

## **ZigBee безжични сензорни мрежи като инструмент за отдалечено събиране и анализиране на данни (с помощта на Python, MySQL и Matlab)**

Светослав Атанасов, Белма Гаази, Пламен Даскалов, Цветелина Георгиева, Веселина Недева

**ZigBee wireless sensor networks as an instrument of remote data measuring and analysis (by means of Python, MySQL and Matlab):** The main purpose of the article is to propose work procedure, method, a step by step algorithm for collecting, processing and analyzing information from wireless sensors remotely. Proposed procedure is applicable to multiple sensors. Extracted from the sensor information is stored on a server on the Internet, wherefrom it can be retrieved, used for processing and analyzed or used to manage processes in industrial environments or in agriculture.

**Key words:** ZigBee, WSN, Python, MySQL, Matlab, Remote Data Measuring and Analyzing, Data acquisition.

### **ВЪВЕДЕНИЕ**

Целта на нашата разработка е да представим методология за отдалечено събиране и обработка на информация. За тази цел се прилага отдалечен достъп до информацията получена от безжичен температурен сензор, която след извличането си ще бъде разположена на достъпно място в интернет – на уеб сървър. Основните ни инструменти са програмния език Python, бази данни MySQL и Matlab.

Предложената от нас схема може да бъде разглеждана като миниатюрен модел на индустриална мрежа от типа свързаност полева мрежа – мрежа от ниво информация. С нашата разработка правим крачка и по посока на елиминиране на често срещан проблем – свързването между двете мрежи от различен тип. Въпреки твърденията на производителите за отвореност и съвместимост, на практика устройствата от различните производители на оборудване за различните нива в полевите мрежи са несъвместими.

Свързването на полева мрежа – мрежа от ниво информация от архитектурна гледна точка става по два начина. Първият е чрез използване на тунелиране на протоколи – рамките на един протокол са скрити е полето за данни в рамките на друг протокол, без да се променя тяхното съдържание. Втория подход е използването на шлюз за преобразуване на протоколите и услугите – от едната страна шлюзът е пълноправен участник в най-ниското ниво на полевите мрежи, а от другата е достъпен с IP базирани механизми [1].

Свързването на тези два типа индустриални мрежи има две предимства: осигуряване на отдалечен достъп до системите за автоматизация и интеграция на информацията. Доставянето на информацията от полевата мрежа в централния офис на компанията например, може да се използва не само за събиране, обработка и анализиране на данни, но и за стратегически операции като управление и планиране на ресурсите [1].

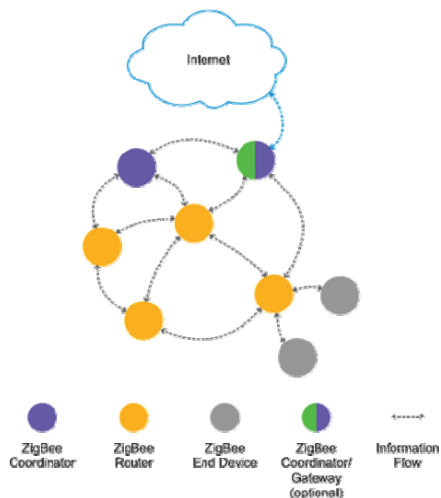
Например, при един промишлен процес (напр. химически завод, хранително-вкусова фабрика или системи за енергиен контрол и т.н.), температурата трябва да се регистрира и контролира по всяко време. За големи обекти, е изгодно информацията от показанията на всички разположени сензори да се концентрира в една точка, която да може да бъде достъпна от разстояние [2].

### **ИЗЛОЖЕНИЕ**

Безжичната нискоскоростна и с ниска консумация мрежа ZigBee (IEEE 802.15.4), има уникални свойства да се самоорганизира и самовъзстановява, тъй като мрежовите устройства в момента на включване самостоятелно формират мрежата, като се идентифицират едно с друго. При излизане от строя на някое от

устройствата мрежата самостоятелно се възстановява и започва предаване по нов маршрут. Данните се предават със скорост 250 Kbit/s което включва и служебната информация на мрежата. Безжичната мрежа ZigBee работи в диапазоните на честотните канали 868 MHz, 915 MHz, 2.4 GHz. В помещенията радиуса на действие е няколко десетки метри, а навън – до няколкостотин метра.

На фигура 1 е представена мрежовата топология на една ZigBee безжична мрежа [3].



Фигура 1. Мрежова топология при ZigBee

На следващата фигура 2 са изобразени използваните в експеримента безжичен сензор и приемник.



Фигура 2. ZigBee безжичен температурен сензор (а) и безжичен приемник (b)

Характеристики на безжичния температурен сензор [4]:

- Диапазон на измерването:  $-40^{\circ}$  to  $100^{\circ}\text{C}$ .
- Обхват на предаване: 150 до 300 m при чиста линия на видимост.
- Точност:  $\pm 0.21^{\circ}\text{C}$  from  $0^{\circ}$  to  $50^{\circ}\text{C}$ .

- Време за реакция: 3 мин. при движение на въздуха 1m/s; 30 секунди във вода.
- Честота на дискретизация: 1 отчитане показанието на сензора на минута.
- Живот на батерията: обикновено 1 година без зареждане; обикновено 3 години със зареждане през соларния панел.
- Размер: 16.5cm x 5cm; 3.7m кабел.
- Резолюция: 0.025° при 25°C.
- Памет: няма.
- Работен диапазон: -20° to 50°C.

Характеристики на безжичния приемник [4]:

- Получава данни от до 50 безжични възли.
- Включва се USB порт на PC.
- Обхват на предаване: 150 до 300 m при чиста линия на видимост.
- USB захранване с резервна батерия.
- Работен стандартен диапазон: -40° to +50°C.
- Комуникация с PC: 1.8m (6') USB кабел.
- Живот на батерията: обикновено 3 години.
- Размери: 16.5 x 5 cm (6.5" x 2").

Един от програмните инструменти с който боравим по време експеримента е езикът Python. Python [5] е интерпретиран, интерактивен, обектно-ориентиран език за програмиране. Тъй като Python е език, който се интерпретира, се спестява значително време за разработка, тъй като не са необходими компилиране и свързване (linking) за тестването на дадено приложение. Освен това, бидейки интерпретиран език с идеология сходна с тази на Java, приложение, написано на него, е сравнително лесно преносимо на множеството от останали платформи (или операционни системи).

Програмите, написани на Python, са доста компактни и четими, като често те са и по-кратки от еквивалентните им, написани на C/C++.

Програмите написани на Python представляват съвкупност от файлове с изходен код. При първото си изпълнение този код се компилира до байткод, а при всяко следващо се използва кеширана версия. Байткодът се изпълнява от интерпретатор на Python.

Python е добър език-спойка за взаимодействие (interfacing) между периферни устройства и интернет [2].

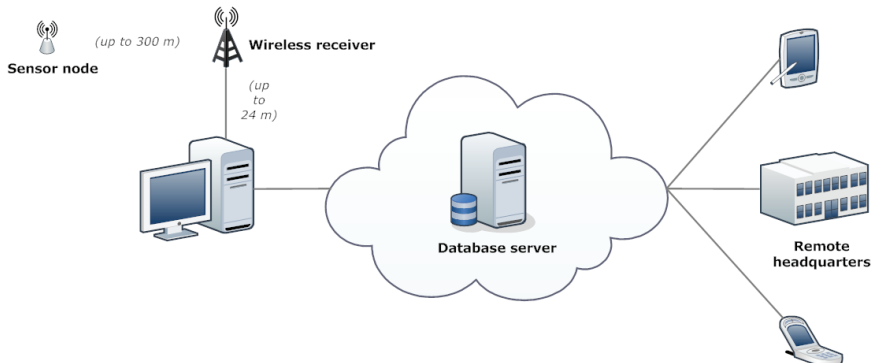
SQLite [6] е софтуерна библиотека, реализираща самостоятелна, безсървърна, без настройки SQL база данни. SQLite е една от най-използваните SQL база данни във света. Често се използва в съвременните устройства, вградени системи (embedded) и при уебсайтове с малък до среден размер.

SQLite е популярен избор за база данни в мобилните телефони, PDA, MP3 плейъри, цифрови декодери и други електронни джаджи. SQLite осигурява ефективно използване на памет, дисково пространство, трафик, изключително надежден е и не изисква поддръжка от администратор на база данни.

PyScripter е безплатна IDE за високо качествена разработка на езика Python. Той е предварително инсталиран в портативната версия на Python и се инсталира лесно [7].

## ЕКСПЕРИМЕНТ

На фигура 3 е представена топологията на експерименталната мрежа. В експеримента участва само един температурен безжичен сензор, но предлаганият метод може да се използва за множество сензори. Ограничения се налагат от самата ZigBee технология - могат да бъдат свързани до 50 устройства [3, 4].



Фигура 3. Схема на топологията за отдалечено събиране на информация

При ясна видимост и липсата на препятствия сензора може да е отдалечен безжично от приемника до 300 м, а самият приемник може да е отдалечен от локалния компютър до 24 м с помощта на удължаващи USB кабели. При проведения експеримент се установи, че при липсата на пряка видимост и наличието на няколко бетонови панелни стени между сензора и приемника, силата на сигнала се намали наполовина и разстоянието на безжичното приемане се намали до под 100 метра, като се увеличи и времето за установяване на връзката между двете устройства. В някои случаи се забеляза и загуба на пакети информация.

Експериментът протича в следните стъпки:

#### **Стъпка 1 - Извличане или получаване на информация от сензора**

Сензорът може да бъде свързан към компютърната система чрез серийния интерфейс (COM или USB), обикновено при безжичните сензори ролята на посредник играе приемникът. За получаване на информацията от него могат да се използват два различни подхода в зависимост от спецификата на самото устройство – първият е сканиране на серийния порт за следене на трафика на пакети (със Python (PySerial – за директно свързани сензори към компютъра по USB, PyUSB [8] – за сензори свързани с приемник към компютъра). Вторият начин за получаване на информация се базира на възможността на съвременните устройства да използват прилежащ софтуер и възможността да записват информацията от работата си в SQLite база от данни. Този формат е удобен и съвместим идеално с Python и неговия фреймуърк Django, но представлява безсъвършна база данни. За да използваме базата данни от показанията на сензора в реална производствена среда е удобно той да бъде конвертиран и качен в MySQL или подобна база от данни (PostgreSQL, Oracle).

Още при започване на работа е добре да конвертираме в базата данни, която планираме да работим в реалната производствена среда.

Съществува и трети начин за получаване на информация от USB порта – чрез така наречените USB подслушватели (sniffers).

#### **Стъпка 2 - Обработки**

Включва обработване и конвертиране на информацията и изчистване на ненужната такава [9, 10]. Съществува и приложение - SQLite Manager – на Mozilla Firefox за управление на SQLite бази от данни.

Обработките с цел оформяне на съвместимост с MySQL включват добавяне на първичен ключ, за да може данните да се редактират, форматиране на полетата да

са целочислени, за да могат да се обработват в Matlab, премахване на излишните колони и записи.

### Стъпка 3 – Запис на информацията онлайн

Изпращане на базата данни на уеб сървър с база данни (MySQL) в интернет.

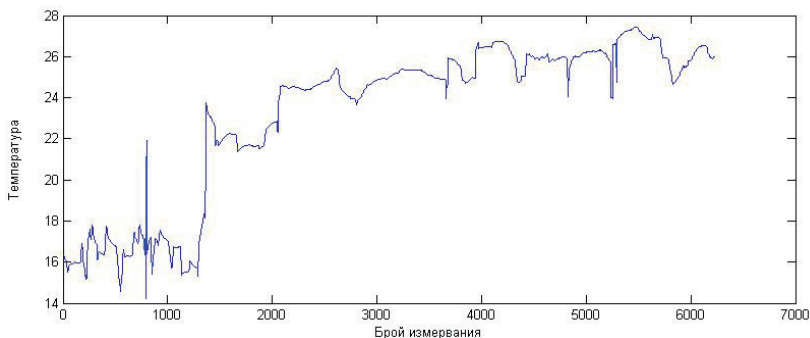
Използване на Python пакета MySQLdb, е възможно да се получи достъп до отдалечен MySQL сървър за бази данни за да се съхранят показанията от температурния сензор. Базата данни се настройва предварително така че да бъде достъпна [1, 11].

### Стъпка 4 – Извличане от отдалечен компютър

Извличане на базата данни от отдалечен компютър от MySQL сървъра в интернет и дообработването ѝ локално на отдалечения компютър – осъществява се със същия пакет MySQLdb. Дообработването включва конвертирането на информацията от базата данни в текстов файл, така че да бъде подходящ за обработка в Matlab – една колона за датата и часа и една колона с температурното показание на сензора [11].

### Стъпка 5 – Обработка и анализ в Matlab

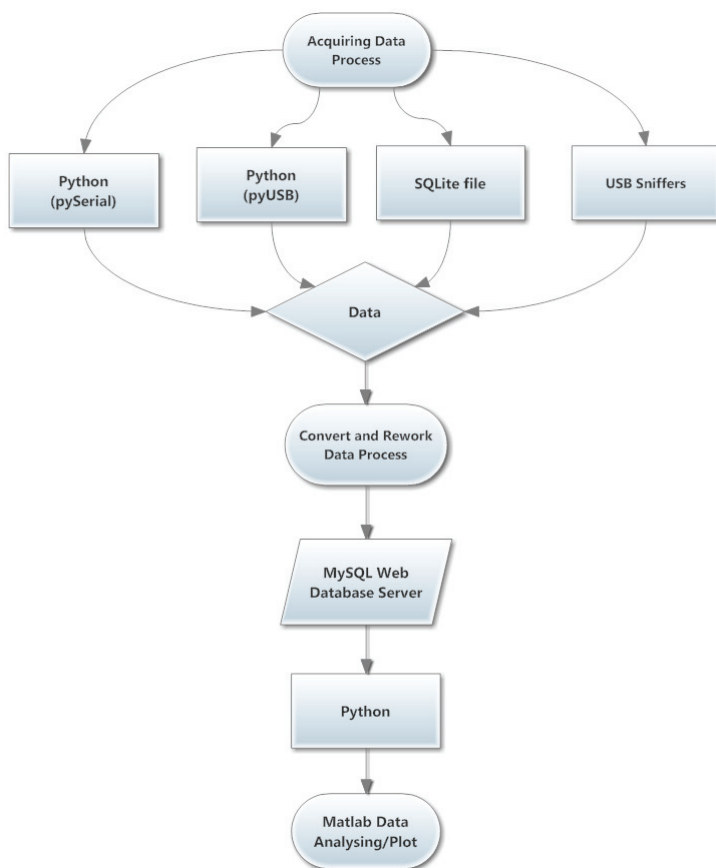
Резултати от обработката, анализи и представянето им графично.



Фигура 4. Примерни резултати и графичното им представяне

Опционално в самата система Matlab може да се създаде web server, който да осигурява обновената информация от базата данни във браузър.

На следващата фигура представяме обобщена схема на процеса:



Фигура 5. Обобщена схема на процеса

### ЗАКЛЮЧЕНИЯ И ИЗВОДИ

Предложена е схема на комуникация и алгоритъм/процедура за работа при извличането, обработката и предаването на отдалечено разстояние на информация от безжичен сензор. Така обработената информация впоследствие може да се използва или за управление на процеси в индустриални условия и в селското стопанство (с помощта на LabView) или подавана обратно на програмируеми логически контролери – тоест информацията от самите безжични сензори може да бъде използвана като инструмент за отдалечен контрол на индустриални мрежи.

На следващ етап, процесите ще са автоматизират, няма да се прави всичко на ръка, а със автоматично действащи скриптове.

Сега, процесът на анализ не изисква синхронна работа и получаване на информацията в реално време, но при продължаване на работата и включване в алгоритъма и процедурите и управление на актуатори (например PLC контролери, задействащи крайни устройства, в зависимост от информацията получена от безжичните сензори), процесите на отчитане, анализиране и задействането ще трябва да стават в реално време.

Може да бъде създаден GUI с помощта Python за управление на процесите.

### **БЛАГОДАРНОСТИ**

Настоящата статия е изготвена с финансовата помощ на Тракийски университет – Стара Загора.

Научен проект № 05-ФТТ /28.05.2013 г. на тема „Управление на технологичните процеси при изкуственото отглеждане на растения чрез отдалечен контрол на технологичните параметри”.

### **ЛИТЕРАТУРА**

- [1] Стоилов Е. (2010), Сигурност в системите за управление на технологични процеси.
- [2] Clarke T. (2013), Electronics Laboratory Handouts.
- [3] <http://www.zigbee.org/>
- [4] <http://www.onsetcomp.com/>
- [5] <http://www.python.org/>
- [6] <http://www.sqlite.org/>
- [7] <http://portablepython.com/>
- [8] <http://pyusb.sourceforge.net/docs/1.0/tutorial.html>
- [9] <http://www.sqlite.org/cli.html>
- [10] [http://www.speqmath.com/tutorials/sqlite\\_export/](http://www.speqmath.com/tutorials/sqlite_export/)
- [11] [http://www.tutorialspoint.com/python/python\\_database\\_access.htm](http://www.tutorialspoint.com/python/python_database_access.htm)

### **За контакти:**

Ас. Светослав Атанасов, Катедра “Електротехника, електроника и автоматика”, Факултет „Техника и технологии“ – Ямбол, Тракийски университет – Стара Загора, e-mail: svetoslav.atanasov@trakia-uni.bg

**Докладът е рецензиран.**