

Arkadiusz ADOLPH  
Arkadiusz KOMKOWSKI  
Andrzej MŁYŃSKI  
Piotr ZAPRAWA

## SYSTEM ZDALNEJ DIAGNOSTYKI POJAZDÓW TRAKCYJNYCH

**STRESZCZENIE** *W artykule zaprezentowano budowę i działanie systemu zdalnej diagnostyki pojazdów trakcyjnych. W jego skład wchodzi: dwuwariantowe urządzenie monitorujące, oprogramowanie centralnego serwera z relacyjną bazą danych oraz aplikacja wizualizacyjna. Omówione zostały również możliwości dalszego rozwoju systemu i jego zastosowań.*

### 1. WSTĘP

---

Współczesna cywilizacja oparta jest w dużej mierze na transporcie osobowym i towarowym. Problematyka usprawnienia transportu, zwłaszcza w dużych aglomeracjach miejskich, zajmuje obecnie poczesne miejsce w Programach Unii

---

**mgr inż. Arkadiusz ADOLPH**  
arkadiusz.adolph@iel.gda.pl

**mgr inż. Arkadiusz KOMKOWSKI**  
arkadiusz.komkowski@iel.gda.pl

**Andrzej MŁYŃSKI**  
andrzej.mlynski@iel.gda.pl

**mgr inż. Piotr ZAPRAWA**  
piotr.zaprawa@iel.gda.pl

Institut Elektrotechniki Oddział w Gdańsku

Europejskiej. Celem działań w tym zakresie jest przekształcenie centrów miast w sfery przyjaznego układu komunikacyjnego poprzez:

- zmniejszenie liczby awarii pojazdów i skrócenie czasu ich usuwania;
- nowe sposoby zarządzania ruchem miejskim;
- wprowadzenie systemów informacyjnych.

Potrzeba poprawy komunikacji miejskiej w dużych aglomeracjach miejskich wynika ze stałego ich rozwoju. Na przedmieściach powstają nowe osiedla mieszkaniowe, co wynika z braku terenów pod zabudowę w okolicach centrów miast, jak również z tendencji do osiedlania się poza centrum z uwagi na mniejszy hałas i mniejsze zanieczyszczenie środowiska. Konieczne jest zapewnienie mieszkańcom tych osiedli możliwości dojazdu do centrum i poruszania się po mieście. Używanie przez mieszkańców transportu własnego zwiększa ruch i hałas w obrębie centrum miasta, powodując zatory komunikacyjne oraz kłopoty z dostateczną ilością miejsc parkingowych. Alternatywą dla własnego transportu jest komunikacja publiczna, jeśli jest ona niezawodna i pozwala na sprawne i łatwe poruszanie się po całym mieście.

Główną wadą komunikacji miejskiej w polskich miastach jest słaba punktualność, wynikająca z dużego natężenia ruchu (gł. samochodów osobowych) w godzinach szczytu i częstych awarii pojazdów lub trakcji (w przypadku tramwajów i trolejbusów). W związku z tym trudno jest określić nawet przybliżony czas przybycia konkretnego pojazdu na przystanek, a zamieszczony tam rozkład jazdy często bywa bezużyteczny. Bez aktualnej informacji o sytuacji na linii dyspozytorzy nie mogą szybko reagować na zaistniałą sytuację, a pasażerowie pozbawieni są możliwości skorzystania na danej trasie z innych środków transportu miejskiego niż ten, na który oczekują.

Na przełomie kilku ostatnich lat uwidocznił się znaczny rozwój technologii telekomunikacyjno-informatycznych. Za przykład może tutaj posłużyć rozwój komunikacji bezprzewodowej bazującej na technologii GSM i GPRS. Jeszcze do niedawna systemy monitoringu i sterowania obejmowały swym działaniem najczęściej obszar pracy będący z nim w bezpośredniej styczności. W przypadku konieczności pracy w ramach systemu na większe odległości, budowano i wykorzystywano sieci dedykowane. Pojawienie się na rynku szerokiego wachlarza usług opartych na technologii GSM/GPRS znacznie zwiększyło możliwości pracy systemów zdalnej kontroli i obniżyło jednocześnie koszt ich realizacji i utrzymania.

Stworzenie nowoczesnego systemu monitoringu i sterowania, takiego jak opracowany w IEL OG, wymagało poznania szeregu nowych zagadnień.

Celem perspektywicznym przedstawionego rozwiązania jest stworzenie możliwości usprawnienia komunikacji miejskiej ze szczególnym uwzględnie-

niem pojazdów trakcyjnych. Do pełnej realizacji tego zadania przewidziane zostały cztery współpracujące ze sobą systemy:

- system diagnostyki pojazdów;
- system monitoringu aktualnego położenia pojazdów;
- system dyspozytorski nadzorujący zbieranie i wysyłanie informacji. System ten umożliwia też analizę i archiwizację zebranych danych;
- system wyświetlania informacji dla podróżnych na przystankach.

Opisany w artykule system zdalnej diagnostyki pojazdów trakcyjnych stanowi istotną część podjętego zadania. W budowie systemu wykorzystano fakt, że praktycznie wszystkie nowe pojazdy trakcyjne posiadają rozbudowane elektroniczne systemy sterownicze. Pozwalają one na stosunkowo łatwą realizację skomplikowanych algorytmów kontroli i diagnostyki zainstalowanych układów energoelektronicznych. W tej roli, w przeważającej liczbie przypadków, stosuje się programowane sterowniki (PLC), które pozwalają m.in. na implementację wielu automatycznych procedur diagnostyki pokładowej, działających w czasie rzeczywistym i wykrywających pojawiające się usterki, nieprawidłowości i zakłócenia w pracy. Informacje te jednak nie są wykorzystywane na bieżąco, a dopiero po przybyciu na miejsce ekipy serwisowej. Bieżące monitorowanie i diagnozowanie usterek w trakcie eksploatacji pojazdów może znacznie usprawnić i przyspieszyć prace serwisu.

System zdalnej diagnostyki pojazdów trakcyjnych, opracowany w Pracowni Energoelektroniki i Sterowania Oddziału Instytutu Elektrotechniki w Gdańsku we współpracy z Zakładem Trakcji Elektrycznej i Zakładem Metrologii i Badań Nieniszczących Instytutu Elektrotechniki w Warszawie, pozwala na ciągły zdalny monitoring pracy urządzeń zasilających, napędowych i sterowniczych pojazdu trakcyjnego przy pomocy terminalu podłączonego do globalnej sieci internetowej. Terminalem może być np. komputer klasy PC wyposażony w standardową przeglądarkę stron WWW z wirtualną maszyną Java.

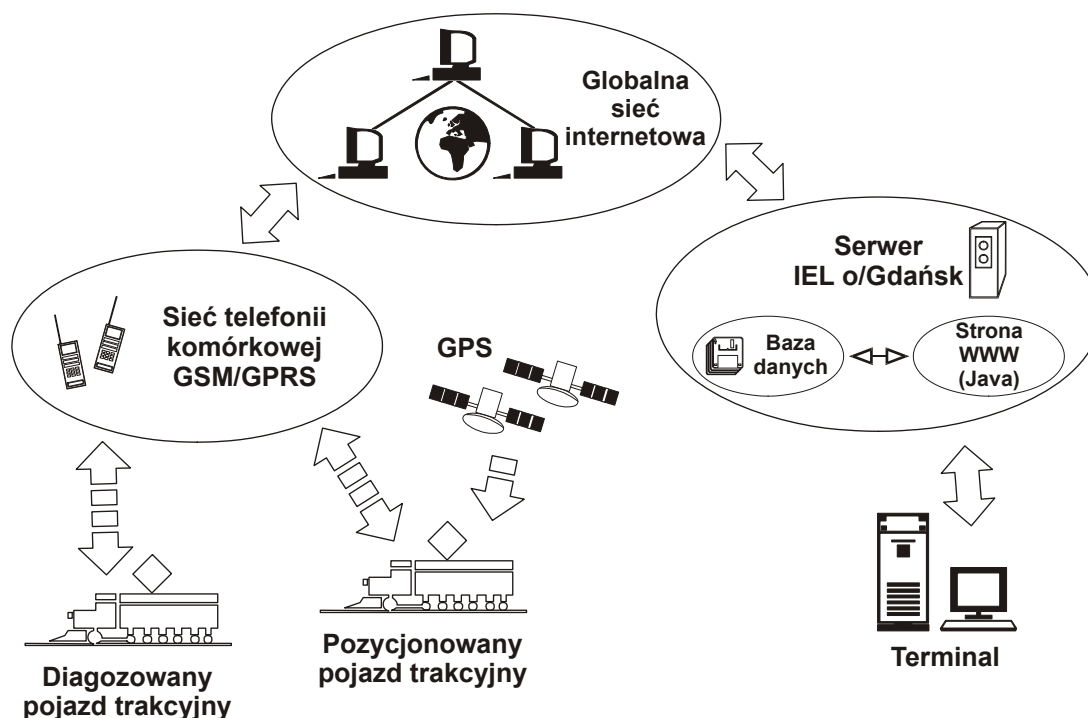
Urządzenia prototypowe zostały zainstalowane w dwóch elektrowozach typu EM-10 będących własnością DOKP w Gliwicach.

W realizacji projektu wykorzystano opracowane specjalnie dla potrzeb komunikacji z rozproszonymi systemami nadzoru oprogramowanie pracujące na serwerze internetowym.

Urządzenie monitorujące, instalowane w pojazdach trakcyjnych, powstało w wyniku modyfikacji i rozwinięcia zaprojektowanego na początku b.r. modułu kontroli położenia pojazdów komunikacji miejskiej.

## 2. OGÓLNA KONCEPCJA SYSTEMU

System zdalnej diagnostyki pojazdów trakcyjnych pozwala na zdalną bezprzewodową kontrolę podstawowych parametrów determinujących prawidłową pracę elektrowozu. Posiada funkcję automatycznego archiwizowania odbieranych informacji w bazie danych na centralnym serwerze sieciowym systemu. Zarządzanie tymi danymi oraz ich analiza jest możliwa z dowolnego miejsca przy pomocy terminalu (np. komputera PC) podłączonego do globalnej sieci internetowej i wyposażonego w funkcję przeglądania stron WWW z obsługą apletów w języku Java. Bezprzewodowym medium komunikacyjnym jest istniejąca sieć telefonii komórkowej GSM/GPRS, której wykorzystanie pozwala na ciągle utrzymywanie kontaktu z poruszającym się pojazdem na terenie całego kraju, a także poza jego granicami. Schemat przepływu danych pomiędzy poszczególnymi systemami komunikacyjnymi pokazano na rys. 1.



Rys. 1. System zdalnej diagnostyki pojazdów trakcyjnych – schemat przepływu danych

Urządzenie monitorujące można w stosunkowo prosty sposób przystosować do pracy w trybie określania położenia pojazdu w terenie, wymieniając moduł diagnostyczny na moduł kontroli położenia, zawierający odbiornik GPS.

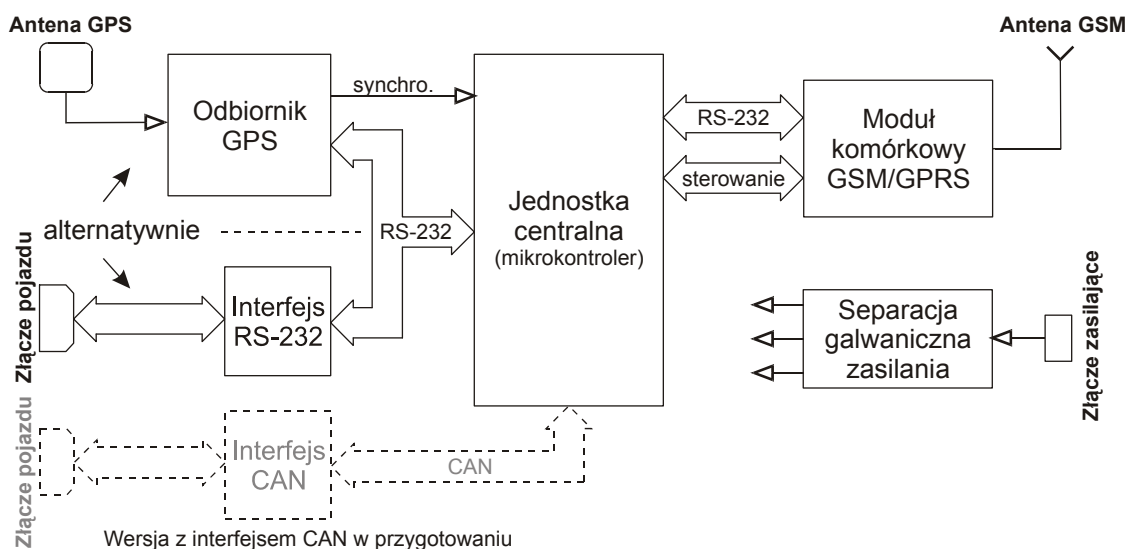
### 3. URZĄDZENIE MONITORUJĄCE

Jednym z głównych elementów składowych systemu diagnostycznego jest urządzenie monitorujące, mogące występować w dwóch podstawowych wersjach: diagnostycznej i pozycjonującej. W obecnym rozwiązaniu urządzenie może realizować tylko jedną z tych funkcji.

Do podstawowych zadań urządzenia, przeznaczonego do instalowania w monitorowanym pojeździe trakcyjnym, należą: pobieranie danych diagnostycznych ze sterownika elektrowozu, przetwarzanie ich na postać pozwalającą na transmisję sieciową oraz przesyłanie do serwera, który archiwizuje je w bazie danych.

#### 3.1. Rozwiązanie sprzętowe

Ogólną budowę urządzenia monitorującego przedstawia rys. 2.



**Rys. 2. Uproszczony schemat blokowy urządzenia monitorującego, pracującego w systemie zdalnej diagnostyki pojazdów trakcyjnych**

Funkcję jednostki centralnej pełni mikrokontroler z rodziny AVR produkowany przez firmę Atmel. Wymagania układu podyktowały konieczność zasto-

sowania najbardziej rozbudowanego układu z tej rodziny ATmega128L, który oprócz dość wydajnego obliczeniowo, jak na jednostki 8-bitowe, procesora o architekturze RISC i obszernej wewnętrznej pamięci programu, zawiera w swej strukturze wiele dodatkowych układów peryferyjnych, w tym 2 sprzętowe układy asynchronicznej komunikacji szeregowej (UART).

W układzie zastosowano moduł telefonu komórkowego g20 firmy Motorola, który posiada wbudowane procedury obsługi protokołów sieciowych, co znacząco uprościło oprogramowanie układu, a tym samym przyspieszyło wykonanie produktu finalnego. Stan pracy modułu jest na bieżąco kontrolowany przez mikrokontroler, dzięki czemu możliwa jest natychmiastowa reakcja na wszelkie zdarzenia pojawiające się w przypadku zaniku sygnału, przerwania połączenia itp. Komunikacja z modułem odbywa się za pośrednictwem interfejsu szeregowego RS-232 z wykorzystaniem rozszerzonego zestawu komend Hayes-AT oraz kilku linii sterujących.

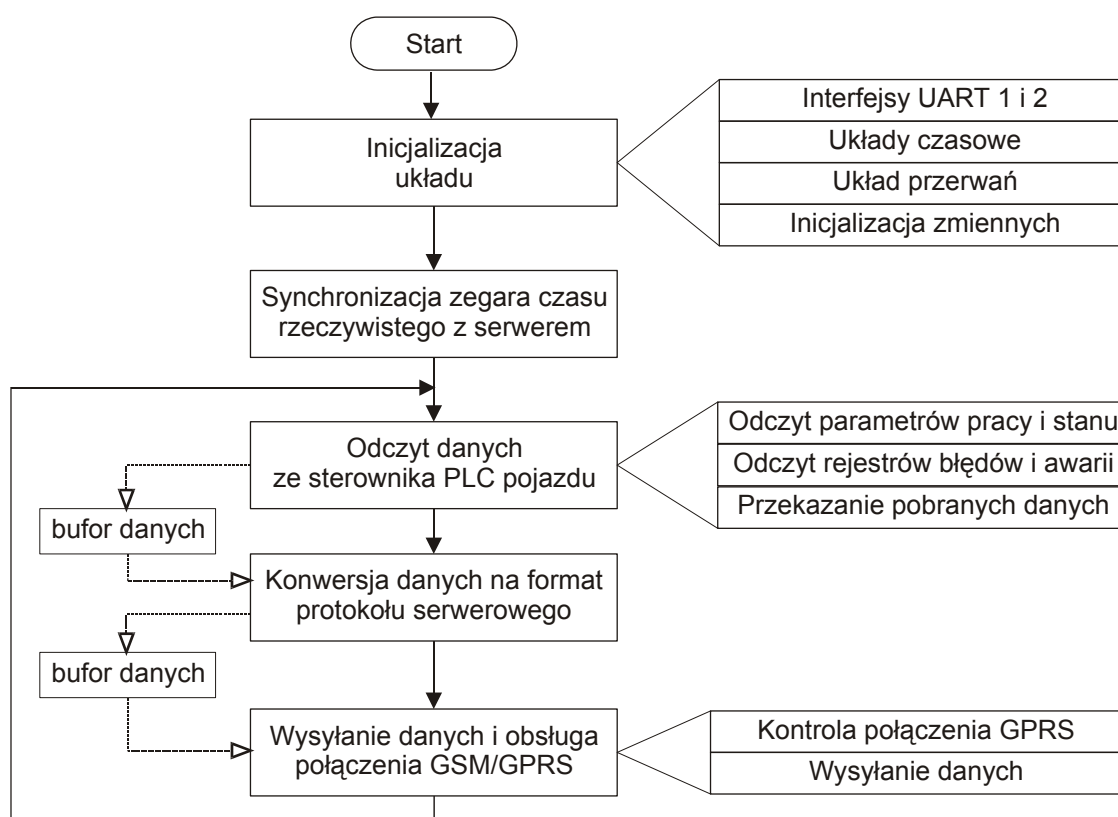
W wersji diagnostycznej jednostka centralna pobiera dane cyklicznie ze sterownika PLC elektrowozu za pośrednictwem interfejsu RS-232, z programowo określonym czasem cyklu. Komunikacja odbywa się z wykorzystaniem powszechnego w takim zastosowaniu protokołu MODBUS-RTU. Pozyskane w ten sposób informacje (parametry pracy, informacje o bieżącym stanie układów i rejestry awarii) są przetwarzane na postać akceptowaną przez protokół komunikacyjny serwera i przekazywane do wysłania przez moduł telefonu komórkowego GSM/GPRS.

Urządzenie monitorujące, pracujące w konfiguracji pozycjonowania pojazdu, nie posiada możliwości pobierania danych diagnostycznych, ale pozwala na zdalne określanie pozycji pojazdu w terenie z dokładnością sięgającą 15 m. Do określania dokładnego położenia geograficznego użyto odbiornika sygnałów GPS Motorola M12 Oncore z zewnętrzną anteną aktywną. Odbiornik komunikuje się z mikrokontrolerem za pośrednictwem łącza szeregowego. Jednostka centralna na bieżąco otrzymuje z odbiornika GPS dane mówiące o aktualnej pozycji geograficznej pojazdu, które podobnie jak w poprzednim przypadku są przesyłane do serwera za pośrednictwem sieci GSM/GPRS.

Obecnie prowadzone są prace nad nową wersją urządzenia monitorującego, łączącego w swej strukturze obie wyżej wymienione funkcje – jednoczesną diagnostykę pojazdu i określanie jego położenia w terenie. Znaczącą zmianą w stosunku do pierwotnej wersji jest zastosowanie innego rodzaju połączenia ze sterownikiem (lub sterownikami) pojazdu trakcyjnego – sieci CAN. Ta modyfikacja pozwoli na bardziej kompleksową diagnostykę i monitoring procesów zachodzących we wszystkich układach sterujących pracą elektrowozu, ponieważ sieć CAN pozwala na pobieranie informacji z wielu sterowników i kontrolerów rozmieszczonych w pojeździe.

### 3.2. Oprogramowanie urządzenia monitorującego

Oprogramowanie mikrokontrolera urządzenia monitorującego napisano w języku C. Algorytm pracy urządzenia w wersji diagnostycznej, przedstawiono na rys. 3.

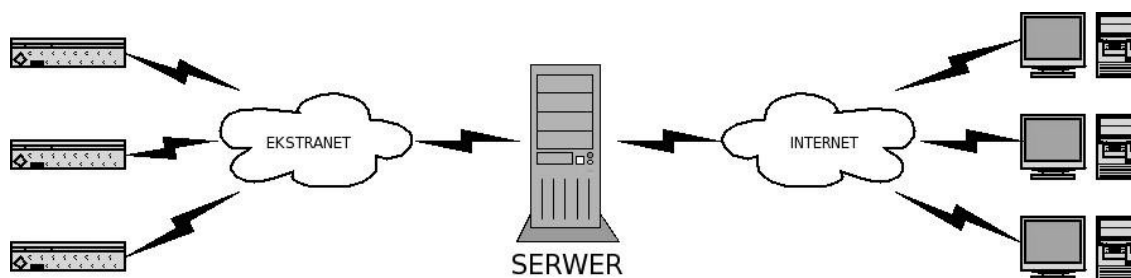


Rys. 3. Uproszczony algorytm działania oprogramowania urządzenia monitorującego

Program pracy w dużym uproszczeniu działa według schematu, pokazanego na rys. 3, jednak jego konfiguracja i konstrukcja tworzy prosty system pseudo-wielozadaniowy. Każde z 3 obsługiwanych zadań jest procedurą o specjalnej jednoprzebiegowej konstrukcji z wyodrębnioną pamięcią stanu pracy. Dane pomiędzy zadaniami przekazywane są za pośrednictwem systemu dedykowanych buforów. Dzięki takiej konstrukcji możliwe jest połączenie różnych procesów, których częścią wspólną jest jedynie zestaw danych.

## 4. SERWER KOMUNIKACYJNY

System monitoringu i sterowania opracowany w Oddziale Instytutu Elektrotechniki w Gdańsku został zrealizowany w oparciu o sieć Internet. Centralną częścią tego systemu jest serwer sieciowy, realizujący komunikację w sieci na poziomie protokołów TCP/IP i UDP/IP. Zadania serwera obejmują integrację rozproszonych obiektów w ramach systemu oraz udostępnianie zdalnym klientom informacji o stanie współpracujących obiektów. W topologii systemu serwer zajmuje pozycję centralną, co przedstawiono na rys. 4.



Rys. 4. Pozycja serwera w strukturze systemu sterowania i monitoringu

Oprogramowanie serwera zostało zrealizowane w oparciu o architekturę modułową. O wyborze tego rozwiązania zdecydowała między innymi łatwość rozbudowy, możliwość dynamicznej konfiguracji oraz ładowania nowych modułów podczas pracy serwera.

Schemat komunikacji jest niezależny od topologii sieci oraz liczby obiektów pracujących w jej ramach. Zespół rozproszonych obiektów, realizujących funkcje pomiarowe, kontrolne i sterujące, komunikuje się bezpośrednio z serwerem za pośrednictwem sieci Internet. Odebrane dane gromadzone są w relacyjnej bazie danych. Dostęp klientów do zgromadzonych danych realizowany jest przez serwer sieciowy, współpracujący z relacyjną bazą danych i umożliwiający uwierzytelnionym klientom równoczesny dostęp do szeregu funkcji z zakresu monitorowania, kontroli oraz sterowania.

Podstawą komunikacji z urządzeniami jest protokół nadrzędny wyższego poziomu, zrealizowany na bazie UDP/IP, opracowany specjalnie do potrzeb sterowania poprzez sieć Internet. Natomiast komunikacja ze zdalnymi klientami zrealizowana została na bazie standardowego protokołu TCP/IP.

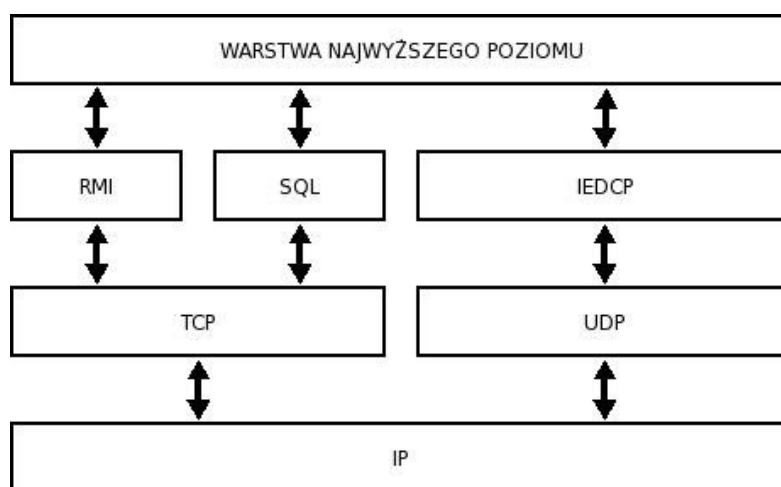
Oprogramowanie serwera zostało zrealizowane przy wykorzystaniu języka JAVA2.



#### 4.1. Zadania serwera komunikacyjnego

Konsekwencją centralnego umieszczenia serwera w topologii systemu jest znaczna ilość zadań, jakie musi on realizować. Poszczególne zadania zostaną przedstawione w zarysie dla każdej z warstw komunikacyjnych.

Najniżej położoną warstwą zadań realizowanych przez serwer jest zapewnienie komunikacji bazującej na określonym protokole sieciowym. Obsługiwane protokoły to TCP/IP oraz UDP/IP. Następną warstwą komunikacyjną działa wykorzystując przedstawione powyżej protokoły. W ramach sieci integrującej zespół obiektów rozproszonych, używany jest protokół wysokiego poziomu opracowany na bazie UDP/IP, umożliwiający realizację komunikacji stratnej i bezstratnej. W ramach sieci udostępniającej szereg usług zdalnym klientom, wykorzystywany jest protokół komunikacji dla zdalnego wywoływania metod Javy. W ramach komunikacji z relacyjną bazą danych, używany jest protokół zdalnej obsługi zapytań. Zależności pomiędzy poszczególnymi warstwami komunikacji przedstawiono na rys. 5.



Rys. 5. Schemat blokowy poszczególnych warstw oraz wzajemnych zależności protokołów komunikacji

Nadrzędna warstwa odpowiedzialna jest za zapewnienie jednoczesnej, równoległej komunikacji wielu obiektom i klientom. Dotyczy to dwukierunkowej komunikacji ze zdalnymi obiektami w sieciach rozproszonych oraz klientami w sieci Internet, a także operacji zapisu i odczytu w bazie danych. Następną warstwą, odpowiedzialną za komunikację pomiędzy poszczególnymi funkcjami serwera, integruje takie elementy jak obsługę zdalnych klientów oraz rozproszonych obiektów z bazą danych. Ostatnia warstwa realizuje nadrzędne za-

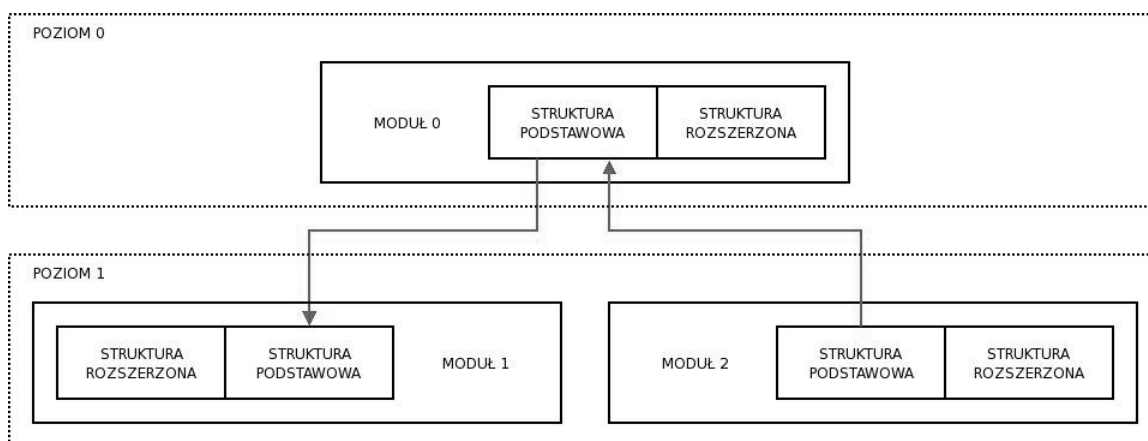
dania serwera, do których zaliczyć można między innymi gromadzenie danych przekazywanych przez obiekty rozproszone oraz sterowanie nimi przez zdalnego klienta.

## 4.2. Modułowa struktura serwera

Wewnętrzna struktura oprogramowania serwera została zrealizowana w oparciu o architekturę modułową. Architektura taka wymaga opracowania protokołu komunikacji pomiędzy poszczególnymi modułami.

Budowa każdego modułu składa się ze struktury podstawowej, tworzącej blok funkcjonalny, oraz struktury rozszerzającej. Struktura podstawowa jest taka sama dla każdego z modułów. Implementuje ona stały zestaw poleceń, realizujący komunikację międzymodułową oraz dynamiczne ładowanie modułów. Zawiera ona także standardowe rozszerzenie do struktury, zależnej od implementacji funkcjonalnej danego modułu.

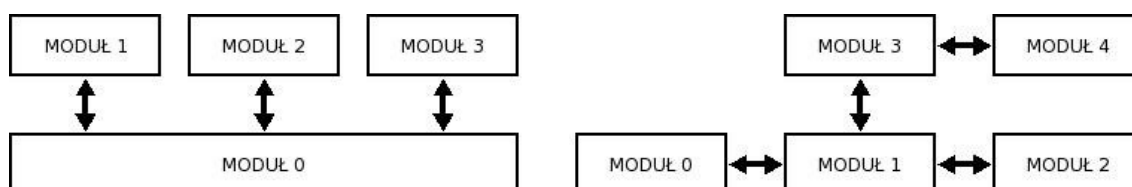
Dzięki temu, że podstawowa struktura, umieszczona w każdym module, jest taka sama, możliwe jest tworzenie dowolnej topologii międzymodułowej. Topologia ta zależy ściśle od zależności pomiędzy modułami i obejmuje głównie usługi, jakie świadczy jeden moduł na rzecz drugiego. Schemat blokowy budowy modułów i komunikacji między nimi przedstawiono na rys. 6.



**Rys. 6. Schemat blokowy budowy modułów i komunikacji między nimi**

Świadczenie usług międzymodułowych realizowane jest na drodze poleceń. Moduł wysyłający polecenie odpowiedzialny jest za jego odpowiednie przygotowanie, obejmujące utworzenie niezależnego obiektu danych, zawierającego rozkaz polecenia wraz ze wszystkim parametrami. Umieszczone są

w nim również informacje o adresie nadawcy i odbiorcy z możliwością sterowania przebiegiem transmisji. Na rysunku 7. przedstawione zostały przykłady konfiguracji modułów i zależności pomiędzy nimi.



Rys. 7. Przykłady konfiguracji modułów i zależności między nimi

Podstawowa struktura, wchodząca w skład każdego modułu, zawiera standardowy zestaw poleceń, umożliwiającą między innymi tworzenie dowolnego modułu potomnego, który umieszczony zostaje poniżej w hierarchii.

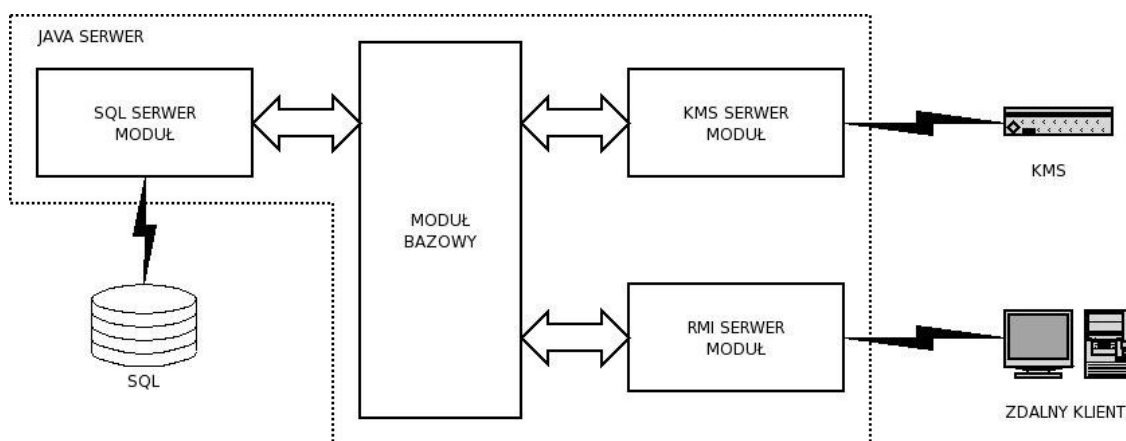
Zapewnienie nowo utworzonemu modułowi potomnemu możliwości komunikacji wymaga nadania mu adresu, który jest jednoznacznym, niepowtarzalnym identyfikatorem wyróżniającym go z pośród wszystkich modułów. W zależności od potrzeb, każdy moduł ma możliwość pobrania dowolnej liczby adresów. Każdy z nich może być użyty do adresowania w obrębie rozszerzeń modułu.

W celu zapewnienia komunikacji międzymodułowej, każdy nowo utworzony moduł musi przeprowadzić dodatkowy proces rejestracji w systemie. Proces ten ma na celu lokalizację modułu w systemie i wyznaczenie ścieżki dostępu do niego.

### 4.3. Realizacja komunikacji modułów

Każdy moduł, realizujący pewne określone zadania, jest kompilacją struktury podstawowej oraz specyficznego rozszerzenia. Struktura podstawowa, wchodząc w skład każdego z modułów, jest odpowiedzialna między innymi za realizację procesu komunikacji między modułowymi. Została ona zrealizowana w oparciu o zlecenia przekazywane pomiędzy modułami. Mają one postać obiektów o ściśle określonej budowie, zawierającego między innymi adres nadawcy i odbiorcy, parametry oraz samo polecenie. Określenie miejsca danego modułu w strukturze serwera, realizowane jest podczas rejestracji modułu, zaraz po jego utworzeniu. Na jego podstawie wyznaczana jest trasa prze-

kazywania polecenia realizowanego według jednoznacznie określonego algorytmu. Zależności komunikacyjne w ramach systemu sterowania zostały przedstawione na rys. 8.



Rys. 8. Komunikacja i zależności w ramach systemu sterowania

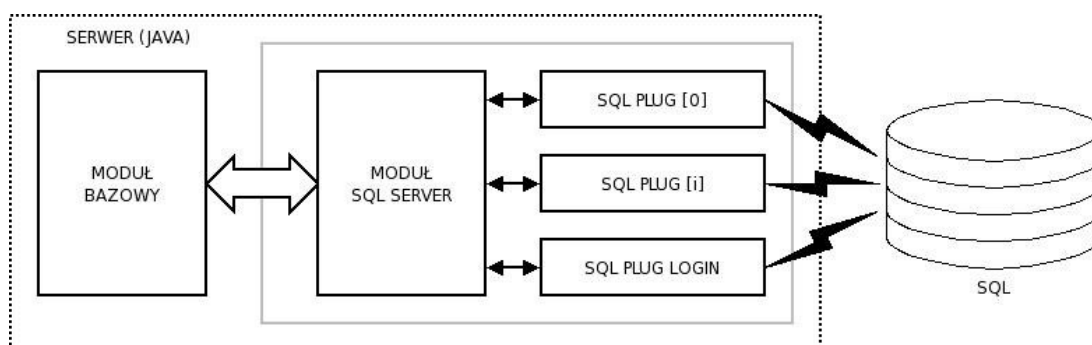
#### 4.4. Wymiana danych z bazą danych

Wszystkie dane, jakie przetwarza i gromadzi serwer, przechowywane są w relacyjnej bazie danych. Znajdują się w niej także wszystkie dane niezbędne do pracy oprogramowania serwera. Komunikacja z bazą danych realizowana jest za pośrednictwem połączeń wykorzystujących mechanizm protokołu TCP/IP. Podczas pracy systemu, połączenia te są realizowane statycznie i dynamicznie.

Oprogramowanie współpracujące z bazą danych zapewnia wykonywanie następujących operacji:

- weryfikację klientów oraz zarządzaniem dostępem;
- wprowadzanie danych pochodzących od urządzeń rozproszonych;
- równoległy i wielowątkowy dostęp do zgromadzonych danych;

Schemat blokowy komunikacji z relacyjną bazą danych przedstawiono na rys. 9.



Rys. 9. Schemat blokowy komunikacji z relacyjną bazą danych

Całkowita liczba połączeń z bazą danych zmienia się dynamicznie, w zależności od aktywności użytkowników. Wszystkie połączenia z bazą danych można podzielić na trzy zasadnicze kategorie. Kategorie pierwsza to połączenia realizowane w sposób ciągły. Za pośrednictwem takich połączeń wprowadzane są do bazy danych wszystkie dane pochodzące od rozproszonych urządzeń zewnętrznych. Kategorie druga to połączenia realizowane dynamicznie, które tworzone są w chwili podłączenia nowych użytkowników. Trzecia kategoria to połączenia tworzone dynamicznie, realizujące asynchroniczną weryfikację nowych użytkowników w oparciu o informacje zawarte w bazie danych.

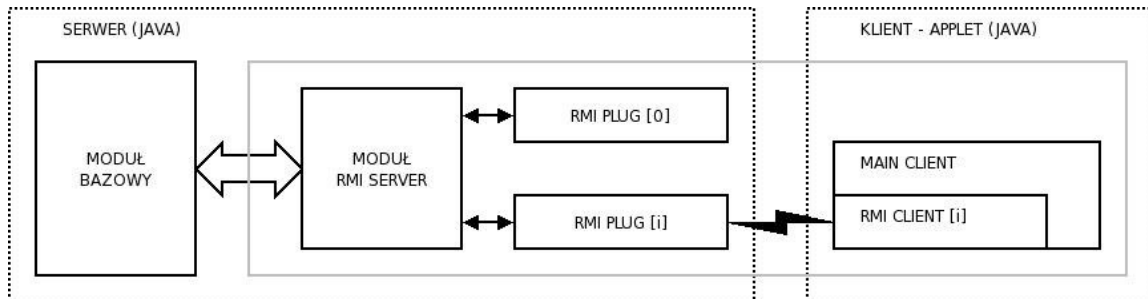
Możliwość dynamicznej realizacji wielu połączeń z bazą danych wynika z wewnętrznej struktury modułu obsługi i obiektów z nim współpracujących. Moduł obsługujący bazę danych tworzony jest podczas konfiguracji serwera bezpośrednio po jego uruchomieniu. Niezależnie od aktywności użytkowników i urządzeń, komunikacja z nim jest utrzymywana w sposób ciągły. Dla każdego nowego połączenia tworzony jest dynamicznie nowy obiekt realizujący kanał komunikacyjny z bazą danych. Obiekt ten zarządzany jest przez odpowiedni, wbudowany mechanizm modułu, który reguluje także tak zwany cykl życia obiektów komunikacyjnych.

#### 4.5. Komunikacja ze zdalnym klientem

Oprogramowanie serwera zapewnia jednoczesną komunikację z wieloma zdalnymi klientami. Moduł realizujący to zadanie przeprowadza autoryzację użytkownika oraz odpowiada za transfer danych z zapewnieniem podstawowego stopnia ich ochrony. Za inicjalizację połączenia z serwerem odpowiedzialny jest klient. Utworzenie indywidualnego kanału komunikacyjnego jest poprzedzone procesem autoryzacji.

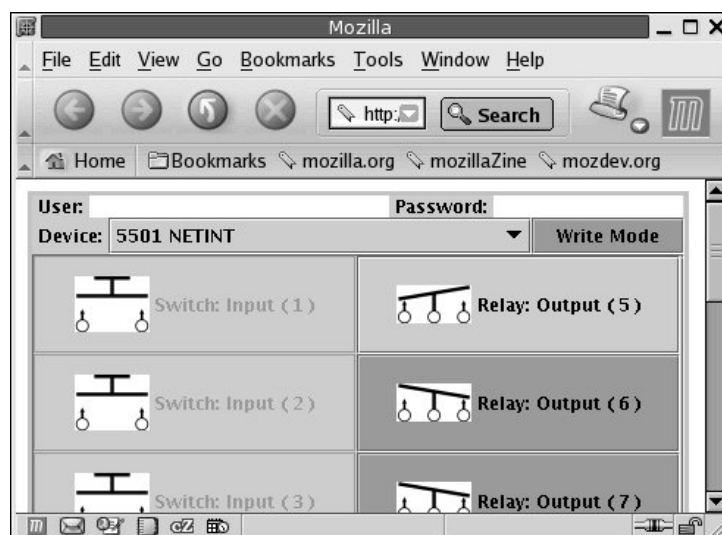
Komunikacja pomiędzy serwerem a klientem odbywa się z wykorzystaniem mechanizmu Remote Method Invocation (RMI), umożliwiającego realiza-

cję zdalnych wywołań metod Javy. Schemat blokowy komunikacji ze zdalnym klientem przedstawiono na rys. 10.



Rys. 10. Schemat blokowy komunikacji z zdalnym klientem

Inicjalizację połączenia z serwerem przeprowadza klient, rozpoczynając proces identyfikacji użytkownika. Moduł komunikacji ze zdalnym klientem posiada wyspecjalizowany obiekt pośredniczący w tym logowaniu. Po jego przeprowadzeniu, moduł rozpoczyna proces otwarcia nowej sesji. Umożliwia to klientowi przełączenie się na jego własny kanał komunikacyjny. Wygląd przykładowego appletu logowania przedstawiony został na rys. 11.



Rys.11. Wygląd przykładowego appletu logowania

Wykorzystanie w komunikacji pomiędzy serwerem a klientem mechanizmu RMI, umożliwia realizację zdalnych wywołań metod Javy. Modułowi komunikującemu się z klientem, mającemu oprogramowanie rozmieszczone na więcej niż jednym komputerze, pozwala to działać w taki sposób aby sieć była niewidoczna.

#### 4.6. Komunikacja z urządzeniami rozproszonymi

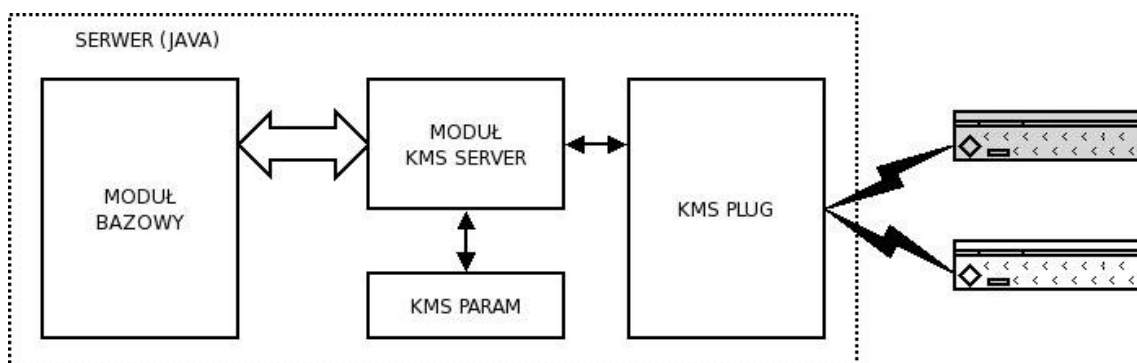
Podstawową cechą zaprojektowanego serwera komunikacyjnego jest możliwość współpracy z obiektami znajdującymi się w sieciach rozproszonych. Bez względu na ich fizyczną alokację, serwer jest odpowiedzialny za integrację ich z systemem oraz bezpośrednio sterowanie nimi. Na bazie protokołu UDP/IP zbudowano protokół wyższego poziomu. Rozszerza on między innymi możliwości protokołu UDP/IP o transmisję bezstratną. Protokół ten, zaprojektowany w IEL OG nazwano protokołem IEDCP.

Obiekty rozproszone, łącząc się z serwerem, realizują dwukierunkową transmisję za pośrednictwem określonego portu serwera. Urządzenia transmitują tą drogą dane zbierane przez serwer oraz pobierają i wysyłają parametry mające wpływ na ich pracę.

Dane zgromadzone w obiektach rozproszonych przesyłane są do określonego portu serwera protokołem IEDCP. Wyspecjalizowany obiekt, pracujący w ramach modułu komunikacji z urządzeniami rozproszonymi, odczytuje i gromadzi na bieżąco nadchodzące dane, a następnie przetwarza je i interpretuje aby ostatecznie przekazać je do modułu współpracującego z bazą danych.

Przyjęty sposób transmisji pozwala na wprowadzenie dodatkowych opcji. Na przykład, każdy pełen proces transmisji można rozszerzyć o jawne otwarcie i zamknięcie sesji.

Przesyłanie danych do obiektów realizowane jest na bazie transmisji bezstratnej. Odbieranie oraz konwersja pakietów UDP przebiega w serwerze analogicznie jak podczas logowania. Dodatkowo jednak aktywowany zostaje obiekt współpracujący z modułem, realizujący proces nadzoru nad każdym otwartym kanałem komunikacji. Zostaje on uaktywniony w momencie otwarcia przez obiekt rozproszony połączenia i zamknięty wraz z jego zakończeniem. Schemat blokowy komunikacji z obiektami rozproszonymi przedstawiono na rys. 12.

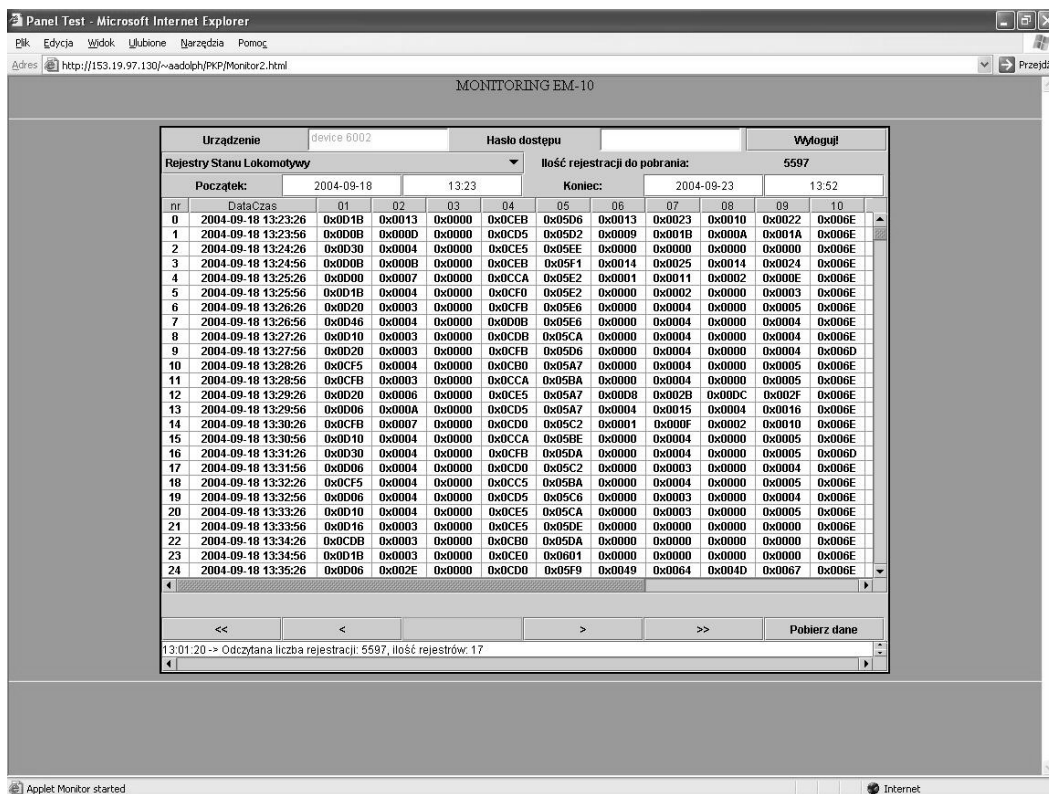


Rys. 12. Schemat blokowy komunikacji z obiektami rozproszonymi

## 5. SERWER – OPROGRAMOWANIE WIZUALIZACYJNE

Ważnym elementem systemu zdalnej diagnostyki pojazdów jest oprogramowanie pozwalające na wizualizację danych zgromadzonych na serwerze. Ponieważ jednym z głównych założeń przy tworzeniu systemu wizualizacyjnego była jego praca na wielu platformach sprzętowych, został on stworzony jako aplet w języku Java. Dzięki temu może być uruchomiony na dowolnym komputerze posiadającym wirtualną maszynę Java, niezależnie od systemu operacyjnego. Do uruchamiania apletu można użyć standardowej przeglądarki internetowej.

Jednym z najważniejszych elementów systemu wizualizacyjnego jest jego interfejs graficzny. Dzięki umieszczeniu całej aplikacji na kontenerze JPanel zostaje ona osadzona na stronie internetowej i wyświetlona w głównym oknie przeglądarki, którego wygląd przedstawiono na rys. 13.



Rys. 13. Widok okna głównego przeglądarki

Okno aplikacji zostało podzielone na 6 obszarów. W górnej części znajduje się pasek logowania, w którym należy wypełnić pola „Urządzenie” i „Hasło dostępu” i poprzez naciśnięcie „Enter” zatwierdzić wprowadzone dane. Wygląd paska logowania przedstawia rys. 14.



Urządzenie		Hasło dostępu	
▼			

Rys. 14. Pasek logowania

Po zalogowaniu pojawia się lista wyboru rejestru z którego będą pobierane dane. Dostępny staje się też przycisk wylogowania. Po wybraniu rejestru określone są minimalne i maksymalne zakresy czasowe rejestracji, które mogą być dowolnie zmieniane. Na bieżąco zliczana jest też ilość rejestracji do pobrania przy podanych zakresach. Wygląd pola wyboru rejestru i zakresu rejestracji przedstawiono na rys. 15.

Urządzenie	device 6002	Hasło dostępu		Wyloguj!
Rejestry Stanu Lokomotywy		Ilość rejestracji do pobrania:		5597
Początek:	2004-09-18	13:23	Koniec:	2004-09-23 13:52

Rys. 15. Lista wyboru rejestru oraz pola do określania minimalnego i maksymalnego czasu rejestracji

Po dokonaniu wyboru rejestru dostępny staje się też przycisk „Pobierz dane”. Jego naciśnięcie powoduje wczytanie wartości wybranej grupy rejestrów. Wygląd obszaru okna zawierającego odczytane z serwera dane dla Rejestrów Stanu Lokomotywy przedstawia rys. 16.

Początek:		2004-09-18	13:23	Koniec:		2004-09-23	13:52					
nr	DataCzas	01	02	03	04	05	06	07	08	09	10	
0	2004-09-18 13:23:26	0x0D1B	0x0013	0x0000	0x0CEB	0x05D6	0x0013	0x0023	0x0010	0x0022	0x006E	▲
1	2004-09-18 13:23:56	0x0D0B	0x000D	0x0000	0x0CD5	0x05D2	0x0009	0x001B	0x000A	0x001A	0x006E	▲
2	2004-09-18 13:24:26	0x0D30	0x0004	0x0000	0x0CE5	0x05EE	0x0000	0x0000	0x0000	0x0000	0x006E	▲
3	2004-09-18 13:24:56	0x0D0B	0x000B	0x0000	0x0CEB	0x05F1	0x0014	0x0025	0x0014	0x0024	0x006E	▲
4	2004-09-18 13:25:26	0x0D00	0x0007	0x0000	0x0CCA	0x05E2	0x0001	0x0011	0x0002	0x000E	0x006E	▲
5	2004-09-18 13:25:56	0x0D1B	0x0004	0x0000	0x0CF0	0x05E2	0x0000	0x0002	0x0000	0x0003	0x006E	▲
6	2004-09-18 13:26:26	0x0D20	0x0003	0x0000	0x0CFB	0x05E6	0x0000	0x0004	0x0000	0x0005	0x006E	▲
7	2004-09-18 13:26:56	0x0D46	0x0004	0x0000	0x0D0B	0x05E6	0x0000	0x0004	0x0000	0x0004	0x006E	▲
8	2004-09-18 13:27:26	0x0D10	0x0003	0x0000	0x0CDB	0x05CA	0x0000	0x0004	0x0000	0x0004	0x006E	▲
9	2004-09-18 13:27:56	0x0D20	0x0003	0x0000	0x0CFB	0x05D6	0x0000	0x0004	0x0000	0x0004	0x006D	▲
10	2004-09-18 13:28:26	0x0CF5	0x0004	0x0000	0x0CB0	0x05A7	0x0000	0x0004	0x0000	0x0005	0x006E	▲
11	2004-09-18 13:28:56	0x0CFB	0x0003	0x0000	0x0CCA	0x05BA	0x0000	0x0004	0x0000	0x0005	0x006E	▲
12	2004-09-18 13:29:26	0x0D20	0x0006	0x0000	0x0CE5	0x05A7	0x00D8	0x002B	0x00DC	0x002F	0x006E	▲
13	2004-09-18 13:29:56	0x0D06	0x000A	0x0000	0x0CD5	0x05A7	0x0004	0x0015	0x0004	0x0016	0x006E	▲
14	2004-09-18 13:30:26	0x0CFB	0x0007	0x0000	0x0CD0	0x05C2	0x0001	0x000F	0x0002	0x0010	0x006E	▲
15	2004-09-18 13:30:56	0x0D10	0x0004	0x0000	0x0CCA	0x05BE	0x0000	0x0004	0x0000	0x0005	0x006E	▲
16	2004-09-18 13:31:26	0x0D30	0x0004	0x0000	0x0CFB	0x05DA	0x0000	0x0004	0x0000	0x0005	0x006D	▲
17	2004-09-18 13:31:56	0x0D06	0x0004	0x0000	0x0CD0	0x05C2	0x0000	0x0003	0x0000	0x0004	0x006E	▲
18	2004-09-18 13:32:26	0x0CF5	0x0004	0x0000	0x0CC5	0x05BA	0x0000	0x0004	0x0000	0x0005	0x006E	▲
19	2004-09-18 13:32:56	0x0D06	0x0004	0x0000	0x0CD5	0x05C6	0x0000	0x0003	0x0000	0x0004	0x006E	▲
20	2004-09-18 13:33:26	0x0D10	0x0004	0x0000	0x0CE5	0x05CA	0x0000	0x0003	0x0000	0x0005	0x006E	▲
21	2004-09-18 13:33:56	0x0D16	0x0003	0x0000	0x0CE5	0x05DE	0x0000	0x0000	0x0000	0x0000	0x006E	▲
22	2004-09-18 13:34:26	0x0CDB	0x0003	0x0000	0x0CB0	0x05DA	0x0000	0x0000	0x0000	0x0000	0x006E	▲
23	2004-09-18 13:34:56	0x0D1B	0x0003	0x0000	0x0CE0	0x0601	0x0000	0x0000	0x0000	0x0000	0x006E	▲
24	2004-09-18 13:35:26	0x0D06	0x002E	0x0000	0x0CD0	0x05F9	0x0049	0x0064	0x004D	0x0067	0x006E	▼

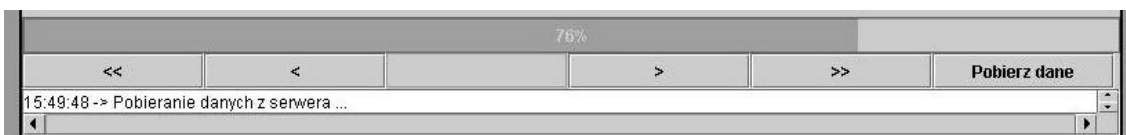
Rys. 16. Obszar okna zawierający odczytane z serwera dane dla Rejestrów Stanu Lokomotywy

Dane rejestrów stanu (RegStan i Rejestry Stanu Lokomotywy) dla każdej rejestracji są podawane w postaci heksadecymalnej. Natomiast stany Rejestru Aktywnych Alarmów oraz Wyjść PLC są podawane w postaci binarnej (gdzie 1 oznacza aktywność). Wygląd obszaru okna zawierającego odczytane z serwera dane dla Rejestru Awarii przedstawia rys. 17.

Początek:		2004-09-12		13:23		Koniec:		2004-09-23		13:52	
nr	DataCzas	01	02	03	04	05	06	07	08	09	10
0	2004-09-13 10:14:33	0	0	0	0	0	0	0	0	0	0
1	2004-09-13 11:00:38	0	0	0	0	0	0	0	0	0	0
2	2004-09-14 13:03:25	0	0	0	0	0	0	0	0	0	0
3	2004-09-14 18:06:12	0	0	0	0	0	0	0	0	0	0
4	2004-09-15 14:15:18	0	0	0	0	0	0	0	0	0	0
5	2004-09-15 15:04:45	0	0	0	0	0	0	0	0	0	0
6	2004-09-15 16:31:49	0	0	0	0	0	0	0	0	0	0
7	2004-09-16 21:38:14	0	0	0	0	0	0	0	0	0	0
8	2004-09-17 10:09:19	0	0	0	0	0	0	0	0	0	0
9	2004-09-17 10:15:12	0	0	0	0	0	0	0	0	0	0
10	2004-09-17 10:17:41	0	0	0	0	0	0	0	0	0	0
11	2004-09-17 10:18:58	0	0	0	0	0	0	0	0	0	0
12	2004-09-17 11:47:01	0	0	0	0	0	0	0	0	0	0
13	2004-09-17 11:54:01	0	0	0	0	0	0	0	0	0	0
14	2004-09-17 14:00:33	0	0	0	0	0	0	0	0	0	0
15	2004-09-18 20:39:05	0	0	0	0	0	0	0	0	0	0
16	2004-09-20 00:17:00	0	0	0	0	0	0	0	0	0	0
17	2004-09-20 16:29:14	0	0	0	0	0	0	0	0	0	0
18	2004-09-20 17:11:22	0	0	0	0	0	0	0	0	0	0
19	2004-09-20 21:16:52	0	0	0	0	0	0	0	0	0	0
20	2004-09-21 05:04:06	0	0	0	0	0	0	0	0	0	0
21	2004-09-21 16:41:33	0	0	0	0	0	0	0	0	0	0
22	2004-09-21 21:15:47	0	0	0	0	0	0	0	0	0	0
23	2004-09-22 08:30:25	0	0	0	0	0	0	0	0	0	0
24	2004-09-22 09:02:43	0	0	0	0	0	0	0	0	0	0

Rys. 17. Obszar okna zawierający odczytane z serwera dane dla Rejestru Awarii

W czasie ładowania danych jest wyświetlany wskaźnik postępu operacji, a ewentualne informacje o błędach są umieszczane na pasku statusu. Każdy wpis jest oznaczany godziną. Istnieje możliwość przewijania paska statusu, tak aby w każdej chwili można było sprawdzić poprzednie komunikaty. Wygląd pasków postępu, nawigacji i statusu przedstawiono na rys. 18.



Rys. 18. Paski postępu, nawigacji i statusu

Powyżej paska stanu po pobraniu pierwszej serii rejestracji (od 0 do 49) zostają uaktywnione przyciski nawigacji pozwalające na przejście do kolejnych

serii. Symbole „<<” i „>>” oznaczają przejście na początek lub koniec, natomiast symbole „<” i „>” umożliwiają przejście do poprzedniej lub następnej serii rejestracji.

## 6. PODSUMOWANIE

---

Wdrożenie przedstawionego w artykule rozwiązania w elektrowozach typu EM-10 DOKP w Gliwicach otwiera drogę do szerokiego stosowania technologii internetowych do zdalnego monitoringu i sterowania pojazdów trakcyjnych. Stanowi również przykład współpracy i właściwego wykorzystania kompetencji różnych komórek organizacyjnych Instytutu Elektrotechniki.

W chwili obecnej prowadzone są prace nad nową, bardziej zintegrowaną wersją urządzenia monitorującego. Urządzenie to będzie się komunikować ze sterownikiem pojazdu za pomocą interfejsu CAN. Uprości to obsługę urządzenia od strony sterownika pojazdu i umożliwi realizację rozszerzonych funkcji diagnostycznych. Nowa wersja urządzenia będzie miała możliwość jednoczesnego wykonywania funkcji diagnostycznych i pozycjonowania pojazdu w terenie. Istnieje możliwość zastosowania takich urządzeń do kontroli położenia pojazdów komunikacji miejskiej. Została złożona oferta współpracy przy realizacji takiego tematu z miastem Gdynia w ramach projektu europejskiego Tellus-Civitas.

Opracowana w ramach realizacji tematu struktura systemu monitoringu, bazująca na centralnie umieszczonym serwerze będzie wykorzystywana również do realizacji innych tematów związanych z zastosowaniem technologii internetowych do kontroli rozproszonych obiektów. Planuje się na przykład zastosowanie tej struktury do kolejnej generacji systemu monitoringu i sterowania przepompowni melioracyjnych.

Modułowa architektura oprogramowania serwera oraz użyty protokół transmisyjny IEDCP przeszły pomyślnie próby wdrożeniowe i mogą zostać przyjęte jako platforma do opracowywania kolejnych aplikacji.

## LITERATURA

1. Eckel B.: „Thinking in JAVA 3rd Edition”, <http://www.mindview.net/Books/TIJ/>.
2. Java 2 Platform, Standard Edition, White Papers: <http://java.sun.com/docs/white/index.html>, Sun Microsystems, Inc.

3. Java 2 Platform, Standard Edition, v 1.4.2 API Specification: <http://java.sun.com/j2se/1.4.2/docs/api/index.html>, Sun Microsystems, Inc.

*Rękopis dostarczono, dnia 8.12.2004 r.*

**Opiniował: dr inż. Stefan Wójtowicz**

## THE REMOTE DIAGNOSTIC SYSTEM FOR TRACTION VEHICLES

Arkadiusz KOMKOWSKI, Arkadiusz ADOLPH  
Piotr ZAPRAWA, Andrzej MŁYŃSKI

**ABSTRACT** *The construction and operation of the remote diagnostic system for traction vehicles are presented in the paper. The system consists of two versions of monitoring devices, software residing on central server machine with relational data base and visualization application. The possibilities of further development of the system and its areas of application are also discussed.*



**Mgr inż. Arkadiusz Komkowski** urodził się w 1973r. Studia na Wydziale Elektrotechniki i Automatyki Politechniki Gdańskiej ukończył w 1998r., gdzie uzyskał tytuł magistra inżyniera o specjalności Urządzenia Elektryczne. Od 1999r. jest asystentem w Oddziale Gdańskiego IEL. Obecnie zajmuje się konstruowaniem urządzeń mikroprocesorowych, układów energoelektronicznych oraz zagadnieniami zastosowania sterowania bezprzewodowego w energoelektronice.

**Mgr inż. Arkadiusz Adolph** urodził się w 1974r. Studia na Wydziale Elektrotechniki i Automatyki Politechniki Gdańskiej ukończył w 2003r., gdzie uzyskał tytuł magistra inżyniera ze specjalnością Napęd Elektryczny i Energoelektronika. Od 2004r. jest asystentem w Pracowni Energoelektroniki i Sterowania Oddziału Gdańskiego IEL. Obecnie zajmuje się opracowywaniem oprogramowania dla systemów sterowania i monitorowania rozproszonych obiektów elektrycznych i energetycznych za pośrednictwem sieci rozległych.



**Mgr inż. Piotr Jerzy Zaprawa**, ur. 1975 roku. Ukończył Wydział Elektrotechniki i Automatyki Politechniki Gdańskiej na specjalności Napędu Elektrycznego i Energoelektroniki. Od 1999 roku pracuje w Oddziale Instytutu Elektrotechniki w Gdańsku na stanowisku asystenta w Pracowni Napędów i Sterowania.

**Andrzej Młyński**, ur. 1976 roku. Ukończył Technikum Budowlane na podbudowie Zasadniczej Szkoły Budowlanej w Gdańsku woj. pomorskie o trzy letnim okresie nauczania w zawodzie technika budownictwa. Od 1998 roku pracuje w Oddziale Instytutu Elektrotechniki w Gdańsku.

