

до стандартів країн реєстрації автотранспортних засобів. Наприклад, в Україні з 2015 року використовується стандарт ДСТУ 4278:2012 [2].

Приклади спільної роботи послідовних етапів нормалізації зображення номера і сегментації символів наведено на рис. 2. У верхньому рядку показані вихідні зображення номерів, у другому рядку – результат їх нормалізації, в нижньому рядку – результат сегментації символів на номері (кожен символ являє собою окреме зображення).



Рис. 2. Приклади роботи алгоритмів нормалізації зображення номера і сегментації символів

Отримані після сегментації зображення символів дозволяють застосовувати їх для вирішення задачі розпізнавання. Запропонований алгоритм сегментації дозволяє також визначити тип номерного знака, а отже, з'ясувати чи кожен із символів буквою або цифрою, що дозволить полегшити аналіз зображень на наступному етапі розпізнавання. Запропоновані досить прості моделі дозволяють виконувати операцію сегментації з високою ефективністю за якістю і швидкості, а також дозволяють в разі потреби без значних додаткових зусиль розширити число використовуваних моделей номерного знаку шляхом введення в розгляд інших можливих розташувань символів.

Список використаних джерел

1. Дэвид А. Форсайт, Джин Понс [Computer Vision: A Modern Approach Компьютерное зрение. Современный подход]. — М.: «Вильямс», 2004. — 928 с.
2. Знаки номерні транспортних засобів (Державний стандарт ДСТУ 4278:2004) [Чинний від 2004-02-20].—К.:Держстандарт України, 2004. – 22 с. – (Національний стандарт України).

УДК 004.738.5:004.77

ПІДВИЩЕННЯ БЕЗПЕКИ ВИКОРИСТАННЯ ПРИСТРОЇВ ІНТЕРНЕТУ РЕЧЕЙ ЗА ДОПОМОГОЮ ТЕХНОЛОГІЇ BLOCKCHAIN

Склярова Д.Ю., студентка групи КІ-162

Науковий керівник: **Красножон О.В.**, старший викладач кафедри біомедичних радіоелектронних апаратів та систем
Чернігівський національний технологічний університет

Інтернет речей (Internet of Things, IoT) – концепція мережі, що складається із взаємозв'язаних фізичних пристроїв, які мають вбудовані датчики, а також програмне забезпечення, що дозволяє здійснювати передачу і обмін даними між фізичним світом і комп'ютерними системами, шляхом використання стандартних протоколів зв'язку. Окрім датчиків, мережа може мати виконавчі пристрої, які вбудовано у фізичні об'єкти, і пов'язано між собою через дротові чи бездротові мережі.

Основною концепцією IoT є можливість підключення багатьох приладів та речей, які людина може використовувати в повсякденному житті: холодильник, кондиціонер, автомобіль, велосипед і навіть кросівки. Всі такі об'єкти повинні оснащуватися вбудованими датчиками (сенсорами), які мають можливість перетворювати інформацію, що надходить із навколишнього середовища, обмінюватися нею і виконувати різні операції в залежності від отриманої інформації [1].

За даними компанії Statista, більш, ніж 23 мільярди пристроїв підключено у всьому світі за допомогою технології IoT. На рисунку 1 зображено статистику, що показує реально існуючу (2015 – 2018 роки) та прогнозовану кількість підключених пристроїв у світі з 2019 по 2025 роки. Так до 2020 року встановлена база пристроїв зросте майже до 31 мільярду [2].

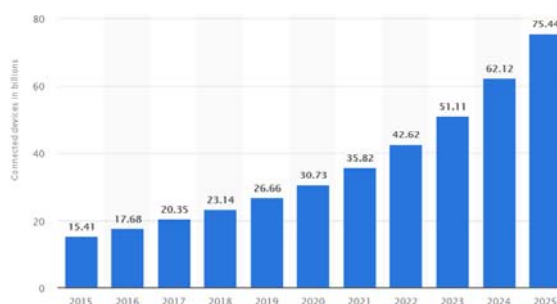


Рис. 1. Прогнозна статистика використання пристроїв IoT

Серед найбільш поширених виділяють 3 способи взаємодії з інтернет-речами:

- прямий доступ;
- доступ через інтернет-шлюз;
- доступ через сервер.

У разі прямого доступу, інтернет-речі повинні мати власну IP-адресу або мережний псевдонім, за яким до них і можна підключитися з клієнтського додатку, тобто, вони повинні виконувати функції веб-сервера. Доступ до таких речей зазвичай виконано у вигляді web-ресурсу з графічним інтерфейсом керування за допомогою веб-браузера.

Якщо інтернет-речі не мають вбудованої підтримки протоколів IP та HTTP, але підтримують приватні протоколи, наприклад, Bluetooth або ZigBee, то для взаємодії з ними можна використовувати спеціальний Інтернет-шлюз.

Третій спосіб взаємодії пристроїв IoT (через сервер) полягає у необхідності посередника між інтернет-речами і користувачем і може бути реалізований за допомогою посередницької платформи даних.

Узагальнену схему взаємодії пристроїв IoT зображено на рисунку 2.

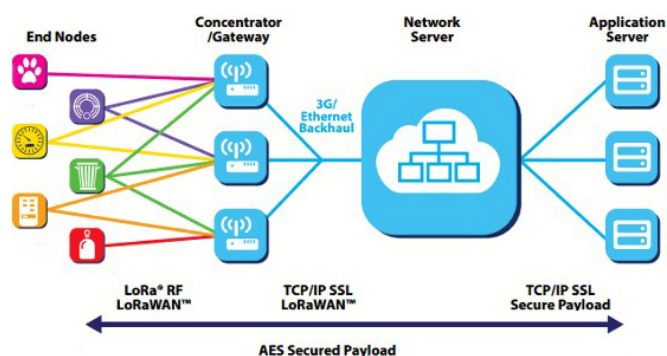


Рис. 2. Узагальнена схема взаємодії пристроїв Інтернету речей

Застосування мережі пристроїв IoT має наступні серйозні проблеми:

- можливість прослуховування/перехоплення даних, тобто збір інформації з метою атаки на мережу;
- вразливість мережі внаслідок помилок при конфігуруванні та адмініструванні;
- вразливість до фізичних атак на мережу з метою навмисного виведення з ладу пристроїв;
- нестабільність роботи пристроїв мережі через втрату електроживлення при раптових відімкненнях;
- відмови або збої у роботі через проблеми з програмним або апаратним забезпеченням;

Інтернет речей як технологія, що швидко розвивається, відчуває ряд «хвороб зростання», серед яких найбільш серйозною є проблема безпеки. Чим більше «розумних» пристроїв підключається, тим суттєвішими стають ризики, пов'язані з несанкціонованим доступом до IoT-мережі і використанням її можливостей зловмисниками. Сьогодні зусилля багатьох компаній і організацій в сфері IT координуються та спрямовуються на пошук рішень, які дозволять мінімізувати загрози безпеки мережі.

Так безпека Інтернету речей стала однією з найперших сфер використання технології Blockchain, що являє собою розподілену базу даних, яка зберігає впорядкований ланцюжок записів (так званих блоків). Завдяки застосуванню розподіленого реєстру, з'явилася можливість досягти високого рівня безпеки IoT-пристроїв в мережі і усунути існуючі обмеження та ризики, пов'язані з централізацією. В цілому, технологія Blockchain є поєднанням децентралізації даних з метою забезпечення довіри до них та забезпечує безпосередню взаємодію з іншим суб'єктами.

Вона дозволяє швидко і безпечно зберігати протоколи обміну і результати взаємодії різних IoT-пристроїв в децентралізованій системі. Саме розподілена архітектура Blockchain гарантує досить високу безпеку всієї IoT-системи. Навіть якщо частина пристроїв мережі є вразливою до зламу, в цілому, це не позначиться на загальній її роботі. Використання ботнетами «розумних» IoT-пристроїв стало можливим внаслідок їх слабкої захищеності. Розподілений тип довірчих відносин дозволяє визначити та відмовитися від використання ураженого пристрою без відчутної шкоди для взаємодії між «здоровими» об'єктами.

Крім того, це допоможе усунути вплив будь-якого окремого вузла та запобігти відмові усій мережі IoT-пристроїв. Впровадження технології Blockchain в IoT може допомогти у забезпеченні відповідального та безпечного спілкування між пристроями мережі.

Послідовність основних етапів проведення операцій (або транзакцій) технології Blockchain має вигляд:

- запит на транзакцію;
- запитана транзакція транслюється в мережу P2P, що складається з комп'ютерів-вузлів;
- мережа вузлів перевіряє транзакцію і стан користувача за допомогою певних алгоритмів;

- в результаті перевірки транзакція або об'єднується з іншими аналогічними для створення нового блоку даних або відкидається;
- новий блок додається до існуючого, таким чином, щоб він був постійним і незмінним;
- транзакцію завершено.

На рисунку 3 зображено послідовні етапи здійснення операцій у Blockchain на прикладі криптовалют.

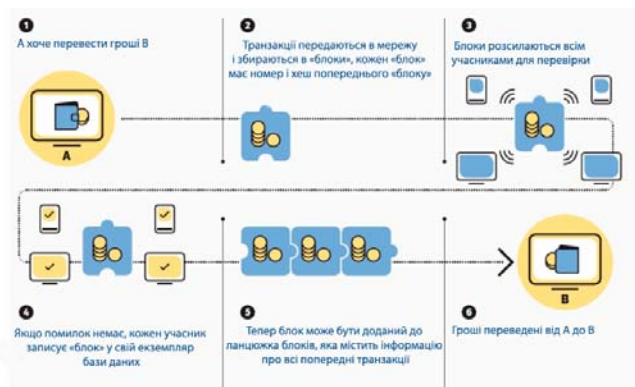


Рис. 3. Послідовність основних етапів проведення операцій (або транзакцій) технології Blockchain

Список використаних джерел

1. Інтернет речей [Електронний ресурс]. – Режим доступу: https://uk.wikipedia.org/wiki/Інтернет_речей.
2. Статистика інтернету речей [Електронний ресурс]. – Режим доступу: <https://www.statista.com>.
3. IoT и проблемы безопасности [Електронний ресурс]. – Режим доступу: <https://habr.com/ru/company/unet/blog/410849/>
4. What is Blockchain Technology? [Електронний ресурс]. – Режим доступу: <https://blockgeeks.com/guides/what-is-blockchain-technology/>

УДК 621.3

КЕРУВАННЯ СВІТЛОДІОДНОЮ RGB МАТРИЦЕЮ НА БАЗІ ПЛАТФОРМИ ARDUINO NANO V3.0 ЗА ДОПОМОГОЮ СМАРТФОНУ

Чеботар Б.Р., студент гр. КІ-1606

Шумська Л.М., керівник проекту, викладач-методист

Чернігівський національний технологічний університет
Коледж транспорту та комп'ютерних технологій ЧНТУ

Пристрої виводу інформації є одним із найпоширеніших типів представлення інформації і дозволяють отримувати інформацію у вигляді тексту, звуку, відео, фото і т. д.

Існує багато різноманітних пристроїв для візуального виводу інформації, наприклад: монітор, проектор, світлодіодні індикатори, матриці на світлодіодних індикаторах, рідкокристалічні дисплеї тощо.

Одним із цікавих напрямків виводу візуальної інформації є розробка пристроїв з використанням матриць на RGB світлодіодах.

RGB матриці можуть бути реалізовані з різними схемами керування: зі спільним анодом, зі спільним катодом, без спільного анода та катода, що допускає різні способи комутації (зазвичай випускається у вигляді SMD компонента). В RGB світлодіоді на одній підкладці встановлені незалежні кристали трьох кольорів світіння (R+G+B) [4].

Яскравість світлодіодів дуже добре піддається регулюванню, але не за рахунок зниження напруги живлення, що до речі не припустимо, а за допомогою методу широтно-імпульсної модуляції (ШІМ), для чого необхідний спеціальний керуючий блок (реально він може бути поєднаний з блоком живлення і конвертором, а також з контролером управління кольором RGB-матриці) [5].

Метод ШІМ полягає в тому, що на світлодіод подається не постійний, а імпульсно-модульований струм, причому частота сигналу повинна становити від сотень до тисяч герц, а ширина імпульсів і пауз між ними може змінюватися. Середня яскравість світлодіода стає керованою, в той же час світлодіод не гасне [5].

Конструктивне виконання матриць також є різноманітним, що дозволяє підібрати конкретну реалізацію для конкретного застосування в залежності від потреб. Це може бути виконання у вигляді квадратної матриці на жорсткій друкованій платі, на гнучкій підложці, різного формату. Було запропоновано використовувати RGB матрицю типу WS2812B.

Дуже зручно використовувати RGB матрицю з керуванням за допомогою смартфона. Для цього потрібно додатково використати мікроконтролер або платформу, наприклад, фірми Arduino, яка пропонує різноманітні платформенні набори з різноманітною периферією. Для даної реалізації було обрано Arduino