

Roman SEREDYŃSKI, Michał MOMOT  
Instytut Techniki i Aparatury Medycznej ITAM  
Alina MOMOT  
Politechnika Śląska, Instytut Informatyki

## SYSTEM DO PROJEKTOWANIA PROTOKOŁÓW SIECI BEZPRZEWODOWYCH ORAZ SYMULACJI ICH DZIAŁANIA

**Streszczenie.** Artykuł opisuje system służący do wspomagania projektowania protokołów komunikacyjnych dla urządzeń przeznaczonych do funkcjonowania w sieci bezprzewodowej. Za pośrednictwem tego systemu można też nawiązać połączenie z prostą bazą danych, co umożliwi gromadzenie szczegółowych informacji o projektowanych sieciach i protokołach. Możliwe jest również przechowywanie danych będących rezultatami symulacji wykonywanych za pomocą tego systemu.

**Słowa kluczowe:** projektowanie protokołów, sieci bezprzewodowe

## SYSTEM FOR DESIGNING OF WIRELESS NETWORKS PROTOCOLS AND SIMULATING OF THEIR PERFORMANCE

**Summary.** This paper describes a system which supports designing of communication protocols for devices working in a wireless network. This system also allows connecting to a simple database which enables collecting detailed information about the designed networks and protocols. It is also possible to store data that are results of simulations performed using this system.

**Keywords:** protocols design, wireless network

### 1. Wprowadzenie

Jednym z istotnych i podstawowych warunków rozwoju nauki jest dostęp do źródeł wiedzy i najnowszych wyników prac badawczych. Dostęp ten możliwy jest dzięki komputerom połączonym siecią komputerową z odpowiednimi bazami danych. Sieć taka umożliwia dostęp do zasobów i dzielenie się nimi [4]. Jeśli do całego procesu gromadzenia i udostępniania

wiedzy, a nie tylko do poszczególnych jego etapów wykorzystano współczesną technikę komputerową, to znacznie szerszy krąg odbiorców zyska dostęp do źródeł informacji i wiedzy [5].

W ostatnich latach można zaobserwować gwałtowny rozwój sieci bezprzewodowych oraz wzrost ich znaczenia w wielu dziedzinach, takich jak edukacja, usługi finansowe, transportowe czy też medyczne [7]. Można też zauważyć, że w pewnych warunkach zastosowanie sieci bezprzewodowych staje się koniecznością, na przykład gdy urządzenia komunikujące się ze sobą rozmieszczone są na dużym obszarze i stworzenie infrastruktury przewodowej wiązałoby się ze zbyt dużym kosztem. Warto tu podkreślić, że zasięg sieci bezprzewodowej w zależności od warunków fizycznych może wynieść nawet 100 km, a w przypadku potrzeby przesyłania dużych ilości informacji w krótkim czasie można wykorzystać łącza laserowe, które umożliwiają uzyskanie bardzo dużych prędkości przesyłu. Innym przykładem warunków, które zwykle wymuszają użycie łączności bezprzewodowej, jest wymaganie mobilności komunikujących się ze sobą urządzeń. Związane jest to z upowszechnianiem się urządzeń mobilnych, takich jak na przykład komputery przenośne wyposażone w urządzenia umożliwiające nadawanie/odbieranie informacji w sposób bezprzewodowy. Sieci bezprzewodowe znajdują również zastosowanie wszędzie tam, gdzie infrastruktura budynków, kanalizacji i instalacji podziemnych utrudnia połączenia kablowe lub znacznie zwiększa ich koszt. Ponadto regiony zagrożone sejsmicznie również wymuszają stosowanie łącz bezprzewodowych, co najmniej w roli połączenia awaryjnego. Rozważając zastosowania łączności bezprzewodowej, warto wspomnieć o łączności optycznej, która staje się szczególnie użyteczna w środowiskach o dużym poziomie zakłóceń generowanych przez silne pola elektromagnetyczne. Przykładami takich środowisk są duże skupiska urządzeń elektrycznych, takie jak fabryki lub oddziały intensywnej opieki medycznej.

Do projektowania sieci, w szczególności bezprzewodowych, używa się często oprogramowania wspomagającego tę pracę. Istotnym etapem procesu projektowania sieci jest wirtualna symulacja jej działania. W tym celu stosowane są narzędzia takie, jak na przykład środowisko symulacji OMNeT++ [1], mające jednak ograniczoną funkcjonalność w zakresie projektowania i symulowania systemów czasu rzeczywistego. Innym tego typu wirtualnym środowiskiem, umożliwiającym specjalistom zasymulowanie pracy każdej infrastruktury sieciowej jako całości, włącznie z technologiami, protokołami i konkretnymi urządzeniami oraz aplikacjami używanymi przez końcowych użytkowników, jest OPNET Modeler [2]. Innym podejściem jest tworzenie i wykorzystywanie specjalizowanych narzędzi do symulacji działania konkretnych typów sieci. Wyróżnić tu można specjalizację w kierunku zastosowania takiej sieci, na przykład sieci telemedyczne [3], lub też specjalizację w kierunku rodzaju wchodzących w jej skład urządzeń, typów łączy itp.

Artykuł opisuje system służący do wspomaganie projektowania protokołów komunikacyjnych i urządzeń przeznaczonych do działania w sieci bezprzewodowej. Opracowane narzędzie posiada interpreter prostego pseudokodu służącego do opisu protokołu, umożliwia symulację działania projektowanej sieci oraz podgląd zawartości struktur danych i wysyłanych ramek. Wykorzystując to narzędzie można określić zasoby potrzebne do implementacji danego protokołu, następnie dysponując fizycznym prototypem urządzenia komunikującego się można przetestować działanie zaprojektowanego protokołu w komunikacji bezprzewodowej, a także przewodowej. Dzięki możliwości podłączenia modułu bezprzewodowego do komputera osobistego za pomocą interfejsu szeregowego RS232 możliwe jest testowanie tego modułu działającego w wirtualnej sieci już na poziomie projektowania protokołu komunikacyjnego. Dodatkowo system może nawiązać połączenie z bazą danych, co umożliwia gromadzenie informacji o projektowanych sieciach i protokołach. Możliwe jest również przechowywanie wyników symulacji wykonywanych w środowisku wirtualnym lub rzeczywistym.

## 2. Problemy przy projektowaniu sieci komputerowych

Podczas procesu projektowania sieci komputerowych należy wziąć pod uwagę następujące zagadnienia [6]:

- a) mechanizm identyfikujący nadawcę i odbiorcę:
  - forma adresowania,
- b) reguły transferu danych:
  - transmisja jednokierunkowa lub dwukierunkowa,
  - liczba kanałów logicznych oraz ich priorytety,
- c) kontrola błędów:
  - kody wykrywania błędów i kody korekcyjne,
  - potwierdzenie odebrania wiadomości,
- d) kolejność przesyłania wiadomości:
  - numerowanie porcji danych,
  - buforowanie porcji danych przychodzących w niewłaściwej kolejności,
- e) sterowanie przepływem danych:
  - przesył informacji kontrolnych,
  - ustalanie szybkości transmisji,
- f) ograniczenie rozmiaru porcji danych:
  - podział, transmisja i scalanie porcji danych,
- g) multipleksowanie lub demultipleksowanie połączeń,

h) wybór trasy.

Zagadnieniem ściśle związanym z projektowaniem sieci komputerowych jest projektowanie protokołów komunikacyjnych rozumianych jako reguły określające zasady wymiany informacji w postaci ramek. Wiąże się z tym opracowanie modelu ramki, czyli struktury danych, która musi zawierać informacje pozwalające na przesłanie ich poprzez sieć do ich miejsca przeznaczenia i odczytanie ich przez odbiorcę w niezmienionym stanie. Podstawowymi polami, które powinna zawierać ramka, są:

- znaki początku ramki,
- adres nadawcy, identyfikujący źródło danych,
- adres odbiorcy, identyfikujący adresata,
- pole danych o odpowiedniej długości,
- dane kontrolne, np. suma kontrolna lub wynik cyklicznej kontroli nadmiarowej CRC,
- znaki końca ramki.

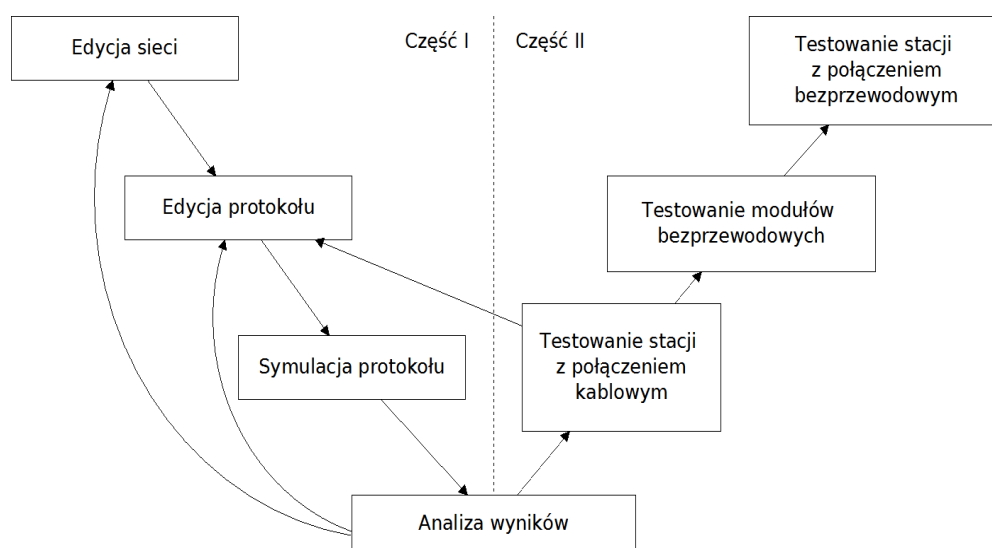
Niezbędnym elementem projektowania protokołu jest ustalenie pól i rozmiaru (lub jego zakresu) ramki, które muszą stać się standardem w sieci. Istotnym elementem jest w tym przypadku stosunek ilości narzutu informacji do ilości danych i określa on skuteczność transmisyjną protokołu.

### **3. Schemat projektowania i testowania działania protokołów**

Podczas opracowywania systemu zaproponowano następujący schemat projektowania i testowania protokołów, obejmujący również symulację ich działania w środowisku wirtualnym i rzeczywistym (rys. 1).

Na schemacie projektowania i testowania można wyróżnić dwie części: na pierwszą składają się etapy projektowania, symulacji i testowania protokołu tylko w środowisku wirtualnym, część druga obejmuje testowanie i uruchamianie projektowanego urządzenia fizycznego.

Na etapie edycji sieci należy zaplanować maksymalną liczbę stacji, ich rozmieszczenie oraz zasięg. Etap edycji protokołu jest głównym etapem projektowania. Biorąc pod uwagę przeznaczenie protokołu, należy opracować strukturę pól ramek i przesyłanych danych oraz algorytm funkcjonalny. Po dokonaniu symulacji działania protokołu w zaprojektowanej sieci i analizie wyników określa się, czy działanie protokołu jest prawidłowe. W przypadku jakichś nieprawidłowości należy powrócić do etapu edycji protokołu lub nawet etapu edycji sieci, jeśli konieczna jest modyfikacja grafu połączeń.



Rys. 1. Etapy projektowania protokołów sieciowych  
Fig. 1. Stages of network protocols designing

Jeśli wynik symulacji działanie protokołu nie wykaże żadnych nieprawidłowości, należy zaimplementować protokół w układzie uruchomieniowym. Etap testowania z połączeniem kablowym ma za zadanie określić prawidłowe działanie protokołu w projektowanym urządzeniu. Jeśli na tym etapie okaże się, że protokół wymaga modyfikacji, należy powrócić do etapu edycji protokołu. Etap testowania modułów bezprzewodowych wymaga wykorzystania programu terminala, za pomocą którego testowany jest przesył danych przez moduły bezprzewodowe z jednego terminala do drugiego. W końcowym etapie testuje się prototypy urządzeń połączone bezprzewodowo z wirtualną siecią.

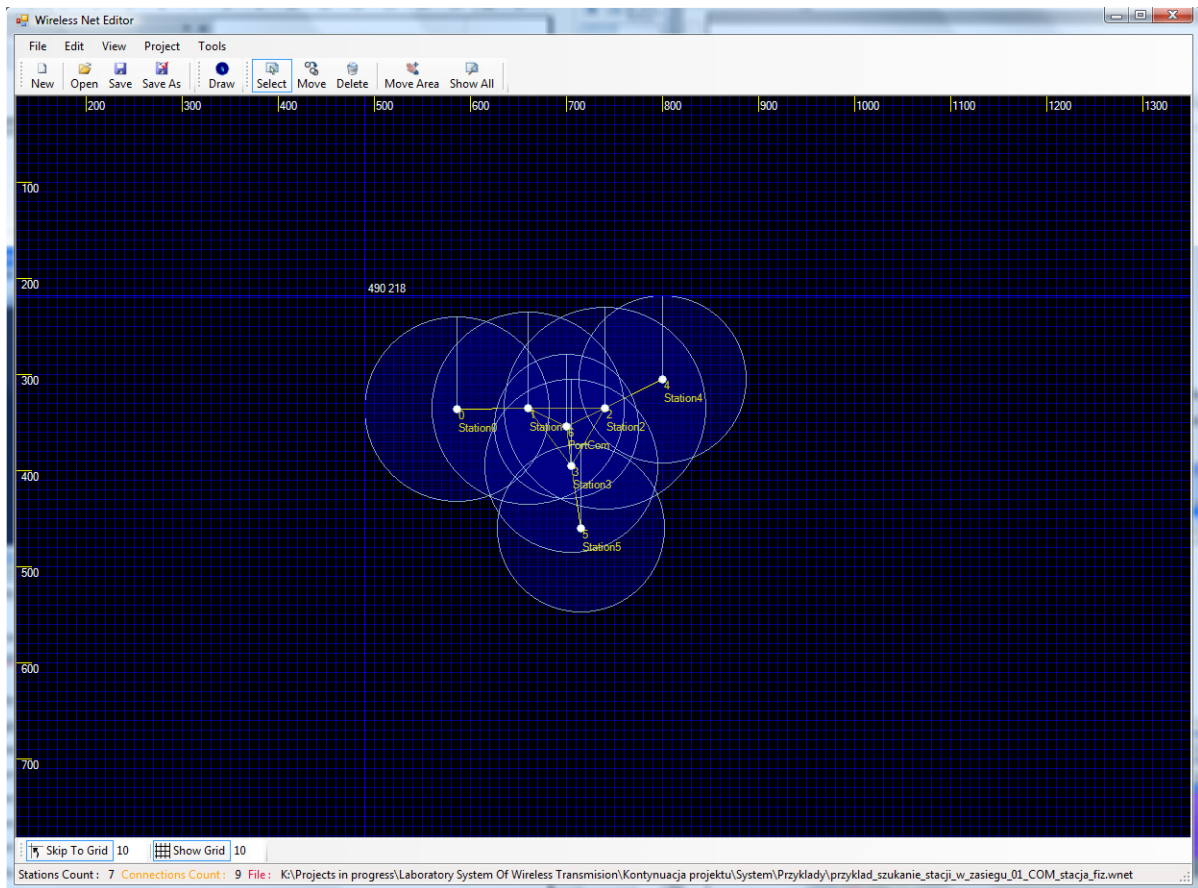
#### 4. Opis środowiska do edycji sieci i protokołów

System służący do wspomaganie projektowania protokołów komunikacyjnych i urządzeń przeznaczonych do działania w sieci bezprzewodowej został zaimplementowany w języku programowania C#, środowisku Visual Studio.

W ramach etapu edycji sieci można graficznie dodawać stacje bezprzewodowe, określając ich zasięg poprzez średnicę okręgu (zakłada się, że stacja będzie mogła komunikować się z każdą stacją znajdującą się w jej zasięgu). W ten sposób powstaje graf połączeń sieci (rys. 2). Naniesione stacje można zaznaczać, zmieniać ich pozycję lub usuwać.

Przejdźcie do kolejnego etapu projektowania, a mianowicie edycji protokołu, następuje po wybraniu opcji **Project** → **Protocol Editor**. W oknie edytora protokołu (rys. 3) można wyróżnić podgląd sieci, w którym można dokonać wyboru stacji, jej protokołu i właściwości. Dla protokołu wybranej stacji widoczne są nazwy pól ramki oraz ich wartości (pole **Frames**), nazwy wektorów danych dołączonych do ramki (pole **DATA**), nazwy zmiennych i stałych

wraz z ich wartościami (pola Variables i Constants) oraz podgląd pewnego obszaru pamięci wybranej stacji (pole Tables). W oknie edytora widoczne są również wyniki przeprowadzonych symulacji w postaci przesłanych ramek oraz okno edytora pseudokodu definiującego działanie protokołu.



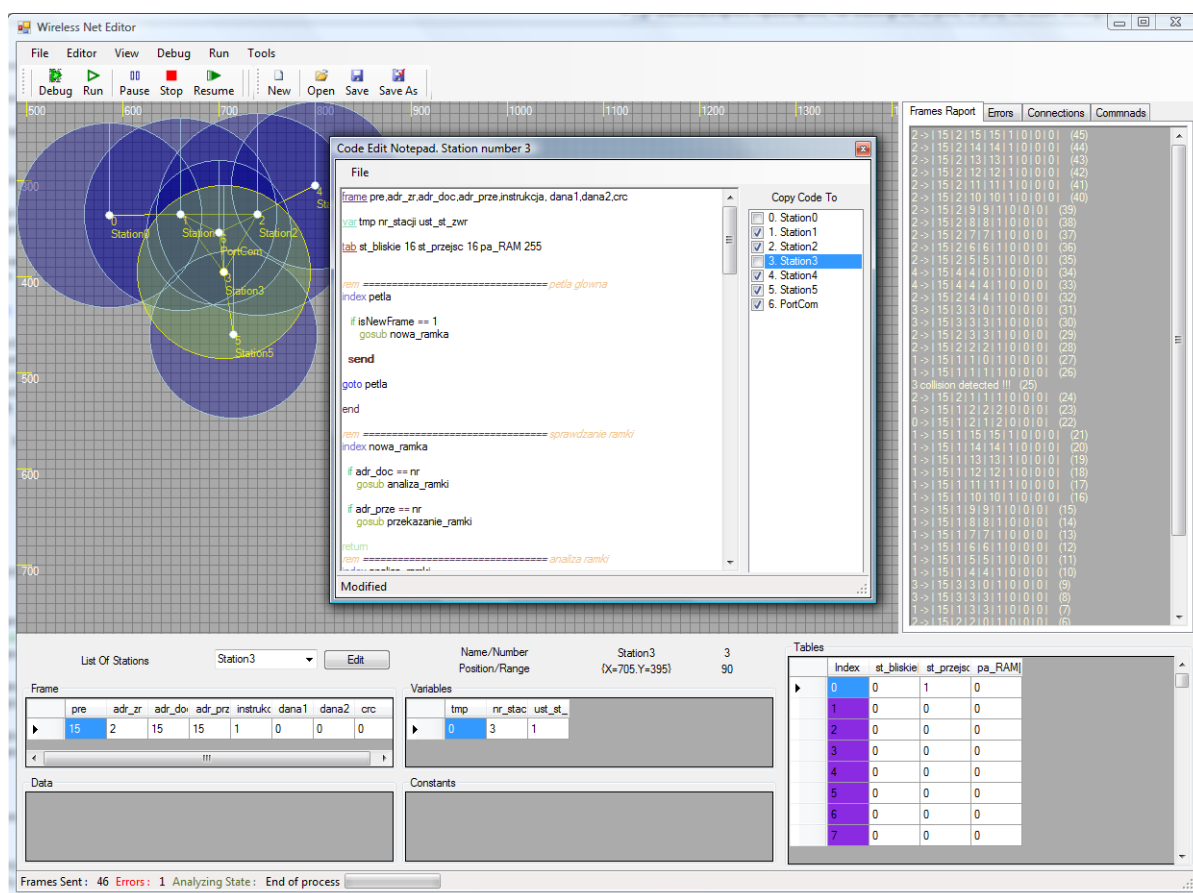
Rys. 2. Edytor sieci  
Fig. 2. Network editor

Symulacja działania każdej stacji odbywa się w oddzielnym wątku programowym. Warto podkreślić, że definiowanie działania protokołu opisuje się za pomocą pseudokodu, specjalnie zaprojektowanego do tego celu, przy użyciu następujących słów kluczowych i symboli:

- Rem – komentarz,
- Frame – deklaracja ramki,
- Data – deklaracja danych dołączonych do ramki,
- Var – deklaracja zmiennej,
- Tab – deklaracja tablicy,
- Const – deklaracja stałej,
- Index – deklaracja etykiety,
- Random – wartość wyznaczona pseudolosowo,
- = – przypisanie wartości,

- If – instrukcja warunkowa,
- Goto – skok do etykiety,
- Gosub – skok do podprogramu,
- Return – powrót z podprogramu,
- Send – wysłanie ramki,
- Recv – odebranie ramki,
- Waitms – wstrzymanie działania przez okres czasu podany w milisekundach,
- End – koniec działania protokołu (wątek kończy się).

Oprócz powyższych dodano jeszcze możliwość prostej komunikacji z bazą danych, w postaci funkcji DBconnect (nawiązanie połączenia z bazą danych) oraz DBadd (zapis rekordu w bazie). Może to służyć, na przykład gdy istnieje potrzeba zastosowania sieci bezprzewodowej w zintegrowanym systemie pomiarowym, umożliwia to w takim przypadku od razu projektowanie protokołu z dostępem do bazy danych, w której te pomiary będą przechowywane.



Rys. 3. Edytor i symulator działania protokołu  
Fig. 3. Protocol editor and simulator

Istnieje możliwość tworzenia raportu z przeprowadzonej symulacji. Raport z symulacji składa się z trzech stron. Pierwsza prezentuje rysunek - graf połączeń sieci. Następna przed-

stawia dane - aktualne wartości pewnych parametrów, które w chwili generowania raportu były widoczne na ekranie. Ostatnia strona obejmuje wykaz wszystkich przesyłanych ramek wraz z identyfikatorami wysyłających je stacji. Podgląd wygenerowanego raportu jest przedstawiany w oddzielnym oknie, które umożliwia podgląd wszystkich stron przygotowanych do wydruku. Można w tym oknie dokonać wyboru drukarki, jej ustawień oraz wydrukować raport. Możliwy jest również zapis wygenerowanego raportu do pliku graficznego w postaci bitmapy.

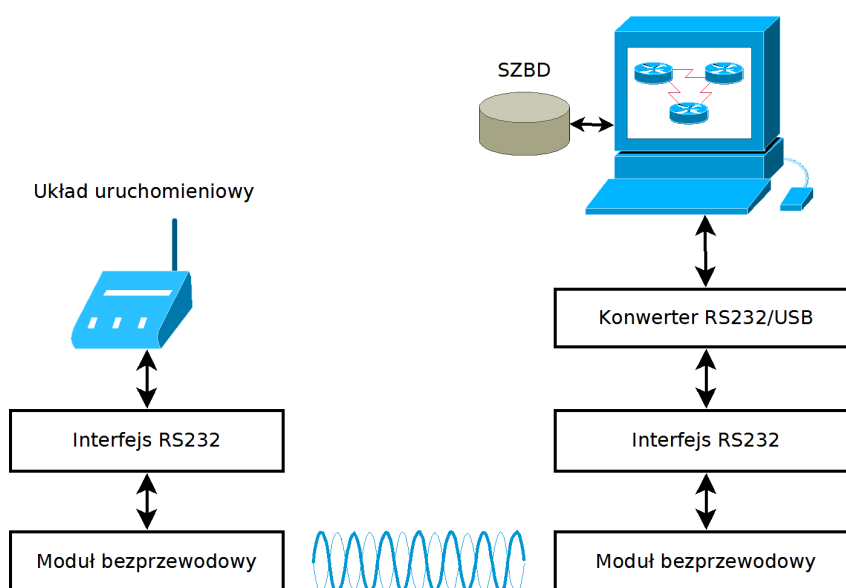
Aplikacja służąca do edycji sieci i protokołów wyposażona jest również w opcję łączenia się z prostą bazą danych obejmującą m.in. tabelę, w której przechowywane są informacje, na podstawie których można wygenerować w programie graf połączeń projektowanej sieci oraz tabelę, w której przechowywane są informacje pozwalające na zdefiniowanie protokołu, którego działanie ma być symulowane w programie.

## 5. Testowanie

Jako przykładowy element sieci bezprzewodowej został wykonany układ elektroniczny, oparty na mikroprocesorze AT89C4051. Układ został wyposażony w zegar czasu rzeczywistego PCF8583. Zegar ten posiada pamięć RAM, w której można przechowywać wyniki pomiarów lub inne wartości. Do prezentacji odebranych ramek, wyników pomiarów, czasu i innych wartości zastosowany został alfanumeryczny 16-znakowy (w dwóch liniach) wyświetlacz LCD. Układ wyposażono również w interfejs RS232C, do połączenia z komputerem za pomocą przewodu, konwertera RS232↔USB lub modułu bezprzewodowego (rys. 4). W tak przygotowanym układzie można testować zaprojektowany wcześniej protokół sieciowy za pomocą połączenia przewodowego. Opracowano i wykonano również bufor bramkowy z interfejsem RS232, służący do podłączenia modułu bezprzewodowego w sposób zaprezentowany na rysunku 4.

Jako narzędzie systemu służące do testowania modułów bezprzewodowych został opracowany również program terminala (rys. 5), który posiada możliwość prezentacji przesyłanych danych w postaci znakowej, bitowej, bajtowej i heksadecymalnej. Program umożliwia również wybór numeru portu i prędkości transmisji. W przypadku transmisji bezprzewodowej program pozwala na kodowanie bitów, aby uzyskać odpowiedni współczynnik wypełnienia przesyłanego sygnału. W menu programu dostępne są opcje otwarcia i zamknięcia połączenia z portem COM. Istnieje możliwość odczytu i ustawienia sygnałów sterowania portem (RTS, CTS, DTR, DSR, DCD). Pasek statusu programu prezentuje numer podłączonego portu, liczbę niewysłanych lub błędnie odebranych bajtów, tryby wprowadzania i odczytu danych oraz kodowanie bitów.





Rys. 4. Schemat środowiska testowego

Fig. 4. Test environment schema

Za pomocą programu terminala testowano zaprojektowane protokoły na poziomie przesyłania danych bezpośrednio pomiędzy dwoma modułami bezprzewodowymi: testowano kodowanie bitów oraz prędkość transmisji w obu kierunkach.

Aby móc zrealizować następny etap testowania systemu bezprzewodowego, konieczne było zaprogramowanie mikrokontrolera (będącego częścią opisanego uprzednio przykładowego elementu sieci bezprzewodowej - prototypu stacji). Program mikrokontrolera ma pozwalać na realizację połączenia przy użyciu projektowanego protokołu i testowanie komunikacji ze stacją poprzez połączenie przewodowe oraz bezprzewodowe.

Aby sprawdzić połączenie układu mikroprocesorowego z portem COM/USB wykorzystano program terminala, dzięki któremu można było opracować i przetestować poszczególne funkcje protokołu w układzie mikroprocesorowym. Testowano odpowiedzi układu na poszczególne ramki, a także działanie programu mikrokontrolera dla różnych danych wejściowych, w tym również generowanych losowo. Poprawność działania układu kontrolowano poprzez obserwację informacji pojawiających się na wyświetlaczu alfanumerycznym LCD. Następnie w podobny sposób przetestowano przesyłanie danych z wykorzystaniem połączenia bezprzewodowego.

Po zweryfikowaniu poprawnego działania wszystkich elementów wchodzących w skład rzeczywistego środowiska testowego (rys.6) przeprowadzono eksperymenty dotyczące projektowania protokołów sieci bezprzewodowej oraz testowano działanie fizycznej stacji z połączeniem przewodowym lub bezprzewodowym wraz z innymi (wirtualnymi) stacjami działającymi w wirtualnej sieci utworzonej na komputerze z wykorzystaniem środowiska opisanego w punkcie 4 niniejszego artykułu. Stacja fizyczna działała w wirtualnej sieci jako stacja



macji o projektowanych sieciach i protokołach. Możliwe jest również przechowywanie danych będących rezultatami symulacji wykonywanych za pomocą tego systemu.

Na obecnym etapie prac możliwe jest projektowanie i testowanie komunikacji bezprzewodowej z wykorzystaniem opracowanego wirtualnego symulatora oraz jednego egzemplarza zaprojektowanego elementu sieci. Na podstawie przeprowadzonych eksperymentów stwierdzono poprawne działanie fizycznej stacji z połączeniem przewodowym lub bezprzewodowym wraz z innymi (wirtualnymi) stacjami działającymi w wirtualnej sieci utworzonej na komputerze z wykorzystaniem opracowanego systemu.

Symulacja pracy sieci w opracowanym systemie opiera się na wątkach programowych, dzięki czemu można symulować jej pracę w czasie rzeczywistym i badać działanie zaprojektowanego fizycznego urządzenia jako jednej z wirtualnych stacji w wirtualnej sieci bezprzewodowej. Każdemu elementowi sieci odpowiada osobny wątek w aplikacji, do którego przypisany jest algorytm opracowany za pomocą prostego pseudokodu, którego interpreter znajduje się w klasie stacji.

Planuje się również rozwinięcie aplikacji poprzez dodanie możliwości ustalania liczby kanałów logicznych tak, aby stacje mogły nadawać równocześnie w kilku kanałach logicznych z różnymi częstotliwościami, oraz możliwości pracy w pełnym trybie duplexowym, gdyż aktualnie interfejs układowy umożliwia testowanie modułów jedynie w trybie półduplexowym.

## BIBLIOGRAFIA

1. Chen J. i in.: The Development of A Realistic Simulation Framework with OMNeT++. 2008 Sec. Intern. Conf. Future Generation Communication and Networking. DOI 10.1109/FGCN.2008.176.
2. Krupanek B.: Badania symulacyjne bezprzewodowego systemu eksperymentalnego opartego na standardzie ZigBee przy wykorzystaniu środowiska OPNET Modeler. XI International PhD Workshop OWD, Gliwice 2009, s. 312÷317.
3. Nadkar T., Unnikrishnan S.: A Bluetooth-based Telemedicine Model: Simulation Approach. Proc. SPIT-IEEE Colloquium and International Conference, Vol. 3, No. 82, Mumbai, India, 2008.
4. Sportach M.: Sieci komputerowe – Księga eksperta. Helion, Gliwice 1999.
5. Świeczkowski J.: Naukowe bazy danych i publikacji STN. Gazeta Uniwersytecka. Miesięcznik Uniwersytetu Śląskiego w Katowicach. Vol. 2, No. 11, 1993.
6. Tanenbaum A.S.: Sieci komputerowe. Helion, Gliwice 2004.
7. Zieliński B.: Bezprzewodowe sieci komputerowe. Helion, Gliwice 2000.

Recenzenci: Dr hab. inż. Zbigniew Huzar, prof. Pol. Wrocławskiej  
Dr inż. Dariusz Mrozek

Wpłynęło do Redakcji 22 stycznia 2010 r.

### **Abstract**

Access to sources of knowledge and latest research results is made possible through a network of computers connected to the relevant computer database. In recent years, one may experience a rapid development of wireless networks and their growing importance in many fields such as education, financial, transport or medical services. And in some circumstances use of wireless networks is becoming a necessity.

Developed system is a tool allowing to specify the basic parameters of the network, define the protocol and test it as well as guide the establishment of its complexity, the demand for program memory and operating memory. This system also allows connecting to a simple database which enables collecting detailed information about the designed networks and protocols. It is also possible to store data that are results of simulations performed using this system.

### **Adresy**

Roman SEREDYŃSKI: Instytut Techniki i Aparatury Medycznej, ul. Roosevelta 118,  
41-800 Zabrze, Polska, roman.seredynski@itam.zabrze.pl

Michał MOMOT: Instytut Techniki i Aparatury Medycznej, ul. Roosevelta 118,  
41-800 Zabrze, Polska, michal.momot@itam.zabrze.pl.

Alina MOMOT: Politechnika Śląska, Instytut Informatyki, ul. Akademicka 16,  
44-101 Gliwice, Polska, alina.momot@polsl.pl.