

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ

До захисту допускається
Т.в.о. завідувача кафедри
_____ Сафонова С.О.
« ____ » _____ 2020 р.

МАГІСТЕРСЬКА РОБОТА

НА ТЕМУ:

Бездротові сенсорні мережі в системах контролю стану промислового
підприємства

Освітньо-кваліфікаційний рівень “Магістр”
Спеціальність 123 “Комп’ютерна інженерія”

Науковий керівник роботи:

(підпис)

О.І. Рязанцев

(ініціали, прізвище)

Консультант з охорони праці:

(підпис)

Я.О. Критська

(ініціали, прізвище)

Студент:

(підпис)

Д.К. Смоляний

(ініціали, прізвище)

Група:

КСМ-16дм

Севєродонецьк 2020

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки

Кафедра комп'ютерних наук та інженерії

Освітньо-кваліфікаційний рівень “магістр”

Спеціальність 123 – “Комп'ютерна інженерія”

(шифр і назва)

Спеціалізація _____

(шифр і назва)

ЗАТВЕРДЖУЮ:

Т.в.о.зав. кафедри КНІ

к.т.н., доц. С.О. Сафонова

« _____ » _____ 2020 р.

З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ

Смолянному Денису Костянтиновичу

(прізвище, ім'я, по батькові)

1. Тема роботи Бездротові сенсорні мережі в системах контролю стану
Промислового підприємства

керівник проекту (роботи) Рязанцев О.І., д.т.н., професор

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу

від _____

№ _____

2. Строк подання студентом роботи 20.01.2020

3. Вихідні дані до роботи Матеріали науково-дослідної практики.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) _____

1. Аналіз літератури по темі дослідження.
2. Варіанти реалізації бездротових сенсорних мереж.
3. Побудова комбінованої мережі промислового підприємства.
4. Діаграма Вороного.
5. Охорона праці.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
Електронні плакати

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада Консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці та безпеки в надзвичайних ситуаціях	<i>ст. викл. Критська Я.О.</i>		

7. Дата видачі завдання 06.09.2019

Керівник _____

(підпис)

Завдання прийняв до виконання _____

(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Отримання завдання до магістерської роботи	<i>10.09.19-17.09.19</i>	
2	Аналіз завдання, огляд літератури	<i>18.09.19-25.09.19</i>	
3	Аналіз технічних засобів	<i>26.09.19-18.10.19</i>	
4	Розробка методу	<i>19.10.19-06.11.19</i>	
5	Програмна реалізація	<i>17.11.19-08.12.19</i>	
6	Охорона праці	<i>09.12.19-15.12.19</i>	
7	Оформлення пояснювальної записки	<i>16.12.19-29.12.19</i>	
8	Підготовка презентації та доповіді	<i>03.01.20-12.01.20</i>	

Студент _____

(підпис)

Смоляний Д.К. _____

(прізвище та ініціали)

Науковий керівник _____

(підпис)

Рязанцев О.І. _____

(прізвище та ініціали)

АНОТАЦІЯ

Смоляний Д.К. Бездротові сенсорні мережі в системах контролю стану промислового обладнання.

Метою роботи є дослідження та створення бездротової сенсорної мережі з метою моніторингу та аналізу умов праці на промислових підприємствах.

У даній роботі представлені найсучасніші технології з моніторингу та управління оточуючими приладами. Електронне управління забезпечує економію електрики, часу людей, надаючи величезний комфорт у керуванні. Зокрема описані технології ZigBee і MeshLogic, одну з яких будемо використовувати в роботі. Продемонстровані технології енергоефективні. Споживання електрики у них знаходиться на низькому рівні в порівнянні з можливими аналогами. А деякі системи можуть самі виробляти електрику, що робить їх повністю автономними незалежно від зовнішніх джерел енергії.

Ключові слова: zigbee, meshlogic, сенсор, датчик, мережа, БСМ, мульти-мікропроцесор, mesh-топология, кластери, специфікація, шифрування, маршрутизатор, підприємство, виробництво.

АННОТАЦИЯ

Смоляной Д.К. Беспроводные сенсорные сети в системах контроля состояния промышленного оборудования.

Целью данной магистерской аттестационной работы является исследование и создание беспроводной сенсорной сети с целью мониторинга и анализа условий труда на промышленных предприятиях.

В данной работе представлены самые современные технологии по мониторингу и управлению окружающими приборами. Электронное управление обеспечивает экономию электричества, времени людей, предоставляя огромный комфорт в управлении. В частности описаны технологии ZigBee и MeshLogic, одну из которых будем использовать в работе. Продемонстрированные технологии энергоэффективны. Потребление электричества у них находится на низком уровне по сравнению с возможными аналогами. А некоторые системы могут сами вырабатывать электричество, что делает их полностью автономными независимо от внешних источников энергии.

Ключевые слова: zigbee, meshlogic, сенсор, датчик, сеть, БСС, мульти-микропроцессор, mesh-топология, кластеры, спецификация, шифрование, маршрутизатор, предприятие, производство.

ABSTRACT

Smoliany D.K. Wireless sensor networks in industrial equipment monitoring systems.

The purpose of this attestation work is research and the creation of a wireless sensor network for the purpose of monitoring and analyzing working conditions in industrial enterprises.

This paper presents the most advanced technologies for monitoring and management of the surrounding devices. Electronic control provides power savings, giving people time to manage a huge comfort. Specifically described MeshLogic ZigBee and technology, one of which we will use in our work.

Demonstrated technologies a energy-efficient , the consumption of electricity from them at a low level in comparison with possible analogs. Some systems may themselves generate electricity, making them completely autonomous, independent of external power sources.

Key words: zigbee, meshlogic, sensor, detector, network, WSN, multi-, microprocessor, mesh-topology clusters specification, encryption, router, enterprise production.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ.....	8
ВСТУП.....	9
1 АНАЛІЗ ЛІТЕРАТУРИ ПО ТЕМІ ДОСЛІДЖЕННЯ.....	11
1.1 Стандарти бездротового зв'язку	11
1.1.1 WLAN (Wireless Local Area Network)	11
1.1.2 Міри безпеки в 802.11.....	14
1.1.3 WPAN (Wireless Personal Area Network).....	15
1.1.4 WMAN (Wireless Metropolitan Area Networks).....	15
1.1.5 WWAN (Wireless Wide Area Network)	16
1.2 Постановка задачі.....	16
2 ВАРІАНТИ РЕАЛІЗАЦІЇ БЕЗДРЮТОВИХ СЕНСОРНИХ МЕРЕЖ ...	18
2.1 Порівнення технологій MeshLogic і ZigBee.....	19
2.1.1 Фізичний рівень.....	19
2.1.2 Стійкість до співканальної інтерференції.....	20
2.1.3 Профілі ZigBee.....	22
2.1.4 Мережеві протоколи.....	24
2.1.5 Рівні, що визначаються стандартом IEEE 802.15.4.....	26
2.1.6 Типи мереж.....	26
2.1.7 Формування мережі ZigBee.....	28
2.1.8 Маршрутизація в мережах ZigBee.....	30
2.1.9 Масштабування мереж ZigBee в мережах.....	34
2.1.10 Бібліотека кластерів ZigBee (ZCL).....	35
2.1.11 Безпека ZigBee.....	37
2.1.12 Введення в експлуатацію.....	38
2.1.13 Енергоспоживання.....	40
2.1.14 Вимоги до апаратних ресурсів.....	41
2.2 Про розвиток модулів ZigBee.....	41
2.2.1 Історія змін програмного забезпечення модулів.....	42
2.2.2 Режим відображення портів вводу-виводу.....	44
2.2.3 Внутрішня структура модулів і використання ПЗ від Freescale.....	44
2.2.4 XBee і Xbee-PRO S2C – нові ZigBee модулі від DIGI.....	45
2.2.5 ZigBee XBee ZNet 2.5 (Series 2).....	48
3 ПОБУДОВА КОМБІНОВАНОЇ МЕРЕЖІ ПРОМИСЛОВОГО ПІДПРИЄМСТВА	52
3.1 Вибір датчиків.....	53
3.2 Розподіл датчиків в приміщенні.....	55
4 ДІАГРАМА ВОРОНОГО.....	58
4.1 Історія виникнення.....	58
4.2 Принцип побудови.....	58
4.3 Алгоритми побудови.....	60
4.3.1 Алгоритм побудови діаграми Вороного «у лоб».....	60
4.3.2 Алгоритм побудови діаграми Вороного шляхом перетину напівплощин за $O(n^2 \cdot \log(n))$	60
4.3.3 Алгоритм Форчуна побудови діаграми Вороного на площині.....	64
4.3.4 Рекурсивний алгоритм побудови діаграми Вороного.....	65
4.4 Використання діаграми Вороного для розподілу елементів БСМ нашого підприємства.....	66
4.4.1 Опис підприємства.....	66
4.4.2 Проектування мережі ZigBee.....	67

Перелік посилань до розділів 1-4.....	69
5 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ	71
5.1 Загальні дані з охорони навколишнього природного середовища	71
5.2 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу.....	71
5.3 Гігієнічні вимоги до параметрів виробничого середовища	73
5.3.1 Освітлення	73
5.3.2 Шум та вібрація, електромагнітне випромінювання	74
5.3.3 Вентилювання	75
5.4 Висновки до розділу 5	79
5.5 Перелік джерел посилань до розділу 5.....	80
ВИСНОВКИ	81
ДОДАТОК А - Графічний матеріал магістерської атестаційної роботи	82
ДОДАТОК Б - Код реалізації побудови алгоритму Вороного шляхом перетину напівплощин	89

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ

ПЗ - Програмне забезпечення
ПК - Персональний комп'ютер
АСУ - Автоматична система управління
LAN – Local Area Network
WAN – World Area Network
ОС – Операційна система
IEEE – Institute of Electrical and Electronics Engineers
БСМ - Бездротова сенсорна мережа
WLAN – Wireless Local Area Network
WPAN – Wireless Personal Area Network
WMAN – Wireless Metropolitan Area Network
MAC – (Media Access Control) управління доступом до середовища
PHY – (Physical layer) управління фізичним рівнем
DHCP – (Dynamic Host Configuration Protocol) протокол динамічного надання мережевих параметрів
DNS – (Domain Name System) система доменних імен
Mesh – мережева топологія мереж на принципі осередків
ІЧ – інфрачервоний
APL – рівень додатків
APS – підрівень підтримки додатків
ZDO – ZigBeeDevice Objес, об'єкт пристрою ZigBee
ZDP – ZigBee Device Profile, профіль пристрою ZigBee
NWK – Network Layer, мережевий рівень
PHY – Physical Layer, фізичний рівень.
SAP – service access point, точки доступу до послуг
SSP – Security Service Provider
ML-SM – бездротова система моніторингу

ВСТУП

В останні роки бездротові сенсорні мережі активно переходять зі стадії науково-дослідних робіт в практичну площину - на ринку з'являється все більше систем і окремих продуктів, в яких в тій чи іншій мірі використовуються технології БСМ. Це відбувається завдяки як розвитку елементної бази (зниження енергоспоживання, вартості і габаритних розмірів радіочастотних приймачів, мікроконтролерів і датчиків), так і появи промислових стандартів (наприклад, ZigBee, 6LowPAN, ISA100.11a і ін.), а також зростанню інтересу до концепції Internet of Things, невід'ємною частиною якої є БСМ. У загальному випадку сенсорними мережами називають бездротові багатоосередкові (mesh) мережі з низькою швидкістю передачі даних і наднизьким енергоспоживанням, основне призначення яких полягає в зборі свідчень від розподілених в просторі датчиків.

Головними особливостями БСМ є самоорганізація і адаптивність до умов експлуатації, тому потрібні мінімальні витрати при розгортанні такої мережі на об'єкті і при подальшому її супроводі в процесі функціонування. Підтримка багатоосередкової топології і здатність вузлів в разі необхідності ретранслювати повідомлення інших пристроїв забезпечують значну площу покриття системи при малій потужності передавачів і стійкість мережі до відмови окремих вузлів або бездротових з'єднань з різних причин (поява перешкод, фізичне пошкодження або переміщення вузла і т. Д.).

Такий підхід дозволяє експлуатувати мережу в будівлях і на промислових об'єктах з несприятливими умовами поширення радіохвиль. При цьому вузли самостійно визначають оптимальні маршрути доставки даних і коригують їх в режимі реального часу при зміні топології мережі. Апаратне забезпечення бездротових вузлів і протоколи мережевої взаємодії між ними оптимізовані по енергоспоживанню для досягнення тривалого терміну експлуатації системи при автономних джерелах живлення (батареї, акумулятори тощо). Залежно від режиму роботи час життя вузла може досягати декількох років.

Як правило, під час обговорення БСМ передбачається, що мережа має багатоосередкову топологію, в якій кожен вузол має кілька сусідів, тобто забезпечується комунікаційна надмірність, що в поєднанні зі здатністю вузлів самостійно автоматично визначати альтернативні шляхи доставки інформації в обхід пошкоджених елементів мережі забезпечує високий рівень надійності і відмовостійкості системи телеметрії. У багатьох додатках це дійсно має місце, але практичний досвід впровадження БСМ диктує необхідність враховувати наступні обставини.

У типових бездротових системах збору даних безліч сенсорних вузлів періодично (або по якомусь більш складному сценарію) опитують підключені до них датчики і передають по радіоканалу результати вимірювань шлюзу, який в свою чергу надає зібрані дані вищестоящим системам з різних дротових або бездротових каналах зв'язку. Тобто інформація від багатьох джерел надходить в одну точку збору, тому надійність всієї системи телеметрії залежить від ступеня доступності шлюзу.

Для зниження цієї залежності рекомендується використовувати в системі кілька шлюзів, при цьому в залежності від вимог прикладної задачі вузли можуть передавати пакети тільки найближчого шлюзу або всім шлюзіам для резервування даних. Крім того, установка в мережі декількох шлюзів дозволяє більш рівномірно розподілити мережевий трафік між вузлами, щоб їх елементи живлення розряджалися приблизно з однаковою швидкістю.

Наприклад, в бездротовій системі моніторингу ML-SM може бути 1 ... 4 шлюзів.

У числі основних функціональних і експлуатаційних переваг БСМ слід виділити:

- можливість самонастроювання і самовідновлення мережі;
- масштабованість мереж з щільним розміщенням вузлів у просторі (від десятків до тисяч пристроїв);
- високі показники надійності і відмовостійкості за рахунок комунікаційної надмірності і наявності безлічі альтернативних маршрутів доставки даних;
- вартість і малі масогабаритні показники вузлів;
- енергетична ефективність (термін експлуатації може досягати декількох років при автономному електроживленні вузлів);
- адаптованість до модифікації топології мережі і змін характеристик середовища поширення радіохвиль;
- здатність вузлів спільно обробляти отримані дані і приймати рішення на базі розподілених алгоритмів;
- можливість швидкого і при необхідності прихованого розгортання мережі.

Інтерес до технологій БСМ постійно зростає, оскільки їх застосування дозволяє вирішувати завдання, в яких використання традиційних провідних каналів зв'язку обмежене або неможливе з технічних, економічних або організаційних причин. Але поки досвід практичного використання подібних систем відносно малий, а кожна нова прикладна задача має свої особливості, тому на поточному етапі розвитку даного напрямку складно говорити про наявність типових рішень в різних додатках [1].

1 АНАЛІЗ ЛІТЕРАТУРИ ПО ТЕМІ ДОСЛІДЖЕННЯ

Бездротові сенсорні мережі - це нова перспективна технологія, на основі якої інтенсивно ведуться прикладні розробки і виконуються Масштабні проекти для різних галузей промисловості і систем військового призначення. Так, наприклад, на основі розробленого вченими університетів США прикладного програмного забезпечення на базі ВПС у Флориді розгорнута БСМ, що забезпечує виявлення проникнення і переміщення людей і транспортних засобів в контрольовані зони. Мережа охоплює зону площею 5 кв.км. і оснащена сенсорами з магнітоелектричними і температурними датчиками для розпізнавання і відстеження різних металевих об'єктів, в тому числі рухомих.

У числі відомих прикладів прикладного використання БСМ - розгортання мережі, побудованої на основі обладнання Intel на борту нафтового танкера компанії BP для моніторингу і профілактичного обслуговування судна. Робота БСМ проходила в умовах екстремальних температур, високої вібрації та значного рівня радіочастотних перешкод від судових систем. При цьому розгорнута БСМ стійко функціонувала, автоматично здійснюючи реконфігурацію і відновлення працездатності при впливі екстремальних факторів. За останній період на розгортання БСМ для моніторингу інфраструктури світової нафто-газовидобувної галузі компанії витрачали щорічно більше 200 млн.\$.

1.1 Стандарти бездротового зв'язку

Різних специфікацій стандартів бездротових мереж сімейства 802.1x існує велика кількість (для позначення одних тільки різновидів стандарту 802.11 використовуються практично всі букви англійського алфавіту). Проте, всі вони поділяються на чотири великі категорії WPAN, WLAN, WMAN, WWAN.

1.1.1 WLAN (Wireless Local Area Network)

Ця категорія бездротової мережі призначена для зв'язку між собою різних пристроїв, подібно LAN на основі виті пари або оптоволокна, і при цьому характеризується високою швидкістю передачі даних на відносно невеликі відстані. Взаємодія пристроїв описується сімейством стандартів IEEE 802.11, що включає в себе більше 20 специфікацій.

Короткий перелік стандартів можна побачити на рисунках 1.1 і 1.2.

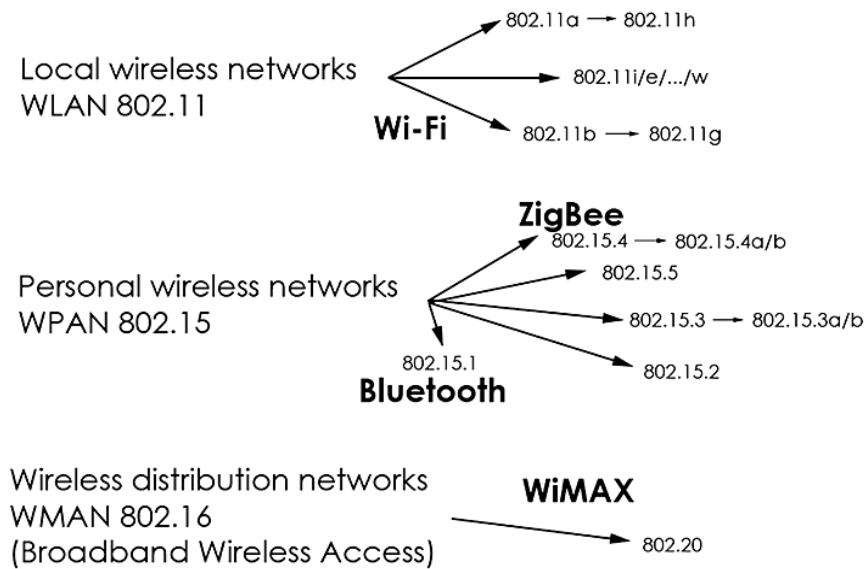


Рисунок 1.1 - Стандарти бездротових мереж сімейства 802.1x

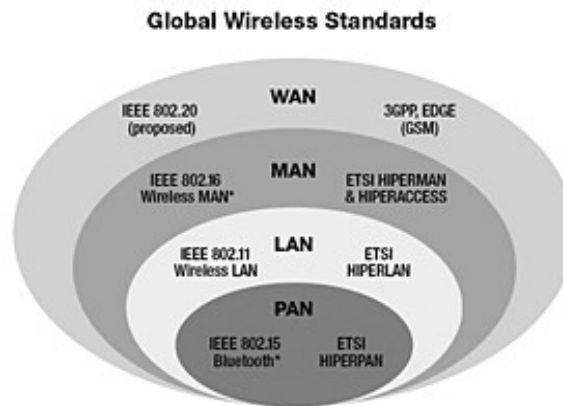


Рисунок 1.2 - Стандарти бездротових мереж

У зв'язку з цим, багато хто помилково не бачать різниці між Wi-Fi і IEEE 802.11. В даний час під Wi-Fi розуміється торгова марка, яка показує, що конкретний пристрій відповідає специфікаціям 802.11a, 802.11b, 802.11g. Таким чином, сімейство IEEE 802.11 можна розділити на три класи 802.11a, 802.11b, 802.11 i / e / ... / w.

IEEE 802.11a - один зі стандартів бездротових локальних мереж, що описує принципи функціонування пристроїв в частотному діапазоні ISM (смуга частот 5,155,825 ГГц) за принципом OFDM (Orthogonal Frequency Division Multiplexing, мультиплексування з поділом по ортогональних частотах). Смуга поділяється на три робочі зони шириною 100 МГц, і для кожної зони визначена максимальна випромінювана потужність 50 мВт, 250 мВт, 1 Вт. Передбачається, що остання зона частот буде використовуватися для організації каналів зв'язку між будівлями або зовнішніми об'єктами, а дві інші зони всередині них. Редакцією стандарту, затвердженою в 1999 р,

визначені три обов'язкових швидкості 6, 12 і 24 Мб/с і п'ять необов'язкових 9, 18, 36, 48 і 54 Мб/с. Однак цей стандарт не прийнятий внаслідок використання частини цього діапазону відомчими структурами. Можливим вирішенням цієї проблеми може стати специфікація 802.11h, яка доповнена алгоритмами ефективного вибору частот для бездротових мереж, а також засобами управління використанням спектру, контролю над випромінюваною потужністю, а також генерації відповідних звітів. Радіус дії пристроїв в закритих приміщеннях становить близько 12 метрів на швидкості 54 Мб / с і до 90 метрів при швидкості 6 Мб / с. У відкритих приміщеннях або в зоні прямої видимості близько 30 метрів (54 Мб / с), і до 300 метрів при 6 Мб / с. Тим не менш, деякі виробники впроваджують у свої пристрої технології прискорення, завдяки яким можливий обмін даними в Turbo 802.11a на швидкостях до 108 Мб / с.

IEEE 802.11b перший стандарт, який отримав широке поширення (саме він спочатку носив торгову марку Wi-Fi) і дозволив створювати бездротові локальні мережі в офісах, будинках, квартирах. Ця специфікація описує принципи взаємодії пристроїв в діапазоні 2,4 ГГц (2,42,4835 ГГц), розділеному на три канали, що не перекриваються, за технологією DSSS (Direct-Sequence Spread-Spectrum, широкої смуги з прямим розширенням спектра) і, опціонально, PBCC (Packet Binary Convolutional Coding, двійкове надточне кодування). Відповідно до цієї технології модуляції, проводиться генерування надмірного набору бітів на кожен переданий біт корисної інформації. Завдяки цьому здійснюється більш висока ймовірність відновлення переданої інформації і краща перешкодозахищеність (шуми і перешкоди ідентифікуються як сигнал з неоднаковим набором бітів і тому фільтруються). Стандартом визначено чотири обов'язкові швидкості 1, 2, 5,5 і 11 Мб / с. Що ж стосується можливого радіусу взаємодії пристроїв, то він становить в закритих приміщеннях близько 30 метрів на швидкості 11 Мб / с, і до 90 метрів при швидкості 1 Мб / с.

У відкритих приміщеннях або в зоні прямої видимості близько 120 метрів (11 Мб / с), і до 460 метрів при 1 Мб / с. В умовах постійно зростаючих потоків даних ця специфікація практично вичерпала себе, і на зміну їй прийшов стандарт IEEE 802.11g.

IEEE 802.11g стандарт бездротової мережі, що з'явився логічним розвитком 802.11b в тому сенсі, що використовує той же частотний діапазон і припускає зворотну сумісність з пристроями, що відповідають стандарту 802.11b (іншими словами, обов'язкове сумісність 802.11g-обладнання з більш старої специфікацією 802.11b) . Одночасно з цим, цей представник сімейства специфікацій, як і годиться, спробував взяти все найкраще від піонерів 802.11b і 802.11a. Отже, основний принцип модуляції запозичений у 802.11a OFDM спільно з технологією CCK (Complementary Code Keying, кодування

комплементарним кодом), а додатково передбачено використання технології РВСС. Завдяки цьому, в стандарті передбачені шість обов'язкових швидкостей 1, 2, 5,5, 6, 11, 12, 24 Мб / с, і чотири опціональних 33, 36, 48 і 54 Мб / с.

Радіус зони дії збільшений в закритих приміщеннях до 30 метрів (54 Мб / с), і до 91 метра при швидкості 1 Мб / с. У межах прямої видимості зв'язок доступний на відстані 120 метрів зі швидкістю 54 Мб / с, а при видаленні на 460 метрів можлива робота зі швидкістю 1 Мб / с ..

Виділений в окремий клас набір специфікацій 802.11 і / е /.../ в головному чином призначений для опису функціонування різних службових компонентів і розробки нових технологій і стандартів бездротового зв'язку. Наприклад, роботи бездротових мостів, вимог до фізичних параметрів каналів (потужність випромінювання, діапазони частот), специфікацій, орієнтованих на різні категорії користувачів і т. Д. У плані надбудов і нових стандартів організації бездротових мереж з цієї групи вже розглянули 802.11.h. В якості ще одного прикладу звернемо увагу на 802.11n. Згідно з повідомленням міжнародного консорціуму EWC (Enhanced Wireless Consortium), використання 802.11n високошвидкісного стандарту, в якому передбачена зворотна сумісність з 802.11a / b / g, а швидкість передачі даних буде досягати 600 Мб / с. Це дозволить використовувати його в задачах, де використання Wi-Fi обмежувалося недостатньою швидкістю [2].

1.1.2 Міри безпеки в 802.11

Спочатку в сімействі протоколів був передбачений комплекс заходів безпеки, об'єднаний під загальною назвою WEP (Wired Equivalent Privacy, безпека, еквівалентна провідній мережі). Незважаючи на те, що WEP підтримується більшістю обладнання 802.11, виявилось, що він має найбільш низький рівень захисту. Причина уразливості WEP полягає в використанні статичних ключів шифрування, відомих всім станціям, а також в тому, що не передбачений процес перевірки автентичності користувача. Для відновлення ключів необхідно внести відповідні зміни на кожному комп'ютері, а якщо залишати криптографічні ключі незмінними, то зламати мережу досить просто. На деяких сайтах для ілюстрації уразливості WEP навіть наводяться покрокові інструкції для злому.

З огляду на уразливості WEP, була розроблена нова технологія захисту WPA (Wi-Fi Protected Access). Технологія WPA лише нагадує WEP і, відповідно до специфікації, щоб включати в себе протокол TKIP (Temporal Key Integrity Protocol), що працює в зв'язці з механізмами 802.1x. Підвищення рівня безпеки забезпечується періодичним генеруванням унікального ключа для кожного користувача. Проте, стандарт WPA піддається атакам

типу DDoS. Для цього достатньо, щоб на сервер доступу кожен секунду приходило кілька невірних запитів на ідентифікацію, при цьому сервер визначає спробу несанкціонованого доступу і виробляє розрив всіх з'єднань на деякий час (близько хвилини). Таким чином, постійна відправка невірних даних може привести до нестабільної роботи.

Наступний розроблений стандарт WPA2. Його відмінність від WPA полягає в більш стійкому до злому алгоритмі шифрування AES (Advanced Encryption Standard) і модифікованому алгоритмі управління ключами.

1.1.3 WPAN (Wireless Personal Area Network)

WPAN бездротова мережа, призначена для організації бездротового зв'язку між різного типу пристроями на обмеженій площі (наприклад, в рамках квартири, офісного робочого місця). Стандарти, що визначають методи функціонування мережі, описані в сімействі специфікацій IEEE 802.15. Одні з найбільш перспективних стандартів 802.15.3, ZigBee і MeshLogic.

IEEE 802.15.3 розроблявся як високошвидкісний стандарт WPAN-мереж для високотехнологічних побутових пристроїв (призначених, як правило, для передачі мультимедійних даних). Використання смуги 2,4 ГГц і технології модуляції OQPSK (Offset Quadrature Phase Shift Keying, квадратурна маніпуляція фазовим зрушенням зі зміщенням) дозволяють досягати швидкості передачі в 55 Мб / с на відстань до 100 метрів. Захист даних може здійснюватися за стандартом AES. У модифікації стандарту 802.15.3a передбачається збільшити пропускну здатність до 480 Мб / с, а в разі специфікації 802.15.3b пропускну спроможність складе від 100 до 400 Мб / с. ZigBee і MeshLogic розглянемо окремо, так як це дві дуже схожі технології і обидві мають місце бути використані в наших цілях [3].

1.1.4 WMAN (Wireless Metropolitan Area Networks)

WMAN - бездротові мережі масштабу міста. Надають широкосмуговий доступ до мережі через радіоканал. Стандарт IEEE 802.16, опублікований в квітні 2002 року, описує wireless MAN Air Interface. 802.16 - це, так звана, технологія «останньої милі», яка використовує діапазон частот від 10 до 66 GHz. Так як це сантиметровий і міліметровий діапазон, то необхідна умова «прямої видимості». Стандарт підтримує топологію point-to-multipoint, технології frequency division duplex (FDD) і time division duplex (TDD), з підтримкою quality of service (QoS). Можлива передача звуку і відео. Стандарт визначає

пропускну здатність 120 Мбіт / с на кожен канал в 25 MHz. Стандарт 802.16a пішов за стандартом 802.16. Він був опублікований в квітні 2003 і використовує діапазон частот від 2 до 11 GHz. Стандарт підтримує комірчасту топологію (mesh networking). Стандарт не накладає умову «прямої видимості» [4].

1.1.5 WWAN (Wireless Wide Area Network)

WWAN - бездротова глобальна обчислювальна мережа, різновид бездротових комп'ютерних мереж для доступу в інтернет.

Глобальні бездротові мережі WWAN відрізняються від локальних бездротових мереж WLAN тим, що для передачі даних в них використовуються бездротові технології стільникового зв'язку, такі як UMTS, GPRS, CDMA2000, GSM, CDPD, Mobitex, HSDPA, 3G або WiMAX (хоча останню правильніше відносити до WMAN - бездротових мереж масштабу міста). Відповідні послуги зв'язку пропонуються, як правило, на платній основі операторами регіонального, національного або навіть глобального масштабу. Технології WWAN дають можливість користувачеві, наприклад, з ноутбуком і WWAN-адаптером отримувати доступ до Всесвітньої павутини, користуватися електронною поштою і підключатися до віртуальних приватних мереж з будь-якої точки в межах зони дії оператора бездротового зв'язку. Багато сучасних портативних комп'ютерів мають вбудовані адаптери WWAN (наприклад, HSDPA) [5].

1.2 Постановка задачі

Об'єктом дослідження є застосовувані сучасні технології для зменшення енергоспоживання промислового підприємства, що дозволить підвищити безпеку персоналу і підприємства в цілому.

Предметом дослідження є промислове підприємство.

У комплексній системі будуть використані такі датчики: датчики освітленості, вологості, температури, обсягу приміщення, присутності.

Ці датчики будуть регулярно сканувати приміщення і передавати дані через шлюз. А за допомогою ПЗ на ПК можна буде регулювати освітленість і вентиляцію приміщення і відключати електрику, коли в приміщенні нікого не буде.

Датчик об'єму приміщення також може використовуватися для охоронних систем, разом з датчиком відкриття дверей і датчиком присутності.

Територія підприємства в довжину 150 метрів і 100 метрів завширшки. Товщина стін може досягати 0.5 метра. Через таку архітектуру радіохвилі не можуть ефективно поширюватися. Ця особливість архітектури буде враховуватися в проектуванні мережі.

Комп'ютерна мережа буде об'єднувати різні корпуси підприємства, між якими буде кабельне з'єднання, а в корпусах радіо.

Метою роботи є дослідження та створення бездротової сенсорної мережі з метою моніторингу та аналізу умов праці на промислових підприємствах. Для досягнення поставленої мети необхідно вирішити такі завдання:

- виконати огляд літератури за вибором комплектуючих вузла зв'язку для промислового об'єкта;
- дослідження існуючих рішень в області БСМ;
- розробка плану впровадження БСМ в підприємства з варіантами установки сенсорів;
- оптимізація обсягів використання необхідного обладнання.

2 ВАРІАНТИ РЕАЛІЗАЦІЇ БЕЗДРОТОВИХ СЕНСОРНИХ МЕРЕЖ

При виникненні завдання, що вимагає застосування технологій БСМ, розробники мають кілька варіантів вирішення проблеми. Перший варіант - побудова власної системи на основі стандарту ZigBee. Сьогодні є досить велика кількість різних ZigBee-продуктів, починаючи від мікросхем приймачів стандарту IEEE 802.15.4 і закінчуючи готовими OEM-модулями з вбудованим програмним забезпеченням мережевого стека ZigBee. Переваги такого підходу - відносна простота і малі терміни отримання працездатного рішення. Недолік - необхідність в витраті часу і ресурсів як на детальне вивчення самого стандарту ZigBee, так і на освоєння реалізації цієї технології в обраному продукті, тому що потрібне глибоке розуміння закладених в ZigBee механізмів для побудови ефективних ZigBee-мереж в реальних умовах експлуатації. Прискорити процес освоєння можна за допомогою налагоджувальних засобів, які є практично у всіх виробників ZigBee-продуктів, однак вартість таких наборів розробника є досить високою. Крім того, при вирішенні поставленого завдання розробник обмежений рамками самого стандарту, тому не завжди можливо отримати «елегантне» рішення, а будь-який відступ від специфікації автоматично позбавить систему сумісності - основної переваги ZigBee.

Іншим варіантом є застосування однієї з доступних пропрієтарних платформ (наприклад, від Sensicast, Millennial Net або Dust Networks). У деяких з них використовуються власні приймачі, а інші засновані на стандарті IEEE 802.15.4. Мережевий стек пропрієтарних платформ реалізований на базі патентованих алгоритмів і протоколів, які забезпечують ряд переваг в порівнянні з ZigBee, але ціною несумісності з рішеннями від інших виробників.

Як правило, подібна продукція являє собою набір апаратних модулів з різною функціональністю, вбудованого ПЗ вузлів і ПЗ персонального комп'ютера, для налагодження та управління мережею. Недоліком цього підходу є закритість пропрієтарної платформи і відсутність можливості внесення змін до логіки мережевої взаємодії для більш точного задоволення вимог конкретного застосування.

Отже, подібні платформи підходять для інтеграції в більшу систему в якості єдиного цілого без будь-якої модифікації. Можна також спробувати самостійно розробити набір мережевих протоколів і реалізувати їх на власній апаратній базі. Таким чином може бути отримано рішення, що найбільш точно відповідає прикладній задачі, але часові та фінансові витрати на розробку можуть бути занадто великі, тому що для цього потрібні серйозні знання в області бездротових сенсорних мереж, що мають масу особливостей і відмінностей від традиційних систем передачі даних. Слід також зазначити, що

налагодження мережі всього з декількох бездротових вузлів вже є досить нетривіальним завданням і вимагає високої кваліфікації розробника.

Таким чином, на даний момент часу існуючі комерційні рішення в області БСМ не відрізняються гнучкістю, яка необхідна для отримання ефективної мережі в конкретному додатку, а також складні в освоєнні нефахівцями в даній області. В результаті підвищуються вартість і терміни розробки системи.

Саме тому було прийнято рішення про створення апаратно-програмної платформи MeshLogic для реалізації бездротових сенсорних мереж в різних областях застосування. Платформа не є готовим продуктом (виробом), який надається кінцевому споживачу, тому що передбачається її використання в якості бази для створення різних бездротових систем під конкретні вимоги замовника і з урахуванням специфіки прикладної задачі. Крім того, платформа може використовуватися як інструмент для проведення досліджень з розробки нових алгоритмів і протоколів в області БСМ.

При розробці платформи MeshLogic ставилося завдання використовувати найбільш сучасні досягнення в області БСМ, а також враховувати переваги і недоліки існуючих комерційних рішень, в тому числі стандарту ZigBee.

Звісно, виникає резонне питання про відмінності платформи MeshLogic від інших існуючих рішень, тому нижче наведено короткий порівняльний аналіз платформи MeshLogic зі стандартом ZigBee. Порівняння з іншими комерційними платформами важко в силу відсутності їх докладного опису [6].

2.1 Порівнення технологій MeshLogic і ZigBee

Далі будуть розглянуті основні характеристики стандарту ZigBee і платформи MeshLogic в різних аспектах, з описом відмінностей цих технологій.

2.1.1 Фізичний рівень

У ZigBee фізичний рівень і рівень доступу до середовища відповідають стандарту IEEE 802.15.4, в якому визначені для використання наступні неліцензовані частотні діапазони: 868 ~ 868,6 МГц, 902 ~ 928 МГц і 2405 ~ 2483,5 МГц.

Як і слід було очікувати, найбільша частина вироблених пристроїв працює в діапазоні 2,4 ГГц, тому будемо розглядати тільки цю область частот.

В IEEE 802.15.4 діапазон 2,4 ~ 2,4835 ГГц розділений на 16 каналів шириною 2 МГц з кроком 5 МГц.

Використовуються квадратурна фазова маніпуляція із зсувом (O-QPSK) і розширення спектра методом прямої послідовності 16 фіксованими квазіортогональними 32-розрядними кодами. В результаті технічна швидкість передачі даних в ZigBee становить 250 Кбіт/с при коефіцієнті розширення спектра (processing gain), що дорівнює 9 дБ.

Поточна реалізація бездротового вузла MeshLogic побудована на основі приймачів Cypress WirelessUSBTM, які також працюють в діапазоні 2,4 ~ 2,4835 ГГц з розбивкою на 78 каналів (з урахуванням необхідного рознесення 39 каналів) шириною 1 МГц. У WirelessUSB застосовуються гауссова частотна маніпуляція і розширення спектра ортогональними послідовностями Голда довжиною 32 або 64 елементарних символу. Приймач WirelessUSB дозволяють змінювати розширюють коди, тому можлива також організація множинного доступу з кодовим розділенням. Залежно від обраного режиму роботи швидкість передачі даних може бути 15625 біт / с, 31250 біт / с або 62500 біт / с, при цьому коефіцієнт розширення спектра становить 18, 15 і 12 дБ відповідно [7].

Таким чином, ZigBee і MeshLogic функціонують в загальнодоступному неліцензованому діапазоні частот 2,4 ~ 2,4835 ГГц, при цьому в ZigBee швидкість передачі даних вище, але фізичний рівень платформи MeshLogic володіє більшою стійкістю до перешкод. Крім того, для типових додатків БСМ швидкість передачі даних не є критичним параметром, в той час як втрати пакетів безпосередньо впливають на якість обслуговування мережі та енергоспоживання вузлів.

Слід зазначити, що принципово платформа MeshLogic дозволяє використовувати на фізичному рівні і інші приймачі, аналогічні Cypress WirelessUSB, в тому числі відповідні стандарту IEEE 802.15.4.

2.1.2 Стійкість до співканальної інтерференції

З огляду на зростання кількості бездротових систем в неліцензованому діапазоні частот 2,4 ~ 2,4835 ГГц, БСМ повинна мати механізми підвищення стійкості до соканальної інтерференції.

Наприклад, в офісній будівлі велика ймовірність появи однієї або декількох мереж стандарту IEEE 802.11b / g, які можуть бути серйозною перешкодою для мережі ZigBee або MeshLogic. Як правило, в ZigBee і MeshLogic потужність корисного сигналу становить приблизно 1 мВт при ширині смуги 2 МГц і 1 МГц відповідно, в той час як в IEEE 802.11b / g потужність може бути до 100 мВт з шириною смуги 22 МГц за рівнем ослаблення 30 дБ.

При створенні ZigBee-мережі необхідно вибрати один з 16 каналів у вільному від інших систем частотному діапазоні. Однак в процесі експлуатації може з'явитися джерело внутрішньосмугових перешкод, через що частина мережі вийде з ладу, тому буде потрібно знову «вручну» задати новий активний канал, причому для всієї мережі. Отже, ускладнюється обслуговування системи.

Розглянемо для прикладу таку ситуацію. У великому будинку існує кілька Wi-Fi-мереж, налаштованих на частотні канали так, що весь діапазон 2,4 ~ 2,4835 ГГц ними перекритий (рис 2.1).

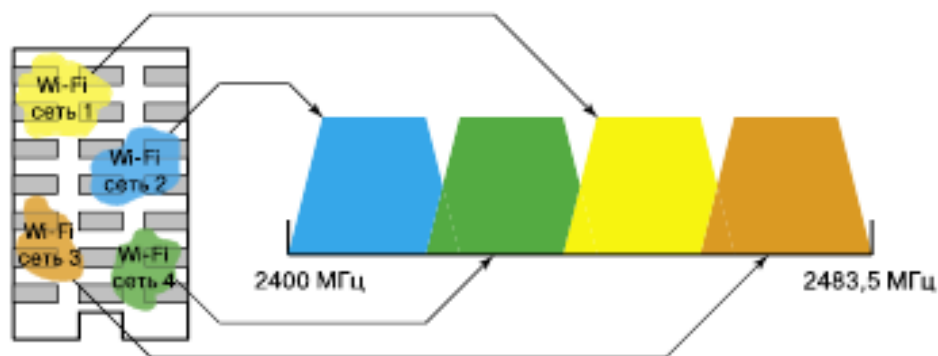


Рисунок 2.1 - Поширення Wi-Fi мереж

Отже, всі 16 каналів стандарту IEEE 802.15.4 будуть потрапляти в область однієї або декількох Wi-Fi-мереж, тому в даних умовах практично неможлива нормальна робота розподіленої по всій будівлі ZigBee-мережі, тому що частина пристроїв не буде функціонувати через вплив розташованій поблизу мереж IEEE 802.11b / g.

Для вирішення подібних проблем в платформі MeshLogic реалізований механізм динамічного виділення каналів: в залежності від локальної перешкодної обстановки вибирається вільний частотний діапазон, при цьому різні ділянки мережі можуть працювати на різних каналах.

Спочатку у всій мережі MeshLogic використовується канал, обраний на етапі інсталяції системи. При досить великій і розподіленій в просторі бездротової мережі, область впливу, що з'явилася в певний момент часу перешкоди на даній частоті, покриє тільки частину вузлів, які після падіння якості зв'язку нижче порогового рівня перейдуть на нову частоту відповідно до алгоритму динамічного виділення каналів.

Відзначимо, що зміну каналу виконують тільки ті вузли, для яких це дійсно необхідно, інша ж частина мережі продовжує функціонувати на колишньому каналі. Звісно, зв'язність всієї мережі зберігається.

Очевидно, даний механізм дозволяє експлуатувати мережу MeshLogic в описаній вище ситуації, тому що різні ділянки мережі будуть використовувати ті чи інші частотні канали, обрані в залежності від сусідньої Wi-Fi-мережі (рис. 2.2). Всі описані дії виконуються автоматично і не вимагають втручання обслуговуючого персоналу.

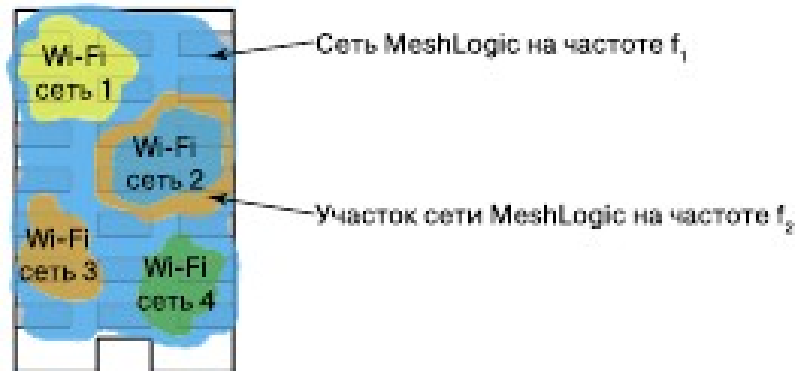


Рисунок 2.2 - Вибір каналів мережі MashLogic

2.1.3 Профілі ZigBee

Структуру стека ZigBee / 802.15.4 складають програмна і апаратна частини. У першій реалізованій код, написаний за специфікацією ZigBee, в другій - механізми MAC і РНУ рівнів, що містяться в стандарті 802.15.4. Код в скомпільованому вигляді, з доданими функціями управління інтерфейсами прийнято називати - профіль.

Як правило, виробники компонентів мережі ZigBee постачають кілька готових профілів. Профіль HomeLightControll управляє режимами освітлення в будинку і передає сигнал вмикання / вимикання від вимикачів до лампочок. Однак для створення мереж зі складною топологією і великою кількістю периферійних пристроїв такі стандартні профілі не завжди є універсальними. Для вирішення подібних завдань розробникам надається можливість створення власних профілів. Ця процедура виконується за допомогою програмного продукту ProfileBuilder, з використанням API постачальника стека ZigBee. Далі, створений профіль завантажується в ZigBee пристрій, який, в залежності від типу профілю, стає координатором, роутером або кінцевим пристроєм. При цьому профіль додатково володіє тим набором функцій, який було визначено користувачем.

Крім широких можливостей по реалізації профілів для побудови бездротової мережі телеметрії, виробники ZigBee пристроїв наділяють свою продукцію досить великою кількістю апаратних інтерфейсів. На сьогоднішній день на ринку доступні ZigBee пристрої, в яких реалізовані такі інтерфейси як UART, SPI, GPIO і ADC / DAC.

Одним із прикладів таких пристроїв є модулі виробництва компанії Cirronet, у яких є такі апаратні інтерфейси 1 SPI, 1 UART, 6 GPIO і 3 АЦП. Ці модулі виконані на базі трансиверів від Chipcon CC2420 і контролера ATmega128 компанії Atmel. Компанія MaxStream випускає аналогічні продукти, на яких доступні 1 UART, 6 GPIO і 6 АЦП. Ці модулі складаються з RF-частини і контролера компанії Freescale, об'єднаних на одній платі розміром 30 * 20 мм. Від більшості подібних продуктів інших виробників ці модулі відрізняє можливість використання AT-команд, завдяки яким суттєво спрощується процедура впровадження модуля в готовий виріб.

Крім модульних платформ деякі виробники представляють інтегральні рішення System-on-Chip (SoC). SoC вдає із себе збірку, в якій rf-частина і контролер об'єднані на одному чіпі.

Профіль стека

Стандарт ZigBee описує профіль стека, який визначає мережу, служби додатків і параметри безпеки для всієї мережі. Профіль стека для конкретної ZigBee мережі вибирається мережевим адміністратором (проектувальником), виходячи з області застосування (автоматизація будівлі, управління техпроцесом, позиціонування). Кожен такий профіль стека потім буде визначати параметри мережі, такі як:

- загальні розміри мережі,
- підтримувані алгоритми маршрутизації,
- розміри таблиць маршрутизації і прив'язки додатків,
- характер служби безпеки для цієї мережі.

Особливий інтерес з точки зору топології мережі являє специфікація трьох параметрів: максимальна глибина мережі (nwkMaxDepth), максимальна кількість дочірніх зв'язків маршрутизатора (nwkMaxChildren), і максимальна кількість зв'язків з дочірніми маршрутизаторами (nwkMaxRouters). Разом вони визначають загальну структуру мережі.

Відповідно до стандарту IEEE 802.15.4 кожен мережевий пристрій має 64-розрядну (довгу) IEEE адресу, якиф унікально ідентифікує цей пристрій. Щоб продовжити життя акумуляторних батарей, для скорочення розмірів пакетів, і, відповідно, часу активної взаємодії пристроїв використовуються більш короткі адреси. Стандарт ZigBee вимагає, щоб обмін повідомленнями після приєднання до мережі виконувався з 16-бітними (короткими) мережевими адресами. 16-бітну мережеву адресу присвоюється пристроям, що знову приєднуються під час обміну повідомленнями в процесі приєднання.

Координатор ZigBee завжди має адресу 0000. Координатор за встановленим стандартом ZigBee алгоритму визначає, виходячи з параметрів профілю стека, діапазони адрес, які присвоюються пристроям, що приєднуються, відповідно до їх розташування в

мережевий ієрархії. Хоча ці діапазони адрес розрізняються, залежно від параметрів профілю стека, діапазони адрес завжди однакові для конкретного набору параметрів профілю стека.

2.1.4 Мережеві протоколи

Стандарти IEEE 802.15.4 і ZigBee 2007 Specification визначають стандартизовані протоколи, які забезпечують мережеву інфраструктуру, необхідну для бездротових мережевих додатків з великим числом датчиків і виконавчих механізмів.

IEEE 802.15.4 визначає фізичний і MAC рівні, а ZigBee визначає мережевий рівень і рівень додатків.

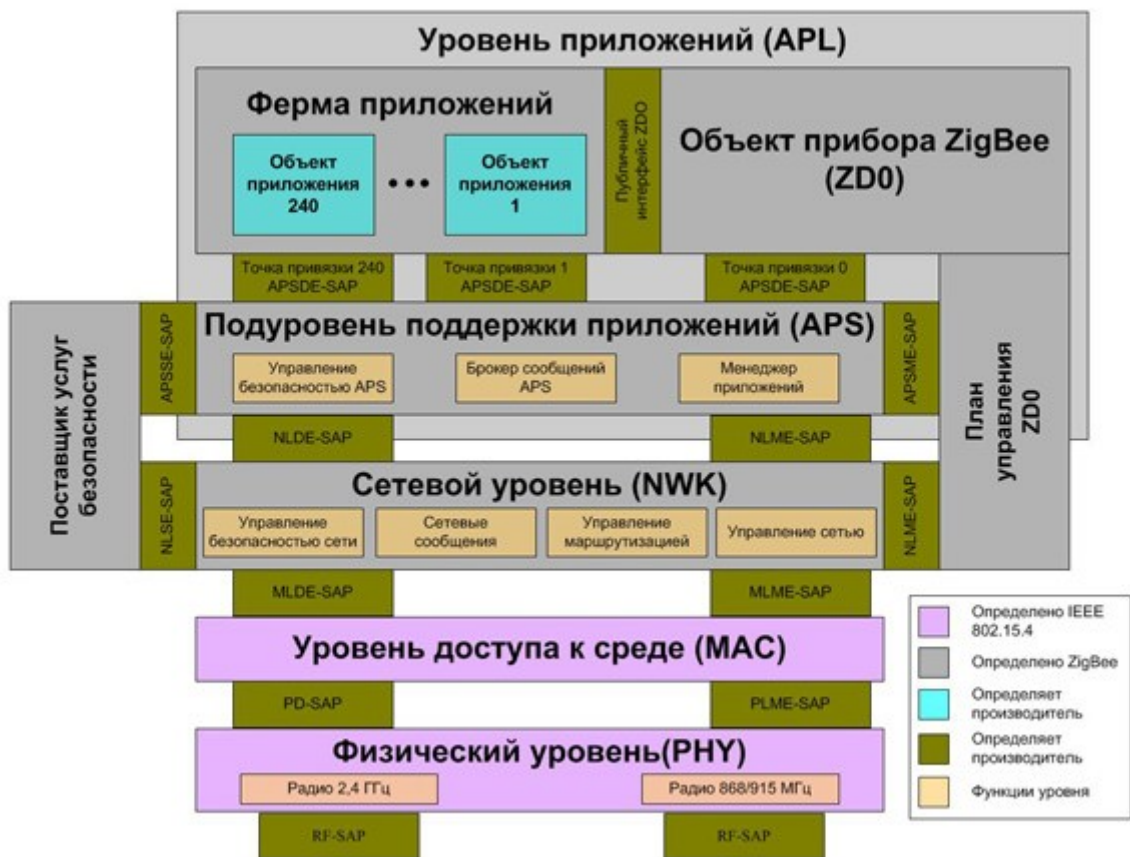


Рисунок 2.3 - Рівні мережевих протоколів ZigBee

Специфікація ZigBee регламентує стек протоколів взаємодії вузлів мережі, в якому протоколи верхніх рівнів використовують сервіси, що надаються протоколами нижче лежачих рівнів.

Рівень додатків (APL) складається з ферми додатків (Application Framework), об'єкта пристрою ZigBee (ZDO) і підрівні підтримки додатків (APS).

Ферма додатків (Application Framework) містить опис порядку створення профілів в стеці ZigBee, що гарантує їх сумісність, а також визначає:

- стандартні типи даних для профілів,
- дескриптори, які допомагають у виявленні служб,
- формати кадрів для транспортування даних і значення пар ключів.

Це дозволяє швидко розробляти прості профілі на основі атрибутів.

Об'єкти додатків (Application Objects) - програмні модулі, що управляють пристроями ZigBee в кінцевих точках. Одна базова станція ZigBee може підтримувати до 240 об'єктів додатків. Об'єкти додатків підтримують кінцеві точки, пронумеровані від 1 до 240. Кінцева точка 0 зарезервована для об'єкта пристрою ZigBee (ZDO).

Об'єкт пристрою ZigBee (ZigBee Device Object – ZDO):

– визначає роль пристрою ZigBee: координатор, маршрутизатор або кінцевий пристрій;

- ініціює запити пошуку і приєднання і / або відповідає на такі запити;
- встановлює безпечний зв'язок між пристроями ZigBee.

Він також надає багатий набір команд управління, які визначаються в профілі пристрою ZigBee (використовуються при введенні в експлуатацію).

План управління ZDO (ZDO Management Plane) підтримує зв'язок ZDO з підрівнем підтримки додатків (APS) та мережевим рівнем (NWK). Дозволяє ZDO обробляти запити додатків на доступ до мережі і забезпечення безпеки, використовуючи повідомлення профілю пристрою ZigBee (ZigBee Device Profile - ZDP).

Підрівень підтримки додатків (Application Support Sublayer - APS). Відповідає за надання даних додатків і профілів пристрою ZigBee. Підрівень також управляє приєднаннями в мережі ZigBee і зберігає дані про такі приєднання в своїй таблиці.

Постачальник послуг безпеки (Security Service Provider - SSP) забезпечує механізми безпеки для рівнів, що використовують шифрування - NWK і APS. Ініціюється і конфігурується об'єктом пристрою ZDO.

Мережевий рівень (Network Layer - NWK) обробляє мережеві адреси і маршрутизацію за викликами MAC рівня. У його завдання також входять:

- запуск мережі (якщо пристрій є координатором);
- призначення мережевих адрес;
- додавання і видалення мережевих пристроїв;
- маршрутизація повідомлень;
- застосування політики безпеки;
- здійснення пошуку маршрутів.

2.1.5 Рівні, що визначаються стандартом IEEE 802.15.4

Рівень управління доступом до середовища (Medium Access Control Layer - MAC) відповідає за надійний зв'язок пристрою з безпосередніми сусідами, допомагає вирішувати колізії і підвищувати ефективність. MAC рівень відповідає також за асемблювання і декомпозицію кадрів і пакетів даних.

Фізичний рівень (Physical Layer - PHY) забезпечує інтерфейс до фізичного середовища передачі - радіо. Фізичний рівень складається з двох рівнів, які працюють в різних діапазонах частот. Рівень, який працює при менших частотах, покриває діапазони 868 МГц для Європи і 915 МГц для таких країн, як США і Австралія. Високочастотний фізичний рівень працює на частоті 2,4 ГГц і застосовується практично у всьому світі.

Зв'язок між елементами стека протоколів ZigBee здійснюється через точки доступу до послуг (service access point - SAP):

- APSDE-SAP – до даних підрівня підтримки додатків;
- APSME-SAP – до управління подуровнем підтримки додатків;
- APSSE-SAP – до управління безпекою підрівня підтримки додатків;
- NLDE-SAP – до даних мережевого рівня;
- NLME-SAP – до управління мережевим рівнем;
- NLSE-SAP – до управління безпекою мережевого рівня;
- MLDE-SAP – до даних підрівня MAC;
- MLME-SAP – до управління подуровнем MAC;
- PD-SAP – до даних фізичного рівня PHY;
- PLME-SAP – до управління фізичним рівнем PHY;
- RF-SAP – до трансмітеру.

2.1.6 Типи мереж

Однією з важливих особливостей мереж ZigBee є можливість вибору топології мережі, що в найбільшій мірі відповідає конкретному застосуванню. Точна структура мережі ZigBee в більшості випадків не може бути заздалегідь визначена, так як залежить від взаємного розташування пристроїв і проходження радіохвиль під час формування мережі. Однак три згаданих параметра профілю стека будуть диктувати загальну структуру (топологію) мережі.

Наприклад, якщо програма має пристрої, фізично розташовані в лінію (наприклад, уздовж конвеєра), установка параметрів `nwkMaxChildren = 1` і `nwkMaxRouters = 1` створить лінійну структуру, як показано на рисунку 2.4.

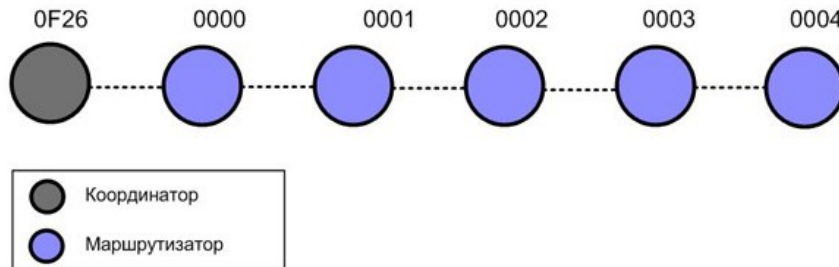


Рисунок 2.4 - Лінійна топологія мережі

Гнучкість мережевих структур ZigBee, що налаштовуються відповідно до галузі застосування, найкращим чином проявляється в ситуаціях, коли більшість інших мереж виявляються непридатні.

Найбільш перспективною в мережі ZigBee є Mesh-мережі (mesh-топологія).

Осередкова мережа - це мережа взаємопов'язаних маршрутизаторів і кінцевих пристроїв, в якій кожен маршрутизатор має, принаймні, два зв'язки і може транслювати повідомлення своїх сусідів. Як показано на рисунку 2.5, осередкова мережа складається з одного координатора і декількох маршрутизаторів і кінцевих пристроїв.

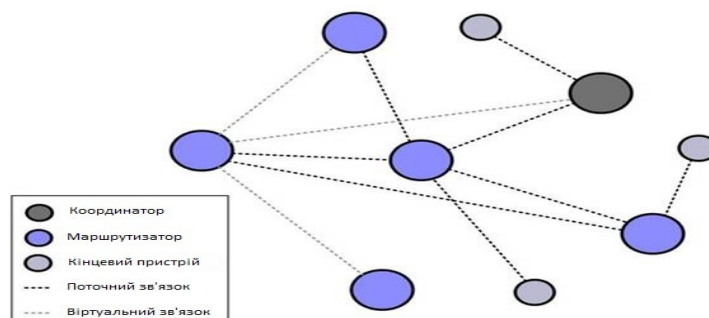


Рисунок 2.5 – Mesh-мережа ZigBee

У такій мережі кожен пристрій може зв'язуватися з будь-яким іншим пристроєм як безпосередньо, так і через проміжні вузли мережі. Mesh-мережі підтримують «багатокроковий» (multi-hop) зв'язок, при якому дані проходять кроками від одного пристрою до іншого, використовуючи найбільш надійні лінії зв'язку і найбільш ефективні маршрути, поки не досягнуть мети.

Здатність до багатокрокової передачі допомагає забезпечити живучість мережі (здатність до самовідновлення). Якщо один з пристроїв відмовляє або виявляється під впливом перешкод, мережа здатна перемаршрутизуватися, використовуючи пристрої, що залишилися.

Принципова відмінність мереж ZigBee від інших бездротових мереж, таких як IEEE 802.11 / WiFi, в тому, що замість централізованої зіркоподібної структури мережі ZigBee допускають ієрархію приєднання до мережі. В результаті може бути досягнуто кілька рівнів приєднання. Така ієрархія пристроїв служить для створення істинно бездротових мереж.

Для порівняння, IEEE 802.11 / WiFi пропонує технологію точки бездротового доступу: бездротове підключення проводиться між, скажімо, ноутбуком або телевізором і точкою доступу, яка не є бездротовим маршрутизатором. Тому далі потік даних направляється на дроти (наприклад, по провідній мережі до інших провідних пристроїв або до іншої точки бездротового доступу). Мережа ZigBee здатна доставляти пакети повністю по бездротовій мережі через кілька проміжних бездротових пристроїв.

Далі представлені переваги осередкової мережі:

1. Mesh-мережі мають високу живучість і надійність. Якщо який-небудь маршрутизатор стає недоступним, можуть бути знайдені і використані альтернативні маршрути.
2. Використання проміжних пристроїв при передачі даних робить mesh-мережу мережу широко масштабованою.
3. Слабкі сигнали і мертві зони можуть бути легко виключені простим додаванням додаткових маршрутів.

2.1.7 Формування мережі ZigBee

Будь-яка мережа ZigBee починається з її формування. Пристрій, здатний бути координатором (він призначається при проектуванні мережі), визначає вільний від перешкод канал і налаштовує себе в якості координатора. У ZigBee цей пристрій називається PAN координатором - координатором персональної (на відміну від локальної - LAN) мережі.

Пристрої, які хотіли б приєднатися до мережі (підключитися, увійти до складу мережі), розсилають циркулярний запит. Спочатку на цей запит відповідає лише PAN координатор. Надалі крім PAN координатора надати приєднання до мережі можуть маршрутизатори ZigBee, які вже приєдналися до PAN координатору безпосередньо або

через інші маршрутизатори. Після того, як пристрій, який бажає приєднатися до мережі, отримує відповідь на циркулярний запит, відбувається обмін повідомленнями, які визначають, чи може пристрій приєднатися до мережі. Ключовим фактором визначення можливості є здатність маршрутизатора обслужити нові пристрої в доповнення до раніше підключених.

Існує два шляхи вступу пристроїв в мережу ZigBee: MAC асоціація і повторне мережеве приєднання (NWK rejoin).

Як очевидно з назви, MAC асоціація здійснюється на MAC рівні. Здатністю до MAC асоціацію має за замовчуванням володіти кожен пристрій ZigBee. Механізм MAC асоціації представлений далі.

1. Координатор або маршрутизатор, який збирається дозволити іншим пристроям приєднатися до нього, повинен виставити на свій MAC рівень дозвіл на приєднання NLME-PERMIT-JOINING.request.

2. У свою чергу, пристрій, який бажає вступити в мережу, виставляє на свій MAC рівень запит на приєднання NLME-NETWORK-DISCOVERY.request.

3. Після цього вступаючий в мережу пристрій передає в ефір циркулярний запит маячка і отримує цей маячок від маршрутизатора(ів), готового(их) підключити новий пристрій.

4. Отримавши маячок (маячки), пристрій визначає, в яку мережу він бажає вступити і до якого конкретно пристрою приєднатися, після чого виставляє на свій MAC рівень вимогу NLME-JOIN.request з встановленим в значення FALSE прапорцем «повторне приєднання» (rejoin).

5. Зробивши це, пристрій що вступає направляє на обраний маршрутизатор запит на приєднання і отримує відповідь, що містить присвоєний йому мережеву адресу.

Відзначимо, що MAC асоціація не є безпечною, так як всі повідомлення передаються у відкритому (не зашифровані) вигляді.

Повторне мережеве приєднання (NWK rejoin) всупереч назві може використовуватися і при першому підключенні пристрою до мережі. І, як зрозуміло з назви, здійснюється на мережевому рівні. Це означає, по-перше, що стандартний механізм MAC асоціації в цьому випадку може або застосовуватися (в цьому випадку ZigBee маршрутизатор виставить дозвіл на приєднання NLME-PERMIT-JOINING.request), або ні. По-друге, це означає, що обмін пакетами може бути безпечним, якщо пристрій, що вступає знає поточний мережевий ключ. Це можливо, якщо пристрій дійсно перепідключатися, або якщо він включається вперше, але отримав мережевий ключ через якийсь сторонній механізм (наприклад, під час налаштування).

При такому підключенні пристрій, який бажає вступити в мережу, виставляє на свій мережевий рівень (NWK) запит на приєднання NLME-JOIN.request і обмінюється з маршрутизатором, що його підключає, пакетами «запит підключення» - «відповідь на запит підключення».

Крім процесу підключення нових пристроїв до зміни структури мережі можуть привести інші процеси. Пристрої можуть, наприклад, залишати мережу або повторно підключатися до мережі в інших місцях (наприклад, при перезавантаженні пристрою).

Рисунок 2.6 показує приклад перепідключення. У цьому прикладі пристрій з короткою адресою «0E3B» перепідключатися до мережі, як «097D» і згодом як «0260». На кожному етапі цей пристрій приєднується до іншого маршрутизатора, який виділяє адресу з наявного в його розпорядженні діапазону адрес.

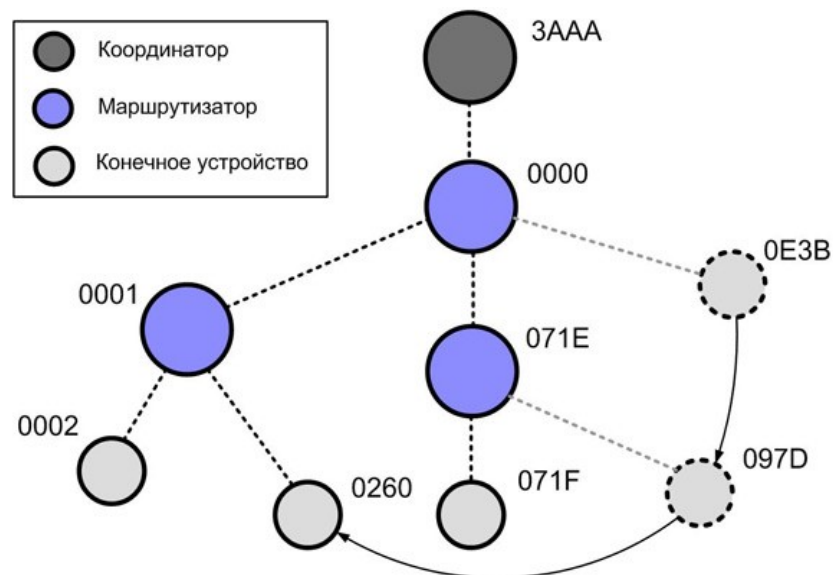


Рисунок 2.6 - Перепідключення кінцевого пристрою в деревовидній мережі

Одне з важливих переваг мережі ZigBee - здатність відстежувати пристрої і топологію мережі в режимі частих включень, відключень і перепідключення пристроїв.

2.1.8 Маршрутизація в мережах ZigBee

Основний алгоритм маршрутизації в мережах ZigBee - «Ad hoc On Demand Distance Vector» (AODV) заснований на понятті «вектор відстані» маршруту, коли кожен маршрутизатор, який бере участь в трансляції запиту маршруту від конкретного джерела до певного пункту призначення створює свій запис в маршрутній таблиці. Цей запис як

мінімум містить «логічну відстань» від джерела запиту і адресу попереднього маршрутизатора.

Алгоритм працює наступним чином:

– коли пристрою «джерело» (I) необхідно відправити пакет на пристрій «адресат» (A), джерело ініціює процес виявлення маршруту шляхом розсилки ширококомовного пакета «запит маршруту», запитуючи маршрут до пристрою A (рис.2.7 (а));

– кожен пристрій, що отримав цей запит маршруту, вносить свій запис в таблицю маршрутизації і може транслювати цей запит, включивши в нього таблицю зі своїм записом (рис. 2.7 (б));

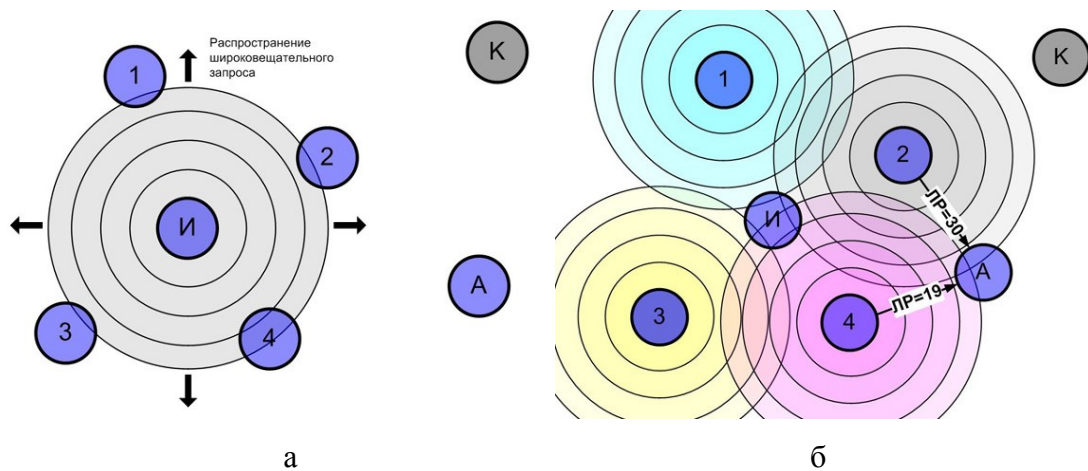


Рисунок 2.7. - Пошук маршруту: а - Крок 1; б - Крок 2

– такі пристрої, які отримали ретрансльований запит, транслюють його далі, і з кожним кроком «логічна відстань», пройдена запитом, збільшується на величину, що залежить від якості зв'язку на черговій ділянці;

– процес триває поки запит не досягає пристрою A, причому зазвичай пристрою A досягає кілька ретрансльованих запитів (від різних маршрутизаторів) і «логічні відстані» (ЛВ) маршрутів цих пакетів різні;

– пристрій A відправляє відповідь по маршруту, що має мінімальну «логічну відстань»;

– відповідь буде повертатися у відповідність з таблицею маршруту по шляху, пройденому попередніми запитом, поки не прибуде на пристрій I (рис. 2.8).

Ця серія переданих в зворотному напрямку відповідей формує прямий маршрут для майбутньої передачі пакетів від I до A (рис. 2.9). Механізм «логічної відстані» дозволяє джерелу і вузлам, розташованим на шляху запиту, вибрати мінімальну «логічну відстань» маршруту від джерела до місця призначення.

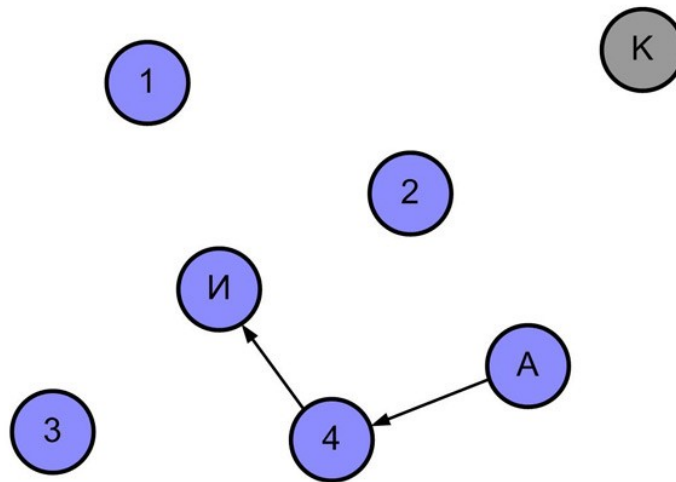


Рисунок 2.8 - Пошук маршруту (Крок 3)

Описаний базовий алгоритм ефективний і універсальний, однак вимагає значного обсягу пам'яті для зберігання маршрутів, що ускладнює і здорожує пристрої.

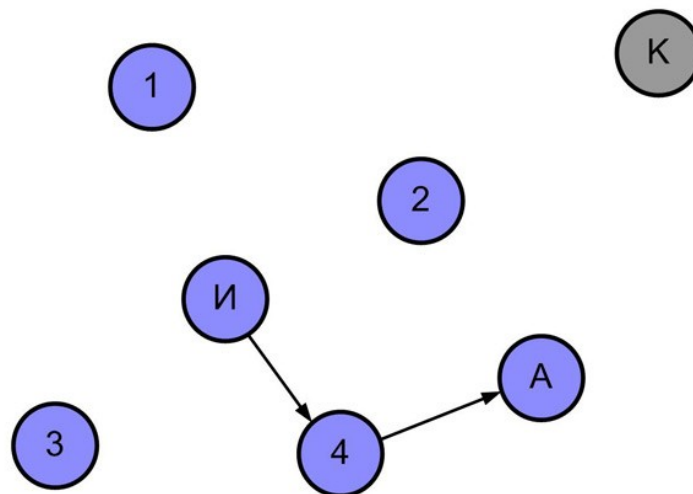


Рисунок 2.9 - Пошук маршруту (Крок 4)

Тому в мережах ZigBee реалізований і інший алгоритм, який дозволяє знизити вимоги до обсягу пам'яті, а іноді і зменшити мережевий трафік, необхідний для пошуку маршрутів. Дія алгоритму заснована на тому, що адреси в ZigBee мережі розподіляються ієрархічно, починаючи з координатора. Пристрій, що не має можливостей маршрутизації або пристрій, можливості маршрутизації якого вичерпані, може скористатися ієрархічною маршрутизацією - менш ефективною, але, тим не менш, цілком практичною.

У процесі формування ZigBee мережі алгоритм розподілу адрес задає діапазони адрес мережевих пристроїв в ієрархічному порядку, починаючи з координатора. В результаті будь-який пристрій в мережі, знаючи свою адресу і адресу одержувача пакета,

може визначити, чи належить конкретна мережева адреса до «низхідної» гілки (і до якої саме), або знаходиться в іншому місці в ієрархії пристроїв. Виходячи з цього, будь-який пристрій може прийняти просте рішення маршрутизації: передавати пакет «вгору» - в напрямку координатора або «вниз» - до дочірнього пристрою.

Приклад ієрархічної маршрутизації представлений на рисунку 2.10.

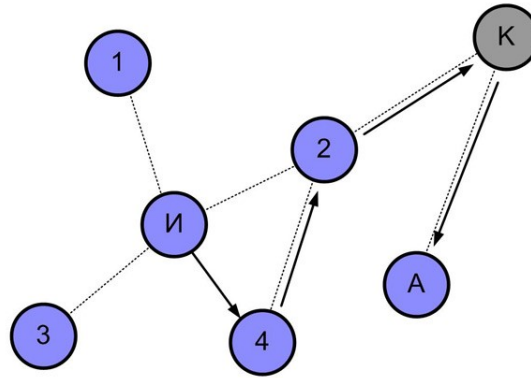


Рисунок 2.10 - Ієрархічна маршрутизація

Як у випадку на рисунку 2.8, пакет, відправлений пристроєм І, призначений для пристрою А. Однак пристрій 4 вичерпав свої можливості маршрутизації, тому він не може транслювати пакет безпосередньо на пристрій А, а замість цього, використовуючи ієрархічну маршрутизацію, направляє цей пакет «вгору» по ієрархічній драбині - на пристрій 2. Далі пакет транслюється на координатор К, який передає його на шукану адресу А.

Переваги ієрархічної маршрутизації полягають в її простоті і меншому використанню ресурсів, що дозволяє створювати дуже недорогі пристрої без можливостей маршрутизації, які, тим не менш, можуть брати участь в будь-якій ZigBee сумісній мережі. Недолік алгоритму в тому, що пакети будуть рухатися вгору і вниз до координатора і назад навіть в тому випадку, коли між джерелом і приймачем можливий прямий зв'язок.

У багатьох додатках, що використовують бездротові мережі, є спеціальний електронний пристрій, часто званий агрегатором, до якого інші мережеві пристрої повинні регулярно направляти дані. З метою економії мережевого трафіку в ZigBee PRO передбачений спеціальний випадок пошуку маршрутів, при якому єдиний циркулярний запит маршруту від агрегатора дозволяє встановити в роутінгових таблицях всіх маршрутизаторів мережі записи з агрегатором як одержувач.

2.1.9 Масштабування мереж ZigBee в мережах

Типова інфраструктура системи складається з декількох зон ZigBee мереж, пов'язаних з сервером через сегменти локальної IP мережі. Кожна з зон є самостійною ZigBee мережею, що складається, як правило, не менше ніж з чотирьох анкерів, щодо яких позиціонуються мітки, що знаходяться в зоні ZigBee мережі. Один з анкерів в кожній зоні завжди є ще й координатором мережі. Два анкера в кожній ZigBee зоні системи забезпечені IP інтерфейсами і виконують функції шлюзів, що зв'язують зону з сервером через незалежні IP канали, що забезпечує резервування і відповідно надійність інфраструктури в цілому. Приклад інфраструктури, яка обслуговує багатоповерховий будинок і прилеглу територію, показаний на рисунку 2.11.

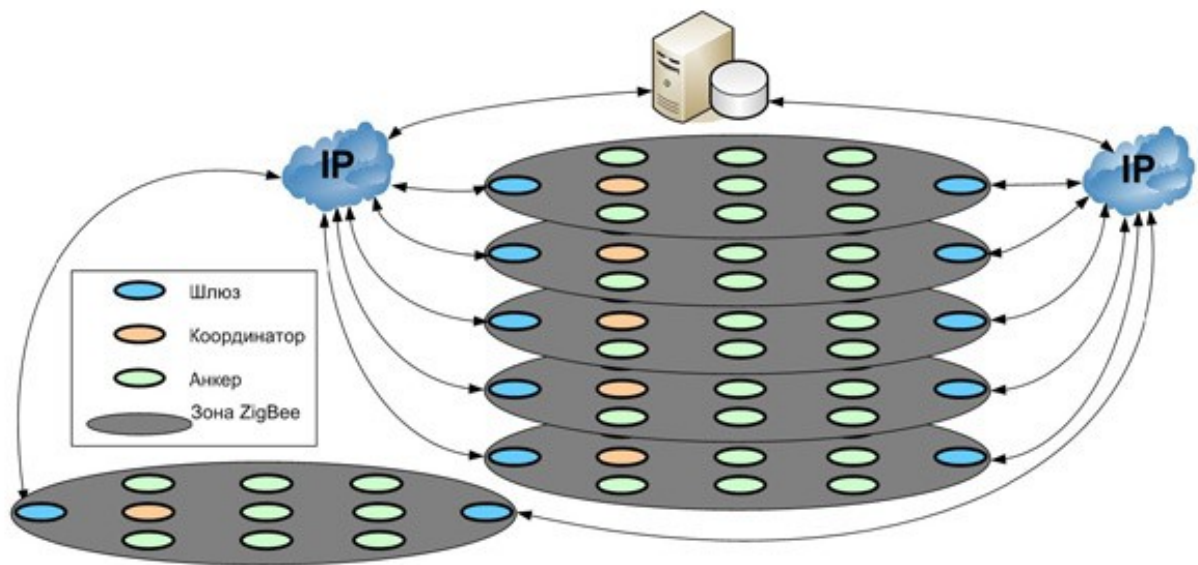


Рисунок 2.11 - Інфраструктура системи позиціонування у реальному часі

Профіль додатку описує набір пристроїв, що використовуються для певної програми, і побічно схеми обміну повідомленнями між цими пристроями. Наприклад, є профілі додатків для «розумного будинку», автоматизації будівлі, системи безпеки і так далі. Для унікальної ідентифікації додатків кожному з них виділяється ідентифікатор профілю.

Існує два типи профілів додатків:

- загальні профілі додатків - сумісне прикладне програмне забезпечення, розроблене альянсом ZigBee, яке виконує конкретне завдання;
- профілі виробників обладнання - приватні профілі додатків, розроблені компаніями для роботи пристроїв ZigBee.

Пристрої в межах профілю додатку спілкуються один з одним за допомогою кластерів, які можуть бути входами або виходами пристрою. Наприклад, в профілі «розумного будинку» є кластер, призначений для підсистем контролю освітлення. Ідентифікатор кластера унікально ідентифікує кластер в рамках конкретного профілю.

Кінцева точка визначає об'єкт зв'язку всередині пристрою, через який реалізується певний додаток. Наприклад, пульт дистанційного керування може виділити кінцеву точку 6 для контролю освітлення в спальні, кінцеву точку 8 для управління системою обігріву та кондиціонування повітря і кінцеву точку 12 для управління системою безпеки. Це дозволяє пульту дистанційного керування незалежно зв'язуватися з цими пристроями і визначати, які пакети призначені для кожної програми і пристрою.

У будь-якому пристрої ZigBee можна використовувати до 240 кінцевих точок (1 - 240). Кінцева точка нуль виділена для об'єкта пристрою ZigBee (ZDO), який забезпечує команди контролю і управління самим пристроєм ZigBee.

2.1.10 Бібліотека кластерів ZigBee (ZCL)

ZCL - це бібліотека кластерів, які можуть використовуватися будь-яким додатком. Це дозволяє використовувати кластери в різних сферах. Зокрема, кластери освітлення можуть використовуватися в будь-якому додатку, який потребує управління освітленням, наприклад, в «розумному будинку» або в системі автоматизації комерційної будівлі.

Кожен кластер має два «кінці» – клієнт і сервер (рис. 2.12).

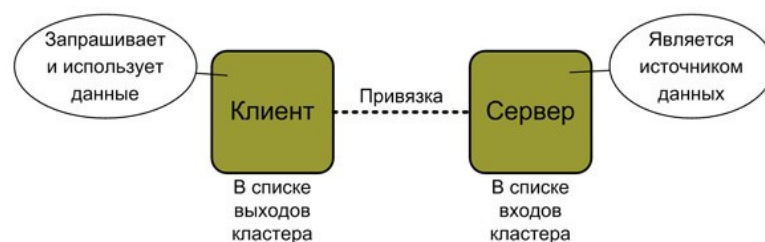


Рисунок 2.12 - Зв'язка «клієнт-сервер»

Кластери в бібліотеці кластерів організовані в ряд функціональних доменів, включаючи:

- освітлення,
- клімат–контроль,
- виміри,
- охорона і безпека,

- протоколи,
- загального призначення,
- і так далі.

Схематичне зображення кластерів і їх використання в профілях додатків можна побачити на рисунку 2.13.



Рисунок 2.13 - Бібліотеки кластерів і профілі додатків ZigBee

Прив'язки є сполуками між двома кінцевими точками, причому кожна прив'язка підтримує конкретний профіль додатку, а кожен тип повідомлення представлений кластером в цьому профілі.

Прив'язки можуть бути створені між окремими кінцевими точками або їх групами, такими, як світильники і вимикачі, які мають кластери введення і виведення, що збігаються (тобто, один і той же ідентифікатор кластера). Пристрій ZigBee може мати до 240 кінцевих точок, тому кожен пристрій може підтримувати безліч прив'язок.

Прив'язки можуть зберігатися всередині вихідного пристрою. Наприклад, в пульті дистанційного керування можуть зберігатися адреси і ідентифікатори кінцевих точок усіх додатків, з якими він взаємодіє. Це називається прямою прив'язкою або прив'язкою джерела.

Або відомості про прив'язку можуть зберігатися в проміжному пристрої, що підтримує таблицю підстановки, яка співвідносить всі кінцеві точки джерел і одержувачів. Це називається непрямою прив'язкою.

2.1.11 Безпека ZigBee

Система безпеки ZigBee, яка заснована на 128-бітному AES алгоритм, доповнює модель безпеки, що надається IEEE 802.15.4. Служби безпеки ZigBee включають методи створення ключів і транспорту, управління пристроями і захисту пакетів даних.

Специфікація ZigBee визначає безпеку для MAC, NWK і APS рівнів. Безпека додатків зазвичай забезпечується через профілі додатків.

Центр управління безпекою визначає, чи слід дозволити або заборонити приєднання нових пристроїв до своєї мережі.

Центр управління безпекою може періодично оновлювати ключ мережі і переходити на новий ключ. Спочатку він транслює новий ключ, зашифрований за допомогою старого ключа мережі. Потім він повідомляє всім пристроям про перехід на новий ключ.

Центром управління безпекою зазвичай є координатор мережі, але це може бути і виділений пристрій.

Центр управління відіграє наступні ролі в забезпеченні безпеки:

- довірений менеджер по перевірці автентичності пристроїв, що бажають приєднатися до мережі;
- менеджер мережі, що підтримує і поширює мережеві ключі;
- диспетчер конфігурації, що забезпечує безпеку взаємодії пристроїв.

ZigBee використовує три типи ключів для управління безпекою: головний ключ, мережевий ключ і ключ каналу зв'язку.

1. Головні ключі.

Ці додаткові ключі не використовуються для шифрування пакетів. Вони використовуються як початковий, що розділяється двома пристроями, секретний код, коли пристрої виконують процедуру генерації ключа каналу зв'язку.

Головні ключі, які створюються центром управління безпекою, називаються основними джерелами центру безпеки, в той час як всі інші ключі називаються основними ключами рівня додатків.

2. Мережеві ключі.

Ці ключі забезпечують безпеку мережевого рівня в мережі ZigBee. Ключ мережі мають всі пристрої в мережі ZigBee.

Мережеві ключі високої безпеки повинні відправлятися по бездротових каналах тільки в зашифрованому вигляді, в той час як стандартні мережеві ключі безпеки можуть відправлятися як зашифрованими, так і не зашифрованими.

3. Ключі каналів зв'язку.

Ці додаткові ключі забезпечують безпеку одноадресної передачі повідомлень між двома пристроями на рівні додатків.

У режимі стандартної безпеки перелік пристроїв, головних ключів, ключів каналів зв'язку і мережевих ключів можна зберігати як в центрі управління безпекою, так і в самих пристроях. Центр управління безпекою, проте, відповідає за підтримання стандартного мережевого ключа і контролює політику прийому в мережу. В цьому режимі вимоги до ресурсів пам'яті центру управління безпекою набагато нижче, ніж для режиму підвищеної безпеки.

У режимі підвищеної безпеки центр управління безпекою веде список пристроїв, головних ключів, ключів каналів зв'язку і мережевих ключів, необхідних для контролю і застосування політики поновлення мережевих ключів і доступу в мережу. В цьому режимі по мірі зростання кількості пристроїв в мережі швидко зростає необхідний центру управління безпекою обсяг пам'яті.

Додаткові можливості безпеки, які забезпечує ZigBee PRO, надзвичайно важливі, тому що мережі ZigBee використовуються в відповідальних додатках.

2.1.12 Введення в експлуатацію

Під введенням в експлуатацію розуміється фізичне розгортання конкретного рішення і логічна прив'язка вузлів для формування функціональної мережі. У широкому сенсі цього слова введення в експлуатацію охоплює широке коло завдань, включаючи:

- розмір радіо і фізичного середовища,
- розміщення пристроїв,
- конфігурацію їх параметрів,
- прив'язку додатків,
- оптимізацію мережі та параметрів пристроїв,
- тестування і підтвердження правильності роботи.

Крім того, при введенні в експлуатацію найчастіше доводиться вирішувати суміжні питання, такі як наявність досвідченого персоналу, наявність необхідного обладнання,

функціональна сумісність використовуваних пристроїв, а також їх сумісність з іншими бездротовими або дротовими системами.

В ідеалі пристрої повинні запускатися самі. Наладчик тільки подає на них напругу, включає і спостерігає, як пристрої працюють поза мережею, до якої вони повинні приєднатися, як вони безпечно включаються в цю мережу, до якого пристрою (або пристроїв) вони приєднуються і з якими пристроями спілкуються.

Але щоб досягти цього ідеалу, наладчик спершу повинен виконати всю свою роботу.

А якщо подивитися на кількість завдань, пов'язаних з введенням в експлуатацію, і врахувати, що багато наладчиків, швидше за все, мають обмежене уявлення про застосовувану технологію, легко зрозуміти цінність налагоджувальних інструментів.

Інструменти наладчика зазвичай працюють на ноутбучі або КПК і розроблені, щоб спростити налагодження і введення в експлуатацію. Ці інструменти мають інтуїтивно зрозумілий інтерфейс, що приховує складність базової технології. А головне, забезпечують гнучку візуалізацію мережі і пристроїв, а також надають кошти для налаштування, запуску та управління системою.

Налагоджувальні інструменти, як правило, не призначені для роботи в складі мережі при поточній експлуатації, а просто, не будучи частиною основного додатку, полегшують введення в експлуатацію та періодичне обслуговування або управління.

На рисунку 2.14 показаний приклад введення ZigBee пристрою в експлуатацію.

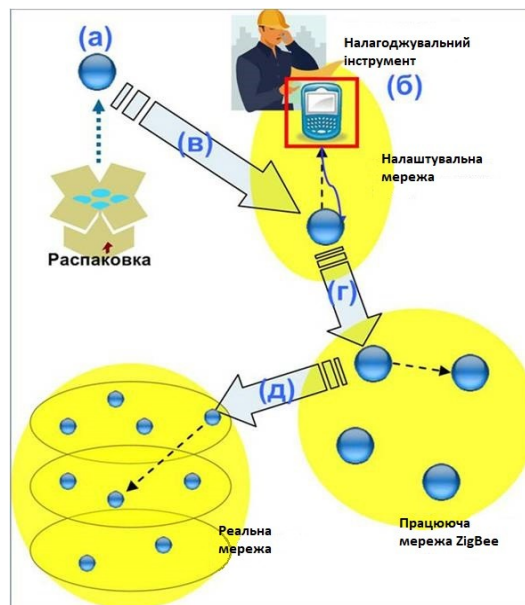


Рисунок 2.14 - Введення в експлуатацію прилади ZigBee

На даному рисунку зображено:

1. Новий розпакований пристрій при включенні готовий до вступу в будь-яку мережу.
2. Наладчик включає налагоджувальний інструмент, який може утворювати з пристроєм настроювальну мережу і за допомогою якого пристрій може бути налаштований.
3. Новий пристрій вступає в настроювальну мережу з налагоджувальним інструментом.
4. Наладчик, використовуючи інструмент, налаштовує пристрій таким чином, щоб він вступив в працюючу мережу ZigBee, використовуючи правильні атрибути безпеки і правильні прив'язки. Пристрій приєднується до працюючої мережі ZigBee.
5. Перевіривши роботу пристрою в працюючій мережі, наладчик переналаштовує пристрій для роботи в реальному мережі, куди цей пристрій і вступає.
(Пунктирними стрілками показані приєднання пристрою).

2.1.13 Енергоспоживання

Одна з ключових особливостей БСМ полягає в енергетичній ефективності. У більшості випадків мінімальне енергоспоживання є більш пріоритетним критерієм, ніж пропускна здатність, латентність, рівномірність надання доступу до середовища і т.п. У цьому полягає основна специфіка мережевих протоколів для БСМ.

У ZigBee найменше енергоспоживання досягається при синхронізованому (beacon mode) доступі до середовища, який дозволяє переводити в «сплячий» режим як кінцеві пристрої, так і маршрутизатори.

Однак на практиці досить важко налаштувати розклад доступу до середовища при складній топології мережі або при нерівномірній швидкості генерації трафіку. Більш простим є множинний доступ до середовища за алгоритмом CSMA / CA, але при цьому координатори повинні постійно перебувати в режимі прослуховування каналу, тому маршрутизаторам потрібна стаціонарна мережа живлення.

Отже, можна очікувати, що в багатьох практичних завданнях тільки кінцеві пристрої будуть працювати на автономному джерелі живлення (батарея), а маршрутизатори і PAN-координатор - від мережі живлення.

У платформі MeshLogic всі вузли здатні працювати від автономного джерела живлення, при цьому не потрібна будь-яка настройка режимів доступу до середовища. Розроблений MAC-протокол динамічно вибирає оптимальний з точки зору енергоспоживання режим роботи кожного вузла в залежності від його поточного

мережевого навантаження. Система самостійно адаптується під мінливі вимоги прикладного рівня, що необхідно, наприклад, в мережах зі змінним періодом формування пакетів даних [8].

2.1.14 Вимоги до апаратних ресурсів

Прагнення мінімізувати енергоспоживання і габарити пристроїв накладає обмеження на апаратне забезпечення бездротового вузла.

Як правило, в подібних пристроях застосовуються 8- або 16-розрядні низькоспоживаючі мікроконтролери, які не відрізняються обчислювальними можливостями і великими обсягами пам'яті. Отже, мережеві протоколи повинні пред'являти мінімальні вимоги до апаратних ресурсів вузла.

Вважається, що реалізація ZigBee-стека може займати від 4 до 32 Кбайт флеш-пам'яті залежно від функціонального типу пристрою.

Очевидно, що розмір стека впливає на величину доступних прикладному рівню ресурсів мікроконтролера, тому багато сучасних ZigBee-платформ припускають двопроцесорну реалізацію вузла, в якій один мікроконтролер займається виконанням мережевих функцій, а другий - рішенням прикладних завдань. В результаті збільшуються енергоспоживання, вартість і габарити.

У платформі MeshLogic вимоги до апаратних ресурсів всіх вузлів однакові і для поточної реалізації становлять близько 12 Кбайт пам'яті програм і 2 Кбайт ОЗУ.

Отже, в більшості випадків буде достатньо одного мікроконтролера в складі вузла [9].

2.2 Про розвиток модулів ZigBee

ZigBee-модулі MaxStream – нові можливості.

Технології IEEE802.15.4 / ZigBee застосовуються для вирішення широкого кола завдань. Ці модулі ідеально підходять і для швидкої організації зв'язку типу «точка-точка». Модулі XBee швидко завоювали популярність серед відчизняних розробників, перш за все з двох причин - простота застосування і низька ціна. Про простоту застосування говорить той факт, що для проведення тесту дальності зв'язку достатньо 10 хвилин, 5 з яких піде на відкриття упаковки відладочного комплекту. Модулі не вимагають ніякої попередньої настройки для організації каналу зв'язку типу «точка-точка», досить підключити до модуля 3 лінії: земля, живлення, UART. Легкість

застосування модулів поєднується з їх великими можливостями, які доступні розробнику через набір простих AT-команд.

За останній час компанія MaxStream кардинально оновила внутрішнє програмне забезпечення модулів XBee / XBeePro. Нові версії прошивки (Firmware) можна завантажити в модуль за допомогою безкоштовної програми X-CTU. Завдяки новому Firmware стало можливим отримання в автоматичному режимі аналогових або цифрових даних з встановленої користувачем частотою опитування. Стан цифрових портів може зчитуватися циклічно, або за фактом зміни рівня. Між циклами опитування радіоприймальна частина модуля може переключатися в режим сну зі споживанням струму менш 35 мкА, що робить модуль ідеальним відповідним рішенням для пристроїв з батарейним харчуванням. Зниженню споживаної потужності також сприяє режим послідовного накопичення даних з подальшою передачею їх в єдиному радіопакеті. Введення 128-бітного AES-шифрування робить бездротовий зв'язок більш надійним та захищеним, що особливо важливо для додатків, що працюють з цінними даними. Додавання нових функцій дозволяє розробнику отримати додатковий вииграш від застосування недорогих і простих у застосуванні модулів і модемів XBee / XBeePro.

2.2.1 Історія змін програмного забезпечення модулів

Перша доступна версія має базовий набір AT-команд. Версія 1.0.6.1 була випущена 08.09.2005. Далі, до можливостей базової версії додані нові функції, вийшла версія 1.0.8.3, яка була випущена 03.05.2006.



Рисунок 2.15 – Формат фрейма API

Управління модулями за допомогою API (Application Programmers Interface - програмний інтерфейс програми). Додатково до прозорого режиму передачі даних доданий режим передачі даних у вигляді фреймів певної структури (рис. 2.15). Це дозволило передавати дані і керуючі команди, не переводячи модуль в спеціальний

командний режим. Завдяки новому режиму з'явилася можливість отримувати значення RSSI (рівень сигналу) і підтвердження про успішну доставку для кожного пакета.

Додані нові AT-команди, що розширюють мережеві (802.15.4) можливості модуля: ATNI, ATND, ATDN, ATCE, ATSC, ATSD, ATA1, ATA2, ATAI, ATDA, ATFP, ATAS, ATED, ATCA, ATMM, ATDP, ATPR, ATEC, ATEA, & ATEH.

Зміни параметрів представлені нижче.

1. Вилучений канал 0x18 в модулі XB24 для відповідності вимогам FCC. Таким чином, модуль XB24 має 16 каналів 0x0B-0x1A і модуль XB24 має 12 каналів 0x0C-0x17.

2. Змінено метод запису кількох AT-команд в одному рядку.

Наступна версія 1.0.A.2. була випущена 20.11.2006. До можливостей версії 1.0.8.3 додані нові функції:

- підтримка роботи з АЦП і функції відображення портів введення-виведення;
- шифрування повідомлень AES-128 (ATEE і ATKY);
- підтримка додаткових (понад IEEE802.15.4) повторів відправки повідомлень (ATTR);
- підтримка повторних відправлень ширококомовних повідомлень (Broadcast Retries);
- підтримка повторних спроб доступу до каналу передачі (CCA-оцінка зайнятості каналу);
- додана команда ATNT, що встановлює час відповіді при подачі команди ATND (Node Discovery - виявлення пристроїв);
- швидкість передачі даних до 115200.

Дана версія ПЗ (1.0.A.2) дозволяє організувати мережі 802.15.4 (зірка), налаштовуючи модуль в режим «координатор» або «кінцевий пристрій» за допомогою команди ATCE. У режимі координатора модуль буде самостійно формувати мережу і підтримувати ретрансляцію повідомлень, виконуючи такі дії:

- перевірку доступності каналів і вибір каналу з найменшим рівнем перешкод;
- запит присутності інших мереж PAN 802.15.4;
- вибір ID і формування мережі PAN;
- підключення кінцевих пристроїв (Association);
- пряма або відкладена ретрансляція повідомлень.

У режимі кінцевого пристрою модуль може виконувати наступні дії:

- здійснювати активне сканування на різних каналах для виявлення координатора(ів);
- підключатися до доступного координатору;
- передавати повідомлення до інших вузлів мережі;

– підтримувати режим циклічного сну з автоматичним відсиленням запитів до координатора (Poll Request) для отримання відкладених повідомлень.

Зрозуміло, розробник може обмежити «самостійність» модуля і жорстко задати основні параметри - на якому частотному каналі потрібно працювати, до якого координатору підключатися і т.д.

2.2.2 Режим відображення портів вводу-виводу

Модуль XBee має 5 каналів АЦП і 8 цифрових портів введення-виведення, які користувач може задіяти в своїх додатках за допомогою АТ-команд. Крім того, модуль XBee підтримує автоматичну передачу даних про стан сигналу на входних лініях на інший модуль. На віддаленому модулі стан відповідних портів буде відображати рівні сигналу на віддаленому модулі. Рівень сигналів АЦП передавального модуля відображається за допомогою двох доступних каналів ШІМ на приймальному модулі. Додатково до відображення сигналу на своїх портах, приймальний модуль опціонально може видавати дані про стан портів віддаленого модуля через UART у вигляді фреймів певної структури. Працюючи в режимі відображення портів, модуль зберігає можливість приймати і передавати повідомлення через UART звичайним чином, в т.ч. і в «прозорому» режимі. Максимальна частота, з якою можуть передаватися дані АЦП, становить 1 кГц. При використанні декількох каналів АЦП частота вибірок повинна пропорційно знижуватися. Режим відображення портів дозволяє організувати передачу інформації про стан цифрових і аналогових датчиків без застосування зовнішнього мікроконтролера і написання спеціальної програми. Досить один раз вказати використовувані порти, частоту опитування, початкові значення для цифрових виходів, кількість зразків в одному пакеті і т.п. Далі модулі будуть автоматично передавати інформацію без додаткового зовнішнього втручання.

2.2.3 Внутрішня структура модулів і використання ПЗ від Freescale

Компанія MaxStream зробила крок назустріч розробникам і опублікувала інформацію по внутрішній структурі модулів XBee / XBeePro. Модулі побудовані на базі ZigBee-трансивера Freescale MC13193 і мікроконтролера MC9S08GT60.

У викладеній на сайті компанії документації XBee_Code_Development докладно описана внутрішня структура модулів і програмне забезпечення третіх виробників, яке може бути завантажено в ці модулі:

- Freescale SMAC;
- Freescale 802.15.4 PHY and MAC;
- Figure 8 ZigBee stack.

Грунтуючись на викладених бібліотеках і прикладах компіляції, розробник може використовувати процесор модуля для виконання власного додатка. При цьому розробник отримує доступ до всіх портів мікроконтролера, вбудованого в модуль. У викладеному на сайті компанії архіві також наводяться принципові схеми перехідних плат, які входять в стартові і налагоджувальні комплекти. Там же наводиться детальна інструкція по програмуванню модулів через вбудований завантажувач за допомогою UART через програму X-CTU або за допомогою програматора-відладчика «P & E Microsystems, Inc. USB HCS08 / HCS12 Multilink ». Для розробки програм можна використовувати Metrowerks CodeWarrior Development Studio для мікроконтролерів серії HC (S) 08 [10].

2.2.4 XBee і Xbee-PRO S2C – нові ZigBee модулі від DIGI

На зміну модулів Digi XBee / XBee-PRO ZB (S2 / S2B) на чіпі Silicon Labs EM250 випущені модулі XBee / XBee-PRO ZB (S2C) на чіпі Silicon Labs EM357 з більшою пам'яттю.

Нові модулі, також як і модулі попередньої серії, доступні в двох варіантах монтажу (TH, SMT), а також OTA сумісні з модулями серії S2 / S2B. Модулі S2C мають менший струм споживання, більшу дальність, що дозволяє створювати більш масштабні мережі, доданий інтерфейс SPI.

Основні параметри модулів XBee / XBee-PRO ZB (S2C) представлені в таблиці 2.1.

Ключовим нововведенням є, то, що модулі XBee / XBee-PRO ZB (S2 / S2B) включають шість типів пристроїв: координатор, координатор API, роутер AT, роутер API, кінцевий вузол AT і кінцевий вузол API, що встановлюються в єдиний образ прошивки, а вибір типу пристрою здійснюється програмно.

Таблиця 2.1 - Параметри модулів XBee/XBee-PRO ZB (S2C)

Параметр	XBee S2C ZigBee		XBee-PRO S2C ZigBee
Система на кристалі	Silicon Labs	EM357 SoC	Silicon Labs EM357

(SoC), Процесор	MC9SO8QE	MC9SO8QE	SoC
Частотний діапазон	2.4 ГГц		
Дальність зв'язку всередині приміщень	60 м	90 м	
Дальність зв'язку на відкритих просторах	1200 м	3200 м	
потужність передачі	3.1 мВт (+5 дБм) / 6.3 мВт (+8 дБм) boost mode (режим максимальної потужності)	63 мВт (+18 дБм)	
Чутливість приймача (1%)	-100 дБм / -102 дБм boost mode	-101 дБм	
Антени	TH: PCB Antenna, U.FL Connector, RPSMA Connector, Wire SMT: RF Pad, PCB Antenna, U.FL Connector	Антени	
Живлення	2.1 – 3.6 В	Живлення	
Струм споживання (передача)	45 мА (+8 дБм, boost mode), 33 мА (+5 дБм, normal mode)	Струм споживання (передача)	
Струм споживання (прийом)	31 мА (boost mode) 28 мА (normal mode)		

Нижче описано, що нового в модулях XBee/XBee-PRO ZB (S2C):

1. Альтернативний послідовний порт, доступний при використанні SPI інтерфейсу в режимі slave (підлеглий пристрій).
2. Образи шести окремих прошивок (Coordinator AT, Coordinator API, Router AT, Router API, End Device AT, and End Device API) об'єднані в одну прошивку.
3. Команда ATCE включає-вимикає режим координатора (1 - включений / 0 - вимкнено).
4. Команда ATSM включає-вимикає режим кінцевого вузла, що підтримує сон (0 - роутер або координатор, що не нуль - сплячий кінцевий вузол).

Примітка: режим координатора не може бути включений, якщо параметр команди ATSM не дорівнює нулю, і назад: не можна встановити ATSM не рівним нулю, якщо

ATCE не дорівнює нулю (тобто не можна включити режим координатора, якщо модуль вже включений в режим кінцевого пристрою і навпаки).

5. Фрагментація тепер доступна і в режимі API, і в прозорому режимі.
6. Порти P3 (DOUT), P4 (DIN), D8 (SleepRq), and D9 (On-Sleep) тепер можуть використовуватися як порти введення-виведення GPIO.
7. Обидва типи підтягують резисторів: підтяжка до землі або підтяжка до живлення, тепер можуть бути використані для висновків, зконфігурованих як вхід.
8. Додана команда ATVL, що показує детальну інформацію про версії прошивки.
9. Додана команда ATDO для конфігурації опцій пристрою.
10. Додана команда ATAS для запуску активного сканування.
11. При передачі на власну адресу статусні повідомлення Tx Status повертають код статусу 0x23.
12. В команду ATDO додані опції HIGH_RAM_CONCENTRATOR і NO_ACK_IO_SAMPLING.
13. Підтримуються зв'язування і багатоадресні передачі.
14. Додана команда AT & X для очищення таблиць зв'язування і групи.
15. В поле Transmission Options фрейму 0x11 додані опції 0x04 (непряма адресація) і 0x08 (многоадресна адресація).
16. Стан розриву (в інтерфейсі UART) тривалістю 5 секунд призведе до скидання XБee; після цього він завантажиться в командний режим зі швидкістю обміну за замовчуванням.
17. Діапазон BD (швидкість передачі) збільшено з 0-7 до 0-0x0A; нестандартні швидкості передачі дозволені, але не гарантуються.
18. Строкові параметри NI, DN, ND допустимо вводити в верхньому і нижньому регістрі.
19. Опція TxOption 0x01 відключає повторні спроби передачі і відновлення маршруту; опція RxOption 0x01 показує, що передавач відключив повторні спроби передачі.
20. FR повертає код статусу модему 0x00 замість 0x01.
21. DC10 – опція докладного звіту в режимі приєднання.
22. Адресовані собі фрагментовані повідомлення тепер повертають код статусу: self-addressed Tx Status code (0x23), замість просто статусу успішно success (0x00).
23. Три символи плюс ("+++"), надруковані поспіль, переводять модуль з режиму API в режим AT команд.
24. Команда ATCN перемикає модуль з режиму AT команд в режим API [11].

2.2.5 ZigBee XBee ZNet 2.5 (Series 2)

Модулі ZigBee XBee ZNet 2.5 (Series 2) є закінченими вузлами, здатними самостійно підключатися до мережі з MESH-топологією і передавати дані, що надходять від зовнішнього хост-процесора. Приклад модуля зображений на рисунку 2.16.

Завдяки вбудованому ZigBee-стеку всі операції по формуванню мережі, приєднання нових пристроїв, прокладання оптимальних маршрутів повідомлень здійснюються автоматично, без участі зовнішнього мікроконтролера. Для передачі даних існує обмежений набір простих команд, що управляють, які можуть виконуватися навіть на самому недорогому 8-розрядному зовнішньому мікроконтролері. Модулі XBee ZNet 2.5 можуть виступати і як абсолютно самостійні вузли. У цьому випадку джерелом або приймачем інформації виступають наявні на модулі периферійні вузли - порти введення-виведення, АЦП і ЦАП. При використанні модуля без зовнішнього мікроконтролера прийом або передача даних буде відбуватися під управлінням команд, що надходять по ефіру від координатора або будь-якого іншого вузла мережі.



Рисунок 2.16 - Модулі XBee ZNet 2.5 (Series 2)

Робота в API-режимі
 Подача команди ATND в режимі API і відповідь від кінцевого пристрою
 Модулі XBee можуть працювати в режимі управління за допомогою AT-команд або в API-режимі. Для управління за допомогою зовнішнього мікроконтролера режим API набагато більш зручний, оскільки дозволяє передавати і дані, і команди управління в загальному потоці. Крім того, деякі можливості в режимі AT-команд просто недоступні. Наприклад, послати по ZigBee-мережі AT-команду віддаленого модулю можна тільки в API-режимі. Робота з API-пакетами вимагає обчислення контрольних сум, що не дуже зручно при ручному формуванні пакета в вікні програми X-CTU (рис. 2.17, 2.18), однак не

представляє великої складності при управлінні XBee-модулем за допомогою зовнішнього мікроконтролера.

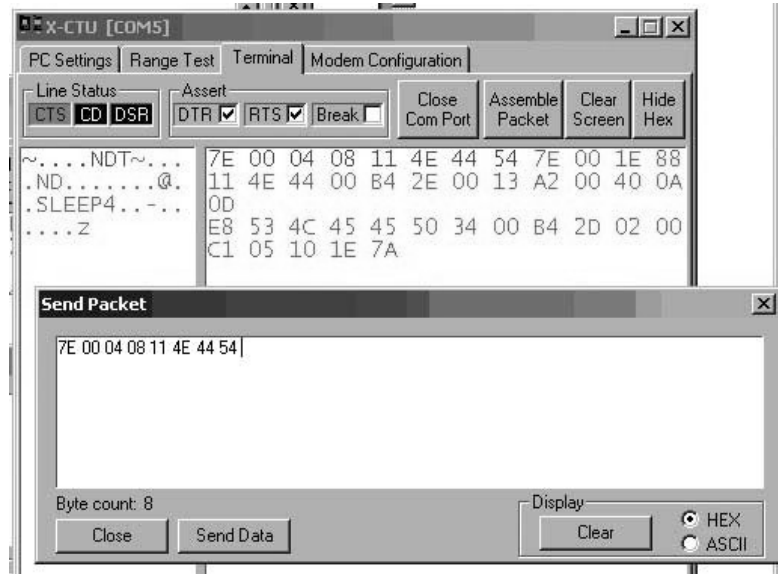


Рисунок 2.17 - Подача команди ATIS в режимі API і відповідь від кінцевого пристрою

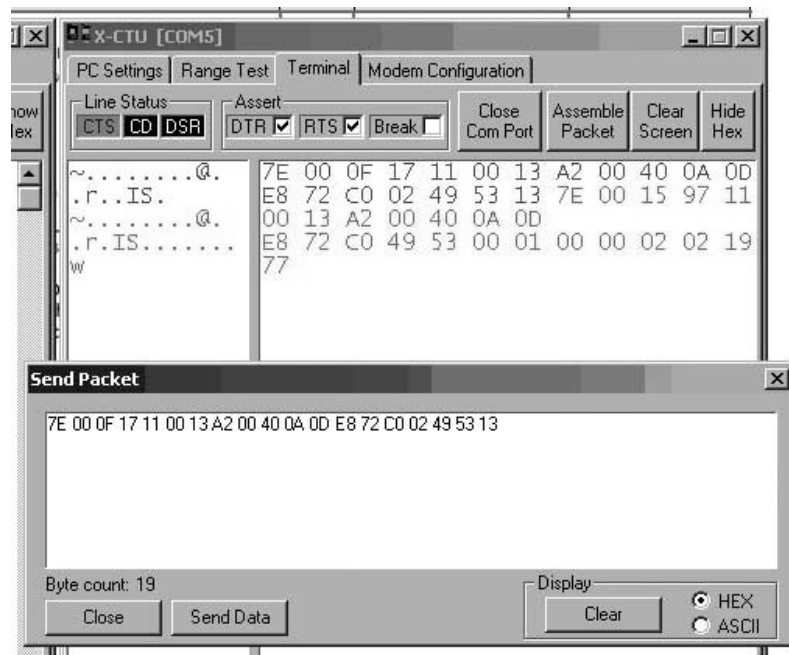


Рисунок 2.18 - Приклади роботи в API-режимі в програмі X-CTU

Для практичних експериментів з API-фреймами виявилось більш зручним використовувати програму-термінал «Terminal v1.9b», яка дозволяє заздалегідь скласти набір API-фреймів у вигляді макросів і відсилати їх натисненням однієї кнопки.

Наприклад, щоб виявити всі вузли мережі, потрібно скласти фрейм AT-команди «ND» і відіслати його з будь-якого модуля мережі. Нижче наведені API-фрейми, отримані на виході UART модуля XBee ZNet 2.5, з якого був відправлений запит на виявлення всіх

вузлів мережі. Всього було отримано 4 відповіді, Як видно з наведених відповідей, в мережі було виявлено 2 роутера і 2 кінцевих вузла (рис. 2.19).

Данные	Пояснения
7E	– стартовый разделитель (признак начала API фрейма)
00 19	– длина API-фрейма
8E	– ID-номер API-пакета (ответ на AT-команду)
11	– номер фрейма, на который получен ответ
4E 44	– ASCII-имя AT-команды, на которую пришел ответ (здесь: «ND»)
0	– байт статуса (00 – ОК)
E1 90	– 16 битный сетевой адрес MY
00 13 A2 00	– 64-битный уникальный адрес SH
40 0A 0C 22	– 64-битный уникальный адрес SL
38 00	– ASCII строковый идентификатор узла NI (здесь: «3», строка заканчивается 00)
FE FE	– адрес «родительского» узла. Действительно только для конечных устройств
01	– тип устройства (0-координатор, 1-роутер, 2-конечное устройство)
00	– статус
C1 05	– профиль устройства (profile ID)
10 1E	– производитель (manufacturer ID)
0C	– контрольная сумма

Рисунок 2.19 - API-фреймы, отримані на виході UART модуля XBee ZNet 2.5

Далі наведені додаткові приклади API-фреймів, отримані від різних вузлів в ZigBee-мережі в різних реальних ситуаціях (рис. 2.20).

Действие	API-фрейм
Отправка AT-команды «ND»	7E 00 0408 11 4e 44 54
Отправка AT-команды «IS» как фрейм запроса удаленной команды	7E 00 0F 17 11 00 13 A2 00 40 0A 0F 47 25 73 02 49 53 4C
Ответ узла на запрос IS идущий от координатора	7E 00 1597 E9 00 13 A2 00 40 0A 0D 4F 44 C5 49 53 00 01 00 00 02 03 3A 3F
API-фреймы, получаемые на выходе координатора при аппаратном сбросе	7E 00 028A 06 6F – фрейм статуса. Состояние «Перезагрузка» 7E 00 028A 06 6F – фрейм статуса. Состояние «Координатор начал работу»
На выходе координатора при подключении 1 роутера	7E 00 2095 00 13 A2 00 40 0A 0B 4C 61 59 0161 59 00 13 A2 00 40 0A 0B 4C 33 00FF FE 01 02 C1 05 10 1E 22
На выходе координатора при подключении конечного устройства	7E 00 2495 00 13 A2 00 40 0A 0D 4F 00 02 0100 02 00 13 A2 00 40 0A 0D 4F 4E 4F 44 45 32 0000 00 02 02 C1 05 10 1E 5F
На выходе координатора при нажатии кнопки DIO1 на конечном устройстве	7E 00 24 95 00 13 A2 00 40 0A 0D 4F 00 02 01 00 02 00 13 A2 00 40 0A 0D 4F 4E 4F 44 45 32 0000 00 02 01 C1 05 10 1E 60
Прием данных в API-режиме. Каждый пакет содержит один ASCII-код символа цифры от 0 до 5	7E 00 0D90 00 13 A2 00 40 0A 0E 18 76 C2 01 30 E1 7E 00 0D90 00 13 A2 00 40 0A 0E 18 76 C2 01 31 E0 7E 00 0D90 00 13 A2 00 40 0A 0E 18 76 C2 01 32 DF 7E 00 0D90 00 13 A2 00 40 0A 0E 18 76 C2 01 33 DE 7E 00 0D90 00 13 A2 00 40 0A 0E 18 76 C2 01 34 DD 7E 00 0D90 00 13 A2 00 40 0A 0E 18 76 C2 01 35 DC

Рисунок 2.20 – Приклади API-фреймів

Різні блоки фрейма виділені кольором і типом шрифту. Наведені приклади можуть бути корисними при вивченні розділу, присвяченому розділу по роботі з API-фреймами в оригінальному описі виробника.

Схема вимірювального модуля на базі Xbee дуже проста (рис. 2.21).

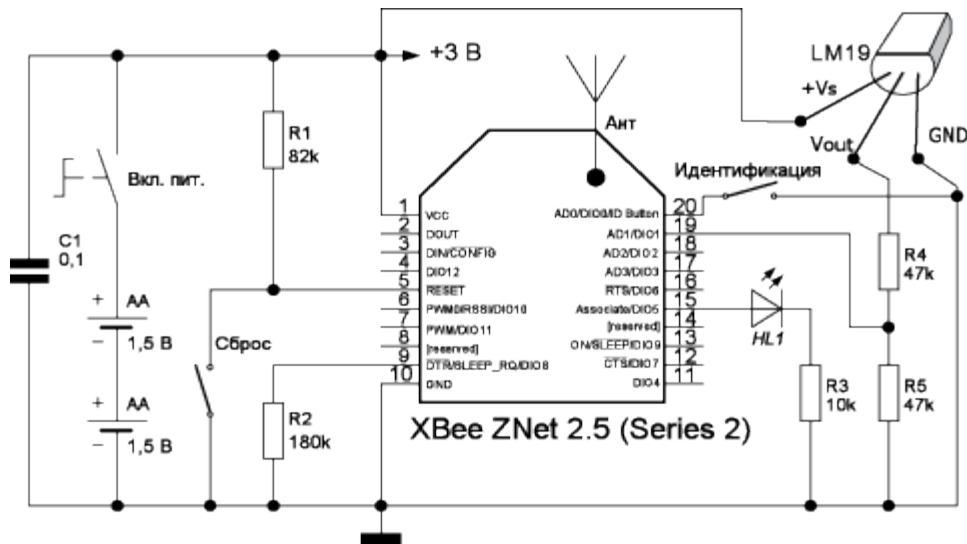


Рисунок 2.21 - Принципова схема температурного датчика

Вимірювання температури проводиться за допомогою аналогового датчика LM19, що перетворює температуру в діапазоні від -55 до 130 °C в вихідну напругу, що вимірюється за допомогою АЦП модуля Xbee ZNet 2.5 (Series 2). У зв'язку з тим, що діапазон вихідної напруги LM19 ($0,303 \dots 2,485$ В) перевищує максимальне вимірюється напруга АЦП модуля ($1,2$ В), в схемі застосований дільник напруги на резисторах, понижуючий вихідну напругу LM19 в 2 рази. Струм споживання LM19 становить менше 10 мкА, тому датчик дозволяє істотно економити енергію батарей при роботі ZigBee-модуля в сплячому режимі. Вимірювальний модуль працює від 2 батарей типу "AA". Світлодіод «HL1» відображає режим роботи модуля: якщо він світиться постійно - модуль не долучився до ZigBee-мережі, якщо блимає 2 рази в секунду - відбулося приєднання модуля до ZigBee-мережі, рідкісні короткі спалахи - модуль знаходиться в сплячому режимі і періодично прокидається для запиту даних від батьківського вузла. У цьому простому проекті застосування вбудованого АЦП і можливість віддаленого відсилання команд дозволили відмовитися від застосування зовнішнього мікроконтролера [12].

3 ПОБУДОВА КОМБІНОВАНОЇ МЕРЕЖІ ПРОМИСЛОВОГО ПІДПРИЄМСТВА

Будівля, в якій планується впровадити БСМ, має розмір 150м * 100м, висота стель 10м. Модулі XBee S2C ZigBee мають радіус зв'язку 60м, XBee-PRO S2C ZigBee - 90м. Так як перший споживає меншу кількість струму в роботі, і його можливостей вистачає для наших цілей -Будемо використовувати саме цю версію, а не PRO, адже наша мета оптимізувати енерговитрати.

Будинок розділений стіною 0,5м на 2 частини - 90м * 100м і 60м * 100м. Приміщення 1 - виробничий цех, і приміщення 2 - оцінний цех (рис. 3.1). Обслуговуватися вони будуть різними координаторами. Живлення маршрутизаторів і координатора постійне, від мережі. Всі датчики будуть мати автономне живлення.

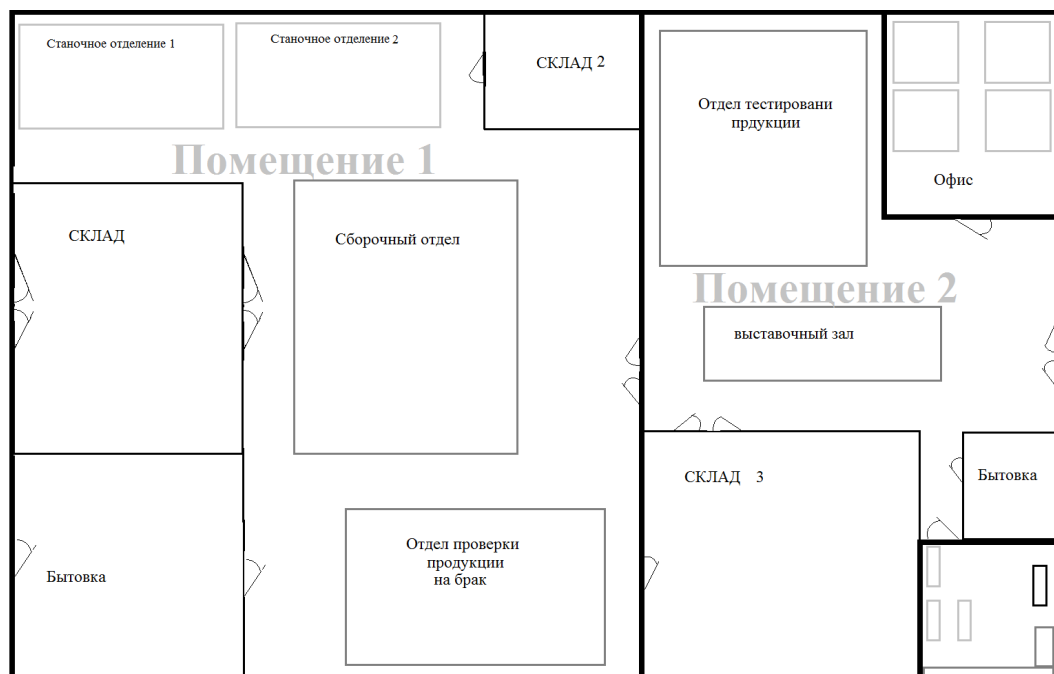


Рисунок 3.1 - Схема будівлі підприємства

Шлюз в мережі грає роль сполучної ланки між ZigBee підмережею і сервером. Приклад готового пристрою зображений на рисунку 3.2. Сервер через шлюз подає по ZigBee підмережі команди на розташований в підмережі координатор для отримання даних з датчиків. Звіти про заміри, в свою чергу, передаються на сервер через шлюз. Підмережа ZigBee і її шлюзи можуть обслуговувати одну або кілька зон.

Шлюзи в мережі дозволяють оптимізувати навантаження мережі. Якщо при проектуванні або в процесі експлуатації виявляється, що пропускної здатності підмережі не вистачає для обслуговування міток, що знаходяться в ній, досить розділити підмережу і

територію, що обслуговується нею, на декілька менших за розміром шляхом додавання необхідного числа шлюзів і призначення ZigBee координаторів додаткових підмереж.

Крім того, використання шлюзів забезпечує практично необмежене масштабування мережі і розширення території, що обслуговується за рахунок створення і підключення нових ZigBee підмереж.



Рисунок 3.2 - Шлюз ZigBee - Ethernet + CSS
інтерфейс компанії RTLS

Основу системи складає інфраструктура, що складається з анкерів (маршрутизаторів) - базових станцій бездротового зв'язку. Приклад зображений на рисунку 3.3.

Особливість інфраструктури системи полягає в тому, що кожен маршрутизатор виконує, щонайменше, дві функції: забезпечує вимірювання відстаней до міток і є вузлом бездротової мережі.

Для цього кожен маршрутизатор має два радіо інтерфейсу: CSS - для вимірювання відстаней і управління мітками і ZigBee - для створення пористої бездротової мережі.



Рисунок 3.3 - Маршрутизатор компанії RTLS

3.1 Вибір датчиків

Всі двері оснащуються двірним/віконним ZigBee датчиком відкриття - магнітоконтат, який контролює відкриття/закриття дверей (рис. 3.4).



Рисунок 3.4 – Двірний/віконний ZigBee датчик відкриття

Так само на всі двері спрямовані інфрачервоний (ІЧ) ZigBee датчик руху поєднаний з температурним датчиком (рис. 3.5). Вимірювання температури від -10 до $+50$ ° С. Кут виявлення 130 °. Радіус виявлення до 6 м.



Рисунок 3.5 - ІЧ ZigBee датчик руху

У приміщенні будуть так само розподілені ZigBee датчики CO₂ - детектор якості повітря (рис. 3.6) (вимірює температуру, вологість, CO₂ і наявність шкідливих летючих органічних суспензій) і ZigBee датчики освітлення (вимір освітленості в діапазоні: $0 \sim 300$ KLUX (рис. 3.7)).



Рисунок 3.6 - ZigBee датчик CO₂



Рисунок 3.7 – ZigBee датчик освітлення

Двері отладоного цеху обладнані розумним дверним ZigBee замком (рис. 3.8). Без традиційного механічного ключа. Відкриває/закриває двері дистанційно набором коду зі

смартфона або натисканням кнопки відкриття брелка. Підібрати код практично неможливо.



Рисунок 3.8 – Розумний двірний ZigBee замок

Для пожежної безпеки використовується високонадійний пожежний ZigBee датчик диму суміщений з температурним датчиком від -10 до $+50$ °C (рис. 3.9). Виявлення диму двома датчиками: іонний і фотоелектричний. Вбудована сирена 85 db. світлодіодні індикатори.



Рисунок 3.9 – Пожежний ZigBee датчик диму

3.2 Розподіл датчиків в приміщенні

Приміщення 1 має 3 кімнати (відділи), які відділені стінами завтовшки до 10 см і заввишки 5 м. Так само є 3 відділи, які розділені тонкими фальшстінами. У кожному відділі встановлюються датчики освітлення, якості повітря і диму. Їх кількість залежить від типу відділу. Наприклад «СКЛАД» це приміщення, в якому люди перебувають за необхідності, коли потрібно що-небудь розвантажити/вивантажити/перенести. Тому там необхідні свідчення загального стану приміщення. А в відділах «Станочное відділення», «Складальний відділ» і «Відділ перевірки продукції на брак» постійно знаходяться люди, тому там необхідні більш точні свідчення, тому біля кожного робочого місця розташовуються датчики освітленості. Датчики якості повітря, що визначають температуру, вологість, CO₂ і наявність шкідливих летючих органічних суспензій розподіляються в декількох точках відділу, враховується середнє значення показників датчиків.

Відділи зі стінами так само мають датчик руху і датчики відкриття/закриття дверей. Схема приміщення 1, з розподілом датчиків зображена на рисунку 3.10.

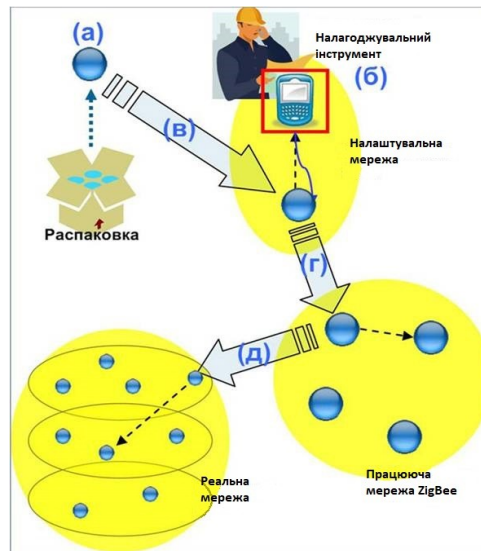


Рисунок 3.10 – Схема розташування датчиків приміщення 1

Приміщення 2 має 6 відділів, 4 з яких виділені в кімнати зі стінами завтовшки до 10 см і заввишки 5 м (офіс, склад, побутівка і щитову поєднану з серверною). Як і в Приміщенні 1, тут є все ті ж датчики, логіка розподілу та ж. У «Офісі» у кожній групі робочих місць є датчик освітленості. Офіс дворівневий, кімната на кожному поверсі має однакове розташування робочих місць і датчиків, доступ до відділу обмежений з використанням розумного замку. На вікна встановлені датчики відкриття \ закриття вікна, вони аналогічні датчикам дверей. ZigBee датчики CO₂ - вимірює температуру, вологість, вміст CO₂ і наявність шкідливих летючих органічних суспензій в кімнаті офісу.

У цьому приміщенні всі двері обладнані розумним ZigBee замком, дверним ZigBee датчиком відкриття і ІЧ ZigBee датчиком руху. Встановлено датчики якості повітря і освітленості для контролю температури, вологості, наявності шкідливих парів в повітрі, і рівня освітленості приміщення.

З офісу за допомогою ПК можна стежити за показаннями всіх датчиків, що дозволить аналізувати стан повітря підприємства.

Вхід в кімнати закритий дверима з використанням розумного двірного замка, доступ обмежений.

Щитова і серверна кімната відповідають за розподіл електроенергії на підприємстві і містить обчислювальні потужності для роботи підприємства та БСМ. Кімната обладнана датчиками присутності, датчиком відкриття \ закриття дверей, пожежними ZigBee датчиками диму, так само, для дотримання необхідних умов праці, встановлені датчик освітленості і датчик якості повітря. Схема приміщення 2 з розподілом датчиків зображена на рисунку 3.11.

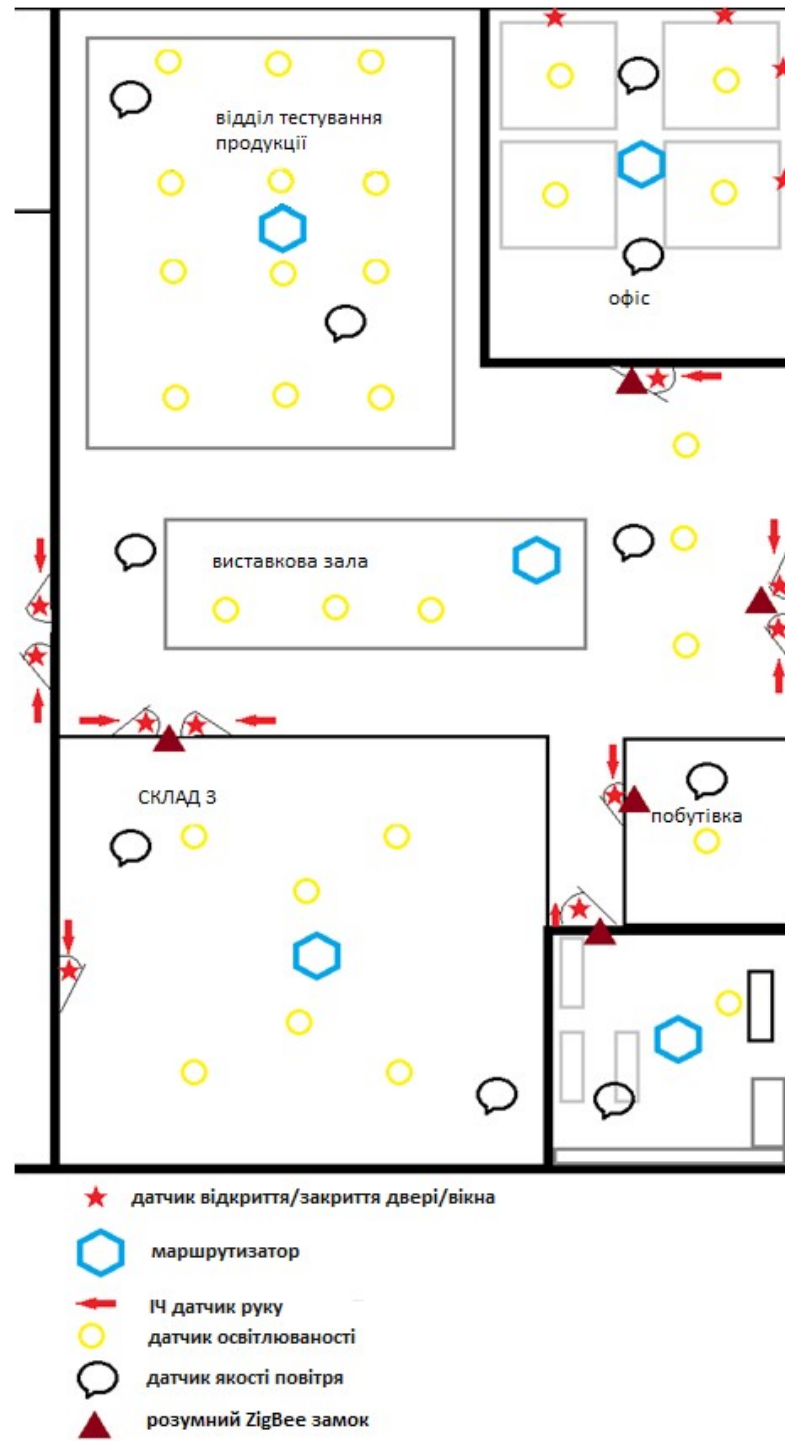


Рисунок 3.11 – Схема розташування датчиків приміщення 2

4 ДІАГРАМА ВОРОНОГО

4.1 Історія виникнення

Перше використання цієї діаграми зустрічається у праці Рене Декарта (1596-1650) «Початки філософії» (1644). Декарт запропонував поділ Всесвіту на зони гравітаційного впливу зірок. Тільки через два століття, відомий німецький математик Йоганн Петер Густав Лежен-Діріхле (1805 - 1859) ввів діаграми для двох- і тривимірного випадків. Тому їх іноді називають діаграмами Діріхле. У 1908 році російський математик Георгій Феодосійович Вороний (16 (28) квітня 1868 - 7 (20) листопада 1908) описав цю діаграму для просторів великих розмірностей, з тих пір діаграма носить його прізвище. Георгій Феодосійович Вороний народився в селі Журавка Полтавської губернії (нині Чернігівська область). З 1889 роки навчався в Санкт-Петербурзькому університеті у Андрія Маркова. У 1894 році захистив магістерську дисертацію «Про цілі числа, що залежать від кореня рівняння третього ступеня». У тому ж році був обраний професором Варшавського університету, де вивчав ланцюгові дроби. У Вороного навчався Вацлав Серпінський. У 1897 році Вороний захистив докторську дисертацію «Про один узагальнений алгоритм безперервних дробів», відзначену премією імені Буняковського. Після цього працював професором математики в Варшавському Університеті. Брав участь у другому всесвітньому конгресі математиків під головуванням Гільберта. Автор шести великих статей. Дві з них (маються на легальному відкритому доступі) французькою мовою надруковані в трьох випусках в німецькому "Журналі чистої і прикладної математики":

1) Sur un problème du calcul des fonctions asymptotiques. Journal für die reine und angewandte Mathematik, 126; 241–282, 1903.

2) Nouvelles applications des paramètres continus à la théorie des formes quadratiques. Deuxième mémoire. Recherches sur les paralléloèdres primitifs. Journal für die reine und angewandte Mathematik, 134; 198–287, 1908. Це введення та перша частина статті.

3) Nouvelles applications des paramètres continus à la théorie des formes quadratiques. Deuxième mémoire. Recherches sur les paralléloèdres primitifs. Journal für die reine und angewandte Mathematik, 136; 67–182, 1909. Це друга частина статті. Помер у Варшаві від желчекам'яної хвороби болезни [13].

4.2 Принцип побудови

Діаграма складається з, так званих, локусів - областей, в яких присутні всі точки, які знаходяться ближче до цієї точки, ніж до всіх інших. У діаграмі Вороного локуси є опуклими багатокутниками.

Як будувати локус? За визначенням він буде будуватися так: нехай дано безліч з n точок, для якої будують діаграму. Візьмемо конкретну точку p , для якої будують локус, і ще одну точку з даної нам безлічі - q (не дорівнює p). Проведемо відрізок, що з'єднає ці дві точки, і проведемо пряму, яка буде серединним перпендикуляром даного відрізка. Ця пряма ділить площину на дві півплощини - в одній лежить точка p , в іншій лежить точка q (рис. 4.1).

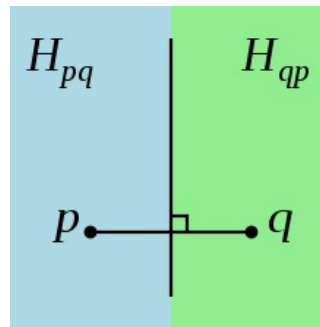


Рисунок 4.1 – побудова локусу

В даному випадку локусами цих двох точок є отримані півплощини. Тобто для того, щоб побудувати локус точки p , потрібно отримати перетин всіх таких напівплощин (тобто на місці q побувають всі точки даної множини, крім p).

Точку, для якої будується локус, називають сайтом (site). На рисунку 4.2 локуси позначені різними кольорами.

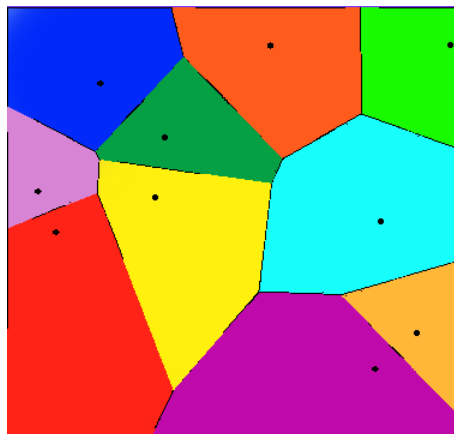


Рисунок 4.2 – Приклад розбиття площі на локуси

Алгоритми побудови діаграми як раз і є не що інше, як алгоритми побудови цих самих локусів для всіх точок з заданого набору. Локуси в даній задачі також називають багатокутниками Вороного або осередками Вороного.

4.3 Алгоритми побудови

Існує 4 алгоритму побудови діаграми Вороного:

1. Алгоритм побудови діаграми Вороного «у лоб». Складність: $O(n^4)$;
2. Алгоритм побудови діаграми Вороного шляхом перетину напівплощин. Складність: $O(n^2 \cdot \log(n))$;
3. Алгоритм Форчун побудови діаграми Вороного на площині. Складність: $O(n \cdot \log(n))$;
4. Рекурсивний алгоритм побудови діаграми Вороного. Складність: $O(n \cdot \log(n))$.

4.3.1 Алгоритм побудови діаграми Вороного «у лоб»

Тут ідея в тому, щоб перетинати НЕ півплощини, а саме серединні перпендикуляри відрізків (бо це простіше), що з'єднують цю точку з усіма іншими точками. Тобто, слідуючи визначенню осередку Вороного, будемо будувати локус для точки p так:

- 1) Отримуємо $n-1$ пряму (серединні перпендикуляри), так як провели серединні перпендикуляри всіх відрізків, що з'єднують цю точку p з іншими;
- 2) Перетинаємо попарно всі прямі, отримуємо $O(n^2)$ точок перетину (бо кожна пряма може перетнути всі інші, в «гіршому випадку»);
- 3) Перевіряємо всі ці $O(n^2)$ точок на приналежність кожної з $n-1$ напівплощин, тобто отримуємо вже асимптотику $O(n^3)$. Відповідно ті точки, які належать всім напівплощинам, і будуть вершинами осередку Вороного точки p ;
- 4) Проробляємо перші три кроки для всіх n точок, отримуємо підсумкову асимптотику $O(n^4)$.

4.3.2 Алгоритм побудови діаграми Вороного шляхом перетину напівплощин за $O(n^2 \cdot \log(n))$

Цей алгоритм вже можна використовувати на практиці, так як він не має настільки великої обчислювальної складності. Для нього нам буде потрібно вміти перетинати відрізки і прямі, вміти перетинати опуклі багатокутники, вміти перетинати півплощини, вміти об'єднувати отримані локуси в діаграму.

Сам алгоритм побудови представлений далі.

Отримуємо $n-1$ пряму для поточного сайту (як в попередньому алгоритмі - серединні перпендикуляри). Це будуть «утворюючі» напівплощини;

Тепер маємо $n-1$ напівплощина. Кожна з цих напівплощин задається будь-якій прямій з попереднього пункту і орієнтацією, тобто з якого боку від прямої вона розташована. Орієнтацію можна визначити за поточним сайтом, для якого будуємо локус - він лежить в шуканій півплощині, а значить і його локус повинен лежати в ній;

Перетинаємо всі півплощини - зможемо робити це за $O(n * \log(n))$ - отримуємо локус для поточного сайту;

Проробляємо перші три кроки для всіх n точок, отримуємо підсумкову асимптотику $n * O(n * \log(n)) = O(n^2 * \log(n))$.

Далі розглянемо приклад реалізації.

Обмежуємо всю область дій обмежувальним прямокутником - півплощини будуть відсікати від нього частини, які і перетнемо один з одним. Це вирішує проблему нескінченних осередків в діаграмі.

Перетин прямих і відрізків - потрібні саме точки перетину, а не просто визначення його наявності. Для реалізації використовувалася бібліотека SplashGeom ©, написана користувачами ресурсу habrhabr.ru Іллею Захаркін, в якій є все необхідне для реалізації багатьох алгоритмів обчислювальної геометрії на площині і деяких в просторі. У SplashGeom © це є - реалізація перетину прямих і відрізків представлена нижче наведеними кодом [14].

```
// перетин прямих
// kInfPoint – прями паралельні, kNegInfPoint – прями збігаються
Point2D Line2D::GetIntersection(const Line2D& second_line) const
{
    double cross_prod_norms = Vector2D(this->A, this->B).OrientedCCW(Vector2D(second_line.A, second_line.B));
    Point2D intersect_point;
    if (fabs(cross_prod_norms) <= EPS) /* A1 / A2 == B1 / B2 */ {
        if (fabs(this->B * second_line.C - second_line.B * this->C) <= EPS) /* .. == C1 / C2 */
        {
            intersect_point = kNegInfPoint2D; }
        else {
            intersect_point = kInfPoint2D; } }
    else {
        double res_x = (second_line.C * this->B - this->C * second_line.B) /
cross_prod_norms;
        double res_y = (second_line.A * this->C - this->A * second_line.C) /
cross_prod_norms;
        intersect_point = Point2D(res_x, res_y);
    }
    return intersect_point;
}
// перетин відрізків
Point2D Segment2D::GetIntersection(const Segment2D& second_seg) const
```

```

{
    Line2D first_line(*this);
    Line2D second_line(second_seg);
    Point2D intersect_point = first_line.GetIntersection(second_line);
    if (intersect_point == kNegInfPoint2D) {
        if (this->Contains(second_seg.b)) { intersect_point = second_seg.b; }
        else if (this->Contains(second_seg.a)) { intersect_point = second_seg.a; }
        else if (second_seg.Contains(this->b)) { intersect_point = this->b; }
        else if (second_seg.Contains(this->a)) { intersect_point = this->a; }
        else { intersect_point = kInfPoint2D; }
    } else if (!(this->Contains(intersect_point) && second_seg.Contains(intersect_point)))
    { intersect_point = kInfPoint2D; }
    return intersect_point; }

```

Тепер реалізуємо перетин багатокутників. Можна, звичайно, робити це за $O(nm)$, де n і m - кількість вершин першого і другого багатокутника відповідно - перетинати кожен бік першого багатокутника з кожною стороною з другого, записуючи точки перетину, а також перевіряти точки на приналежність іншому багатокутнику, але щоб досягти кращої швидкості, будемо перетинати за алгоритмом О'Рурка [15].

Алгоритм: Поки не пройде максимальна кількість ітерацій (доводиться, що вона не більше $2 * (n + m)$), виконуємо наступні інструкції: а). Беремо поточні ребра першого і другого багатокутників; б). Якщо вони перетинаються - тоді розглядаємо точку з перетину: може бути так, що ми тільки що її додали в багатокутник перетину, тоді просто ігноруємо додавання, інакше - якщо вона не є початковою точкою перетину (тобто ми вже зробили коло), то додаємо в перетин, інакше закінчуємо алгоритм - зустріли точку перетину, яка була на початку; в).

Далі (незалежно від того, перетинаються поточні ребра чи ні) викликаємо функцію Рух, яка відповідає за пересування вікна першого (або другого) багатокутника на одне ребро вперед, а також за можливе додавання вершини в багатокутник перетину. Основні дії відбуваються саме в русі. Якщо не було точок перетину, визначити, чи лежить один багатокутник всередині іншого. Якщо лежить, то повернути його, інакше повернути пустий перетин.

Функцію Руху можна реалізовувати по-різному, у нас вона повертає номер (мітку) багатокутника, ребро якого рухаємо. Концептуально вона всередині себе робить три речі: - визначає номер випадку - поточне взаємне розташування ребра першого багатокутника і ребра другого багатокутника. Це основна ідея алгоритму - перегляд випадків положення ребер відносно один одного. Всі ці чотири положення добре описані в книзі, в нашій реалізації положення визначається за допомогою косої добутку векторів на площині; - визначає, ребро якого багатокутника зараз «всередині», тобто воно лежить «зліва» від

іншого ребра, це теж перевіряється косим твором (лежить «зліва», якщо кінцева точка лежить «зліва»); - вирішує, чи записувати поточний кінець одного з ребер в багатокутник перетину. Якщо це відповідає потрібній нагоді і ребро всередині, то потрібно додати, інакше ні.

Реалізація заснована на описі алгоритму в книзі з деякими доповнюючими ідеями. Дана реалізація не враховує випадки, коли у багатокутників є загальні сторони (як зазначено в книзі, цей випадок вимагає окремого розгляду), однак випадки із загальними вершинами працюють коректно [16].

Код опису алгоритму в SplashGeom © представлений в додатку В.

Тепер є все необхідне для побудови перетину напівплощин. В силу асоціативності операції перетину напівплощин, можемо перетинати їх в будь-якому порядку, а значить перетнемо дві площини, потім перетнемо ще дві, а потім перетнемо їх перетин, це ж уже швидше, ніж перетинати все по-окремо.

Приклад коду представлений нижче.

```

Convex2D GetHalfPlanesIntersection(const Point2D& cur_point, const
vector<Line2D>& halfplanes, const Rectangle& border_box)
{
    if (halfplanes.size() == 1) {
        Convex2D cur_convex(border_box.GetIntersectionalConvex2D(cur_point,
halfplanes[0]));
        return cur_convex;
    } else {
        int middle = halfplanes.size() >> 1;
        vector<Line2D> first_half(halfplanes.begin(), halfplanes.begin() +
middle);
        vector<Line2D> second_half(halfplanes.begin() + middle,
halfplanes.end());
        Convex2D first_convex(GetHalfPlanesIntersection(cur_point, first_half,
border_box));
        Convex2D second_convex(GetHalfPlanesIntersection(cur_point,
second_half, border_box));
        return first_convex.GetIntersectionalConvex(second_convex);
    }
}

```

Тепер маємо багатокутник Вороного даної точки. Можна додати його в діаграму. Проблем з цим тут особливо не виникає, тому що отримуємо області в довільному порядку, і між собою вони ніяк не пов'язані (на відміну від реалізації в алгоритмі Форчуна, де можна перейти на сусідній локус за вказівником на ребро). Код реалізації діаграми Вороного представлений нижче.

```

// будуємо локус для поточного сайту
Voronoi2DLocus VoronoiDiagram2D::MakeVoronoi2DLocus(const Point2D& site,
const vector<Point2D>& points, const Rectangle& border_box)
{
    Voronoi2DLocus cur_locus;
    vector<Line2D> halfplanes;
    for (auto cur_point : points) {
        if (cur_point != site) {
            Segment2D cur_seg(site, cur_point);
            Line2D cur_halfplane(cur_seg.GetCenter(),
cur_seg.NormalVec());
            halfplanes.push_back(cur_halfplane);
        }
    }
    *cur_locus.region_ = GetHalfPlanesIntersection(site, halfplanes, border_box);
    cur_locus.site_ = site;
    return cur_locus;
}
// будуємо діаграму Вороного
VoronoiDiagram2D VoronoiDiagram2D::MakeVoronoiDiagram2DHalfPlanes(const
vector<Point2D>& points, const Rectangle& border_box)
{
    Voronoi2DLocus cur_locus;
    for (auto cur_point : points) {
        cur_locus = MakeVoronoi2DLocus(cur_point, points, border_box);
        this->diagram_.push_back(cur_locus);
    }
    return *this;
}

```

Так, можемо будувати діаграму Вороного за $O(n^2 \cdot \log(n))$, вона має вигляд вектора (списку) локусів. Недолік даного рішення - інформації про сусідів не одержати (можливо, цей недолік можна ліквідувати поліпшеною реалізацією).

4.3.3 Алгоритм Форчуна побудови діаграми Вороного на площині

У 1987 році Стів Форчун (Steve Fortune) запропонував алгоритм побудови діаграми за $O(n \cdot \log(n))$. Звичайно, він є не єдиним алгоритмом побудови з такою асимптотикою, але він досить зрозумілий і не дуже складний в реалізації.

Основна ідея алгоритму - це так звана замітаюча пряма (ЗП) (sweep line). Вона застосовується в багатьох алгоритмах обчислювальної геометрії, тому що дозволяє зручно моделювати рух прямої по деякій безлічі об'єктів (наприклад, в алгоритмі перетину n відрізків теж використовується sweep line).

Суть алгоритму полягає в наступному. Є n сайтів (точок на площині). Є замітаюча пряма, яка рухається (наприклад) «зверху вниз», тобто від сайту з найбільшою ординатою

до сайту з меншою (від події до події, якщо бути точним). Відразу варто відзначити, що вплив на побудову діаграми надають тільки ті сайти, які знаходяться вище або на заметаючій прямій.

Коли ЗП потрапляє на черговий сайт (відбувається подія точки (point event)), створюється нова парабола (arch), фокусом якої є даний сайт, а директрисою - замітаюча пряма. Ця парабола ділить площину на дві частини - «внутрішня» область параболи відповідає точкам, які зараз ближче до сайту, а «зовнішня» область - точкам, які ближче до sweep line, ну а точки, що лежать на параболі - рівновіддалені від сайту і ЗП. Парабола буде змінюватися в залежності від положення ЗП до сайту - чим далі ЗП йде від сайту вниз, тим більше розширюється парабола, однак на самому початку вона взагалі є відрізком («спрямованим» вгору).

Далі парабола розширюється, у неї з'являються дві контрольні точки (break points) - точки її перетину з іншими параболою («береговою лінією»). У «берегової лінії» ми зберігаємо дуги парабол від однієї точки перетину їх один з одним до іншої, так і виходить beach line. По суті, в цьому алгоритмі моделюємо рух цієї «берегової лінії», так як ці самі break point'и рухаються якраз по ребрах осередків Вороного (адже виходить, що контрольні точки рівновіддалені від обох сайтів, яким відповідають ці параболи, та ще й від ЗП).

І як раз в той момент, коли дві контрольні точки - по одній з різних парабол - «зустрічаються», тобто як би перетворюються в одну, ця точка і стає вершиною осередку Вороного (відбувається подія кола (circle event)), причому в цей час та дуга, яка перебувала між цими двома точками - «схлопується» і видаляється з «берегової лінії». Далі просто з'єднуємо цю точку з попередньою відповідною їй і отримуємо ребро осередку Вороного [17].

4.3.4 Рекурсивний алгоритм побудови діаграми Вороного

Алгоритм полягає в наступному:

- ділимо всю безліч сайтів S на дві приблизно рівні частини (може бути непарна кількість точок) S_1 і S_2 ;
- рекурсивно будуємо діаграми Вороного для S_1 і S_2 ;
- об'єднуємо отримані діаграми і отримуємо діаграму для S .

Загальний опис алгоритму не складний, проте в алгоритмі є свої тонкощі [18].

4.4 Використання діаграмі Вороного для розподілу елементів БСМ нашого підприємства

4.4.1 Опис підприємства

Метою роботи є дослідження та створення бездротової сенсорної мережі з метою моніторингу та аналізу умов праці на промислових підприємствах, для початку позначимо його відділи. Будинок розділений на 2 частини, Приміщення 1 - виробничий цех, і приміщення 2 - оцінний цех. Обидва цеху розділені на відділи, деякі з них ізольовані стінами (рис. 4.3).

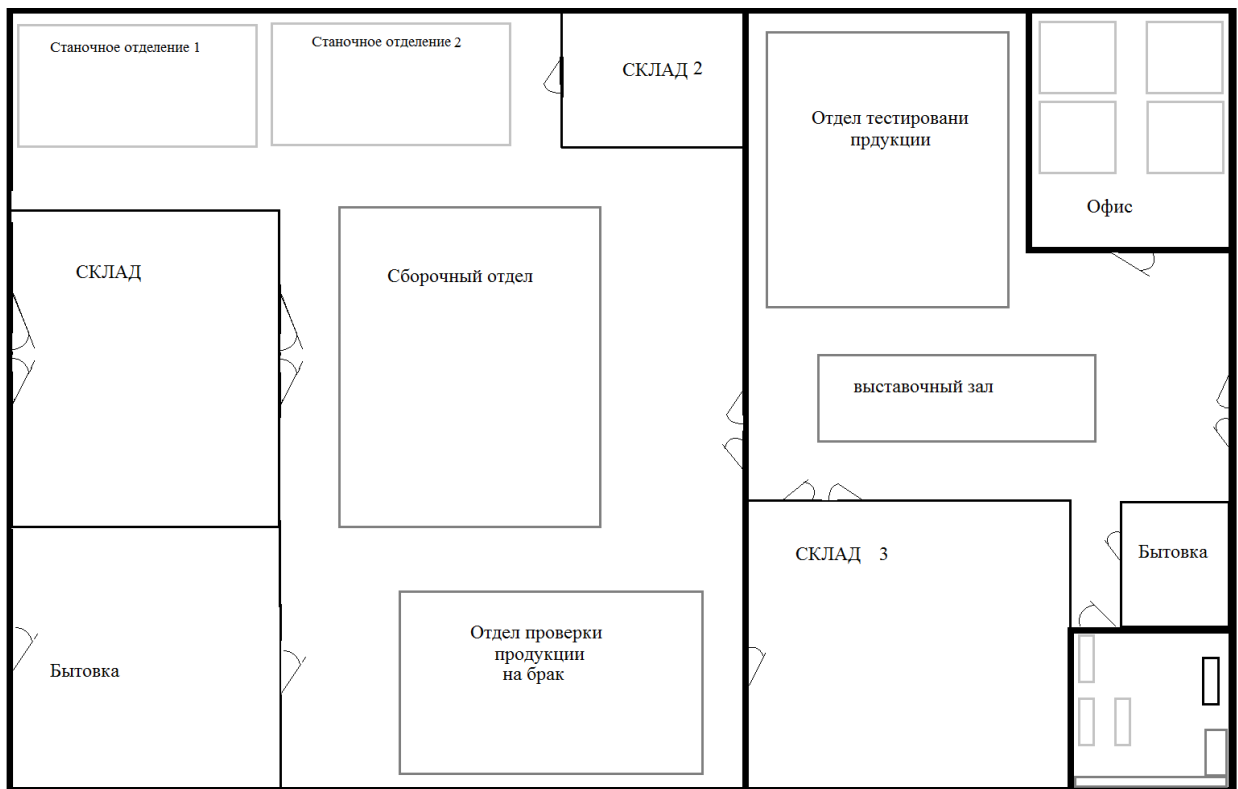


Рисунок 4.3 – Схема підприємства

У кожному відділі знаходяться люди, тому в них необхідний контроль якості повітря і освітленості. Для безпеки так само використовуються датчики руху і відкриття/закриття дверей/вікон. Кожен відділ буде оснащений своїм маршрутизатором. Всі маршрутизатори підключені до координатора, він же є шлюзом для передачі даних на сервер. Доступ до приміщення оцінчного цеху закритий з використанням дверей і розумного ZigBee замку, доступ обмежений. Всі двері в цеху обладнані даними замком.

4.4.2 Проектування мережі ZigBee

Для розподілу маршрутизаторів необхідно визначити їх місце в кожному відділі. Для будівлі необхідно 11 маршрутизаторів. Встановивши всі таким чином, буде легше проводити моніторинг і управління приміщень [19].

У мінімальному випадку можна обійтися 3 маршрутизаторами, але в цьому випадку при виході одного з них певна зона залишиться без можливості бути обслугованою. Зони їх покриття можна визначити за допомогою Діаграми Вороного. У приміщенні 2 маршрутизатор ставиться в геометричному центрі. У приміщенні 1 для покриття всієї площі необхідно 2 маршрутизатора (точки - сайти), кожен з них буде охоплювати половину приміщення. Для визначення зони роботи необхідно на центрі прямий, що з'єднує сайти, провести перпендикуляр. Отримаємо локуси - це і буде поділ зон дії [20]. Приклад на рисунку 4.4.

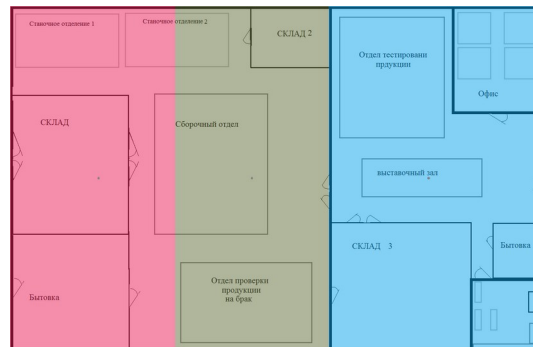


Рисунок 4.4 – Обслуговування будівлі з використання 3х маршрутизаторів

При подібному розміщенні при виході з ладу одного з маршрутизаторів зони дії іншого недостатньо для покриття необхідної площині (рис. 4.5).

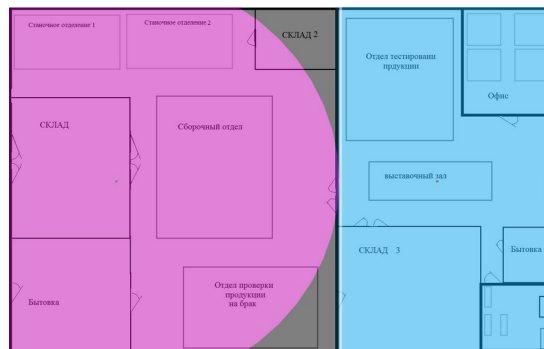


Рисунок 4.5 – Приклад випадку, коли один з маршрутизаторів вийшов з ладу

При використанні окремого маршрутизатора в області кожного відділу отримуємо простоту контролю і управління, а також відмовостійкість системи. Для їх розподілу необхідна схема підприємства. Для початку розставимо сайти по центрах відділів, проведемо серединні перпендикуляри. Так можемо бачити локуси для всіх сайтів. Далі змінюючи положення сайтів, і перебудовуючи заново серединні перпендикуляри, добиваємося необхідних локусів для всіх сайтів. Таким чином, отримаємо оптимальне розміщення маршрутизаторів в їх зоні роботи (рис. 4.6).

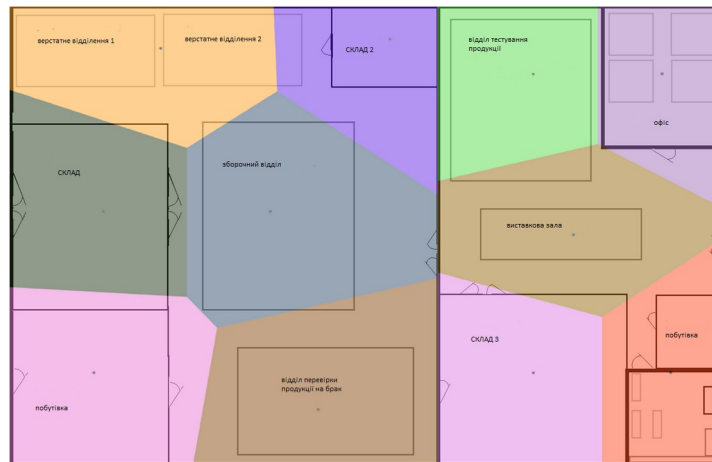


Рисунок 4.6 – Схема покриття площі будівлі усіма встановленими маршрутизаторами ZigBee

При виході з ладу одного або декількох маршрутизаторів система продовжить функціонувати в нормальному режимі. Кінцеві пристрої підключаються до найближчих робочих маршрутизаторів і продовжать роботу. Приклад роботи системи при несправності 5 маршрутизаторів представлений на рисунку 4.7.

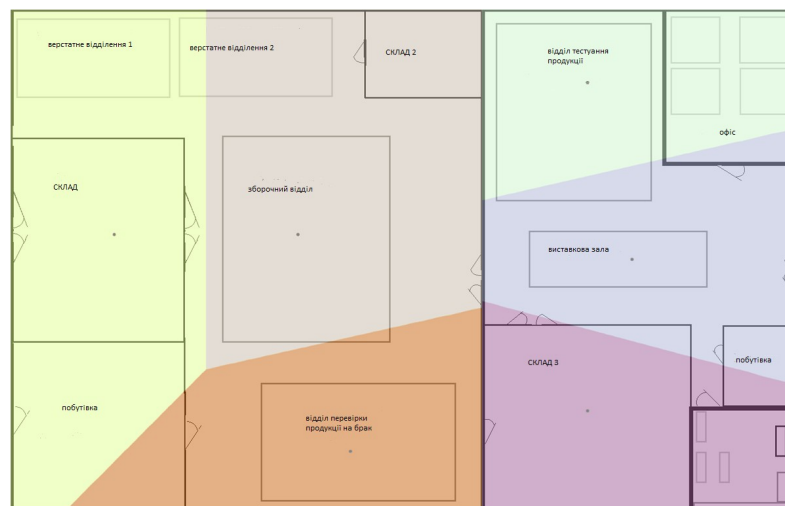


Рисунок 4.7 – Покриття зони з декількома несправними маршрутизаторами

Перелік посилань до розділів 1-4

1. Баскаков С.С. Досвід практичного впровадження бездротових сенсорних мереж / С.С. Баскаков // Журнал Автоматизація в промисловості. - січень 2016. – С. 7-11.
2. Вишневський В.М. Широкосмугові бездротові мережі передачі інформації / В.М. Вишневський, А.І. Ляхов, С. Кравець, І.В. Шахновіч. - М. : Техносфера, 2005. – 592 с.
3. Колесніков С. Бездротові мережі. Стандарти [Електронний ресурс] / С.КОЛЕСНИКОВ // онлайн газета КОМП'ЮТЕР-ІНФОРМ. - Режим доступу: http://old.ci.ru/inform19_05/p_24.htm. - Дата доступу: 11.10.2016. - Загл. з екрану.
4. Wireless MAN [Електронний ресурс] // Матеріал з Вікіпедії - вільної енциклопедії. - Режим доступу https://ru.wikipedia.org/wiki/Wireless_MAN. - Дата доступу: 11.10.2016. - Загл. з екрану.
5. WWAN [Електронний ресурс] // Матеріал з Вікіпедії - вільної енциклопедії. - Режим доступу <https://ru.wikipedia.org/wiki/WWAN>. - Дата доступу: 11.10.2016. - Загл. з екрану.
6. Терентьев М.Н. Бездротові сенсорні мережі / М.Н. Терентьев // Навчальний посібник. - М. : Видавництво МАІ, 2008. - 96 С.
7. Баскаков С.С. Бездротові сенсорні мережі на базі платформи MeshLogic / С.С. Баскаков, В.І. Оганов // Електронні компоненти. - Москва. - 2006. - №8. - С. 65-69.
8. Восків Л.С. Енергоефективний комбінований метод локалізації в бездротових сенсорних мережах / Л.С. Восків, Р.О. Курпатов. // Журнал «Датчики і системи». – М.: ООО «СенСіДат-Контрол». – №4. – 2011. – 153 с.
9. Пушкарьов О. ZigBee-модулі MaxStream нові можливості / О. Пушкарьов // Журнал Новини Електроніки. – № 2. – 2007. – С. 25-27.
10. XBee і XBee-PRO S2C - нові ZigBee модулі від DIGI [Електронний ресурс] // Москва, компанія «Компел». - Режим доступу: <http://www.compel.ru/2016/09/14/xbee-i-xbee-pro-s2c-novyye-zigbee-moduli-ot-digi>. - Дата доступу: 14.10.2016. - Загл. з екрану.
11. Пушкарьов О. Передача даних в ZigBee мережі за допомогою модулів XBee ZNet 2.5 / О. Пушкарьов // Журнал Новини Електроніки. - 2008. - № 3. -С. 27-32.
12. Шахновіч І.А. Сучасні технології бездротового зв'язку. / Шахновіч І.А. .. М. : Техносфера – 2006. – 288 с.
13. Вороний, Г. Ф. [Електронний ресурс] / Матеріал з Вікіпедії - вільної енциклопедії. - Режим доступу:

https://ru.wikipedia.org/wiki/Вороной,_Георгий_Феодосьевич. - Дата доступу: 21.10.2016. - Загл. з екрану.

14. Захаркін І. SPLashGeom (Open-source C ++ library for geometry and linear algebra) [Електронний ресурс] / І.Захаркін. - Режим доступу: <https://github.com/izaharkin/SplashGeom>. Дата доступу: 25.10.2016. - Загл. з екрану.

15. O'Rourke J. Computational geometry in C / J. O'Rourke. – New York:Cambridge University press. – 1998. – 358 p.

16. Препарату Ф. Обчислювальна геометрія. Введення / М. Шеймос, Ф. Препарату. - Москва, «МИР». – 1989. – 478с.

17. Mark de Berg *Computational Geometry* (вид. 2nd revised) / Mark de Berg, Marc van Kreveld, Mark Overmars, and Otfried Schwarzkopf. – [Springer-Verlag](https://www.springer.com). – 388p.

18. Вишневський В.М. Теоретичні основи проектування комп'ютерних мереж / В.М. Вишневський. - Москва: «Техносфера». – 2003. – 512 с.

19. Горелов В.П. Огляд технології ZIGBEE / В.П.Горелов, Г.А.Данілов, М.А.Кручинін, М.В.Рябов // Наук. пробл. трансп. Сиб. і Дав. сх. – №2. – 2012. – С. 239 - 243.

20. Р. Педжман Основи побудови бездротових локальних мереж стандарту 802.15.4. / Рошан Педжман, Ліері Джонатан // Пер. з англ. - М .: Видавничий дом«Вільяме ». – 2004. – 304 с.

5 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

В даному розділі наведено дані з охорони навколишнього природного середовища, проведено аналіз потенційних небезпечних та шкідливих виробничих факторів, причин пожеж. Розглянуті заходи, які дозволяють забезпечити гігієну праці і виробничу санітарію. На підставі аналізу розроблені заходи з техніки безпеки та рекомендації з пожежної профілактики.

Аналіз потенційно небезпечних і шкідливих виробничих чинників виконується для персонального комп'ютера, на якому буде виконуватися розробка.

5.1 Загальні дані з охорони навколишнього природного середовища

Вплив на атмосферне повітря, діяльність за темою магістерської роботи(створення бездротової сенсорної мережі) при нормальних умовах праці не оказує, бо не має в приміщенні сканерів, принтерів та інших джерел викиду забруднюючих речовин в повітря робочої зони.

В процесі створення/розробки програми на робочому місці виникають такі відходи:

- Змінні носії інформації - IV клас безпеки
- Макулатура - IV клас безпеки
- Побутові відходи - IV клас безпеки

5.2 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу

Аналіз небезпечних та шкідливих виробничих факторів виконується у табличній формі (табл. 5.1). Роботу, пов'язану з ЕОП з ВДТ, у тому числі на тих, які мають робочі місця, обладнані ЕОМ з ВДТ і ПП, виконують із забезпеченням виконання виконують із забезпеченням виконання НПАОП 0.00-7.15-18 [6] «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями», які встановлюють вимоги безпеки до обладнання робочих місць, до роботи із застосуванням ЕОМ з ВДТ і ПП. Основними робочими характеристиками персонального комп'ютера є:

- робоча напруга $U=+220\text{В} \pm 5\%$;
- робочий струм $I=2\text{А}$;
- споживана потужність $P=350\text{ Вт}$.

Таблиця 5.1 – Аналіз небезпечних і шкідливих виробничих факторів

Небезпечні і шкідливі виробничі фактори	Джерела факторів (види робіт)	Кількісна оцінка	Нормативні документи
Фізичні			
- підвищений рівень напруги електричної мережі, замикання якої може відбутися через тіло людини	система організаційних і технічних заходів і засобів, що забезпечують захист від шкідливого і небезпечного впливу електричного струму, електричної дуги, електромагнітного поля і статичної електрики.	4	[4]
- недостатність природного світла	порушення умов праці (вимог до приміщень)	2	[5]
- недостатнє освітлення робочої зони	порушення гігієнічних параметрів	3	[5]
психофізіологічні:			
- нервово-психічна перевантаження (розумове, перенапруження аналізаторів-зорових)	<ul style="list-style-type: none"> - пошук інформації для постановки теми; - пошук та аналіз аналогів і літератури; - пошук наявних технологій, моделювання та аналіз алгоритмів; - виконання роботи за темою диплома, тестування; - оформлення роботи 	4	[6] [3]

Продовження таблиці 5.1

- нервово-психічна перевантаження (розумове, перенапруження аналізаторів-зорових)	<ul style="list-style-type: none"> - пошук інформації для постановки теми; - пошук та аналіз аналогів і літератури; - пошук наявних технологій, моделювання та аналіз алгоритмів; - виконання роботи за темою диплома, тестування; - оформлення роботи 	4	[6] [3]
---	---	---	------------

5.3 Гігієнічні вимоги до параметрів виробничого середовищ**5.3.1 Освітлення**

Світло є природною умовою існування людини. Воно впливає на стан вищих психічних функцій і фізіологічні процеси в організмі. Хороше освітлення діє тонізуюче, створює гарний настрій, покращує протікання основних процесів вищої нервової діяльності.

У приміщенні, де розташовані ЕОМ передбачається природне бічне освітлення, рівень якого відповідає ДБН В.2.5-28:2018 [5]. Джерелом природного освітлення є сонячне світло. Регулярно повинен проводитися контроль освітленості, який підтверджує, що рівень освітленості задовольняє ДБН і для даного приміщення в світлий час доби достатньо природного освітлення.

Розрахунок освітлення.

Для виробничих та адміністративних приміщень світловий коефіцієнт приймається не менше $1/8$, в побутових – $1/10$:

$$S_b = \left(\frac{1}{5} \div \frac{1}{10} \right) \cdot S_n, \quad (5.1)$$

де S_b – площа віконних прорізів, m^2 ;

S_n – площа підлоги, m^2 .

$$S_n = a \cdot b = 5 \cdot 5 = 25 \text{ м}^2,$$

$$S = 1/8 \cdot 25 = 3,125 \text{ м}^2.$$

Приймаємо 2 вікна площею $S=1,6 \text{ м}^2$ кожне.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для створення заданої освітленості при загальному рівномірному освітленні.

Розрахунок кількості світильників n виробляється по формулі (5.2):

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M}, \quad (5.2)$$

де E – нормована освітленість робочої поверхні, визначається нормами – 300 лк;

S – освітлювана площа, м^2 ; $S = 25 \text{ м}^2$;

Z – поправочний коефіцієнт світильника ($Z = 1,15$ для ламп розжарювання та ДРЛ; $Z = 1,1$ для люмінесцентних ламп) приймаємо рівним 1,1;

K – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1,5;

U – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. – 0,575

M – число люмінесцентних ламп в світильнику – 2;

F – світловий потік лампи – 5400лм (для ЛБ-80).

Підставивши числові значення у формулу (5.2), отримуємо:

$$n = \frac{300 \cdot 25 \cdot 1,1 \cdot 1,5}{5400 \cdot 0,575 \cdot 2} \approx 2,0$$

Приймаємо освітлювальну установку, яка складається з 2-х світильників, які складаються з двох люмінесцентних ламп загальною потужністю 160 Вт, напругою – 220 В.

5.3.2 Шум та вібрація, електромагнітне випромінювання

Рівень шуму, зумовлений як роботою системного блоку, клавіатури, так і друкуванням на принтері, а також зовнішніми чинниками, коливається у межах 50–65 дБА ДСН 3.3.6.042-99 [2].

Віброізоляцію можливо здійснювати за допомогою спеціальної прокладки під

системний блок, яка послаблює передачу вібрацій робочого столу. Вібрація на робочому місці в приміщенні, що розглядається, відповідає нормам ДСН 3.3.6.042-99 [2].

5.3.3 Вентилювання

У приміщенні, де знаходяться ЕОМ, повітрообмін реалізується за допомогою природної організованої вентиляції (вентиляційні шахти), тобто при V приміщення > 40 м³ на одного працюючого допускається природна вентиляція. Цей метод забезпечує приток потрібної кількості свіжого повітря, що визначається в СНіП.

Також має здійснюватися провітрювання приміщення, в залежності від погодних умов, тривалість повинна бути не менше 10 хв. Найкращий обмін повітря здійснюється при наскрізному провітрюванні.

Розрахунок захисного заземлення (забезпечення електробезпеки будівлі).

Згідно з класифікацією приміщень за ступенем небезпеки ураження електричним струмом, приміщення в якому проводиться робота відноситься до першого класу (без підвищеної небезпеки). Коефіцієнт використання вертикальних заземлювачів η_v в залежності від розміщення заземлювачів та їх кількості знаходиться в межах 0,4...0,99. Взаємну екрануючу дію горизонтального заземлювача (з'єднувальної смуги) враховують за допомогою коефіцієнта використання горизонтального заземлювача η_c .

Послідовність розрахунку.

1) Визначається необхідний опір штучних заземлювачів $R_{шт.з.}$:

$$R_{шт.з.} = \frac{R_d \cdot R_{пр.з.}}{R_{пр.з.} - R_d}, \quad (5.3)$$

де $R_{пр.з.}$ – опір природних заземлювачів; R_d – допустимий опір заземлення. Якщо природні заземлювачі відсутні, то $R_{шт.з.} = R_d$.

Підставивши числові значення у формулу (5.3), отримуємо:

$$R_{шт.з.} = \frac{4 \cdot 40}{40 - 4} \approx 4 \text{ Ом}$$

1) Опір заземлення в значній мірі залежить від питомого опору ґрунту ρ , Ом·м. Приблизне значення питомого опору глини приймаємо $\rho=40$ Ом·м (табличне значення).

2) Розрахунковий питомий опір ґрунту, $\rho_{\text{розр}}$, Ом·м, визначається відповідно для вертикальних заземлювачів $\rho_{\text{розр.в}}$, і горизонтальних $\rho_{\text{розр.г}}$, Ом·м за формулою:

$$\rho_{\text{розр.}} = \psi \cdot \rho, \quad (5.4)$$

де ψ – коефіцієнт сезонності для вертикальних заземлювачів І кліматичної зони з нормальною вологістю землі, приймається для вертикальних заземлювачів $\rho_{\text{розр.в}}=1,7$ і горизонтальних $\rho_{\text{розр.г}}=5,5$ Ом·м.

$$\rho_{\text{розр.в}} = 1,7 \cdot 40 = 68 \text{ Ом} \cdot \text{м}$$

$$\rho_{\text{розр.г}} = 5,5 \cdot 40 = 220 \text{ Ом} \cdot \text{м}$$

3) Розраховується опір розтікання струму вертикального заземлювача $R_{\text{в}}$, Ом, за (5.5).

$$R_{\text{в}} = \frac{\rho_{\text{розр.в}}}{2 \cdot \pi \cdot l_{\text{в}}} \cdot \left(\ln \frac{2 \cdot l_{\text{в}}}{d_{\text{ст}}} + \frac{1}{2} \cdot \ln \frac{4 \cdot t + l_{\text{в}}}{4 \cdot t - l_{\text{в}}} \right), \quad (5.5)$$

де $l_{\text{в}}$ – довжина вертикального заземлювача (для труб - 2–3 м; $l_{\text{в}}=3$ м);

$d_{\text{ст}}$ – діаметр стержня (для труб - 0,03–0,05 м; $d_{\text{ст}}=0,05$ м);

t – відстань від поверхні землі до середини заземлювача, яка визначається за ф.

(5.6):

$$t = h_{\text{в}} + \frac{l_{\text{в}}}{2}, \quad (5.6)$$

де $h_{\text{в}}$ – глибина закладання вертикальних заземлювачів (0,8 м); тоді $t = 0,8 + \frac{3}{2} = 2,3$

м

$$R_{\text{в}} = \frac{68}{2 \cdot \pi \cdot 3} \cdot \left(\ln \frac{2 \cdot 3}{0,05} + \frac{1}{2} \cdot \ln \frac{4 \cdot 2,3 + 3}{4 \cdot 2,3 - 3} \right) = 18,5 \text{ Ом}$$

4) Визначається теоретична кількість вертикальних заземлювачів n штук, без урахування коефіцієнта використання η_v :

$$n = \frac{2 \cdot R_B}{R_d} = \frac{2 \cdot 18,5}{4} = 9,25 \quad (5.7)$$

Γ визначається коефіцієнт використання вертикальних електродів групового заземлювача без врахування впливу з'єднувальної стрічки $\eta_v = 0,57$ (табличне значення).

5) Визначається необхідна кількість вертикальних заземлювачів з урахуванням коефіцієнта використання n_B , шт:

$$n_B = \frac{2 \cdot R_B}{R_d \cdot \eta_v} = \frac{2 \cdot 18,5}{4 \cdot 0,57} = 16,2 \approx 16 \quad (5.8)$$

6) Визначається довжина з'єднувальної стрічки горизонтального заземлювача l_c , м:

$$l_c = 1,05 \cdot L_B \cdot (n_B - 1), \quad (5.9)$$

де L_B – відстань між вертикальними заземлювачами, (прийняти за $L_B = 3$ м);

n_B – необхідна кількість вертикальних заземлювачів.

$$l_c = 1,05 \cdot 3 \cdot (16 - 1) \approx 48 \text{ м}$$

7) Визначається опір розтіканню струму горизонтального заземлювача (з'єднувальної стрічки) R_Γ , Ом:

$$R_\Gamma = \frac{\rho_{\text{розр.}\Gamma}}{2 \cdot \pi \cdot l_c} \cdot \ln \frac{2 \cdot l_c^2}{d_{\text{см}} \cdot h_\Gamma}, \quad (5.10)$$

де $d_{\text{см}}$ – еквівалентний діаметр смуги шириною b , $d_{\text{см}} = 0,95b$, $b = 0,15$ м;

h_Γ – глибина закладання горизонтальних заземлювачів (0,5 м);

l_c - довжина з'єднувальної стрічки горизонтального заземлювача l_c , м

$$R_{\Gamma} = \frac{220}{2 \cdot \pi \cdot 48} \cdot \ln \frac{2 \cdot 48^2}{0,95 \cdot 0,15 \cdot 0,5} = 8,1 \text{ Ом}$$

8) Визначається коефіцієнт використання горизонтального заземлювача η_c відповідно до необхідної кількості вертикальних заземлювачів n_b .

Коефіцієнт використання з'єднувальної смуги $\eta_c = 0,3$ (табличне значення).

9) Розраховується результуючий опір заземлювального електроду з урахуванням з'єднувальної смуги:

$$R_{\text{заг}} = \frac{R_b \cdot R_{\Gamma}}{R_b \cdot \eta_c + R_{\Gamma} \cdot n_b \cdot \eta_b} \leq R_d. \quad (5.11)$$

Висновок: дане захисне заземлення буде забезпечувати електробезпеку будівлі, так як виконується умова: $R_{\text{заг}} < 4 \text{ Ом}$, а саме:

$$R_{\text{заг}} = \frac{18,5 \cdot 8,1}{18,5 \cdot 0,3 + 8,1 \cdot 16 \cdot 0,57} = 1,9 \leq R_d$$

1) При виникненні пожеж при роботі на ПЕОМ від таких можливими джерел запалювання як:

- іскри і дуги коротких замикань;
- перегрів провідників, резисторів та інших радіодеталей ПЕОМ, від тривалої перевантаження та наявності перехідного опору;
- іскри при розмиканні і розмиканні ланцюгів;
- розряди статичної електрики;
- необережному поводженню з вогнем, а також вибухи газо-повітряних і пароповітряних сумішей.

5.4 Висновки до розділу 5

В результаті проведеної роботи було зроблено аналіз умов праці, шкідливих та небезпечних чинників, з якими стикається робітник. Було визначено параметри і певні характеристики приміщення для роботи над запропонованим проектом, описано, які заходи потрібно зробити для того, щоб дане приміщення відповідало необхідним нормам і було комфортним і безпечним для робітника. Приведені рекомендації щодо організації робочого місця, а також важливу інформацію щодо пожежної та електробезпеки. Були наведені розміри приміщення та наведено значення температури, вологості й рухливості повітря, необхідна кількість і потужність ламп та інші параметри, значення яких впливає на умови праці робітника, а також – наведені інструкції з охорони праці, техніки безпеки при роботі на комп'ютері.

Зараз як ніколи актуальною є проблема забруднення навколишнього середовища. Робота пов'язана з інформаційними технологіями також несе загрозу забруднення довкілля, адже наслідками використання несправного обладнання або його неправильної експлуатації може стати шкода екології. Так само потрібно сортувати і утилізувати відходи в процесі роботи. Виходячи з вищесказаного можна зробити висновок що дотримуватися нормативних документів охорони праці є обов'язковим.

5.5 Перелік джерел посилань до розділу 5

1. Закон України «Про охорону праці». Режим доступу: <https://zakon.rada.gov.ua/laws/show/2694-12> - 10.14.1992 р.
2. Державні санітарні норми. ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень» - Режим доступу: <https://zakon.rada.gov.ua/rada/show/va042282-99> - 01.02.1999 р.
3. Державні санітарні правила і норми. ДСанПіН 3.3.2.007-98 «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» - Режим доступу: <https://zakon.rada.gov.ua/rada/show/v0007282-98> - 10.12.1998_р.
4. Державний стандарт України. ДСТУ Б В.2.5-82:2016 «Електробезпека в будівлях і спорудах. Вимоги до захисних заходів від ураження електричним струмом» - Режим доступу: <http://epicentre.co.ua/dstu/doc28522.html> - 01.07.2016 р.
5. Державні будівельні норми. ДБН В.2.5-28:2018 «Природне і штучне освітлення» - Режим доступу: <http://www.minregion.gov.ua/wp-content/uploads/2018/12/V2528-1.pdf> - 03.10.2018
6. Нормативно-правовий акт з охорони праці. НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» - Режим доступу: <https://zakon.rada.gov.ua/laws/show/z0508-18> - 14.02.2018 р.

ВИСНОВКИ

Проблеми, пов'язані з безпекою і контролем стану промислових об'єктів, є актуальною темою в сучасному світі. У зв'язку з цим доцільно постійне спостереження за показниками, які можуть бути на даних об'єктах, з метою організації безпечних умов праці, прогнозування та швидкого реагування на надзвичайні ситуації.

Бездротові сенсорні мережі, інтегровані в промислове підприємство, припускають організацію вузлів зв'язку, в які входять датчики і передавальні пристрої, які здійснюватимуть збирання і передачу даних на сервер для прийняття рішення про стан об'єкта.

У цій атестаційній роботі був виконаний вибір апаратного рішення і розглянутий спосіб побудови бездротової сенсорної мережі для виробничого підприємства.

Поставлені завдання:

- виконано огляд літератури за вибором комплектуючих вузла зв'язку для промислового об'єкта;
- досліджено існуючі рішення в області БСМ;
- розроблений план впровадження БСМ в підприємства з варіантами установки сенсорів;
- оптимізовані обсяги використання необхідного обладнання.

На підставі поставлених завдань запропонована модель вузла зв'язку, що складається з сервера, шлюзу, координатора і різних датчиків. Була розроблена БСМ для виробничого підприємства, яка дозволить оптимізувати витрати на електрику, поліпшити робочі умови персоналу і підвищити безпеку підприємства.

Датчики збирають необхідні дані і передають на сервер, де відбувається їх обробка. В результаті чого, за допомогою ПЗ, можна контролювати процеси вентиляції, обігріву приміщення і т.д. Надалі можлива доробка систем управління на основі ZigBee. З їх допомогою можливо буде автоматично регулювати закриття/відкриття жалюзі на вікнах, зміна рівня освітленості і вентиляції, обігрів приміщення. В результаті установки даної системи з'являється можливість відслідковувати пересування персоналу в приміщенні виробництва, а так само автоматизувати транспортування виробів в межах приміщень.

ДОДАТОК А

Графічний матеріал магістерської атестаційної роботи

АТЕСТАЦІЙНА РОБОТА

Тема:

Бездротові сенсорні мережі в системах контролю стану
промислового обладнання

Студент:
Смоляний Денис Костянтинович

Науковий керівник:
д.т.н., проф. Рязанцев Олександр Іванович

2020 р.



Рисунок А.1 – слайд 1

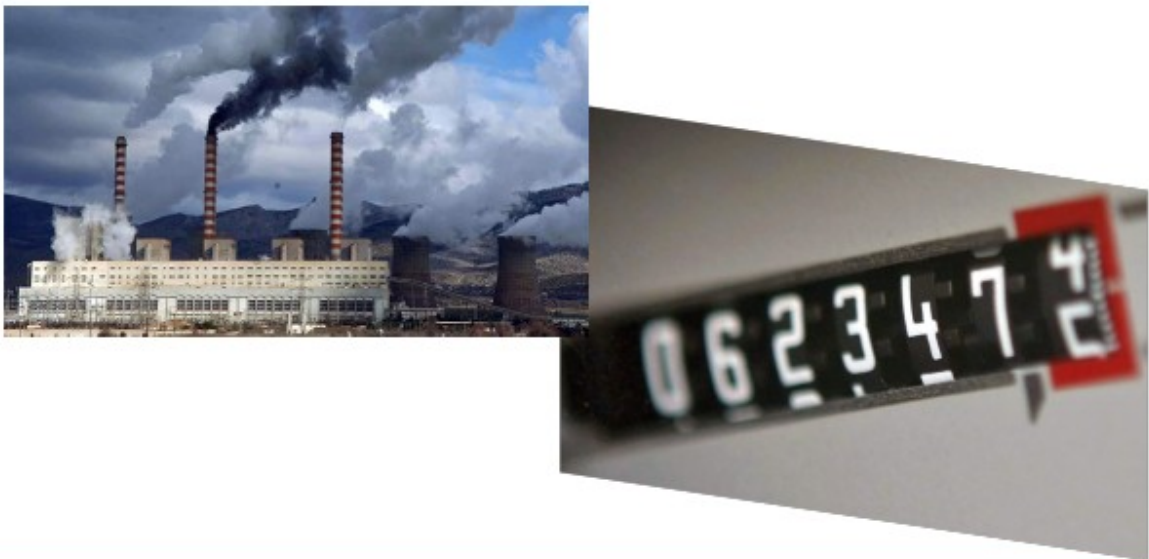
Практично всі сфери життя в 21 столітті залежать від інформаційно-комунікаційних технологій



2

Рисунок А.2 – слайд 2

Нерациональне використання ресурсів і виробничих потужностей



3

Рисунок А.3 – слайд 3

МЕТА РОБОТИ –

Дослідження та створення бездротової сенсорної мережі з метою моніторингу та аналізу умов праці на промислових підприємствах.

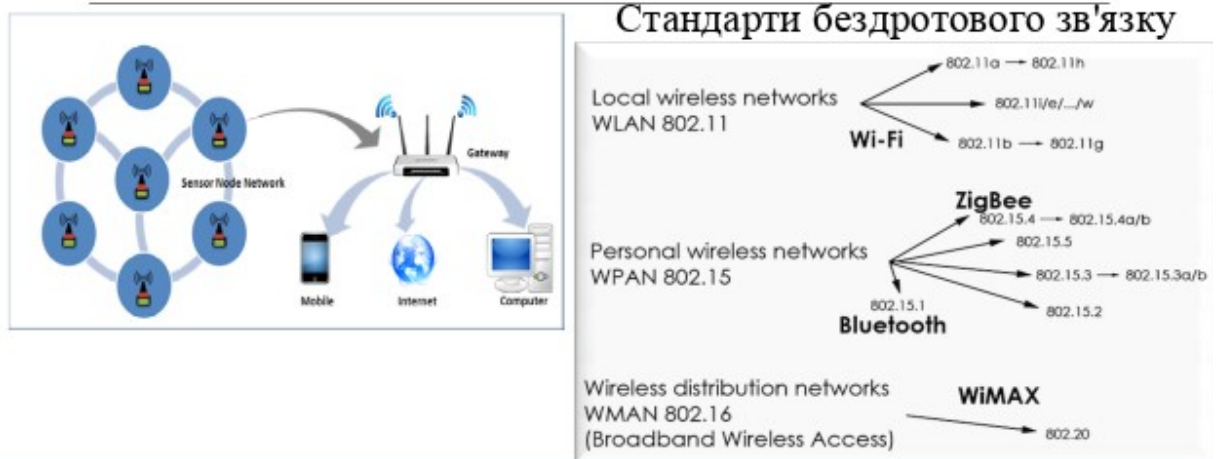
Постановка задачі роботи

- ➔ виконати огляд літератури за вибором комплектуючих вузла зв'язку для промислового об'єкта;
- ➔ дослідження існуючих рішень в області БСМ;
- ➔ розробка плану впровадження БСМ в підприємства з варіантами установки сенсорів;
- ➔ оптимізувати обсяги використання необхідного обладнання.

4

Рисунок А.4 – слайд 4

Бездротова сенсорна мережа



5

Рисунок А.5 – слайд 5

ВАРІАНТИ РЕАЛІЗАЦІЇ БЕЗДРОТОВИХ СЕНСОРНИХ МЕРЕЖ

Перший варіант - побудова власної системи на основі стандарту ZigBee

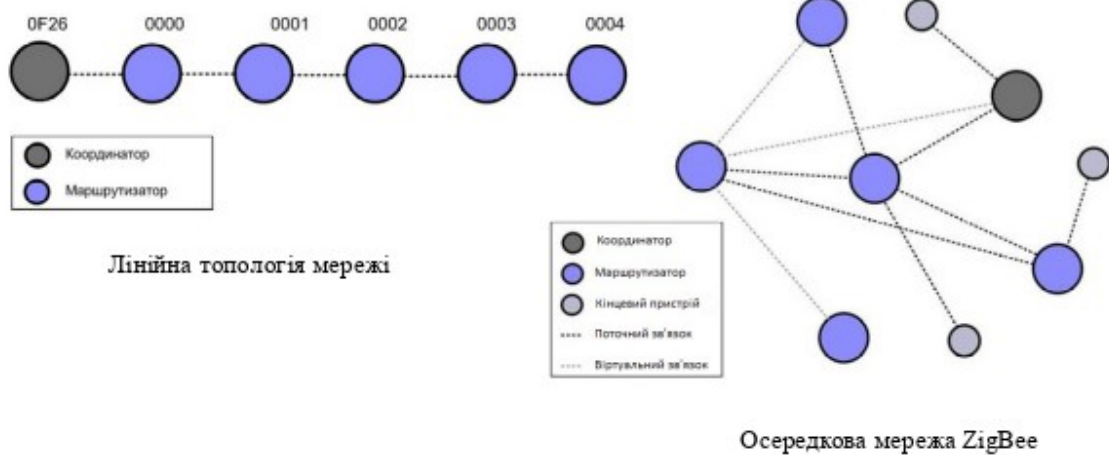


Другий варіант - застосування однієї з доступних пропрієтарних платформ (наприклад, від Sencicast, Millennium Net або Dust Networks).

6

Рисунок А.6 – слайд 6

ТИПИ МЕРЕЖ



7

Рисунок А.7 – слайд 7

Формування мережі ZigBee

MAC асоціація

1. Координатор або маршрутизатор, який збирається дозволити іншим пристроям приєднатися до нього, повинен виставити на свій MAC рівень дозвіл на приєднання NLME-PERMIT-JOINING.request.
2. У свою чергу, пристрій, що бажає вступити в мережу, виставляє на свій MAC рівень запит на приєднання NLME-NETWORK-DISCOVERY.request.
3. Після цього пристрій, що вступає в мережу передає в ефір широкомовний запит маячка і отримує цей маячок від маршрутизатора (ів, готового(их) підключити новий пристрій).
4. Отримавши маячок (маячки), пристрій визначає, в яку мережу він бажає вступити і до якого конкретно пристрою приєднатися, після чого виставляє на свій MAC рівень вимогу NLME-JOIN.request з встановленим в значення FALSE прапорцем «повторне приєднання» (rejoin).
5. Зробивши це, пристрій, що вступає, направляє на обраний маршрутизатор запит на приєднання і отримує відповідь, що містить присвоєну йому мережеву адресу.

Повторне мережеве приєднання (NWK rejoin).

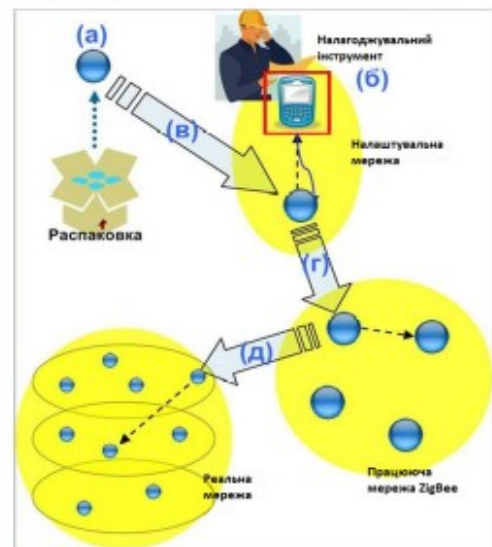
Стандартний механізм MAC асоціації в цьому випадку може або застосовуватися, або ні. Обмін пакетами може бути безпечним, якщо пристрій, що вступає, знає поточний мережевий ключ. Це можливо, якщо пристрій дійсно перепідключається, або якщо він влючається вперше, але отримав мережевий ключ через якийсь сторонній механізм (наприклад, при налаштуванні).

8

Рисунок А.8 – слайд 8

Введення в експлуатацію

- вимір радіо і фізичного середовища,
- розміщення пристроїв,
- конфігурація їх параметрів,
- прив'язка додатків,
- оптимізація мережі та параметрів пристроїв,
- тестування і підтвердження правильності роботи.



9

Рисунок А.9 – слайд 9

ПОБУДОВА КОМБІНОВАНОЇ МЕРЕЖІ ПРОМИСЛОВОГО ПІДПРИЄМСТВА

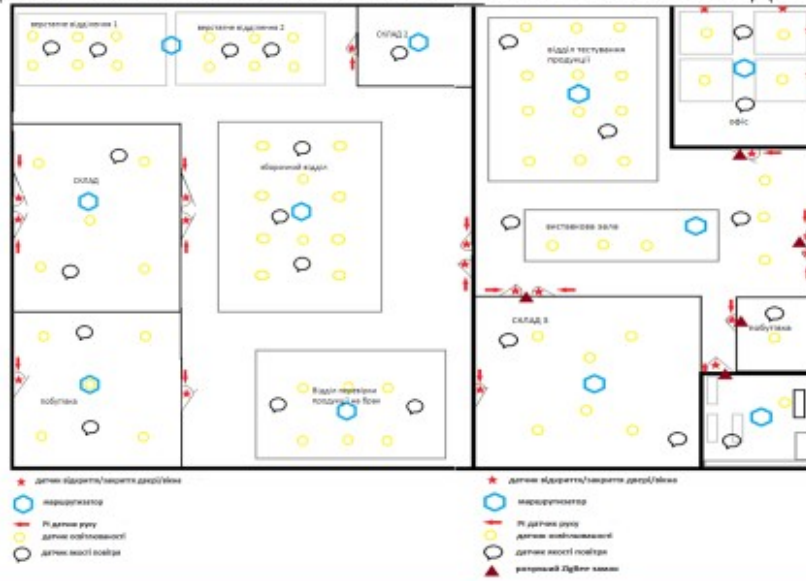
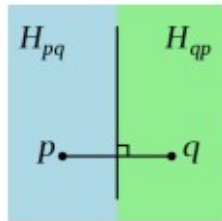


Рисунок А.10 – слайд 10



Діаграма Вороного

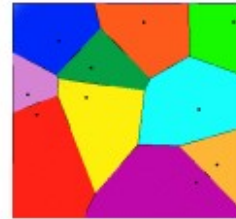
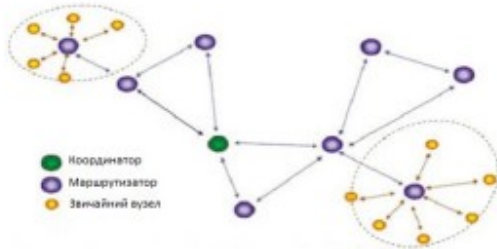
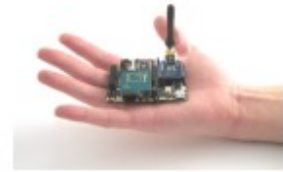


Рисунок А. 11 – слайд 11

ВИСНОВКИ

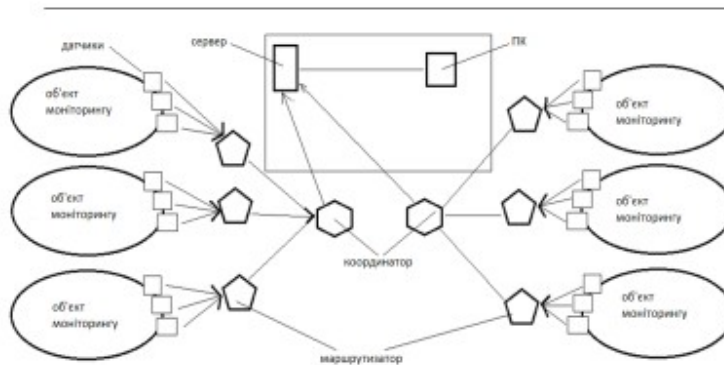
- виконано огляд літератури, обрані комплектуючі вузла зв'язку для промислового об'єкта;
- досліджено існуючі рішення в області БСМ;
- розроблений план впровадження БСМ в підприємства з варіантами установки сенсорів;
- оптимізовані обсяги використання необхідного обладнання.



12

Рисунок А.12 – слайд 12

Вузол зв'язку на промисловому об'єкті



Маршрутизатори збирають дані з датчиків і через шлюз (координатор) передають на сервер для подальшої обробки. Оператор за ПК може спостерігати за показаннями датчиків.

13

Рисунок А.13 – слайд 13

ДОДАТОК Б

Код реалізації побудови алгоритму Вороного шляхом перетину напівплощин

```
// для отримання багатокутників, які напівплощини відсікають від прямокутника
1. Convex2D Rectangle::GetIntersectionalConvex2D(const Point2D& cur_point, const
   Line2D& halfplane) const
2. {
3.   vector<Point2D> convex_points;
4.   Segment2D cur_side;
5.   Point2D intersection_point;
6.   for (int i = 0, sz = vertices_.size(); i < sz; ++i) {
7.     int j = (i + 1) % sz;
8.     cur_side = Segment2D(vertices_[i], vertices_[j]);
9.     intersection_point = halfplane.GetIntersection(cur_side);
10.    if (intersection_point != kInfPoint2D)
11.        convex_points.push_back(intersection_point);
12.    if (halfplane.Sign(cur_point) == halfplane.Sign(vertices_[i]))
13.        convex_points.push_back(vertices_[i]);
14.    }
15.   Convex2D result_polygon(MakeConvexHullJarvis(convex_points));
16.   return result_polygon;
17. }
// визначає номер випадка взаємного розташування ребер
18. NumOfCase EdgesCaseNum(const Segment2D& first_edge, const Segment2D&
   second_edge)
19. {
20.   bool first_looks_at_second = first_edge.LooksAt(second_edge);
21.   bool second_looks_at_first = second_edge.LooksAt(first_edge);
22.   if (first_looks_at_second && second_looks_at_first) {
23.     return NumOfCase::kBothLooks;
24.   } else if (first_looks_at_second) {
25.     return NumOfCase::kFirstLooksAtSecond;
26.   } else if (second_looks_at_first) {
27.     return NumOfCase::kSecondLooksAtFirst;
28.   } else {
```

```

27.     return NumOfCase::kBothNotLooks;
28. }
29. }
// визначає, яке ребро зараз "всередині"
30. WhichEdge WhichEdgeIsInside(const Segment2D& first_edge, const Segment2D&
    second_edge)
31. {
32.     double                first_second_side                =
        Vector2D(second_edge).OrientedCCW(Vector2D(second_edge.a, first_edge.b));
33.     double                second_first_side                =
        Vector2D(first_edge).OrientedCCW(Vector2D(first_edge.a, second_edge.b));
34.     if (first_second_side < 0) {
35.         return WhichEdge::kSecondEdge;
36.     } else if (second_first_side < 0) {
37.         return WhichEdge::kFirstEdge;
38.     } else {
39.         return WhichEdge::Unknown;
40.     }
41. }
// функція Руху
42. WhichEdge MoveOneOfEdges(const Segment2D& first_edge, const Segment2D&
    second_edge, Convex2D& result_polygon)
43. {
44.     WhichEdge now_inside = WhichEdgeIsInside(first_edge, second_edge);
45.     NumOfCase case_num = EdgesCaseNum(first_edge, second_edge);
46.     WhichEdge which_edge_is_moving;
47.     switch (case_num) {
48.         case NumOfCase::kBothLooks: {
49.             if (now_inside == WhichEdge::kFirstEdge) {
50.                 which_edge_is_moving = WhichEdge::kSecondEdge;
51.             } else {
52.                 which_edge_is_moving = WhichEdge::kFirstEdge;
53.             }
54.             break;
55.         }

```

```

56. case NumOfCase::kFirstLooksAtSecond: {
57.     which_edge_is_moving = WhichEdge::kFirstEdge;
58.     break;
59. }
60. case NumOfCase::kSecondLooksAtFirst: {
61.     which_edge_is_moving = WhichEdge::kSecondEdge;
62.     break;
63. }
64. case NumOfCase::kBothNotLooks: {
65.     if (now_inside == WhichEdge::kFirstEdge) {
66.         which_edge_is_moving = WhichEdge::kSecondEdge;
67.     } else {
68.         which_edge_is_moving = WhichEdge::kFirstEdge;
69.     }
70.     break;
71. }
72. }
73. if (result_polygon.Size() != 0 && (case_num == NumOfCase::kFirstLooksAtSecond
    || case_num == NumOfCase::kSecondLooksAtFirst)) {
74.     Point2D vertex_to_add;
75.     if (now_inside == WhichEdge::kFirstEdge) {
76.         vertex_to_add = first_edge.b;
77.     } else if (now_inside == WhichEdge::kSecondEdge) {
78.         vertex_to_add = second_edge.b;
79.     } else {
80.         if (case_num == NumOfCase::kFirstLooksAtSecond)
81.             vertex_to_add = first_edge.b; // ?!
82.         else
83.             vertex_to_add = second_edge.b;
84.     }
85.     if (vertex_to_add != result_polygon.GetCurVertex())
86.         result_polygon.AddVertex(vertex_to_add);
87. }
88. return which_edge_is_moving;
89. }

```

// перетин опуклих багатокутників (посилання не константне тільки тому, що в другому багатокутнику рухається вікно)

```

90. Convex2D Convex2D::GetIntersectionalConvex(Convex2D& second_polygon)
91. {
92. Convex2D result_polygon;
93. size_t max_iter = 2 * (this->Size() + second_polygon.Size());
94. Segment2D cur_fp_edge; // current first polygon edge
95. Segment2D cur_sp_edge; // current second polygon edge
96. Point2D intersection_point;
97. bool no_intersection = true;
98. WhichEdge moving_edge = WhichEdge::Unknown;
99. for (size_t i = 0; i < max_iter; ++i) {
100.     cur_fp_edge = this->GetCurEdge();
101.     cur_sp_edge = second_polygon.GetCurEdge();
102.     intersection_point = cur_fp_edge.GetIntersection(cur_sp_edge);
103.     if (intersection_point != kInfPoint2D) {
104.         if (result_polygon.Size() == 0) {
105.             no_intersection = false;
106.             result_polygon.AddVertex(intersection_point);
107.         } else if (intersection_point != result_polygon.GetCurVertex()) {
108.             if (intersection_point == result_polygon.vertices_[0]) {
109.                 break; // we already found the intersection polygon
110.             } else {
111.                 result_polygon.AddVertex(intersection_point);
112.             }
113.         }
114.     }
115.     moving_edge = MoveOneOfEdges(cur_fp_edge, cur_sp_edge,
        result_polygon);
116.     if (moving_edge == WhichEdge::kFirstEdge) {
117.         this->MoveCurVertex();
118.     } else {
119.         second_polygon.MoveCurVertex();
120.     }
121. }

```

```
122. if (no_intersection == true) {
123.     if (second_polygon.Contains(this->GetCurVertex())) {
124.         result_polygon = *this;
125.     } else if (this->Contains(second_polygon.GetCurVertex())) {
126.         result_polygon = second_polygon;
127.     }
128. }
129. return result_polygon;
130. }
```