

# Integracja z multiroom NuVo

W tym tutorialu przedstawimy jak wykorzystać moduły **DOMIQ** do integracji z dowolnymi urządzeniami wykorzystującym protokół transmisji szeregową RS-232.

Moduł **Serial-2SG** umożliwia komunikację szeregową przy użyciu protokołu RS-232. Obsługiwane są prędkości: 9600, 19200, 38400 oraz 57600 bps oraz formaty ramek: 8N1, 8N2, 8E1, 8O1. Do modułu **DOMIQ/Base** możliwe jest podłączenie maksymalnie dwóch modułów **Serial-2SG** jednocześnie.

Domyślnie moduły **Serial-2SG** są sprzedawane w wersji z adresem 1. W przypadku, kiedy planowane jest podłączenie dwóch modułów **Serial-2SG** do modułu **Base** należy drugi moduł zamówić w wersji z adresem 2.

W dalszej części tego tutoriala przedstawimy proces integracji systemu **DOMIQ** z urządzeniami wykorzystującymi protokół transmisyjny RS-232 na przykładzie systemu multiroom firmy **NuVo**. Czytając tę instrukcję dowiesz się jak:

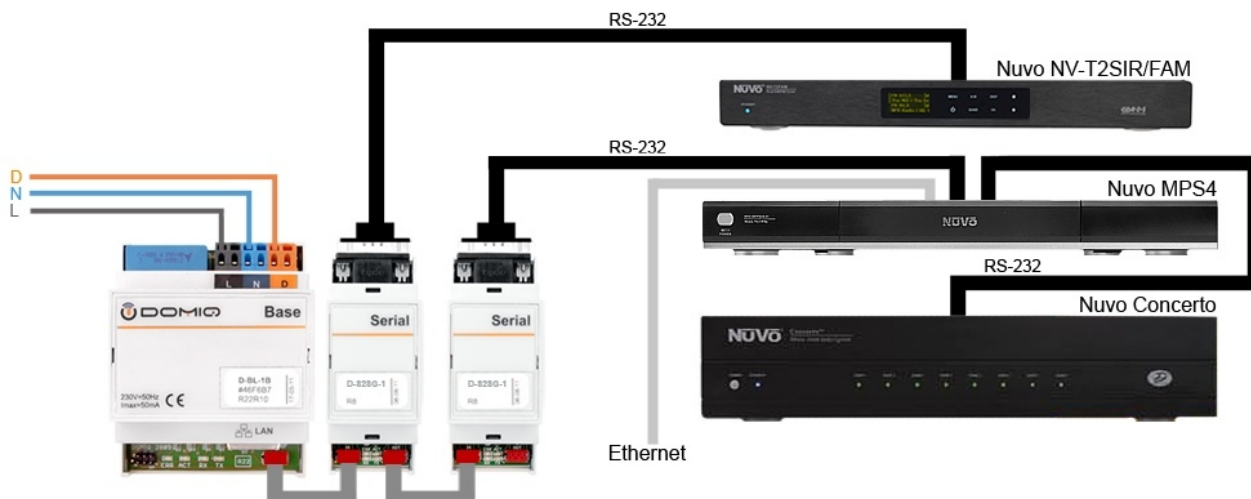
- Wysyłać i odbierać komunikaty w protokole RS-232 przy użyciu modułów **DOMIQ**.
- Zintegrować multiroom z systemem automatyki domowej.
- Przetwarzać dane odczytywane przez moduł **Serial-2SG**.
- Sterować systemem multiroom z wizualizacji **Display**.

## 1. Multiroom NuVo

System multiroom to zaawansowany system audio umożliwiający wielostrefowe odwarzanie dźwięku ze stacji radiowych, plików mp3. Każda strefa może odtwarzać dźwięk z dowolnego dostępnego źródła audio. Zestaw udostępniony nam do testów składał się z następujących urządzeń:

- **NuVo Concerto** - wzmacniacz muzyczny. Obsługa 6 stref.
- **NuVo NV-T2SIR/FAM** - urządzenie z wbudowanymi dwoma niezależnymi tunerami radiowymi. Umożliwia słuchanie stacji radiowych jako dwa niezależne strumienie audio.
- **NuVo MPS4** - serwer plików MP3. Posiada wbudowane niezależne 4 źródła sygnału audio. Jeżeli urządzenie posiada aktywne połączenie z Internetem możliwe jest również odtwarzanie internetowych stacji radiowych. Każde ze źródeł może odtwarzać jednocześnie inne pliki muzyczne lub internetowe stacje radiowe. Jeżeli serwer MPS4 jest podłączony do sieci Ethernet możliwe jest sterowanie nim przy użyciu protokołu TCP/IP.
- **NuVo NV-I8GCP** - naścienne panele dotykowe z wyświetlaczem umożliwiające sterowanie urządzeniami (zmiana ustawień, wybór źródła dźwięku, sterowanie głośnością, wybór utworów itp).

## 2. Połączenia



Zdjęcia urządzeń NuVo zaczerpnięte z [www.NuVotechnologies.com](http://www.NuVotechnologies.com).

Połączenia oznaczone jako "RS-232" należy wykonać standardowym kablem zakończonym wtykami DB-9. Połączenia sygnałów audio pomiędzy urządzeniami **NuVo** należy wykonać zgodnie z odpowiednimi instrukcjami producenta.

## 3. Konfiguracja modułów DOMIQ

### 3.1. Konfiguracja DOMIQ/Serial-2SG

Po podłączeniu modułów **DOMIQ** należy skonfigurować moduł **Serial-2SG** poprzez ustalenie parametrów transmisji RS-232. Wykorzystywane jest do tego polecenie o następującej składni:

```
LC.SER.config.<adres modułu>=<prędkość> <format>
```

na przykład:

```
LC.SER.config.1=38400 8N1
```

Obsługiwane prędkości transmisji danych: 9600, 19200, 38400 oraz 57600 bps oraz formaty ramek danych: 8N1, 8N2, 8E1, 8O1.

Prędkość transmisji i format danych musi zostać ustawiony przy każdym uruchomieniu modułu **Base**. W tym celu należy wykorzystać zakładkę **Logika** i umieścić tam fragment kodu o następującej składni:

```
command("LC.SER.config.<adres>=<prędkość> <format>")
```

uzupełniając polecenie odpowiednimi parametrami. Polecenie to będzie wysyłane przy każdym uruchomieniu modułu **Base** lub modułu **Serial-2SG**.

### 3.2. Polecenia sterujące

System multiroom NuVo jest w pełni konfigurowalny. Dysponując opisami protokołów komunikacyjnych urządzeń NuVo oraz modułami **DOMIQ** można sterować wszystkimi dostępnymi funkcjami oferowanymi przez system multiroom takimi jak: włączanie/wyłączanie poszczególnych stref, zmiana źródła sygnału w strefach, sterowanie głośnością, zmiana nastaw tunera radiowego, wyświetlanie informacji RDS itd. oraz integrować system multiroom z systemem automatyki domowej.

Wykorzystywane są do tego dwa polecenia:

- **Wysyłanie poleceń:** LC.SER.line.<adres modułu>=<polecenie>

Przykład: LC.SER.line.1=\*Z1ON - włączenie strefy pierwszej.

- **Odbiór poleceń:** LE.SER.line.<adres modułu>=<polecenie>

Jeżeli serwer muzyczny NuVo MPS4 jest podłączony do sieci Ethernet, wówczas jest możliwość sterowania nim przy użyciu protokołu **TCP/IP**. Wykorzystywane jest do tego polecenie o następującej składni:

```
C.TCP.send.<adres IP serwera muzycznego:port>=<polecenie>
```

Przykład: C.TCP.send.192.168.10.182:5004=PlayPause. - symulowanie naciśnięcia przycisku Play/Pause.

Lista dostępnych poleceń zawarta jest w opisach protokołów komunikacyjnych urządzeń NuVo.

Polecenia sterujące mogą być wysyłane jako:

- Wynik działania zdarzenia warunkowego.
- Wynik działania timera.
- Efekt wciśnięcia przycisku.
- Element funkcji logicznej.

## 4. Integracja

Dzięki modułowi **DOMIQ/Serial 2SG** możliwa jest pełna integracja systemu multiroom z systemem automatyki budynkowej. Mamy wówczas do dyspozycji cały potencjał automatyki domowej do sterowania systemem multiroom i na odwrót.

### 4.1. Budzik muzyczny

Wykorzystanie timerów i logiki w połączeniu z systemem multiroom pozwala na zdefiniowanie inteligentnego budzika muzycznego. System automatyki uruchomi system multiroom w ustalonych strefach o określonej godzinie, puści ulubioną płytę lub stację radiową oraz ustawi głośność na określoną wartość. Definicja inteligentnego budzika muzycznego przebiega dwuetapowo. Pierwszym etapem jest zdefiniowanie funkcji logicznych. W drugim etapie definiowane są timery.

#### 4.1.1. Definiowanie funkcji logicznych

1. Wybierz zakładkę **Logika** w konfiguratorze modułu **Base**.
2. Wprowadź kod bazując na kodzie na przedstawionym poniżej:

```
function budzik()  
    command("LC.SER.line.1=*Z1ON\r*Z1SRC5\r*Z1VOL50")  
    command("LC.SER.line.2=*T'A'FM98.8")  
    command("LC.SER.line.1=*Z2ON\r*Z2SRC6\r*Z2VOL45")  
    command("LC.SER.line.2=*T'B'FM103.0")  
    command("LC.SER.line.1=*Z3ON\r*Z3SRC1\r*Z3VOL45")  
    command("C.TCP.send.<adres IP >=PlayPlaylist <nazwa> True")  
end
```



soby, sekcja **Skrypty**) skrypt **parser.lua**, który jest dostępny do pobrania jako załącznik do niniejszego tutoriala. Kod parsera jest uniwersalny i może być wykorzystywany do przetwarzania dowolnych komunikatów.

Parser porównuje otrzymane komunikaty z wzorcami dopasowania. Jeżeli komunikat pokrywa się ze wzorcem, wywołuje wówczas jedną ze zdefiniowanych funkcji, która przetwarza odebrane dane.

```
import 'parser'
p = parser()
p:add("ADD(%d)", function(n) print(n+1) end)
p:add("SUB(%d)", function(n) print(n-1) end)
p:parse("ADD1")
p:parse("SUB2")
```

Wykonanie powyższego przykładu spowoduje wyświetlenie:

```
2
1
```

Bardziej praktycznym sposobem użycia danych jest zapisanie danych w stanie **Base**, co pozwala na ich wyświetlenie na ekranie wizualizacji **Display** lub w menu **Remote**.

Szczegółowy opis wzorców dopasowania znajduje się w rozdziale 5.4.1 dokumentu "Lua Reference Manual", dostępnego pod adresem <http://www.lua.org/manual/5.1/>

W dalszej części tego rozdziału przedstawimy przykładowe zastosowanie parsera do wyświetlania informacji RDS z tunera radiowego **NuVo** na wizualizacjach **Display**.

## 5.1. Parsowanie komunikatów RDS

Komunikaty radiowe RDS są przekazywane za pomocą komendy **#SsDISPLINE**, gdzie **s** to numer źródła sygnału audio. Aby wyświetlać informacje RDS należy dodać do zakładki **Logika** następujący kod programu:

```
import 'parser'

function nuvo(tab)
    local dev = {}
    local zones = {}
    local tun = assert(tab.tun)
    local amp = assert(tab.amp)
    local pamp = parser()
    local ptun = parser()

    -- Parsowanie komunikatów z tunera
    function tun:onchange(data)
        print("TUN: "..data)
        ptun:parse(data)
    end
end
```

```
-- Parsowanie komunikatów ze wzmacniacza
function amp:onchange(data)
    print("AMP: "..data)
    pamp:parse(data)
end

-- Parsowanie komunikatów wyświetlania
pamp:add([[#S(%d)DISPLINE(%d),"([^\"]+)"]],function(s,l,txt)
    local sid = tonumber(s)
    for k,v in pairs(zones) do
        if v == sid then set("NUVO.line"..k.." "..l,txt) end
    end
end)

-- Parsowanie komunikatów o statusie poszczególnych stref
pamp:add([[#Z(%d),ON,SRC(%d),VOL(%d+),DND(%d),LOCK(%d)]],
    function(z,src,vol,dnd,lock)
        local prefix="NUVO"..z..".."
        set(prefix.."vol",vol)
        set(prefix.."src",src)
        set(prefix.."act",1)
        zones[tonumber(z)] = tonumber(src)
        print ("SRC"..src)
    end)

-- To samo, ale jak strefa jest wyłączona
pamp:add([[#Z(%d),OFF]],function(z)
    set("NUVO"..z.."..".."act",0)
end)
--
-- TU WSTAW KOD ŹRÓDŁOWY PRZEDSTAWIONY W ROZDZIALE 6.
--
return dev
end

-- Inicjalizacja powyższego sterownika.
-- Jako parametry przekazujemy obiekty od dwu modułów SG
n = nuvo {tun = use 'SER.line.1', amp = use 'SER.line.2'}
```

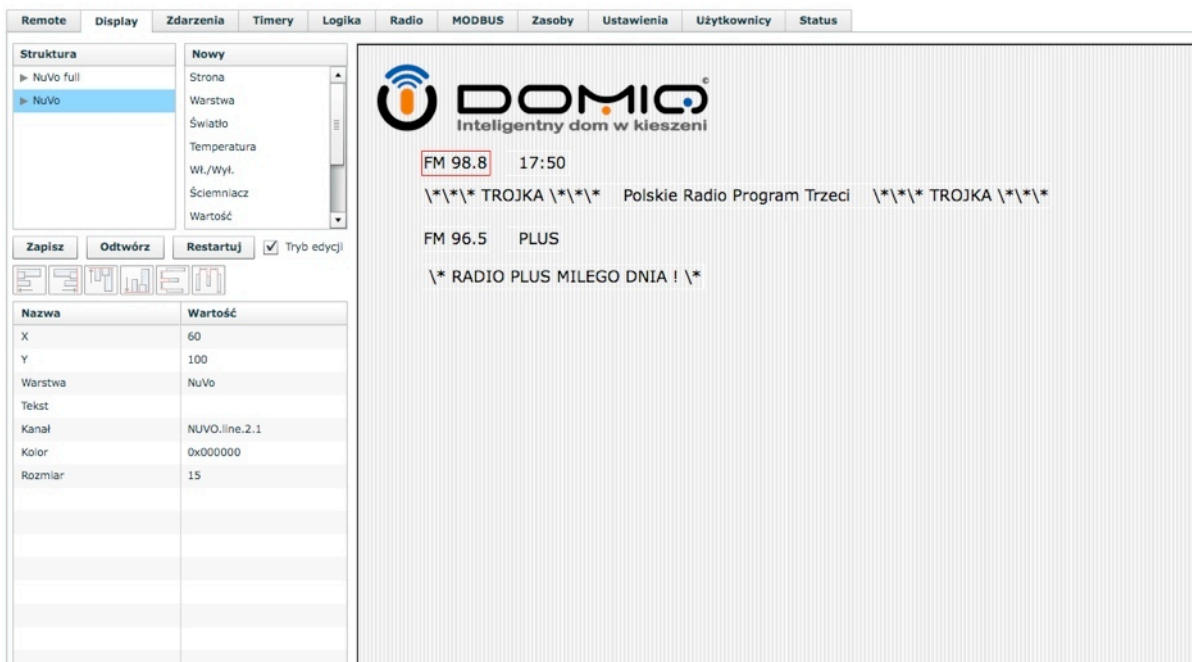
Kilka słów objaśnienia do przedstawionego kodu:

Powyższy kod pokazuje jeden z typowych tzw "wzorców projektowych" używanych w programowaniu w Lua, tzw. tworzenie obiektu. Idea tego wzorca polega na "schowaniu" przed innym kodem wszystkich zmiennych definiowanych jako local w treści funkcji, co pozwala na tworzenie wielu instancji kodu (np. do obsługi kilku systemów NuVo).

Po wprowadzeniu powyższego kodu programu można przystąpić do tworzenia strony wizualizacji, gdzie będą wyświetlane komunikaty RDS:

1. Przejdź do zakładki **Display**.
2. Dodaj nową **Stronę**, nadaj jej **Identyfikator** oraz wybierz tło wizualizacji.
3. Dodaj nowy element typu **Tekst**.
4. W polu **Kanał** wpisz: `NUVO.line.<numer strefy muzycznej>.<numer linii>`.  
Przykład: `NUVO.line.2.3` - wyświetlenie linii 3 ze strefy nr 2. Ilość linii informacji oraz numeracja linii różni się w zależności od stacji radiowej. System NuVo umożliwia wyświetlanie maksymalnie 6 linii informacji. W przypadku stacji radiowych najczęściej wykorzystywane są linie od 1 do 4.
5. Powtórz punkty nr 3 i 4. w zależności od liczby stref oraz linii informacji.

W efekcie otrzymamy:



## 6. Sterowanie NuVo przy użyciu wizualizacji

W tym rozdziale przedstawimy, jak sterować systemem multiroom przy użyciu wizualizacji (panel **Display** lub aplikacji **Remote**). Zrealizowana zostanie następująca funkcjonalność:

- Przyciski do załączania/wyłączania poszczególnych stref.
- Przyciski wyciszania poszczególnych stref
- Strojenie tunera radiowego.
- Zmiana głośności.
- Centralny przycisk wyłączający wszystkie strefy.

- Zmiana źródła sygnału w poszczególnych strefach.

Powyższe funkcje uzupełnią opisane w rozdziale 5. wyświetlanie informacji RDS. Definiowanie sterowania urządzeniami NuVo z poziomu wizualizacji przebiega dwuetapowo. W pierwszym etapie utworzone zostaną reguły logiczne. W drugim natomiast elementy wizualizacji. Zaprezentowana numeracja stref i źródeł sygnału audio może różnić się w zależności od połączenia urządzeń NuVo.

### 6.1. Definiowanie reguł logicznych

Aby zrealizować założoną we wstępie tego rozdziału funkcjonalność, należy rozszerzyć kod źródłowy z zakładki **Logika**. Przedstawiony poniżej kod programu należy wkleić w zakładce **Logika** w miejsce komentarza „-- TU WSTAW KOD ŹRÓDŁOWY PRZEDSTAWIONY W ROZDZIALE 6”.

```
-- Włączanie stref
function dev:on(z)
  amp:send("*Z"..z.."ON")
end

-- Wyłączanie stref
function dev:off(z)
  if z == nil then
    for i=1,6 do
      amp:send("*Z"..i.."OFF")
    end
  else
    amp:send("*Z"..z.."OFF")
  end
end

-- Zwiększanie glosnosci
function dev:volup(z)
  amp:send("*Z"..z.."VOL+")
end

-- Zmniejszanie glosnosci
function dev:voldown(z)
  amp:send("*Z"..z.."VOL-")
end

-- MUTE
function dev:mute(z)
```



```
amp:send ("*Z"..z.."MUTE")
end

-- Zmiana źródła sygnału
function dev:src(z)
  amp:send ("*Z"..z.."SRC+")
end

-- Strojenie tunera radiowego
-- Funkcja pomocnicza
local function tuner(ch, fun)
  tun:send ("*T"..ch.." "..fun )
end

function dev:seek(ch, dir)
  tuner(ch, "SEEK"..dir)
end
```

*Kilka słów objaśnienia do przedstawionego kodu:*

Powyższy kod źródłowy to zbiór funkcji realizujących wybrane działania w systemie multi-room **NuVo**. Pierwsze dwie funkcje to włączanie i wyłączenie wybranych stref oraz funkcja wyłącznika centralnego. Kolejne trzy funkcje to sterowanie głośnością w wybranych strefach oraz wyciszanie poszczególnych stref. Funkcja nr 6 odpowiada za zmianę źródła sygnału audio w strefach. Ostatnie dwie funkcje odpowiadają za strojenie tunera radiowego. Strojenie odbywa się na zasadzie przeszukiwania częstotliwości od stacji do stacji.

Przedstawiony kod źródłowy można swobodnie poszerzać o inne funkcje oferowane przez urządzenia **NuVo**.

## 6.2. Tworzenie wizualizacji

Do zwizualizowania nowej funkcjonalności wykorzystana zostanie strona utworzona w rozdziale 5.

Wszystkie utworzone przyciski sterujące są uniwersalne i mogą być wykorzystane do sterowania poszczególnymi strefami. Należy jedynie pamiętać o zmianie numeru strefy we wpisywanych poleceniach. Procedury przedstawione w dalszej części rozdziału (z wyjątkiem 6.2.5) sterują strefą nr 1.

### 6.2.1. Załączanie/wyłączanie stref

1. Dodaj nowy element **Wł./Wył.**
2. W polu **Kanał** wpisz: NUVO.1.act.
3. W polu **Włącz** wpisz: LOGIC=n:on(1).
4. W polu **Wyłącz** wpisz: LOGIC=n:off(1).

5. Aby zwizualizować stan strefy w formie lampki, dodaj element **Światło**. Z listy tematów wybierz **Green**. W polu kanał wpisz `NUVO.1.act`. Element umieść w narożniku przycisku Wł./Wył.

### 6.2.2. Wyciszanie strefy

1. Dodaj nowy **Przycisk**.
2. Wpisz opis w polu **Etykieta**, np. Mute.
3. W polu **Polecenie** wpisz: `LOGIC=n:mute(1)`.

### 6.2.3. Strojenie tunera

1. Dodaj nowy **Przycisk**.
2. Wpisz opis w polu **Etykieta**, np. Seek+.
3. W komórce **Polecenie** wpisz: `LOGIC=n:seek('A','+')`.
4. Powtórz punkty od 1 do 3 zmieniając opis w **Etykiecie** na Seek- oraz zawartość komórki **Polecenie** na `LOGIC=n:seek('A','-')`.

### 6.2.4. Zmiana głośności

1. Dodaj nowy **Przycisk**.
2. Wpisz opis w polu **Etykieta**, np. VOL+.
3. W komórce **Polecenie** wpisz: `LOGIC=n:volup(1)`.
4. Powtórz punkty od 1 do 3 zmieniając opis w **Etykiecie** na VOL- oraz zawartość komórki **Polecenie** na `LOGIC=n:volup(1)`.

Punkty od 6.2.1 od 6.2.4 powtórz dla pozostałych stref.

### 6.2.5. Centralny przycisk włącz/wyłącz

1. Dodaj nowy **Przycisk**.
2. Wpisz opis w polu **Etykieta**, np. ALL OFF.
3. W komórce **Polecenie** wpisz: `LOGIC=n:off()`.

Przykładowy efekt końcowy:



ALL OFF

Strefa 1

ON Seek A+ VOL+ SRC+  
5

MUTE Seek A- VOL-

FM 98.8 POLSKIE RADIO  
\*\*\* TROJKA \*\*\* Polskie Radio Program Trzeci \*\*\*TROJKA \*\*\*

Strefa 2

ON Seek B+ VOL+ SRC+  
6

MUTE Seek B- VOL-

FM 96.5 PLUS  
\* RADIO PLUS MILEGO DNIA ! \*