

MAŁGORZATA KOCOT, Jakub OLSZYNA, Wiesław WINIECKIPOLITECHNIKA WARSZAWSKA, INSTYTUT RADIOELEKTRONIKI,
ul. Nowowiejska 15/19, 00-665 Warszawa**Zdalnie konfigurowany rozproszony system pomiarowy****Inż. Małgorzata KOCOT**

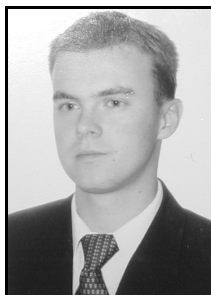
Absolwentka studiów I stopnia na Wydziale Elektroniki i Technik Informatycznych PW, autorka pracy inżynierskiej pt. Rozproszony system do pomiarów klimatycznych z wykorzystaniem standardu ZigBee. Obecnie studentka studiów II stopnia. Aktualne obszary zainteresowań: rozproszone systemy pomiarowo-sterujące, technologie transmisji bezprzewodowej krótkiego i średniego zasięgu, języki skryptowe.



e-mail: M.Kocot@stud.elka.pw.edu.pl

Mgr inż. Jakub OLSZYNA

Absolwent Wydziału Elektroniki i Technik Informatycznych PW, specjalności Radiokomunikacja i Techniki Multimedialne. Obecnie doktorant w Instytucie Radioelektroniki, członek IEEE. Autor lub współautor 15 publikacji naukowych. Aktualne obszary zainteresowań: rozproszone systemy pomiarowo-sterujące, sieci czujnikowe, arytmetyka modularna w kryptografii klucza publicznego.



e-mail: J.Olszyna@ire.pw.edu.pl

Prof. dr hab. inż. Wiesław WINIECKI

Prof. nzw. na Wydziale Elektroniki i Technik Informatycznych PW. Kierownik zespołu Komputerowej Techniki Pomiarowej. Członek Komitetu Metrologii i Aparatury Naukowej PAN, wiceprezes POLSPAR, członek IEEE. Autor lub współautor 4 książek i ponad 170 publikacji naukowych. Obszary zainteresowań: systemy pomiarowe, przyrządy wirtualne, nowoczesne technologie komunikacyjne i programowe w skupionych i rozproszonych systemach pomiarowo-kontrolnych.



e-mail: W.Winięcki@ire.pw.edu.pl

Streszczenie

W artykule przedstawiono możliwości zdalnego konfigurowania rozproszonych systemów pomiarowych, na przykładzie systemu wykorzystującego technologię ZigBee. Opisany system pomiarowy bazuje na zestawie deweloperskim Professional Development Kit for XBee ZB w wersji XK-Z11-PD-WE. Dzięki skryptom napisanym w języku Python system może być w trakcie działania przystosowywany do nowych zadań, na drodze wymiany oprogramowania kontrolera sieci.

Słowa kluczowe: rozproszone systemy pomiarowe, ZigBee, Python.

Remotely configurable distributed measurement system**Abstract**

The paper presents the possibility of remote configuration of distributed measurement systems on the example of a measurement system utilizing ZigBee technology [7]. Due to the increasing number of potential applications of distributed measurement systems [1], [2], the possibility to adapt the entire system to new applications (changes of system behaviour, network topology etc.) may prove to be an interesting and useful feature. The following design objectives have been adopted: ability to replace software without restart and recompilation, ability to change the network topology to fit the application functionality and possibility to control the entire system through LAN. The designed distributed measurement system is based on the XK-Z11-PD-WE version of the Professional Development Kit for XBee ZB and is controlled by an application created in LabView. System can be adapted to new tasks during operation using Python scripts [8] through an exchange of the network controller software. The proposed solution is better than upgrading firmware through ZigBee network using OAD approach, mainly because of faster execution (simple change of the script instead of firmware upgrade performed on all the elements of the system) and larger amount of a memory available. Correctness of the system has been experimentally verified. The obtained results confirm the usefulness of the proposed functionality in distributed measurement systems.

Keywords: distributed measurement systems, ZigBee, Python.

1. Wstęp

Stosowanie rozproszonych systemów pomiarowych konieczne są wszędzie tam, gdzie elementy systemu rozmieszczone są w dużej odległości i wymagają stałej kontroli [1, 2]. Jako potencjalne obszary zastosowań tego rodzaju systemów można wskazać m.in. zakłady przemysłowe, infrastrukturę komunalną oraz duże budynki. Jednym z czynników przyspieszających rozpowszechnianie rozproszonych systemów pomiarowych jest dostępność tanich, szybkich i niezawodnych komputerów, rozwój technologii teleinformatycznych, a także coraz łatwiejszy dostęp do platform komunikacyjnych. Ze względu na rozmieszczenie terytorialne oraz dużą liczbę urządzeń wchodzących w skład systemu, istotna z punktu widzenia użyteczności, elastyczności i obsługi systemu może się okazać możliwość inicjowanej zdalnie adaptacji systemu do warunków pomiaru. Idea adaptacji polega na zmianie funkcjonalności systemu bez konieczności ponownej kompilacji aplikacji i wykonania restartu całego systemu. Adaptacja w rozważanym kontekście dotyczy m.in. zmiany sposobu działania systemu oraz zmiany topologii sieci, która może odbywać się na drodze wymiany oprogramowania niektórych elementów systemu. Ze względu na rosnącą liczbę potencjalnych zastosowań rozproszonych systemów pomiarowych, możliwość adaptacji systemu do nowych zastosowań (zmiana sposobu działania, topologii sieci itp.) może okazać się przydatną cechą.

Adaptacja systemu może polegać na aktualizacji oprogramowania każdego z elementów. W odniesieniu do ZigBee metoda ta nosi nazwę OAD (ang. *Over-Air Download*) i jest wykorzystywana zarówno w standardzie IEEE 802.15.4, jak i w specyfikacji ZigBee. W przypadku rozproszonych systemów pomiarowych, w których elementy systemu są rozmieszczone terytorialnie, zazwyczaj stosowana jest jedynie komunikacja bezprzewodowa. W takim przypadku wzrost liczby urządzeń powoduje, że przeprowadzanie indywidualnej aktualizacji dla każdego elementu staje się trudne. Dobrym rozwiązaniem jest wówczas zmiana funkcjonalności wszystkich elementów naraz, inicjowana zdalnie z poziomu kontrolera systemu. Takie rozwiązanie ułatwia zarządzanie systemem i zwiększa jego użyteczność. W artykule przedstawiono możliwości zdalnego konfigurowania na przykładzie rozproszonego systemu pomiarowego wykorzystującego technologię ZigBee. Opisany wcześniej proces adaptacji jest możliwy dzięki zastosowaniu języka skryptowego Python w module kontrolera systemu.

2. Standard ZigBee

Technologia ZigBee została zaprojektowana z myślą o wykorzystaniu w rozproszonych systemach pomiarowych i sieciach czujnikowych.

Standard ten zakłada niski koszt wytworzenia urządzeń, niski poziom złożoności oraz bardzo niski pobór mocy dla elementów łączności bezprzewodowej w różnorodnych środowiskach pracy. Ta cecha czyni ZigBee niezwykle przydatnym dla różnorodnych aplikacji począwszy od detektorów ruchu wykorzystywanych w systemach alarmowych, a kończąc na czujnikach wilgotności ściółki leśnej w systemach ostrzegania przed pożarami. Mały pobór energii przekłada się na niewielki koszt eksploatacji, mobilność urządzeń oraz skalowalność (rozumianą jako zachowanie wydajności systemu przy zwiększaniu liczby jego elementów). Największe zużycie energii typowego urządzenia nadawczo-odbiorczego występuje podczas przesyłania informacji - dzięki zastosowaniu krótkiego czasu transmisji ramek oraz długiego okresu pomiędzy kolejnymi cyklami przesyłania danych możliwa jest wówczas minimalizacja zużycia energii. ZigBee cechuje względnie mała szybkość transmisji oraz duży stopień bezpieczeństwa i niezawodności [3, 4, 5].

Specyfikacja ZigBee w warstwie fizycznej i MAC bazuje na normie IEEE 802.15.4 [6, 7]. Maksymalna szybkość transmisji wynosi 250 kbit/s w paśmie ISM 2,4 GHz. ZigBee wykorzystuje dwa tryby pracy (tryb aktywny oraz tryb uśpienia), charakteryzuje się również możliwością dołączenia wielu urządzeń. Ze względu na funkcjonalność można wyróżnić dwa typy urządzeń: urządzenia o pełnej funkcjonalności FFD (ang. *Full Function Devices*) oraz urządzenia o ograniczonej funkcjonalności RFD (ang. *Reduced Function Devices*). Urządzenia RFD to urządzenia końcowe (np. czujniki), które mają ograniczone zasoby, korzystają z uproszczonej implementacji stosu co sprawia, że mogą pracować tylko w sieci o topologii gwiazdy i komunikować jedynie z koordynatorem sieci. Urządzenia FFD, jak koordynator czy router, korzystają z pełnej wersji stosu i mogą komunikować się ze wszystkimi urządzeniami w sieci. Istotną cechą standardu ZigBee jest to, że pozwala na tworzenie sieci w topologii gwiazdy (ang. *star topology*), drzewa (ang. *cluster tree topology*) i kraty (ang. *mesh topology*). W topologii gwiazdy urządzenia mogą komunikować się tylko z koordynatorem, w łączu typu punkt-punkt. Sieć w topologii drzewa składa się z wielu mniejszych sieci w topologii gwiazdy, których centralne węzły komunikują się bezpośrednio z koordynatorem systemu. Topologia kraty pozwala na tworzenie połączeń pomiędzy dowolnymi urządzeniami (ang. *peer-to-peer*), w szczególności nowe drogi połączeń mogą być organizowane *ad hoc*.

3. Języki skryptowe - Python

Języki skryptowe (ang. *scripting languages*) zalicza się do języków programowania wysokiego poziomu, których instrukcje są interpretowane podczas wykonywania, a nie kompilowane do postaci języka maszynowego. Dzięki temu funkcjonalność aplikacji napisanej w języku skryptowym może być modyfikowana w sposób dynamiczny, w trakcie działania programu.

Python jest skryptowym językiem interpretowalnym – programy są przetwarzane w trakcie wykonania przez interpreter [8]. Dzięki spełnieniu zasad programowania strukturalnego, obiektowego i funkcjonalnego język ten nie ogranicza stylu programowania. Biblioteka standardowa zawiera bardzo dużą liczbę funkcji umożliwiających wykonanie względnie złożonych zadań (np. aplikacje do modelowania 3D). Proste konstrukcje składniowe umożliwiają tworzenie funkcjonalnego kodu w krótszym czasie niż w przypadku kompilowanych języków programowania jak C, C++ czy Java. Python działa na wielu platformach, także w systemach wbudowanych. Implementacje Pythona zostały przygotowane dla systemów Windows, Linux/Unix i Mac OS X oraz dla maszyn wirtualnych Java oraz .NET.

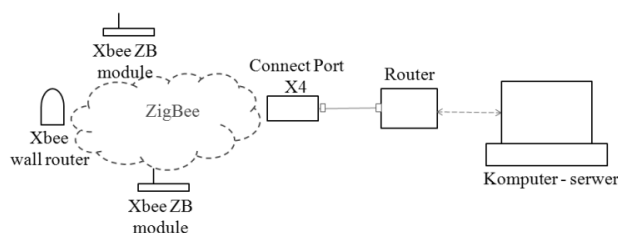
Podstawową zaletą stosowania języka Python (oraz innych języków skryptowych) w procesie adaptacji rozproszonych systemów pomiarowych jest działanie skryptów w ograniczonej przestrzeni (brak możliwości ingerencji w oprogramowanie firmowe).

Błędnie napisany skrypt w języku Python uruchomi się i zakończy działanie w sposób nieoczekiwany, ale nie zablokuje w sposób trwały możliwości działania systemu, co mogłoby się stać np. w przypadku błędnie działającego oprogramowania firmowego (zaktualizowanego za pomocą wspomnianej wcześniej metody OAD).

4. Opis zdalnie konfigurowanego rozproszonego systemu pomiarowego

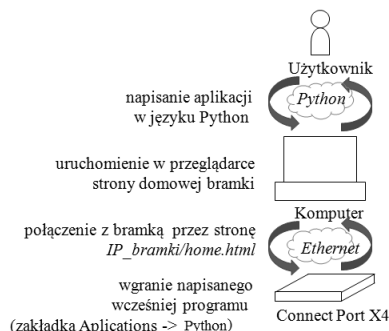
W celu sprawdzenia możliwości adaptacji systemu na drodze zdalnej wymiany oprogramowania zaprojektowano rozproszony system pomiarowy wykorzystujący technologię ZigBee. Zostały przyjęte następujące założenia projektowe: wymiana oprogramowania bez konieczności ponownej kompilacji i restartu systemu, zmiana topologii sieci w zależności od aplikacji oraz sterowanie systemem przy użyciu sieci komputerowej (Ethernet). Na rynku pojawił się zestaw dobrze nadający się do realizacji tego zadania – iDigi Professional Development Kit for Xbee, model ZK-Z11-PD-WE [9]. W skład zestawu wchodzi: bramka ethernetowa ConnectPort X4, moduły Xbee i Xbee-PRO wraz z płytkami montażowymi oraz router Xbee-PRO, wyposażony w czujniki temperatury i natężenia oświetlenia. Moduły Xbee i Xbee-PRO w trybie uśpienia pobierają nie więcej prądu, niż 10 μ A. Moduł Xbee ma zasięg do 40 m w budynku i do 120 m na zewnątrz, moc nadajnika wynosi 1,25 mW, a czułość -96dBm. W przypadku modułu Xbee-PRO zasięg wewnątrz budynku wynosi 60m, na zewnątrz 1,6 km, moc nadajnika jest równa 50 mW, zaś czułość to -102 dBm. Moduł routera Xbee-PRO zawiera zintegrowane czujniki światła i temperatury. Zakres pomiaru temperatury wynosi od -20 °C do +70 °C, a pomiary prowadzone są z dokładnością ± 2 °C. Zakres pomiaru natężenia światła obejmuje fale o długości od 360 nm (ultrafiolet) do 970 nm (podczerwień), czyli cały zakres długości fal rejestrowanych przez ludzkie oko.

Istotnym elementem zestawu jest bramka ethernetowa Connect Port X4, która stanowi interfejs pomiędzy sieciami Ethernet i ZigBee. Bramka implementuje protokoły UDP/TCP i DHCP oraz umożliwia sterowanie systemem pomiarowym z poziomu sieci komputerowej. Funkcjonalność bramki jest uzyskiwana za pomocą skryptów napisanych w języku Python, które mogą być zdalnie dodawane, usuwane oraz modyfikowane - jedynym ograniczeniem jest rozmiar pamięci wewnętrznej, który w tym przypadku wynosi 16 MB. Ta cecha pozwala na realizację idei adaptacji systemu, co stanowi element nowości w stosunku do innych prac obejmujących zagadnienia związane z rozproszonymi systemami pomiarowymi, wykorzystującymi technologię ZigBee. Schemat blokowy zaprojektowanego rozproszonego systemu pomiarowego przedstawiono na rysunku 1.



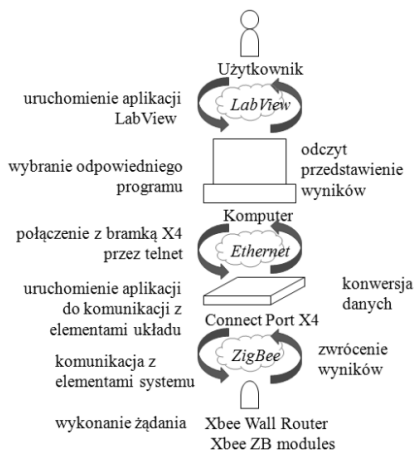
Rys. 1. Schemat blokowy systemu
Fig. 1. Block diagram of the system

Proces zdalnej wymiany oprogramowania został przedstawiony na rysunku 2. Pierwszym etapem jest stworzenie aplikacji w języku Python. Następnie możliwe jest zapisanie jej w pamięci bramki ethernetowej. Oprogramowanie bramki umożliwia zarządzanie zainstalowanymi aplikacjami z poziomu przeglądarki internetowej.



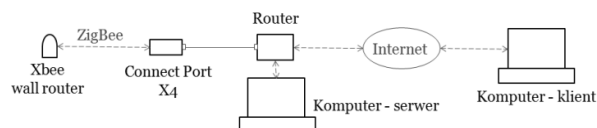
Rys. 2. Proces zdalnej wymiany oprogramowania
Fig. 2. Remote software Exchange process

Zaprojektowany system jest sterowany przez aplikację napisaną w środowisku LabView (rys. 3). Użytkownik wybiera odpowiedni program, po czym następuje nawiązanie połączenia z bramką ethernetową za pomocą protokołu Telnet, przesłanie odpowiedniej komendy uruchamiającej program w bramce, wykonanie programu i zwrócenie wyników. Aplikacja w LabView odczytuje dane z terminala Telnet i przedstawia wyniki użytkownikowi.



Rys. 3. Schemat funkcjonowania systemu
Fig. 3. Functional diagram of the system

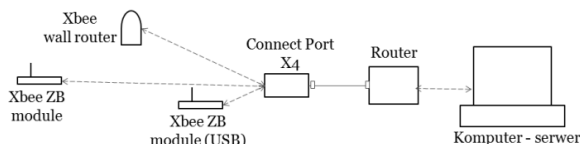
Aby przedstawić możliwość adaptacji systemu na drodze zdalnej wymiany oprogramowania, zostały zaprojektowane dwie przykładowe aplikacje. Pierwsza służy do odczytu temperatury i natężenia oświetlenia z routera Xbee Wallrouter (rys. 4).



Rys. 4. Schemat blokowy dla aplikacji pomiaru temperatury
Fig. 4. Block diagram for measuring temperature

Druga z aplikacji umożliwia pomiar wartości parametru RSSI (ang. *Received Signal Strength Indicator*) w łączu pomiędzy bramką ethernetową a każdym z pozostałych urządzeń sieci ZigBee (rys. 5). W obu przypadkach komunikacja między urządzeniami odbywa się poprzez sieć ZigBee. Bramka ethernetowa podłączona jest do routera zarządzającego bezprzewodową siecią lokalną, dzięki czemu możliwe jest zdalne sterowanie systemem z poziomu aplikacji działającej na komputerze pełniącym rolę serwera. Powyższe przykłady pokazują ideę adaptacji systemu

poprzez zmianę topologii sieci – w przypadku pierwszego przykładu jest to topologia *peer-to-peer*, w przypadku drugiego – topologia gwiazdy.



Rys. 5. Schemat blokowy dla aplikacji do pomiaru RSSI
Fig. 5. Block diagram for measuring the RSSI

5. Podsumowanie

W artykule opisano właściwości standardu ZigBee. Omówiono zalety języków skryptowych pozwalających na zmianę funkcjonalności aplikacji bez konieczności ponownej kompilacji i restartu systemu. Na przykładzie zrealizowanego rozproszonego systemu pomiarowego, bazującego na zestawie deweloperskim Professional Development Kit for Xbee ZB w wersji XK-Z11-PD-WE, przedstawiono możliwość adaptacji systemu na drodze zdalnie inicjowanej wymiany oprogramowania. Dzięki skryptom napisanym w języku Python system może być w trakcie działania dynamicznie przystosowywany do nowych zadań lub warunków pomiaru. Rozwiązanie opisane w artykule jest dużo lepsze niż aktualizacja oprogramowania metodą OAD, ponieważ programy w formie skryptów mają dostęp do znacząco większej pamięci (pamięć ulotna, flash) niż oprogramowanie firmowe. Dzięki temu możliwe jest pisanie elastycznych aplikacji składających się z dużej liczby instrukcji. Dodatkowo błędnie działający skrypt nie zablokuje całego systemu, jak mogłoby to się stać w przypadku niepoprawnej pracy oprogramowania firmowego. Poprawność działania systemu została zweryfikowana eksperymentalnie. Uzyskane rezultaty potwierdzają przydatność proponowanego rozwiązania w rozproszonych systemach pomiarowych.

6. Literatura

- [1] Akyildiz F., Su W., Sankarasubramanian Y., Cayirci E.: A survey on sensor networks, *Communications Magazine*, t. 40 (8), s. 102–114, 2002.
- [2] Nawrocki W.: *Rozproszone systemy pomiarowe*. WKiŁ, Warszawa, 2006.
- [3] Li J., Zhu X., Tang N., Sui J.: Study on ZigBee network architecture and routing algorithm. *Proc. 2nd International Conference on Signal Processing Systems*, Dalian, China, 2010, s. V2-389-V2-393.
- [4] Ramya C.M., Shanmugaraj M., Prabakaran R.: Study on ZigBee technology, *Proc. 3rd International Conference on Electronics Computer Technology*, Kanyakumari, India, 2011, s. 297-301.
- [5] Evans-Pughe C.: Bzzzz zzz [ZigBee wireless standard], *IEEE Review*, 2003, s. 28-31.
- [6] Zucatto, F.L., Biscassi, C.A., Monsignore, F., Fidelix, F., Coutinho, S., Rocha, M.L.: ZigBee for building control wireless sensor networks, *Proc. The 2007 SBMO/IEEE MTT-S Int. Microwave and Optoelectronics Conference*, Salvador, Brazil, 2007, s. 511-515.
- [7] Olszyna J.: Technologie bezprzewodowe w przemyśle na przykładzie standardu ZigBee, *PAR*, 12 (2008), s.12-15.
- [8] <http://www.python.org/>
- [9] http://www.digi.com/pdf/ds_idigatewaydevkit.pdf

otrzymano / received: 28.06.2012

przyjęto do druku / accepted: 02.08.2012

artykuł recenzowany / revised paper