

ПРОЕКТУВАННЯ СИСТЕМИ ОБМІНУ ДАНИМИ ДАТЧИКА ІНТЕРНЕТУ РЕЧЕЙ

Зростання кількості пристроїв, що використовуються для автоматизації процесів моніторингу, керування та інших дій в рамках інтернету речей вимагає розуміння вимог до побудови серверної частини цієї мережі. Впровадження сукупності фізичних пристроїв взаємодії призводить до виникнення ситуації перевантаження мережі та зменшення пропускної спроможності.

Ключові слова: інтернет речей, датчик, сервер.

N.I. PRAVORSKA, N.V. HRIPINSKA

Khmelnitskyi National University

DEVELOPMENT OF SENSOR FOR INTERNET OF THINGS DATA TRANSFERRING

Internet of Things grows with a high speed. In a nearest years total amount of IoT expected over 20 billions devices. As a result, global providing of Internet of Things requires a lot of resources. One of them is a network. The second important recourses are hardware for IoT. IoT provides simple logic, so IoT can be build with use of 8-bit devices like Arduino devices.

In a main part is shown local module of IoT requires periodically send data to main server. Data transferring starts at moments of events or periodically by local timer events. Increasing of IoT connections will generate situation when connections are large then available throughput.

Model of Bianka used for modelling of Data transferring ability. It is shown what with rizing of retries counts received higher throughput. But with rizing of connection count we could receive delays and even DoS. Keywords: Internet of Things, sensor, server, DoS.

Вступ

Інтернет речей (Internet of Things, IoT) – це міжмережева взаємодія фізичних пристроїв, транспортних засобів (що також називаються "підключеними пристроями", або "розумними пристроями"), будівель й інших предметів, забезпечених електронікою, програмним забезпеченням, датчиками, актуаторами і мережевим підключенням, які дозволяють цим об'єктам збирати і обмінюватися даними. У 2013 році Світова ініціатива стандартизації з Інтернету речей (Global Standards Initiative on Internet of Things, IoT – GSI) визначила інтернет речей як "інфраструктуру інформаційного суспільства". Інтернет речей дозволяє пристроям бути виявленим або контрольованим видалено через існуючу мережеву інфраструктуру, створюючи можливості для прямої інтеграції фізичного світу в комп'ютерну систему, в результаті підвищуючи ефективність, точність, економічну вигоду на додаток до скорочення втручання людини. Коли інтернет речей доповнюється датчиками і виконавчими механізмами, ця технологія стає базою для загальніших кіберфізичних систем, які також охоплюють такі технології, як розумні мережі, розумні будинки, розумний транспорт і розумні міста. Кожна річ унікально ідентифікується через вбудовану обчислювальну систему, але здатна взаємодіяти з існуючою інфраструктурою інтернету. За оцінками експертів, до 2020 року в інтернет речей буде включено майже 50 мільярдів об'єктів [1].



Рис. 1. Суть інтернету речей

Згідно зі звітом компанії Gartner Inc. до 2020 року інтернет речей складатиметься з, приблизно, 20.8 мільярдів пристроїв. Інтеграція з мережею інтернет має на увазі, що пристрої повинні мати унікальні ідентифікатори, якими є IP-адреси. Проте, у зв'язку з обмеженістю пулу адрес IPv4 (4.3 мільярдів унікальних

адрес), об'єкти в мережі інтернету речей повинні використати IPv6 для розміщення надзвичайно величезного адресного простору. Ці об'єкти представляють собою не лише пристрої з сенсорними можливостями, але і активації (замки, контрольовані через інтернет), що забезпечують таку можливість. Для подальшого росту, майбутнє інтернету речей не представляється можливим без підтримки IPv6; і, отже, глобальне впровадження IPv6 найближчими роками матиме вирішальне значення для успішного розвитку інтернету речей в майбутньому.

З іншого боку, інтернет речей може відповідати і за виконання різних дій, а не тільки за сприйняття речей. Інтелектуальні торгові системи, наприклад, можуть проводити моніторинг певних купівельних звичок користувача в якому-небудь магазині за допомогою відстеження їх мобільних телефонів.

Основна частина

Інтеграція сенсорних і таких, що управляють системою, підключених до інтернету, може оптимізувати споживання енергії в цілому. Очікується, що в облаштування інтернету речей будуть інтегровані усі види енергоспоживаючих пристроїв (перемикачів, розеток, лампочок, телевізорів і так далі) і зможуть передавати інформацію компанії, що постачає комунальні послуги, щоб ефективно збалансувати виробництво і використання енергії. Такі пристрої також запропонують споживачам можливість видаленого доступу або централізованого управління за допомогою хмарних серверів, і підключення додаткових функцій, таких як планування (видалене включення або виключення обігрівачів, управління духовими шафами, зміна умов освітлення і так далі).

Окрім домашнього управління енергією, інтернет речей особливо актуальний для SmartGrid мереж, оскільки він надає системи збору і обробки інформації про енергію і потужність в автоматичному режимі з метою підвищення ефективності, надійності, стійкості і економічності розподілу і виробництва електроенергії. Використання облаштувань розширеної вимірювальної інфраструктури (AMI), підключених до інтернету, облаштування електропостачання можуть не лише збирати інформацію від кінцевих споживачів, але і управляти іншими облаштуваннями автоматизації розподілу, такими як трансформаторами і АПВ (автоматичне повторне включення).

Мережева архітектура та зразки реалізації апаратної частини

Інтернет речей вимагає величезної масштабованості в мережевому просторі для обробки швидкорослої кількості пристроїв. Стандарт 6LoWPAN використовуватиметься для підключення пристроїв до IP-мереж. З мільярдами пристроїв, що додаються в мережевий простір, IPv6 відіграватиме важливу роль в обробці масштабованості мережевого рівня. Протокол обмежених застосувань, MQTT і ZeroMQ забезпечать легке перенесення даних. Туманні обчислення (Fog computing) є реальною альтернативою запобігання такому великому потоку даних, що проходить через інтернет. Обчислювальна потужність граничних мережевих пристроїв (маршрутизатори, комутатори) може використовуватися для аналізу і обробки даних, що забезпечує простоту масштабування в режимі реального часу.

Наприклад, застосування мережевих додатків зараз не викликає складнощів, оскільки доступні готові платформи швидкої розробки користувацьких проектів. Наприклад, в якості контролера буде використаний контролер з сімейства Arduino.



Рис. 2. Контролер Arduino Uno R3

Програмна реалізація

Програми, написані в середовищі розробки Arduino IDE для Arduino, називаються нарисами або скетчами. Нариси зберігаються на комп'ютері розробника у вигляді файлів з розширенням .ino.

Arduino IDE підтримує мови C і C++, використовуючи спеціальні правила структуризації. У Arduino IDE вбудована бібліотека з проекту Wiring, що надає доступ до безлічі простих процедур введення/виводу. Код, написаний користувачем, може містити всього 2 базових функції, для запуску нарису і основного циклу програми, які компілюються і зв'язуються з функцією-заглушкою main() у виконувану циклічну програму за допомогою GNU toolchain, також включеною в дистрибутив IDE. У Arduino IDE

використовується програма avrdude, що перетворює виконуваний код в текстовий файл в шістнадцятиричному форматі, який завантажується на плату Arduino програмою завантажувачем.

Однією з важливих деталей облаштувань інтернету речей є можливість сполучатися один з одним без участі користувача, відповідно такі пристрої мають бути підключені до мережі. У моєму випускній кваліфікаційній роботі пристрій також буде підключений до мережі, для забезпечення доступу до пристрою з будь-якої точки земної кулі, де є сигнал мережі.

На поточний момент часу існує величезна безліч різних стандартів, протоколів і специфікацій мережевих протоколів, що дозволяють пристроям виходити в мережу. Одна з найпоширеніших технологій – це Wi-Fi. У зв'язку з цим, мною було вирішено використати її для реалізації виходу пристрою в мережу і передачі даних на сервер.

Оцінювання пропускної спроможності мережі

Для розрахунку пропускної спроможності слід використати модель Бьянки (рис. 3). Основним вкладом моделі Бьянки є аналітичний розрахунок пропускної спроможності насичення в замкнутому виді. Модель також обчислює вірогідність збою передачі пакету із-за зіткнення. Вона припускає, що канал знаходиться в ідеальних умовах, тобто немає прихованого терміналу і ефекту захоплення. Бьянки використовує двовимірний ланцюг Маркова з $m + 1$ східцями відведення, в якій кожен етап представляє лічильник часу відстрочення вузла. Перехід відбувається при зіткненні і успішній передачі на «більш високу» стадію (наприклад, із стадії $i - 1$ на стадію i) і до найнижчого ступеня (тобто стадії 0), відповідно [9].

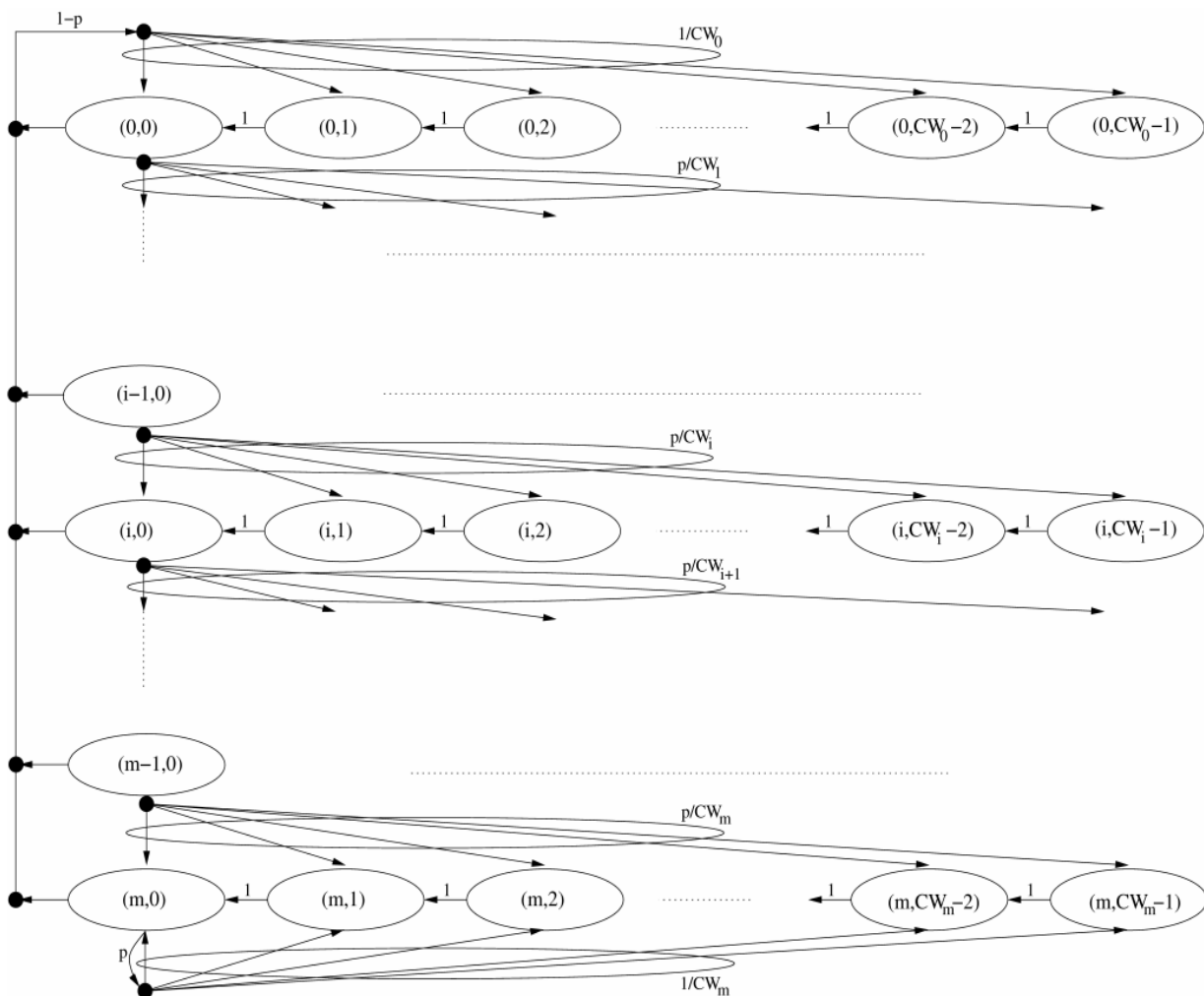


Рис. 3. Ланцюг Маркова для моделі Бьянки

Ця модель припускає, що в кожній спробі передачі, незалежно від кількості повторних передач, кожен пакет стикається з постійною і незалежною вірогідністю p виникнення цієї події.

Іншими словами, p – вірогідність того, що впродовж інтервалу часу, щонайменше, одна з $N-1$ станцій, що залишилися, також передає пакет даних. Якщо в стійкому стані кожна станція, що залишилася, передає пакет з вірогідністю π , p можна записати так:

$$p = 1 - (1 - \pi)^{N-1}.$$

Вірогідність доступу до каналу π вузла обчислюється як функція кількості рівнів ступеня відкату m , мінімального значення вікна конкуренції W_{min} і вірогідність зіткнення p :

$$\pi = \sum_{i=0}^m b_{i,o} = \frac{b_{o,o}}{1-p} = \frac{2(1-2p)}{(1-2p)(W_{\min} + 1) + pW_{\min}(1-(2p)^m)} = \frac{2}{W_{\min} + 1 + pW_{\min} \sum_{k=0}^{m-1} (2p)^k}$$

Для оцінювання проведено моделювання при різних значеннях рівня відкату m від 3 до 7. Результат моделювання наведено на рис. 4.

В цілому при кількості станцій до 1000 швидкість передачі даних зменшується до 3-х разів, а час затримки передачі не перевищує 10 секунд. В цілому такі швидкість і затримка цілком підходять для реалізації проєктів IoT, і можна сміливо говорити про те, що проєкт реалізовується практично на будь-якій території. Проте слід розуміти, що швидкість безпроводної передачі даних залежить від безлічі зовнішніх чинників і неможливо точно сказати, яка буде кінцева швидкість для певного варіанту реалізації.

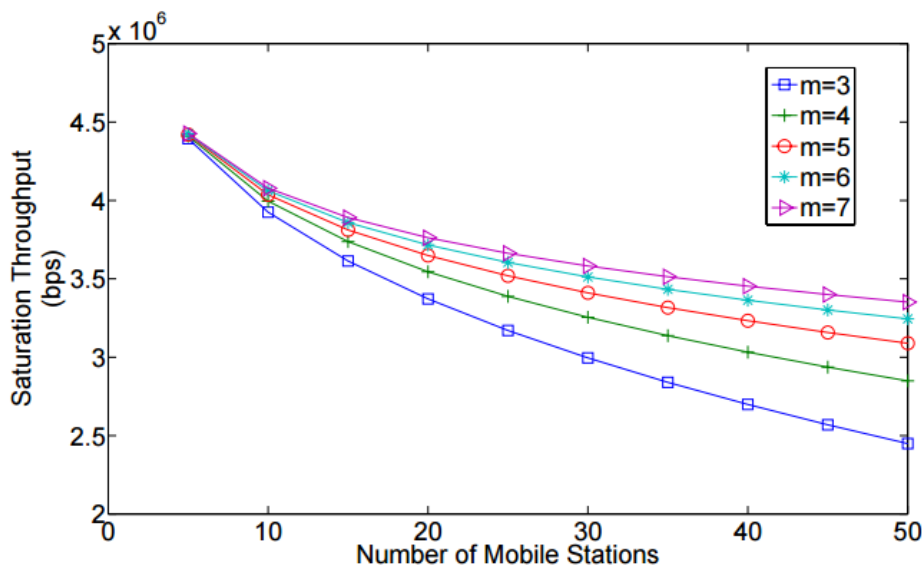


Рис. 4. Графік рівня відкату та вплив на пропускну здатність (Saturation throughput), Мбіт/с

Створення сервера

Для забезпечення доступу до даних, отриманих з пристрою, необхідно створити сервер, куди буде передаватися інформація з датчиків. Потужність сервера має бути достатня для забезпечення потрібної кількості запитів від датчиків весь період часу. Передача інформації від датчиків виникає зазвичай у детерміновані моменти часу. Використання програмованих контролерів зазвичай надає можливість вимірювання часових періодів, синхронізованих між собою, оскільки контролер містить точні опорні генератори.

Як показано на рис. 4, зростання таких датчиків може привести до виникнення повторних запитів на передачу інформації. А це обумовить досягнення стану неможливості обслуговування запитів. Виникне ситуація відмови в обслуговуванні – Denial of Service (DoS), відомий тип атаки на сервера для виведення їх з роботи.

Доступ до перегляду інформації, що зберігається на сервері зможе отримати будь-яка людина, що має вихід в інтернет та має доступ до серверу.

Для отримання даних з пристрою і їх подальшого відображення створюється **.php** файл з функціями очікування POST запиту. Виглядає така функція таким чином:

```
"$DataValue2=$_POST["humidity"];".
```

З таким функціоналом сторінка чекатиме від отримування POST-запиту і оновлюємо сторінку актуальною інформацією по отриманню такого запиту.

```
<?php
echo "<pre>";
echo "<br>Sensor state ..... : ";
echo $_POST["Sensor"];
echo "<br>Sensor voltage ... : ";
echo $_POST["Voltage"];
echo "<br>Sensor current ... : ";
echo $_POST["Current"];
echo "</pre>";
?>
```

Рис. 5. Приклад *php*-файла на стороні сервера

Висновки

Створення серверу обробки даних від пристроїв мережі інтернету речей вимагає аналізу періодичності отриманні запитів від кінцевих пристроїв до серверу. Оскільки інтервали між запитами можуть змінюватись, виникатиме ситуація, коли запити на вході серверу будуть накладатись в часі. Це призведе до виникнення повторних перевідправлень даних до сервера.

Зростання кількості датчиків збільшує ймовірність виникнення подібної ситуації. Також слід розраховувати на можливість перевантаження серверу запитами та можливості виникнення ситуації з відмовою в обслуговуванні запитів.

Література

1. Internet of Things [Електронний ресурс]. – Режим доступу : https://en.wikipedia.org/wiki/Internet_of_things (дата обращения 16.03.2017).
2. Privacy of the Internet of Things [Електронний ресурс]. – Режим доступу : <https://arxiv.org/ftp/arxiv/papers/1611/1611.03340.pdf> (дата обращения 22.04.2017)
3. Горященко, К.Л. Впровадження стандарту IEEE 1901 передачі інформації по провідних лініях / К. Л. Горященко, В. П. Нездоровін, В. С. Коваль // Вісник Хмельницького національного університету. Технічні науки. – 2017. – № 3. – С. 242–245.
4. Arduino [Електронний ресурс]. – Режим доступу : <https://en.wikipedia.org/wiki/Arduino> (дата обращения 15.03.2017).
5. Современные сетевые технологии, технологии Интернет [Електронний ресурс]. – Режим доступу : <http://www.itlab.unn.ru/Uploads/part2.pdf> (дата звернення 15.03.2017).

References

1. Internet of Things [Elektronnyi resurs]. – Rezhym dostupu : https://en.wikipedia.org/wiki/Internet_of_things (data obrashcheniya 16.03.2017).
2. Privacy of the Internet of Things [Elektronnyi resurs]. – Rezhym dostupu : <https://arxiv.org/ftp/arxiv/papers/1611/1611.03340.pdf> (data obrashcheniya 22.04.2017)
3. Horiashchenko, K.L. Vprovadzhennia standartu IEEE 1901 peredachi informatsii po provodovykh liniyakh / K. L. Horiashchenko, V. P. Nezdorovin, V. S. Koval // Visnyk Khmelnytskoho natsionalnoho universytetu. Tekhnichni nauky. – 2017. – № 3. – S. 242–245.
4. Arduino [Elektronnyi resurs]. – Rezhym dostupu : <https://en.wikipedia.org/wiki/Arduino> (data obrashcheniya 15.03.2017).
5. 5. Sovremennye setevye tehnologii, tehnologii Internet [Elektronnij resurs]. – Rezhim dostupu : <http://www.itlab.unn.ru/Uploads/part2.pdf> (data zvernennja 15.03.2017).

Рецензія/Peer review : 11.9.2018 р.

Надрукована/Printed : 8.11.2018 р.
Рецензент: д.т.н., доц. Любчик В.Р.