

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Радіоелектроніки і телекомунікацій

(повне найменування факультету)

Мікро- та наноелектроніки

(повне найменування кафедри)

Пояснювальна записка

до дипломного проекту (роботи)

магістра

(ступінь вищої освіти)

на тему: «Розробка системи безпеки жилого приміщення на базі мікроконтролера»

Виконав: студент(ка) II курсу, групи РТ – 311м
Спеціальності 153 - Мікро- та наносистемна техніка
(код і найменування спеціальності)

Освітня програма (спеціалізація)

«Мікро- та наноелектронні прилади і пристрої»

Олехнович Д. О.

Керівник Рева В.І.

(прізвище та ініціали)

Рецензент Манукян О.А.

(прізвище та ініціали)

2022

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
(повне найменування закладу вищої освіти)

Факультет Радіoeлектроніки і телекомунікацій

Кафедра мікро-та наноелектроніки

Ступінь вищої освіти магістр

Спеціальність 153 Мікро- та наносистемна техніка

(код і найменування)

Освітня програма (спеціалізація) «Мікро- та наноелектронні прилади і пристрої»

(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри мікро- та
наноелектроніки

Г. В. Сніжної
« » 2022 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА
Олехнович Даяни Олександрівни

(прізвище, ім'я, по батькові)

1. Тема проєкту (роботи) «Розробка системи безпеки жилого приміщення на базі мікроконтролера»

Завідувач проєкту (роботи)

доц., канд.фіз-мат.наук Рева В.І.,

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

Затверджені наказом закладу вищої освіти від «21» листопада 2022 року №393

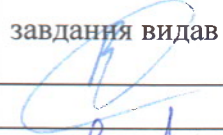
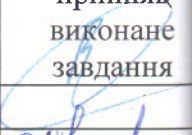
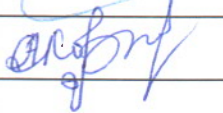
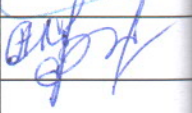
2. Строк подання студентом проєкту (роботи) _____

3. Вихідні дані до проєкту (роботи): Arduino Mega 2560, ESP8266 - Wi-Fi модуль, Клавiатурний модуль для Arduino с 12 кнопками на гнучкій основі, ультразвуковий датчик відстані HC-SR04

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)
Розробка системи безпеки жилого приміщення на базі мікроконтролера. Розділ 1
Поняття розумний будинок. Розділ 2 Структурні особливості системи. Розділ 3
Проектування та реалізація системи.

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

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-3	Рева В.І. доцент каф. МіНЕ		
Нормо-контроль	Коротун А. В., к.ф.-м.н., доцент каф. МіНЕ		

7. Дата видачі завдання «24» жовтень 2022 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Збір матеріалу	26.09 – 09.10	виконано
2	Аналіз зібраного матеріалу	10.10 – 20.10	виконано
3	Виконання 1 розділу	21.10 – 30.10	виконано
4	Виконання 2 розділу	01.11-12.11	виконано
5	Виконання 3 розділу	13.11-25.11	виконано
6	Написання програми системи захисту	15.11-01.12	виконано
7	Перевірка роботи консультантами	02.12-07.12	виконано
8	Захист роботи у ЕК	19.12	виконано

Студент

(підпис)

(прізвище та ініціали)

Керівник проекту (роботи)

(підпис)

(прізвище та ініціали)

АНОТАЦІЯ

ПЗ: 71 с., 29 рис., 6 табл., 2 додатків, 30 джерел.

Мета роботи: Розробка системи безпеки жилого приміщення на базі мікроконтролера.

Об'єкт та предмет дослідження: об'єктом дослідження виступають система безпеки жилого приміщення, яка створена на базі мікроконтролера.

Методи дослідження: науково-аналітичний.

Результати: результатом дослідження є програма розроблена на мовах програмування C++, Python.

Рекомендації щодо впровадження: проект може бути рекомендований для розташування на дверних, віконних та інших прорізів. Де може відстежуватися відчинення чи зачинення прорізів.

Практична цінність: проект може бути використаний для систем забезпечення безпеки житлового приміщення.

СИСТЕМИ БЕЗПЕКИ, РОЗУМНИЙ ДІМ, МІКРОКОНТРОЛЕР,
ПРОГРАМУВАННЯ, C++, ARDUINO, ARDUINOMEGA2560.

ЗМІСТ

ВСТУП.....	6
РОЗДІЛ 1 ПОНЯТТЯ РОЗУМНИЙ БУДИНОК.....	11
1.1 Визначення розумного будинку	11
1.2 Концепція розумного будинку.....	13
1.3 Можливості розумного будинку.....	14
1.4 Технології розумного будинку	17
1.5 Розвинуті будинки в історії.....	19
1.6 Економічні аспекти	21
1.7 Інтегрована система управління інформацією	21
1.8 Віддалене управління розумним будинком.....	23
РОЗДІЛ 2 ООБЛИВОСТІ СТРУКТУРИ СИСТЕМИ	25
2.1 Складові системи розумного будинку	25
2.1.1 Клімат контроль	25
2.1.2 Розумне освітлення	25
2.1.3 Система безпеки.....	26
2.1.4 Нижній рівень системи.....	26
2.1.5 Wi-Fi ESP8266	28
2.2 Середній рівень системи.....	35
РОЗДІЛ 3 ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ СИСТЕМИ	42
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ	57
ДОДАТОК А	60
ДОДАТОК Б.....	71

ВСТУП

Автоматизований або «розумний» дім представляє собою живе, максимально сучасне приміщення. Усе в цьому приміщенні направлене на кращу організацію та якість життя людини. Це досягається за рахунок обладнання, що є дуже технологічним та максимально автоматизованим. Під цим терміном «розумність», ховається ціла система, основним завданням якої є комфорт, економія та що найважливіше безпека людини.

Але прямо цей термін розуміти не можна, бо те що зазвичай є «Smart house», не може самостійно використовувати інтелектуальне управління. Це набір алгоритмів, які задаються ще на стадії програмування. Файли, що зберігають у собі ці налаштування так звані "конфігурації". Але, зазвичай у таких систем, є ще режим, у якому саме користувач, може вирішувати що вмикати або вимикати і у якому порядку. Цей спосіб називають дистанційним керуванням. Але з часом, новими відкриттями, системи штучного інтелекту розвиваються, і таким чином збільшується радіус їх впливу. Бо століття назад, навіть простий датчик руку, який призводить до ввімкнення освітлення, був чимось шокуючим чи магічним. Але в нас час, такі речі можна легко зустріти на вулиці. І зараз, повний контроль будинку, це не тільки можливо, але і має свій попит на світовому ринку.

Найпоширенішою є ідея, що мисляча будівля, зможе сама приймати рішення, у мінливих та непостійних обставинах. Метою цієї роботи є створення проекту, що забезпечить компроміс між безпекою і таким бажаним комфортом. Будь який з подібних винаходів містить у собі різні підсистеми, що відповідають за відокремленні функції, які в сукупності і є розумною системою. І чим більше таких маленьких

кластерів, тим важче ними керувати, бо це потребує більше ресурсів. Ідея "smart house", не настільки нова, як здається. Вона отримала першу конкретику у семидесятих роках минулого століття, коли до неї додали ще одну вимогу - ефективно використовувати простір. Популярність вона здобула останнє десятиліття. Під цією ідеєю, мається на увазі система, що може розрізняти види ситуацій, які виникають у будинку, і вміти на них правильно реагувати. У такому проекті, усе має спільну систему управління. Це невідоміша і найпрогресивніша концепція, при якій людина взаємодіє з штучним інтелектом. Така концепція обов'язково має такі доктрини:

1. Вбудована система керування житловим приміщенням це системи у яких є ресурси і можливості, щоб контролювати і задовольнити всі потреби підсистем, що існують у приміщенні.
2. Заміна усіх робочих на території жилого приміщення.
3. Можливість як найшвидше передати людини повний контроль будь якою підсистемою чи повністю переключитися у дистанційний режим.
4. Можливість правильної роботи будь якої підсистеми без зв'язку з керуючою програмою.
5. Економічність в обслуговуванні 6. Присутність системи комунікації.

Сенс таких проектів у тому, щоб людина декількома простими маніпуляціями, зі зручним їй інтерфейсом могла повністю управляти тим, де увімкняться музика, де потрібно закрити штори, ставні або жалюзі, чи вимкнути світло, змінити його яскравість або задати йому новий алгоритм. Така система завжди слідує за усіма параметрами, щоб підтримувати все згідно даним, у файлах конфігурацій. І найцікавіше, що таку систему модно "навчити" економити ваші кошти, бо їй можна задати

не одне таке налаштування, а декілька. Наприклад для вечірки, сплячого режиму, режиму економії, або наприклад ситуації коли довго нікого не буде вдома. В користувача є можливість, щоб такі зміни виконувалися по якійсь команді, чи наприклад жесті, або по часу.

Також в таких системах є відчутне поліпшення рівня безпеки. Багато людей встановлюють такі системи саме по цій причині. Розумний дім розкаже і покаже вам усе, що трапилось за ті години, або місяці коли вас не було вдома. І також, дистанційно теж можна спостерігати за своїм майном. Така система розкриє які люди до вас приходили, скільки часу вони провели біля дому або у середні, чи може просто підозрілу особу. Система безпомилково збереже у своїй бездоганній пам'яті будь який рук, вираз або рису обличчя. За допомогою новітніх методів зв'язку, дуже легко контролювати усе в своєму будинку навіть на іншій половині земної кулі. Для цього можна використати всесвітню павутину. В деяких випадках навіть вона не потрібна. Деякі системи з легкістю відрізняють голос свого господаря. Тобто достатньо просто подзвонити наприклад на домашній телефон, щоб передати свою команду. Такі методи дозволяють економічно підрозділяти ресурси, і скорочують витрати. А також така практика дозволяє використовувати дружній до людини інтерфейс, щоб було простіше керувати.

У середньому біля тридцяти окремих підсистем можуть бути зібрані в одному розумному домі, що дозволяє оптимізувати все найкращім чином. Кожна з цих тридцяти підсистем повинна проводити не прості математичні і логічні виміри та розрахунки. Вже сьогодні такі системи мають функціонал близький до фантастичних романів. Досягнуто майже стовідсоткового рівня автоматизації, навіть домашня техніка свої функції виконує без участі людини. Такі проекти суттєво спрощують життя, забираючи на себе усі побутові проблеми, дозволяючи своєму

господарю відпочивати, або присвятити час чомусь дійсно важливому. У будь – якого розумного будинку обов’язково присутня ціла система автоматизації усіх процесів що відбуваються всередині. Вона представляє собою добірку програм що з легкістю візьмуть на себе будь які турботи. А також через цю систему можна спостерігати як за окремими її частинами, так і за усім в цілому. Теж саме стосується і управління. Це потрібно, щоб обслуговування було як найкращім, і максимально індивідуальним, тобто комфортним кожному окремо. Щоб цього досягнути, у цьому проекті будуть знайдені вирішення для таких завдань:

1. Дослідження сфери;
2. Створена система безпеки для «Smart house»;
3. Був створений максимально дружній до людини інтерфейс, через який може відбуватися керування і моніторинг системи безпеки.
4. Був проведений тестовий запуск проекту за допомогою програмного емулятора. Ця робота містить вступ, три розділи, висновки, список використаних джерел, та двох додатків.

У першому розділі здійснюється аналіз предметної області. Наведено загальні відомості про технологію «Розумний будинок» і управління в ній. Більш детально розглянуто специфіку управління за профілями і його недоліки. У другому розділі описуються компоненти бази розумного будинку та взаємозв’язок між ними. У третьому розділі описується створення інтелектуальної системи керування, на основі розробленої методики для технології «Розумний будинок». На основі результатів, показана ефективність застосування даної методики. Дано визначення розумному будинку. Визначення, сформульоване в Вашингтоні в 70-х роках: «Будівля, що забезпечує продуктивне й ефективне використання робочого простору ...», абсолютно нічого не пояснює. І справді, багато з того, що представлено на ринку, і продається під назвою «розумного будинку» автоматикою, і системами

управління не є. Часто ми стикаємося з дуже дорогими (це не означає, що вони хороші) «розумними будинками», які робляться на потіху багатю і не дуже освіченою публіці. Працездатність і надійність цих конструкцій невитримують ніякої критики. Інша крайність - пластмасові «китайські» коробочки з дешевим процесором для виконання якоїсь однієї задачі. Тому спочатку дамо визначення, що розуміємо під назвою «Будинок розумний». Це система автоматики, яка забезпечує комфортні умови існування в будинку, незалежно від зовнішніх факторів. Оскільки ми живемо в такий час, коли екологія (особливо великих міст) жахлива. Насправді справа з автоматикою з'явилася не випадково, а мала логічне пояснення. У різних кліматичних зонах нашої планети життя різне. У полярних і тропічних регіонах умови життя різні. Однак у житла все ж є певна спільність – кілька стін і даху, які ізолюють людину від зовнішнього впливу. Не дарма часто говорять про «дах над головою». Такі будинки часто оснащуються «підками» для приготування їжі та опалення. До цих пір люди так живуть. Лише наприкінці 1970-х років з'явилися перші мікропроцесори, придатні для автоматизації повсякденного життя. Сучасна автоматизація – продукт 21 століття. Наші домівки повні пристроїв, які використовуються і можуть бути «адаптовані» до потреб дому. Почали з'являтися концепції та технології, які вже можна назвати «розумними будинками». Електроніка значно дешевша, і ми можемо виробляти автоматизацію для дому. Те, що двадцять років тому коштувало десятки тисяч доларів, сьогодні коштує сто доларів. Тут можна говорити про прогрес. Можна говорити про автономне спілкування індивідуальних будинків. Сьогодні такий спосіб будівництва більш ефективний, дешевий і швидкий. Зараз будують швидше, ніж погоджують.

РОЗДІЛ 1 ПОНЯТТЯ РОЗУМНИЙ БУДИНОК

1.1 Визначення розумного будинку

Будь-яка будівля, будь-яке адміністративне, виробниче або житлове приміщення складається з певного набору підсистем, що відповідають за виконання певних функцій, які вирішують різноманітні завдання під час експлуатації цієї будівлі. У міру того як ці підсистеми ускладнюються і виконують все більше функцій, управління ними стає все важче. Крім того, стрімко зростають витрати на утримання обслуговуючого персоналу, ремонт і обслуговування цих підсистем. Ці проблеми виникли вперше під час функціонування великих адміністративно-виробничих комплексів. Цей тип сучасної архітектури є уособленням міста. У ньому працюють практично всі служби, які раніше були невід'ємною частиною економіки міста. У таких будівлях часто є адміністративні офіси або адміністратори, які використовують та обслуговують систему майже 24 години на добу. Хоча є чимало засобів автоматики, які самі справляються з покладеними на них завданнями, такими, як опалення, вентиляція, підтримка мікроклімату, освітлення, пожежна сигналізація, димо-фільтрація, контроль входу / виходу і т.п., але управління і обслуговування всіх цих систем вимагає наявності адміністратора персоналу. Його обов'язком є контроль роботи цих підсистем і вжиття заходів у разі виходу їх з ладу. Але є ситуації, коли навіть дії кваліфікованого персоналу можуть виявитися неефективними. Це випадки виникнення загрози будівлі і знаходяться в ньому людям, що мають глобальний характер - пожежа, землетрус і інші стихійні лиха, терористичні атаки. Тут потрібно вживати екстраординарних заходів в лічені частки секунди. Реакція і коректність дій людей в критичній ситуації може виявитися недостатньою. Традиційні системи забезпечення різних аспектів життєдіяльності в минулому

проектувалися як автономні. Такі системи, що створювалися окремо для кожної функції і об'єднані для довільної частини будівлі. У будівлях встановлювалися системи тільки з тими можливостями і з тим ступенем складності, які були необхідні на поточний момент побудови будівлі.[1] Подальше розширення і модернізація даних систем були складними і дорогими завданнями через безліч різних чинників. Витрати на експлуатацію такої системи складаються з витрат на експлуатацію кожної автономної системи в окремо, вартості навчання персоналу. Вартість експлуатації цих систем висока - в силу їх автономності кожна з них підтримується окремо.

Вартість навчання персоналу настільки ж висока, оскільки оператори повинні бути ознайомлені з експлуатацією кожної автономної системи. Розумний будинок (англ. Smart House) - житловий будинок сучасного типу, організований для комфорту проживання людей за допомогою сучасних високотехнологічних пристроїв. Саме поняття «розумний будинок» було сформульовано Інститутом інтелектуальної будівлі у Вашингтоні (округ Колумбія) в 1970-х роках: Будівля, що забезпечує продуктивне й ефективне використання робочого простору. Принцип «Системи інтелектуального управління будівлею» передбачає абсолютно новий підхід в організації життєзабезпечення будівлі, в якому за рахунок комплексу програмно-апаратних засобів значно зростає ефективність функціонування і надійність управління всіх систем і виконавчих пристроїв будівлі.[5]

Під «розумним будинком» слід розуміти систему, яка повинна вміти розпізнавати конкретні ситуації, що відбуваються в будівлі, і відповідним чином на них реагувати: одна з систем може управляти поведінкою інших по заздалегідь виробленим алгоритмам. Основною особливістю інтелектуальної будівлі є об'єднання окремих підсистем в єдиний керований комплекс. Важливою особливістю і властивістю "Розумного будинку" відрізняє його від інших способів організації

життєвого простору є те, що це найбільш прогресивна концепція взаємодії людини з житловим простором, коли людина однією командою задає бажану обстановку, а вже автоматика відповідно до зовнішніми і внутрішніми умовами задає і відстежує режими роботи всіх інженерних систем і електроприладів. У цьому випадку виключається необхідність користуватися кількома пультами при перегляді ТБ, десятками вимикачів при управлінні освітленням, окремими блоками при управлінні вентиляційними і опалювальними системами, системами відеоспостереження та сигналізації, воротами і іншим. У будинку, обладнаному системою "Розумний дім" досить одним натисканням на настінної клавіші (або пульті ДУ, сенсорної панелі і т. Д.). Будинок сам налаштує роботу всіх систем відповідно побажанням, часом доби, становищем власника в будинку, погодою, зовнішньої освітленістю і т.д.

1.2 Концепція розумного будинку

Основні положення терміну інтелектуальна будівля. «Розумна дім» У цьому підході розуміється, що є певна ситуація і на неї треба реагувати. Однак, згідно з дослівним перекладом з англійської, IZ можна трактувати як «розумно побудований» [3]. Це означає, що будівля має бути спроектована таким чином, щоб усі служби могли бути інтегровані одна з одною з мінімальними витратами (фінансовими, часовими та зусильними), а їх обслуговування було організовано оптимальним чином.[7]

Концепція інтелектуальної будівлі включає наступні положення: Створення інтегрованої системи управління будівлею - системи зі здатністю забезпечити інтегровану роботу всіх інженерних систем

будівлі: освітлення, опалення, вентиляції, кондиціонування, водопостачання, доступу, контроль та багато інших. Ліквідація всього обслуговуючого персоналу будівлі та передача функцій управління та прийняття рішень підсистемам інтегрованої системи управління будівлею. У ці підсистеми вбудований «інтелект» будівлі — як вона буде реагувати на зміни параметрів датчиків системи та інші події, наприклад, надзвичайні ситуації.

Реалізація механізму миттєвого відключення та передачі, при необхідності, особі, яка керує будь-якою підсистемою розумної будівлі. При цьому людина повинна мати зручний і рівноправний доступ до управління і відображення всіх підсистем і частин «розумного будинку». Забезпечення коректної роботи окремих підсистем у разі відмови загальної системи управління або інших частин системи. Мінімізація витрат на технічне обслуговування та модернізацію будівельних систем, що має забезпечуватися застосуванням єдиних стандартів при побудові підсистем, автоматичного налаштування та виявлення нових пристроїв і модулів при їх додаванні в систему. Наявність у будівлі посиленої комунікаційної середовища для підключення до неї обладнання та системних модулів. При цьому передбачена можливість використання в системі управління як середовища зв'язку різних типів фізичних каналів: слабкострумів ліній, електричних ліній, радіоканалу.[9]

1.3 Можливості розумного будинку

Розумна будівля має багато переваг. Система керування дозволяє власникам створювати скільки завгодно складних і інтелектуальних робочих процедур, оскільки всі потужні системи можуть працювати злагоджено та разом. Тому впровадження багатьох ресурсозберігаючих процедур:

- 1) контроль доступу та безпека;
- 2) облік і контроль майже всіх параметрів системи та швидке реагування на їх критичні зміни, при цьому відповідь полягає в комплексному та миттєвому дистанційному контролі та управлінні будівлею, оскільки всі інформаційні та керуючі канали зв'язку в такій системі цифрові.[4]

Всього одним дотиком ви можете перетворити порожній будинок в затишний будинок для гостей: вмикається освітлення, створюється приємний мікроклімат, опускаються штори і наповнюється ванна.[16]

Обід можна розігріти навіть не встаючи - просто хтось повинен заздалегідь поставити його на плиту. За допомогою сенсорних панелей можна керувати «домашнім кінотеатром», а також аудіо- та відеотехнікою. За допомогою спеціальних світлорегуляторів можна змінювати не тільки яскравість, з якою світиться лампа після включення, але і час, необхідний для досягнення цієї яскравості. Функція постійного регулювання освітлення, призначена спеціально для офісних приміщень, дозволяє підтримувати задане освітлення робочої поверхні, незалежно від того, світить сонце або небо вкрите хмарами. Автоматична активація зовнішнього оповіщення в залежності від часу доби та наявності людей не тільки забезпечить додатковий комфорт, але й відлякає непроханих гостей. Система постійно вимірює температуру окремо в кожній кімнаті та підтримує її на заданому рівні, безпосередньо керуючи вентилями радіаторів або заслінками кондиціонера та автоматично вмикаючи або вимикаючи вентиляцію, якщо це необхідно. Це допомагає економити щодня завдяки різним режимам роботи системи: комфортний режим, нічний режим, режим «нікого немає вдома». Зміна режимів відбувається

за планом або за командою. Для кожного з режимів потрібно лише один раз встановити температуру на сенсорному дисплеї в кімнаті. Якщо вікно кімнати відкрито для провітрювання (вказує на контакт рами), система опалення або кондиціювання повітря автоматично вимикається для економії енергії. Влітку їх ламелі автоматично повертаються під кутом, запобігаючи проникненню надмірного сонячного світла всередину, не зменшуючи його. Таким чином, вони запобігають нагріванню приміщення та допомагають економити споживану кондиціонером електроенергію. Розумний будинок розповідає про події, що відбулися під час відсутності господаря: хто і коли приходив, як довго були в будинку, по будинку довго блукали підозрілі люди. Їхні обличчя та рухи глибоко врзалися в його пам'ять. Непрохані гості підносять неприємні сюрпризи у вигляді сліпучих вогнів і сирен. Крім того, вам повідомлять по телефону, що вони зайшли в приміщення, і повідомте охорону. При надзвичайній ситуації (наприклад, витік води) необхідно не тільки повідомити відповідну службу, а й вжити необхідних заходів для ліквідації аварії (припинення подачі води). Коли ви відсутні, будинок може імітувати типовий спосіб життя власника, включаючи освітлення та вечірню музику для створення ефекту присутності. Встановивши програму vision на домашній комп'ютер і скориставшись модемом, можна відключити ноутбук від світла, який хтось залишив, щоб впустити гостей в будинок. Він забезпечує надійний зв'язок навіть без комп'ютера під рукою: просто наберіть номер домашнього телефону. Ви входите в зону управління будинком із встановленим маркером. Кожен системний пристрій має свій серійний номер. Викликаючи його, ви можете віддати кілька команд: увімкнути обігрів, розігріти сауну, наповнити ванну. Сучасна технологія 4G дозволяє використовувати мобільні телефони як інструменти для управління системою, контролю та діагностики. Не встаючи з дивана, ви можете легко змінювати яскравість внутрішнього освітлення, закривати

жалюзі та виставляти бажану температуру за допомогою інфрачервоного пульта дистанційного керування.

1.4 Технології розумного будинку

Високотехнологічний «розумний» будинок Не за горами той день, коли речі навколо нас навчатися мислити і стануть набагато кориснішими для суспільства. Н. Вінер. 1940 Smart House - це інтелектуальна система управління, яка забезпечує злагоджену та автоматичну роботу всіх технічних мереж у будинку. Така система розумно розподіляє ресурси, знижує експлуатаційні витрати та забезпечує чіткий інтерфейс контролю та управління. В даний час технічне проектування елітних заміських будинків за авторськими проектами здійснюється дизайнерами та архітекторами у співпраці з великими будівельними компаніями. Він враховує роботу всіх інженерних комунікацій і об'єднує це з індивідуальним проектуванням в єдиний комплекс: водопостачання, каналізація, мікроклімат. У проект розумного будинку може бути включено до сорока окремих інженерних систем, кожна з яких потребує комплексу інженерні розрахунки. На перший погляд, сучасний розумний будинок оснащений фантастичними функціями. У розумному будинку все автоматизовано, навіть побутова техніка в ньому виконує більшість завдань самостійно. Розумна будівля живе за готовим сценарієм, який робить життя людини комфортним і мобільним, вільним від суєти і домашніх турбот. Це не вигадка - це реальність. Проекти розумних будинків мають різноманітні функції, які охоплюють усі сфери життя. Будь-який «розумний» будинок відрізняється наявністю системи автоматизації будівлі. Це комплекс програмного забезпечення для управління пристроєм - моніторингу,

оптимізації та управління. Метою автоматизації будівель є ефективне та безпечне обслуговування всіх пристроїв у житловому будинку чи іншій будівлі. Управління здійснюється за допомогою системи, яка активується при виконанні умов, заданих програмою. Автоматизація будівель стає можливою завдяки сучасним пристроям, станціям і модулям управління, які можуть контролювати (блокувати, оптимізувати) роботу всіх пристроїв. Процес автоматизації дозволяє людині забути про складність управління системами зв'язку. Особливістю автоматизації будівель є можливість одночасного і точного обліку всіх найважливіших факторів: контроль споживання води, електроенергії, температури, джерел безперебійного живлення та ін. Монтаж інженерних мереж, що забезпечують безперебійну роботу котеджних систем в автономному режимі, , що також економить ваші гроші [5]. Створення систем зв'язку. Основним завданням фахівців, які розробляють автоматизовані рішення для житлових комплексів, є створення інженерних мереж та управління ними. Впровадження інженерних комунікацій відбувається поетапно: спочатку створюється опалення, кондиціонування та вентиляція, потім «розумне» освітлення та мультимедійні технології. Використовуючи ці мережі, можна економити від 10 до 40% електроенергії. Управління технічними комунікаціями здійснюється комп'ютером за допомогою спеціальних програм - контролерів. Координація роботи здійснюється через сенсорні панелі, через Інтернет або мобільний телефон, а також за допомогою одно-, дво- або N-клавійного вимикача. Останні здатні контролювати всю систему будівлі. Всі вбудовані автоматичні вимикачі відповідають вимогам.[2] Окрім централізованого управління всіма комунікаціями будівлі, пропонується встановити систему спостереження, яка завдяки спеціальним приладам дозволить контролювати інженерні мережі: датчики освітлення, руху та температури. Інформація про стан комунікацій дозволяє своєчасно проводити профілактику та заміну

несправних деталей. Індивідуальне проектування котеджів - це цілий комплекс робіт, починаючи з безпосереднього зведення конструкції і закінчуючи створенням системи. «розумний» чохол. Процес розробки проекту повинен відповідати технічним вимогам, а також нормам єдиної системи проектної документації. Проектування замських будинків «Інтелектуальні функції» включають[19]:

- 1) розробка проекту системи управління освітленням;
- 2) створення концепції мультимедійних систем (домашні кінотеатри, музичні центри);
- 3) установка кліматичного контролю;
- 4) керування вентиляцією, кондиціонуванням, опаленням і водопостачанням;
- 5) установка системи силової електрики;
- 6) інтеграція системи каналізації;
- 7) установка системи безпеки, диспетчеризації;
- 8) підключення будь-яких систем за бажанням замовника (система ландшафтного поливу, система анти зледеніння покрівлі, сходинок, доріжок і водостоків).

Індивідуальне проектування котеджів і установку систем безпеки, в яких потребують замські будинки, - охоронної та пожежної сигналізації, пожежогасіння, контролю доступу, виклику служби охорони і т.д. Проектування «розумного» житла супроводжується гарантійним і післягарантійним обслуговуванням встановленої системи.[6]

1.5 Розвинуті будинки в історії

Почнемо з визначення того, що ми розуміємо під цим терміном. Визначення, сформульоване у Вашингтоні в 1970-х роках: «Будівля, яка продуктивно та ефективно використовує робочий простір...» не пояснює абсолютно нічого. Дійсно, багато з того, що представлено на ринку і продається під назвою «розумний дім», — це домашня автоматизація, а системи управління — ні. Ми часто зустрічаємо дуже дорогі «розумні будинки» (це не означає, що вони хороші), які створені для задоволення заможної та неосвіченої публіки. Працездатність і надійність цих конструкцій не витримують ніякої критики. Інша крайність — пластик

Тому давайте спочатку визначимося, що ми розуміємо під назвою «Розумний будинок». Це система автоматизації, яка забезпечує комфортні (задані) умови проживання в будинку, незалежно від зовнішніх (тривожних) впливів. Оскільки ми живемо в час, коли екологія (особливо у великих містах) нікуди не годиться, а аварії на комунікаціях стають нормою, тематика «smart house» набирає популярності.

Домашня автоматизація з'явилася з першими мікропроцесорами лише в кінці 70-х років 20 століття. Автоматизація – продукт 21 століття. Наш будинок повний цих пристроїв, які використовуються, і їх можна "адаптувати" до потреб сім'ї. Сьогодні народжуються концепції та технології, які вже можна охарактеризувати як «розумні будинки». Електроніка набагато дешевша. У будинку загальне навантаження може досягати двадцяти кіловат, або 20 кВт/год при включенні теплих підлог. 480 на день При такому підході автоматизація окупається протягом місяця. Інше – це комфорт, простота управління та інтуїтивно зрозумілий інтерфейс. Якщо використовувати аналогію, то зараз ми знаходимося на рівні технології «розумного будинку» 1970-х років, коли формувалися основи персональних комп'ютерів.

1.6 Економічні аспекти

У всіх розвинутих країнах з ринковою економікою доходи і витрати платників податків є прозорими, платниками податків є практично всі громадяни. Чіткий контроль над доходами та витратами дозволяє ефективно працювати таким установам, як іпотека (видача кредитів на будівництво та купівлю житла). Це призводить до того, що більшість працездатного населення може дозволити собі купити житло в кредит. Той, хто взяв кредит на покупку житла, буде «жити в борг»: лівова частка його доходу піде на погашення кредиту. Без страхування ризиків у цьому випадку неможливі правовідносини в цих економічних процесах. Тому в багатьох країнах вже існує практика обов'язкового страхування житла. Цілком природно, що страхові компанії зацікавлені в якості та надійності застрахованого житла. «Розумна будівля» здатна надати експертам цей «чорний ящик», зі змісту якого будуть зроблені висновки. У результаті розробка та впровадження ІВ у країнах з розвиненою ринковою економікою зараз прогресує швидкими темпами. Крім того, майбутні власники таких будівель зацікавлені в їх інтелекті, адже він, безсумнівно, приносить своєму власнику значну економію за рахунок точного обліку та контролю всіх систем будівлі, а також раціонального використання таких ресурсів, як електроенергія, вода, опалення.[5]

1.7 Інтегрована система управління інформацією

Ці системи виникають через необхідність підтримки гетерогенних обчислювальних середовищ, де кілька платформ підтримуватимуться на різних рівнях: різні комп'ютери (мейнфрейми, сервери, робочі станції та персональні комп'ютери), різні методи, обладнання, виробники та різний рівень розумності тощо. Спираючись на надані послуги нижнього рівня, було створено остаточний рівень, який об'єднує всі аспекти управління інфраструктурою будівлі в систему, яка виконує різні функції:

- 2) управління параметрами середовища;
- 3) контроль доступу в будівлю;
- 4) сигналізація злочину;
- 5) управління ліфтами;
- 6) телевізійне спостереження;
- 7) реєстрація часу перебування;
- 8) управління освітленням;
- 9) контроль використання електричної енергії;
- 10) опалення, вентиляція, підтримка мікроклімату;

Крім виконання цільових функцій на неї покладаються функції керування інформаційною інфраструктурою:

- 1) Контроль доступу до інформації та управління безпекою;
- 2) Управління робочим навантаженням;
- 3) Контроль продуктивності;
- 4) Управління подіями;
- 5) Автоматизоване управління зберіганням даних
- 6) Управління проблемами;
- 7) Управління транспортом даних;
- 8) Керування розсилкою звітів;
- 9) Управління Web-серверами;
- 10) Управління мережею.

Остання функція - управління мережею - включає в себе автоматичне розпізнавання об'єктів управління і топології мережі, підвищення рівня контролю мережевого обладнання, наявність засобів розробки для управління нестандартним мережним устаткуванням, інтеграція з уже існуючими в будівлі системами управління мережею і наявність засобів ведення політики управління мережею.

1.8 Віддалене управління розумним будинком

Люди можуть керувати інтелектуальним будинком за допомогою засобів керування що розташовані в середині будівлі або біля неї, але в тому самому комунікаційному середовищі, що й усі інші компоненти системи. Однак у багатьох випадках дистанційне керування підсистемами розумного дому є необхідним або бажаним. Перелічимо основні з них: Людина хоче мати можливість стежити за станом свого будинку під час тривалої відсутності та при необхідності керувати деякими системами. На випадок непередбачуваних ситуацій, збоїв модулів і перед оповіщенням підсистем. і. Концепція розумного будинку дає можливість створити підсистему дистанційного управління в рамках інтегрованої системи управління. Цей метод дозволяє отримувати інформацію про події та стан деяких параметрів і дистанційно передавати команди управління всією системою або її окремими компонентами. Це найпримітивніша модель штучного інтелекту, про яку наголошують у ЗМІ. Де «розумний дім» — диво техніки, яке обіцяє багато зручностей. Об'єднання самотійних об'єктів в єдиний комплекс шляхом підключення розумного дому до мережі дозволить здійснювати моніторинг систем життєзабезпечення, енергозбереження, безпеки тощо. Вирішувати будь-які питання, пов'язані з цими системами та надавати

різноманітні послуги через мережу, стане простіше та ефективніше. Найвища форма розумного дому: використання стандартних цифрових протоколів для обміну інформацією всередині та поза домом. Всі компоненти системи об'єднані в одній точці контакту.

Цей розділ проливає світло на концепцію «розумного дому» та її історичну появу. Системам «розумного будинку», які ми бачимо сьогодні не більше десятиріччя. Спочатку проводами здійснили зв'язок між групами управління підсистемами, їх датчиками і керуючим комп'ютером «розумного будинку». Потім розвиток Інтернету призвів до бездротового керування. Системами розумного будинку тепер можна керувати через локальну мережу Wi-Fi з будь-якої точки планети, якщо є підключення до Інтернету. Основним недоліком установки системи «розумний дім» є висока вартість обладнання. Для багатьох, хто чув про гнучкість і ефективність системи, цей екран став перешкодою для покупки. У той же час, з дорогим обладнанням, якщо частина системи виходить з ладу, витрати на відновлення цілісності системи можуть бути непідйомними.

РОЗДІЛ 2 СТРУКТУРНІ ОСОБЛИВОСТІ СИСТЕМИ

2.1 Складові системи розумного будинку

2.1.1 Клімат контроль

Сучасні житлові приміщення оснащені комплексами технічних рішень для створення сприятливого мікроклімату будинку - це системи вентиляції, опалення, кондиціонування, пристрої для зволоження повітря. І вся ця система живлення вимагає регулярного контролю і уваги: налаштування режимів роботи, включення і виключення вікон. Однак за допомогою системи клімат-контролю «Розумний дім» усіма пристроями можна керувати через браузер з персонального комп'ютера, планшета чи телефону. Також можна централізовано встановити бажану температуру в кожній кімнаті, відстежувати поточну, а також стан погоди. [21]

2.1.2 Розумне освітлення

Сьогодні збереглося використання однієї люстри в центрі кімнати. У будинку, як правило, використовується багато освітлювальних приладів, як за кількістю, так і за типом. Система «Розумний дім» дозволяє керувати джерелами світла за допомогою запрограмованих опцій активації певні групи світильників. Наприклад: «Максимальне світло», «Середнє», «Мінімальне» та «Все вимкнено». У веб-програмі для кожного сценарію є своя кнопка. Отже, не потрібно обходити всі кімнати в будинку, щоб перевірити, чи все вимкнено. Досить активувати певний сценарій з персонального комп'ютера, телефону або планшета. А за допомогою зручної системи управління та контролю через браузер пристрою не буде пропущено жодної лампи.

2.1.3 Система безпеки

Однією з найважливіших підсистем системи розумний дім є підсистема безпеки. Це охоронно-пожежна сигналізація, відеоспостереження та домофон, а також технічна сигналізація (газ, вода та ін.). Вся інформація з датчиків і камер збирається і доступна в браузері. Ви контролюєте свій житловий простір і його системи, навіть не перебуваючи в ньому[11].

Структура системи «Розумний будинок» складається з трьох рівнів, а саме нижнього, середнього та верхнього. Нижній рівень складається з датчиків і виконавчих елементів, таких як сервоприводи, датчики вологості і температури, ультразвукові датчики і т.д. Середній рівень системи «Розумний будинок» в нашому випадку представлений у вигляді контролера Arduino Mega, до якого підключені всі датчики та виконавчі елементи. Верхні рівні – SCADA, і ці системи реалізовані на промислових контролерах, що призводить до високої вартості кінцевого продукту. Наша система реалізована на базі контролера Arduino Mega. Який входить у доступний ціновий сегмент

2.1.4 Нижній рівень системи

Датчик освітленості Модуль освітленості на LM393, використовується для вимірювання інтенсивності світла в різних пристроях, таких як, автоматизація світла (включенні світла вночі), роботах (визначення дня або ночі) і приладів контролюючих рівень освітленості. Вимірювання здійснюється за допомогою світлочутливого елемента (фоторезистора), який змінює опір в залежності від освітленості. Модуль освітленості з

чотири виводами містить два вихідних контакти, аналоговий і цифровий і два контакти для підключення живлення. Для зчитування аналогового сигналу передбачений окремий вивід «АО», з якого можна зняти показання напруги з 0В - 3.3В або 5В в залежності від використовуваного джерела живлення. Цифровий вивід DO, встановлюється в лог "0" або лог «1», в залежності від яскравості, чутливість виходу, можна регулювати за допомогою поворотного потенціометра. Вихідний струм цифрового виходу, здатний видати більше 15 мА, що дуже спрощує використання модуля і дає можливість використовувати його минаючи контролер Arduino і підключаючи його безпосередньо до входу одноканального реле або одного з входів двоканального реле [13]. Характеристики чутливості наведені в табл. 2.1.

Таблиця 2.1 – Характеристики чутливості модулю освітленості

Освітлення	Опір фоторезистора	Приклад
0,1 лк	600 кОм	Безмісячна ніч
1 лк	70 кОм	Місячна ніч
10 лк	10 кОм	Темна кімната
100 лк	1,5 кОм	Дуже похмурий день
1000 лк	300 Ом	Похмурий день

Технічні характеристики модуля RCK205502 наведені у таблиці 2.1.

Таблиця 2.1 – Технічні характеристики модуля RCK205502

Напруга живлення	+3,3В~+5,5В
Струм	10мА

Формат сигналу цифрового виходу	TTL (0/1)
Рівень сигналу аналогового виходу	0.. Vcc
Підключається безпосередньо до мікроконтролера	
Робоча температура	Від 0°C~70°C
Розміри	45×14×7 мм
Позначення виводів датчика:	
Vcc	Плюс живлення
GND	Минус живлення
DO	Цифровий вихід
AO	Аналоговий вихід

2.1.5 Wi-Fi модуль ESP8266

ESP8266 - це мікроконтролер, розроблений у 2014 році і що випускається компанією Espressif Systems - китайською компанією із Шанхаю. Він являє собою мережеве рішення з Wi-Fi-трансівером на борту плюс можливість виконання додатків, що записуються в його пам'ять.

Існує безліч модифікацій плат, які називаються зазвичай від ESP-01 до ESP-. Сьогодні вже з'явилися ще інші назви плат від сторонніх розробників. Відмінності в платах полягає в основному в портах введення-виведення, кількості флеш-пам'яті, виду конекторів і т.п. Процесор - той самий, отже з погляду програмування немає значення яку плату програмувати.[123,27]

Специфікація ESP8266:

- 1) Напруга живлення: 3,3 В
- 2) Енергоспоживання: 10 мкА ... 170 мА
- 3) Флеш-пам'ять: до 16 мб максимум (зазвичай 512 кб)

- 4) Процесор: Tensilica L106, 32 біта
- 5) Швидкість процесора: 80...160 МГц
- 6) ОЗП: 32 кб + 80 кб
- 7) Порти введення-виведення загального призначення:
17(мультиплексовані з іншими функціями)
- 8) АЦП: 1 введення з роздільною здатністю 102
- 9) Підтримка 802.11: b/g/n/d/e/i/k/r
- 10) Максимальна кількість підключень TCP: 5

Зі специфікації видно, що питання як довго працюватиме ESP8266 від батарейок не може бути легко визначений. Споживання енергії змінюється в дуже широкому діапазоні – при передачі на повній потужності воно становить 170 міліампер, а в режимі сну – лише 10 мА



Рисунок 2.1 - Зовнішній вигляд Wi-Fi модулю ESP8266

ESP8266 розроблений так, що він може використовувати підключений до нього модуль пам'яті, і це зазвичай флеш-пам'ять. Нагадаємо, що кількість циклів перезапису на таку пам'ять становить 10000 разів. Це цілком достатньо для випадків коли програма записує в пам'ять свої

налаштування або веде якийсь лог даних, але якщо ваша програма записує свої дані занадто швидко, пам'ять незабаром перестане працювати.[24,28]]

ESP8266 - це Wi-Fi пристрій, отже, і підключитися до нього можна через Wi-Fi, проте перед цим слід його налаштувати - процесор не знає назвою вашої локальної мережі та пароля для підключення до неї, а також інших можливих налаштувань. Це, звичайно, справедливе для випадку коли ми бажаємо щоб модуль підключався до нашої мережі. Для випадку, коли саммодуль працює в режимі точки доступу, все трохи складніше.

Для спрощення роботи з модулем на стадії програмування та налагодження своєї програми можна використовувати послідовний порт (UART). ESP8266 має спеціальний послідовний порт для цього – два порти, зазначені Rx і Tx. Tx - служить передачі даних, а Rx - прийому. Цими портами модуль з'єднується з відповідними портами. Найзручніше підключити цей порт до комп'ютера за допомогою перехідника USB-UART. При такому підключенні ми можемо відправляти модулю команди з програми терміналу прямо з клавіатури та отримувати відповіді від модуля до терміналу або записувати програму до модуля.

При підключенні через UART слід встановити однакову швидкість портів. У процесі завантаження модуль ESP8266 намагається автоматично визначити швидкість підключення пристрою-партнера і встановити в себе таку саму.

Модуль ESP8266 має також другий послідовний порт. Головне його призначення - виведення діагностичної та налагоджувальної інформації. Це може бути дуже корисним під час перевірки своєї програми. Пін Tx другого послідовного порту мультиплексовано з піном GPIO2.

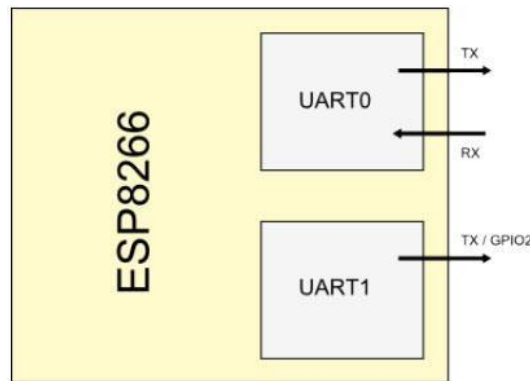


Рисунок 2.2 - Прийом та віддача сигналів Wi-Fi модулем ESP8266

Працюючи з пристроями стандарту Wi-Fi, бажано мати розуміння принципу його роботи у бездротовій мережі. На високому рівні Wi-Fi - це бездротова мережа для з'єднань TCP/IP. Wi-Fi – це набір протоколів бездротової мережі, описаних у стандарті IEEE 802.11.

Пристрій, званий Wireless Access Point (AP) - бездротовою точкою доступу працює як вузол комунікацій. Зазвичай він підключений або працює в режимі роутера. Наприклад, Wi-Fi роутер у вашому будинку працює у такому режимі.

Модуль ESP8266 може працювати як у режимі точки доступу (Access Point), так і в режимі клієнта – робочої станції (Station), а може і в обох режимах одночасно. Найчастіше точка доступу має підключення до інтернету та працює як міст між пристроєм та інтернетом. Декілька робочих станцій у локальній мережі спілкуються між собою також через точку доступу. Станція одночасно може бути підключена лише до однієї точки доступу. Кожен пристрій у мережі має власну унікальну MAC-адресу - 48-бітове значення.[10,25]

Якщо в межах видимості знаходиться кілька точок доступу, їх потрібно якось розрізняти, тому кожна точка доступу має мережний ідентифікатор, званий SSID (Service Set Identifier, іноді також званий BSSID) - це ім'я мережі, що має довжину до 32 символів.

Найшвидший спосіб поспілкуватися з модулем ESP8266 – це передати йому AT-команду та отримати відповідь. Набір AT-команд - це спеціальний набір інструкцій, які знає наш модуль і може виконувати певні дії при їх отриманні і видавати в термінал результат їх виконання. Програма, звана процесор AT-команд, вже встановлена в модулі ESP8266 і готова до прийому по послідовному порту. Ці команди починаються із символів "AT"-команди. Коли модуль підключено до терміналу комп'ютера, ми можемо відправити найпростішу команду - "AT". У відповідь на неї модуль повинен надіслати відповідь "OK". Синтаксис AT-команд наведений у таблиці 2.2

Таблиця 2.2 - Синтаксис AT-команд

Тип	Формат	Опис
Тест	AT+<x>=?	Запит параметрів та діапазону можливих значень
Запит	AT+<x>?	Запит поточних значень параметрів
Встановлення	AT+<x>=<...>	Встановлення значень параметрів
Виконання	AT+<x>	Виконання команд

Всі команди закінчуються символами "\r\n".

Модуль ESP8266 дозволяє записувати власні програми для їх запуску. Ви можете скомпілювати код з мови C і завантажити модуль. Така процедура називається "прошивкою" (flashing). Для того, щоб ваш додаток виконував корисні функції, у нього має бути можливість відправляти та приймати дані по мережі та/або працювати із зовнішніми датчиками, входами та виходами. ESP8266 оснащений базовими функціями для цього, набір яких є примітивною "операційною

системою". Служби цієї ОС можуть бути викликані вашим додатком. Вони повністю документовані і вам буде дуже корисно вміти користуватися ними.[6]

Наприклад, якщо потрібно підключитися до точки доступу WiFi, API має команду для цього. Для отримання IP-адреси також є API, для отримання часу із запуску модуля також є API. таких функцій дуже багато і вони дозволяють робити з модулем дуже багато корисних речей. Запам'ятати всі функції, звичайно, неможливо, але варто знати про їхнє існування.[26]

Коли модуль ESP8266 завантажується, він перевіряє значення рівнів на пінах MTDO (GPIO15), GPIO0 та GPIO2. Комбінація високих та низьких рівнів на цих пінах дозволяє отримати 3-бітове число з вісьмома можливими значеннями від 000 до 111. Кожне значення інтерпретується модулем. Інтерпретація наведена у таблиці 2.3

Таблиця 2.3 - Інтерпретація значень модулем

Значення	Десятичне значення	Роз'яснення
000	0	Немає даних
001	1	Завантаження даних по UART. Також дані для прошивки модуля.
010	2	Швидкий старт
011	3	Завантаження з флеш-пам'яті
100	4	SDIO low speed V2
101	5	SDIO high speed V1
110	6	SDIO low speed V1
111	7	SDIO high speed V2

На практиці це означає, що якщо ми хочемо, щоб модуль

завантажувався нормально, нам потрібно завантажити його з флеш-пам'яті (значення на контактах дорівнює 011), а якщо нам потрібно перепрошити модуль, тоді контакти повинні мати значення 001.

SDK (Software Development Kit - набір засобів розробника) для модуля ESP8266 для мови C являє собою набір файлів, званих заголовками з розширеннями .h. Простою мовою це файли визначення типів даних та функцій, які можуть знадобитися під час компіляції програми. SDK ESP8266 містить папку з файлами від компанії Espressif для нашого модуля.[14]

Програми для модуля ESP8266 пишуться зазвичай мовою C. Перед завантаженням програми в модуль її слід скомпілювати з тексту в машинні коди. Редагувати текст програми найзручніше в якомусь редакторі, який має підсвічування синтаксису, вбудовану довідку та інші корисні функції, і ще краще - в інтегрованому середовищі розробки (Integrated Development Enviroment). Працюючи в такому середовищі, ви можете написати текст програми, скомпілювати її і відразу ж завантажити в модуль.

Широко відомі такі середовища розробки як Eclipse та Arduino IDE. Eclipse дуже потужне середовище, розроблене компанією IBM і отримало статус відкритої багато років тому. Вона написана на Java, що означає можливість роботи в різних ОС: Windows, Linux, OS X. До цього середовища розробки можна підключати безліч додатків, а набір таких доповнень для мови C називається "C Developers Tools" (CDT). CDT не включає компілятор мови. Його слід вибрати самому. [27]

Arduino IDE - набагато простіше середовище розробки, яке, тим не менш, дозволяє писати і завантажувати в модуль свої програми. Як це робити описано у нашій статті "Arduino IDE + ESP8266".

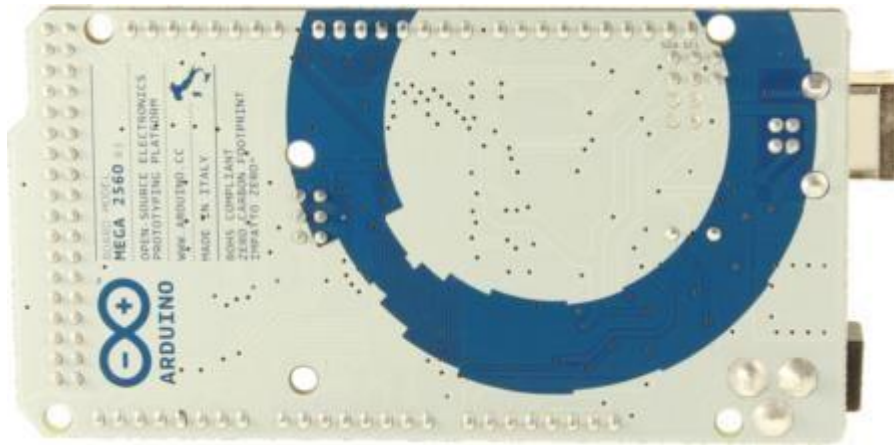
При написанні програми дуже часто вони працюють не так, як очікувалося. Для модуля ESP8266 налагодження (отримання службової інформації та стану системи) полегшується наявністю послідовного порту спеціально для

виведення налагоджувальної інформації. Ви можете надрукувати в UART1 (GPIO2), що хочете за допомогою функції `os_printf()`. Якщо підключити на піну GPIO2 модуля перетворювач UART-USB, ви зможете бачити цю інформацію на екрані комп'ютера в реальному часі. Таким чином, маючи один порт UART для прошивки модуля, а другий для налагодження, вам не доведеться нічого перемикає під час створення своєї програми. Увімкнення або вимкнення виведення інформації керується функцією `system_set_os_print()`.

2.2 Середній рівень системи

На середньому рівні системи буде використана плата Arduino UNO для керування всіма вище описаними датчиками. Arduino MEGA - плата від компанії Arduino (рис.2.3), побудована на мікроконтролері ATmega 2560. Для початку роботи з пристроєм досить просто подати живлення від AC/DC-адаптера або батарейки або підключити його до комп'ютера за допомогою USB- кабелю. Arduino Mega сумісний з більшістю плат розширення, розроблених для Arduino Duemilanove та Diecimila. Через перехідник живлення плата також може працювати і від батареї формату Крона, так і від акумулятора. [15]





Параметри плати 6,9x5,4см відповідно. 3) PWM: контакти 2-13 та 44-46. Функція `analogWrite()` потрібна, щоб виводити 8-бітовий ШІМ.

4) Інтерфейс SPI: контакти 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS) з бібліотекою SPI, ці виходи забезпечують зв'язок через інтерфейс SPI. Кабель SPI також прокладається до Arduino Uno, Роз'єми ICSP, сумісні з Duemilanove та Diecimila.

1) TWI: Виходи 20 (SDA) і 21 (SCL). При використанні бібліотеки Wire ці виходи забезпечують зв'язок через інтерфейс TWI.

Arduino Mega 2560 має 16 аналогових входів, кожен з яких може представити аналогову напругу у вигляді 10-бітного числа (1024 різних значення). За замовчуванням, вимірювання напруги здійснюється щодо діапазону від 0 до 5 В. Проте верхню межу цього діапазону можна змінити, використовуючи висновок AREF і функцію `analogReference()`.

Можна зробити ще кілька висновків:

1) AREF – може використовуватися як параметр для `analogReference()`.

2) Reset. Подача параметра LOW призведе до первантаження мікроконтролера та виконання циклу знову.

У Arduino Mega 2560 є цілий список методів для зв'язку з комп'ютером, іншим Arduino, чи іншими мікроконтролерами. Контролер ATmega2560 має 4 апаратних антени UART для підключення інтерфейсів. ATmega16U2 підключає один до USB-порту комп'ютера та, при підключенні, до ПК, дозволяє ідентифікувати Arduino, як віртуальний COM-порт. 4) Інтерфейс

SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS) з бібліотеками SPI, цей вихід дозволяє зв'язок через кабельний інтерфейс SPI. Uno, Du Milano та DiSimila.

5) Світлодіод: 13. Вбудований світлодіод підключено до контакту 13. Якщо параметр HIGH, то сигнал присутній.

1) TWI: Клеми 20 (SDA) і 21 (SCL) Під час використання бібліотеки Wire ці виходи дозволяють зв'язок через інтерфейс TWI. містить програму SerialMonitor. Існує спеціальний, який дозволяє читати та надсилати звичайні текстові дані на Arduino під час передачі чіпа. Світлодіоди ATmega8U2/ATmega16U2 RX і TX на платі флеш-пам'яті, коли USB під'єднано до ПК. (Для послідовної передачі даних за допомогою контактів 0 і 1 без USB-конвертера ці світлодіоди не використовуються.)

Для послідовного з'єднання цифрового виходу Mega2560 можна встановити та використати бібліотеку SoftwareSerial.

У мікроконтролері ATmega2560 реалізована підтримка інтерфейсів SPI і TWI на апаратному рівні. Для полегшення роботи з шиною TWI програмне забезпечення Arduino містить бібліотеку Wire. Програмування інтерфейсу Arduino MEGA відбувається через Arduino IDE.

Підключення мікроконтролера ATmega2560 здійснюється згідно інструкції STK500. Встановлення і використання нових програм для мікроконтролера можливо без додаткового програмування, в зв'язку з тим, що встановлюється з завантажувача.

Обминаючи завантажувач, мікроконтролер прошивається через ICSP порт.

У репозиторіях Arduino містяться данні по вихідне забезпечення для мікроконтролера ATmega16U2 (або ATmega8U2 для плат R1 і R2). Для увімкнення режиму DFU, потрібно:

- 1) Для плат версії R1 потрібно на звороті плати переключити перемичку, а потім скинути значення 8U2.

2) Для плат R2 та вище маєтся резистор для переходу у режим DFU, який підключає лінію мікроконтролера(8U2/16U2) HWB до GND. Є ще інший варіант, це встановлення за допомогою зовнішнього програматора, через роз'єм ISP. У цьому випадку завантажувач DFU буде тремтіти.

Arduino Mega 2560 збудований таким чином, щоразу перед завантаженням програми необхідно натискати кнопку reset. Це можна зробити програмно, завдяки підключеному комп'ютеру. Завдяки конденсатору 100нФ відбувається підключення до порту перезавантаження мікроконтролера ATmega2560. Це потрібно для виконання маніпуляцій за потоками даних(DTR), завдяки контролеру ATmega8U2. У тому разі, якщо лінія DTR видає нульове значення, то пін скидання також видає параметр LOW, що призводить до перезавантаження. Цей метод використовується щоб було можливо, при програмуванні ArduinoMega, виконати перезавантаження одним натисканням кнопки. Ця архітектура скорочує час завантажувача, тому що такий процес неодмінно зв'язаний зі зниженням рівня сигналу у лінії DTR.

Але такий метод призводить до наслідків, і це не треба ігнорувати. Для персональних комп'ютерів, на яких встановлені Mac OS X або Linux, при підключенні Mega 2560 одночасно відбувається скидання контролера, і це відбувається щоразу.

Завантажувач Arduino Mega2560 вмикається кожного разу, на протязі 0,5 сек., після обнови. Відразу після встановлення з'єднання завантажувач, хоча і запрограмований ігнорувати сторонні данні(будь яких, що не пов'язані із процесом відновлення програми), з пакета, що був надісланий на карту, може взяти декілька перших байтів даних. Працююча програма на Arduino, якщо очікує отримати якісь вхідні дані чи параметри при

старті персонального комп'ютера, необхідно бути впевненим, що повідомлення надсилається через секунду, після встановлення підключення до системи.

Плата Mega 2560 містить канал підключення (зазначену як «RESET-EN»), роз'єднавши підключення якого є можливість відключити перезавантаження мікроконтролера в авто – режимі. Для того щоб знов отримати можливість користуватися функцією перезавантаження в авто – режимі, потрібно знов з'єднати між собою канали, які розташовуються з торцевих сторін, цього каналу. Авто -перезавантаження можна відключити іншим чином, тобто додати в електричну схему резистор з супротивом 110 Ом, між каналами пере підключення та плюсом.

В електричній схемі Arduino Mega 2560 додані запобіжники які відновлюються, вони створені для протидії коротким замиканням та надмірному навантаженню портів USB персонального комп'ютера. Не дивлячись на це, у ПК є і свої системи захисту, але у такому разі запобіжники мікроконтролеру стають ще одним рівнем. У тому разі, якщо вхідний струм більший за 500мА, такі запобіжники обов'язково розірвуть підключення до усунення причини перевантаження або стрибок напруги.

Більшість плат розширення, які актуальні для Arduino Uno, Diecimila та Duemilanove, також будуть коректно працювати і з мікроконтрлером Arduino Mega2560. З цією метою усі аналогові (A0 – A5) і цифрові (D0 – D13) піни, а також живлення і ICSP-роз'єм на таких платах мають однакове розташування. Окрім цього, в указаних раніше пристроях, канали основного приймача UART підключені до однакових виводів (0 та 1), як і канали зовнішнього перезавантаження (висновки 2 і 3 відповідно). Канали для підключення інтерфейсу взаємодії SPI, в обох випадках підключені до роз'єму ICSP на Mega2560, так і на Duemilanove/Diecimila.

Потрібно добре пам'ятати, що на таких платах як Arduino Mega місце підключення I2C має інше положення.

Найвищий рівень системи SCADA – це пакет програм, пряма задача яких забезпечувати або підключатися до системі у цей самий час.

Його задача це обробка, відбір, архівація, відображення та збереження інформації. Система SCADA – може бути використана у будь яких шаблонах автоматизації, перевірки інформації або дослідів. Їх завдання це пошук вирішення поставленої задачі:

- 1) миттєве отримання та посилення даних між системою, платою та її пристроями;
- 2) розрахунки даних у той же час;
- 3) Видача логічних команд;

Але у них є східні по роду діяльності програми. У цей час є багато варіантів продуктів SCADA систем. Між собою вони відрізняються тим, які системи можуть підтримувати, своїм обладнанням, методами і звісно протоколами. Від цього залежить і їх вартість, але вони мають різноманітну цінову політику. У цьому проекті використана Web SCADA, котра являється системою, реалізованою через мову розмітки гіпертексту та таблиці каскадних стилей. Це несе з а собою велику перевагу, бо систему можна контролювати і через браузер. Зазвичай, вона використовується з метою видачі користувачу доступу до даних, а також їх подальшому збору та контролю. До складу входять такі програми як:

1. SCADA-Web, про який було детально розказано раніше.
2. SCADA-Сервер. Призначений для математичних розрахунків, управління базою даних системи і надання інформації за запитом
3. SCADA-Комутатор. Його завдання це збір, використання,

вираховування інформації, яка у подальшому, передається на сервера.

У використанні таких програм є плюси: простота роботи, безкоштовність та можливість змінювати код.

Але є мінуси:

1. Такі системи підтримують тільки заводські та промислові контроллери.
2. Також частіше за усе, підтримується тільки протокол Modbus і інтерфейсу OPC; У цьому проекті використання Rapid SCADA не є логічним. Бо він не працює з контролерами такого типу як Arduino. Для того, щоб усе таки використати у цьому разі Rapid, потрібно завантажити і використати додакові бібліотеки, як прописано у протоколі Modbus. Але такі гібриди, не будуть стабільно працювати. З метою надання більшої стабільності проекту, була задіяна платформа ASP.NET MVC.

Платформа ASP.NET MVC також має плюси:

- 1) Не зачепає інформацію про стан, і не використовує сервера, завдяки чому, можна отримати повну владу над додатком.
- 2) Вона опрацьовує електричну схему мікроконтролера, завдяки цьому вдається підтримувати розгалужену систему архітектури.
- 3) Допомогою внутрішніх протоколів, може розрізнити в додатку три окремі частини: уявлення, контролер і модель.

У цьому розділі мною було описану структуру системи «розумний будинок», яка складається з трьох основних рівнів. Кожен рівень відповідає частині пристрою, включаючи апаратне та програмне забезпечення. На найнижчому рівні система використовуватиме різні датчики (у вигляді окремих системних модулів для зчитування параметрів мікроклімату приміщення, а також присутності домашніх користувачів або «непроханих гостей». Інформація з модулів CES буде На наступному проміжному рівні системи буде використовуватися контролер на основі стандартної плати "Arduino MEGA", який буде підключений до всіх нижніх рівнів системи. Цей контролер з'єднає

датчики в єдину систему і збирає всі отриману інформацію для подальшого аналізу та прийняття необхідних рішень. На останньому третьому етапі системи буде програмне забезпечення, яке буде розташоване на окремому ПК-сервері, яке оброблятиме інформацію, зібрану від адміністратора. системна інформація, аналіз, виведення та модифікації на спеціальній платформі, доступній для користувачів, які використовують syst smart home time. Хоча домашній і smart — це дорога технологія, її потрібно починати, як тільки будується будинок і з самого початку планується якісне обладнання, але є й інші рішення. З точки зору дизайну, найпростіший будинок можна доповнити спеціальним прогресивним обладнанням, яке розширить функціональність житлового приміщення і модернізує транспорт.

РОЗДІЛ 3 ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ СИСТЕМИ

Постановка задачі. Ставлячи перед собою мету спроектувати дану систему, були сформульовані наступні завдання: аналіз технічних засобів розробки, створення і структурування бази даних і реалізація веб-додатки. Функціональні вимоги до системи «Розумний будинок»: Обов'язкова реєстрації користувача для повноцінної роботи в системі. Надання клієнту можливості скласти попередню компоновку і опис приміщення, в якому буде встановлена система розумного будинку.

1. Реалізація можливості управління системою розумного будинку користувачами, пов'язаними з клієнтом.
2. Реалізувати можливість зв'язку з оператором для виклику, отримання довідки з потрібних питань, а також усунення неполадок системи.
3. Реалізація системи батьківського контролю для клієнта.
4. Реалізувати фізичний доступ до системи за допомогою ідентифікації поелектронному ключу.
5. Реалізація системи оповіщення про екстрених ситуаціях за допомогою смс-повідомлень.
6. Контроль і управління температурою і вологістю повітря в приміщенні за допомогою модулів і плагінів web-інтерфейсу.
7. Створення системи аварійного відключення електроживлення.
8. Контроль і керування освітленням за допомогою модулів і плагінів web-інтерфейсу.
9. Реалізувати доступ до web-інтерфейсу з будь-якого пристрою (мобільного телефону, планшета, персонального комп'ютера і так далі).
10. Обов'язкове надання цілодобової доступності web-інтерфейсу.

11. Надання можливості установки системи розумного будинку на різні площі (житлові, робочі і так далі).

14.Наявність мобільного пристрою під управління Android або Ios. Наявність персонального комп'ютера, що має доступ до інтернету.

15. Наявність окремого контуру харчування для кожного габаритного побутового приладу.

16.Забезпечення електроенергією центрального контролера, модулів введення-виведення, GSM-модему, роутера, комутаторів LAN безперебійним джерелом живлення.

Алгоритми роботи системи:

Для початку імпортуємо всі бібліотеки для роботи нам потрібно <ESP8266WiFi.h>, <ESP8266WebServer.h>, <Keypad.h>, <UniversalTelegramBot.h>.

ESP8266WiFi - головна бібліотека для роботи з ESP, за допомогою якою і проходить головне підключення до інтернет мережі.

ESP8266WebServer - друга бібліотека для роботи з ESP, за допомогою цієї бібліотеки, створюється сервер, на який потім виграється код сайту.

Keypad - бібліотека яка дозволяє приєднати матричну клавіатуру, та виконувати з неї введення символів.

UniversalTelegramBot - за допомогою цієї бібліотеки ми підключаємо телеграм бота, який буде надсилати повідомлення, якщо хтось буде намагатись відчинити двері.

#define BOTtoken - за допомогою BOTFATHER(треба знайти в телеграмі),створюється бот та його токен, потім токен потрібно вставити в лапки тим самим ми приписуємо шлях до бота.

#define CHAT_ID - цей define також треба зробити через бота в телеграм підназвою CHAT_ID це нам дозволить напряду надсилати команди боту які треба буде виконати.

`#define Relay 6` - цим дефайном ми говоримо нашій програмі на якому піну стоїть реле. Реле нам необхідно за для відкриття електромагнітного замку, методом подання напруги.

`const char* ssid` та `const char* password` - за допомогою цих двох змінних ми говоримо нашій платі ESP до якої саме WiFi мережі треба приєднатися в поле `ssid` ми вводимо назву WiFi мережі, а в поле `password` відповідно пароль від даної мережі.

`ESP8266WebServer server(80)` - ініціалізація серверу на 80 порту.

`unsigned long currentTime = millis(); unsigned long previousTime = 0; const long timeoutTime = 1000;` - цією конструкцією робимо таймер

`String save; String code; String doorpos; int wrong = 0;` - ці строки задають головний механізм коду введення паролю. Зміна `save` відповідає за запам'ятання символів які були натиснуті; зміна `code` відповідає за пароль, який треба ввести для того, щоб відкрити двері; `doorpos` ця зміна віддає команди боту закриті чи відкриті двері після введення паролю; `wrong` зміна, що відповідає за кількість невірно введених паролей.

З 38 по 57 строку коду їде ініціалізація клавіатури, її підключення до відповідних пінів.

З 60 по 62 строку коду їде надання `root` прав телеграм боту.

З 66 по 69 строку прописано спроби під'єднання до WiFi та передачі даних на сервер.

`void setup()` - у цьому місці ми ініціалізуємо Serial Monitor, реле, та телеграм бота. Також виводиться спроби з'єднання з WiFi, та запуск серверу. якщо все вдало пройшло код читається далі, якщо деє з'явилася помилка, то він вирішує її до того часу поки не будуть виконуватись всі команди.

`void key()` - саме цей пункт коду відповідає за відкриття дверей та перевірку правильності паролю. за допомогою конструкції `if, else` ми перевіряємо чи натиснуто будь яку клавішу, якщо так то записуємо її, якщо була натиснута `#`, то ми перевіряємо схожість змінної `save` та

змінної code, якщо вони однакові, то реле відчиняє двері, якщо різні, то видає помилку та додається одна невірна спроба, якщо зібрати три невірні спроби, то буде викличена поліція. Якщо була натиснута * тоді все, що вводилось до цього у цієї спробіанулюється.

void message() - цей void віддає команди боту, який надсилає їх нам. Якщо була натиснута кнопка #, то робиться перевірка змінних code та save, та якщо вони однакові людині надсилається повідомлення, що двері відчинено, якщо були різні, то людина отримає повідомлення, що хтось намагається проникнути всередину.

Морфологічний аналіз засобів реалізації В даному розділі був проведений аналіз засобів реалізації системи «Розумний будинок». Морфологічний аналіз проведений в п'ять кроків, перший крок потрібен для вибору об'єкта аналізу, другий - вибір основних характеристик, за якими аналізується обраний об'єкт, третім кроком вказуються можливі варіанти реалізації характеристик, четвертим кроком розглядаються всі можливі варіації характеристик і п'ятим кроком проводиться вибір кращого варіанта за узагальненими критеріями.

Крок 1. Вибір об'єкта. Система «Розумний будинок»

Крок 2. Вибір основних характеристик об'єкта, які виражаються абстрактним поняттям.

1. Контролер
2. Середовище розробки
3. База даних
4. Мова програмування

Крок 3. Вказівка всіляких варіантів реалізації характеристик, обраних на кроці 2.

1. Контролер: Arduino UNO, Raspberry PI
2. Середовище розробки: Visual Studio, Eclipse
3. База даних: Microsoft SQL Server Standard 2016, Oracle Database StandardEdition 2

4. Мова програмування: C #, JavaКритерії оцінки:

1. Зв'язаність компонентів
2. Складність реалізації
3. Тимчасові витрати на розробку
4. Крок 4. Розгляд різних отриманих комбінацій.

Варіант 1:

Arduino UNO + Visual Studio + Microsoft SQL Server Standard 2016 + C #

Варіант 1 задовольняє обраному нами критерієм пов'язаності компонентів, так як середовище розробки Visual Studio безпосередньо пов'язана з мовою програмування C #, а також з Microsoft SQL Server Standard 2016. Arduino UNO також підтримує роботу з даними програмними рішеннями. Складність реалізації для даного випадку незначна, так як мова розробки відомий і знайома середовище розробки. Отже, тимчасові витрати також будуть незначними.

Варіант 2:

Raspberry PI + Eclipse + Oracle Database Standard Edition 2 + Java

Варіант 2 задовольняє обраному нами критерієм пов'язаності компонентів, так як середовище розробки Eclipse безпосередньо пов'язана з мовою програмування Java і база даних Oracle Database Standard Edition 2 може бути пов'язана за допомогою даних програмних рішень. Raspberry PI також підтримує роботу з даними програмним забезпеченням. Складність реалізації для даного випадку висока, так як мова розробки не відомий, проте знайома середовище розробки. З цього випливає, що тимчасові витрати будуть значними.

Варіант 3:

Raspberry PI + Visual Studio + Microsoft SQL Server Standard 2016 + C #

Варіант 3 задовольняє обраному нами критерієм пов'язаності компонентів, так як середовище розробки Visual Studio безпосередньо пов'язана з мовою програмування C #, а також з Microsoft SQL Server Standard 2016. Raspberry PI також підтримує роботу з даними

програмними рішеннями. Складність реалізації для даного випадку значна, так як мова розробки відомий, знайома середовище розробки, але контролер вимагає більш докладного вивчення, ніж чим Arduino UNO. Отже, тимчасові витрати також будуть значними.

Варіант 4:

Arduino UNO + Eclipse + Oracle Database Standard Edition 2 + Java

Варіант 4 задовольняє обраному нами критерієм пов'язаності компонентів, так як середовище розробки Eclipse безпосередньо пов'язана з мовою програмування Java і база даних Oracle Database Standard Edition 2 може бути пов'язана за допомогою даних програмних рішень. Arduino UNO також підтримує роботу з даними програмним забезпеченням. Складність реалізації для даного випадку висока, так як мова розробки не відомий, однак знаком контролер. З цього випливає, що тимчасові витрати будуть значними.

Варіант 5:

Arduino UNO + Eclipse + Microsoft SQL Server Standard 2016 + C #

Варіант 5 не задовольняє обраному нами критерієм пов'язаності компонентів, так як мова програмування C # не може бути скомпільовано за допомогою середовища розробки Eclipse. Отже, даний варіант не підлягає подальшому аналізу.

Варіант 6:

Raspberry PI + Visual Studio + Oracle Database Standard Edition 2 + Java

Варіант 6 не задовольняє обраному нами критерієм пов'язаності компонентів, так як мова Java не може бути скомпільовано за допомогою середовища розробки Visual Studio. Отже, даний варіант не підлягає подальшому аналізу.

Варіант 7:

Arduino UNO + Visual Studio + Oracle Database Standard Edition 2 + C #

Варіант 7 задовольняє обраному нами критерієм пов'язаності компонентів, так як середовище розробки Visual Studio безпосередньо пов'язана з мовою програмування C # і база даних Oracle Database Standard

Edition 2 може бути пов'язана за допомогою даних програмних рішень. Arduino UNO також підтримує роботу з даними програмним забезпеченням. Складність реалізації для даного випадку вкрай висока, так як робота з Oracle Database Standard Edition 2 з використанням C # можлива лише при реалізації десятків обхідних рішень, що спричинить за собою написання величезної кількості коду. Тимчасові витрати оцінити неможливо в зв'язку з вкрай високою складністю завдання.

Варіант 8:

Raspberry PI + Eclipse + Microsoft SQL Server Standard 2016+ Java
Варіант 8 задовольняє обраному нами критерієм пов'язаності компонентів, так як середовище розробки Eclipse безпосередньо пов'язана з мовою програмування Java і база даних Microsoft SQL Server Standard 2016 може бути пов'язана за допомогою даних програмних рішень. Raspberry PI також підтримує роботу з даними програмним забезпеченням. Складність реалізації для даного випадку висока, так як робота з Microsoft SQL Server Standard 2016 з використанням Java можливе лише за умови встановлення спеціальних драйверів, що може привести некоректної роботи системи. Тимчасові витрати будуть значними, так як розробники на мові програмування Java вузькоспеціалізовані.

Варіант 9:

Arduino UNO + Visual Studio + Microsoft SQL Server Standard 2016 + Java
Варіант 9 не задовольняє обраному нами критерієм пов'язаності компонентів, так як мова Java не може бути скомпільовано за допомогою середовища розробки Visual Studio. Отже, даний варіант не підлягає подальшому аналізу.

Варіант 10:

Raspberry PI + Eclipse + Oracle Database Standard Edition 2 + C #

Варіант 10 не задовольняє обраному нами критерієм пов'язаності компонентів, так як мова C # не може бути скомпільовано за допомогою середовища розробки Eclipse. Отже, даний варіант не підлягає подальшому аналізу.

Варіант 11:

Arduino UNO + Visual Studio + Oracle Database Standard Edition 2 + Java

Варіант 11 не задовольняє обраному нами критерієм пов'язаності компонентів, так як мова Java не може бути скомпільовано за допомогою середовища розробки Visual Studio. Отже, даний варіант не підлягає подальшому аналізу.

Варіант 12:

Arduino UNO + Visual Studio + Oracle Database Standard Edition 2 + Java

Варіант 12 не задовольняє обраному нами критерієм пов'язаності компонентів, так як мова Java не може бути скомпільовано за допомогою середовища розробки Visual Studio. Отже, даний варіант не підлягає подальшому аналізу.

Варіант 13:

Arduino UNO + Eclipse + Oracle Database Standard Edition 2 + C #

Варіант 13 не задовольняє обраному нами критерієм пов'язаності компонентів, так як мова C # не може бути скомпільовано за допомогою середовища розробки Eclipse. Отже, даний варіант не підлягає подальшому аналізу.

Варіант 14:

Raspberry PI + Visual Studio + Microsoft SQL Server Standard 2016 + Java

Варіант 14 не задовольняє обраному нами критерієм пов'язаності компонентів, так як мова Java не може бути скомпільовано за допомогою середовища розробки Visual Studio. Отже, даний варіант не підлягає подальшому аналізу.

Варіант 15:

Arduino UNO + Eclipse + Microsoft SQL Server Standard 2016 + Java

Варіант 15 задовольняє обраному нами критерієм пов'язаності компонентів, так як середовище розробки Eclipse безпосередньо пов'язана з мовою програмування Java і база даних Microsoft SQL Server Standard 2016 може бути пов'язана за допомогою даних програмних рішень. Raspberry PI також підтримує роботу з даними програмним забезпеченням. Варіант 16:

Raspberry PI + Visual Studio + Oracle Database Standard Edition 2 + C #

Варіант 16 задовольняє обраному нами критерієм пов'язаності компонентів, так як середовище розробки Visual Studio безпосередньо пов'язана з мовою програмування C # і база даних Oracle Database Standard Edition 2 може бути пов'язана за допомогою даних програмних рішень. Arduino UNO також підтримує роботу з даними програмним забезпеченням. Складність реалізації для даного випадку вкрай висока, так як робота з Oracle Database Standard Edition 2 з використанням C # можлива лише при реалізації десятків обхідних рішень, що спричинить за собою написання величезної кількості коду. Тимчасові витрати оцінити неможливо в зв'язку з вкрай високою складністю завдання.

Крок 5. Вибір найкращого варіанта за частковими критеріями показаний в таблиці 3.1. Порівняльна таблицю наведено у таблиці 3.2.

Таблиця 3.1 – Оцінки за частковими критеріями

Відмінно	100
Дуже добре	85
Добре	75
Задовільно	65
Посередньо	50
Незадовільно	0

Таблиця 3.2 – Вибір найкращого варіанта за узагальненими критеріями

Критерій №	Зв'язаність компонен.	Складність реалізації	Тимчасові витрати на розробку	Підсумок
Варіант 1 • Arduino UNO • Visual Studio • Microsoft SQL Server Standard 2016 • C#	100	75	75	250
Варіант 2: • Raspberry PI • Eclipse • Oracle Database Standard Edition2 • Java	100	65	50	215
Варіант 3: • Raspberry PI • Visual Studio • Microsoft SQL Server Standard 2016 • C#	100	50	50	200
Варіант 4: • Arduino UNO • Eclipse • Oracle Database Standard Edition2 • Java	100	50	0	150

Варіант 5: • ArduinoUNO • Visual Studio • Oracle Database Standard Edition 2 • C#	50	50	0	100
Варіант 6: • Raspberry PI • Eclipse • MicrosoftSQL ServerStandard 2016 • Java	65	0	50	110
Варіант 7: • ArduinoUNO • Eclipse • MicrosoftSQL ServerStandard 2016 • Java	65	50	50	165
Варіант 8: • RaspberryPI • VisualStudio • OracleDatabase Standard Edition 2 • C#	50	0	0	50

Після проведення морфологічного аналізу варіант 1 виявився найкращим для використання в розробці системи «Розумний будинок».

В даному розділі поставлена задача проектування системи «розумний

будинок» та описані програмні і апаратні вимоги даної системи.

Описані алгоритми роботи підсистеми для регулювання мікроклімату приміщення та підсистеми сповіщення, яка буде інформувати користувача про дії в будинку, які будуть відбуватися без його присутності в ньому. Також в розділі описано структуру додатку для керування всією системою «розумного будинку», описані діаграми використання класів додатку та описаний взаємозв'язок між класами та між користувачем. Розглянутий веб-додаток, який використовується для керування системою та описані всі його основні функції. У додатку А наведено програму системи захисту. У додатку Б наведено електричну схему системи захисту.

ВИСНОВКИ

Результатом роботи, є розробка системи безпеки жилого приміщення на базі мікроконтролера, створено проект, що зможе захищати власність людини чи її життя. Проект представляє собою сигналізацію з кодовим ключем та ультразвуковим сонаром. За допомогою датчика відбувається зчитування відчинення дверей чи вікна. Вся інформація подається в максимально зручному вигляді. Користувачу відправляється сповіщення у месенджер Телеграмм, за допомогою спеціального чат-боту.

Виконано наступні завдання:

- створено проект, який буде достатньо практичним у реальних умовах.
- більш детально вивчено роботу плати ArduinoMega.
- досліджено поняття системи «Розумний дім».
- використано декілька ступенів захисту.
- спрощено вже існуючі прототипи, тим самим знизивши собівартість, без втрати функціоналу.

Об'єктом дослідження було представлено безпеку житлового приміщення на базі мікроконтролера.

Предметом дослідження було заявлено керування системою охорони на прикладі системи розумного будинку та розробка системи безпеки жилого приміщення на базі мікроконтролера.

Відносно невелика ціна виготовлення такої охоронної системи на Arduino зробить її доступною для більшості населення. Новизною цього проекту є зручне для кожного використання та контроль за цією системою охорони будинку, що проявляється у підключенні API телеграм-боту напряму до програми, записаної на мікроконтролері та збереженні повної конфіденційності цієї інформації завдяки 256-шести бітному симетричному шифруванню AES, при алгоритмі PSA-2048 і

обміну ключами доступу методом Діффі-Геллмана.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Протоколи розумного будинку [Електронний ресурс]-Режим доступу: <https://idomus.company/> (дата звернення 01.10.2022).
2. Велт Т.Дж. "Умный дом" строим сами / Велт Т.Дж.,Элсенпитер Р.К.;Питер : КудИЦ-образ, 2005.384 с.
3. Сопер М.Э. Практические советы и решения по созданию "умного дома" : самоучитель. Москва : NT Press, 2007. 421 с.
4. Гололобов В.Н. "Умный дом" своими руками. Москва : NT Press, 2007. 216 с.
5. Умный дом Xiaomi[Електронний ресурс]-Режим доступу : <https://xiaomismarthome.ru/zigbee/> (дата звернення 01.11.2022).
6. Arduino и Raspberry PI: закляте враги или лучшие друзья? [Електронний ресурс]-Режим доступу : <https://habrahabr.ru/>
7. Arduino Leonardo [Електронний ресурс]-Режим доступу : <http://arduino.ua> (дата звернення 17.11.2022).
8. "Умный дом": методические указания для слушателей курсов повышения квалификации.Владимир : Изд-во ВлГУ, 2014. 123 с.
9. Петін В.В.Створення розумного будинку на базі Arduino.Житомир : ДМК Прес,2018. 180 с.
- 10.Android и Arduino. [Електронний ресурс] – Режим доступа:<https://studbooks.net> (дата звернення 29.10.2022).
11. Кокорева И. Системы "Интеллектуальное здание" и "Умный дом":эффективные решения и новейшие разработки. Электроника: НТБ, 2008, №7, с. 22–27.
12. Білостоцький Т., Осухівська Г. М. Математичне моделювання передачі даних в комп'ютерних мережах. Матеріали II науково-технічної конференції «Інформаційні моделі, системи та технології»,

2012. С.36.

13. Менахем Домб. Системи розумного дому, заснованого на основі інтернет речей. [Електронний ресурс] – Режим доступа <https://www.intechopen.com/books/internet-of-things-iot-for-automated-and-smart-applications/smart-home-systems-based-on-internet-of-things> (дата звернення 05.11.2022).

14. Канали ZigBee та Wi-Fi. Співіснування ZigBee та WiFi Networks.[Електронний ресурс] – Режим доступа URL: <https://support.metageek.com/hc/en-us/articles/203845040-ZigBee-and-WiFi-Coexistence> (дата звернення: 15.11.2022).

15. Кузик А. О. Спектральний та кореляційний аналіз сигналів з неперервною та дискретною частотною модуляцією. Нац. Ун. Львівська політехніка. 31-44 с.

16. МІЙ ДОМОВИК. Розумний будинок без проводів і ремонту [Електронний ресурс]-Режим доступа до ресурсу <https://moy-domovoy.ru/> (Дата звернення: 01.11.2022);

17. Дужак І. О. Розумний будинок. *Автоматизація технологічних і бізнес-процесів*. 2013. №13. С. 31.

18. Харке В.Н. Розумний будинок. Об'єднання в мережу побутової техніки та систем комунікацій в житловому будівництві *Техносфера*, 2006. 292 с.

19. Бондарев О. Хто в дом і господар. Розумні будинки через кілька років набудуть широкої популярності. *Корреспондент*. 2012. №30. С.42-46;

20. Основні проблеми розумних будинків і як їх можна вирішити? [Електронний ресурс] // Кластер. Інженетрі системи та мережі. 2019. Режим доступа до ресурсу: <https://klaster.ua/ua/stati-i-obzory/osnovnye-problemy-umnyh-domov-i-kak-ih-mozhno-reshit/> (Дата звернення: 08.10.2022)

21. Зовнішній датчик руху «Розумного» будинку: [Електронний ресурс]

Режим доступу до ресурсу: <http://sutem.com.ua/7135smartbus.php> (Дата звернення: 09.10.2022);

22. Шлюзи зв'язку «Розумного» будинку: [Електронний ресурс]: Режим доступу до ресурсу: <http://sutem.com.ua/7141smartbus.php> (Дата звернення: 15.10.2022);

23. Керуючі пристрої «Розумного» будинку: [Електронний ресурс]: Режим доступу до ресурсу: <http://sutem.com.ua/7121smartbus.php> (Дата звернення: 26.10.2022);

24. Arduino Uno [Електронний ресурс]. Режим доступу: <https://doc.arduino.ua/ru/hardware/Uno> (Дата звернення: 11.11.2022);

25. Arduino Uno [Електронний ресурс]. Режим доступу: <http://arduino.ru/Hardware/ArduinoBoardUno> (Дата звернення: 14.11.2022);

26. Arduino Uno и Raspberry Pi 3. От схемотехники к интернету вещей. Макаров С. Л. 2018. 202с.: ил

27. Функції та комфорт розумного дому: [Електронний ресурс]: Режим доступу до ресурсу: <http://melask.com.ua/rozumniy-dim/komfort.html> (Дата звернення: 28.11.2022);

28. Розумний будинок: [Електронний ресурс]: Режим доступу до ресурсу: <http://lady.tochka.net/ua/58959-что-такое-umnyy-dom/> (Дата звернення: 28.11.2022);

29. Розумний дім – технологія економії: [Електронний ресурс]: Режим доступу до ресурсу: <http://right-choice.com.ua/rozumniy-dimtehnologiya-ekonomiyi-zruchnosti-i-komfortu-visokogo-rivnya> (Дата звернення: 12.11.2022);

30. Зовнішній датчик руху «Розумного» будинку: [Електронний ресурс]: Режим доступу до ресурсу: <http://sutem.com.ua/7135smartbus.php> (Дата звернення: 26.11.2022).

ДОДАТОК А ПРОГРАМА СИСТЕМИ ЗАХИСТУ

```
//підключаємо бібліотеки
#include <ESP8266WiFi.h>//бібліотека інтернет контроллеру
#include <ESP8266WebServer.h>//бібліотека інтернет контроллеру
#include <ArduinoJson.h>
#include <Keypad.h>//бібліотека клавіатури

const char* ssid = "2/18-1"; //ваш WiFi
const char* password = "05064632091";//пароль від інтернету

ESP8266WebServer server(80); // ініціалізація сервера на 80 порт

String header;

//додавання таймеру
unsigned long currentTime = millis();
unsigned long previousTime = 0;
const long timeoutTime = 1000;

String save; // зміна для передачі натиснутих клавіш
String code = "1111"; // зміна, в якій вказано правильний пароль
int wrong = 0; //зміна не вірних спроб

// ініціалізація клавіатури
const byte ROWS = 4;
const byte COLS = 4;
```

```
//присвоєння кожній клавіші свого символу
```

```
char hexaKeys[ROWS][COLS] = {
    {'1', '2', '3', 'A'},
    {'4', '5', '6', 'B'},
    {'7', '8', '9', 'C'},
    {'*', '0', '#', 'D'}
};
```

```
//піни підключення клавіатури
```

```
byte rowPins[ROWS] = {2, 3, 4, 5}; // строчки
byte colPins[COLS] = {8, 9, 10, 11}; // колонки
```

```
Keypad passwordKeypad = Keypad(makeKeymap(hexaKeys), rowPins, colPins, ROWS, COLS);
```

```
// конект до WiFi
```

```
const uint16_t lengt=24; //кількість спроб
uint16_t tick=0;
time_t tnow[lengt];
float h[lengt], t[lengt];
```

```
void key(){
```

```
    char passwordKey = passwordKeypad.getKey();//перевод в розуміючу мову для коду яка клавіша була натиснута
```

```
    if (passwordKey){//якщо будь яка клавіша натиснута
```

```
        if (passwordKey == '#')// якщо натиснута клавіша "#", то йде перевірка чи правильно було введено пароль
```

```
        {
```

```
            if (save == code) {//якщо пароль введено правильно, то відкрити двері
```

```
                Serial.println("Pass");// вивести напис, що дверь відчинена
```

```
                wrong = 0;//та анулювати кількість невірних спроб
```

```

    }
    else { //в іншому випадку вивести неправильний код
        Serial.println("Wrong pass");//вивод напису
        wrong = wrong + 1; //та додати одну невірну спробу
        if (wrong > 3) { //якщо кількість невірних спроб більше трьох
            Serial.println("Call police");//виведеться напис
        }
    }
    save = ""; //завжди анулювати зміну для її відновлення
}
else{
    save = save + passwordKey; //в іншому випадку додати до змінюю "save" ще
    один символ
    Serial.println(save); // вивести пароль який вводить людина
}
    if (passwordKey == '*') { //якщо натиснута "*", то анулювати все, що було
    введено до цього
        save = "";
    }
}
}

```

```

//запуск сайту
void handleRoot() {
    String trendstr;
    trendstr = F("<html>\n
<head>\n
    <meta charset='UTF-8'>\n
    <meta http-equiv='X-UA-Compatible' content='IE=edge'>\n
    <meta name='viewport' content='width=device-width, initial-scale=1.0'>\n
    <link rel='stylesheet' href='style.css'>\n
    <title>Door</title>\n
</head>\n
<body>\n

```

```

<div class='passplace'>\
  <h1>Your Home - Your Safe</h1>\
  <h3>Pass: </h3>\
  <input type='text' id='pass' name='pass'> \
  <br>\
  <input type='submit' value='змінити' class='btn'>\
</div>\
\
  <script>\
    let reg = /[A-D]+[0-9]/;\
    let inp = document.querySelector('#pass');\
  \
    document.querySelector('.btn').onclick = function(e){\
  \
    e.preventDefault();\
    if(!reg.test(inp.value)){\
      alert('введіть пароль за прикладом: спочатку введіть будь яку кількість літер від A-D, а потім цифри')\
    }\
    else{\
      alert('успішно змінено')\
    }\
  }\
  \
  </script>\
</body>\
<style>\
*{\
  margin: 0;\
  padding: 0;\
}\
body{\
  background-color: rgb(76, 76, 76);\
}\
.passplace{\
  margin: 10px;\

```

```

    text-align: center;\
    color: aliceblue;\
    background-color: black;\
    border: 50px, 50px, 50px, 50px;\
}\
</style>\
</html>");

```

```

    server.send ( 200, F("text/html"), trendstr );
}
//виведення даних на Serial monitor
void setup() {
    Serial.begin(115200);
    Serial.print("Приєднуюсь до: "); //
    Serial.println(ssid);
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }

    Serial.println("");
    Serial.println("WiFi під'єння.");
    Serial.println("IP адреса: ");
    Serial.println(WiFi.localIP());
    server.on("/", handleRoot);
    server.begin();
    Serial.println("приєднано до серверу");
}

void loop(){
    server.handleClient();
}

```

ДОДАТОК Б ЕЛЕКТРИЧНА СХЕМА СИСТЕМИ ЗАХИСТУ

