Міністерство освіти і науки України

Державний університет «Житомирська політехніка»

Факультет інформаційно-комп'ютерних технологій

Кафедра інженерії програмного забезпечення

ПОЯСНЮВАЛЬНА ЗАПИСКА

до випускної кваліфікаційної роботи бакалавра

на тему: «Arduino “Smart House”.Система керування функціями Arduino»

Виконала студентка 4-го курсу, групи ПІ-54  
спеціальності 121 «Інженерія програмного забезпечення»

В.О. Раздобудова

Керівник к.т.н., доцент, доцент кафедри ІПЗ

Ю.М.Єфремов

Рецензент к.т.н., доцент, КІ та КБ

О.І. Грабар

Житомир – 2020

ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»

ФАКУЛЬТЕТ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(назва прописними)

КАФЕДРА\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(назва прописними)

СПЕЦІАЛЬНІСТЬ\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(назва прописними)

ЗАТВЕРДЖУЮ

Зав. кафедри \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(назва кафедри)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_\_\_\_ р.

**ЗАВДАННЯ**

на випускну кваліфікаційну роботу

Студентки \_Раздобудової Валерії Олексіївни\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Тема роботи: \_«Arduino “Smart House”. Система керування функціями Arduino»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Затверджена Наказом університету від «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_ 20\_\_\_ р. № \_\_\_\_\_\_\_\_\_\_\_

Термін здачі студентом закінченої роботи \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Вихідні дані роботи (зазначається предмет і об’єкт дослідження) \_дослідження концепції ‘розумного’\_будинку; розробка функціональної системи керування функціями контролера\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Консультанти з випускної кваліфікаційної роботи із зазначенням розділів, що їх стосуються

|  |  |  |  |
| --- | --- | --- | --- |
| Розділ | Консультант | Підпис, дата | |
| Завдання видав | Завдання прийняв |
| 1 | Єфремов Ю.М. | 04.02.2020 | 04.02.2020 |
| 2 | Єфремов Ю.М. | 15.04.2020 | 15.04.2020 |
| 3 | Єфремов Ю.М. | 01.05.2020 | 01.05.2020 |

Керівник \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис)

**Календарний план**

|  |  |  |  |
| --- | --- | --- | --- |
| №  з/п | Назва етапів випускної кваліфікаційної роботи | Термін виконання етапів роботи | Примітка |
| 1 | Постановка задачі. Аналізування вимог до програмного продукту | 4 лютого 2020 –  20 лютого 2020 | виконано |
| 2 | Пошук, огляд та аналіз аналогічних розробок. Формулювання технічного завдання. Опрацювання літературних джерел. | 21 лютого 2020 –  1 березня 2020 | виконано |
| 3 | Вивчення матеріалу. Планування структури програми. | 2 березня 2020 –  2 квітня 2020 | виконано |
| 4 | Розробка макету | 3 квітня 2020 –  10 квітня 2020 | виконано |
| 5 | Проектування структури системи | 11 квітня 2020 –  15 квітня 2020 | виконано |
| 6 | Написання програмного коду | 16 квітня 2020 –  26 квітня 2020 | виконано |
| 7 | Тестування | 27 квітня 2020 –  29 квітня 2020 | виконано |
| 8 | Передзахист | 30 квітня 2020 | виконано |
| 9 | Захист | 12 червня 2020 | виконано |

Студент\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис)

Керівник\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис)

РЕФЕРАТ

Зм.

Арк.

№ докум.

Підпис

Дата

Арк.

4

ІПЗ.КР.Б – 121 – 20 – ПЗ

Розроб.

Раздобудова В.О

Керівник

.

Єфремов Ю.М.

Рецензент

Грабар О.І.

Зав. каф.

*І.В.Пулеко*

**Arduino”Smart House”.Система керування функціями Arduino**

**Пояснювальна записка**

Літ.

Аркушів

Житомирська політехніка,

група ПІ-54

Випускна робота бакалавра являє собою програмний комплекс системи контролерів для реалізації концепції "розумний" будинок. Пояснювальна записка до випускної роботи містить 61 сторінок, 45 ілюстрацій та 16 таблиць.

Метою роботи є розробка системи керування функціями контролерів, встановлення протоколів зв’язку елементів системи, створення схеми системи управління та створення програми управління виконавчими приладами.

В роботі визначено основні завдання на розробку системи, проаналізовано аналоги мікрокомп’ютерів розробленої системи. Досліджено функції та можливості мікроконтролерів і датчиків.

В ході виконання випускної роботи було розроблено тестовий макет розумного будинку з датчиками вологості, температури, тиску, газу, освітлення, руху, загоряння. Представлено програмний комплекс та можливість керування деякими приладами.

Ключові слова: ARDUINO, "РОЗУМНИЙ" БУДИНОК, ТЕМПЕРАТУРА, ВОЛОГІСТЬ, ЗАГОРЯННЯ, ОСВІТЛЕННЯ.

ABSTRACT

The bachelor's thesis is a software package of controllers for the implementation of the concept of "smart" house. The explanatory note to the thesis contains 61 pages, 45 illustrations and 16 tables.

The purpose of the work is to develop a control system for the functions of controllers, to establish protocols for communication of system elements, to create a scheme of the control system and to create a control program for actuators. The work also contains main tasks of the system development; analogs of the developed system were analyzed.

The main tasks for system development are defined in the work, analogues of microcomputers of the developed system are analyzed. The functions and capabilities of microcontrollers and sensors are studied.

During the final work, a test model of a smart home with sensors of humidity, temperature, pressure, gas, lighting, movement, ignition was developed. The software package and the ability to control some devices are presented.

KEYWORDS: ADUINO, "SMART" HOUSE, TEMPERATURE, HUMIDITY, FIRE, LIGHTING.

ЗМІСТ

[**РЕФЕРАТ 4**](#_Toc514019495)

[**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ 7**](#_Toc514019497)

[**ВСТУП** 6](#_Toc514019498)

[**РОЗДІЛ 1. ТЕХНОЛОГІЯ “SMART HOUSE”** 9](#_Toc514019499)

[1.1. ІСТОРІЯ ТЕХНОЛОГІЇ 9](#_Toc514019500)

[1.2. ПРИНЦИП РОБОТИ СИСТЕМИ ТА ОСНОВНІ ХАРАКТЕРИСТИКИ 10](#_Toc514019501)

[1.3.ІСНУЮЧІ АНАЛОГИ.ПЕРЕВАГИ ТА НЕДОЛІКИ 14](#_Toc514019502)

[1.4. ОБҐРУНТУВАННЯ ВИБОРУ ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ ТА ВИМОГИ ДО АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ 16](#_Toc514019503)

[ВИСНОВКИ ДО РОЗДІЛУ 1 19](#_Toc514019504)

[**РОЗДІЛ 2. ПРОЕКТУВАННЯ СИСТЕМИ ТА АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ** 21](#_Toc514019505)

[2.1. АНАЛІЗ МОЖЛИВОСТЕЙ ТА МІКРОКОМП’ЮТЕРІВ 21](#_Toc514019510)

[2.2. АНАЛІЗ МОЖЛИВОСТЕЙ ПЕРИФЕРІЙНИХ ПРИСТРОЇВ 33](#_Toc514019511)

[2.3. АНАЛІЗ МОЖЛИВОСТЕЙ ДАТЧИКІВ 37](#_Toc514019512)

[2.4.СХЕМА ТЕСТОВОГО СТЕНДУ 39](#_Toc514019513)

[ВИСНОВКИ ДО РОЗДІЛУ 2 44](#_Toc514019514)

[**РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ** 45](#_Toc514019515)

[3.1. ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ARDUINO 45](#_Toc514019516)

[3.2. ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ANDROID ТА БАЗИ ДАНИХ. 45](#_Toc514019517)

[ВИСНОВКИ ДО РОЗДІЛУ 3 55](#_Toc514019519)

[**ВИСНОВКИ** 56](#_Toc514019520)

[**СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ** 58](#_Toc514019521)

[**ДОДАТКИ** 59](#_Toc514019522)

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ПЗ – Програмне забезпечення.

ОС – Операційна система.

framework – Готовий до використання комплекс програмних рішень.

скетч – Блок коду для Arduino.

SPI – Послідовний периферійний інтерфейс.

КМ – Контактна мережа (дроти живлення).

КБ – Одиниця вимірювання даних.

МБіт– Кількість інформації, біт.

ВСТУП

**Актуальність теми.** Оскільки стрімка зміна технологій завжди напрямлена на службу людству, бажання жити простим, але прогресивним життям зростає. Автоматичні системи у світі обираються замість ручних. Зі стрімким зростанням кількості користувачів Інтернетом за останнє десятиліття, Інтернет став частиною людського життя.

Вбудовані фізичні прилади, такі як побутові прилади стають все розумнішими і розумнішими. Вони остнащені вбудоманими мікропроцесорами та бездротовими прийомопередавачами, надаючи нам комунікаційні можливості та забезпечуючи хорошу роботу за невеликі кошти.

Щоденні об’єкти оснащені невеликими, дешевими мобільними процесорами, датчиками та виконуючими механізмами. Датчики та бездротові сенсорні функції впроваджуються в рішення ‘розумного будинку’, точно визначаючи умови навколишнього середовища в будинку.

Ефективне задоволення потреб у сфері ‘розумного будинку’ потребує багатопрофільного співпрацювання. Як правило, в таких розробках приймають участь спеціалісти з області архітектури, єлектротехніки, радіотехніки. Деякі розробки можуть також потребувати спеціалістів з області медицини або біоінженерії.

Домашня автоматизація включає в себе централізоване управління освітленням, температурою та вологістю, контролем загорання та датчиком руху. А також приладами управління та інше. Автоматизація ‘розуманого будинку’ напрямлена на зростання ефективності та безпеки використання енергії для надання комфорту користувачу.

Задачею інтеректуального будинку є не лише практичність, але й зниження використання ресурсів таких як енергія. Скорочуються не лише фінансові витрати, але ця технологія також набула ще більшой актуальності завдяки екологічної користі.

**Метою випускної роботи** є розробка дистанційно керована система автоматизації ‘розумного будинку’, призначена для моніторингу і контролю за температурою, освітленням, процентом вологості, витіком газу, пожежою.

Встановлена мета обумовлює наступні завдання:

* проведення аналізу мікроконтролерів та різних датчиків;
* визначення архітектури та узагальненої структури системи;
* обґрунтування та вибір засобів реалізації системи;
* розробка програмного забезпечення системи;
* проектування тестового стенду;
* тестування програмного комплексу.

**Об’єктом дослідження** є концепція та розробка функціональної схеми упарвління 'розумним будинком'.

**Предметом дослідження** є датчики, мікроконтролери та мікрокомп’ютери на базі Arduino.

Розроблена система може бути використана для навчання базовим навичкам автоматизації системи керування функціями ‘розумного будинку’.

Пояснювальна записка до випускної роботи містить 61 сторінок, 45 ілюстрацій та 16 таблиць, список літературних джерел містить 26 найменувань.

1. ТЕХНОЛОГІЯ “SMART HOUSE”

## Постановка задачі

Головною метою даної роботи є ознайомлення з технологією ‘розумного’ будинку та представлення недорогої, гнучкої та надійної системи домашньої автоматизації з використанням мікроконтролера Arduino, з IP-з’єднанням через локальну WIFI мережу для доступу та керуванням приладами авторизованим користувачем віддалено за допомогою смартфона. Запропонавана система не залежить від серверу та допомагає користувачу контролювати роботу різних приборів, параметри мікроклімату у приміщенні. Користувач зможе приймати рішення на основі зворотнього зв’язку з датчиками дистанційно.

Для демонстрації ефективності та реалізації цієї системи, в даній випускній роботі представлено систему домашньої автоматизації, яка використовує мікроконтролер Arduino Mega 2560 та Ethernet шилд W5100. Для реалізації поставленого завдання, основними етапами є:

1. Розробка макету
   1. Створити тестовий стенд для підключення основного мікроконтролеру
   2. Підключеня модулів керування на основі Arduino:
      1. Модуль W5100 Ethernet Shield, що встановлюється на плату Arduino Mega 2560
      2. Модуль SD Card Reader
      3. Підключення датчиків DHT22, BMP180
      4. Підключення фоторезистору
      5. Підключення сенсору вогню, гасу
      6. Підключення світлодіодів
2. Проектування та розробка програмного забезпечення, яке буде виконуватися на Arduino
   1. Слідкувати за значенням сенсорів
   2. Взаємодіяти з Сервером через протокол TCP/IP
   3. Зчитувати дані з SD-карти та виконувати відповідні функції
   4. Модуль погоди
3. База даних та методи роботи з нею
4. Файл для створення та редагування конфігурацій для міктроконтролера

Результатом реалізації поставленого завдання є функціональна система керування за концепцією ‘розумний’ будинок.

## Принцип роботи системи та основні характеристики

Майже будь-яка система, що допомагає спростити життя людині та зробить її більш комфортною може вважатия ‘розумнім’ будинком, адже цей термін не має однозначного визначення. Рішення ‘на всі випадки життя’ не існує, саме тому, проектування починається з визначенням поставлених задач. Це дозволяє розширити коло реалізацій з ріним рівнем інтеграції та основним принципом роботи. Але все ж вбудовані системи можна поділити на такі три групи:

* З центральним контролером;
* Без центрального контролера;
* Система з інтеграцією, налаштовується;

До першої групи відносяться вже налаштована виробником система, яка керується через центральний обчислювальний пристрій. Ця система не передбачає взаємодії компонентів між собою. Все налаштування зберігаються на сервері, а периферійні пристрої тільки виконують відповідні інструкції. Також зазвичай не мають вбудованої пам’яті. Принцип роботи системи зображений на рис. 1.1.



Рисунок 1.1 – Принцип роботи вбудованої системи з центральним контролером

До другої групи відносяться напівавтономні пристрої. В такому випадку у пам’ять кожному пристрою прописується алгоритм, а для того щоб щось змінити, потрібно буде перепрограмовувати прилад. Принцип роботи на рис.1.2.



Рисунок 1.2 – Принцип роботи вбудованої системи без центрального контролера

Третю групу складають контролери, що приєднуються до звичайних приладів та залежно від показання вбудованих сенсорів, регулюють роботу цих приладів. Налаштовуються через Інтеренет або хмарний сервіс, але все ж можуть мати центральний комп’ютер. Контролери є незалежними між собою, тому для налаштування зв’язку можут знадобитися додаткові датчики або сенсори. Принцип роботи на рис. 1.3.



Рисунок 1.3 – Принцип роботи системи з інтеграцією, що налаштовується

Перечислюючи основні характеристики 'розумного' дому, можна виділити декілька основних слів: комфорт, енергоефектвність, безпека, безперебійність, автоматизація приміщення та функціональність. Дана система має можливість прилаштовуватися до звичок користувача і таким чином 'турбуватися' про нього.

Таким чином, наприклад, система регулює витрачення електроенергії пристроїв, які не використовуються в певний момент. Також, система потурбується про безпеку, зачинивши вхідні двері та поставить на охорону периметр всього будинку. Для того щоб проконтролювати свій будинок на відстані достатньо буде використати лише смартфон.

'Розумний' будинок визначається такими основними параметрами:

* Безпека-охорона. Забезпечує безпеку будинку як у присутності, так і у відсутності господаря
* Безпека-контроль. Застерігає від витіку гасу, води, контролює момент займання
* Функція сервісу та комфорту. Підтримує в приміщенні оптимальну температуру. Автоматично регулювати освітлення залежно від часу доби або по бажанню господаря. Відкривати/закривати жалюзі, ворота.
* Взаємодія. Одна з головних особливостей 'розумного' будинку полягає в об'єднаянні різних пристроїв в одну систему. Якщо взяємодія пристроїв проходить максимально легко – така система буде вважатися найбільш відкритою. Багато виробників застосовують технології базовані на Z-Wawe, що включає в себе бездротове управління. Цей елемент допомагає всім пристроям злагоджено працювати між собою.
* Віддалений доступ. Інша основна характеристика будь-якого ‘розумного’ будинку полягає у можливості користувача швидко і легко змінювати налаштування за необхідності.

Вище згадані параметри є основними характеристиками концепції 'розумний' будинок. Реалізація даних функцій залежить від віробника та призначення. Сама реалізація не стандаризується, тому виробник сам вирішує яким чином реалізувати дану концепцію.

## Існуючі аналоги. Переваги та недоліки.

На даний момент сучасний ринок представляє величезний вибір різних систем для автоматизації житлового приміщення. В США та Канаді такі програмно-апаратні комплекси можуть використовуватися у користувачів з різними фінансовими можливостями, так як таке обладнання сьогодні досить доступне. Важливо й те, що можливість коритуватися функціями інтелектуального будинку людьми, які мають якісь фізичні вади. Тому ми ознайомимося та проаналізуємо системи 'розумного' будинку від різних виробників. [1]

Умовно різні системи поділилися на три основні групи за призначенням:

* Мультимедійний простір
* Контроль мікрокліматичних параметрів приміщення
* Змішана система

Також виділимо основні критерії оцінювання:

* Складові частини системи та її можливість до масштабування
* Порядок та спосіб підключення пристроїв між собою
* Спосіб зв'язку з користувачем
* Вартість
* Радіус дії сигналу

Перша позиція – рішення від Ajax. Є вітчизняною розробкою та має можливість не лише забезбечити комфорт в управлінні приміщенням, але й гарантовано захистить житло в повній мірі. Працює на захищеному шифруванням радіозв’язку Jeweller (власна розробка). Переваги: простий у використанні та встановленні, бездротовий канал зв’язку, велика зона дії сигналу(до 2000), Wi-Fi і GSM-зв’язок, різні способи інформування користувача, показує витрати електроенергії, управління за допомогою як Android та і iOS, наявність тривожної кнопки на пульті, невисока вартість комплекту. Недоліки: функціонування лише з роботи центрального контролера, немає автономних датчиків, відсутня камера відеонагляду, управління тільки через смартфон. Приклад цієї системи наведено на рис. 1.4.



Рисунок 1.4 – Приклад системи Ajax

Рішення від Orvibo являє собою достпуний по ціні комплект обладнання, основною задачею якого є надання безпеки житлу. Також можна організувати повноцінну систему ‘розумний’ будинок. Виробник: Китай. Присутній англійський інтерфейс. Переваги: простий у використанні, віддалене управління через смартфон, можна додатково встановити різноманітні додаткові пристрої, можливість масштабування, наявне відеоспостереження, бездротовий простокол взяємодії між контролером і датчиками (ZigBee), вибір сценарії для роботи з житлом, доступна вартість. Недоліки: невелика зона дії сигналу (до 30м), невелика базова комплектація, для надійної роботи потрібне дротове підключення, відсутність резервного живлення, примітивні незахищені від взлому датчики, камера спостереження розрахована лише на роботу всередині приміщення. Приклад цієї системи наведений на рис. 1.5.



Рисунок 1.5 – Приклад системи Orvibo

Рішення від Xiaomi належить до бюджетного варіанту ‘розумного’ будинку, але в тей же час дозволяє управляти різними пристроями максимально просто і зручно. Якщо використовувати додаткові сенсори та пристрої можна створити досить функціональну систему контролю житла. Переваги: пристрої автономні, можливість масштабування, наявність власної камери, компактність, зручне управління за допомогою смартфона, низька вартість. Недоліки, маленька зона дії (до 10м), невелика базова комплектація з виконавчих пристрої, різні датчики потребують своє положення. Приклад цієї системи наведений на рис. 1.6.



Рисунок 1.6 – Приклад системи Xiaomi

Порівняльну таблицю рішень наведено у таблиці 1.1.

Таблиця 1.1

|  |  |  |  |
| --- | --- | --- | --- |
| Параметр | Ajax | Orvibo | Xiaomi |
| Вартість | Середня | Середня(але без додаткових модулів) | Невелика |
| Установка | Проста | Потребує налаштування | Проста |
| Налаштування | Не потребує | Прошивка | Додаткове налаштування для датчиків |
| Готові модулі | Багато | Багато | Мало |
| Масштабованість | Масштабується | Масштабується | Масштабується |
| Взаємодія компонентів | Лише через смартфон | Через сам пристрій | Через хмарний додаток |
| Функціонал | Майже необмежений | Базовий | Базовий |

## Обґрунтування вибору інструментальних засобів та вимоги до апаратного забезпечення

Реалізація розумного контролю мікрокліматичних параметрів з вбудованою функцією енергозбереження. Головна задача проекту – створити недорогу систему управління функціями на базі Arduino. Дана в проекті система відноситься до групи з вбудованим центральним контролером та архітектурою модульного типу. Розробка повинна дозвляти звичайному користувачу налаштувати систему так, як буде зручно. Скаладається з трьох основних частин:

* Модулі для контролю кліматотехніки, освітлення
* Центральний контролер
* Модулі контролю займання, гасу

База для реалізації ‘розумного’ будинку на рис.1.6.



Рисунок 1.6 – База для реалізації ‘розумного’ будинку

Для запрограмування макету Arduino можна скористатися офіційним програмним забезпеченням (повністью безкштовне для завантаження). Окрім великого різноманіття підключаємих приладів, більшої гручкості їй додає середовище програмування С++. Користувач має можливість самостійно запрограмувати дії компонентів системи.

## Висновки до розділу 1

Проаналізувавши предметну область можна підсумувати, що ‘розумний’ будинок є сучасно оснащене житло, організоване для полегшення життя людини за допомогою автоматизованої системи різних приладів та високотехнологічних пристроїв.

Основні концепції:

* Мультимедійне житло
* Контроль мікрокліматичних параметрів приміщення
* Змішана система

Основні хаактеристики:

* Віддалений доступ
* Масштабованість
* Взаємодія
* Установка

Основні характеристикаи для власної системи:

* Енергозбреження
* Збір інформації за допомогою датчиків
* Кастомізація
* Реалізація прийняття рішень на подаваємі команди

Основні функції власної системи:

* Безпека-охорона
* Безпека-контроль
* Функція сервісу та комфорту

1. [ПРОЕКТУВАННЯ](file:///C:\Users\Anna\AppData\Roaming\Skype\бакалаврат\fragment_bakalavr.docx#_Toc212894718) СИСТЕМИ ТА АПАРАТНЕ ЗАБЕЗПЕЧЕННЯ

## Аналіз можливостей мікроконтролерів та мікрокомп’ютерів

Домашня автоматизація є достатньо складна задача не лише для розробника, але й для користувача. Розробка має задовольнити всі потреби майбутнього користувача, а для конструювання необхідна досить велика кількість різних датчиків, модулів та пристроїв.

Мікроконтролер – спеціалізований комп’ютер у вигляді мікросхеми, що включає в себе мікропроцесор, оперативну або постійну пам’ять, порти вводу-виводу. Мікроконтролер являє собою один з центральних елементів ‘розумного’ будинку. Керує різними пристроями, а також знімає показники з датчиків.

Arduino – платформа для програмування електроних пристроїв. Основними компонентами є плата вводу-виводу. Відкрита система та архітектура з середовищем розробки на мові Processing-Writing. Іншими словами – це невелики пристрій, забезпечуючий керування датчиками, системами освітлення, прийому та передачі даних.[7]

**Arduino UNO** мікроконтролер на базі ATmega328. Представлений на рис. 2.1 та рис.2.2. Характеристика мікроконтролера наведена у таблиці 2.1.

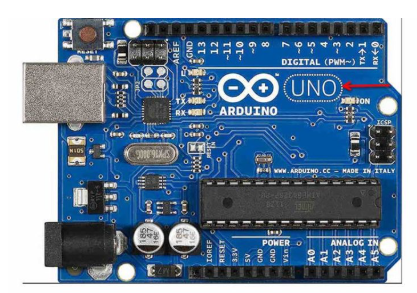


Рисунок 2.1 – Arduino UNO, вид зверху[7]

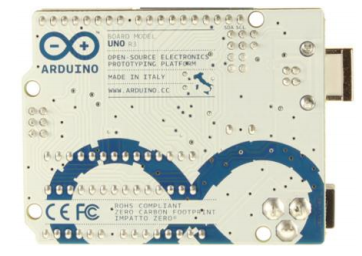


Рисунок 2.2 – Arduino UNO, вид знизу

Таблиця 2.1[7]

|  |  |
| --- | --- |
| **Параметр** | **Значення** |
| Мікроконтролер | ATmega328 |
| Робоча напруга | 5 В |
| Вхідна напруга (рекомедоване) | 7-12 В |
| Вхідна напруга (границя) | 6-20 В |
| Цифрові Входи/Виходи | 14 (6 з яких можуть використовуватися в якості ШІМ) |
| Аналогові входи | 6 |
| Постійний струм через вхід/вихід | 40 мА |
| Постійний струм для виводу 3.3 В | 50 мА |
| Флеш-пам’ять | 32 КБ (ATmega328) з яких 0.5 КБ використовуються завантажувачем |
| ОЗУ | 2 Кб (ATmega328) |
| EEPROM (незалежна пам’ять) | 1 Кб (ATmega328) |
| Тактова частина | 16 МГц |

**Arduino Mega2560** мікроконтролер на основі процесора ATmega2560. Представлений на рис. 2.3 та 2.4. Характеристика мікроконтролера наведена у таблиці 2.2.[8]

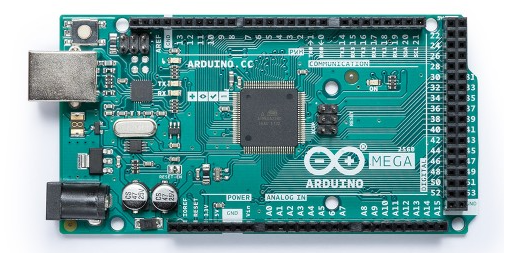


Рисунок 2.3 – Arduino Mega2560 вид зверху[8]

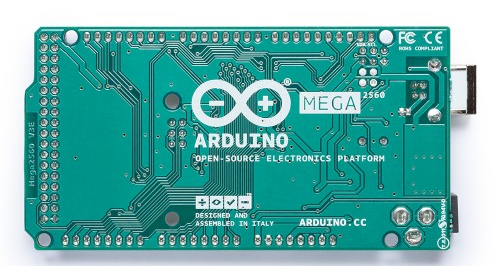


Рисунок 2.4 – Arduino Mega2560 вид знизу

Таблиця 2.2[8]

|  |  |
| --- | --- |
| **Параметр** | **Значення** |
| Мікроконтролер | ATmega2560 |
| Робоча напруга | 5B |
| Вхідна напруга (рекомендоване) | 7-12B |
| Вхідна напруга (границя) | 6-20B |
| Цифрові Входи/Виходи | 54 (14 з яких можуть працювати також в якості ШІМ) |
| Аналогові виходи | 16 |
| Постійний струм через вхід/вихід | 40 мА |
| Постійний струм для виведення 3.3 В | 50 мА |
| Флеш-пам’ять | 256 KB (з яких 8 КВ використовуються для завантажувача) |
| ОЗУ | 8 КБ |
| Енергонезалежна пам’ять | 4 КБ |
| Тактова частота | 16 МГц |

**RobotDyn DUE XPRO** плата на базі мікроконтролера Atmel SAM3X8E ARM Cortex-M3 та на базі 32-бітного ядра ARM. Представлений на рис. 2.5 та 2.6. Характеристика мікроконтролера наведена у таблиці 2.3.[10]

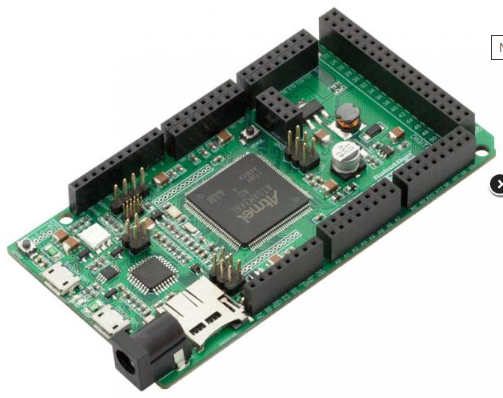


Рисунок 2.5 – RobotDyn DUE XPRO вид зверху[9]

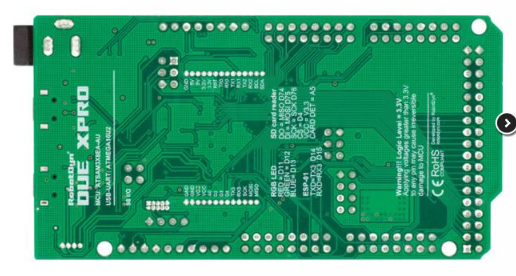


Рисунок 2.6 – RobotByn DUE XPRO вид знизу

Таблиця 2.3[9]

|  |  |
| --- | --- |
| **Параметр** | **Значення** |
| Мікроконтролер | RobotDyn DUE XPRO |
| Робоча напруга | 5В |
| Вхідна напруга (рекомендоване) | 7-12В |
| Вхідна напруга (границя) | 6-20В |
| Цифрові Входи/Виходи | До 98 |
| Аналогові виходи | 12 |
| Постійний струм через вхід/вихід | 40мА |
| Постійний струм для виведення 3.3 В | 50мА |
| Флеш-пам’ять | 512 Кб (2 блоки по 256 Кб) |
| Тактова частота | 84 МГц |

**Crowduino Uno-SD Elecrow** системна плата, яка може бути повністю сумісна з Arduino UNO. Побудована на мікроконтролері Atmega328P. Представлений на рис. 2.7 та 2.8. Характеристика наведена у таблиці 2.4.[10]

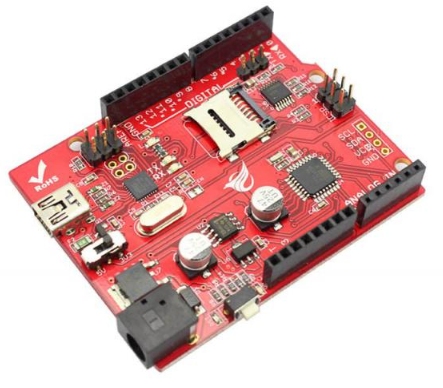


Рисунок 2.7 – Crowduino UNO-SD вид зверху[10]

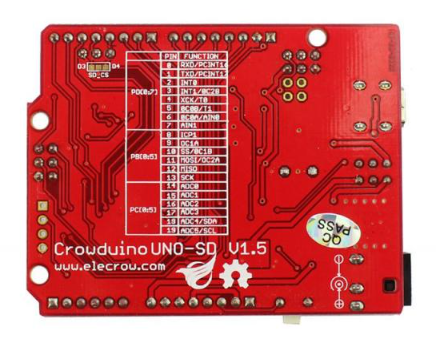


Рисунок 2.8 – Crowduino UNO-SD вид знизу

Таблиця 2.4[10]

|  |  |
| --- | --- |
| Параметр | Значення |
| Мікроконтролер | Atmega16U2 |
| Робоча напруга | 5В |
| Вхідна напруга (рекомендоване) | 7-12В |
| Вхідна напруга (границя) | 6-20В |
| Цифрові Входи/Виходи | 14 (з яких 6 можуть використовуватися як ШІМ) |
| Аналогові входи | 6 |
| Постійний струм через вхід/вихід | 40мА |
| Постійний струм для виводу 3.3 В | 50мА |
| Флеш-пам’ять | 32кБ |
| Тактова частина | 16 МГц |
| Енергонезалежна пам’ять даних | 1кБ |
| ОЗУ | 2кБ |

**Raspberry Pi 3 Model A+** одноплатий міні-комп’ютер з процесором ARM. Плата виділяється своїм маленьким розміром, користується великою популярністю. Використовується здебільшого для мобільних приладів. Представлений на рис.2.9 та 2.10. Характеристка мікроконтролера наведена у таблиці 2.5.[11]

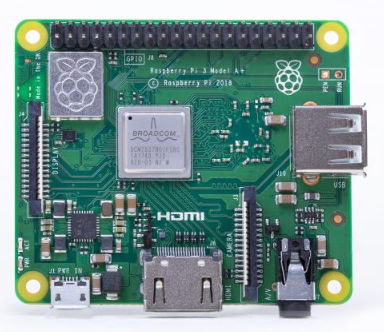


Рисунок 2.9 – Raspberry Pi 3 Model A+ вид зверху[11]



Рисунок 2.10 – Raspberry Pi 3 Model A+ вид снизу

Таблиця 2.5[11]

|  |  |
| --- | --- |
| **Параметр** | **Значення** |
| Мікропроцесор | BCM2837B0 |
| Робоча напруга | 5В |
| Архітектруа ядра | ARM11 |
| ЦПУ | 700МГц |
| gpio | 40pin |
| ОЗУ | 512МБ |
| Тактова частота | 1.4МГц |

## Аналіз можливостей периферійних пристроїв

**W5100 Ethernet Shield** один з важливих єлементів у підключенні пристроїв – дана плата розширеня бере участь у підключенні мікроконтролера до локальної мережі. Має влаштовану можливість взаємодіяти з microSD. Представлений на рис. 2.11 та 2.12. Характеристика шилду наведена у табиці 2.6.[12]



Рисунок 2.11 – Шилд W5100 Ethernet Shield вид звехру[12]

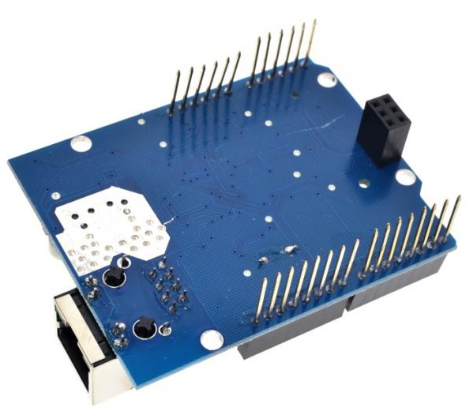


Рисунок 2.12 – Шилд W5100 Ethernet Shield вид знизу

Таблиця 2.6[12]

|  |  |
| --- | --- |
| **Параметр** | **Значення** |
| Плата розширення, шилд | W5100 Ethernet Shield |
| Робоча напруга | 5В(від основної плати Arduino) |
| Вбудований буфер | 16КБ |
| Швидкість | 10/100Мбіт |
| Взіємодія з мікроконтролером | Через SPI шину |

**SD Card Reader** в даному проекті дані будуть зберігатися на модулі карти пам’яті. Модуль представений на рис. 2.13 та 2.14.



Рисунок 2.13 – SD Card reader

**Relay Module 5V** високоякісне реле з напругою 5В, комутує до 10А 30В ДС та 10А 250В АС. Можна підключати напряму до виводу мікроконтролера. Реле дозволяє включати зовнішні прилади за допомогою розмикання та замикання електричної мережі, в якій вони знаходяться. Це допомагає керувати користувачу, наприклад, освітленням. Реле представлене на рис. 2.14.



Рисунок 2.14 – 5V Relay Module

**Реле SRD-5VDC-SL-C** електромеханічне реле. Напруга живлення 5В, комутує нагрузку до 10А 250В. До виводу мікроконтролера підключати не можна. Реле представлене на рис. 2.15.



Рисунок 2.15 – SRD-5VDC-SL-C

## Аналіз можливостей датчиків

**DHT11** даний цифровий датчик температури і вологості будемо використовувати в макеті. Його можна віднести до сімейсвта датчиків DHT11/DHT22/DHT21. Відрізняються точністью вимірювання за рахунок того, що містять в собі АЦП для перетворення аналогоих значень вологості та температури. Також є захищені від зовнішнього довкілля. Представлений на рис. 2.16. та 2.17. Характеристики датчиків представлені у таблиці 2.7.[14]



Рисунок 2.16 – Датчик DHT11

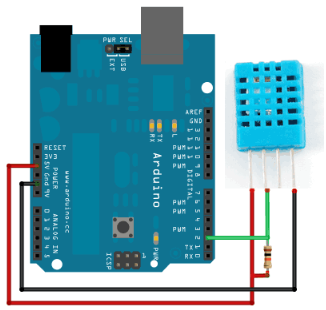


Рисунок 2.17 – Схема підключення DHT11

Таблиця 2.7[14]

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Датчик** | Діапазон  температури | Діапазон  вологості | Вага, гр | Округлення | Споживання | Умови |
| **DHT11** | 0-50 ºC ± 2%(макс.) | 20-90% RH ± 5%(макс.) | 1 | цілі | 0,1 | Приміщення |
| **DHT22** | -40...+80°C, ±0.5°C | 0-100%, ±2% | 2.2 | десяті | 0.14 | Приміщення |
| **DHT21** | -40...+125°C, ±0.5°C | 0-100%, ±2% | 11 | десяті | 1.36 | Вулиця |

**LM35DZ** аналоговий датчик з хорошою лінійністью та економічністью, що вимірює температуру в діапазоні від -55 до +150°C. Швидкодієвий і точний. Може бути альтернативою цифровому датчику. Представлений на рис. 2.18.

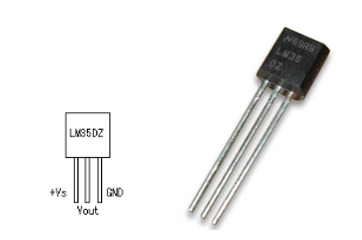


Рисунок 2.18 – датчик LM35DZ

**BMP180** – барометр, є цифровим датчиком для вимірювання тиску та температури. Діапазон вимірювання тиску: 300 – 1100 гПа; діапазон вимірювання температури: -40 до +85°C (+-2 градуси). Представлений на рис. 2.19.[15]

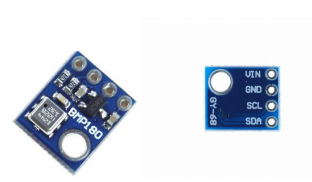


Рисунок 2.19 – датчик BMP180[15]

Датчик вогню **Flame Senson** базується на принципі інфрачервоних променів, що є надчутливі до вогню. Може вловлювати вогонь з довдиною хвилі 760 – 1100нм. Робоча температура датчика від -25 до 85°C, тому треба слідкувати за тим, щоб не наближати датчик надто близько до вогнища. Представлений на рис. 2.20.



Рисунок 2.20 – Датчик вогню

Датчик вогню взаємодіє з **Баззером**, простий електроний компонент, що сигналізує про займання. Також баззери широко використовуються в різноманітній побутовій техніці та іграшках. Перетворюють команди, що базуються на двубітній системі числення 1 та 0, в звукові сигнали. Представлений на рис. 2.21.



Рисунок 2.21 – Компонент баззер

**Газовий сенсор** – цей модуль має викоку чутливість до природного газу, пропану, ізобутану, коксовому газу та легко виявляє їх в приміщенні. При цьому, нечутливий до сигаретного диму або пару від приготування. Значення зміни супротиву використовується для розрахунку концентрації газу. Використовується для сигналізації витіку газу як в житлі так і на підприємствах. Швидко реагує, стабільний у роботі та має довгий строк служби. Простий у підключенні. Модуль, використаний у проекті – аналоговий. Представлений на рис. 2.22.



Рисунок 2.22 – Газовий датчик

**PIR (піроелектричний) датчик руху** – інфрачервоний датчик руху, що дозволяє зафіксувати рух від людини або доиашньої тварини на відстані до 3-х метрів, також можна відстань можна регулюати. Має два входи живлення та один цифровий для монтіторингу показників.

Даний міні-датчик ми будемот використовувати у проекті для автоматичного включення світла. Представлений на рис. 2.23. Характеристика датчика наведена у таблиці 2.8.



Рисунок 2.23 – PIR датчик руху

Таблиця 2.8

|  |  |
| --- | --- |
| **Параметр** | **Значення** |
| Датчик | PIR сенсор руху |
| Робоча напруга | 4,5В – 20В |
| Вихідна напруга | 3.3V TTL логіка |
| Дистанція виявлення | 3 м |
| Кут детектування | 80°-100° |
| Тривалість імпульсу при виявленні | 8 с |
| Робоча температура | -20 до +80 °C |
| Діаметр лінзи | 10 мм |

**Фотоелектричний аналоговий сенсор GL5516** – це резистор, супротив якого варіюється в залежності від сили падаючого світла. Базується на фотоелектричного ефекта полупровідника. Тобто, якщо світло більш інтенсивне – супротив зменшується і навпаки. Фоторезистор використовується для вимірювання та управління світлом. Застосування: детектор фотоспалаху, автоспалах для камери, контроль на підприємстві, фотоперемикач, електронні іграшки. Представлений на рис. 2.24. Характеристика сенсора наведена у таблиці 2.9.

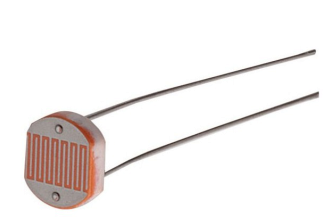


Рисунок 2.24 – Фотоелектричний сенсор GL5516

Таблиця 2.9

|  |  |
| --- | --- |
| **Параметр** | **Значення** |
| Сенсор | Аналоговий фотоелектричний GL5516 |
| Максимальна напруга | 150В |
| Максимальна потужність | 90мВт |
| Світлочутливий опір | 5-10кОм |
| Темновий опір | 0.5МОм |
| Час відгуку | 30 мс |
| Робоча температура | від -30 до +70 °C |

В проекті був використаний **фоторезисторний модуль світла від** **RobotDyn** освітленості цифровий та аналоговий. Представлений на рис. 2.25. Характеристика датчика наведена у таблиці 2.10.

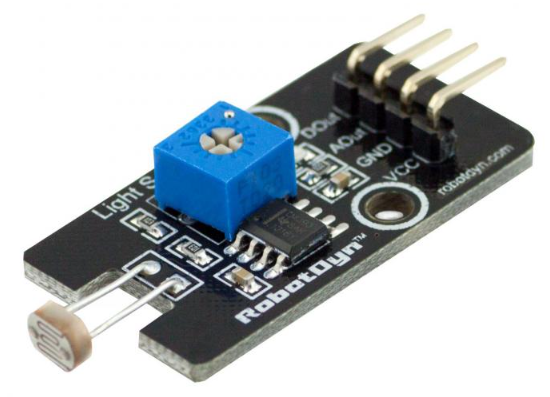


Рисунок 2.25 – фоторезисторний датчик від RobotDyn

Таблиця 2.10

|  |  |
| --- | --- |
| **Параметр** | **Значення** |
| Датчик | Фоторезисторний |
| Вихід компаратора | >15мА |
| Робоча напруга | Від 3.3В до 5В |
| Компаратор | LM393 |

## Схема тестового стенд

Спеціально розроблений тестовий стенд, представлений на рис. 2.26, в якому використали такі складові:

* Arduino Mega 2560
* W5100 Ethernet Shield
* SD Card Reader
* 5V Relay Module
* SRD-5VDC-SL-C Relay
* DHT11
* LM 35
* BMP 180
* Датчик вогню
* Баззер
* Газовий сенсор
* PIR датчик руху
* Фоторезистор GL 5516
* Фоторезистор датчик RobotDyn
* Світлодіоди (3 шт)
* Резистори (4.7 кОм)

В макеті використовуються компоненти від Keyestudio.[25] Схема підключення представлена на рис.2.27.

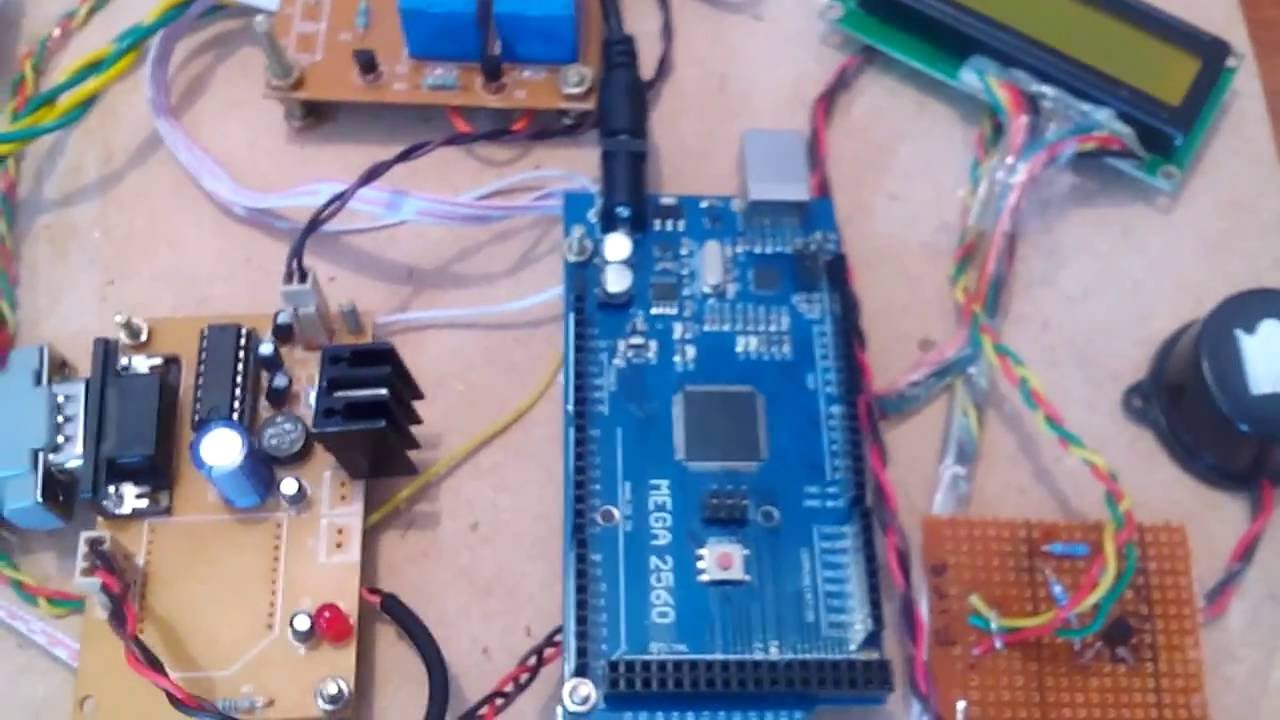


Рисунок 2.26 – Тестовий стенд

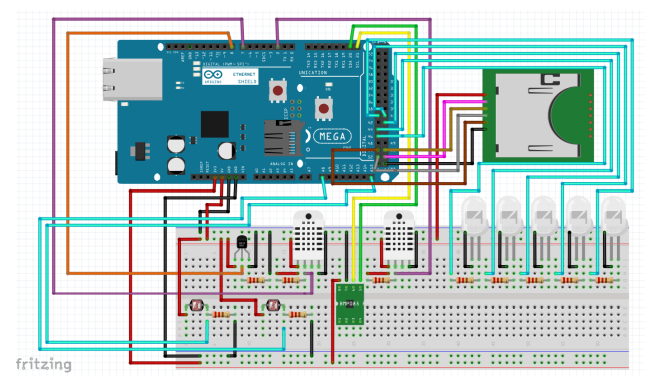


Рисунок 2.27 – Приклад схеми підключення

Підкючення виконується наспуними кроками:

1. Встановлюємо Ethernen шилд зверху на плату Arduino Mega 2560
2. Виконуємо підключення SD Card Reader’у

* CS до 52-го порту
* MOSI до 51-го порту
* SCK до 52-го порту
* MISO до 50-го порту
* GND до порту GND

1. Підключення датчиків

* DHT11 до 2-го і 7-го портів
* LM35 до 8-го порту
* BMP180 до роз’єму живлення 3.3В, GND, SCL до 21-го та SDA до 20-го портів

1. Підключення датчику газу до роз’єму живлення 5В, GND та A0
2. Під’єднуємо до нього датчик вогню – до живлення 5В і A0 та баззер через макетну плату – до 8-го порту
3. Підкючаємо на мекетну плану фоторезистори до 8-го та 15-го аналогових портів
4. PIR датчик руху підє’днуємо до живлення 5В, GND та 3-го порту
5. Для використання реле треба виконату підключення наступним чином

* GND до потру GND
* VCC до живлення 5В
* IN до відподнго цифрового порту

## Висновки до розділу 2

Для впровадження нашої домашньої системи ‘розумного’ будинку була розроблема експериментальна установка, в базі якої ми викроистали основний блок Arduino Mega 2560. Цей мікроконтролер доступний по ціні та задовльняє поставлені вимоги.

Так як ми будемо зберігати дані на карті пам’яті також було використано SD Card Reader та W5100 Ethernet шилд для роботи через TCP/IP.

Для зняття показників роботи нашої автоматизації були обрані датчики цифровий DHT11 температури і рівня вологості, аналоговий LM35 для температури, барометр BMP180 для зняття показників тиску.

Для виконання функції безпеки у приміщенні були використані сенсор вогню та газу, що під’єднані до звукового сигналу.

Для роботи з регулюванням світла було вирішено використати PIR сенсор руху, аналоговий та цифровий фоторезисторний датчик освітленості. Також для сигналізації про роботу реле були використані світодіоди.

Усі компоненти були надані виробником Keyestudio[25], це дозволяє створити нам тестовий макет для експериментальної реалізації концепції ‘розумний’ будинок.

РОЗДІЛ 3. РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## Програмне забезпечення Arduino

Програмування підключаємих елементів буде відбуватися в редакторі IDE.

В мережі присутній весь необхідний софт від Arduino Sofware. Рис.3.1[17]

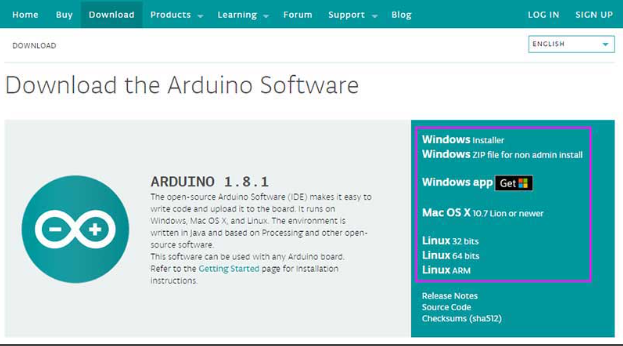


Рисунок 3.1 – Середовище розробки

Після того як ми підключаємо основну плату за допомогою USB кабеля, виконуємо встановлення необхідних драйверів для роботи рис. 3.2

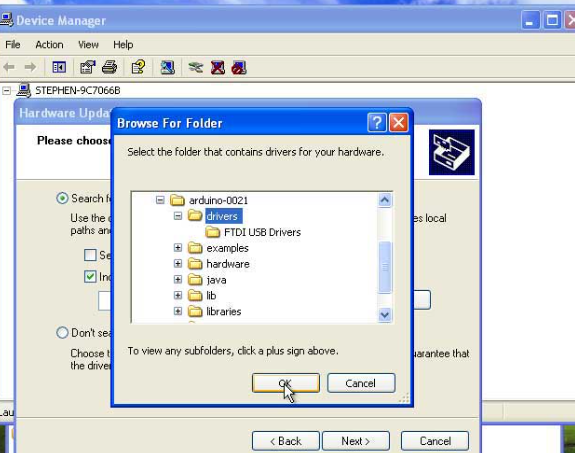


Рисунок 3.2 – Налаштування драйверів

Приклад середовища розробки зображений на рис. 3.3

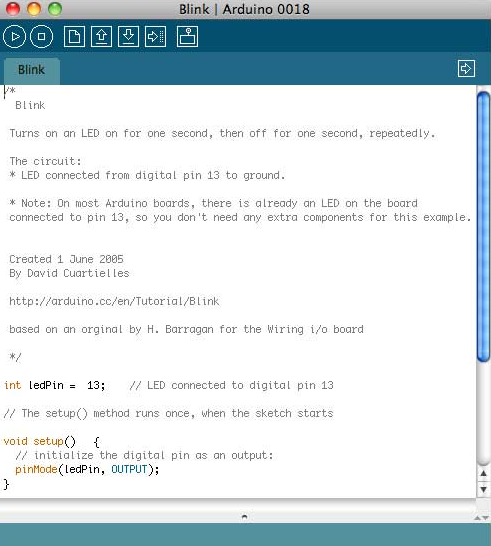


Рисунок 3.3 – Середовище розробки

Для роботи потрібно обрати необхідну нам плату рис. 3.4.

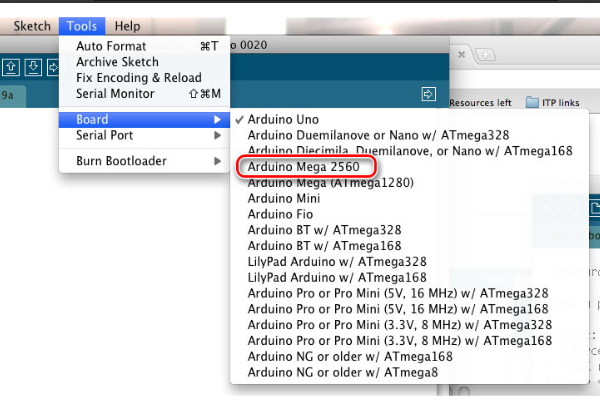


Рисунок 3.4 – Вибір необхідної плати

В даній випускній роботі необхідно розробити ПЗ, що буде виконувати наступне:

* Зчитувати встановлені конфігурацї з SD Car Reader
* Контролювати значення датчиків
* Влючати та виключати пристрої
* Робота через сервер TCP/IP

Будемо використовувати такі бібліотеки:

* SPI.h
* SD.h
* Ethernet.h
* Servo.h
* DHT.h
* SFE\_BMP180.H
* stdio.h
* string.h
* Wire.h
* OneWire.h
* DallasTemperature.h
* SoftwareSerial

Опис використання бібліотек наведений у таблиці 3.1.

Таблиця 3.1

Інформація про використання бібліотек

|  |  |
| --- | --- |
| **Бібліотека** | **Опис** |
| SPI.h | Необхідна для SD Card Reader та W5100 Ethernet Shield |
| SD.H | Відповідає за SD Card Reader |
| Ethernet.h | Відповідає за взаємодію з W5100 Ethernet Shield |
| DHT.h | Відповідає за зв’язок з датчиком DHT |
| stdio.h | Відповідає за роботу зі строками |
| string.h | Відповідає за роботу зі стандартними функціями |
| SFE\_BMP180.h | Відповідає за роботу з датчиком BMP180 |
| OneWire.h | Відповідає за роботу з датчиком LM35 |
| DallasTemperature.h | Відповідає за роботу з датчиком LM35 |
| Servo.h | Відповідає за роботу керуванням світлоіодами |
| SoftwareSerial.h | Дозволяє реалізувати послідовний інтерфейс |

Для реаліації текстового інтерфейсу по мережіта перердачу команд мікроконтролеру і отримання відповідей буде проходити через протокол Telnet.

Призначення даного протоколу полягає у наданні спільного, двонаправленого, восьмибітового байт орієнтованго засобу зв’язку. Тобто це допоможе злагодженій роботі між собою термінальним пристроям та процесам.

Код скетча містить функції та їх опис представлені у таблиці 3.2.

Таблиця 3.2

|  |  |
| --- | --- |
| **Функція** | **Виконує** |
| run\_config() | Зчитування файлу конфігурацій з карти пам’яті, первині налаштування мікроконтролеру |
| loop() | Працює під час роботи мікроконтролеру |
| setup() | Первині ініціалізації та запуск run\_config() |
| rewrite\_etalon() | Виконує видалення-перезапис функцій |
| check\_light() | Ввімкнення або вимкнення пристроїв за певних показників сенсора |
| SendTextMessage(uint8\_t warningType) | Відповідає за відправку текстового повідомлення на вказаний номер( з урахуванням коду країни), як попередження про загрозу |
| CheckFire() | Перевірка на загрозу займання, у разі небезпеки відправляється повідомлення |

У даному проекті було по можливості використано менше функцій за для швидкої роботи мікроконтролера. Також, через це ми зменшили використання ООП.

У початковий процес при первинному налаштуванні ми включаємо лексичний аналіз конфіурацій. Створюємо та заповнюємо відповідні масиви для збереження даних та зчитуванння сенсорів, а також відбуважться зчитування еталоних параметрів. Результат виводу представлений на рис. 3.5.

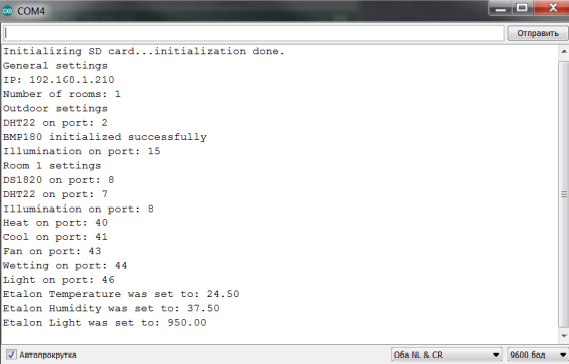


Рисунок 3.5 – Функція run\_config

У таблиці 3.3 наведений список команд, які були використані.

Таблиця 3.3

|  |  |
| --- | --- |
| **Команда** | **Функція** |
| set | За допомогою цієї команди ми можемо встановлювати значення таких параметрів як: temperature, humidity, ilumination. Також встановлюється еталоний параметр |
| Sensor\_request | Запит на показники датчиків: outdoor – зовнішній та room – кімната |
| Get\_rooms\_amount | Кількість кімнат у приміщенні, я яких зчитуються показники |
| logout | Роз’єднання протоколу передачі даних |

Команда set представлена на рис. 3.6.

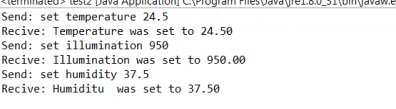


Рисунок 3.6 – Команда set

Команда sensor\_request представлена на рис. 3.7.

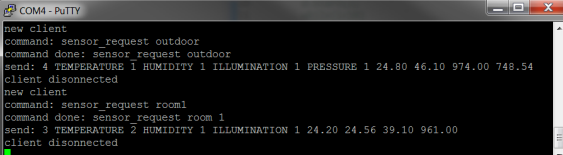


Рисунок 3.7 – Команда sensor\_request

Команда get\_rooms\_amount представлена на рис. 3.8.

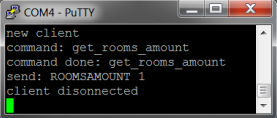


Рисунок 3.8 – Команда get\_room\_request

## Програмне забезпечення Android та Бази Даних

В нашому проекті реалізація передбачає використання пристрою з ОС Android. Саме тому необхідно розробити відповідні класи для взаємодії з Arduino. В якості сережовища розробки була використана Android Studio.

Для реалізації мають виконуватися певні функції, а саме:

* Зчитувати дані з мікроконтролера
* Задавати дані на мікроконтролер
* Обробляти та зберігати показники
* Запам’ятовувати конфігурації з’єднання
* Автономно контролювати датичики всередині на зовні приміщення

Розроблений Arduino Telnet, який несе в собі такі класи:

* Arduino
* Conector
* Room
* Sensor
* SmsMaker

Діаграма представлена у додатках. Класи та опис наведено у таблиці 3.4.

Таблиця 3.4

|  |  |
| --- | --- |
| **Класс** | **Опис** |
| Arduino | Базовий; Ініціалізує всі компоненти, запуск та зупинку команд. Наявні два конструктори:   * public Arduino(String hostnam, int hotport, SensorDataBase) * public Arduino(String hostname, int hotpost)   Методи: |

Продовження таблиці 3.4

|  |  |
| --- | --- |
| Arduino | * void start (int) * void stop() * Sensors getLost() * void setTemperature(float) * void setHumidity(float) * void setIlumination(float) |
| Connector | З’єднання з мікроконтролером; Наявні два контсруктори:   * Conector(sring hostname) * Conector(string hostname, int) |
| Room | Зберігає показники датчиків у приміщенні і має такі поля:   * humidity (вологість) * temperature (температура) * illumination (освітленість)   Методи:   * synchronized void set Humiditi(float) * get Humiditi() * void set Temperature(float) * get Temperature() * void set Ilumination(float) * get Illumination |

Продовження таблиці 3.4

|  |  |
| --- | --- |
| Sensors | Збереження інформації датчиків як у приміщенні, так і ззовні  Поля:   * roomsNunber * humidity * temperatute * ilumination * pressure   Та масив:   * rooms[]   Методи:   * void admeasuremnts(float temperature, humidity, float ilumination, presure) * add Room byIndex(int index room) * void set Temperature(float) * void set Humiditi(float) * void set Ilumination(float) * void set Presure(float) * int get Room number() |
| SmsMaker | Від реакції відповідних датчиків, відправляє текстове повідомлення на номер телефону ( з урахуванням коду країни), smsMaker(uint8\_t, int messageType) |

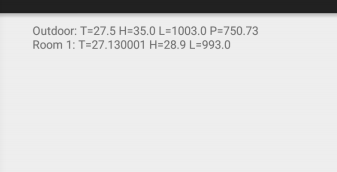


Рисунок 3.9 – Дані отримані з Arduino

Для зручності кориснуванням інформацією, її збереження та вільного доступу до неї була реалізована база даних. Також, при бажанні, дані можна експортувати.

Саме для цього була використана мова SQ Lite.[20]

Під ОС Android був розроблений пакет, в якому наявні такі каси:

* SensorDBHelper
* SensorDB

Діаграма класів представлена на рис.3.10

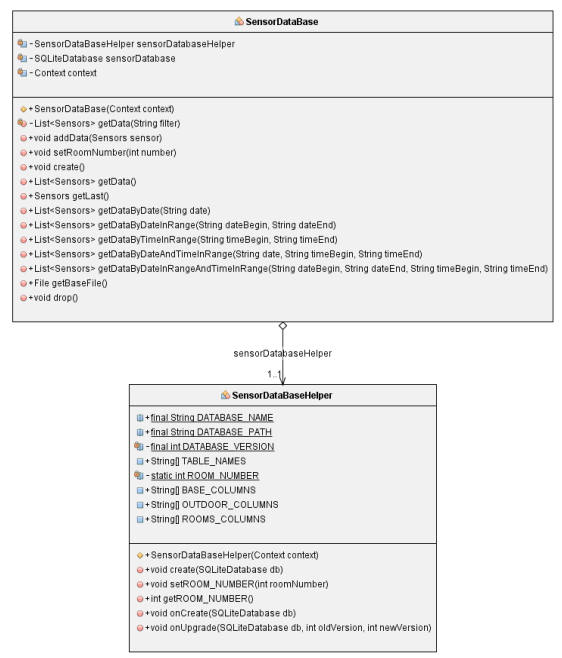


Рисунок 3.10 – UML діаграма класів

Перейдемо до створення бази даних. Нижче представлена база даних на прикладі однієї кімнати рис. 3.11 та дані про неї в таблиці 3.5. Ми будемо мати дані про стан як зсередини приміщення, так і зовнішній.

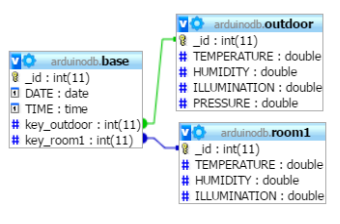


Рисунок 3.11 – База даних на прикладі однієї з кімнат

Таблиця 3.5

|  |  |
| --- | --- |
| **Стовбці** | **Опис** |
| base | * \_id (унікальний) * Date(дата запису) * Time(дані про час) * keyoutdoor * keyroom1 |
| outdoor | * \_id(унікальний) * temperature(температура) * humidity(вологість) * ilumination(світло) * pressure(тиск) |
| room | * \_id * temperature * humiditi * ilumination |

Результат роботи з базою даних представлений на рис. 3.12.

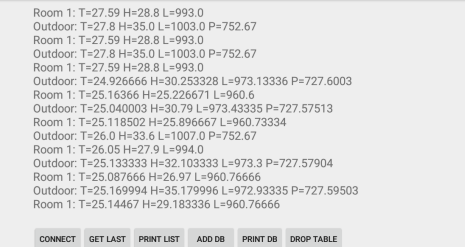


Рисунок 3.12 – Результат від бази даних

## Висновки до розділу 3

В даному розділі була проведена розробка програмного забезпечення для Arduino використовуючи протокол Telnet. Тако, був розроблений скеч з усіма необхідними функціями. Для зручності використання користувачем системи атоматизації був розроблений пакет Arduino Telnet під ОС Android.

Для зручного користування, зчиування, збереження та експорту інформації також була розроблена База даних на базі SQ Lite.

# ВИСНОВКИ

У ході виконанння випускної роботи бакалавра було реалізовану концепцію системи керування ‘розумним’ будинком Smart House, а саме частина з керування функціями Arduino. Метою цієї випускної роботи бакалавра була реалізація концепції інтеректуальної системи з атоматизації будинку Smart House.

Під час дослідження та розробки було вирішено наступні завдання:

1. Досліджено та проведено аналіз різноманітних мікроконтролерів різних виробників. Виходячи з цього, був обраний найбільш оптимальний варінт мікроконтролера – Arduino Mega 2560. Одна з найбільших переваг такої системи автоматизації є економічність та фінансова доступність. По своїм характеристикам він є практичним та доступним по ціні.
2. Був проведений аналіз периферійних пристроїв та різноманітни датчиків, адже однією з основних задач є контроль навколишнього середовища житла та моніторинг електроних приладів. Були підібрані підходящі датчики для моніторингу показників.
3. У проекті ми задовольнили такі функції системи, як: сервіс-комфорт та безпека-контроль. А саме: моніторинг вологості, тиску та температурних датчиків. Присутній сенсор витіку газу та сигнальна і текстова реакція на займання всередині приміщення. А також автоматизація освітленості за допомогою датчка руху, що дає нам економію енергії.
4. У проекті реалізоване ПЗ для мікроконтролера. Значною перевагою є те, що присутня можливість зміни налаштувань під вимоги користувача без втручання у код програми.
5. В роботі була використана локальна Інтернет мережа. А взаємодія між мікрокомп’ютером та мережею проходить через протокол Telnet.
6. Для передачі та контролю даних на OC Android був розроблений пакет Arduino.
7. Для функції збереження показників при виконанні даної роботи була створена База даних на мові SQ Lite.
8. Було детально розглянуто всіх аспектів реалізації: опис роботи мікроконтролера, його взаємодія з приладами, описано класи та методи, що були використані.
9. Система автоматизації, яка була реалізована видає усі необхідні дані про стан датчиків та навколишнього середовища у приміщенні та зовні.

Накінець, в ході виконання даної випускної роботи ми отримали гнучку систему автоматизації з концепцією ‘розумного’ будинку Smart House. Було розроблене підходяще програмне забезпечення системи та база даних для зручного моніторинку та взаємодії з даними.

Так як система має пряме практичне застосування, в подальшому може бути розширена та удосконалена.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Vencom [Електронний ресурс] – Режим доступу до ресурсу: <https://vencon.ua/ua/articles/reyting-sistem-umnyy-dom-po-proizvoditelyam>.
2. Iot [Електронний ресурс] – Режим доступу до ресурсу: <https://iot.ru/wiki/umnyy-dom>.
3. Lutron Electronics, Inc. - Dimmers And Lighting Controls. [Електронний ресурс] – Режим доступу до ресурсу:

<http://www.lutron.com/en-US/Company-Info/Pages/AboutUS/OurStory.aspx>.

1. Pico Electronics. [Електронний ресурс] – Режим доступу до ресурсу: http://www.picodenshi.com/about-pico30238.html.
2. HomeKit [Електронний ресурс] – Режим доступу до ресурсу:

https://developer.apple.com/homekit/.

1. Products - Clipsal by Schneider Electric. [Електронний ресурс] – Режим доступу до ресурсу: https://www.clipsal.com/Home-Owner.
2. Arduino Uno. [Електронний ресурс] – Режим доступу до ресурсу:

http://arduino.ua/ru/hardware/Uno.

1. Arduino Mega 2560. [Електронний ресурс] – Режим доступу до ресурсу: http://arduino.ua/ru/hardware/Mega2560.
2. chipKIT Uno32™ Prototyping Platform. [Електронний ресурс] – Режим доступу до ресурсу: http://www.digilentinc.com/Products/Detail.cfm?Prod=CHIPKIT-UNO32.
3. chipKIT Max32™ Prototyping Platform. [Електронний ресурс] – Режим доступу до ресурсу:

http://www.digilentinc.com/Products/Detail.cfm?Prod=CHIPKIT-MAX32.

1. RASPBERRY PI 1 MODEL A+. [Електронний ресурс] – Режим доступу до ресурсу: https://www.raspberrypi.org/products/model-a-plus/.
2. Плата расширения Arduino Ethernet. [Електронний ресурс] – Режим доступу до ресурсу: http://arduino.ua/ru/hardware/EthernetShield.
3. DS18B20 - Датчик температуры цифровой. [Електронний ресурс] – Режим доступу до ресурсу: http://www.sinava.ru/DS18B20.php.
4. Датчики влажности DHT11 и DHT22. [Електронний ресурс] – Режим доступу до ресурсу: http://homessmart.ru/index.php/oborudovanie/datchiki/datchiki-vlazhnosti-dht11-i-dht22.
5. BMP180 цифровой модуль атмосферного давления Arduino. [Електронний ресурс] – Режим доступу до ресурсу:

http://greenchip.com.ua/23-0-100-0.html.

1. СФ2-5а. [Електронний ресурс] – Режим доступу до ресурсу:

http:// www.giricond.ru/files/sf2a.pdf.

1. SD Library. [Електронний ресурс] – Режим доступу до ресурсу:

http://www.arduino.cc/en/Reference/SD.

1. Ethernet library. [Електронний ресурс] – Режим доступу до ресурсу:

http://www.arduino.cc/en/Reference/Ethernet.

1. Android Studio Overview. [Електронний ресурс] – Режим доступу до ресурсу: http://developer.android.com/tools/studio/index.html.
2. Android: SQLite. [Електронний ресурс] – Режим доступу до ресурсу: http://developer.android.com/reference/android/database/sqlite/SQLiteDatabase.html.
3. ElektrikExpert [Електронний ресурс] – Режим доступу до ресурсу: <https://elektrikexpert.ru/arduino.html>.
4. Future2day [Електронний ресурс] – Режим доступу до ресурсу: <https://future2day.ru/umnyj-dom-na-osnove-arduino/>.
5. SovetIngenera [Електронний ресурс] – Режим доступу до ресурсу: <https://sovet-ingenera.com/umniy-dom/chto-takoe-umnyj-dom.html>.
6. Arduino.ua [Електронний ресурс]

<https://arduino.ua/prod3587-kontroller-due-xpro-arm-na-cortex-atsam3x8ea-au>

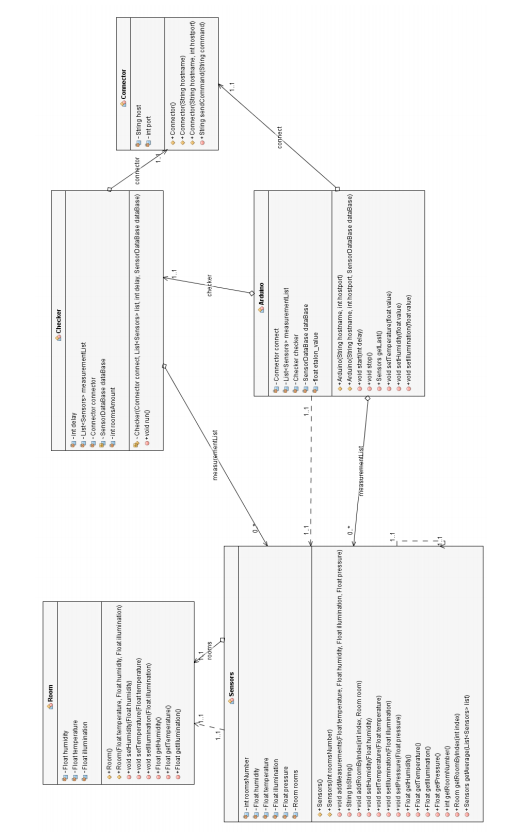
1. Keyestudio <https://wiki.keyestudio.com/Ks0077(78,_79)_keyestudio_Super_Learning_Kit_for_Arduino#Project_23:_PIR_Motion_Sensor>
2. GitLab Valeria Razdobudova

<https://gitlab.com/bachelors-2020/pi-54/razdobudova-valeria/diplomaproject>

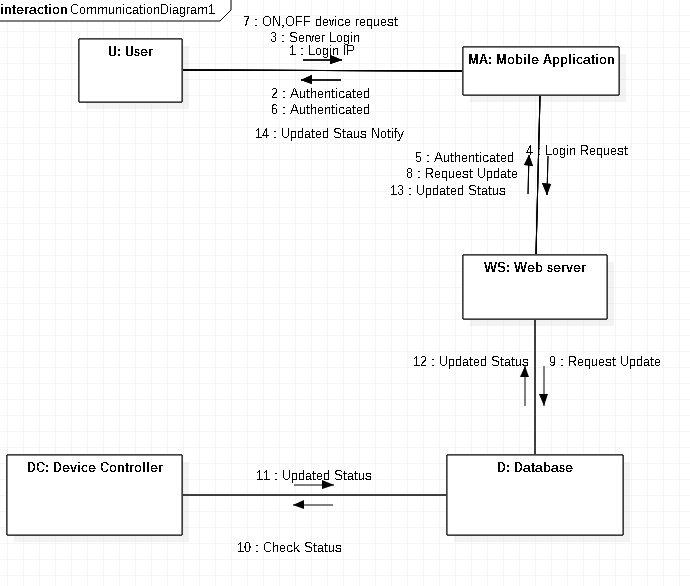
# ДОДАТКИ

# ДОДАТОК А

UML Діаграма класів основних елементів системи



Діаграма комунікацій



ДОДАТОК Б

Метод, що реагує на запалення у приміщенні та надсилає повідомлення

//---------------Перевірка на займання у приміщенні------------//

void CheckFire() {

Temp\_alert\_val = CheckTemp();

Serial.print(Temp\_alert\_val);

if (Temp\_alert\_val > 45) {

Fire\_Set = 1;

while (sms\_count < 3)//Кількість повідомлень для відправки

{

SmsMaker();

}

}

}

#include "SmsMaker.h"

extern volatile uint8\_t delayEspired;

SmsMaker::SmsMaker(Modem \*m) {

modem = m;

}

/\*

Відправка повідомлення

\*/

bool SmsMaker::sendWarningSms(uint8\_t clientOrder, int messageType) {

if (digitalRead(DUCT\_PIN) == LOW) {

Serial.print(F(" спроба send\_WarningSms відправки повідомлення користувачу"));

Serial.print(String(clientOrder) + " msgTYPE=" + String(messageType));

}

char mess[MAX\_MESS\_LEN];

char phone[49];

if (modem->fillUCS2PhoneNumber(clientOrder, phone) == false) {

#if DEBUG

Serial.println(F(" помилка! такий номер не можливий"));

#endif

return false;

}

phone[48] = '\0';

strcpy\_P(mess, (char \*)pgm\_read\_word(&(messages[messageType])));

#if DEBUG

/\*

uint8\_t i = 0;

Serial.print(F(" text msg: "));

while (mess[i] != '\0') {

Serial.write(mess[i]);

i++;

}

Serial.println(F(" :"));

\*/

#endif

return sendSms("UCS2\0", phone, mess);

}