

Paweł DYMORA, Mirosław MAZUREK, Piotr MACIĄG

UNIVERSITY OF TECHNOLOGY,
Al. Powstańców Warszawy 12, 35-959 Rzeszów

System monitorujący pracę bezprzewodowej sieci sensorowej oparty na przestrzennych typach danych

Dr inż. Paweł DYMORA

Absolwent Wydziału Elektrotechniki i Informatyki Politechniki Rzeszowskiej. stopień doktora w dyscyplinie informatyki zdobył w 2009 roku. Zajmuje się problematyką projektowania, eksploatacji i optymalizacji sieci komputerowych w szczególności sieci sensorowych.



e-mail: Pawel.Dymora@prz.edu.pl

Dr inż. Mirosław MAZUREK

Absolwent Wydziału Elektrotechniki i Informatyki Politechniki Rzeszowskiej. stopień doktora w dyscyplinie informatyki zdobył w 2009 roku. Zajmuje się problematyką eksploatacji i optymalizacji systemów operacyjnych oraz zagadnieniami bezpieczeństwa sieci i systemów komputerowych.



e-mail: Miroslaw.Mazurek@prz.edu.pl

Inż. Piotr MACIĄG

Absolwent Wydziału Elektrotechniki i Informatyki Politechniki Rzeszowskiej. Zainteresowania naukowe dotyczą wykorzystania sieci sensorowych w projektowaniu inteligentnych budynków, systemów monitoringowych czy automatyki domowej. Autor szeregu sterowników wykorzystujących układy mikroprocesorowe.



e-mail: Piotr.Maciag@prz.edu.pl

Streszczenie

Sieci sensorowe znajdują zastosowanie nie tylko w budowie systemów ostrzegających przed zagrożeniami naturalnymi, ale również w projektowaniu inteligentnych budynków czy specjalistycznej aparatury medycznej. System sensorowy składa się zazwyczaj z pewnej liczby węzłów pomiarowych, których zadaniem jest dostarczanie informacji o pewnych pożądanych parametrach. Parametry te mogą mówić np. o stanie poziomu wód rzecznych, stężeniu niebezpiecznych gazów czy występowaniu substancji niebezpiecznych dla środowiska naturalnego. Dobrze zaprojektowana sieć sensorowa powinna cechować się redundancją, możliwością routowania danych pomiędzy węzłami pomiarowymi oraz wykorzystaniem odpowiedniego mechanizmu dostępu do pasma komunikacyjnego. Istotną częścią systemu sensorowego stanowi aplikacja monitorująca oraz wizualizująca stan sensorów. W niniejszym artykule przedstawiono oryginalne podejście do budowy sieci sensorowej i wizualizacji jej działania z użyciem przestrzennych typów danych.

Słowa kluczowe: przestrzenne typy danych, mikroprocesor AVR, protokoły bezprzewodowych sieci sensorowych.

Wireless sensor network monitoring system based on spatial data types

Abstract

Because of their low cost, high availability and redundancy, wireless sensor networks are used in wide variety of applications. Today building monitoring systems which can inform about and prevent natural disasters, designing smart houses and supporting contemporary medicine are achieved by using sensor networks. When implementing a wireless sensor network, designer have to elaborate protocol which can be used to provide communication between sensors, user interface for monitoring and controlling system and database in which should be stored data from sensors. This paper presents capabilities of sensor network and spatial data types in designing smart house.

Keywords: spatial data types, microcontrollers AVR, wireless sensor networks protocols.

1. Wprowadzenie

W chwili obecnej jedną z najbardziej dynamicznie rozwijających się dziedzin informatyki jak również i mikroelektrotechniki są sieci sensorowe. Dzięki rozwojowi techniki, a w szczególności miniaturyzacji, pojawiły się zupełnie nowe możliwości w projektowaniu układów elektronicznych, a tym samym systemów sterowania tj. systemy monitoringu procesów produkcyjnych czy inteligentne budynki. Inteligentnie zaprojektowany dom potrafi wnieść w nasze życie nową jakość. Dzięki zdalnemu sterowaniu urządzeniami, zaawansowanym systemom bezpieczeństwa czy technologiom zwiększającym jakość odpoczynku, inteligentne domy zyskują coraz większą popularność [1, 7].

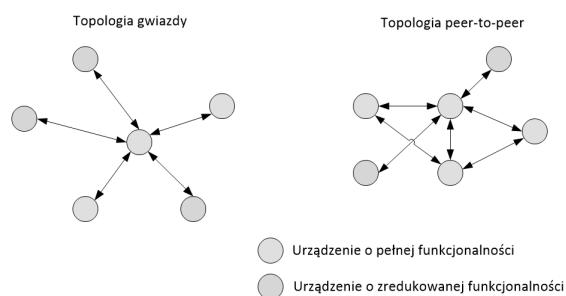
System monitorujący pracę inteligentnego budynku składa się zazwyczaj z kilku części. Po pierwsze istnieje sieć sensorów czyli urządzeń monitorujących i sterujących pracą budynku. Sensory tworzą sieć, poprzez którą następuje wymiana danych. Wykorzystywane są sieci przewodowe i bezprzewodowe chociaż te drugie są bardziej preferowane ze względu na swoją nieinwazyjność, nie zaburzanie istniejącej struktury budynku. Wśród sensorów istnieje zazwyczaj jeden wyróżniony układ, który pełni rolę koordynatora sieci oraz zbiera dane z poszczególnych czujników. Układ taki cechuje się zazwyczaj zwiększonymi możliwościami obliczeniowymi oraz większymi zasobami pamięciowymi. Jeśli taki koordynator spełnia ponadto rolę łącznika do innych sieci to możemy mówić o bramie sieciowej [1, 2, 3, 5]. W niniejszej artykule zostanie zaprezentowana kompletna implementacja (projekt, realizacja, oprogramowanie) systemu inteligentnego budynku w tym bramy sieciowej, która odbiera dane od węzłów pomiarowych i następnie przy pomocy odpowiednich aplikacji umieszcza te dane w bazie danych. Ostatnią część systemu stanowi interfejs, za pomocą którego użytkownik może monitorować stan budynku oraz sterować poszczególnymi urządzeniami peryferyjnymi. Intensywnie rozwijające się przestrzenne typy danych idealnie nadają się do zdefiniowania schematu budynku oraz prezentacji stanu poszczególnych sensorów, co jest wyróżnikiem stworzonej aplikacji [4].

2. Koncepcje projektowania bezprzewodowych sieci sensorowych

Istnieją rozmaite standardy opisujące działanie bezprzewodowych sieci sensorowych. Większość z nich określa sposób dostępu do nośnika danych (którym są fale radiowe) oraz definiuje sposoby adresacji, struktury ramki czy też metody uwierzytelniania i kontroli poprawności danych. Standard IEEE 802.15.4 opisuje stos protokołów, za pomocą których można sterować pracą nadajników i odbiorników radiowych. W standardzie tym zostały dokładnie określone funkcje jakie mogą pełnić poszczególne sensory, sposoby formowania ramki czy też tworzenia połączeń pomię-

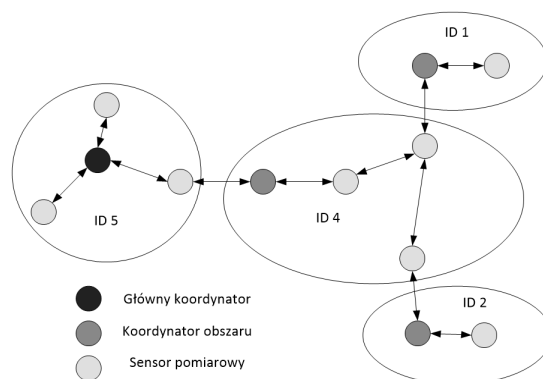
dzy węzłami pomiarowymi. Wszystkie te funkcjonalności dotyczą jednak tylko warstwy fizycznej oraz warstwy dostępu do łącza. Wyższe warstwy, takie jak warstwa sieciowa odpowiadająca za routowanie danych czy warstwa aplikacji udostępniająca interfejs programowy pozostają do zdefiniowania w przypadku budowania nowej technologii. Dobrze obrazuje to stos protokołów technologii ZigBee. W technologii tej, dwie pierwsze warstwy: fizyczna i dostępu do łącza zostały określone według reguł standardu 802.15.4, ale już zachowania w warstwie sieciowej zostały zdefiniowane przez twórców technologii ZigBee. Przed problemem opracowania wspomnianego stosu protokołów 802.15.4 staje twórca, który decyduje się na oprogramowanie „od podstaw” modułów radiowych pracujących w sieci sensorowej. Problem ten został poruszony w niniejszym artykule, gdzie do implementacji komunikacji w budowanym systemie sensorowym wykorzystano moduły RFM12, których twórcy określają tylko podstawowe mechanizmy transmisji danych, takie jak: format ramki, wybór kanału transmisji czy szybkość transmisji [1, 2, 7, 8, 10]. W sieciach sensorowych wyróżniamy dwa typy urządzeń. Urządzenia o zredukowanej funkcjonalności (RFD, ang. Reduced Function Device) przeznaczone do zbierania informacji z poszczególnych czujników czy też sterowania układami peryferyjnymi. Mogą to być np. urządzenia odczytujące temperaturę, wykrywające obecność szkodliwych gazów czy sterujące pracą rolet lub bram. Urządzenia tego typu zazwyczaj są zasilane za pomocą baterii i większość czasu są w stanie uśpienia. Ponadto, urządzenia RFD nie mogą routować danych pomiędzy sensorami. Potrafią jedynie wysyłać dane do sensorów o pełnej funkcjonalności. Drugi typ urządzeń stanowią urządzenia o pełnej funkcjonalności (FFD, ang. Full Function Device). Urządzenia tego typu, oprócz wykonywania czynności standardowych dla sensorów, czyli odczytu danych i sterowania, mogą routować dane pomiędzy innymi urządzeniami FFD oraz RFD. W sieci sensorowej wyróżniamy zazwyczaj jeden główny węzeł, który pełni rolę koordynatora sieci. Zadaniem koordynatora jest sterowanie dostępem do łącza lub wykrywanie nowo dodanych węzłów. Jest to urządzenie wyposażone w większe zasoby obliczeniowe niż typowe urządzenia FFD czy RFD [1, 5, 6, 7, 8].

Sieci sensorowe mogą być budowane w oparciu o dwie topologie. Pierwsza topologia typu gwiazda, umożliwia komunikację urządzeniom tylko z koordynatorem (ale nadal jest możliwe routowanie ruchu przez urządzenie FFD). Druga topologia, typu peer-to-peer, umożliwia zestawianie połączeń pomiędzy poszczególnymi sensorami (połączenia mogą być zestawiane pomiędzy urządzeniami FFD, RFD nadal komunikują się bezpośrednio z koordynatorem) [1, 5, 7]. Rys. 1. przedstawia obie topologie.



Rys. 1. Topologie wykorzystywane w sieciach sensorowych [1, 7]
Fig. 1. Topologies used in sensor networks [1, 7]

Sieci sensorowe mogą być organizowane w złożone struktury, np. można skonstruować sieć w oparciu o drzewo. W takim przypadku korzeniem drzewa będzie koordynator sieci, węzłami wewnętrznymi urządzenia FFD a liśćmi urządzenia RFD. Urządzenia mogą być organizowane w obszary. Każdy obszar musi mieć koordynatora, a ponadto musi być wyróżniony jeden koordynator dla całego systemu. Takie podejście do projektowania sieci sensorowych przedstawia Rys. 2. Każdy obszar jest identyfikowany poprzez numer ID [7].



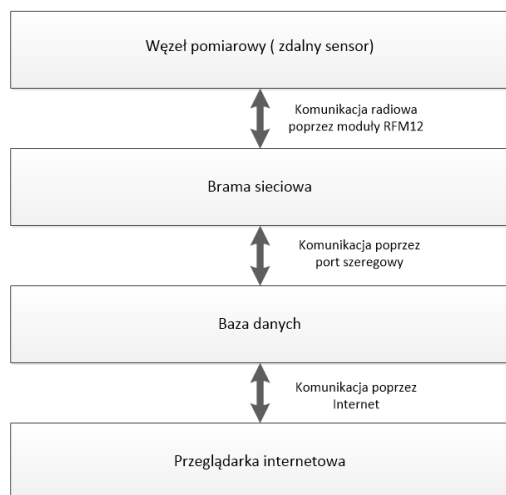
Rys. 2. Grupowanie sensorów w obszary [7]
Fig. 2. Sensors grouping in areas [7]

Jak już wspomniano, standard IEEE 802.15.4 definiuje protokoły warstwy fizycznej i dostępu do łącza danych w bezprzewodowych sieciach sensorowych. Warstwa fizyczna określa sposób modulacji sygnału, precyzowania kanału i pasma transmisji, szybkość wymiany danych oraz przełączanie modułów radiowych w stan odbiornika lub nadajnika. Twórcy modułów RFM12 zdefiniowali powyższe operacje za pomocą poleceń dostępnych dla programisty [5, 6, 7, 9]. Nie zostały jednak w żaden sposób określone sposoby adresowania sensorów, mechanizmy dostępu do łącza czy wybierania koordynatora. W interesie programisty leży zaprogramowanie tych działań. W najprostszym przypadku nie są one potrzebne. Po prostu należy uruchomić jeden moduł w stan odbiornika, drugi w stan nadajnika, ustawić odpowiednie prędkości transmisji oraz kanały wymiany danych i połączenie powinno zostać nawiązane. Rozwiązanie takie jest możliwe w przypadku wykorzystania jedynie dwóch sensorów. W przypadku większej liczby układów (niektóre sieci sensorowe mogą liczyć do tysięcy komunikujących się urządzeń) programista musi zadbać o zaprojektowanie mechanizmów wymiany danych. W pierwszej kolejności należy opracować schemat adresacji. Twórcy modułów RFM12 określili, że każda ramka wymieniana pomiędzy urządzeniami ma rozpoczynać się od przesłania trzech bajtów preambuły oraz dwóch bajtów synchronizacyjnych. Następnie może zostać przesłana dowolna ilość bajtów użytkownika. Na koniec powinny zostać przesłane dwa bajty stopki. Wobec tego, adres urządzenia docelowego oraz urządzenia źródłowego może zostać umieszczony na samym początku danych użytkownika. Adres powinien składać się z dwóch części. Po pierwsze, należy przesyłać identyfikatory obszarów, do których należą urządzenia źródłowe oraz docelowe. Druga część adresu powinna zawierać adres samego urządzenia docelowego bądź źródłowego. Załóżmy dla uproszczenia, że wykorzystywana jest topologia złożonej gwiazdy. Jako topologie złożonej gwiazdy możemy przyjąć zbiór gwiazd, których koordynatorzy komunikują się z głównym koordynatorem sieci. Główny koordynator sieci stanowi bramę sieciową dla innych technologii sieciowych, np. dla sieci Internet. Główny koordynator sieciowy komunikuje się tylko z koordynatorami poszczególnych gwiazd. Z kolei koordynatorzy poszczególnych gwiazd sterują pracą wewnętrznych sensorów. Gwiazdy są identyfikowane poprzez ID obszarów. Proponowany schemat wymiany danych w takiej sieci jest następujący. Urządzenia pomiarowe przysyłają dane do koordynatora w losowo określonych odstępach czasu. Jeśli koordynator nie odesłał potwierdzenia odebrania danych to: 1) był zajęty „swoimi sprawami” i nie mógł w danej chwili odebrać danych pomiarowych, albo 2) mogła nastąpić kolizja z innym węzłem, także przysyłającym dane. W przypadku 2 żaden z węzłów nie odbierze potwierdzenia. Każdy z węzłów pomiarowych oczekuje na odebranie potwierdzenia przez ściśle określony czas. Jeśli w tym czasie potwierdzenie od koordynatora nie nadejdzie, węzeł pomiarowy powinien ponownie odczekać losową ilość czasu i ponowić próbę nadania danych do koordynatora. Jeśli węzeł pomiarowy odebrał potwierdzenie od koordynato-

ra, to natychmiast zaczyna oczekiwać na dane od innych węzłów pomiarowych. Dane od innych układów pomiarowych przesyłane są przez koordynatora zaraz po przesłaniu potwierdzenia dla węzła pomiarowego. Jeśli węzeł pomiarowy poprawnie odebrał dane, to przesyła potwierdzenie do koordynatora. W takim przypadku urządzenia wracają do punktu startowego, inaczej dane przeznaczone dla węzła są przesyłane przy następnym nawiązanym połączeniu z koordynatorem. Weryfikacja poprawności przesyłanych danych odbywa się na podstawie sumy kontrolnej. Czasy przez jakie węzły oczekują na ponowienie próby nawiązania połączenia z koordynatorem powinny być na tyle długie, aby niezbyt często zakłócały wymianę danych innym węzłom w sieci. Te zasady wymiany danych przypominają nieco mechanizm CSMA/CA.

3. Warstwa sprzętowa zaprojektowanego systemu

W przypadku projektowania sieci sensorowej dla inteligentnego budynku, zadania poszczególnych węzłów pomiarowych polegają na sterowaniu pracą urządzeń peryferyjnych takich jak: piece, wentylatory, pompy, rolety, bramy, itp. Z punktu widzenia użytkownika końcowego istotna bywa informacja o temperaturze panującej w budynku, obecności trujących gazów, wilgotności powietrza. Ponadto użytkownik może zażyczyć sobie możliwość sterowania oświetleniem mieszkania czy sprawdzania obecności nieproszonych gości w pobliżu mieszkania. Zaprojektowany węzeł pomiarowy pozwala na sterowanie pracą kilku urządzeń peryferyjnych oraz umożliwia kontrolę takich parametrów jak temperatura w mieszkaniu czy obecność gazów szkodliwych. Z drugiej strony użytkownik końcowy ma możliwość sterowania pracą centralnego ogrzewania i dostosowywania temperatury w mieszkaniu do przewidywanej prognozy pogody. Rys. 3 przedstawia kompletny schemat zaprojektowanego systemu sensorowego.



Rys. 3. Kompletny schemat zaprojektowanego systemu sensorowego
Fig. 3. Complete schema of designed sensor system

Centralną część węzła pomiarowego stanowi mikroprocesor rodziny Atmel AVR - ATmega. Odpowiada on za odbieranie danych pomiarowych z czujników (temperatury, czadu, metanu) oraz za sterowanie pracą układów peryferyjnych. Ponadto węzeł pomiarowy został wyposażony w wyświetlacz LCD, na którym można sprawdzać stan poszczególnych układów. Ważnym zadaniem mikroprocesora jest sterowanie działaniem modułu radiowego RFM12. Ustawienie odpowiedniego pasma do komunikacji, wybór kanału oraz szybkości transmisji danych następuje zaraz po uruchomieniu węzła pomiarowego. Następnie cyklicznie następuje przełączanie modułu kolejno w stan nadajnika i odbiornika. W stanie nadajnika zostają przesłane dane dla bramy sieciowej, a w stanie odbiornika węzeł pomiarowy oczekuje na odebranie poleceń od użytkownika. W tym przypadku nie zostało wykorzystane adresowanie urządzeń – nie jest to konieczne. Na rys. 4.

przedstawiono węzeł pomiarowy. Przy budowie bramy sieciowej ponownie wykorzystano mikroprocesor AVR ATmega. Jednak w tym przypadku mikroprocesor spełnia nieco inne zadania. Po pierwsze, podobnie jak w przypadku węzła pomiarowego, inicjalizuje moduł radiowy do pracy na odpowiednim kanale i z odpowiednią szybkością. Następnie moduł radiowy jest cyklicznie przełączany kolejno w stan nadajnika i odbiornika. Dane odebrane od węzła pomiarowego są przekazywane do serwera za pomocą połączenia szeregowego. W tym celu wykorzystano układ FT232RL będący bramą sieciową [5, 6, 8, 9, 10].



Rys. 4. Węzeł pomiarowy
Fig. 4. Measuring sensor node

Po stronie serwera została uruchomiona aplikacja przygotowana w języku JAVA. Zadaniem aplikacji jest odbiór danych z portu szeregowego i zapisywanie ich do bazy danych. Aplikacja składa się z kilku wątków. Po pierwsze, uruchomiony został wątek do obsługi interfejsu użytkownika. Inny wątek odpowiada za obsługę danych na porcie szeregowym. Do obsługi portu szeregowego została wykorzystana biblioteka dynamiczna *rxTxSerial.dll*. Dostęp do bazy danych odbywa się poprzez sterownik JDBC. Fragment programu odbierającego dane na porcie szeregowym oraz zapisujący te dane do bazy danych przedstawiono na Rys. 5. Biblioteka *rxTxSerial.dll* udostępnia kilka zdarzeń związanych z obsługą portu szeregowego, na które może reagować program użytkownika. W omawianej aplikacji wykorzystano jedynie zdarzenie pojawiające się po wykryciu danych na porcie szeregowym.

```

case SerialPortEvent.DATA_AVAILABLE: // int = 1

try
{
    inputStream = serialPort.getInputStream();
    int nb = inputStream.available();

    if(nb == 6)
    {
        str=new StringBuffer("");
        byte[] readBuffer = new byte[nb];
        inputStream.read(readBuffer);
        str.append(new String(readBuffer));
        jTextArea1.replaceRange("", 0, jTextArea1.getText().length());

        String czujniki;
        if(str.charAt(0) != '0')
            temp = new String(str).substring(0, 2);
        else
            temp = new String(str).substring(1, 2);

        czujniki = new String(str).substring(2, 6);

        T = czujniki.charAt(0)-48;
        G = czujniki.charAt(1)-48;
        W = czujniki.charAt(2)-48;
        P = czujniki.charAt(3)-48;
        jTextArea1.append("Temperatura: " + temp + "\n");
        jTextArea1.append("Metan: " + G + "\n");
        jTextArea1.append("Tlenek węgla: " + T + "\n");
        jTextArea1.append("Wentylator: " + W + "\n");
        jTextArea1.append("Pompa: " + P + "\n");

        jTextArea1.setCaretPosition(jTextArea1.getText().length());
    }
    inputStream.close();
}
catch (IOException ex) {
    ex.printStackTrace();
}
}
  
```

Rys. 5. Fragment kodu źródłowego aplikacji
Fig. 5. Fragment of the application source code

Brama sieciowa przesyła dane pomiarowe w sześciu bajtach. Stany poszczególnych układów przesyłane są jako znaki kodu ASCII. Pod odebraniem danych oraz ich przetworzeniu, aplikacja wyświetla stan sensorów w podręcznym okienku oraz dokonuje zapisu do bazy danych. Następnie następuje odczytanie z bazy danych poleceń użytkownika i za pośrednictwem bramy sieciowej przekazanie ich do węzła pomiarowego.

4. Wizualizacja stanu sensorów z wykorzystaniem przestrzennych typów danych

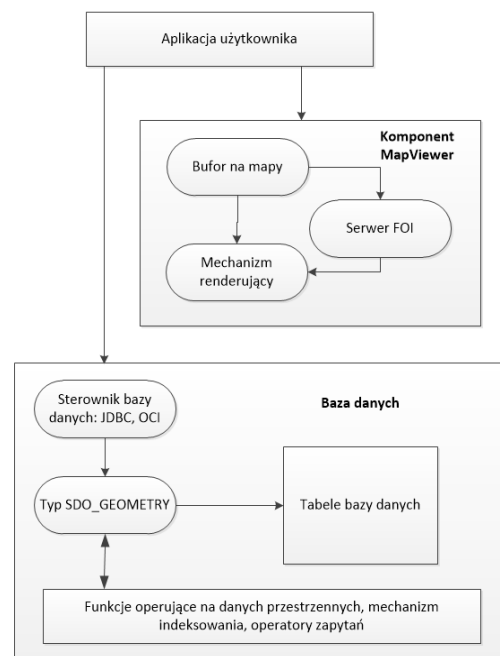
Użytkownik końcowy może monitorować stan inteligentnego budynku poprzez obserwowanie stanu sensorów na interaktywnym schemacie budynku. W celu zdefiniowania odpowiedniego schematu budynku oraz piktogramów reprezentujących poszczególne układy wykorzystano przestrzenne typy danych. Wiele współczesnych baz danych udostępnia specjalny typ danych, który może zostać wykorzystany do przechowywania informacji geograficznych, modelowania obiektów przestrzennych czy wizualizacji obiektów w przestrzeni trójwymiarowej. Informacje przechowywane przy pomocy przestrzennych typów danych cechują się dwiema właściwościami. Po pierwsze określa ona lokalizację obiektu, która zależy od układu odniesienia. Współczesne systemy baz danych udostępniają co najmniej dwa układy odniesienia: układ współrzędnych geograficznych oraz układ, w którym co prawda podajemy współrzędne geograficzne, ale nie jest uwzględniana krzywizna Ziemi. W projektowaniu niniejszego systemu wykorzystano bazę danych Oracle. Udostępnia ona przestrzenny typ danych o nazwie SDO_GEOMETRY. Istotne jest to, że baza danych Oracle wprowadza jeszcze jeden dodatkowy układ odniesienia. Możliwe jest tutaj definiowanie obiektów na podstawie kartezjańskiego układu współrzędnych, co wykorzystano w artykule. Oprócz wspomnianej informacji o lokalizacji, podczas definiowania obiektu przestrzennego należy określić kształt obiektu. Typy przestrzenne udostępniają możliwości definiowania punktów, linii, wielokątów oraz zbiorów tych obiektów geometrycznych. Zagadnieniem sposobu przechowywania danych przestrzennych zajmują się takie komitety jak Open GIS Consortium (OGC) czy Urząd Komitetu Integracji Europejskiej (dyrektywa INSPIRE). Jak już wspomniano dane przestrzenne w bazie Oracle przechowywane są za pomocą typu SDO_GEOMETRY [4].

Aby wizualizować stan sensorów nie wystarczy samo zdefiniowanie obiektów za pomocą przestrzennego typu danych. Kolejne kroki obejmują zdefiniowanie wyświetlania poszczególnych obiektów przestrzennych oraz powiązanie bazy danych z serwerem aplikacji, na którym została wdrożona usługa pozwalająca na renderowanie map oraz schematów budynków (Oracle Map Builder). Spojrzenie na całość systemu pozwalającego na przedstawianie danych przestrzennych przedstawia rys. 6 [4].

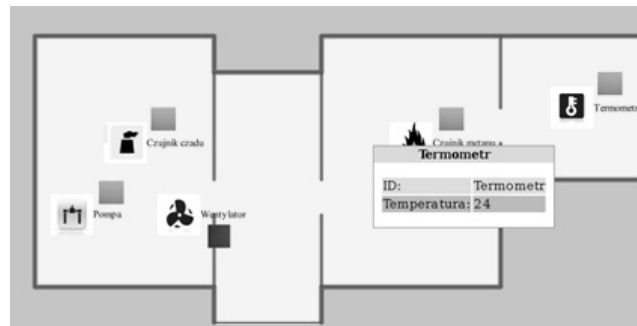
Oprócz przestrzennego typu danych technologia Oracle Spatial udostępnia szereg funkcji pozwalających na przetwarzanie obiektów geograficznych. Udostępniane są takie funkcjonalności jak możliwość wyszukiwania najbliższego sąsiada, przekształcanie adresów obiektów na współrzędne geograficzne, mechanizm indeksowania czy przechowywania plików graficznych reprezentujących obiekty przestrzenne. Rys. 6 Przedstawia uproszczoną architekturę komponentu MapViewer. Komponent MapViewer jest to usługa, która odpowiedzialna jest za przedstawianie schematów budynków oraz map użytkownikowi końcowemu, a która powinna zostać wdrożona na serwerze aplikacji. W niniejszej pracy wykorzystano serwer aplikacji WebLogic. Do powiązania plików graficznych z wprowadzonymi danymi przestrzennymi wykorzystano narzędzie Oracle Map Builder [4].

Twórcy technologii Spatial udostępnili szereg bibliotek pozwalających na wprowadzenie interakcji pomiędzy użytkownikiem a schematem budynku czy mapą. W niniejszej pracy wykorzystano bibliotekę języka JavaScript. Za pomocą odpowiednio przygotowanych funkcji języka JavaScript możliwe jest umieszczenie mapy w odpowiednim znaczniku języka HTML. Rys. 7 przedstawia schemat budynku wraz z sensorami zamieszczony na stronie

internetowej, przygotowanej dla użytkownika końcowego. Użytkownik po najechaniu kursorem myszki na piktogramy reprezentujące czujniki ma możliwość podglądu stanu sensorów.



Rys. 6. Technologia Oracle Spatial [4]
Fig. 6. Oracle Spatial technology [4]



Rys. 7. Schemat budynku wraz z zaznaczonymi stanami sensorów
Fig. 7. Building schema with sensor state messages

Układ	Stan
Termometr	20°C
Czad	Nie wykryto
Gaz	Nie wykryto
Pompa	Działa
Wentylator	Działa

Rys. 8. Okienko przedstawiające stan sensorów
Fig. 8. Message box showing sensors state

Ostatnią część systemu stanowi interfejs zaprojektowany z użyciem języka PHP, za pomocą którego użytkownik może sterować pracą poszczególnych układów. Użytkownik poprzez formularze języka HTML, odwołujące się do skryptów PHP może manipulować danymi sterującymi przechowywanymi w bazie danych. Ponadto skrypty języka PHP zostały wykorzystane w celu przedstawienia podręcznego okienka informującego o stanach poszczególnych sensorów. Przykładowe okno stanu sensorów przedstawia rys. 8.

5. Podsumowanie

Niniejszy artykuł porusza zagadnienie projektowania systemów sensorowych. Omówiono podstawy projektowania sieci sensorowych, a w szczególności sposoby dostępu do łącza oraz wymiany danych w bezprzewodowych sieciach sensorowych. Niewątpliwie zaprojektowanie sposobu wymiany danych pomiędzy sensorami w przypadku użycia modułów radiowych RFM12 stanowi wyzwanie. Należy tak opracować protokół wymiany danych, funkcjonalność poszczególnych sensorów, aby wymiana danych była niezawodna, a każde z urządzeń miało możliwość przesłania informacji w rozsądnym czasie. Mikroprocesory AVR ATmega są chętnie wykorzystywane ze względu na niską cenę, wysoką niezawodność oraz niewielkie zużycie energii elektrycznej. Szansą na efektywną wizualizację systemów sensorowych stanowią przestrzenne typy danych. Ma to szczególne znaczenie w przypadku definiowania sieci złożonych z tysięcy sensorów, rozsypanych po różnych obszarach geograficznych.

6. Literatura

[1] Dymora P., Mazurek M., Nieroda S.: Sensor network infrastructure for intelligent building monitoring and management system, *Annales UMCS Informatica*, 2012.

- [2] Dymora P., Mazurek M.: A comparative study of self-adopting fault tolerant protocols in wireless sensor networks. *Przegląd elektrotechniczny*, ISSN 0033-2097, R. 90 NR 1/2014, s.167 - 170, 2014.
- [3] Dymora P., Mazurek M.: Delay analysis in wireless sensor network protocols. *PAK* 2013 nr 10, s. 1054-1056, 2013.
- [4] Kothuri R., Godfrind A., Beinat E.: *Pro Oracle Spatial for Oracle Database 11g*. Apress, New York, 2012.
- [5] Baranowski R.: *Mikrokontrolery AVR ATmega w praktyce*, Wyd. BTC, Warszawa 2005.
- [6] Francuz T.: *Język C dla mikrokontrolerów AVR: od podstaw do zaawansowanych aplikacji*, Helion, Gliwice 2011.
- [7] Flaudi R.: *Building Wireless Sensor Networks*. O'Rielly Media, New York 2010.
- [8] AVR ATmega16/32 reference note.
- [9] FT232RL reference note.
- [10] RFM12 reference note.

otrzymano / received: 07.07.2014

przyjęto do druku / accepted: 02.09.2014

artykuł recenzowany / revised paper

INFORMACJE

Informacje dla Autorów

Redakcja przyjmuje do publikacji tylko prace oryginalne, nie publikowane wcześniej w innych czasopismach. Redakcja nie zwraca materiałów nie zamówionych oraz zastrzega sobie prawo redagowania i skracania tekstów oraz streszczeń.

Artykuły naukowe publikowane w czasopiśmie PAK są formatowane jednolicie zgodnie z ustaloną formatką zamieszczoną na stronie redakcyjnej www.pak.info.pl. Dlatego artykuły przekazywane redakcji należy przygotowywać w edytorze Microsoft Word 2003 (w formacie DOC) z zachowaniem:

- wielkości czcionek,
- odstępów między wierszami tekstu,
- odstępów przed i po rysunkach, wzorach i tabelach,
- oznaczeń we wzorach, tabelach i na rysunkach zgodnych z oznaczeniami w tekście,
- układu poszczególnych elementów na stronie.

Osobno należy przygotować w pliku w formacie DOC notki biograficzne autorów o objętości nie przekraczającej 450 znaków, zawierające podstawowe dane charakteryzujące działalność naukową, tytuły naukowe i zawodowe, miejsce pracy i zajmowane stanowiska, informacje o uprawianej dziedzinie, adres e-mail oraz aktualne zdjęcie autora o rozmiarze 3,8 x 2,7 cm zapisane w skali odcieni szarości lub dołączone w osobnym pliku (w formacie TIF).

Wszystkie materiały:

- artykuł (w formacie DOC),
- notki biograficzne autorów (w formacie DOC),
- zdjęcia i rysunki (w formacie TIF lub CDR),

prosimy przesyłać w formie plików oraz dodatkowo jako wydruki na białym papierze (lub w formacie PDF) na adres e-mail: wydawnictwo@pak.info.pl lub pocztą zwykłą, na adres: Redakcja Czasopisma Pomiar Automatyka Kontrola, Asystent Redaktora Naczelnego mgr Agnieszka Skórkowska, ul. Akademicka 10, p.21A, 44-100 Gliwice.

Wszystkie artykuły naukowe są dopuszczane do publikacji w czasopiśmie PAK po otrzymaniu pozytywnej recenzji. Autorzy materiałów nadesłanych do publikacji są odpowiedzialni za przestrzeganie prawa autorskiego. Zarówno treść pracy, jak i wykorzystane w niej ilustracje oraz tabele powinny stanowić dorobek własny Autora lub muszą być opisane zgodnie z zasadami cytowania, z powołaniem się na źródło cytatu.

Przedrukowywanie materiałów lub ich fragmentów wymaga pisemnej zgody redakcji. Redakcja ma prawo do korzystania z utworu, rozporządzania nim i udostępniania dowolną techniką, w tym też elektroniczną oraz ma prawo do rozpowszechniania go dowolnymi kanałami dystrybucyjnymi.