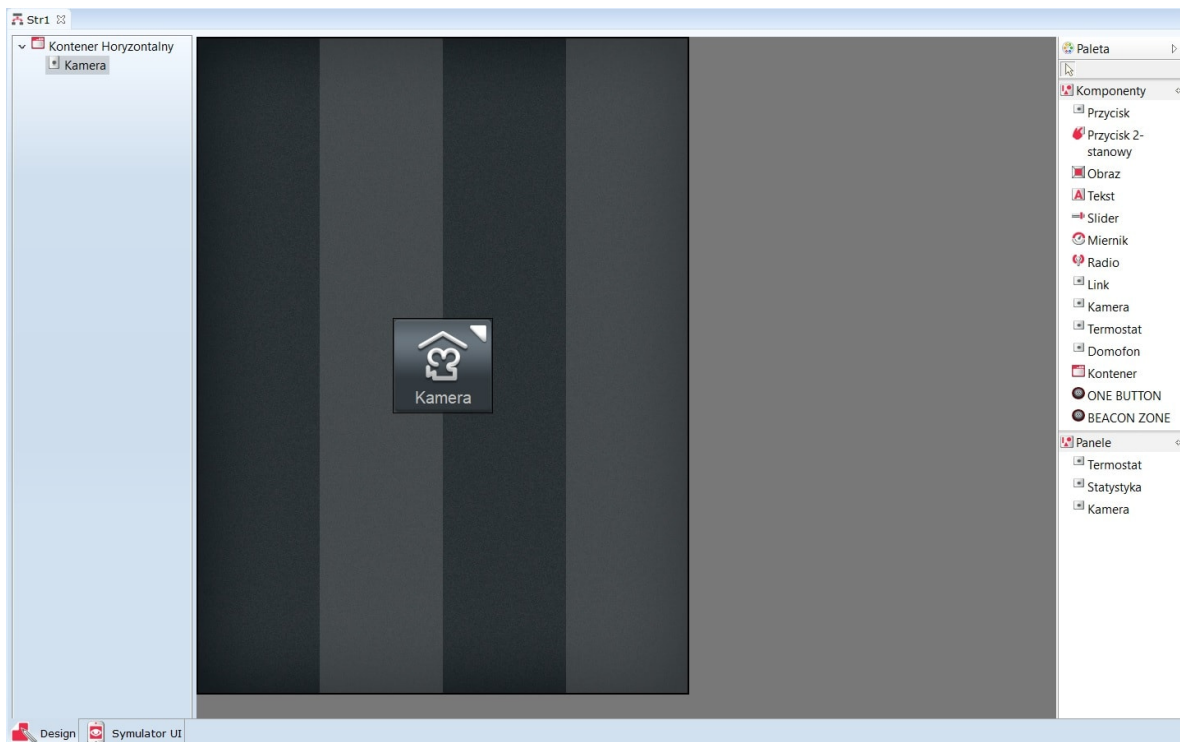
7. Obsługa obrazu z kamer

Aplikacja Home Manager umożliwia dokonywanie podglądu obrazu z kamer IP w dowolnym interfejsie. Nie ma ograniczeń w liczbie obsługiwanych kamer, jednakże obraz z nich będzie wyświetlany pojedynczo.

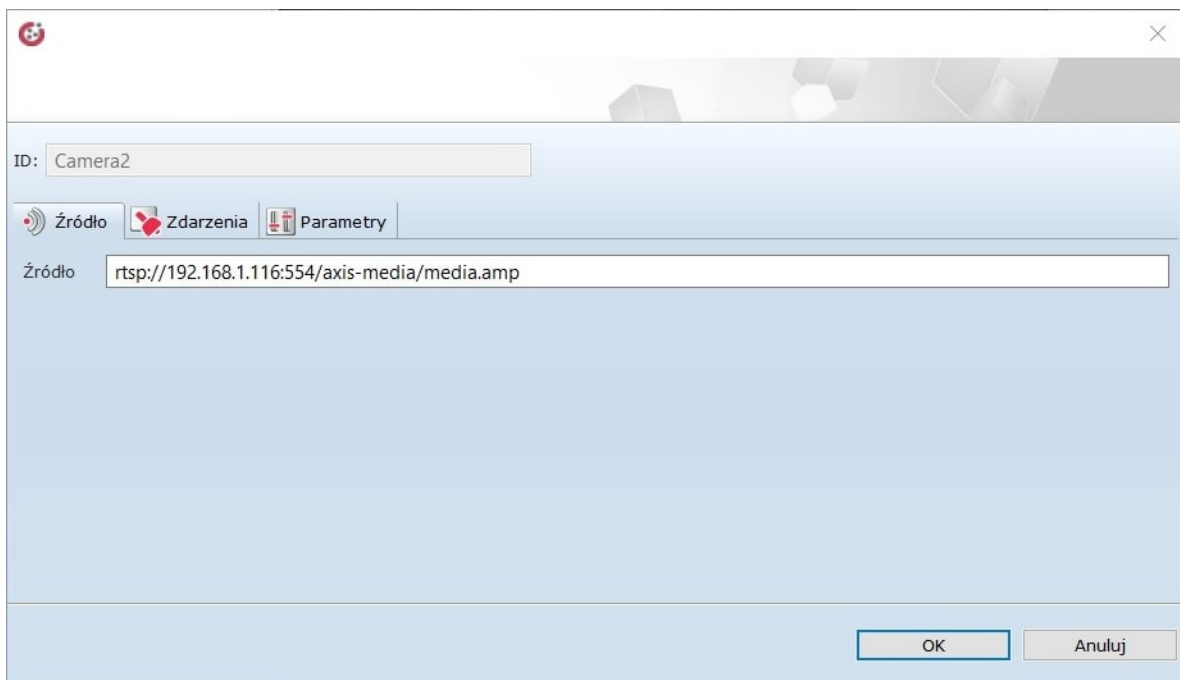
UWAGA! Aplikacja Home Manager poprawnie wyświetla obrazy z kamer obsługujących protokół RTSP oraz kodek H.264 w strumieniu MPEG.

A. Dodawanie komponentu *Kamera*

Chcąc dodać do interfejsu obraz z kamery, należy przeciągnąć do obszaru roboczego komponent *Kamera* dostępny na liście obiektów:



Następnie - jako źródło dla dodanego obiektu, konieczne jest wpisanie adresu kamery, której obraz ma być wyświetlany. Dodawana kamera musi zostać uprzednio skonfigurowana w taki sposób, aby możliwe było otwarcie podglądu z niej za pośrednictwem protokołu RTSP.

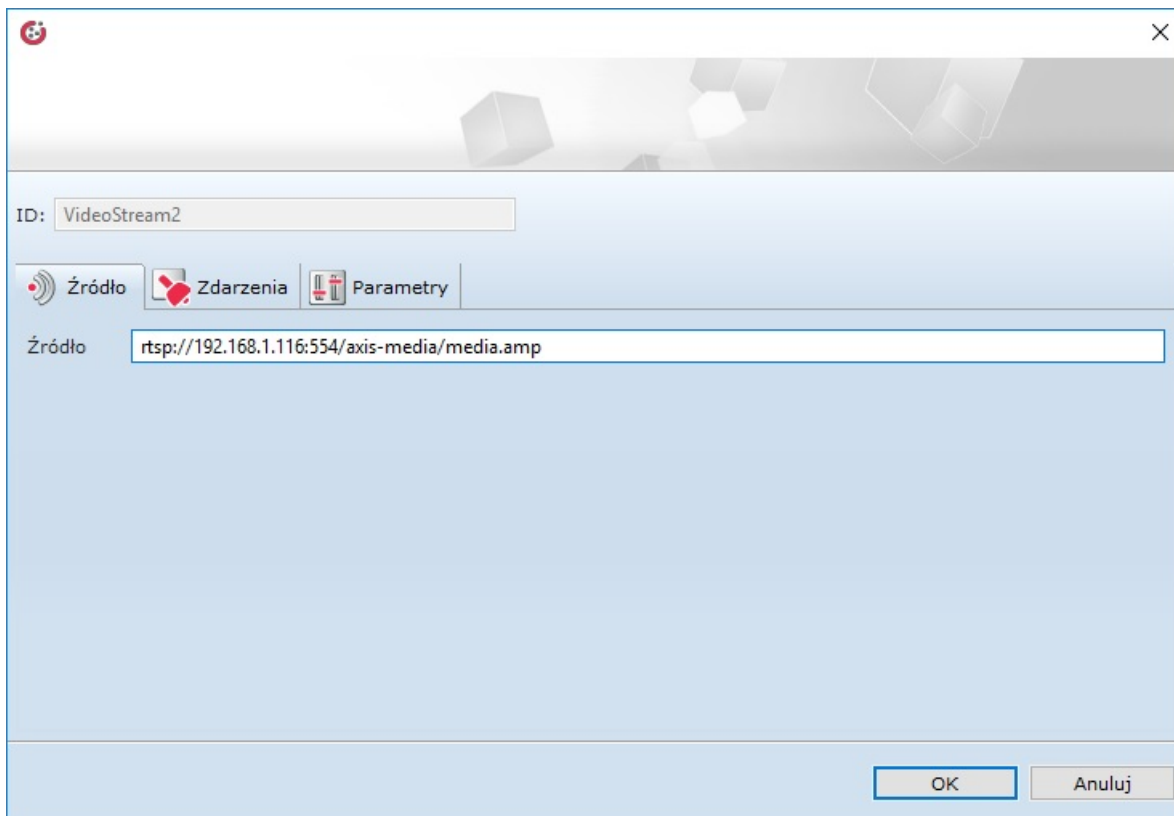


Po wysłaniu utworzonego interfejsu, obraz z kamery zostanie wyświetlony na ekranie urządzenia mobilnego po naciśnięciu wstawionego obiektu.

B. Dodanie panelu *Kamera*

Istnieje możliwość dodania do interfejsu obrazu z kamery przy pomocy panelu *Kamera*. W tym celu należy przeciągnąć go na pustą stronę interfejsu.

Następnie - jako źródło dla dodanego obiektu, konieczne jest wpisanie adresu kamery, której obraz ma być wyświetlany. Dodawana kamera musi zostać uprzednio skonfigurowana w taki sposób, aby możliwe było otwarcie podglądu z niej za pośrednictwem protokołu RTSP.



Po wysłaniu utworzonego interfejsu, obraz z kamery zostanie wyświetlony na ekranie urządzenia mobilnego po naciśnięciu strony z dodanym panelem *Kamera*.

8. Zdalny dostęp aplikacji mobilnej do systemu

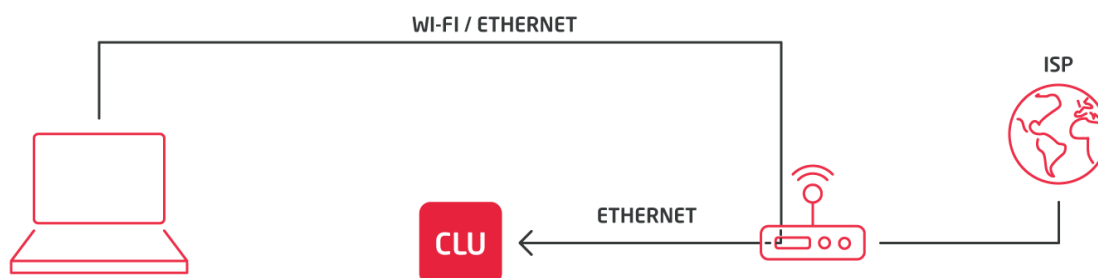
System Grenton daje możliwość swobodnego sterowania domem z każdego miejsca na świecie. Siedząc w pracy, czy będąc w podróży służbowej, możemy bez problemu kontrolować stan naszej inwestycji i zarządzać jego funkcjami w bardzo prosty sposób.

By zdalny dostęp do systemu Grenton był możliwy, powinien spełniać następujące wymagania:

- system Grenton musi być w pełni skonfigurowany;
- utworzone interfejsy aplikacji mobilnej muszą być przesłane do urządzeń mobilnych, z których realizowany ma być zdalny dostęp;
- dostawca usług internetowych musi zapewnić dostęp do stałego, zewnętrznego adresu IP;
- router/punkt dostępu musi mieć możliwość przekierowania portów.

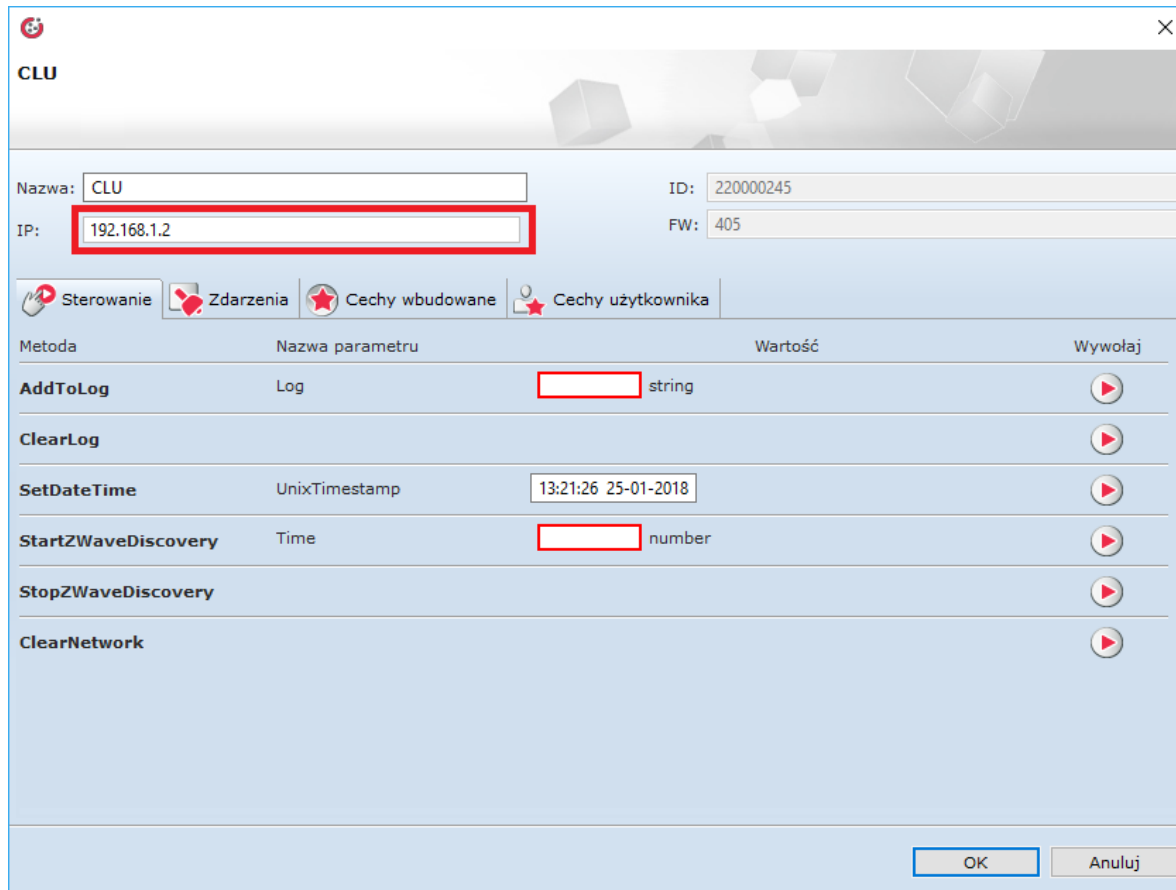
8.1. Konfiguracja systemu

Instrukcja została przygotowana dla układu, w którym jednostka centralna połączona jest do routera widocznego przez zewnętrzny, stały adres IP.



Przed przystąpieniem do konfiguracji zdalnego dostępu należy:

- upewnić się, czy jednostka centralna została połączona do routera sieci lokalnej oraz czy adres jednostki centralnej został nadany z puli adresów tego routera;
- sprawdzić adres jednostki centralnej nadanej przez router sieci lokalnej (w tym celu kliknąć dwukrotnie na ikonę jednostki centralnej);
- w nowo otwartym oknie odczytać informacje z zaznaczonego poniżej pola:



The screenshot shows a window titled "CLU" with a close button in the top right corner. Below the title bar, there are four input fields: "Nazwa:" with the value "CLU", "ID:" with "220000245", "IP:" with "192.168.1.2" (highlighted with a red box), and "FW:" with "405". Below these fields are four tabs: "Sterowanie", "Zdarzenia", "Cechy wbudowane", and "Cechy użytkownika". Under the "Sterowanie" tab, there is a table with the following columns: "Metoda", "Nazwa parametru", "Wartość", and "Wywołaj".

Metoda	Nazwa parametru	Wartość	Wywołaj
AddToLog	Log	<input type="text" value=""/> string	
ClearLog			
SetDateTime	UnixTimestamp	13:21:26 25-01-2018	
StartZWaveDiscovery	Time	<input type="text" value=""/> number	
StopZWaveDiscovery			
ClearNetwork			

At the bottom right of the window are two buttons: "OK" and "Anuluj".

Dla analizowanego przypadku adres jednostki centralnej to: *192.168.1.2*. Adres ten będzie wykorzystywany do wykonania przekierowania portów.

8.2. Ustawienie przekierowania portów w routerze sieci lokalnej

UWAGA! Ustawienia przekierowania portów dla każdego routera mogą się różnić! Poniżej przedstawiona została ogólna procedura postępowania.

W celu ustawienia przekierowania portów konieczne jest:

- wejście do ustawień routera sieci lokalnej - by tego dokonać, wymagane jest podłączenie do sieci lokalnej, w której znajduje się jednostka centralna;
- otwarcie przeglądarki internetowej i wpisanie w polu adresu, adresu IP routera sieci lokalnej (w celu wejścia do jego ustawień) - domyślny adres znajduje się najczęściej na jego spodzie;
- zalogowanie się przy pomocy danych logowania - domyślny login i hasło znajdują się najczęściej w formie naklejki na spodzie routera sieci lokalnej (domyślne dane routera można również znaleźć w dedykowanych narzędziach internetowych);

UWAGA Jeżeli wprowadzony adres IP lub dane logowania są nieprawidłowe, oznacza to, że zostały one zmienione przez administratora sieci. W celu uzyskania dostępu do ustawień routera należy się z nim skontaktować.

- odszukanie w ustawieniach routera pozycji dotyczącej przekierowania portów (*Port Forwarding* lub podobne);
- wykonanie przekierowania zewnętrznego portu 1234 na wewnętrzny port 1234 lokalnego adresu jednostki centralnej z wykorzystaniem protokołu UDP - przykładowa konfiguracja została zamieszczona poniżej:

The screenshot shows the Tomato router's OpenLinksys web interface. The left sidebar contains a navigation menu with categories like Status, Bandwidth, IP Traffic, Tools, Basic, Advanced, Port Forwarding (selected), Access Restriction, QoS, Bandwidth Limiter, USB and NAS, VPN Tunneling, Administration, and About. The main content area is titled 'Port Forwarding' and shows a table with one rule. The rule is active (On), uses the UDP protocol, and forwards external port 1234 to internal port 1234 on the internal IP address 192.168.1.2. The description is 'CLU1'. Below the table, there are instructions for the fields: Src Address (optional), Ext Ports, Int Port (optional), and Int Address.

On	Proto	Src Address	Ext Ports	Int Port	Int Address	Description
On	UDP		1234	1234	192.168.1.2	CLU1

TCP

- **Src Address** (*optional*) - Forward only if from this address. Ex: "1.2.3.4", "1.2.3.4 - 2.3.4.5", "1.2.3.0/24", "me.example.com".
- **Ext Ports** - The ports to be forwarded, as seen from the WAN. Ex: "2345", "200,300", "200-300,400".
- **Int Port** (*optional*) - The destination port inside the LAN. If blank, the destination port is the same as *Ext Ports*. Only one port per entry is supported when forwarding to a different internal port.
- **Int Address** - The destination address inside the LAN.

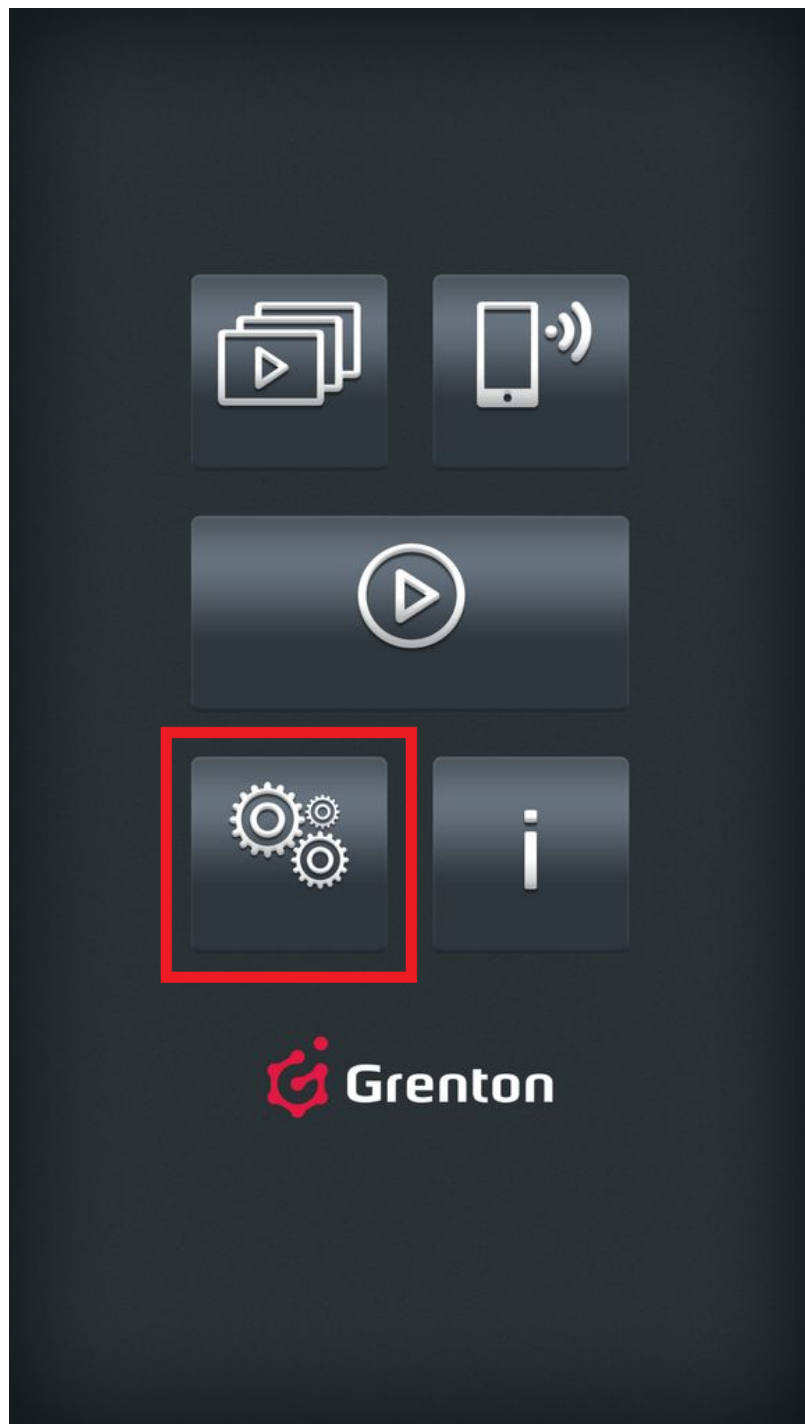
- zapisanie ustawień routera - w niektórych przypadkach konieczne może być zrestartowanie urządzenia.

UWAGA! Należy się upewnić, czy komunikacja z zewnątrz nie jest blokowana przez wewnętrzne ustawienia routera.

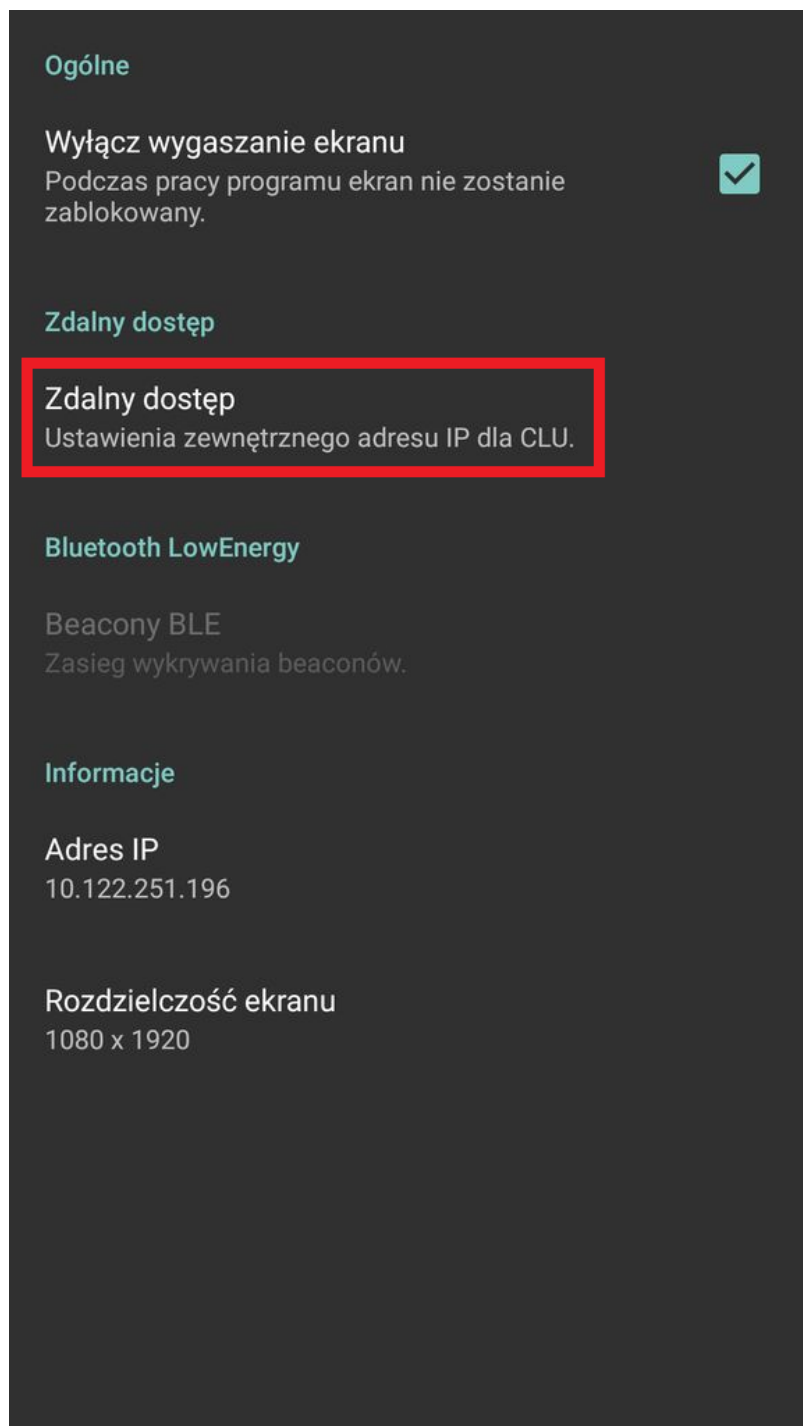
8.3. Konfiguracja aplikacji mobilnej Home Manager

Przy tworzeniu konfiguracji należy kolejno:

- uruchomić aplikację mobilną Home Manager;
- upewnić się, że do aplikacji mobilnej wgrany został interfejs, za pomocą którego realizowana będzie funkcjonalność zdalnego dostępu;
- przejść do ekranu głównego aplikacji mobilnej i wejść do *Ustawień* (klikając w ikonę koła zębatego znajdującą się w lewym dolnym rogu ekranu):



- w ustawieniach kliknąć w pozycję *Zdalny dostęp, Ustawienia zewnętrznego adresu IP dla CLU*:



- z listy dostępnych interfejsów wybrać ten, dla którego ma zostać skonfigurowany zdalny dostęp;
- następnie wyświetli się okno z aktualną konfiguracją sieciową systemu z informacjami o adresie:
 - lokalnym (lokalny adres IP jednostki centralnej);
 - zdalnym (zewnętrzny adres IP sieci, do której podłączona jest jednostka centralna wraz z numerem portu, który został jej przypisany):

Adres lokalny: 192.168.1.2

Adres zdalny: 79.189.67.44:1234

UWAGA! Jeżeli podany adres zdalny różni się od rzeczywistego zewnętrznego adresu IP, należy dokonać zmiany klikając w okno adresu. W nowo otwartym oknie konieczne jest dokonanie zmian zgodnie z rzeczywistym adresem IP urządzenia. W celu zaakceptowania zmian należy nacisnąć *OK*.

Adres lokalny: 192.168.1.2
Adres zdalny: 79.189.67.44:1234

Ustawienia adresu zdalnego

IP

Port

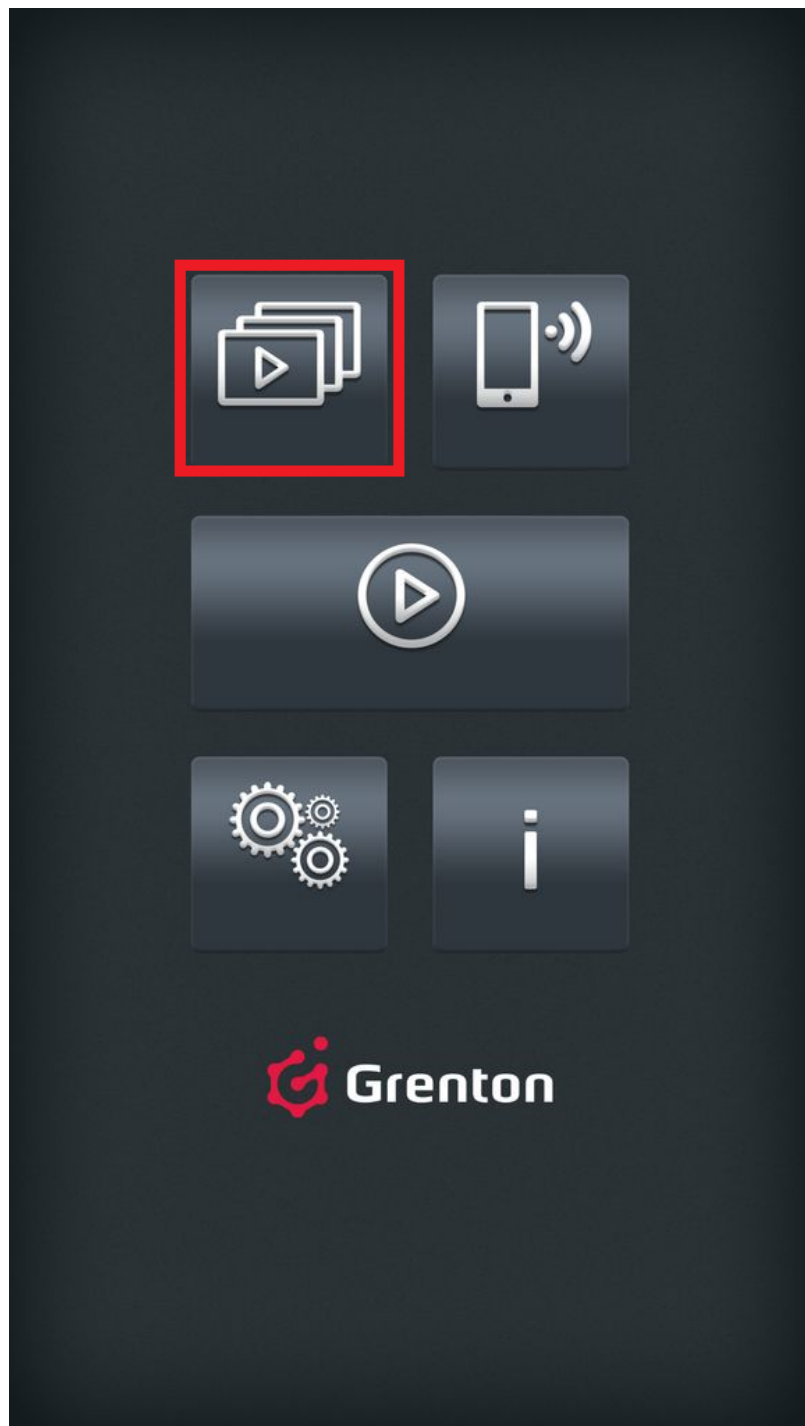
ANULUJ OK

9.4. Uruchomienie zdalnego dostępu

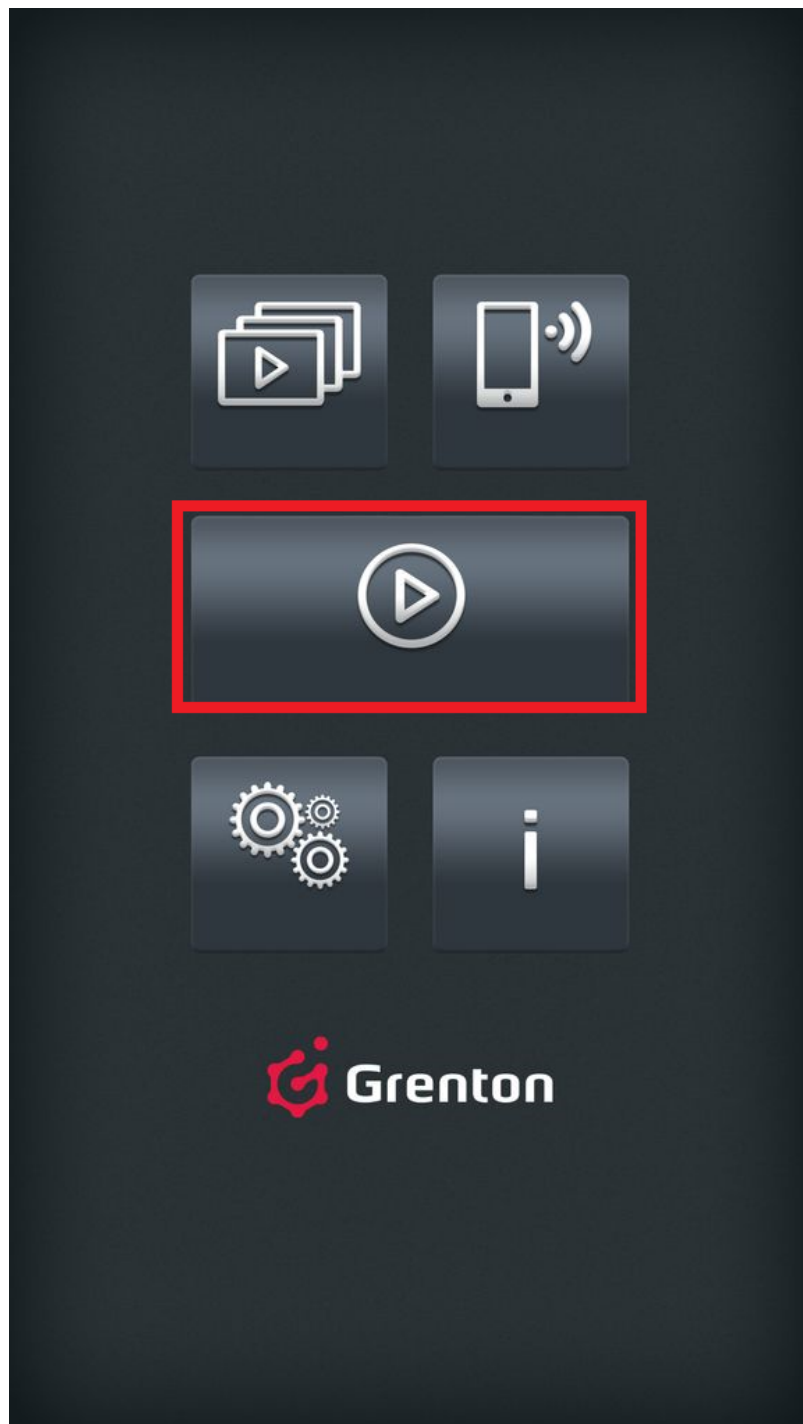
Aplikacja mobilna Grenton Home Manager automatycznie przełącza się z komunikacji lokalnej na komunikację zdalną. Aby zdalny dostęp był możliwy, urządzenie mobilne musi spełniać następujące warunki:

- zdalny dostęp musi być poprawnie skonfigurowany;
- urządzenie musi być podłączone do sieci internetowej innej niż lokalna (inna niż ta, do której podłączony jest system) lub musi mieć włączone dane sieci komórkowej (*internet w telefonie*).

W celu uruchomienia zdalnej komunikacji z systemem należy otworzyć interfejs, dla którego wykonywana była konfiguracja zdalnego dostępu poprzez wybranie go z listy interfejsów:



Jeśli dany interfejs był ustawiony jako domyślny należy kliknąć przycisk:



W pierwszej kolejności aplikacja Home Manager podejmie próbę nawiązania połączenia poprzez sieć lokalną. W momencie wykrycia braku takiej możliwości nastąpi przełączenie na komunikację zdalną.

IX. Obiekty CLU

1. Timery

Timery są wirtualnymi obiektami tworzonymi w ramach danego CLU. Object Manager umożliwia utworzenie maksymalnie 64 timerów. Timery mogą być wykorzystywane wszędzie tam, gdzie potrzebne jest wywołanie metody po określonym czasie lub też jej cykliczne wywoływanie.

Sam timer jest również obiektem w OM i tak jak każdy inny obiekt posiada: swoje cechy, metody i zdarzenia oraz wartości początkowe. Timer może pracować w dwóch trybach:

- **Countdown** Po wystartowaniu, odlicza ustalony czas. Po zakończeniu odliczania uruchamiana jest metoda powiązana ze zdarzeniem `OnTimer`, a timer zatrzymuje się i nie odlicza, aż do następnego uruchomienia metodą `Start`.
- `Interval` Timer cykliczny - po starcie zaczyna odliczać ustawiony czas. Po jego upływie timer wywołuje metodę powiązaną ze zdarzeniem `OnTimer`, a sam timer ponownie zaczyna odliczać zadany czas. Sytuacja powtarza się, aż do momentu zatrzymania metodą `Stop`.

A. Tworzenie timerów

W celu utworzenia timera w danym CLU należy zaznaczyć to, w którym ma się pojawić, a następnie wybrać z menu górnego ikonę `Dodaj obiekt CLU`.



Po kliknięciu ikony pojawia się okno wyboru z listą dostępnych obiektów, na której należy odszukać i wybrać obiekt `Timer`. Po zaznaczeniu, wciśnięciu `OK`, konieczne jest nadanie nazwy nowemu timerowi. Następnie należy ustawić czas timera [wartość podawana jest w ms] oraz wybrać tryb pracy [`Countdown` lub `Interval`]. Ustawiony czas będzie jednocześnie czasem w warunkach początkowych. Utworzony timer pojawi się na liście obiektów zaznaczonego CLU.

Utworzony timer również jest obiektem CLU i dlatego podobnie jak inne, fizyczne obiekty, jest obsługiwany przez konfigurator obiektów - [patrz pkt VIII.4.1.](#)

B. Parametry konfiguracyjne obiektu Timer

CECHY

Nazwa	Opis
<code>Time</code>	Zliczany czas (w ms)
<code>Mode</code>	Tryb pracy timera: 0 – zliczenie w dół (countdown), 1 – cykliczny (interval)
<code>State</code>	Aktualny stan pracy timera: 0 – zatrzymany (stopped), 1 – liczy (counting), 2 – pauza (paused)
<code>value</code>	Czas jaki pozostał do wyzwolenia zdarzenia <code>OnTimer</code> (w ms)

METODY

Nazwa	Opis
<code>SetTime</code>	Ustawia czas timera (w ms)
<code>SetMode</code>	Ustawia tryb pracy
<code>Start</code>	Uruchamia timer
<code>Stop</code>	Zatrzymuje timer
<code>Pause</code>	Pauza timera

ZDARZENIA

Nazwa	Opis
OnTimer	Zdarzenie wywoływane przy zliczeniu timera
OnStart	Zdarzenie wywoływane przy uruchomieniu timera
OnStop	Zdarzenie wywoływane przy zatrzymaniu timera
OnPause	Zdarzenie wywoływane przy pauzie timera

2. Kalendarz

Kalendarze, podobnie jak timery, są wirtualnymi obiektami tworzonymi przez użytkownika w CLU. Możliwe jest utworzenie maksymalnie 64 kalendarzy na jednym CLU. Jeden kalendarz utworzony na CLU to jedna reguła wykonywana w określonym dniu i godzinie lub generowana cyklicznie w wybrane dni, miesiące czy godziny (z dokładnością co do jednej minuty). Reguły można tworzyć przy pomocy interfejsu graficznego lub też używając składni zgodnej z regułami CRON systemu LINUX.

A. Tworzenie kalendarza

Chcąc utworzyć kalendarz należy zaznaczyć CLU, w ramach którego ma się znaleźć, a następnie z menu górnego uruchomić **Dodaj obiekt CLU**.



W otwartym oknie wybrać **CaTendar**. Po wciśnięciu **OK**, wpisać nazwę dla tworzonego kalendarza. Object Manager wyświetli okno właściwości utworzonego obiektu.

UWAGA! Kalendarz po utworzeniu i wysłaniu konfiguracji do CLU automatycznie staje się aktywny – aby zatrzymać pracę kalendarza należy wywołać metodę **STOP**.

B. Właściwości kalendarza

Okno właściwości kalendarza zawiera cztery zakładki:

CLU->kalendarz

Nazwa: Typ:

Sterowanie
 Zdarzenia
 Cechy wbudowane
 Reguła

Metoda	Nazwa parametru	Wartość	Wywołaj
Start			<input type="button" value="▶"/>
Stop			<input type="button" value="▶"/>
SetRule	Rule	<input type="text" value=""/> string	<input type="button" value="▶"/>
CancelNext	count	<input type="text" value=""/> number	<input type="button" value="▶"/>

- **Sterowanie** – zawiera metody kalendarza;
- **Zdarzenia** – zawiera zdarzenia kalendarza;
- **Cechy wbudowane** – zawiera listę cech kalendarza;
- **Reguła** – zawiera interfejs umożliwiający w prosty sposób formułowanie reguł.

C. Reguły kalendarza

Wpisywanie reguł dla kalendarza może odbywać się na dwa sposoby:

- za pomocą interfejsu graficznego w zakładce **Reguła**;
- poprzez wpisanie reguły CRON z wykorzystaniem metody **SetRule** w zakładce sterowanie lub **Rule** jako cechy wbudowanej.

D. Tworzenie reguły kalendarza za pomocą interfejsu graficznego

W zakładce **Reguła** znajduje się graficzny interfejs, za pomocą którego użytkownik może w prosty sposób ustawić parametry reguły dla kalendarza.

UWAGA! Po wpisaniu parametrów reguły za pomocą interfejsu graficznego, wartość **Rule** w **Cechach wbudowanych** wypełniana jest automatycznie zgodnie z zaznaczonymi kryteriami.

CLU->kalendarz

Nazwa: Typ:

Czas: :

Kryteria:

<input type="checkbox"/> Miesiąc <input type="checkbox"/> Styczeń <input type="checkbox"/> Luty <input type="checkbox"/> Marzec <input type="checkbox"/> Kwiecień <input type="checkbox"/> Maj <input type="checkbox"/> Czerwiec <input type="checkbox"/> Lipiec <input type="checkbox"/> Sierpień <input type="checkbox"/> Wrzesień <input type="checkbox"/> Październik <input type="checkbox"/> Listopad <input type="checkbox"/> Grudzień	<input type="checkbox"/> Dzień miesiąca <input type="text"/>	<input type="checkbox"/> Dzień tygodnia <input type="checkbox"/> Niedziela <input type="checkbox"/> Poniedziałek <input type="checkbox"/> Wtorek <input type="checkbox"/> Środa <input type="checkbox"/> Czwartek <input type="checkbox"/> Piątek <input type="checkbox"/> Sobota	<input type="checkbox"/> Dni parzyste/nieparzyste <input type="radio"/> Dni parzyste <input type="radio"/> Dni nieparzyste
---	---	--	--

Znajdują się tam dwie sekcje, w których użytkownik wybiera parametry dla reguły:

- **czas** – zawiera dwa pola: pierwsze, w którym wpisywana jest godzina (lub zakres godzin), druga oznacza minutę (lub zakres minut). Wartości w tych polach należy wpisywać zgodnie regułą CRON;
- **kryteria** – zawiera pozostałe parametry dla reguły. Użytkownik dokonuje wyboru poprzez postawienie znacznika przy odpowiednim polu.

E. Tworzenie reguły kalendarza zgodnie z formatem CRON

Reguły kalendarza tworzy się poprzez wpisanie reguły w *Cechach wbudowanych* utworzonego kalendarza - w polu `Rule` lub poprzez metodę `setRule`. Szczegółowe informacje na temat sposobu tworzenia reguł w formacie CRON można znaleźć w dokumentacji kalendarza CRON.

F. Parametry konfiguracyjne obiektu Kalendarz

CECHY

Nazwa	Opis
<code>Rule</code>	Reguła kalendarza w formacie CRON lub <code>ERROR</code> w przypadku wprowadzenia błędnej reguły
<code>SinceLastRun</code>	Czas (w minutach) od ostatniego spełnienia warunku reguły
<code>ToNextRun</code>	Czas (w minutach) do następnego wywołania akcji kalendarza
<code>State</code>	Stan działania kalendarza: 1 (kalendarz aktywny) lub 0 (nieaktywny)

METODY

Nazwa	Opis
<code>Start</code>	Przełączenie w stan aktywny (<code>State =1</code>)
<code>Stop</code>	Przełączenie w stan zatrzymania (<code>State =0</code>)
<code>SetRule</code>	Ustawienie reguły kalendarza
<code>CancelNext</code>	Anulowanie wywołania wskazanej liczby najbliższych akcji kalendarza

ZDARZENIA

Nazwa	Opis
<code>OnCalendar</code>	Zdarzenie generowane w momencie wywołania akcji kalendarza
<code>OnStart</code>	Zdarzenie generowane w momencie wznowienia pracy kalendarza
<code>OnStop</code>	Zdarzenie generowane w momencie zablokowania pracy kalendarza
<code>OnCancel</code>	Zdarzenie generowane w momencie anulowania najbliższych akcji kalendarza

3. Harmonogram

Harmonogram jest obiektem wirtualnym służącym do ustawiania wartości dowolnej cechy w rozkładzie tygodniowym. Wartości te ustawiane są przy pomocy interfejsu graficznego dla każdego dnia i każdej godziny z rozdzielczością 15-minutową, 30-minutową lub godzinną.

W jednym CLU, można utworzyć do 64 harmonogramów.

UWAGA! Po utworzeniu harmonogramu (po wysłaniu nowej konfiguracji do CLU) staje się on automatycznie aktywny. Chcąc zatrzymać pracę harmonogramu, należy wywołać metodę `STOP`.

A. Tworzenie harmonogramu

W celu utworzenia harmonogramu należy zaznaczyć CLU, w ramach którego ma zostać umieszczony, a następnie z menu górnego uruchomić **Dodaj obiekt CLU**.



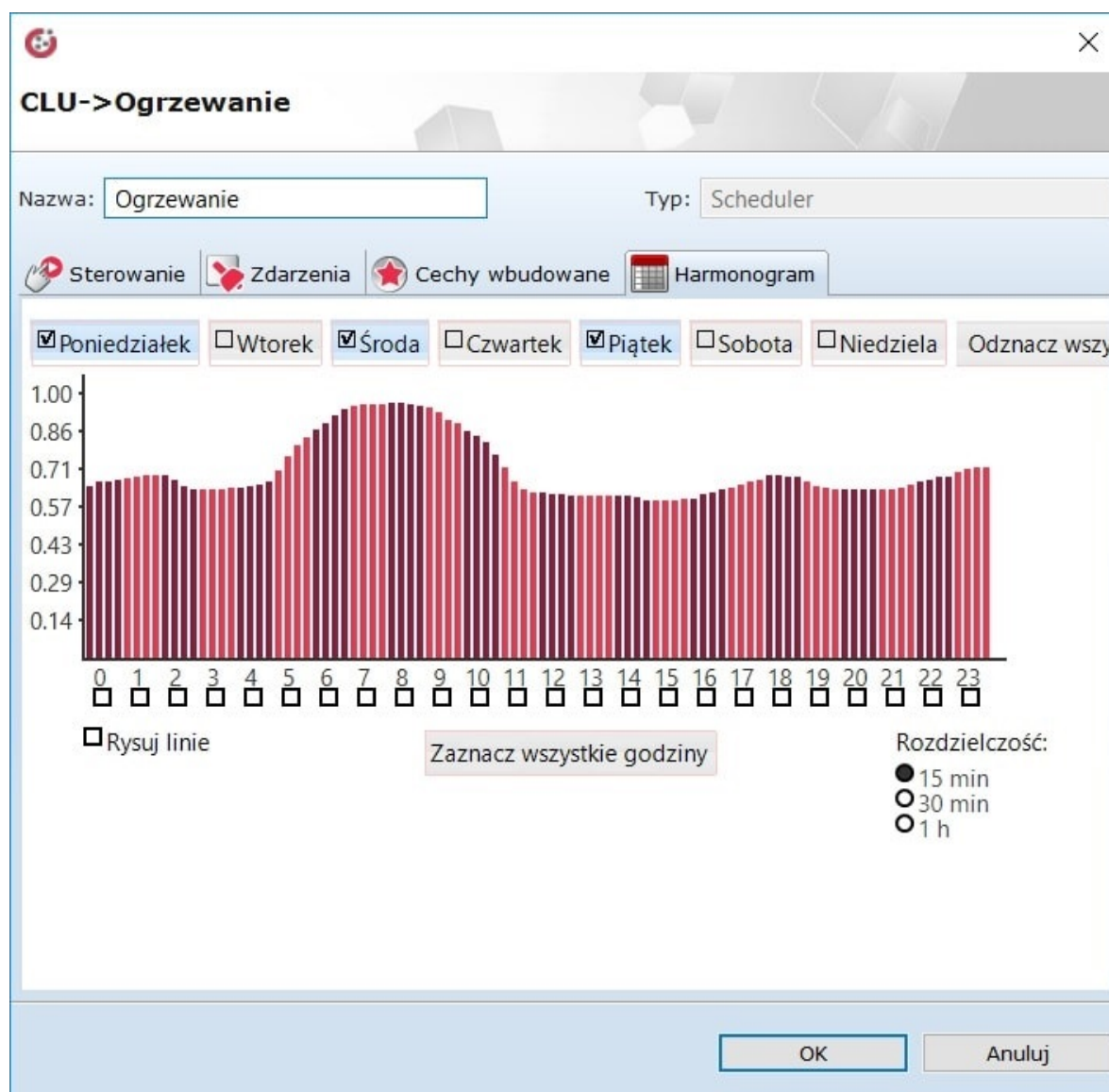
W otwartym oknie wyboru konieczne jest odszukanie i wybranie obiektu **ScheduTer**. Po wpisaniu nazwy, na ekranie otworzy się okno właściwości harmonogramu.

W oknie tym znajdują się cztery zakładki:

- **Sterowanie** – zawiera metody harmonogramu;
- **Zdarzenia** – zawiera zdarzenia harmonogramu;
- **Cechy wbudowane** – zawiera listę cech harmonogramu;
- **Harmonogram** – zawiera interfejs graficzny umożliwiający proste formułowanie wartości dla całego zakresu harmonogramu.

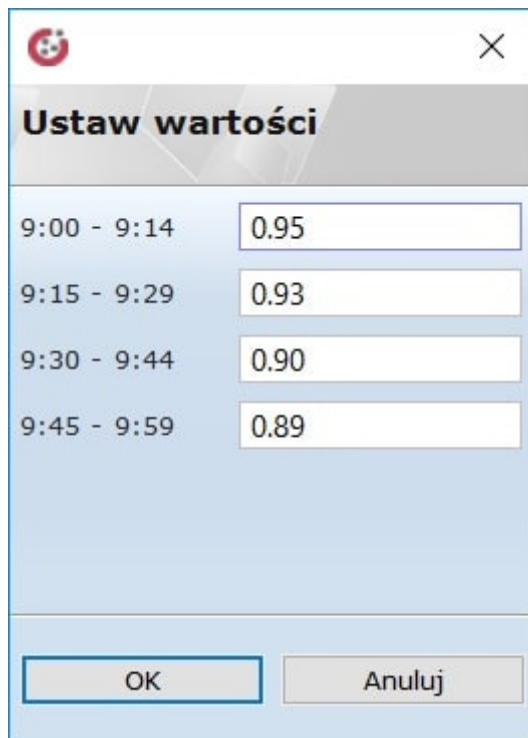
B. Formułowanie wartości dla harmonogramu

W zakładce *Harmonogram* (w oknie właściwości) znajduje się interfejs graficzny, dzięki któremu istnieje możliwość zdefiniowania wartości dla wybranego wyjścia.



Harmonogram pozwala na ustawienie wartości dla 7 dni (w zakresie jednego tygodnia) z rozdzielczością 15-minutową. Można ustawić wartości dla każdego dnia z osobna lub kilku dni jednocześnie. Dzień, dla którego aktualnie wpisywane są wartości, wyróżniony jest czarnym znacznikiem z lewej strony nazwy. Przełączenie na inny dzień następuje po kliknięciu na jego nazwę.

W celu jednoczesnego wprowadzania wartości dla kilku dni, należy kliknąć znaczniki przy nazwach, dla których wartości będą ustawiane. Wartości można ustawiać bezpośrednio na wykresie posługując się myszką lub wpisywać ręcznie w oknie wartości, które otwiera się po kliknięciu na wybraną godzinę.



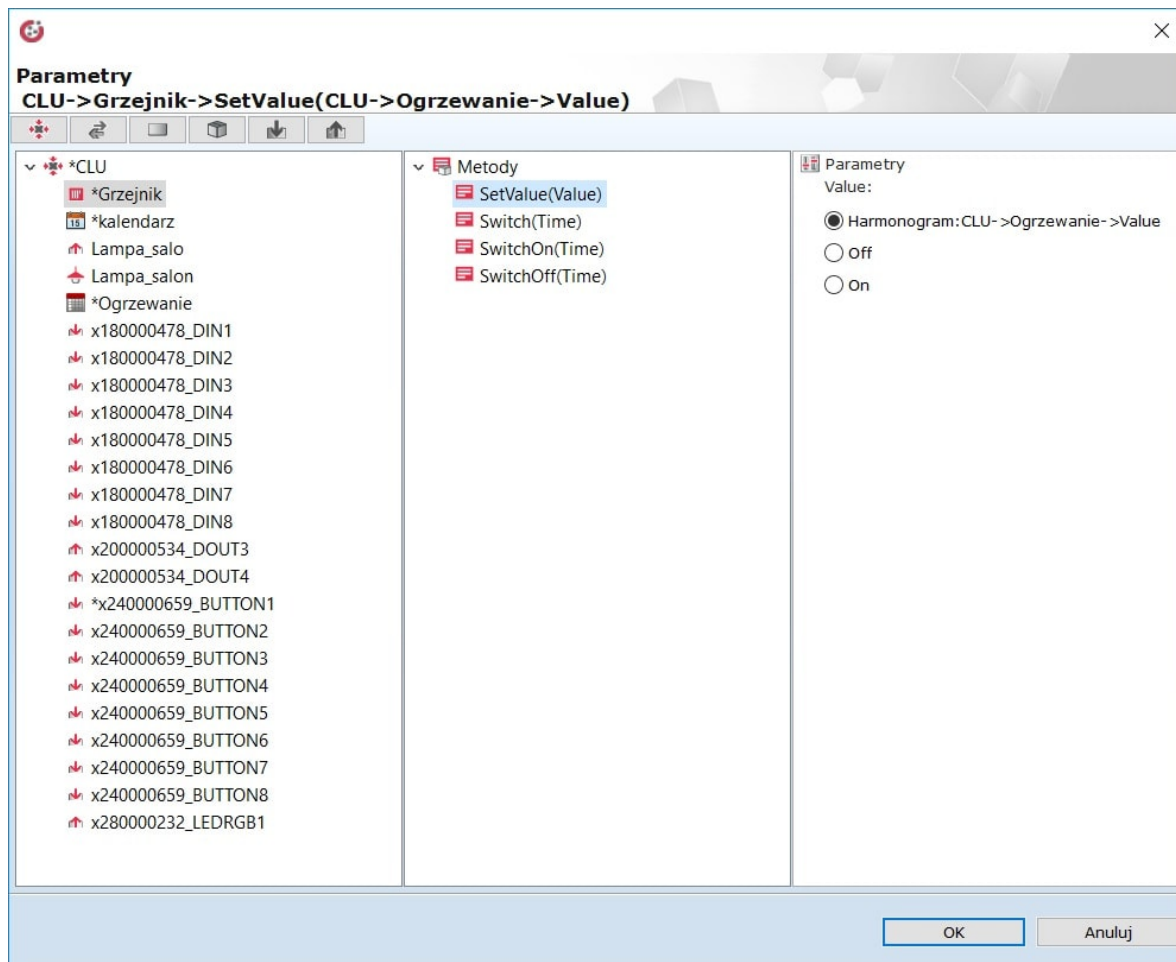
The image shows a dialog box titled "Ustaw wartości" (Set values). It contains a table with four rows, each representing a 15-minute interval. The values are entered into text input fields. At the bottom, there are two buttons: "OK" and "Anuluj" (Cancel).

Time Interval	Value
9:00 - 9:14	0.95
9:15 - 9:29	0.93
9:30 - 9:44	0.90
9:45 - 9:59	0.89

C. Ustawianie wartości wyjścia za pomocą harmonogramu

Zmiana wartości w ustawionym harmonogramie wywołuje zdarzenie `OnHarmonogram`.

By wartości ustawione w harmonogramie zostały przepisane do wartości wybranego wyjścia, do zdarzenia `onHarmonogram` trzeba dodać metodę `setValue` dla wybranego wyjścia, a jako parametr tej metody należy wybrać `Harmonogram`.



Co 15 minut wartość (`value`) dla tego wyjścia będzie ustawiana zgodnie z wartością zapisaną w harmonogramie.

UWAGA! Należy pamiętać, aby zakres wartości ustawionych w harmonogramie odpowiadał zakresowi, w jakim wybrane wyjście może być sterowane. Zakres wartości harmonogramu można zmienić metodami `SetMax` i `SetMin` oraz poprzez zmianę cechy wbudowanych `Min` i `Max`.

D. Parametry konfiguracyjne obiektu Harmonogram

CECHY

Nazwa	Opis
<code>Data</code>	Ciąg znaków definiujący harmonogram zmian wartości
<code>State</code>	Stan działania harmonogramu: 1 (harmonogram aktywny) lub 0 (nieaktywny)
<code>value</code>	Wartość wyjściowa, zmieniana co 15 minut zgodnie z rozkładem
<code>Min</code>	Minimalna wartość dla ustawiania zakresu wartości interfejsu graficznego
<code>Max</code>	Maksymalna wartość dla ustawiania zakresu wartości interfejsu graficznego

METODY

Nazwa	Opis
Start	Przełączenie w stan aktywny (state =1)
Stop	Przełączenie w stan zatrzymania (state =0)
SetData	Ustawienie tygodniowego harmonogramu

ZDARZENIA

Nazwa	Opis
OnHarmonogram	Zdarzenie wywoływane w momencie zmiany wartości cechy value
OnStart	Zdarzenie wywoływane przy wznowieniu pracy
OnStop	Zdarzenie wywoływane przy zatrzymaniu pracy

4. Regulator PID

CLU umożliwia utworzenie do 64 regulatorów PID (proporcjonalno-całkująco-różniczkujących) służących do utrzymywania zadanej wartości wyjściowej na stałym poziomie w zależności od wartości wejściowej.

Najpopularniejszym sposobem wykorzystania regulatora PID jest regulacja temperatury na podstawie informacji odczytywanej z czujnika temperatury.

UWAGA! Regulator PID pracujący w trybie *AUTO* po rozpoczęciu pracy (po pierwszym uruchomieniu lub po resecie CLU) przeprowadza procedurę kalibracji obiektu, podczas której temperatura sterowanego obiektu może zostać podniesiona od kilku do kilkunastu procent powyżej temperatury zadanej. W związku z tym - nie zaleca się stosowania regulatorów PID dla obiektów o dużej bezwładności cieplnej, np. do ogrzewania lub kontrolowania temp. wody w akwariach.

A. Tworzenie regulatora

W celu utworzenia regulatora PID należy zaznaczyć CLU, w ramach którego ma się znaleźć, a następnie z menu górnego uruchomić Dodaj obiekt CLU.



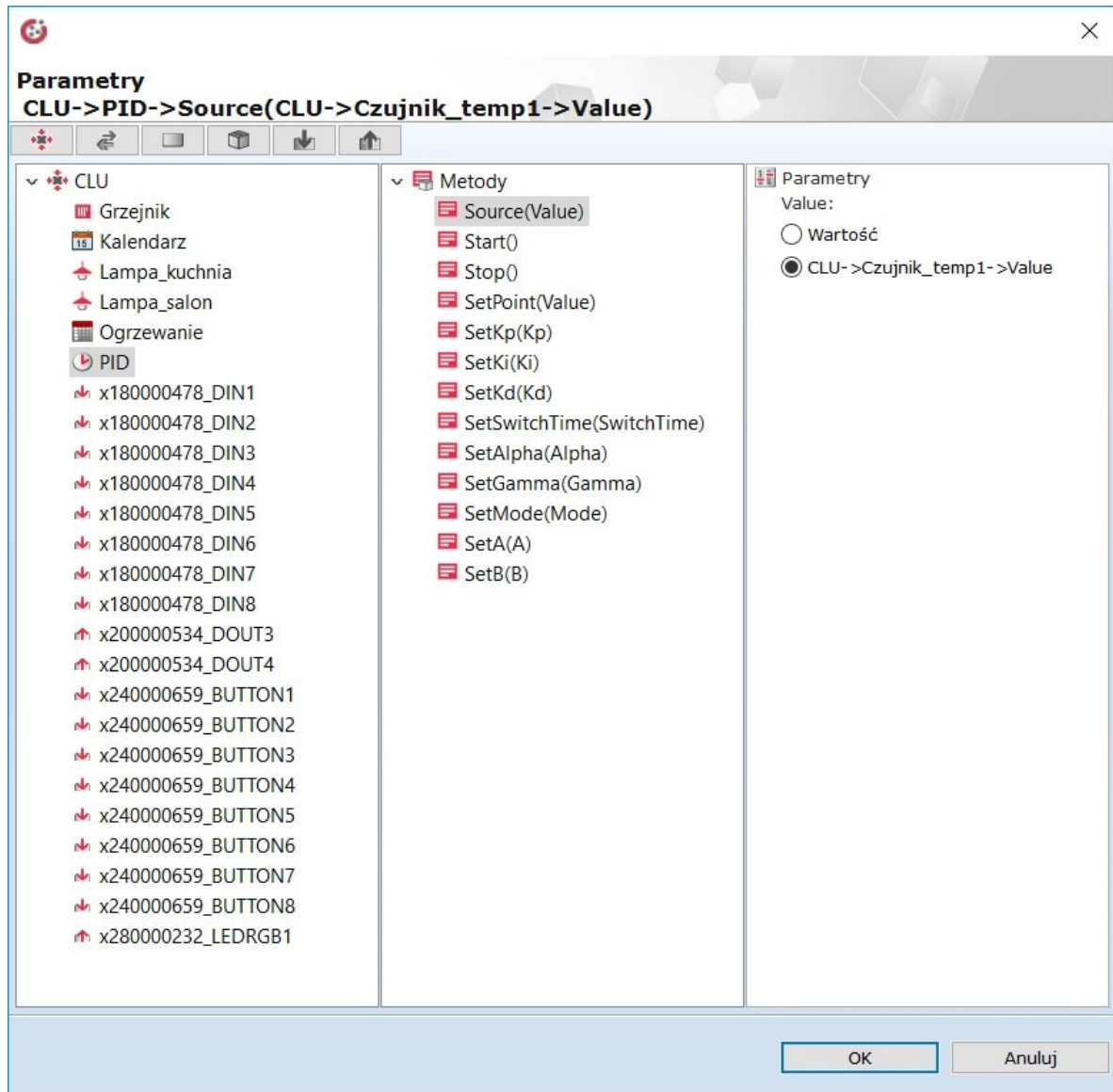
W otwartym oknie konieczne jest odnalezienie i zaznaczenie obiektu PIDcontroler, a następnie nadanie mu nazwy. Na ekranie pojawi się okno właściwości nowo utworzonego regulatora, w którym znajdują się trzy zakładki:

- **Sterowanie** – zawiera metody regulatora;
- **Zdarzenia** – zawiera zdarzenia regulatora;
- **Cechy wbudowane** – zawiera listę cech regulatora.

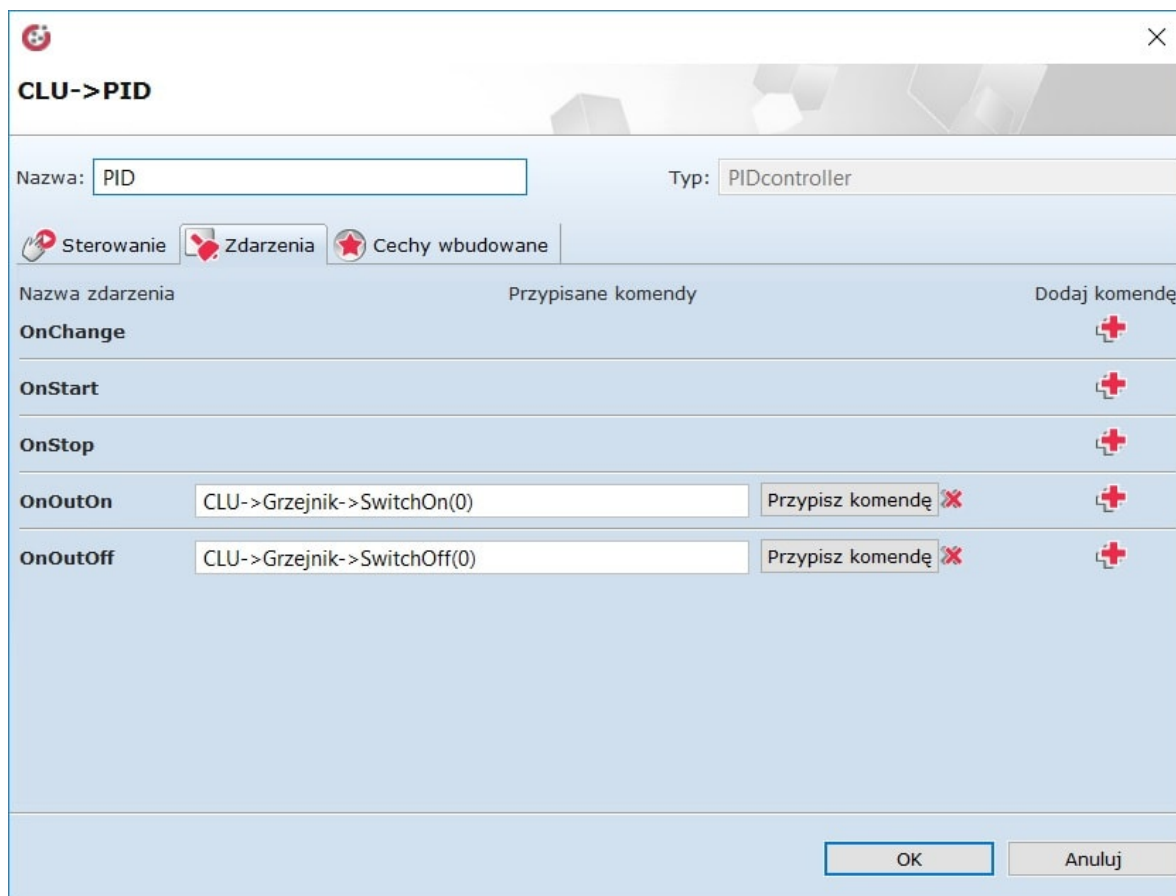
B. Sterowanie za pomocą regulatora

Sterowanie wartościami wyjściowymi z wykorzystaniem regulatora będzie możliwe, gdy zostanie poprawnie powiązany z obiektami wejściowym i wyjściowym. W tym celu kolejno:

- Do metody Source przypisać wartość źródłową, np. cechę `value` czujnika temperatury (w czujniku temperatury do zdarzenia `onChange` należy wybrać regulator PID, a do metody `source`, jako parametr, przypisać wartość z czujnika temperatury).

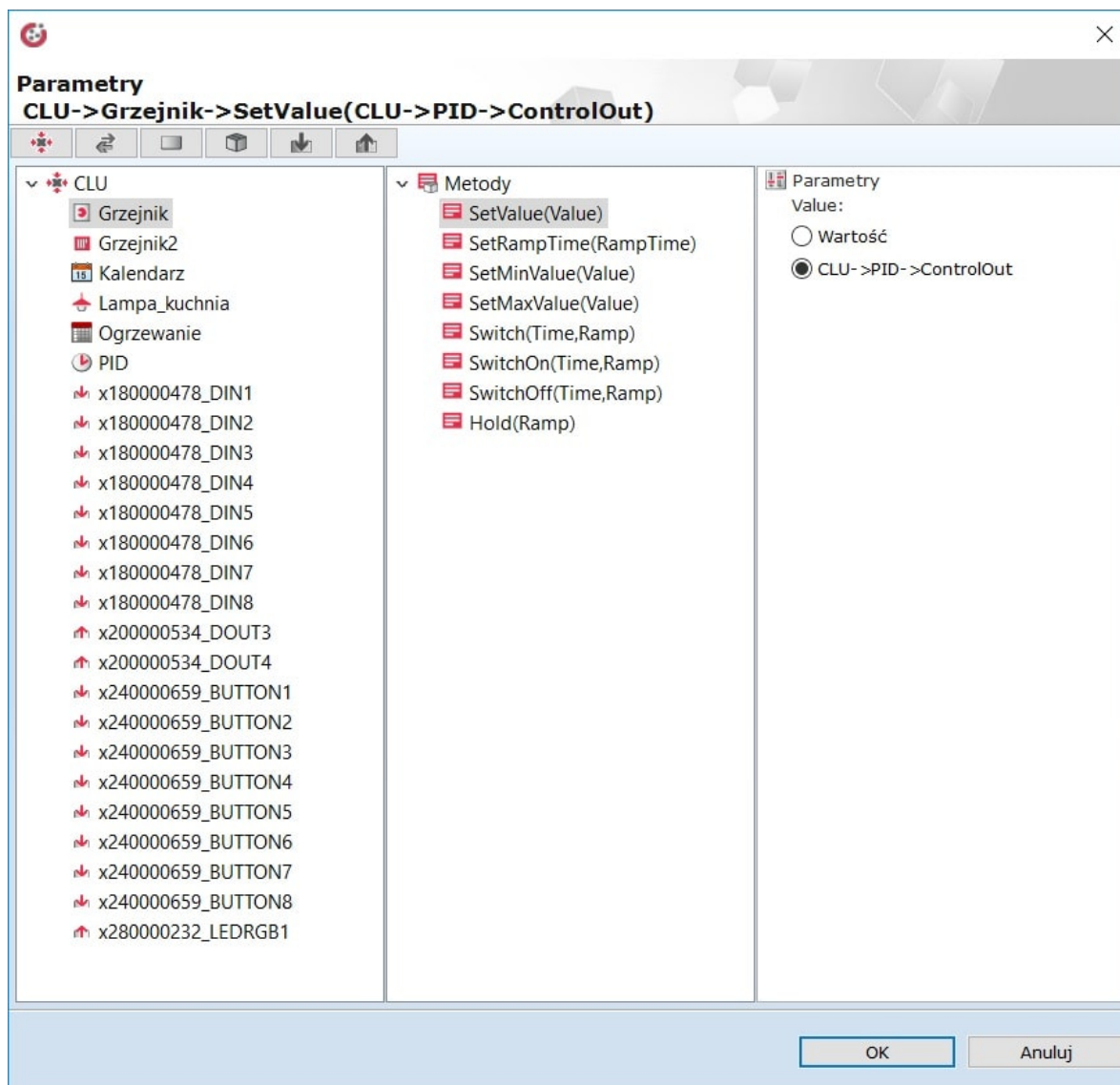


- Należy powiązać moduł wyjściowy z odpowiednimi zdarzeniami obiektu PID.
By tego dokonać, do zdarzeń `onouton` i `onoutoff` przypisać metody sterujące dla obiektu.
W przypadku sterowania temperaturą, należy do zdarzenia `onouton` w regulatorze przypisać metodę `switchon` wyjścia, z którego sterowane jest źródło ciepła (do którego podpięty jest grzejnik, piec, zawór sterujący grzejnikiem), natomiast do zdarzenia `onoutoff` należy przypisać metodę `switchoff` tego wyjścia.



Alternatywnie - jeśli interfejs modułu wyjściowego na to pozwala, można wykorzystać jedynie zdarzenie `onChange`.

By to zrobić, należy do zdarzenia `onChange` w regulatorze przypisać metodę `setValue (value)` wyjścia sterującego źródłem ciepła (dane wyjście musi posiadać taką metodę), a następnie jako parametr wskazać cechę `controlout` regulatora PID.



W takiej konfiguracji, gdy wartość wyjścia sterującego w regulatorze ulegnie zmianie, zostanie ona przepisana do wartości wyjścia.

C. Tryby pracy

Regulator ma dwa możliwe tryby pracy:

- **Sterowanie automatyczne (Auto)** Sterowanie w tym trybie oparte jest na algorytmie automatycznego strojenia, w którym wszystkie istotne parametry dostrajane są automatycznie na podstawie odczytywanych danych.
- **Sterowanie ręczne (Norma1)** Użytkownik w tym trybie ma możliwość ustawienia wszystkich istotnych parametrów wykorzystywanych do sterowania w regulatorze PID z ręcznym doбором nastaw (parametry Kp, Ki, Kd). Tryb **Norma1** przeznaczony jest dla zaawansowanych użytkowników znających zasady strojenia regulatorów PID.

Chcąc ustawić regulator w danym trybie pracy, zmień wartość cechy **Mode** na:

- **Norma1** – dla trybu ręcznego;
- **Auto** – dla trybu automatycznego.

W zależności od wybranego trybu pracy zmienia się możliwość ustawiania wartości poszczególnych cech – np. parametry A i B wykorzystywane są tylko dla algorytmu **Auto**, a parametry Kp, Ki oraz Kd – tylko dla trybu **Norma1**.

UWAGA! Parametry A i B nie mogą być zmieniane w trakcie trwania sterowania, gdyż są one na bieżąco aktualizowane przez algorytm.

D. Zasada działania regulatora PID

Regulator steruje cechą `ControlOut` poprzez ustawianie jej wartości na 0 lub 1 z częstotliwością ustawiania cechy `SwitchTime`, poprzez zmianę wypełnienia impulsu.

Przed rozpoczęciem sterowania, regulator najpierw przeprowadza procedurę szacowania bezwładności obiektu sterowanego i na tej podstawie wyznacza dopuszczalne granice wartości `SwitchTime`. Po zakończeniu tego etapu, wartość cechy `SwitchTime` ustawiana jest automatycznie w połowie wyznaczonego zakresu.

UWAGA! W przypadku, gdy regulacja prowadzona jest w sposób automatyczny, ręczna zmiana wartości `SwitchTime` nie jest możliwa.

E. Parametry konfiguracyjne obiektu Regulator PID

CECHY

Nazwa	Opis
<code>ControlOut</code>	Wartość wyjścia sterującego (binarna, przełączana w cyklu określonym przez <code>SwitchTime</code>)
<code>State</code>	Stan działania regulatora: 1 (harmonogram aktywny) lub 0 (nieaktywny)
<code>SetPoint</code>	Wejście regulatora – wartość docelowa
<code>Kp</code>	Wzmocnienie członu proporcjonalnego regulatora PID
<code>Ki</code>	Wzmocnienie członu całkującego regulatora PID
<code>Kd</code>	Wzmocnienie członu różniczkującego regulatora PID
<code>SwitchTime</code> *	Czas przełączania
<code>Alpha</code>	Parametr α w algorytmie Kaczmarza (zabezpieczenie przed zerowaniem mianownika)
<code>Gamma</code>	Parametr γ w algorytmie Kaczmarza (dynamika zmian oszacowania a i b)
<code>Mode</code>	Tryb pracy regulatora: 1 – „ręczny” PID lub 2 – automatyczny algorytm Kaczmarza
<code>A</code> *	Parametr a w algorytmie Kaczmarza
<code>B</code> *	Parametr b w algorytmie Kaczmarza

- Ustawianie tych parametrów nie jest możliwe we wszystkich stanach pracy regulatora.

METODY

Nazwa	Opis
<code>Source</code>	Podanie nowej wartości wejścia dla sterownika (pętla sprzężenia zwrotnego)
<code>Start</code>	Przełączenie w stan aktywny (<code>state =1</code>)
<code>Stop</code>	Przełączenie w stan zatrzymania (<code>state =0</code>)
<code>SetPoint</code>	Ustawienie docelowej wartości regulatora
<code>SetKp</code>	Ustawienie wartości wzmacnienia członu proporcjonalnego
<code>SetKi</code>	Ustawienie wartości wzmacnienia członu całkującego
<code>SetKd</code>	Ustawienie wartości wzmacnienia członu różniczkującego
<code>SetSwitchTime</code>	Ustawienie czasu przełączania
<code>SetAlpha</code>	Ustawienie parametru Alpha w algorytmie Kaczmarza, zabezpieczającego przed zerowaniem mianownika
<code>SetGamma</code>	Ustawienie parametru Gamma w algorytmie Kaczmarza
<code>SetMode</code>	Ustawienie trybu pracy regulatora – ręczny PID (Normal PID) lub automatyczny algorytm Kaczmarza (Auto-Kaczmarz)
<code>SetA</code>	Ustawienie parametru a w algorytmie Kaczmarza
<code>SetB</code>	Ustawienie parametru b w algorytmie Kaczmarza

ZDARZENIA

Nazwa	Opis
<code>OnChange</code>	Zdarzenie wywoływane w momencie zmiany wartości cechy <code>ControlOut</code>
<code>OnStart</code>	Zdarzenie wywoływane w momencie wznowienia pracy
<code>OnStop</code>	Zdarzenie wywoływane przy zablokowaniu pracy
<code>OnOutOn</code>	Zdarzenie wywoływane w momencie przełączenia wartości cechy <code>ControlOut</code> na 1
<code>OnOutOff</code>	Zdarzenie wywoływane w momencie przełączenia wartości cechy <code>ControlOut</code> na 0

5. Termostat

Termostat to obiekt wirtualny, który służy do stworzenia konfiguracji sterującej ogrzewaniem lub chłodzeniem w zależności od podanego czujnika temperatury i wprowadzonego harmonogramu ogrzewania bądź chłodzenia w rozkładzie tygodniowym. Wartości temperatury ustawiane są przy pomocy interfejsu graficznego dla każdego dnia i godziny z rozdzielczością 15-minutową, 30-minutową lub godzinną.

W jednym CLU można utworzyć do 64 termostatów.

UWAGA! Po utworzeniu termostatu (po wysłaniu nowej konfiguracji do CLU) staje się on automatycznie aktywny. Chcąc zatrzymać jego pracę, należy wywołać metodę **Stop**.

A. Tworzenie termostatu

W celu utworzenia termostatu należy zaznaczyć CLU, w ramach którego ma zostać umieszczony, a następnie z menu górnego uruchomić **Dodaj obiekt CLU**.



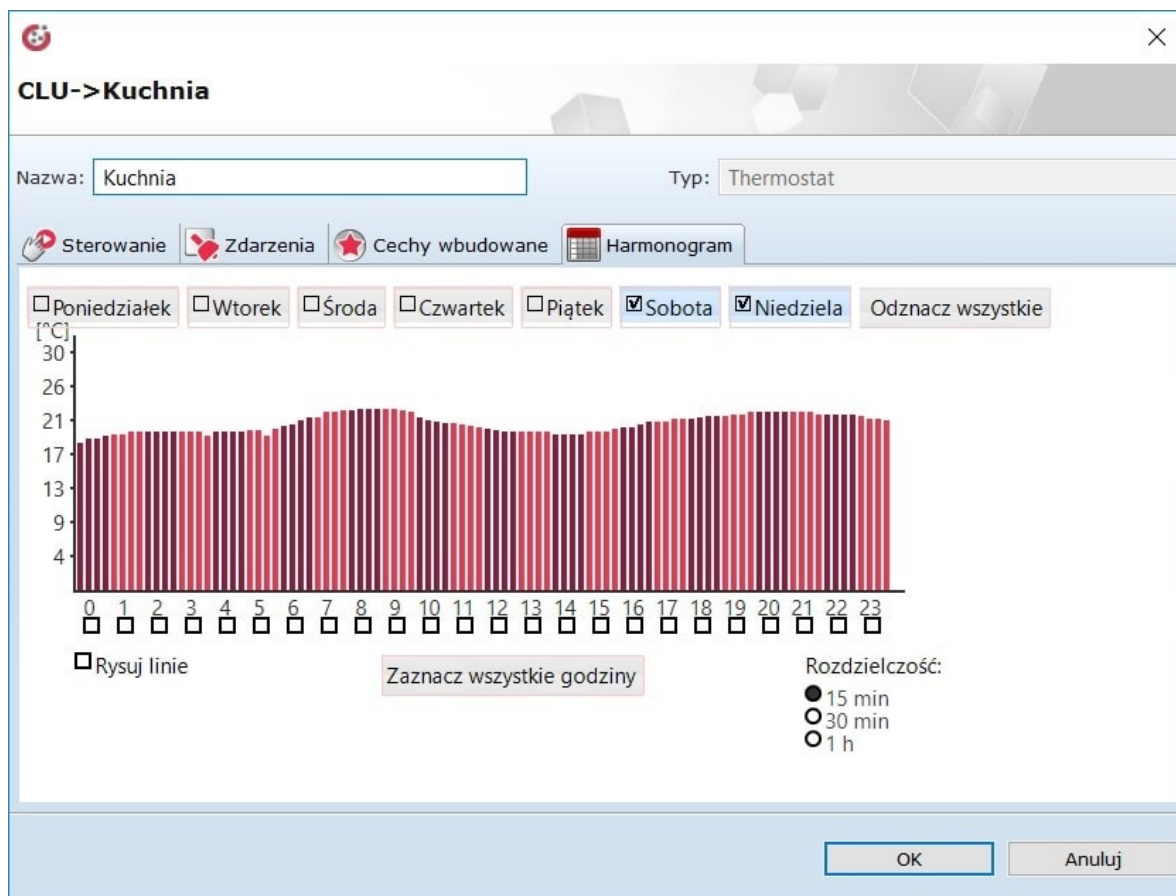
W otwartym oknie wyboru należy odszukać i wybrać **Termostat**. Po wpisaniu nazwy, wprowadzeniu źródła (którym powinien być czujnik temperatury odpowiedzialny za daną strefę grzewczą) oraz wyborze odbiornika (będącego wyjściem, do którego podłączone jest urządzenie odpowiedzialne za daną strefę grzewczą – np. głowica kaloryfera, ogrzewanie podłogowe) dla utworzonego obiektu na ekranie otworzy się okno właściwości harmonogramu.

W oknie tym znajdują się cztery zakładki:

- **Sterowanie** – zawiera metody harmonogramu;
- **Zdarzenia** – zawiera zdarzenia harmonogramu;
- **Cechy wbudowane** – zawiera listę cech harmonogramu;
- **Harmonogram** – zawiera interfejs graficzny, umożliwiający proste formułowanie wartości dla całego zakresu harmonogramu.

B. Formułowanie wartości dla termostatu

W zakładce *Harmonogram* (w oknie właściwości) znajduje się interfejs graficzny, dzięki któremu istnieje możliwość zadawania wartości.



Harmonogram pozwala na wpisanie wartości dla 7 dni (w zakresie jednego tygodnia) z rozdzielczością 15-minutową. Można ustawić wartości dla każdego dnia z osobna lub dla kilku dni jednocześnie. Dzień, dla którego aktualnie wpisywane są wartości, wyróżniony jest czarnym znacznikiem z lewej strony nazwy. Przełączenie na inny dzień następuje po kliknięciu na jego nazwę.

W celu jednoczesnego wprowadzania wartości dla kilku dni, należy kliknąć znaczniki przy nazwach, dla których wartości będą ustawiane. Wartości można ustawiać bezpośrednio na wykresie posługując się myszką lub wpisywać ręcznie w oknie wartości, które otwiera się po kliknięciu na wybraną godzinę.

The 'Ustaw wartości' dialog box displays a list of time intervals and their corresponding temperature values. The values are: 20.7 for 16:00 - 16:14, 20.7 for 16:15 - 16:29, 21.1 for 16:30 - 16:44, and 21.4 for 16:45 - 16:59. The dialog includes 'OK' and 'Anuluj' buttons at the bottom.

Time Interval	Temperature (°C)
16:00 - 16:14	20.7
16:15 - 16:29	20.7
16:30 - 16:44	21.1
16:45 - 16:59	21.4

Termostat reaguje na harmonogram, gdy znajduje się w trybie `Auto`. Wybór trybu pracy odbywa się przy pomocy aplikacji lub przez metody obiektu.

C. Parametry konfiguracyjne obiektu Termostat

CECHY

Nazwa	Opis
<code>Source</code>	Wejście termostatu, powiązanie z czujnikiem temperatury
<code>Control</code>	Wyjście termostatu, powiązanie z elementem wykonującym
<code>OutputType</code>	Określenie typu wyjścia (-1 – autodetekcja, 0 – wyjście cyfrowe, 1 – wyjście analogowe)
<code>PointValue</code>	Wartość temperatury zadana ręcznie
<code>HolidayModeValue</code>	Wartość temperatury dla trybu wakacyjnego
<code>Hysteresis</code>	Wartość histerezy – określenie granic załączenia i wyłączenia termostatu
<code>State</code>	Stan pracy (1 – termostat aktywny, 0 – nieaktywny)
<code>ControlDirection</code>	Kierunek pracy (0 – tryb normalny (grzenie), 1 – tryb odwrotny (chłodzenie))
<code>Mode</code>	Tryb pracy (0 – tryb ręczny (wykorzystanie <code>PointValue</code>), 1 – tryb wakacyjny (<code>HolidayModeValue</code>), 2 – tryb automatyczny (wartość <code>AutoMode</code> z Harmonogramu), 3 – tryb podgrzewania (wartość <code>HeatUp</code>))
<code>Data</code>	Ciąg znaków definiujący harmonogram zmian wartości
<code>Min</code>	Dolna wartość zakresu wbudowanego harmonogramu
<code>Max</code>	Górna wartość zakresu wbudowanego harmonogramu
<code>TargetTemp</code>	Aktualna wartość temperatury docelowej
<code>ControlOutValue</code>	Wartość przypisana do wyjścia sterującego ogrzewaniem

METODY

Nazwa	Opis
Start	Przełączenie termostatu w stan aktywny (State =1)
Stop	Przełączenie termostatu w stan nieaktywny (State =0)
IncreaseDegree	Zwiększenie PointValue o 1°C
DecreaseDegree	Zmniejszenie PointValue o 1°C
HeatUp	Zwiększenie PointValue o zadaną wartość w określonym czasie
HolidayModeStart	Uruchomienie trybu wakacyjnego
HolidayModeStop	Zatrzymanie trybu wakacyjnego
AutoModeStart	Uruchomienie trybu AutoMode (pobieranie temperatury z harmonogramu)
AutoModeStop	Zatrzymanie trybu AutoMode
SetData	Ustawienie tygodniowego harmonogramu
SetOutputType	Ustawienie typu wyjścia (Auto – autodetekcja, Digital – wyjście cyfrowe, Analog – wyjście analogowe)
SetPointValue	Ustawienie ręcznie zadanej temperatury
SetHolidayModeValue	Ustawienie wartości temperatury dla trybu wakacyjnego
SetHysteresis	Ustawienie wartości histerezy
SetControlDirection	Ustawienie kierunku pracy (0 – tryb normalny (grzenie), 1 – tryb odwrotny (chłodzenie))

ZDARZENIA

Nazwa	Opis
OnChange	Zdarzenie generowane przy zmianie wartości cechy PointValue
OnStart	Zdarzenie generowane przy wznowieniu pracy termostatu
OnStop	Zdarzenie generowane przy zatrzymaniu pracy termostatu
OnOutOn	Zdarzenie wywoływane przy wystawieniu na wyjściu OutValue wartości większej od zera
OnOutOff	Zdarzenie wywoływane przy wystawieniu na wyjściu OutValue wartości mniejszej od zera
OnHolidayModeOn	Zdarzenie generowane przy uruchomieniu trybu wakacyjnego
OnHolidayModeOff	Zdarzenie generowane przy wyłączeniu trybu wakacyjnego

X. Pomiar mediów

1. Uruchomienie pomiaru mediów po stronie Object Managera

Object Manager pozwala na przeprowadzenie pomiaru mediów, który umożliwi szacunkową prezentację zużytej energii (bazując na czasie załączenia urządzenia oraz podanej w konfiguracji mocy odbiornika). Konfiguracja pomiaru mediów odbywa się w OM i należy ją uruchomić dla każdego wejścia i wyjścia z osobna – tak, by CLU zbierało dane dotyczące zużycia energii. Pomiar mediów jest rejestrowany co 15 minut, rozpoczynając odliczanie od pełnej godziny - bazując na zegarze CLU (*cecha CLU->TIME*).

UWAGA! Pomiar mediów dostępny jest dla Object Managera w wersji 1.2.0.180202 i wyższej oraz dla CLU z firmware 04.07.29-1802 i wyższym.

Pomiar mediów można uruchomić dla modułów:

- Wejściowych (Digital IN) – w trybie ciągłym (zliczającym czas pracy) lub impulsowym (zliczającym impulsy pojawiające się na wejściu binarnym):

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
Intertion	0	0	ms	[0-2000]
HoldDelay	500	500	ms	[0-5000]
HoldInterval	100	50	ms	[0-2000]
Value	0		bool	[0-1]
StatisticState	0		number	0,1,2
Load	0		number	

- Wyjściowych (Relay, Led RGB, Dimmer) – w trybie ciągłym (zliczającym czas pracy):

CLU_220000541->x200000534_DOUT1

Nazwa: Źródło/Odbiornik:

Identyfikacja: 2 Typ:

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
Value	0	Off	bool	0,1
StatisticState	0	Ciągly	number	0,1
Load	0	<input type="text" value=""/>	number	

Auto odświeżanie

UWAGA! Pomiar mediów wyżej wymienionych modułów dotyczy modułów na szynę DIN oraz podtynkowych Tf-bus! Ustawienie pomiaru nie jest dostępne dla modułów Z-Wave!

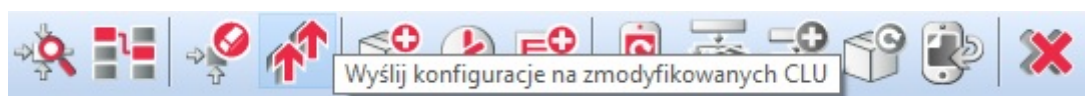
A. Stworzenie konfiguracji

Chcąc utworzyć konfigurację należy:

- Dwukrotnie kliknąć myszką na wybrany moduł z listy modułów w widoku głównym programu (dotyczy to wyżej wypisanych modułów umożliwiających obsługę pomiaru mediów);
- Przejść do zakładki *Cechy wbudowane*;
- Zmienić wybór cechy `StatisticState` na: `Ciągly` lub `Impulsowy` (w przypadku wejść binarnych modułu Digital In);
- Poniżej pojawi się cecha `Load` – do jej wartości początkowej wpisać pobieraną moc czynną urządzenia podłączonego do wejścia lub wyjścia w watach na godzinę (np. 60) – CLU poda wartość przeliczy impulsowo lub ciągle (mnożąc przez czas w godzinach):

StatisticState	0	Ciągly	number	0,1
Load	0	<input type="text" value="60"/>	number	

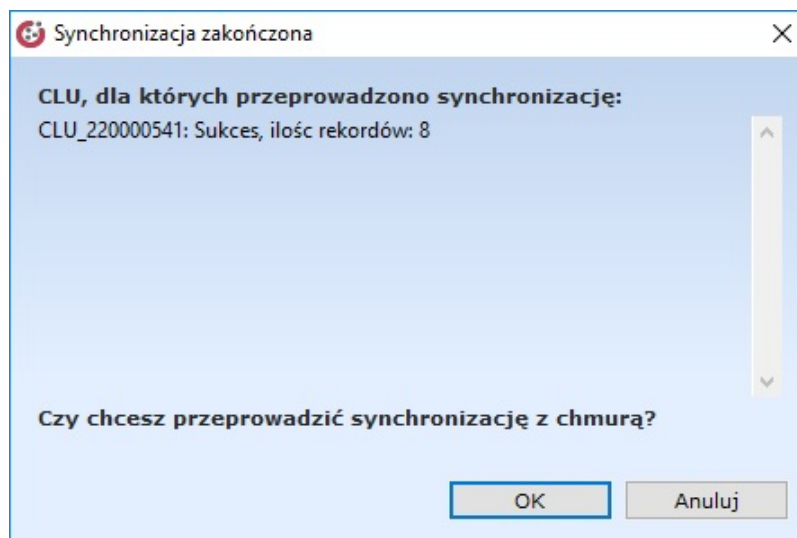
- Zatwierdzić przyciskiem *OK*;
- Dodać ustawienia pomiaru mediów dla kolejnych modułów – powtórzyć kroki 2b-2e;
- Wysłać konfigurację do CLU.



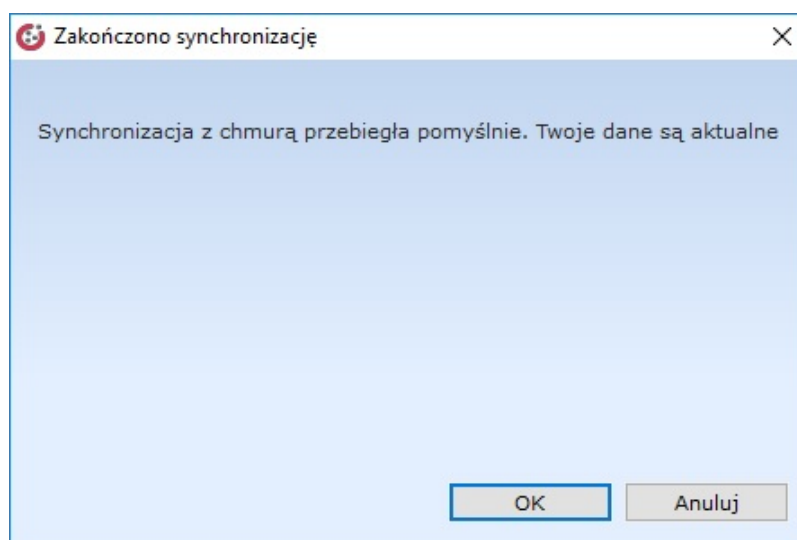
B. Odczyt pomiaru mediów w Object Managerze

W celu odczytania pomiaru mediów w programie Object Manager należy:

- Odczekać przynajmniej do pierwszego planowanego zarejestrowania pomiarów przez CLU (do XX.00 lub XX.15 lub XX.30 lub XX.45 – gdzie XX to godzina);
- Wybrać **Narzędzia->Pobierz plik z pomiarami**;
- Wyświetli się okno z informacją o pobranych rekordach:

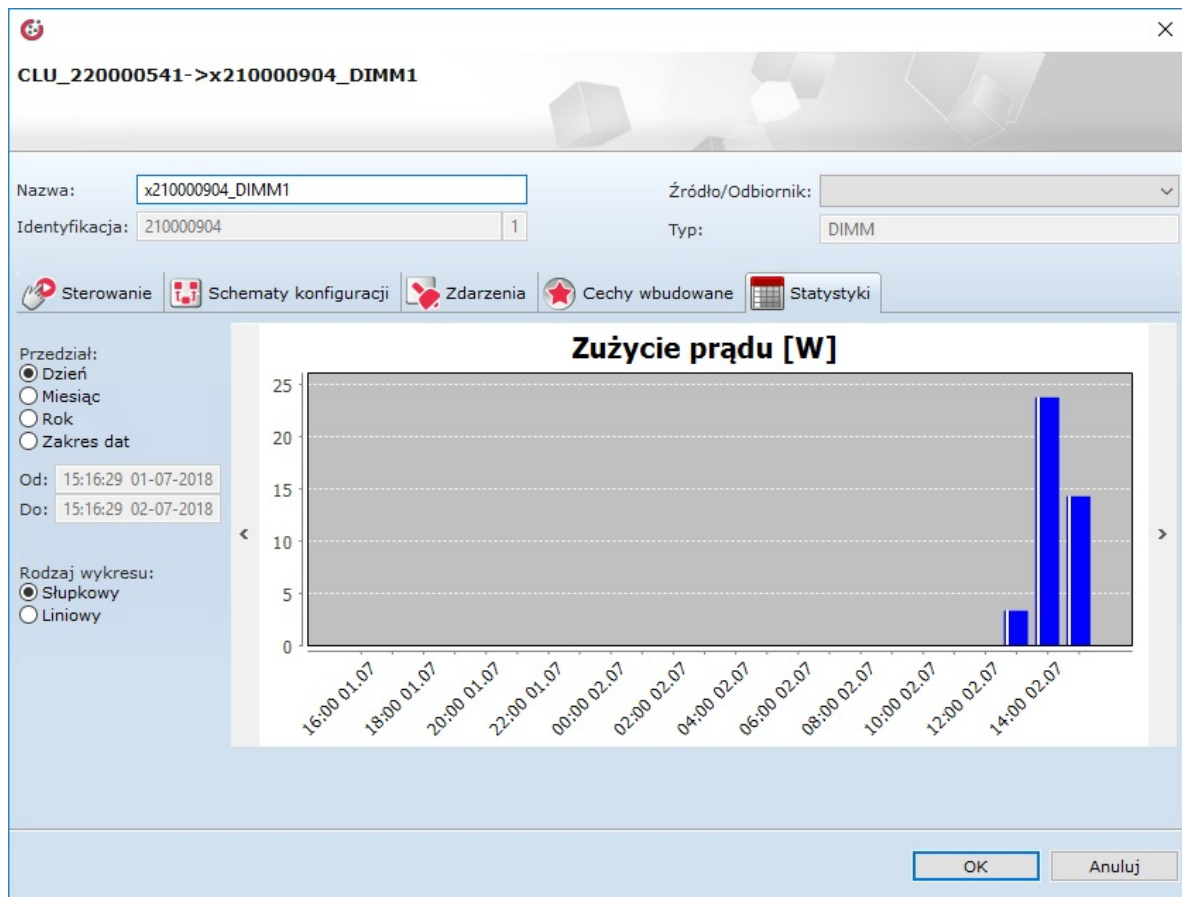


- Kliknąć *OK*;
- Object Manager następnie zsynchronizuje pobrane dane z chmurą;
- Po zakończonej synchronizacji wcisnąć *OK*:



UWAGA! W przypadku błędu synchronizacji prosimy o kontakt z działem Support!

- W celu upewnienia się, czy pomiar mediów został zarejestrowany, należy kliknąć dwukrotnie na wybrany moduł, dla którego uruchomiony został pomiar mediów;
- Następnie należy przejść do zakładki *Statystyki*:
 - Można wybrać rodzaj wyświetlanego wykresu: słupkowy lub liniowy - w obu przypadkach na wykresie pojawia się zsumowana wartość zużytej energii (w watach) dla każdej godziny;
 - Można również wybrać przedział przeglądanego pomiaru mediów: dzień, miesiąc, rok lub samodzielnie wybrać zakres dat – w zależności od wybranego przedziału wyświetlony zostanie odpowiedni wykres.



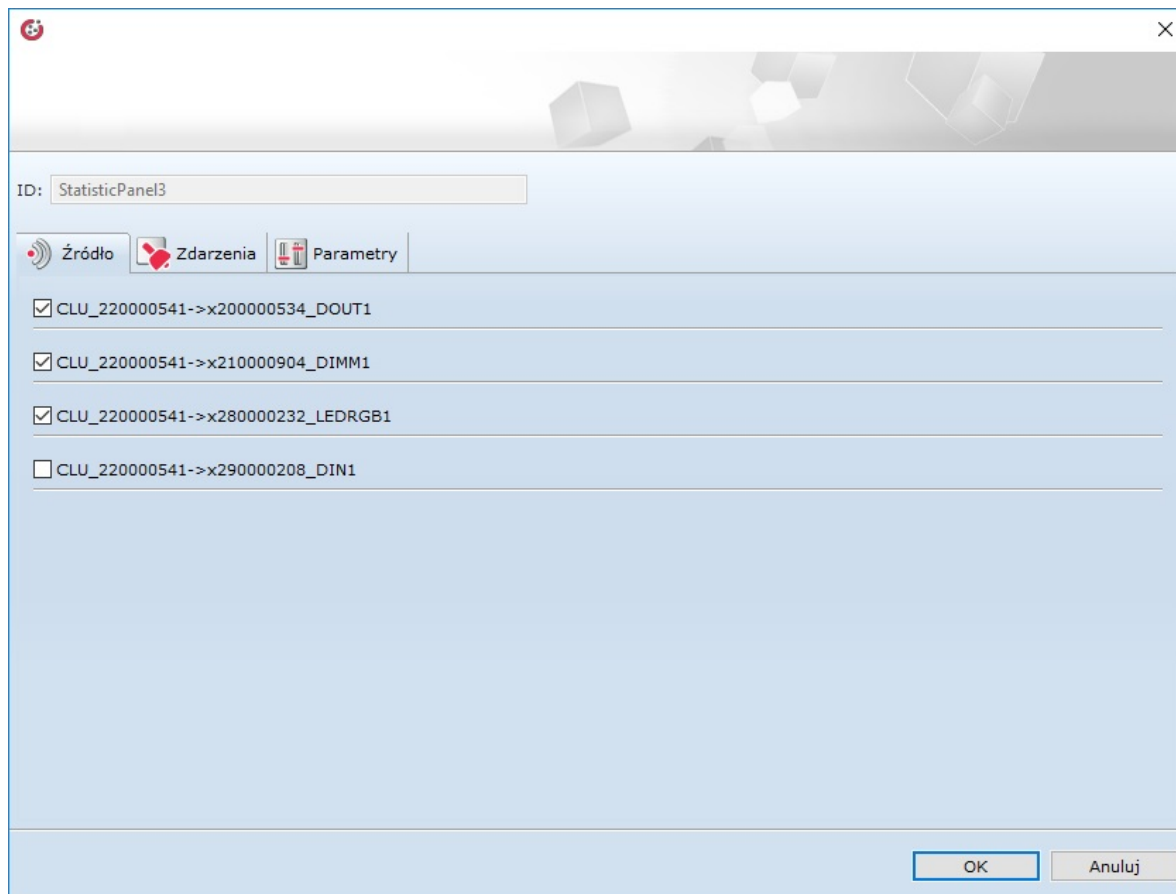
C. Skonfigurowanie pomiaru mediów dla interfejsu aplikacji

Konfiguracja pomiaru mediów dla interfejsu aplikacji musi przebiegać zgodnie z poniższym schematem:

- Dodać nowy interfejs aplikacji:



- Wprowadzić nazwę tworzonej aplikacji;
- Ustawić rozdzielczość, skórkę, dodać przynajmniej jedną stronę, kliknąć OK;
- Z zasobnika paneli przeciągnąć panel *Statystyka* do obszaru edytowalnego interfejsu aplikacji;
- W zakładce *Źródło* zaznaczyć checkboxy dla modułów, których wykresy pomiaru mediów mają być wyświetlane w panelu statystyk w aplikacji;



- Kliknąć OK;
- Wysłać interfejs do urządzenia mobilnego - [patrz pkt VIII.4.7.](#)

2. Używanie pomiaru mediów po stronie aplikacji Home Manager

UWAGA! Pomiar mediów dostępny jest dla Home Manager w wersji 1.1.110 lub wyższej.

By prawidłowo korzystać z pomiaru mediów w aplikacji mobilnej, należy w pierwszej kolejności pobrać pomiary z CLU oraz - jeśli jest to potrzebne - synchronizować pomiary z chmurą.

A. Pobieranie pomiarów:

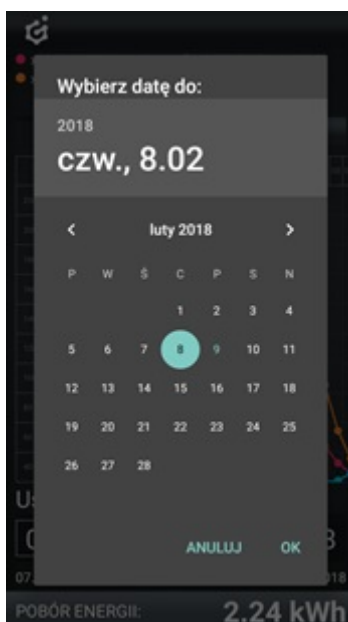
- Należy wejść do ustawień aplikacji z menu głównego (ikona koła zębatego).
- Z listy ustawień wybrać: *Pobierz pomiary z CLU*.
- Po chwili wyświetli się komunikat: *Sukces dla CLU: X, Y* ⁶.
- Uruchomić interfejs aplikacji – pomiary powinny być zaktualizowane i wyświetlone na wykresie.

B. Opcje widoku panelu pomiaru mediów:

- Zmiana wyświetlanych danych konkretnych *wejść/wyjść* – po kliknięciu na wypisane moduły, w górnej belce panelu pomiaru mediów wyświetla się okno dostępnych modułów dodanych do panelu, które domyślnie są zaznaczone – ich odznaczenie powoduje brak pokazywania zmierzonych wartości dla konkretnych *wejść/wyjść*:



- W tym samym oknie, w którym są widoczne moduły, możliwa jest zmiana widoku wykresu - domyślne jest to wykres liniowy, można jednak również wybrać słupkowy, kołowy lub ranking;
- Zmiana zakresu czasowego wyświetlanych przebiegów – można tego dokonać przy pomocy przycisków „dzienny” (sumujący pomiary dla każdej godziny dnia), „miesięczny” (sumujący wartości dla każdego dnia w miesiącu) oraz „roczny” (sumujący pomiary dla każdego miesiąca z osobna);
- Możliwy jest również wybór własnego zakresu czasowego – po kliknięciu w daną godzinę wyświetla się okno wyboru dnia początkowego oraz końcowego:



C. Synchronizacja oraz pobieranie pomiarów:

- Pobieranie pomiarów z CLU, które było dokonywane wcześniej, odbywało się przy połączeniu lokalnym z CLU. By pomiary mogły być wyświetlane podczas dostępu zdalnego, należy zsynchronizować je z chmurą;
- W celu zsynchronizowania pomiarów z chmurą należy wejść do menu głównego aplikacji Home Manager - w ustawienia i na samym dole wybrać: *Synchronizuj pomiary z chmurą*.

XI. Funkcje serwisowe CLU

1. Przywracanie ustawień fabrycznych CLU - *Hard Reset*

Uruchomienie funkcji *Hard Reset CLU* powoduje:

- Usunięcie zapisanej konfiguracji;
- Sformatowanie partycji pamięci flash;
- Usunięcie wszystkich utworzonych obiektów LUA;
- Wyczyszczenie wszystkich danych kontrolera Z-Wave;
- Usunięcie informacji o podłączonych modułach Z-Wave.

W celu przywrócenia ustawień fabrycznych CLU funkcją *Hard Reset* należy wykonać następujące czynności (zgodnie z podaną kolejnością):

- Odłączyć zasilanie od modułu CLU;
- Naciśnąć i przytrzymać przycisk *Link* na module;
- Podłączyć zasilanie do modułu CLU;
- Trzymać wciśnięty przycisk *Link* przez co najmniej 10 sekund – obydwie diody na CLU będą świecić światłem ciągłym;
- Po upływie 10 sekund zwolnić przycisk *Link* – prawidłowe wykonanie resetu zostanie potwierdzone 5-krotnym mrugnięciem obydwu diod.

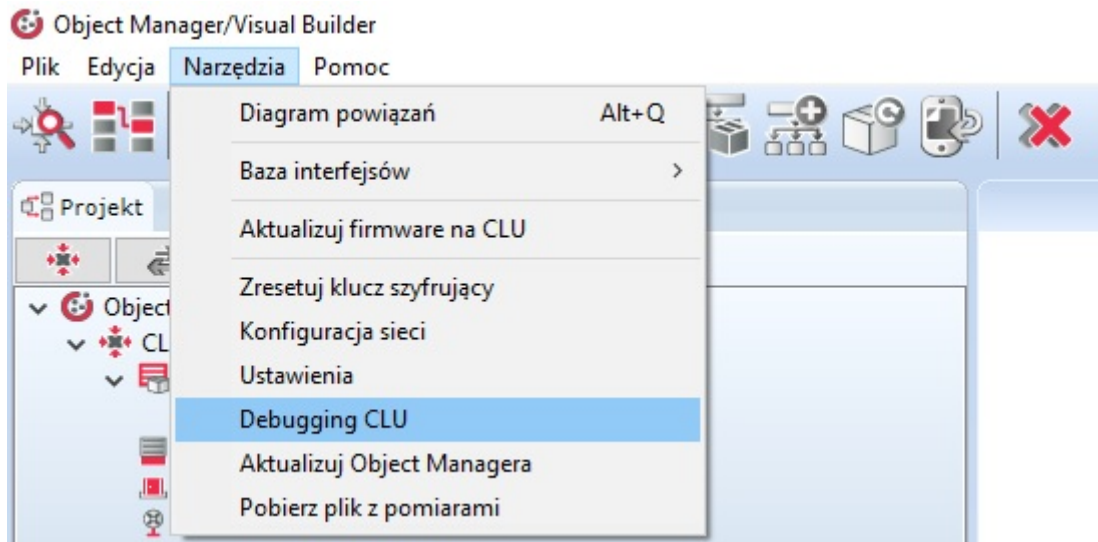
UWAGA! Jeśli przed uruchomieniem funkcji *Hard Reset* do modułu CLU były dodane moduły Z-Wave, to po wykonaniu resetu niezbędne będzie wykonanie procedury usuwania i ponownego dodawania każdego modułu Z-Wave!

2. Autodiagnostyka systemu - *Debugging CLU*

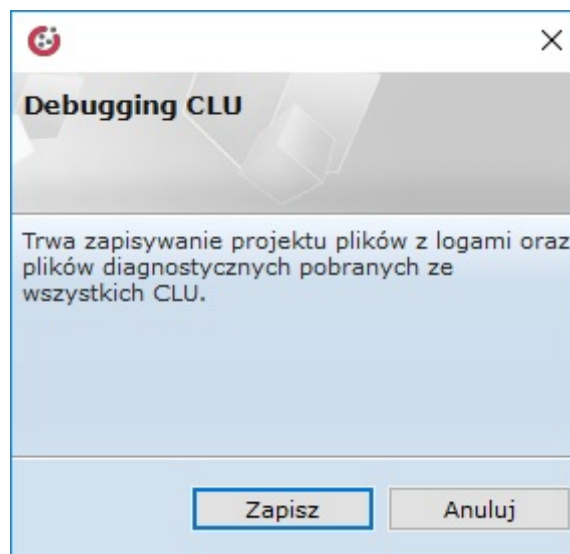
Debugging CLU wykorzystywany jest do diagnostyki jednostki centralnej CLU oraz do szybkiego odnajdywania ewentualnych problemów w stworzonym projekcie.

W celu przeprowadzenia autodiagnostyki systemu należy:

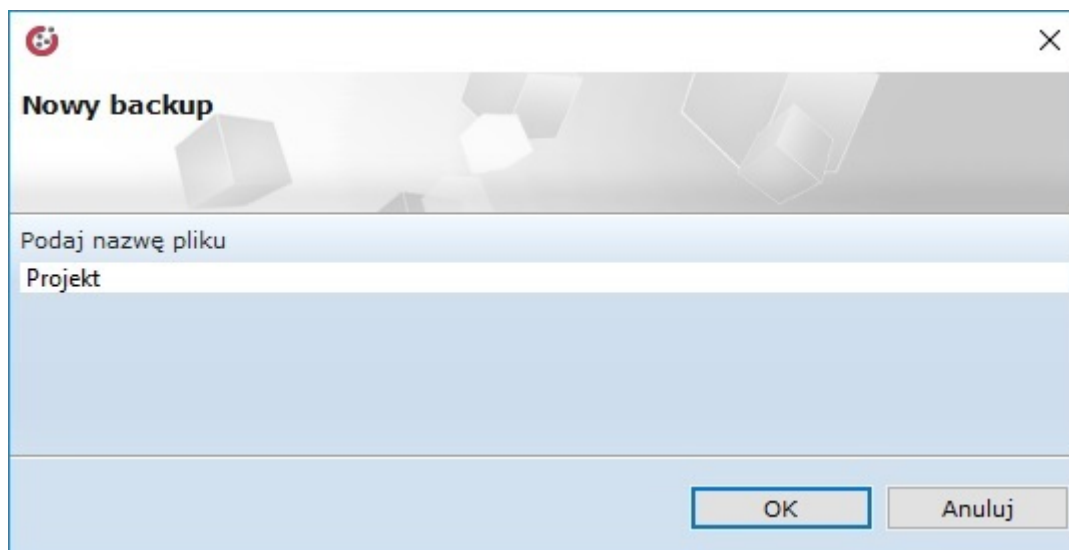
- Otworzyć projekt w Object Managerze;
- Z paska zadań wybrać *Narzędzia*, a z rozwiniętej listy funkcję *Debugging CLU*:



- W otwartym oknie wybrać opcję *Zapisz*:



- Określić miejsce zapisu paczki plików oraz nadać nazwę kopii zapasowej:



- Następnie w wybranej lokalizacji pojawi się folder w formacie *.zip*, którego zawartość będzie prezentowała się następująco:

Name	Type	Compressed size	Password ...	Size
CLU_0d1cf11d	File folder			
interfaces	Compressed (zipped) Fol...	255 KB	No	296 KB
om	Text Document	77 KB	No	1 518 KB
Projekt	Compressed (zipped) Fol...	22 KB	No	23 KB
Projekt_backup_18-06-20_09-49.omp	OMP File	23 KB	No	23 KB

- Utworzony w ten sposób folder zawiera:
 - pliki konfiguracyjne CLU;
 - aktualną bazę interfejsów wykorzystaną w projekcie;
 - plik z wyszczególnionymi logami aplikacji;
 - informacje na temat projektu oraz jego kopię zapasową.

XII. SMART PANEL

1. Wyposażenie Smart Panelu

Smart Panel składa się z:

- Wyświetlacza OLED;
- Czterech przycisków dotykowych;
- Sensora rozpoznającego cztery gesty;
- Sensora zbliżenia/obecności;
- Sensora temperatury;
- Sensora natężenia światła;
- Buzzera.

2. Podłączenie Smart Panelu do CLU

UWAGA!

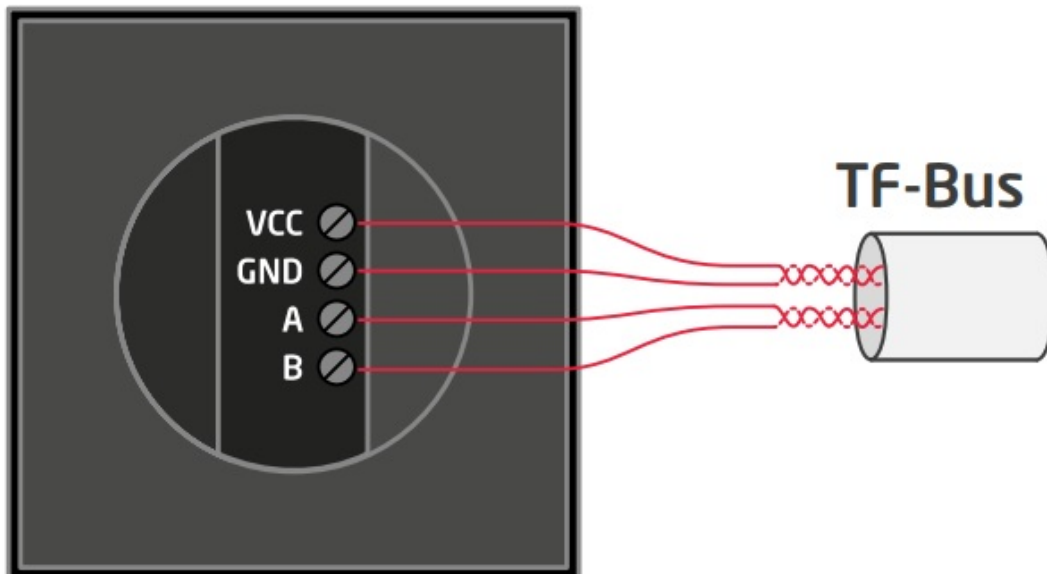
Smart Panel dostępny jest dla Object Managera w wersji 1.2.0.180202 i wyższej oraz dla CLU z firmware 04.07.29-1802 i wyższym.

UWAGA!

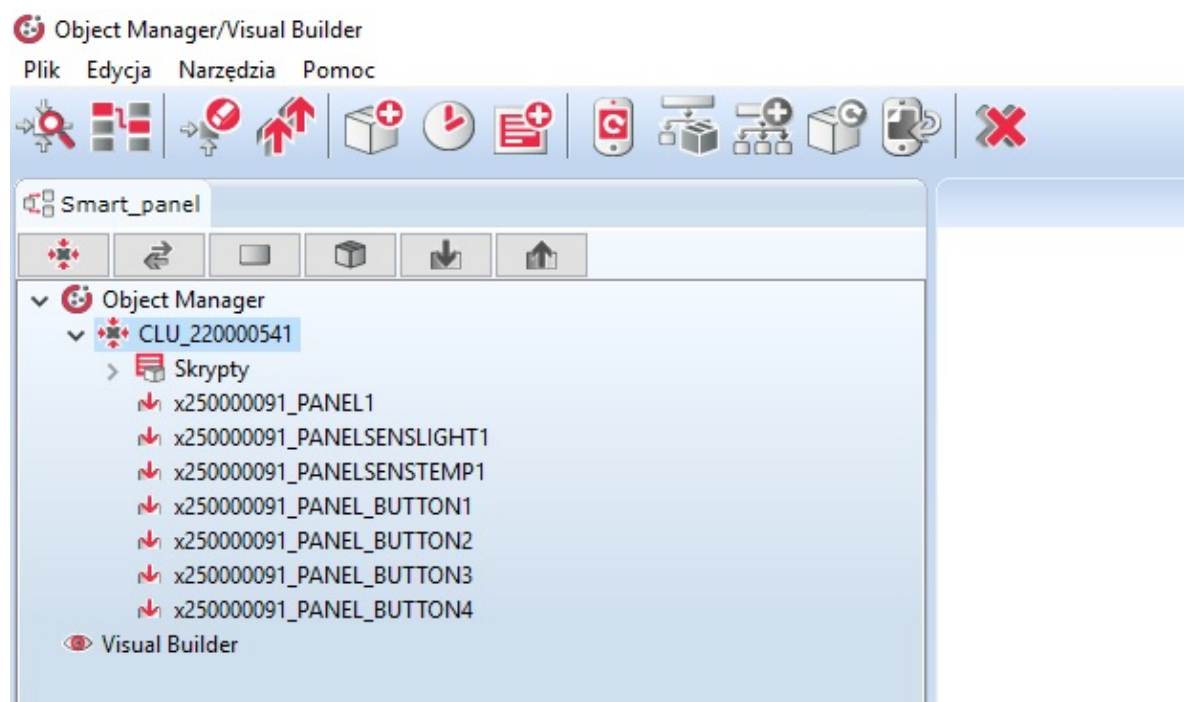
Smart Panel w wersji v4 dostępny jest dla Object Managera w wersji 1.2.1.190201 i wyższej oraz dla CLU z firmware 04.07.49-1912 i wyższym.

Podłączenie Smart Panelu do systemu odbywa się przy pomocy skrętki. Do odpowiednich zacisków złącza ARK należy wyprowadzić od Smart Panelu dwie pary skręconych przewodów - schemat podłączenia przedstawia poniższy rysunek:

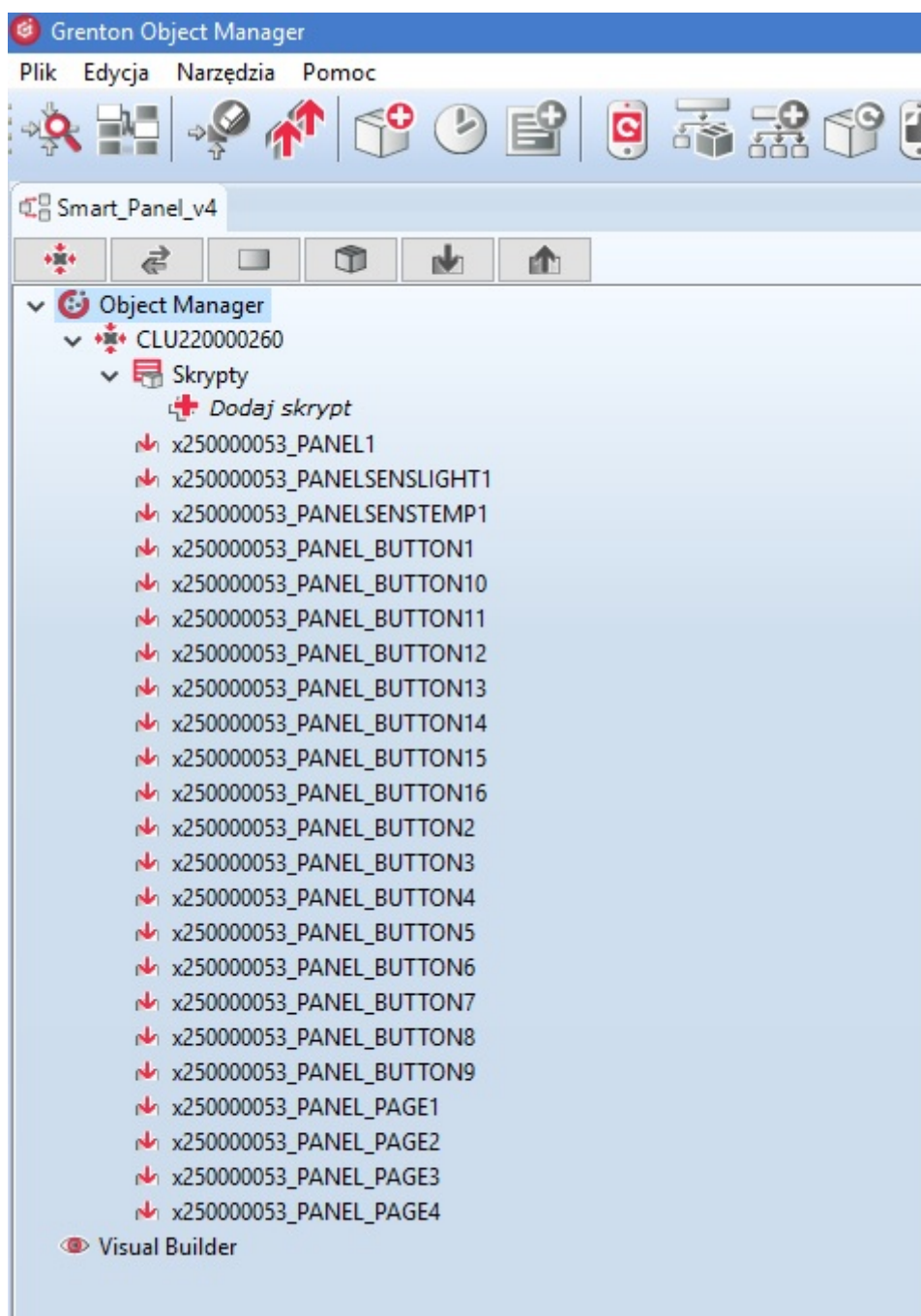
- Jeden przewód z pierwszej skręconej pary (np. kabla UTP) podłączyć do zacisku Vcc;
- Drugi przewód z pary podłączyć do zacisku GND;
- Po jednym przewodzie z drugiej pary podłączyć do zacisków A i B.



Po podłączeniu i przeprowadzeniu w projekcie operacji *CLU Discovery* na liście modułów pojawiają się następujące elementy Smart Panelu v3:



Po podłączeniu i przeprowadzeniu w projekcie operacji *CLU Discovery* na liście modułów pojawiają się następujące elementy Smart Panelu dla wersji v4:



W przypadku poprawnego dodania elementów do projektu, można przejść do tworzenia konfiguracji.

UWAGA! W przypadku niepowodzenia należy skontaktować się z działem Support!

3. Informacje pomocne przy tworzeniu konfiguracji

1. Konfiguracja panelu z wyświetlaczem różni się od konfiguracji klasycznego panelu dotykowego Grenton między innymi tym, że oprócz: cech, metod oraz zdarzeń każdego przycisku, czujnika temperatury/natężenia światła, użytkownik ma również do dyspozycji: sensor gestów, a także cechy, metody i zdarzenia dla samego *Smart Panelu*.

Od wersji 04.03.04.1910 dostępne są nowe funkcjonalności *Smart Panelu*, takie jak: obiekt konfiguracyjny strony *PANEL_PAGE* czy nowe cechy, metody i zdarzenia w obiekcie *PANEL*.

2. Wyświetlacz, w który wyposażony jest panel dotykowy, ma rozdzielczość 128x64 pikseli.
3. Smart Panel v3 może pracować w dwóch trybach pracy: wyświetlającym ikony (wyświetlacz podzielony jest na 4 pola) lub w trybie rysowania (wykorzystując całe pole wyświetlacza).

Smart Panel v4 może pracować w czterech trybach pracy:

1. Tryb kompatybilności wstecz (domyślna konfiguracja) - `Inactive`,
 2. Tryb wyświetlania ikon (wyświetlacz podzielony na 4 pole) - `Buttons`,
 3. Tryb rysowania (wykorzystując całe pole wyświetlacza) - `FreeDraw`,
 4. Tryb obsługi termostatów - `Thermostats`.
4. Panel dotykowy wyposażony jest w slot karty microSD, która służy do przechowywania domyślnych ikon wyświetlanych na panelu. Pliki muszą być umieszczone w głównym katalogu karty z rozszerzeniem `.bmp`.
5. Ekran Smart Panelu domyślnie jest wygaszony. Zapala się w chwili, gdy zadziała czujnik zbliżenia (czas wyświetlania jest pobierany z cechy `Panel->ProximityTimeout` – po tym czasie, gdy panel nie wykrywa obecności, wyświetlacz wygasza się).
6. Czujnik obecności działa w zależności od odległości ustawianej przy pomocy czułości – cecha `ProximitySens`. Po wykryciu obecności generowane jest zdarzenie `OnProximityDetect`.

4. Konfiguracja modułu Smart Panel w wersji v3

4.1. Parametry konfiguracyjne

A. Panel

CECHY

Nazwa	Opis
<code>GestureIconUp</code>	Nazwa pliku BMP z ikoną dla gestu w górę (bez rozszerzenia)
<code>GestureIconDown</code>	Nazwa pliku BMP z ikoną dla gestu w dół (bez rozszerzenia)
<code>GestureIconLeft</code>	Nazwa pliku BMP z ikoną dla gestu w lewo (bez rozszerzenia)
<code>GestureIconRight</code>	Nazwa pliku BMP z ikoną dla gestu w prawo (bez rozszerzenia)
<code>ProximitySens</code>	Czułość sensora zbliżeniowego
<code>ProximityTimeout</code>	Czas, po którym wyświetlacz zostanie wygaszony
<code>ProximityValue</code>	Sygnal sensora zbliżeniowego (wartość bezwymiarowa)
<code>BuzzerValue</code>	Sterowanie sygnalizacją dźwiękową (załącz/wyłącz)

METODY

Nazwa	Opis
<code>SwitchOnDisplay</code>	Wybudza wyświetlacz z trybu uśpienia
<code>ShowButtons</code>	Zmienia tryb wyświetlacza na <i>buttons</i>
<code>ClearScreen</code>	Czyści zawartość wyświetlacza w trybie <i>freedraw</i>
<code>PrintText</code>	Wyświetla tekst w trybie <i>freedraw</i>
<code>PrintFloat</code>	Wyświetla liczbę w trybie <i>freedraw</i>
<code>DrawLine</code>	Rysuje linię w trybie <i>freedraw</i>
<code>DrawPoint</code>	Rysuje punkt w trybie <i>freedraw</i>
<code>DrawIcon</code>	Rysuje ikonę (bmp) w trybie <i>freedraw</i>
<code>DisplayContent</code>	Wyświetla zawartość bufora pamięci graficznej; zmienia tryb wyświetlacza na <i>freedraw</i>
<code>SetGestureIconUp</code>	Ustawia plik BMP z ikoną dla gestu w górę
<code>SetGestureIconDown</code>	Ustawia plik BMP z ikoną dla gestu w dół
<code>SetGestureIconLeft</code>	Ustawia plik BMP z ikoną dla gestu w lewo
<code>SetGestureIconRight</code>	Ustawia plik BMP z ikoną dla gestu w prawo
<code>SetProximitySens</code>	Ustawia czułość sensora zbliżeniowego
<code>SetProximityTimeout</code>	Ustawia czas, po którym wyświetlacz zostanie wygaszony
<code>SetBuzzerValue</code>	Załącza/wyłącza sygnalizację dźwiękową

ZDARZENIA

Nazwa	Opis
<code>OnGestureUp</code>	Zdarzenie związane z gestem w górę
<code>OnGestureDown</code>	Zdarzenie związane z gestem w dół
<code>OnGestureLeft</code>	Zdarzenie związane z gestem w lewo
<code>OnGestureRight</code>	Zdarzenie związane z gestem w prawo
<code>OnProximityDetect</code>	Zdarzenie wywołane w przypadku wykrycia osoby zbliżającej się do wyświetlacza panelu

B. Przyciski

CECHY

Nazwa	Opis
<code>Mode</code>	Zwraca ustawiony tryb działania przycisku: 0 - monostabilny, 1 - bistabilny, 2 - zablokowany (dioda świeci na czerwono światłem ciągłym)
<code>HoldDelay</code>	Czas (w milisekundach), po jakim wyzwolone zostanie zdarzenie <code>onHold</code> (przy wciśnięciu i trzymaniu przycisku)
<code>HoldInterval</code>	Cykliczny odstęp czasu (w milisekundach), po jakim podczas trzymania przycisku wyzwolane będzie zdarzenie <code>onHold</code>
<code>value</code>	Zwraca stan wejścia (0 lub 1)
<code>Label</code>	Tekst opisujący przycisk (wyświetlany zamiast ikony)
<code>IconA</code>	Nazwa pliku ikony przypisanej do przycisku w trybie monostabilnym oraz bistabilnym w pozycji <i>OFF</i> ; nazwa poprzedzona „~” wyświetli grafikę w negatywie; <code>IconA</code> ma priorytet nad cechą <code>Label</code>
<code>IconB</code>	Nazwa pliku ikony przypisanej do przycisku w trybie bistabilnym w pozycji <i>ON</i> ; nazwa poprzedzona „~” wyświetli grafikę w negatywie

METODY

Nazwa	Opis
<code>SetMode</code>	Ustawia tryb działania przycisku: 0 - monostabilny, 1 - bistabilny, 2 - zablokowany (dioda świeci na czerwono światłem ciągłym)
<code>SetHoldDelay</code>	Ustawia wartość <code>HoldDelay</code>
<code>SetHoldInterval</code>	Ustawia wartość <code>HoldInterval</code>
<code>setLabel</code>	Ustawia tekst opisujący przycisk
<code>setIconA</code>	Ustawia plik ikony A
<code>setIconB</code>	Ustawia plik ikony B
<code>ShowOK</code>	Powoduje miganie zielonej diody na przycisku przez dwie sekundy (częstotliwość 500ms)
<code>ShowError</code>	Powoduje miganie czerwonej diody na przycisku przez dwie sekundy (częstotliwość 500ms)
<code>LedSwitchOn</code>	Załącza zieloną diodę na przycisku
<code>LedSwitchOff</code>	Wyłącza zieloną diodę na przycisku

ZDARZENIA

Nazwa	Opis
<code>OnChange</code>	Zdarzenie wywoływane przy zmianie stanu (niezależnie od wartości)
<code>OnSwitchOn</code>	Zdarzenie wywoływane w momencie ustawienia stanu wysokiego na wejściu
<code>OnSwitchOff</code>	Zdarzenie wywoływane w momencie ustawienia stanu niskiego na wejściu
<code>OnShortPress</code>	Zdarzenie wywoływane po naciśnięciu przycisku na okres 500 - 2000ms
<code>OnLongPress</code>	Zdarzenie wywoływane po naciśnięciu przycisku na okres 2000 - 5000ms
<code>OnHold</code>	Zdarzenie wywołwane po raz pierwszy po upłynięciu czasu <code>HoldDelay</code> , a następnie cyklicznie co czas <code>HoldInterval</code>
<code>OnClick</code>	Zdarzenie wywołwane po naciśnięciu przycisku na czas poniżej 500ms

C. Czujniki temperatury i oświetlenia

CECHY

Nazwa	Opis
<code>Threshold</code>	Wielkość histerezy (dokładność 0,1) określająca czułość, przy której następuje wygenerowanie zdarzeń: <code>OnChange</code> , <code>OnLowerValue</code> , <code>OnRaiseValue</code>
<code>Sensitivity</code>	Czas (w ms), dla którego próbkowane wartości są uśredniane
<code>MinValue</code>	Minimalna wartość cechy <code>value</code> , której przekroczenie wywołuje zdarzenie <code>OnOutOfRange</code>
<code>MaxValue</code>	Maksymalna wartość cechy <code>value</code> , której przekroczenie wywołuje zdarzenie <code>OnOutOfRange</code>
<code>value</code>	Wartość wejścia: dla czujnika temperatury (od 0 do 45°C) lub dla czujnika światła (0 - 100%)

ZDARZENIA

Nazwa	Opis
<code>OnChange</code>	Zdarzenie wywołwane w momencie zmiany stanu wejścia (niezależnie od wartości)
<code>OnRaiseValue</code>	Zdarzenie wywołwane przy przekroczeniu górnego progu histerezy
<code>OnLowerValue</code>	Zdarzenie wywołwane przy przekroczeniu dolnego progu histerezy
<code>OnOutOfRange</code>	Zdarzenie wywołwane, gdy wartość na wyjściu znajduje się poza wyznaczonym zakresem

4.2. Tworzenie konfiguracji przycisków i wyświetlacza

W celu stworzenia konfiguracji należy:

- Otworzyć obiekt *PANEL_BUTTONX* (gdzie X to numer jednego z 4 przycisków) poprzez dwukrotne kliknięcie na liście modułów;
- Przejść do zakładki *Zdarzenia*;
- Skonfigurować działanie przycisku poprzez przypisanie metod do określonych zdarzeń (klikając w „+” po prawej stronie okna):

Nazwa zdarzenia	Przypisane komendy	Dodaj komendę
OnChange	CLU_220000541->x200000534_DOUT1->Switch(0)	Przypisz komendę ✖ +
OnSwitchOn	CLU_220000541->x210000904_DIMM1->SwitchOn(0,500)	Przypisz komendę ✖ +
OnSwitchOff	CLU_220000541->x210000904_DIMM1->SwitchOff(0,500)	Przypisz komendę ✖ +
OnShortPress		+ ✖
OnLongPress		+ ✖
OnHold		+ ✖
OnClick		+ ✖

- Wybrać zakładkę *Cechy wbudowane* i zdefiniować obiekty wyświetlane na ekranie danego przycisku:
 - **Label** – cecha określająca tekst przypisany do danego przycisku;
 - **IconA** – cecha określająca nazwę ikony przypisanej do danego przycisku, gdy znajduje się w trybie monostabilnym, bądź dla trybu bistabilnego dla cechy `value=0`;
 - **IconB** – cecha określająca nazwę ikony przypisanej do danego przycisku, gdy znajduje się w trybie bistabilnym dla cechy `value=1`. Chcąc przypisać tę samą ikonę, ale o odwróconej kolorystyce należy nazwę pikrogramu poprzedzić znakiem „~” (np. `~lamp1on`):

CLU_220000541->x250000091_PANEL_BUTTON1
✕

Nazwa:

Identyfikacja: 1

Źródło/Odbiornik:

Typ:

Sterowanie
 Schematy konfiguracji
 Zdarzenia
 Cechy wbudowane
 Statystyki

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
Mode	1	<input type="text" value="Bistabilny"/>		0,1,2
HoldDelay	1000	<input type="text" value="1000"/>	ms	[0-5000]
HoldInterval	100	<input type="text" value="50"/>	ms	[0-2000]
Value	0		bool	0,1
Label	-	<input type="text" value=""/>	string	[0-15]
IconA	lamp2off	<input type="text" value="lamp2off"/>	string	[0-9]
IconB	~lamp2on	<input type="text" value="~lamp2on"/>	string	[0-9]

Auto odświeżanie
 Odśwież

OK
Anuluj

Powyzsze cechy mozna ustawic zarowno w zakladce *Cechy wbudowane*, jak rowniez za posrednictwem metod: `SetLabel`, `SetIconA`, `SetIconB`.

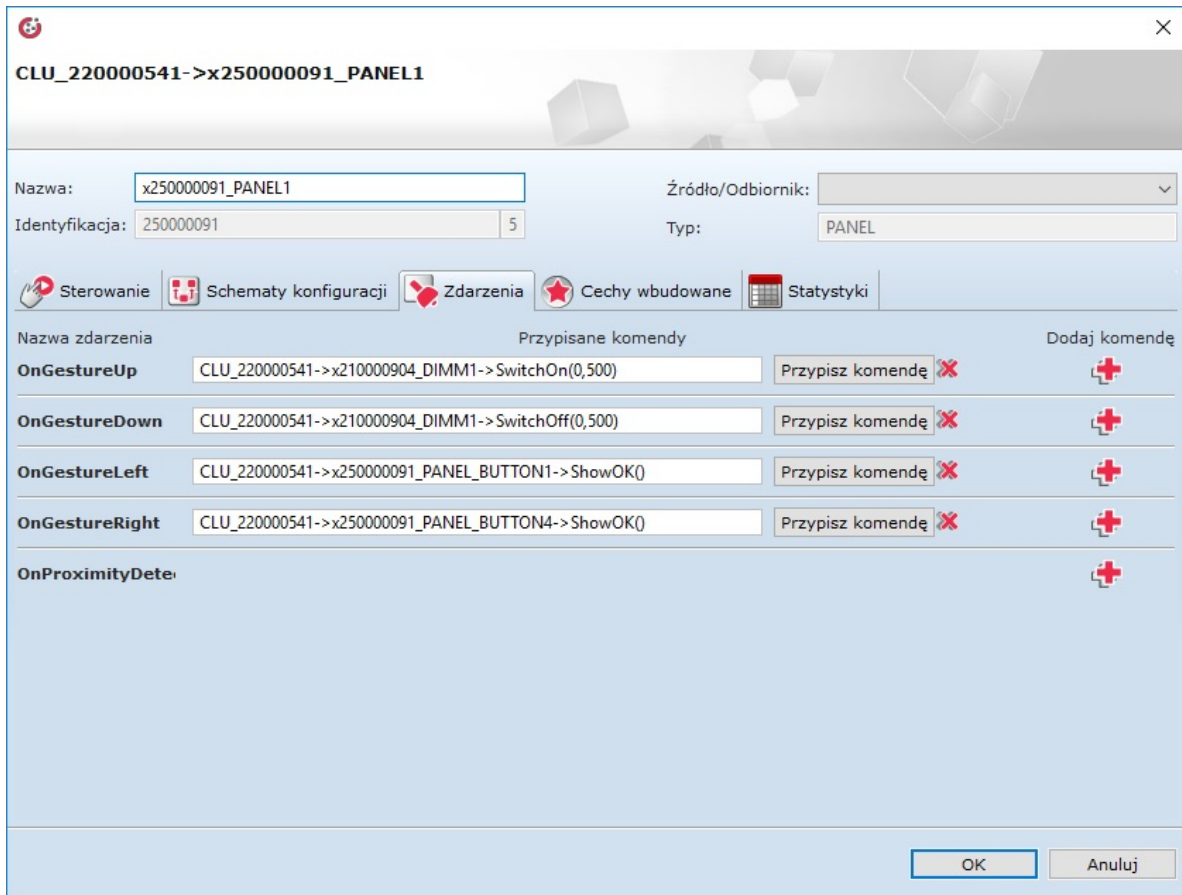
UWAGA! Metoda `setIcon` ma w systemie wyzszy priorytet niz metoda `setLabel`!

- Wyslac konfiguracje do CLU.

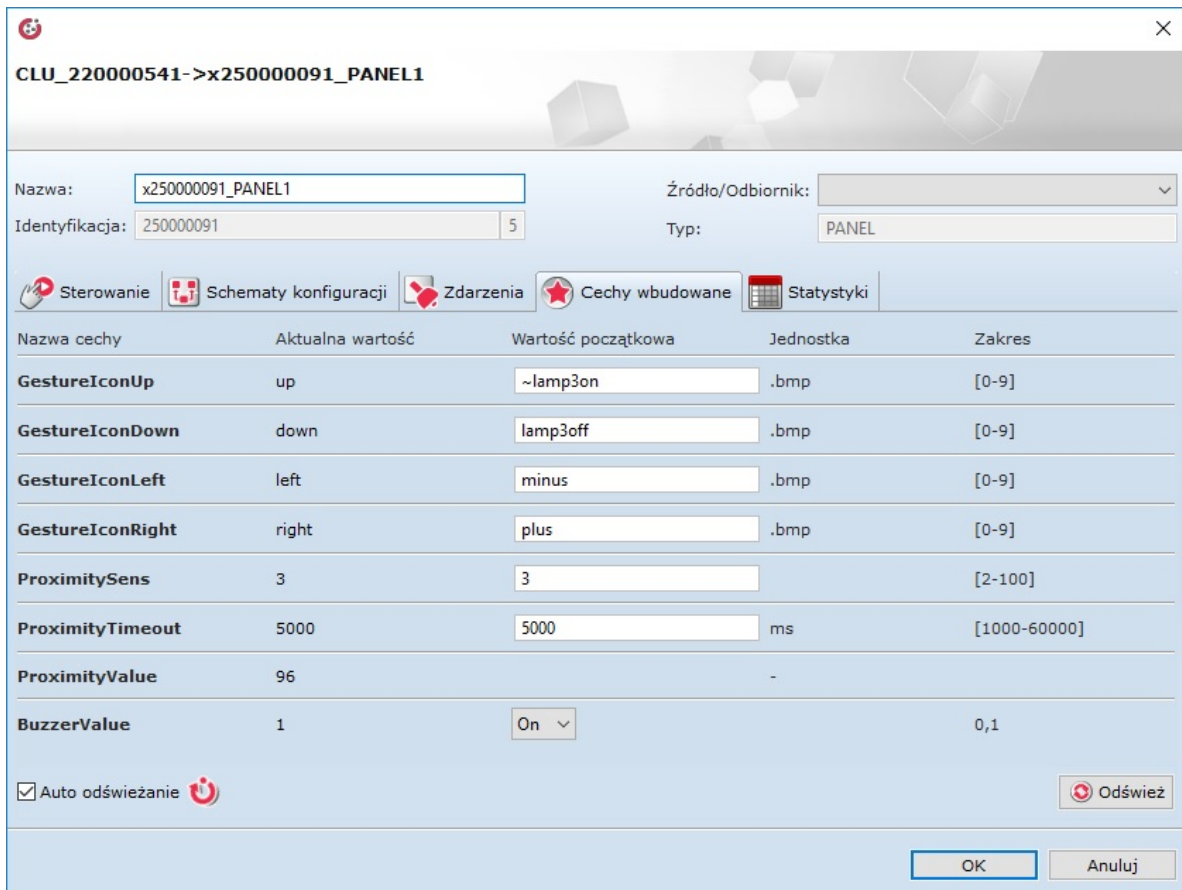
4.3. Tworzenie konfiguracji sensora gestow

Chcac utworzyc konfiguracje dla sensora gestow nalezy:

- Otworzyc - przez dwuklik - obiekt *Panel*;
- Przejsc do zakladki *Zdarzenia*;
- Przypisac metody do zdarzen `OnGestureUp`, `OnGestureDown`, `OnGestureLeft`, `onGestureRight` (klikajac w „+” po prawej stronie kazdej z metod):



Możliwa jest podmiana domyślnie wyświetlanych ikon przy wywoływaniu gestów – w tym celu należy przejść do zakładki *Cechy wbudowane* i wpisać nazwy żądanych ikon bez rozszerzenia *.bmp*:



Użycie ikon będzie możliwe, gdy będą wgrane na kartę microSD z rozszerzeniem *.bmp*.

- Zatwierdzić okno konfiguracji przyciskiem *OK*;
- Wysłać konfigurację do CLU.

4.4. Konfiguracja czujnika zbliżenia

W celu ustawienia parametrów czujnika zbliżenia należy:

- Otworzyć – przez dwuklik - obiekt *Panel*;
- Przejść do zakładki *Cechy wbudowane*, gdzie znajdują się 3 cechy odnoszące się do czujnika zbliżenia:
 - `ProximitySens` - określa czułość sensora;
 - `ProximityTimeout` - definiuje czas, po jakim wyświetlacz zostaje wygaszony, gdy nie zostanie wykryty ruch;
 - `ProximityValue` - zwraca przybliżoną odległość w centymetrach od panelu do obiektu;

CLU_220000541->x250000091_PANEL1

Nazwa: x250000091_PANEL1 Źródło/Odbiornik:
 Identyfikacja: 250000091 5 Typ: PANEL

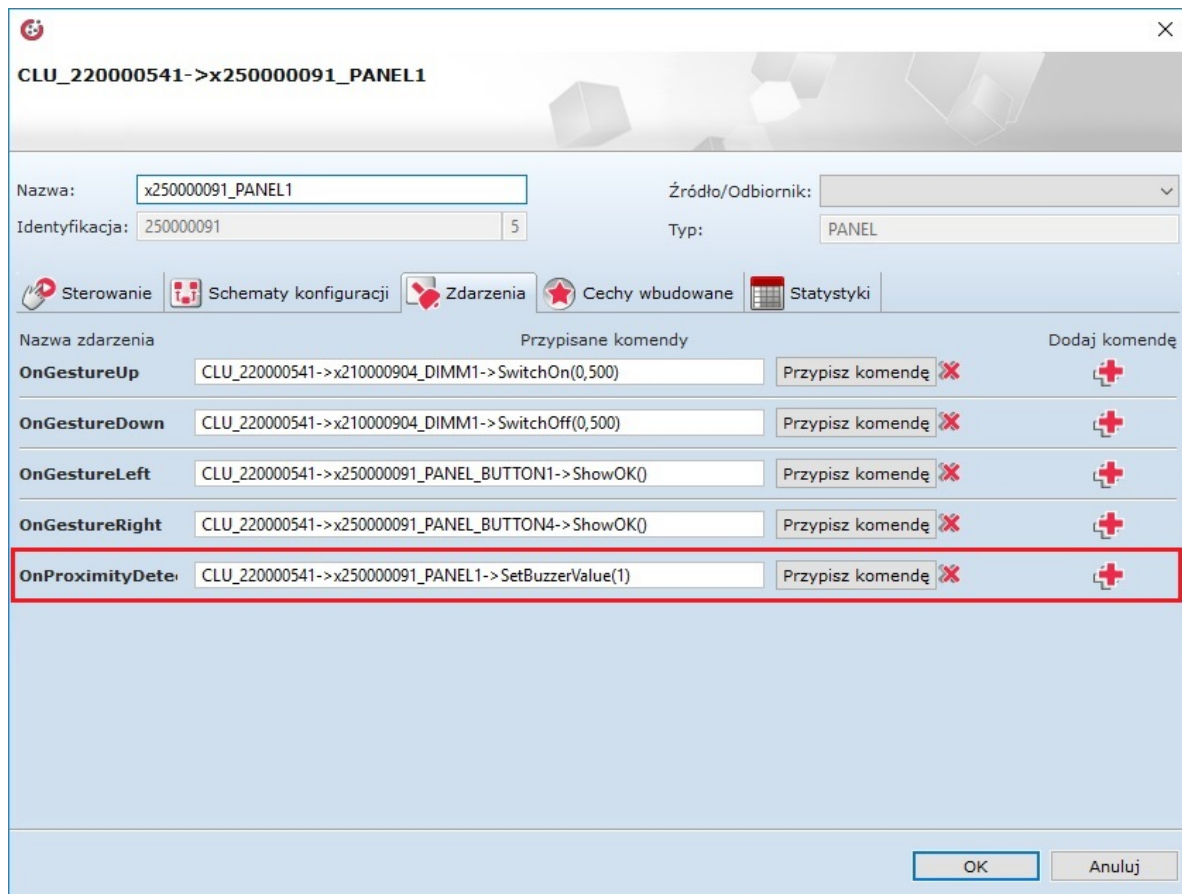
Sterowanie Schematy konfiguracji Zdarzenia **Cechy wbudowane** Statystyki

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
GestureIconUp	~lamp3on	~lamp3on	.bmp	[0-9]
GestureIconDown	lamp3off	lamp3off	.bmp	[0-9]
GestureIconLeft	minus	minus	.bmp	[0-9]
GestureIconRight	plus	plus	.bmp	[0-9]
ProximitySens	5	5		[2-100]
ProximityTimeout	10000	10000	ms	[1000-60000]
ProximityValue	96		-	
BuzzerValue	0	Off		0,1

Auto odświeżanie

Powyższe cechy można ustawiać zarówno w zakładce *Cechy wbudowane*, jak również przy pomocy metod: `SetProximitySens` oraz `SetProximityTimeout` (w metodach obiektu *Panel*).

- Reakcja czujnika zbliżenia generuje zdarzenie `onProximityDetect`, do którego można dodać dodatkowe metody:

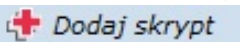


- Wysłać konfigurację do CLU.

4.5. Tworzenie konfiguracji wielu stron panelu dotykowego

Chcąc rozpocząć tworzenie konfiguracji panelu obsługującego wiele stron, na CLU należy utworzyć cechę typu *number* (określa numer strony startowej) o przykładowej nazwie *strona* - w tym celu kliknąć dwukrotnie na CLU, przejść do zakładki *Cechy użytkownika* i wybrać przycisk:



Aby panel mógł wyświetlać żądaną zawartość na ekranie, konieczne jest utworzenie skryptu (np. *Wyswietlacz*) z kilkoma stronami - w tym celu wybrać przycisk przy lewej krawędzi okna programu Object Manager: 

UWAGA! Nazwa skryptu nie może zawierać polskich znaków!

- **STRONA Z PRZYCISKAMI** - Do skryptu należy dodać warunek sprawdzający, jaki jest aktualny numer strony (wartość *Cechy użytkownika: strona*), a dla spełnionego warunku - dla konkretnej strony - dodać akcję przypisania ikon wszystkich 4 przycisków (metody `setIconA` dla elementów `PANEL_BUTTON1-4`) oraz metodę `ShowButtons` wyświetlającą na ekranie panelu wybrane ikony;

UWAGA! Poza przypisaniem ikon do określonych przycisków wymagane jest wywołanie metody `ShowButtons`, gdyż samo ich przypisanie nie spowoduje, że pojawią się na wyświetlaczu!

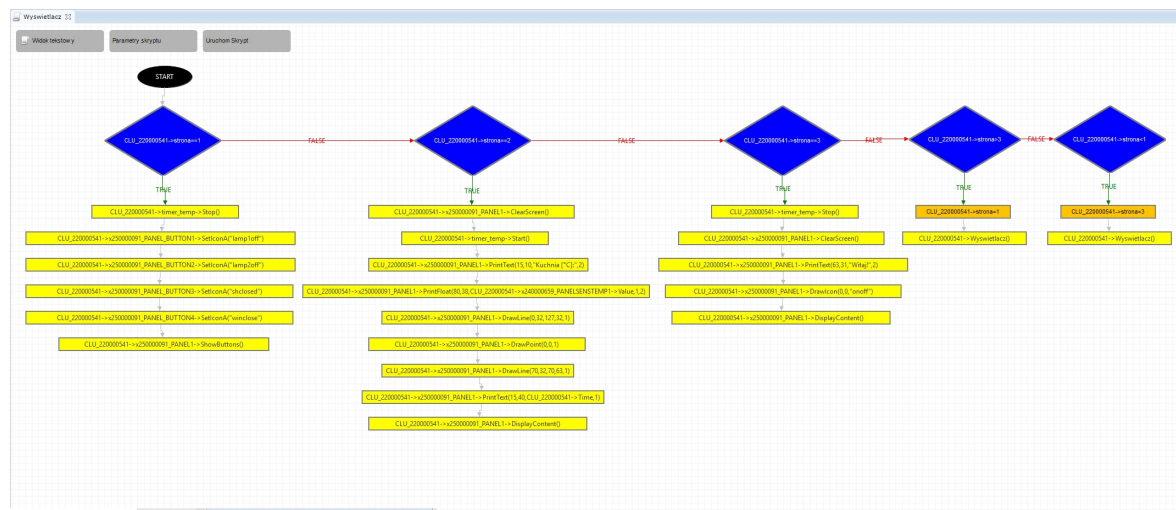
UWAGA! W przypadku tworzenia wielu stron, ustawienie przycisku w tryb bistabilny - przy pomocy cechy/metody - nie będzie poprawnie odczytywało stanu przełącznika (z uwagi na różne funkcjonalności przycisków przy zmianach stron)!!

- **STRONA Z GRAFIKAMI I TEKSTAMI** – Przy projektowaniu strony zawierającej grafiki oraz teksty należy dodać:
 - warunek sprawdzający numer strony (nie może być to strona z przyciskami);
 - akcję `PANEL->ClearScreen()` ;
 - akcje ustawienia tekstu oraz linii (opisane poniżej);
 - akcję `PANEL->DisplayContent()` .

Akcje ustawiania tekstu i linii:

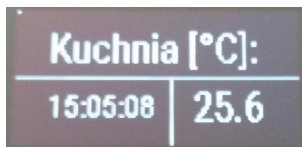
- `PANEL->PrintText` – metoda powodująca wypisanie tekstu lub cechy – do jej wywołania należy podać cztery parametry: początkowe współrzędne na ekranie (x, y), tekst oraz rozmiar czcionki (gdzie 1 – 10 pkt, 2 – 14 pkt, 3 – 28 pkt);
 - `PANEL->PrintFloat` – metoda działająca tak samo jak `PrintText`, z tą różnicą, że posiada dodatkowy parametr *Precision*, odpowiedzialny za ilość miejsc po przecinku parametru *number*;
 - `PANEL->DrawLine` – metoda rysująca linię – do jej wywołania konieczne jest podanie 5 parametrów: współrzędne początkowe (x, y), końcowe (xe, ye) oraz kolor linii (gdzie 0 – czarny, 1 - biały);
 - `PANEL->DrawPoint` – metoda rysująca punkt – do jej wywołania należy podać 3 parametry: współrzędne (x, y) oraz kolor (parametr działa jak przy wywołaniu metody `DrawLine`);
 - `PANEL->DrawIcon` – metoda rysująca ikonę – do jej wywołania trzeba podać 3 parametry: współrzędne początkowe (x, y) oraz nazwę ikony z zasobnika.
- **ZAPĘTLENIE SKRYPTU** – Do skryptu należy dodać warunki, które spowodują, że przy wygenerowaniu gestu w prawo na ostatniej stronie, panel wróci do pierwszej strony (i na odwrót) - tak, by uzyskać działanie pętli.

Realizację wszystkich powyżej opisanych metod zaprezentowano na zrzucie ekranu przykładowego skryptu:



Powyższy skrypt zamieszczono na końcu dokumentu w wersji tekstowej (punkt 3.).

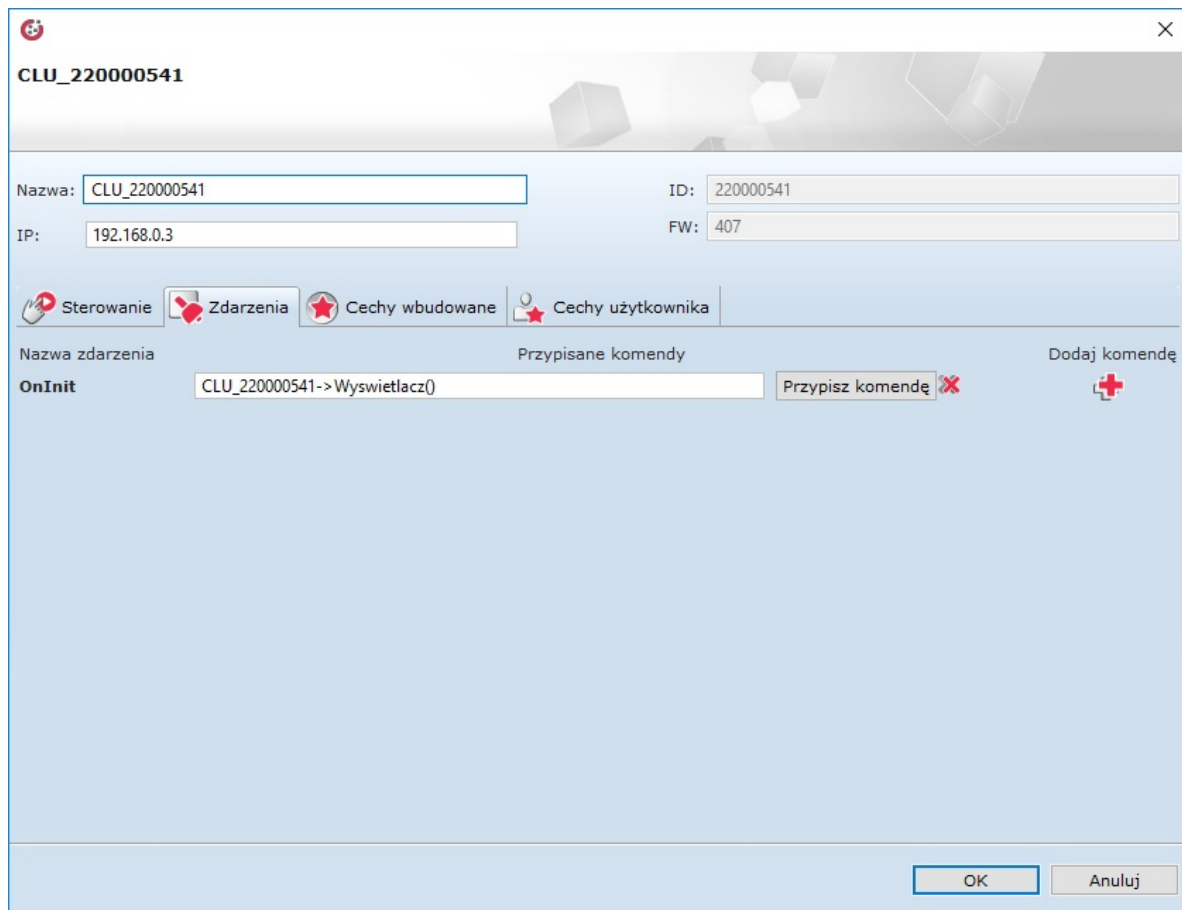
Druga strona zaprogramowana w skrypcie będzie wyglądać następująco:



- W następnym kroku - do gestów panelu w lewo i w prawo - należy przypisać operacje zwiększania zmiennej użytkownika *strona* oraz uruchamiania skryptu *Wyswietlacz* jak na rysunku poniżej:

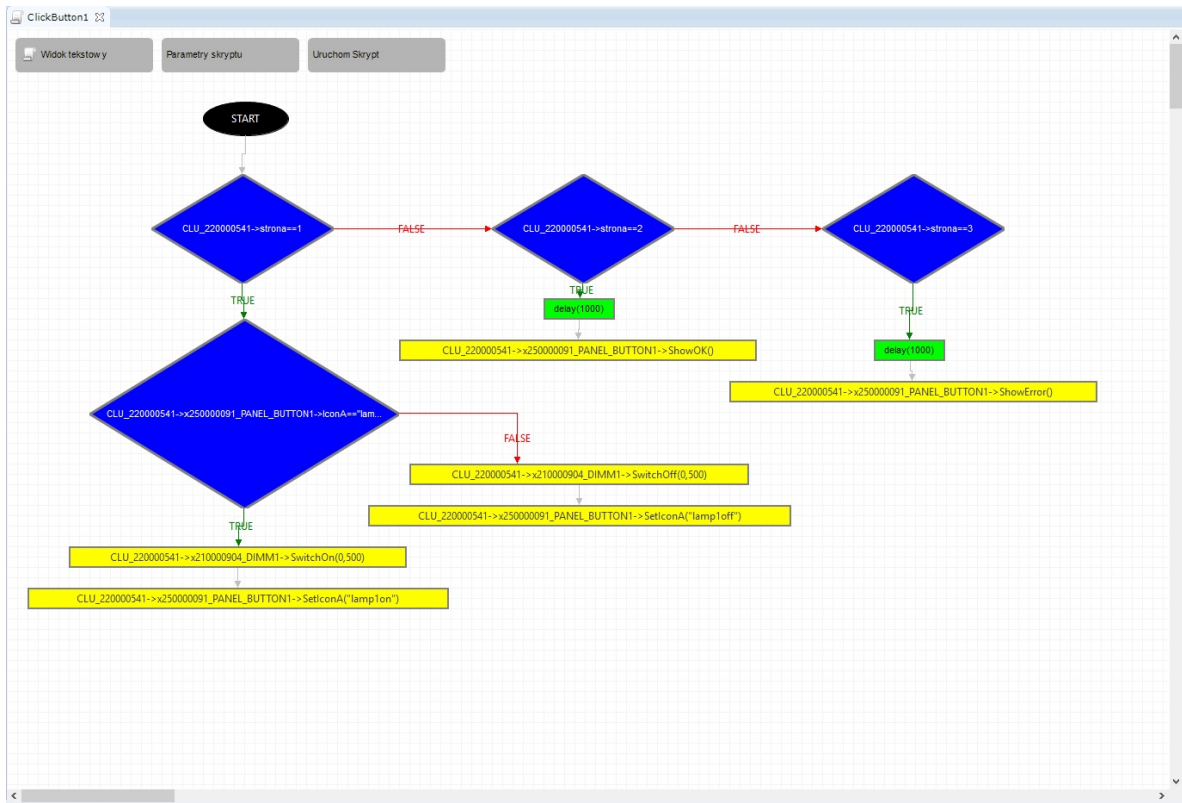
Nazwa zdarzenia	Przypisane komendy	Dodaj komendę
OnGestureUp		
OnGestureDown	CLU_220000541->Wyswietlacz()	
	CLU_220000541->strona=1	
OnGestureLeft	CLU_220000541->strona=CLU_220000541->strona-1	
	CLU_220000541->Wyswietlacz()	
OnGestureRight	CLU_220000541->strona=CLU_220000541->strona+1	
	CLU_220000541->Wyswietlacz()	
OnProximityDete		

- Przypisać do zdarzenia *CLU-> onInit* wywołanie skryptu *Wyswietlacz*:



- Stworzyć skrypt (np. *ClickButton1*) do obsługi zdarzenia `onClick` jednego wybranego przycisku na każdej stronie – należy utworzyć osobne skrypty dla każdego przycisku:
 - Dodać warunek sprawdzający numer strony;
 - Chcąc zrealizować funkcję trybu bistabilnego dla przycisku, dodać kolejny warunek sprawdzający aktualny stan ikony i podejmujący odpowiednie działania (załączający lub wyłączający, np. oświetlenie);
 - Dodać kolejne warunki sprawdzające numer strony.

Realizację pokazano na poniższym zrzucie ekranu:



Powyższy skrypt zamieszczono na końcu dokumentu w wersji tekstowej (punkt 4.)

UWAGA! Operacja na zmiennych użytych w trybie graficznym panelu nie powoduje odświeżania, dlatego w powyższym skrypcie zastosowano akcję ponownego wygenerowania strony!

- Na koniec należy dodać kolejne skrypty do wszystkich przycisków oraz używanych zdarzeń - odpowiednio: skrypt *ClickButton1* do zdarzenia *PANEL_BUTTON1->onClick*.

OnHold			
OnClick	CLU_220000541->ClickButton1()	Przypisz komendę	

3. Skrypt *Wyswietlacz* w wersji tekstowej:

```

if(not (CLU_220000541->strona==1)) then
if(not (CLU_220000541->strona==2)) then
if(not (CLU_220000541->strona==3)) then
if(CLU_220000541->strona>3) then
CLU_220000541->strona=1
CLU_220000541->wyswietlacz()
else
if(CLU_220000541->strona<1) then
CLU_220000541->strona=3
CLU_220000541->wyswietlacz()
end
end
else
CLU_220000541->timer_temp->Stop()
CLU_220000541->x250000091_PANEL1->ClearScreen()

```

```

CLU_220000541->x250000091_PANEL1->PrintText(63,31,"witaj!",2)
CLU_220000541->x250000091_PANEL1->DrawIcon(0,0,"onoff")
CLU_220000541->x250000091_PANEL1->DisplayContent()
end
else
CLU_220000541->x250000091_PANEL1->ClearScreen()
CLU_220000541->timer_temp->Start()
CLU_220000541->x250000091_PANEL1->PrintText(15,10,"kuchnia [°C]:",2)
CLU_220000541->x250000091_PANEL1->PrintFloat(80,38,CLU_220000541->
>x240000659_PANELSENSTEMP1->value,1,2)
CLU_220000541->x250000091_PANEL1->DrawLine(0,32,127,32,1)
CLU_220000541->x250000091_PANEL1->DrawPoint(0,0,1)
CLU_220000541->x250000091_PANEL1->DrawLine(70,32,70,63,1)
CLU_220000541->x250000091_PANEL1->PrintText(15,40,CLU_220000541->Time,1)
CLU_220000541->x250000091_PANEL1->DisplayContent()
end
else
CLU_220000541->timer_temp->Stop()
CLU_220000541->x250000091_PANEL_BUTTON1->SetIconA("lamp1off")
CLU_220000541->x250000091_PANEL_BUTTON2->SetIconA("lamp2off")
CLU_220000541->x250000091_PANEL_BUTTON3->SetIconA("shclosed")
CLU_220000541->x250000091_PANEL_BUTTON4->SetIconA("winclose")
CLU_220000541->x250000091_PANEL1->ShowButtons()
end

```

4. Skrypt *ClickButton1* w wersji tekstowej:

```

if(not (CLU_220000541->strona==1)) then
if(not (CLU_220000541->strona==2)) then
if(CLU_220000541->strona==3) then
SYSTEM.Wait(1000)
CLU_220000541->x250000091_PANEL_BUTTON1->ShowError()
end
else
SYSTEM.Wait(1000)
CLU_220000541->x250000091_PANEL_BUTTON1->ShowOK()
end
else
if(CLU_220000541->x250000091_PANEL_BUTTON1->IconA=="lamp1off") then
CLU_220000541->x210000904_DIMM1->SwitchOn(0,500)
CLU_220000541->x250000091_PANEL_BUTTON1->SetIconA("lamp1on")
else
CLU_220000541->x210000904_DIMM1->SwitchOff(0,500)
CLU_220000541->x250000091_PANEL_BUTTON1->SetIconA("lamp1off")
end
end

```

5. Konfiguracja modułu Smart Panel w wersji v4

UWAGA!

Smart Panel w wersji v4 dostępny jest dla Object Managera w wersji 1.2.1.190201 i wyższej oraz dla CLU z firmware 04.07.49-1912 i wyższym.

5.1. Parametry konfiguracyjne

A. Panel

CECHY

Nazwa	Opis
GestureIconUp	Nazwa pliku BMP z ikoną dla gestu Góra (bez rozszerzenia)

Nazwa	Opis
<code>GestureIconDown</code>	Nazwa pliku BMP z ikoną dla gestu Dół (bez rozszerzenia)
<code>GestureIconLeft</code>	Nazwa pliku BMP z ikoną dla gestu Lewo (bez rozszerzenia)
<code>GestureIconRight</code>	Nazwa pliku BMP z ikoną dla gestu Prawo (bez rozszerzenia)
<code>ProximitySens</code>	Czułość sensora zbliżeniowego (mniejsza wartość - większa czułość)
<code>ProximityTimeout</code>	Czas, po którym wyświetlacz zostanie wygaszony
<code>ProximityValue</code>	Sygnał sensora zbliżeniowego (wartość bezwymiarowa)
<code>BuzzerValue</code>	Sterowanie sygnalizacją dźwiękową: 0 - off, 1 - on
<code>GestureMode</code>	Wybór orientacja gestów: 0 - off, 1 - vertical, 2 - horizontal, 3 - vert+horiz
<code>GestureSens</code>	Wybór czułości gestów: 1 - Low, 2 - Mid, 3 - High
<code>PageNr</code>	Numer wyświetlanej aktualnie strony
<code>PageDisplayMode</code>	Informacja przed zmianą strony: 0 - ShowImmediately, 1 - ShowIconOrName, 2 - ShowGesture
<code>ButtonsLEDMode</code>	Lokalizacja przycisków za pomocą słabego światła LED: 0 - LocationLedOFF, 1 - LocationLedON, 2 - LocationLedONforActive
<code>PageControlMode</code>	Źródło, które przełącza strony: 0 - Command (przełączanie za pomocą metod SetNextPage oraz SetPrevPage) 1 - Gesture/Command (przełączanie za pomocą gestów oraz metod SetNextPage i SetPrevPage)
<code>GestureDisplayMode</code>	Wyświetlanie informacji o aktualnie wykonanym geście: 0 - off, 1 - on

METODY

Nazwa	Opis
SwitchOnDisplay	Wybudza wyświetlacz z trybu uśpienia
ShowButtons	Zmienia tryb wyświetlacza na <i>buttons</i> . Czyści wyświetlacz i wyświetla ponownie ikony (lub tekst) dla wszystkich przycisków
ClearScreen	Czyści zawartość wyświetlacza w trybie <i>freedraw</i>
PrintText	Wyświetla tekst w trybie <i>freedraw</i> z użyciem parametrów: <code>x</code> , <code>y</code> , <code>txt</code> , <code>font size</code> , gdzie: <code>x</code> oraz <code>y</code> to współrzędne wyrażone w pikselach, <code>txt</code> to string, <code>font size</code> to rozmiar czcionki (1:10p, 2:14p, 3:32p)
PrintFloat	Wyświetla liczbę w trybie <i>freedraw</i> z użyciem parametrów: <code>x</code> , <code>y</code> , <code>number</code> , <code>precision</code> , <code>font size</code> , gdzie: <code>x</code> oraz <code>y</code> to współrzędne wyrażone w pikselach, <code>number</code> to liczba, <code>precision</code> to ilość miejsc po przecinku, <code>font size</code> to rozmiar czcionki (1:10p, 2:14p, 3:32p)
DrawLine	Rysuje linie w trybie <i>freedraw</i> z użyciem parametrów: <code>x</code> , <code>y</code> , <code>xe</code> , <code>ye</code> , <code>color</code> , gdzie: <code>x</code> oraz <code>y</code> to współrzędne początkowe, <code>xe</code> oraz <code>ye</code> to współrzędne końcowe, <code>color</code> to kolor linii (0 - czarny, 1 - biały). Współrzędne początkowe oraz końcowe są wyrażone w pikselach
DrawPoint	Rysuje punkt w trybie <i>freedraw</i> z użyciem parametrów: <code>x</code> , <code>y</code> , <code>color</code> , gdzie: <code>x</code> oraz <code>y</code> to współrzędne wyrażone w pikselach, <code>color</code> to kolor punktu (0 - czarny, 1 - biały)
DrawIcon	Rysuje ikonę (bmp) w trybie <i>freedraw</i> z użyciem parametrów: <code>x</code> , <code>y</code> , <code>Filename</code> , gdzie: <code>x</code> oraz <code>y</code> to współrzędne wyrażone w pikselach, <code>Filename</code> to nazwa ikony (bez rozszerzenia)
DisplayContent	Wyświetla zawartość bufora pamięci graficznej. Zmienia tryb wyświetlacza na <i>freedraw</i>
SetGestureIconUp	Ustawia ikonę dla wykonania gestu w górę
SetGestureIconDown	Ustawia ikonę dla wykonania gestu w dół
SetGestureIconLeft	Ustawia ikonę dla wykonania gestu w lewo
SetGestureIconRight	Ustawia ikonę dla wykonania gestu w prawo
SetProximitySens	Ustawia wartość ProximitySens
SetProximityTimeout	Ustawia czas w sekundach po jakim wyświetlacz gaśnie

Nazwa	Opis
<code>SetBuzzerValue</code>	Sterowanie sygnalizacją dźwiękową (On/Off)
<code>SetGestureMode</code>	Wybór orientacji gestów
<code>SetGestureSens</code>	Wybór czułości gestów
<code>SetBeep</code>	Generuje dźwięk o danej częstotliwości[Hz], czasie trwania[ms] i głośności
<code>SetPageNr</code>	Ustawia numer wyświetlanej strony
<code>SetPageDisplayMode</code>	Ustawia tryb wyświetlania informacji przed zmianą strony
<code>SetButtonsLEDMode</code>	Ustawia tryb lokalizacji przycisków za pomocą diod LED
<code>SetPageControlMode</code>	Ustawia źródło, które przełącza strony (komendy/strony)
<code>SetGestureDisplayMode</code>	Ustawia tryb wyświetlania informacji o wykonanym geście
<code>SetNextPage</code>	Wyświetla następną stronę
<code>SetPrevPage</code>	Wyświetla poprzednią stronę
<code>Draw</code>	Wyzwala wywołanie zdarzenia OnDraw w przypadku gdy OLED jest aktywny

ZDARZENIA

Nazwa	Opis
<code>OnGestureUp</code>	Zdarzenie wywoływane w przypadku wykonania gestu w górę
<code>OnGestureDown</code>	Zdarzenie wywoływane w przypadku wykonania gestu w dół
<code>OnGestureLeft</code>	Zdarzenie wywoływane w przypadku wykonania gestu w lewo
<code>OnGestureRight</code>	Zdarzenie wywoływane w przypadku wykonania gestu w prawo
<code>OnProximityDetect</code>	Zdarzenie wywołane w przypadku wykrycia zbliżającej się do wyświetlacza osoby
<code>OnPageChange</code>	Zdarzenie wywoływane w przypadku zmiany strony w panelu

B. Przyciski

CECHY

Nazwa	Opis
Mode	Zwraca ustawiony tryb działania przycisku: 0 - monostabilny (monostable), 1 - bistabilny (bistable), 2 - zablokowany (locked)
HoldDelay	Czas w milisekundach, po jakim po wciśnięciu i przytrzymaniu przycisku wyzwalane jest zdarzenie <code>onHold</code>
HoldInterval	Odstęp cykliczny w milisekundach, po jakim podczas trzymania przycisku wyzwalane jest zdarzenie <code>onHold</code>
Value	Zwraca stan przycisku jako 0 lub 1
Label	Tekst, który opisuje przycisk (wyświetlany zamiast ikony)
IconA	Nazwa pliku ikony przypisanej do przycisku w trybie monostabilnym oraz bistabilnym w pozycji <i>OFF</i> ; nazwa poprzedzona „~” wyświetli grafikę w negatywie; <code>IconA</code> ma priorytet nad cechą <code>Label</code>
IconB	Nazwa pliku ikony przypisanej do przycisku w trybie bistabilnym w pozycji <i>ON</i> ; nazwa poprzedzona „~” wyświetli grafikę w negatywie

METODY

Nazwa	Opis
SetMode	Ustawia tryb działania przycisku: 0 - monostabilny (monostable), 1 - bistabilny (bistable), 2 - zablokowany (locked)
SetHoldDelay	Ustawia wartość <code>HoldDelay</code>
SetHoldInterval	Ustawia wartość <code>HoldInterval</code>
SetLabel	Ustawia wartość <code>Label</code> (tekst opisujący przycisk)
SetIconA	Ustawia nazwę pliku ikony A (bez rozszerzenia)
SetIconB	Ustawia nazwę pliku ikony B (bez rozszerzenia)
ShowOK	Powoduje miganie zielonej diody na przycisku przez dwie sekundy (częstotliwość 500 ms). Czerwona dioda przycisku pozostaje zgaszona
ShowError	Powoduje miganie czerwonej diody na przycisku przez dwie sekundy (częstotliwość 500 ms). Zielona dioda przycisku pozostaje zgaszona
LedSwitchOn	Załącza zieloną diodę na przycisku
RedLedSwitchOn	Załącza czerwoną diodę na przycisku
LedSwitchOff	Wyłącza wszystkie diody na przycisku

ZDARZENIA

Nazwa	Opis
<code>onChange</code>	Zdarzenie wywoływane przy zmianie stanu na przeciwny
<code>onSwitchOn</code>	Zdarzenie wywoływane w momencie ustawienia stanu wysokiego na wejściu
<code>onSwitchOff</code>	Zdarzenie wywoływane w momencie ustawienia stanu niskiego na wejściu
<code>onShortPress</code>	Zdarzenie wywoływane po naciśnięciu przycisku na czas 500 ms - 2000 ms
<code>onLongPress</code>	Zdarzenie wywoływane po naciśnięciu przycisku na czas 2000 ms – 5000 ms
<code>onHold</code>	Zdarzenie wywoływane gdy wejście jest w stanie wysokim, pierwszy raz po upływie czasu <code>holdDelay</code> , a następnie cyklicznie co wartość <code>holdInterval</code>
<code>onClick</code>	Zdarzenie wywoływane po naciśnięciu przycisku na czas krótszy niż 500ms

C. Konfiguracja stron (Panel_Page)

CECHY

Nazwa	Opis
PageType	Typ strony wyświetlanej na Smart Panelu: 0 - Inactive, 1 - Buttons, 2 - Thermostats, 3 - FreeDraw
PageName	Nazwa strony/nazwa ikony wyświetlana na Smart Panelu (przy przechodzeniu pomiędzy stronami)
Object_1_Id	ID obiektu termostatu lub numer przycisku w zależności od typu strony, np.: Dla strony typu <i>Thermostats</i> : - w przypadku termostatu na lokalnym CLU: THE1325 - w przypadku termostatu na zdalnym CLU: CLU220000001->THE4321 W przypadku cechy PageType ustawionej na <i>Buttons/FreeDraw</i> należy wpisać numer przycisku (1..16)
Object_1_Name	Nazwa termostatu wyświetlana na stronie Smart Panelu. Dotyczy tylko strony <i>Thermostats</i> (brak nazwy - termostat nieaktywny). W przypadku cechy PageType ustawionej na <i>Buttons/FreeDraw</i> cecha pozostaje pusta
Object_2_Id	ID obiektu termostatu lub numer przycisku w zależności od typu strony, np.: Dla strony typu <i>Thermostats</i> : - w przypadku termostatu na lokalnym CLU: THE1325 - w przypadku termostatu na zdalnym CLU: CLU220000001->THE4321 W przypadku cechy PageType ustawionej na <i>Buttons/FreeDraw</i> należy wpisać numer przycisku (1..16)
Object_2_Name	Nazwa termostatu wyświetlana na stronie Smart Panelu. Dotyczy tylko strony <i>Thermostats</i> (brak nazwy - termostat nieaktywny). W przypadku cechy PageType ustawionej na <i>Buttons/FreeDraw</i> cecha pozostaje pusta
Object_3_Id	ID obiektu termostatu lub numer przycisku w zależności od typu strony, np.: Dla strony typu <i>Thermostats</i> : - w przypadku termostatu na lokalnym CLU: THE1325 - w przypadku termostatu na zdalnym CLU: CLU220000001->THE4321 W przypadku cechy PageType ustawionej na <i>Buttons/FreeDraw</i> należy wpisać numer przycisku (1..16)
Object_3_Name	Nazwa termostatu wyświetlana na stronie Smart Panelu. Dotyczy tylko strony <i>Thermostats</i> (brak nazwy - termostat nieaktywny). W przypadku cechy PageType ustawionej na <i>Buttons/FreeDraw</i> cecha pozostaje pusta
Object_4_Id	ID obiektu termostatu lub numer przycisku w zależności od typu strony, np.: Dla strony typu <i>Thermostats</i> : - w przypadku termostatu na lokalnym CLU: THE1325 - w przypadku termostatu na zdalnym CLU: CLU220000001->THE4321 W przypadku cechy PageType ustawionej na <i>Buttons/FreeDraw</i> należy wpisać numer przycisku (1..16)

Nazwa	Opis
<code>Object_4_Name</code>	Nazwa termostatu wyświetlana na stronie Smart Panelu. Dotyczy tylko strony <i>Thermostats</i> (brak nazwy - termostat nieaktywny). W przypadku cechy <code>PageType</code> ustawionej na <i>Buttons/FreeDraw</i> cecha pozostaje pusta

METODY

Nazwa	Opis
<code>SetPageType</code>	Ustawia typ strony wyświetlanej na Smart Panelu
<code>SetPageName</code>	Ustawia nazwę strony/nazwę ikony wyświetlanej na Smart Panelu (przy przechodzeniu pomiędzy stronami)
<code>SetObject_1_Id</code>	Ustawia wartość <code>Object_1_Id</code>
<code>SetObject_1_Name</code>	Ustawia wartość <code>Object_1_Name</code>
<code>SetObject_2_Id</code>	Ustawia wartość <code>Object_2_Id</code>
<code>SetObject_2_Name</code>	Ustawia wartość <code>Object_2_Name</code>
<code>SetObject_3_Id</code>	Ustawia wartość <code>Object_3_Id</code>
<code>SetObject_3_Name</code>	Ustawia wartość <code>Object_3_Name</code>
<code>SetObject_4_Id</code>	Ustawia wartość <code>Object_4_Id</code>
<code>SetObject_4_Name</code>	Ustawia wartość <code>Object_4_Name</code>

ZDARZENIA

Nazwa	Opis
<code>OnPageOpen</code>	Zdarzenie wywoływane po otwarciu strony
<code>OnPageClose</code>	Zdarzenie wywoływane po zamknięciu strony
<code>OnDraw</code>	Zdarzenie sygnalizujące potrzebę przerysowania. Generowanie wyłącznie w trybie pracy <i>freedraw</i> , po wejściu na daną stronę lub w momencie wywołania metody <code>Draw</code> i wybudzeniu ekranu

D. Czujniki temperatury i oświetlenia

CECHY

Nazwa	Opis
Threshold	Wielkość histerezy (dokładność 0.1°C / 0.1 %) określająca czułość, przy której następuje wygenerowanie zdarzeń: <code>OnChange</code> , <code>OnLowerValue</code> , <code>OnRaiseValue</code>
Sensitivity	Okres (w ms), w którym próbkowane wartości są uśredniane
MinValue	Minimalna wartość cechy <code>value</code> , której przekroczenie wywołuje zdarzenie <code>OnOutOfRange</code>
MaxValue	Maksymalna wartość cechy <code>value</code> , której przekroczenie wywołuje zdarzenie <code>OnOutOfRange</code>
Value	Wartość wejścia: dla czujnika temperatury od 0.0 do 45.0°C lub dla czujnika światła 0 – 100%

ZDARZENIA

Nazwa	Opis
<code>OnChange</code>	Zdarzenie wywoływane w przypadku zmiany wartości cechy <code>Value</code>
<code>OnRaiseValue</code>	Zdarzenie wywoływane przy zmianie wartości na wyższą (zbocze narastające)
<code>OnLowerValue</code>	Zdarzenie wywoływane przy zmianie wartości na niższą (zbocze opadające)
<code>OnOutOfRange</code>	Zdarzenie wywoływane, gdy wartość na wejściu znajduje się poza wyznaczonym zakresem (<code>MinValue</code> ; <code>MaxValue</code>)

5.2. Tworzenie konfiguracji sensora gestów

Chcąc utworzyć konfigurację dla sensora gestów należy:

- Otworzyć - przez dwuklik - obiekt *Panel*;
- Przejść do zakładki *Zdarzenia*;
- Przypisać metody do zdarzeń `OnGestureUp`, `OnGestureDown`, `OnGestureLeft`, `OnGestureRight` (klikając w "+" po prawej stronie każdej z metod):

Właściwości obiektu

Nazwa: Źródło/Odbiornik:

Id: Numer seryjny:

Typ:

Nazwa zdarzenia	Przypisane komendy	Dodaj komendę
OnGestureUp	<input type="text" value="CLU220000260->x200000319_DOUT4->SwitchOn(0)"/> <input type="button" value="Przypisz komendę"/> ✖	<input type="button" value="Dodaj komendę"/> +
OnGestureDown	<input type="text" value="CLU220000260->x200000319_DOUT4->SwitchOff(0)"/> <input type="button" value="Przypisz komendę"/> ✖	<input type="button" value="Dodaj komendę"/> +
OnGestureLeft	<input type="text" value="CLU220000260->x270000003_BUTTON1->LedSwitchOn()"/> <input type="button" value="Przypisz komendę"/> ✖	<input type="button" value="Dodaj komendę"/> +
OnGestureRight	<input type="text" value="CLU220000260->x270000003_BUTTON1->LedSwitchOff()"/> <input type="button" value="Przypisz komendę"/> ✖	<input type="button" value="Dodaj komendę"/> +
OnProximityDetect		<input type="button" value="Dodaj komendę"/> +
OnPageChange		<input type="button" value="Dodaj komendę"/> +

UWAGA!

W przypadku konfiguracji zawierającej konfigurację stron (Buttons/FreeDraw/Thermostats) metody przypisane do zdarzeń OnGestureLeft oraz OnGestureRight nie będą wykonywane. Jest to związane z predefiniowaną funkcjonalnością polegającą na przechodzeniu pomiędzy stronami. Istnieje możliwość zmiany sposobu przewijania stron. W tym celu należy zmienić ustawienie cechy `PageControlMode` na wartość `Command`. Po wykonaniu tej czynności metody przypisane do zdarzeń będą wykonywane.

PageControlMode	1	<input type="text" value="Command"/>	0,1
-----------------	---	--------------------------------------	-----

Możliwa jest również podmiana domyślnie wyświetlanych ikon przy wykonywaniu gestów - w tym celu należy przejść do zakładki *Cechy wbudowane* i wpisać nazwy żądanych ikon bez rozszerzenia *.bmp*:

Właściwości obiektu

Nazwa: Źródło/Odbiornik:

Id: Numer seryjny:

Typ:

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
GestureIconUp	up	<input type="text" value="lamp3on"/>	.bmp	[0-9]
GestureIconDown	down	<input type="text" value="lamp3off"/>	.bmp	[0-9]
GestureIconLeft	left	<input type="text" value="shclosed"/>	.bmp	[0-9]
GestureIconRight	right	<input type="text" value="shopen"/>	.bmp	[0-9]
ProximitySens	5	<input type="text" value="5"/>		[2-100]
ProximityTimeout	5000	<input type="text" value="5000"/>	ms	[1000-60000]
ProximityValue	165		-	
BuzzerValue	1	<input type="button" value="On"/>		0,1
GestureMode	3	<input type="button" value="Vert+ Horiz"/>		0,1,2,3
GestureSens	2	<input type="button" value="Mid"/>		1,2,3
PageNr	0	<input type="text" value="1"/>		
PageDisplayMode	0	<input type="button" value="ShowImmediately"/>		0,1,2
ButtonsLEDMode	1	<input type="button" value="LocationLedON"/>		0,1,2
PageControlMode	1	<input type="button" value="Gesture/Command"/>		0,1
GestureDisplayMode	1	<input type="button" value="On"/>		0,1

Auto odświeżanie

Użycie ikon będzie możliwe, gdy będą wgrane na kartę microSD z rozszerzeniem *.bmp*.

Dodatkowo od wersji 04.03.04.1910 istnieje możliwość wyboru orientacji rozpoznawanych gestów oraz ich czułości. W tym celu należy przejść do zakładki *Cechy wbudowane* i wybrać żądaną orientację i czułość rozpoznawania gestów:

Właściwości obiektu

Nazwa: Źródło/Odbiornik:

Id: Numer seryjny: |

Typ:

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
GestureIconUp	up	<input type="text" value="lamp3on"/>	.bmp	[0-9]
GestureIconDown	down	<input type="text" value="lamp3off"/>	.bmp	[0-9]
GestureIconLeft	left	<input type="text" value="shclosed"/>	.bmp	[0-9]
GestureIconRight	right	<input type="text" value="shopen"/>	.bmp	[0-9]
ProximitySens	5	<input type="text" value="5"/>		[2-100]
ProximityTimeout	5000	<input type="text" value="5000"/>	ms	[1000-60000]
ProximityValue	160		-	
BuzzerValue	1	<input type="button" value="On"/>		0,1
GestureMode	3	<input type="button" value="Vert+Horiz"/>		0,1,2,3
GestureSens	2	<input type="button" value="Mid"/>		1,2,3
PageNr	0	<input type="text" value="1"/>		
PageDisplayMode	0	<input type="button" value="ShowImmediately"/>		0,1,2
ButtonsLEDMode	1	<input type="button" value="LocationLedON"/>		0,1,2
PageControlMode	1	<input type="button" value="Command"/>		0,1
GestureDisplayMode	1	<input type="button" value="On"/>		0,1

Auto odświeżanie

Cechy wbudowane za pośrednictwem których można dokonywać wyboru orientacji i czułości to:

- **GestureMode** - możliwa zmiana kierunku detekcji gestu:
 - Off - gesty nie są rozpoznawane;
 - Vertical - rozpoznawane są wyłącznie gesty w górę oraz w dół;
 - Horizontal - rozpoznawane są wyłącznie gesty w lewo oraz w prawo;
 - Vert+Horiz - rozpoznawane są gesty zarówno w górę, w dół, jak również w lewo oraz w prawo.
- **GestureSens** - możliwa zmiana czułości wykrywania gestów:
 - Low - gest wykonywany blisko urządzenia w sposób dokładny;
 - Mid - gest wykonywany zarówno blisko urządzenia, jak również z niewielkiej odległości;
 - High - gest wykonywany z dalszej odległości, istnieje możliwość detekcji błędnego gestu.

Powyższe cechy można ustawiać zarówno w zakładce *Cechy wbudowane*, jak również przy pomocy metod: `SetGestureIconUp`, `SetGestureIconDown`, `SetGestureIconLeft`, `SetGestureIconRight`, `SetGestureMode`, `SetGestureSens` (w metodach obiektu `Panel`).

- Zatwierdzić okno konfiguracji przyciskiem *OK*;
- Wysłać konfigurację do *CLU Z-Wave*.

5.3. Konfiguracja czujnika zbliżenia

W celu ustawienia parametrów czujnika zbliżenia należy:

- Otworzyć - przez dwuklik - obiekt `Panel`;
- Przejść do zakładki *Cechy wbudowane*, gdzie znajdują się 3 cechy odnoszące się do czujnika zbliżenia:
 - `ProximitySens` - określa czułość sensora;
 - `ProximityTimeout` - definiuje czas, po jakim wyświetlacz zostaje wygaszony, gdy nie zostanie wykryty ruch;
 - `ProximityValue` - zwraca przybliżoną odległość w centymetrach od panelu do obiektu;

Właściwości obiektu

Nazwa: Źródło/Odbiornik:

Id: Numer seryjny:

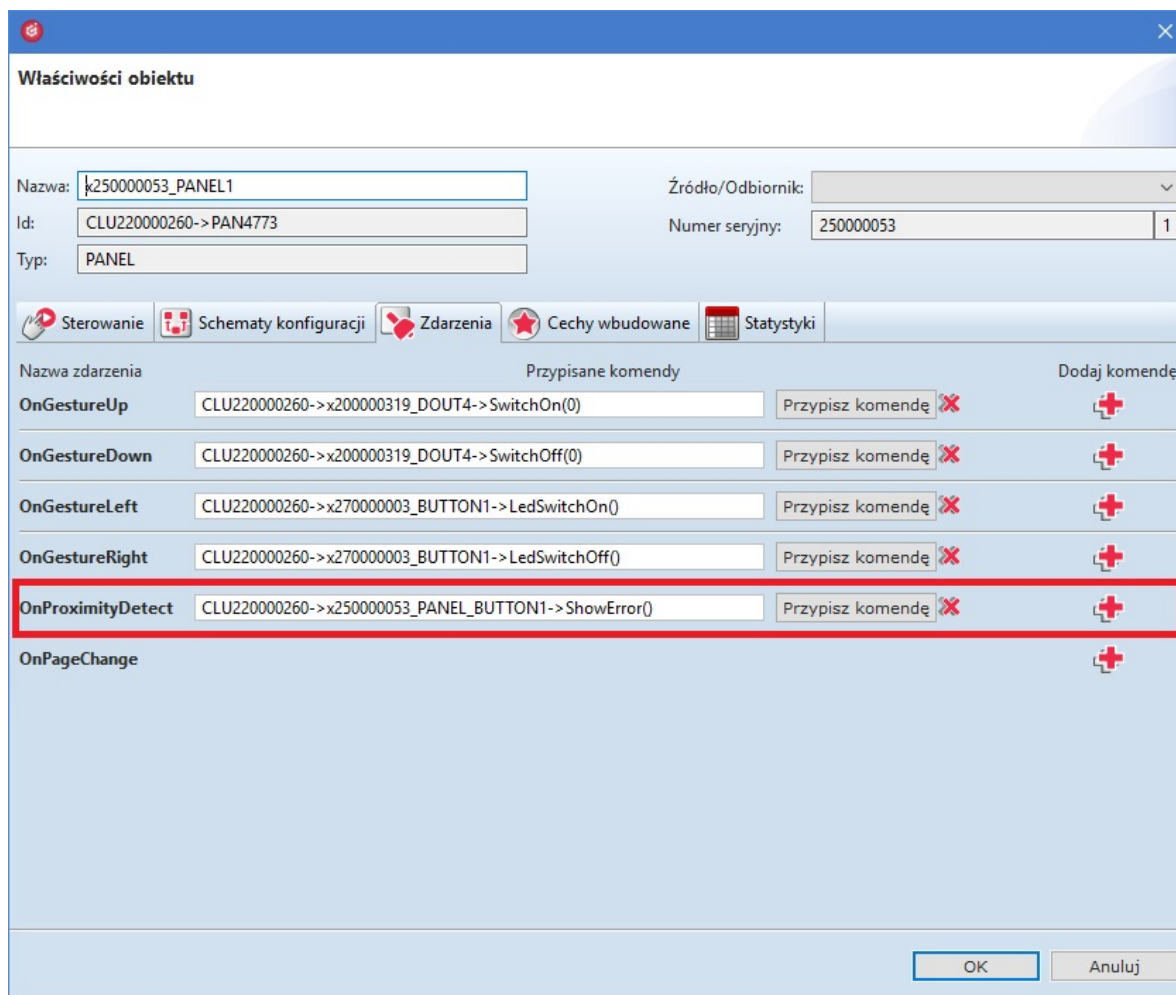
Typ:

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
GestureIconUp	up	<input type="text" value="up"/>	.bmp	[0-9]
GestureIconDown	down	<input type="text" value="down"/>	.bmp	[0-9]
GestureIconLeft	left	<input type="text" value="left"/>	.bmp	[0-9]
GestureIconRight	right	<input type="text" value="right"/>	.bmp	[0-9]
ProximitySens	5	<input type="text" value="5"/>		[2-100]
ProximityTimeout	5000	<input type="text" value="5000"/>	ms	[1000-60000]
ProximityValue	164		-	
BuzzerValue	1	<input type="button" value="On"/>		0,1
GestureMode	3	<input type="button" value="Vert+Horiz"/>		0,1,2,3
GestureSens	2	<input type="button" value="Mid"/>		1,2,3
PageNr	0	<input type="text" value="1"/>		
PageDisplayMode	0	<input type="button" value="ShowImmediately"/>		0,1,2
ButtonsLEDMode	1	<input type="button" value="LocationLedON"/>		0,1,2
PageControlMode	1	<input type="button" value="Gesture/Command"/>		0,1
GestureDisplayMode	1	<input type="button" value="On"/>		0,1

Auto odświeżanie

Powyższe cechy można ustawiać zarówno w zakładce *Cechy wbudowane*, jak również przy pomocy metod: `SetProximitySens` oraz `SetProximityTimeout` (w metodach obiektu Panel).

- Reakcja czujnika zbliżenia generuje zdarzenie `OnProximityDetect`, do którego można dodać dodatkowe metody:



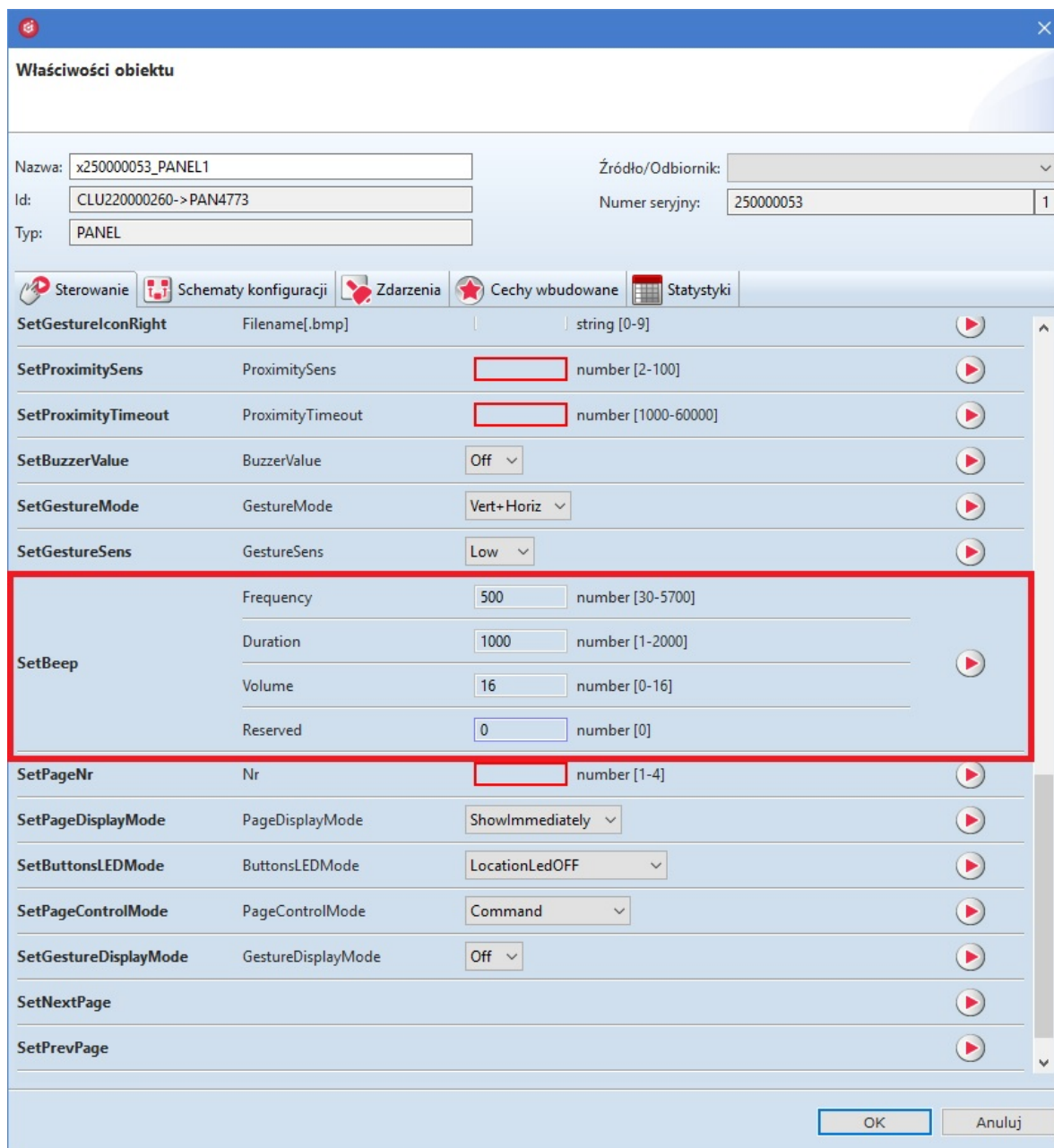
- Wysłać konfigurację do CLU Z-Wave.

5.4. Obiekt Panel - nowa funkcjonalność

W najnowszej wersji modułu Smart Panel (od 04.03.04.1910) w obiekcie Panel wprowadzona została nowa funkcjonalność umożliwiająca między innymi:

- wygenerowanie dźwięku;
- zarządzanie podświetleniem LED przycisków;
- możliwość włączenia/wyłączenia informowania o wykrytym geście;
- mechanizm zarządzania stronami, który zostanie opisany dokładnie w kolejnym podrozdziale.

Pierwszą z wprowadzonych nowości jest możliwość wygenerowania dźwięku o zadanej częstotliwości, długości oraz głośności. Do tego celu służy metoda `SetBeep`:



Kolejną funkcjonalnością dostępną od najnowszej wersji oprogramowania jest możliwość lokalizacji przycisków za pomocą słabego światła LED. W tym celu należy przejść do zakładki *Cechy wbudowane* i ustawić żadaną wartość cechy `ButtonsLEDMode`:

- `LocationLedOFF` - przyciski na module Smart Panel są nie podświetlone;
- `LocationLedOn` - przyciski na module SmartPanel są lekko podświetlone;
- `LocationLedforActive` - podświetlone zostają wyłącznie przyciski, które znajduje się w jednym z dwóch trybów pracy *Monostable/Bistable*. Jeżeli przycisk znajduje się w trybie *Locked*, jego dioda pozostaje zgaszona.

Oprócz możliwości zarządzania podświetleniem przycisków, istnieje możliwość włączenia/wyłączenia informowania o wykryciu gestu. W tym celu w zakładce *Cechy wbudowane* odnaleźć cechę `GestureDisplayMode`, ustawiając żadaną wartość:

- `Off` - informacja o wykryciu gestu nie jest wyświetlana na ekranie modułu;
- `On` - informacja o wykryciu gestu jest wyświetlana na ekranie modułu.

Powyższe cechy wbudowane można ustawiać również przy pomocy metod: `SetButtonsLEDMode` oraz `SetGestureDisplayMode`.

5.5. Obiekt Panel - mechanizm zarządzania stronami

Smart Panel w wersji v4 wprowadza nowy mechanizm zarządzania stronami. W jego skład wchodzi cechy, metody oraz zdarzenie, które umieszczone zostały w obiekcie Panel:

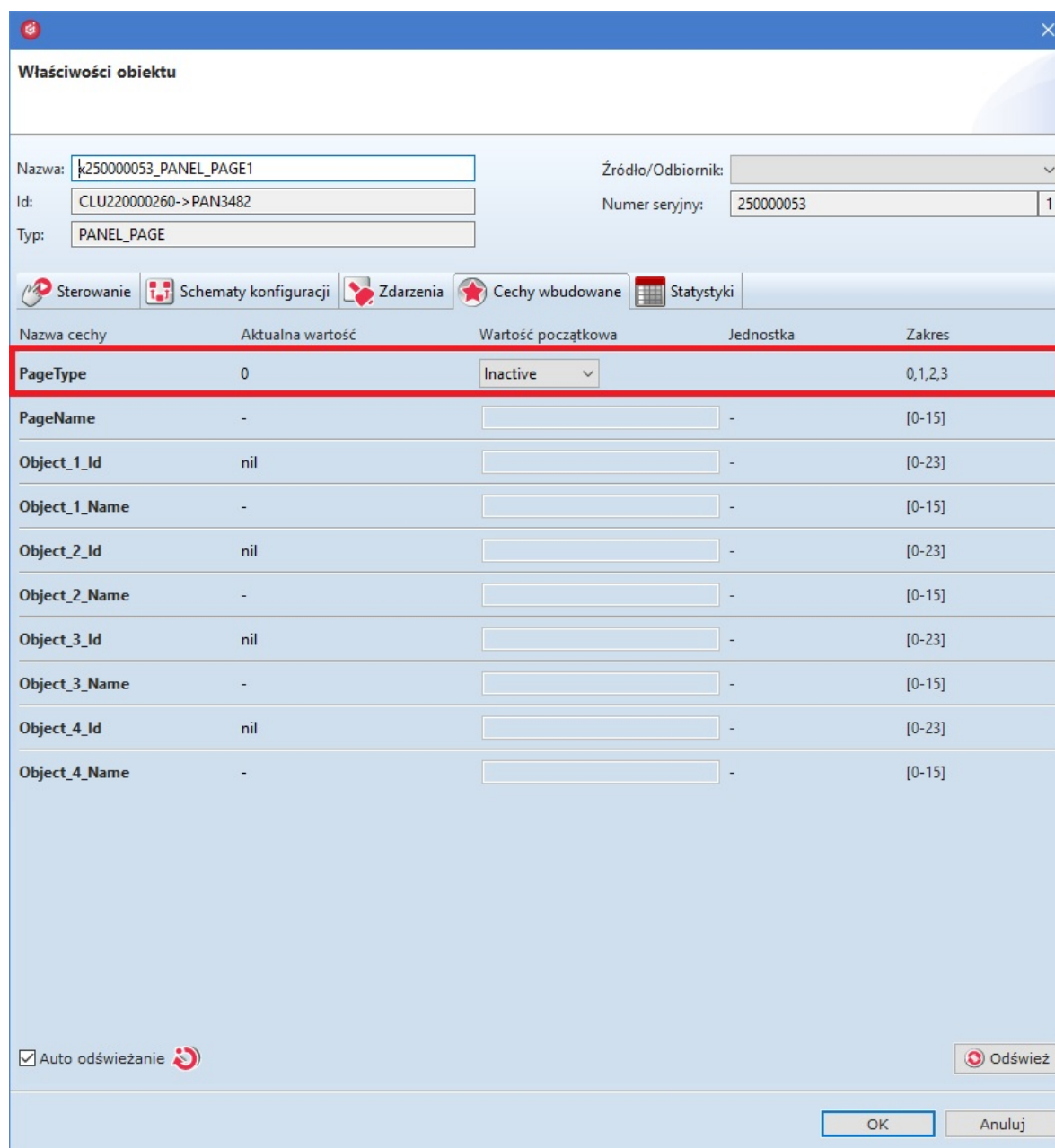
- Metody/Cechy:
 - `SetPageNr/PageNr` - za pomocą niniejszej metody/cechy istnieje możliwość bezpośredniego przejścia pomiędzy większą ilością stron jednocześnie. Podając w parametrze numer strony, a następnie wywołując metodę, na ekranie wyświetlona zostanie żądana strona (możliwa potrzeba wybudzenia ekranu);
 - `SetPageDisplayMode/PageDisplayMode` - za pośrednictwem metody/cechy istnieje możliwość ustawienia sposobu przechodzenia pomiędzy stronami. Do wyboru jest trzy tryby:
 - ShowImmediately (0) - przejście pomiędzy stronami odbywa się natychmiastowo, nie jest poprzedzone wyświetleniem komunikatu/ikony/nazwy;
 - ShowIconOrName (1) - przejście pomiędzy stronami poprzedza wyświetlenie ikony lub nazwy wprowadzonej w cechę `PageName`;
 - ShowGesture (2) - przejście pomiędzy stronami poprzedza wyświetlenie ikony wprowadzonej w cechę `GestureIconLeft` lub `GestureIconRight`, w zależności od wykonanego gestu;
 - `SetPageControlMode/PageControlMode` - przy użyciu metody/cechy istnieje możliwość zmiany źródła, za pomocą którego dokonywana jest zmiana strony:
 - Command (0) - przejście do poprzedniej/następnej strony następuje wyłącznie za pomocą metod `SetPrevPage` oraz `SetNextPage`. Dodatkowo gesty w lewo oraz w prawo stają się aktywne, co oznacza, iż istnieje możliwość przypisania do zdarzeń `OnGestureLeft` oraz `OnGestureRight` akcji;
 - Gesture/Command (1) - przejście do poprzedniej/następnej strony możliwe jest za pomocą gestów w lewo oraz w prawo, jak również z wykorzystaniem metod `SetPrevPage` oraz `SetNextPage`. W przypadku ustawienia takiej wartości cechy, gesty w lewo oraz w prawo posiadają predefiniowaną funkcjonalność, która ma wyższy priorytet nad akcjami przypisanymi do zdarzeń `OnGestureLeft` oraz `OnGestureRight`. Oznacza to, iż akcje przypisane do tych zdarzeń nie będą wykonywane;
 - `SetNextPage` - metoda umożliwia przejście do następnej strony w konfiguracji;
 - `SetPrevPage` - metoda umożliwia przejście do poprzedniej strony w konfiguracji;
 - `Draw` - metoda służąca do wygenerowania zdarzenia `OnDraw` w momencie kiedy OLED jest aktywny;
- Zdarzenie:
 - `OnPageChange` - zdarzenie generowane w momencie przejścia pomiędzy stronami

UWAGA!

Mechanizm zarządzania stronami dostępny jest wyłącznie dla konfiguracji stron wykonanych za pośrednictwem obiektów Panel_Page (Buttons/FreeDraw/Thermostats). W przypadku konfiguracji, która została stworzona w sposób dotychczasowy (podrozdział 4.5), powyższe cechy, metody oraz zdarzenie są ignorowane.

5.6. Kompatybilność wstecz

Rozpoczynając pracę z nową wersją modułu Smart Panel, urządzenie znajduje się domyślnej konfiguracji, która jest kompatybilna wstecz. Wszystkie cztery obiekty Panel_Page posiadają ustawioną cechę wbudowaną `PageType` na wartość `Inactive`. Umożliwia to pracę z panelem w taki sam sposób jak dotychczas (w wersji v3). Dostępne są wyłącznie pierwsze cztery przyciski widoczne na liście obiektów. Przyciski od 5 do 16 pomimo możliwości konfiguracji są nieaktywne. Tworzenie konfiguracji wielu stron odbywa się zgodnie z procedurą opisaną w podrozdziale 4.5.



Właściwości obiektu

Nazwa: Źródło/Odbiornik:

Id: Numer seryjny:

Typ:

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
PageType	0	Inactive		0,1,2,3
PageName	-	<input type="text"/>	-	[0-15]
Object_1_Id	nil	<input type="text"/>	-	[0-23]
Object_1_Name	-	<input type="text"/>	-	[0-15]
Object_2_Id	nil	<input type="text"/>	-	[0-23]
Object_2_Name	-	<input type="text"/>	-	[0-15]
Object_3_Id	nil	<input type="text"/>	-	[0-23]
Object_3_Name	-	<input type="text"/>	-	[0-15]
Object_4_Id	nil	<input type="text"/>	-	[0-23]
Object_4_Name	-	<input type="text"/>	-	[0-15]

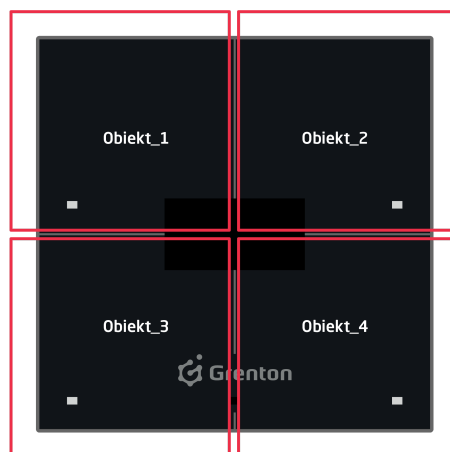
Auto odświeżanie

5.7. Tworzenie konfiguracji z wykorzystaniem obiektu strony Buttons

W trybie pracy *Buttons* do dyspozycji jest 4 fizyczne przyciski dotykowe i do 16 przycisków wirtualnych rozłożonych na 4 stronach, z których każdy może realizować niezależne funkcje. Istnieje również możliwość łączenia/scalania 2,3,4 obiektów w jeden przycisk (szerzej opisane w podrozdziale XII.5.10).

UWAGA!

W trybie pracy *Buttons* rysowanie treści na wyświetlaczu jest zablokowane.



Strona typu „Buttons/FreeDraw”

Tworzenie konfiguracji panelu obsługującej stronę lub strony typu *Buttons* najlepiej rozpocząć od konfiguracji przycisków, które mają zostać wykorzystane. W celu ich parametryzacji należy:

- Otworzyć obiekt *PANEL_BUTTONX* (gdzie X to numer jednego z 16 przycisków) poprzez dwukrotne kliknięcie na liście modułów;
- Przejście do zakładki Zdarzenia;
- Skonfigurować działanie przycisku poprzez przypisanie metod do określonych zdarzeń (klikając "+" po prawej stronie okna):

Właściwości obiektu

Nazwa: Źródło/Odbiornik:

Id: Numer seryjny: |

Typ:

Nazwa zdarzenia	Przypisane komendy	Dodaj komendę
OnChange	<input type="text" value="CLU220000260->x270000003_BUTTON1->ShowOK()"/> <input type="button" value="Przypisz komendę"/> ✖	<input type="button" value="Dodaj komendę"/> +
OnSwitchOn	<input type="text" value="CLU220000260->x200000319_DOUT1->SwitchOn(0)"/> <input type="button" value="Przypisz komendę"/> ✖	<input type="button" value="Dodaj komendę"/> +
OnSwitchOff	<input type="text" value="CLU220000260->x200000319_DOUT1->SwitchOff(0)"/> <input type="button" value="Przypisz komendę"/> ✖	<input type="button" value="Dodaj komendę"/> +
OnShortPress		<input type="button" value="Dodaj komendę"/> +
OnLongPress		<input type="button" value="Dodaj komendę"/> +
OnHold		<input type="button" value="Dodaj komendę"/> +
OnClick		<input type="button" value="Dodaj komendę"/> +

- Wybrać zakładkę *Cechy wbudowane* i zdefiniować obiekty wyświetlane na ekranie danego przycisku:
 - `Label` - cecha określająca tekst przypisany do danego przycisku;
 - `IconA` - cecha określająca nazwę ikony przypisanej do danego przycisku, gdy znajduje się w trybie *Monostable*, bądź dla tryb *Bistable* dla pozycji OFF;
 - `IconB` - cecha określająca nazwę ikony przypisanej do danego przycisku, gdy znajduje się w trybie *Bistable* w pozycji ON. Chcąc przypisać tę samą ikonę, ale o odwróconej kolorystyce należy nazwę piktogramu poprzedzić znakiem "~" (np. `~heaton`):

Właściwości obiektu

Nazwa: Źródło/Odbiornik:

Id: Numer seryjny:

Typ:

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
Mode	0	<input type="text" value="Monostable"/>		0,1,2
HoldDelay	1000	<input type="text" value="1000"/>	ms	[1-5000]
HoldInterval	100	<input type="text" value="50"/>	ms	[1-2000]
Value	0		bool	0,1
Label	Lampa 3	<input type="text" value="Lampa 3"/>	string	[0-15]
IconA	lamp3off	<input type="text" value="lamp3off"/>	string	[0-9]
IconB	~lamp3on	<input type="text" value="~lamp3on"/>	string	[0-9]

Auto odświeżanie

Powyższe cechy wbudowane można ustawić zarówno w zakładce *Cechy wbudowane*, jak również za pośrednictwem metod: `SetLabel`, `SetIconA`, `SetIconB`.

UWAGA!

Metoda `SetIconA` ma w systemie wyższy priorytet niż metoda `SetLabel`!

- Wysłać konfigurację do CLU Z-Wave.

Kolejnym krokiem w tworzeniu konfiguracji jest konfiguracja obiektów `Panel_Page` w zależności od ilości przycisków. Jeden obiekt `Panel_Page` obsługuje do 4 przycisków. W tym celu należy:

- Otworzyć obiekt `PANEL_PAGEX` (gdzie X to numer kolejnej strony) poprzez dwukrotne kliknięcie na liście modułów;
- Przejście do zakładki *Zdarzenia*;
- Skonfigurować działanie strony poprzez przypisanie metod do określonych zdarzeń (klikając w "+" po prawej stronie okna):

Właściwości obiektu

Nazwa: Źródło/Odbiornik:

Id: Numer seryjny: |

Typ:

Nazwa zdarzenia	Przypisane komendy	Dodaj komendę
OnPageOpen	<input type="text" value="CLU220000260->x270000003_BUTTON2->LedSwitchOn()"/> <input type="button" value="Przypisz komendę"/> ✖	<input type="button" value="Dodaj komendę"/> +
OnPageClose	<input type="text" value="CLU220000260->x270000003_BUTTON2->LedSwitchOff()"/> <input type="button" value="Przypisz komendę"/> ✖	<input type="button" value="Dodaj komendę"/> +
OnDraw		<input type="button" value="Dodaj komendę"/> +

UWAGA!

W przypadku typu strony *Buttons* zdarzenie `OnDraw` nie jest generowane.

- Wybrać zakładkę *Cechy wbudowane* i zdefiniować obsługiwany typ strony i powiązać obiekty strony z przyciskami:
 - `PageType` - cecha określająca typ strony, należy ustawić ją na wartość *Buttons (1)*;
 - `PageName` - cecha określająca nazwę strony lub ikonę, która będzie wyświetlana w momencie przechodzenia pomiędzy stronami (działa wyłącznie przy ustawionej cesze `PageDisplayMode` na wartość 1 (ShowIconOrName) w obiekcie Panel);
 - `Object_X_Id` - identyfikator/numer przycisku. W celu należy odczytać wartość znajdującą się w polu *Numer seryjny* obiektu *PANEL_BUTTONX*

Właściwości obiektu

Nazwa: Źródło/Odbiornik:

Id: Numer seryjny:

Typ:

- o `Object_X_Name` - nazwa termostatu. W przypadku typu strony *Buttons* cechę należy pozostawić pustą;

Właściwości obiektu

Nazwa: Źródło/Odbiornik:

Id: Numer seryjny:

Typ:

Sterowanie
 Schematy konfiguracji
 Zdarzenia
 Cechy wbudowane
 Statystyki

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
PageType	1	<input type="text" value="Buttons"/>		0,1,2,3
PageName	Strona 1	<input type="text" value="Strona 1"/>	-	[0-15]
Object_1_Id	1	<input type="text" value="1"/>	-	[0-23]
Object_1_Name	-	<input type="text" value=""/>	-	[0-15]
Object_2_Id	2	<input type="text" value="2"/>	-	[0-23]
Object_2_Name	-	<input type="text" value=""/>	-	[0-15]
Object_3_Id	7	<input type="text" value="7"/>	-	[0-23]
Object_3_Name	-	<input type="text" value=""/>	-	[0-15]
Object_4_Id	8	<input type="text" value="8"/>	-	[0-23]
Object_4_Name	-	<input type="text" value=""/>	-	[0-15]

Auto odświeżanie

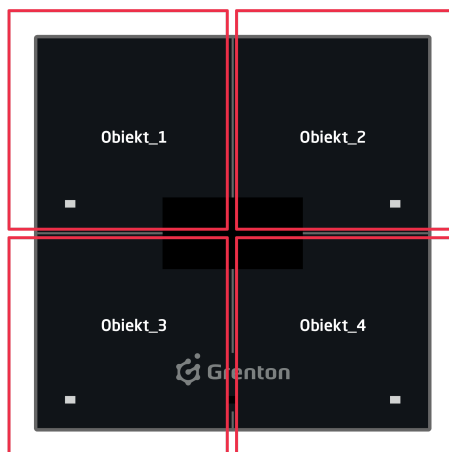
UWAGA!

Wysłanie konfiguracji wyłącznie ze zdefiniowanym typem strony, bez ustawionych powiązań obiektów z przyciskami wiąże się z uruchomieniem trybu pracy panelu jako *Buttons*. Jednakże przyciski na module będą nieaktywne. Ma to związek z nieuzupełnieniem cech `Object_X_Id`.

- Wysłać konfigurację do CLU Z-Wave.

5.8. Tworzenie konfiguracji z wykorzystaniem obiektu strony FreeDraw

W trybie pracy *FreeDraw* podobnie jak w przypadku *Buttons* do dyspozycji są 4 fizyczne przyciski dotykowe i do 16 przycisków wirtualnych rozłożonych na 4 stronach, z których każdy może realizować niezależne funkcje. Istnieje również możliwość łączenia/scalania obiektów w jeden przycisk. Wyświetlacz OLED działa w trybie *FreeDraw*, czyli jest w pełni dostępny dla skryptów LUA użytkownika. Stworzony został również mechanizm rysowania, w którym skrypty rysujące wywoływane są zdarzeniem `OnDraw` generowanym przez panel w momencie, gdy jest to konieczne. System wywołuje metodę `Draw` w momencie, kiedy rysowany na module kontent uległ zmianie.



Strona typu „Buttons/FreeDraw”

A. Ogólne zasady tworzenia konfiguracji

Tworzenie konfiguracji panelu obsługującej stronę lub strony typu *FreeDraw* najlepiej rozpocząć od konfiguracji przycisków, które mają zostać wykorzystane. Ich parametryzację opisano w poprzednim podrozdziale.

Następnym krokiem w tworzeniu konfiguracji powinno być stworzenie skryptów rysujących treść na wyświetlaczu modułu Smart Panel. Ich tworzenie jest analogiczne jak w wersji v3 modułu Smart Panel (patrz rozdział XII.4).

Przykład skryptu rysującego treść na wyświetlaczu (*Strona1*):

```
CLU220000260->x250000053_PANEL1->ClearScreen()
CLU220000260->x250000053_PANEL1->PrintText(15,10,"kuchnia [°C]:",2)
CLU220000260->x250000053_PANEL1->PrintFloat(80,38,CLU220000260->
->x240000659_PANELSENSTEMP1->value,1,2)
CLU220000260->x250000053_PANEL1->DrawLine(0,32,127,32,1)
CLU220000260->x250000053_PANEL1->DrawPoint(0,0,1)
CLU220000260->x250000053_PANEL1->DrawLine(70,32,70,63,1)
CLU220000260->x250000053_PANEL1->PrintText(15,40,CLU220000260->Time,1)
CLU220000260->x250000053_PANEL1->DisplayContent()
```

UWAGA!

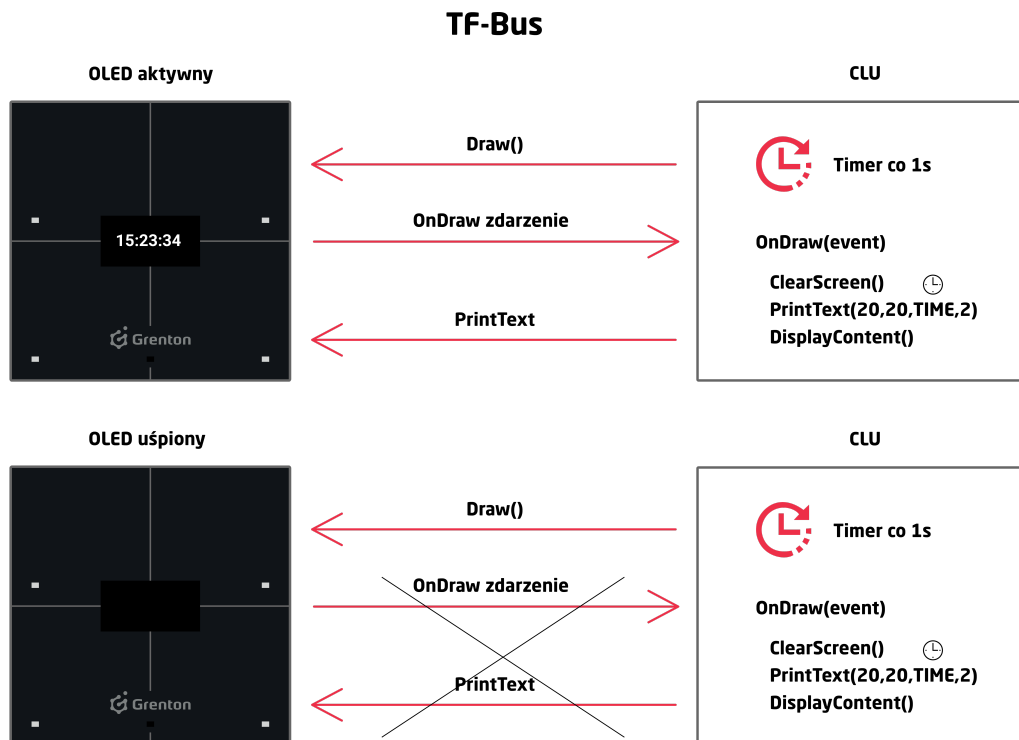
W mechanizmie rysowania wprowadzone została ograniczenie. CLU Z-Wave oczekuje 2 sekundy na zakończenie rysowania metodą `DisplayContent`. W przeciwnym wypadku na ekranie wyświetlony zostanie komunikat o następującej treści:

"page: PageName

free draw

! TIMEOUT !"

Poniższy rysunek przedstawia obecny mechanizm rysowania.



Kolejnym krokiem w tworzeniu konfiguracji jest konfiguracja obiektów Panel_Page w zależności od ilości przycisków. Jeden obiekt Panel_Page obsługuje do 4 przycisków. W tym celu należy:

- Otworzyć obiekt *PANEL_PAGEX* (gdzie X to numer kolejnej strony) poprzez dwukrotne kliknięcie na liście modułów;
- Przejście do zakładki *Zdarzenia*;
- Skonfigurować działanie strony poprzez przypisanie metod do określonych zdarzeń (klikając w "+" po prawej stronie okna):

Właściwości obiektu

Nazwa: Źródło/Odbiornik:

Id: Numer seryjny:

Typ:

Nazwa zdarzenia	Przypisane komendy	Dodaj komendę
OnPageOpen	<input type="text" value="CLU220000260->x270000003_BUTTON2->LedSwitchOn()"/>	Przypisz komendę <input type="button" value="✘"/> <input type="button" value="✚"/>
OnPageClose	<input type="text" value="CLU220000260->x270000003_BUTTON2->LedSwitchOff()"/>	Przypisz komendę <input type="button" value="✘"/> <input type="button" value="✚"/>
OnDraw	<input type="text" value="CLU220000260->Strona1()"/>	Przypisz komendę <input type="button" value="✘"/> <input type="button" value="✚"/>

UWAGA!

W przypadku typu strony *FreeDraw* należy uzupełnić zdarzenie `OnDraw`.

- Wybrać zakładkę *Cechy wbudowane* i zdefiniować obsługiwany typ strony i powiązać obiekty strony z przyciskami:
 - `PageType` - cecha określająca typ strony, należy ustawić ją na wartość *FreeDraw (3)*;
 - `PageName` - cecha określająca nazwę strony lub ikonę, która będzie wyświetlana w momencie przechodzenia pomiędzy stronami (działa wyłącznie przy ustawionej cieszce `PageDisplayMode` na wartość 1 (ShowIconOrName) w obiekcie Panel);
 - `Object_X_Id` - identyfikator/numer przycisku. W tym celu należy odczytać wartość znajdującą się w polu *Numer seryjny* obiektu *PANEL_BUTTONX*

Właściwości obiektu

Nazwa: Źródło/Odbiornik:

Id: Numer seryjny:

Typ:

- `Object_X_Name` - nazwa termostatu. W przypadku typu strony *FreeDraw* cechę należy pozostawić pustą;

Właściwości obiektu

Nazwa: Źródło/Odbiornik:

Id: Numer seryjny:

Typ:

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
PageType	3	<input type="text" value="FreeDraw"/>		0,1,2,3
PageName	Strona 1	<input type="text" value="Strona 1"/>	-	[0-15]
Object_1_Id	1	<input type="text" value="1"/>	-	[0-23]
Object_1_Name	-	<input type="text"/>	-	[0-15]
Object_2_Id	0	<input type="text"/>	-	[0-23]
Object_2_Name	-	<input type="text"/>	-	[0-15]
Object_3_Id	1	<input type="text" value="1"/>	-	[0-23]
Object_3_Name	-	<input type="text"/>	-	[0-15]
Object_4_Id	1	<input type="text" value="1"/>	-	[0-23]
Object_4_Name	-	<input type="text"/>	-	[0-15]

Auto odświeżanie

UWAGA!

Wysłanie konfiguracji wyłącznie ze zdefiniowanym typem strony, bez ustawionych powiązań obiektów z przyciskami wiąże się z uruchomieniem trybu pracy panelu jako *FreeDraw*. Jednakże przyciski na module będą nieaktywne. Ma to związek z nie uzupełnieniem cech `Object_X_Id`.

- Wysłać konfigurację do CLU Z-Wave.

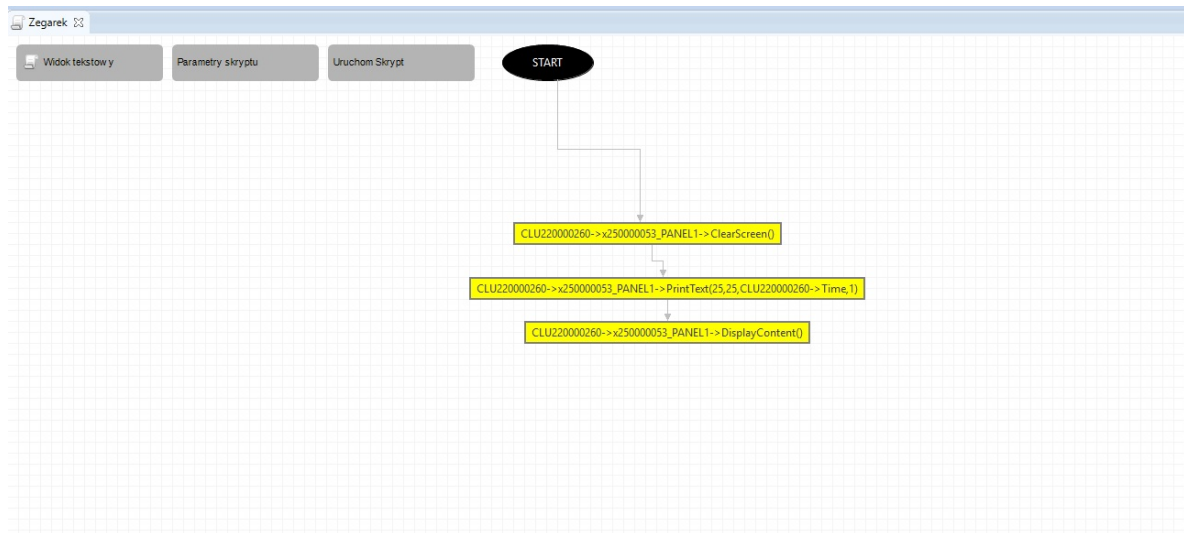
UWAGA!

Istnieje możliwość nadpisania treści wyświetlacza poprzez wywołanie metod rysujących z poziomu aplikacji Object Manager lub za pośrednictwem innych skryptów, które nie są przypisane do zdarzenia `OnDraw`. Jednakże, nadpisana treść zostanie wyczyszczona w momencie przejścia do innej strony lub wywołaniu metody `Draw` i wybudzeniu ekranu.

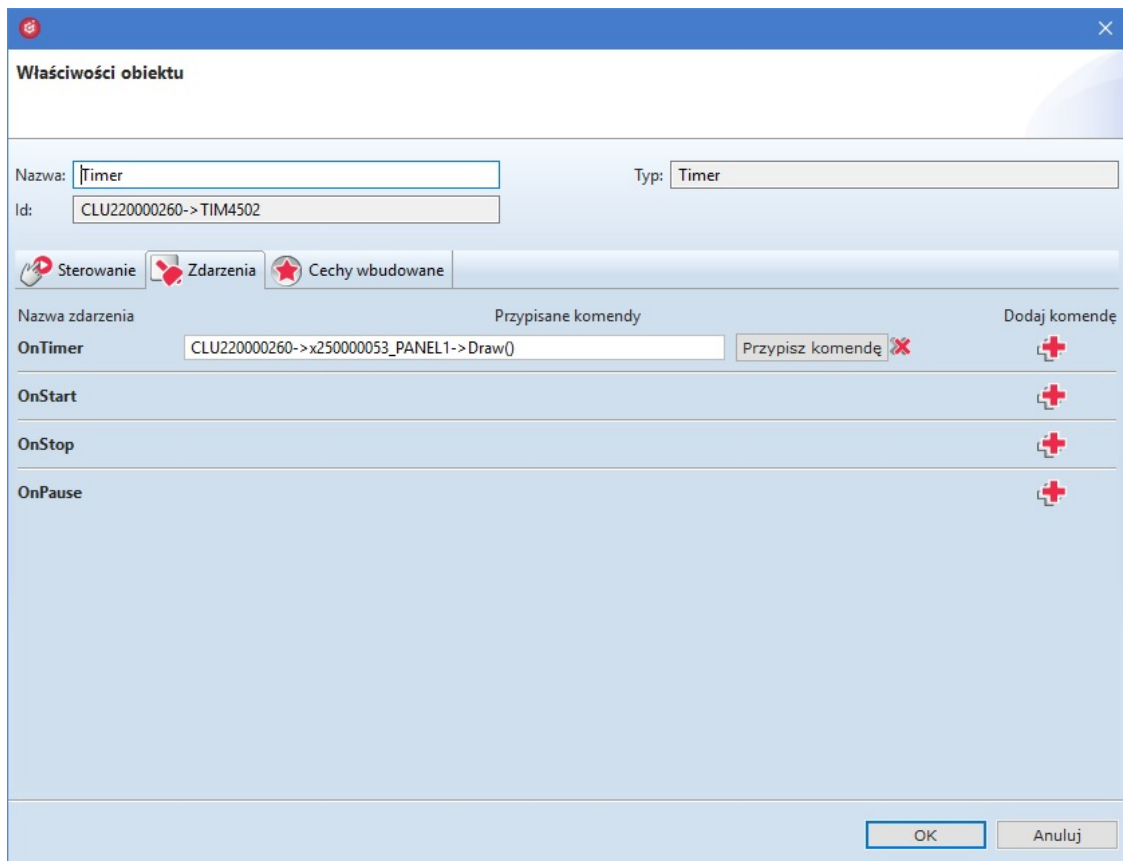
B. Konfiguracja strony jako zegarka

Chcąc skonfigurować stronę jako zegarek należy:

- Stworzyć skrypt wyświetlający aktualny czas (*Zegarek*);



- Stworzyć obiekt wirtualny Timer:
 - Przejść do zakładki *Zdarzenia*;
 - Skonfigurować działanie obiektu wirtualnego poprzez przypisanie metody `Draw` obiektu *Panel* do zdarzenia `OnTimer`:



- Wybrać zakładkę *Cechy wbudowane* i zdefiniować parametry konfiguracyjne obiektu:


Właściwości obiektu

Nazwa: Typ:

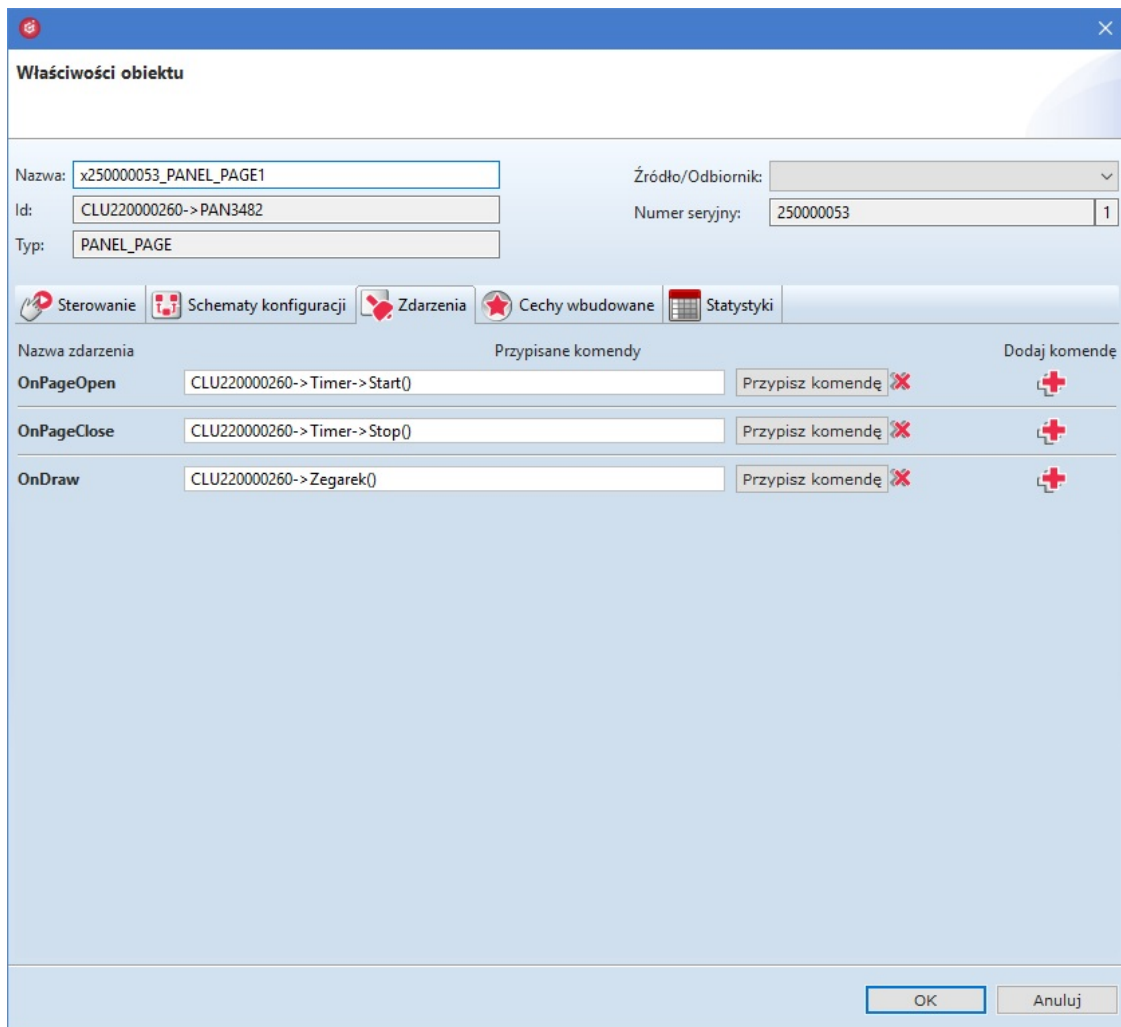
Id:

Sterowanie
 Zdarzenia
 Cechy wbudowane

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
Time	1000	<input type="text" value="1000"/>	ms	
Mode	1	<input type="text" value="Interval"/>		0,1
State	0			0,1,2
Value	0		ms	

Auto odświeżanie 

- Otworzyć obiekt *PANEL_PAGEX* (gdzie X to numer strony) poprzez dwukrotne kliknięcie na liście obiektów:
 - Przejść do zakładki *Zdarzenia*
 - Skonfigurować działanie strony poprzez przypisanie metod do określonych zdarzeń (klikając w "+" po prawej stronie okna):



- Wybrać zakładkę *Cechy wbudowane* i zdefiniować parametry konfiguracyjne obiektu;
- Wysłać konfigurację do CLU Z-Wave.

Skrypt *Zegarek* w wersji tekstowej:

```

CLU220000260->x25000053_PANEL1->CTearScreen()
CLU220000260->x25000053_PANEL1->PrintText(25,25,CLU220000260->Time,1)
CLU220000260->x25000053_PANEL1->DisplayContent()

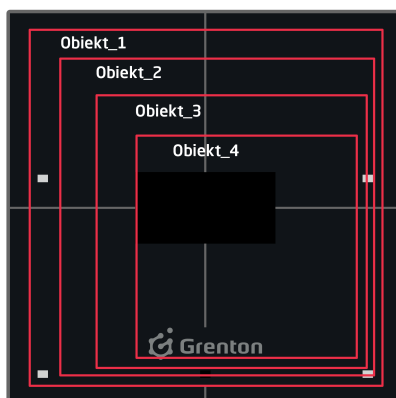
```

5.9. Tworzenie konfiguracji z wykorzystaniem obiektu strony *Thermostats*

W trybie pracy *Thermostats* do dyspozycji jest strona składająca się z 4 obiektów (łącznie obsługa do 16 obiektów na 4 stronach), dla których przypisywane są obiekty termostatów zdefiniowanych w systemie. Możliwa jest zmiana parametrów termostatów takich jak temperatura zadana czy tryb pracy. Istnieje również możliwość włączenia lub wyłączenia danego termostatu.

UWAGA!

W trybie pracy *Thermostats* przyciski, jak również rysowanie treści na wyświetlaczu jest zablokowane.



Strona typu „Thermostats”

Tworzenie konfiguracji panelu obsługującej stronę lub strony typu *Thermostats* najlepiej rozpocząć od stworzenia termostatów, które mają zostać wykorzystane w konfiguracji. Opis tworzenia oraz pracy obiektu wirtualnego *Thermostat* opisany został w podrozdziale IX.5.

Wersja v4 modułu Smart Panel obsługuje dwa rodzaje termostatów:

- Termostat lokalny - jest to obiekt wirtualny typu *Thermostat* stworzony na module CLU Z-Wave, do którego podłączony jest moduł Smart Panel z tworzoną aktualnie konfiguracją;
- Termostat zdalny - jest to obiekt wirtualny typu *Thermostat* stworzony na innym module CLU Z-Wave;

Za pośrednictwem modułu Smart Panel istnieje możliwość zmiany takich parametrów obiektu wirtualnego *Thermostat* jak:

- `PointValue` - temperatura zadana, możliwość odczytu aktualnie ustawionej temperatury, jak również zmiany na nową wartość;
- `Mode` - tryb pracy termostatu:
 - W trybie automatycznym `Auto (2)` wartość temperatury odczytywana jest z harmonogramu. Za pośrednictwem modułu Smart Panel nie ma możliwości zmiany tej temperatury;
 - W trybie ręcznym `Manual (0)` wartość temperatury odczytywana jest z cechy `PointValue`. Za pośrednictwem modułu Smart Panel istnieje możliwość zmiany tej temperatury;
- `State` - aktualny stan termostatu: wyłączony (`off (0)`) / włączony (`on (1)`).

A. Tworzenie konfiguracji z termostatem lokalnym

Chcąc stworzyć konfigurację z wykorzystaniem termostatu lokalnego należy:

- Stworzyć termostat na CLU Z-Wave, do którego podłączony jest moduł Smart Panel;
- Skonfigurować obiekt wirtualny zgodnie z założeniami;
- Otworzyć obiekt `PANEL_PAGEX` (gdzie X to numer jednej z 4 stron) poprzez dwukrotne kliknięcie na liście obiektów
- Wybrać zakładkę *Cechy wbudowane* i zdefiniować obiekty wyświetlane na ekranie:
 - `PageType` - cecha określająca typ strony, należy ustawić ją na wartość *Thermostats (2)*;

- o `PageName` - cecha określająca nazwę strony lub ikonę, która będzie wyświetlana w momencie przechodzenia pomiędzy stronami (działa wyłącznie przy ustawionej cieszce `PageDisplayMode` na wartość 1 (ShowIconOrName) w obiekcie Panel);
- o `Object_X_Id` - identyfikator termostatu. W tym celu należy odczytać wartość znajdującą się w polu `Id` obiektu wirtualnego *Thermostat*. Identyfikator termostatu lokalnego nie jest poprzedzony identyfikatorem CLU:

Właściwości obiektu

Nazwa: Typ:

Id:

- o `Object_X_Name` - nazwa termostatu. Brak nazwy termostatu w parametrze powoduje, iż termostat nie jest wyświetlany;

Właściwości obiektu

Nazwa: Źródło/Odbiornik:

Id: Numer seryjny:

Typ:

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
PageType	2	Thermostats		0,1,2,3
PageName	Strona 1	Strona 1	-	[0-15]
Object_1_Id	THE3749	THE3749	-	[0-23]
Object_1_Name	Kuchnia	Kuchnia	-	[0-15]
Object_2_Id	THE5081	THE5081	-	[0-23]
Object_2_Name	Salon	Salon	-	[0-15]
Object_3_Id	THE4059	THE4059	-	[0-23]
Object_3_Name	Hol	Hol	-	[0-15]
Object_4_Id	THE2718	THE2718	-	[0-23]
Object_4_Name	Łazienka	Łazienka	-	[0-15]

Auto odświeżanie

UWAGA!

Wysłanie konfiguracji wyłącznie ze zdefiniowanym typem strony, bez ustawionych powiązań obiektów z przyciskami wiąże się z uruchomieniem trybu pracy panelu jako *Thermostats*. Na wyświetlaczu będą widoczne kropki ("..."). Ma to związek z nie uzupełnieniem cech `Object_X_Id` oraz `Object_X_Name`.

- Wysłać konfigurację do CLU Z-Wave.

B. Tworzenie konfiguracji z termostatem zdalnym

Chcąc stworzyć konfigurację z wykorzystaniem termostatu zdalnego należy:

- Stworzyć termostat na CLU Z-Wave, do którego nie podłączony jest moduł Smart Panel z aktualnie wykonywaną konfiguracją;
- Skonfigurować obiekt wirtualny zgodnie z założeniami;
- Otworzyć obiekt *PANEL_PAGEX* (gdzie X to numer jednej z 4 stron) poprzez dwukrotne kliknięcie na liście obiektów
- Wybrać zakładkę *Cechy wbudowane* i zdefiniować obiekty wyświetlane na ekranie:
 - `PageType` - cecha określająca typ strony, należy ustawić ją na wartość *Thermostats (2)*;
 - `PageName` - cecha określająca nazwę strony lub ikonę, która będzie wyświetlana w momencie przechodzenia pomiędzy stronami (działa wyłącznie przy ustawionej cesze `PageDisplayMode` na wartość 1 (ShowIconOrName) w obiekcie Panel);
 - `Object_X_Id` - identyfikator termostatu. W tym celu należy odczytać wartość znajdującą się w polu *Id* obiektu wirtualnego *Thermostat*. Identyfikator termostatu zdalnego musi zostać poprzedzony identyfikatorem CLU:

Właściwości obiektu

Nazwa:

Typ:

Id:

- `Object_X_Name` - nazwa termostatu. Brak nazwy termostatu w parametrze powoduje, iż termostat nie jest wyświetlany;

Właściwości obiektu

Nazwa: Źródło/Odbiornik:

Id: Numer seryjny:

Typ:

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
PageType	2	<input type="text" value="Thermostats"/>		0,1,2,3
PageName	Strona 1	<input type="text" value="Strona 1"/>	-	[0-15]
Object_1_Id	CLU220000331->THE5372	<input type="text" value="CLU220000331->THE5372"/>	-	[0-23]
Object_1_Name	Sypialnia I	<input type="text" value="Sypialnia I"/>	-	[0-15]
Object_2_Id	CLU220000331->THE6721	<input type="text" value="CLU220000331->THE6721"/>	-	[0-23]
Object_2_Name	Sypialnia II	<input type="text" value="Sypialnia II"/>	-	[0-15]
Object_3_Id	CLU220000331->THE9021	<input type="text" value="CLU220000331->THE9021"/>	-	[0-23]
Object_3_Name	Hol piętro	<input type="text" value="Hol piętro"/>	-	[0-15]
Object_4_Id	CLU220000331->THE5542	<input type="text" value="CLU220000331->THE5542"/>	-	[0-23]
Object_4_Name	Łazienka I	<input type="text" value="Łazienka I"/>	-	[0-15]

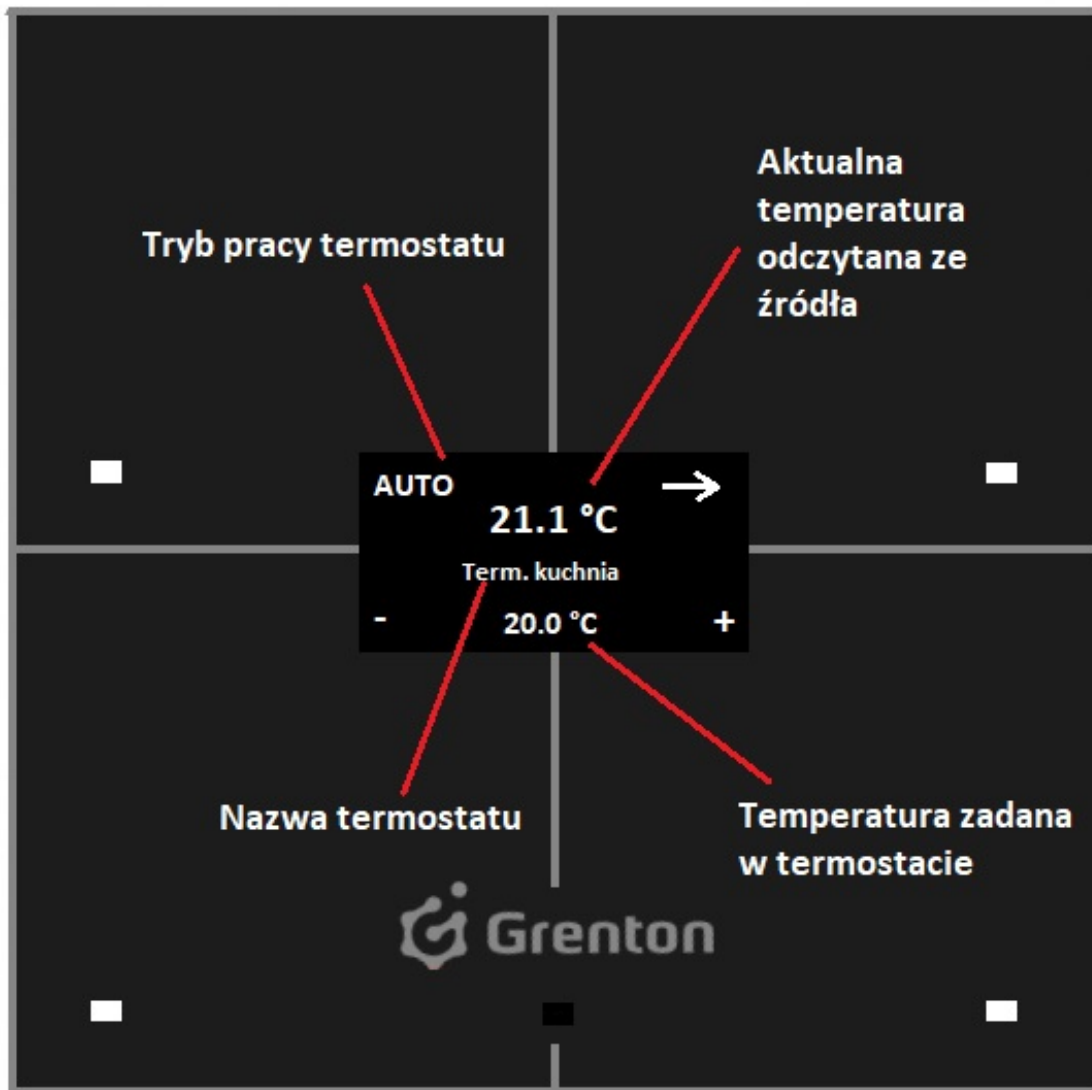
Auto odświeżanie

UWAGA!

Wysłanie konfiguracji wyłącznie ze zdefiniowanym typem strony, bez ustawionych powiązań obiektów z przyciskami wiąże się z uruchomieniem trybu pracy panelu jako *Thermostats*. Na wyświetlaczu będą widoczne kropki ("..."). Ma to związek z nie uzupełnieniem cech `Object_X_Id` oraz `Object_X_Name`.

- Wysłać konfigurację do CLU Z-Wave.

Poniższy schemat przedstawia poglądowy wygląd termostatu na ekranie Smart Panelu. Za pośrednictwem strzałki użytkownik może przejść do kolejnego termostatu na stronie. Natomiast za pomocą "-" / "+" istnieje możliwość zmiany zadanej temperatury.



C. Predefiniowane zachowanie przycisków

Przycisk	Krótkie/długie przycisnięcie	Opis zachowania
Lewy górny	Krótkie naciśnięcie (kliknięcie)	Zmiana trybu pracy termostatu: Manual/Auto
Lewy górny	Długie naciśnięcie (przytrzymanie)	Zmiana stanu termostatu: Off/On
Prawy górny	Krótkie naciśnięcie (kliknięcie)	Przejdźcie do następnego termostatu na stronie
Prawy górny	Długie naciśnięcie (przytrzymanie)	Brak zdefiniowanej funkcjonalności
Lewy dolny	Krótkie naciśnięcie (kliknięcie)	Zmniejszenie temperatury zadanej (<code>PointValue</code>) o 0.1°C
Lewy dolny	Długie naciśnięcie (przytrzymanie)	Zmniejszenie temperatury zadanej (<code>PointValue</code>) - dopóki przycisk jest przytrzymany
Prawy dolny	Krótkie naciśnięcie (kliknięcie)	Zwiększenie temperatury zadanej (<code>PointValue</code>) o 0.1 °C
Prawy dolny	Długie naciśnięcie (przytrzymanie)	Zwiększenie temperatury zadanej (<code>PointValue</code>) - dopóki przycisk jest przytrzymany

5.10. Łączenie obiektów w większe przyciski

Nowa wersja modułu Smart Panel wprowadza również możliwość łączenia/scalania 2, 3 lub 4 obiektów w jeden większy przycisk. Funkcjonalność dostępna jest wyłącznie w trybie stron *Buttons* oraz *FreeDraw*. W celu stworzenia większego przycisku należy:

- Skonfigurować obiekty `PANEL_BUTTOX` (gdzie X to numer przycisku):
 - W zakładce *Zdarzenia* skonfigurować działanie przycisku poprzez przypisanie metod do określonych zdarzeń;
 - W zakładce *Cechy wbudowane* zdefiniować obiekty wyświetlane na ekranie danego przycisku;
- Otworzyć obiekt `PANEL_PAGEX` (gdzie X to numer strony);
- Przejsć do zakładki *Zdarzenia*;
- Skonfigurować działanie strony poprzez przypisanie metod do określonych zdarzeń;
- Przejsć do zakładki *Cechy wbudowane*;
- Ustawić cechę `PageType` na wartość *Buttons* lub *FreeDraw*;
- Ustawić cechy `Object_X_Id` zgodnie z żadaną wersją łączenia:
 - Scalanie 2 obiektów w jeden przycisk w poziomie - ikona ustawiona dla przycisku jest wyświetlana na środku w górnej części ekranu (dla obiektów `Object_1_Id` oraz `Object_2_Id`) lub dolnej części ekranu (dla obiektów `Object_3_Id` oraz `Object_4_Id`);
 - Scalanie 2 obiektów w jeden przycisk w pionie - ikona ustawiona dla przycisku jest wyświetlana na środku po lewej części ekranu (dla obiektów `Object_1_Id` oraz `Object_3_Id`) lub po prawej części ekranu (dla obiektów `Object_2_Id` oraz `Object_4_Id`);

- Scalanie 3 obiektów w jeden przycisk - wyświetlana są dwie identyczne ikony, w zależności od sposobu łączenia obiektów;
- Scalanie 4 obiektów w jeden przycisk - ikona ustawiona dla przycisku jest wyświetlana na środku ekranu

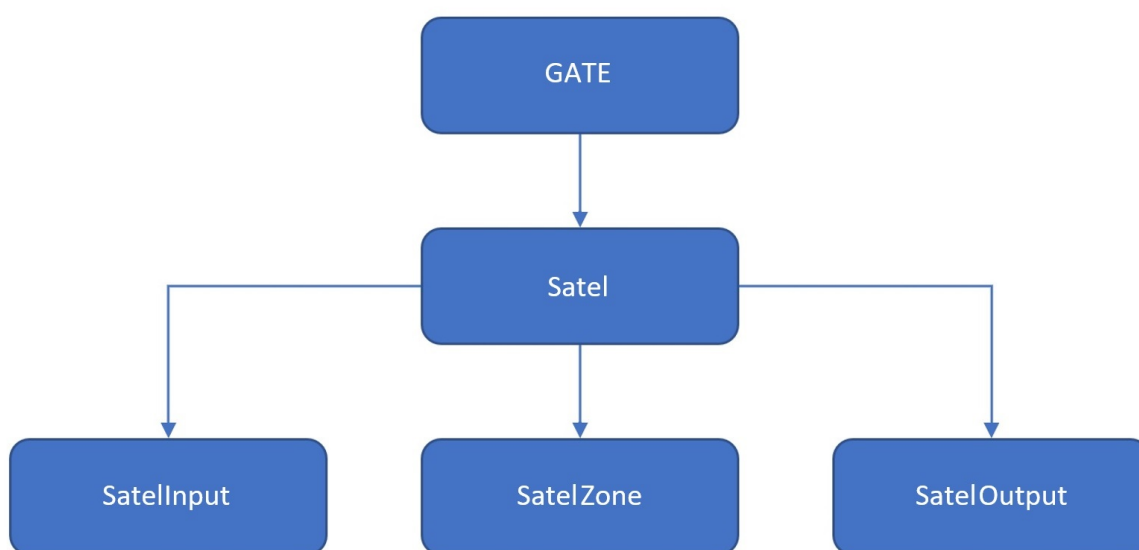
XIII. Moduł GATE Alarm

1. Integracja z centralą Satel

1.1. Informacje ogólne

Integracja systemu Grenton z centralą alarmową Satel jest możliwa za pośrednictwem modułu ETHM-1. Można maksymalnie utworzyć 64 obiekty wirtualne typu: *SatelZone*, *SatelInput*, *SatelOutput*. Możliwe jest również wykorzystanie kodowania integracji oferowanego przez firmę Satel.

Konstrukcja konfiguracji wygląda następująco:



Obiekt wirtualny:

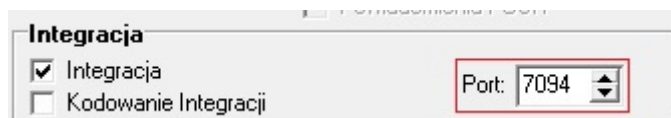
- **Satel** – umożliwia przeprowadzenie konfiguracji pozwalającej na integrację systemu z centralą alarmową Grenton;
- **SatelZone** – pozwala utworzyć strefę, do której dostęp będzie możliwy po wprowadzeniu hasła jednego z użytkowników lub hasła samego administratora;
- **SatelInput** – daje możliwość monitorowania stanu wybranego wejścia;
- **SatelOutput** – pozwala monitorować oraz ustawiać stan wybranego wyjścia po podaniu hasła użytkownika lub administratora.

1.2. Konfiguracja modułu GATE Alarm

UWAGA! Wszystkie wymagane informacje można znaleźć w konfiguracji modułu ETHM – korzystając z manipulatora podłączonego do centrali Satel bądź za pomocą dedykowanego programu DLOADX.

Przed rozpoczęciem konfiguracji należy posiadać informacje dotyczące centrali Satel oraz modułu ETHM-1:

- Adres IP modułu ETHM (Satel) – dostępny w konfiguracji Satela (*DLOADX -> Dane -> Struktura i Sprzęt -> zakładka Sprzęt -> Manipulatory -> ETHM-1 -> sekcja Adres IP serwera*);
- Port integracji ETHM – dostępny w konfiguracji Satela (*DLOADX -> Dane -> Struktura i Sprzęt -> zakładka Sprzęt -> Manipulatory -> ETHM-1 -> sekcja Integracja*);

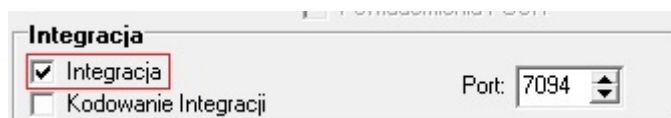


Integracja

Integracja Port: 7094

Kodowanie Integracji

- Hasło administratora/użytkowników - domyślne hasło w konfiguracji Satela dla administratora to: 1111 (*DLOADX -> Użytkownicy -> Użytkownicy*);
- Integracja po stronie modułu ETHM **musi być załączona** (*DLOADX -> Dane -> Struktura i Sprzęt -> zakładka Sprzęt -> Manipulatory -> ETHM-1 -> sekcja Integracja*);

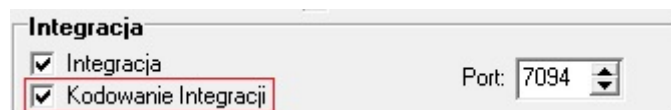


Integracja

Integracja Port: 7094

Kodowanie Integracji

- W sytuacji, gdy szyfrowanie - *Kodowanie integracji* jest załączone, należy również znać klucz kodowania (*DLOADX -> Dane -> Struktura i Sprzęt -> zakładka Sprzęt -> Manipulatory -> ETHM-1 -> sekcja Integracja*);

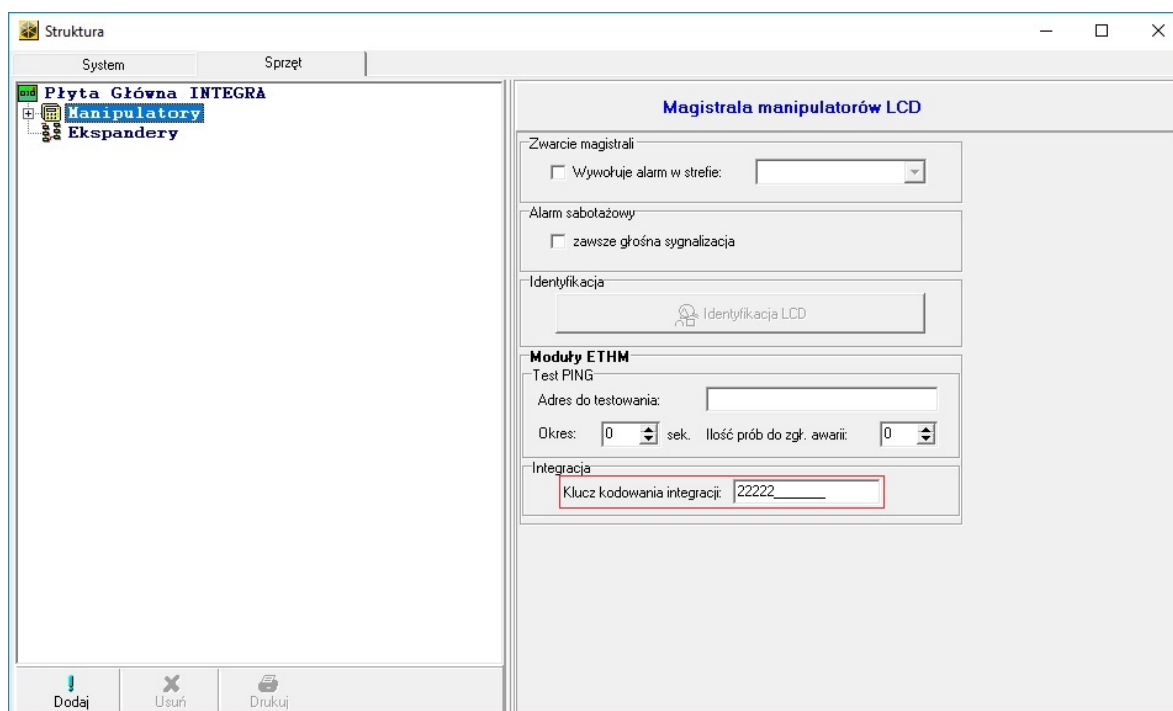


Integracja

Integracja Port: 7094

Kodowanie Integracji

- Klucz kodowania można odnaleźć w konfiguracji Satela (*DLOADX -> Dane -> Struktura i Sprzęt -> zakładka Sprzęt -> Manipulatory*) lub odczytać go za pomocą manipulatora (*Manipulator -> Tryb serwisowy -> Opcje -> Klucz integracji*).



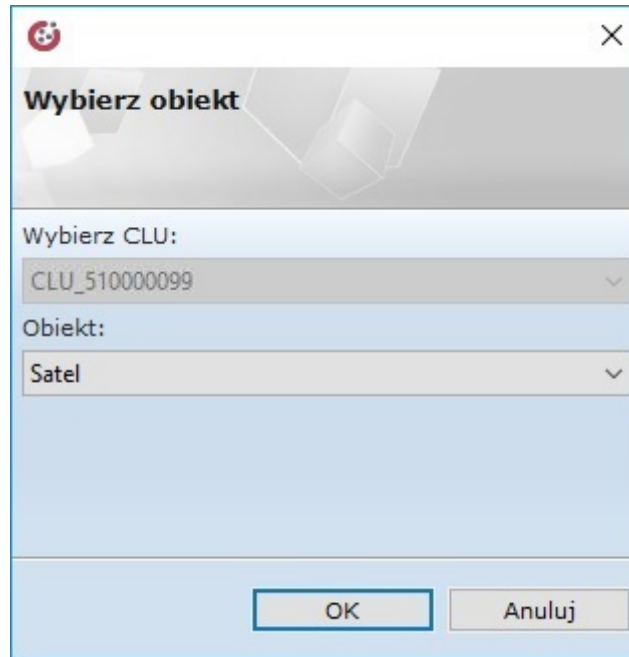
1.3. Obiekty wirtualne

UWAGA! Przed rozpoczęciem pracy z modułem GATE Alarm wymagana jest aktualizacja bazy interfejsów!

A. Satel

Chcąc przeprowadzić poprawną konfigurację modułu GATE Alarm należy kolejno:

- Utworzyć obiekt wirtualny *Satel*:



- Przejść do konfiguracji – zakładka *Cechy wbudowane* i wprowadzić:
 - **IP** – adres IP modułu ETHM (Satel);
 - **Port** – port integracji ETHM;
 - **AdminPassword** – hasło administratora;
 - **EncryptionEnabled** - załączenie kodowania – ustawić w sytuacji, gdy integracja na module ETHM ma zaznaczoną opcję *Kodowanie integracji*;
 - **Encryption Key** – klucz integracji (dla załączonego kodowania):

CLU_510000099->Satel1

Nazwa: Typ:

Sterowanie
 Zdarzenia
 Cechy wbudowane

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
State	0		bool	0,1
LastError	0			
IP	192.168.0.10	<input type="text" value="192.168.0.10"/>	string	
Port	7094	<input type="text" value="7094"/>		[1-65535]
AdminPassword	0	<input type="text"/>	string	
UpdateTime	1000	<input type="text" value="1000"/>		[1000-20000]
EncryptionEnabled	true	<input type="text" value="true"/>	bool	0,1
EncryptionKey	0	<input type="text"/>	string	

Auto odświeżanie
 Odśwież

Informacje na temat tego, gdzie można znaleźć potrzebne informacje znajdują się w drugim podrozdziale - [patrz pkt XIII.1.2.](#)

- Wysłać konfigurację oraz zweryfikować połączenie – zakładka *Cechy wbudowane*, cecha `state` (1 – poprawnie połączono z centralą, 0 – brak połączenia):

CLU_510000099->Satel1

Nazwa: Typ:

Sterowanie
 Zdarzenia
 Cechy wbudowane

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
State	1		bool	0,1
LastError	0			

B. Strefa

Moduł GATE Alarm umożliwia dodanie obiektu wirtualnego *Strefa*:

- Utworzyć obiekt *SatelZone*:

- Zdefiniować Nr (numer wybranej strefy) oraz podać hasło użytkownika:

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
Value	0		-	
Nr	1	1	number	[1-32]
UserPassword	2222	2222	number	

- Wysłać konfigurację oraz zweryfikować połączenie - zakładka *Cechy wbudowane*, cecha `Value` (-1 to brak połączenia z centralną, pozostałe oznaczają poprawne połączenie i zwracany jest stan strefy: 0 lub 1);
- Uzbroić/rozbroić strefę – metody `ArmZone` oraz `DisarmZone`.

C. Wyjście

GATE Alarm umożliwia dodanie wirtualnego obiektu *Wyjście*:

- Utworzyć obiekt *SatelOutput*:

- Zdefiniować Nr (numer wybranego wyjścia na płytce Satel) oraz podać hasło użytkownika:

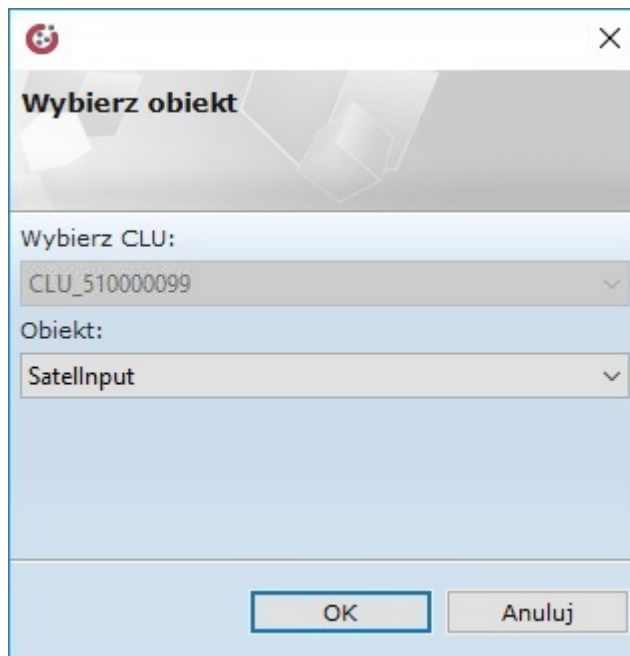
Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
Value	0		bool	[0-1]
Nr	1	1	number	[1-256]
UserPassword	1234	1234	number	[0-99999]

- Wysłać konfigurację oraz zweryfikować połączenie - zakładka *Cechy wbudowane*, cecha `Value` (-1 to brak połączenia z centralką, pozostałe oznaczają poprawne połączenie i zwracany jest stan strefy: 0 lub 1);
- Załączyć/wyłączyć wyjście – metody `switchon` i `switchoff`.

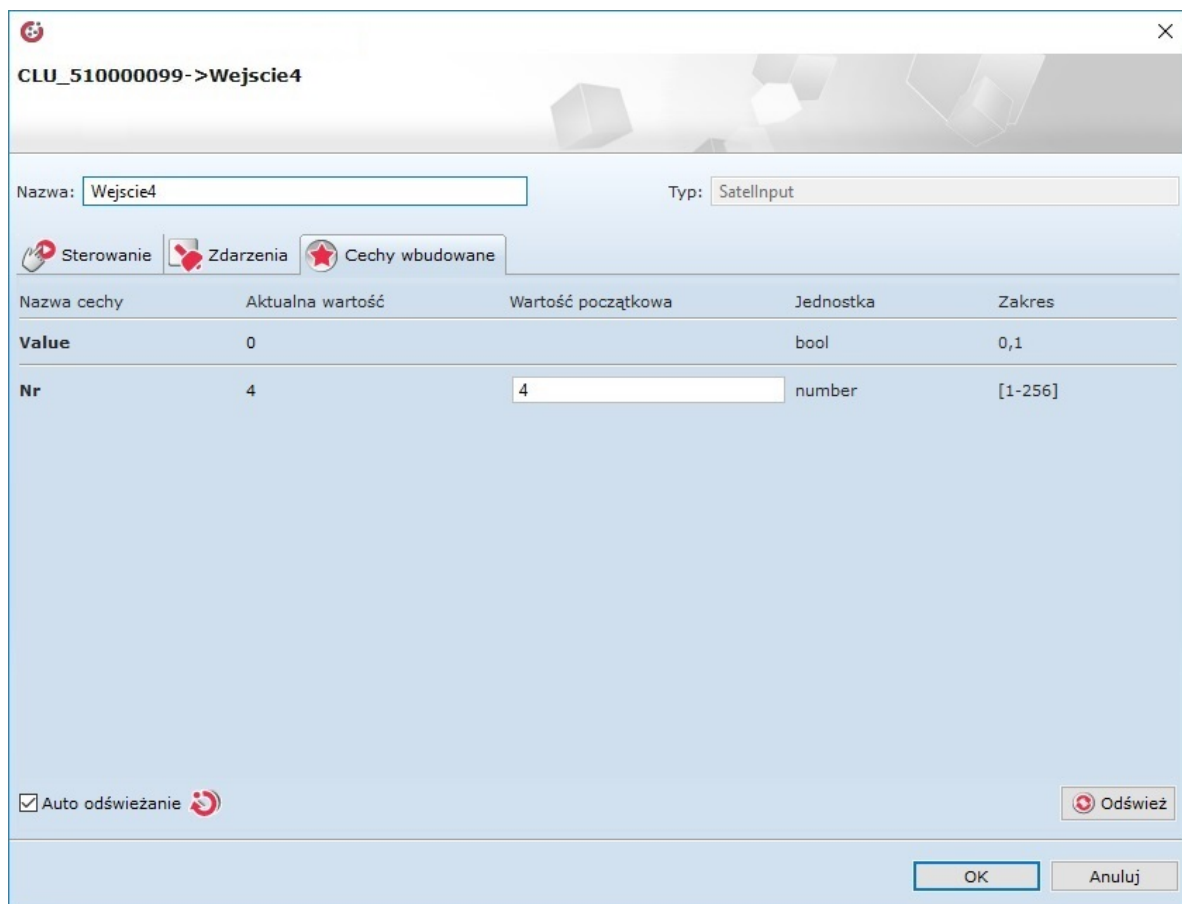
D. Wejście

GATE Alarm umożliwia dodanie wirtualnego obiektu *Wejście*:

- Utworzyć obiekt *SatellInput*:



- Zdefiniować Nr (numer wybranego wejścia na płycie Satel):



- Wysłać konfigurację oraz zweryfikować połączenie - zakładka *Cechy wbudowane*, cecha `Value` (-1 to brak połączenia z centralką, pozostałe oznaczają poprawne połączenie i zwracany jest stan strefy: 0 lub 1).

2. Przywracanie ustawień fabrycznych - *Hard Reset*

UWAGA! Kroki procedury dotyczą modułu GATE Alarm w wersji fw. **18.23.4.1457** lub niższej
Uruchomienie funkcji *Hard Reset* na module GATE Alarm powoduje:

- Usunięcie zapisanej konfiguracji;

- Sformatowanie partycji pamięci flash;
- Usunięcie wszystkich utworzonych obiektów LUA;
- Utratę komunikacji pomiędzy OM / HM a modułem Gate.

W celu przywrócenia ustawień fabrycznych funkcją *Hard Reset* należy wykonać następujące czynności (zgodnie z podaną kolejnością):

- Odłączyć zasilanie od modułu Gate;
- Nacisnąć i przytrzymać przycisk *Reset* na module (przycisk znajduje się pod dolną zaślepką modułu);
- Podłączyć zasilanie do modułu Gate;
- Trzymać wciśnięty przycisk *Reset* przez co najmniej 3 sekundy - prawidłowe wykonanie resetu zostanie potwierdzone 3-krotnym mrugnięciem diody zielonej.
- Po upływie 3 sekund zwolnić przycisk *Reset*
- Odczekać około 60 sekund aż do momentu, gdy na module diody - zielona oraz czerwona - będą mrugać naprzemiennie (tryb *Emergency*)

Po wykonaniu procedury na module zostanie wyczyszczona konfiguracja, natomiast sam moduł przestanie być widoczny (brak odpowiedzi na *Keep-Alive*) w projekcie z poziomu Object Managera. Aby ponownie przywrócić moduł, należy wykonać CLU Discovery a następnie wysłać konfigurację.

3. Parametry konfiguracyjne

CECHY

Nazwa	Opis
Uptime	Czas pracy urządzenia od ostatniego resetu (w sekundach)
UnixTime	Aktualny uniksowy znacznik czasu
LibraryVersion	Wersja oprogramowania Gate
ClientReportInterval	Okres raportowania o zmianach cech
State	Stan centralki (0 – brak połączenia z centralką, 1 – połączono z centralką)
LastError	Ostatni kod błędu modułu ETHM (0 – ok, 1 – nieprawidłowe hasło)
IP	Adres IP modułu ETHM (Satel)
Port	Port modułu ETHM (Satel)
AdminPassword	Hasło administratora Satel
UpdateTime	Czas uaktualniania stanu centralki
EncryptionEnabled	Stan szyfrowania (<i>true</i> – załączone, <i>false</i> – wyłączone)
EncryptionKey	Klucz szyfrowania Satel
value	Zwraca aktualny stan (1 – dla strefy uzbrojonej, wejścia naruszonego, wyjścia załączonego; 0 – dla strefy rozbrojonej, wejścia nienaruszonego, wyjścia wyłączonego; -1 – brak informacji na temat stanu z powodu braku połączenia)
Nr	Parametr definiujący strefę, wejście lub wyjście, do których odnosi się obiekt
UserPassword	Hasło użytkownika (dla „_” zastosuje hasło administratora)

METODY

Nazwa	Opis
SetDateTime	Ustawia datę i czas
SetClientReportInterval	Ustawia okres raportowania o zmianach cech
SetUpdateTime	Ustawia czas uaktualniania centralki
SetIP	Ustawia adres IP modułu ETHM (Satel)
SetPort	Ustawia port modułu ETHM (Satel)
SetAdminPassword	Ustawia hasło administratora
SetEncryptionEnabled	Włącza/wyłącza szyfrowanie
SetEncryptionKey	Ustawia klucz szyfrowania Satel
ArmZone	Uzbraja strefę
DisarmZone	Rozbraja strefę
SetNr	Ustawia parametr definiujący, do której strefy, wejścia lub wyjścia odnosi się obiekt
SetUserPassword	Ustawia hasło użytkownika (dla _ zastosuje hasło administratora)
Switchon	Załącza wyjście
Switchoff	Wyłącza wyjście

ZDARZENIA

Nazwa	Opis
OnInit	Zdarzenie wywołane w momencie inicjalizacji urządzenia
OnConnected	Zdarzenie wywołane po nawiązaniu połączenia z centralą
OnDisconnected	Zdarzenie wywołane po utraceniu połączenia z centralką
OnError	Zdarzenie wywołane po wystąpieniu błędu centralki (LastError)
OnChange	Zdarzenie wywołane przy zmianie stanu (niezależnie od wartości)
OnSwitchon	Zdarzenie wywołane przy załączeniu wyjścia lub naruszeniu wejścia
OnSwitchoff	Zdarzenie wywołane przy wyłączeniu wyjścia lub ustawieniu stanu normalnego na wejściu
OnArm	Zdarzenie wywołane przy uzbrojeniu strefy
OnDisarm	Zdarzenie wywołane przy rozbrojeniu strefy

XIV. Moduł GATE Modbus

1. Informacje ogólne

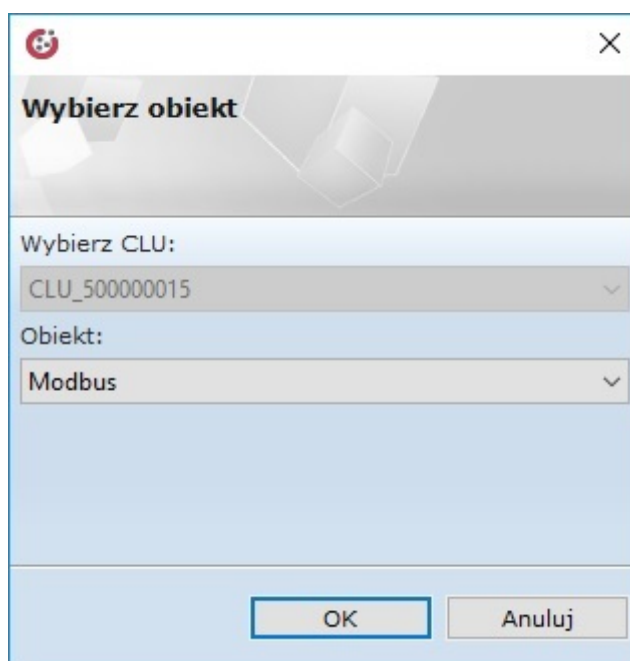
Moduł GATE Modbus umożliwia integrację systemu Grenton ze wszystkimi urządzeniami wspierającymi standard MODBUS RTU. Maksymalnie można utworzyć 64 obiekty wirtualne typu Modbus. Przed rozpoczęciem konfiguracji należy uzyskać informacje dotyczące wybranego urządzenia Slave wspierającego standard MODBUS RTU - konieczna będzie znajomość m.in.: adresu urządzenia, typu oraz adresów rejestrów, a także prędkości transmisji.

2. Konfiguracja modułu GATE Modbus

UWAGA! Przed rozpoczęciem jakiegokolwiek pracy z modułem GATE Modbus wymagana jest aktualizacja bazy interfejsów!

Chcąc przeprowadzić poprawną konfigurację modułu GATE Modbus należy kolejno:

- Utworzyć obiekt wirtualny *Modbus*:



- Przejsć do konfiguracji – zakładka *Cechy wbudowane* i wprowadzić:
 - **DeviceAddress** – adres urządzenia Slave;
 - **AccessRights** - tryb pracy (*read* - odczyt wartości z rejestru; *read/write* - umożliwia zapis wartości do ustawionego rejestru);
 - **RegisterAddress** – adres obsługiwanego rejestru;
 - **TransmissionSpeed** – prędkość transmisji;
 - **RefreshInterval** - okres odpytywania rejestru urządzenia Slave przez GATE Modbus;
 - **ResponseTimeout** - czas urządzenia Slave na odpowiedź (po jego przekroczeniu zwracany jest `ErrorCode == 2`);
 - **Divisor** - dzielnik (dla `valueType = number/float`);
 - parametry odpowiednie dla wybranego typu rejestru urządzenia Slave - [patrz pkt XIV.3.](#)

CLU_50000015- >mod1

Nazwa: Typ:

Sterowanie
 Zdarzenia
 Cechy wbudowane

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
DeviceAddress	111	<input type="text" value="111"/>	number	[0-255]
AccessRights	0	<input type="text" value="Read"/>	-	0,1
RegisterAddress	141	<input type="text" value="141"/>	number	[0-65535]
TransmissionSpeed	9600	<input type="text" value="9600"/>	bps	1200,2400,4800,9600,19200,38400,57600,115200
ValueType	1	<input type="text" value="number"/>		1,2,3
BitPosition	0	<input type="text" value="0"/>	number	[0-15]
BitCount	16	<input type="text" value="16"/>	number	[1-32]
RefreshInterval	2000	<input type="text" value="2000"/>	number	[0-65535]
ResponseTimeout	500	<input type="text" value="500"/>	number	[10-65535]
Divisor	1	<input type="text" value="1"/>	number	[1-65535]
Endianness	3	<input type="text" value="Swap words"/>	-	0,1,2,3
RegisterType	2	<input type="text" value="Rejestry pamiętające"/>	-	0,1,2,3
ErrorCode	0		number	
Value	639	<input type="text" value="0"/>	number	
RegisterValue	639		number	

Auto odświeżanie

- Wysłać konfigurację oraz zweryfikować połączenie – zakładka *Cechy wbudowane*, cecha `ErrorCode=0` (poprawny odczyt/zapis):

CLU_50000015->mod1
×

Nazwa:
Typ:

Sterowanie
 Zdarzenia
 Cechy wbudowane

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
DeviceAddress	111	<input type="text" value="111"/>	number	[0-255]
AccessRights	0	<input type="text" value="Read"/>	-	0,1
RegisterAddress	141	<input type="text" value="141"/>	number	[0-65535]
TransmissionSpeed	9600	<input type="text" value="9600"/>	bps	1200,2400,4800,9600,19200,38400,57600,115200
ValueType	1	<input type="text" value="number"/>		1,2,3
BitPosition	0	<input type="text" value="0"/>	number	[0-15]
BitCount	16	<input type="text" value="16"/>	number	[1-32]
RefreshInterval	2000	<input type="text" value="2000"/>	number	[0-65535]
ResponseTimeout	500	<input type="text" value="500"/>	number	[10-65535]
Divisor	1	<input type="text" value="1"/>	number	[1-65535]
Endianness	3	<input type="text" value="Swap words"/>	-	0,1,2,3
RegisterType	2	<input type="text" value="Rejestry pamiętające"/>	-	0,1,2,3
ErrorCode	0		number	
Value	639	<input type="text" value="0"/>	number	
RegisterValue	639		number	

Auto odświeżanie

3. Parametry rejestrów

W zależności od typu danego rejestru urządzenia Slave, należy ustawić odpowiednio kolejne dostępne parametry.

A. Rejestry 16-bitowe

Odczyt 16-bitowych rejestrów pamiętających (Read Holding Registers, *FunctionCode=03*):

- AccessRights: read;
- ValueType: number;
- BitPosition: wartość domyślna;
- BitCount: 16;
- Endianness: wartość domyślna;
- RegisterType: rejestry pamiętające;

CLU_50000015->mod1

Nazwa: Typ:

Sterowanie
 Zdarzenia
 Cechy wbudowane

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
DeviceAddress	1	<input type="text" value="1"/>	number	[0-255]
AccessRights	0	<input type="text" value="Read"/>	-	0,1
RegisterAddress	1	<input type="text" value="1"/>	number	[0-65535]
TransmissionSpeed	115200	<input type="text" value="115200"/>	bps	1200,2400,4800,9600,19200,38400,57600,115200
ValueType	1	<input type="text" value="number"/>		1,2,3
BitPosition	0	<input type="text" value="0"/>	number	[0-15]
BitCount	16	<input type="text" value="16"/>	number	[1-32]
RefreshInterval	5000	<input type="text" value="5000"/>	number	[0-65535]
ResponseTimeout	1000	<input type="text" value="1000"/>	number	[10-65535]
Divisor	1	<input type="text" value="1"/>	number	[1-65535]
Endianness	0	<input type="text" value="No swap"/>	-	0,1,2,3
RegisterType	2	<input type="text" value="Rejestry pamiętające"/>	-	0,1,2,3
ErrorCode	0		number	
Value	55555	<input type="text" value="0"/>	number	
RegisterValue	55555		number	

Auto odświeżanie

Odczyt 16-bitowych rejestrów wejściowych (`Read Input Registers`, `FunctionCode=04`):

- `AccessRights`: read;
- `ValueType`: number;
- `BitPosition`: wartość domyślna;
- `BitCount`: 16;
- `Endianness`: wartość domyślna;
- `RegisterType`: rejestry wejściowe.

Zapis 16-bitowych rejestrów pamiętających (`Preset/write Single Holding Register`, `FunctionCode=06`):

- `AccessRights`: read/write;
- `ValueType`: number;
- `BitPosition`: wartość domyślna;
- `BitCount`: 16;
- `Endianness`: wartość domyślna;
- `RegisterType`: rejestry pamiętające.

B. Pola w rejestrach 16-bitowych

Odczyt pól bitowych w 16-bitowym rejestrze pamiętającym (`Read Holding Registers`, `FunctionCode=03`):

- `AccessRights`: read;

- `valueType`: bit;
- `BitPosition`: 0-15 (pozycja pierwszego interesującego bitu);
- `BitCount`: 1-16 (ilość odczytywanych kolejno bitów);
- `Endianness`: wartość domyślna;
- `RegisterType`: rejestry pamiętające;

CLU_500000015->mod2

Nazwa: Typ:

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
DeviceAddress	2	<input type="text" value="2"/>	number	[0-255]
AccessRights	0	<input type="button" value="Read"/>	-	0,1
RegisterAddress	1	<input type="text" value="1"/>	number	[0-65535]
TransmissionSpeed	115200	<input type="button" value="115200"/>	bps	1200,2400,4800,9600,19200,38400,57600,115200
ValueType	3	<input type="button" value="bit"/>	-	1,2,3
BitPosition	2	<input type="text" value="2"/>	number	[0-15]
BitCount	1	<input type="text" value="1"/>	number	[1-32]
RefreshInterval	10000	<input type="text" value="10000"/>	number	[0-65535]
ResponseTimeout	1000	<input type="text" value="1000"/>	number	[10-65535]
Divisor	1	<input type="text" value="1"/>	number	[1-65535]
Endianness	0	<input type="button" value="No swap"/>	-	0,1,2,3
RegisterType	2	<input type="button" value="Rejestry pamiętające"/>	-	0,1,2,3
ErrorCode	0		number	
Value	1	<input type="text" value="0"/>	number	
RegisterValue	4		number	

Auto odświeżanie

Odczyt pól bitowych w 16-bitowym rejestrze wejściowym (`Read Input Registers`, `FunctionCode=04`):

- `AccessRights`: read;
- `valueType`: bit;
- `BitPosition`: 0-15 (pozycja pierwszego interesującego bitu);
- `BitCount`: 1-16 (ilość odczytywanych kolejno bitów);
- `Endianness`: wartość domyślna;
- `RegisterType`: rejestry wejściowe.

Zapis pól bitowych w 16-bitowym rejestrze pamiętającym (`Preset/write Single Holding Register`, `FunctionCode=06`):

- `AccessRights`: read/write;
- `valueType`: bit;
- `BitPosition`: 0-15 (pozycja pierwszego interesującego bitu);
- `BitCount`: 1-16 (ilość odczytywanych kolejno bitów);
- `Endianness`: wartość domyślna;
- `RegisterType`: rejestry pamiętające.

C. 32-bitowe wartości całkowite rejestrów

Odczyt 32-bitowych wartości całkowitych rejestru pamiętającego (`Read Holding Registers`, `FunctionCode=03`):

- `AccessRights`: read;
- `valueType`: number;
- `BitPosition`: wartość domyślna;
- `BitCount`: 32;
- `Endianness`: w przypadku 32-bitowych rejestrów urządzenia Slave najczęściej wymagają zmiany kolejności bajtów i słów - *Swap bytes and words*; szczegółowe informacje należy odnaleźć w karcie urządzenia Slave;
- `RegisterType`: rejestry pamiętające:

CLU_500000015->mod3

Nazwa: mod3 Typ: Modbus

Sterowanie Zdarzenia Cechy wbudowane

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
DeviceAddress	3	3	number	[0-255]
AccessRights	0	Read	-	0,1
RegisterAddress	1	1	number	[0-65535]
TransmissionSpeed	115200	115200	bps	1200,2400,4800,9600,19200,38400,57600,115200
ValueType	1	number	-	1,2,3
BitPosition	0	0	number	[0-15]
BitCount	32	32	number	[1-32]
RefreshInterval	10000	10000	number	[0-65535]
ResponseTimeout	1000	1000	number	[10-65535]
Divisor	1	1	number	[1-65535]
Endianness	1	Swap bytes and words	-	0,1,2,3
RegisterType	2	Rejestry pamiętające	-	0,1,2,3
ErrorCode	0		number	
Value	200000000	0	number	
RegisterValue	200000000		number	

Auto odświeżanie Odśwież

OK Anuluj

Odczyt 32-bitowych wartości całkowitych rejestru wejściowego (`Read Input Registers`, `FunctionCode=04`):

- `AccessRights`: read;
- `valueType`: number;
- `BitPosition`: wartość domyślna;
- `BitCount`: 32;
- `Endianness`: w przypadku 32-bitowych rejestrów urządzenia Slave najczęściej wymagają zmiany kolejności bajtów i słów - *Swap bytes and words*; szczegółowe informacje należy odnaleźć w karcie urządzenia Slave;
- `RegisterType`: rejestry wejściowe.

Zapis 32-bitowych wartości całkowitych w rejestrze pamiętającym (Preset/Write Multiple Holding Registers, *FunctionCode=16*):

- **AccessRights**: read/write;
- **ValueType**: number;
- **BitPosition**: wartość domyślna;
- **BitCount**: 32;
- **Endianness**: w przypadku 32-bitowych rejestrów urządzenia Slave najczęściej wymagają zmiany kolejności bajtów i słów - *Swap bytes and words*; szczegółowe informacje należy odnaleźć w karcie urządzenia Slave;
- **RegisterType**: rejestry pamiętające.

D. 32-bitowe wartości zmiennoprzecinkowe rejestrów

Odczyt 32-bitowych wartości zmiennoprzecinkowych rejestru pamiętającego (Read Holding Registers, *FunctionCode=03*):

- **AccessRights**: read;
- **ValueType**: float;
- **BitPosition**: wartość domyślna;
- **BitCount**: 32;
- **Endianness**: w przypadku 32-bitowych rejestrów urządzenia Slave najczęściej wymagają zmiany kolejności bajtów i słów - *Swap bytes and words*; szczegółowe informacje należy odnaleźć w karcie urządzenia Slave;
- **RegisterType**: rejestry pamiętające:

Nazwa: Typ:

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
DeviceAddress	4	<input type="text" value="4"/>	number	[0-255]
AccessRights	0	<input type="text" value="Read"/>	-	0,1
RegisterAddress	1	<input type="text" value="1"/>	number	[0-65535]
TransmissionSpeed	115200	<input type="text" value="115200"/>	bps	1200,2400,4800,9600,19200,38400,57600,115200
ValueType	2	<input type="text" value="float"/>	-	1,2,3
BitPosition	0	<input type="text" value="0"/>	number	[0-15]
BitCount	32	<input type="text" value="32"/>	number	[1-32]
RefreshInterval	10000	<input type="text" value="10000"/>	number	[0-65535]
ResponseTimeout	1000	<input type="text" value="1000"/>	number	[10-65535]
Divisor	1	<input type="text" value="1"/>	number	[1-65535]
Endianness	1	<input type="text" value="Swap bytes and words"/>	-	0,1,2,3
RegisterType	2	<input type="text" value="Rejestry pamiętające"/>	-	0,1,2,3
ErrorCode	0		number	
Value	100.00	<input type="text" value="0"/>	number	
RegisterValue	1120403456		number	

Auto odświeżanie

Odczyt 32-bitowych wartości zmiennoprzecinkowych rejestru wejściowego (`Read Input Registers` , `FunctionCode=04`):

- `AccessRights` : read;
- `ValueType` : float;
- `BitPosition` : wartość domyślna;
- `BitCount` : 32;
- `Endianness` : w przypadku 32-bitowych rejestrów urządzenia Slave najczęściej wymagają zmiany kolejności bajtów i słów - *Swap bytes and words*; szczegółowe informacje należy odnaleźć w karcie urządzenia Slave;
- `RegisterType` : rejestry wejściowe.

Zapis 32-bitowych wartości zmiennoprzecinkowych w rejestrze pamiętającym (`Preset/Write Multiple Holding Registers` , `FunctionCode=16`):

- `AccessRights` : read/write;
- `ValueType` : float;
- `BitPosition` : wartość domyślna;
- `BitCount` : 32;
- `Endianness` : w przypadku 32-bitowych rejestrów urządzenia Slave najczęściej wymagają zmiany kolejności bajtów i słów - *Swap bytes and words*; szczegółowe informacje należy odnaleźć w karcie urządzenia Slave;
- `RegisterType` : rejestry pamiętające.

E. Dyskretne wejścia/wyjścia

Odczyt dyskretnych wyjść/wejść bitowych (`Read Coil Status` , `FunctionCode=01`):

- `AccessRights` : read;
- `ValueType` : number;
- `BitPosition` : wartość domyślna;
- `BitCount` : 1-32;
- `Endianness` : wartość domyślna;
- `RegisterType` : wyjścia bitowe:

CLU_50000015->mod5

Nazwa: Typ:

Sterowanie
 Zdarzenia
 Cechy wbudowane

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
DeviceAddress	5	<input type="text" value="5"/>	number	[0-255]
AccessRights	0	<input type="text" value="Read"/>	-	0,1
RegisterAddress	1	<input type="text" value="1"/>	number	[0-65535]
TransmissionSpeed	115200	<input type="text" value="115200"/>	bps	1200,2400,4800,9600,19200,38400,57600,115200
ValueType	1	<input type="text" value="number"/>		1,2,3
BitPosition	0	<input type="text" value="0"/>	number	[0-15]
BitCount	8	<input type="text" value="8"/>	number	[1-32]
RefreshInterval	10000	<input type="text" value="10000"/>	number	[0-65535]
ResponseTimeout	1000	<input type="text" value="1000"/>	number	[10-65535]
Divisor	1	<input type="text" value="1"/>	number	[1-65535]
Endianness	0	<input type="text" value="No swap"/>	-	0,1,2,3
RegisterType	1	<input type="text" value="Wejścia dwustanowe"/>	-	0,1,2,3
ErrorCode	0		number	
Value	30	<input type="text" value="0"/>	number	
RegisterValue	30		number	

Auto odświeżanie

Odczyt dyskretnych wejść dwustanowych (*Read Discrete Inputs*, *FunctionCode=02*):

- `AccessRights`: read;
- `ValueType`: number;
- `BitPosition`: wartość domyślna;
- `BitCount`: 1-32;
- `Endianness`: wartość domyślna;
- `RegisterType`: wejścia dwustanowe.

Zapis dyskretnych wyjść/wejść bitowych (*Force/write single coil*, *FunctionCode=05*;
Force/write multiple coils, *FunctionCode=15*):

- `AccessRights`: read/write;
- `ValueType`: number;
- `BitPosition`: wartość domyślna;
- `BitCount`: 1-32;
- `Endianness`: wartość domyślna;
- `RegisterType`: wyjścia bitowe.

4. Przywracanie ustawień fabrycznych - *Hard Reset*

UWAGA! Kroki procedury dotyczą modułu GATE Modbus w wersji fw. **18.21.4.1453** lub niższej

Uruchomienie funkcji *Hard Reset* na module GATE Modbus powoduje:

- Usunięcie zapisanej konfiguracji;
- Sformatowanie partycji pamięci flash;
- Usunięcie wszystkich utworzonych obiektów LUA;
- Utratę komunikacji pomiędzy OM / HM a modułem Gate.

W celu przywrócenia ustawień fabrycznych funkcją *Hard Reset* należy wykonać następujące czynności (zgodnie z podaną kolejnością):

- Odłączyć zasilanie od modułu Gate;
- Nacisnąć i przytrzymać przycisk *Reset* na module (przycisk znajduje się pod dolną zaślepką modułu);
- Podłączyć zasilanie do modułu Gate;
- Trzymać wciśnięty przycisk *Reset* przez co najmniej 3 sekundy - prawidłowe wykonanie resetu zostanie potwierdzone 3-krotnym mrugnięciem diody zielonej.
- Po upływie 3 sekund zwolnić przycisk *Reset*
- Odczekać około 60 sekund aż do momentu, gdy na module diody - zielona oraz czerwona - będą mrugać naprzemiennie (tryb *Emergency*)

Po wykonaniu procedury na module zostanie wyczyszczona konfiguracja, natomiast sam moduł przestanie być widoczny (brak odpowiedzi na *Keep-Alive*) w projekcie z poziomu Object Managera. Aby ponownie przywrócić moduł, należy wykonać CLU Discovery a następnie wysłać konfigurację.

5. Parametry konfiguracyjne

CECHY

Nazwa	Opis
Uptime	Czas pracy urządzenia od ostatniego resetu (w sekundach)
UnixTime	Aktualny uniksowy znacznik czasu
LibraryVersion	Wersja oprogramowania Gate
ClientReportInterval	Okres raportowania o zmianach cech
DeviceAddress	Adres urządzenia Modbus typu Slave
AccessRights	Tryb pracy: <i>read</i> (0 – odczyt); <i>read/write</i> (1 - odczyt/zapis)
RegisterAddress	Adres obsługiwanego rejestru
TransmissionSpeed	Prędkość transmisji
ValueType	Typ zmiennej (1 – <i>number</i> ; 2 – <i>float</i> ; 3 – <i>bit</i>)
BitPosition	Pozycja bitu (dotyczy dostępu bitowego do 16-bitowych rejestrów)
BitCount	Liczba bitów rejestru do odczytania
RefreshInterval	Czas odświeżania
ResponseTimeout	Czas na odpowiedź
Divisor	Dzielnik
Endianness	Kolejność bajtów i słów ⁷ : <i>No swap</i> (0 – bez zamiany); <i>Swap bytes and words</i> (1 - zamiana kolejności bajtów i słów); <i>Swap bytes</i> (2 - zamiana kolejności bajtów w obrębie każdego ze słów); <i>Swap words</i> (3 - zamiana słów)
RegisterType	Typ rejestru Modbus (0 – wejścia/wyjścia bitowe, 1 – wejścia dwustanowe, 2 – rejestry pamiętające, 3 – rejestry wejściowe)
ErrorCode	Kod błędu: (-3 - błąd ramki; -2 - przekroczenie czasu odpowiedzi; -1 - nieaktualna wartość ostatniego odczytanego rejestru; 0 – poprawny odczyt/zapis rejestru; 1 - niedozwolona funkcja; 2 - niedozwolony numer rejestru; 3 - niedozwolona wartość danej; 4 - uszkodzenie w przyłączonym urządzeniu; 5 - potwierdzenie pozytywne; 6 - brak gotowości/komunikat usunięty; 7 - potwierdzenie negatywne; 8 - błąd parzystości pamięci)
Value	Wartość odczytu/zapisu
RegisterValue	Nieprzeskalowana wartość rejestru

METODY

Nazwa	Opis
SetDateTime	Ustawia datę i czas
SetClientReportInterval	Ustawia okres raportowania o zmianach cech
SetDeviceAddress	Ustawia adres urządzenia Modbus typu Slave
SetAccessRights	Ustawia tryb pracy: odczyt lub odczyt/zapis
SetRegisterAddress	Ustawia adres obsługiwanego rejestru
SetTransmissionSpeed	Ustawia prędkość transmisji
SetValueType	Ustawia typ zmiennej
SetBitPosition	Ustawia pozycję bitu
SetBitCount	Ustawia liczbę bitów rejestru do odczytania
SetRefreshInterval	Ustawia czas odświeżania
SetResponseTimeout	Ustawia czas oczekiwania na odpowiedź
SetDivisor	Ustawia dzielnik
SetEndianness	Ustawia typ kolejności bajtów
SetRegisterType	Ustawia typ rejestru Modbus
SetValue	Ustawia wartość odczytu/zapisu

ZDARZENIA

Nazwa	Opis
OnInit	Zdarzenie wywołane w momencie inicjalizacji urządzenia
OnChange	Zdarzenie wywołane przy zmianie stanu (niezależnie od wartości)
OnError	Zdarzenie wywołane, gdy urządzenie slave zgłasza błąd

XV. Moduł GATE HTTP

1. Informacje ogólne

Moduł GATE Http to urządzenia umożliwiające systemową integrację z zewnętrznymi serwisami posługującymi się protokołem HTTP, a także szeroką grupą urządzeń i systemów zewnętrznych/trzecich firm - np. urządzeń AV z interfejsami HTTP.

2. Konfiguracja modułu GATE HTTP

UWAGA! Przed rozpoczęciem jakiegokolwiek pracy z modułem GATE HTTP wymagana jest aktualizacja bazy interfejsów!

2.1 Obiekty wirtualne

2.1.1. HTTPRequest

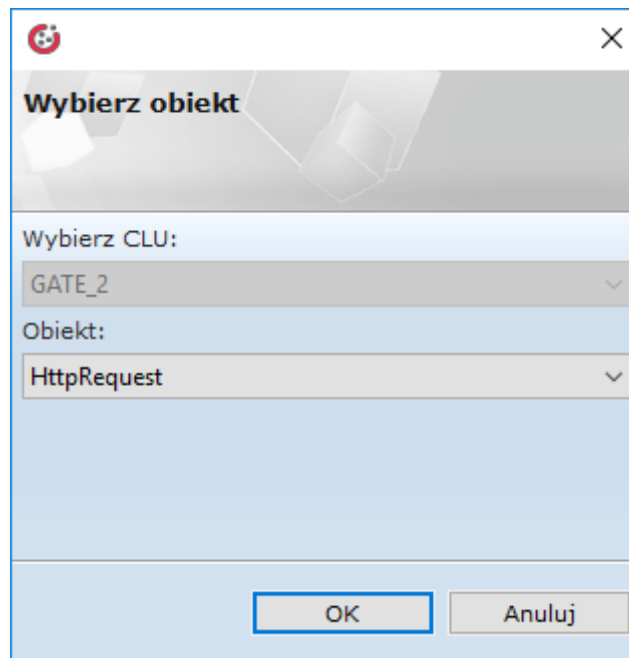
Dla HttpRequest przykładowo wykorzystywany jest serwis pogodowy <http://api.openweathermap.org>

Według przykładu na stronie openweathermap.org, zapytanie API wygląda następująco:

API call: <http://api.openweathermap.org/data/2.5/weather?q=London&APPID={APIKEY}>

HttpRequest - służy do wysyłania zapytań HTTP (typu GET, POST) do określonego hosta. Obsługiwane są standardowe Typy zawartości (content-type) m.in. JSON, XML.

Aby zastosować moduł Gate do odbierania zapytań, należy utworzyć obiekt wirtualny HttpRequest:



- W obiekcie HttpRequest należy ustawić następujące parametry:

GATE_2->http_request

Nazwa: Typ:

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
Host	http://api.openweathermap.org	<input type="text" value="http://api.openweathermap.org"/>	string	
Path	/data/2.5/weather	<input type="text" value="/data/2.5/weather"/>	string	
QueryStringParams	-	<input type="text" value="q=London,uk&APPID={APIKEY}"/>	string	
Method	GET	<input type="text" value="GET"/>	string	
Timeout	5	<input type="text" value="5"/>	s	[1-255]
RequestType	2	<input type="text" value="JSON"/>	-	0,1,2,3,4,5
ResponseType	2	<input type="text" value="JSON"/>	-	0,1,2,3,4,5
RequestHeaders	-	<input type="text" value="\z"/>	string	
RequestBody	-	<input type="text" value="\z"/>	string	
ResponseBody	-	<input type="text" value="\z"/>	string	
StatusCode	0		-	

Auto odświeżanie

- **Host:** api.openweathermap.org
- **Path:** /data/2.5/weather
- **QueryStringParams:** q=London&APPID={APIKEY}
- **Method:** GET
- **RequestType:** JSON
- **ResponseType:** JSON

UWAGA! Obiekt Gate Http umożliwia obsługę połączeń szyfrowanych TLS. Jeżeli wymagane jest takie połączenie, należy na początku wartości w polu Host podać 'https://'. Jeżeli wartość nie zostanie podana, zostanie wykorzystane standardowe połączenie http.

UWAGA! Gate Http nie obsługuje wszystkich połączeń szyfrowanych TLS, dlatego zalecamy przetestowanie połączenia z danym hostem.

UWAGA! Podczas połączenia https czas nawiązania połączenia oraz otrzymania odpowiedzi od hosta jest dłuższy niż w przypadku połączenia http, dlatego należy zwiększyć wartość dla parametru Timeout.

UWAGA! Cechy opisane jako nieustawialne są cechami zawierającymi odpowiedzi. Wartości początkowe tych cech należy pozostawić niezmiennymi. Wszelkie operacje na tych zmiennych należy wykonywać na skryptach (oraz zmiennych lokalnych).

Po wysłaniu konfiguracji i wywołaniu Metody SendRequest, StatusCode przyjmuje wartość 200 (OK).

Otrzymana odpowiedź na zapytanie jest przetrzymywana w cesze `ResponseBody`. Dla ustawionego `ResponseType JSON`, odpowiedź jest parsowana z json do tabeli. Wartość cechy jest niewidoczna z poziomu OM. Wartości odpowiedzi należy wyciągnąć z odpowiedzi z poziomu skryptu.

2.1.2. Pobieranie określonych wartości z otrzymanej odpowiedzi (XML,JSON)

UWAGA! Uzyskaną odpowiedź `ResponseBody` należy przypisać do zmiennej lokalnej (w skrypcie).

Przykładowo:

```
local resp = GATE->http_openweather_json->ResponseBody
```

Następnie w skryptach należy wykonywać operację na zmiennej `resp`!

Otrzymane odpowiedzi w zależności od ich typu (`ResponseType`) są odpowiednio parsowane do postaci tabeli.

Przykładowe odczyty wartości są zapisywane do zmiennych lokalnych (wewnątrz skryptu).

Aby była możliwość wykorzystania zmiennej (np. do wyświetlania w aplikacji), należy przypisywać do zmiennych globalnych (cech użytkownika).

Poniżej przykładowe odpowiedzi w formacie XML oraz JSON oraz sposób odczytania danej wartości (w przedstawionych przykładach wykorzystano odpowiedzi z serwisu pogodowego openweathermap.org)

A. JSON:

Przykładowa odpowiedź (openweathermap.org):

```
resp = [[
{"coord":
{"lon":145.77,"lat":-16.92},
"weather":[{"id":803,"main":"Clouds","description":"broken
cLouds","icon":"04n"}],
"base":"cmc stations",
"main":
{"temp":293.25,"pressure":1019,"humidity":83,"temp_min":289.82,"temp_max":295.37
},
"wind":{"speed":5.1,"deg":150},
"clouds":{"all":75},
"rain":{"3h":3},
"dt":1435658272,
"sys":
{"type":1,"id":8166,"message":0.0166,"country":"AU","sunrise":1435610796,"sunset
":1435650870},
"id":2172797,
"name":"Cairns",
"cod":200}
]]
```

Jak odczytać:

- Wartość parametru **lon**

```
{ "coord":  
  { "lon":145.77,"lat":-16.92},  
  "weather":[{"id":803,"main":"Clouds","description":"broken  
clouds","icon":"04n"}],  
  "base":"cmc stations",  
  "main":  
  { "temp":293.25,"pressure":1019,"humidity":83,"temp_min":289.82,"temp_max":295.37  
  },  
}
```

W skrypcie:

```
local lon = resp.coord.lon
```

Po wywołaniu skryptu do zmiennej lokalnej (zmienna skryptu) zostanie przypisana wartość 145.77

- Wartość parametru **description**

```
{ "coord":  
  { "lon":145.77,"lat":-16.92},  
  "weather":[{"id":803,"main":"Clouds","description":"broken  
clouds","icon":"04n"}],  
  "base":"cmc stations",  
  "main":  
  { "temp":293.25,"pressure":1019,"humidity":83,"temp_min":289.82,"temp_max":295.37  
  },  
}
```

W skrypcie:

```
local description = resp.weather[1].description
```

Po wywołaniu skryptu do zmiennej lokalnej (zmienna skryptu) zostanie przypisana wartość „broken clouds”.

B. XML:

Przykładowa odpowiedź (openweathermap):

```
resp= [[  
<current>  
  <city id="2643741" name="City of London">  
    <coord lon="-0.09" lat="51.51">  
    <country>GB</country>  
    <sun rise="2015-06-30T03:46:57" set="2015-06-30T20:21:12">  
  </city>  
  <temperature value="72.34" min="66.2" max="79.88" unit="fahrenheit"/>  
  <humidity value="43" unit="%">  
  <pressure value="1020" unit="hPa">  
  <wind>  
    <speed value="7.78" name="Moderate breeze">  
    <direction value="140" code="SE" name="SouthEast">  
  </wind>
```

```
<clouds value="0" name="clear sky">
<visibility value="10000">
<precipitation mode="no">
<weather number="800" value="Sky is Clear" icon="01d">
<lastupdate value="2015-06-30T08:36:14">
</current>
]]
```

Jak odczytać:

- Wartość atrybutu id w tagu **city**

```
<current>
  <city id="2643741" name="City of London">
    <coord lon="-0.09" lat="51.51">
      <country>GB</country>
    <sun rise="2015-06-30T03:46:57" set="2015-06-30T20:21:12">
  </city>
```

W skrypcie:

```
local city_id = resp[1].id
```

Po wywołaniu skryptu do zmiennej lokalnej (zmienna skryptu) zostanie przypisana wartość 2643741.

- Wartość znajdująca się pomiędzy tagiem

```
<current>
  <city id="2643741" name="City of London">
    <coord lon="-0.09" lat="51.51">
      <country>GB</country>
    <sun rise="2015-06-30T03:46:57" set="2015-06-30T20:21:12">
  </city>
```

W skrypcie:

```
local country = resp[1][2][1]
```

Po wywołaniu skryptu do zmiennej lokalnej (zmienna skryptu) zostanie przypisana wartość „GB”.

- Nazwa tagu

```
<current>
  <city id="2643741" name="City of London">
    <coord lon="-0.09" lat="51.51">
      <country>GB</country>
    <sun rise="2015-06-30T03:46:57" set="2015-06-30T20:21:12">
  </city>
```

W skrypcie:

```
local nameTag = resp[1][2].xmlTag
```

Po wywołaniu skryptu do zmiennej lokalnej (zmienna skryptu) zostanie przypisana wartość „country”.

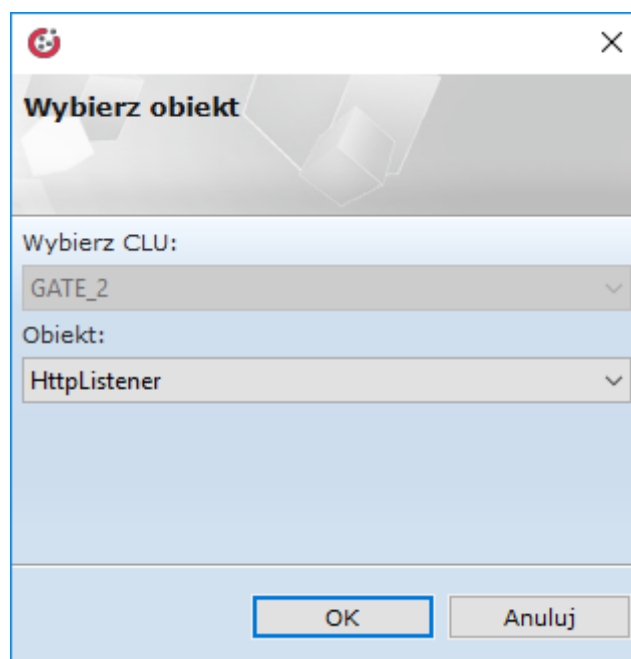
2.2.1. HttpListener

Obiekt HttpListener służy do otrzymywania zapytań HTTP (typu GET, POST). Wysyłana odpowiedź zwrotna może być serializowana do jednego ze standardowych typów m.in. JSON, XML. W obiekcie HttpListener ważne jest, aby na każdy przychodzący Request odesłać odpowiedź (Response).

W przypadku nasłuchiwanie na zapytanie Request do modułu Gate – przykładowo (korzystając np. z przeglądarki internetowej):

GET 192.168.4.12/grentontest/xml

Należy utworzyć obiekt wirtualny HttpListener



GATE_2->http_listener

Nazwa: Typ:

Sterowanie
 Zdarzenia
 Cechy wbudowane

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
Path	-	<input type="text" value="/grentontest/xml"/>	string	
Method	-		string	
QueryStringParams	-	<input type="text" value="\z"/>	string	
RequestType	-		-	0,1,2,3,4,5
RequestBody	-	<input type="text" value="\z"/>	string	
ResponseType	-	<input type="text" value="XML"/>	-	0,1,2,3,4
ResponseBody	-	<input type="text" value="\z"/>	string	
StatusCode	-	<input type="text" value="200"/>	-	

Auto odświeżanie

W obiekcie HttpListener należy ustawić następujące parametry:

- **Path:** /grentontest/xml
- **ResponseType:** XML
- **StatusCode:** 200

UWAGA!

Cechy opisane jako nieustawialne są cechami zawierającymi odpowiedzi. Wartości początkowe tych cech należy pozostawić niezmienione. Wszelkie operacje na tych zmiennych należy wykonywać na skryptach (oraz zmiennych lokalnych)

Do zdarzenia OnRequest należy utworzyć skrypt, który będzie tworzył poprawną odpowiedź i wysyłał ją zwrótnie.

2.2.2. Przygotowanie odpowiedzi wysyłanej do serwera

Odpowiedź jest tworzona w zmiennej lokalnej resp.

Po przygotowaniu odpowiedzi należy ją ustawić dla cechy ResponseBody(resp), a następnie wysłać za pomocą metody SendResponse()

A. XML:

Aby w odpowiedzi wysłać wartość danej cechy:

```

local resp = "<clu><temperature>" .. CLUZ->x103478262_ONEW_SENSOR1->Value .. "
</temperature></clu>"
GATE_2->Listener_XML->SetResponseBody(resp)
GATE_2->Listener_XML->SendResponse()
  
```

Przesłana odpowiedź wygląda następująco:


```
<clu>
  <temperature>22.5</temperature>
</clu>
```

B.JSON:

```
local resp = {
  Temp = CLUZ->x103478262_ONEW_SENSOR1->Value
}
GATE_2->Listener_JSON->SetResponseBody(resp)
GATE_2-> Listener_JSON->SendResponse()
```

Przesłana odpowiedź wygląda następująco:

```
{"Temp": 22.6}
```

2.2.3. Odczyt wartości kluczy z parametru querystringparams

Zgodnie z opisem cechy QueryStringParams jej wartość nie jest ustawialna, można odczytać ją w skrypcie. Jeżeli w zapytaniu zostaje wysłane querystring z kluczami (keys), to z poziomu skryptu można odczytać daną wartość – jest zapisana w postaci tabeli.

Poszczególne wartości kluczy można uzyskać na zasadzie:

```
value1 = qs.klucz1
```

Dla otrzymanego zapytania:

192.168.1.12/grentontest/query?light1=on&light2=off&light3=on

Należy utworzyć skrypt:

```
local qs = HTTP_L->grentontest_query_listener->QueryStringParams

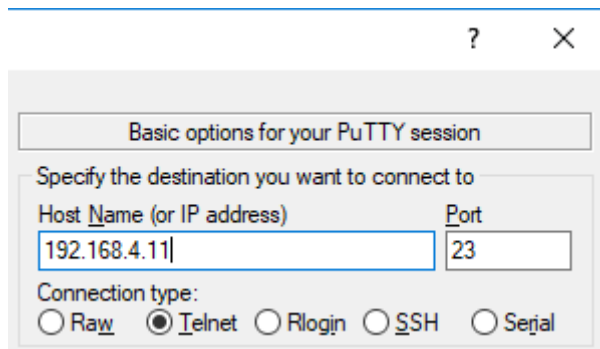
local test0 = qs.light1
local test1 = qs.light2
local test2 = qs.light3

HTTP_L->grentontest_query_listener->SetResponseBody()
HTTP_L->grentontest_query_listener->SendResponse()
```

Wszystkie wartości kluczy zostaną zapisane w zmiennych lokalnych (test0, test1, test2).

3. Możliwość połączenia z Gate za pomocą TELNET

Dla modułu Gate Http możliwy jest podgląd skryptów Lua. W przypadku błędu konfiguracji (tryb emergency) możliwy jest podgląd miejsca błędu w utworzonej konfiguracji LUA. Połączenie jest nawiązywane za pomocą protokołu Telnet – w tym celu można wykorzystać np. program PuTTY. Przykładowe parametry do nawiązania połączenia:



Do wywołania połączenia po stronie Gate'a wykorzystane mogą być dwie metody:

- `StartConsole` – Uruchamia konsolę Lua. W momencie wywołania metody, użytkownik ma 10s na ustawienie połączenia z Gate. Przy poprawnym połączeniu, na terminalu (klient) zostanie zwrócona informacja o poprawnym połączeniu:

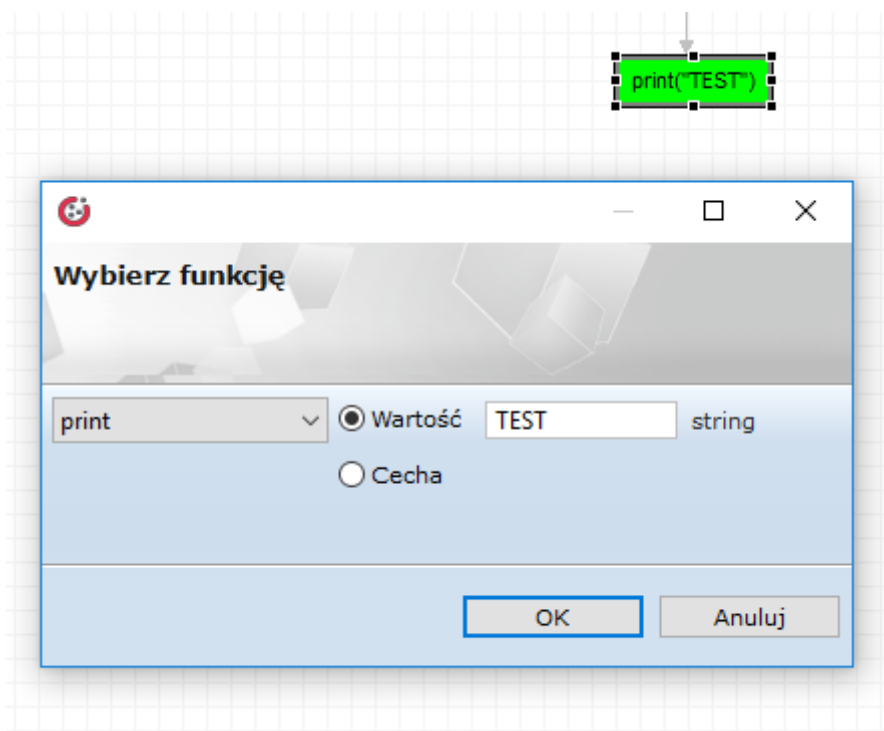
```
CLU SN Telnet session started.
```

- `StartConsoleOnReboot` – umożliwia nawiązanie połączenia przy następnym restarcie Gate'a. Po restarcie użytkownik ma 10s na ustawienie połączenia z Gate. Przy poprawnym połączeniu, na terminalu (klient) zostanie zwrócona informacja o poprawnym połączeniu

```
CLU SN initializing...
CLU: running user.lua...
CLU: running om.lua...
CLU: running OnInit...
CLU: Project loaded.
```

Uwaga! Niezalecane jest przypisanie metody `StartConsole` oraz `StartConsoleOnReboot` do zdarzenia `onInit` modułu GATE Http.

Aby na konsoli wyświetlić np. wartość danej cechy, należy wykorzystać komponent *Blok funkcyjny* i wybrać metodę *Print*, a następnie wybrać żadaną cechę.



4. Przywracanie ustawień fabrycznych - *Hard Reset*

Uruchomienie funkcji *Hard Reset* na module GATE Http powoduje:

- Usunięcie zapisanej konfiguracji;
- Sformatowanie partycji pamięci flash;
- Usunięcie wszystkich utworzonych obiektów LUA;
- Utratę komunikacji pomiędzy OM / HM a modulem Gate.

W celu przywrócenia ustawień fabrycznych funkcją *Hard Reset* należy wykonać następujące czynności (zgodnie z podaną kolejnością):

- Odłączyć zasilanie od modułu Gate;
- Nacisnąć i przytrzymać przycisk *Reset* na module (przycisk znajduje się pod dolną zaślepką modułu);
- Podłączyć zasilanie do modułu Gate;
- Trzymać wciśnięty przycisk *Reset* przez co najmniej 10 sekund - podczas resetu dioda zielona będzie świecić światłem ciągłym. Prawidłowe wykonanie resetu zostanie potwierdzone 3-krotnym mrugnięciem diody zielonej.
- Po upływie 10 sekund zwolnić przycisk *Reset*
- Odczekać około 60 sekund aż do momentu, gdy na module diody - zielona oraz czerwona - będą mrugać naprzemiennie (tryb *Emergency*)

Po wykonaniu procedury na module zostanie wyczyszczona konfiguracja, natomiast sam moduł przestanie być widoczny (brak odpowiedzi na *Keep-Alive*) w projekcie z poziomu Object Managera. Aby ponownie przywrócić moduł, należy wykonać CLU Discovery a następnie wysłać konfigurację.

5. Parametry konfiguracyjne

A. Obiekt GATE

CECHY

Nazwa	Opis
<code>Uptime</code>	Czas pracy urządzenia od ostatniego resetu (w sekundach)
<code>UnixTime</code>	Aktualny uniksowy znacznik czasu
<code>LibraryVersion</code>	Wersja oprogramowania Gate
<code>ClientReportInterval</code>	Okres raportowania o zmianach cech

METODY:

Nazwa	Opis
<code>SetDateTime</code>	Ustawia datę i czas
<code>SetClientReportInterval</code>	Ustawia okres raportowania o zmianach cech
<code>SetUpdateTime</code>	Ustawia datę i czas, co jaki stan centralki jest uaktualniany
<code>StartConsole</code>	Uruchamia konsolę Lua
<code>StartConsoleOnReboot</code>	Uruchamia konsolę Lua przy kolejnym uruchomieniu modułu

ZDARZENIA:

Nazwa	Opis
<code>OnInit</code>	Zdarzenie wywoływane jednorazowo w momencie inicjalizacji urządzenia

B. Obiekt HttpRequest

Uwaga! Cechy opisane jako nieustawialne są cechami zawierającymi odpowiedzi. Wartości początkowe tych cech należy pozostawić niezmienione. Wszelkie operacje na tych zmiennych należy wykonywać na skryptach (oraz zmiennych lokalnych).

CECHY

Nazwa	Opis
Host	Adres hosta
Path	Ścieżka zapytania
QueryStringParams	Parametry zapytania. \z oznacza brak
Method	Typ metody wysyłanej w zapytaniu np. GET, POST
Timeout	Dopuszczalny czas odpowiedzi
RequestType	<p>Typ zawartości wysyłanego zapytania. Definiuje parametr <i>content-type</i> w nagłówku zapytania. W zależności od wybranego typu zawartość cechy <code>RequestBody</code> jest odpowiednio serializowana:</p> <p>0 - <code>None</code> - niezdefiniowany. W nagłówku nie jest wysłane <i>content-type</i>. Zawartość cechy <code>RequestBody</code> nie jest serializowana.</p> <p>1 - <code>Text</code> - <i>content-type: text/plain</i>. Zawartość cechy <code>RequestBody</code> nie jest serializowana.</p> <p>2 - <code>JSON</code> - <i>content-type: application/json</i>. Zawartość cechy <code>RequestBody</code> jest serializowana do formatu JSON.</p> <p>3 - <code>XML</code> - <i>content-type: text/xml</i>. Zawartość cechy <code>RequestBody</code> jest serializowana do formatu XML.</p> <p>4 - <code>FormData</code> - <i>content-type: application/x-www-form-urlencoded</i>. Zawartość cechy <code>RequestBody</code> jest serializowana do tabeli.</p> <p>5 - <code>Other</code> - typ zawartości (<i>content-type</i>) jest inny niż wbudowany. Typ można zdefiniować umieszczając go w nagłówku (cecha <code>RequestHeaders</code>). Zawartość nie jest serializowana.</p>
ResponseType	<p>Typ oczekiwanej odpowiedzi. Definiuje parametr <i>Accept</i> w nagłówku zapytania. W zależności od wybranego typu zawartość otrzymanej odpowiedzi (cecha <code>ResponseBody</code>) jest odpowiednio parsowana do tabeli:</p> <p>0 - <code>None</code> - parametr <i>Accept</i> nie jest wysyłany w nagłówku wysyłanego zapytania. Odpowiedź (cecha <code>ResponseBody</code>) nie jest parsowana.</p> <p>1 - <code>Text</code> - <i>Accept: text/plain</i>. Odpowiedź (cecha <code>ResponseBody</code>) nie jest parsowana.</p> <p>2 - <code>JSON</code> - <i>Accept: application/json</i>. Odpowiedź (cecha <code>ResponseBody</code>) jest parsowana z JSON.</p> <p>3 - <code>XML</code> - <i>Accept: text/xml</i>. Odpowiedź (cecha <code>ResponseBody</code>) jest parsowana z XML.</p> <p>4 - <code>FormData</code> - <i>Accept: application/x-www-form-urlencoded</i>. Odpowiedź (cecha <code>ResponseBody</code>) jest parsowana.</p> <p>5 - <code>Other</code> - parametr <i>Accept</i> nagłówka jest inny niż wbudowany. Parametr można zdefiniować umieszczając go w nagłówku (cecha <code>RequestHeaders</code>).</p>
RequestHeaders	Dodatkowe nagłówki zapytania HTTP. \z oznacza brak zawartości.
RequestBody	Zawartość wiadomości wysyłanej w zapytaniu. \z oznacza brak zawartości

Nazwa	Opis
<code>ResponseBody</code>	Zawartość wiadomości otrzymanej po wysłaniu zapytania. (cecha wykorzystywana do odczytu w skryptach - nieustawialna)
<code>StatusCode</code>	Status odpowiedzi HTTP

METODY

Nazwa	Opis
<code>SendRequest</code>	Wysyła zapytanie
<code>AbortRequest</code>	Przerywa obsługę zapytania
<code>Clear</code>	Usuwa treść zapytania
<code>SetHost</code>	Ustawia adres hosta
<code>SetPath</code>	Ustawia ścieżkę zapytania
<code>SetQueryStringParams</code>	Ustawia parametry zapytania
<code>SetMethod</code>	Ustawia metodę zapytania
<code>SetTimeout</code>	Ustawia dopuszczalny czas odpowiedzi
<code>SetRequestType</code>	Ustawia typ zawartości wysłanego zapytania (content-type)
<code>SetResponseType</code>	Ustawia typ oczekiwanej odpowiedzi na zapytanie
<code>SetRequestHeaders</code>	Ustawia dodatkowe nagłówki HTTP w zapytaniu
<code>SetRequestBody</code>	Ustawia zawartość wiadomości w zapytaniu

ZDARZENIA

Nazwa	Opis
<code>OnRequestSent</code>	Zdarzenie wywoływane w momencie wysłania zapytania
<code>OnResponse</code>	Zdarzenie wywoływane w momencie otrzymania odpowiedzi

C. Obiekt `HttpListener`

Uwaga! Cechy opisane jako nieustawialne są cechami zawierającymi odpowiedzi. Wartości początkowe tych cech należy pozostawić niezmienione. Wszelkie operacje na tych zmiennych należy wykonywać na skryptach (oraz zmiennych lokalnych)

CECHY

Nazwa	Opis
Path	Ścieżka zapytania
Method	Typ metody otrzymanej w zapytaniu np. GET , POST
QueryStringParams	Zwraca parametry zapytania HTTP (cecha wykorzystywana do odczytu w skryptach - nieustawialna)
RequestType	<p>Typ otrzymanego zapytania. W zależności od wybranego typu, zawartość otrzymanego zapytania (cechy RequestBody) jest odpowiednio parsowana do tabeli:</p> <p>0 - None - Odpowiedź nie jest parsowana.</p> <p>1 - Text - Odpowiedź nie jest parsowana.</p> <p>2 - JSON - Odpowiedź jest parsowana z JSON.</p> <p>3 - XML - Odpowiedź jest parsowana z XML.</p> <p>4 - FormData - Odpowiedź jest parsowana.</p> <p>5 - Other - Odpowiedź nie jest parsowana. Cecha RequestBody zwraca treść zapytania HTTP (cecha wykorzystywana do odczytu w skryptach - nieustawialna).</p>
RequestBody	Zwraca treść zapytania HTTP (cecha wykorzystywana do odczytu w skryptach - nieustawialna)
ResponseType	<p>Typ zawartości wysłanej odpowiedzi na zapytanie. Definiuje parametr <i>content-type</i> w nagłówku odpowiedzi. W zależności od wybranego typu, zawartość cechy ResponseBody jest odpowiednio serializowana:</p> <p>0 - None - niezdefiniowany. W nagłówku nie jest wysłane <i>content-type</i>. Zawartość nie jest serializowana.</p> <p>1 - Text - <i>content-type: text/plain</i>. Zawartość nie jest serializowana.</p> <p>2 - JSON - <i>content-type: application/json</i>. Zawartość RequestBody jest serializowana do formatu JSON.</p> <p>3 - XML - <i>content-type: text/xml</i>. Zawartość RequestBody jest serializowana do formatu XML.</p> <p>4 - FormData - <i>content-type: application/x-www-form-urlencoded</i>. Zawartość RequestBody jest serializowana.</p> <p>5 - Other - parametr <i>Accept</i> nagłówka jest inny niż wbudowany. Parametr można zdefiniować umieszczając go w nagłówku (cecha RequestHeaders).</p>
ResponseBody	Zwraca treść odpowiedzi HTTP (cecha wykorzystywana do odczytu w skryptach).

Nazwa	Opis
StatusCode	Status wysyłanej odpowiedzi HTTP. Obsługiwane statusy: 200 - OK 201 - Utworzono 202 - Przyjęto 204 - Brak zawartości 205 - Przywróć zawartość 400 - Nieprawidłowe zapytanie 403 - Zabroniony 404 - Nie znaleziono 405 - Nieozwolona metoda 406 - Nieozwolone 408 - Koniec czasu oczekiwania na żądanie 409 - Konflikt 410 - Zniknął (usunięto)

METODY

Nazwa	Opis
SendResponse	Wysyła odpowiedź na zapytanie
Clear	Usuwa treść odpowiedzi
SetPath	Ustawia ścieżkę zapytania
SetResponseType	Ustawia typ oczekiwanej odpowiedzi na zapytanie
SetResponseBody	Ustawia treść odpowiedzi
SetStatusCode	Ustawia status odpowiedzi

ZDARZENIA

Nazwa	Opis
OnRequest	Zdarzenie wywołane w momencie otrzymania zapytania

XVI. Moduły Z-Wave

Niniejszy rozdział przedstawia opis zakresu wsparcia modułów Z-Wave innych producentów, które są dostępne w systemie Grenton.

UWAGA! Pełna lista urządzeń jest dostępna na stronie <https://support.grenton.pl/pl/support/solutions> w artykule 'Jakie bezprzewodowe moduły Z-Wave są obsługiwane?'

1. Fibaro UBS

Wersja modułu: FGBS-001 v2.1.

1.1. Informacje ogólne

Moduł Z-Wave Fibaro UBS posiada dwa wejścia bezpotencjałowe. Umożliwia odczyt wartości nawet z czterech czujników 1-Wire. Ponadto daje możliwość zmiany parametrów konfiguracyjnych (interfejs konfiguracyjny Fibaro).

UWAGA! Dodawanie/usuwanie odbywa się przez trzykrotne kliknięcie przycisku w module podczas inclusion/exclusion.

1.2. Obiekty

A. ZWAVE_DIN

Wejścia bezpotencjałowe.

CECHY

Nazwa	Opis
<code>value</code>	Zwraca stan wejścia
<code>holdDelay</code>	Czas, po którym wciśnięcie i przytrzymanie przycisku wywoła zdarzenie <code>onHold</code>
<code>holdInterval</code>	Odstęp cykliczny (w ms), po jakim podczas trzymania przycisku wyzwalane są kolejne zdarzenia <code>onHold</code>

METODY

Nazwa	Opis
<code>setHoldDelay</code>	Ustawia wartość <code>holdDelay</code>
<code>setHoldInterval</code>	Ustawia wartość <code>holdInterval</code>

ZDARZENIA

Nazwa	Opis
<code>onChange</code>	Zdarzenie wywoływane w przypadku zmiany stanu na przeciwny
<code>onSwitchOn</code>	Zdarzenie wywoływane w momencie ustawienia stanu wysokiego na wejściu
<code>onSwitchOff</code>	Zdarzenie wywoływane w momencie ustawienia stanu niskiego na wejściu
<code>onShortPress</code>	Zdarzenie wywoływane po naciśnięciu przycisku na okres 500-2000ms
<code>onLongPress</code>	Zdarzenie wywoływane po naciśnięciu przycisku na okres 2000-5000ms
<code>onHold</code>	Zdarzenie wywoływane, gdy wejście jest w stanie wysokim, pierwszy raz po upływie czasu <code>holdDelay</code> , a następnie cyklicznie co wartość <code>holdInterval</code>
<code>onClick</code>	Zdarzenie wywoływane po naciśnięciu przycisku na czas krótszy niż 500ms

B. ZWAVE_1W_SENSOR

Obiekt odpowiada za czujnik 1-Wire. Dla każdego czujnika tworzony jest osobny obiekt. Do modułu UBS Fibaro można podłączyć maksymalnie 4 czujniki 1-Wire (DS18B20).

Obiekty ZWAVE_1W_SENSOR są zawsze dodawane wraz z dodaniem modułu Fibaro UBS do CLU/projektu w OM, niezależnie od ilości podłączonych czujników. O tym, czy dany czujnik jest podłączony, informuje cecha `Discovered` – zwracająca informację, czy podczas Discovery pod dany obiekt zgłosił się czujnik 1-Wire i jest podłączony do modułu UBS.

Przy podłączeniu lub odłączeniu czujników 1-Wire, należy usunąć a następnie ponownie dodać moduł UBS do modułu CLU Z-Wave. Moduł Fibaro UBS zgłosi się z nowym numerem seryjnym – istnieje możliwość przepisania konfiguracji obiektów (automatyczna lub ręczna). Po ponownym dodaniu czujników może nastąpić ponowne przeindeksowanie kolejności czujników do obiektów ZW_1W_SENSOR.

Moduł Fibaro UBS dla czujnika 1-Wire nie zwraca informacji, czy podczas działania systemu czujnik został rozłączony – przechowywana jest ostatnia pobrana wartość, dlatego niezalecane jest wykorzystywanie tych czujników jako źródło regulacji temperatury.

W momencie zwarcia na magistrali 1-Wire, wszystkie czujniki podłączone do modułu Fibaro UBS (dostępne/widoczne w OM) zwracają wartość 0.00 – dlatego przy dłuższym (nieplanowanym) wystąpieniu tej wartości należy sprawdzić poprawność podłączenia magistrali 1-Wire.

CECHY

Nazwa	Opis
<code>value</code>	Wartość wejścia
<code>minValue</code>	Minimalna wartość wejścia
<code>maxValue</code>	Maksymalna wartość wejścia
<code>Discovered</code>	Informacja zwracana podczas CLU Discovery o podłączeniu czujnika do modułu

ZDARZENIA

Nazwa	Opis
<code>OnChange</code>	Zdarzenie wywoływane przy zmianie wartości wyjścia
<code>OnRise</code>	Zdarzenie wywoływane po przekroczeniu górnego progu histerezy (złocze rosnące)
<code>OnLower</code>	Zdarzenie wywoływane po przekroczeniu dolnego progu histerezy (złocze opadające)
<code>OnOutOfRange</code>	Zdarzenie wywoływane, gdy wartość na wyjściu znajduje się poza wyznaczonym zakresem (<code>minValue</code> : <code>maxValue</code>)
<code>OnInRange</code>	Zdarzenie wywoływane przy powrocie wartości do przedziału wewnątrz wartości progowych (<code>minValue</code> : <code>maxValue</code>)

C. ZWAVE_CONFIG

Obiekt wyświetla informacje odnośnie parametrów oraz komunikacji z modułem w sieci Z-Wave. Umożliwia ustawienie zaawansowanych parametrów konfiguracyjnych danego modułu (określone indywidualnie w instrukcji).

CECHY

Nazwa	Opis
NodeID	Numer modułu (węzła) w sieci Z-Wave (nadawany dla każdego modułu Z-Wave po dodaniu go do kontrolera)
Banned	Informacja o zablokowaniu komunikacji Z-Wave z modułem: 0 – komunikacja z modułem nie jest zablokowana, 1 – zablokowana komunikacja z modułem (moduł zbanowany). Zablokowanie następuje w momencie, gdy 3 kolejne próby komunikacji z modułem zakończą się niepowodzeniem (inkrementacja cechy <code>FailCount</code> o 3). Do zbanowanego modułu, co 1,5 minuty wysyłane jest zapytanie – jeśli CLU dostanie odpowiedź, wówczas zablokowanie zostanie usunięte i możliwa jest ponowna próba wysłania rozkazu do modułu
FailCount	Liczba nieudanych prób komunikacji z modułem Z-Wave. W przypadku niepowodzenia komunikacji z modułem (brak odpowiedzi, potwierdzenia itp.) następuje inkrementacja cechy o 1, następnie próba ponawiania jest dwukrotnie (w interwałach 15s). W przypadku niepowodzenia komunikacja z modułem zostaje zablokowana (<code>Banned</code> =1)
Register	Numer rejestru (parametru) konfiguracyjnego, który ostatnio został odczytany/ustawiony za pomocą dostępnych metod
value	Wartość rejestru (parametru) konfiguracyjnego

METODY

Nazwa	Opis
<code>RemoveBan</code>	Zdejmuje blokadę komunikacji z modułem Z-Wave (w przypadku, gdy cecha <code>Banned</code> =1). Wywołanie metody umożliwi ponowne wysłanie rozkazu do modułu. Uwaga! Cecha <code>RemoveBan</code> nie jest jednoznaczne z ponowną poprawną komunikacją z modułem – umożliwia ponowne wysłanie rozkazu/zapytania do modułu! W przypadku niepowodzenia cały proces blokowania jest rozpoczynany na nowo!
<code>ClearFailCount</code>	Czyści liczbę nieudanych prób komunikacji
<code>Set</code>	Ustawia wartość danego rejestru (parametru) konfiguracyjnego: 1 - <code>Register</code> (numer rejestru lub parametru), 2 - <code>value</code> (wartość rejestru bądź parametru), 3 - <code>size</code> (rozmiar wysyłanej wartości rejestru lub parametru – w bajtach)
<code>Get</code>	Pobiera wartość danego rejestru (parametru) konfiguracyjnego
<code>SetDefault</code>	Ustawia wartość domyślną dla danego rejestru (parametru) konfiguracyjnego

ZDARZENIA

Nazwa	Opis
<code>OnBanned</code>	Zdarzenie wywoływane gdy urządzenie zostanie zbanowane

3. NEO Coolcam Motion Sensor (PIR)

Wersja modułu: NAS-PD01ZE HW: 66 FW: 3.80

3.1. Informacje ogólne

Moduł Z-Wave Neo Coolcam Motion Sensor umożliwia odczyt: stanu czujnika ruchu (PIR), poziomu oświetlenia oraz poziomu baterii. Ponadto daje możliwość ustawienia/odczytu czasu wybudzenia modułu.

UWAGA! Dodawanie/usuwanie odbywa się przez trzykrotne kliknięcie przycisku w module Neo podczas inclusion/exclusion. Poprawnie przeprowadzony proces zostanie potwierdzony pięciokrotnym mrugnięciem diody.

3.2. Obiekty

A. BINARY_SENSOR

Obiekt umożliwia odczyt stanu czujnika ruchu.

CECHY

Nazwa	Opis
<code>value</code>	Zwraca stan wejścia: 0 - brak naruszenia, 1 - naruszenie

ZDARZENIA

Nazwa	Opis
<code>OnChange</code>	Zdarzenie wywoływane w przypadku zmiany stanu na przeciwny
<code>OnSwitchOn</code>	Zdarzenie wywoływane w momencie ustawienia stanu wysokiego na wejściu
<code>OnSwitchOff</code>	Zdarzenie wywoływane w momencie ustawienia stanu niskiego na wejściu

B. ANALOG_SENSOR

Obiekt umożliwia odczyt natężenia oświetlenia mierzonego w luksach.

CECHY

Nazwa	Opis
<code>Value</code>	Aktualna wartość sensora
<code>MinValue</code>	Wartość, poniżej której generowane jest zdarzenie <code>onOutOfRange</code>
<code>MaxValue</code>	Wartość, powyżej której generowane jest zdarzenie <code>onOutOfRange</code>

METODY

Nazwa	Opis
<code>SetMinValue</code>	Ustawia dolną wartość progową zdarzenia <code>onOutOfRange</code>
<code>SetMaxValue</code>	Ustawia górną wartość progową zdarzenia <code>onOutOfRange</code>

ZDARZENIA

Nazwa	Opis
<code>OnChange</code>	Zdarzenie wywoływane przy zmianie wartości sensora
<code>OnValueRaise</code>	Zdarzenie wywoływane przy zmianie wartości sensora na wyższą niż poprzednia
<code>OnValueDrop</code>	Zdarzenie wywoływane przy zmianie wartości sensora na niższą niż poprzednia
<code>OnOutOfRange</code>	Zdarzenie wywoływane przy przekroczeniu jednej z wartości progowych <code>MinValue</code> / <code>MaxValue</code>
<code>OnInRange</code>	Zdarzenie wywoływane przy powrocie wartości do przedziału wewnątrz wartości progowych (<code>MinValue</code> : <code>MaxValue</code>)

C. ZWAVE_BATTERY

Obiekt umożliwia odczyt stanu baterii. Odczyt odbywa się cyklicznie, co ustawiony czas, dla cechy `Interval` obiektu `ZWAVE_WAKEUP` (domyślnie 3600s).

CECHY

Nazwa	Opis
BatteryLevel	Poziom baterii modułu Z-Wave (w procentach)
warningLevel	Poziom baterii, poniżej którego generowane są zdarzenia ostrzegawcze

METODY

Nazwa	Opis
setWarningLevel	Ustawia poziom ostrzegawczy baterii modułu Z-Wave

ZDARZENIA

Nazwa	Opis
OnChange	Zdarzenie wywoływane przy zmianie wartości poziomu baterii
OnLowBattery	Zdarzenie wywoływane po wykryciu spadku poziomu baterii poniżej poziomu ostrzegawczego
OnBatteryGood	Zdarzenie wywoływane po wykryciu powrotu poziomu baterii do wartości powyżej poziomu ostrzegawczego

D. ZWAVE_WAKEUP

Obiekt umożliwia ustawienie oraz odczyt czasu wybudzenia baterijnego modułu Z-Wave. Domyślna wartość ustawiana przez CLU wynosi 3600s (60 minut). Minimalna wartość to 300s (5 minut); maksymalna 16777200s (około 194 dni). Możliwe jest ustawienie wartości w kroku 60s (360s, 420s, 480s itd.)

CECHY

Nazwa	Opis
Interval	Czas samoczynnego wybudzania modułu Z-Wave z trybu uśpienia (w sekundach)
LastWakeUp	Czas ostatniego wybudzenia modułu Z-Wave z trybu uśpienia

METODY

Nazwa	Opis
setInterval	Ustawia czas samoczynnego wybudzenia modułu Z-Wave z trybu uśpienia

ZDARZENIA

Nazwa	Opis
onWakeup	Zdarzenie wywoływane po wykryciu wybudzenia modułu Z-Wave z trybu uśpienia

E. ZWAVE_CONFIG

Obiekt wyświetla informacje odnośnie parametrów oraz komunikacji z modułem w sieci Z-Wave. Umożliwia ustawienie zaawansowanych parametrów konfiguracyjnych danego modułu (określone indywidualnie w instrukcji).

CECHY

Nazwa	Opis
NodeID	Numer modułu (węzła) w sieci Z-Wave (nadawany dla każdego modułu Z-Wave po dodaniu go do kontrolera)
Banned	<p>Informacja o zablokowaniu komunikacji Z-Wave z modułem:</p> <ul style="list-style-type: none"> 0 – komunikacja z modułem nie jest zablokowana, 1 – zablokowana komunikacja z modułem (moduł zbanowany). <p>Zablokowanie następuje w momencie, gdy 3 kolejne próby komunikacji z modułem zakończą się niepowodzeniem (inkrementacja cechy <code>FailCount</code> o 3). Do zbanowanego modułu, co 1,5 minuty wysyłane jest zapytanie – jeśli CLU dostanie odpowiedź, wówczas zablokowanie zostanie usunięte i możliwa jest ponowna próba wysłania rozkazu do modułu</p>
FailCount	Liczba nieudanych prób komunikacji z modułem Z-Wave. W przypadku niepowodzenia komunikacji z modułem (brak odpowiedzi, potwierdzenia itp.) następuje inkrementacja cechy o 1, następnie próba ponawiania jest dwukrotnie (w interwałach 15s). W przypadku niepowodzenia komunikacja z modułem zostaje zablokowana (<code>Banned</code> =1)
Register	Numer rejestru (parametru) konfiguracyjnego, który ostatnio został odczytany/ustawiony za pomocą dostępnych metod
value	<p>Wartość rejestru (parametru) konfiguracyjnego</p> <p>UWAGA! Parametr 2, 3, 5 oraz 8 odnosi się do asocjacji modułów, która nie jest wspierana przez system Grenton!</p> <p>UWAGA! Parametr 3 – zmiana wartości parametru nie powoduje wysłania jej podczas wykrycia ruchu!</p> <p>UWAGA! Parametr 4 – poprawne ustawianie wartości parametru, jednakże sam moduł nie przestawia trybu pracy!</p> <p>UWAGA! Parametr 7 oraz 9 – poprawne ustawianie wartości parametru, jednakże ustawiona wartość nie została przetestowana ze względu na uszkodzony czujnik!</p> <p>UWAGA! Parametr 1, 6 – brak zauważalnych zmian w pracy modułu po zmianie wartości!</p> <p>UWAGA! Parametr 9 – mniejszy zakres ustawianych wartości (do 100 luxów)!</p> <p>UWAGA! W dokumentacji brak informacji o rejestrze numer 11 (Motion Event Report One Time Enable)!</p>

METODY

Nazwa	Opis
<code>RemoveBan</code>	Zdejmuje blokadę komunikacji z modułem Z-Wave (w przypadku, gdy cecha <code>Banned</code> =1). Wywołanie metody umożliwia ponowne wysłanie rozkazu do modułu. UWAGA! <code>RemoveBan</code> nie jest jednoznaczne z ponowną poprawną komunikacją z modułem – umożliwia ponowne wysłanie rozkazu/zapytania do modułu! W przypadku niepowodzenia cały proces blokowania jest rozpoczynany na nowo!
<code>ClearFailCount</code>	Czyści liczbę nieudanych prób komunikacji
<code>Set</code>	Ustawia wartość danego rejestru (parametru) konfiguracyjnego: <code>Register</code> (numer rejestru lub parametru), <code>Value</code> (wartość rejestru bądź parametru), <code>Size</code> (rozmiar wysyłanej wartości rejestru lub parametru – w bajtach) UWAGA! Wywołanie metody <code>Set</code> musi być wykonane po wybudzeniu modułu baterijnego! W celu wybudzenia modułu należy trzykrotnie kliknąć przycisk w module - po wybudzeniu zamruga czerwona dioda!
<code>Get</code>	Pobiera wartość danego rejestru (parametru) konfiguracyjnego UWAGA! Wywołanie metody <code>Get</code> musi być wykonane po wybudzeniu modułu baterijnego! W celu wybudzenia modułu należy trzykrotnie kliknąć przycisk w module - po wybudzeniu zamruga czerwona dioda!
<code>setDefault</code>	Ustawia wartość domyślną dla danego rejestru (parametru) konfiguracyjnego UWAGA! Wywołanie metody <code>setDefault</code> musi być wykonane po wybudzeniu modułu baterijnego! W celu wybudzenia modułu należy trzykrotnie kliknąć przycisk w module - po wybudzeniu zamruga czerwona dioda!

ZDARZENIA

Nazwa	Opis
<code>OnBanned</code>	Zdarzenie wywoływane, gdy urządzenie zostanie zbanowane

4. NEO Coolcam Door / Window Sensor

Wersja modułu: NAS-DS01Z

4.1. Informacje ogólne

Moduł Z-Wave Neo Coolcam Door/Window Sensor umożliwia odczyt: stanu kontaktronu (NC) oraz poziomu baterii. Ponadto daje możliwość ustawienia/odczytu czasu wybudzenia modułu.

UWAGA! Dodawanie/usuwanie odbywa się przez trzykrotne kliknięcie przycisku w module Neo podczas inclusion/exclusion. Poprawnie przeprowadzony proces zostanie potwierdzony pięciokrotnym mrugnięciem diody.

4.2. Obiekty

A. BINARY_SENSOR

Obiekt umożliwia odczyt stanu otwarcia/zamknięcia kontaktronu.

CECHY

Nazwa	Opis
<code>value</code>	Zwraca stan wejścia: 0 - zamknięcie, 1 - otwarcie

ZDARZENIA

Nazwa	Opis
<code>OnChange</code>	Zdarzenie wywoływane w przypadku zmiany stanu na przeciwny
<code>onSwitchOn</code>	Zdarzenie wywoływane w momencie ustawienia stanu wysokiego na wejściu
<code>onSwitchOff</code>	Zdarzenie wywoływane w momencie ustawienia stanu niskiego na wejściu

B. ZWAVE_BATTERY

Obiekt umożliwia odczyt stanu baterii. Odczyt odbywa się cyklicznie, co ustawiony czas, dla cechy `Interval` obiektu `ZWAVE_WAKEUP`.

CECHY

Nazwa	Opis
<code>BatteryLevel</code>	Poziom baterii modułu Z-Wave (w procentach)
<code>warningLevel</code>	Poziom baterii, poniżej którego generowane są zdarzenia ostrzegawcze

METODY

Nazwa	Opis
<code>setWarningLevel</code>	Ustawia poziom ostrzegawczy baterii modułu Z-Wave

ZDARZENIA

Nazwa	Opis
<code>OnChange</code>	Zdarzenie wywoływane przy zmianie wartości poziomu baterii
<code>onLowBattery</code>	Zdarzenie wywoływane po wykryciu spadku poziomu baterii poniżej poziomu ostrzegawczego
<code>onBatteryGood</code>	Zdarzenie wywoływane po wykryciu powrotu poziomu baterii do wartości powyżej poziomu ostrzegawczego

C. ZWAVE_WAKEUP

Obiekt umożliwia ustawienie oraz odczyt czasu wybudzenia baterijnego modułu Z-Wave. Domyślna wartość ustawiana przez CLU wynosi 3600s (60 minut). Minimalna wartość to 300s (5 minut); maksymalna 16777200s (około 194 dni). Możliwe jest ustawienie wartości w kroku 60s (360s, 420s, 480s itd.)

CECHY

Nazwa	Opis
<code>Interval</code>	Okres samoczynnego wybudzania modułu Z-Wave z trybu uśpienia (w sekundach)
<code>LastwakeUp</code>	Czas ostatniego wybudzenia modułu Z-Wave z trybu uśpienia

METODY

Nazwa	Opis
<code>SetInterval</code>	Ustawia okres samoczynnego wybudzenia modułu Z-Wave z trybu uśpienia

ZDARZENIA

Nazwa	Opis
<code>OnwakeUp</code>	Zdarzenie wywoływane po wykryciu wybudzenia modułu Z-Wave z trybu uśpienia

D. ZWAVE_CONFIG

Obiekt wyświetla informacje odnośnie parametrów oraz komunikacji z modułem w sieci Z-Wave. Umożliwia ustawienie zaawansowanych parametrów konfiguracyjnych danego modułu (określone indywidualnie w instrukcji).

CECHY

Nazwa	Opis
NodeID	Numer modułu (węzła) w sieci Z-Wave (nadawany dla każdego modułu Z-Wave po dodaniu go do kontrolera)
Banned	<p>Informacja o zablokowaniu komunikacji Z-Wave z modułem:</p> <p>0 – komunikacja z modułem nie jest zablokowana, 1 – zablokowana komunikacja z modułem (moduł zbanowany).</p> <p>Zablokowanie następuje w momencie, gdy 3 kolejne próby komunikacji z modułem zakończą się niepowodzeniem (inkrementacja cechy FailCount o 3). Do zbanowanego modułu, co 1,5 minuty wysyłane jest zapytanie – jeśli CLU dostanie odpowiedź, wówczas zablokowanie zostanie usunięte i możliwa jest ponowna próba wysłania rozkazu do modułu</p>
FailCount	Liczba nieudanych prób komunikacji z modułem Z-Wave. W przypadku niepowodzenia komunikacji z modułem (brak odpowiedzi, potwierdzenia itp.) następuje inkrementacja cechy o 1, następnie próba ponawiania jest dwukrotnie (w interwałach 15s). W przypadku niepowodzenia komunikacja z modułem zostaje zablokowana (Banned =1)
Register	Numer rejestru (parametru) konfiguracyjnego, który ostatnio został odczytany/ustawiony za pomocą dostępnych metod
Value	<p>Wartość rejestru (parametru) konfiguracyjnego</p> <p>UWAGA! Parametr 1 i 2 odnosi się do asocjacji modułów, która nie jest wspierana przez system Grenton!</p>

METODY

Nazwa	Opis
<code>RemoveBan</code>	Zdejmuje blokadę komunikacji z modułem Z-Wave (w przypadku, gdy cecha <code>Banned</code> =1). Wywołanie metody umożliwi ponowne wysłanie rozkazu do modułu. UWAGA! <code>RemoveBan</code> nie jest jednoznaczne z ponowną poprawną komunikacją z modułem – umożliwia ponowne wysłanie rozkazu/zapytania do modułu! W przypadku niepowodzenia cały proces blokowania jest rozpoczynany na nowo!
<code>ClearFailCount</code>	Czyści liczbę nieudanych prób komunikacji
<code>Set</code>	Ustawia wartość danego rejestru (parametru) konfiguracyjnego: <code>Register</code> (numer rejestru lub parametru), <code>Value</code> (wartość rejestru bądź parametru), <code>Size</code> (rozmiar wysyłanej wartości rejestru lub parametru – w bajtach) UWAGA! Wywołanie metody <code>Set</code> musi być wykonane po wybudzeniu modułu baterijnego! W celu wybudzenia modułu należy trzykrotnie kliknąć przycisk w module - po wybudzeniu zamruga czerwona dioda!
<code>Get</code>	Pobiera wartość danego rejestru (parametru) konfiguracyjnego UWAGA! Wywołanie metody <code>Get</code> musi być wykonane po wybudzeniu modułu baterijnego! W celu wybudzenia modułu należy trzykrotnie kliknąć przycisk w module - po wybudzeniu zamruga czerwona dioda!**
<code>setDefault</code>	Ustawia wartość domyślną dla danego rejestru (parametru) konfiguracyjnego UWAGA! Wywołanie metody <code>setDefault</code> musi być wykonane po wybudzeniu modułu baterijnego! W celu wybudzenia modułu należy trzykrotnie kliknąć przycisk w module - po wybudzeniu zamruga czerwona dioda!**

ZDARZENIA

Nazwa	Opis
<code>OnBanned</code>	Zdarzenie wywoływane, gdy urządzenie zostanie zbanowane

5. INFIBITY Motion Sensor (PIR) [NEO Coolcam]

Wersja modułu: NAS-PD01ZE HW: 66 FW: 3.80

5.1. Informacje ogólne

Moduł Z-Wave Infibity Motion Sensor umożliwia odczyt: stanu czujnika ruchu (PIR), poziomu oświetlenia, temperatury oraz poziomu baterii. Ponadto daje możliwość ustawienia/odczytu czasu wybudzenia modułu.

UWAGA! Dodawanie/usuwanie odbywa się przez trzykrotne kliknięcie przycisku w module Infibity podczas inclusion/exclusion. Poprawnie przeprowadzony proces zostanie potwierdzony pięciokrotnym mrugnięciem diody.

5.2. Obiekty

A. BINARY_SENSOR

Obiekt umożliwia odczyt stanu czujnika ruchu.

CECHY

Nazwa	Opis
<code>value</code>	Zwraca stan wejścia: 0 - brak naruszenia, 1 - naruszenie

ZDARZENIA

Nazwa	Opis
<code>onChange</code>	Zdarzenie wywoływane w przypadku zmiany stanu na przeciwny
<code>onSwitchOn</code>	Zdarzenie wywoływane w momencie ustawienia stanu wysokiego na wejściu
<code>onSwitchOff</code>	Zdarzenie wywoływane w momencie ustawienia stanu niskiego na wejściu

B. ANALOG_SENSOR

Obiekt umożliwia odczyt natężenia oświetlenia mierzonego w luxach (ANALOG_SENSOR1) oraz temperatury (ANALOG_SENSOR2).

CECHY

Nazwa	Opis
<code>value</code>	Aktualna wartość sensora
<code>minValue</code>	Wartość, poniżej której generowane jest zdarzenie <code>onOutOfRange</code>
<code>maxValue</code>	Wartość, powyżej której generowane jest zdarzenie <code>onOutOfRange</code>

METODY

Nazwa	Opis
<code>setMinValue</code>	Ustawia dolną wartość progową zdarzenia <code>onOutOfRange</code>
<code>setMaxValue</code>	Ustawia górną wartość progową zdarzenia <code>onOutOfRange</code>

ZDARZENIA

Nazwa	Opis
<code>OnChange</code>	Zdarzenie wywoływane przy zmianie wartości sensora
<code>OnValueRaise</code>	Zdarzenie wywoływane przy zmianie wartości sensora na wyższą niż poprzednia
<code>OnValueDrop</code>	Zdarzenie wywoływane przy zmianie wartości sensora na niższą niż poprzednia
<code>OnOutOfRange</code>	Zdarzenie wywoływane przy przekroczeniu jednej z wartości progowych <code>MinValue</code> / <code>MaxValue</code>
<code>OnInRange</code>	Zdarzenie wywoływane przy powrocie wartości do przedziału wewnątrz wartości progowych (<code>MinValue</code> : <code>MaxValue</code>)

C. ZWAVE_BATTERY

Obiekt umożliwia odczyt stanu baterii. Odczyt odbywa się cyklicznie, co ustawiony czas, dla cechy `Interval` obiektu `ZWAVE_WAKEUP` (domyślnie 3600s).

CECHY

Nazwa	Opis
<code>BatteryLevel</code>	Poziom baterii modułu Z-Wave (w procentach)
<code>WarningLevel</code>	Poziom baterii, poniżej którego generowane są zdarzenia ostrzegawcze

METODY

Nazwa	Opis
<code>SetWarningLevel</code>	Ustawia poziom ostrzegawczy baterii modułu Z-Wave

ZDARZENIA

Nazwa	Opis
<code>OnChange</code>	Zdarzenie wywoływane przy zmianie wartości poziomu baterii
<code>OnLowBattery</code>	Zdarzenie wywoływane po wykryciu spadku poziomu baterii poniżej poziomu ostrzegawczego
<code>OnBatteryGood</code>	Zdarzenie wywoływane po wykryciu powrotu poziomu baterii do wartości powyżej poziomu ostrzegawczego

D. ZWAVE_WAKEUP

Obiekt umożliwia ustawienie oraz odczyt czasu wybudzenia baterijnego modułu Z-Wave. Domyślna wartość ustawiana przez CLU wynosi 3600s (60 minut). Minimalna wartość to 300s (5 minut); maksymalna 16777200s (około 194 dni). Możliwe jest ustawienie wartości w kroku 60s (360s, 420s, 480s itd.)

CECHY

Nazwa	Opis
<code>Interval</code>	Okres samoczynnego wybudzania modułu Z-Wave z trybu uśpienia (w sekundach)
<code>Lastwakeup</code>	Czas ostatniego wybudzenia modułu Z-Wave z trybu uśpienia

METODY

Nazwa	Opis
<code>SetInterval</code>	Ustawia okres samoczynnego wybudzenia modułu Z-Wave z trybu uśpienia

ZDARZENIA

Nazwa	Opis
<code>Onwakeup</code>	Zdarzenie wywoływane po wykryciu wybudzenia modułu Z-Wave z trybu uśpienia

E. ZWAVE_CONFIG

Obiekt wyświetla informacje odnośnie parametrów oraz komunikacji z modułem w sieci Z-Wave. Umożliwia ustawienie zaawansowanych parametrów konfiguracyjnych danego modułu (określone indywidualnie w instrukcji).

CECHY

Nazwa	Opis
NodeID	Numer modułu (węzła) w sieci Z-Wave (nadawany dla każdego modułu Z-Wave po dodaniu go do kontrolera)
Banned	<p>Informacja o zablokowaniu komunikacji Z-Wave z modułem:</p> <p>0 – komunikacja z modułem nie jest zablokowana, 1 – zablokowana komunikacja z modułem (moduł zbanowany).</p> <p>Zablokowanie następuje w momencie, gdy 3 kolejne próby komunikacji z modułem zakończą się niepowodzeniem (inkrementacja cechy FailCount o 3). Do zbanowanego modułu, co 1,5 minuty wysyłane jest zapytanie – jeśli CLU dostanie odpowiedź, wówczas zablokowanie zostanie usunięte i możliwa jest ponowna próba wysłania rozkazu do modułu</p>
FailCount	Liczba nieudanych prób komunikacji z modułem Z-Wave. W przypadku niepowodzenia komunikacji z modułem (brak odpowiedzi, potwierdzenia itp.) następuje inkrementacja cechy o 1, następnie próba ponawiania jest dwukrotnie (w interwałach 15s). W przypadku niepowodzenia komunikacja z modułem zostaje zablokowana (Banned =1)
Register	Numer rejestru (parametru) konfiguracyjnego, który ostatnio został odczytany/ustawiony za pomocą dostępnych metod
Value	<p>Wartość rejestru (parametru) konfiguracyjnego</p> <p>UWAGA! Parametr 2, 3, 5 oraz 8 odnosi się do asocjacji modułów, która nie jest wspierana przez system Grenton!</p> <p>UWAGA! Parametr 1, 6 oraz 7 – brak zauważalnych zmian w pracy modułu po zmianie wartości!</p> <p>UWAGA! Parametr 9 – mniejszy zakres ustawianych wartości (do 100 luksów)!</p>

METODY

Nazwa	Opis
<code>RemoveBan</code>	Zdejmuje blokadę komunikacji z modułem Z-Wave (w przypadku, gdy cecha <code>Banned</code> =1). Wywołanie metody umożliwi ponowne wysłanie rozkazu do modułu. UWAGA! <code>RemoveBan</code> nie jest jednoznaczne z ponowną poprawną komunikacją z modułem – umożliwia ponowne wysłanie rozkazu/zapytania do modułu! W przypadku niepowodzenia cały proces blokowania jest rozpoczynany na nowo!
<code>ClearFailCount</code>	Czyści liczbę nieudanych prób komunikacji
<code>Set</code>	Ustawia wartość danego rejestru (parametru) konfiguracyjnego: <code>Register</code> (numer rejestru lub parametru), <code>Value</code> (wartość rejestru bądź parametru), <code>Size</code> (rozmiar wysyłanej wartości rejestru lub parametru – w bajtach) UWAGA! Wywołanie metody <code>Set</code> musi być wykonane po wybudzeniu modułu baterijnego! W celu wybudzenia modułu należy trzykrotnie kliknąć przycisk w module - po wybudzeniu zamruga czerwona dioda!
<code>Get</code>	Pobiera wartość danego rejestru (parametru) konfiguracyjnego UWAGA! Wywołanie metody <code>Get</code> musi być wykonane po wybudzeniu modułu baterijnego! W celu wybudzenia modułu należy trzykrotnie kliknąć przycisk w module - po wybudzeniu zamruga czerwona dioda!
<code>setDefault</code>	Ustawia wartość domyślną dla danego rejestru (parametru) konfiguracyjnego UWAGA! Wywołanie metody <code>setDefault</code> musi być wykonane po wybudzeniu modułu baterijnego! W celu wybudzenia modułu należy trzykrotnie kliknąć przycisk w module - po wybudzeniu zamruga czerwona dioda!

ZDARZENIA

Nazwa	Opis
<code>OnBanned</code>	Zdarzenie wywoływane, gdy urządzenie zostanie zbanowane

6. INFIBITY Door/Window Sensor [NEO Coolcam]

Wersja modułu: NAS-DS01Z HW: 65 FW: 3.61

6.1. Informacje ogólne

Moduł Z-Wave Infibity Door/Window Sensor umożliwia odczyt: stanu kontaktronu (NC) oraz poziomu baterii. Ponadto daje możliwość ustawienia/odczytu czasu wybudzenia modułu.

UWAGA! Dodawanie/usuwanie odbywa się przez trzykrotne kliknięcie przycisku w module Infibity podczas inclusion/exclusion. Poprawnie przeprowadzony proces zostanie potwierdzony pięciokrotnym mrugnięciem diody.

6.2. Obiekty

A. BINARY_SENSOR

Obiekt umożliwia odczyt stanu otwarcia/zamknięcia kontaktronu.

CECHY

Nazwa	Opis
<code>value</code>	Zwraca stan wejścia: <code>0</code> - zamknięcie, <code>1</code> - otwarcie

ZDARZENIA

Nazwa	Opis
<code>onChange</code>	Zdarzenie wywoływane w przypadku zmiany stanu na przeciwny
<code>onSwitchOn</code>	Zdarzenie wywoływane w momencie ustawienia stanu wysokiego na wejściu
<code>onSwitchOff</code>	Zdarzenie wywoływane w momencie ustawienia stanu niskiego na wejściu

B. ZWAVE_BATTERY

Obiekt umożliwia odczyt stanu baterii. Odczyt odbywa się cyklicznie, co ustawiony czas, dla cechy `interval` obiektu ZWAVE_WAKEUP.

CECHY

Nazwa	Opis
<code>batteryLevel</code>	Poziom baterii modułu Z-Wave (w procentach)
<code>warningLevel</code>	Poziom baterii, poniżej którego generowane są zdarzenia ostrzegawcze

METODY

Nazwa	Opis
<code>setWarningLevel</code>	Ustawia poziom ostrzegawczy baterii modułu Z-Wave

ZDARZENIA

Nazwa	Opis
<code>onChange</code>	Zdarzenie wywoływane przy zmianie wartości poziomu baterii
<code>onLowBattery</code>	Zdarzenie wywoływane po wykryciu spadku poziomu baterii poniżej poziomu ostrzegawczego
<code>onBatteryGood</code>	Zdarzenie wywoływane po wykryciu powrotu poziomu baterii do wartości powyżej poziomu ostrzegawczego

C. ZWAVE_WAKEUP

Obiekt umożliwia ustawienie oraz odczyt czasu wybudzenia baterijnego modułu Z-Wave. Domyślna wartość ustawiana przez CLU wynosi 3600s (60 minut). Minimalna wartość to 300s (5 minut); maksymalna 16777200s (około 194 dni). Możliwe jest ustawienie wartości w kroku 60s (360s, 420s, 480s itd.)

CECHY

Nazwa	Opis
<code>Interval</code>	Okres samoczynnego wybudzania modułu Z-Wave z trybu uśpienia (w sekundach)
<code>LastWakeUp</code>	Czas ostatniego wybudzenia modułu Z-Wave z trybu uśpienia

METODY

Nazwa	Opis
<code>setInterval</code>	Ustawia okres samoczynnego wybudzenia modułu Z-Wave z trybu uśpienia

ZDARZENIA

Nazwa	Opis
<code>onwakeup</code>	Zdarzenie wywoływane po wykryciu wybudzenia modułu Z-Wave z trybu uśpienia

D. ZWAVE_CONFIG

Obiekt wyświetla informacje odnośnie parametrów oraz komunikacji z modułem w sieci Z-Wave. Umożliwia ustawienie zaawansowanych parametrów konfiguracyjnych danego modułu (określone indywidualnie w instrukcji).

CECHY

Nazwa	Opis
NodeID	Numer modułu (węzła) w sieci Z-Wave (nadawany dla każdego modułu Z-Wave po dodaniu go do kontrolera)
Banned	<p>Informacja o zablokowaniu komunikacji Z-Wave z modułem:</p> <p>0 – komunikacja z modułem nie jest zablokowana, 1 – zablokowana komunikacja z modułem (moduł zbanowany).</p> <p>Zablokowanie następuje w momencie, gdy 3 kolejne próby komunikacji z modułem zakończą się niepowodzeniem (inkrementacja cechy FailCount o 3). Do zbanowanego modułu, co 1,5 minuty wysyłane jest zapytanie – jeśli CLU dostanie odpowiedź, wówczas zablokowanie zostanie usunięte i możliwa jest ponowna próba wysłania rozkazu do modułu</p>
FailCount	Liczba nieudanych prób komunikacji z modułem Z-Wave. W przypadku niepowodzenia komunikacji z modułem (brak odpowiedzi, potwierdzenia itp.) następuje inkrementacja cechy o 1, następnie próba ponawiania jest dwukrotnie (w interwałach 15s). W przypadku niepowodzenia komunikacja z modułem zostaje zablokowana (Banned =1)
Register	Numer rejestru (parametru) konfiguracyjnego, który ostatnio został odczytany/ustawiony za pomocą dostępnych metod
Value	<p>Wartość rejestru (parametru) konfiguracyjnego</p> <p>UWAGA! Parametr 1 i 2 odnosi się do asocjacji modułów, która nie jest wspierana przez system Grenton!</p>

METODY

Nazwa	Opis
<code>RemoveBan</code>	Zdejmuje blokadę komunikacji z modułem Z-Wave (w przypadku, gdy cecha <code>Banned</code> =1). Wywołanie metody umożliwia ponowne wysłanie rozkazu do modułu. UWAGA! <code>RemoveBan</code> nie jest jednoznaczne z ponowną poprawną komunikacją z modułem – umożliwia ponowne wysłanie rozkazu/zapytania do modułu! W przypadku niepowodzenia cały proces blokowania jest rozpoczynany na nowo!
<code>ClearFailCount</code>	Czyści liczbę nieudanych prób komunikacji
<code>Set</code>	Ustawia wartość danego rejestru (parametru) konfiguracyjnego: <code>Register</code> (numer rejestru lub parametru), <code>Value</code> (wartość rejestru bądź parametru), <code>Size</code> (rozmiar wysyłanej wartości rejestru lub parametru – w bajtach) UWAGA! Wywołanie metody <code>Set</code> musi być wykonane po wybudzeniu modułu baterijnego! W celu wybudzenia modułu należy trzykrotnie kliknąć przycisk w module - po wybudzeniu zamruga czerwona dioda!
<code>Get</code>	Pobiera wartość danego rejestru (parametru) konfiguracyjnego UWAGA! Wywołanie metody <code>Get</code> musi być wykonane po wybudzeniu modułu baterijnego! W celu wybudzenia modułu należy trzykrotnie kliknąć przycisk w module - po wybudzeniu zamruga czerwona dioda!
<code>setDefault</code>	Ustawia wartość domyślną dla danego rejestru (parametru) konfiguracyjnego UWAGA! Wywołanie metody <code>setDefault</code> musi być wykonane po wybudzeniu modułu baterijnego! W celu wybudzenia modułu należy trzykrotnie kliknąć przycisk w module - po wybudzeniu zamruga czerwona dioda!

ZDARZENIA

Nazwa	Opis
<code>OnBanned</code>	Zdarzenie wywoływane, gdy urządzenie zostanie zbanowane

7. INFIBITY Water Sensor [NEO Coolcam]

Wersja modułu: NAS-WS02ZU HW: 32 FW: 2.133

7.1. Informacje ogólne

Moduł Z-Wave Infibity Water Sensor umożliwia odczyt: stanu czujnika zalania oraz poziomu baterii. Ponadto daje możliwość ustawienia/odczytu czasu wybudzenia modułu.

UWAGA! Dodawanie/usuwanie odbywa się przez trzykrotne kliknięcie przycisku w module Infinity podczas inclusion/exclusion. Poprawnie przeprowadzony proces zostanie potwierdzony pięciokrotnym mrugnięciem diody.

UWAGA! Moduł w Object Managerze zgłasza się jako NEO COOLCAM!

7.2. Obiekty

A. BINARY_SENSOR

Obiekt umożliwia odczyt stanu czujnika zalania.

CECHY

Nazwa	Opis
value	Zwraca stan wejścia: 0 - suchy, 1 - zalany

ZDARZENIA

Nazwa	Opis
onChange	Zdarzenie wywoływane w przypadku zmiany stanu na przeciwny
onSwitchOn	Zdarzenie wywoływane w momencie ustawienia stanu wysokiego na wejściu
onSwitchOff	Zdarzenie wywoływane w momencie ustawienia stanu niskiego na wejściu

B. ZWAVE_BATTERY

Obiekt umożliwia odczyt stanu baterii. Odczyt odbywa się cyklicznie, co ustawiony czas, dla cechy `Interval` obiektu `ZWAVE_WAKEUP`.

CECHY

Nazwa	Opis
BatteryLevel	Poziom baterii modułu Z-Wave (w procentach)
warningLevel	Poziom baterii, poniżej którego generowane są zdarzenia ostrzegawcze

METODY

Nazwa	Opis
SetWarningLevel	Ustawia poziom ostrzegawczy baterii modułu Z-Wave

ZDARZENIA

Nazwa	Opis
<code>OnChange</code>	Zdarzenie wywoływane przy zmianie wartości poziomu baterii
<code>OnLowBattery</code>	Zdarzenie wywoływane po wykryciu spadku poziomu baterii poniżej poziomu ostrzegawczego
<code>OnBatteryGood</code>	Zdarzenie wywoływane po wykryciu powrotu poziomu baterii do wartości powyżej poziomu ostrzegawczego

C. ZWAVE_WAKEUP

Obiekt umożliwia ustawienie oraz odczyt czasu wybudzenia baterijnego modułu Z-Wave. Domyślna wartość ustawiana przez CLU wynosi 3600s (60 minut). Minimalna wartość to 300s (5 minut); maksymalna 16777200s (około 194 dni). Możliwe jest ustawienie wartości w kroku 60s (360s, 420s, 480s itd.)

CECHY

Nazwa	Opis
<code>Interval</code>	Okres samoczynnego wybudzania modułu Z-Wave z trybu uśpienia (w sekundach)
<code>LastWakeup</code>	Czas ostatniego wybudzenia modułu Z-Wave z trybu uśpienia

METODY

Nazwa	Opis
<code>setInterval</code>	Ustawia okres samoczynnego wybudzenia modułu Z-Wave z trybu uśpienia

ZDARZENIA

Nazwa	Opis
<code>onwakeup</code>	Zdarzenie wywoływane po wykryciu wybudzenia modułu Z-Wave z trybu uśpienia

D. ZWAVE_CONFIG

Obiekt wyświetla informacje odnośnie parametrów oraz komunikacji z modułem w sieci Z-Wave. Umożliwia ustawienie zaawansowanych parametrów konfiguracyjnych danego modułu (określone indywidualnie w instrukcji).

CECHY

Nazwa	Opis
NodeID	Numer modułu (węzła) w sieci Z-Wave (nadawany dla każdego modułu Z-Wave po dodaniu go do kontrolera)
Banned	<p>Informacja o zablokowaniu komunikacji Z-Wave z modułem:</p> <p>0 – komunikacja z modułem nie jest zablokowana, 1 – zablokowana komunikacja z modułem (moduł zbanowany).</p> <p>Zablokowanie następuje w momencie, gdy 3 kolejne próby komunikacji z modułem zakończą się niepowodzeniem (inkrementacja cechy FailCount o 3). Do zbanowanego modułu, co 1,5 minuty wysyłane jest zapytanie – jeśli CLU dostanie odpowiedź, wówczas zablokowanie zostanie usunięte i możliwa jest ponowna próba wysłania rozkazu do modułu</p>
FailCount	Liczba nieudanych prób komunikacji z modułem Z-Wave. W przypadku niepowodzenia komunikacji z modułem (brak odpowiedzi, potwierdzenia itp.) następuje inkrementacja cechy o 1, następnie próba ponawiania jest dwukrotnie (w interwałach 15s). W przypadku niepowodzenia komunikacja z modułem zostaje zablokowana (Banned =1)
Register	Numer rejestru (parametru) konfiguracyjnego, który ostatnio został odczytany/ustawiony za pomocą dostępnych metod
Value	<p>Wartość rejestru (parametru) konfiguracyjnego</p> <p>UWAGA! Parametr 7 odnosi się do asocjacji modułów, która nie jest wspierana przez system Grenton!</p>

METODY

Nazwa	Opis
<code>RemoveBan</code>	Zdejmuje blokadę komunikacji z modułem Z-Wave (w przypadku, gdy cecha <code>Banned</code> =1). Wywołanie metody umożliwi ponowne wysłanie rozkazu do modułu. UWAGA! <code>RemoveBan</code> nie jest jednoznaczne z ponowną poprawną komunikacją z modułem – umożliwia ponowne wysłanie rozkazu/zapytania do modułu! W przypadku niepowodzenia cały proces blokowania jest rozpoczynany na nowo!
<code>ClearFailCount</code>	Czyści liczbę nieudanych prób komunikacji
<code>Set</code>	Ustawia wartość danego rejestru (parametru) konfiguracyjnego: <code>Register</code> (numer rejestru lub parametru), <code>Value</code> (wartość rejestru bądź parametru), <code>Size</code> (rozmiar wysyłanej wartości rejestru lub parametru – w bajtach) UWAGA! Wywołanie metody <code>Set</code> musi być wykonane po wybudzeniu modułu baterijnego! W celu wybudzenia modułu należy trzykrotnie kliknąć przycisk w module - po wybudzeniu zamruga czerwona dioda!
<code>Get</code>	Pobiera wartość danego rejestru (parametru) konfiguracyjnego UWAGA! Wywołanie metody <code>Get</code> musi być wykonane po wybudzeniu modułu baterijnego! W celu wybudzenia modułu należy trzykrotnie kliknąć przycisk w module - po wybudzeniu zamruga czerwona dioda!
<code>setDefault</code>	Ustawia wartość domyślną dla danego rejestru (parametru) konfiguracyjnego UWAGA! Wywołanie metody <code>setDefault</code> musi być wykonane po wybudzeniu modułu baterijnego! W celu wybudzenia modułu należy trzykrotnie kliknąć przycisk w module - po wybudzeniu zamruga czerwona dioda!

ZDARZENIA

Nazwa	Opis
<code>OnBanned</code>	Zdarzenie wywoływane, gdy urządzenie zostanie zbanowane

8. Heiman Smart Smoke Sensor

Wersja modułu: **HS1SA-Z (HS1SA-Z HW: 255 FW: 1.10)**

8.1. Informacje ogólne

Moduł Z-Wave Heiman Smart Smoke Sensor umożliwia odczyt: stanu czujnika dymu oraz poziomu baterii. Ponadto daje możliwość ustawienia/odczytu czasu wybudzenia modułu.

UWAGA! Dodawanie/usuwanie odbywa się przez trzykrotne kliknięcie przycisku w module HEIMAN podczas inclusion/exclusion. Poprawnie przeprowadzony proces zostanie potwierdzony pięciokrotnym mrugnięciem diody.

UWAGA! Obsługa modułu dostępna na CLU z firmware 04.07.41 (Build 183201) i nowszym.

8.2. Obiekty

A. BINARY_SENSOR

Obiekt umożliwia odczyt stanu czujnika dymu.

CECHY

Nazwa	Opis
<code>value</code>	Zwraca stan wejścia: 0 - brak naruszenia, 1 - naruszenie (dym)

METODY

-

ZDARZENIA

Nazwa	Opis
<code>onChange</code>	Zdarzenie wywoływane w przypadku zmiany stanu na przeciwny
<code>onSwitchOn</code>	Zdarzenie wywoływane w momencie ustawienia stanu wysokiego na wejściu
<code>onSwitchOff</code>	Zdarzenie wywoływane w momencie ustawienia stanu niskiego na wejściu

B. ZWAVE_BATTERY

Obiekt umożliwia odczyt stanu baterii. Odczyt odbywa się cyklicznie, co ustawiony czas, dla cechy `Interval` obiektu `ZWAVE_WAKEUP`.

CECHY

Nazwa	Opis
<code>BatteryLevel</code>	Poziom baterii modułu Z-Wave (w procentach)
<code>warningLevel</code>	Poziom baterii, poniżej którego generowane są zdarzenia ostrzegawcze

METODY

Nazwa	Opis
<code>SetWarningLevel</code>	Ustawia poziom ostrzegawczy baterii modułu Z-Wave

ZDARZENIA

Nazwa	Opis
<code>OnChange</code>	Zdarzenie wywoływane przy zmianie wartości poziomu baterii
<code>OnLowBattery</code>	Zdarzenie wywoływane po wykryciu spadku poziomu baterii poniżej poziomu ostrzegawczego
<code>OnBatteryGood</code>	Zdarzenie wywoływane po wykryciu powrotu poziomu baterii do wartości powyżej poziomu ostrzegawczego

C. ZWAVE_WAKEUP

Obiekt umożliwia ustawienie oraz odczyt czasu wybudzenia baterijnego modułu Z-Wave. Domyślna wartość ustawiana przez CLU wynosi 3600s (60 minut). Minimalna wartość to 300s (5 minut); maksymalna 16777200s (około 194 dni). Możliwe jest ustawienie wartości w kroku 60s (360s, 420s, 480s itd.)

CECHY

Nazwa	Opis
<code>Interval</code>	Okres samoczynnego wybudzania modułu Z-Wave z trybu uśpienia (w sekundach)
<code>LastWakeup</code>	Czas ostatniego wybudzenia modułu Z-Wave z trybu uśpienia

METODY

Nazwa	Opis
<code>setInterval</code>	Ustawia okres samoczynnego wybudzenia modułu Z-Wave z trybu uśpienia

ZDARZENIA

Nazwa	Opis
<code>onwakeup</code>	Zdarzenie wywoływane po wykryciu wybudzenia modułu Z-Wave z trybu uśpienia

D. ZWAVE_CONFIG

Obiekt wyświetla informacje odnośnie parametrów komunikacji z modułem w sieci Z-Wave.

CECHY

Nazwa	Opis
<code>NodeID</code>	Numer modułu (węzła) w sieci Z-Wave (nadawany dla każdego modułu Z-Wave po dodaniu go do kontrolera)
<code>Banned</code>	Informacja o zablokowaniu komunikacji Z-Wave z modułem: 0 – komunikacja z modułem nie jest zablokowana, 1 – zablokowana komunikacja z modułem (moduł zbanowany). Zablokowanie następuje w momencie, gdy 3 kolejne próby komunikacji z modułem zakończą się niepowodzeniem (inkrementacja cechy <code>FailCount</code> o 3). Do zbanowanego modułu, co 1,5 minuty wysyłane jest zapytanie – jeśli CLU dostanie odpowiedź, wówczas zablokowanie zostanie usunięte i możliwa jest ponowna próba wysłania rozkazu do modułu
<code>FailCount</code>	Liczba nieudanych prób komunikacji z modułem Z-Wave. W przypadku niepowodzenia komunikacji z modułem (brak odpowiedzi, potwierdzenia itp.) następuje inkrementacja cechy o 1, następnie próba ponawiania jest dwukrotnie (w interwałach 15s). W przypadku niepowodzenia komunikacja z modułem zostaje zablokowana (<code>Banned</code> =1)

METODY

Nazwa	Opis
<code>RemoveBan</code>	Zdejmuje blokadę komunikacji z modułem Z-Wave (w przypadku, gdy cecha <code>Banned</code> =1). Wywołanie metody umożliwia ponowne wysłanie rozkazu do modułu. UWAGA! <code>RemoveBan</code> nie jest jednoznaczne z ponowną poprawną komunikacją z modułem – umożliwia ponowne wysłanie rozkazu/zapytania do modułu! W przypadku niepowodzenia cały proces blokowania jest rozpoczynany na nowo!
<code>ClearFailCount</code>	Czyści liczbę nieudanych prób komunikacji

ZDARZENIA

Nazwa	Opis
<code>OnBanned</code>	Zdarzenie wywoływane, gdy urządzenie zostanie zbanowane

9. INFIBITY Siren Alarm [NEO Coolcam]

Wersja modułu: *NAS-AB01Z HW:48 FW: 2.90*

9.1. Informacje ogólne

Obsługa modułu Infibity Siren Alarm obejmuje możliwość załączania / wyłączenia sygnału syreny, odczyt poziomu baterii, a także ustawienie oraz odczyt wybudzenia modułu. Dodatkowo istnieje możliwość zmiany parametrów konfiguracyjnych.

Sposób dodawania / usuwania: W celu dodania / usunięcia modułu należy 3x kliknąć na przycisku w module Infibity podczas inclusion / exclusion – w przypadku poprawnego dodania / usunięcia, czerwone LEDy zamrugają 5 razy.

Uwaga! Po restarcie (wysłaniu konfiguracji) CLU, należy odczekać 10s przed pierwszą próbą załączenia modułu Siren Alarm.

9.2. Obiekty

A. ZWAVE_DOUT

Obiekt umożliwia załączenie / wyłączenie oraz odczyt aktualnego stanu syreny.

CECHY

Nazwa	Opis
<code>value</code>	Zwraca stan wyjścia (0 – wyłączone; 1 – załączone)

METODY

Nazwa	Opis
<code>setValue</code>	Ustawia stan wyjścia jako 1 lub 0
<code>switch</code>	Przełącza wyjście. Parametr Time określa na jak długo następuje zmiana stanu, dla 0 jest ona stała
<code>switchon</code>	Załącza wyjście. Parametr Time określa na jak długo następuje zmiana stanu, dla 0 jest ona stała
<code>switchoff</code>	Wyłącza wyjście. Parametr Time określa na jak długo następuje zmiana stanu, dla 0 jest ona stała

ZDARZENIA

Nazwa	Opis
<code>onChange</code>	Zdarzenie wywoływane w przypadku zmiany stanu na przeciwny
<code>onswitchon</code>	Zdarzenie wywoływane w momencie ustawienia stanu wysokiego na wyjściu
<code>onswitchoff</code>	Zdarzenie wywoływane w momencie ustawienia stanu niskiego na wyjściu

B. ZWAVE_BATTERY

Obiekt umożliwia odczyt stanu baterii. Odczyt stanu odbywa się cyklicznie co ustawiony czas dla cechy Interval obiektu ZWAVE_WAKEUP

CECHY

Nazwa	Opis
BatteryLevel	Poziom baterii modułu Z-Wave w procentach
warningLevel	Poziom baterii, poniżej którego generowane są zdarzenia ostrzegawcze

METODY

Nazwa	Opis
setWarningLevel	Ustawia poziom ostrzegawczy baterii modułu Z-Wave

ZDARZENIA

Nazwa	Opis
OnChange	Zdarzenie wywoływane, gdy urządzenie zostanie zbanowane
OnLowBattery	Zdarzenie wywoływane po wykryciu spadku poziomu baterii poniżej poziomu ostrzegawczego
OnBatteryGood	Zdarzenie wywoływane po wykryciu powrotu poziomu baterii do wartości powyżej poziomu ostrzegawczego

C. ZWAVE_WAKEUP

Obiekt umożliwiający ustawienie oraz odczyt czasu wybudzenia baterijnego modułu Z-Wave. Domyślna wartość ustawiania przez CLU wynosi 3600s (5 minut). Minimalna wartość to 60s (1 minuta); maksymalna 16777200s (około 194 dni).

CECHY

Nazwa	Opis
Interval	Okres samoczynnego wybudzania modułu Z-Wave z trybu uśpienia w sekundach
LastWakeUp	Czas ostatniego wybudzenia modułu Z-Wave z trybu uśpienia

METODY

Nazwa	Opis
setInterval	Ustawia okres samoczynnego wybudzania modułu Z-Wave z trybu uśpienia

ZDARZENIA

Nazwa	Opis
OnWakeUp	Zdarzenie wywoływane po wykryciu wybudzenia modułu Z-Wave z trybu uśpienia

D. ZWAVE_CONFIG

Obiekt wyświetla informacje odnośnie parametrów komunikacji z modułem w sieci Z-Wave. Umożliwia również ustawienie zaawansowanych parametrów konfiguracyjnych danego modułu (określone indywidualnie w instrukcji).

Ustawienie rejestru 7 zmienia tryb pracy syreny:

- Jako **Alarm** – syrena działa zgodnie z ustawieniami parametrów: 1,2,5,8
- Jako **DoorBell** – syrena działa zgodnie z ustawieniami parametrów: 3,4,6,9

CECHY

Nazwa	Opis
NodeID	Numer modułu (węzła) w sieci Z-Wave (nadawany dla każdego modułu Z-Wave po dodaniu go do kontrolera)
Banned	Informacja o zablokowaniu komunikacji Z-Wave z modułem: 0 – komunikacja z modułem nie jest zablokowana, 1 – zablokowana komunikacja z modułem (moduł zbanowany). Zablokowanie następuje w momencie, gdy 3 kolejne próby komunikacji z modułem zakończą się niepowodzeniem (inkrementacja cechy <code>FailCount</code> o 3). Do zbanowanego modułu, co 1,5 minuty wysyłane jest zapytanie – jeśli CLU dostanie odpowiedź, wówczas zablokowanie zostanie usunięte i możliwa jest ponowna próba wysłania rozkazu do modułu
FailCount	Liczba nieudanych prób komunikacji z modułem Z-Wave. W przypadku niepowodzenia komunikacji z modułem (brak odpowiedzi, potwierdzenia itp.) następuje inkrementacja cechy o 1, następnie próba ponawiania jest dwukrotnie (w interwałach 15s). W przypadku niepowodzenia komunikacja z modułem zostaje zablokowana (<code>Banned</code> =1). UWAGA! Po restarcie CLU, do modułu wysyłana jest komenda Switch Binary Switch na którą moduł nie odpowiada, przez co <code>FailCount</code> zostaje zwiększony o 1.
Register	Numer rejestru (parametru) konfiguracyjnego, który ostatnio został odczytany/ustawiony za pomocą dostępnych metod
value	Wartość rejestru (parametru) konfiguracyjnego

METODY

Nazwa	Opis
RemoveBan	Zdejmuje blokadę komunikacji z modułem Z-Wave (w przypadku, gdy cecha Banned =1). Wywołanie metody umożliwia ponowne wysłanie rozkazu do modułu. UWAGA! RemoveBan nie jest jednoznaczne z ponowną poprawną komunikacją z modułem – umożliwia ponowne wysłanie rozkazu/zapytania do modułu! W przypadku niepowodzenia cały proces blokowania jest rozpoczynany na nowo!
ClearFailCount	Czyści liczbę nieudanych prób komunikacji
Set	Ustawia wartość danego rejestru (parametru) konfiguracyjnego: Register (numer rejestru lub parametru), Value (wartość rejestru bądź parametru), Size (rozmiar wysyłanej wartości rejestru lub parametru – w bajtach) UWAGA! Wywołanie metody set musi być wykonane po wybudzeniu modułu baterijnego! W celu wybudzenia modułu należy trzykrotnie kliknąć przycisk w module - po wybudzeniu zamruga czerwona dioda!
Get	Pobiera wartość danego rejestru (parametru) konfiguracyjnego
SetDefault	Ustawia wartość domyślną dla danego rejestru (parametru) konfiguracyjnego

ZDARZENIA

Nazwa	Opis
OnBanned	Zdarzenie wywoływane, gdy urządzenie zostanie zbanowane

10. Danfoss Living Connect

Wersja modułu: EU HW: 00 FW: 1.1

10.1. Informacje ogólne

Obsługa modułu Danfoss Living Connect obejmuje możliwość ustawiania temperatury zadanej na głowicy, a także załączania / wyłączenia blokady przycisków. Możliwy jest również odczyt poziomu naładowania baterii urządzenia oraz definiowanie okresu wybudzania modułu.

Sposób dodawania / usuwania: Aby dodać / usunąć urządzenie należy 1x kliknąć środkowy przycisk na module podczas inclusion / exclusion (wywołane na CLU) – podświetlenie wyświetlacza zacznie szybko mrugać, a następnie zaświeci światłem ciągłym. Jeśli po dłuższym czasie szybkiego mrugania podświetlenie wyświetlacza zacznie mrugać wolniej, to oznacza, że proces dodawania się nie udał. Przed dodaniem urządzenia należy wyjść z trybu montażu sygnalizowanego na wyświetlaczu symbolem „M”.

10.2. Obiekty

A. ZWAVE_THERMOSTAT

Obiekt umożliwiający ustawianie temperatury zadanej na głowicy a także załączania / wyłączenia blokady przycisków.

UWAGA! Obsługa nie obejmuje odczytu ustawionej temperatury za pomocą przycisków na głowicy.

CECHY

Nazwa	Opis
<code>PointValue</code>	Zwraca wartość zadanej temperatury (4°C ÷ 28°C)
<code>ProtectionState</code>	Zwraca stan blokady klawiszy: 0 – wyłączona, 2 – włączona

METODY

Nazwa	Opis
<code>SetPointValue</code>	Ustawia wartość zadanej temperatury (cecha PointValue)
<code>SetProtectionState</code>	Ustawia stan blokady klawiszy

ZDARZENIA

Nazwa	Opis
<code>OnPointValueChange</code>	Zdarzenie wywoływane w przypadku zmiany wartości zadanej temperatury
<code>OnProtectionChange</code>	Zdarzenie wywoływane w momencie zmiany stanu blokady klawiszy
<code>OnProtectionOn</code>	Zdarzenie wywoływane w momencie włączenia blokady klawiszy
<code>OnProtectionOff</code>	Zdarzenie wywoływane w momencie wyłączenia blokady klawiszy

B. ZWAVE_BATTERY

Obiekt umożliwia odczyt stanu baterii. Odczyt stanu odbywa się cyklicznie co ustawiony czas dla cechy Interval obiektu ZWAVE_WAKEUP

CECHY

Nazwa	Opis
<code>BatteryLevel</code>	Poziom baterii modułu Z-Wave w procentach
<code>warningLevel</code>	Poziom baterii, poniżej którego generowane są zdarzenia ostrzegawcze

METODY

Nazwa	Opis
<code>SetwarningLevel</code>	Ustawia poziom ostrzegawczy baterii modułu Z-Wave

ZDARZENIA

Nazwa	Opis
<code>OnChange</code>	Zdarzenie wywoływane, gdy urządzenie zostanie zbanowane
<code>OnLowBattery</code>	Zdarzenie wywoływane po wykryciu spadku poziomu baterii poniżej poziomu ostrzegawczego
<code>OnBatteryGood</code>	Zdarzenie wywoływane po wykryciu powrotu poziomu baterii do wartości powyżej poziomu ostrzegawczego

C. ZWAVE_WAKEUP

Obiekt umożliwiający ustawienie oraz odczyt czasu wybudzenia baterijnego modułu Z-Wave. Domyślna wartość ustawiania przez CLU wynosi 300s (5 minut). Minimalna wartość to 60s (1 minuta); maksymalna 1800s (30 minut). Możliwe jest ustawienie wartości w kroku 60s (60s, 120s, 180s itd.)

CECHY

Nazwa	Opis
<code>Interval</code>	Okres samoczynnego wybudzania modułu Z-Wave z trybu uśpienia w sekundach
<code>LastWakeUp</code>	Czas ostatniego wybudzenia modułu Z-Wave z trybu uśpienia

METODY

Nazwa	Opis
<code>SetInterval</code>	Ustawia okres samoczynnego wybudzania modułu Z-Wave z trybu uśpienia

ZDARZENIA

Nazwa	Opis
<code>OnWakeUp</code>	Zdarzenie wywoływane po wykryciu wybudzenia modułu Z-Wave z trybu uśpienia

D. ZWAVE_CONFIG

Obiekt wyświetla informacje odnośnie parametrów komunikacji z modułem w sieci Z-Wave.

CECHY

Nazwa	Opis
NodeID	Numer modułu (węzła) w sieci Z-Wave (nadawany dla każdego modułu Z-Wave po dodaniu go do kontrolera)
Banned	Informacja o zablokowaniu komunikacji Z-Wave z modułem: 0 – komunikacja z modułem nie jest zablokowana, 1 – zablokowana komunikacja z modułem (moduł zbanowany). Zablokowanie następuje w momencie, gdy 3 kolejne próby komunikacji z modułem zakończą się niepowodzeniem (inkrementacja cechy FailCount o 3). Do zbanowanego modułu, co 1,5 minuty wysyłane jest zapytanie – jeśli CLU dostanie odpowiedź, wówczas zablokowanie zostanie usunięte i możliwa jest ponowna próba wysłania rozkazu do modułu
FailCount	Liczba nieudanych prób komunikacji z modułem Z-Wave. W przypadku niepowodzenia komunikacji z modułem (brak odpowiedzi, potwierdzenia itp.) następuje inkrementacja cechy o 1, następnie próba ponawiania jest dwukrotnie (w interwałach 15s). W przypadku niepowodzenia komunikacja z modułem zostaje zablokowana (Banned =1)

METODY

Nazwa	Opis
RemoveBan	Zdejmuje blokadę komunikacji z modułem Z-Wave (w przypadku, gdy cecha Banned =1). Wywołanie metody umożliwi ponowne wysłanie rozkazu do modułu. UWAGA! <i>RemoveBan nie jest jednoznaczne z ponowną poprawną komunikacją z modułem – umożliwi ponowne wysłanie rozkazu/zapytania do modułu! W przypadku niepowodzenia cały proces blokowania jest rozpoczynany na nowo!</i>
ClearFailCount	Czyści liczbę nieudanych prób komunikacji

ZDARZENIA

Nazwa	Opis
OnBanned	Zdarzenie wywoływane, gdy urządzenie zostanie zbanowane

11. POPP Z-Weather

Wersja modułu: EU HW: 01 FW: 1.0

11.1. Informacje ogólne

Obsługa modułu POPP Z-Weather obejmuje możliwość odczytu parametrów klimatycznych ze stacji pogodowej. Możliwy jest również odczyt poziomu naładowania baterii urządzenia, a także definiowanie okresu wybudzania modułu.

Sposób dodawania / usuwania: Aby dodać / usunąć urządzenie należy 3x kliknąć przycisk na module w czasie 1,5s podczas inclusion / exclusion (wywołane na CLU) – czerwona dioda na module zamruga 3x podczas dodawania lub 1x podczas usuwania.

Sposób wybudzenia urządzenia: 1x kliknąć na urządzeniu

11.2. Obiekty

A. ZWAVE_WEATHER

Obiekt umożliwiający odczyt parametrów klimatycznych - temperatury, luminancji, wilgotności względnej, prędkości wiatru, ciśnienia barometrycznego oraz temperatury punktu rosy.

CECHY

Nazwa	Opis
Temperature	Zwraca wartość zmierzonej temperatury powietrza (-10°C ÷ 60°C)
Luminance	Zwraca wartość zmierzonej luminancji (0% ÷ 100%)
Humidity	Zwraca wartość zmierzonej wilgotności względnej (0% ÷ 100%)
windSpeed	Zwraca wartość zmierzonej prędkości wiatru (0m/s ÷ 31m/s)
Pressure	Zwraca wartość zmierzonego ciśnienia barometrycznego (600hPa ÷ 1200hPa)
DewPoint	Zwraca wartość zmierzonej temperatury punktu rosy (-56,4°C ÷ 60°C)

ZDARZENIA

Nazwa	Opis
OnTemperatureChange	Zdarzenie wywoływane w przypadku zmiany wartości temperatury powietrza
OnLuminanceChange	Zdarzenie wywoływane w momencie zmiany wartości luminancji
OnHumidityChange	Zdarzenie wywoływane w momencie zmiany wartości wilgotności względnej
OnWindSpeedChange	Zdarzenie wywoływane w momencie zmiany wartości prędkości wiatru
OnPressureChange	Zdarzenie wywoływane w momencie zmiany wartości ciśnienia barometrycznego
OnDewPointChange	Zdarzenie wywoływane w momencie zmiany wartości temperatury punktu rosy

B. ZWAVE_BATTERY

Obiekt umożliwia odczyt stanu baterii. Odczyt stanu odbywa się cyklicznie co ustawiony czas dla cechy Interval obiektu ZWAVE_WAKEUP

CECHY

Nazwa	Opis
BatteryLevel	Poziom baterii modułu Z-Wave w procentach
warningLevel	Poziom baterii, poniżej którego generowane są zdarzenia ostrzegawcze

METODY

Nazwa	Opis
setWarningLevel	Ustawia poziom ostrzegawczy baterii modułu Z-Wave

ZDARZENIA

Nazwa	Opis
OnChange	Zdarzenie wywoływane, gdy urządzenie zostanie zbanowane
OnLowBattery	Zdarzenie wywoływane po wykryciu spadku poziomu baterii poniżej poziomu ostrzegawczego
OnBatteryGood	Zdarzenie wywoływane po wykryciu powrotu poziomu baterii do wartości powyżej poziomu ostrzegawczego

C. ZWAVE_WAKEUP

Obiekt umożliwiający ustawienie oraz odczyt czasu wybudzenia baterijnego modułu Z-Wave. Domyślna wartość ustawiania przez CLU wynosi 600s (około 10 minut). Minimalna wartość to 600s (około 10 minut), maksymalna 17180s (około 286 minut). Możliwe jest ustawienie wartości w kroku 1s (600s, 601s, 602s itd.)

CECHY

Nazwa	Opis
Interval	Okres samoczynnego wybudzania modułu Z-Wave z trybu uśpienia w sekundach
LastWakeUp	Czas ostatniego wybudzenia modułu Z-Wave z trybu uśpienia

METODY

Nazwa	Opis
setInterval	Ustawia okres samoczynnego wybudzania modułu Z-Wave z trybu uśpienia

ZDARZENIA

Nazwa	Opis
onWakeUp	Zdarzenie wywoływane po wykryciu wybudzenia modułu Z-Wave z trybu uśpienia

D. ZWAVE_CONFIG

Obiekt wyświetla informacje odnośnie parametrów komunikacji z modułem w sieci Z-Wave.

CECHY

Nazwa	Opis
NodeID	Numer modułu (węzła) w sieci Z-Wave (nadawany dla każdego modułu Z-Wave po dodaniu go do kontrolera)
Banned	Informacja o zablokowaniu komunikacji Z-Wave z modułem: 0 – komunikacja z modułem nie jest zablokowana, 1 – zablokowana komunikacja z modułem (moduł zbanowany). Zablokowanie następuje w momencie, gdy 3 kolejne próby komunikacji z modułem zakończą się niepowodzeniem (inkrementacja cechy FailCount o 3). Do zbanowanego modułu, co 1,5 minuty wysyłane jest zapytanie – jeśli CLU dostanie odpowiedź, wówczas zablokowanie zostanie usunięte i możliwa jest ponowna próba wysłania rozkazu do modułu
FailCount	Liczba nieudanych prób komunikacji z modułem Z-Wave. W przypadku niepowodzenia komunikacji z modułem (brak odpowiedzi, potwierdzenia itp.) następuje inkrementacja cechy o 1, następnie próba ponawiania jest dwukrotnie (w interwałach 15s). W przypadku niepowodzenia komunikacja z modułem zostaje zablokowana (Banned =1)

METODY

Nazwa	Opis
RemoveBan	Zdejmuje blokadę komunikacji z modułem Z-Wave (w przypadku, gdy cecha Banned =1). Wywołanie metody umożliwi ponowne wysłanie rozkazu do modułu. UWAGA! <i>RemoveBan nie jest jednoznaczne z ponowną poprawną komunikacją z modułem – umożliwia ponowne wysłanie rozkazu/zapytania do modułu! W przypadku niepowodzenia cały proces blokowania jest rozpoczynany na nowo!</i>
ClearFailCount	Czyści liczbę nieudanych prób komunikacji

ZDARZENIA

Nazwa	Opis
OnBanned	Zdarzenie wywoływane, gdy urządzenie zostanie zbanowane

12. FAKRO AMZ Solar

Wersja modułu: HW: 31 FW: 1.01:01.01

12.1. Informacje ogólne

Obsługa modułu FAKRO AMZ Solar obejmuje możliwość sterowania oknem - zarówno poprzez maksymalne otwarcie / zamknięcie, jak i ustawienie wartości procentowej otwarcia okna, zmianę trybu pracy (również trybu sezonowego), a także definiowanie parametrów działających w danym trybie. Ponadto daje możliwość zmiany parametrów konfiguracyjnych (interfejs konfiguracyjny Fakro).

Sposób dodawania / usuwania: Dodawanie / usuwanie urządzenia odbywa się poprzez naciśnięcie przycisku 'P' na urządzeniu podczas inclusion / exclusion (wywołane na CLU).

12.2. Obiekty

ZWAVE_FAKRO

Obiekt umożliwiający sterowanie otwarciem markizy i odczyt ustawionego procentu otwarcia. Dostępne jest ustawianie na wartość maksymalną (otwarcie / zamknięcie), a także podając procentową wartość otwarcia markizy (0-100%). Dodatkowo możliwe jest ustawianie trybów pracy urządzenia oraz parametrów dotyczących poszczególnych trybów pracy.

UWAGA! Informacje dotyczące poszczególnych trybów pracy znajdują się w dokumentacji urządzenia dostarczanej przez producenta.

CECHY

Nazwa	Opis
State	Stan urządzenia: 0 - stoi, 1 - ruch w górę, 2 - ruch w dół
Percent	Wartość procentowa otwarcia markizy, gdzie: 0% - okno zamknięte, 100% - okno otwarte UWAGA! Wartość cechy <code>Percent</code> jest odświeżana w momencie, gdy sterownik markizy zakończy daną pracę - należy mieć to na uwadze przy wykorzystywaniu tej cechy np. dla komponentu Slider.
Mode	Tryb pracy urządzenia: 0 - Manual - Ręczny, 1 - Semiauto - Półautomatyczny, 2 - Auto - Automatyczny
SeasonMode	Tryb sezonowy urządzenia: 0 - Summer - Lato, 1 - winter - Zima Uwaga! Parametr nie dotyczy trybu ręcznego <code>Mode = 0</code>
OpeningTime	Czas otwarcia markizy w trybie półautomatycznym
Sensitivity	Czułość poziomu nasłonecznienia dla markizy w trybie automatycznym

Uwaga! Wartość ustawionych parametrów konfiguracyjnych jest odświeżana w momencie `wakeUp`'u danego urządzenia (pobierane są wartości od urządzenia Z-Wave).

METODY

Nazwa	Opis
Up	Markiza do góry
Down	Markiza w dół
Stop	Stop, jeśli markiza jest w ruchu
Start	Markiza do góry jeśli poprzednio ruch w dół, markiza w dół jeśli poprzednio ruch w górę
SetPercent	Ustawia wartość procentową, gdzie 100% - markiza otwarta
SetMode	Ustawia tryb pracy urządzenia
SetSeasonMode	Ustawia tryb sezonowy
SetOpeningTime	Ustawia czas otwarcia markizy
SetSensitivity	Ustawia czułość poziomemu nasłonecznieniu

ZDARZENIA

Nazwa	Opis
OnChange	Zdarzenie wywoływane w momencie zmiany stanu sterownika okna
OnUp	Zdarzenie wywoływane w momencie zmiany stanu ze Stop na Up
OnDown	Zdarzenie wywoływane w momencie zmiany stanu ze Stop na Down
OnStart	Zdarzenie wywoływane w momencie wywołania komendy Start
OnStop	Zdarzenie wywoływane w momencie wywołania komendy Stop

ZWAVE_CONFIG

Obiekt wyświetla informacje odnośnie parametrów komunikacji z modułem w sieci Z-Wave. Umożliwia ustawienie zaawansowanych parametrów konfiguracyjnych danego modułu (określone indywidualnie w instrukcji).

CECHY

Nazwa	Opis
<code>NodeID</code>	Numer modułu (węzła) w sieci Z-Wave (nadawany dla każdego modułu Z-Wave po dodaniu go do kontrolera)
<code>Banned</code>	Informacja o zablokowaniu komunikacji Z-Wave z modułem: 0 – komunikacja z modułem nie jest zablokowana, 1 – zablokowana komunikacja z modułem (moduł zbanowany). Zablokowanie następuje w momencie, gdy 3 kolejne próby komunikacji z modułem zakończą się niepowodzeniem (inkrementacja cechy <code>FailCount</code> o 3). Do zbanowanego modułu, co 1,5 minuty wysyłane jest zapytanie – jeśli CLU dostanie odpowiedź, wówczas zablokowanie zostanie usunięte i możliwa jest ponowna próba wysłania rozkazu do modułu
<code>FailCount</code>	Liczba nieudanych prób komunikacji z modułem Z-Wave. W przypadku niepowodzenia komunikacji z modułem (brak odpowiedzi, potwierdzenia itp.) następuje inkrementacja cechy o 1, następnie próba ponawiania jest dwukrotnie (w interwałach 15s). W przypadku niepowodzenia komunikacja z modułem zostaje zablokowana (<code>Banned</code> =1)
<code>Register</code>	Numer rejestru (parametru) konfiguracyjnego, który ostatnio został odczytany/ustawiony za pomocą dostępnych metod
<code>value</code>	Wartość rejestru (parametru) konfiguracyjnego

METODY

Nazwa	Opis
<code>RemoveBan</code>	Zdejmuje blokadę komunikacji z modułem Z-Wave (w przypadku, gdy cecha <code>Banned</code> =1). Wywołanie metody umożliwia ponowne wysłanie rozkazu do modułu. UWAGA! <code>RemoveBan</code> nie jest jednoznaczne z ponowną poprawną komunikacją z modułem – umożliwia ponowne wysłanie rozkazu/zapytania do modułu! W przypadku niepowodzenia cały proces blokowania jest rozpoczynany na nowo!
<code>ClearFailCount</code>	Czyści liczbę nieudanych prób komunikacji
<code>Set</code>	Ustawia wartość danego rejestru (parametru) konfiguracyjnego: <code>Register</code> (numer rejestru lub parametru), <code>value</code> (wartość rejestru bądź parametru), <code>size</code> (rozmiar wysyłanej wartości rejestru lub parametru – w bajtach)
<code>Get</code>	Pobiera wartość danego rejestru (parametru) konfiguracyjnego
<code>SetDefault</code>	Ustawia wartość domyślną dla danego rejestru (parametru) konfiguracyjnego

ZDARZENIA

Nazwa	Opis
OnBanned	Zdarzenie wywoływane, gdy urządzenie zostanie zbanowane

13. FAKRO ARF

13.1. Informacje ogólne

Obsługa modułu FAKRO ARF obejmuje możliwość sterowania roletą - zarówno maksymalne otwarcie / zamknięcie, jak i ustawienie wartości procentowej otwarcia rolety.

Sposób dodawania / usuwania: Dodawanie / usuwanie urządzenia odbywa się poprzez naciśnięcie przycisku 'P' na urządzeniu podczas inclusion / exclusion (wywołane na CLU).

13.2. Obiekty

A. ZWAVE_FAKRO

Obiekt umożliwiający sterowanie roletą i odczyt ustawionego procentu otwarcia. Dostępne jest ustawianie na wartość maksymalną (otwarcie / zamknięcie) a także podając procentową wartość otwarcia rolety (0-100%).

CECHY

Nazwa	Opis
State	Stan rolety: 0 - stoi, 1 - ruch w górę, 2 - ruch w dół
Percent	Wartość procentowa otwarcia rolety, gdzie: 0% - roleta zamknięta, 100% - roleta otwarta UWAGA! Wartość cechy <code>Percent</code> jest odświeżana w momencie, gdy roleta zakończy daną pracę - należy mieć to na uwadze przy wykorzystywaniu tej cech np. dla komponentu Slider. UWAGA! Wywołanie metody <code>Stop</code> podczas jazdy rolety nie powoduje odświeżenia cechy <code>Percent</code>

METODY

Nazwa	Opis
Up	Roleta do góry
Down	Roleta w dół
Stop	Stop, jeśli roleta jest w ruchu
Start	Roleta do góry jeśli poprzednio ruch w dół, roleta w dół jeśli poprzednio ruch w górę
SetPercent	Ustawia wartość procentową, gdzie 100% - roleta otwarta

ZDARZENIA

Nazwa	Opis
OnChange	Zdarzenie wywoływane w momencie zmiany stanu rolety
OnUp	Zdarzenie wywoływane w momencie zmiany stanu ze Stop na Up
OnDown	Zdarzenie wywoływane w momencie zmiany stanu ze Stop na Down
OnStart	Zdarzenie wywoływane w momencie wywołania komendy Start
OnStop	Zdarzenie wywoływane w momencie wywołania komendy Stop

B. ZWAVE_CONFIG

Obiekt wyświetla informacje odnośnie parametrów komunikacji z modułem w sieci Z-Wave.

CECHY

Nazwa	Opis
NodeID	Numer modułu (węzła) w sieci Z-Wave (nadawany dla każdego modułu Z-Wave po dodaniu go do kontrolera)
Banned	Informacja o zablokowaniu komunikacji Z-Wave z modułem: 0 - komunikacja z modułem nie jest zablokowana, 1 - zablokowana komunikacja z modułem (moduł zbanowany). Zablokowanie następuje w momencie, gdy 3 kolejne próby komunikacji z modułem zakończą się niepowodzeniem (inkrementacja cechy <code>FailCount</code> o 3). Do zbanowanego modułu, co 1,5 minuty wysyłane jest zapytanie - jeśli CLU dostanie odpowiedź, wówczas zablokowanie zostanie usunięte i możliwa jest ponowna próba wysłania rozkazu do modułu
FailCount	Liczba nieudanych prób komunikacji z modułem Z-Wave. W przypadku niepowodzenia komunikacji z modułem (brak odpowiedzi, potwierdzenia itp.) następuje inkrementacja cechy o 1, następnie próba ponawiania jest dwukrotnie (w interwałach 15s). W przypadku niepowodzenia komunikacja z modułem zostaje zablokowana (<code>Banned</code> =1)

METODY

Nazwa	Opis
<code>RemoveBan</code>	Zdejmuje blokadę komunikacji z modułem Z-Wave (w przypadku, gdy cecha <code>Banned</code> =1). Wywołanie metody umożliwi ponowne wysłanie rozkazu do modułu. UWAGA! <code>RemoveBan</code> nie jest jednoznaczne z ponowną poprawną komunikacją z modułem – umożliwia ponowne wysłanie rozkazu/zapytania do modułu! W przypadku niepowodzenia cały proces blokowania jest rozpoczynany na nowo!
<code>ClearFailCount</code>	Czyści liczbę nieudanych prób komunikacji

ZDARZENIA

Nazwa	Opis
<code>OnBanned</code>	Zdarzenie wywoływane, gdy urządzenie zostanie zbanowane

14. FAKRO FTP_V

Wersja modułu: HW: 25 FW: 1.01:01.01

14.1. Informacje ogólne

Obsługa modułu FAKRO FTP_V obejmuje możliwość sterowania oknem - zarówno poprzez maksymalne otwarcie / zamknięcie, jak i ustawienie wartości procentowej otwarcia okna.

Sposób dodawania / usuwania: Dodawanie / usuwanie urządzenia odbywa się poprzez naciśnięcie przycisku 'P' na urządzeniu podczas inclusion / exclusion (wywołane na CLU).

14.2. Obiekty

A. ZWAVE_FAKRO

Obiekt umożliwiający sterowanie otwarciem oknem i odczyt ustawionego procentu otwarcia. Dostępne jest ustawianie na wartość maksymalną (otwarcie / zamknięcie), a także podając procentową wartość otwarcia okna (0-100%).

CECHY

Nazwa	Opis
State	Stan urządzenia: 0 - stoi, 1 - otwieranie, 2 - zamykanie
Percent	Wartość procentowa otwarcia okna, gdzie: 0% - okno zamknięte, 100% - okno otwarte UWAGA! Wartość cechy <code>Percent</code> jest odświeżana w momencie, gdy sterownik okna zakończy daną pracę - należy mieć to na uwadze przy wykorzystywaniu tej cech np. dla komponentu <code>Slider</code> .
WaterSensor	Wartość z czujnika deszczu

METODY

Nazwa	Opis
Open	Otwarcie okna
Close	Zamknięcie okna
Stop	Stop, jeśli okno jest otwierane lub zamykane
Start	Zamknięcie okna jeśli poprzednio było otwierane, otwieranie okna jeśli poprzednio było zamykane
SetPercent	Ustawia wartość procentową, gdzie 100% - okno otwarte

ZDARZENIA

Nazwa	Opis
OnChange	Zdarzenie wywoływane w momencie zmiany stanu sterownika okna
OnOpen	Zdarzenie wywoływane w momencie zmiany stanu ze Stop na Open
OnClose	Zdarzenie wywoływane w momencie zmiany stanu ze Stop na Close
OnStart	Zdarzenie wywoływane w momencie wywołania komendy Start
OnStop	Zdarzenie wywoływane w momencie wywołania komendy Stop
OnRainChange	Zdarzenie wywoływane w przypadku zmiany stanu czujnika na przeciwny
OnRainOn	Zdarzenie wywoływane w momencie ustawienia stanu wysokiego na czujniku
OnRainOff	Zdarzenie wywoływane w momencie ustawienia stanu niskiego na czujniku

B. ZWAVE_CONFIG

Obiekt wyświetla informacje odnośnie parametrów komunikacji z modułem w sieci Z-Wave.

CECHY

Nazwa	Opis
NodeID	Numer modułu (węzła) w sieci Z-Wave (nadawany dla każdego modułu Z-Wave po dodaniu go do kontrolera)
Banned	Informacja o zablokowaniu komunikacji Z-Wave z modułem: 0 – komunikacja z modułem nie jest zablokowana, 1 – zablokowana komunikacja z modułem (moduł zbanowany). Zablokowanie następuje w momencie, gdy 3 kolejne próby komunikacji z modułem zakończą się niepowodzeniem (inkrementacja cechy <code>FailCount</code> o 3). Do zbanowanego modułu, co 1,5 minuty wysyłane jest zapytanie – jeśli CLU dostanie odpowiedź, wówczas zablokowanie zostanie usunięte i możliwa jest ponowna próba wysłania rozkazu do modułu
FailCount	Liczba nieudanych prób komunikacji z modułem Z-Wave. W przypadku niepowodzenia komunikacji z modułem (brak odpowiedzi, potwierdzenia itp.) następuje inkrementacja cechy o 1, następnie próba ponawiania jest dwukrotnie (w interwałach 15s). W przypadku niepowodzenia komunikacja z modułem zostaje zablokowana (<code>Banned</code> =1)
Register	Numer rejestru (parametru) konfiguracyjnego, który ostatnio został odczytany/ustawiony za pomocą dostępnych metod
value	Wartość rejestru (parametru) konfiguracyjnego

METODY

Nazwa	Opis
RemoveBan	Zdejmuje blokadę komunikacji z modułem Z-Wave (w przypadku, gdy cecha <code>Banned</code> =1). Wywołanie metody umożliwia ponowne wysłanie rozkazu do modułu. UWAGA! <i>RemoveBan nie jest jednoznaczne z ponowną poprawną komunikacją z modułem – umożliwia ponowne wysłanie rozkazu/zapytania do modułu! W przypadku niepowodzenia cały proces blokowania jest rozpoczynany na nowo!</i>
ClearFailCount	Czyści liczbę nieudanych prób komunikacji
Set	Ustawia wartość danego rejestru (parametru) konfiguracyjnego: <code>Register</code> (numer rejestru lub parametru), <code>value</code> (wartość rejestru bądź parametru), <code>Size</code> (rozmiar wysyłanej wartości rejestru lub parametru – w bajtach)
Get	Pobiera wartość danego rejestru (parametru) konfiguracyjnego
SetDefault	Ustawia wartość domyślną dla danego rejestru (parametru) konfiguracyjnego

ZDARZENIA

Nazwa	Opis
OnBanned	Zdarzenie wywoływane, gdy urządzenie zostanie zbanowane

XVII. Aplikacja mobilna myGrenton

1. Instalacja myGrenton

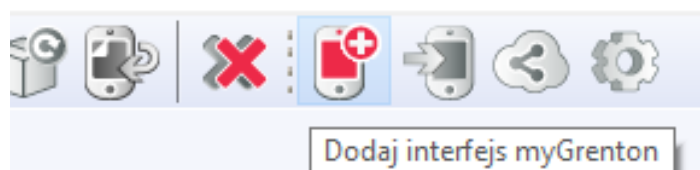
Aktualna wersja aplikacji dostępna jest w Sklepie Play dla telefonów lub tabletów z systemem **Android (wersja 5.0 lub wyższa)**. Po otwarciu aplikacji pokazuje się okno powitalne z informacjami o aplikacji oraz możliwość dodania nowego interfejsu.

UWAGA! Wymagana minimalna wersja CLU **05.03.06**

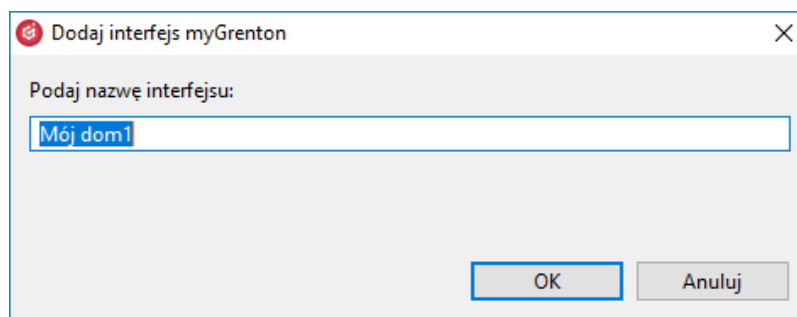
UWAGA! Aplikacja może prosić m.in. o pozwolenie na wykonywanie zdjęć i filmów wideo, pełny dostęp do sieci, wyświetlanie połączeń sieciowych, zapobieganie przejściu telefonu w stan uśpienia, odbieranie danych z internetu. Do poprawnego działania aplikacji należy wyrazić zgodę na powyższe warunki.

2. Tworzenie interfejsu

Aby utworzyć interfejs do aplikacji myGrenton należy kliknąć na ikonę **Dodaj interfejs myGrenton** w Menu Głównym Object Managera:



Następnie pojawi się okno z możliwością zmiany nazwy interfejsu:



Po zaakceptowaniu nazwy zostanie utworzony pusty interfejs:



2.1. Dodawanie strony do interfejsu

Aby dodać kolejną stronę interfejsu należy kliknąć ikonę **+**, która znajduje się obok grafiki telefonu.

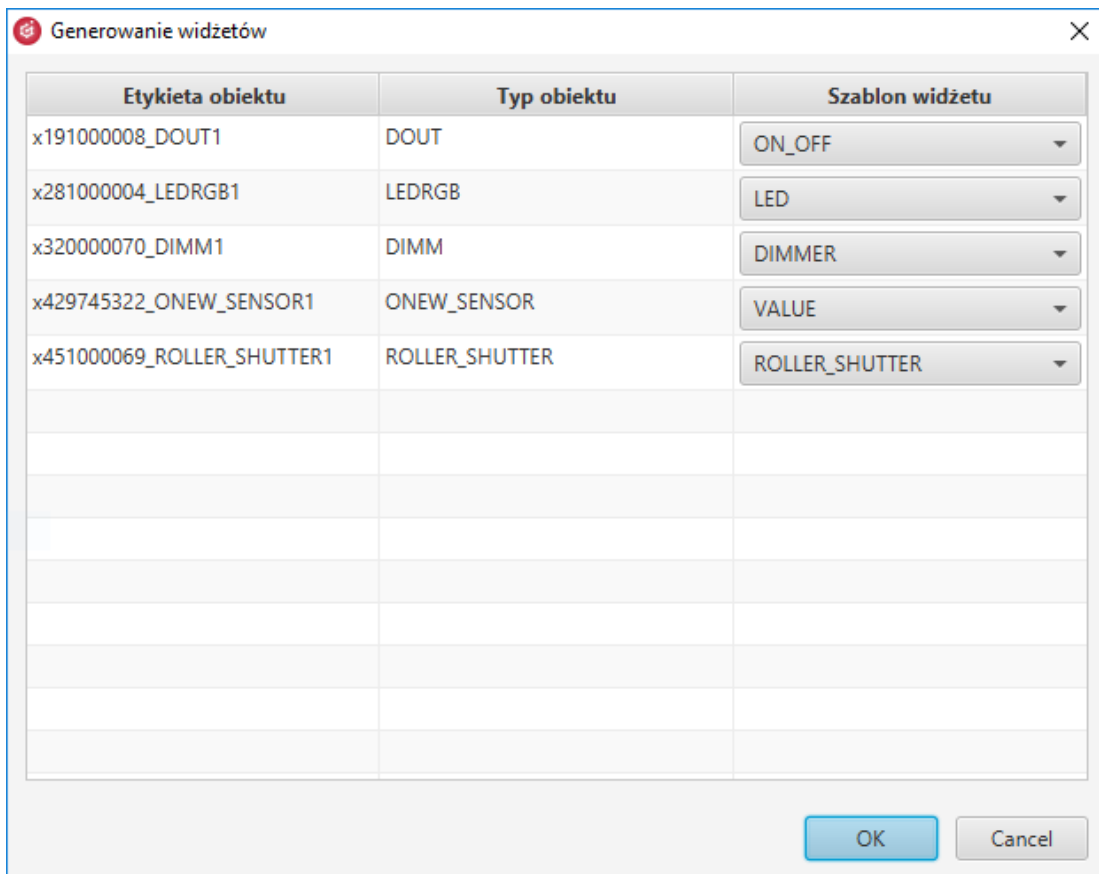
2.2. Usuwanie strony z interfejsu

Aby usunąć stronę z interfejsu należy kliknąć ikonę **x**, która znajduje się w prawym górnym rogu telefonu a następnie potwierdzić usunięcie strony.

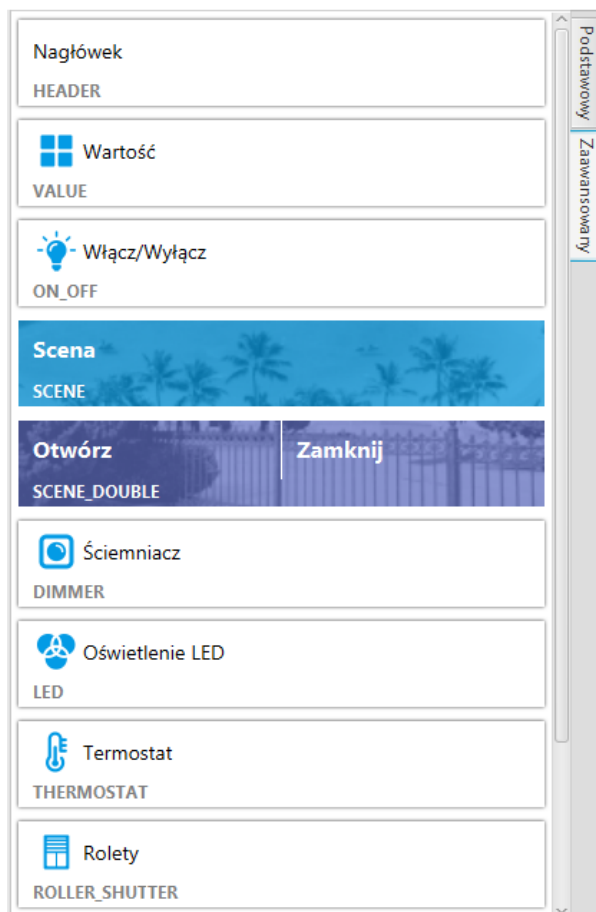
3. Widżety

Do sterowania systemem za pomocą aplikacji myGrenton z poziomu telefonu lub tabletu służą widżety. Każdy z nich oferuje inną funkcjonalność. Widżety mogą zostać dodane do interfejsu na 2 sposoby:

1. Przeciągając konkretny obiekt z listy obiektów w Object Managerze (wtedy utworzony widżet będzie posiadał uprzednio zdefiniowany szablon)



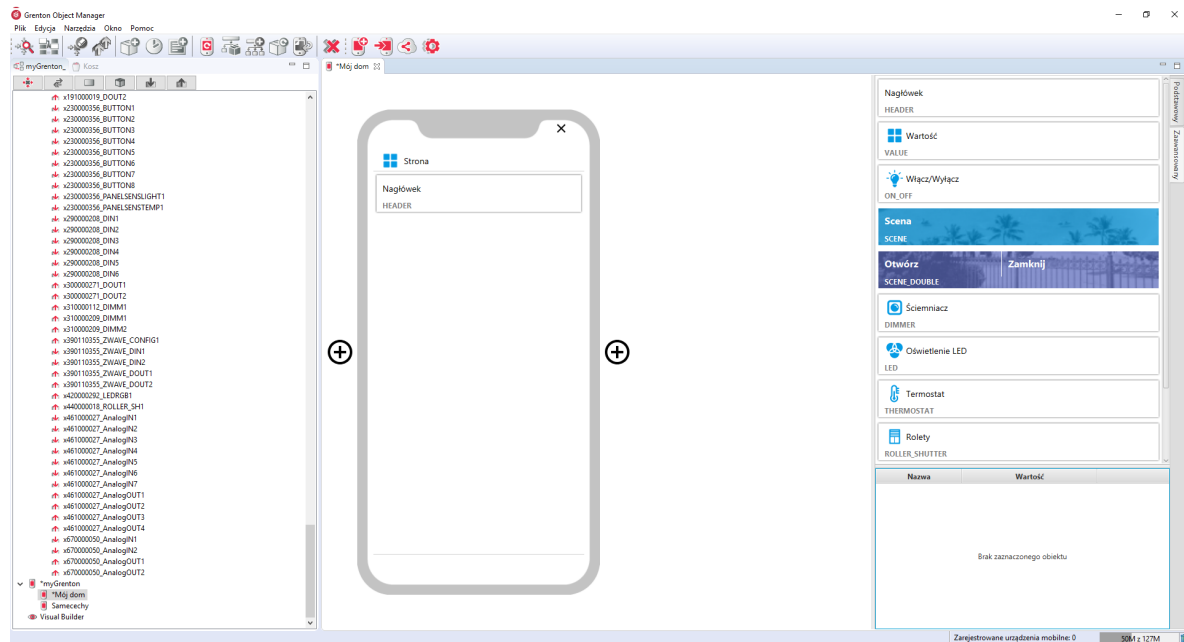
2. Przeciągając konkretny widżet z zakładki **Zaawansowany** a następnie uzupełniając go o pożądane parametry. Po przejściu do zaawansowanej konfiguracji z prawej strony ekranu pojawia się tabela z dostępnymi widżetami:



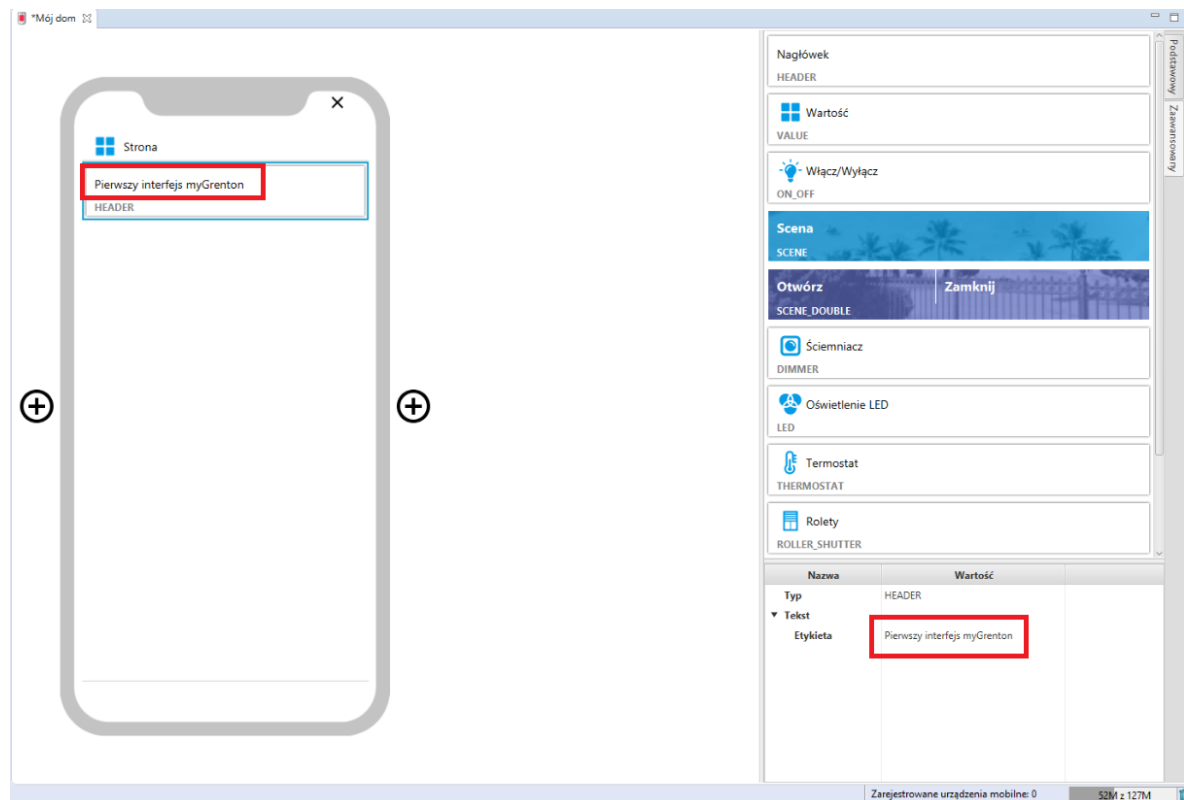
UWAGA! Maksymalna ilość widgetów na stronę wynosi 30.

3.1. Nagłówek (HEADER)

Służy do wyświetlania ciągu znaków. Obsługiwane są wszystkie znaki alfabetyczne, numeryczne oraz znaki specjalne.

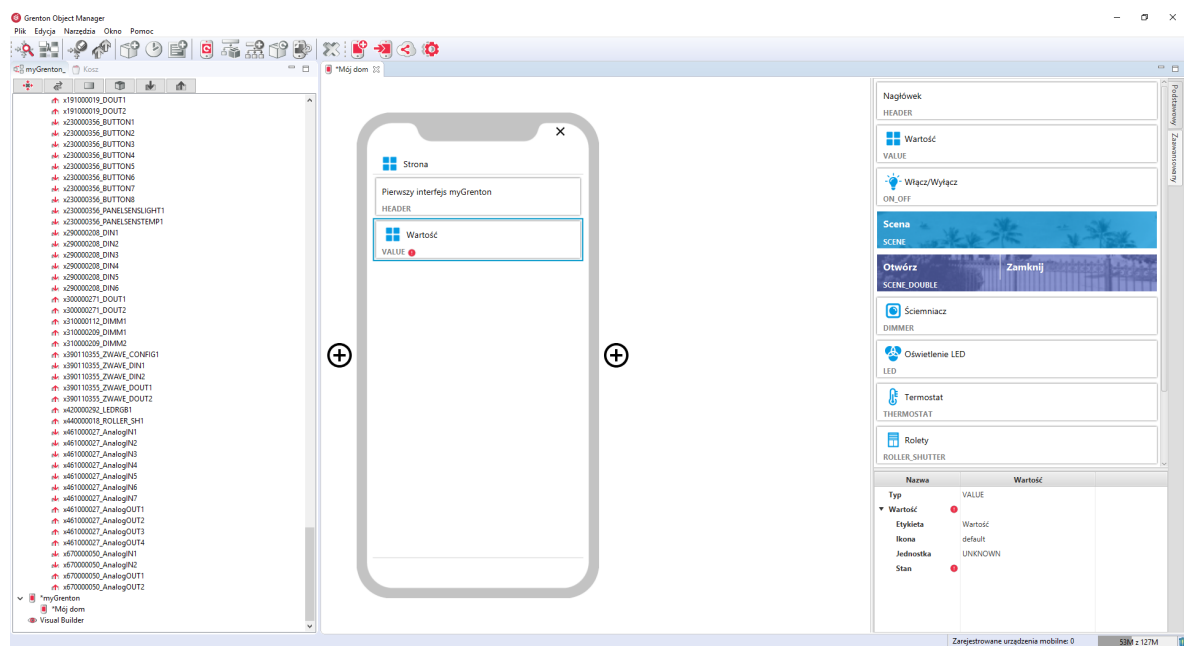


Aby zmienić nagłówek należy dwukrotnie kliknąć napis **Nagłówek** w polu telefonu lub uzupełnić **Etykieta** w polu wartość.

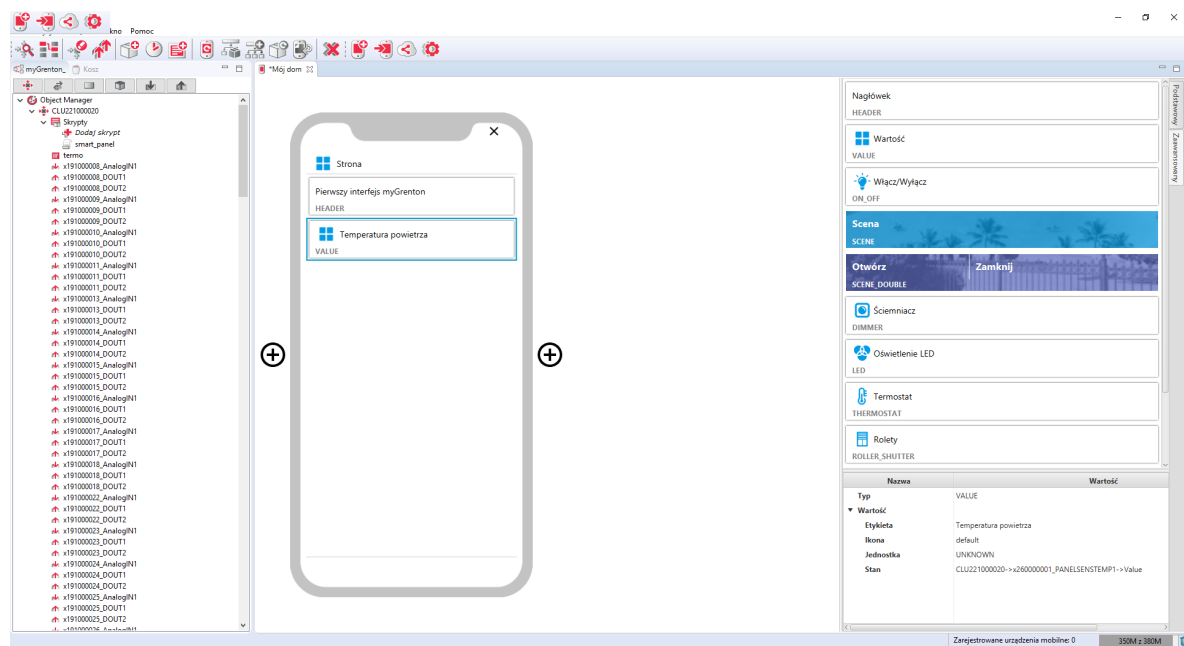


3.2 Wartość (VALUE)

Widget zwraca wartości cechy danego obiektu. Widget ten dysponuje trzema jednostkami do wyboru: niezdefiniowana (UNKNOWN), procentowa (PERCENT) oraz stopniowa (DEGREE). Ponadto istnieje możliwość opisanego widgetu w polu **Etykieta** a także zmiany ikony w polu **Ikona**.

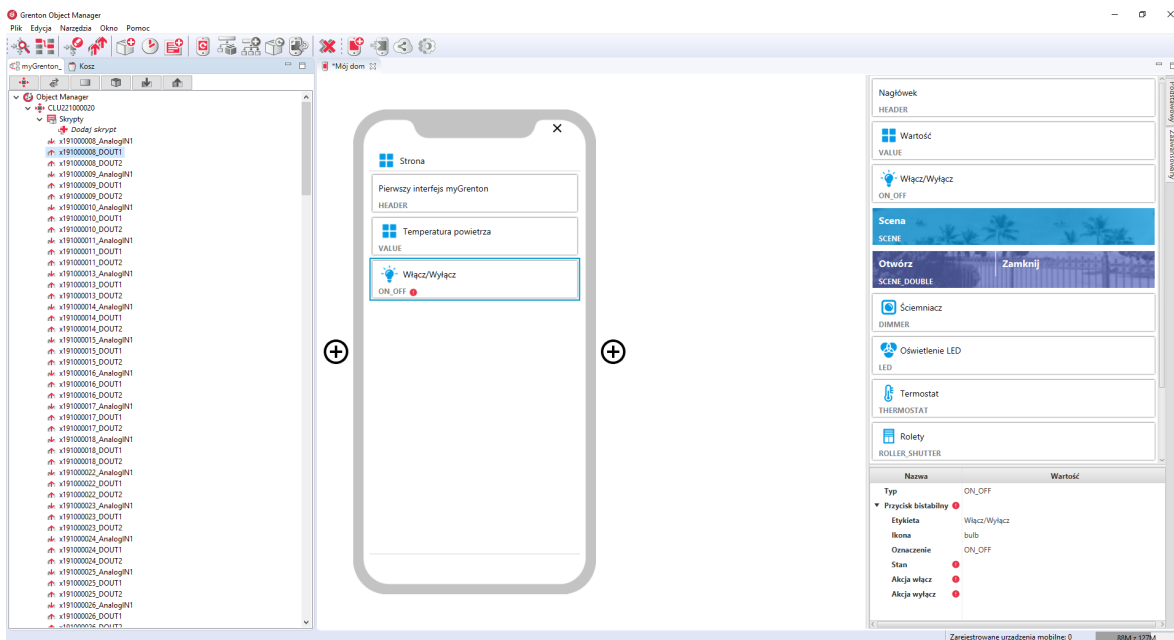


Uzupełniony widget:

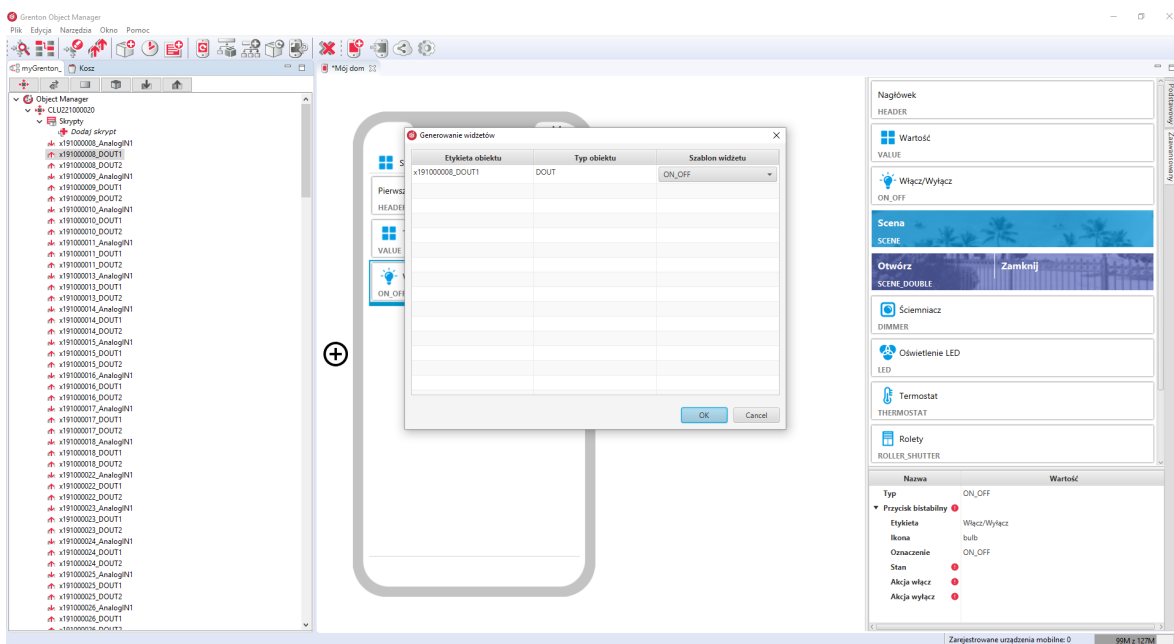


3.3 Włącz/Wyłącz (ON_OFF)

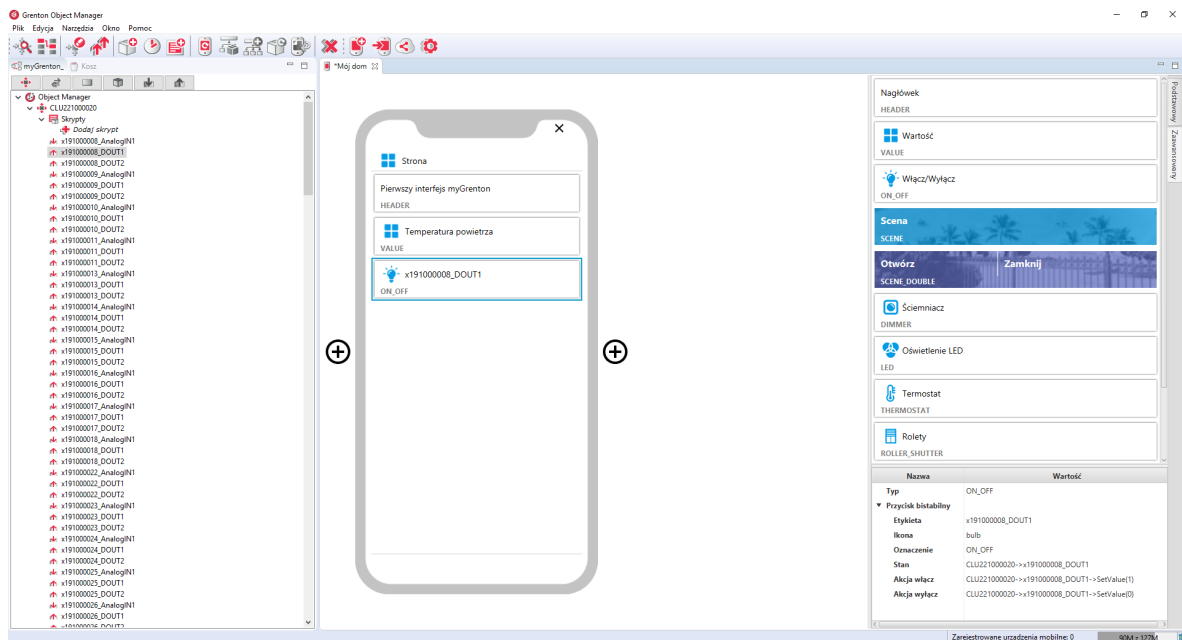
Widget służy do sterowania elementami wykonawczymi. Przeciągając widget z zakładki **Zaawansowany** należy ją uzupełnić o wartości w polach **Stan**, **Akcja włącz**, **Akcja wyłącz**. Widget dedykowany głównie do sterowania wyjściami przekaźnikowymi, jednak może być zastosowany m.in. do załączania/wyłączania oświetlenia LED, uruchamiania obiektów wirtualnych.



Dla obiektów wyjść cyfrowych zdefiniowane są gotowe szablony dla widgetu ON_OFF. Aby dodać widget ON_OFF z pożądanym obiektem wyjścia cyfrowego należy przeciągnąć obiekt DOUT z listy obiektów na stronę interfejsu:

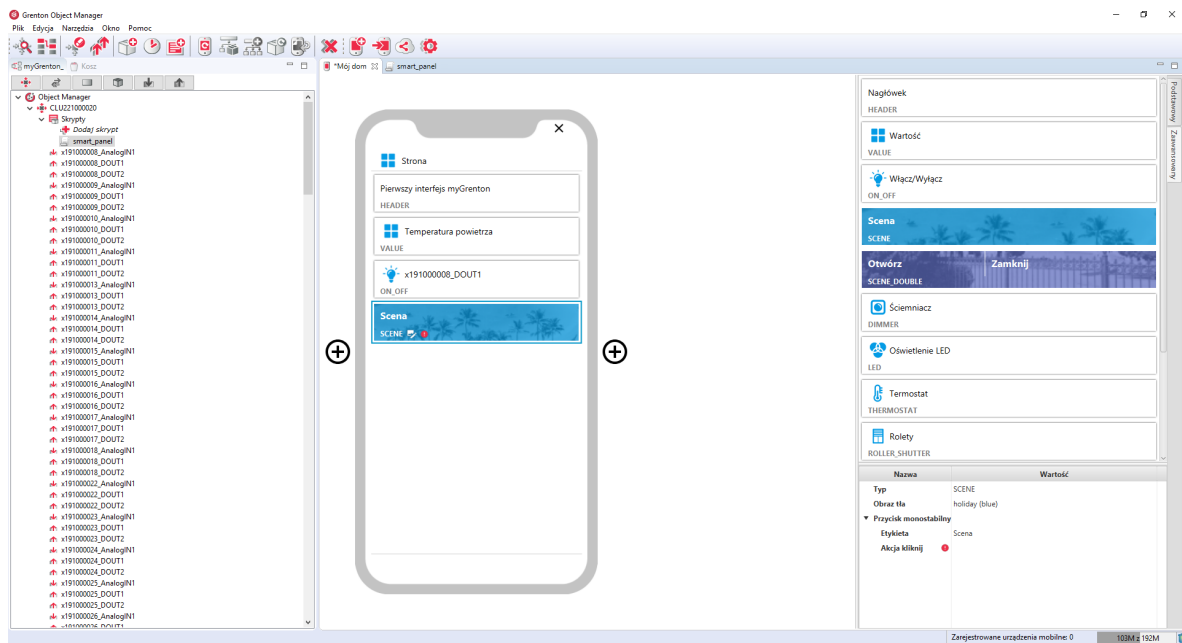


Utworzony widget będzie wyglądał następująco:

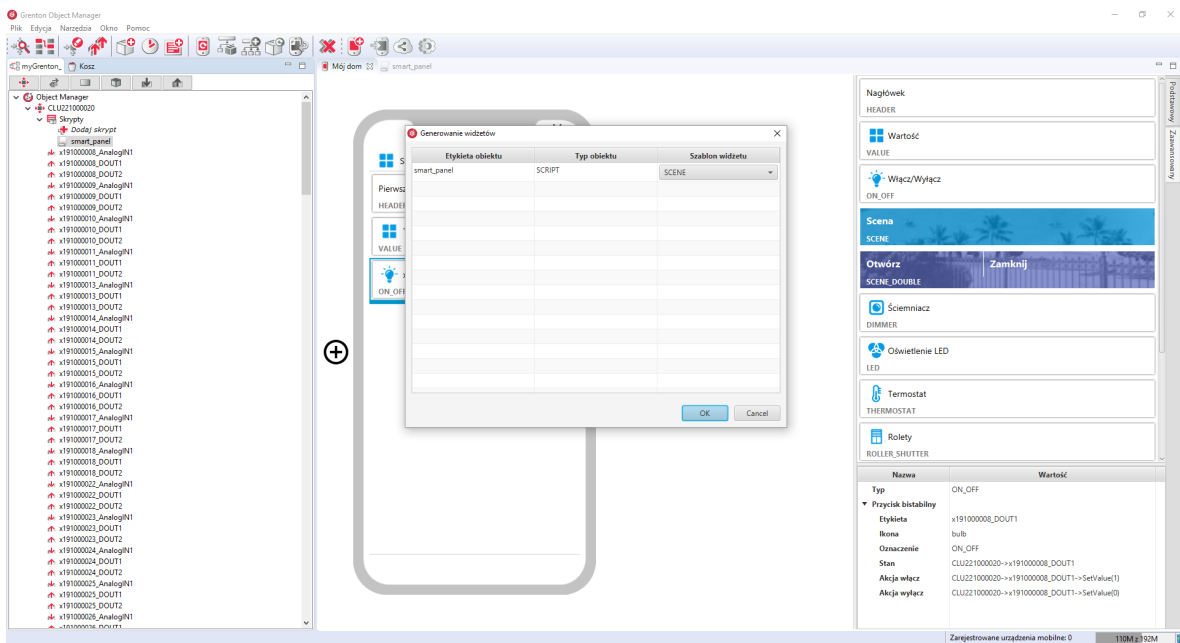


3.4 Scena (SCENE)

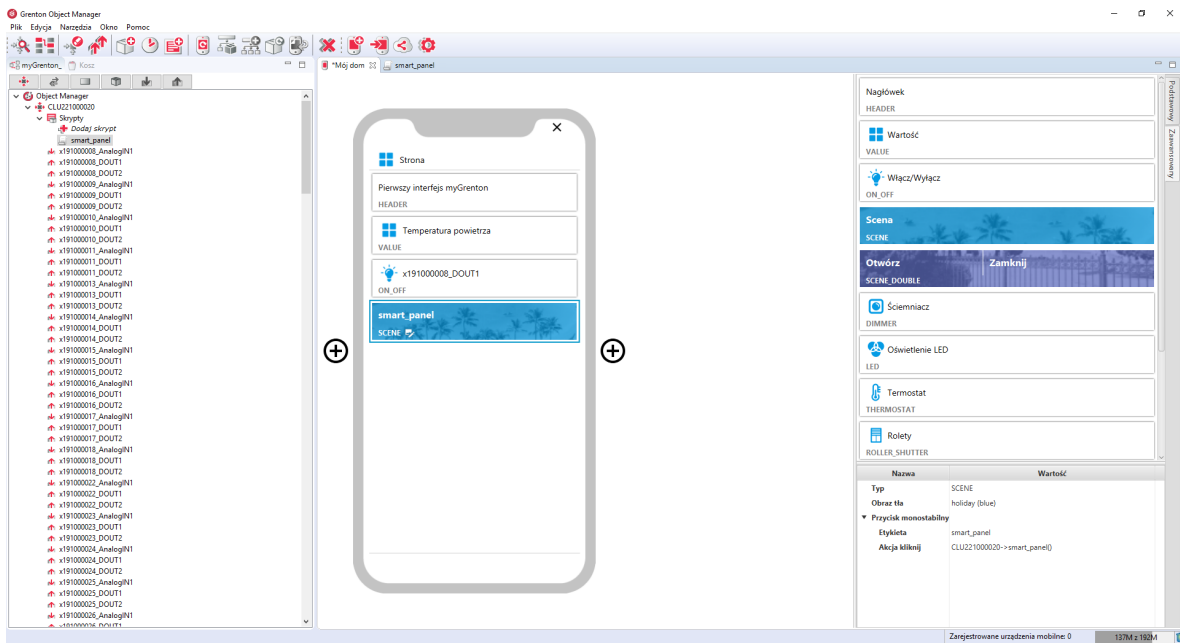
Widget dedykowany jest do wywoływania utworzonych skryptów.



Dla skryptów zdefiniowane są gotowe szablony dla widgetu SCENE. Aby dodać widget SCENE z pożądanym obiektem wyjścia cyfrowego należy przeciągnąć skrypt z listy obiektów na stronę interfejsu:

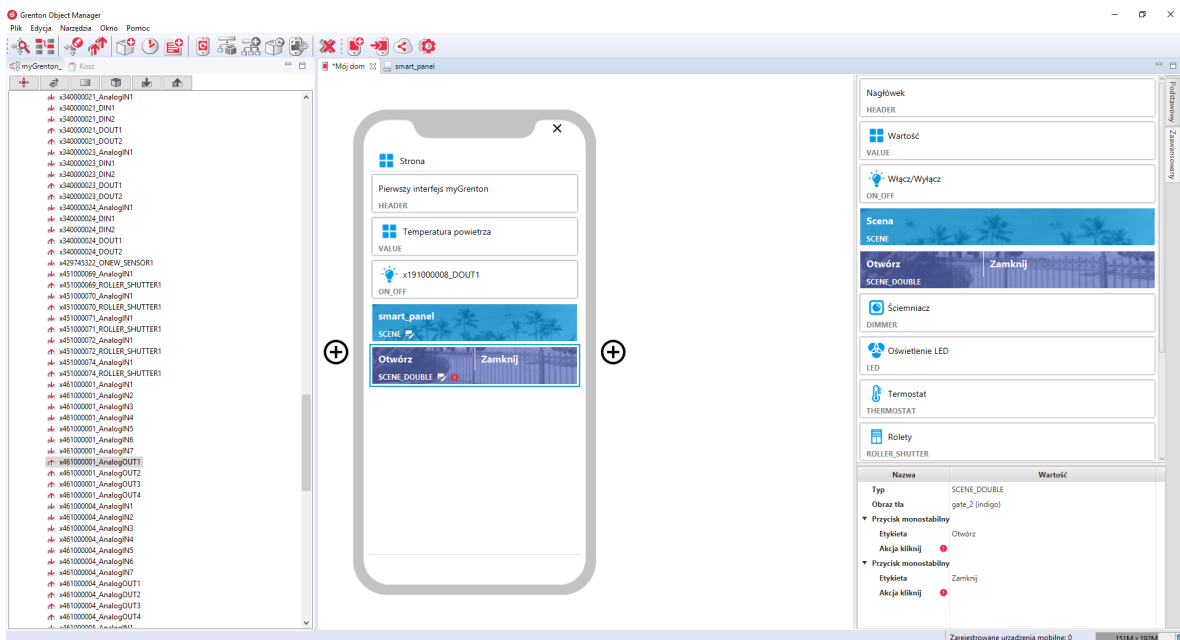


Utworzony widżet będzie wyglądać następująco:

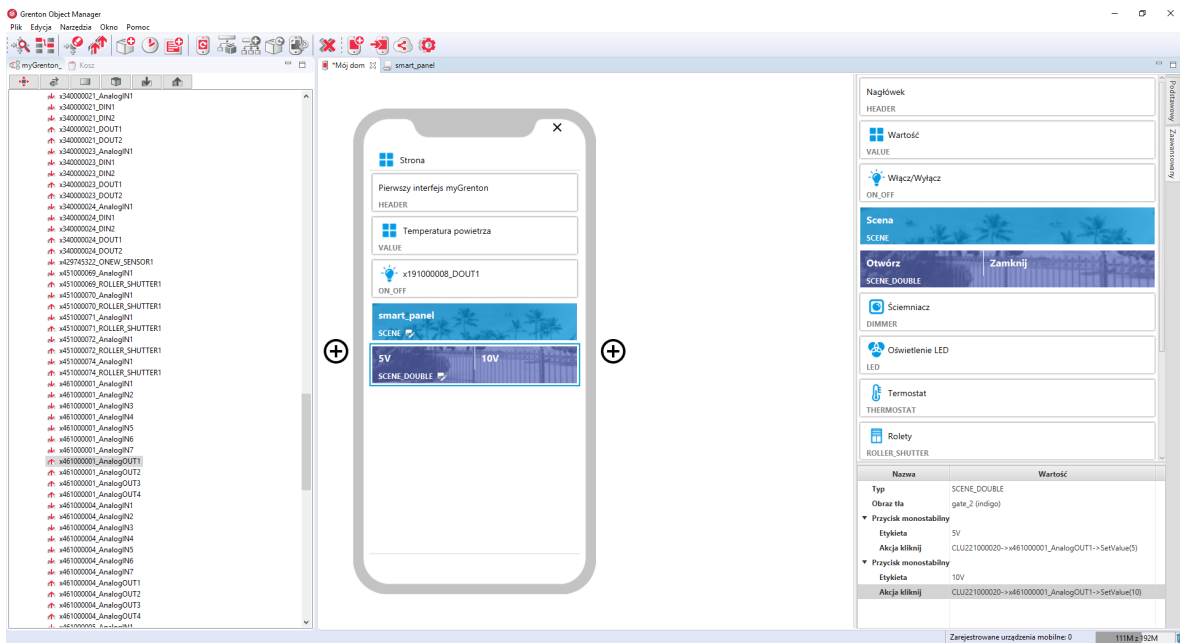


3.5 Otwórz/Zamknij (SCENE_DOUBLE)

Widżet służy do wywoływania skryptów (metod obiektów). W odróżnieniu do widżetu SCENE posiada 2 przyciski na akcję.

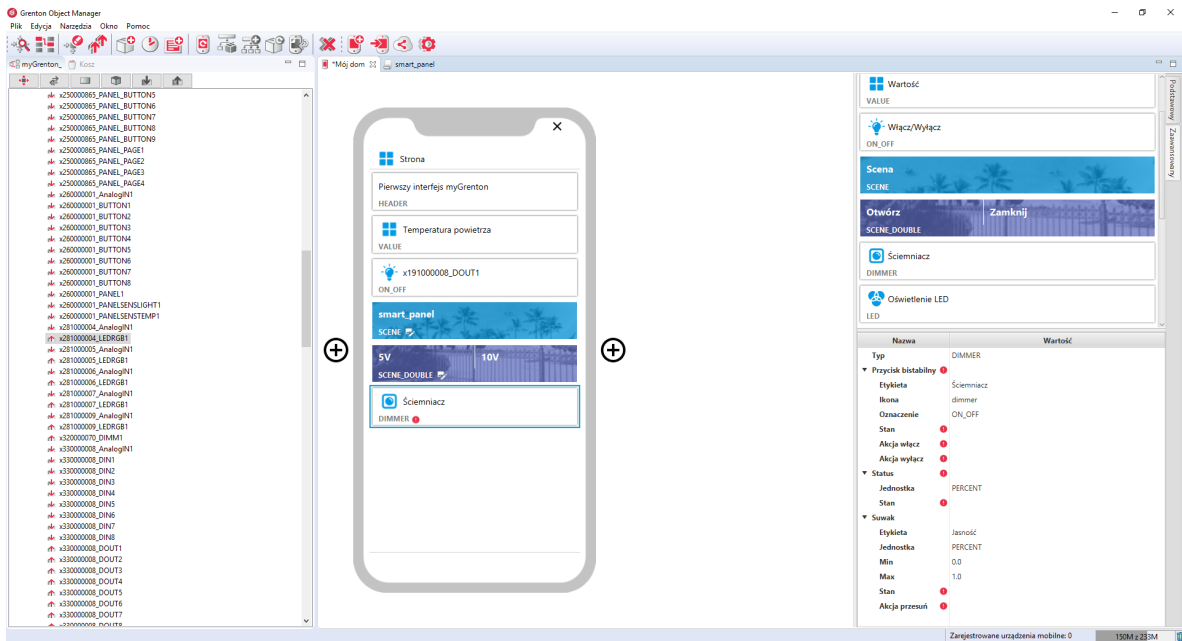


Przykładowo uzupełniono pola Etykiety i akcji na kliknięcie:

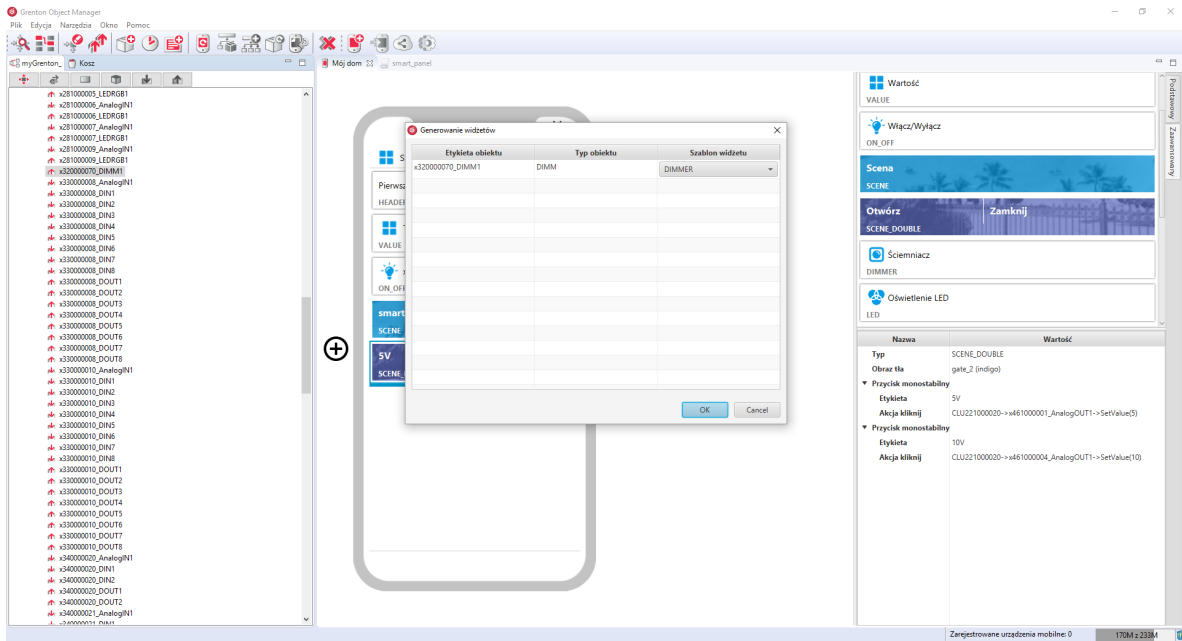


3.6. Ściemniacz (DIMMER)

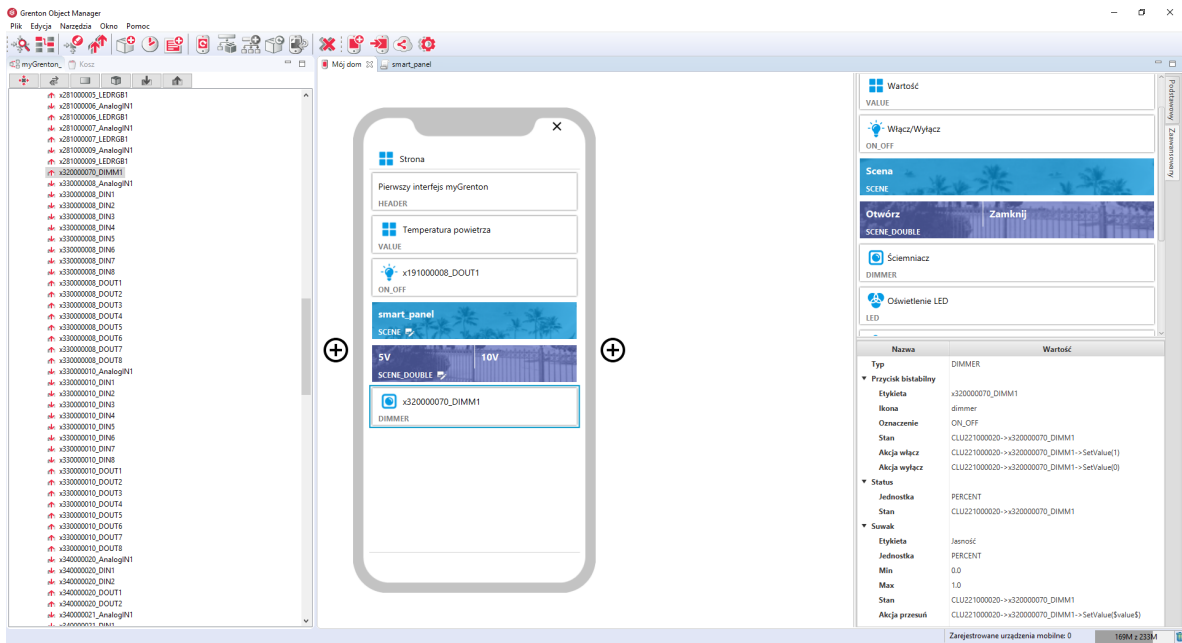
Widget dedykowany dla modułów DIMMER i LEDRGB. Przeciągając widget z zakładki **Zaawansowany** należy ją uzupełnić o wartości w polach **Stan**, **Akcja włącz**, **Akcja wyłącz**, **Jednostka**, **Min**, **Max**, **Akcja przesun**. Widget ten posiada suwak, przez co może umożliwić sterowanie w danym zakresie.



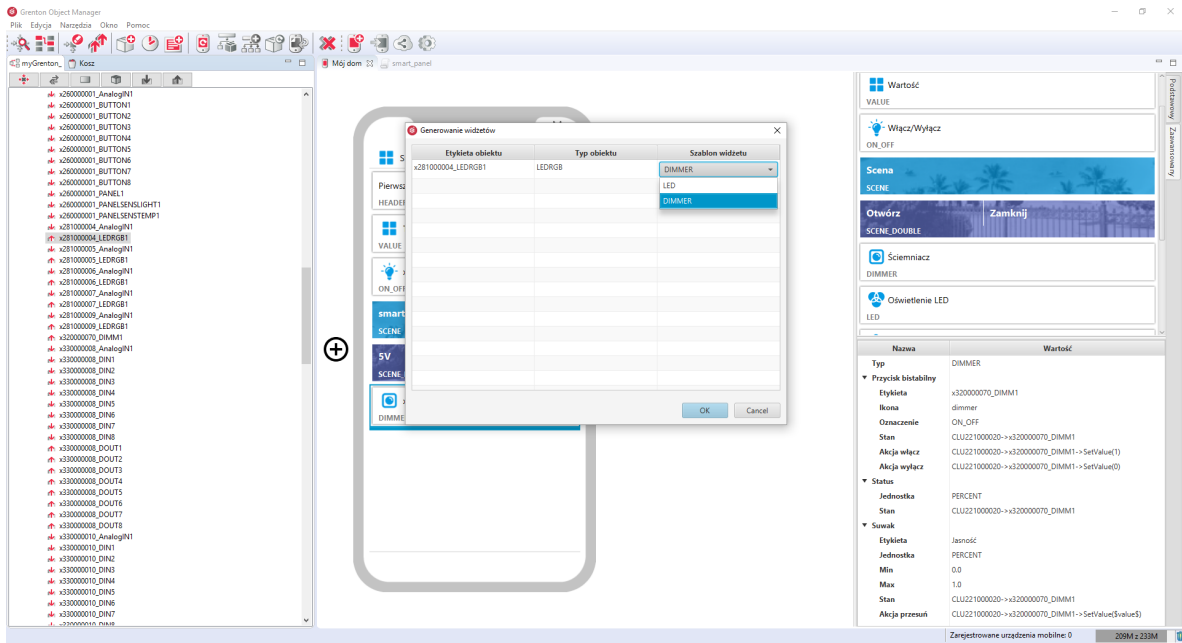
Dla obiektu DIMM zdefiniowane są gotowe szablony dla widgetu DIMMER. Aby dodać widget DIMMER z z gotowym szablonem należy przeciągnąć obiekt DIMM z listy obiektów na stronę interfejsu:



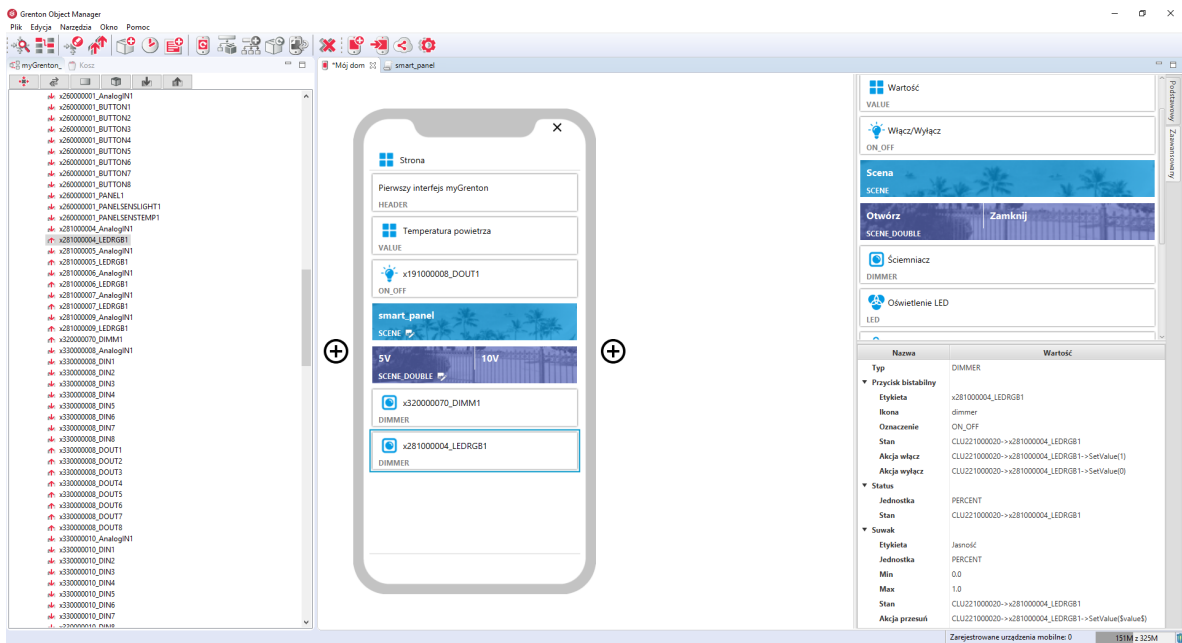
Utworzony widget prezentuje się następująco:



Widget DIMMER posiada gotowy szablon także dla obiektu LEDRGB:

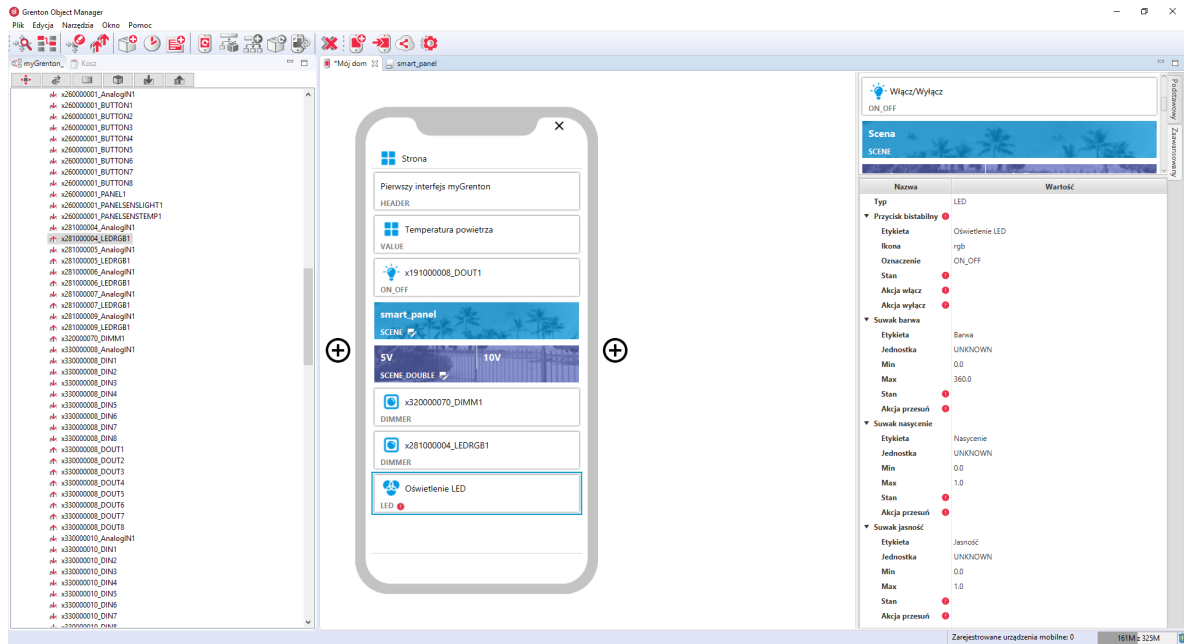


Utworzony widget:

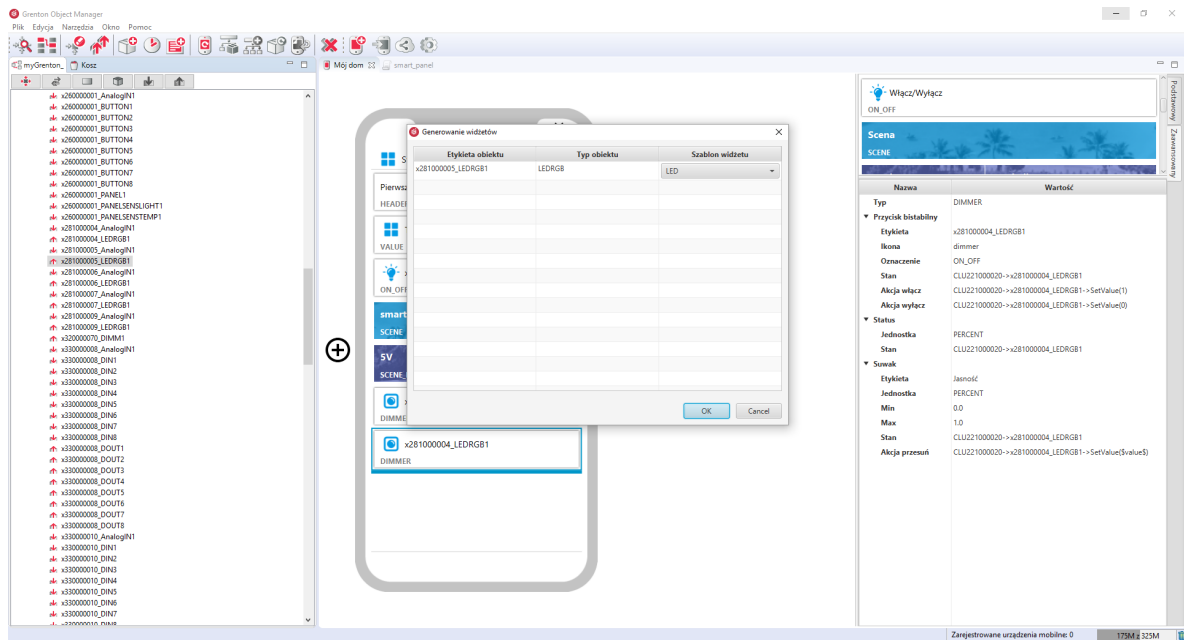


3.7. Oświetlenie LED (LED)

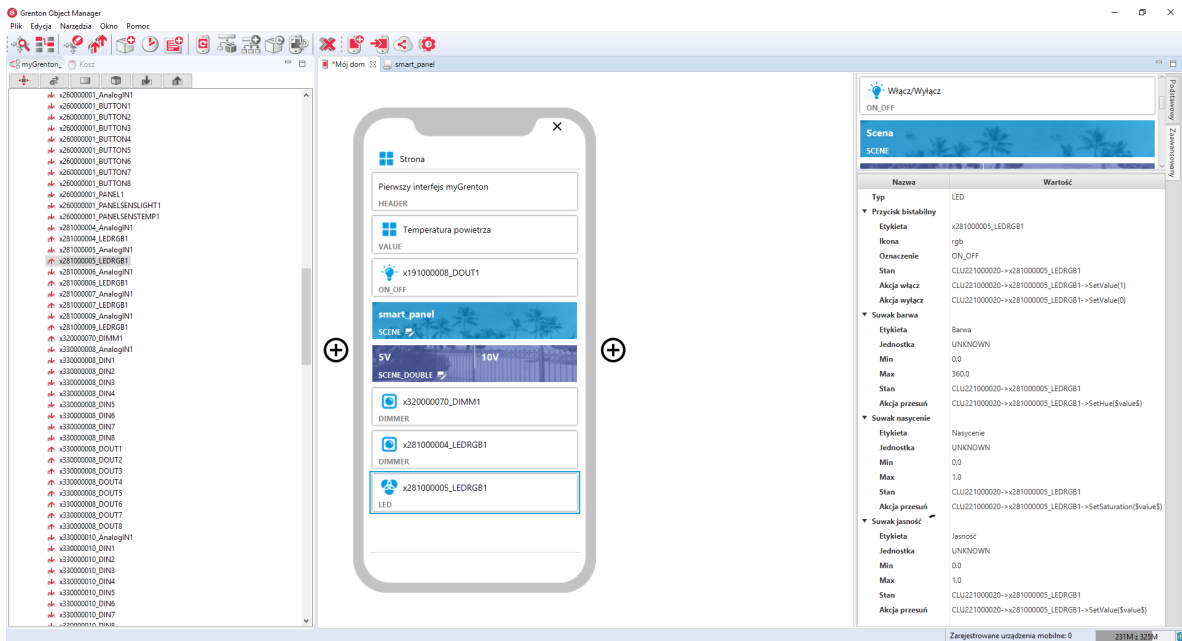
Widget dedykowany do oświetlenia LED. Posiada 1 przycisk bistabilny działający na zasadzie załącz/wyłącz oraz 3 suwaki: Suwak sterujący barwą, Suwak sterujący nasyceniem, Suwak sterujący jasnością.



Dla oświetlenia LED zdefiniowane są gotowe szablony dla widgetu LED. Aby dodać widget LED z gotowym szablonem należy przeciągnąć obiekt LEDRGB z listy obiektów na stronę interfejsu:



Utworzony widget:

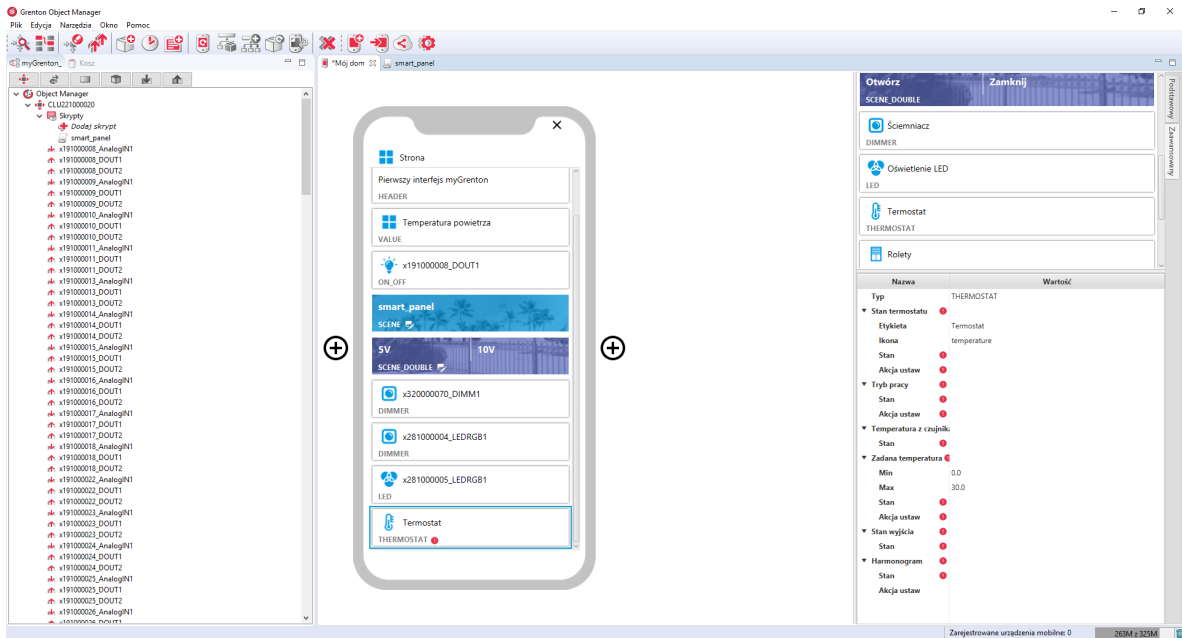


3.8 Termostat (THERMOSTAT)

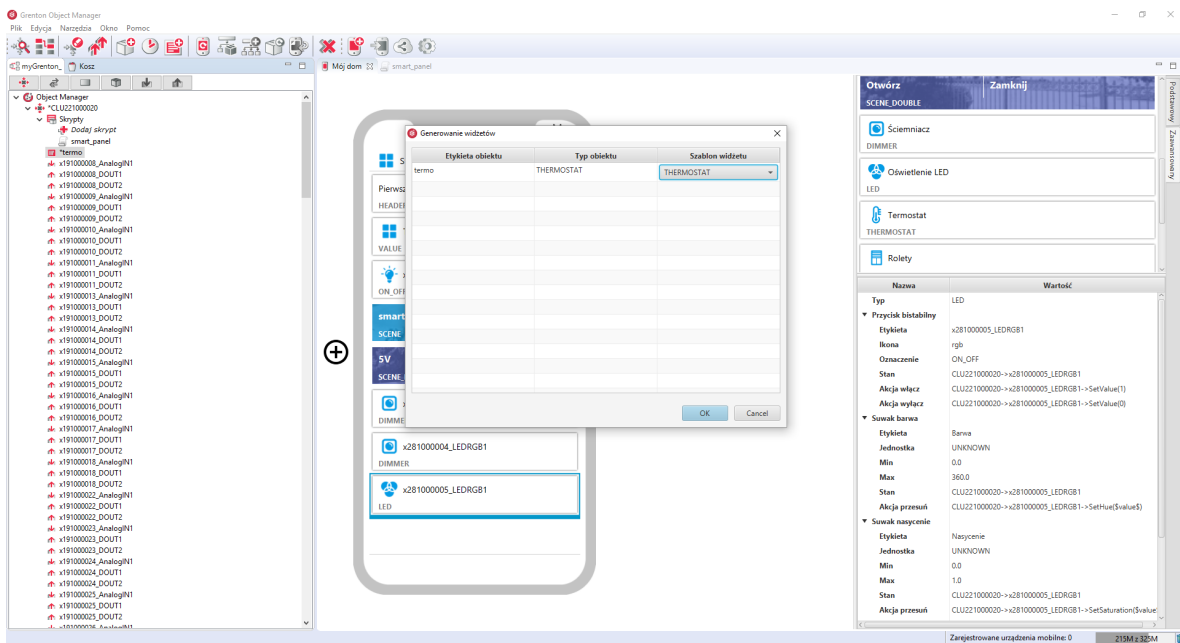
Uwaga!!

Widget obsługiwany jest dla termostatów utworzonych w **CLU 2.0 !!**

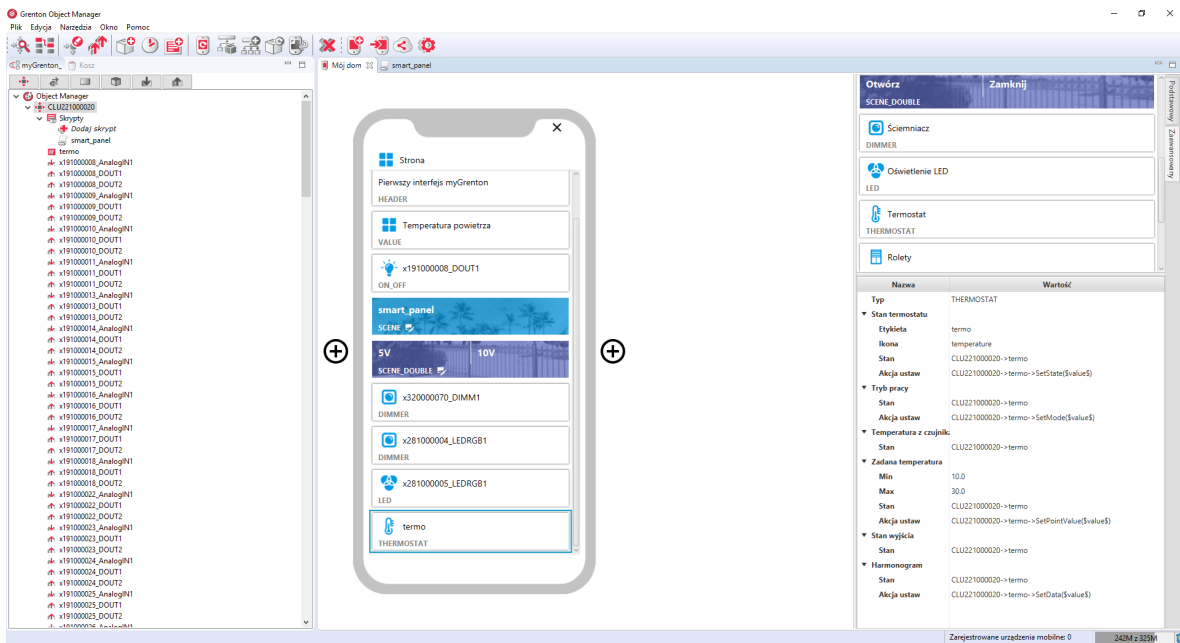
Widget dedykowany dla obiektów wirtualnych typu termostat. W przypadku gdy przeciągamy z listy obiektów zdefiniowany już termostat do interfejsu, utworzony widget jest uzupełniony w oparciu o podane cechy wejścia i wyjścia termostatu.



Dla termostatów zdefiniowane są gotowe szablony dla widgetu THERMOSTAT. Aby dodać widget THERMOSTAT z gotowym szablonem należy przeciągnąć obiekt wirtualny termostat z listy obiektów na stronę interfejsu:

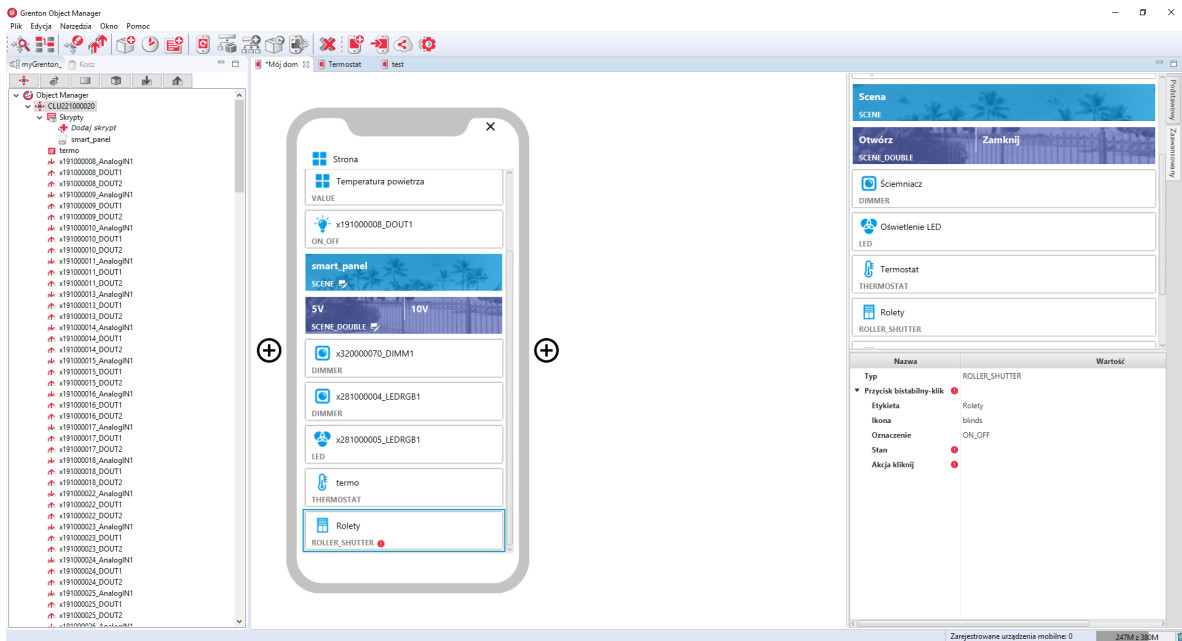


Uzupełniony widget THERMOSTAT:

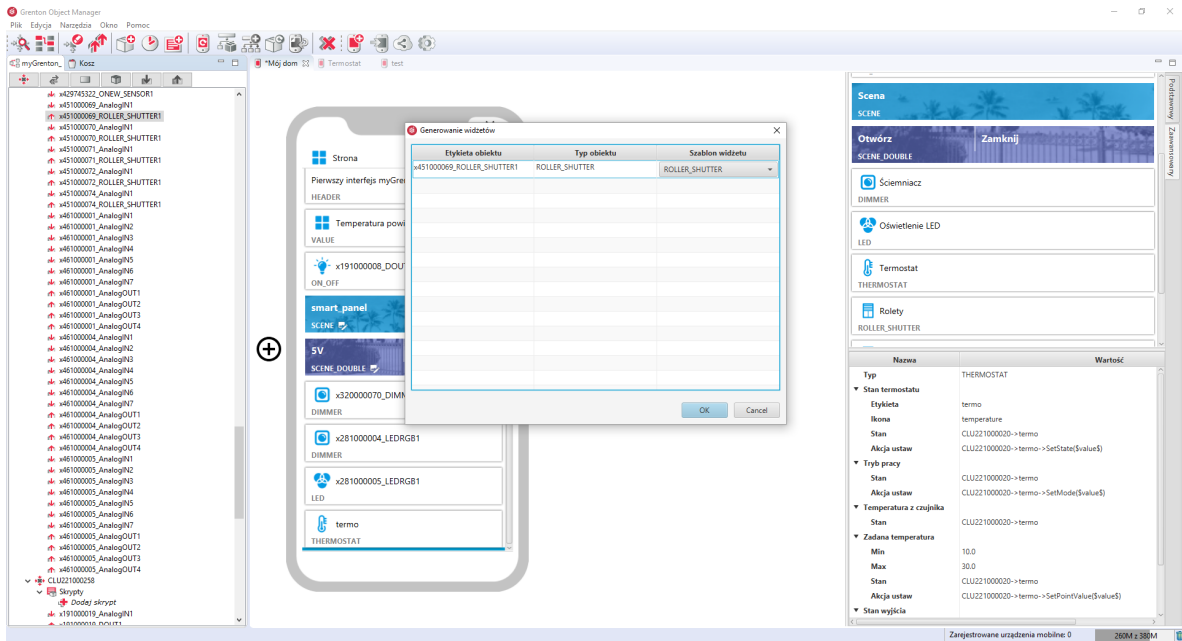


3.9. Rolety (ROLLER_SHUTTER)

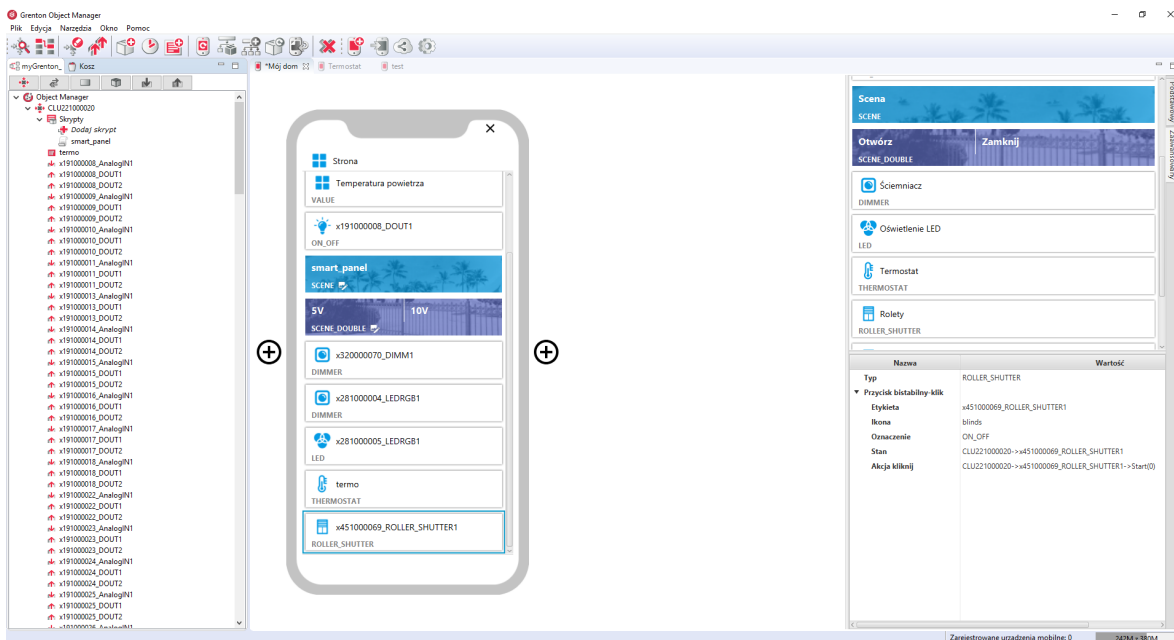
Widget służy do sterowania roletami. Aplikacja wyświetla obecny stan rolet (ZATRZYMANE, ZAMYKANIE, OTWIERANIE).



Dla sterowników rolet zdefiniowane są gotowe szablony dla widgetu ROLLER_SHUTTER. Aby dodać widget ROLLER_SHUTTER z gotowym szablonem należy przeciągnąć obiekt ROLLER_SHUTTER z listy obiektów na telefon:



Gotowy widget:

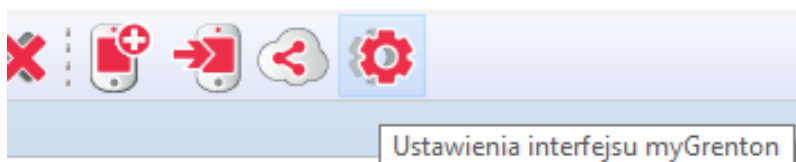


3.10. Usuwanie widgetu

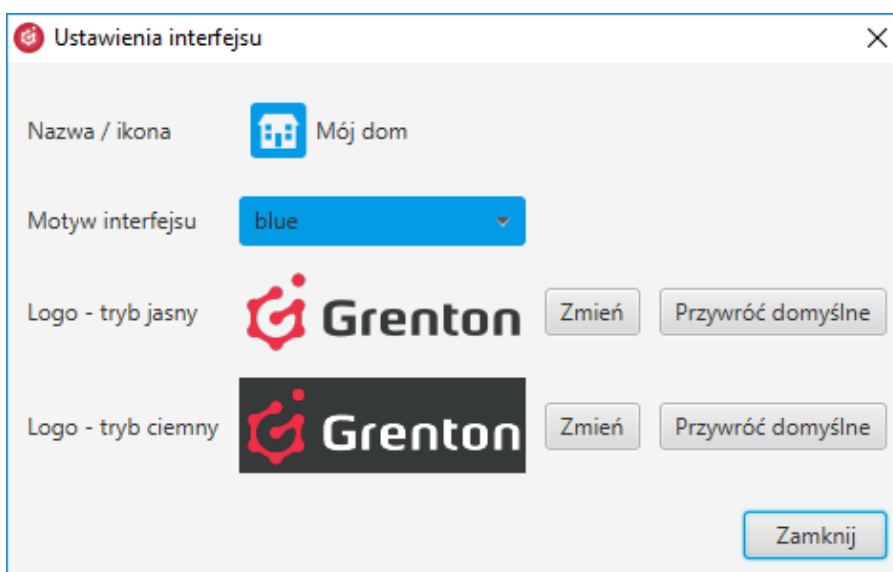
Aby usunąć widget należy go zaznaczyć i wcisnąć klawisz `Delete`. Na urządzeniu z zainstalowanym systemem operacyjnym macOS należy użyć kombinacji znaków `Fn + Backspace`.

4. Personalizacja interfejsu

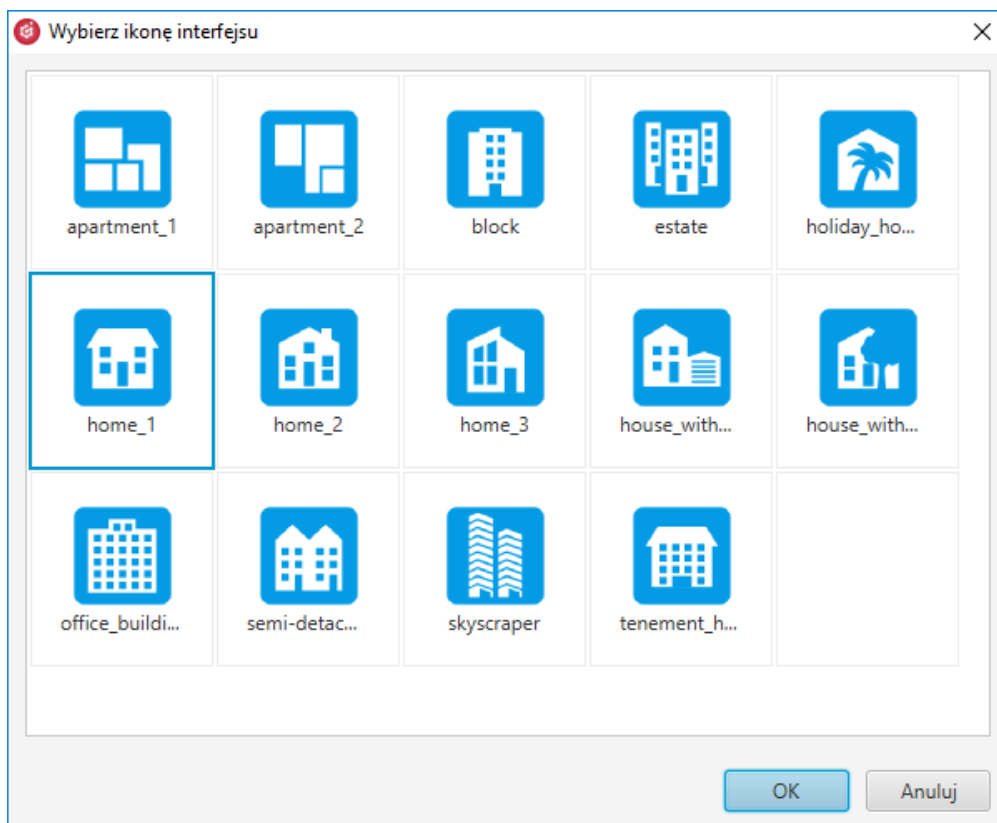
Istnieje możliwość dostosowania wyglądu interfejsu do swoich preferencji. Aby spersonalizować interfejs należy kliknąć ikonę `Ustawienia interfejsu myGrenton` w pasku narzędzi:



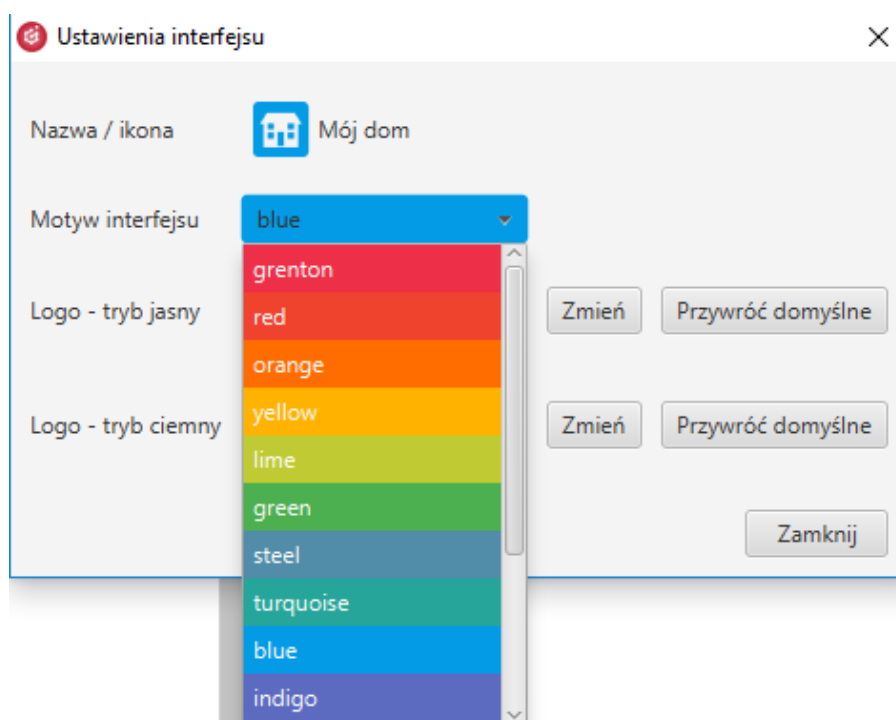
Po kliknięciu ikony zębatki pojawi się okno z ustawieniami interfejsu:



Możliwa jest zmiana nazwy interfejsu (dwuklik na nazwę - tu `mój dom`) oraz wyświetlanej ikony:



Istnieje również możliwość zmiany koloru motywu interfejsu:



5. Wysyłanie interfejsu na urządzenie

Dla utworzonego interfejsu możliwe jest jego wysłanie na urządzenie mobilne na 2 sposoby:

5.1. Wysyłanie na telefon interfejs myGrenton

Aby wysłać interfejs na telefon należy kliknąć w Object Managerze ikonę `wyślij na telefon interfejs myGrenton`:



Wyświetlone okno umożliwia przesłanie interfejsu poprzez skanowanie kodu QR bądź podając parametry interfejsu:

Wyślij na urządzenie mobilne

Teraz możesz wysłać interfejs na urządzenie mobilne

Uruchom w urządzeniu mobilnym aplikację myGrenton, w menu wybierz opcję 'Dodaj nowy interfejs' i zeskanuj kod QR wyświetlony po lewej stronie tego tekstu.

Jezeli nie masz możliwości zeskanowania kodu QR, wybierz w aplikacji myGrenton manualny tryb ładowania interfejsu i wprowadź następujące dane:

Adres IP: 192.168.3.3
Port: 9998
Token: 948 572

Uwaga!

- Komputer z aplikacją Object Manager i urządzenie mobilne z aplikacją myGrenton muszą być podłączone do tej samej sieci LAN.
- Okno dialogowe musi pozostać otwarte do momentu ukończenia przesyłania interfejsu.

Zamknij

Uwaga!! Komputer z aplikacją Object Manager i urządzenie mobilne z aplikacją myGrenton muszą być podłączone do tej samej sieci LAN. Okno dialogowe musi pozostać otwarte do momentu ukończenia przesyłania interfejsu.

Na telefonie należy wybrać opcję skanowania kodu QR lub wpisania ręcznego. W przypadku podania danych, należy uzupełnić dane wyświetlone w oknie Object Managera.

11:31 LTE 87

×

Podaj dane by wczytaj interfejs

Te dane są niezbędne do pobrania interfejsu z aplikacji Object Manager.

Adres IP
192.168.1.1
np. 192.168.1.2

Port
9998

Token
123456

6 / 6

WCZYTAJ INTERFEJS

Po prawidłowym przesłaniu aplikacja automatycznie uruchomi wczytany interfejs.

5.2. Udostępnienie w chmurze interfejsu myGrenton a następnie wysłanie na telefon



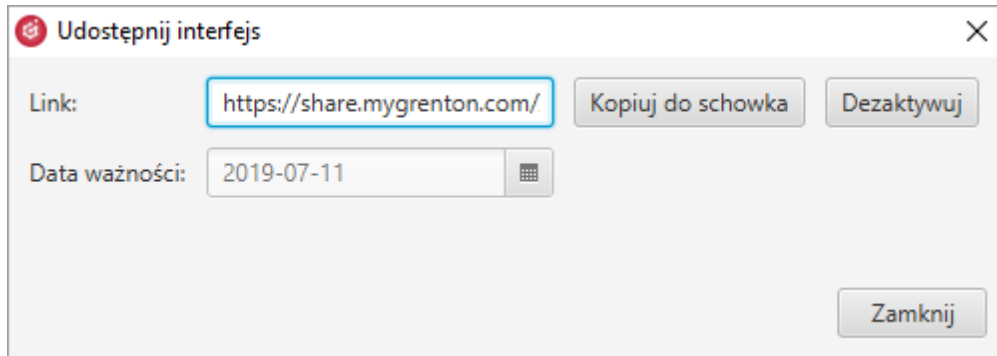
Po kliknięciu ikony **Udostępnij w chmurze interfejs myGrenton** pojawi się okno udostępniania interfejsu do chmury Grenton:

Udostępnij interfejs ×

Link:

Data ważności:

Możliwe jest ustawienie daty ważności linku z interfejsem. Po kliknięciu **Pobierz link** pojawia się link do strony z wygenerowanym interfejsem:

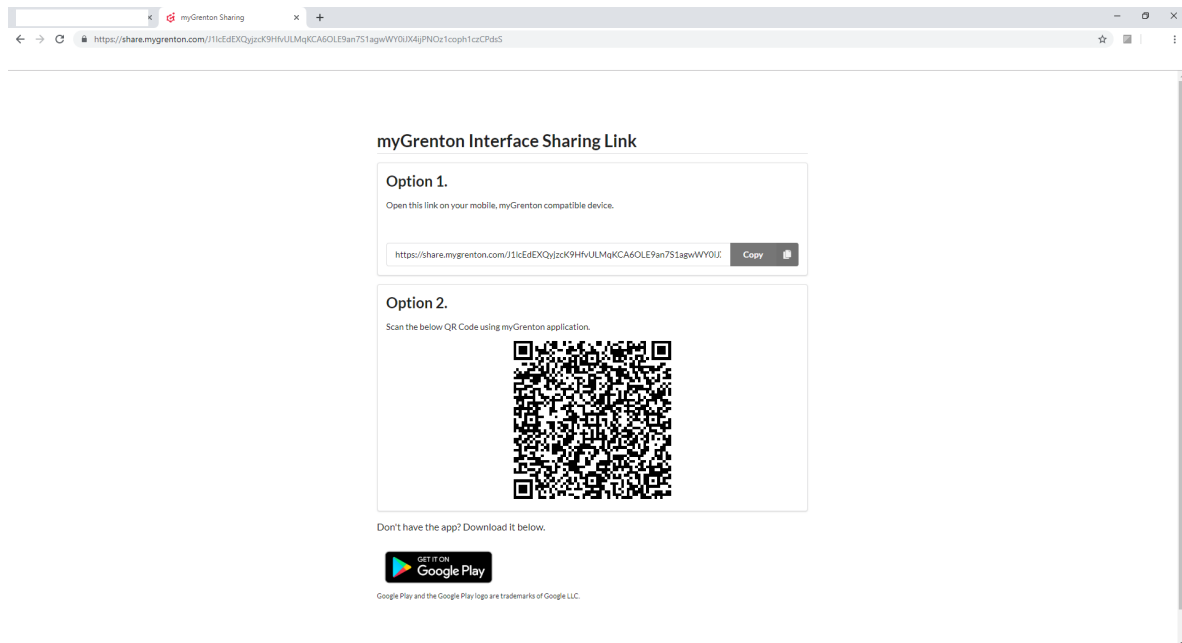


Udostępnij interfejs

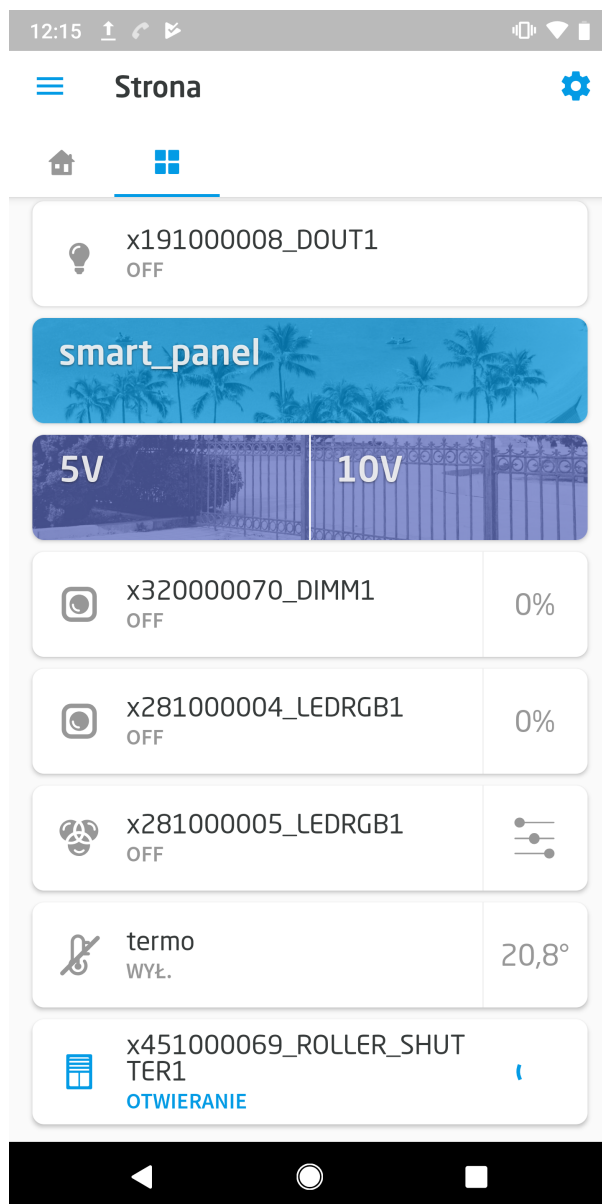
Link:

Data ważności:

Wygenerowany link można skopiować do schowka. Wklejając link w dowolną przeglądarkę pokaże się kod QR z interfejsem.



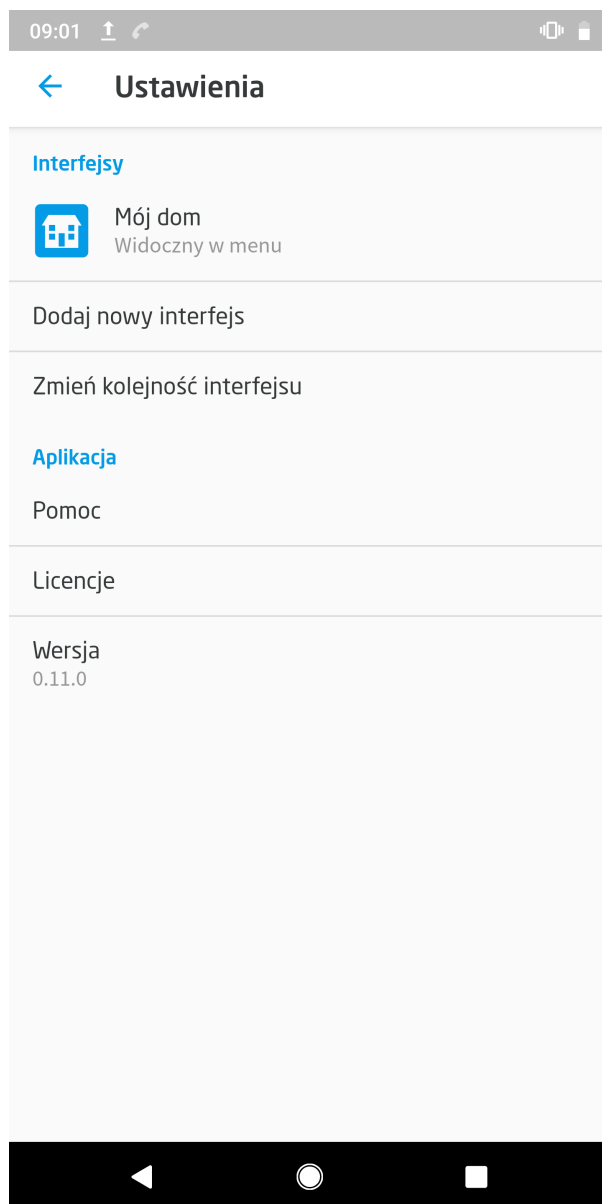
Utworzony kod QR skanujemy w aplikacji. Po poprawnym wczytaniu otwiera się interfejs w telefonie:



6. Ustawienia aplikacji oraz interfejsu

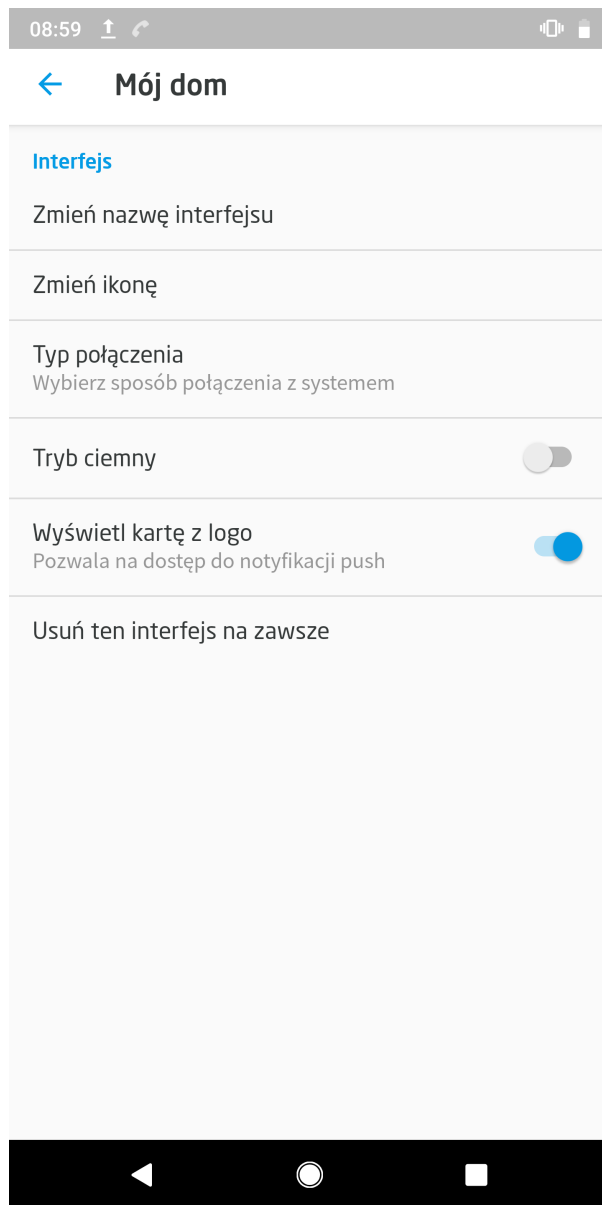
6.1. Ustawienia aplikacji

W ustawieniach dysponujemy możliwością dodania nowego interfejsu i zmianę kolejności interfejsów. Dodatkowo przycisk **Pomoc** przekierowuje na stronę wsparcia technicznego, gdzie można znaleźć przydatne informacje o systemie Grenton. **Licencje** przenoszą użytkownika na nową stronę, gdzie wypisane zostały wszystkie użyte licencje. Pole **wersja** wyświetla wersję używanej aplikacji.



6.2. Ustawienia interfejsu

Klikając w interfejs przechodzimy do ustawień interfejsu. Po wejściu w ustawienia istnieje możliwość dostosowania jego wyglądu. Możliwa jest zmiana nazwy interfejsu, zmiana ikony, wybór typu połączenia, wybór trybu jasnego bądź ciemnego, tryb wyświetlania informacji o wiadomościach a także możliwość usunięcia interfejsu na zawsze.



UWAGA! Wymagana minimalna wersja CLU do obsługi chmury:**05.03.06**

UWAGA! Aby poprawnie połączyć się przez chmurę wymagane jest połączenie z chmurą ze strony CLU. Aby to zrobić należy ustawić parametr `useCloud==true` a następnie wysłać konfigurację na CLU. Poprawne połączenie z chmurą zasygnalizuje parametr `cloudConnection==true`.

Właściwości CLU

Nazwa: Numer seryjny:

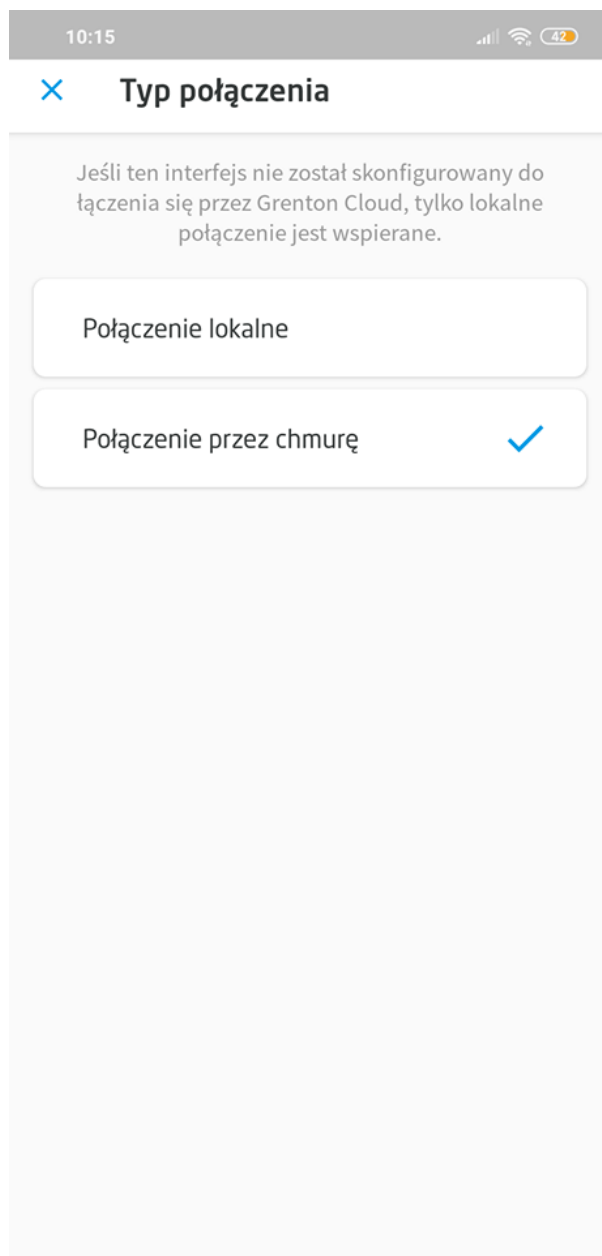
IP: FW:

Sterowanie
 Zdarzenia
 Cechy wbudowane
 Cechy użytkownika

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
Uptime	34		s	
Log	nil			
State	1			0,1,2,3,4,5,6,7,8
IsLocalPower	true		bool	
Date	2019-07-11			
Time	13:04:14			
Day	11			[1-31]
Month	7			[1-12]
Year	2019			
DayOfWeek	4			[0-6]
Hour	13		h	[0-23]
Minute	4		m	[0-59]
UnixTime	1562850254		s	
FirmwareVersion	05.03.06			
UseCloud	true	True ▾	bool	
cloudConnection	true		bool	

Auto odświeżanie

UWAGA! Jeżeli CLU połączyło się z chmurą to interfejs wysłany na aplikację mobilną będzie domyślnie używał połączenia zdalnego. Aby przejść na komunikację lokalną należy wybrać lokalny typ połączenia z systemem.



XVIII. Safe Mode

UWAGA! Funkcjonalność dostępna jest tylko dla modułów z serii **Grenton 2.0!!**

W przypadku braku komunikacji modułów z jednostką centralną CLU możliwa jest bezpośrednia komunikacja modułów w celu wywoływania prostych akcji pomiędzy wejściami a wyjściami. Funkcjonalność jest dostępna dla modułów, które mają załączony Safe Mode.

Ustawienie trybu Safe Mode dostępny jest dla następujących modułów:

- DIGITAL IN DIN
- RELAY 4HP DIN
- RELAY 2HP DIN
- DIMMER MOSFET DIN
- IO MODULE DIN 8
- IO MODULE FM

UWAGA! Funkcjonalność dostępna jest tylko dla modułów z serii **Grenton 2.0!!**

Właściwości obiektu

Nazwa: Źródło/Odbiornik:

Id: Numer seryjny:

Typ:

Nazwa cechy	Aktualna wartość	Wartość początkowa	Jednostka	Zakres
Value	0	Off <input type="button" value="v"/>	bool	0,1
VoltageType	2	Signal <input type="button" value="v"/>		0,1,2
VoltageValue	230	<input type="text" value="230"/>	V	[0-230]
Power	0		W	[0-3000]
Overload	3000	<input type="text" value="3000"/>	W	[0-3000]
SafeModeGroup	0	<input type="text" value="0"/>		[0-10000]

Auto odświeżanie

Ustawienie Safe Mode odbywa się dla każdego obiektu IN/OUT danego modułu, poprzez cechę `safeModeGroup`. Domyślna wartość `safeModeGroup` jest równa 0, co oznacza wyłączenie trybu.

UWAGA! Moduły przy pierwszym podłączeniu do magistrali mają załączony Safe Mode na wartość 1 (Default Mode). Po wykonaniu CLU Discovery i wysłaniu konfiguracji, następuje wyłączenie trybu.

W przypadku utraty połączenia modułów z CLU oraz wykrycia zdarzenia dla obiektu wejściowego (obiekt IN danego modułu) rozsyłana jest wiadomość do każdego obiektu wyjściowego (obiekt OUT danego modułu), który posiadają tę samą wartość cechy `SafeModeGroup`. W wyniku otrzymania wiadomości na obiekcie wyjściowym zostaje wywołane odpowiednia akcja przypisane do otrzymanego zdarzenia.

Dostępne akcje podczas działania Safe Mode:

- Załączenie obiektu wejściowego IN (SwitchOn) --> Załączenie danego wyjścia OUT/DIMM
- Wyłączenie obiektu wejściowego IN (SwitchOff) --> Wyłączenie danego wyjścia OUT/DIMM
- Krótka zmiana stanu obiektu wejściowego IN (Click) --> Zmiana stanu na przeciwny OUT/DIMM

UWAGA! Zdarzenia oraz wywoływane akcje w przypadku Safe Mode są ustawione statycznie i nie ma możliwości ich zmiany

W przypadku gdy do danego obiektu wartość cechy `SafeModeGroup` ustawiona jest na wartość 1, działa on w trybie **Default Mode**. Jest to specjalny tryb działania defaultowo ustawiany dla każdego obiektu:

- moduł wejściowy (Digital IN) - steruje wszystkimi modułami wyjściowymi (RELAY 4HP DIN, RELAY 2HP DIN, DIMMER MOSFET DIN) w sieci TFBUS będące również w Default Mode, przykładowo:
 - Digital IN1--> Relay 4HP OUT1 | Relay 2HP OUT1 | DIMMER MOSFET
 - Digital IN2--> Relay 4HP OUT2 | Relay 2HP OUT2

- Digital IN3--> Relay 4HP OUT3
- Digital IN4--> Relay 4HP OUT4
- moduł posiadający wejścia oraz wyjścia (IO MODULE DIN 8,IO MODULE FM) - steruje własnymi kanałami o adekwatnym numerze (IN1->OUT1, IN2->OUT2, itd), przykładowo dla IO MODULE FM:
 - IO MODULE FM IN1 --> IO MODULE FM OUT1
 - IO MODULE FM IN2 --> IO MODULE FM OUT2

W momencie, gdy komunikacja pomiędzy CLU a modułami zostanie przywrócona, wartość cech Value danych obiektów zostaje zaktualizowana do rzeczywistej wartości wejść / wyjść (na wartość zmienioną podczas działania Safe Mode'a) a moduły wykonują działania zgodnie z zaprogramowaną logiką.

-
1. W zależności od rodzaju używanego routera, jego interfejs może się różnić od ogólnej instrukcji konfiguracji portów.[↪](#)
 2. Jest to domyślny port dla potrzeb obsługi strumienia kamery *rtsp*.[↪](#)
 3. Jego adres IP można znaleźć na liście aktualnie podłączonych do sieci urządzeń w interfejsie routera.[↪](#)
 4. W zależności od tego jakiego rodzaju urządzenie jest w użyciu, jego konfiguracja może się różnić od podanej w instrukcji.[↪](#)
 5. Oprócz ustawień połączenia w tej samej sekcji istnieje możliwość zaznaczenia pola wyboru, decydującego o używaniu trybu głośnomówiącego po odebraniu połączenia.[↪](#)
 6. Gdzie X i Y oznaczają nazwy CLU.[↪](#)
 7. W rozumieniu instrukcji słowo składa się z dwóch bajtów.[↪](#)