

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ
«ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»**

Спеціальність:

123 – «Комп'ютерна інженерія»

Освітня програма:

«Обслуговування комп'ютерних систем і мереж»

Група: 4КС-56

Дипломний проект

**студента денної форми навчання
КС 56.18.000.00 ДП**

***ПРОХОРОВА
МИХАЙЛА
МИХАЙЛОВИЧА***

**м. Одеса
2023 р.**

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «Одеський технічний фаховий коледж ОНТУ»

Спеціальність 123 – «Комп'ютерна інженерія»

Освітня програма «Обслуговування комп'ютерних систем і мереж»

Група 4КС-56

ПОЯСНЮВАЛЬНА ЗАПИСКА

До дипломного проекту (роботи) на тему: _____

***Проектування системи клімат-контролю
на базі платформи Arduino***

Проектний матеріал складається з пояснювальної записки на 76 сторінках та графічного матеріалу на 15 аркушах.

Дипломник  (Прохоров М.М.)

Керівник проекту  (Скорняков В.С.)

Консультанти:

з економічної частини  (Копайгородська Т.Г.)

з охорони праці  (Чорновол Н.І.)

за дотриманням вимог ЄСКД  (Петрашова В.І.)

старший консультант  (Кривченко Ю.В.)

До захисту допущений

Голова циклової комісії  (Кривченко Ю.В.)

Завідувач відділенням  (Скорнякова О.В.)

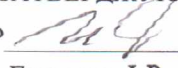
Захист «20» сервіс 2023 р. Протокол ДКК № 2

Оцінка ДКК 4/добре

Секретар ДКК 

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «Одеський технічний фаховий коледж ОНТУ»

Відділення комп'ютерних систем Комісія КТ і ПІ
Спеціальність 123 "Комп'ютерна інженерія"

ЗАТВЕРДЖУЮ:
Заст. дир. з НВР 
Беркань І.В.
" " 2023 р.

ЗАВДАННЯ

на дипломний проект (роботу)

Здобувачу освіти Прохорову Михайлу Михайловичу
(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Проектування системи клімат-контролю
на базі платформи Arduino

затверджена наказом по коледжу від "17" жовтня 2022 року № 235-А2-ОД

2. Термін здачі здобувачем освіти закінченого проекту (роботи) 10.06.2023

3. Вихідні дані до проекту (роботи) Компоненти системи контролю мікроклімату. Програмовані плати, мікроконтролери датчики температури, вологості, освітлення, датчика якості повітря, вуглекислого газу. Блок зв'язку. Модуль годинника. Елементи індикації. РКІ.

4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити)
1. Технологічний розділ. 2. Економічний розділ. 3. Охорона праці. Список використаних джерел. Висновки. Додатки.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
1. Тема роботи. 2. Вступ. 3. Завдання для проектування. 4. Об'єкт для проектування. 5. Функції системи. 6. Структурна схема системи клімат-контролю. 7. Вибір елементної бази. 8. Принципова схема підключення датчиків. 9. Принципова схема підключення виконавчих пристроїв. 10. Блок-схема роботи системи клімат-контролю. 11. Алгоритм роботи системи увімкнення освітлення. 12. Алгоритм роботи системи оповіщення при перевищенні температури на кухні. 13. Алгоритм регулювання температури. 14. Перспективи розвитку ідеї. 15. Дякую за увагу.

6. Консультанти по проекту (роботі), із зазначенням розділів проекту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
Розділ 1-3	Скорняков В.С.		
Економічний розділ	Копайгородська Т.Г.		
Охорона праці	Чорновол Н.І.		
Нормоконтроль	Петрашова В.І.		

7. Дата видачі завдання _____

Керівник

Скорняков В.С.

(підпис)

Завдання прийняв до виконання

(підпис)

КАЛЕНДАРНИЙ ПЛАН

Пор. №	Назва етапів дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1.	Огляд літератури. Огляд існуючих рішень	20.02.2023	
2.	Формування кінцевого завдання на розробку. Вступ.	01.03.2023	
3.	Аналітичний огляд існуючих рішень	20.03.2023	
4.	Конструкторський розділ. Вибір елементної бази	10.04.2023	
5.	Розробка алгоритму та управляючої програми	17.04.2023	
6.	Економічний розділ. Проведення розрахунків щодо економічної доцільності розробки	01.05.2023	
7.	Виконання розділу «Охорона праці»	15.05.2023	
8.	Виконання графічної частини дипломного проекту	22.05.2023	
9.	Підготовка до попереднього захисту, підготовка до захисту	01.06.2023	
10.	Підготовка доповіді та презентації для захисту	10.06.2023	
11.	Отримання рецензії, відповіді на зауваження рецензента	до 19.06.2023	
12.	Захист роботи	до 30.06.2023	

Дипломник _____

(підпис)

Керівник проекту _____

(підпис)

ЗМІСТ

ВСТУП	7
1 ТЕХНОЛОГІЧНИЙ РОЗДІЛ	9
1.1 Аналіз існуючих рішень	9
1.2 Розробка схем та вибір елементної бази	13
1.3 Розробка алгоритмів та управляючої програми	33
2. ЕКОНОМІЧНИЙ РОЗДІЛ	43
3. ОХОРОНА ПРАЦІ	49
3.1 Аналіз небезпечних і шкідливих факторів, що впливають на програміста при розробці даного програмного комплексу	49
3.2 Гігієнічні вимоги до виробничого середовища	50
3.2.1 Вимоги до приміщення	50
3.2.2 Освітлення	51
3.2.3 Шум	51
3.2.4 Вимоги до організації робочого місця працівника	51
3.3. Пожежна безпека	53
ВИСНОВКИ	55
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	56
ДОДАТКИ	58

ВСТУП

Системи типу «Розумний будинок» останнім часом стали звичайною справою навіть для жителів невеликих міст, і тим більше для столиць і мегаполісів. В даний час ні для кого не секрет, що за подібними технологіями майбутнє, і вони будуть розвиватися. Актуальним залишається питання про те, якою стане стратегія розвитку систем подібного роду, які технології будуть використовуватися для їх розробки і експлуатації.

«Розумний будинок» - це одна з пріоритетних областей розвитку Інтернету, в якій працює велика кількість людей, компаній і структур. Всі вони в даний час вельми активно включають до складу своїх рішень різноманітні функції управління споживчими пристроями, а також реалізують обмін даними між цими пристроями. Використання сучасних каналів зв'язку, датчиків, включених до складу побутових пристроїв і обладнання, а також наявність програмного забезпечення, що забезпечує узгодження всіх компонентів системи і управління ними, дозволяє створити умови, при яких будинок стає не тільки більш безпечним, але й істотно економити ресурси. Термостати, які з'являються зараз, дозволяють не тільки дистанційно керувати кліматом і налаштовувати його на необхідні параметри, а й по-різному підходити до вирішення проблеми. Наприклад, в просторій квартирі або великому будинку немає причин для нагріву кухні в нічний час.

Проектування розумного будинку може включати в себе управління світлом, кліматом, системою безпеки, керованими шторами і навіть деякою побутовою технікою (наприклад, пилесмоком, телевізором, аудіоустановкою і т.д.). Управління відбувається за рахунок отримання і обробки даних, що подаються з датчиків руху, освітлення і температури, задимленості, вогню та полум'я, потоку і рівня води, охорони (на двері і вікна). Власне, основний інтерес становлять вхідні

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		7

сигнали датчиків, тому, систему регулювання домашнього освітлення можна реалізувати так само, як і охоронну сигналізацію, тобто на базі датчика руху.

Зацікавленість до аналізу цієї теми не випадкова, через те що з одної точки зору вона пов'язана з основними проблемами фізики, електроніки та штучним інтелектом. З іншого боку, інтерес до створення розумних будинків пов'язана з пошуком нових незвичних методів аналізу електронної структури і майбутніх розробок, які в найближчому майбутньому з'являться в нашому повсякденному житті.

У даній роботі проектується система клімат-контролю на базі платформи Arduino. Для контролю житла використовуються датчики і супутнє обладнання. Всі прилади підключаються до єдиної системи управління, в нашому випадку контролера Arduino, за допомогою якого проводиться їх моніторинг і управління. У роботі розглядаються питання функціональної специфікації, апаратної та програмної реалізації. У специфікації розписані дані про об'єкт управління і детально описані функції, які повинна виконувати система. В апаратній частині наведені конкретні апаратні засоби, їх опис, вхідні і вихідні параметри, спосіб підключення і принцип реалізації необхідних функцій. У програмній частині наведені програми, що реалізують запропоновані системі функції.

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		8

1 ТЕХНОЛОГІЧНИЙ РОЗДІЛ

1.1 Аналіз існуючих рішень

«Розумний будинок» – це поняття, яке виникло у контексті сучасного заміського будинку середнього класу. Деякі уривчасті данні з публікацій встановлюють їх виникнення у часи «технологічної революції» розвитку побутових приладів в ХХ столітті. Річард Харпер у своїй книзі «Всередині розумного дому» розглядає два елементи, що стали поворотними моментами для впровадження та просування побутових технологій у сім'ях середнього класу: проведення електрики в будинки на початку ХХ століття і впровадження інформаційних технологій у другій половині ХХ століття [12].

В даний час стали популярними також різноманітні системи безпеки, управління освітленням та ін. Це все є лише прикладами окремих підсистем «Розумного будинку», які, тим не менш, не будучи взаємопов'язаними, не можуть називатися єдиною системою. Функціональність, стиль, комфорт, безпека, і це далеко не повний список того, чим нас може порадувати «Розумний будинок». Сучасні інженерні технології можуть миттєво підлаштовуватися під споживача, надаючи масу можливостей. В цьому і полягає суть системи «Розумний будинок».

«Розумним» називають сучасний будинок, всі комунікації якого зв'язані за допомогою високотехнологічних пристроїв таким чином, щоб людині в ньому було зручно. Повсякденні прилади в «Розумному будинку» з'єднуються в одну загальну домашню мережу з можливістю виходу в інтернет. Сигнали від датчиків, встановлених в кожній кімнаті, надходять до центрального комп'ютера, який обробляє отримані сигнали і в залежності від поставленого завдання генерує керуючі команди для пристроїв, які слід використовувати [10].

Типові основні функції «Розумного будинку» включають в себе управління наступними системами [13, 16, 10, 11, 21]: електропостачання та освітлення;

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		9

інтернет, мобільний зв'язок, система оповіщення; телебачення, аудіо- та відеосистеми; дистанційне керування; водопостачання і каналізація; клімат-контроль, опалення та вентиляція; забезпечення безпеки і відеоспостереження; пожежна сигналізація; моніторинг поломок (наприклад, витоків газу або протікання води).

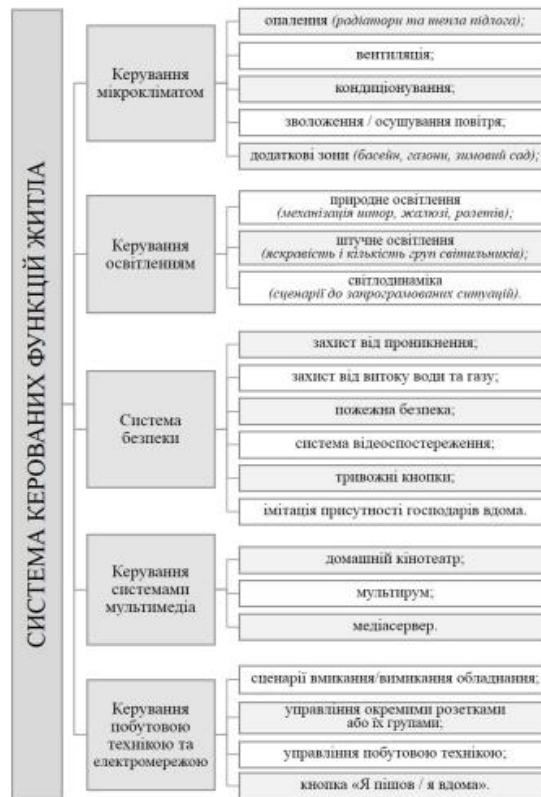


Рис.1.1 – Складові системи «розумного будинку»

Управління температурою. Створення і підтримка оптимального мікроклімату житла – найважливіша умова високої працездатності, продуктивного відпочинку і здоров'я мешканців будинку або квартири. Забезпеченням цих умов займаються наступні інженерні системи: опалення (тепла підлога); вентиляція; кондиціонування; зволоження/осушування повітря.

Система клімат-контролю працює відповідно до закладених у неї алгоритмів, що дозволяють підтримувати параметри середовища і різних кліматичних зон в приміщеннях при мінімальних затратах енергоресурсів. Для автоматичного

регулювання систем опалення використовуються спеціальні прилади – терморегулятори. З їх допомогою і за допомогою програмного забезпечення системи можна налаштувати температурний режим, а також встановити співвідношення між роботою теплої підлоги і радіаторів. Для підтримки оптимальної вологості повітря (40 - 60%) можуть використовуватися зволожувачі й осушувачі. Перші актуальні в холодну пору року, коли радіатори «висушують» повітря, а другі – в приміщеннях з високою вологістю – у ванній, басейні [12]. За комфортну температуру в вашому домі відповідають такі нагрівальні пристрої, як радіатори, бойлери, нагрівальні кабелі статі і т. Д. Кожне з цих пристроїв має свій власний принцип роботи і виконує свою власну функцію.

Система управління кліматом в приміщенні дає можливість встановлювати оптимальний рівень температури, вологості, величину притоку свіжого повітря, управляти роботою системи фільтрації повітря, створювати індивідуальну кліматичну систему для кожного члена сім'ї (наприклад, в дитячій кімнаті постійний приток свіжого повітря за відсутності протягів). У той же час система клімат-контролю, забезпечує економію фінансових коштів і вирішує проблему енергозбереження. Як показує практичний експеримент, зниження кімнатної температури на 1°C може знизити споживання енергії на 6%. Якщо під час відсутності людей в приміщенні температура падає на 3 °С, то ви можете заощадити 18% енергії.

Крім створення комфорту, застосування таких систем значно подовжує термін служби електроприладів, а також чимало сприяє енергозбереженню.

Клімат-контроль в системі вентиляції. Впровадження кліматичного контролю безпосередньо впливає на забезпечення оптимальних видів споживання енергії для кліматичних систем. Опалення та кондиціонування повітря в «Розумному будинку» з правильними кліматичними режимами ніколи не працюють «назустріч один одному» і не споживають енергію даром. Справа в тому, що система опалення

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

знає про роботу кондиціонера, і кондиціонер не може охолоджувати повітря, поки працює система опалення та навпаки.

Для комфортного мікроклімату потрібно орієнтуватися на показник відносної вологості, яка вимірюється в процентному вмісті. На відносну вологість сильно впливає температура – при її зниженні вологість збільшується, а при підвищенні, навпаки, знижується. Оптимальний показник відносної вологості в квартирі коливається в межах 45-60%. Саме цей комплекс пристроїв на підставі заданих параметрів і показань датчиків створює сприятливий мікроклімат у приміщеннях. Такі системи найчастіше входять до складу «розумного будинку». По суті, це єдина система вентиляції, в яку входить кондиціонер, опалювальна система, зволожувач, очищувач, різні фільтри і інші прилади. Це одне з найпростіших пояснень, як виглядає система клімат контроль у квартирі.

Управління освітленням. Система освітлення - це перше, що спадає на думку при згадці «розумного будинку». Дійсно, забезпечення штучного світла, це сама ресурсномістка функція. Управління освітленням - одна з найпопулярніших областей автоматизації квартири або будинку. Індивідуально спроектована система освітлення створить затишний будинок, заощадить енергію, приємно здивує ваших гостей. Кількість, тип і потужність світильників завжди вибираються на основі таких факторів, як призначення кімнати, її колірна гамма, інтер'єр, вуличне освітлення і багато інших. За допомогою сенсорного пристрою ви також можете включити систему освітлення кімнати, в якій ви перебуваєте, або включити весь будинок, а також визначити окремі групи освітлення або інтенсивність світіння. Логіка для роботи освітлювальних приладів проста: датчики світла і датчики руху встановлені в ключових місцях кімнати. Інформація з датчиків обробляється відповідно до встановленого алгоритму і підключає відповідне обладнання, забезпечуючи необхідний рівень яскравості в приміщенні.

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

Таким чином, в нашій роботі будуть впроваджені наступні основні системи: система контролю температури з можливістю забезпечувати вентиляцію повітря і додатково - керувати освітленням.

1.2 Розробка схем та вибір елементної бази

Отже, розглянемо компоненти системи, яка проектується у даній роботі. Для початку, визначимося з об'єктом, для якого розробляється дана система. Це може бути багатокімнатна квартира, котедж, заміський будинок, промисловий об'єкт, тощо. Нехай це буде одноповерховий будинок площею 120 м², план якого наведено на рис.1.2.

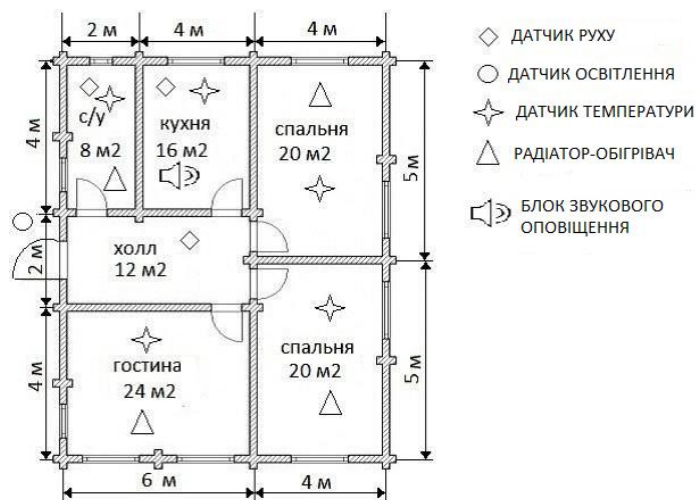


Рис.1.2 - План приміщення

Система управління «Розумний будинок» буде включати в себе:

1. автоматичне регулювання температури повітря за допомогою:

- аналогового датчика температури - показання якого будуть враховані при установці інтенсивності роботи опалювального елемента в вітальні (24 кв.м) і в двох спальнях (20 кв.м кожна)
- цифрового датчика температури - показання якого будуть також враховані при установці інтенсивності роботи опалювального елемента в с/у (8 кв.м.)

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

- електричного радіатора опалення, який буде підтримувати встановлену температуру.

2. ручне та автоматичне керування освітленням:

- при спрацьовуванні датчика руху - включення освітлення в холі (12 м²), В санвузлі (с/у) (8 м²) і на кухні (16 м²).

- при спрацьовуванні датчика освітленості - включення вуличного освітлення на веранді у парадних дверей.

3. звукову сигналізацію про перевищення порога температури:

- при спрацьовуванні релейного датчика температури буде відтворюватися звуковий сигнал, який вказує на перевищення заданого рівня температури на кухні (16 кв.м.)

Проаналізуємо типи та кількість необхідних датчиків, складемо план будинку – на рис.1.2 наведено план розташування датчиків по приміщенням, з урахуванням площі приміщень, розташування нагрівальних елементів, вікон.

Структурна схема системи представлена на рис.1.3.

«Розумний будинок» - комплексна система управління всім інженерним обладнанням будівлі. На структурній схемі видно, що для реалізації системи управління необхідні: центральний блок, який буде реалізований на платі, датчики, виконавчі механізми, що представляють собою джерела світла, звукова сигналізація і опалювальні прилади; пристрій взаємодії з користувачем, наприклад, персональний комп'ютер.

Вибір обігрівальних приладів ще більш великий, це можуть бути тепло вентилятори, теплові гармати, теплові завіси, інфрачервоні обігрівачі, електричні конвектори, масляні радіатори, трубчасті електронагрівачі і т.д. Джерелами світла можуть бути обрані світлодіодні системи освітлення, люмінесцентні лампи, компактні люмінесцентні лампи, газорозрядні лампи, лампи розжарювання, галогенні лампи розжарювання, лампи спеціального призначення і т.д.

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

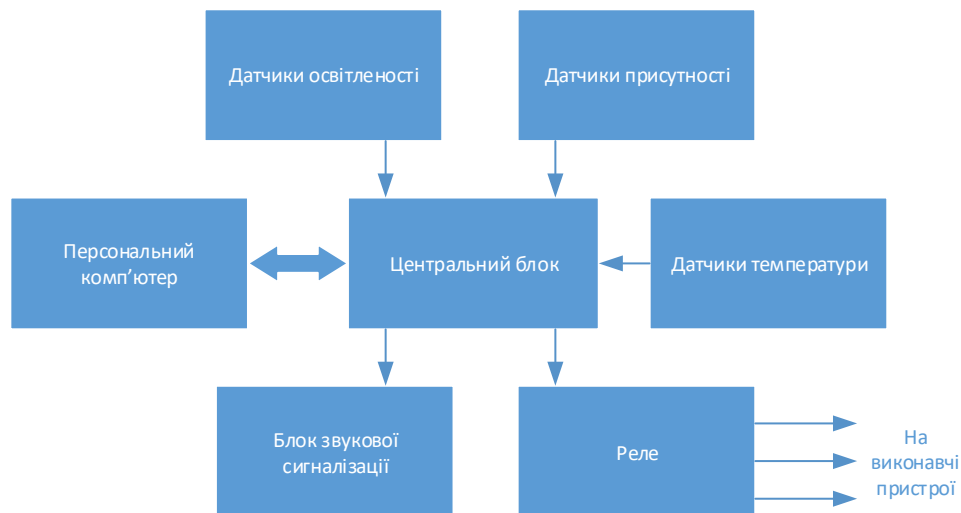


Рис.1.3 - Структурна схема системи клімат-контролю

Розглянемо функції, що будуть реалізовані системою клімат-контролю:

Управління температурою. Управління температурою необхідно для створення комфортної атмосфери в приміщенні. Також буде передбачена можливість прогрівати приміщення завчасно або відключати систему опалення при відсутності людей вдома, що дозволить заощадити значні кошти. Таким чином, система дозволить віддалено підтримувати сприятливий мікроклімат.

За рахунок калібрування датчика температури користувач може завдати бажану температуру, нижче якої система не дозволить опуститися. На жаль, така система може обмежити тільки нижню межу температури, тому що обігрівач не має можливості охолоджувати повітря.

Управління освітленням. Користувач може включити освітлення в кімнаті через вимикач, тоді світло буде горіти незалежно від присутності або відсутності людини в цій кімнаті. Якщо світло не включено через вимикач, тоді в залежності від присутності або відсутності людини в кімнаті світло буде включатися або вимикатися. Такий сценарій буде закладений на мікроконтролер через сигнали, що надходять з датчиків присутності. При відсутності необхідності включення світла, за бажанням користувача, він може відключити автоосвітлення через ПК.

Моніторинг має на увазі вивід на монітор ПК інформації про температуру, зафіксовану термодатчиком, інформації про спрацьовування датчика руху і освітленості, індикації включення і виключення освітлювальних, акустичного та обігрівальних приладів. Налагодження передбачає включення і відключення функції автоматичної системи освітлення, регулювання температури, а також примусове віддалене включення виконавчих пристроїв, в незалежності від даних отриманих з датчиків.

Як будуть реалізовані ці функції – розглянемо більш детально.

На рисунках 1.4 і 1.5 представлені структурно-функціональна схеми, що наочно представляють організацію управління такими функціями:

Освітлення. На сьогоднішній день існує велика різноманітність автоматичних пристроїв, призначених для управління освітленням як на вулиці, так і в приміщення. Згідно з технічним завданням будуть застосовані наступні компоненти:

- сутінкові реле (застосовуються на вулиці). Такі прилади спрацьовують тоді, коли рівень освітленості на вулиці знизиться до певного показника;
 - світильники, оснащені датчиками руху, будуть застосовані в приміщенні.
- Такого роду прилади ефективно працюють поблизу споруд - гаражні ворота, центральний і чорний вхід в будинок, веранда і т.д. Активація відбувається тоді, коли в радіусі дії з'являється рух.

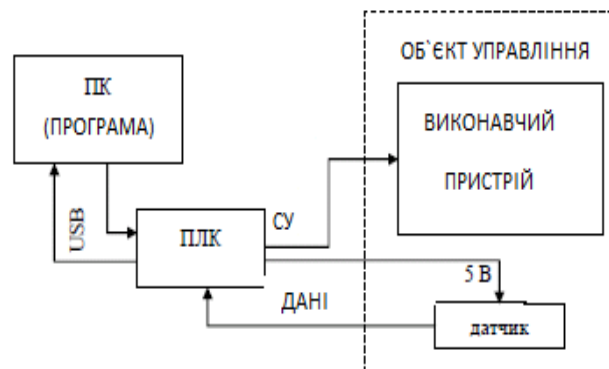


Рис.1.4 - Структурно-функціональна схема системи управління

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

Обидва варіанти, які працюють в режимі автоматичного включення підсвічування, можна легко встановити своїми руками. Для підключення тут знадобиться лише схема, яка додається виробником.

Температура. Реалізують:

- термоопір (термістори). Застосовуються досить часто, так як спосіб вимірювання температури, що полягає в зміні опору від температури, досить простий і в залежності від способу зняття показань можна отримати високоточні дані.
- цифровий датчик. Має високу точність, широкий інтервал вимірюваних температур і простий в експлуатації. Його недоліком є низька швидкодія, що не грає ролі при автоматизації житлового приміщення.
- напівпровідниковий датчик. Так як в якості термодатчиків використовуються діоди і біполярні транзистори, то вони мають низьку вартість, високу точність, лінійну характеристику і прості у використанні. Як правило, діапазон вимірюваних температур дорівнює -50 до 150 $^{\circ}\text{C}$.

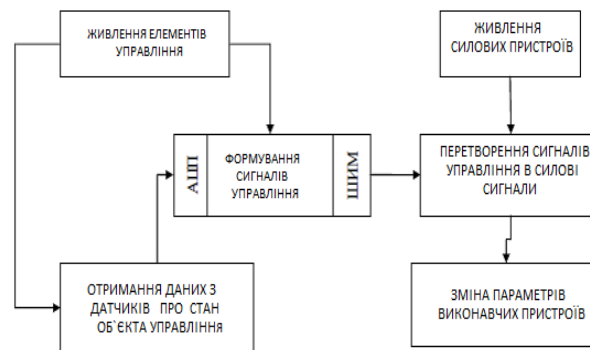


Рис. 1.5 - Функціональна схема системи управління

Для реалізації будь-якої системи автоматичного управління необхідні дві складові: апаратна реалізація через елементну базу та відповідне програмне забезпечення. Додатковим елементом може стати візуалізація роботи системи, призначена для користувачів; вона забезпечить комфортне спостереження і

керування, що виключає внесення змін до базових налаштувань системи, які можуть привести до непередбачених несприятливих наслідків.

Центральний блок реалізовано на платі Arduino.

Arduino - апаратна обчислювальна платформа, основними компонентами якої є плата вводу/виводу та середовище розробки на мові Processing/Wiring. Це інструмент для проектування електронних пристроїв (електронний конструктор) більше взаємодіючих з навколишнім фізичним середовищем, ніж стандартні персональні комп'ютери, які фактично не виходять за рамки віртуальності.

Плата Arduino складається з мікроконтролера Atmel AVR, а також елементів для програмування та інтеграції з іншими пристроями. На багатьох платах наявний лінійний стабілізатор напруги +5В або +3,3В. Тактування здійснюється на частоті 16 або 8 МГц кварцовим резонатором. У мікроконтролер записаний завантажувач (bootloader), тому зовнішній програматор не потрібен.

Arduino застосовується для створення електронних пристроїв з можливістю прийому сигналів від різних цифрових і аналогових датчиків, які можуть бути підключені до неї, і управління різними виконавчими пристроями. Проекти пристроїв, засновані на Arduino, можуть працювати самостійно або взаємодіяти з програмним забезпеченням на комп'ютері (наприклад Flash, Processing, MaxMSP). Плати можуть бути зібрані користувачем самостійно або куплені в зборі. Середовище розробки програм з відкритим вихідним кодом доступна для безкоштовного скачування. Оригінальні плати Arduino виробляються фірмою Smart Projects. На даний момент доступно 20 версій плат, які різняться характеристиками мікроконтролера та кількістю аналогових і цифрових виводів.

Для реалізації всіх функцій нашої системи контролер повинен мати шість цифрових входів і три аналогових і дев'ять цифрових виходів для підключення твердотільного реле, яке в свою чергу буде комутувати нагрівальні, освітлювальні і звуковий елементи. Відповідно, для нашого випадку підходящим є конструкція

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		18

плати Arduino Micro на базі мікроконтролера ATmega32u4. Його параметри (табл.1.1), повністю задовольняють вимогам системи.

Таблиця 1.1 – Параметри плати Arduino Micro

Параметр	Характеристики
Мікроконтролер	ATmega32u4
Робоча напруга	5 В
Вхідна\вихідна напруга	7-12 В\6-20 В
Цифрові\аналогові входи\виходи	20\12
Канали ШИМ	7
Постійний струм через вхід\вихід	40 мА
Постійний струм для виводу 3,3В	50 мВ
Флеш-пам'ять	32 Кб
ОЗП	2,6 Кб
EEPROM	1 Кб
Тактова частота	16 МГц



Рис.1.6 – Зовнішній вигляд плати Arduino Micro

Підключення Ардуіно до ПК здійснюється через шину USB. Завдяки своїй універсальності і здатності ефективно передавати різномірний трафік, вона буде застосована для підключення контролера до ПК і покликана замінити традиційний порти PC - COM. Універсальний зовнішній послідовний інтерфейс - COM-порт

(Communications Port - комунікаційний порт) передає біти послідовно один за одним і забезпечує асинхронний обмін за стандартом RS-232C. Так як передбачається передача даних між контролером і ПК на швидкостях 9600 біт/с і вище, то COM-порт реалізований на мікросхемі UART 16550A.

ATmega32u4 має вбудовану підтримку USB-з'єднання, завдяки чому дуже просто реалізується взаємодія з ПК через послідовний COM-порт.

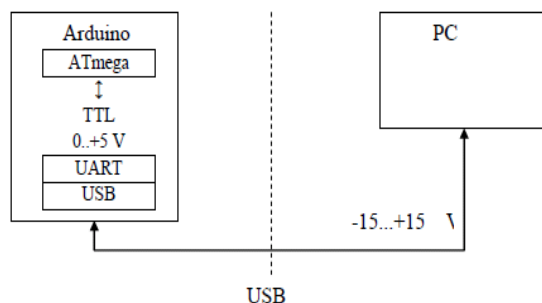


Рис.1.7 - Обмін даними між Arduino та ПК через COM-порт

Розглянемо характеристики мікроконтролера ATmega32u4.

ATmega32 u 4 - економічний 8-бітний КМОП мікроконтролер, виконаний на основі RISC ЦПУ AVR. За рахунок виконання більшості інструкцій за один цикл синхронізації ATmega32U4 досягає продуктивності 1 MIPS/МГц, що дозволить розробнику оптимізувати співвідношення споживаної потужності і продуктивності. Вбудована програмована флеш-пам'ять підтримує можливість внутрішньо-системного перепрограмування за допомогою послідовного інтерфейсу SPI.

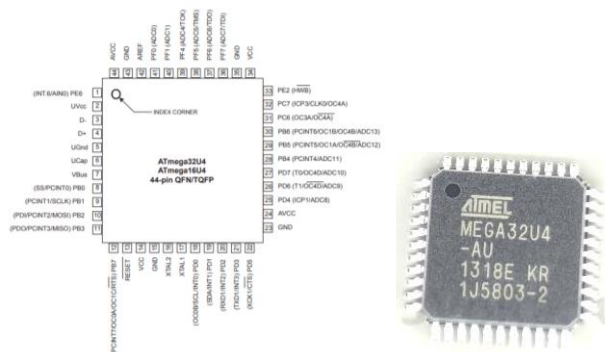


Рис.1.8 – Розташування виводів та зовнішній вигляд МК ATmega32u4

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		20

Поєднання в одному кристалі 8-бітного RISC ЦПУ з внутрішньо-системною самопрограмованою флеш-пам'яттю робить мікроконтролер ATmega32u4 ефективним і недорогим інструментом для вирішення завдань вбудованого управління. Мікроконтролер ATmega32u4 підтримується повним набором апаратних і програмних засобів для проектування, в т.ч. C-компілятори, макроасемблера, програмні відладчики/симулятори, внутрішньо-схемні емулятори та оціночні набори.

Маємо такі характеристики мікроконтролера ATmega32u4:

Ядро	8bit megaAVR
Максимальна бистродія	16 МГц (16 MIPS)
Пам`ять програм (flash)	32 КБ
Пам`ять даних (RAM)	2,5 КБ
Пам`ять EEPROM	1 КБ
Кількість ліній вводу/вивода	26
Кількість входів АЦП	12
Розрядність АЦП	10
Кількість АЦП	1
Бистродія АЦП	15 тис.відліків/с
Кількість UART	1
USB	передбачено
Мінімальна напруга живлення	2,7 В
Максимальна напруга живлення	5,5 В
Корпус	TQFP-44-0.8
Робоча температура	-40...85 °С

Детальний опис мікроконтролера приведено у [6].

Під час запису програми і передачі даних в графічний інтерфейс Ардуіно буде живитися через USB підключення. Під час роботи живлення буде надходити через порт Vin від зовнішнього джерела живлення постійного струму (рис.1.9). Згідно

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		21

рекомендованій вхідній напрузі 7-12В (табл.1.1), буде використаний універсальний стабілізований імпульсний блок живлення з вихідною напругою 12 В фірми Jazzway (табл.1.2). Схема підключення представлена на рисунку 1.10.

Використані виводи живлення:

- а) 5V. Регульована напруга живлення, яке буде використовуватися для живлення компонентів.
- б) 3V. 3,3 В - генерується вбудованим стабілізатором. Буде використаний для компонентів, діапазон напруги живлення яких включає 3,3 В.
- с) GND. Виводи заземлення - буде використовуватися для всіх компонентів.



Рис.1.9 – Адаптер на 12В

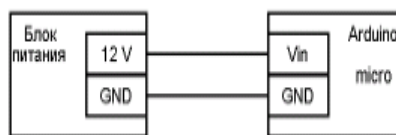


Рис.1.10 – Схема підключення живлення до Ардуіно

Використані входи і виходи:

- а) ШІМ: буде потрібно 4 виходи 8-бітного ШІМ для регулювання інтенсивності нагрівального елемента.
- б) Аналогові входи: буде потрібно чотири аналогових виходи, для підключення аналогових датчиків температури. Всього Micro має 12 аналогових входів, з яких потрібно тільки один. Всі вони також можуть використовуватися в якості цифрових входів/виходів. Кожен аналоговий вхід забезпечує роздільну здатність 10 біт (тобто 1024 різних значення). За замовчуванням, вимірювання на всіх

аналогових входах виробляються від потенціалу землі до 5 В, але верхню межу цього діапазону можна змінити.

Таблиця 1.2. Характеристики адаптера

Параметр	Характеристики
Напруга живлення	110-240 В, 50-60 Гц
Вихідна напруга живлення	12 В
Вживана потужність	12 Вт
Робоча температура експлуатації	0...30 °С
Ступінь захисту світлодіодного модуля	IP20

Наступний крок – вибір порогового датчика температури. Датчик температури буде розташований в центрі кожної з кімнат, перерахованих в технічному завданні.

Мінімальні вимоги у всіх термодатчиків однакові: нижня межа діапазону вимірюваної температури повинен бути не вище 20 С°, верхня межа - не нижче 70 С°. Так як буде регулюватися мікроклімат житлового приміщення, то швидкодія датчика не грає великої ролі, а вартість повинна бути низькою. На кухні єдиною вимогою є реєстрація перевищення допустимого значення температури, тобто необхідний датчик, який буде передавати 0 або 1, залежно від того чи перевищила температура на кухні встановленого значення чи ні. Повинна бути присутнім можливість регулювання температура, перевищення якої неприпустимо. Також датчик повинен володіти високою швидкістю.

Для кухні підійде датчик на основі термістора (опір якого залежить від температури), а на виході датчика буде сигнал високого або низького рівня (рис.1.11). Цей сигнал буде результатом порівняння різниці потенціалів на термісторі і різниці потенціалів на потенціометрі (опір якого потрібно вручну відрегулювати). Щоб гарантувати високий рівень на логічному виході між виходом

і живленням 5В буде підключений підтягаючий резистор (рис.1.12). Обран тип МР DKM-046.

Таблиця 1.3. Параметри термодатчика МР DKM-046

Параметр	Характеристики
Робоча напруга живлення	3,3-5 В
Вихідний сигнал	0 або 1
Діапазон робочих температур	20...80 °С



Рис.1.11 - Релейний термодатчик для кухні МР DKM-046

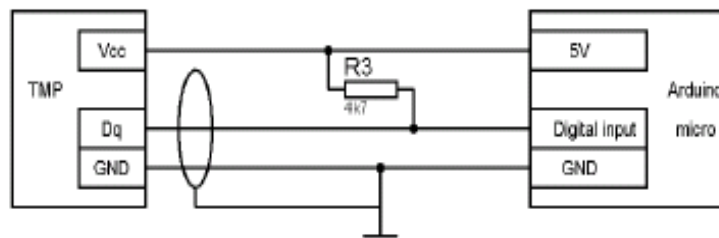


Рис.1.12 - Схема підключення релейного термодатчика

Схема датчика представлена в Додатку 1 (рис.1).

Наступний крок – вибір аналогового датчика температури. Аналоговий датчик буде встановлений в холі, так як у ванній через підвищеної вологості показання вихідної напруги можуть бути перекручені. За вимогами діапазону температур, вихідної напруги і споживаної напруги підходить датчик LM35.

Це напівпровідниковий датчик, його вихідна напруга пропорційна С°, що дуже зручно для перетворення напруги в температуру. Так як споживаний струм датчиком становить 60 мкА, його розігрів менш С°, що також є великою перевагою.

Точність 0.5 °С не погіршиться, навіть якщо опорна напруга на АЦП мікроконтролера буде максимальною - 5В. У цьому випадку 5 вольт відповідала б

температурі 500 С°, якій в свою чергу відповідав би показник 1023 на виході АЦП. Отже, одному значення на виході АЦП відповідає $500/1023 = 0,4888$ С°/ розряд.



Рис.1.13 - Аналоговий датчик температури

Таблиця 1.4. Параметри аналогового датчика температури

Параметр	Характеристики
Залежність напруги від температури	Лінійна 10мВ\°С
Робоча напруга живлення	4-30 В
Діапазон робочих температур	0...100 °С
Гарантована точність	0,5°С (при 25°С)
Вживана потужність	60 мкА

При передачі електричних сигналів відбувається взаємодія з зовнішніми полями, і на чим більшу відстань передається сигнал, тим більше сумарний вплив, внаслідок чого на вході приймача крім вимірювального сигналу присутня і перешкода. За рекомендацією розробників, щоб уникнути їх впливу необхідно використовувати низькочастотний фільтр, який є пристроєм, що пропускає з постійним коефіцієнтом передачі низькі частоти, і викликає ослаблення високих частот з запізненням по фазі. Використовуємо найпростіший RC-фільтр низьких частот, з граничною частотою 50 Гц. Опір приймемо $R = 2\text{кОм}$, тоді з формули граничної частоти висловимо ємність конденсатора C :

$$f_g = \frac{1}{2\pi RC} = 50 \Rightarrow C = \frac{1}{2\pi R f_g} = \frac{1}{2 * 3.14 * 2000 * 50} = 1.59 \text{ мкФ},$$

де f_g -гранична частота, Гц; R - опір, Ом; π -констата.

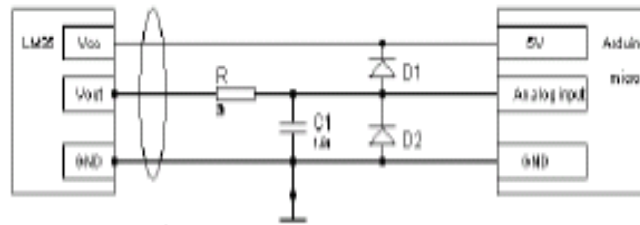


Рис.1.14 - Схема підключення аналогового датчика температури

Схема датчика представлена в Додатку 1 (рис.2).

Наступний крок – вибір цифрового датчика температури. Датчик фірми Даллас DS18B20 (рис. 1.15) підходить як по напрузі живлення, так і за умовами експлуатації (табл. 1.6). Завдяки передачі сигналу через протокол інтерфейсу 1-Wire у вигляді цифрового коду (табл. 1.5), він є перешкодостійким, його можна підключити на великі відстані і підвищена вологість не чинитиме впливу на його показники.

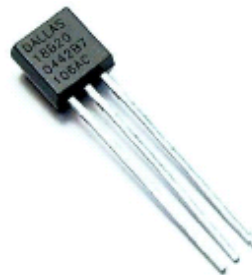


Рис.1.15 - Датчик температури DS18B20

Розрядність передачі температури в вигляді цифрового коду варується 9- 12 біт. При максимальній останні чотири біта відповідають значенням після коми, таким чином мінімальна похибка при перетворенні становитиме $2^{-4} = 0,0625 \text{ C}^\circ$. Для лінії зв'язку потрібен підтягаючий резистор близько 4,7 кОм (рис. 2.15), так як всі елементи фізично підключені до загальної шини і використовують вихід з трьома станами або вихід типу відкритий стік.

Таблиця 1.5. Передача сигналу через протокол 1-Wire в вигляді цифрового кода

Температура	Двоичний код	HEX код
+125	0000 0111 1101 0000	07D0h
+85	0000 0101 0101 0000	0550h
+25,0625	0000 0001 1001 0001	0191h

+10,125	0000 000 1010 0010	00A2h
+0,5	0000 0000 0000 1000	0008h
0	0000 0000 0000 0000	0000h
-0,5	1111 1111 1111 1000	FFF8h
-10,125	1111 1111 0101 1110	FF5Eh
-25,0625	1111 1110 0110 1111	FE6Fh
-55	1111 1100 1001 0000	FC90h

Таблиця 1.6. Характеристики датчика температури

Параметр	Характеристики
Інтерфейс	1-Wire однопровідний
Напруга живлення	3,0-5,0 В
Діапазон виміру температур	-55...+125 °С
Мінімальна погрішність	0,5°С (при t від -10...+85°С)
Дозвіл перетворень	60 мкА
Час вимірювання	Не більше 750 мс

Зчитування інформації з датчика ділиться на 3 етапи:

1) ініціалізація: мається на увазі вивід датчика із сплячого режиму і підготовці його до обміну інформацією. Ініціалізація включає в себе передачу контролером логічного нуля і прийомом логічного нуля, в разі успіху (рис.1.17).

2) робота з ПЗУ (постійним запам'ятовуючим пристроєм) дозволяє визначити чіп датчика і необхідна, якщо є кілька датчиків серії DS для вибору одного з них через індивідуальний номер або одночасної роботи з усіма.

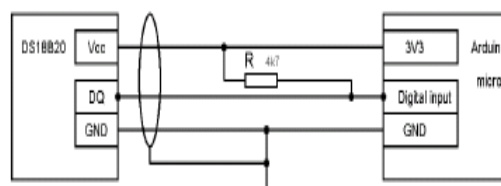


Рис.1.16 - Схема підключення цифрового датчика температури

3) отримання даних про температуру.

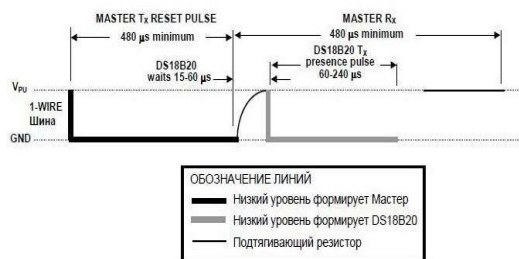


Рис. 1.17 - Ініціалізація датчика DS18B20

Детальна схема датчика представлена в Додатку 1 (рис.3).

Наступний крок – вибір датчика руху (присутності). Основною вимогою для датчика присутності є радіус виявлення, таким чином, згідно з планом будинку максимальний радіус повинен бути не менше 3 м. Даному і представленим вимогам відповідає модуль присутності HC-SR501 (рис. 1.18) на основі ППР-елемента.

Використані в датчику чутливі елементи забезпечують ефект поляризації в моменти, коли на них падає випромінювання (теплове випромінювання від людського тіла). При різкій зміні характеристик в спостережуваній зоні змінюється і напруженість в електричному полі кристала. Власне, з цієї причини інфрачервоний датчик PIR також називається піроелектричним. Як і всі детектори, такі пристрої не ідеальні. Залежно від умов вони можуть спрацьовувати на помилкові сигнали або не визначати цільові явища. Однак за сукупністю експлуатаційних властивостей в більшості випадків вони виправдовують своє застосування.

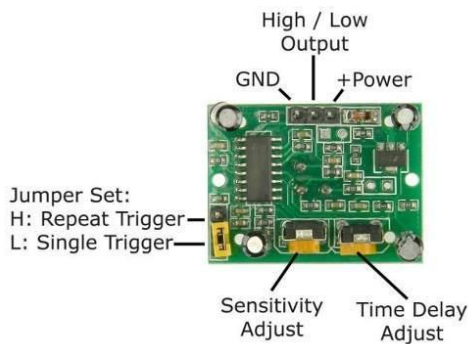


Рис. 1.18 - Датчик руху HC-SR501

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		28

Таблиця 1.7. Характеристики датчику руху

Параметр	Характеристики
Напруга живлення	5-20 В
Вихідна напруга живлення	3,3 В (високий рівень)
Вживаний струм (очікування\робочий режим)	50 мкА\ 65 мкА
Робоча температура	-15...70 °С
Радіус обнаруження	3-7 метра
Утримання високого рівня	20-300 с.
Затримка увімкнення	2,5-5 с.
Кут огляду	120 ⁰

Детальна схема датчика представлена в Додатку 1 (рис. 4).

Режим роботи датчика встановлюється за допомогою перемички Н і L. У режимі «L» логічна одиниця на виході після реєстрації руху зберігається протягом певного часу, встановленим таймером P1 і не пізніше, незалежно від наявності руху під час роботи таймера. Цей режим можна використовувати в якості охоронної сигналізації.

У нашому випадку необхідно використовувати режим «Н», в якому логічна одиниця на виході присутня протягом часу, встановленого таймером після останньої фіксації переміщення, так як кожен об'єкт, що рухається, перезапускає таймер.

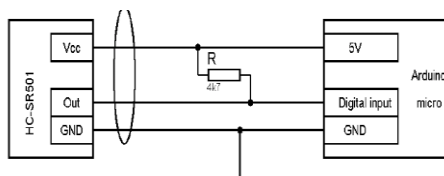


Рис.1.19 - Схема підключення датчика руху

Перемикування в високий рівень відбувається з мінімальною затримкою 0,2 с. після останнього переходу з високого стану в низький. Цей режим підходить для організації автоматичного освітлення. Щоб гарантувати високий рівень на

логічному виході між виходом і живленням 5В, буде підключений підтягаючий резистор 4,7 Ком (рис. 1.19).

Детальна схема датчика представлена в Додатку 1 (рис. 5).

Наступний крок – вибір датчика освітлення. До датчика освітлення така ж вимога така ж, як і до порогового датчика температури: можливість регулювання порога спрацьовування, тобто, ступеня освітленості, при якому з датчика на контролер буде надходити сигнал і включатися вуличне освітлення. Для цих цілей підійде датчик на основі фоторезистора GL5528 (опір якого залежить від ступеня освітленості, рис.1.20), а на виході датчика буде сигнал високого або низького рівня. Цей сигнал буде результатом порівняння різниці потенціалів на фоторезисторі і різниці потенціалів на потенціометрі (опір якого потрібно вручну відрегулювати). Щоб гарантувати високий рівень на логічному виході між виходом і живленням 5В буде підключений підтягаючий резистор 4,7Ком (рис.1.21).



Рис.1.20 - Датчик освітлення на фоторезисторі GL5528

Таблиця 1.8. Характеристики датчика освітлення на фоторезисторі GL5528

Параметр	Характеристики
Напруга живлення	3,3 - 5,0 В
Діапазон робочих температур	20...80 °С
Вихідний сигнал	0 або 1

Принцип повністю ідентичний роботі порогового датчика температури.

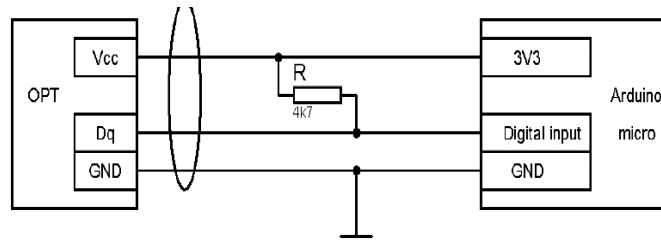


Рис.1.21 - Схема підключення датчика освітлення

Детальна схема датчика представлена в Додатку 1 (рис. 6).

У нашій системі в якості виконавчих пристроїв виступають електричні радіатори опалення, освітлювальні прилади і звукова сигналізація. Підключатися до виходів контролера вони будуть за допомогою согласуючої схеми на основі транзистора, яка посилить керуючий сигнал по потужності. Так як силова частина живиться від мережі змінного струму з напругою 220 В, то необхідна гальванічна розв'язка, яку забезпечить електромагнітне реле. У навантаження електронного ключа VT (рис. 1.22) підключається реле. Контролер за допомогою ключа регулює стан реле. Діод VD захищає схему від ЕРС самоіндукції, яка виникає в котушці реле при закритті ключа VT. При виборі реле буде враховуватися напруга комутації (в нашому випадку 12В), струм і напруга для виконавчих контактів реле. Як ключ використовуватимемо польовий транзистор, і будемо використовувати проміжне електромагнітне реле фірми Finder серії S40 (рис.1.23), яке підходить нашим вимогам (табл.1.9). Для вибору транзистора висловимо струм, що протікає через котушку реле, рівний струму стоку:

$$I_d = I_r = \frac{P_r}{U_r} = \frac{1.2}{12} = 1.2 \text{ A,}$$

де I_r - струм, що протікає через котушку, P_r - номінальна потужність, споживана котушкою, при постійному струмі, U_r - номінальна напруга котушки, I_d -струм стоку.

Зазначеним вище вимогам задовольняє польовий транзистор IRFL 4310 (табл.1.10). Його гранична напруга 2-4 В нижче напруги на виході контролера 5 В,

якого буде достатньо для відкриття переходу стік-витік. Струм 1,2 А, що протікає через котушку і рівний току стоку, потрапляє в робочий діапазон струму стік-витік.

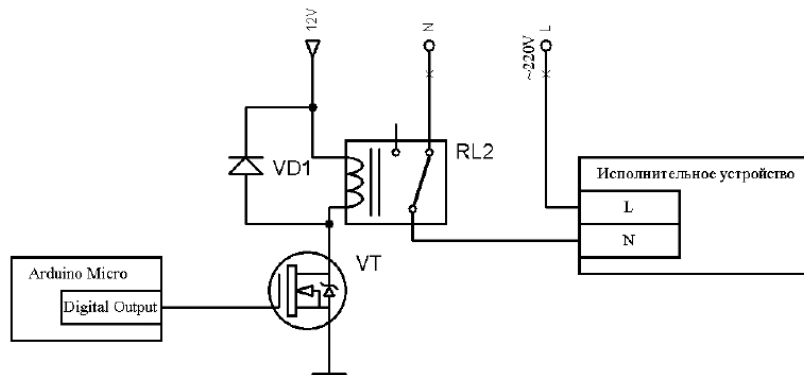


Рис. 1.22 - Схема підключення виконавчих пристроїв



Рис. 1.23 - Електромагнітне реле фірми Finder S40

Таблиця 1.9. Характеристики електромагнітного реле фірми Finder S40

Параметри	Характеристики
Номінальна напруга котушки увімкнення	12 В
Номінальна напруга ланцюга контактів	230 В
Кількість контактів перемикання	1
Номінальний струм контактів	10 А
Мінімальний струм контактів	0,3 А
Діапазон робочих температур	-40...+85 С°
Номінальна потужність котушки при постійному струмі	1,2 Вт

Принципові схеми підключення пристроїв вводу та виводу представлені на рисунках 1.24 та 1.25 відповідно.

Таблиця 1.10. Характеристики польового транзистора

Параметри	Характеристики
Структура	Канал n-типу
Пороговий струм	2-4 В
Постійний струм витоку	0,91 А
Максимальна напруга сток-виток	100 В
Максимальний струм сток-виток	1,6А
Максимальна напруга затвір-виток	20 В

1.3 Розробка алгоритмів та управляючої програми

Розробка програми для мікроконтролера проводилась у середовищі програмування Arduino IDE.

Arduino IDE - це багатоплатформовий додаток на Java. Середовище розробки складається з вбудованого текстового редактора програмного коду, області повідомлень, вікна виведення тексту (консолі), панелі інструментів з кнопками часто використовуваних команд і декількох меню. Для завантаження програм, середовище розробки підключається до апаратної частини Arduino. Дане програмне забезпечення засноване на мові програмування Processing та спроектована для програмування новачками, не знайомими близько з розробкою програмного забезпечення. Мова програмування аналогічна мові Wiring. Строго кажучи, це C ++, доповнений деякими бібліотеками. Програми обробляються за допомогою препроцесора, а потім компілюється за допомогою AVR-GCC [8].

Програма, написана в середовищі Arduino IDE, називається скетч. Скетч пишеться в текстовому редакторі, що має інструменти вирізки/вставки, пошуку/заміни тексту. Під час збереження та експорту проекту в області повідомлень з'являється пояснення, також можуть відображатися виниклі помилки. Вікно виведення тексту (консоль) показує повідомлення, що включають

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		33

повні звіти про помилки та іншу інформацію. Кнопки панелі інструментів дозволяють перевірити і записати програму, створити, відкрити та зберегти скетч, відкрити моніторинг послідовної шини.

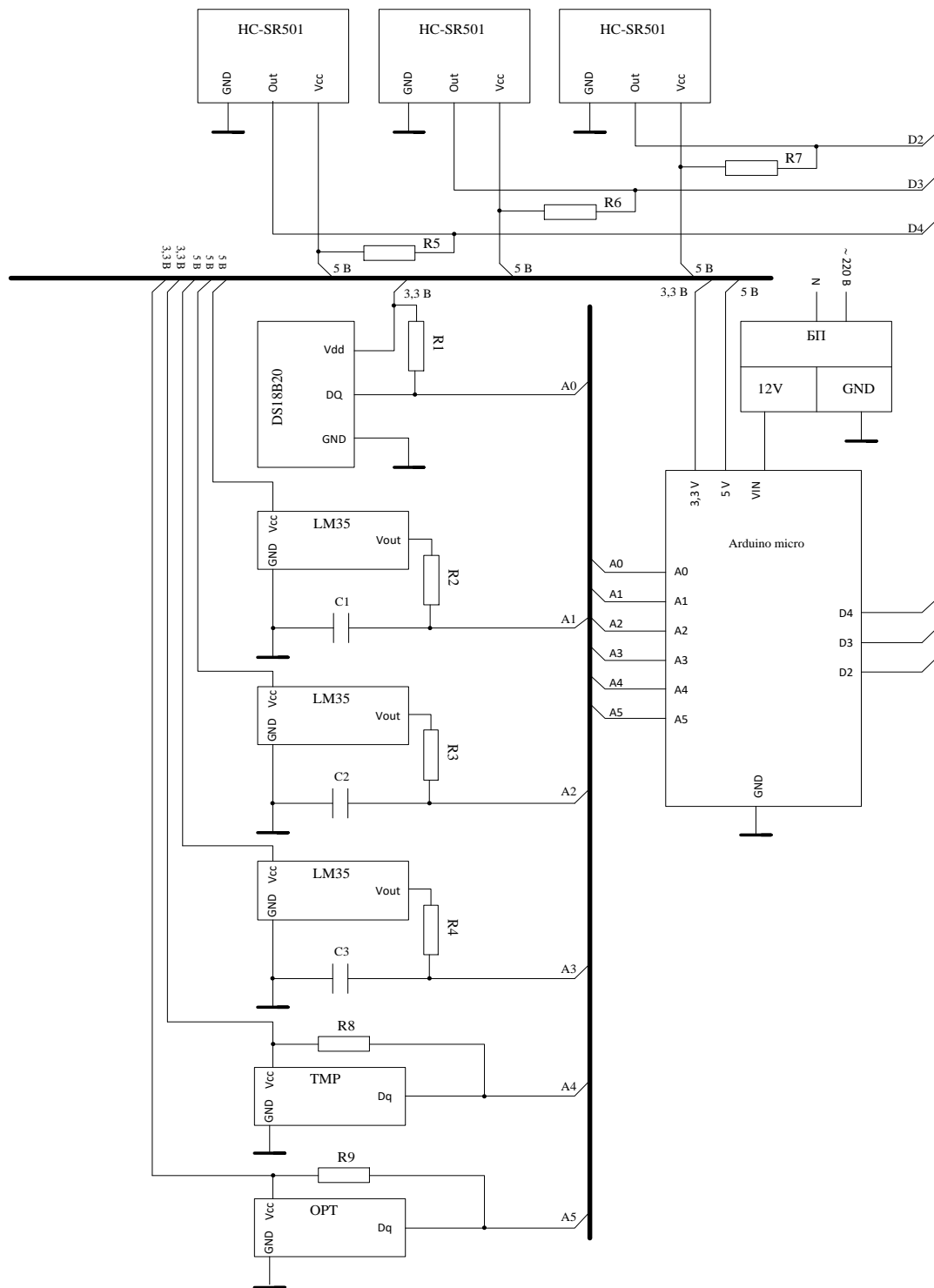


Рис. 1.24 – Принципова схема підключення датчиків

Змн.	Арк.	№ докум.	Підпис	Дата

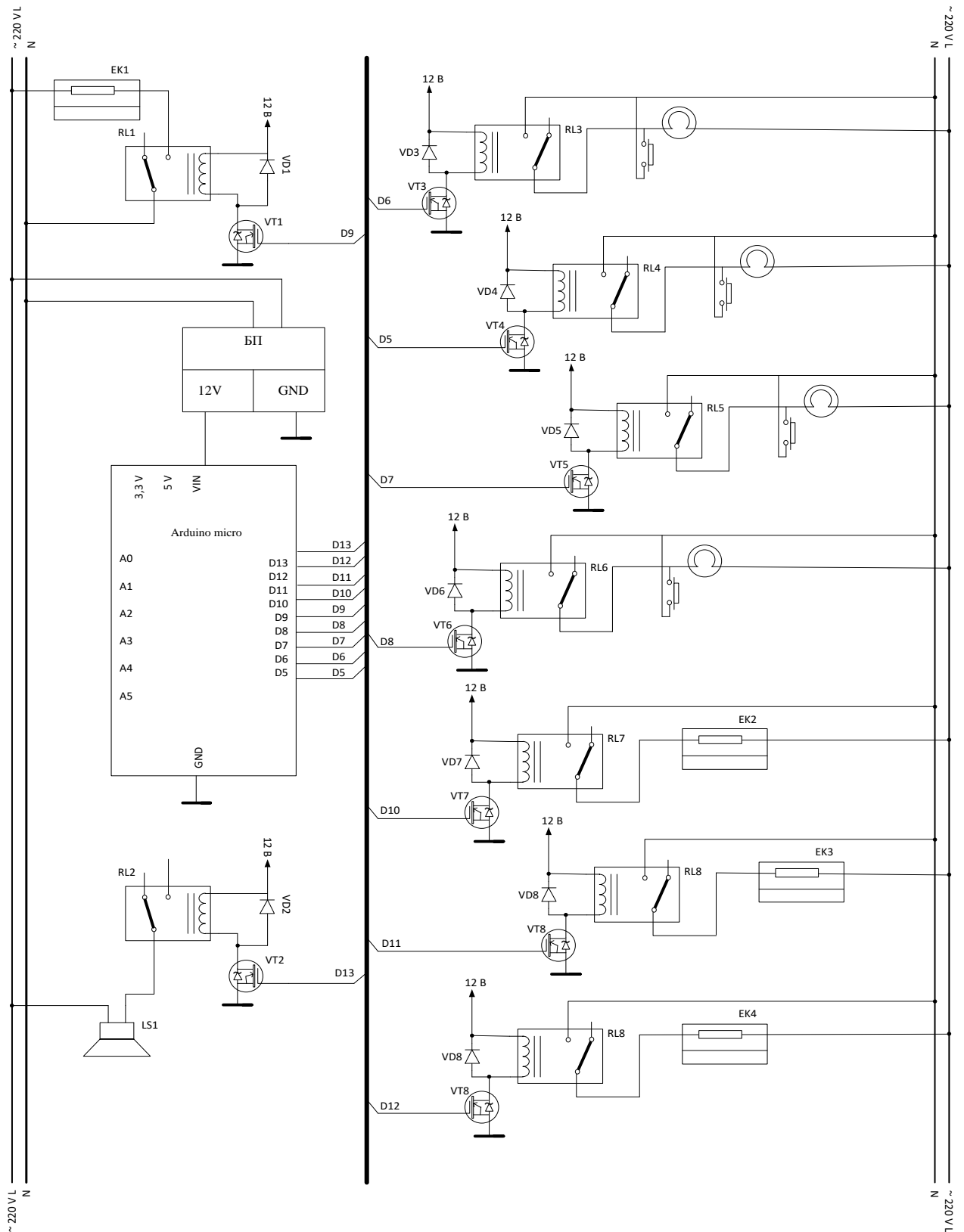


Рис. 1.25 – Принципова схема підключення виконавчих пристроїв

Змн.	Арк.	№ докум.	Підпис	Дата

КС 56. 18. 000. 00 ДП ПЗ

Арк.

35

Середовищем Arduino IDE використовується принцип блокнота - стандартне місце для зберігання програм (скетчів). Скетчі з блокнота відкриваються через команду Sketchbook з меню File та кнопкою Open на панелі інструментів. При першому запуску програми, автоматично створюється директорія для блокнота. Розташування блокнота змінюється через діалогове вікно Preferences.

Бібліотеки додають додаткову функціональність скетчам, наприклад, при роботі з апаратною частиною або при обробці даних. Для використання бібліотеки необхідно вибрати команду Import Library з меню Sketch. Одна або кілька директив #include будуть розміщені на початку коду скетчу з подальшою компіляцією бібліотек і разом зі скетчем. Завантаження бібліотек вимагає додаткового місця в пам'яті Arduino. Невикористовуванні бібліотеки можна видалити з скетчу прибравши директиву #include. Моніторинг послідовної шини відображає дані, які відправляються в платформу Arduino. Для відправлення даних необхідно ввести текст та натиснути кнопку Send або Enter. Потім вибирається швидкість передачі з випадаючого списку, відповідна значенню Serial.begin в скетчі.

Перед завантаженням скетчу (програми) в Ардуіно необхідно в налаштуваннях вибрати тип плати *Arduino Micro* (рис.1.26) і послідовний порт *COM7* (рис. 1.27). Визначення порту USB проводиться в полі *Послідовна шина USB Диспетчера пристроїв Windows* (рис.1.28).

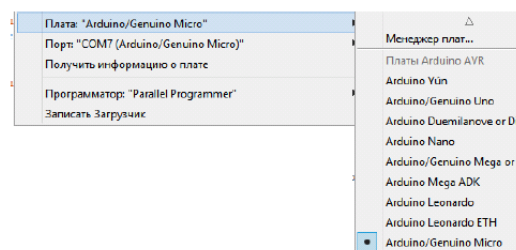


Рис.1.26 – Вибір плати в настройках Arduino IDE

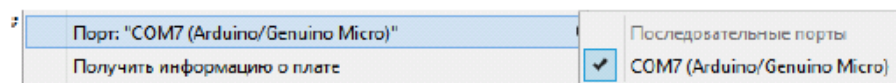


Рис.1.27 – Вибір порту в настройках Arduino IDE

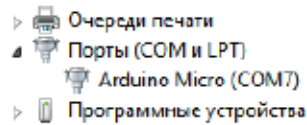


Рис.1.28 – Вибір порта в настройках Arduino IDE

Передача даних здійснюється через послідовний інтерфейс UART у вигляді послідовності біт. Кожен біт передається за рівні проміжки часу, які визначаються швидкістю передачі, яка вимірюється у біт/с (бод). Крім бітів даних і біти синхронізації: стартовий і стоповий, тобто, передача байта інформації займає 10 бітів, а не 8. Похибка часових інтервалів передачі бітів повинна бути не більше 5% (рекомендується не більше 1,5%).

Перетворювач інтерфейсу ATmega32U4 дозволяє підключати плату Ардуіно до комп'ютера через USB порт. Arduino IDE створює на комп'ютері віртуальний COM-порт, через який і відбувається обмін. Незважаючи на те, що плата підключена до комп'ютера через USB-порт, всі програми обмінюються даними через віртуальний COM-порт, не підозрюючи, що порт віртуальний.

Для роботи з апаратними UART контролерами в Arduino IDE існує вбудований клас Serial, який дозволяє здійснювати двосторонній обмін даними через UART. У нашій програмі через Serial з'єднання будуть відправлятися дані, отримані з датчиків, а також сигнали управління, які Ардуіно передала виконавчим пристроям, у вигляді масиву даних. Для цього в класі Serial є спеціальна функція, що дозволяє відправляти і зчитувати масив байтів. Перший елемент масиву буде фіксованим, щоб по ньому можна було визначити, що дані передані Ардуіно. Прийом даних управління, встановлені користувачем через інтерфейс, також буде у вигляді масиву байтів.

Опис роботи програми описано в блок-схемах на рисунках 1.29, 1.30, 1.31 і 1.32, сама програма наведена в Додатку 2.

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		37

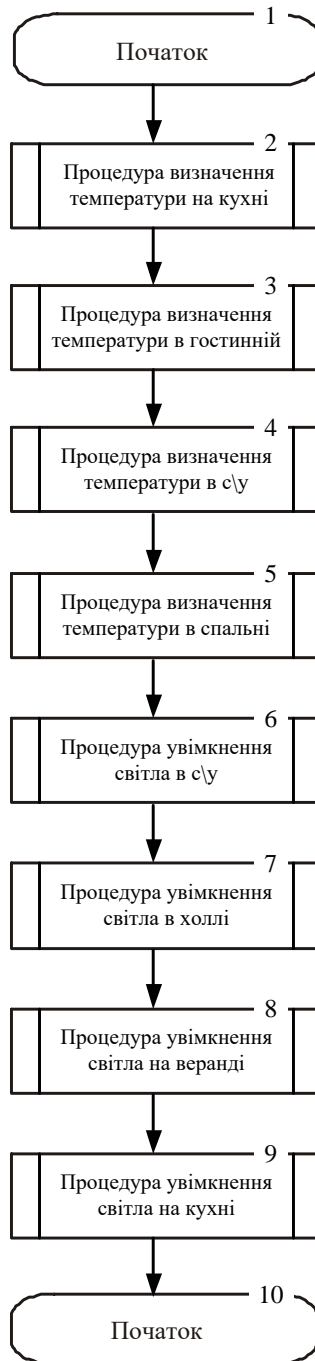


Рис.1.29 – Блок-схема роботи системи клімат-контролю

Змн.	Арк.	№ докум.	Підпис	Дата

КС 56. 18. 000. 00 ДП ПЗ

Арк.

38

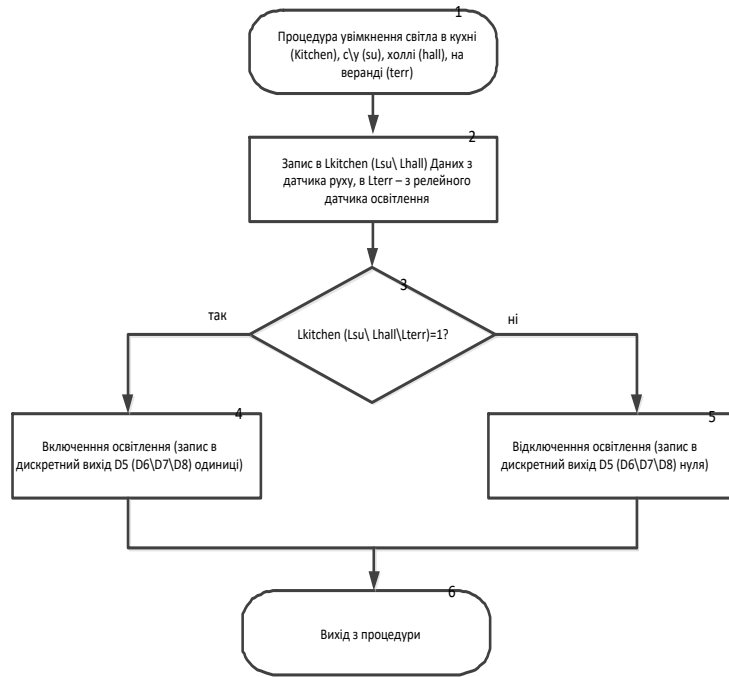


Рис.1.30 – Алгоритм роботи системи увімкнення освітлення

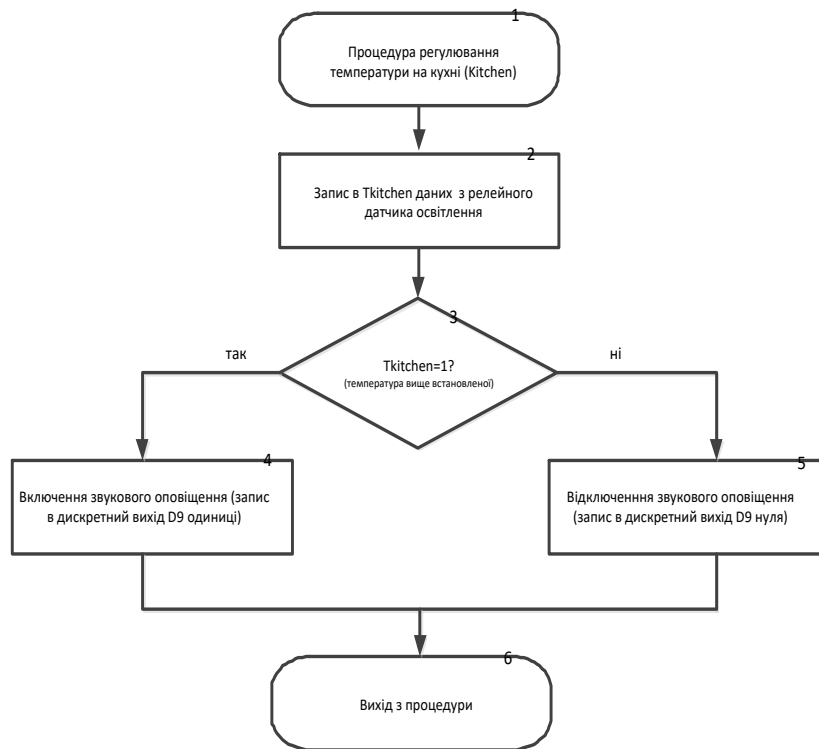
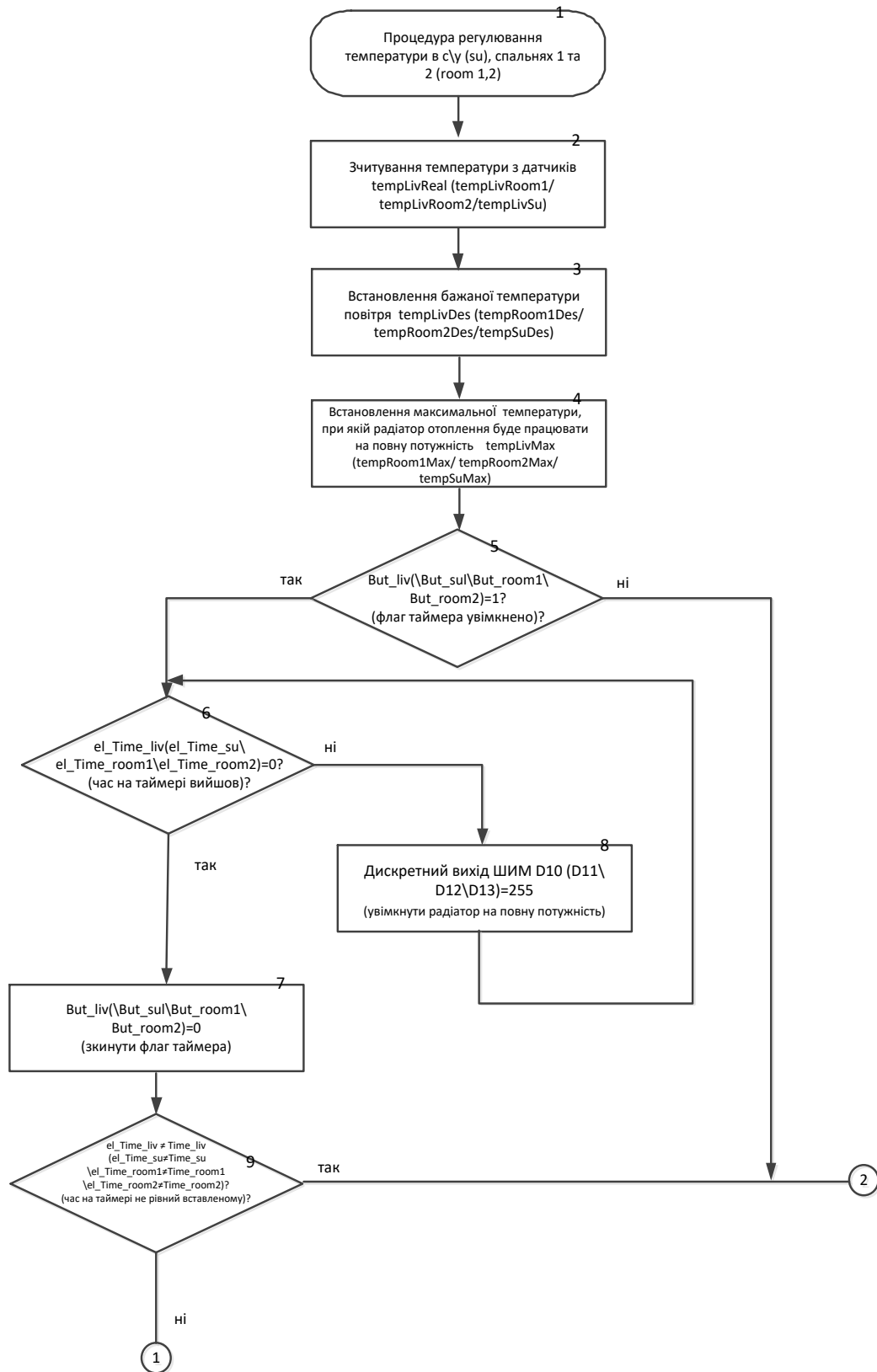


Рис.1.31 – Алгоритм роботи системи оповіщення при перевищенні температури на кухні

Змн.	Арк.	№ докум.	Підпис	Дата



Змн.	Арк.	№ докум.	Підпис	Дата

КС 56. 18. 000. 00 ДП ПЗ

Арк.

40

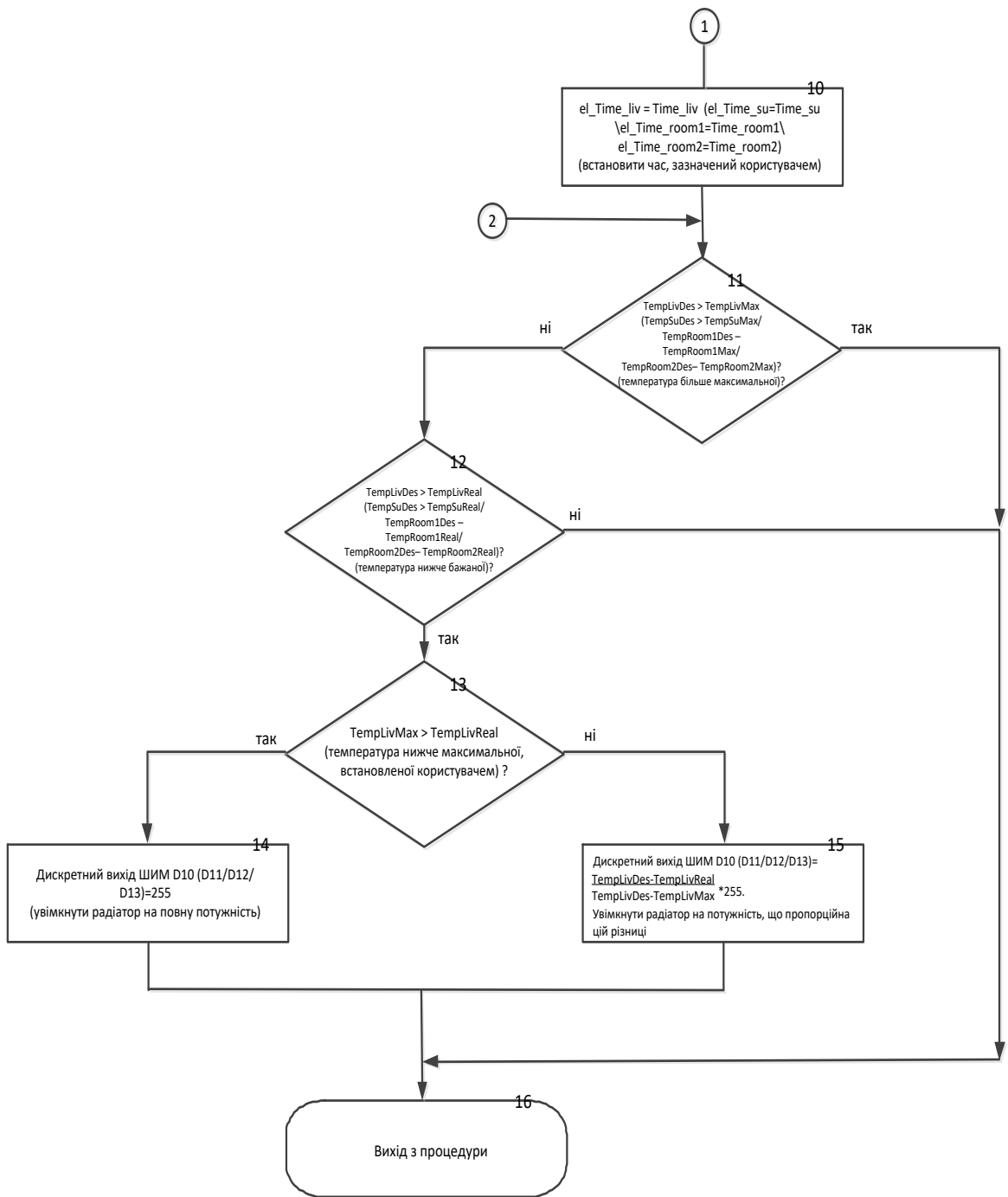


Рис.1.32 – Алгоритм регулювання температури

Робота системи розпочинається з визначення температури на кухні. Якщо показання не перевищують показань «комфортної бажаної температури», то система аналізує температурні показники в гостинній кімнаті, потім у с\у та спальні. Якщо у вказаних приміщеннях температурні показники знаходяться поза

Змн.	Арк.	№ докум.	Підпис	Дата

межами «комфортних» показань, то тут можливі варіанти. Якщо температура менше вказаних комфортних показань, то може увімкнутися обігрівач (при цьому можливе регулювання потужності обігрівача). Крім того, при перевищенні максимально встановлених температурних показників на кухні – вмикається звукове оповіщення. Так, наприклад, можна контролювати задимленість або пожежу на кухні.

Після роботи системи регулювання температури – відслідковується робота системи контролю за освітленням. У с\у, холлі, веранді та на кухні. Система працює просто – якщо в межах дії датчика руху з'являється людина - увімкнеться світло. Вимкнути можна або власноруч або автоматично після зникнення руху у зоні дії датчика. На веранді світло увімкнеться автоматично при сутінках.

Отже, в даному розділі створено структурну схему розробки, здійснено вибір елементної бази, представлено принципову схему розробки та наведено алгоритми, що пояснюють принципи роботи систем та реалізації їх функцій – увімкнення освітлення та моніторинг стану температури повітря.

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		42

2 ЕКОНОМІЧНИЙ РОЗДІЛ

У даній роботі проектується система клімат-контролю на базі платформи Arduino. Для контролю житла використовуються датчики і супутнє обладнання. Всі прилади підключаються до єдиної системи управління, в нашому випадку контролера Arduino, за допомогою якого проводиться їх моніторинг і управління.

У роботі розглядаються питання функціональної специфікації, апаратної реалізації та програмної реалізації. В функціональній специфікації розписані необхідні для автоматизації дані про об'єкт управління і детально описані функції, які повинна виконувати система. В апаратній частині наведені конкретні апаратні засоби, тобто контролер і датчики, відповідні даному об'єкту управління, їх опис, вхідні і вихідні параметри, спосіб підключення і принцип реалізації необхідних функцій. У програмній частині наведені програми, що реалізують запропоновані системі управління функції. Програми забезпечують отримання з датчиків даних про стан об'єкта управління і інтерпретацію цих даних в доступну форму.

Метою даних розрахунків є обчислення вартості виконання науково-дослідної роботи «Проектування системи клімат-контролю на базі платформи Arduino»

Оцінка якості розробленого проекту включає визначення трудомісткості і вартості його створення. Розрахунок трудомісткості НДР здійснений в наступній послідовності:

1) Складений перелік всіх етапів і видів робіт, які необхідно виконати в ході даної НДР. Після узгодження з керівником проекту допущено виключення, доповнення, об'єднання окремих етапів і видів робіт;

2) По кожному виду робіт визначений кваліфікаційний рівень виконавців.. Розподіл робіт по етапах і видах виконавців вироблений формою, наведено в таблиці 2.1.

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		43

Таблиця 2.1 - Розподіл робіт по етапах і видах виконавців

Етап проведення НДР	Вигляд робіт	Посада виконавця
Розробка технічного завдання (ТЗ)	1.Складання і затвердження ТЗ для НДР «Проектування системи клімат-контролю на базі платформи Arduino»	Дипломник, керівник
Вибір напрямку дослідження	1. Збір і вивчення науково-технічної літератури. 2. Формулювання можливих напрямів вирішення завдань, поставлених в технічному завданні НДР . 3. Вибір напрямку проведення досліджень для подальшої розробки. 4. Розробка плану проведення досліджень для подальшої розробки.	Дипломник керівник
Теоретичні і експериментальні дослідження	1 ТЕХНОЛОГІЧНИЙ РОЗДІЛ 1.1 Аналіз існуючих рішень 1.2 Розробка схем та вибір елементної бази 1.3 Розробка алгоритмів та управляючої програми	Дипломник керівник консультанти
Узагальнення і оцінка результатів досліджень	1. Узагальнення результатів попередніх етапів роботи.	Дипломник керівник консультанти

	2. Складання і оформлення звіту. Розгляд результатів проведеною НДР і прийняття результатів в цілому.	
--	--	--

В умовах відсутності нормативної бази тривалість виконання окремих робіт розраховується на основі вірогідних оцінок робіт, що задаються виконавцями.

Таблиця 2.2. - Очікувана трудомісткість робіт

Вигляд роботи	Очікуваний час виконання (дні)
1. Складання і затвердження ТЗ для НДР «Проектування системи кондиціонування приміщення за технологією «Розумний будинок»»	2
2. Збір і вивчення науково – технічної літератури, технічної документації і інших матеріалів.	3
3. Формулювання можливих напрямів вирішення завдань, поставлених в технічному завданні НДР і їх порівняльна оцінка.	2
4. Вибір напрямку проведення досліджень і способів вирішення поставлених завдань. Розробка плану проведення досліджень для подальшої розробки.	3
5. Аналітичний розділ. Огляд існуючих рішень	3
6. Технічний розділ.	4
7. Конструкторський розділ	4
8. Економічна частина	2

10. Охорона праці	2
Всього:	26

Результатом виконання НДР є науково-технічна продукція, що є закінчені науково – дослідницькі роботи, виконані відповідно до вимог, передбачених договором, і прийнятими замовником. Розрахунок собівартості і ціни виконання НДР включає наступні статті витрат: витрати на матеріали, основна і додаткова заробітна плата, відрахування до єдиного соціального фонду страхування, витрати на роботи, що виконуються сторонніми організаціями, і деякі інші.

1) Витрати на матеріали, купувальні комплектуючі, напівфабрикати визначають на основі розрахунку потреби в них за оптовими цінами, що діють і складають 170 грн.

2) До витрат «Основна заробітна плата» відносяться оплата праці виконавців, безпосередньо притягнених до її виконання. Розмір основної зарплати встановлюється виходячи з чисельності різних категорій виконавців, трудомісткості, що витрачається ними на виконання різних видів робіт, а також їх середньої заробітної плати (ставки) за один робочий день. Відповідно до статті 8 «Закону про Державний бюджет України на 2023» встановлено мінімальну заробітну плату у місячному розмірі з 1 січня 2023 року - 6700 гривень; мінімальну погодинну тарифну ставку – 40,46 грн.

Середня зарплата за один робочий день для кожного виконавця визначена по формулі:

$$Зден = п.т.с. * 8;$$

де п.т.с – погодинна тарифна ставка, грн..;

8 – тривалість робочого дня, год.

$$Зден дипломника = 40,46 * 8 = 323,68 \text{ грн.}$$

$$Зден керівника = 72 * 8 = 576 \text{ грн.}$$

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		46

Зден консультантів = 69 *8= 552 грн.

Витрати на основну заробітну плату, НДР, що включаються в собівартість, приведені в таблиці 2.3.

Таблиця 2.3 - Витрати на основну заробітну плату

Виконавець	Погодинна тарифна ставка, грн	Денна ставка, грн	Трудомісткість робочих днів	Сума основної зарплати, грн
Дипломник	40,46	323,68	26	8415,68
Керівник	72	576	1	576
Консультант по економічній частині	69	552	0,25	139,5
Консультант по охороні праці	69	552	0,25	139,5
Нормоконтроль	69	552	0,25	139,5
Всього (Зо)				9392,18

3) Витрати на додаткову заробітну плату визначаються у відсотках від основної і враховують виплати за час, що не пропрацював, встановлений законом. У наукових закладах додаткова заробітна плата складає 10-12% від основної заробітної плати.

$$Зд=12\%Zo; \quad Zd = 9392,18*0,12 = 1127,06 \text{ грн}$$

4) До складу собівартості НДР включаються податки, збори і інші обов'язкові платежі, встановлені системою оподаткування що діє. Сума до єдиного соціального внеску складає:

Відрахування до єдиного соціального внеску складає:

$$Зесв=0,22*(Zo+Zd);$$

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		47

$$З_{\text{св}}=0,22*(9392,18+1127,06) = 2314,23 \text{ грн.}$$

5) До накладних витрат відносять витрати на управління і господарське обслуговування, що відноситься до всіх виконуваних НДР.. У наукових закладах накладні витрати складають 40 -120% від основної і додаткової заробітної плати.

$$R_{\text{накл}}= (30+3д)*0,6;$$

$$R_{\text{накл}}= (9392,18+1127,06)* 0,6 = 6311,54 \text{ грн.}$$

На підставі отриманих даних по окремих статтях витрат складена калькуляція планової собівартості в цілому НДР за формою, приведеною в таблиці 2.4.

Таблиця 2.4 - Калькуляція планової собівартості

Статті витрат	Сума, грн.
1. Матеріали	170,00
2. Основна заробітна плата	9392,18
3. Додаткова заробітна плата	1127,06
4. Відрахування до єдиного соціального внеску	2314,23
5. Накладні витрати	6311,54
Планова собівартість (Спл)	19315,01

Плановий прибуток визначений по формулі:

$$Ппл = 0,1*Спл= 0,1*19315,01=1931,5 \text{ грн}$$

Де 0,1 – норматив, який враховує граничний рівень рентабельності, встановлений чинним законодавством для науково-технічної продукції.

Договірна ціна визначається по формулі:

$$Ц_{\text{нір}} = Спл + Ппл =19315,01+1931,5 =21246,51 \text{ грн}$$

Звідси ціна реалізації становить:

$$Ц_{\text{р}} = Ц_{\text{нір}} + ПДВ;$$

$$Ц_{\text{р}} = 21246,51 +21246,51 *0,2= 25495,81 \text{ грн.}$$

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		48

3 ОХОРОНА ПРАЦІ

Одним із головних завдань є збільшення продуктивності праці, поліпшення якості виробів, досягнення високих економічних показників. Все це нерозривно пов'язане з умовами праці, розробкою та впровадженням заходів до попередження впливу шкідливих та небезпечних факторів на працівників.

Тому у даному розділі дипломного проєкту приведено аналіз виробничих факторів, що мають місце в роботі програміста при проектуванні системи клімат контролю на базі платформи Arduino.

3.1 Аналіз небезпечних і шкідливих факторів, що впливають на програміста при розробці даного програмного комплексу

В дипломному проєкті використовується праця програміста, тому умови праці та забезпечення безпеки його роботи здійснюються при дотриманні всіх норм та вимог безпечного використання комп'ютера і його можливостей для працівника.

Основними факторами шкідливого впливу ПК на організм є:

- електромагнітні поля;
- електромагнітні випромінювання;
- розгортка зображення на моніторі;
- мелькання зображення на екрані;
- тривала нерухомість пози оператора;

Зменшити вплив перерахованих факторів ризику і зберегти здоров'я людини, що постійно використовує в роботі ПК, дозволяє наступне:

- скорочення тривалості роботи з ПК;
- збільшення відстані між користувачем і ПК;

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		49

- технічний захист (правильне підключення комп'ютера, тобто правильна організація робочого місця з точки зору електромережі - обов'язково трьох-проводна мережу і заземлення системного блоку).

3. 2 Гігієнічні вимоги до виробничого середовища

Дипломним проектом враховані санітарні нормативи освітлення, вимоги до параметрів мікроклімату (температура, відносна вологість), ступеня і сили вібрації, звукового шуму і вогнестійкості приміщення, а також характеристики електромагнітного, ультрафіолетового та інфрачервоного полів.

3.2.1 Вимоги до приміщення

Площа приміщення має бути не менше 6,0 кв. м. із розрахунку на одне робоче місце, а об'єм – не менше 20,0 куб. м. Віконні прорізи приміщень для роботи з персональними комп'ютерами мають бути обладнані регульованими пристроями (жалюзі, завіски, зовнішні козирки. Для внутрішнього оздоблення приміщень з персональними комп'ютерами слід використовувати дифузно-відбивні матеріали з коефіцієнтами відбиття для стелі 0,7-0,8, для стін 0,5-0,6.

Покриття підлоги повинне бути матовим з коефіцієнтом відбиття 0,3-0,5. Поверхня підлоги має бути рівною, неслизькою, з антистатичними властивостями. Забороняється для оздоблення інтер'єру приміщень з персональними комп'ютерами застосовувати полімерні матеріали (деревинно-стружкові плити, шпалери, що миються, рулонні синтетичні матеріали, шаруватий паперовий пластик тощо), що виділяють у повітря шкідливі хімічні речовини. Приміщення можуть обладнуватись шафами для зберігання документів, магнітних дисків, полицями, стелажми, тумбами тощо з урахуванням вимог до площі приміщень.

Слід щоденно робити вологе прибирання. Крім того, ці приміщення мають бути оснащені аптечками першої медичної допомоги.

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		50

3.2.2 Освітлення

Приміщення, в яких встановлені персональні комп'ютери, повинні мати природне та штучне освітлення. Природне освітлення здійснюється через світові прорізи (вікна), орієнтовані переважно на північ чи північний схід. Штучне освітлення в приміщенні здійснюється системою загального рівномірного освітлення. На поверхні столу в зоні розміщення документів штучне освітлення має становити 300-500лк. Вимога виконується.

3.2.3 Шум

Так як шум має 35Дб, сприйняття шуму людським вухом межується від 20Дб до 120 дб, це означає, що при роботі за ЕОМ шум не заважає, працівнику працювати.

Для запобігання виникнення інших шумів у відповідності з ГОСТ 12.1.029-80 зниження шуму й вібрації в приміщенні дипломним проектом передбаченні звукоізоляція вікон та дверей.

3.2.4 Вимоги до організації робочого місця працівника

Робоче місце має ключове значення в продуктивності праці. Грамотно організоване робоче місце не тільки сприяє максимальної концентрації на завданнях, а й підвищує настрій, оскільки люди вважають, що комфортні умови є результатом позитивного ставлення керівництва компанії до своїх співробітників

Відзначимо найважливіші фактори, які впливають на продуктивність роботи співробітників:

- Ергономічний комфорт (зручне крісло, меблі, швидкий доступ до необхідних деталей)
- Хороший рівень освітлення і оптимальний температурний режим;
- Відсутність зовнішніх подразників, шуму;

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		51

– Наявність зони відпочинку, свіжої води, кави, снєків і іншого.

Грамотно організоване робоче місце є комплексним питанням, тобто сукупністю факторів зручності, а не наявністю одного з них. Важливим фактором, крім наведених, є свобода офісного простору.

Розмір одного робочого місця має становити не менше 6 квадратних метрів, об'єм – 20 кубічних метрів. На столі працівника можливо розмістити допоміжні для роботи пристрої (принтери, колонки, сканери), а також місця для зберігання документів, за умови, що це не обмежуватиме видимість екрану і не заважатиме працівнику. Робочий стілець робітника має бути підйомно-поворотним, легко регульованим за висотою та забезпечувати належну підтримку та зручне положення спини і хребта особи.



Рис. 3.1 – Організація робочого місця

При початку роботи кожна особа має пройти лікарський огляд. Окрім того, при подальшій трудовій діяльності, така особа підлягає регулярному лікарському огляду не рідше ніж раз на 2 роки. Обов'язковим є проходження таких лікарів як терапевта, невропатолога та офтальмолога. Мають бути чітко встановлені перерви для відпочинку працівників (окрім обідньої), як правило, тривалістю 10-15 хвилин раз на годину або дві, в залежності від складності роботи. В будь-якому випадку, повинен передбачатися такий розпорядок роботи, щоб час неперервної роботи з

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		52

комп'ютером був не більше ніж 4 години. Додатково, для збереження належного рівня здоров'я та професійної придатності робітників, рекомендується виділити окреме побутове приміщення для перепочинку працівників і зняття ними нервово-емоційного напруження, що виникає при роботі з комп'ютером.

3.3. Пожежна безпека

Пожежна безпека може бути забезпечена заходами пожежної профілактики і активного пожежного захисту. Пожежна профілактика включає комплекс заходів, спрямованих на попередження пожежі або зменшення його наслідків.

Пожежна безпека входить в комплекс заходів з охорони праці, і організаційна робота в цій сфері на об'єктах господарювання включає широкий спектр заходів, а саме:

- 1) створення умов для безпечної праці,
- 2) мінімізації ризику виникнення пожеж,
- 3) своєчасне і повноцінне забезпечення технічними засобами для запобігання займання та усунення самих пожеж та їх наслідків,
- 4) контроль дотримання протипожежних вимог і норм законодавства,
- 5) розробка і впровадження регламентів по гасінню пожеж, евакуації та порятунку з місць пожежі й задимлення людей і майна (матеріальних цінностей),
- б) внутрішнє і зовнішнє навчання співробітників.

Пожежна безпека при роботі з комп'ютером передбачає обережність при обслуговуванні, ремонтних та профілактичних роботах та виконувannya всіх інструкцій працівником щодо пожежної безпеки. Заземлені конструкції, що знаходяться в приміщеннях, де розміщені робочі місця (батареї опалення, водопровідні труби, кабелі із заземленим відкритим екраном), мають бути надійно захищені діелектричними щитками або сітками з метою недопущення потрапляння працівника під напругу. Приміщення, де розміщені робочі місця, мають бути

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		53

оснащені системою автоматичної пожежної сигналізації і вогнегасниками відповідно до вимог чинного законодавства України. Проходи до засобів пожежогасіння мають бути вільними

Всі приміщення повинні бути забезпечені первинними засобами пожежогасіння: пожежним водопостачанням (пожежні крани ПК), пожежні щити з набором пожежного інструменту, вуглекислотними або порошковими вогнегасниками



Рис. 3.2 – Організація робочого місця

У випадку виникнення пожежі необхідно відключити електроживлення, викликати по телефону 101 пожежну команду, евакуювати людей із приміщення відповідно плану евакуації.

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		54

ВИСНОВКИ

В результаті роботи згідно з технічним завданням була спроектована система клімат-контролю на базі платформи Arduino.

Були реалізовані:

- система автоматичного вуличного освітлення, по спрацюванню датчика освітленості;
- система автоматичного освітлення на кухні, в санвузлі, в холі по спрацюванню датчика руху;
- система терморегулювання за рахунок отримання даних з датчиків температури;
- система звукової сигналізації на кухні, після спрацювання порогового термодатчика при перевищенні встановленого порога температури.

Дану систему можна вдосконалити, додавши GSM-модем, що дозволяє користувачеві віддалено отримувати свідчення об'єкта управління і відправляти сигнали. Також можна додати вивід повідомлень про відмови в системі управління і оповіщення оператора про аварійні ситуації. Є можливість доопрацювати систему на предмет відстеження показників датчиків і роботу виконавчих пристроїв через графічний інтерфейс, тим самим встановлювати параметри роботи опалювальних елементів через графічний інтерфейс (для кожної кімнати окремо), встановлювати через графічний інтерфейс час таймера безперервної роботи опалювальних елементів (для кожної кімнати окремо).

Можна розширити функціонал нашого «Розумного будинку», додавши систему пожежної сигналізації, систему сигналізації несанкціонованого проникнення, сигналізацію при можливості затоплення і багато іншого.

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		55

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Вікіпедія – вільна енциклопедія, стаття „Закон Мура“. [Електронний ресурс]. – Режим доступу: wikipedia.org
2. Вікіпедія – свободная энциклопедия, стаття „Три закона роботехніки“. [Електронний ресурс]. – Режим доступу: wikipedia.org
3. Популярная робототехника – «Сенсор». [Електронний ресурс]. – Режим доступу: rorobot.eu
4. Інформаційний сайт по розробці роботів Home roboticlab – „Инфракрасный датчик расстояния“. [Електронний ресурс]. – Режим доступу: home.roboticlab.eu
5. Офіційний сайт Ардуіно. [Електронний ресурс]. – Режим доступу: arduino.cc
6. ATmega32U4 8-битный AVR микроконтроллер с 32 КБайт программируемой в системе FLASH памятью и USB контролером. [Електронний ресурс]. – Режим доступу: http://catalog.gaw.ru/index.php?page=component_detail&id=21540
7. Белов А.В. - Конструирование устройств на микроконтроллерах. - СПб.: Наука и Техника, 2005. – 256 с.
8. Кузин А.В. , Жаворонков М.А. Микропроцессорная техника: учебник для студ. сред. проф. образования. – 3-е издание, стер.-М: Издательский центр «Академия», 2007. - 304с.
9. Петин В.А. Проекты с использованием контроллера Arduino.-2-е изд. перераб. и доп. – СПб.: БХВ-Петербург, 2015. - 464с.
10. Соммер У. Программирование микроконтроллерных плат Arduino/ Freeduino. – СПб.: БХВ-Петербург, 2012.-256с.
11. Гук М. Аппаратные интерфейсы ПК. – СП.: Издательский дом «Питер», 2002. – 607с.

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		56

12. Полякова О. В. Класифікація функціональних складових елементів системи інтелектуального керування середовищем при проектуванні житла. – Вісник КНУТД №4 (100), 2016. Стр.133-141
13. Гололобов, В.Н. «Умный дом» своими руками [Текст] / В.Н. Гололобов. – М.: НТ Пресс, 2007. – 416 с.
14. Микроконтроллер – это... Что такое Микроконтроллер? [Электронный ресурс]. – <https://dic.academic.ru/dic.nsf/ruwiki/69464>. – Дата обращения: 17.04.2018.
15. Основы программирования микроконтроллеров / А.Бачинин, В. Панкратов, В. Наколряков. – М.: ООО «Амперка», 2013. – 207 с.
16. Сопер, М.Э. Практические советы и решения по созданию «Умного дома» / М.Э. Сопер. – М.: НТ Пресс, 2007. – 432 с.
17. Тесля, Е.В. «Умный дом» своими руками. Строим интеллектуальную цифровую систему в своей квартире / Е.В. Тесля. – СПб.: Питер, 2008. – 219 с.: ил.
18. Харке В.Н. Умный дом. Объединение в сеть бытовой техники и системы коммуникаций в жилищном строительстве / В.Н. Харке. – М.: Техносфера, 2006. – 290 с.
19. Эксперимент 10. Кнопочный переключатель [Амперка – Вики] [Электронный ресурс]. – <http://wiki.amperka.ru/%D0%BA%D0%BE%D0%BD%D1%81%D0%BF%D0%B5%D0%BA%D1%82-arduino:%D0%BA%D0%BD%D0%BE%D0%BF%D0%BE%>
20. Эксперимент 16. Метеостанция [Амперка – Вики] [Электронный ресурс]. – <http://wiki.amperka.ru/%D0%BA%D0%BE%D0%BD%D1%81%D0%BF%D0%B5%D0%BA%D1%82-arduino:%D0%BC%D0%B5%D1%82%D0%B5%D0%BE%D1%81%D1%82%D0%B0%D0%BD%D1%86%D0%B8%D1%8F>
21. Элсенпитер, Т.Р. «Умный Дом строим сами»/Т.Р. Элсенпитер, Дж. Велт. – М.: КУДИЦ-ОБРАЗ, 2005. – 384 с.

					КС 56. 18. 000. 00 ДП ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		57

Додаток 1. Принципові схеми датчиків

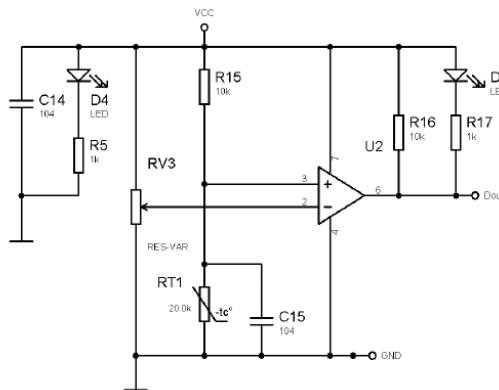


Рис.1 - Принципова схема релейного датчика температури

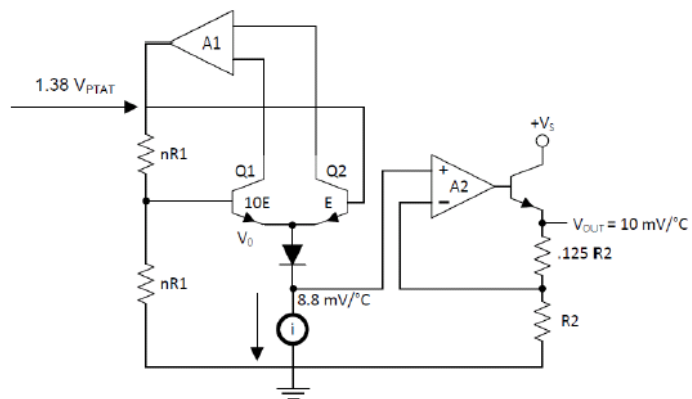


Рис.2 - Принципова схема аналогового датчика температури LM35

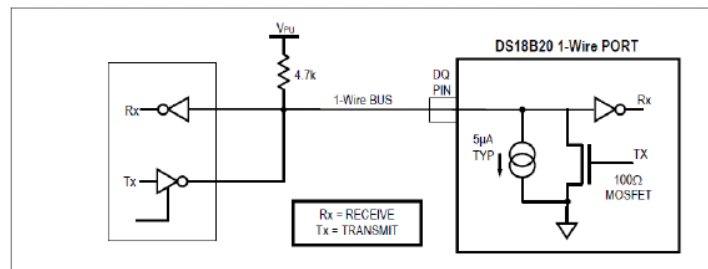


Рис.3 - Принципова схема цифрового датчика температури DS18B20

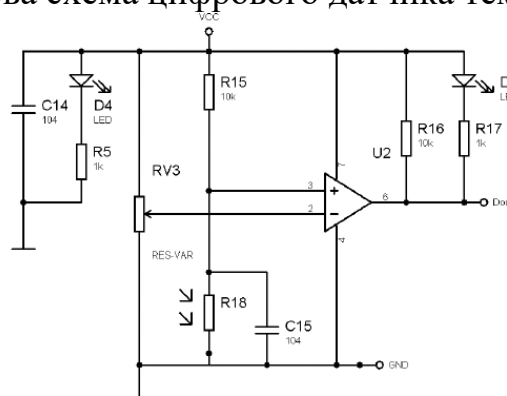


Рис.4 - Принципова схема релейного датчика освітлення

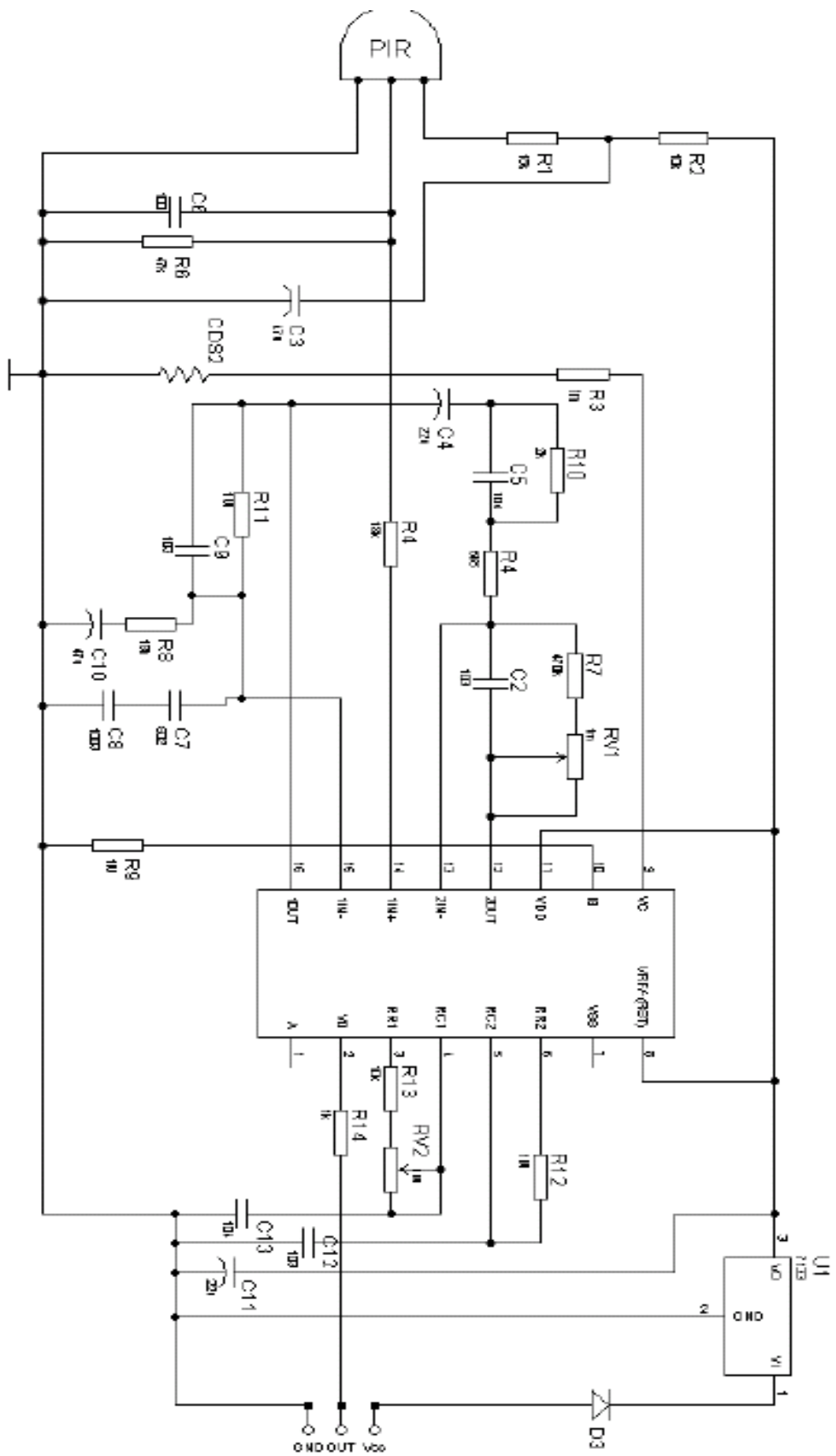


Рис.5 - Принципова схема датчика руху HC-SR501

Додаток 2. Код програми

Програма на Arduino IDE:

```
byte input [18], inputcheck [19], output [19];
/*
input [0] - прапор таймера вітальні
input [1] - максимальна температура вітальні
input [2] - бажана температура вітальні
input [3] - прапорець таймера ванній
input [4] - максимальна температура ванній
input [5] - бажана температура ванній
input [6] - прапорець таймера кімнати 1
input [7] - максимальна температура кімнати 1
input [8] - бажана температура кімнати 1
input [9] - прапорець таймера кімнати 2
input [10] - максимальна температура кімнати 2
input [11] - бажана температура кімнати 2
input [18] = 253 завжди
*/
/*
A0 output [0] - температура С / У
D10 output [1] - включений радіатор С / У
D2 output [2] - датчик присутності С / У
D5 output [3] - включено освітлення С / У
A4 output [4] - релейний датчик температури на кухні
D9 output [5] - включена сирена на кухні
D3 output [6] - датчик присутності на кухні
D6 output [7] - включено освітлення на кухні
A1 output [8] - температура в кімнаті 1
D11 output [9] - включений радіатор в кімнаті 1
A2 output [10] - температура в кімнаті 2
D12 output [11] - включений радіатор в кімнаті 2
A3 output [12] - температура в гостинній
D13 output [13] - включений радіатор в вітальні
D4 output [14] - датчик присутності в холі
D7 output [15] - включено освітлення в холі
D5 output [16] - датчик освітленості на веранді
D8 output [17] - включено освітлення на веранді
output [18] - завжди 254
*/
void setup () {
// put your setup code here, to run once:
Serial.begin (9600);
pinMode (A0, INPUT); // дані з датчика температури у ванній
pinMode (A1, INPUT); // дані з датчика температури в кімнаті 1
pinMode (A2, INPUT); // дані з датчика температури в кімнаті 2
pinMode (A3, INPUT); // дані з датчика температури в вітальні
pinMode (A4, INPUT); // дані з релейного датчика температури на кухні
pinMode (A5, INPUT); // дані з релейного датчика освітленості на веранді
pinMode (D2, INPUT); // дані з датчика присутності в ванній
pinMode (D3, INPUT); // дані з датчика присутності на кухні
pinMode (D4, INPUT); // дані з датчика присутності в холі
```

```

pinMode (D5, OUTPUT); // освітлення на кухні
pinMode (D6, OUTPUT); // освітлення в холі
pinMode (D7, INPUT); // освітлення у ванній
pinMode (D8, INPUT); // освітлення у ванній
pinMode (D9, INPUT); // звукова сигналізація
pinMode (D10, INPUT); // радіатор опалення у ванній
pinMode (D11, INPUT); // радіатор опалення в кімнати 1
pinMode (D12, INPUT); // радіатор опалення в кімнати 2
pinMode (D13, INPUT); // радіатор опалення вітальні
void loop ()
{
Serial.readBytes (inputcheck, 19);
while (input [18] != 253) {}
for (int i = 0; i <18; i ++)
input [i] = inputcheck [i];
if (digitalRead (A4)) // включення звукової сигналізації на кухні, якщо спрацював датчик
{DigitalWrite (D9, HIGH); output [5] = 1; } Else {digitalWrite (D9, LOW); output [5] = 0; }
if (digitalRead (A5)) // включення освітлення на веранді, якщо спрацював датчик освітленості
{DigitalWrite (D8, HIGH); output [17] = 1; } Else {digitalWrite (D8, LOW); output [17] = 0; }
if (digitalRead (D2)) // включення освітлення у ванній, якщо спрацював датчик присутності
{DigitalWrite (D7, HIGH); output [15] = 1; } Else {digitalWrite (D7, LOW); output [15] = 0; }
if (digitalRead (D3)) // включення освітлення на кухні, якщо спрацював датчик присутності
{DigitalWrite (D5, HIGH); output [3] = 1;} else {digitalWrite (D5, LOW); output [3] = 0}
if (digitalRead (D4)) // включення освітлення в холі, якщо спрацював датчик присутності
{DigitalWrite (D6, HIGH); output [7] = 1; } Else {digitalWrite (D6, LOW); output [7] = 0; }
analogWrite (D10, ReadRadiator (input [3], input [4], input [5])); // управління радіатором по
датчику температури у ванній
if (ReadRadiator (input [3], input [4], input [5])> 0) output [1] = 1; else output [1] = 0;
analogWrite (D13, ReadRadiator (input [0], input [1], input [2])); // управління радіатором датчику
температури в вітальні
if (ReadRadiator (input [0], input [1], input [2])> 0) output [13] = 1; else output [13] = 0;
analogWrite (D11, ReadRadiator (input [6], input [7], input [8])); // управління радіатором датчику
температури в кімнаті 1
if (ReadRadiator (input [6], input [7], input [8])> 0) output [9] = 1; else output [9] = 0;
analogWrite (D12, ReadRadiator (input [9], input [10], input [11])); // управління радіатором
датчику температури в кімнаті 2
if (ReadRadiator (input [9], input [10], input [11])> 0) output [11] = 1; else output [11] = 0;
Serial.readBytes (inputcheck, 19); // зчитує символи, що надходять через послідовний порт, і
поміщає їх в приймальний буфер.
if (inputcheck [18] == 253) // якщо зчитали масив, відправлений в порт з інфрачервоного
інтерфейсу
{
for (int i = 0; i <18; i ++)
input [i] = inputcheck [i]
}
WryteInfo ();
Serial.write (output, 19); // запис в послідовний порт даних з датчиків}
}
byte ReadRadiator (byte tempDes, byte tempMax, byte tempReal, byte flag)
{
// бажана менше температури, при якій радіатор включений на повну потужність, отож
користувач щось наплутав

```

```

if (tempMax > tempDes)
{
return 0;
}
if (flag == 1) // якщо запущений таймер, примусово включається радіатор
return 254;
if (tempReal < tempDes) // якщо температура менше бажаної
{
if (tempReal > tempMax) // якщо температура менше температури, при якій радіатор повинен
бути включений на повну потужність
return 254; // повертаємо максимальне число
else
{// повертаємо число, пропорційне відносній різниці температур
return round ((tempDes-tempReal) * 254 / (tempDes-tempMax));
}
}
else {return 0;} // якщо температура не менше бажаної
}
void WryteInfo ()
{
output [0] = round (analogRead (A0) /2.046); // температура в санвузлі
output [2] = digitalRead (D2); // - датчик присутності С / У
output [4] = digitalRead (A4) // - релейний датчик температури на кухні
output [6] = digitalRead (D3); // - датчик присутності на кухні
output [8] = round (analogRead (A1) /2.046); // - температура в кімнаті 1
output [10] = round (analogRead (A2) /2.046); // - температура в кімнаті 2
output [12] = round (analogRead (A3) /2.046); // - температура в гостинній
D4 output [14] = digitalRead (D4) // - датчик присутності в холі
D5 output [16] = digitalRead (D5) // - датчик освітленості на веранді
output [18] = 254;
}
#include <OneWire.h>
// OneWire DS18S20, DS18B20, DS1822 Temperature Example
OneWire ds (2); // on pin 2 (a 4.7K resistor is necessary)
int vcc = 19;
void setup (void) {
pinMode (vcc, OUTPUT);
Serial.begin (9600);
}
void loop (void) {
digitalWrite (vcc, HIGH);
byte i;
byte present = 0;
byte type_s;
byte data [12];
byte addr [8];
float celsius, fahrenheit;
if (! ds.search (addr)) {
ds.reset_search ();
delay (250);
return;
}
}

```

```

for (i = 0; i <8; i ++ ) {
}
if (OneWire :: crc8 (addr, 7) != addr [7]) {
Serial.println ( "CRC is not valid!");
return;
}
Serial.println ();
ds.reset ();
ds.select (addr);
ds.write (0x44, 1); // start conversion, with parasite power on at the end
delay (1000); // maybe 750ms is enough, maybe not
present = ds.reset ();
ds.select (addr);
ds.write (0xBE); // Read Scratchpad
data [i] = ds.read ();
}
int16_t raw = (data [1] << 8) | data [0];
if (type_s) {
raw = raw << 3; // 9 bit resolution default
if (data [7] == 0x10) {
raw = (raw & 0xFFF0) + 12 - data [6];
}
} Else {
byte cfg = (data [4] & 0x60);
// at lower res, the low bits are undefined, so let's zero them
if (cfg == 0x00) raw = raw & ~ 7; // 9 bit resolution, 93.75 ms
else if (cfg == 0x20) raw = raw & ~ 3; // 10 bit res, 187.5 ms
else if (cfg == 0x40) raw = raw & ~ 1; // 11 bit res, 375 ms
}
celsius = (float) raw / 16.0;
fahrenheit = celsius * 1.8 + 32.0;
Serial.print (celsius);
}

```

Програма в Processing:

```

import com.dhchoi.CountdownTimer;
import com.dhchoi.CountdownTimerService;
import processing.serial. *;
Serial myPort;
float [] input;
byte inputLenght = 18;
/*
A0 input [0] - температура С / У
D10 input [1] - включений радіатор С / У
D2 input [2] - датчик присутності С / У
D5 input [3] - включено освітлення С / У
A4 input [4] - релейний датчик температури на кухні
D9 input [5] - включена сирена на кухні
D3 input [6] - датчик присутності на кухні
D6 input [7] - включено освітлення на кухні
A1 input [8] - температура в кімнаті 1
D11 input [9] - включений радіатор в кімнаті 1
A2 input [10] - температура в кімнаті 2

```

```

D12 input [11] - включений радіатор в кімнаті 2
A3 input [12] - температура в гостинній кімнаті
D13 input [13] - включений радіатор в вітальні
D4 input [14] - датчик присутності в холі
D7 input [15] - включено освітлення в холі
A5 input [16] - датчик освітленості на веранді
D8 input [17] - включено освітлення на веранді
input [18] - завжди 254
*/
byte [] output;
/*
output [0] - прапор таймера вітальні
output [1] - максимальна температура вітальні
output [2] - бажана температура вітальні
output [3] - прапор таймера в ванній
output [4] - максимальна температура в ванній
output [5] - бажана температура в ванній
output [6] - прапор таймера кімнати 1
output [7] - максимальна температура кімнати 1
output [8] - бажана температура кімнати 1
output [9] - прапор таймера кімнати 2
output [10] - максимальна температура кімнати 2
output [11] - бажана температура кімнати 2
*/
int text_but_size = 15; // розмір тексту в кнопках
color cS = # 2A4ED8, cOn = # FF2962, cOff = # 79142F, // колір тексту і кнопки
// кольору датчиків і виконавчих пристроїв
green = # 00FF3D, greenD = # 3D6446, greyD = # 000000, grey = # 8D9D91, blue = # 0074FF,
orange = # FFAC05, yellow = # F6FF00, yellowD = # 828352, red = # FC1703, redD = # 9B615C;
PImage bat1, bat2;
// Створення об'єктів, які буду імітувати виконавчі пристрої
PShape batRoom1, batRoom2,
lampKitch, // освітлення на кухні
lampSu, // освітлення в санвузол
lampHall, // освітлення в холі
lampOut, // освітлення на веранді
radiatorSu, // опалення у ванній
radiatorLiv, // опалення в вітальні
radiatorRoom1, // опалення в кімнаті 1
radiatorRoom2, // опалення в кімнаті 2
soundKitch, // звукове сповіщення на кухні
// Створення об'єкта таймера, для фіксації секунд
CountdownTimer timer;
// Вивід залишку часу до включення опалення
String timeTextLiv = "", // в вітальні
timeTextRoom1 = "", // в першій спальні
timeTextRoom2 = "", // в другій спальні
timeTextSu = ""; // в санвузлі
int timeSecondsLiv = 0, timeMinutesLiv = 0, timeHoursLiv = 0; // залишковий час, який буде
виводитися для кухні
int timeSecondsSu = 0, timeMinutesSu = 0, timeHoursSu = 0; // залишковий час, який буде
виводитися для санвузла

```

```

int timeSecondsRoom1 = 0, timeMinutesRoom1 = 0, timeHoursRoom1 = 0; // залишковий час, який
буде виводитися для спальні
int timeSecondsRoom2 = 0, timeMinutesRoom2 = 0, timeHoursRoom2 = 0; // залишковий час, який
буде виводитися для спальні
int eTimeLiv = 0, eTimeSu = 0, eTimeRoom1 = 0, eTimeRoom2 = 0; // Час, що залишився. Для
роботи таймера
PImage bg;
Slider sliderLiv, sliderLiv_Tmax, sliderLiv_Tdes; // слайдер для установки часу таймера, бажаної
температури, максимальної
Slider sliderRoom1, sliderRoom1_Tmax, sliderRoom1_Tdes;
Slider sliderRoom2, sliderRoom2_Tmax, sliderRoom2_Tdes;
Slider sliderSu, sliderSu_Tmax, sliderSu_Tdes;
int timeLiv = 4, timeRoom1 = 4, timeRoom2 = 4, timeSu = 4; // час таймера, що встановлено
користувачем
int tempLivMax, tempLivDes, // максимальна і бажана температура, встановлюються
користувачем
tempSuMax, tempSuDes, //
tempRoom1Max, tempRoom1Des, //
tempRoom2Max, tempRoom2Des;
float tempSuReal = 24.5, tempRoom1Real = 25.3, tempRoom2Real = 25.0, tempLivReal = 26.1;
SimpleButtonOff buttonOffLiv, buttonOffSu, buttonOffRoom1, buttonOffRoom2;
void setup ()
{
// String portName = Serial.list () [0];
// myPort = new Serial (this, portName, 9600); // обмін даними 9600 Мбіт / с
size (1240, 640);
bg = loadImage ( "home2.jpg");
background (bg);
// можливість обмінюватися даними в реальному часі
Interactive.make (this);
liv ();
su ();
Room1 ();
Room2 ();
timer = CountdownTimerService.getNewCountdownTimer (this) .configure (1000, 36000000) .start ();
input = new float [inputLenght];
for (int i = 0; i <inputLenght; i ++)
{Input [i] = 1;}
}
void draw ()
{
String read;
int in = myPort.read ();
background (bg);
LivTemp ();
SuTemp ();
Room1Temp ();
Room2Temp ();
Out ();
Hall ();
Kitch ();
if (! timer.isRunning ()) {timer.start ();}

```

```

// tempC = myPort.analogRead (aPin) / 2;
 textSize (12);
// text (myPort.analogRead (aPin), 100,100);
 delay (10);
 }
 void onTickEvent (CountdownTimer t, long timeLeftUntilFinish)
 {
 if (buttonOffLiv.on)
 {--ElTimeLiv; }
 if (buttonOffSu.on)
 {--ElTimeSu; }
 if (buttonOffRoom1.on)
 {--ElTimeRoom1; }
 if (buttonOffRoom2.on)
 {--ElTimeRoom2; }
 updateTimeTextSu ();
 updateTimeTextLiv ();
 updateTimeTextRoom1 ();
 updateTimeTextRoom2 ();
 }
 void Hall ()
 {
 if (input [14] == 1) // якщо датчик працює
 fill (red);
 else fill (redD);
 ellipse (750,270,30,30);
 if (input [15] == 1) // якщо світло увімкнено
 fill (yellow);
 else fill (yellowD);
 ellipse (750,320,60,60);
 }
 void Kitch ()
 {
 if (input [6] == 1) // якщо датчик присутності працює
 fill (red);
 else fill (redD);
 ellipse (700,50,30,30);
 if (input [7] == 1) // якщо світло увімкнено
 fill (yellow);
 else fill (yellowD);
 ellipse (700,100,60,60);
 if (input [4] == 1) // якщо релейний датчик температури працює
 fill (0);
 else fill (grey);
 ellipse (830,50,30,30);
 if (input [5] == 1) // якщо звукова сигналізація включена чи включений
 fill (green);
 else fill (greenD);
 rect (810,100,40,80);
 }
 void liv ()
 {

```

```

lampKitch = createShape (ELLIPSE, 800, 100, 50, 50);
sliderLiv = new Slider (10, 70, 400, 10, color (# 801A98), color (# BE60D3));
Interactive.on (sliderLiv, "valueChanged", this, "Livget");
// установка максимальної температури
sliderLiv_Tmax = new Slider (10, 120, 170, 10, color (# 801A98), color (# BE60D3));
Interactive.on (sliderLiv_Tmax, "valueChanged", this, "LivgetTmax");
// установка Жеда температури
sliderLiv_Tdes = new Slider (200, 120, 170, 10, color (# 801A98), color (# BE60D3)); ;
Interactive.on (sliderLiv_Tdes, "valueChanged", this, "LivgetTdes");
// make the manager
Interactive.make (this);
// кнопка, що запускає та зупиняє таймер
buttonOffLiv = new SimpleButtonOff (20, 30, 100,30, cOn, cOff, "Вітальня", text_but_size, cc);
}
void LivTemp ()
{
textSize (20);
fill (0);
// вивід на панель часу таймера максимальної, бажаної і реальної температур
text ("Час таймера:", 150,60); text (timeTextLiv, 310,60);
text ("Мак. т:", 10,110); text (tempLivMax, 150,110);
text ("Желаемая.т:", 210,110); text (tempLivDes, 350,110);
text ("Температура:", 500,450); text (tempLivReal, 500,470);
if (buttonOffLiv.on) // кнопка натиснута
{
if (elTimeLiv == 0) // якщо час скінчився
{
buttonOffLiv.on = false; // віджати кнопку
lampKitch.setFill (color (# F5FA05)); // вимкнути лампу
}
else
{
lampKitch.setFill (color (# F5FA05)); // включити лампу
// image (bat2,1000,150);
}
}
else // кнопка не нажата
{
lampKitch.setFill (color (# 4C4D12)); // лампа не горить
if (elTimeLiv! = timeLiv) // час має дорівнювати уставці
{ElTimeLiv = timeLiv; }
}
if (input [13] == 1) // якщо радіатор включений
fill (orange);
else fill (blue);
rect (500,500,80,40);}
void Livget (float vv)
{TimeLiv = int (map (vv, 0, 100, 2, 86400));}
void LivgetTmax (float vv)
{TempLivMax = int (map (vv, 0, 100, 10, 40));}
void LivgetTdes (float vv)
{TempLivDes = int (map (vv, 0, 100, 10, 40));}

```

```

void updateTimeTextLiv ()
{
timeHoursLiv = elTimeLiv / 3600;
timeSecondsLiv = elTimeLiv% 60;
timeMinutesLiv = elTimeLiv / 60 - timeHoursLiv * 60;
timeTextLiv = nf (timeHoursLiv, 2) + ':' + nf (timeMinutesLiv, 2) + ':' + nf (timeSecondsLiv, 2);
}
void Out ()
{
if (input [16] == 1) // якщо датчик працює
fill (red);
else fill (redD);
ellipse (490,270,30,30);
if (input [17] == 1) // якщо світло увімкнено
fill (yellow);
else fill (yellowD);
ellipse (490,320,60,60);
}
void Room1 ()
{
batRoom1 = createShape (RECT, 50, 50,500, 500);
sliderRoom1 = new Slider (10, 370, 400, 10, color (# 801A98), color (# BE60D3));
Interactive.on (sliderRoom1, "valueChanged", this, "Room1get"); // читання часу з таймера
// установка максимальної температури
sliderRoom1_Tmax = new Slider (10, 420, 170, 10, color (# 801A98), color (# BE60D3));
Interactive.on (sliderRoom1_Tmax, "valueChanged", this, "Room1getTmax");
// установка желаймой температури
sliderRoom1_Tdes = new Slider (200, 420, 170, 10, color (# 801A98), color (# BE60D3)); ;
Interactive.on (sliderRoom1_Tdes, "valueChanged", this, "Room1getTdes");
// make the manager
Interactive.make (this);
// кнопка, яка запускає і зупиняє таймер
buttonOffRoom1 = new SimpleButtonOff (20, 330, 100,30, cOn, cOff, "Комната_1", text_but_size,
cc);
}
void Room1Temp ()
{
// shape (lampRoom1);
textSize (20);
fill (0);
// вивід на панель часу таймера. максимальної, бажаної і реальної температур
text ("Час таймера:", 150,360); text (timeTextRoom1,310,360);
text ("Макс. т:", 10,410); text (tempRoom1Max, 150,410);
text ("Желаемая.т:", 210,410); text (tempRoom1Des, 350,410);
text ("Температура:", 1000,130); text (tempRoom1Real, 1000,150);
if (buttonOffRoom1.on) // кнопка натиснута
{
if (elTimeRoom1 == 0) // якщо час скінчився
{
buttonOffRoom1.on = false; // віджати кнопку
batRoom1.setFill (color (# F5FA05)); // вимкнути,
}
}
}

```

```

else
{BatRoom1.setFill (color (# F5FA05)); // включити лампу}
}
else // кнопка не нажата
{
if (elTimeRoom1 != timeRoom1) // час має дорівнювати уставці
{ElTimeRoom1 = timeRoom1; }}
if (input [9] == 0) // якщо радіатор включений
fill (orange);
else fill (blue);
rect (1000,170,80,40);
}
void Room1get (float vv)
{TimeRoom1 = int (map (vv, 0, 100, 2, 86400));}
void Room1getTmax (float vv)
{TempRoom1Max = int (map (vv, 0, 100, 10, 40));}
void Room1getTdes (float vv)
{TempRoom1Des = int (map (vv, 0, 100, 10, 40));}
void updateTimeTextRoom1 ()
{
timeHoursRoom1 = elTimeRoom1 / 3600;
timeSecondsRoom1 = elTimeRoom1 % 60;
timeMinutesRoom1 = elTimeRoom1 / 60 - timeHoursRoom1 * 60;
timeTextRoom1 = nf (timeHoursRoom1, 2) + ':' + nf (timeMinutesRoom1, 2) + ':' + nf
(timeSecondsRoom1, 2);
}
void Room2 ()
{
batRoom2 = createShape (RECT, 1100, 450,50, 50);
sliderRoom2 = new Slider (10, 520, 400, 10, color (# 801A98), color (# BE60D3));
// читання часу з таймера
Interactive.on (sliderRoom2, "valueChanged", this, "Room2get");
// установка максимальної температури
sliderRoom2_Tmax = new Slider (10, 570, 170, 10, color (# 801A98), color (# BE60D3));
Interactive.on (sliderRoom2_Tmax, "valueChanged", this, "Room2getTmax");
// установка бажаної температури
sliderRoom2_Tdes = new Slider (200, 570, 170, 10, color (# 801A98), color (# BE60D3));
Interactive.on (sliderRoom2_Tdes, "valueChanged", this, "Room2getTdes");
// make the manager
Interactive.make (this);
// кнопка, яка запускає і зупиняє таймер
buttonOffRoom2 = new SimpleButtonOff (20, 480, 100,30, cOn, cOff, "Комната_2", text_but_size,
cc); }
void Room2Temp ()
{ textSize (20);
fill (0); // вивід на панель часу таймера. максимальної, бажаної і реальної температур
text ("Час таймера:", 150,510); text (timeTextRoom2,310,510);
text ("Макс. т:", 10,560); text (tempRoom2Max, 150,560);
text ("Желаемая.т:", 210,560); text (tempRoom2Des, 350,560);
text ("Температура:", 1000,400); text (tempRoom2Real, 1000,420);
if (buttonOffRoom2.on) // кнопка натиснута
{

```

```

if (elTimeRoom2 == 0) // якщо час скінчився
{
buttonOffRoom2.on = false; // віджати кнопку
batRoom2.setFill (color (# F5FA05)); // вимкнути,
}
else
{
batRoom2.setFill (color (# F5FA05)); // включити лампу
}}
else // кнопка ненажатом
{
if (elTimeRoom2! = timeRoom2) // час має дорівнювати уставці
{ElTimeRoom2 = timeRoom2; }
}
if (input [11] == 0) // якщо радіатор включений
fill (orange);
else fill (blue);
rect (1000,450,80,40);
}
void Room2get (float vv)
{TimeRoom2 = int (map (vv, 0, 100, 2, 86400));}
void Room2getTmax (float vv)
{TempRoom2Max = int (map (vv, 0, 100, 10, 40));}
void Room2getTdes (float vv)
{TempRoom2Des = int (map (vv, 0, 100, 10, 40));}
void updateTimeTextRoom2 ()
{
timeHoursRoom2 = elTimeRoom2 / 3600;
timeSecondsRoom2 = elTimeRoom2% 60;
timeMinutesRoom2 = elTimeRoom2 / 60 - timeHoursRoom2 * 60;
timeTextRoom2 = nf (timeHoursRoom2, 2) + ':' + nf (timeMinutesRoom2, 2) + ':' + nf
(timeSecondsRoom2, 2);
}
import de.bezier.guido. *;
public class Slider
{
float x, y, width, height;
float valueX = 0, value;
boolean on;
color slid, strok;
Slider (float xx, float yy, float ww, float hh, color sl, color str)
{X = xx;
y = yy;
width = ww;
height = hh;
slid = sl;
strok = str;
valueX = x;
Interactive.add (this); }
void mouseEntered ()
{On = true; }
void mouseExited ()

```

```

{On = false; }
void mouseDragged (float mx, float my)
{ ValueX = mx - height / 2;
if (valueX <x) valueX = x;
if (valueX> x + width-height) valueX = x + width-height;
value = map (valueX, x, x + width-height, 0, 100);
// подія, отправляє значення значення слайдера
Interactive.send (this, "valueChanged", value);
}
public void draw ()
{
noStroke ();
fill (stroke); // колір скролла
rect (x, y, width, height);
fill (on? 0: slid); // колір повзунка
rect (valueX, y, height, height);
}}
void su ()
{
lampSu = createShape (ELLIPSE, 100, 200, 50, 50);
sliderSu = new Slider (10, 220, 400, 10, color (# 801A98), color (# BE60D3));
Interactive.on (sliderSu, "valueChanged", this, "Suget"); // читання часу з таймера
// установка максимальної температури
sliderSu_Tmax = new Slider (10, 270, 170, 10, color (# 801A98), color (# BE60D3));
Interactive.on (sliderSu_Tmax, "valueChanged", this, "SugetTmax");
// установка бажаної температури
sliderSu_Tdes = new Slider (200, 270, 170, 10, color (# 801A98), color (# BE60D3)); ;
Interactive.on (sliderSu_Tdes, "valueChanged", this, "SugetTdes");
// make the manager
Interactive.make (this);
// кнопка, яка запускає і зупиняє таймер
buttonOffSu = new SimpleButtonOff (20, 180, 100,30, cOn, cOff, "Ванна", text_but_size, cc);
}
void SuTemp ()
{
textSize (20);
fill (0);
// вивід на панель часу таймера. максимальної, бажаної і реальної температури
text ("Час таймера:", 150,210); text (timeTextSu, 310,210);
text ("Макс. т:", 10,260); text (tempSuMax, 150,260);
text ("Желаемая.т:", 210,260); text (tempSuDes, 350,260);
text ("Температура:", 450,130); text (tempSuReal, 450,150);
if (buttonOffSu.on) // кнопка натиснута
{else
{LampSu.setFill (color (# F5FA05)); // включити лампу}
}
else // кнопка ненажатом
{LampSu.setFill (color (# 4C4D12)); // лампа не горить
if (elTimeSu != timeSu) // час має дорівнювати уставці
{ElTimeSu = timeSu; }}
if (input [2] == 1) // якщо датчик присутності працює
fill (red);
}

```

```

else fill (redD);
ellipse (550,20,30,30);
if (input [3] == 1) // якщо світло увімкнено
fill (yellow);
else fill (yellowD);
ellipse (550,70,60,60);
if (input [1] == 0) // якщо радіатор включений
fill (orange);
else fill (blue);
rect (500,170,80,40);
}
void Suget (float vv)
{TimeSu = int (map (vv, 0, 100, 2, 86400));}
void SugetTmax (float vv)
{TempSuMax = int (map (vv, 0, 100, 10, 40));}
void SugetTdes (float vv)
{TempSuDes = int (map (vv, 0, 100, 10, 40));}
void updateTimeTextSu ()
{
timeHoursSu = elTimeSu / 3600;
timeSecondsSu = elTimeSu% 60;
timeMinutesSu = elTimeSu / 60 - timeHoursSu * 60;
timeTextSu = nf (timeHoursSu, 2) + ':' + nf (timeMinutesSu, 2) + ':' + nf (timeSecondsSu, 2);
}
import com.dhchoi.CountdownTimer;
import com.dhchoi.CountdownTimerService;
import processing.core.PApplet;
public class Timer extends PApplet
{CountdownTimer timer;
int elapsedTime = 0;
String timeText = "";
final int timeTextX = 5, timeTextY = 35; // upper left corner of displayed text
int timeTextSeconds = 0, timeTextMinutes = 0; // the seconds and minutes to be displayed
color timeTextColor = color (255, 0, 0); // color of text (red: stopped, green: running)
long value;
boolean run;
Timer (long v)
{Value = v;
timer = CountdownTimerService.getNewCountdownTimer (this) .configure (1000, 1000 * v) ;;
Interactive.add (this);
}
void updateTimeText ()
{TimeTextSeconds = elapsedTime% 60;
timeTextMinutes = elapsedTime / 60;
timeText = nf (timeTextMinutes, 2) + ':' + nf (timeTextSeconds, 2);
}
void onTickEvent (CountdownTimer t, long timeLeftUntilFinish)
{++ elapsedTime;
updateTimeText ();}
}

```

Додаток 3. Матеріали презентації до проекту

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «ОДЕСЬКИЙ ТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ ОНТУ»

ДИПЛОМНИЙ ПРОЕКТ НА ТЕМУ: ПРОЕКТУВАННЯ СИСТЕМИ КЛІМАТ-КОНТРОЛЮ НА БАЗІ ПЛАТФОРМИ ARDUINO



Дипломник: Прохоров М.М.
Керівник: Скорняков В.С.

ВСТУП

Будь-який будинок - будь-то адміністративний, виробничий або житловий складається з деякого набору підсистем, що відповідають за виконання певних функцій, які вирішують різні завдання в процесі функціонування цієї будівлі.

Сучасною реалізацією такого будинку є технологія під назвою «розумний будинок». В роботі проектується система клімат-контролю на базі платформи Arduino. Загалом, системи клімат-контролю є елементами «Розумного будинку».



ЗАВДАННЯ ДЛЯ ПРОЕКТУВАННЯ:

В результаті роботи згідно з технічним завданням потрібно спроектувати систему клімат-контролю на базі платформи Arduino.

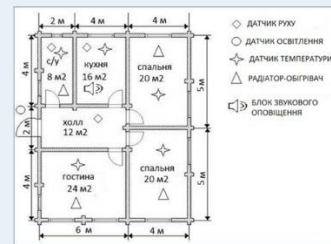
Реалізувати наступні функції системи:

- автоматичного вуличного освітлення, по спрацюванню датчика освітленості;
- автоматичного освітлення на кухні, в санвузлі, в холі по спрацюванню датчика руху;
- терморегулювання за рахунок отримання даних з датчиків температури;
- звукова сигналізація на кухні, після спрацювання порогового термодатчика при перевищенні встановленого порога температури.



В роботі будуть впроваджені наступні основні системи: **система контролю температури з можливістю забезпечувати вентиляцію повітря і додатково - керувати освітленням.**

Для початку, визначимо з об'єктом, для якого розробляється дана система. Це може бути багатокімнатна квартира, котедж, багатопверховий замський будинок, промисловий об'єкт, тощо. Нехай це буде одноповерховий будинок площею 100 квадратних метрів.



Система управління «Розумний будинок» буде включати в себе:

1. автоматичне регулювання температури повітря за допомогою:

- аналогового датчика температури - показання якого будуть моніторитися в вітальні (24 кв.м) і в двох спальнях (20 кв.м кожна)
- цифрового датчика температури - показання якого будуть моніторитися в с/у (8 кв.м.)
- електричного радіатора опалення, який буде підтримувати встановлену температуру.

2. ручне та автоматичне керування освітленням:

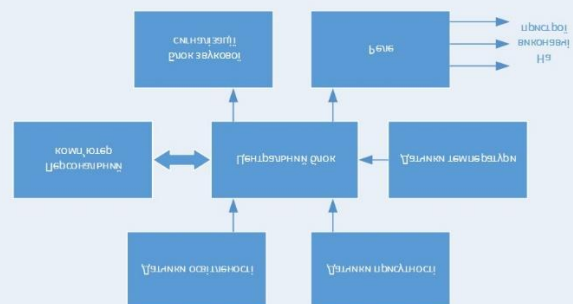
- при спрацюванні датчика руху - включення освітлення в холі (12 м²), в санвузлі (с/у) (8 м²) і на кухні (16 м²).
- при спрацюванні датчика освітленості - включення вуличного освітлення на веранді у парадних дверей.

3. звукову сигналізацію про перевищення порога температури:

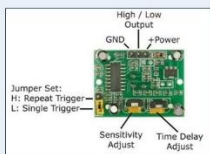
- при спрацюванні релеїного датчика температури буде відтворюватися звуковий сигнал, який вказує на перевищення заданого рівня температури на кухні (16 кв.м.)

Синхронізація: цільову взаємодію з користувачем здійснюватиме персональний комп'ютер. Персональний комп'ютер який одне безпосередньо на площі будівлі. Учасники елементів зв'язкової мережі зв'язують схему відео, що для безпечної системи управління необхідні:

СІМ'ЮЛЯЦІЙНА СХЕМА СИСТЕМИ КЛІМАТ-КОНТРОЛЮ



ВИБІР ЕЛЕМЕНТНОЇ БАЗИ



Датчик руху HC-SR501



Датчик освітлення на фоторезисторі GL5528



Електромагнітне реле фірми Finder 540



Імпульсний блок живлення з вихідною напругою 12 В фірми Jazzway

ВИБІР ЕЛЕМЕНТНОЇ БАЗИ



Зовнішній вигляд плати Arduino Micro



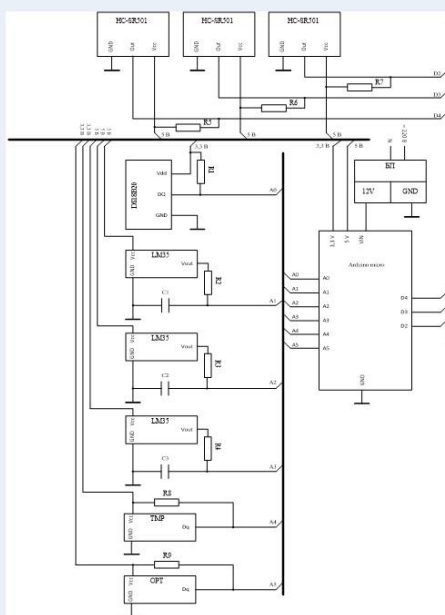
Релейний термодатчик для кухні MP DKM-046



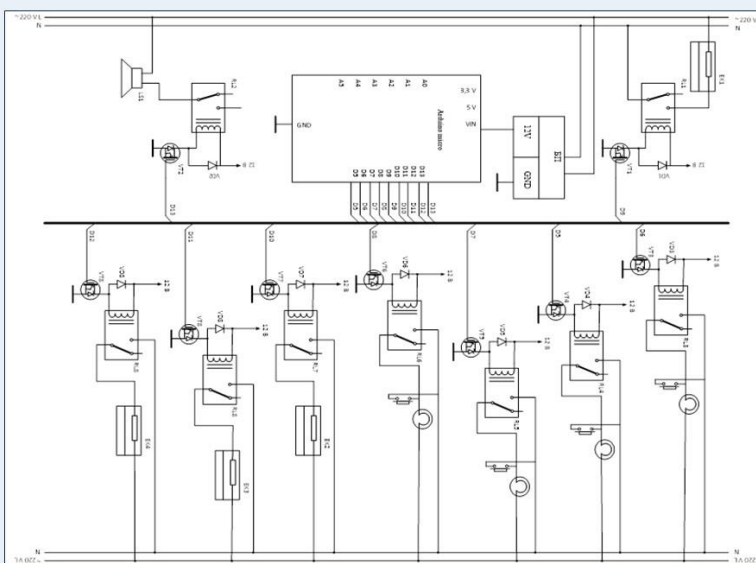
Датчик температури DS18B20



Аналоговий датчик температури



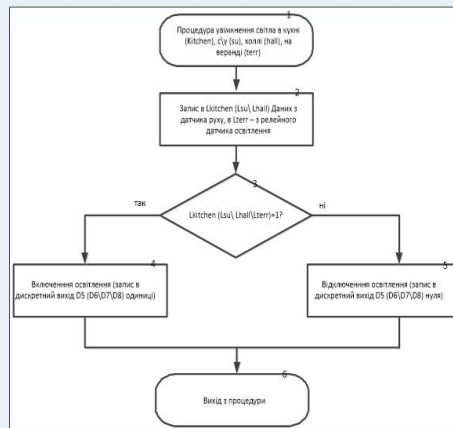
Принципова схема підключення датчиків



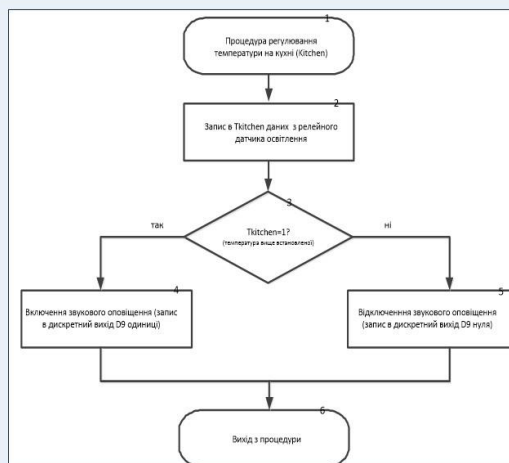
Принципова схема підключення виконавчих пристроїв



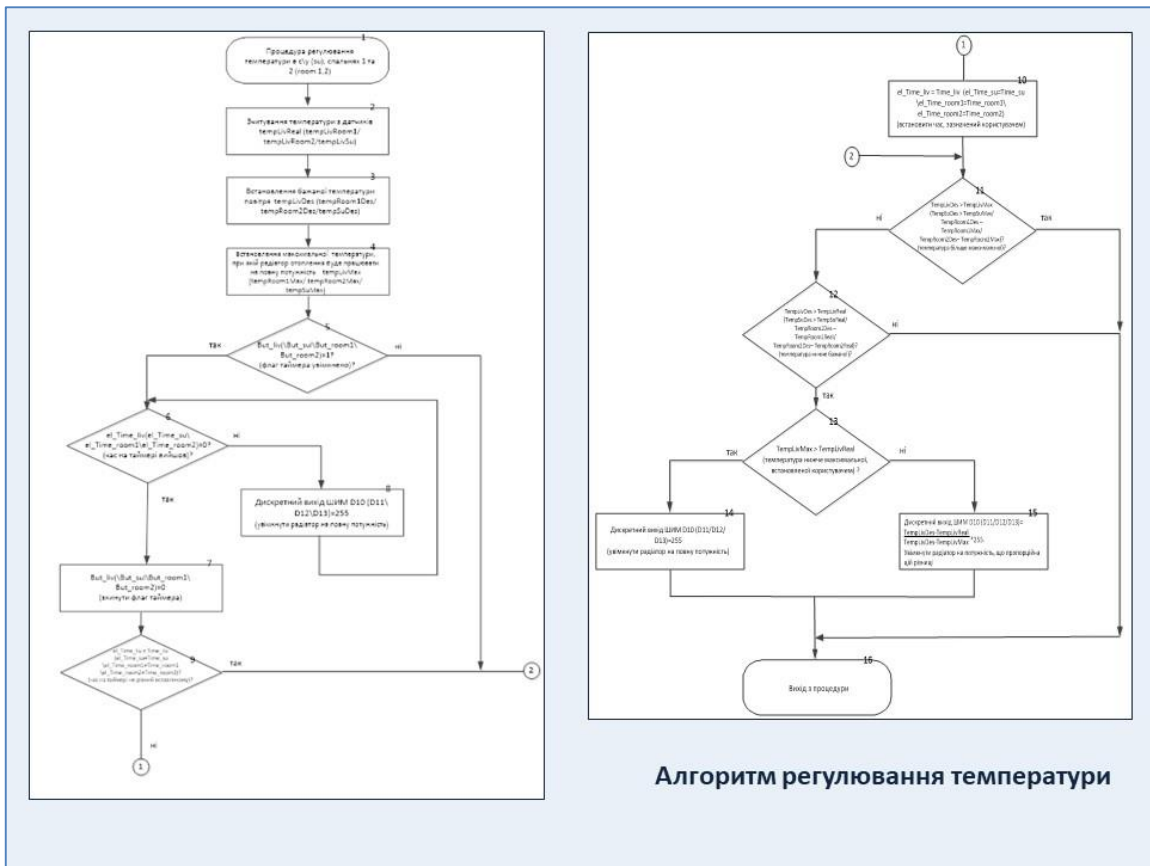
Блок-схема роботи системи клімат-контролю



Алгоритм роботи системи увімкнення освітлення



Алгоритм роботи системи оповіщення при перевищенні температури на кухні



ПЕРСПЕКТИВИ РОЗВИТКУ ІДЕЇ

Дану систему можна вдосконалити, додавши GSM-модем, що дозволяє користувачеві віддалено отримувати свідчення об'єкта управління і відправляти сигнали. Також можна додати вивід повідомлень про відмови в системі управління і оповіщення оператора про аварійні ситуації.

Є можливість доопрацювати систему на предмет відстеження показників датчиків і роботу виконавчих пристроїв через графічний інтерфейс, тим самим встановлювати параметри роботи опалювальних елементів через графічний інтерфейс (для кожної кімнати окремо), встановлювати через графічний інтерфейс час таймера безперервної роботи опалювальних елементів (для кожної кімнати окремо).

ДЯКУЮ ЗА УВАГУ

РЕЦЕНЗІЯ

на дипломний проект (роботу) здобувача (здобувачки) освіти
відділення комп'ютерних систем

Прохорова Михайла Михайловича

(прізвище, ім'я та по батькові)

Спеціальність **123 "Комп'ютерна інженерія"**

Освітня програма **Обслуговування комп'ютерних систем та мереж**

Керівник дипломного проекту (роботи) **Скорняков Вячеслав Сергійович**

(прізвище, ім'я та по батькові)

Тема дипломного проекту (роботи):

Проектування системи клімат-контролю на базі платформи Arduino

Обсяг розрахунково-пояснювальної записки _____ сторінок

Обсяг графічної (презентаційної) частини _____ аркушів (слайдів)

ХАРАКТЕРИСТИКА ДИПЛОМНОГО ПРОЕКТУ (РОБОТИ)

а) заключення про ступінь відповідності виконаного дипломного проекту (роботи) завданню

Дипломний проект повністю відповідає завданню до дипломного проектування

б) характеристика виконання кожного розділу дипломного проекту (роботи) _____

Пояснювальна записка дипломного проекту виконана у повному обсязі та відповідає поставленим завданням. Конкретизовано на основі проведеного теоретичного аналізу вимоги до дипломного проекту, визначено завдання та визначено технічні рішення, що дозволяють реалізувати завдання дипломного проекту, здійснено вибір елементної бази для побудови системи клімат-контролю

в) оцінка якості виконання пояснювальної записки та графічної частини дипломного проекту (роботи) _____

Презентаційні матеріали виконані якісно, демонстративно та відповідають вмісту теоретичного матеріалу

г) перелік позитивних якостей дипломного проекту (роботи) _____

Серед позитивних якостей проекту - детальний аналітичний огляд існуючих рішень, виважений підхід до реалізації завдань до дипломного проекту та вибору елементної бази

д) основні недоліки дипломного проекту (роботи) _____

В роботі зустрічаються відхилення від вимог щодо оформлення пояснювальної записки дипломного проекту.

Етапи розробки моделі пристрою слід було описати докладніше

Оцінка розрахункової частини _____ **4 (добре)**

Оцінка графічної частини _____ **4 (добре)**

Загальна оцінка _____ **4 (добре)**

Прізвище, ім'я, по батькові рецензента _____ **Кривченко Юрій Вікторович**

Місце роботи і посада рецензента _____

ВСП "Одеський технічний фаховий коледж ОНТУ", голова циклової комісії комп'ютерних технологій та програмної інженерії

Підпис: _____

« **16** » _____ **червня** 2023 р.

**ДОЗВІЛ
НА РОЗМІЩЕННЯ
ВИПУСКНОЇ КВАЛІФІКАЦІЙНОЇ РОБОТИ
В ЕЛЕКТРОННОМУ РЕПОЗИТАРІЇ ВСП «ОТФК ОНТУ»**

Ми, що нижче підписалися,

Прохоров Михайло Михайлович

здобувач освіти гр. 4 КС-56, та

Скорняков Вячеслав Сергійович,

керівник дипломного проекту,

не заперечуємо щодо розміщення електронного варіанту пояснювальної записки до випускної кваліфікаційної роботи молодшого спеціаліста на тему:

***«Проектування системи клімат-контролю на базі платформи Arduino»
(автор роботи – Прохоров М.М., керівник роботи – Скорняков В.С.)***

виконаного у ВСП «Одеський технічний фаховий коледж Одеського національного технологічного університету» в 2023 році, у повному обсязі в електронному репозитарії ВСП «ОТФК ОНТУ» для вільного доступу через мережу Інтернет.

Несемо відповідальність за ідентичність електронного та друкованого варіантів випускної кваліфікаційної роботи, і даємо згоду на обробку персональних даних.

Виконавець



/ Прохоров М.М./

Керівник



/ Скорняков В.С./

« 10 » червня 20 23 р.

Ім'я користувача:
Наталія Вікторівна Копусь

ID перевірки:
1015174289

Дата перевірки:
22.05.2023 11:59:18 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
22.05.2023 12:04:56 EEST

ID користувача:
100011688

Назва документа: 4КС56 - Прохоров Михайло

Кількість сторінок: 49 Кількість слів: 9057 Кількість символів: 63875 Розмір файлу: 1.59 MB ID файлу: 1014853637

23.2% Схожість

Найбільша схожість: 4.79% з Інтернет-джерелом (<https://www.bibliofond.ru/view.aspx?id=787281>)

23.2% Джерела з Інтернету

816

Сторінка 51

Не знайдено джерел з Бібліотеки

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0% Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи

10

ВСП «Одеський технічний фаховий коледж ОНТУ»

ВІДГУК

Керівника на дипломний проект здобувача освіти

Прохорова Михайла Михайловича

(прізвище, ім'я та по батькові)

Спеціальність: 123 «Комп'ютерна інженерія»

Тема дипломного проекту:

Проектування системи клімат-контролю
на базі платформи Arduino

ХАРАКТЕРИСТИКА ДИПЛОМНОГО ПРОЕКТУ

а) Обсяг і якість виконання проекту (графічного матеріалу і розрахунково-пояснювальної записки): Пояснювальна записка дипломного проекту виконана здобувачем якісно, у повному обсязі. В дипломному проекті проведено огляд існуючих рішень та аналогів системи, що проектується, проаналізовано та здійснено вибір елементної бази. Розроблено структурну та функціональну схеми системи, блок-схеми та управляючої програми. В дипломному проекті в останніх розділах проаналізовано питання економічної доцільності та охорони праці. Створено презентацію до захисту.

б) Самостійність роботи над проектом: Здобувач самостійно визначався з напрямом роботи, дослухався до рекомендацій керівника дипломного проекту, своєчасно надавав результати роботи, якісно виконував основні етапи роботи за вимогою керівника.

в) Теоретична підготовка випускника: _____

Теоретична підготовка випускника в цілому відповідає державним вимогам до фахівців відповідного рівня кваліфікації

г) Вміння розв'язувати виробничі і конструкторські питання на базі останніх досліджень науки і техніки, передових методів виробництва _____

В процесі роботи над дипломним проектом здобувач продемонстрував уміння використовувати останні досягнення науки та техніки в предметній галузі на підставі відповідної навчальної та науково-технічної літератури, впевнено користувався програмним забезпеченням при роботі над дипломним проектом та створенням презентації.

Оцінка розрахункової частини _____ *добре*

Оцінка графічної частини _____ *добре*

Загальна оцінка _____ *добре*

Прізвище, ім'я, по батькові Скорняков Вячеслав Сергійович

Місце роботи і посада керівника дипломного проекту: викладач комісії КТ та ПІ ВСП «Одеський технічний фаховий коледж Одеської національного технологічного університету»

Підпис _____ *JK*

« *12* » _____ *06* _____ 2023 р.