

## SIECIOWY ROZPROSZONY SYSTEM POMIAROWY W LABORATORIUM DYDAKTYCZNYM

Beata PAŁCZYŃSKA<sup>1</sup>, Krystyna Maria NOGA<sup>2</sup>

1. Katedra Telekomunikacji Morskiej, Wydział Elektryczny, Akademia Morska w Gdyni  
tel.: 58 5586552 e-mail: palbeata@am.gdynia.pl
2. Katedra Automatyki Okrętowej, Wydział Elektryczny, Akademia Morska w Gdyni  
tel.: 58 5586458 e-mail: k.noga@we.am.gdynia.pl

**Streszczenie:** Zaprezentowano możliwości wykorzystania do celów dydaktycznych specjalistycznego narzędzia programistycznego, zdecydowanie upraszczającego oprogramowanie rozproszonego systemu pomiarowego. Przedstawiono system pomiarowy, zaprojektowany w graficznym środowisku programistycznym LabVIEW. Opisano stanowisko laboratoryjne, składające się z kilku węzłów z dostępem do sieci komputerowej, w którym funkcje procesu pomiaru i sterowania oraz kontrola pomiaru są wykonywane jako rozproszone w różnych węzłach systemu pomiarowego. Scharakteryzowano wykorzystane w rozwiązaniu elementy programowe środowiska LabVIEW; projekt i sieciowe zmienne współdzielone. Przedstawiono organizację zaprojektowanego rozproszonego systemu pomiarowego, zawarto uwagi dotyczące konfigurowania i uruchamiania systemu.

**Słowa kluczowe:** rozproszone systemy pomiarowe, technologia inteligencji rozproszonej.

### 1. WPROWADZENIE

Jako rozwiązanie zagadnienia zarządzania danymi we współczesnych rozproszonych systemach pomiarowych (RSP) najpowszechniej stosuje się architekturę typu klient – serwer [1, 2, 3]. Daje to ogromne możliwości nie tylko przestrzennego, ale także funkcjonalnego rozproszenia takiego systemu. Taka struktura aplikacji pociąga za sobą jednak konieczność nietypowego podejścia do oprogramowania systemu pomiarowego. Powinno ono uwzględniać różne uwarunkowania związane m.in. z jednoczesnym wykorzystaniem procesorów różnego typu, zapewniać efektywne współdzielenie danych przez te procesory, koordynację wszystkich węzłów rozproszonych jako jednego systemu, w tym regulację czasową i synchronizację pomiędzy tymi węzłami, a także integrować różne typy wejść/wyjść. W środowisku programistycznym LabVIEW (np. <https://www.youtube.com/user/Labview>) oprogramowanie rozproszonego systemu pomiarowe umożliwiają takie elementy programowe jak zmienne współdzielone i LabVIEW projekt [4, 5]. W artykule przedstawiono możliwości zastosowania tego oprogramowania w laboratorium dydaktycznym w Katedrze Telekomunikacji Morskiej w Akademii Morskiej w Gdyni. Celem ćwiczenia laboratoryjnego jest zaprezentowanie całego procesu projektowania i uruchamiania aplikacji RSP w środowisku programowym LabVIEW, wspomagającym projektowanie systemów pomiarowych. Zaprezentowano

możliwości komunikacji pomiędzy węzłami rozproszonego systemu pomiarowego, wykorzystującego cztery aktywne węzły. Dwa z nich to węzły pomiarowe, jeden to węzeł odbiorczy, zaś ostatni, najważniejszy, to węzeł sterowania całym systemem. Struktura RSP jest wstępnie przygotowana przez prowadzącego laboratorium, w trakcie trwania ćwiczeń jest modyfikowana w zależności od indywidualnych aplikacji uruchomionych w poszczególnych węzłach. Kompletnie zaprojektowany RSP jest wspólnie uruchamiany oraz demonstrowane jest jego działanie.

### 2. NARZĘDZIE DO OPROGRAMOWANIA SIECIOWEGO SYSTEMU POMIAROWEGO

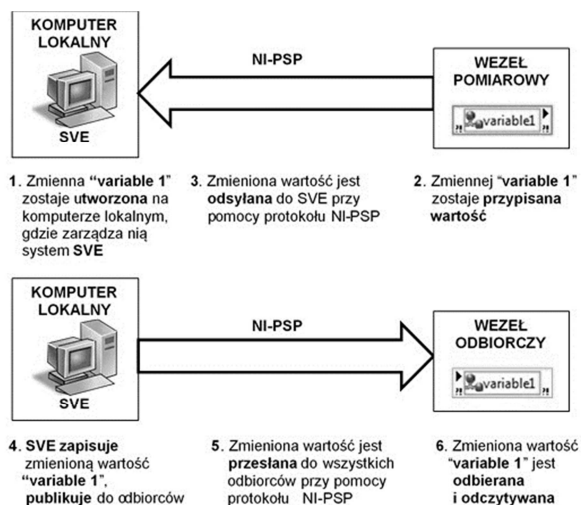
W środowisku LabVIEW projektant ma do dyspozycji zestaw narzędzi programowych, które umożliwiają efektywne projektowanie, uruchamianie i sterowanie systemami rozproszonymi. W prezentowanym systemie pomiarowym wykorzystano dwa elementy programowe tego typu: sieciowe zmienne współdzielone (*ang. Network Shared Variables*) oraz projekt (*ang. LabVIEW Project*).

#### 2.1. Sieciowe zmienne współdzielone

Zmienne współdzielone są programowo konfigurowalnymi elementami, dzięki którym użytkownik może przysyłać dane pomiędzy przyrządami wirtualnymi, w aplikacjach nawet na różnych komputerach. Wykorzystanie ich wprowadza znaczne uproszczenie podczas tworzenia kodu programu.

W LabVIEW użytkownik ma do wyboru trzy rodzaje zmiennych współdzielonych: jednoprosesową, sieciową i taktowaną czasem. Zmienna sieciowa jest wykorzystywana do przesyłania danych przez sieć. Komunikacja sieciowa jest całkowicie realizowana przez zmienną bez żadnej ingerencji użytkownika. Protokołem wykorzystywanym przez zmienne współdzielone sieciowe do wysyłania i odbioru danych przez sieć jest stworzony przez National Instruments protokół NI-PSP (*ang. Publish-Subscribe Protocol*), na bazie szybkiego protokołu bezpołączeniowego UDP (*ang. User Datagram Protocol*). W celu poprawnego działania programu zmienne sieciowe należy przypisać do Systemu Zarządzania Zmiennymi Współdzielonymi SVE (*ang. Shared Variable Engine*), który konfigurowany i uruchamiany przy starcie LabVIEW umożliwia przesyłanie zmiennych przez Internet. SVE rozmieszcza i przechowuje

zmienne, a następnie publikuje je w taki sposób, że odbiorca ma możliwość uaktualniania ich wartości (rys. 1). Sieciowa zmienna współdzielona identyfikuje się w programie poprzez ścieżkę sieciową, która zawiera nazwę komputera, nazwę biblioteki projektu i nazwę zmiennej.



Rys. 1. Proces przekazywania zmienionej wartości sieciowej zmiennej współdzielonej w systemie [4]

Aby umożliwić wykorzystanie w systemie sieciowych zmiennych współdzielonych SVE musi działać przynajmniej na jednym węźle systemu rozproszonego (najlepiej na tym, który zawsze pracuje on-line, w prezentowanym systemie w węźle kontrolera).

## 2.2. LabVIEW Projekt

Administrowanie wszystkimi plikami, związanymi z aplikacją rozproszoną, np. plikami źródłowymi przyrządów wirtualnych, dokumentacją projektu, plikami pomocniczymi, kodem zewnętrznym, plikami danych i konfiguracjami sprzętowymi, wykonywane jest za pomocą LabVIEW Projekt. Wszystkie zmienne współdzielone są częścią biblioteki projektu. Każda aplikacja w węźle to osobny projekt. Okno projektu (rys. 2) umożliwia edytowanie projektu i grupowanie plików połączonych w wirtualne foldery, które mają tylko charakter logiczny i nie posiadają swojego odbicia w strukturze plików na dysku.

## 3. ORGANIZACJA ROZPROSZONEGO SYSTEMU POMIAROWEGO

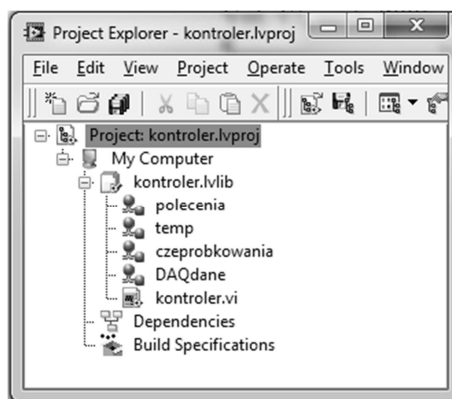
Zaprojektowany RSP składa się z czterech węzłów z dostępem do sieci, które są kontrolowane i sterowane z głównego komputera (SERWERA), na którym działa główny program sterujący *kontroler.vi* oraz uruchomione jest oprogramowanie SVE. Pozostałe funkcje procesu pomiaru, tzn. zbieranie danych pomiarowych oraz prezentacja wyników, zostały rozproszone i są wykonywane w różnych węzłach systemu pomiarowego (KLIENCI). Dwa przyrządy wirtualne VIs (*ang. virtual instruments*) zainstalowane są w węzłach pełniących funkcję węzłów pomiarowych *DAQ.vi* oraz *metex.vi*, zaś trzeci program *Uzytkownik.vi* pracuje w węźle odbiorczym. Wszystkie komputery, na których znajdują się wyżej wymienione przyrządy, wchodzą w skład sieci komputerowej z zaimplementowanym protokołem TCP/IP. W związku z tym, przed uruchomieniem systemu w każdym węźle RSP należy aktywować komunikację VI Servera z wykorzystaniem

protokołu TCP/IP oraz dopisać na odpowiednich komputerach systemu wszystkie numery IP, które mogą mieć dostęp do przyrządów wirtualnych umieszczonych na danej maszynie.

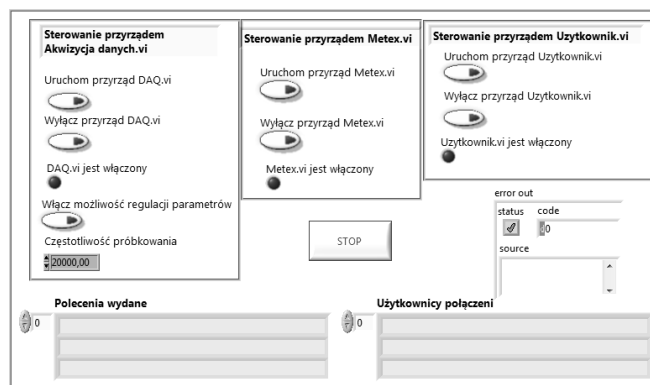
### 3.1. Przyrząd wirtualny sterujący RSP

Biblioteka projektu *kontroler.lvlib* zawiera przyrząd wirtualny zainstalowany w węźle oraz wszystkie sieciowe zmienne współdzielone, które występują we wszystkich węzłach RSP (rys. 2). Najważniejszym programem jest *kontroler.vi*, który działa na głównym komputerze. Z jego poziomu można kontrolować przebieg wykonywania pomiarów w węzłach pomiarowych oraz nadzorować stan pracy wszystkich węzłów systemu. Istnieje możliwość sterowania poszczególnymi węzłami rozproszonymi poprzez włączanie bądź wyłączanie poszczególnych przyrządów wirtualnych w węzłach oraz zdalną regulację ich parametrów. Na panelu czołowym programu (rys. 3) można wyodrębnić trzy obszary, związane z poszczególnymi węzłami RSP, sterowanie akwizycją danych z karty DAQ z możliwością ustawiania częstotliwości próbkowania karty, sterowanie multimetrem, mierzącym temperaturę w otoczeniu węzła oraz udostępnianie wyników pomiaru, wykonywanych w węzłach pomiarowych użytkownikowi. Ponadto, po uruchomieniu programu na panelu przyrządu pojawia się lista połączonych użytkowników w RSP oraz zapisywane są wszystkie polecenia wydawane z kontrolera.

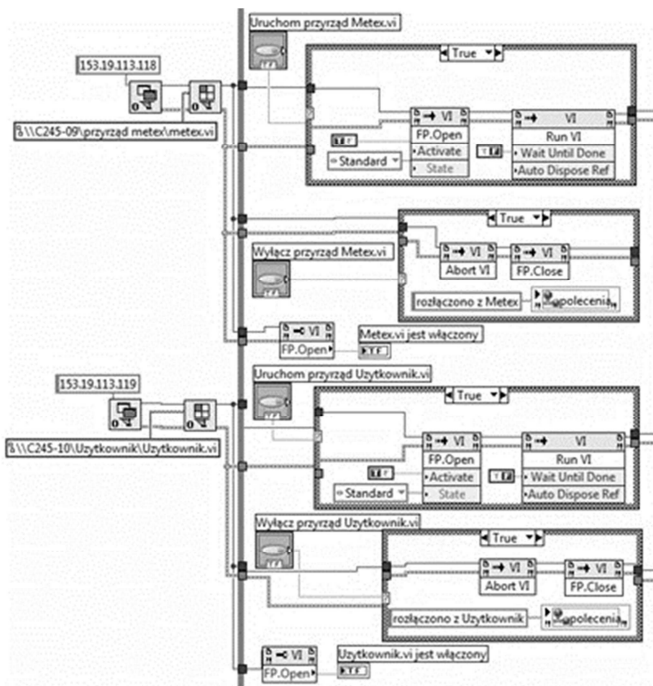
W kodzie źródłowym przyrządu *kontroler.vi* zawarte są informacje dotyczące adresów IP oraz ścieżki dostępu do wszystkich przyrządów wirtualnych pracujących w RSP (rys. 4). W prezentowanym systemie, z poziomu kontrolera można w bardzo prosty sposób zdalnie ustawiać częstotliwość próbkowania karty DAQ, pracującej w węźle pomiarowym (rys. 5).



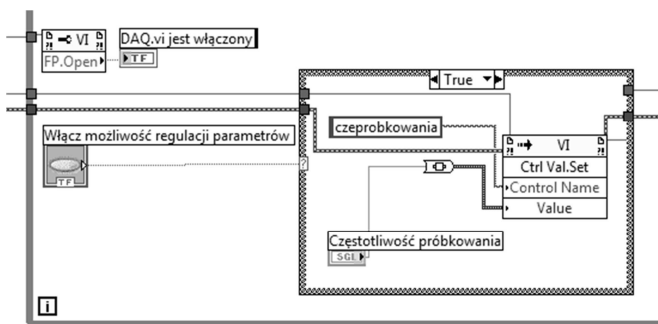
Rys. 2. Okno projektu *kontroler.lvproj*



Rys. 3. Wygląd panelu czołowego wirtualnego przyrządu pomiarowego *kontroler.vi*



Rys. 4. Część diagramu przyrządu *kontroler.vi* odpowiedzialna za możliwość uruchamiania zdalnych przyrządów wirtualnych *metex.vi* i *uzytkownik.vi*



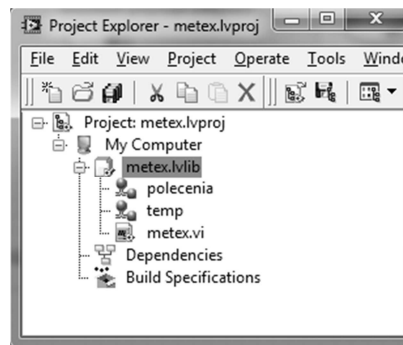
Rys. 5. Część diagramu przyrządu *kontroler.vi* odpowiedzialna za możliwość regulacji parametrów w węzle pomiarowym

### 3.2. Przyrząd wirtualny *Metex.vi*

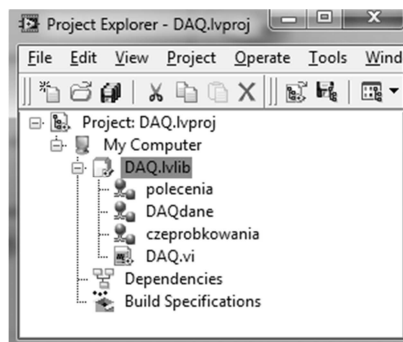
Projekt *metex.lvproj* obsługuje węzeł pomiarowy, w którym za pomocą wielofunkcyjnego multimetru METEX M3640D, połączonego z komputerem klasy PC interfejsem szeregowym RS-232C, wykonywany jest pomiar temperatury (rys. 6). W bibliotece zdefiniowane zostały dwie zmienne współdzielone: zmienna *polecenia*, do której zapisywane są komendy przesyłane z kontrolera oraz zmienna *temp*, zawierająca dane pomiarowe.

### 3.3. Przyrząd wirtualny *DAQ.vi*

W drugim węzle pomiarowym, opartym na karcie pomiarowej DAQ NI PCI 6281, za pomocą przyrządu wirtualnego *DAQ.vi* pobierane są dane pomiarowe z generatora m.cz. Ponadto, z poziomu kontrolera możliwe jest zdalne ustawianie częstotliwości próbkowania karty pomiarowej. W projekcie *DAQ.lvproj* zdefiniowane zostały trzy sieciowe zmienne współdzielone: *polecenia*, *czeprobkowania* i *DAQdane*, związane, odpowiednio z komendami przesyłanymi z kontrolera, częstotliwością próbkowania karty pomiarowej i danymi rejestrowanymi przez kartę DAQ (rys. 7).



Rys. 6. Okno projektu *metex.lvproj*

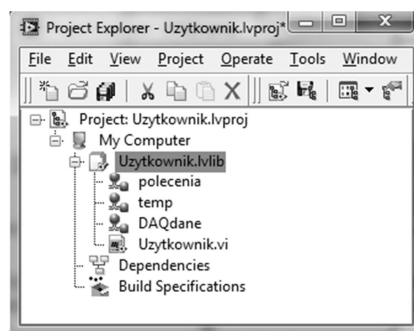


Rys. 7. Okno projektu *DAQ.lvproj*

W związku z tym, że dane pomiarowe są przesyłane za pomocą zmiennej współdzielonej, w celu poprawnego działania programu wprowadzono ograniczenia związane z częstotliwością próbkowania karty pomiarowej. Przy dużych częstotliwościach próbkowania istnieje niebezpieczeństwo utraty (pominięcia) niektórych danych.

### 3.4. Przyrząd wirtualny *Uzytkownik.vi*

Kontroler może udostępniać Użytkownikowi dane z obu węzłów pomiarowych RSP. W związku z tym, w projekcie *uzytkownik.lvproj* oprócz sieciowej zmiennej współdzielonej *polecenia* muszą zostać skonfigurowane obie zmienne, w których zapisane zostały dane pomiarowe (rys. 8).

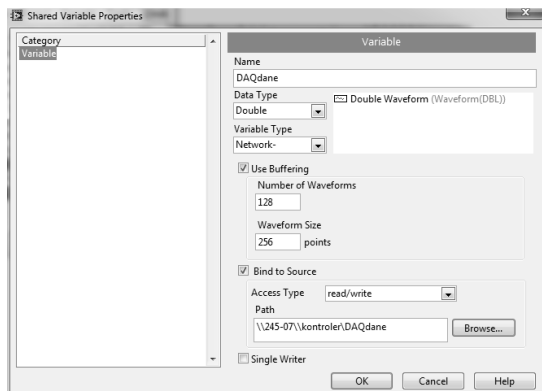


Rys. 8. Okno projektu *Uzytkownik.lvproj*

## 4. KONFIGURACJA I URUCHOMIENIE RSP

Poprawne funkcjonowanie zaprezentowanego RSP zależy od właściwej konfiguracji systemu, stworzenia odpowiednich projektów w węzłach oraz określenia warunków dostępu do poszczególnych zmiennych współdzielonych. Wszystkie komputery pracujące w węzłach RSP powinny być widoczne w jednej sieci lokalnej. Proces konfiguracji systemu rozpoczyna się od uruchomienia programu *kontroler.vi*, w którym włącza się

usługę Zarządzania Zmiennymi Współdzielonym SVE. Komputer, na którym działa SVE, przechowuje jednocześnie wszystkie wykorzystywane zmienne współdzielone. Pamiętać, zatem należy o prawidłowej konfiguracji tych zmiennych w pozostałych węzłach systemu pomiarowego. W każdym z nich, w oknie właściwości zmiennej współdzielonej, zmienną współdzieloną należy związać z odpowiadającą jej zmienną sieciową na głównym komputerze, na którym zainstalowany jest program *kontroler.vi* (rys. 9). We wszystkich węzłach systemu pomiarowego należy aktywować komunikację VI Servera za pomocą protokołu TCP/IP oraz ustawić listy dostępu do poszczególnych adresów IP komputerów pracujących w RSP. Następnie włączyć z poziomu kontrolera wszystkie przyrządy wirtualne RSP.



Rys. 9. Okno właściwości zmiennej współdzielonej *DAQdane* w projekcie *DAQ.lvproj*

## 5. CELE DYDAKTYCZNE LABORATORIUM

Celem tak zaprojektowanego stanowiska laboratoryjnego jest zaprezentowanie studentom procesu projektowania, modyfikowania i uruchamiania aplikacji RSP. Każda grupa studencka projektuje i konfiguruje jeden z węzłów RSP oraz organizuje LabVIEW Projekt dla tego węzła. Struktura RSP jest wstępnie przygotowana przez prowadzącego laboratorium. W trakcie trwania ćwiczeń jest ona modyfikowana w zależności od indywidualnych projektów węzłów. Studenci tworzą wirtualne przyrządy pomiarowe w węzłach pomiarowych, w oparciu o autonomiczne urządzenie pomiarowe (multimetr), interfejs szeregowy RS 232C lub kartę pomiarową DAQ. Następnie uruchamiają i testują programy w poszczególnych węzłach

RSP. Po zintegrowaniu całego systemu pomiarowego w głównym projekcie *kontroler.lvproj*, RSP jest konfigurowany i wspólnie uruchamiany.

Otwarty forma zarówno sprzętowa jak i programowa systemu rozproszonego umożliwia studentom zapoznanie się z nowoczesnymi rozwiązaniami zorganizowania takiego systemu w oparciu o Internet.

## 6. WNIOSKI KOŃCOWE

Referat prezentuje temat z dwóch perspektyw. Pierwsza pokazuje możliwości wykorzystania dla celów dydaktycznych specjalistycznego narzędzia programistycznego zdecydowanie upraszczającego oprogramowanie rozproszonego systemu pomiarowego RSP. Druga przedstawia organizację systemu pomiarowego, opartego na narzędziach programowych środowiska LabVIEW. W artykule zaprezentowano rozwiązanie zastosowane w laboratorium dydaktycznym Katedry Telekomunikacji Morskiej Akademii Morskiej w Gdyni, którego celem jest pokazanie całego procesu projektowania i uruchamiania aplikacji rozproszonego systemu pomiarowego (RSP) w środowisku programowym wspomagającym projektowanie systemów pomiarowych. Możliwości wykorzystania zaprojektowanego RSP w laboratorium dydaktycznym mogą być znacznie rozszerzone dzięki dołączeniu do projektu dodatkowych węzłów pomiarowych i odbiorczych.

## 7. BIBLIOGRAFIA

1. Arpaia P., Cennamo F., Daponte P.: A distributed laboratory based on object-oriented measurement systems, *Measurement*, 19 (3-4), 1996, ss. 207-215.
2. Pianegiani F., Macii D., Carbone P.: An Open Distributed Measurement System Based on an Abstract Client-Server Architecture, *IEEE Transactions on Instrumentation and Measurement*, vol. 52, No 3, 2003, ss. 686-692.
3. Grimaldi D., Marinov M.: Distributed measurement systems, *Measurement*, 30, 2001, ss. 279-287.
4. National Instruments: Architecting Systems with NI Software, *NI Newsletter*, Vol. 17, no 2, 2005, ss. 8-9.
5. National Instruments: Delivers Distributed Intelligence, *NI Newsletter*, Vol. 17, no 5, 2005, ss. 2-6.

## NETWORK DISTRIBUTED MEASUREMENT SYSTEM IN DIDACTIC LABORATORY

This paper presents the application possibilities of the commercially available software tools oriented to the design of distributed measurement system for educational purposes. The network distributed measurement system based on graphical programming environment LabVIEW is shown. It has been designed and applied at the Department of Maritime Telecommunications in Gdynia Maritime University for the tutored and self-education purposes. Students have an opportunity to execute a remote measurement as well as design, carry out and supervise their own projects using the graphical programming environment with ready-made instrument drivers (serial and DAQ boards). The position of the laboratory, consisting of several nodes with access to a computer network, in which the functions of the process and control of measurement have been dispersed is described. The advantages of the implemented components of the LabVIEW programming environment are indicated. The using such tools as the LabVIEW project, shared variables interface with the Shared Variable Engine significantly simplifying the programming of distributed application, which requires various forms of communication and data sharing. The presented distributed system assigned for the didactic laboratory is easily extensible. The application possibilities are virtually unlimited thanks to attaching additional measurement and client nodes.

**Keywords:** distributed measurement systems, distributed intelligence technology.