

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

ННК “Інститут прикладного системного аналізу”  
(повна назва інституту/факультету)

Кафедра Системного проектування  
(повна назва кафедри)

«До захисту допущено»

Завідувач кафедри

\_\_\_\_\_ А.І.Петренко  
(підпис) (ініціали, прізвище)

“ ” \_\_\_\_\_ 2015 р.

**Дипломна робота**

першого (бакалаврського) рівня вищої освіти  
(першого (бакалаврського), другого (магістерського))

зі спеціальності 7.050102, 8.050102 Інформаційні технології проектування  
7.050103, 8.050103 Системне проектування  
(код та назва спеціальності)

на тему: "Система управління розумним домом. Програмне забезпечення контролера виконавчих пристроїв"

Виконав (-ла): студент (-ка) 4 курсу, групи ДА-12  
(шифр групи)

\_\_\_\_\_ Чернацький Артем Юрійович \_\_\_\_\_  
(прізвище, ім'я, по батькові) (підпис)

Керівник \_\_\_\_\_ ст. викладач Бритов О.А. \_\_\_\_\_  
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант Охорона праці доцент, к.б.н., Гусєв А.М. \_\_\_\_\_  
(назва розділу) (посада, вчене звання, науковий ступінь, прізвище, ініціали) (підпис)

Рецензент \_\_\_\_\_ доцент, к.т.н., Тимошенко Ю.О. \_\_\_\_\_  
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Нормоконтроль \_\_\_\_\_ ст. викладач Бритов О.А. \_\_\_\_\_  
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій дипломній роботі немає запозичень з праць інших авторів без відповідних посилань.

Студент \_\_\_\_\_  
(підпис)

Київ – 2015 року



3. Розробити програмне забезпечення для контролера виконавчих пристроїв
4. Розробити пакет для Android, що дозволяє взаємодіяти з мікроконтролером.
5. Розробити базу даних та створити пакет для роботи з нею.
6. Розробити програмне забезпечення для генерації файлів конфігурацій.
7. Виконати тестування роботи системи у комплексі.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслеників, плакатів тощо)

1. Схема підключення сенсорів – плакат.
2. UML Діаграма класів пакету ArduinoTelnet – плакат.
3. UML Діаграма класів пакету ShomeDB – плакат.
4. Інтерфейс генератора конфігурацій – плакат.

6. Консультанти розділів роботи \*

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	Гусев А.М., доцент		

7. Дата видачі завдання 01.02.2015

#### Календарний план

№ з/п	Назва етапів виконання дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Отримання завдання	01.02.2015	
2	Збір інформації	15.02.2015	
3	Аналіз можливостей мікроконтролерів, мікрокомп'ютерів, периферійних пристроїв та датчиків.	28.02.2015	
4	Розробка тестового стенду	02.03.2015	
5	Розробка програми для мікроконтролера	10.03.2015	
6	Розробка пакету для Android	15.03.2015	
7	Розробка бази даних та пакету для неї	25.03.2015	
8	Розробка генератора конфігурацій	25.04.2015	
9	Тестування роботи моделі	30.04.2015	
	Оформлення дипломної роботи	31.05.2015	
10	Отримання допуску до захисту та подача роботи в ДЕК	08.06.2015	

Студент

\_\_\_\_\_

(підпис)

А.Ю. Чернацький

(ініціали, прізвище)

Керівник роботи

\_\_\_\_\_

(підпис)

О.А. Бритов

(ініціали, прізвище)

\*Консультантом не може бути зазначено керівника дипломного проекту (роботи).

# АНОТАЦІЯ

Бакалаврської дипломної роботи Чернацького Артема Юрійовича на тему: "Система управління розумним домом. Програмне забезпечення контролера виконавчих пристроїв."

Дипломна робота присвячена розробці програмного забезпечення контролера виконавчих пристроїв для реалізації концепції "розумного" будинку. Тема роботи є актуальною у зв'язку зі зростанням ціни на енергоносії та складною економічною ситуацією у нашій країні.

Цілю дипломної роботи є розробити функціональну схему системи управління, обрати протоколи зв'язку елементів системи та створити програми управління виконавчими приладами.

В роботі проведено аналіз мікроконтролерів та мікрокомп'ютерів, досліджено можливості різних датчиків.

В ході виконання дипломної роботи було побудовано робочу модель системи, а також тестовий стенд з датчиками температури, вологості, тиску та можливістю керування деякими приладами.

В результаті роботи було побудовано робочу модель розумного дому з елементами штучного інтелекту.

Загальний обсяг роботи – 90 сторінок, 50 рисунків, 13 таблиць, 1 додаток, 32 бібліографічних найменувань.

Ключові слова: Arduino, Android, "розумний" будинок, температура, вологість, тиск, освітлення.

# АННОТАЦИЯ

Бакалаврской дипломной работы Чернацкого Артема Юрьевича  
на тему: "Система управления умным домом. Программное обеспечение контроллера исполнительных устройств"

Дипломная работа посвящена разработке программного обеспечения контроллера исполнительных устройств для реализации концепции "умного" дома. Тема работы является актуальной в связи с подорожанием цены на энергоносители и сложной экономической ситуацией в нашей стране.

Целью работы является разработать функциональную схему системы управления, выбрать протоколы связи элементов системы и создать программы управления исполнительными устройствами.

В работе проведен анализ микроконтроллеров и микрокомпьютеров, исследованы возможности различных датчиков.

В ходе выполнения дипломной работы было построено рабочую модель системы, а так же тестовый стенд с датчиками температуры, влажности, давления и возможностью управления некоторыми приборами.

В результате работы было построено рабочую модель умного дома с элементами искусственного интеллекта.

Общий объем работы - 90 страниц, 50 рисунков, 13 таблиц, 1 приложение, 32 библиографических наименований.

Ключевые слова: Arduino, Android, "умный" дом, температура, влажность, давление, освещение.

# ANNOTATION

For the bachelor's degree work of Chernatsky Artem Yuriyovich  
on "smart home control system. The software controller actuators"

This work is devoted to developing software controller actuators to implement the concept of "smart" home. Topic is important in because energy prices have rised.

Aim of the work is to develop a functional scheme management system, choose communication protocols and elements of the system and create a program management executive devices.

The analyze of microcontrollers and microcomputers and investigation of possibility of different sensors were made in this work.

A working model of the system with sensors of temperature, humidity, pressure and the ability to control some devices was built .

As a result of working model was built smart house with elements of artificial intelligence.

The total amount of work - 90 pages 50 images, 13 tables, 1 addition, 32 bibliographic titles.

Tags: Arduino, Android, "smart" house, temperature, humidity, pressure, light.

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ .....	10
ВСТУП .....	11
1 ТЕХНОЛОГІЯ "РОЗУМНИЙ" БУДИНОК .....	13
1.1 Історія технології "розумний" будинок .....	13
1.2 Принцип роботи системи .....	15
1.3 Основні характеристики системи .....	17
1.4 Існуючі концепції .....	19
1.5 Переваги та недоліки існуючих рішень .....	21
1.6 Власний варіант реалізації .....	22
1.7 Висновки .....	24
2. АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ .....	25
2.1 Аналіз можливостей мікроконтролерів та мікрокомп'ютерів .....	25
2.1.1 Arduino .....	25
2.1.1.1 Arduino Uno .....	25
2.1.1.2 Arduino Mega2560 .....	27
2.1.2 ChipKit .....	29
2.1.2.1 Uno32 .....	29
2.1.2.2 Max32 .....	31
2.1.3 Raspberry PI .....	33
2.1.3.1 Model A+ .....	33
2.1.3.2 Model B+ .....	35
2.1.3.3 Model 2 .....	36
2.1.4 Висновки щодо мікроконтролера/мікрокомп'ютера .....	38
2.2 Аналіз можливостей периферійних пристроїв .....	38
2.2.1 W5100 Ethernet Shield .....	38
2.2.2 SD Card Reader .....	40
2.2.3 Реле SONGLE SRD-05VDC .....	40

2.3 Аналіз можливостей датчиків.....	41
2.3.1 DS1820/DS18B20/DS18S20 .....	41
2.3.2 DHT11/DHT22/DHT21.....	42
2.3.3 BMP180 .....	44
2.3.4 Фоторезистор СФ2-5а.....	44
2.4 Схема тестового стенду .....	45
2.5 ASUS NEXUS 7 2012 16GB .....	47
2.6 Висновки .....	48
3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ .....	49
3.1 Програмне забезпечення Arduino .....	49
3.2 Програмне забезпечення Android.....	55
3.2.1 Клас Arduino .....	56
3.2.2 Клас Connector.....	57
3.2.3 Клас Room.....	59
3.2.4 Клас Sensors .....	59
3.2.4 Клас Checker .....	61
3.3 Програмне забезпечення бази даних.....	62
3.3.1 База даних .....	63
3.3.2 Клас SensorDataBaseHelper .....	65
3.3.3 Клас SensorDataBase .....	66
3.4 Генератор файлів конфігурацій .....	68
3.5 Висновки .....	73
4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ .....	74
4.1 Вступ .....	74
4.2 Аналіз умов праці в приміщенні .....	74
4.2.1 Оцінка санітарно-гігієнічних умов праці .....	74
4.3 Стан повітряного середовища приміщення .....	77
4.4 Освітлення приміщення .....	79
4.5 Оцінка умов безпеки праці.....	81
4.5.1 Вимоги електробезпеки.....	81



4.5.2 Оцінка пожежної безпеки .....	82
4.6 Висновки .....	83
ВИСНОВКИ.....	85
ПЕРЕЛІК ПОСИЛАНЬ.....	87
ДОДАТОК А.....	90
UML діаграма класу ArduinoTelnet .....	90

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

3NF	Third normal form, Третя нормальна форма
framework	Готовий до використання комплекс програмних рішень
ПЗ	Програмне забезпечення
БД	База даних
скетч	Блок коду для Arduino
SPI	Serial Peripheral Interface, послідовний периферійний інтерфейс
КБ	Одиниця вимірювання обсягу даних, що дорівнює 1024 байтам
МБіт	Кількість інформації, $10^6$ або 1000000 (мільйон) біт.
KNX	Комунікаційна шина асоціації KONNEX

## ВСТУП

У наші дні спостерігається бурхливий ріст та розвиток інформаційних технологій. Те, що здавалось фантастикою декілька десятиліть назад, зараз є буденною звичайністю. Швидкість зростання обсягів даних та обчислювальних потужностей надзвичайно стрімкий. Сучасні мобільні пристрої зазвичай мають технічні характеристики в рази кращі, ніж комп'ютери минулого, а їхня собівартість зменшується з кожним роком.

За такого рівня технічного прогресу у більшості звичайних побутових пристроїв вже є власний процесор, що керує ним та дозволяє автоматизувати роботу. В результаті таких досягнень виникає питання, а чи можливо створити єдину систему контролю пристроїв, що нас оточують? Чи можна автоматизувати більшу частину побутових процесів?

Відповіддю на ці питання стали розробки системи "розумного" будинку. Багато різноманітних відомих та не дуже компаній займаються розробкою та створенням таких систем.

Основними перевагами системи "розумного" будинку є можливість автоматизації роботи пристроїв у домі. Це дозволяє автоматично контролювати параметри мікроклімату у приміщенні, вмикати та вимикати різного роду мультимедійні пристрої.

Іншою корисною функцією даної системи є реалізація можливості енергозбереження. Якщо система контролює охолоджувальні та нагрівальні установки, то вона здатна ефективно витратити енергію, адже очевидним є той факт, що немає ніякої необхідності підтримувати комфортну температуру у приміщенні за умови відсутності господаря. Для нашої країни енергозбереження є дуже актуальним питанням, адже ціни на енергоносії суттєво зросли за останній час.

При всіх позитивних моментах використання системи "розумного" будинку вона має і недоліки. Основним з них є вартість готової системи, адже

комплексні технічні рішення від виробників зазвичай коштують занадто багато, щоб створити масовий інтерес до продукту.

Іншим недоліком є фактична відсутність стандартизації на даний момент. Кожен з виробників реалізує концепцію "розумного" будинку виключно у своєму баченні. Він створює свою специфікацію, стандартизацію і модулі, які зазвичай не будуть працювати в системі від іншого виробника. Це створює складності для користувачів, адже при необхідності модернізації чи розширення системи вони будуть змушені використовувати тільки певні фіксовані елементи.

Також проблемою готових рішень під ключ є неможливість модернізації їх користувачами власноруч. Очевидно, що потенційними користувачами систем концепції "розумного" будинку є люди з технічною освітою, адже вони як ніхто інший розуміють усі її переваги. Відсутність можливості власноруч щось змінити позбавляє ці рішення однієї великої складової - товариства користувачів. Ці організації фанатів зазвичай і займаються власноручною модернізацією, покращенням та оптимізацією проектів. Усім чудово відомі товариства користувачів Linux та Unix, які займаються розвитком проектів, їх модернізацією та покращенням. Якби у людей була можливість приймати участь у створенні системи, то це б підвищило інтерес та прискорило процес її становлення.

Враховуючи усі недоліки готових систем, які наведені вище, було прийнято рішення розробити власну відкриту концепцію "розумного" будинку та виконати її реалізацію. Метою дипломної роботи є розробка функціональної схеми системи управління, вибір протоколів зв'язку елементів системи та розробка програми управління виконавчими приладами.

# 1 ТЕХНОЛОГІЯ "РОЗУМНИЙ" БУДИНОК

"Розумний" будинок - житловий будинок сучасного типу, організований для проживання людей за допомогою автоматизації і високотехнологічних пристроїв. Під "розумним" будинком слід розуміти систему, яка забезпечує безпеку та збереження ресурсів, а також комфорт для всіх користувачів. У найпростішому випадку вона повинна вміти розпізнавати конкретні ситуації, що складаються в будинку, і відповідним чином на них реагувати: одна з систем може керувати поведінкою інших по заздалегідь виробленим алгоритмам. Крім того, від автоматизації декількох підсистем забезпечується синергетичний ефект для всього комплексу.

Зі збільшенням обчислювальної здатності пристроїв концепція "розумного" будинку отримала своє логічне продовження – систему "Інтернет речей", згідно з якою була проведена первинна стандартизація та визначені основні правила та рекомендації до побудови готового продукту на рівні як системи загалом, так і окремих компонентів. Незважаючи на відносну новизну, вже зараз існує кілька десятків різних рішень.

## 1.1 Історія технології "розумний" будинок

Історія "розумного" будинку почалася в 1961 році, коли Джоель і Рут Спіра винайшли і запатентували спеціальний пристрій для плавного регулювання світла - димер. Саме цей винахід став приводом для створення всесвітньо відомої сьогодні компанії Lutron Electronics Company, Inc [1]. Дана фірма продовжувала працювати над "розумними" технологіями, паралельно впроваджуючи в побут такі поняття, як світлова зона і сцена.

Значною подією у подальшому розвитку технології "розумного" будинку було створення шведською компанією Pico Electronics побутової автоматики в 1975 році, яку вперше почали використовувати для управління музичними програвачами [2]. Удосконалили домашню автоматику американці Скотт і

Росслин Міллер.

Першим повноцінним проектом "розумного" будинку став невеликий житловий будинок на південному березі Англії. В основу його автоматики лягло використання широкосмугової KNX-системи, що відповідає за управління освітленням, сигналізацією, жалюзями, опаленням та дверима гаража. Також в даному будинку був створений басейн, який згодом доповнили системою світлодіодного освітлення з оригінальними колірними ефектами.

У 1987 році організація ASHRAE створила новий протокол домашньої автоматизації, який надалі був вдосконалений групою компаній Berker, Merten, Insta, Gira, Jung та Siemens і перетворений в абсолютно нову модель автоматики European Installation Bus. У 1999 році на її основі було розроблено нове покоління польових шин KNX, які досі вважаються кращим стандартом європейських систем домашньої автоматизації.

Сучасні системи пішли далеко вперед, істотно розширивши свої технічні можливості. Сьогодні в них використовуються вбудовані домашні кінотеатри, об'єднуються всі інженерні системи, застосовується інтелектуальне управління на основі спеціального ПЗ. Завдяки модульності системи у користувачів з'явилася можливість самостійно обирати функціонал "розумного" будинку.

У 2008-2009 роках була сформована концепція "Інтернет речей", яка стала логічним продовженням переходу до модульної архітектури та зробила тісну інтеграцію будь-яких підтримуваних приладів простою як ніколи. Інтернет речей - це не просто безліч різних приладів і датчиків, об'єднаних між собою дротяними та бездротовими каналами зв'язку і підключених до мережі Інтернет, а це більш тісна інтеграція реального та віртуального світів, в якому спілкування виробляється між людьми і пристроями.

У 2014 році компанія Apple зробила анонс нового комплексу програмних рішень під назвою HomeKit, який створений для інтеграції екосистеми "розумного" будинку з пристроями на iOS та автомобілями з системою Apple CarPlay чи Google Car [3].

## 1.2 Принцип роботи системи

Термін "розумний" будинок не має чіткого визначення, а тому під це поняття підпадає будь-яка система з автоматизованим керуванням приладами, яка спрощує життя людини та підвищує рівень її комфорту. Через нечіткі рамки виникло багато реалізацій з різним рівнем інтеграції та принципом роботи. Їх можна умовно поділити на три групи:

- Вбудовані системи з центральним контролером;
- Вбудовані системи без центрального контролера;
- Системи з інтеграцією, що налаштовується.

Перша група представляє з себе повністю налаштовану і встановлену виробником систему, яка керується центральним обчислювальним пристроєм і не передбачає прямої взаємодії своїх компонентів між собою. Всі призначені для користувача налаштування зберігаються на центральному пристрої (сервері), а периферія лише виконує отримані від нього інструкції і зазвичай не має вбудованої пам'яті і обчислювальних потужностей. Принцип роботи системи зображений на рис. 1.1.



Рисунок 1.1 – Діаграма, що представляє принцип роботи вбудованої системи з центральним контролером

Друга група є системою з напіваавтономними пристроями. Алгоритми взаємодії прописуються з програми контролера безпосередньо в пам'ять кожного пристрою і для їх зміни пристрій буде необхідно перепрограмувати. У зв'язку з відсутністю центрального компонента, зв'язки між приладами відбуваються безпосередньо і є можливість створення автономних груп, замкнутих один на одного. Принцип роботи на рис. 1.2.



Рисунок 1.2 – Діаграма, що представляє принцип роботи вбудованої системи без центрального контролера

Третя група - це зовнішні контролери, які приєднуються до звичайних приладів і залежно від показання своїх сенсорів і вбудованого алгоритму регулюють їх роботу. Можуть мати центральний контролер, але часто керуються і налаштовуються з Інтернет або хмарного сервісу. Функціонують здебільшого як незалежні модулі і для налаштування прямого зв'язку можуть знадобитися додаткові датчики / сенсори. Принцип роботи на рис. 1.3.





Рисунок 1.3 – Діаграма, що представляє принцип роботи системи з інтеграцією, що налаштовується

### 1.3 Основні характеристики системи

Основне призначення системи "розумний" будинок - підвищення комфорту за рахунок збільшення рівня автоматизації рутинних процесів. Тобто, в першу чергу, система повинна бути зручною у використанні і вимагати найменшу кількість маніпуляцій з боку користувача. Для досягнення цієї мети "розумний" будинок визначається такими основними параметрами:

- **Взаємодія.** Особливість системи "розумного" будинку полягає в її здатності об'єднувати різні пристрої в єдину систему. Злагоджена робота пристроїв може бути організована просто чи складно залежно від "відкритості" системи автоматизації. Найбільш відкритою системою вважається та, де взаємодія пристроїв проходить максимально легко. Для підтримки взаємодії між декількома електронними пристроями виробники систем "розумний" будинок дуже часто укладають партнерські відносини між собою. Це дозволяє більш серйозно підійти до питання інтеграції всіх систем будинку: від архітектурного освітлення та поливу газону до забезпечення роботи домашнього кінотеатру. Ще один спосіб взаємодії - це робота на основі технологічних стандартів. Багато

виробників впроваджують у свої продукти бездротове управління на базі технології Z-Wave [4]. Саме цей загальний елемент дозволяє пристроям чудово працювати один з одним. Чим більше у провайдера партнерів, тим ширше асортимент для вашого вибору в якості клієнта;

- Віддалений доступ. Користувачам системи потрібна можливість швидко і легко змінювати налаштування за необхідності. Дуже часто зробити це потрібно, коли ви знаходитесь не вдома. Саме тому однією з найбільш важливих особливостей системи "розумний" будинок є можливість віддаленого управління і доступу до системи. Він дозволяє контролювати що відбувається в будинку і навколо нього, змінювати налаштування освітлення, термостатів та іншого обладнання за допомогою ноутбука, смартфона або планшета. Віддалений доступ також дозволяє установникові налаштувати систему без необхідності його присутності в будинку, що підвищує зручність і рівень сервісу;

- Масштабованість. Необхідність цього параметра визначається тим, що технології постійно розвиваються, виплескуючи на ринок товари нового покоління. У майбутньому це дає можливість додати нові приміщення до системи без необхідності купувати нову систему. Також масштабованість дає можливість користувачеві самому визначати необхідні сенсори і функції системи при цьому не позбавляючи можливість додати їх у майбутньому. З цих та інших причин дуже важливо, щоб в "розумному" будинку можна було додавати нові функції і пристрої (вертикальне розширення) або нові приміщення (горизонтальне розширення). Виробники часто підтримують обидва типи розширення за допомогою розробки системи на одній мові мережі, наприклад IP (Internet Protocol), а також можливістю бездротового дооснащення продуктами, які можуть взаємодіяти за допомогою існуючої домашньої мережі або провідних пристроїв.

Наведені параметри є основою концепції "розумний" будинок, при цьому повна або часткова реалізація їх лежить на виробниках і може залежати від

призначення системи та специфіки регіону установки. Обмежень реалізації жорсткими рамками немає, стандартизація стосується тільки протоколів взаємодії пристроїв між собою. Сама реалізація залишається на розсуд компаній і не стандартизується.

## 1.4 Існуючі концепції

В ідеальному варіанті "розумний" будинок є системою, в якій кожен прилад інтегрований в загальну екосистему, проте, зважаючи на різні протоколи зв'язку приладів, необхідності у здешевленні системи виробником та налаштування під конкретні завдання відбувається поділ на три основні групи за призначенням:

- Мультимедійний простір;
- Система розумного контролю мікрокліматичних параметрів приміщення;
- Змішана система.

Мультимедійне простір або мультимедійна система "розумний" будинок - це цілісна екосистема з мультимедійних пристроїв з загальним хабом пам'яті для контенту і розширеними можливостями взаємодії.

Ця концепція заснована на переважаючій функції контенту і складається з таких компонентів як Smart TV з доступом до Інтернет ресурсів і додатків, мультимедіа хаб, система об'ємного звуку, просунуті функції енергозбереження для техніки, віддалений контроль пристроїв. З появою готових пристроїв Smart TV і хмарних сервісів концепція може бути легко реалізована будь-яким користувачем без необхідності докуповувати щось окрім самих пристроїв. Є найбільш популярною в США і Канаді.

З появою доповненої реальності концепція мультимедійного "розумного" будинку знайшла нове життя у вигляді так званих шоломів віртуальної реальності. До основних функцій була додана модуляція віртуальних об'єктів на реальність і можливість взаємодіяти з результатом як з цілісною

середовищем. Варто відзначити, що тепер для побудови системи потрібно всього лише один пристрій без додаткових аксесуарів і функціональність системи залежить тільки від вбудованого в шолом ПЗ. Приклад мультимедійної системи розумного будинку з доповненою реальністю на рис. 1.4.



Рисунок 1.4 – Приклад роботи Microsoft HoloLens [5]

Система розумного контролю мікрокліматичних параметрів приміщення - це система, заснована навколо ідеї створення найбільш оптимальних параметрів мікроклімату з найменшими енерговитратами. У даному випадку під контроль автоматики передаються різні кліматичні пристрої, а також джерела світла і опційно - інша побутова електроніка. Сама концепція системи будується навколо об'єднання перерахованої техніки в єдину екосистему, що налаштовується. По суті, система представляє з себе набір керованих алгоритмів, які запускаються при настанні стартової умови. Наприклад, при недостатній освітленості загоряються додаткові світильники, або система опалення працює на мінімумі поки користувач не вдома.

Змішана система - це система, яка має функціонал обох попередніх. Він може бути реалізований як у повній мірі, так і частково. В основному будується навколо центрального контролюючого пристрою, щоб була можливість

управління різними типами пристроїв по різних протоколах.

## **1.5 Переваги та недоліки існуючих рішень**

Рішення від Meizu. "Розумний" будинок від Meizu - це сукупність різних смарт-девайсів від різних компаній, які об'єднані єдиним ПЗ (LifeKit) та вимогами Meizu. На сьогодні в "розумну" екосистему потрапили такі смарт-девайси, як ваги RyFit (32 \$), лампа X-Light Plus (19 \$), розетки, очищувач повітря Air Cube та інше. Система відноситься до змішаного типу. Переваги: наявність готових пристроїв, простота встановлення і налаштування, функції керуючого пристрою бере на себе телефон. Недоліки: порівняно малий функціонал, обмежений вибір устаткування, прив'язка до пристроїв від Meizu, сумарна дорожнеча системи.

Рішення від Allone або Allone WiFi Smart Remote Control. Являє собою центральний контролер, керуючий підтримуваними приладами через wi-fi [6]. Ціна 48 \$. Переваги: велика кількість модулів, масштабованість. Недоліки: порівняно висока ціна модулів, не підходить для великих будинків з товстими стінами через використання wi-fi для управління пристроями.

Рішення від Clipsal. Являє собою невеликі модулі з сенсором, які підключаються до електроніки і керують нею за рахунок вбудованого алгоритму [7]. Переваги: низька вартість, масштабованість системи, немає потреби купувати техніку з вбудованими функціями взаємодії. Недоліки: відносно складне налаштування, низька взаємодія компонентів один з одним. Порівняння рішень наведено у таблиці 1.1.

Таблиця 1.1 – Порівняння існуючих варіантів реалізації системи "розумний" будинок.

Параметр	Meizu	Allone	Clipsal
Вартість	Середня	Висока (разом з модулями)	Невелика за модуль, середня за систему в цілому
Установка	Проста	Проста	Потребує попереднього налаштування
Налаштування	Не потребує	Через веб-сервіс	Через прошивку
Готові модулі	Мало	Багато	Не потребує
Масштабованість	Масштабується	Масштабується, але залежить від розміру приміщення	Масштабується
Взаємодія компонентів	Через смартфон	Через сам пристрій	Через хмарний додаток
Функціонал	Базовий	Майже необмежений	Майже необмежений

## 1.6 Власний варіант реалізації

Варіант реалізації SHome представляє з себе систему розумного контролю мікрокліматичних параметрів з вбудованими функціями енергозбереження та прогнозу енергоспоживання. Система відноситься до класу з вбудованим центральним контролером. Архітектура модульного типу. Розширюваність забезпечується за рахунок сумісних с Arduino модулів, однак для підхоплення нових сенсорів необхідно завантажити нові налаштування на Arduino. Система складається з трьох частин: модулі керування кліматотехніки на основі

Arduino, центральний контролер і мобільний додаток для управління системою та відображення інформації у візуально-зрозумілому вигляді. Відмінність від інших рішень полягає у використанні тільки вільних компонентів під вільною ліцензією як у випадку з апаратним забезпеченням, так і з програмним. Абсолютно будь-який користувач може налаштувати систему під себе, використовуючи недорогі компоненти для Arduino і малопотужний пристрій в якості центрального контролера, не написавши при цьому жодного рядка коду.

Основна мета проекту - надати користувачам недорогу систему на безоплатній основі з просунутими можливостями в налаштуванні енергетичних планів і кастомізації. Необхідне для розгортання реалізації від SHome на рис. 1.5.



Рисунок 1.5 – Все необхідне для розгортання системи SHome

## 1.7 Висновки

Отже, "розумний" будинок - житловий будинок сучасного типу, організований для проживання людей за допомогою автоматизації і високотехнологічних пристроїв.

За принципом побудови буває:

- Вбудовані системи з центральним контролером
- Вбудовані системи без центрального контролера
- Системи з інтеграцією, що налаштовується.

За концепцією:

- Мультимедійний простір
- Система розумного контролю мікрокліматичних параметрів приміщення
- Змішана система

Основні характеристики:

- Взаємодія
- Масштабованість
- Віддалений доступ

Основні риси власної системи SHome:

- Енергозбереження
- Енергопрогнозування
- Компоненти під вільною ліцензією
- Кастомізація



## **2. АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ**

Реалізація концепції "розумного" будинку досить складна задача. Для побудови та тестування такої системи необхідна значна кількість різних датчиків, модулів та пристроїв. Розглянемо компоненти, необхідні для виконання поставленої задачі.

### **2.1 Аналіз можливостей мікроконтролерів та мікрокомп'ютерів**

Мікроконтролер або мікрокомп'ютер - один з центральних елементів системи "розумного" будинку. Він відповідає за керування різними пристроями навантаження, такими як обігрівачі, освітлення, установки зволоження повітря та інші, а також знімає показники з різноманітніших датчиків. Розглянемо деякі варіанти мікроконтролерів та мікрокомп'ютерів.

#### **2.1.1 Arduino**

Arduino – апаратна обчислювальна платформа, основними компонентами якої є плата вводу/виводу та середовище розробки на мові Processing/Wiring. Arduino може використовуватися як для створення автономних інтерактивних об'єктів, так і підключатися до програмного забезпечення, яке виконується на комп'ютері (наприклад: Adobe Flash, Processing, Max/MSP, Pure Data, SuperCollider).

##### **2.1.1.1 Arduino Uno**

Arduino Uno є плата мікроконтролера на базі ATmega328. Його зображення наведено на рис. 2.1 та рис. 2.2, характеристики у таблиці 2.1 [8].

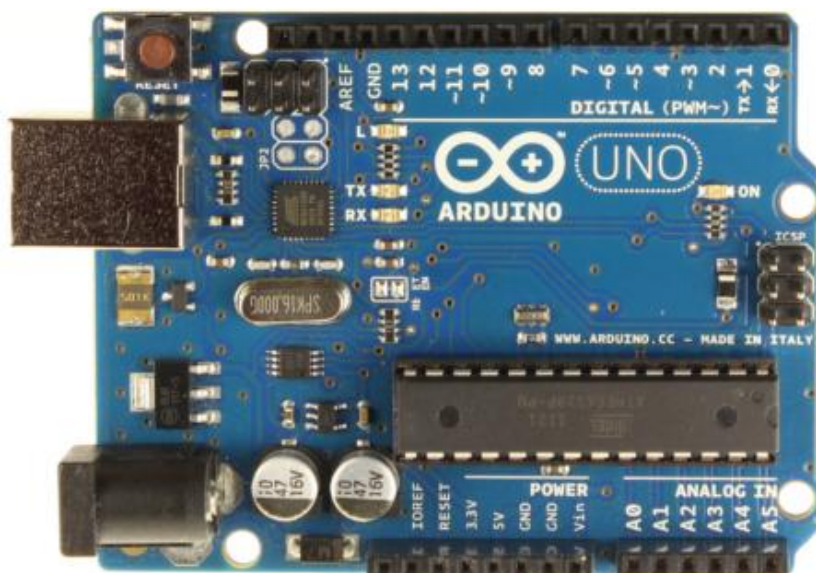


Рисунок 2.1 – Arduino UNO вид зверху [8]

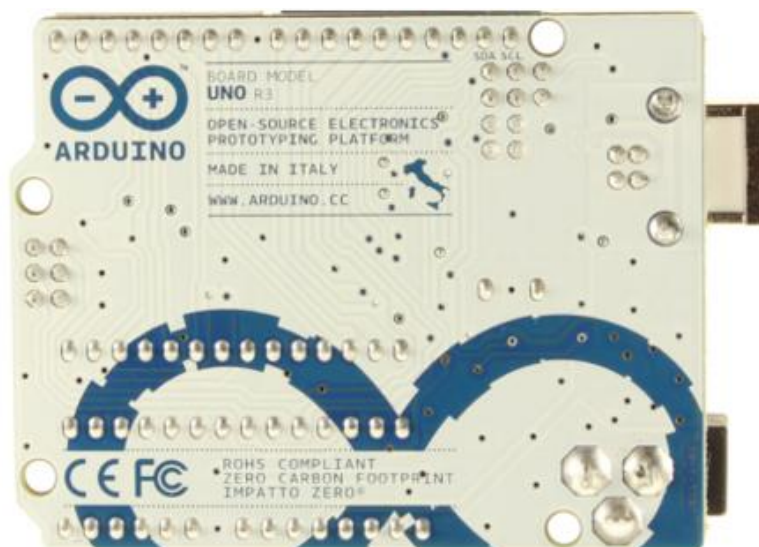


Рисунок 2.2 – Arduino UNO вид знизу [8]

Таблиця 2.1 – Характеристики Arduino UNO

Параметр	Значення
Мікроконтролер	ATmega328
Робоча напруга	5В
Вхідна напруга (рекомендований)	7-12В
Вхідна напруга (граничне)	6-20В
Цифрові Входи / Виходи	14 (з них 6 можуть використовуватися в якості ШІМ-виходів)
Аналогові входи	6
Постійний струм через вхід / вихід	40 mA
Постійний струм для виведення 3.3 В	50 mA
Флеш-пам'ять	32 КБ з яких 0.5 КБ використовуються завантажувачем
ОЗУ	8 КБ
Незалежна пам'ять	4 КБ
Тактова частота	16 MHz

Вартість плати на сайті aliexpress.com складає \$2.91 [9].

### 2.1.1.2 Arduino Mega2560

Arduino Mega 2560 мікроконтролер на основі процесора ATmega2560. Його зображення наведено на рис. 2.3 та рис. 2.4, характеристики у таблиці 2.2 [10].

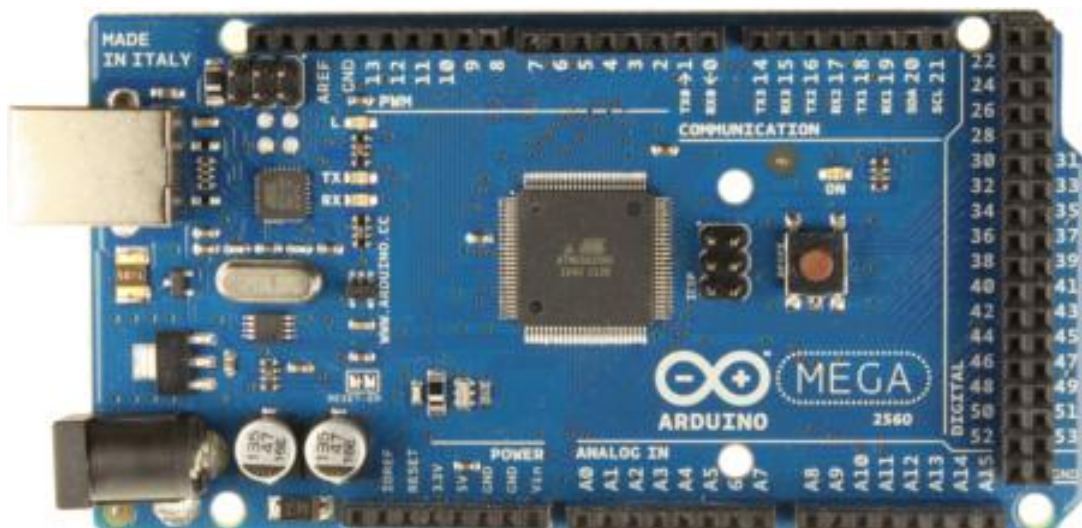


Рисунок 2.3 – Arduino Mega 2560 R3 вид сверху [10]

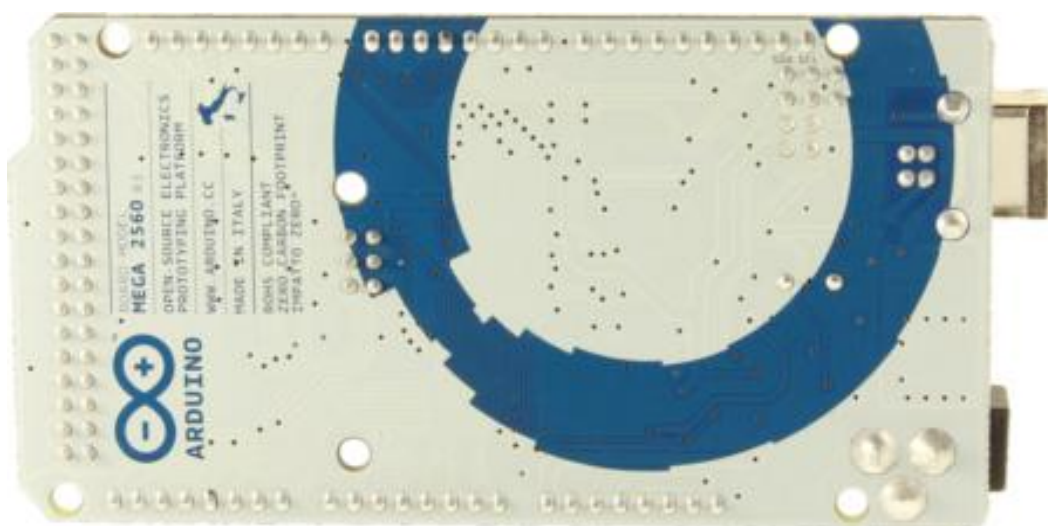


Рисунок 2.4 – Arduino Mega2560 R3 вид снизу [10]

Таблиця 2.2 – Характеристики Arduino Mega2560 R3

Параметр	Значення
Мікроконтролер	ATmega2560
Робоча напруга	5В
Вхідна напруга (рекомендований)	7-12В
Вхідна напруга (граничне)	6-20В
Цифрові Входи / Виходи	54 (14 з яких можуть работат також як виходи ШІМ)
Аналогові входи	16
Постійний струм через вхід / вихід	40 mA
Постійний струм для виведення 3.3 В	50 mA
Флеш-пам'ять	256 КВ (з яких 8 КВ використовуються для завантажувача)
ОЗУ	8 КВ
Незалежна пам'ять	4 КВ
Тактова частота	16 MHz

Вартість плати на сайті aliexpress.com складає \$11.29 [9].

## 2.1.2 ChipKit

### 2.1.2.1 Uno32

ChipKIT Uno32 заснований на популярній апаратній платформі Arduino прототипів, але додає продуктивності за рахунок мікроконтролера Microchip PIC32. Його зображення наведено на рис. 2.5 та рис. 2.6, характеристики у таблиці 2.3 [11].



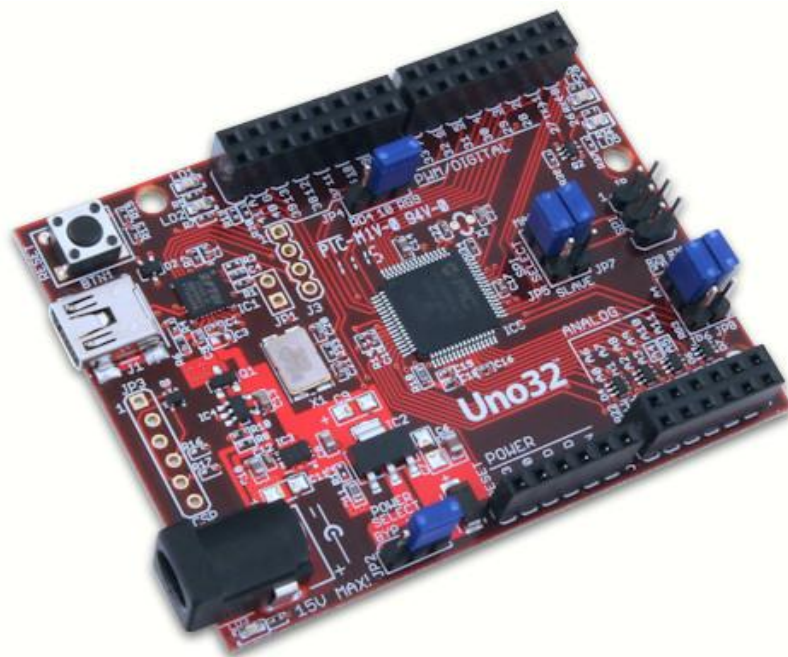


Рисунок 2.5 – ChipKIT Uno32 вид зверху [11]

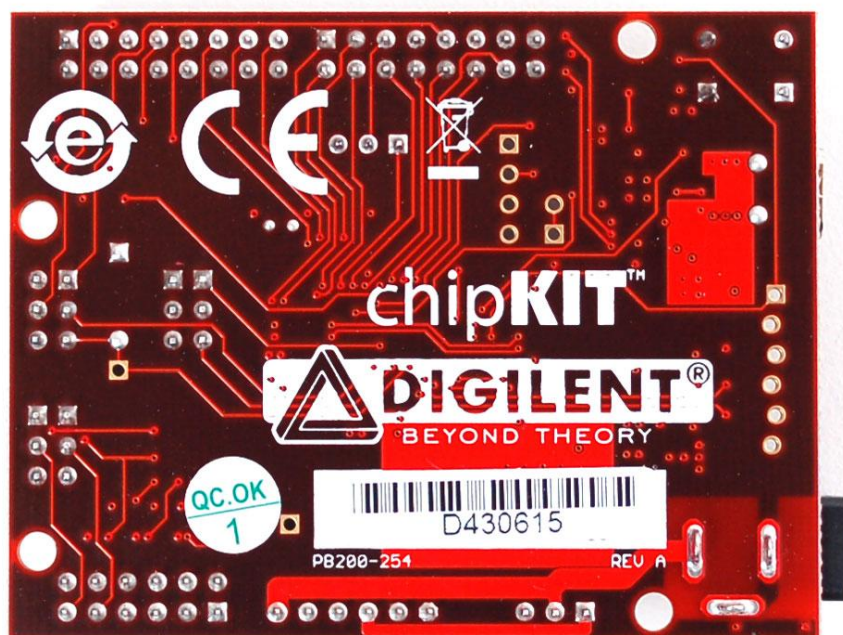


Рисунок 2.6 – ChipKIT Uno32 вид знизу [11]

Таблиця 2.3 – Характеристики ChpKIT Uno32

Параметр	Значення
Мікроконтролер	PIC32MX320F128H
Робоча напруга	3.3В
Вхідна напруга (рекомендований)	7-15В
Вхідна напруга (граничне)	20В
Цифрові Входи / Виходи	42
Аналогові входи	12
Постійний струм через вхід / вихід	75 mA
Флеш-пам'ять	128 KB
ОЗУ	16 KB
Тактова частота	80 MHz

Вартість плати на сайті [www.digilentinc.com](http://www.digilentinc.com) складає \$26.95 [11].

### 2.1.2.2 Max32

ChpKIT Max32 заснований на популярній апаратній платформі Arduino, але додає продуктивності за рахунок мікроконтролера мікроконтролера Microchip PIC32. Max32 того ж форм-фактора, як Arduino Mega. Його зображення наведено на рис. 2.7 та рис. 2.8, характеристики у таблиці 2.4 [12].

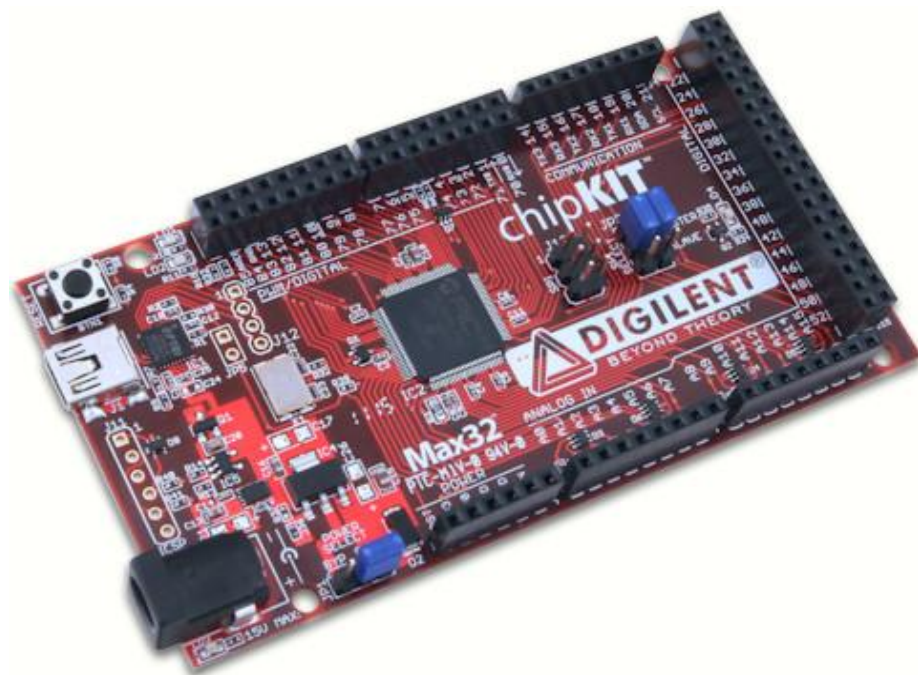


Рисунок 2.7 – ChipKIT Max32 вид сверху [12]



Рисунок 2.8 – ChipKIT Max32 вид снизу



Таблиця 2.4 – Характеристики ChipKIT Max32

Параметр	Значення
Мікроконтролер	PIC32MX795F512L
Робоча напруга	3.3В
Вхідна напруга (рекомендований)	7-15В
Вхідна напруга (граничне)	20В
Цифрові Входи / Виходи	83
Аналогові входи	16
Постійний струм через вхід / вихід	75 mA
Флеш-пам'ять	512 KB
ОЗУ	128 KB
Тактова частота	80 MHz

Вартість плати на сайті [www.digilentinc.com](http://www.digilentinc.com) складає \$49.5 [12].

### 2.1.3 Raspberry PI

Raspberry Pi) – одноплатний комп'ютер розміром з банківську карту, спочатку розроблений як бюджетна система для навчання інформатиці. Згодом отримав набагато більш широке застосування і популярність, ніж очікували його автори.

#### 2.1.3.1 Model A+

Зображення Raspberry PI Model A+ наведено на рис. 2.9 та рис. 2.10, характеристики у таблиці 2.5 [13].



Рисунок 2.9 – Raspberry PI Model A+ вид зверху [13]

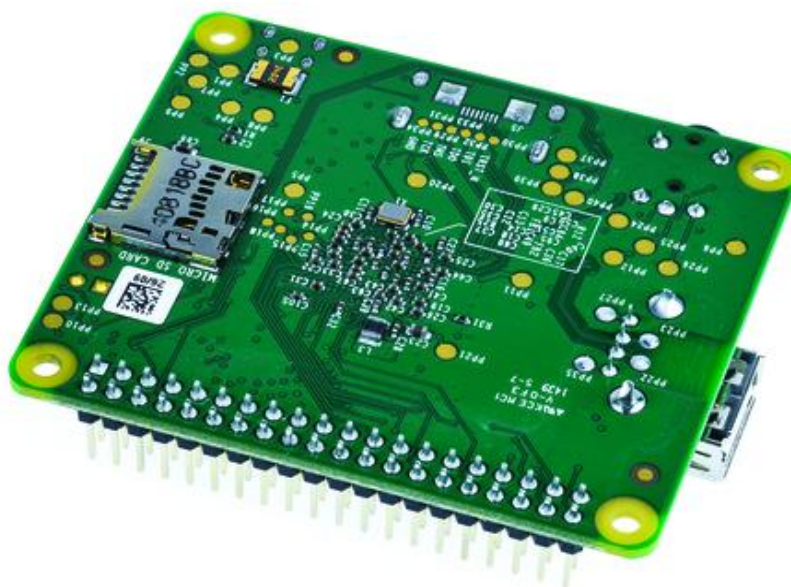


Рисунок 2.10 – Raspberry PI Model A+ вид знизу [13]

Таблиця 2.5 – Характеристики Raspberry PI Model A+

Параметр	Значення
Чіп	Broadcom BCM2835
Архітектура ядра	ARM11
ЦПУ	700 MHz Low Power ARM1176JZFS Applications Processor
ОЗУ	256MB SDRAM
Локальна мережа	-
GPIO	40-pin
Живлення	Micro USB socket 5V, 2A

Вартість плати на сайті [www.raspberrypi.org](http://www.raspberrypi.org) складає \$25 [13].

### 2.1.3.2 Model B+

Зображення Raspberry PI Model B+ наведено на рис. 2.11 та рис. 2.12, характеристики у таблиці 2.6 [14].



Рисунок 2.11 – Raspberry PI Model B+ вид зверху [14]

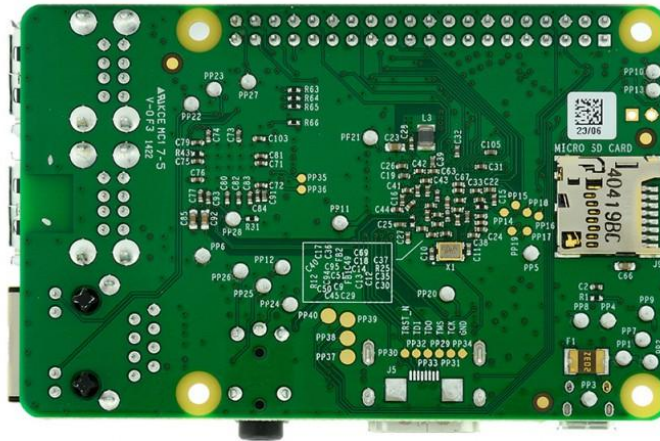


Рисунок. 2.12 – Raspberry PI Model B+ вид знизу [14]

Таблиця 2.6 – Характеристики Raspberry PI Model B+

Параметр	Значення
Чіп	Broadcom BCM2835 SoC
Архітектура ядра	ARM11
ЦПУ	700 MHz Low Power ARM1176JZFS Applications Processor
ОЗУ	512MB SDRAM
Локальна мережа	10/100 BaseT Ethernet socket
GPIO	40-pin
Живлення	Micro USB socket 5V, 2A

Вартість плати на сайті [www.raspberrypi.org](http://www.raspberrypi.org) складає \$35 [14].

### 2.1.3.3 Model 2

Зображення Raspberry PI Model 2 наведено на рис. 2.13 та рис. 2.14, характеристики у таблиці 2.5 [15].



Рисунок 2.13 – Raspberry PI Model 2 вид зверху [15]

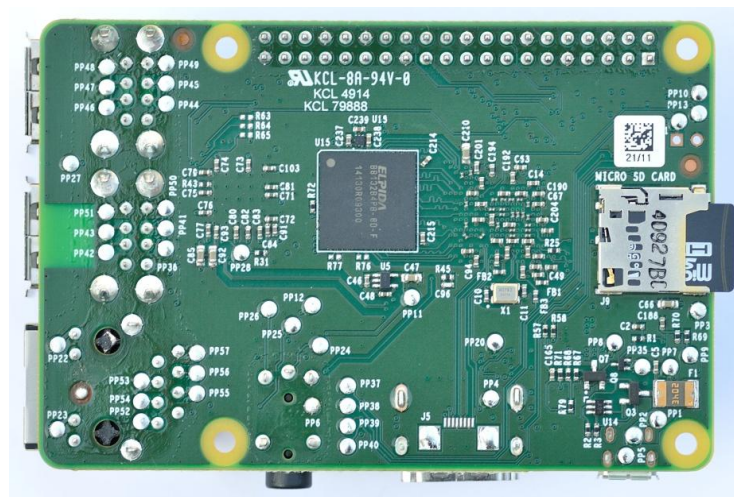


Рисунок 2.14 – Raspberry PI Model 2 вид знизу [15]

Таблиця 2.7 – Характеристики Raspberry PI Model 2

Параметр	Значення
Чіп	Broadcom BCM2836 SoC
Архітектура ядра	Quad-core ARM Cortex-A7
ЦПУ	900 MHz
ОЗУ	1GB LPDDR2
Локальна мережа	10/100 BaseT Ethernet socket
GPIO	40-pin
Живлення	Micro USB socket 5V, 2A



Вартість плати на сайті [www.raspberrypi.org](http://www.raspberrypi.org) складає \$37 [15].

#### **2.1.4 Висновки щодо мікроконтролера/мікрокомп'ютера**

Було розглянуто деякі мікроконтролери та мікрокомп'ютери 3-х виробників.

Аналоги від ChipKit коштують на рівні мікрокомп'ютерів, до того ж у них відсутній Ethernet роз'єм, що робить їх використання економічно не вигідним.

Мікрокомп'ютери від Raspberry, мають високу продуктивність та можливості, але найдешевша з них майже у 2.5 рази дорожчий за Arduino Mega 2560 і не має Ethernet роз'єму. Версії ж з вбудованим Ethernet роз'ємом коштують приблизно у 3.3 рази дорожче.

Arduino Uno найдешевший з представлених варіантів, але у нього і найменший обсяг вбудованої пам'яті, що може стати перешкодою для реалізації проекту.

Таким чином, з наведених варіантів найбільш доцільним для використання в проекті є мікроконтролер Arduino Mega 2560. Він має достатню кількість входів/виходів та достатній обсяг вбудованої пам'яті, а його вартість одна з найменших. Для реалізації проекту нам також знадобиться модулі для роботи з Ethernet так картами пам'яті.

## **2.2 Аналіз можливостей периферійних пристроїв**

### **2.2.1 W5100 Ethernet Shield**

Для реалізації підключення мікроконтролера до локальної мережі використаємо плату розширення W5100 Ethernet Shield з вбудованою можливістю взаємодії з картами пам'яті microSD. Його зовнішній вигляд наведено на рис. 2.15 та рис. 2.16.

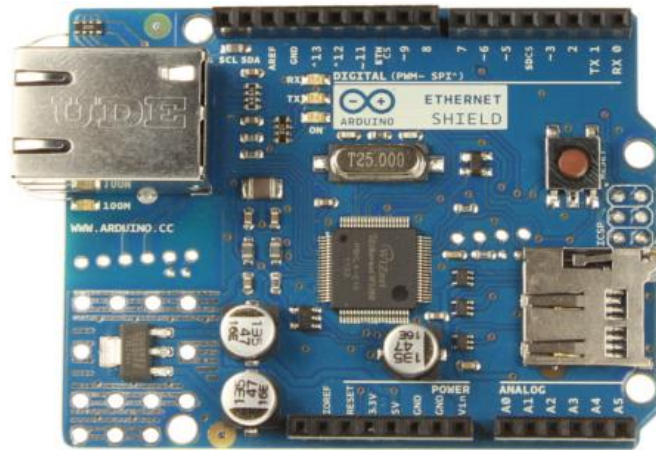


Рисунок 2.15 – Плата розширення W5100 Ethernet Shield вид зверху [16]

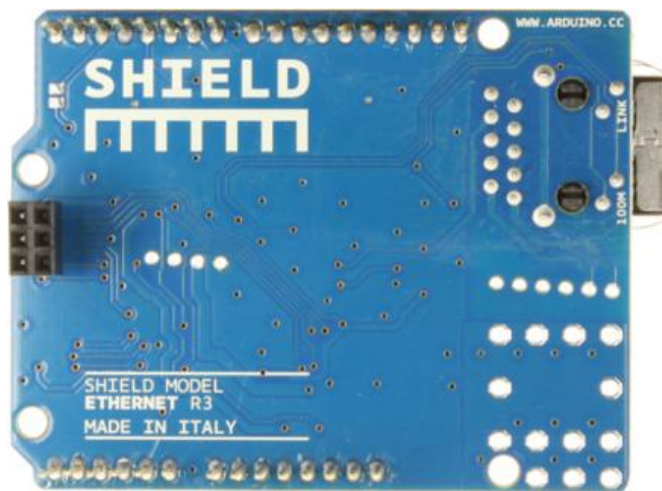


Рисунок 2.16 – Плата розширення W5100 Ethernet Shield вид знизу [16]

Характеристики [16]:

- Робоча напруга 5В (подається від плати Ардуіно)
- Ethernet-контролер W5100 з вбудованим буфером об'ємом 16 КБ
- Швидкість з'єднання 10/100 Мбіт
- Взаємодія з Ардуіно здійснюється через інтерфейс SPI

Вартість модуля на сайті [aliexpress.com](http://aliexpress.com) складає \$5.22 [9].

### 2.2.2 SD Card Reader

Так як файли з налаштуваннями ми будемо зберігати на карті пам'яті SD, то нам буде потрібен модуль для роботи з нею. Зовнішній вигляд такого модуля наведений на рис. 2.17.

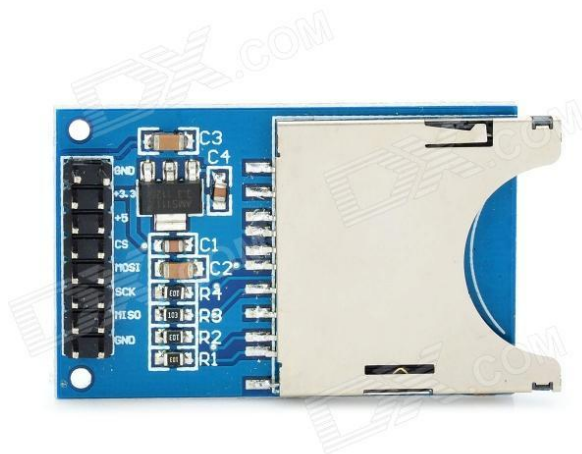


Рисунок 2.17 – SD Card Reader [9]

Вартість модуля на сайті [aliexpress.com](http://aliexpress.com) складає \$0.84 [9].

Для взаємодії з мікроконтролером використовується SPI шина.

### 2.2.3 Реле SONGLE SRD-05VDC

Дане реле управляється напругою 5V і здатне комутувати до 10A 30V DC і 10A 250V AC. Зовнішній вигляд наведений на рис. 2.18 та рис. 2.19.





Рисунок 2.18 – SONGLE SRD-05VDC вид зверху [9]



Рисунок 2.19 – SONGLE SRD-05VDC вид знизу [9]

Вартість модуля на сайті aliexpress.com складає \$1.06 [9].

## 2.3 Аналіз можливостей датчиків

### 2.3.1 DS1820/DS18B20/DS18S20

Сімейство датчиків DS1820/DS18B20/DS18S20 використовується для отримання вимірювань температури. Вони відрізняються точністю, робочими діапазонами та робочою напругою. Схема підключення у всіх трьох однакова. Для тестування буде використано датчик DS18B20. Зовнішній вигляд наведено на рис. 2.20, характеристики у таблиці 2.8 [17].



Рисунок 2.20 – Датчик DS1820 [17]

Таблица 2.8 – Характеристики DS1820/DS18B20/DS18S20

Виріб	DS1820	DS18S20	DS18B20
Маркування	DS1820	DS1820	DS18B20
Корпус	PR-35 (подовжений TO-92)	TO-92	TO-92
Розрядність	9-біт	9-біт	9-12біт
Час перетворення	200mS (тип.) 500nS(max)	750nS (max)	750nS(max)
Точність вимірювання $\pm 0.5\%$ в області температур	0 ....+70°C	-10 ....+85°C	-10 ....+85°C
Напруга живлення для точності вимірювання $\pm 0.5\%$	4,3-5,5V	3,0-5,5V	3,0-5,5V

Вартість датчика DS18B20 на сайті aliexpress.com складає \$0.7 [9].

### 2.3.2 DHT11/DHT22/DHT21

Датчики сімейства DHT11/DHT22/DHT21 є об'єднаними датчиками температури та вологості. Вони відрізняються точністю вимірювань та рівнем захищеності від впливу довкілля. Для тестування буде використано датчик DHT22. Його зовнішній вигляд наведено на рис. 2.21, а характеристики у

таблиці 2.9 [18].

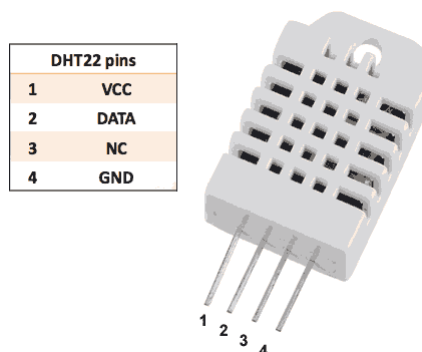


Рисунок 2.21 – Датчик DHT22 [18]

Таблиця 2.9 – Характеристики DHT11/DHT22/DHT21

	<b>DHT11</b>	<b>DHT22</b>	<b>DHT21</b>
Діапазон вимірювання температур	0..60°C, ±2%	-40...+80°C, ±0.5°C	-40...+125°C, ±0.5°C
Діапазон вимірювання вологості	20..90%, ±2%	0-100%, ±2%	0-100%, ±2%
Вага, гр.	1	2.2	11
Округлення результатів	до цілих	до десятих	до десятих
Споживання струму при 5В	0,1	0,14	1,36
Умови використання	в приміщенні	в приміщенні	вулиця

Вартість датчика DHT22 на сайті aliexpress.com складає \$2.58 [9].

### 2.3.3 BMP180

BMP180 - це цифровий датчик тиску та температури. Зовнішній вигляд наведено на рис. 2.22.

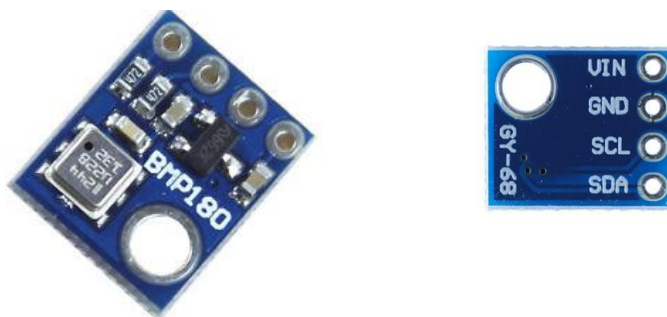


Рисунок 2.22 – Датчик BMP180 [19]

Характеристики [19]:

- Діапазон вимірювання тиску: 300-1100 гПа (9000m для -500m над рівнем моря) до 0.03hPa / 0,25 м
- Діапазон вимірювання температури: -40 До + 85 ° С з точністю + -2 ° С

Вартість датчика на сайті [aliexpress.com](http://aliexpress.com) складає \$1.26 [9].

### 2.3.4 Фоторезистор СФ2-5а

Для зняття показників освітленості скористаємось фоторезистором СФ2-5а зовнішній вигляд наведено на рис. 2.23, характеристики у таблиці 2.10 [20].

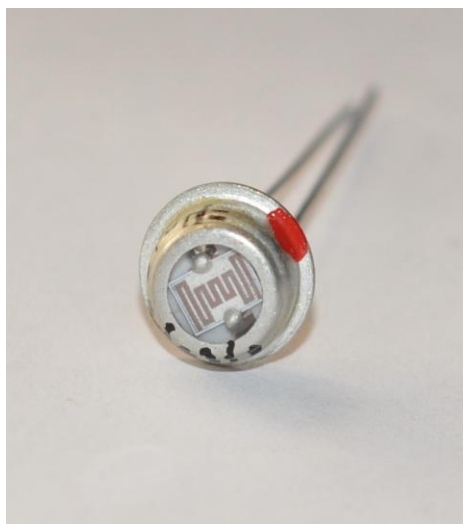


Рисунок 2.23 – Фоторезистор СФ2-5а

Таблиця 2.10 – Характеристики СФ2-5а

Параметр	Значення
Робоча напруга, В	1,3
Максимальна напруга, В	10
Загальний струм при освітленості 1 лк, мкА	5,0...50
Загальний струм при освітленості 200 лк, мкА, не менше	500

## 2.4 Схема тестового стенду

Для тестування роботи системи був створений тестовий стенд рис. 2.24.

Під час тестування буди використані наступні компоненти:

- Arduino Mega 2560
- W5100 Ethernet Shield
- SD Card Reader
- BMP180
- DHT22 (2 шт.)
- DS18B20
- Світлодіоди (5 шт)
- Резистори (4.7 кОм)

- Фоторезистор (2 шт.)

Схема підключення наведена на рис. 2.25.

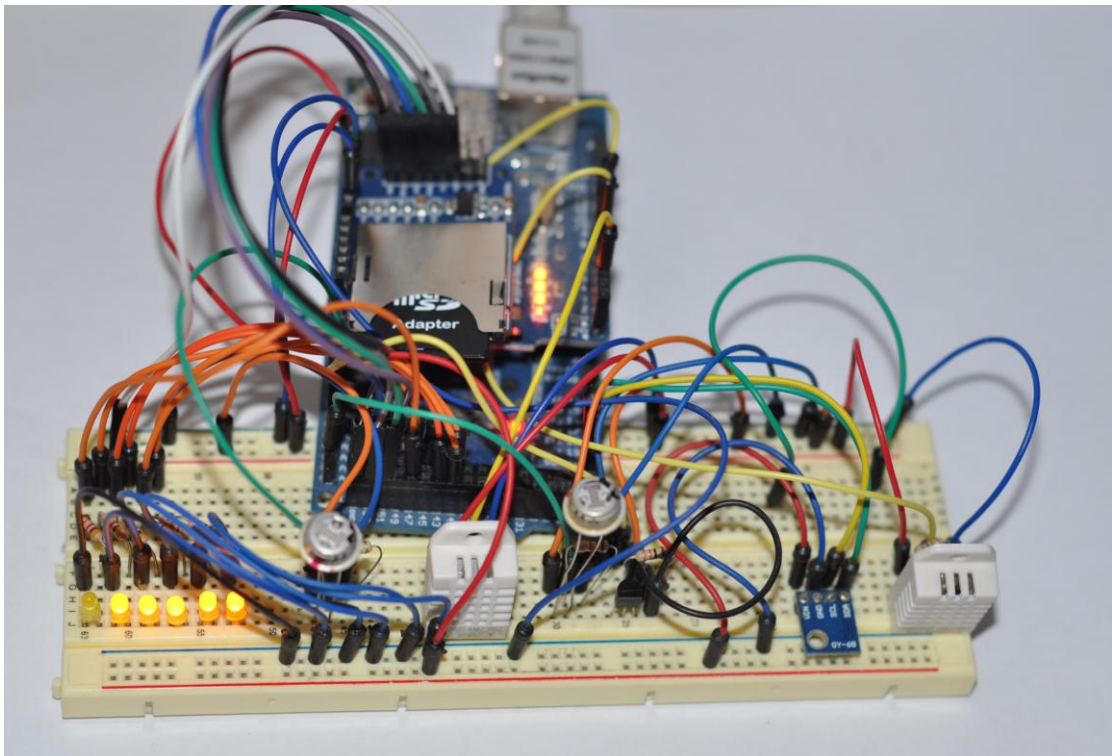


Рисунок 2.24 – Тестовий стенд

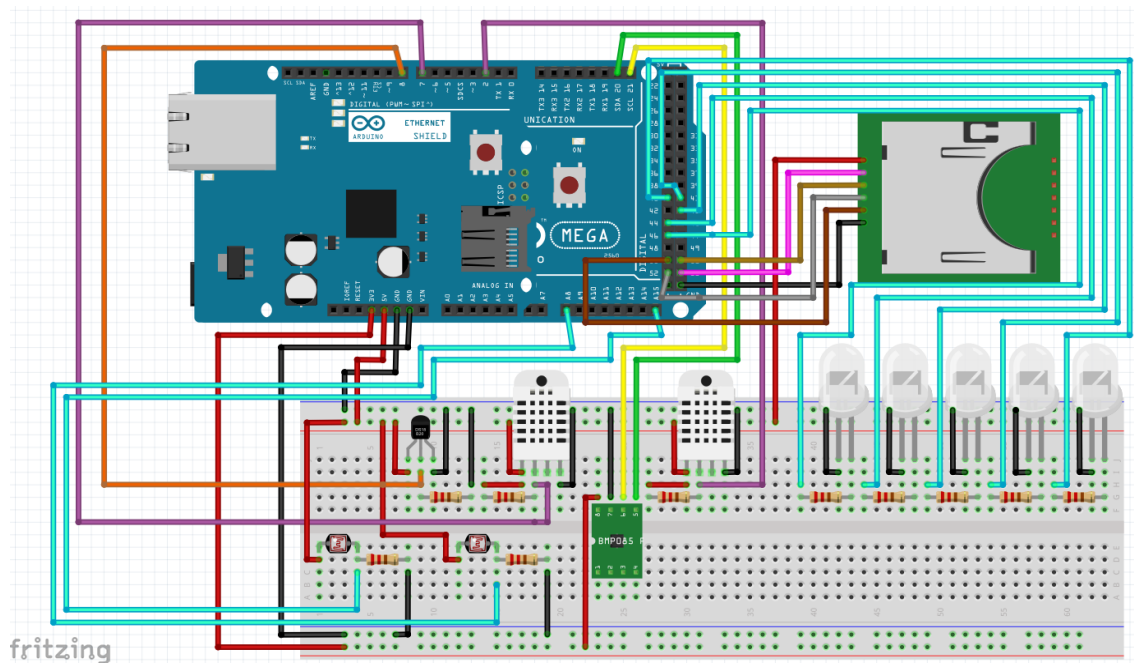


Рисунок 2.25 – Схема підключення

Підключення слід виконувати так чином:

Модуль W5100 Ethernet Shield встановлюється зверху на плату Arduino Mega 2560.

Модуль SD Card Reader підключається до мікроконтролера так:

- +5 до роз'єму живлення 5В
- CS до 52-го порту
- MOSI до 51-го порту
- SCK до 52-го порту
- MISO до 50-го порту
- GND до порту GND

Датчики DHT22 підключаються до 2-го та 7-го порту.

Датчик DS18B20 до 8-го порту.

Датчик BMP180 підключимо таким чином:

- VCC до роз'єму живлення 3.3В
- GND до порту GND
- SCL до 21-го порту
- SDA до 20-го порту

Фоторезистори підключимо до 8-го та 15-го аналогових портів.

Світлодіоди підключимо до портів 40, 41, 43, 44 та 46.

Якщо ж використовувати реле, то підключати їх потрібно так:

- GND до порту GND
- VCC до роз'єму живлення 5В
- IN до відповідного цифрового порту

## **2.5 ASUS NEXUS 7 2012 16GB**

Для тестування роботи системи скористаємось планшетом ASUS NEXUS 7 2012 16GB. Зовнішній вигляд його наведено на рис. 2.26.



Рисунок 2.26 – ASUS NEXUS 7 2012 16GB

На планшеті встановлена остання версія ОС - Android 5.1.1 Lollipop. Це дозволить протестувати можливість роботу коду з найновішими версіями Android пристроїв.

## 2.6 Висновки

Для реалізація концепції "розумного" будинку було обрано мікроконтролер Arduino Mega 2560, його ціна одна з найнижчих, а характеристики відповідають поставленим вимогам.

Для роботи через протокол TCP/IP та зчитування інформації з карт пам'яті було вирішено використати модулі W5100 Ethernet Shield та SD Card Reader.

Для тестування режими роботи були обрані датчики сімейства DS18B20 та DHT, а також датчик барометричного тиску BMP180 і фоторезистори. В якості емуляції роботи реле були використані світлодіоди. Це дозволило створити тестовий полігон і перевірити на ньому роботу системи.



## 3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Для реалізація роботи мікроконтролера а також взаємодії з ним необхідно розробити відповідне програмне забезпечення. Потрібно створити:

- Програмне забезпечення, яке буде виконуватися на Arduino
- Програмне забезпечення для комунікації між Android та Arduino
- Базу даних та методи роботи з нею
- Файл для створення та редагування файлів конфігурацій для мікроконтролера

Розглянемо поетапно ці завдання.

### 3.1 Програмне забезпечення Arduino

Для роботи мікроконтролера необхідно розробити ПЗ, яке буде виконувати наступні функції:

- Слідкувати за значеннями сенсорів
- Включати та виключати пристрої
- Взаємодіяти з Сервером через протокол TCP/IP
- Зчитувати файл конфігурацій з карти пам'яті та виконувати відповідні налаштування

Для розробки ПЗ скористаємось офіційним середовищем розробки від Arduino. Нам знадобляться наступні бібліотеки:

- SPI.h
- SD.h
- Ethernet.h
- DHT.h
- stdio.h
- string.h
- SFE\_BMP180.h
- Wire.h

- OneWire.h
- DallasTemperature.h

Бібліотека SPI.h необхідна для роботи з SD Card Reader та W5100 Ethernet Shield.

Бібліотека SD.h відповідає за роботу з SD Card Reader [21].

Бібліотека Ethernet.h використовується для взаємодії з W5100 Ethernet Shield [22].

Бібліотека DHT.h необхідна для зв'язку з датчиками сімейства DHT.

Бібліотеки stdio.h та string.h використовуються для роботи зі строками та стандартними функціями.

Бібліотека SFE\_BMP180.h відповідає за взаємодію з датчиком барометричного тиску BMP180.

Бібліотеки OneWire.h та DallasTemperature.h необхідні для роботи з датчиками DS1820/DS18B20/DS18S20.

Передача команд мікроконтролеру та його відповідей буде відбуватися через протокол TELNET, що дозволить легко тестувати роботу системи без необхідності використовувати додаткове ПЗ.

Призначення протоколу TELNET у наданні достатньо спільного, двонаправленого, восьмибітового байт-орієнтованого засобу зв'язку. Його основне завдання полягає в тому, щоб дозволити термінальним пристроям і термінальним процесам взаємодіяти один з одним. Передбачається, що цей протокол може бути використаний для зв'язку виду термінал-термінал ("зв'язування") або для зв'язку процес-процес ("розподілені обчислення").

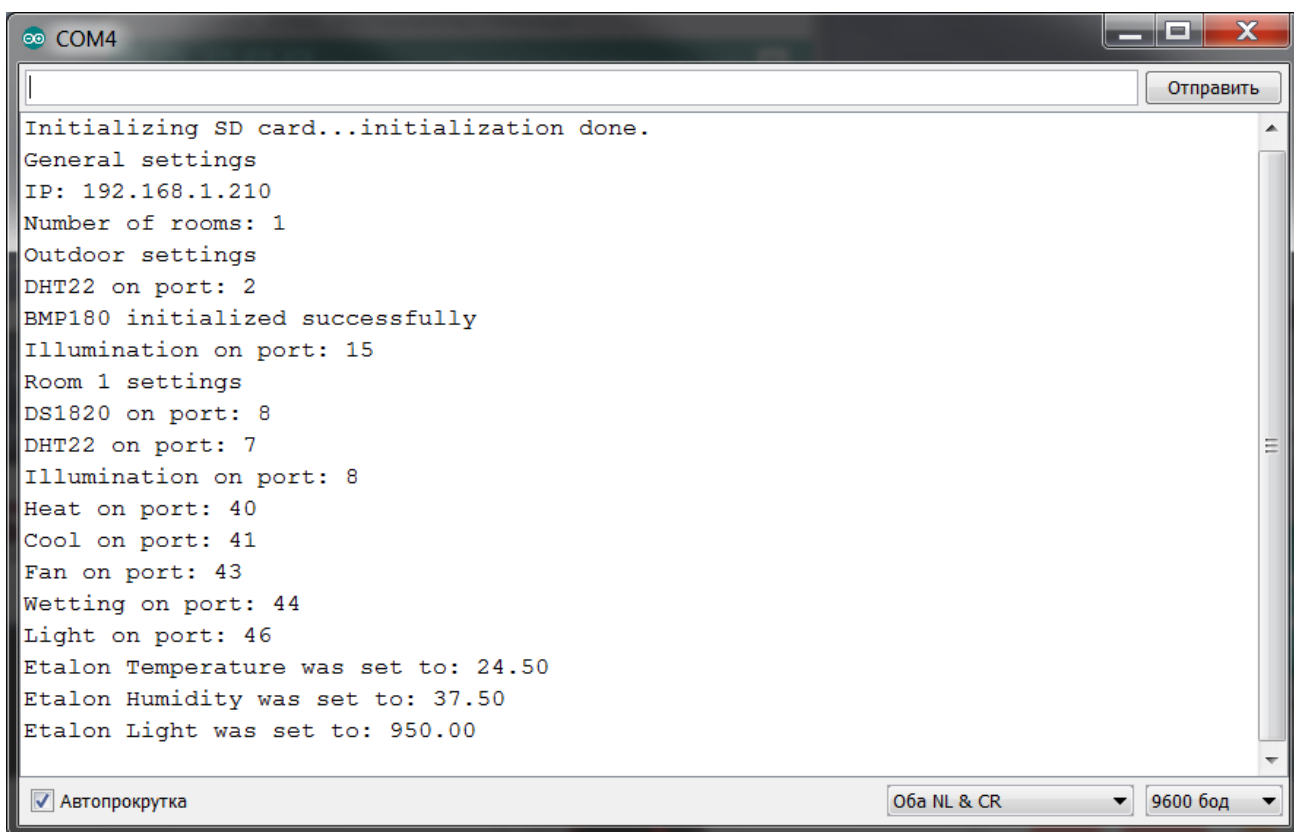
Програмний код скетча включає всього 4 функції:

- run\_config()
- rewrite\_etalon()
- setup()
- loop()

Функції будемо намагатися використовувати менше, щоб отримати максимальну швидкість роботи мікроконтролера. З цієї ж самою причини

відмовимось від використання ООП, на скільки це можливо.

Функція `run_config` виконує зчитування файлу конфігурацій з карти пам'яті та первинне налаштування мікроконтролера для роботи. Цей процес включає лексичний аналіз файлу конфігурацій, створення та заповнення відповідних масивів для збереження даних та взаємодії з сенсорами та засобами керування навантаженнями. Також у цьому сегменті коду відбувається ініціалізація датчиків і зчитування еталонних параметрів. Результат роботи може виводитися на комп'ютер через під'єднання до СОМ порту. Приклад наведено на рис. 3.1.



```
COM4
Initializing SD card...initialization done.
General settings
IP: 192.168.1.210
Number of rooms: 1
Outdoor settings
DHT22 on port: 2
BMP180 initialized successfully
Illumination on port: 15
Room 1 settings
DS1820 on port: 8
DHT22 on port: 7
Illumination on port: 8
Heat on port: 40
Cool on port: 41
Fan on port: 43
Wetting on port: 44
Light on port: 46
Etalon Temperature was set to: 24.50
Etalon Humidity was set to: 37.50
Etalon Light was set to: 950.00
 Автопрокрутка
Оба NL & CR 9600 бод
```

Рисунок 3.1 – Робота функції `run_config`

Функція `rewrite_etalon` виконує видалення файлу та перезапис нових встановлених еталонних функцій.

Функція `setup` запускається при старті мікроконтролера і виконую первині ініціалізації компонентів та запуск функції `run_config`.

Функція `loop` виконується увесь час під час роботи мікроконтролера. В

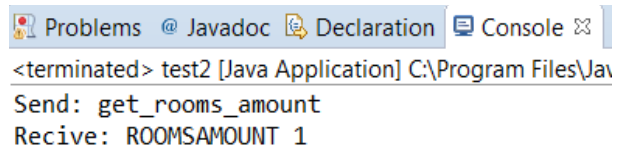
кодi цієї функції відбувається оновлення показників датчиків та ввімкнення або вимкнення пристроїв у відповідності з показниками з сенсорів. Також ця функція відповідає за обробку запитів. Після отримання команди з сервера програма проводить лексичний аналіз команди, правильність введених параметрів. У випадку отримання не коректної команди програма надсилає відповідь про помилку. Список команд:

- logout
- get\_rooms\_amount
- sensor\_request
- set

Команда `logout` використовується для роз'єднання протоколу передачі.

Команда `get_rooms_amount` повертає кількість кімнат, які обслуговуються мікроконтролером.

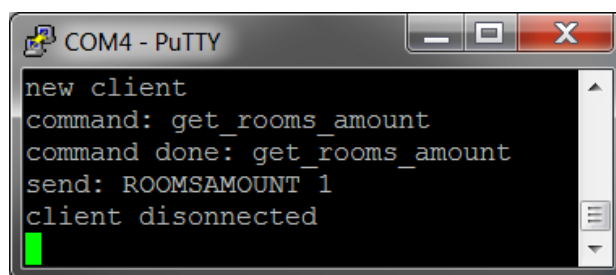
Приклад роботи команд `get_rooms_amount` наведено на рис. 3.2 та 3.3.



```

Problems @ Javadoc Declaration Console
<terminated> test2 [Java Application] C:\Program Files\Jav
Send: get_rooms_amount
Recive: ROOMSAMOUNT 1
  
```

Рисунок 3.2 – Робота команд `get_rooms_amount`



```

COM4 - PuTTY
new client
command: get_rooms_amount
command done: get_rooms_amount
send: ROOMSAMOUNT 1
client disonnected
  
```

Рисунок 3.3 – Робота команд `get_rooms_amount` PUTTY

Команда `sensor_request` виконує передачу значень з датчиків у відповідності до параметрів:

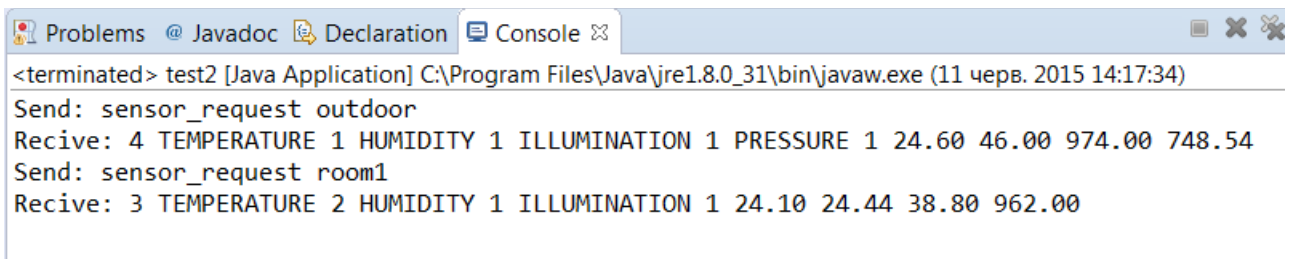
- outdoor

- room%

Параметр outdoor вказує на необхідність надіслати вимірювання зовнішніх датчиків.

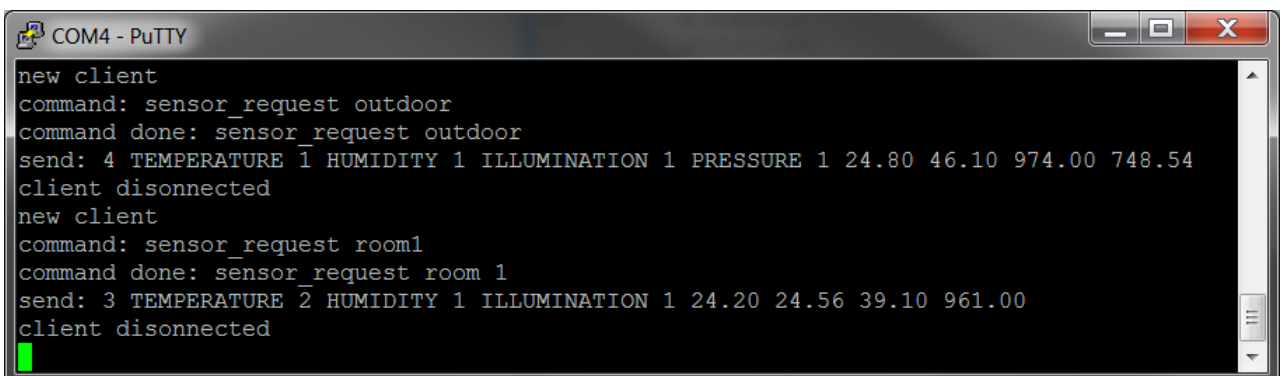
Параметр room% вказує на необхідність надіслати вимірювання датчиків, розташованих в кімнаті, де % - номер кімнати.

Приклад роботи команд sensor\_request наведено на рис. 3.4 та 3.5.



```
<terminated> test2 [Java Application] C:\Program Files\Java\jre1.8.0_31\bin\javaw.exe (11 черв. 2015 14:17:34)
Send: sensor_request outdoor
Recive: 4 TEMPERATURE 1 HUMIDITY 1 ILLUMINATION 1 PRESSURE 1 24.60 46.00 974.00 748.54
Send: sensor_request room1
Recive: 3 TEMPERATURE 2 HUMIDITY 1 ILLUMINATION 1 24.10 24.44 38.80 962.00
```

Рисунок 3.4 – Робота команд sensor\_request



```
COM4 - PuTTY
new client
command: sensor_request outdoor
command done: sensor_request outdoor
send: 4 TEMPERATURE 1 HUMIDITY 1 ILLUMINATION 1 PRESSURE 1 24.80 46.10 974.00 748.54
client disconnected
new client
command: sensor_request room1
command done: sensor_request room 1
send: 3 TEMPERATURE 2 HUMIDITY 1 ILLUMINATION 1 24.20 24.56 39.10 961.00
client disconnected
```

Рисунок 3.5 – Робота команд sensor\_request PUTTY

Команда set використовується для встановлення значень відповідних параметрів:

- temperature
- humidity
- illumination

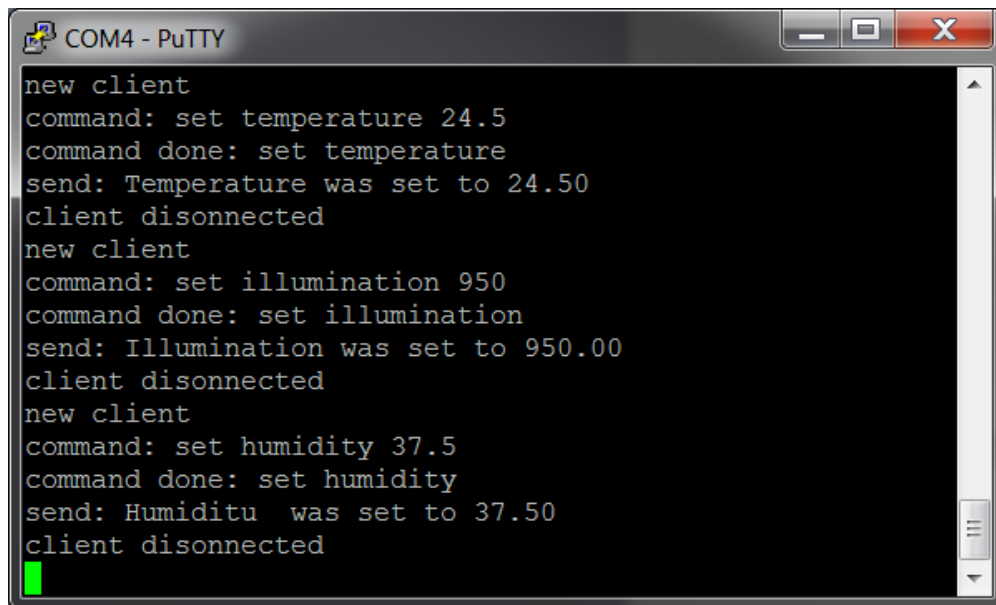
Дані параметри відповідають за температура, вологість та освітленість. Наступним після них має буде число, яке необхідно встановити в якості еталонного параметру (рис 3.6 та 3.7).

```

Problems @ Javadoc Declaration Console
<terminated> test2 [Java Application] C:\Program Files\Java\jre1.8.0_31\bin\javaw.exe (11 черв. 2015 14:29:50)
Send: set temperature 24.5
Recive: Temperature was set to 24.50
Send: set illumination 950
Recive: Illumination was set to 950.00
Send: set humidity 37.5
Recive: Humiditu was set to 37.50

```

Рисунок 3.6 – Робота команди set



```

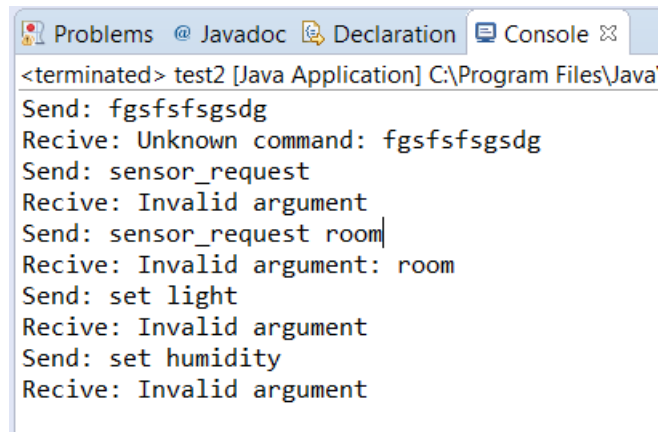
COM4 - PuTTY
new client
command: set temperature 24.5
command done: set temperature
send: Temperature was set to 24.50
client disonnected
new client
command: set illumination 950
command done: set illumination
send: Illumination was set to 950.00
client disonnected
new client
command: set humidity 37.5
command done: set humidity
send: Humiditu was set to 37.50
client disonnected

```

Рисунок 3.7 – Робота команди set PUTTY

Лексичний аналіз усіх команд та файлу конфігурацій відбувається за допомогою використання команд розбиття строки на лексеми `strtok()`. Далі за допомогою порівнянь визначаються основні команди або ключові слова та параметри.

Усі команди, які отримує мікроконтролер, перевіряються на коректність, якщо були знайдені помилки, то мікроконтролер надішле відповідне повідомлення (рис. 3.8 та 3.9).

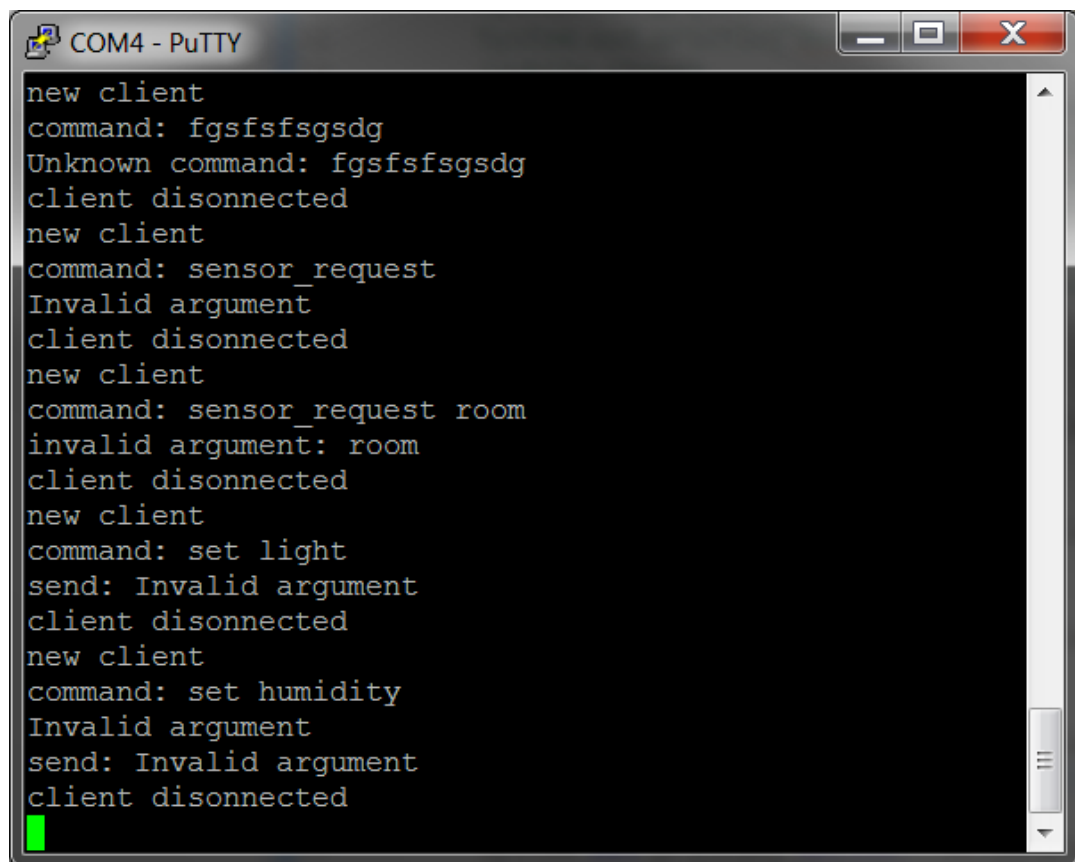


```

Problems @ Javadoc Declaration Console
<terminated> test2 [Java Application] C:\Program Files\Java
Send: fgsfsfsgsdg
Recive: Unknown command: fgsfsfsgsdg
Send: sensor_request
Recive: Invalid argument
Send: sensor_request room
Recive: Invalid argument: room
Send: set light
Recive: Invalid argument
Send: set humidity
Recive: Invalid argument

```

Рисунок 3.8 – Обробка помилкових команд



```

COM4 - PuTTY
new client
command: fgsfsfsgsdg
Unknown command: fgsfsfsgsdg
client disconnected
new client
command: sensor_request
Invalid argument
client disconnected
new client
command: sensor_request room
invalid argument: room
client disconnected
new client
command: set light
send: Invalid argument
client disconnected
new client
command: set humidity
Invalid argument
send: Invalid argument
client disconnected

```

Рисунок 3.9 – Обробка помилкових команд PUTTY

## 3.2 Програмне забезпечення Android

Оскільки в якості сервера буде використано пристрій з ОС Android, то необхідно розробити класи для його взаємодій з Arduino через протокол TELNET. Для розробки програмного коду скористаємось офіційною середою

розробки від Android - Android Studio [23]. Вона має зручний інтерфейс, можливість завантаження програми на пристрій, а також ефективні засоби тестування роботи програм.

Щоб виконати поставлену задачу, нам необхідно розробити пакет, який буде виконувати наступні функції:

- Відправляти дані на мікроконтролер
- Отримувати дані від мікроконтролера
- Обробляти результати запитів
- Зберігати результати вимірів
- Розраховувати середнє значення вимірів
- Зберігати параметри з'єднання
- Автоматично опитувати датчики на вулиці та в кімнатах у фоновому потоці

Для реалізації поставленої задачі було розроблено пакет ArduinoTelnet. Він включає в себе наступні класи:

- Arduino
- Connector
- Room
- Sensors
- Chercker

Діаграма класів наведена у додатку А.

### 3.2.1 Клас Arduino

Клас Arduino є базовим, користувач з ним взаємодіє, використовуючи відповідні методи. Цей клас відповідає за ініціалізацію всіх компонентів, запуск та зупинку потоку перевірки вимірювань. Клас має 2 конструктори:

- `public Arduino (String hostname, int hostport)`
- `public Arduino (String hostname, int hostport, SensorDataBase dataBase)`

Різниця між конструкторами полягає в тому, що другий розрахований на



роботу з базою даних. Перші 2 параметри відповідають за адресу підключення та порту на який потрібно надсилати запити.

Функціонал класу складається з таких методів:

- void start(int delay)
- void stop()
- Sensors getLast()
- void setTemperature(float value)
- void setHumidity(float value)
- void setIllumination(float value)

Функція start() використовується для запуску потоку у фоні, який буде опитувати мікроконтролер та конвертувати отриману інформацію у певний формат, а також додавати середні значення до бази даних (якщо вона була ініціалізована). В якості параметра delay виступає час у мс, який потік має чекати перед тим як надіслати новий запит.

Функція stop() зупиняє фоновий потік, використовуючи метод interrupt(). Це дозволяє коректно завершити потік у відповідності з документацією java [24].

Функція getLast() повертає об'єкт типу Sensors з останніми показниками вимірювань.

Функція setTemperature() виконує встановлення температури на мікроконтролері у відповідності з параметром value. Для передачі повідомлення функція запускає окремий потік, який і надсилає команду.

Функція setHumidity() аналогічно до функції setTemperature() встановлює вологість.

Функція setIllumination() аналогічно до функції setTemperature() встановлює значення освітленості.

### **3.2.2 Клас Connector**

Клас Connector використовується для встановлення з'єднання з мікроконтролером та обміну повідомленнями. Клас має 2 конструктори:

- Connector(String hostname)
- Connector(String hostname, int hostport)

Різниця між конструкторами полягає в тому, що в першому випадку в якості параметра виступає лише адрес підключення, а в другому випадку користувач може вибрати ще й порт. В першому конструкторі за замовчуванням використовується порт 23, стандартний порт протоколу TELNET.

Функцію клас має одну String sendCommand(String command). Її код наведено нижче:

```
public synchronized String sendCommand(String command) {
    String result = "";
    String com = command + "\n";
    boolean success = false;
    while (!success) {
        try {
            Socket s = new Socket(host, port);
            s.getOutputStream().write(com.getBytes());
            byte buf[] = new byte[64 * 1024];
            int r = s.getInputStream().read(buf);
            String data = new String(buf, 0, r);
            result = result + data;
            s.close();
            success = true;
        } catch (Exception ex) {}
    }
    return result;
}
```

Функція намагається у циклі while() відправити повідомлення серверу, якщо їй це вдається, то отримані дані конвертуються у строку формату String та повертаються назад, якщо ж виконати підключення не вийшло, то функція спробує зробити це знов.

### 3.2.3 Клас Room

Клас Room використовується для збереження показників вимірів з кімнат, для цього в нього є відповідні поля:

- humidity
- temperature
- illumination

Вони відповідають за зберігання даних про вологість, температуру та освітленість у приміщенні.

З методів у класі реалізовані такі функції:

- synchronized void setHumidity(Float humidity)
- synchronized void setTemperature(Float temperature)
- synchronized void setIllumination(Float illumination)
- synchronized Float getHumidity ()
- synchronized Float getTemperature ()
- synchronized Float getIllumination ()

Функції setHumidity(Float humidity), setTemperature(Float temperature) та setIllumination(Float illumination) встановлюють значення вологості, температури та освітлення.

Функції getHumidity (), getTemperature () та getIllumination () повертають поточне значення вологості, температури та освітленості.

### 3.2.4 Клас Sensors

Клас Sensors зберігає показники значень з усіх кімнат, а також датчиків, розташованих на вулиці. Для цього в ньому є відповідні поля та масив:

- roomsNumber
- humidity
- temperature
- illumination
- pressure

- rooms[]

Для доступу та роботи з класом у ньому реалізовані наступні методи:

- synchronized void addMeasurements (Float temperature, Float humidity, Float illumination, Float pressure)
- synchronized String toString()
- synchronized void addRoomByIndex (int index, Room room)
- synchronized void setHumidity(Float humidity)
- synchronized void setTemperature(Float temperature)
- synchronized void setIllumination(Float illumination)
- synchronized void setPressure(Float pressure)
- synchronized Float getHumidity ()
- synchronized Float getTemperature ()
- synchronized Float getIllumination ()
- synchronized Float getPressure ()
- synchronized int getRoomNumber ()
- synchronized Room getRoomByIndex (int index)
- synchronized Sensors getAverage (List<Sensors> list)

Функція addMeasurements додає нові дані до поточного об'єкта. Параметрами виступають, температура, вологість, освітленість та барометричний тиск.

Функція toString використовується для конвертації даних, які зберігає клас, у зміну формату String.

Функція addRoomByIndex додає показники вимірювань для кімнати. В якості параметрів необхідно вказати порядковий номер кімнати та об'єкт типу Room.

Функції setHumidity, setTemperature, setIllumination та setPressure використовуються для встановлення нових значень вологості, температури, освітленості та барометричного тиску.

Функції getHumidity, getTemperature, getIllumination, getPressure

повертають значення вимірів вологості, температури, освітлення та барометричного тиску.

Функція `getRoomNumber` повертає кількість кімнат.

Функція `getRoomByIndex` повертає кімнату у відповідності з вказаним її порядковим номером.

Функція `getAverage` знаходить середнє значення всіх параметрів з переданого їй списку та повертає новий об'єкт з цими значеннями.

### **3.2.4 Клас Checker**

Клас `Checker` є спадкоємцем класу `Thread` та виконується у фоновому потоці. Він надсилає запити мікроконтролеру, виконує лексичний розбір, отриманої інформації та додає нові вимірювання до списку. При умові наявності бази даних з кожних 30 вимірювань розраховує середнє і додає до бази даних. Клас складається лише з перезавантаженого метода `run`, який виконується при запуску об'єкта цього класу.

Приклад перевірки роботи пакету `ArduioTelnet` на `ASUS Nexus 7 2012 16GB` наведено на рис 3.10.

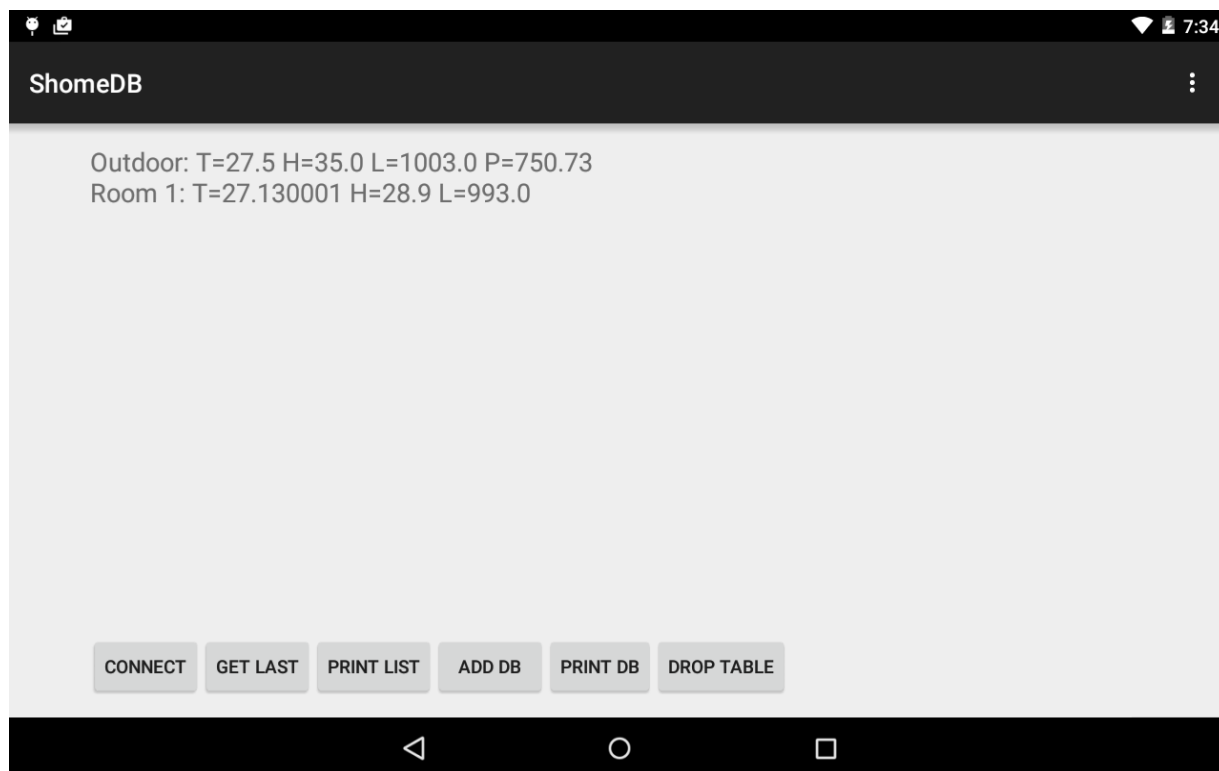


Рисунок 3.10 - Дані, отримані з Arduino

### 3.3 Програмне забезпечення бази даних

Для роботи систем прогнозування, а також для оптимізації збереження інформації, спрощення доступу до неї необхідно реалізувати базу даних. Це дозволить легко виконувати збереження даних у хмарі, імпорт та експорт даних. Для реалізації цієї задачі доцільним буде скористатися можливостями програмування для Android, а саме базою даних на мові SQLite.

Використання SQLite дозволяє зберігати базу даних локально у директорії програми. Для експорту та імпорту достатньо буде просто отримати доступ до файлу и виконати переписування або зчитування файлу.

Для реалізації роботи з базою даних на Android було створено пакет SensorDB. Він складається з наступних класів:

- SensorDataBaseHelper
- SensorDataBase

UML діаграма класів наведена на рис. 3.11.



Рисунок 3.11 – UML діаграма класів для роботи з БД

### 3.3.1 База даних

Для зручності роботи з даними необхідно розробити власне саму базу даних. Створимо базу даних 3 нормальної форми [25]. У відповідності до 3NF сформуємо по таблиці на кожную кімнату, таблицю для вуличних вимірювань та

базову таблицю. При цьому у базовій таблиці будемо мати посилання на таблиці кімнат та зовнішніх вимірів. Діаграма бази даних для випадку 1 кімнати наведена на рис 3.12

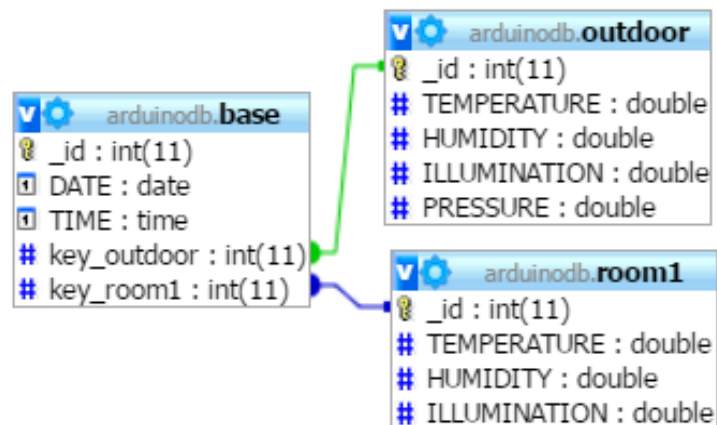


Рисунок 3.12 – Схема БД

Базова таблиця outdoor складається з таких стовбців:

- `_id`
- `DATE`
- `TIME`
- `key_outdoor`
- `key_room1`

Стовпець `_id` є унікальним ідентифікатором запису у таблиці.

У полі `DATE` будемо зберігати дату, коли було зроблено запис.

У полі `TIME` зберігаємо час запису.

Поле `key_outdoor` є зовнішнім посиланням на запис у таблиці `outdoor`.

Поле `key_room` є зовнішнім посиланням на запис у таблиці `room1`.

Таблиця `outdoor` складається з наступних стовбців:

- `_id`



- TEMPERATURE
- HUMIDITY
- ILLUMINATION
- PRESSURE

Стовпець `_id` є унікальним ідентифікатором запису у таблиці.

У полі `TEMPERATURE` зберігається значення температури на вулиці.

У полі `HUMIDITY` зберігається значення вологості на вулиці.

У полі `ILLUMINATION` зберігається значення освітленості на вулиці.

У полі `PRESSURE` зберігається значення атмосферного тиску на вулиці.

Таблиця `room1` має такі стовпці:

- `_id`
- `TEMPERATURE`
- `HUMIDITY`
- `ILLUMINATION`

Стовпець `_id` є унікальним ідентифікатором запису у таблиці.

У полі `TEMPERATURE` зберігається значення температури.

У полі `HUMIDITY` зберігається значення вологості.

У полі `ILLUMINATION` зберігається значення освітленості.

### 3.3.2 Клас `SensorDataBaseHelper`

У відповідності з правилами роботи з SQLite на Android ми маємо створити клас успадкований від класу `SQLiteOpenHelper` [26]. Далі необхідно перевизначити методи `onCreate` та `onUpgrade`. Вони відповідають за створення бази або її модифікацію, якщо версії бази даних відрізняються.

Базову таблицю створимо командою:

```
create table base (_id integer primary key autoincrement, DATE date NOT NULL, TIME time NOT NULL, key_outdoor integer, key_room1 integer);
```

Таблицю для зовнішніх вимірів створимо командою:

```
create table outdoor (_id integer primary key autoincrement, TEMPERATURE real default null, HUMIDITY real default null, ILLUMINATION real default null,
```

PRESSURE real default null);

Таблицю для вимірів у кімнаті створимо командою:

```
create table room1 (_id integer primary key autoincrement, TEMPERATURE
real default null, HUMIDITY real default null, ILLUMINATION real default null);
```

### 3.3.3 Клас SensorDataBase

Для роботи користувача з базою даних було створено клас SensorDataBase. Він реалізує функціонал, щоб забезпечити безпечну роботу з базою даних, та позбавити користувача писати SQL запити. У класі присутні наступні методи:

- List<Sensors> getData(String filter)
- void addData(Sensors sensor)
- void setRoomNumber(int number)
- void create ()
- List<Sensors> getData ()
- Sensors getLast()
- List<Sensors> getDataByDate (String date)
- List<Sensors> getDataByDateInRange (String dateBegin, String dateEnd)
- List<Sensors> getDataByTimeInRange (String timeBegin, String timeEnd)
- List<Sensors> getDataByDateAndTimeInRange (String date, String timeBegin, String timeEnd)
- List<Sensors> getDataByDateInRangeAndTimeInRange (String dateBegin, String dateEnd, String timeBegin, String timeEnd)
- File getBaseFile ()
- void drop()

Функція getData (з фільтром " String filter ") виконує вибір даних з бази даних відповідно до фільтру. Далі данні конвертуються у формат Sensors та додаються до списку, який і повертається.

Функція addData додає дані типу Sensors до бази.

Функція setRoomNumber встановлює кількість кімнат та перебудовує

таблиці за необхідності.

Функція `create` створює таблиці.

Функція `getData` повертає усі значення з бази даних у форматі списку.

Функція `getDataByDate` виконує вибірку інформації з бази певну дату та повертає список.

Функція `getDataByDateInRange` виконує вибірку з бази даних за дату у вказаному проміжку.

Функція `getDataByTimeInRange` проводить вибірку інформації у вказаний період часу та повертає список

Функція `getDataByDateAndTimeInRange` виконує вибірку даних з бази за певну дату у певний проміжок часу.

Функція `getDataByDateInRangeAndTimeInRange` фільтрує дані з бази за певний період дати та часу.

Функція `getBaseFile` повертає посилання на файл, в якому зберігається БД.

Функція `drop` видаляє всі таблиці разом з даними.

Результати роботи бази даних наведені на рис. 3.13. В цьому випадку на екран виведені усі дані з таблиць та конвертовані у зручний формат.



Рисунок 3.13 – Робота з БД на Android

### 3.4 Генератор файлів конфігурацій

Для полегшення налаштування мікроконтролера під конкретний випадок для користувача була створена програма SHome Configuration Creator. У програмі є графічний інтерфейс, що дозволяє користувачу без проблем редагувати чи створювати файли з налаштуваннями для Arduino.

Для розробки програми скористаємось Microsoft Visual Studio, ліцензію на яку студентам КПІ надає Microsoft [27]. Розробляти проект будемо на мові C#.

На головній вкладці, яка називається General, користувач може обрати кількість кімнат, адрес підключення та наявність датчика барометричного тиску. Зовнішній вигляд вкладки наведено на рис. 3.14.

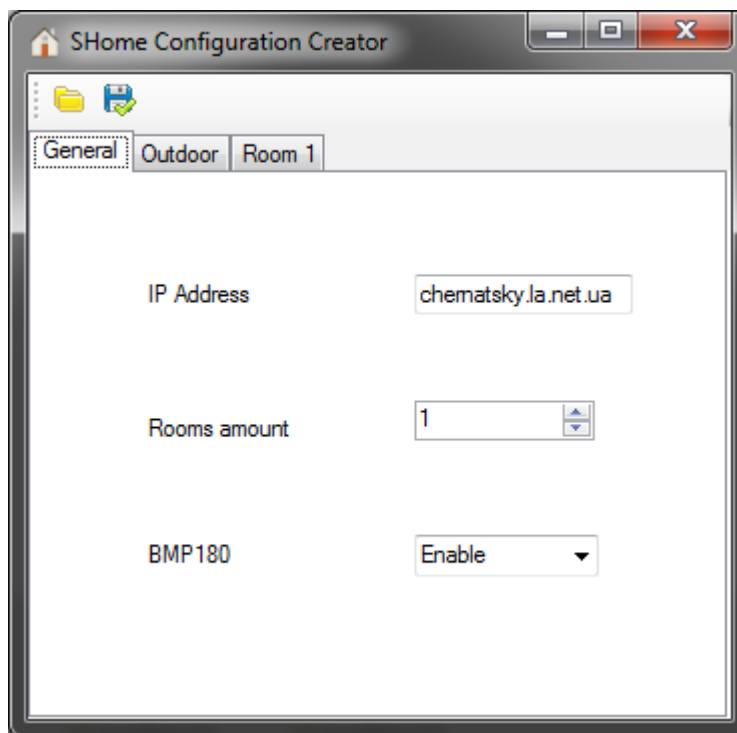


Рисунок 3.14 – Вкладка налаштувань General

На вкладці Outdoor користувач може обрати датчики, які розташовані на вулиці, для цього слід з першого випадаючого списку обрати тип датчика, а з другого - порт, до якого він підключений. Для роботи доступні такі датчики:

- DHT11
- DHT21
- DHT22
- DS18B20
- ILLUMINATION

Зовнішній вигляд вкладки наведено на рис. 3.15.

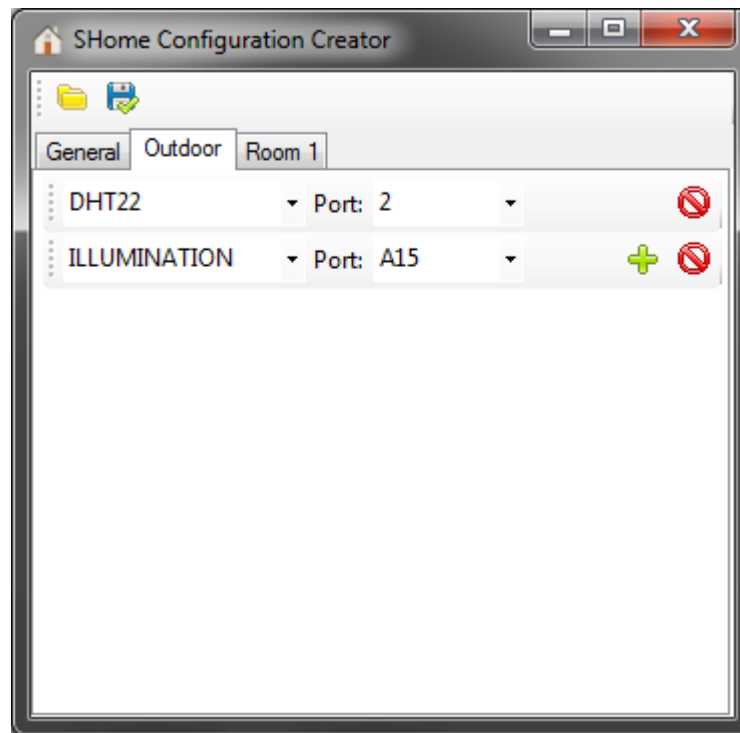


Рисунок 3.15 – Вкладка налаштування Outdoor

Для кожної з кімнат у відповідних вкладках можна додавати датчики та обладнання. Перечень датчиків такий самий, як для вкладки Outdoor. З обладнання користувачам є доступ до такого обладнання:

- HEAT
- COOL
- FAN
- WETTING
- LIGHT

Зовнішній вигляд вкладки наведено на рис. 3.16.

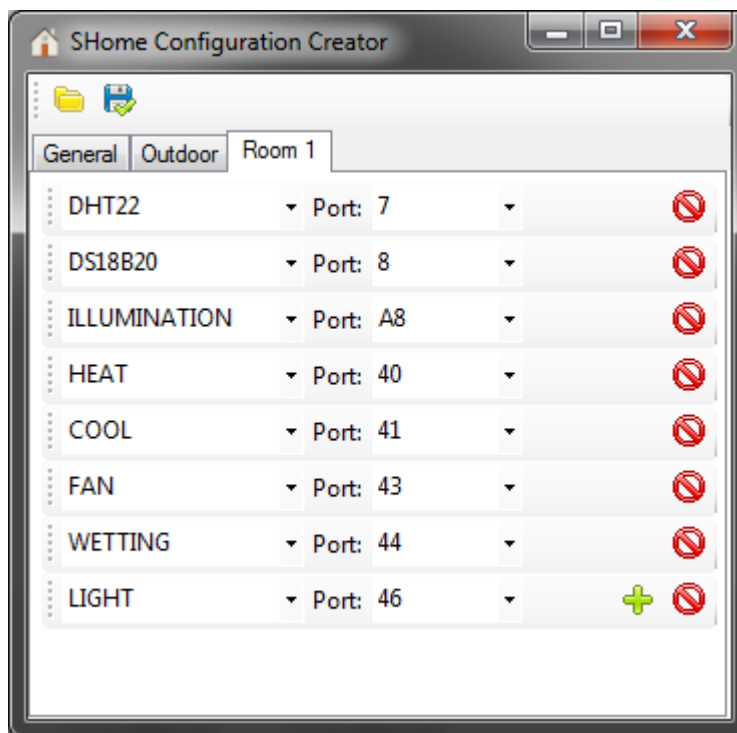


Рисунок 3.16 – Вкладка налаштування Room 1

Для редагування файлу можна скористатися функцією відкриття. Далі користувач за допомогою діалогового вікна проведе вибір необхідного йому файлу рис. 3.17.

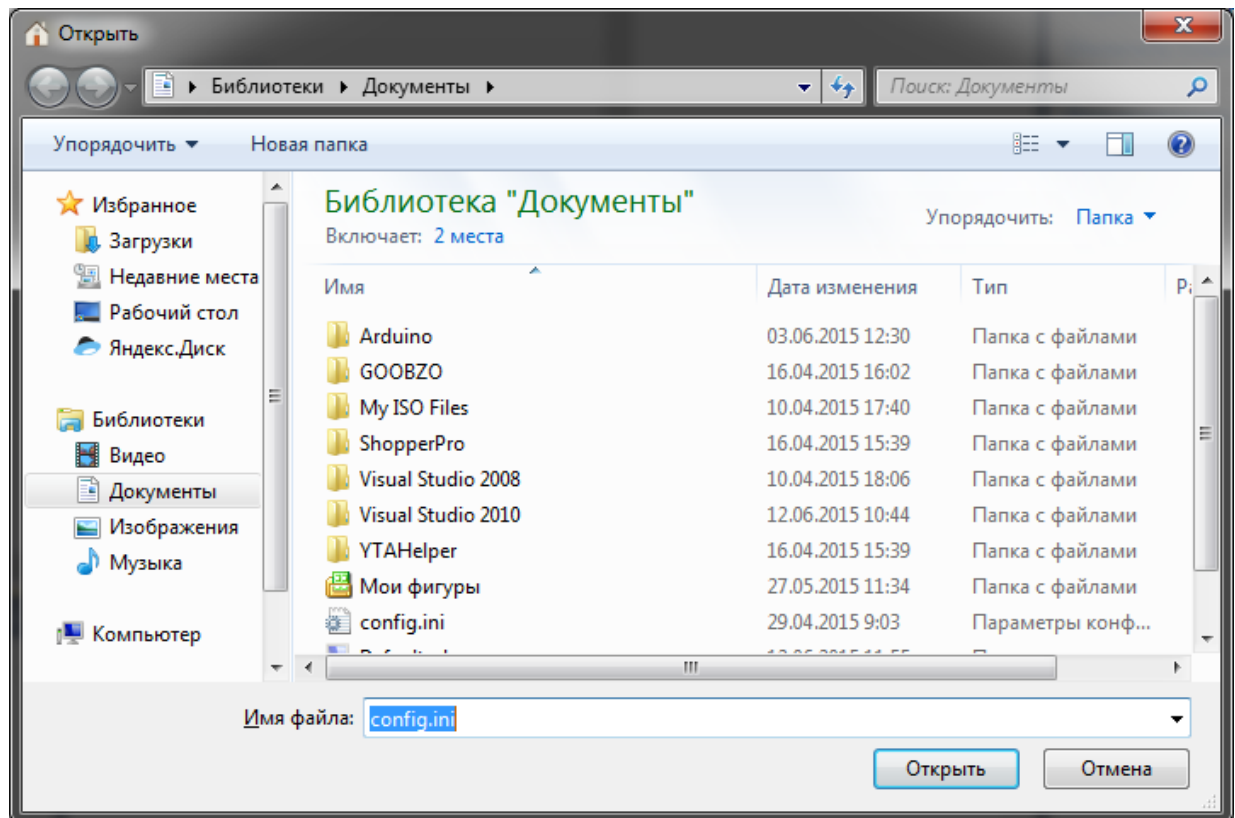


Рисунок 3.17 – Вікно відкриття файлу

Відредагований файл можна зберегти, для цього необхідно натиснути на кнопку у вигляді дискети. Далі відкриється діалогове вікно для збереження файлу. Його зовнішній вигляд наведено на рис. 3.18.



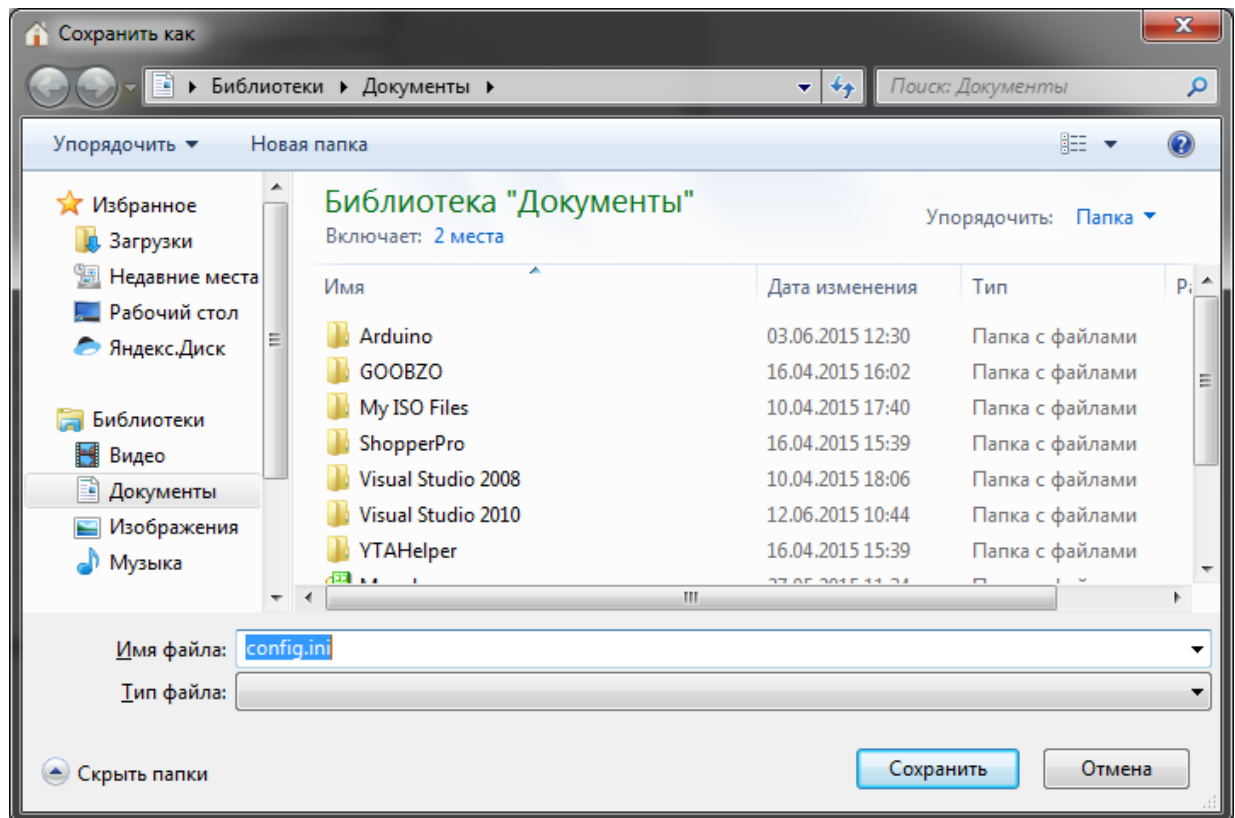


Рисунок 3.18 – Вікно збереження файлу

### 3.5 Висновки

В ході розробки програмного забезпечення для взаємодії користувача з Arduino було реалізовано систему керування мікроконтролером через протокол TELNET. Для цього був створений пакет ArduionoTelnet для Android та написаний скетч для мікроконтролера. У скетчі для мікроконтролера був впроваджений метод зчитування налаштувань з карти пам'яті.

Для полегшення налаштування було створено програму SHome Configuration Creator, яка дозволяє у зручному форматі створювати, редагувати та зберігати файли конфігурацій.

Також для збереження інформації було розроблено та реалізовано базу даних у пакеті SensorDB. Це дозволяє зберігати дані локально на пристрої у баз SQLite та виконувати імпорт та експорт бази даних у хмару.

## **4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ**

### **4.1 Вступ**

У даному розділі проводиться аналіз середовища, в якому проводилося розробка програмного продукту, на основі санітарних норм України.

При роботі з персональною комп'ютерною технікою змінюються фізичні і хімічні чинники середовища: електромагнітні випромінювання, статична електрика, температура і вологість повітря, вміст кисню і озону. Повітря забруднюється шкідливими хімічними речовинами антропогенного походження за рахунок деструкції полімерних матеріалів, що використовуються для обробки приміщень та обладнання.

Неправильна організація робочого місця сприяє загальній та локальній напрузі м'язів шії, тулуба, верхніх кінцівок, викривлення хребта і розвитку остеохондрозу.

### **4.2 Аналіз умов праці в приміщенні**

#### **4.2.1 Оцінка санітарно-гігієнічних умов праці**

План приміщення наведений на рис. 4.1.

Основні характеристики приміщення, що розглядається, наведені в табл. 4.1.

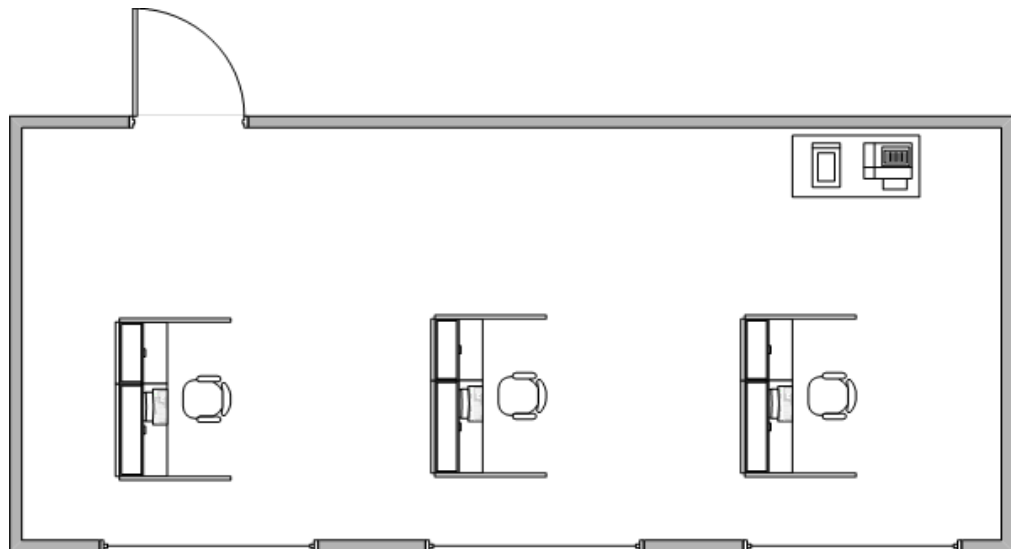


Рисунок 4.1 – План приміщення

Таблиця 4.1 – Характеристики робочого приміщення

Параметр	Позначення	Величина
Довжина, м	A	4
Ширина, м	B	8
Висота, м	H	2,75
Кількість робочих місць	N	3
Площа, м <sup>2</sup>	S	32
Об'єм, м <sup>3</sup>	V	88

Відповідно до [28], площа  $S'$ , виділена для одного робочого місця з персональною ЕОМ, повинна складати не менше 6 кв. м, а об'єм  $V'$  – не менше 20 куб. м. Розрахуємо фактичні значення цих показників, розділивши загальну площу та об'єм приміщення на кількість працюючих:

$$S' = \frac{S}{N} = \frac{4 \cdot 8}{3} = 10,6 \left( \frac{\text{м}^2}{\text{люд.}} \right) \quad (4.1)$$

$$V' = \frac{V}{N} = \frac{4 \cdot 8 \cdot 2,75}{3} = 29,33 \left( \frac{\text{м}^3}{\text{люд.}} \right) \quad (4.2)$$

Отже, за характеристиками площі і об'єму приміщення відповідає нормам згідно [28].

Параметри вікон:

Висота – 1,5 м.

Ширина – 2 м.

Відстань від підлоги – 1 м.

Вікна виходить на північ, можуть відкриватися та мають жалюзі.

Двері відчиняються назовні, ширина коридору 3 м, висота до перекриття 2,75 м. Ширина дверей у приміщенні 0,9 м.

У освітленні приміщення, що розглядається, застосовується бокове природне освітлення (вікна: висота = 1,5 м, ширина = 6 м), штучне, створюване електричними лампами (8 світильників, розташованих на стелі в два ряди).

Розглянемо тепер відповідність характеристик робочого місця нормативним. Для цього зведемо основні вимоги до організації робочого місця з [28] і відповідні фактичні значення для робочого місця, за яким виконується робота, у табл. 4.2.

Крісло є підйомно-поворотним, має підлокітники і можливість регулювання за висотою і кутом нахилу спинки. Екран монітору знаходиться на відстані 0.7 м. Таким чином, за всіма параметрами робоче місце відповідає нормативним вимогам.

Таблиця 4.2 – Характеристики робочого місця

Найменування параметра	Значення	
	фактичне	Нормативне
Висота робочої поверхні, мм	750	680 - 800
Висота простору для ніг, мм	700	не менше 600
Ширина простору для ніг, мм	600	не менше 500
Глибина простору для ніг, мм	700	не менше 650
Висота поверхні сидіння, мм	450	400 - 500
Ширина сидіння, мм	400	не менше 400
Глибина сидіння, мм	400	не менше 400
Висота поверхні спинки, мм	500	не менше 300
Ширина опорної поверхні спинки, мм	400	не менше 380
Радіус кривини спинки в горизонтальній площині, мм	400	400
Відстань від очей до екрану дисплея, мм	750	700 - 800

З обладнання в приміщенні знаходиться 3 комп'ютери та 3 монітори ASUS VN248H 23,8", а також принтер. На все обладнання є паспорт та інструкція з експлуатації українською мовою, згідно з супроводжувальною документацією обладнання відповідає стандартам України і його можна використовувати без загрози здоров'ю та життю працюючого.

### 4.3 Стан повітряного середовища приміщення

На стан організму працівника, його працездатність значний вплив має мікроклімат у виробничих приміщеннях. Під мікрокліматом виробничих приміщень розуміють клімат внутрішнього середовища виробничого приміщення, який визначається поєднаними діями на організм людини, температури, вологості, швидкості руху повітря та теплових випромінювань.

Мікроклімат приміщення, визначається наступними параметрами:

- температура повітря,  $t$  ( $^{\circ}\text{C}$ )
- відносна вологість повітря,  $\varphi$  (%)
- швидкість руху повітря,  $v$  (м/с)
- інтенсивність теплового випромінювання,  $j$  ( $\text{Вт}/\text{м}^2$ )
- температура поверхонь будівельних конструкцій,  $t_{\text{п}}$  ( $^{\circ}\text{C}$ )

Перші три параметри встановлюються відповідно до пори року і категорії роботи за енерговитратами.

Перші три параметри встановлюються відповідно до пори року і категорії роботи за енерговитратами. Робота з ЕОМ, яка розглядається, виконується сидячи і не потребує фізичного напруження; витрати енергії становлять до 120 ккал/год. Відповідно така робота відноситься до категорії Ia, і нормовані параметри мікроклімату визначені у [29] і наведені у табл. 4.3.

Таблиця 4.3 – Оптимальні параметри мікроклімату

Пора року	Категорія робіт	Температура повітря, град. С	Відносна вологість повітря, %	Швидкість руху повітря, м/с
		оптимальна	оптимальна	оптимальна
Холодна	легка-1 а	22 – 24	40 - 60	0,1
Тепла	легка-1 а	23 – 25	40 - 60	0,1

Особлива увагу при роботі з ЕОМ слід звернути на те, щоб параметри мікроклімату були оптимальними, тобто такі, які забезпечують відчуття теплового комфорту та створюють передумови для високого рівня працездатності.

Температура повітря у приміщенні, що розглядається, визначається температурою атмосферного повітря і джерелами виділення тепла. Ними є електрообладнання, сонячна радіація і теплота, яку виділяє організм людини. Сумарна кількість теплоти, що виділяється у приміщенні, не призводить до

виходу температури за встановлені межі. Суттєвого підвищення температури внаслідок дії сонячної радіації вдається уникнути, закривши вікна шторами; проникаюча радіація не спричиняє будь-якого помітного теплового ефекту внаслідок низької теплопровідності будівельних конструкцій. В даному випадку приміщення обладнане системою опалення та кондиціонером LG S12SWT, який розрахований на приміщення площею 36 м<sup>2</sup>. При низьких температурах у холодну пору року стабільність температури повітря підтримує опалювальна система. Температура приміщення становить 22<sup>0</sup> С взимку та 24<sup>0</sup> С влітку, що відповідає нормі. Тобто, протягом року температура повітря у приміщенні не виходить за встановлені межі. Тому по цим параметрам приміщення відповідає нормам викладених у [28].

#### 4.4 Освітлення приміщення

Відносно вікон робоче місце повинно бути розміщено так, щоб природне світло було збоку, переважно з лівого. Робоче місце, обладнане ПК повинно бути розташоване так, щоб уникнути попадання в очі прямого світла. Джерела штучного світла рекомендується розташувати з обох сторін від екрану паралельно напрямку зору. Вікна приміщень повинні мати регулювальні пристрої для відкривання. Як в приміщення джерелом світла є штучне освітлення, то застосовуватися, як правило, люмінесцентні лампи [32].

В приміщенні знаходяться 3 вікна з однієї сторони. Їх характеристики:

Висота:  $L=1.5$  м; ширина:  $W=2$  м, то загальна площа одного вікна:  $S=L*W=3$  м<sup>2</sup>, а загальна  $S=9$  м<sup>2</sup>

Робота за дисплеєм ПЕОМ за розрядом зорових робіт відноситься до III розряду. При загальному висвітленні освітленість робочого місця повинна становити від 200 до 400 лк.

При штучному освітленні нормуються наступні параметри:

E (лк) - найменша припустима освітленість;

M - показник дискомфорту;

$K_p$  (%) - коефіцієнт пульсації освітленості;

Перевіримо, чи відповідають нормам фактичні параметри штучного освітлення в приміщенні. Номінальний світловий потік лампи білого світіння ЛБ-40  $\Phi_{л} = 3120$  лм.

У приміщенні застосовуються світильники, у яких встановлені 2 лампи.

Висоту підвісу світильника визначимо з формули :

$$h = H - h_c - h_p - h_n \quad (4.3)$$

$H$  - висота приміщення, м;  $h_c$  - висота світильника, м;  $h_n$  - відстань від стелі до підвісу, м;  $h_p$  - висота робочої поверхні, м.

Для розглянутого приміщення :

$$H = 2,75 \text{ м}; h_c = 0,2 \text{ м}; h_n = 0,18 \text{ м}; h_p = 0,75 \text{ м}.$$

звідси :

$$h = 2,75 - 0,2 - 0,18 - 0,75 = 1,62 \text{ м}. \quad (4.4)$$

Світильники розташовані в 2 ряди. Відстань між рядами 2 метра, відстань від ряду до стіни 0,5 метра. Приміщення має наступні габарити:

довжина  $A = 4$  метрів,

ширина  $B = 8$  метрів.

Визначимо освітленість у робочій точці. Для розрахунку загальної рівномірної освітленості при горизонтальній робочій поверхні використаємо метод коефіцієнта використання світлового потоку.

Розрахункова формула для світлового потоку світильника має вигляд:

$$\Phi_{л} = \frac{E \cdot K_3 \cdot S \cdot Z}{N \cdot n} \quad (4.5)$$

$N$  - число світильників у приміщенні,  $N = 2 \cdot 4 \cdot 2 = 16$ ;

$n$  - коефіцієнт використання світлового потоку;

$\Phi_{л}$  - світловий потік ламп;

$K_3$  - коефіцієнт запасу,  $K_3 = 1.5$ ;

$Z$  - коефіцієнт нерівномірності;

$S$  - площа приміщення;



$E$  - освітленість, створювана всіма світильниками.

Звідси одержуємо формулу для розрахунку освітленості на робочому місці :

$$E = \frac{\Phi_{\text{л}} \cdot N \cdot n}{K_3 \cdot S \cdot Z} \quad (4.6)$$

Коефіцієнт використання світлового потоку залежить від:

ККД, кривій розподілу сили світла світильника;

Коефіцієнта відбиття стелі  $R_c$  і стін  $R_c$ ;

Висоти підвісу світильників  $h_{\text{п}}$ ;

Показника приміщення і обчислимо за формулою:

$$i = \frac{A \cdot B}{h \cdot (A + B)} \quad (4.7)$$

$$i = (8 \cdot 4) / (1,62 \cdot (8 + 4)) = 1,64 \quad (4.8)$$

Нам відомо, що стеля й стіни пофарбовані в світло-сірий і світло-бежевий кольори. Приймаємо:

$R_{\text{п}} = 50\%$ ,  $R_c = 30\%$ .

Звідси:  $n = 42\%$ .

$$E = \frac{3120 \cdot 16 \cdot 0,42}{1,5 \cdot 32 \cdot 1,1} = 397,09 \text{ лк} \quad (4.9)$$

Виходячи з того, що по розряду зорової роботи робота за дисплеєм ПЕОМ відноситься до III розряду, тому при загальному освітленні освітленість робочого місця повинна становити від 200 до 400 лк. Фактична освітленість на робочому місці становить 397,09 лк. У такий спосіб для роботи з дисплеєм цілком достатньо існуючих джерел світла.

## 4.5 Оцінка умов безпеки праці

### 4.5.1 Вимоги електробезпеки

Технічні рішення із запобігання електротравм від контакту з нормально струмовідними елементами електроустаткування:

- величина напруги мережі 380<sup>x</sup>220В (міжфазна лінійна і фазна)
- всі нормально струмовідні елементів (в першу чергу електричні дроти) вкриті ізоляційними матеріалами
- в джерелі безперебійного живлення персонального комп'ютера використовується механічне захисне блокування, що забезпечує вимикання напруги при його відкриванні
- електромережа в приміщенні розведена в спеціальних каналах стін і підлоги

Дане приміщення задовольняє вимоги до електробезпеки у приміщенні, в якому встановлені ЕОМ.

#### **4.5.2 Оцінка пожежної безпеки**

З огляду на можливість виникнення пожежі слід з'ясувати, які речовини і матеріали можуть горіти. У приміщенні, що розглядається, можуть горіти вироби з дерева, пластмас, тканини і паперу. Горючі рідини, пил та волокна у приміщенні не використовуються і не виділяються. Тому приміщення, що аналізується, відноситься, відповідно до нормативної документації, до зони П-Іа [30] і до категорії пожежної небезпеки В.

Ймовірними причинами виникнення пожежу можуть бути несправність електрообладнання (кабелів, розеток), короткі замикання внаслідок виходу з ладу чи експлуатації несправного електроустаткування (ПЕОМ, периферійних пристроїв), порушення правил протипожежної безпеки тощо.

Експлуатація ліній електромережі практично повністю унеможлиблює виникнення електричного джерела загоряння в наслідок короткого замикання та перевантаження проводів. Застосовуються дроти з важкогорючою і негорючою ізоляцією.

Для своєчасного попередження пожеж та підвищення оперативності реагування при їх виникненні у приміщенні використовується такий комплекс заходів:

- обов'язковий інструктаж персоналу з питань охорони праці

- зокрема, правила пожежної безпеки у приміщеннях з ЕОМ
- заборона використання відкритого вогню у приміщенні
- наявність системи автоматичної пожежної сигналізації з димовими пожежними оповіщувачами
- ступінь вогнестійкості будівлі, у якій розташовано приміщення – II
- наявність шляхів евакуації при виникненні пожежі
- розміщення схеми евакуації людей при пожежі і ознайомлення з нею персоналу

Приміщення має один вихід, оскільки в ньому працює менше 25 чоловік. Ширина проходу між робочими місцями у приміщенні перевищує 1 м. Будинок має два виходи – головний і запасних. Коридор між приміщеннями має два виходи на різні сходи, одні з яких ведуть до головного виходу, а другі - до спеціального евакуаційного виходу.

Для гасіння пожежі кожна кімната обладнана ручними вуглекислотними вогнегасниками ВВК-1,4 [31]. У загальному коридорі встановлені пінні вогнегасники ВВП. Призначена відповідальна особа, що відповідає за дотримання персоналом вимог пожежної безпеки. Розроблено план евакуації персоналу і найбільш коштовного устаткування (майна). Співробітники ознайомлені з порядком і планом евакуації.

Отже, шляхи евакуації з приміщення повністю відповідають нормам.

## **4.6 Висновки**

У даному розділі було проведено аналіз умов праці у приміщенні, де розробляється дипломний проект. Були наведені характеристики робочого приміщення: довжина – 4 м., ширина – 8 м., висота – 2,75 м., кількість робочих місць – 3, площа – 32 м<sup>2</sup>, об'єм - 88 м<sup>3</sup>. Площа та об'єм виділені для кожного робочого місця відповідають вимогам та становлять 10,6 м<sup>2</sup> та 29,33 м<sup>3</sup> відповідно. У приміщенні застосовується бокове природне освітлення та штучне (два ряди світильників, у кожному з яких знаходиться по чотири лампи

типу ЛБ-40). Фактична освітленість на робочому місці становить 397,09 лк, що також задовольняє вимогам. Визначено, що характер робіт складності є допустимим рівнем напруженості і рекомендовано робити перерви по 10 хвилин після кожної години роботи. Встановлено, що температура повітря у приміщенні становить 22-24<sup>0</sup> С, в залежності від пори року. Зазначено, що приміщення за групою електробезпечності відноситься до приміщень без підвищеної небезпеки ураження струмом. Також визначені вогнегасники – ВВК-1,4 та ВВП.

Таким чином робоче приміщення, в якому проводиться розробка дипломного проекту повністю відповідає усім необхідним вимогам.

## ВИСНОВКИ

На сьогоднішній день концепція "розумного" будинку є дуже актуальна. Вона дозволяє користувачам економити енергоносії та створювати для себе комфортний мікроклімат у приміщенні, тому важливість цього питання є дуже високою, особливо для нашої країни в її складному економічному та енергетично залежному становищі.

В даній дипломній роботі було проведено аналіз мікроконтролерів та мікрокомп'ютерів від різних виробників. Для реалізації концепції був обраний мікроконтролер Arduino Mega 2560, його характеристики дозволяють забезпечити користувачів можливістю горизонтального та вертикального розширення системи, а вартість одна з найнижчих серед розглянутих аналогів, що робить цей вибір економічно вигідним.

Для контролю параметрів навколишнього середовища, а також мікроклімату приміщення був визначений перелік розповсюджених сенсорів. Також було наведені характеристики цих сенсорів. Такий підхід дозволяє користувачеві самому визначити параметри для контролю, та побудувати систему відповідно до його вподобань.

Для реалізації системи управління виконавчими пристроями був розроблений ПЗ для мікроконтролера. Головною його особливістю є гнучка можливість керування мікроконтролером, а також можливість зміни конфігурації системи без переписування коду. Для цього достатньо лише завантажити новий файл конфігурації на карту пам'яті та перезавантажити мікроконтролер.

Щоб полегшити налаштування мікроконтролера, була розроблена програма SHome Configuration Creator. Вона дозволяє людині, яка не вміє програмувати, самостійно вносити зміни до своєї системи.

В якості зв'язку з мікроконтролером була використана звичайна локальна мережа. У наші дні, майже кожен має вдома власну локальну мережу. Таким

чином, інтеграція мікроконтролера та зв'язок з ним відбуваються легко на базі існуючих рішень. Передача повідомлень між мікроконтролером та сервером відбувається через протокол TELNET. Це дозволяє використовувати для контролю навіть звичайний термінал. Для роботи на ОС Android був розроблений пакет ArduinoTelnet. Він розроблений на мові JAVA, що дозволяє при необхідності перенести його у будь-яку інші програму на цій мові.

Для збереження даних температури, вологості, атмосферного тиску та освітленості була створена база даних на мові SQLite. Це дозволило інтегрувати її в пакет додатку на Android та зберігати локально. Для роботи з базою був розроблений спеціальний пакет SHomeDB.

Таким чином, при виконанні даної дипломної роботи була розроблена функціональна схема системи управління, обрано протоколи зв'язку елементів системи та створено програми управління виконавчими приладами. В результаті ми отримали діючу модель системи з датчиками температури, вологості, тиску та здатністю керування деякими пристроями.

Результати даної дипломної роботи можуть бути використані для роботи комплексної системи SHome з елементами штучного інтелекту, або як окремий модуль керування пристроями для розробки власної системи.

## ПЕРЕЛІК ПОСИЛАНЬ

1. Lutron Electronics, Inc. - Dimmers And Lighting Controls. – Режим доступу : <http://www.lutron.com/en-US/Company-Info/Pages/AboutUS/OurStory.aspx> . – Дата доступу : 05.06.2015.
2. Pico Electronics. – Режим доступу : <http://www.picodenshi.com/about-pico-30238.html>. – Дата доступу : 05.06.2015.
3. HomeKit - Apple Developer. – Режим доступу : <https://developer.apple.com/homekit/> . – Дата доступу : 05.06.2015.
4. Домашняя автоматизация с Z-Wave — это доступно. – Режим доступу : <http://rus.z-wave.me>. – Дата доступу : 05.06.2015.
5. First Look at Microsoft's HoloLens. – Режим доступу : <http://www.pcmag.com/article2/0,2817,2475581,00.asp>. – Дата доступу : 05.06.2015.
6. Smart Control Allone. – Режим доступу : [http://www.orvibo.com/en\\_products\\_view.asp?mid=15&pid=57&id=232](http://www.orvibo.com/en_products_view.asp?mid=15&pid=57&id=232) . – Дата доступу : 05.06.2015.
7. Products - Clipsal by Schneider Electric. – Режим доступу : <https://www.clipsal.com/Home-Owner> . – Дата доступу : 05.06.2015.
8. Arduino Uno. – Режим доступу : <http://arduino.ua/ru/hardware/Uno> . – Дата доступу : 05.06.2015.
9. GREAT WALL Electronics Co., Ltd. – Режим доступу : <http://ru.aliexpress.com/store/731260> . – Дата доступу : 05.06.2015.
10. Arduino Mega 2560. – Режим доступу : <http://arduino.ua/ru/hardware/Mega2560> . – Дата доступу : 05.06.2015.
11. chipKIT Uno32™ Prototyping Platform. – Режим доступу : <http://www.digilentinc.com/Products/Detail.cfm?Prod=CHIPKIT-UNO32> . – Дата доступу : 05.06.2015.
12. chipKIT Max32™ Prototyping Platform. – Режим доступу :

- <http://www.digilentinc.com/Products/Detail.cfm?Prod=CHIPKIT-MAX32>. –  
Дата доступа : 05.06.2015.
13. RASPBERRY PI 1 MODEL A+. – Режим доступа :  
<https://www.raspberrypi.org/products/model-a-plus/> . – Дата доступа :  
30.04.2015.
14. RASPBERRY PI 1 MODEL B+. – Режим доступа :  
<https://www.raspberrypi.org/products/model-b-plus/> . – Дата доступа :  
30.04.2015.
15. RASPBERRY PI 2 MODEL B. – Режим доступа :  
<https://www.raspberrypi.org/products/raspberry-pi-2-model-b/> . – Дата  
доступа : 30.04.2015.
16. Плата расширения Arduino Ethernet. – Режим доступа :  
<http://arduino.ua/ru/hardware/EthernetShield> . – Дата доступа : 05.06.2015.
17. DS18B20 - Датчик температуры цифровой. – Режим доступа :  
<http://www.sinava.ru/DS18B20.php> . – Дата доступа : 05.06.2015.
18. Датчики влажности DHT11 и DHT22. – Режим доступа : <http://homesmart.ru/index.php/oborudovanie/datchiki/datchiki-vlazhnosti-dht11-i-dht22> .  
– Дата доступа : 05.06.2015.
19. BMP180 цифровой модуль атмосферного давления Arduino. – Режим  
доступа : <http://greenchip.com.ua/23-0-100-0.html> . – Дата доступа :  
05.06.2015.
20. СФ2-5а. – Режим доступа : [http:// www.giricond.ru/files/sf2a.pdf](http://www.giricond.ru/files/sf2a.pdf) . – Дата  
доступа : 05.06.2015.
21. SD Library. – Режим доступа : <http://www.arduino.cc/en/Reference/SD> . –  
Дата доступа : 05.06.2015.
22. Ethernet library. – Режим доступа : [http](http://www.arduino.cc/en/Reference/Ethernet)  
<http://www.arduino.cc/en/Reference/Ethernet> . – Дата доступа : 05.06.2015.
23. Android Studio Overview. – Режим доступа :  
<http://developer.android.com/tools/studio/index.html> . – Дата доступа :  
05.06.2015.



24. Java - Thread - Oracle Documentation. – Режим доступу : <https://docs.oracle.com/javase/1.5.0/docs/api/java/lang/Thread.html> . – Дата доступу : 05.06.2015.
25. Третья нормальная форма (3NF). – Режим доступу : [http://www.e-reading.club/chapter.php/99163/67/Bazy\\_dannyh\\_\\_konspekt\\_lekciii.html](http://www.e-reading.club/chapter.php/99163/67/Bazy_dannyh__konspekt_lekciii.html). – Дата доступу : 05.06.2015.
26. Android: SQLite. – Режим доступу : <http://developer.android.com/reference/android/database/sqlite/SQLiteDatabase.html> . – Дата доступу : 05.06.2015.
27. National Technical University of Ukraine Kiev Polytechnic Institute (KPI) - Information Technology - DreamSpark Premium. – Режим доступу : <https://e5.onthehub.com/WebStore/ProductsByMajorVersionList.aspx?ws=0cf64030-16db-e011-b09a-f04da23e67f6&vsro=8>. – Дата доступу : 05.06.2015.
28. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСанПіН 3.3.2.007-98 (затверджено Постановою Головного державного санітарного лікаря України від 10.12.1998 р. № 7).
29. ДСН 3.3.6.042-99. Санітарні норми мікроклімату виробничих приміщень [Текст] / К., 2000.- 16 с.
30. Норми визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою. НАПБ Б.03.002-2007. (затверджено наказом МНС України від 03.12.2007 № 833)
31. Типові норми належності вогнегасників (затверджено наказом Міністерства України з питань надзвичайних ситуацій та у справах захисту населення від наслідків Чорнобильської катастрофи від 2 квітня 2004 р. N 151)
32. Природне і штучне освітлення : ДБН В.2.5-28:2006. – К. : Міністерство будівництва, архітектури та житлово-комунального господарства України, 2006. – 68 с. – (Національні стандарти України).

# UML діаграма класу ArduinoTelnet

