

ДВНЗ «Донецький національний технічний університет»
Факультет комп'ютерно- інтегрованих технологій, автоматизації,
електроінженерії та радіоелектроніки
(повне найменування інституту, назва факультету)

Автоматика та телекомунікації

(повна назва кафедри)

«До захисту допущено»
Завідувач кафедри

(підпис) (ініціали, прізвище)

“ ” 20_ р.

Випускна кваліфікаційна робота

бакалавра

(освітньо-кваліфікаційний рівень)

на тему «Розробка концепції мережі для розумної квартири з використанням технологій IoT»

Виконав: студент 4 курсу, групи ТКР-17

(шифр групи)

спеціальності

172

телекомунікації та радіотехніка

(шифр і назва напрямку підготовки, спеціальності)

Резніков Євген Сергійович

(прізвище та ініціали)

(підпис)

Керівник ст.викл.каф АТ Ступак Г.В.

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Рецензент

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

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

Студент _____

(підпис)

Покровськ – 2021 р.

ДВНЗ «Донецький національний технічний університет»
Факультет комп'ютерно- інтегрованих технологій, автоматизації,
електроінженерії та радіоелектроніки
 (повне найменування інституту, назва факультету)

Автоматика та телекомунікації

(повна назва кафедри)

Захист відбувся _____
 (дата)

з оцінкою _____
 секретар ДЕК _____
 (підпис)

Випускна кваліфікаційна робота

бакалавра

(освітньо-кваліфікаційний рівень)

на тему «Розробка концепції мережі для розумної квартири з використанням технологій IoT»

Виконав студент групи ТКР-17

(підпис, дата)

Є.С. Рєзніков

(ініціали, прізвище)

Керівник _____

(підпис, дата)

Г.В. Ступак

(ініціали, прізвище)

Зав. каф. автоматики та телекомунікацій _____

(підпис, дата)

І.С. Лактіонов

(ініціали, прізвище)

Консультанти

(підпис, дата)

(ініціали, прізвище)

(підпис, дата)

(ініціали, прізвище)

(підпис, дата)

(ініціали, прізвище)

Нормоконтролер _____

(підпис, дата)

Д.О. Жуковська

(ініціали, прізвище)

Покровськ – 2021 р.

ДВНЗ «Донецький національний технічний університет»

Факультет комп'ютерно-інтегрованих технологій, автоматизації
електроінженерії та радіоелектроніки

Кафедра автоматика та телекомунікації

Освітній ступінь бакалавр

Спеціальність 172 «Телекомунікації та радіотехніка»

ЗАТВЕРДЖУЮ:

Завідувач кафедри

/_____/

“ ”

_____ 20__ року

З А В Д А Н Н Я НА ВИПУСКНУ КВАЛІФІАЦІЙНУ РОБОТУ СТУДЕНТУ

Резнікову Євгену Сергійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи «Розробка концепції мережі для розумної квартири з
використанням технологій IoT»

керівник роботи: Ступак Гліб Володимирович, ст. викл. каф. АТ

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

затверджені наказом від “ ” року №

2. Строк подання студентом роботи 9 червня 2021р

3. Вихідні дані до роботи: схема об'єкту, матеріали курсового
проектування, звіт з практики

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

1) Аналіз предметної області

2) Дослідження основних апаратних платформ

3) Розробка рішення на базі обраного варіанту

4) Порівняння платформ, розробка веб-інтерфейсу

5) Охорона праці

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових
креслень): структурна, функціональна, схема з'єднань.

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		видав	прийняв

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз завдання	25.04.2021	
2	Аналіз об'єкту	28.04.2021	
3	Аналіз предметної області	30.04.2021	
4	Визначення ключових трендів	04.05.2021	
5	Вимоги розумних рішень	07.05.2021	
6	Аналіз існуючих підходів	10.05.2021	
7	Порівняння апаратних платформ	13.05.2021	
8	Розробка схем	18.05.2021	
9	Апаратний синтез, вибір обладнання	22.05.2021	
10	Програмно-апаратне налаштування	26.05.2021	
11	Охорона праці	28.05.2021	
12	Оформлення	31.05.2021	

Студент

Керівник проекту (роботи)

Лист зауважень

Посада П.І.Б.	Суть зауваження, оцінка та підпис

АНОТАЦІЯ

Резніков Є.С. Розробка концепції мережі для розумної квартири з використанням технологій IoT / Випускна кваліфікаційна робота на здобуття освітнього ступеня «бакалавр» за спеціальністю 172 «Телекомунікації та радіотехніка». – ДВНЗ ДонНТУ, Покровськ, 2021.

Робота містить 100 сторінок, складається з вступу, трьох розділів, висновків, переліка використаних джерел з 16 найменувань, 21 рисунків, 2 таблиці.

Зроблено огляд існуючих варіантів побудови систем «розумний будинок». На базі апаратних засобів TATU SmatrLab, Arduino Uno, Raspberry pi-3 було реалізовано три варіанти системи «розумний будинок». Провівши дослідження розроблених прототипів було визначено, що найбільш оптимальним є застосування системи побудованої на базі платформи Raspberry pi-3.

Практична значимість – розроблені прототипи можуть бути використанні як в реальному житті так і для дослідження при вирішенні освітніх задач.

Ключові слова: IP-АДРЕСА, РОЗУМНИЙ БУДИНОК, КОНТРОЛЕР, ВІДДАЛЕНЕ КЕРУВАННЯ, ARDUINO UNO, TATU SMART LAB, RASPBERRY PI-3, ПРОТОТИПУВАННЯ, МЕРЕЖА.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	9
ВСТУП	10
1 РОЗУМНИЙ БУДИНОК. ЗАГАЛЬНІ ВІДОМОСТІ	12
1.1 Історія технології «Розумний» будинок	12
1.2 Принцип роботи системи	14
1.3 Основні характеристики системи	16
1.4 Існуючі концепції	18
1.5 Переваги та недоліки існуючих рішень	20
1.6 Власний варіант реалізації	21
Висновки до розділу 1	Ошибка! Закладка не определена.
2 РЕАЛІЗАЦІЯ ПРОТОТИПУ НА ПЛАТФОРМІ TATU SMART LAB TSL TOOL BOXES	23
2.1 Опис обладнання	23
2.2 Розробка схеми підключення	28
2.3 Приклад налаштування	29
Висновки до розділу 2	Ошибка! Закладка не определена.
3 РЕАЛІЗАЦІЯ ПРОТОТИПУ НА ПЛАТФОРМІ ARDUINO	34
3.1 Опис обладнання	34
3.2 Розробка схеми підключення і програмування	37
Висновки до розділу 3	Ошибка! Закладка не определена.
4 РЕАЛІЗАЦІЯ ПРОТОТИПУ НА ПЛАТФОРМІ RASPBERRY	43
4.1 Опис обладнання	43
4.2 Розробка схеми підключення і програмування	45
4.3 Налаштування серверної частини Raspberry Pi	47
4.3.1 Встановлення FTP-сервера	47
4.3.2 Встановлення Web-сервера	48
4.3.3 Написання необхідних PHP-сценаріїв	50

4.1.3 Налаштування відео-потoku	51
4.3.4 Налаштування сервера Ubuntu	60
Висновки до розділу 4	Ошибка! Закладка не определена.
5 ПОРІВНЯННЯ РОЗРОБЛЕНИХ РІШЕНЬ	67
ВИСНОВКИ	69
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	Ошибка! Закладка не определена.
ДОДАТОК А – Охорона праці	73
А.1 Служба охорони праці та її основні задачі	73
А.2. Характеристика підприємства і стану охорони праці на ньому	74
А.3 Основні шкідливі виробничі фактори, що впливають на умови праці	75
А.4 Заходи щодо поліпшення умов праці робітників	76
А.4.1 Аналіз розміщення робочих місць	77
А.4.2 Розрахунок параметрів системи кондиціонування та вентиляції для приміщення офісу	79
А.4.3 Розрахунок системи загального рівномірного освітлення з лампами розжарювання для приміщення, в якому використовуються зорові роботи високої точності	82
А.5 Пожежна безпека	84
А.6 Безпека при надзвичайних ситуаціях на підприємстві	86
ДОДАТОК Б. Специфікація	90
ДОДАТОК А. Лістинг налаштувань платформи Arduino	91
ДОДАТОК В. Лістинг налаштувань мікро-комп'ютера Raspberry pi-3	95

ПЕРЕЛІК СКОРОЧЕНЬ

KNX	Комунікаційна шина
ОС	Операційна система
ПК	Персональний комп'ютер
БД	База даних
IP	Internet Protocol Address
MAC	Media Access Control
AWG	American Wire Gauge
FTDI	Future Technology Devices International
USB	Universal Serial Bus
SPI	Serial Peripheral Interface

ВСТУП

Світ технологій розвивається безперервно і динамічно. З кожним роком з'являються нові концепції та їх реалізації, а існуючі розширюють свої функціональні характеристики і можливості. Не нова і мета цієї спіралі прогресу - зробити умови життя людини якомога комфортнішими і продуктивними. Однією з причин такого бурхливого зростання є широке поширення Internet і значне збільшення швидкості передачі даних. Як зазначають в Google, до 2008 року був «Інтернет людей», тепер настав «Інтернет речей», так як пристроїв, підключених до світової мережі стало більше ніж жителів планети. Завдяки цьому взаємодію їх з людиною стало можливо винести на зовсім інший рівень.

Концепція «Розумний» будинок вперше з'явилася в 1961 році з появою пристрою для регулювання інтенсивності освітлення - диммера і з тих пір зазнала значних змін. Безумовно, на тому етапі називати систему «Розумним» будинком було занадто рано, проте думки більшості сходяться на тому, що саме це відкриття стало початком ери розумної домашньої електроніки.

Говорячи про технологію «Розумний» будинок та її новизну, варто зазначити скоріше не інноваційність, а перехід кількості в якість. На початку 2000х існувало більше десятка протоколів синхронізації з невеликою кількістю підтримуваних пристроїв і жорсткою прив'язкою до одного виробника, однак ситуація змінилася. У наш час практично будь-який пристрій може працювати з будь-якою платформою завдяки синхронізації через віддалені сервіси або другий варіант - він поєднує в собі функції всіх інших і просто підключається в домашню мережу.

Вже зараз багато відомих корпорацій, орієнтованих на споживчий ринок, розширюють можливості своїх систем за рахунок тісної інтеграції пристроїв і сервісів в екосистеми розумних будинків. Зацікавленість

великих корпорацій в системі «Розумний» будинок тільки підтверджує актуальність теми.

Суть концепції полягає у створенні автономної екосистеми з домашньої електроніки, що підвищує комфорт проживання і виводить його на якісно новий рівень. Вона не продиктована життєвою необхідністю, проте як і багато девайсів 21го століття економить час користувачів і підвищує продуктивність їх праці та відпочинку.

Не можна не визнати, що потенціал технології величезний і обмежується тільки відносно високою вартістю готової системи. Проте, в 2010 році була представлена перша плата для домашньої автоматизації - Arduino Mega, яка дозволяла налаштувати взаємозв'язок сенсорів з контролерами на C-образній мові скетчів без потреби глибоко вникати в саму суть процесів функціонування модулів. Саме наявність великої кількості готових бібліотек для найрізноманітнішої електроніки поряд з невисокою ціною дозволило зробити «Розумний» будинок загальнодоступним.

Однак для користувачів, які незнайомі з програмуванням і не мають хоча б базових знань по роботі з настроюваними мікроконтроллерами, розгортання такої системи так і залишилося нетривіальним завданням.

Мета роботи – побудова протиптної моделі системи «розумний» будинок для подальшого дослідження і застосування інфокомунікаційних технологій

Об'єкт дослідження – апаратні засоби для реалізації прототипу системи «розумний будинок».

Задачі проекту полягають в:

- опис загальної концепції;
- реалізації прототипу на різних платформах;
- вибір оптимальної платформи з перспективою можливості подальших досліджень.

1 РОЗУМНИЙ БУДИНОК. ЗАГАЛЬНІ ВІДОМОСТІ

Розумний будинок - це сучасний тип житлового будинку, організованого для людей, що проживають через автоматизацію та високотехнологічні об'єкти. Розумний будинок повинен розуміти систему, яка забезпечує безпеку та збереження ресурсів (включаючи комфорт) для всіх користувачів. У найпростішому випадку він повинен вміти визнати конкретні ситуації, що виникають у себе вдома, і відповідним чином реагувати: одна з систем може керувати іншою поведінкою у попередньо виконаних алгоритмах. Крім того, автоматизація декількох підсистем дає весь комплексний синергетичний ефект.

Зі збільшенням обчислювальних здібностей пристроїв, концепція "розумного" будинку отримала логічне продовження - систему онлайнових предметів, яка була проведена первинною стандартизацією та основними правилами та рекомендаціями для будівництва готової продукції як в цілому, так і окремих компонентів. Незважаючи на відносно інновації, вже існує кілька різних рішень.

1.1 Історія технології «Розумний» будинок

"Розумна історія будинку почалася в 1961 році, коли сили Джоеля та Рут винайшли та запатентовані спеціальні пристрої плавно регулюють світло - регулятор. Цей даний винахід був причиною створення глобальної компанії" Lutron Electronics Company, Inc. Ця компанія продовжувала працювати. Допомога інтелектуальних технологій паралельно, що представляє такі поняття як легка зона та сцена.

Значна подія в подальшому розвитку інтелектуальних технологій інтелектуальних будинків полягала у створенні автоматизації господарської

економіки шведської електроніки в 1975 році, коли ви вперше почали використовувати музичні гравці. Покращена домашня автоматизація американців Скотта та Россі Міллер.

Перший повноцінний проект "Smart House був невеликим житловим будинком на південному узбережжі Англії". Його автоматизація - використання широкосмугового KNX - система, яка відповідає за контрольне освітлення, тривоги, жалюзі, опалювальні та гаражні двері. Крім того, Басейн був створений у цьому будинку, який згодом доповнювався світлодіодною системою з вихідними кольорами.

У 1987 році. Організація ASHRAE розробила новий протокол автоматизації дому, який додатково покращив Беркер, Мертен, Інста, Гіра, Юнг та Сіменс та перетворився на нову модель європейської моделі автоматизації шин. У 1999 році нове покоління зовнішніх шин KNX, які все ще вважаються найкращим стандартом для систем європейської домашньої автоматизації, були створені.

Сучасні системи значно розширювали свої технічні можливості. Сьогодні вони використовують вбудовані домашні театри, всі інженерні системи об'єднуються з інтелектуальним контролем відповідно до спеціального програмного забезпечення. Через модульність системи користувачі мають можливість вибрати "розумну" функціональність.

2008-2009 рр. Створено поняття "веб-суб'єктів", яка стала логічним продовженням переходу до модульної архітектури та тісно інтегруючи будь-які підтримуються пристрої, більш ніж будь-коли. Інтернет-цілі - це не тільки різні пристрої та датчики, а також дротові та бездротові канали та підключені до Інтернету, що ближче до інтеграції реальних та віртуальних світів, де спілкування здійснюється між людьми та об'єктами.

У 2014 році компанія Apple оголосила про нову систему, яка називається HomeKit, яка призначена для інтеграції Smart Home Ecosystem з пристроями iOS та автомобілями з Apple Carplay або Google Car.

1.2 Принцип роботи системи

Термін "інтелектуальний" вдома не має чіткого визначення, тому існує система з автоматизованими пристроями, які спрощують людське життя та підвищує рівень комфорту. Через розмиті кадри було багато установок з різними рівнями інтеграції та принципу роботи. Їх можна розділити на три групи:

- Вбудовані системи з центральним контролером;
- Вбудовані системи без центрального контролера;
- Системи з нестандартною інтеграцією.

Перша група повністю налаштована та встановлена системою виробника, керованим центральним комп'ютерним пристроєм і не включає взаємодію її прямих компонентів. Всі користувацькі налаштування зберігаються на центральному пристрої (сервер), а периферія виконує лише інструкції, отримані від нього, і часто не має вбудованої пам'яті та обчислювальної продуктивності. Системний принцип показаний на рисунку 1.



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

Друга група - це система вихідних об'єктів. Алгоритми взаємодії виділяються з програми контролера безпосередньо до пам'яті кожного

пристрою та змінюють пристрій доведеться перепрограмувати. Через відсутність компонента центральної лінії між пристроями, вони встановлюються безпосередньо і є здатністю створювати автономні групи. Принцип роботи відображається на рис. 1.2.



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

Третя група є зовнішніми контролерами, що підключаються до звичайних пристроїв, і залежно від його датчиків та інтегрованих алгоритмів регулюють свою роботу. Може мати центральний контролер, але часто керується та налаштовано з Інтернету або хмарної серії. Працює в основному як незалежні модулі, а пряме з'єднання можуть вимагати додаткових датчиків / сеп. Принцип роботи відображається на рис. 1.3.



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

1.3 Основні характеристики системи

Основною метою "розумної" системи є підвищення рівня автоматизації звичайних процесів. Це, перш за все, система повинна бути проста у використанні та вимагати кількості мінімальних маніпуляцій від користувача. Особливістю Smart Home System є підключення різних пристроїв у одну систему. Координаційна робота пристроїв може бути організована просто або складною залежно від системи "OpenMea". Найбільш відкрита система також обговорюється, коли взаємодія пристроїв проходить так само легко. Для того, щоб підтримувати взаємодію між електронними пристроями, системи розумних виробників домашнього господарства дуже часто є партнерством один з одним, що дозволяє вам більш серйозно ставитися до інтеграції всіх домашніх систем: від архітектурного освітлення та газону, щоб забезпечити роботу домашнього кінотеатру. Іншим способом взаємодії - це робота технологічних стандартів. Багато виробників доставляють продукти бездротового зв'язку за технологією Z-Wave. Саме цей загальний елемент дозволяє установкам мати можливість працювати один з одним.

Чим більше постачальника партнерів буде діапазоном замовника;

- Віддалений логін. Користувачі системи швидко та легко змінюють налаштування, якщо це терміново. Це дуже часто необхідно, коли клієнт не вдома. Ось чому одна з найбільш вимогливих особливостей інтелектуальної системи є здатність дистанційного керування та доступу до системи. Це дозволяє контролювати те, що відбувається вдома та навколо будинку, змінюють освітлення, термостат та інше обладнання, використовуючи ноутбук, смартфон або планшет. Віддалений доступ також дозволяє налаштувати адаптуватися до системи без його присутності вдома, що збільшує рівень зручності та обслуговування;

- зміна масштабу. Необхідність цього параметра визначається тим, що технологія постійно розвивається, розпилюючи нове покоління товарів. У майбутньому це дозволяє додавати нові номери в системі без покупки нового. Крім того, клони дозволяють користувачеві одночасно встановлювати необхідні системні датчики та функції, не порушуючи можливість їх прикріплення. З цих та інших причин дуже важливо, щоб функції та обладнання (вертикальне розширення) або нових приміщень (горизонтальний розвиток) можна додати до "розумного" будинку. Виробники часто підтримують як типи розширення в одній мережі, такі як IP (Інтернет-протокол), а також можливість бездротових продуктів, які можуть взаємодіяти з існуючою домашньою мережею або провідними пристроями.

Наведені вище параметри є основою концепції "розумного", з повною або частковою реалізацією виробників і може залежати від особливостей об'єкта системи та регіону. Немає обмежень щодо жорстких кадрів, стандартизація відноситься лише до взаємодії протоколів між пристроями. Реалізація себе залишається на розсуд компанії та не є стандартизованою.

1.4 Існуючі концепції

В ідеальному варіанті «Розумний» будинок - це система, в якій кожен прилад інтегрований в загальну екосистему, проте зважаючи на відмінності різних протоколів зв'язку приладів, необхідності у здешевленні системи виробником та налаштування під конкретні завдання, відбувається поділ на три основні групи за призначенням:

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

Мультимедійний простір або мультимедійна система «Розумний будинок» - це цілісна екосистема з мультимедійних пристроїв із загальним хабом пам'яті для контенту і розширеними можливостями взаємодії.

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

З появою доповненої реальності концепція мультимедійного розумного будинку знайшла нове життя у вигляді так званих шоломів віртуальної реальності. До основних функцій було додано модуляцію віртуальних об'єктів на реальність і можливість взаємодіяти з результатом як з цілісним середовищем. Варто відзначити, що тепер для побудови системи потрібно всього лише один пристрій без додаткових аксесуарів і функціональність системи залежить тільки від вбудованого в шолом ПЗ.

Приклад мультимедійної системи розумного будинку з доповненою реальністю на рис. 1.4.



Рисунок 1.5 – приклад роботи Microsoft HoloLens

Система параметрів мікроклімату - це система, заснована на ідеї створення найбільш оптимальних параметрів мікроклімату з найнижчим споживанням енергії. У цьому випадку кліматичні пристрої передаються відповідно до контролю автоматизації, а також джерела світла та додаткової домашньої електроніки. Сама поняття побудована навколо комбінованої техніки в одній необов'язковій екосистемі. Фактично, система є набором керованих алгоритмів, введених на початку статусу стану. Наприклад, додаткові світильники або системи опалення запалюються з недостатньою освітленням, тоді як користувач знаходиться вдома і TD.

Змішана система - це система з попередніми функціями. Він може бути розгорнутий у повному обсязі. Він по суті побудований навколо центрального блоку управління, який може контролювати різні типи пристроїв відповідно до різних протоколів.

1.5 Переваги та недоліки існуючих рішень

"Розумний" Будинок Мейзу "- це збірник різних інтелектуальних пристроїв з різних компаній, які поєднують одне програмне забезпечення ("Lifkit ") та Meizu вимоги. Сьогодні "Інтелектуальна екосистема має такі інтелектуальні пристрої, як калькуляції Ryfit

(32 USD), лампа X-Light Plus (19 \$), розетки, очищувач повітря та так далі. Система вказує на змішаний тип. Переваги: наявність, монтаж та простота встановлених пристроїв, функції керування переймаються по телефону. Недоліки: у порівнянні з низькою функціональністю, обмеженим відбором обладнання, обов'язковим для пристроїв MEZES, всі високі ціни.

Рішення з ALLONE або ALLONE Wi-Fi Smart Remote - це центральний контролер, що доповнюють пристрої через Wi-Fi. Ціна 48 грн. Переваги: Багато модулів, масштабування. Недоліки: відносно велика ціна модуля не підходить для великих будинків з товстими стінками через пристрої управління Wi-Fi.

Рішення. Це невеликі модулі з датчиком, який з'єднує просто електроніку та контролює його на рахунку вбудованого алгоритму. Переваги: низькі витрати, масштабовані системи, не купують методи з вбудованими функціями взаємодії. Недоліки: відносно складне налаштування, невеликі інгредієнти взаємодії один з одним. Порівняння рішень показано в таблиці 1.1.

Таблиця 1.1 – Порівняння існуючих варіантів реалізації системи «Розумний»пбудинок

Параметр	Meizu	Allone	Clipsal
Вартість	Середня	Висока (разом з модулями)	Невелика за модуль, середня за систему в цілому

Установка	Проста	Проста	Потребує попереднього налаштування
Налаштування	Не потребує	Через веб-сервіс	Через прошивку
Готові модулі	Мало	Багато	Не потребує
Масштабованість	Масштабується	Масштабується, але залежить від розміру приміщення	Масштабується
Взаємодія компонентів	Через смартфон	Через сам пристрій	Через хмарні додаток
Функціонал	Базовий	Майже необмежений	Майже необмежений

1.6 Власний варіант реалізації

Власний варіант реалізації - це система розумного контролю мікрокліматичних параметрів з вбудованими функціями енергозбереження та додатковим керуванням освітлення, та відеоспостереження. Система відноситься до класу з вбудованим центральним контролером. Архітектура модульного типу. Розширюваність забезпечується за рахунок сумісних с Arduino, PROFinet/PROFibus, чи Raspberry pi-3 модулів, в залежності від контролера, однак для підхоплення нових сенсорів потрібно додати необхідне програмне забезпечення. Система складається з трьох частин: модулі управління кліматом, центральний контролер і веб-сторінка, для відображення та віддаленого керування системою. У випадку використання Arduino, або Raspberry pi-3, відмінність від інших рішень полягає у використанні тільки вільних компонентів під вільною ліцензією як у випадку з апаратним забезпеченням, так і з програмним. Абсолютно будь-який користувач може налаштувати систему під себе ,

використовуючи недорогі компоненти і малопотужний пристрій в якості центрального контролера, не написавши при цьому жодного рядка коду. Також у проекті було використано та проаналізовано нестандартне рішення щодо побудови системи розумний будинок з використанням промислового обладнання на базі TATU SmartLab мережі PROFinet.

Основна мета проекту - надати користувачам недорогого систему на безоплатній основі з розширеними можливостями в налаштуванні енергетичних планів, відеоспостереження і кастомізації.

Висновки до розділу 1

На даному етапі я розглянув деякі, найбільш популярні методи реалізації розумного будинку. Визначив основні параметри для впровадження системи в життя. Такі як низька вартість обладнання, легкість налаштування та керування системою, а також наступний функціонал:

- Відеоспостереження в режимі реального часу.
- Керування електроприладами.
- Збір даних о стані зовнішньої середовища.
- Автоматичний клімат-контроль будинку.

2 РЕАЛІЗАЦІЯ ПРОТОТИПУ НА ПЛАТФОРМІ TATU SMART LAB TSL TOOL BOXES

2.1 Опис обладнання

Обладнання TATU Smart Lab (TSL) складається із трьох окремих функціональних блоків (TSL Box 1 - PLC & PROFINET, TSL Box 2 - PROFIBUS, TSL Box 3 - Process Simulation) кожний з яких призначений для певних цілей, і може використовуватися незалежно від інших.

Обладнання TATU Smart Lab це:

- розум і гнучкість;
- портативний, навчальний набір для сучасних автоматизованих технологій.

На рисунку 2.1 представлена загальна схема підключення стендів з різними варіаціями підключення до персонального комп'ютера.

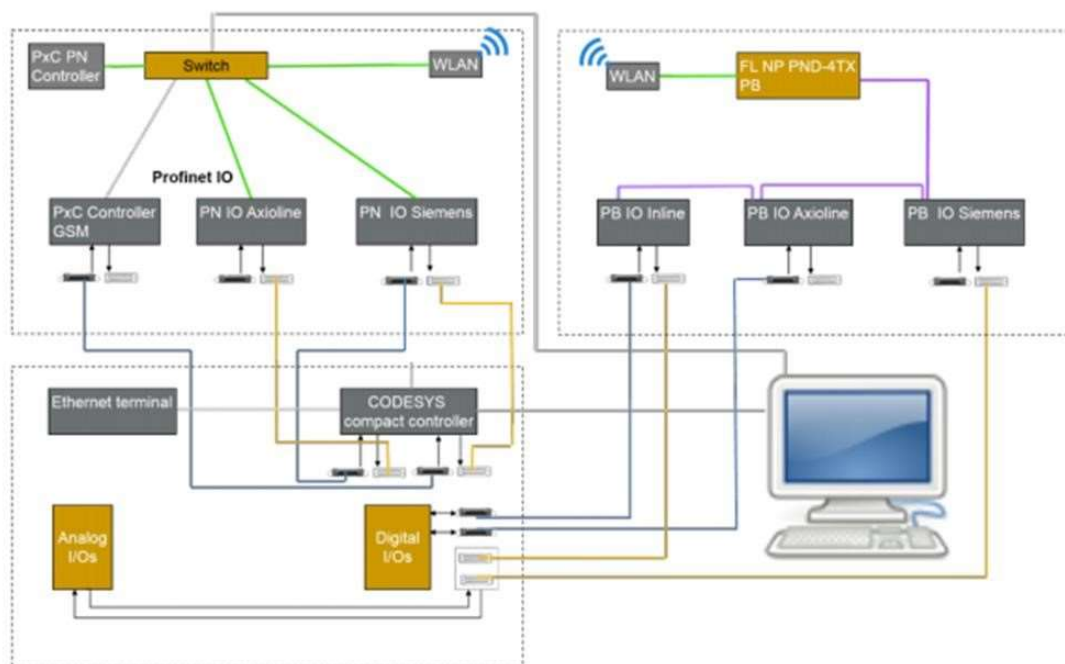


Рисунок 2.1 – Схема лабораторного обладнання

Розглянемо детальніше TSL Box 1 - PLC & PROFINET, TSL Box 2 - PROFIBUS й TSL Box 3 - Process Simulation.

Зовнішній вигляд функціонального блоку TSL Box 1 – PLC & PROFINET представлений на рисунку 2.2.



Рисунок 2.2 – Зовнішній вигляд TSL Box 1 – PLC & PROFINET

Даний блок складається з контролера з GSM/GPRS модулем (ILC 151 GSM/GPRS Inline controller), контролера (програмуємий логічний контролер АХС 3050), конвертера інтерфейсів (AXL F BK PN Axioline), комутатора (FL SWITCH SMCS 8TX-PN), а також безпроводової точки доступу. На рисунку 2.3 наведена функціонально схема блоку TSL Box 1 – PLC & PROFINET.

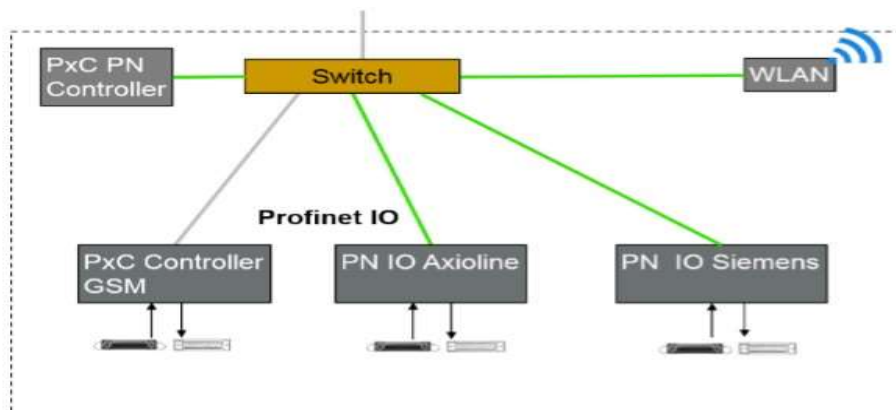


Рисунок 2.3 – Функціональна схема блоку TSL Box 1 – PLC & PROFINET

Контролери між собою можуть обмінюватися даними з використанням протоколу TCP/IP, або Modbus TCP. Пристрої PROFinet IO підключаються до АХС 3050, або ж до ILC 151 GSM/GPRS.

АХС 3050 являє собою компактний контролер з вбудованим Ethernetпортом і можливістю підключення до локальної шини Axioline F. АХС 3050 послідовно конфігурується й програмується відповідно до IEC 61131 з використанням програмного забезпечення для автоматизації PC WORX. АХС 3050 оснащений двома USB-інтерфейсами, які пропонують різні функції. АХС 3050 має внутрішню параметричну пам'ять. Ця пам'ять може використовуватися для зберігання програм і конфігурацій, які належать до проекту, наприклад, проект візуалізації. Максимальна кількість додаткових модулів Axionline підключених до контролера не повинне перевищувати шістдесяти трьох.

ILC 151 GSM/GPRS Inline controller являє собою модульний дрібносерійний контролер з вбудованими Ethernet й INTERBUS портами. Крім того, ILC 151 GSM / GPRS має вбудований GPRS-модем та GSM-модуль. Контролер конфігурується й програмується відповідно до IEC 61131 з використанням програмного забезпечення для автоматизації PC WORX. PC WORX може управлятися через мережу (Ethernet). Цей потужний процесор може бути запрограмований у всіх п'яти мовах програмування (відповідно до IEC 61131) й забезпечує швидку обробку завдань керування. Інтегровані функції зв'язку дозволяють прямий й ефективний обмін даними через Ethernet. З'єднання Ethernet забезпечує універсальні можливості для обміну даними з контролером Inline.AXL F BK PN Axioline F bus coupler for PROFINET сполучений із шиною PROFINET і зв'язує мережу із системою Axioline F. Має можливість підключати до 63 пристроїв Axioline F до існуючої системи PROFINET.

IM151-3 PN HF Siemens PROFINET buscoupler - периферійна система, що являє собою пристрій вводу-виводу на шину PROFINET IO. В представленому функціональному блоці стенду вона складається з

інтерфейсного модуля IM151-3 PN, модуля живлення, одного цифрового вхідного модуля, двох цифрових вихідних модулів.

FL SWITCH SMCS 8TX-PN – інтелектуальний керований комутатор представляє серію промислових комутаторів Ethernet. Дане устаткування працює на другому рівні моделі OSI. Конструкція пристрою використовує вентилятор, резервне джерело живлення, а також відповідність із усіма промисловим стандартам з погляду електромагнітної сумісності, кліматичних, механічних навантажень і т.д. Забезпечує максимально високий рівень доступності. Стандарти: (Rapid) Spanning Tree Protocol або MRP (протокол надмірності мережі) забезпечують безпечну роботу всієї мережі незалежно від топології, навіть у випадку переривання кабелю. Вбудований веб-сервер й агент SNMP призначені для діагностики, обслуговування й конфігурації через мережу. Дзеркалювання портів може бути використано для моніторингу трафіку даних на мережних з'єднаннях або в якості важливої сервісної функції.

Заключний пристрій на даному функціональному модулі – FL WLAN EPA, що представляє собою високопродуктивний промисловий безпроводовий інтерфейс WLAN для Ethernet або PROFINET-сумісного устаткування. Прозорий протокол використовується для передачі даних на двох рівнях, що забезпечує легку інтеграцію в промислових мережах Ethernet, таких як PROFINET або Modbus / TCP.

TSL Box 2 - PROFIBUS, або PROFIBUS Box складається з 3 PROFIBUS IO-пристроїв, PROFINET IO проксі для PROFIBUS DP з вбудованим комутатором і точкою доступу WLAN. Зовнішній вигляд TSL Box 2 – PROFIBUS представлений на рисунку 2.4.



Рисунок 2.4 – Вигляд TSL Box 2 – PROFIBUS

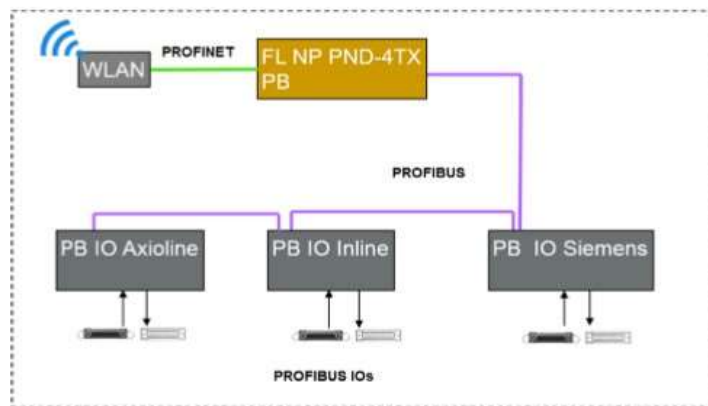


Рисунок 2.5 – Функціональна схема TSL Box 2 - PROFIBUS

TSL Box 2 може бути використаний для навчання PROFIBUS технології. Він повинен використовуватись в сполученні з TSL Box 1 або просто як PROFINET або PROFIBUS PLC.

TSL Box 3 - Process Simulation дозволяє моделювати процеси. . Зовнішній вигляд TSL Box 3 представлений на рисунку 2.5.



Рисунок 2.5 – Вигляд TSL Box 3 - Process Simulation

На рисунку 2.6 наведена функціональна схема.

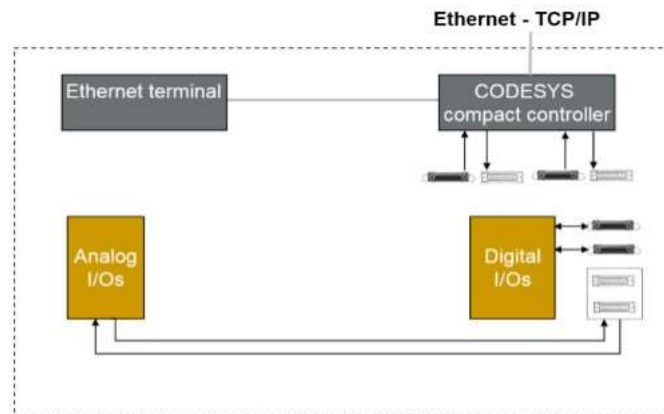


Рисунок 2.6 – Функціональна схема TSL Box 3 - Process Simulation

2.2 Розробка схеми підключення

Імітаційні моделі повинні бути розроблені в CoDeSys v.3.x і завантажені на EC2250 EtherCAT. Графічний інтерфейс можна переглядати в веб-браузері або на екрані вводу виводу даного функціонального модуля.

Первинне налаштування системи в цілому проводиться в програмному пакеті PCWorx.

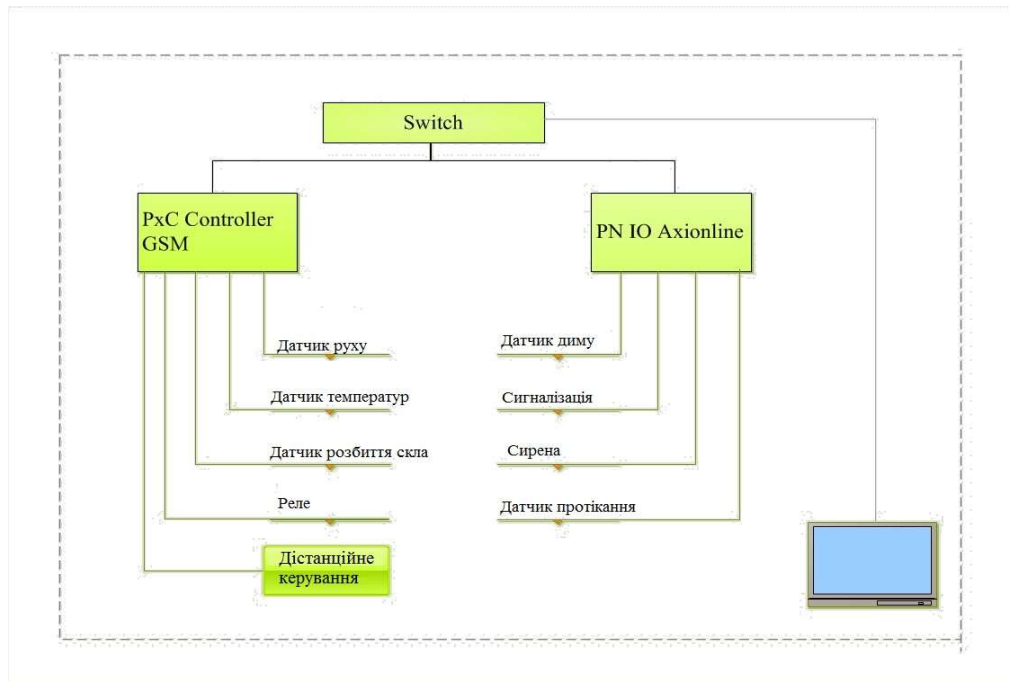


Рисунок 2.7 - Схема підключення обладнання на базі PROFinet

У дипломному проекті я використовував наступні функціональні блоки: контролер с GSM/GPRS модулем (ILC 151 GSM/GPRS Inline controller), також AXL F BK PN Axionline и комутатор (FL SWITCH SMCS 8TX-PN).

2.3 Приклад налаштування

Підключення функціональних блоків TSL Box 1:

1. Підключаємо персональний комп'ютер до АХС 3050 через usb-порт.
2. В відповідності до завдання привласнюємо IP-адресу для вашого ПК.
3. У програмному середовищі PC WORX створюємо новий проект АХС 3050.
4. У вкладці «Bus structure» встановлюємо діапазон адрес, а також привласнюємо IP-адреса для відповідних портів АХС 3050. Натискаємо «Send» і перезапускаємо контролер.

5. У вкладці шукаємо пункт меню «PROFINET device interface», вибираємо третій порт (порт AXC 3050 до якого підключена промислова мережа через комутатор FL SWITCH SMCS 8TX-PN), знову натискаємо «Send» і відключаємо usb.
6. З'єднуємо комутатор FL SWITCH SMCS 8TX-PN та Ваш персональний комп'ютер за допомогою Ethernet-кабелю.
7. Задаємо IP-адресу з діапазону, що вказаний на контролері. На верхній панелі переходимо до вкладки «Extras» і перевіряємо тип використовуваного з'єднання.
8. У вікні «Bus Structure» знаходимо profinet, натискаємо правою кнопкою й обираємо вважати з'єднання таким як profinet.
9. Вибираємо з переліку доступних пристроїв, такі, що представлені в схемі лабораторного стенду.
10. Прописуємо IP-адреси, dns-імена доданим пристроям згідно до індивідуального завдання.
11. Переходимо до вкладки "PROFINET Stationnames" і ставимо галочку зберегти IP, після чого натискаємо "Assing ip".
12. До ВК PN підключаємо модулі, натискаємо на "Axionline" правою кнопкою, вибираємо пункт прочитати шину, після чого вибираємо підключені пристрої з наданого списку.
13. Якщо ви все зробили вірно, то в схемі лабораторного стенду ви маєте три налаштовані блоки. Перевірити зв'язок з ними ви можете за допомогою команди ping, виконав її з командного рядка cmd операційної системи Windows. Також правильність проведених налаштувань можна перевірити за допомогою http запитів до вебінтерфейсів налаштованих пристроїв, заздалегіть заборонивши використання проху-серверу в вашому браузері для доступу до локальних адрес..

14. На заключному етапі аналогічним чином до зібраної моделі мережі додаємо контролер ILC 151 GSM / GPRS і підключаємо до нього модулі.

Перевірку проведених налаштувань проводимо відповідно до пункту 13.

Програмування контролерів

1. Фізично сполучаємо за допомогою фірмового кабелю TATU SmartLab функціональні модулі лабораторного стенду TSL Box 1 – PLC & PROFINET та TSL Box 3 – Process Simulation.
2. Заходимо у вкладку програмування «IEC programming Worckspace».
3. У вікні «Project Tree Window» переходимо у вкладку «Logical POUs», вибираємо останній файл «Main».
4. Натискаємо «Variable», у вікні, що відкрилося, задаємо ім'я (ONBOARD_INPUT_(BI0-15)). Як тип змінної ставимо «BOOL», VAR_GLOBAL.
5. Повторюємо крок «3», назву задаємо унікальну, в відповідності до індивідуального завдання. Приміром, «out» для наочності. З'єднуємо вхід і вихід. Компілюємо.
6. Якщо помилки відсутні, переходимо у вкладку «Process Data Workspace». У вікнах, що відкрилися, вибираємо програмувальний нами контролер, після чого вибираємо модуль, що цікавить нас, конкретний біт (або слово цілком) і переміщуємо його до назви нашого висновку (у цьому випадку в «out»). Знову компілюємо.
7. Наступний крок – зашиваємо програму. Для цього у верхніх вкладках шукаємо «Project Control Dialog», натискаємо, у вікні, що відкрилося, стоп (на порожніх контролерах це не обов'язково).
8. Після чого натискаємо «Download» і згодом переводимо систему в режим «Cold».

9. Для наочності включаємо режим «Debug» і перевіряємо роботу системи в цілому.
10. Якщо все зроблено вірно, то при натисканні відповідної кнопки на панелі призведе до відображення логічної одиниці, чи нуля на вході/виході у схемі, а також до загорання відповідної кнопки на панелі управління.
11. також до загорання відповідної кнопки на панелі управління.

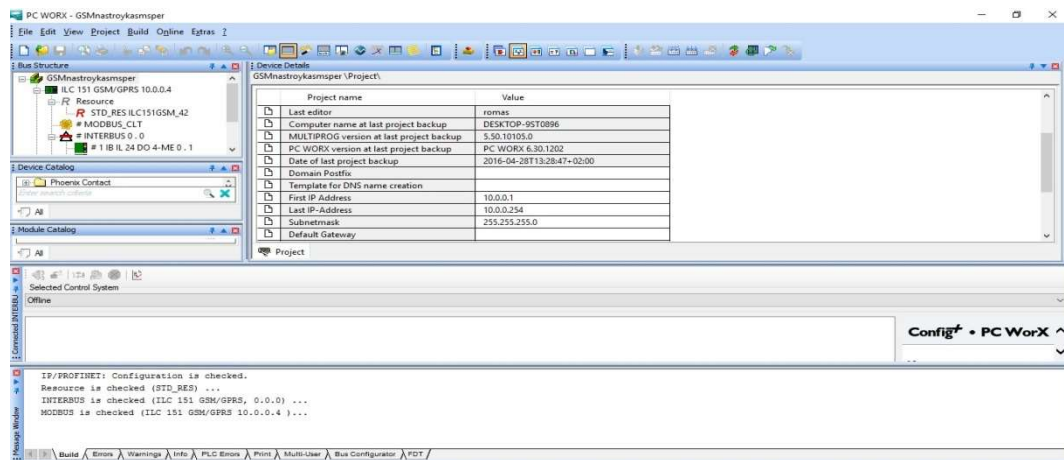


Рисунок 2.8 - Налаштування ILC 151 GSM/GPRS

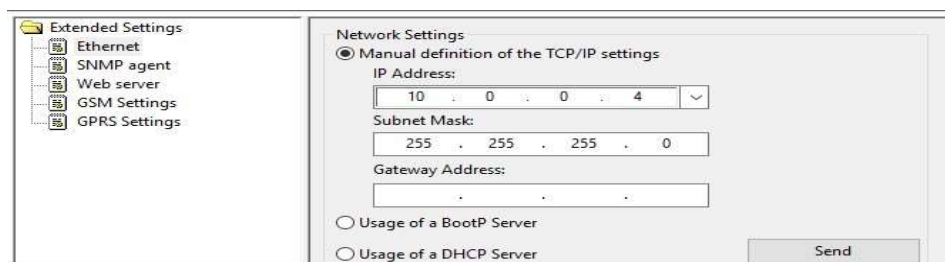


Рисунок 2.9 - Задавання ip-адрес

Висновки до розділу 2

Даний підхід дозволяю створити досить надійну систему розумного будинку, потужності процесорів достатньо, щоб організувати відео

спостереження, роботу безліч датчиків та реле. Але у такого рішення є три серйозні мінуси:

- Складність налаштування системи та складність підключення нового обладнання.
- Обмеженість варіантів керування. Спілкування з системою може здійснюватися з використанням GSM/GPRS-модуля за смс, або email.
- Вартість обладнання. В межах розумного будинку використання потужного промислового контролера не має сенсу, бо є більш дешевші та прості в використанні контролери, призначені саме для таких задач.

3 РЕАЛІЗАЦІЯ ПРОТОТИПУ НА ПЛАТФОРМІ ARDUINO

3.1 Опис обладнання

Arduino Uno - це мікроконтролер ATMEGA328 (аркуш даних). У своєму складі необхідно для комфортного контролера Micro: 14 цифрових входів / виходів (з яких можна використовувати як PWM-OUT), 6 аналогових входів, 16 МГц кварцовий резонатор, піднятий "ЕМ USB, троянди ЕМ, троянда Кнопка програмування (ICSP) та кнопка відновлення. Щоб почати працювати з пристроєм, легко потужність від адаптера AC / DC або акумулятора або підключення до комп'ютера за допомогою USB-кабелю.

На відміну від усіх попередніх таблиць Arduino, UNO, як USB Converter використовує Milka16U2 мікроконтролер (Memaga8U2 до версії R2) замість чіпів FTDI. «Arduino Board» R2 »Версія обробки мікропрограми було додано, що затримка HWB HWB 8U2 мікроконтролер HWB.

По-перше - IOREF - дозволяє збільшити плату за регулювання відповідно до напруги робочої напруги Arduino. Цей висновок очікується, що 5 V-Arduino розширення панельного розширення на основі мікроконтролерів AVR та 3, 3 рівнів Arduino. Друге виробництво не пов'язане та зарезервоване для майбутніх цілей.

Характеристика Microcontroller ATMEGA328

- 5 В напруги роботи
- потужність (рекомендовано) 7-12 В
- напруга живлення (максимальна) 6-20 В
- цифрові / виходи 14 (з них 6 можуть бути використані як PWM-OUT)
- аналогові входи 6.
- максимальний одноразовий вхідний струм 40 мА
- максимальний вихідний струм 3, 3 В 50 мА
- Flash-РАМ 32 КВ (МЕМА328), з якого було зібрано 0, 5 КБ

- SRAM 2 KB (MEMA328)
- EEPROM 1 KB (MEMA328)
- 16 МГц частота

Arduino UNO може потужність від USB або поза зовнішнім харчуванням - тип джерела автоматично вибирається.

В якості зовнішньої їжі (не USB) може бути використана на мережах акумулятора акумулятора або акумуляторів акумуляторів. Адаптер плагіна повинен бути вставлений у відповідну "дієту на диску. На акумуляторі / батареї його дроти повинні бути" Unite GND та VIN POWER.

Напруга зовнішнього джерела живлення може становити від 6 до 20 В. Однак, зменшення напруги живлення нижче 7В обумовлена зменшеною напругою 5 В, що може призвести до нестабільного пристрою. Напруга для використання над 12 В може перегріти стабілізатор напруги та відмови. На цьому тлі рекомендується використовувати напругу від 7 до 12В. Нижче наводяться на борту:

- Вхідна напруга в Arduino безпосередньо з зовнішньої їжі (не пов'язана з 5 В USB або іншою стабілізованою напругою). Як результат, ви можете використовувати зовнішню потужність та споживати електрику, коли пристрій працює на зовнішньому адаптері.

- 5 В. Вихід отримує стабілізатор напруги напруги 5 В, незалежно від того, як джерело живлення: від адаптера (7 - 12В) від USB (5b) або VIN (7 - 12V). Не рекомендується збирати 5 В або 3 В3, як у цьому випадку, стабілізатор напруги не використовується, що може спричинити невдачу.

- 3v3. 3.3b, отриманий від напруги столу стабілізатора. Найвищий струм, спожитий з цієї продукції, становить 50 мА.

- GND. Виходи Землі.

- Цей випуск надає інформацію про плату за розробку на мікроконтролерах Arduino. Залежно від напруги, збільшується з IORF, розширення картки може перейти до відповідних перетворювачів джерела живлення або використовувати, що дозволяє 5V та 3.3 V пристрою.

Обсяг "atmega328" - 32 Кб. Мікроконтроллер також має 2 Кб SRAM та 1 Кб EEPROM (з якого ви можете читати або зберегти бібліотеку EEPROM).

Використання PINMODE (), Digitalwrite () та DigitalRead () Кожен з 14 цифрових виходів може діяти як запис або вихід. Рівень напруги обмежений до 5 В. Найбільший струм, який може забезпечити або споживати одну вихідну 40 мА. Всі підсвічування з внутрішніми резисторами з номінальною вартістю 20-50 кІм. Крім того, деякі аргументи «Arduino» можуть спричинити додаткові функції:

- Серійний інтерфейс: видалення 0 (RX) та 1 (TX). Використовується для отримання послідовного інтерфейсу (RX) та передачі даних (TX). Ці результати пов'язані з відповідними мікроорганізаторами ACEGA8U2, які виконують роль USBOANAAAAARAAR Converter.

- Зовнішні перерви: 2 та 3 виходи.

- PHIM: виходи 3, 5, 6, 9, 10 та 11. З аналоговими ознаками (функціями) 8-бітними аналоговими значеннями відображаються у вигляді ShimenSignal.

- інтерфейс SPI: Exodus 10 (SS), 11 (Max), 12 (MIS), 13 (SCK). Використовуючи бібліотеку SPI, виходи можуть бути розгорнуті за допомогою інтерфейсу SPI.

- Світлодіод: 13. Вбудований світлодіод, підключений до виходу 13. Надсилання високої світлодіодної вартості, коли надсилається низька - виключена.

Arduino uno 6 аналогових входів (A0 - A5) Кожен може уявити аналогову напругу як кількість 10 біт (1024 різних значень). Вимірювання напруги заснована від 0 до 5 В. Однак, верхня межа цього діапазону може бути змінена за допомогою Alagerference () ASFF. Крім того, деякі аналогові входи мають додаткові функції: Twi: виводячи до A4 або SDA та вимкніть A5 або SCL. За допомогою бібліотеки вихідної проводки ви можете встановити інтерфейс TwiWi.

Є кілька результатів на борту:

- ISF. Підкладка напруги для аналогових входів. Можна використовувати функцію аналізу ().

- відновлення. Розвиток низької (низької) у цьому висновнику буде відновлено. Зазвичай ефективність використовується для встановлення кнопки скидання для розширення картки.

"Arduino Uno пропонує багато можливостей для реалізації" Оцінка комп'ютера, інша арка або інші мікроконтролери. Мемга328 - це приймач Petter, який дозволяє зробити послідовне з'єднання з використанням цифрових результатів 0 (Rx) та 1 (TX). Мікроконтроллер АМТЕ16U2 на диску є У цьому приймачі з комп'ютером USB-порту та підключення до комп'ютера. Дозволяє визначити Arduino як віртуальний портовий порт з зовнішніх драйверів. Для платформи Windows потрібен лише правильний файл .inf. . Це дозволяє читати прості дані "Arduino". USB- з'єдність з вашим комп'ютером, світлодіоди RX та TX-діоди блимають.

Бібліотека програмного забезпечення дозволяє серію підключатися до будь-якої цифрової продукції.

MICROCONTROLLER ME328 також встановлюється послідовно підтримувати I2C (TWI) та інтерфейси SPI. Програмне забезпечення Arduino - це бібліотека дроту, яка дозволяє спростити шину I2C. Використовуйте бібліотеку SPI, щоб працювати з інтерфейсом SPI.

3.2 Розробка схеми підключення і програмування

Arduino Uno програмується за допомогою програмного забезпечення Ардуіно. Для цього з меню Tools> Board необхідно вибрати "Arduino Uno" з мікро контролером, відповідним вашій платі.

АТmega328 в Arduino Uno випускається з прошитим завантажувачем, що дозволяє завантажувати в мікроконтролер нові програми без

необхідності використання зовнішнього програматора. Взаємодія з ним здійснюється за оригінальним протоколу STK500.

Проте, мікроконтролер можна прошити і через роз'єм для внутрисхемного програмування ICSP (In-Circuit Serial Programming), не звертаючи уваги на завантажувач.

Вихідний код прошивки мікроконтролера ATmega16U2 (або 8U2 на платах версії R1 і R2) знаходиться у вільному доступі. Прошивка ATmega16U2 / 8U2 включає в себе DFU-завантажувач (Device Firmware Update), що дозволяє оновлювати прошивку мікроконтролера. Для активації режиму DFU необхідно замкнути перемикач на звороті плати (біля зображення Італії), після чого скинути 8U2.

Після переходу в DFU-режим для завантаження нової прошивки можна використовувати програмне забезпечення Atmel's FLIP (для Windows) або DFU programmer (для Mac OS X і Linux). Альтернативний варіант - прошити мікроконтролер через роз'єм для внутрисхемного програмування ISP за допомогою зовнішнього програматора, проте в цьому випадку DFUзавантажувач затреться.

Мова програмування Arduino є стандартним C ++ (використовується компілятор AVR-GCC) з деякими особливостями, які полегшують новачкам написання першої працюючої програми.

- програми зберігаються в файлах з розширенням .ino. Ці файли перед компіляцією обробляються препроцесором Ардуіно. Також існує можливість створювати і підключати до проекту стандартні файли C++;
- обов'язкову в C ++ функцію main () препроцесор Arduino створює сам, вставляючи туди необхідні «чорнові» дії;
- програміст повинен написати дві обов'язкові для Arduino функції setup () і loop (). Перша викликається одноразово при старті, друга виконується в нескінченному циклі;

- у текст своєї програми (скетчу) програміст не зобов'язаний вставляти заголовки використовуваних стандартних бібліотек. Ці заголовки додасть препроцесор Arduino відповідно до конфігурації проекту. Однак призначені для користувача бібліотеки потрібно вказувати;
- менеджер проекту Arduino IDE має нестандартний механізм додавання бібліотек. Бібліотеки у вигляді вихідних текстів на стандартному C ++ додаються в спеціальну папку в робочому каталозі IDE. При цьому назва бібліотеки буде додано до списку бібліотек в меню IDE. Програміст зазначає потрібні бібліотеки і вони вносяться до списку компіляції;
- Arduino IDE не пропонує ніяких налаштувань компілятора і мінімізує інші настройки, що спрощує початок роботи для новачків і зменшує ризик виникнення проблем.

Найпростіша Arduino-програма складається з двох функцій:

- `setup ()`: функція викликається одноразово при старті мікроконтролера.
- `loop ()`: функція викликається після `setup ()` в нескінченному циклі весь час роботи мікроконтролера.

На рисунку 3.1 наведена принципова схема прототипу моделі розумного будинку на базі апаратних засобів Arduino.

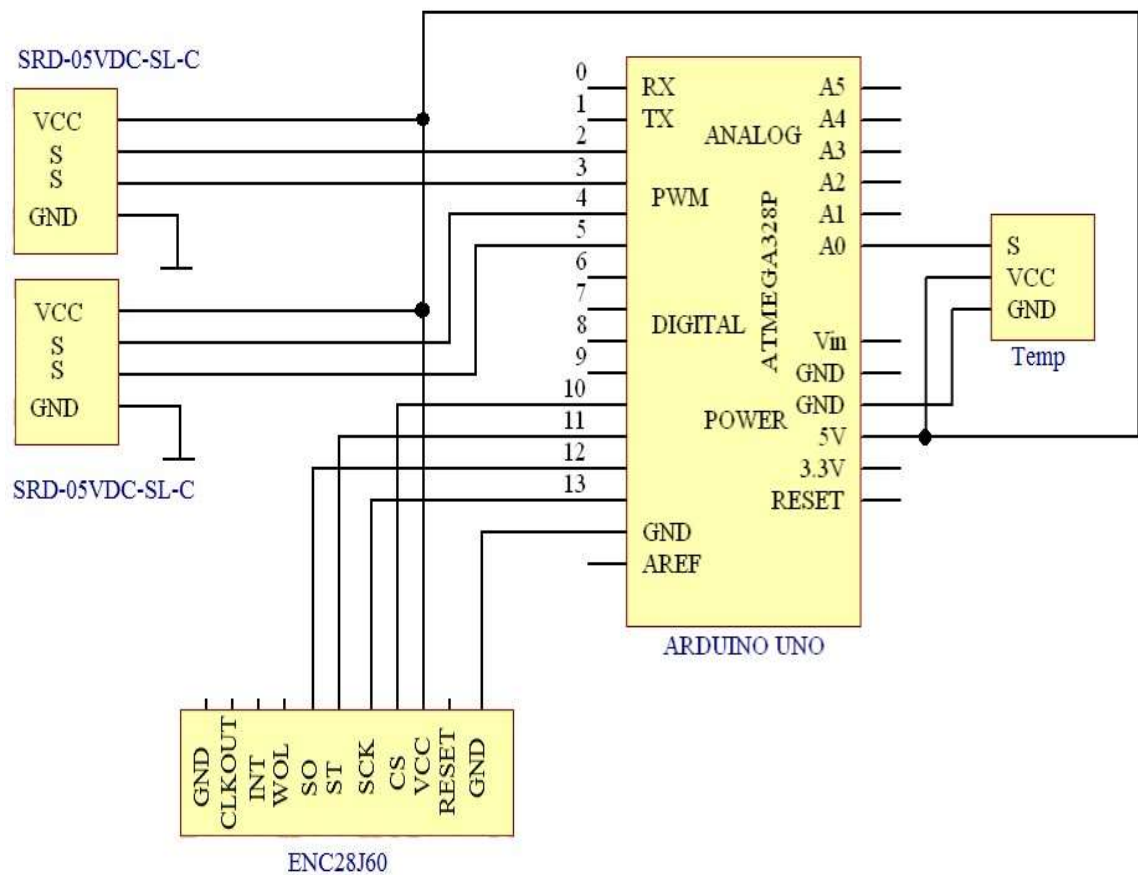


Рисунок 3.1 – Принципова схема «розумного» будинку на базі Arduino

Двоканальні реле і езернет-шилд підключаються до цифрових загальних виводів плати, та джерела живлення 5В. Датчик температури підключен до джерела живлення 5В та аналогового входу А0, через який відбувається передача рівня напруги з датчика, котрий оброблюється внутрішніми алгоритмами Arduino та перетворюється у цифрове значення температури за градусом Цельсія.

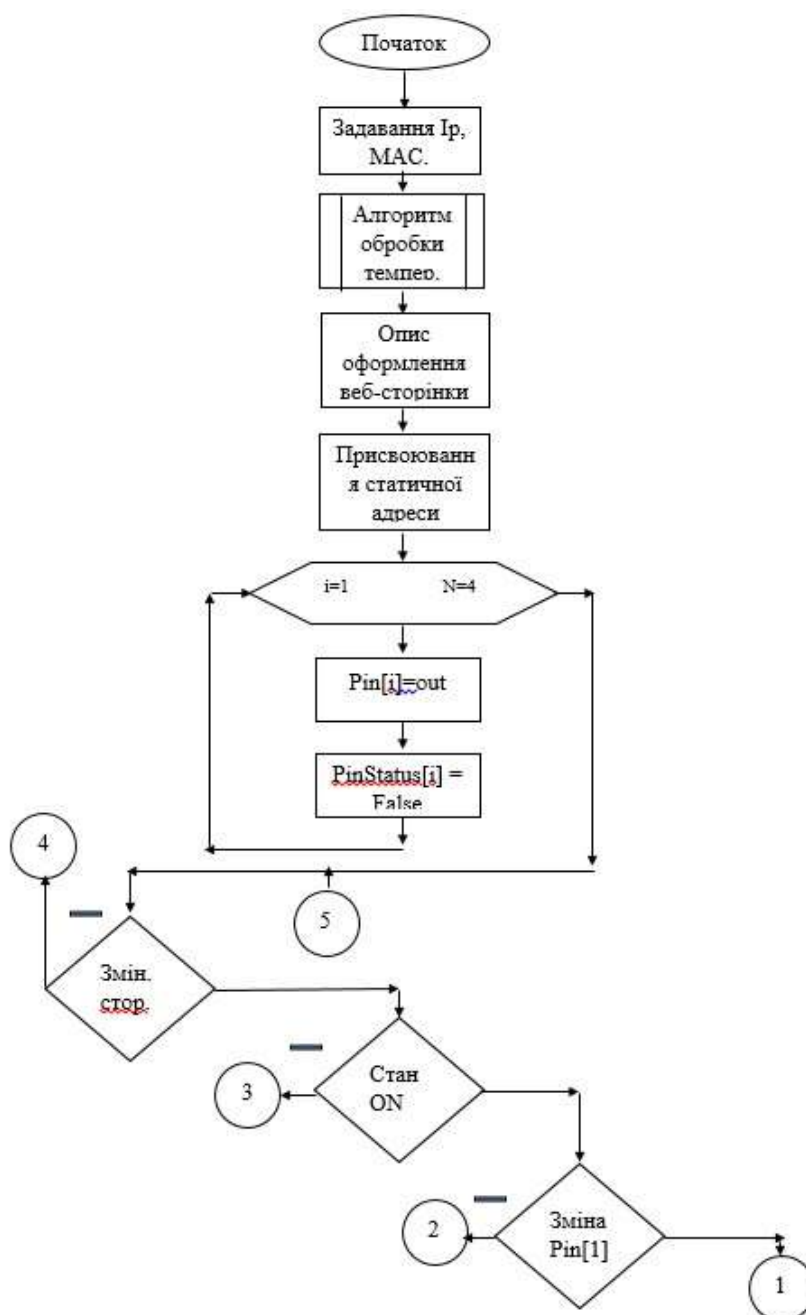


Рисунок 3.2 – Блок-схема алгоритму програми

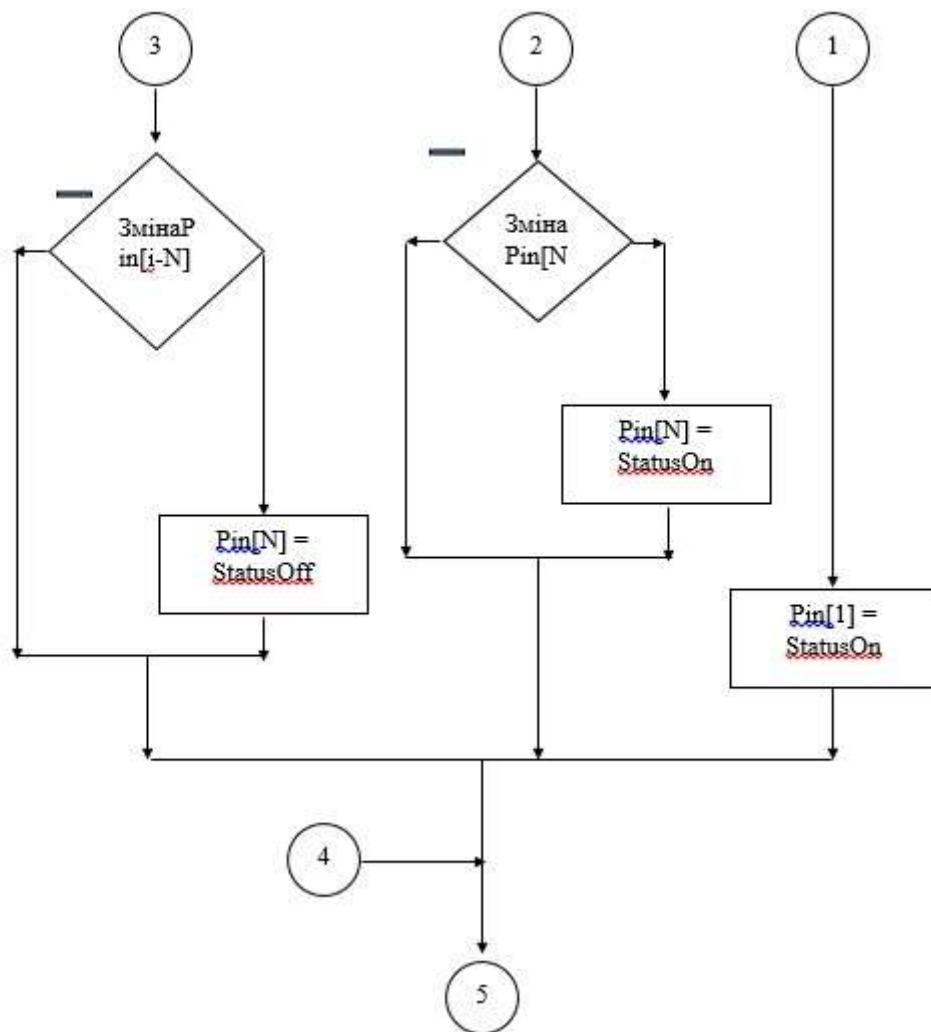


Рисунок 3.3 – Продовження блок-схеми

Програмова реалізація алгоритму знаходиться у додатку Б.

Висновки до розділу 3

Використання плати Arduino в якості головного контролера дозволило створити систему керування реле та збору даних з датчика температури через веб-інтерфейс. Головним недоліком такої системи є мала ресурсоемність контролера, яка не дозволяє працювати з потоковим відео, та обмежує функціонал використання веб-інтерфейсу.

4 РЕАЛІЗАЦІЯ ПРОТОТИПУ НА ПЛАТФОРМІ RASPBERRY

4.1 Опис обладнання

Raspberry Pi - найпопулярніший в світі одноплатний комп'ютер розміром з кредитну картку. Сфера застосування Raspberry Pi дуже широка - мультимедійні системи, управління роботами, системи "розумного будинку", навчання дітей азам програмування, робота з простими документами і вихід в інтернет, клієнт віддаленого робочого столу. Для початку роботи потрібно лише вставити MicroSD картку з прошивкою і подати харчування на роз'єм MicroUSB. Для Raspberry Pi є широкий вибір операційних систем. В основному це різні реалізації Linux. Найбільш популярні - OpenELEC / LibreELEC,

Raspbian і WTware. Незважаючи на мініатюрні розміри «малинки», вона має цілком пристойними характеристиками. З часу початку виробництва Raspberry Pi в лютому 2012 р змінилося вже декілька поколінь і модифікацій. На сьогодні актуальні дві версії - Raspberry Pi 2 Model B і Raspberry Pi 3 Model B .

Розміри обох версій однакові, корпусу підходять до обох. Для підключення до телевізора є роз'єм HDMI версії 1.4, а також аналоговий відеоаудіо вихід 3.5мм (розташування контактів нестандартно!). Можлива робота Raspberry Pi з моніторами через перехідник HDMI-DVI і настройка потрібного дозволу екрану. Відеоядро Broadcom VideoCore IV забезпечує роботу з дозволом до 1080p60 і має можливість апаратно декодувати відеокодеки H.264, MPEG-2 і VC1 (для апаратного декодування MPEG-2 і VC1 необхідно придбання ліцензії).

Таблиця 4.1 - Технічні характеристики різноманітних контролерів

Raspberry Pi

Технічні характеристики	Raspberry Pi 3 Model B	Raspberry Pi 2 Model B	Raspberry Pi Model B+	Raspberry Pi Model B
Процесор	Broadcom 2837 quad-core ARM Cortex-A53 64bit (1,2GHz)	Broadcom BCM2836 quadcore ARMv7 (900MHz)	Broadcom BCM2835, ARMv6 (700MHz)	Broadcom BCM2835, ARMv6 (700MHz)
Оперативна пам'ять	1Gb	1Gb	512Mb	512Mb
Відеовихід	HDMI	HDMI	HDMI	HDMI
A/V вихід	A/V выход 3.5мм jack 4 pin	A/V выход 3.5мм jack 4 pin	A/V выход 3.5мм jack 4 pin	Аудиовыход 3,5мм stereo jack; Композитный видеовыход RCA
USB порти	USB 2.0 x 4	USB 2.0 x 4	USB 2.0 x 4	USB 2.0 x 2
Мережа	WiFi 802.11n, 10/100Mb RJ45 Ethernet	10/100Mb RJ45 Ethernet	10/100Mb RJ45 Ethernet	10/100Mb RJ45 Ethernet
Bluetooth	Bluetooth 4.1, Bluetooth Low Energy	--	--	--
Слот пам'яті	Micro SD	Micro SD	Micro SD	SD
GPIO	40	40	40	26

Особливістю відеоядра є доступність для нього повною документації, що дозволило створити широкий спектр додатків. На платі Raspberry Pi є CSI роз'єм для підключення камери, DSI - для підключення дисплея, а також 40-контактний роз'єм GPIO для підключення великої кількості зовнішніх пристроїв. 4 USB роз'єму дозволяють використовувати різні периферійні пристрої - USB накопичувачі, клавіатури, миші, безпроводні пульти, Wi-Fi адаптери, DVB-пристрої (наприклад, MyGica T230 і TBS

5980). При використанні пристроїв, які споживають значний струм, необхідно використання USB-хаба з додатковим харчуванням. Увага! Raspberry Pi споживає струм до 600-800 мА і чутливий до напруги живлення 5В. Якщо напруга нижче норми (близько 4,7 В) - «малинка» сигналізує про це кольоровим квадратом у правому верхньому кутку екрану. При появі такої індикації потрібно вимкнути пристрій і знайти більш якісний кабель. На жаль, більшість продаються кабелів USB-MicroUSB мають недостатню товщину жил, що призводить до падіння напруги нижче допустимого. Щоб уникнути проблем з функціонуванням, використовуйте тільки якісні кабелі з товщиною жил 20-22

AWG. Для невеликої довжини кабелю допустимо 24 AWG .

4.2 Розробка схеми підключення і програмування

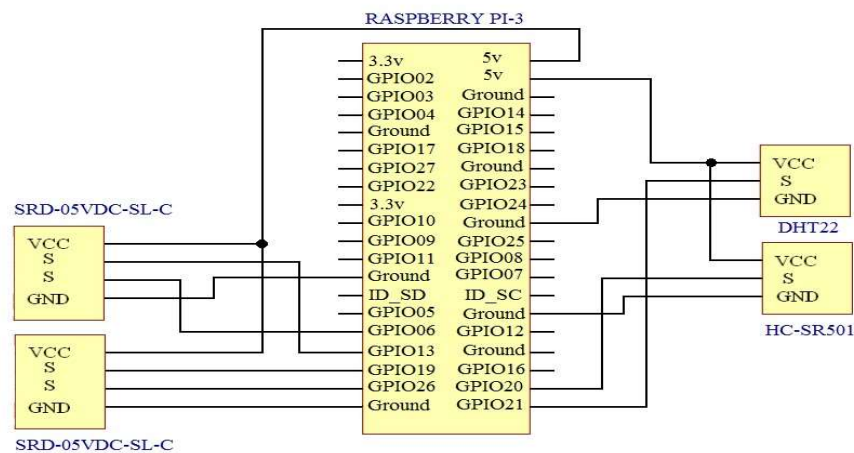


Рисунок 4.1 Принципова схема комутації елементів до Raspberry pi-3

Усі елементи живляться від п'яти вольт. Інформаційні виходи пристроїв підключенні до загальних виводів, режим роботи кожного виводу описується програмно.

Далі приведені блок-схеми, які відображають принцип роботи датчика руху та принцип керування реле. Датчик температури не

відображен у наступних алгоритмах тому, що логіка його роботи прописана у програмному коді, що знаходиться у бібліотеці даного модуля. Програмована реалізація блоксхем знаходиться у додатку В.

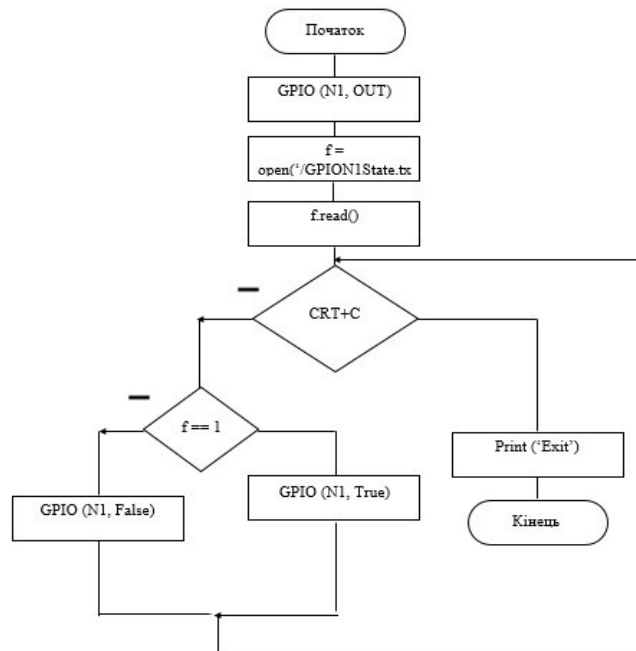


Рисунок 4.2 Блок-схема алгоритму роботи реле

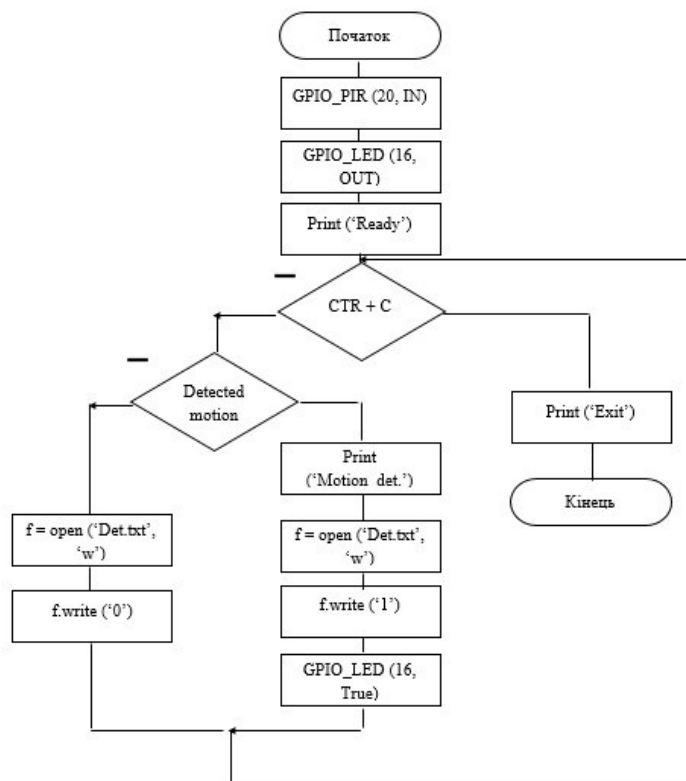


Рисунок 4.3 Блок-схема алгоритму роботи датчика руху

4.3 Налаштування серверної частини Raspberry Pi

4.3.1 Встановлення FTP-сервера

FTP (File Transfer Protocol - протокол передачі файлів) - стандартний протокол, призначений для передачі файлів через TCP-мережі (наприклад, через Інтернет). FTP часто використовується для завантаження веб-сторінок та інших документів з персонального комп'ютера розробника на відкриті сервера хостингу.

Протокол побудований на архітектурі «клієнт-сервер» і використовує різні мережеві з'єднання для передачі команд і даних між клієнтом та сервером. Користувачі FTP можуть пройти аутентифікацію, передаючи логін і пароль відкритим текстом, або ж, якщо це дозволено на сервері, можуть підключитися анонімно. Можна використовувати протокол SSH для безпечної передачі, що приховує (шифрує) логін і пароль, а також шифрувати вміст.

Для встановлення сервера на Raspberry відкриваємо термінал і вводимо

```
sudo bash //Запуск вдосконаленої командної оболонки з правами
root
// Встановлюємо vsftpd:
apt-get install vsftpd де vsftpd (Very Secure FTP Daemon) -
служба захищеного FTP-сервера з підтримкою IPv6 і SSL.
```

FTP-сервер є за замовчуванням в багатьох операційних системах і обслуговує офіційні репозиторії ftp.debian.org, ftp.redhat.com, ftp.openbsd.org, ftp.freebsd.org. Також використовується на офіційному FTP ядра Linux.

Редагуємо конфігурацію:

```
nano /etc/vsftpd.conf
```

Редагуємо до такого стану:

```
listen=NO # Встановлюємо запуск через
inetd anonymous_enable=NO # Забороняємо анонімне
```

```

підключення local_enable=YES # Дозволяємо авторизацію
користувачам системи write_enable=YES # Дозволяємо
користувачам запис в своїх каталогах
    local_umask=022 #
    ascii_upload_enable=YES # Тип даних ASCII для
завантаження на сервер ascii_download_enable=YES # Тип
даних ASCII для завантаження з сервера

```

Зберігаємо файл CTRL+O і виходимо через CTRL+X.

Перезапускаємо vsftpd

```
/etc/init.d/vsftpd restart
```

Це основні (необхідні для коректного встановлення) налаштування ftrсервера. При необхідності можна розширити їх додатковими функціями. Але вже на цьому етапі сервером можна користуватися в рамках дозволених прав користувачів.

4.3.2 Встановлення Web-сервера

Веб-сервер (англійський сервер) - це сервер, який підтверджує запит HTTP-клієнта, дає їм відповідь HTTP, як правило, з HTML, переглядом, файлом, потоком середовища або інших даних. У нашій системі потрібно керувати потоками відео для моніторингу відео та контролера та підключення до сервера за допомогою HTTP запитів.

Apache - це популярний Інтернет-сервер, який можна встановити за допомогою малини Pi для обробки веб-сайтів. Apache подають HTML-файли через HTTP та додаткові модулі можуть обробляти динамічні веб-сайти за допомогою таких скриптів, як PHP, що буде дуже необхідним для реалізації функції Smart Home.

Щоб встановити Apache та PHP, використовуйте цю команду та дотримуйтесь проблему процесу встановлення:

```
sudo apt-get install apache2 php5 libapache2-mod-php5
```

Виконуємо такі команди для створення окремого системного користувача для веб-сервера:

```
sudo groupadd www-data sudo usermod -g www-data www-data
```

Перезавантажуємо Apache за допомогою наступної команди:

```
sudo service apache2 restart
```

Після цього веб-сервер повинен бути доступний для роботи, але повністю реалізуючи функції, необхідні для установки MySQL.

MySQL - безкоштовна система управління базами даних.

MySQL створив TCH, щоб збільшити швидкість великих баз даних. Ця система управління базами даних (SKBD) з відкритим джерелом була розроблена як альтернатива комерційним системам. MySQL з самого початку був дуже схожий на MSQl, але з часом він розширив все, і тепер MySQL є однією з найпоширеніших систем управління базами даних. Він використовується, зокрема, для створення динамічних сайтів, оскільки вона має відмінну підтримку з різних мов програмування.

Пристрій MySQL, використовуючи цю команду та дотримуйтесь процесу встановлення:

```
sudo apt-get install mysql-server mysql-client php5-mysql
```

Якщо все встановлено без помилок, після цього контролер готовий для написання php-коду та веб-сторінок, що зможуть зручно відображати інформацію, що отримується з датчиків та сенсорів та передавати її на сервер, звідки до неї зможе мати доступ клієнт.

4.3.3 Написання необхідних PHP-сценаріїв

PHP (PHP: гіпертекстпроцесор - PHP: гіпертекстпровід), попередня назва: персональний домашній інструмент - мова програмування SCET була розроблена для створення HTML-сторінки з боку вашого сайту. PHP - одна з найпоширеніших мов, що використовуються в області WebWork (разом з Java, .net, Perl, Python, Ruby). PHP підтримується переважними постачальниками хостингу. PHP - Відкрийте дизайн програмного забезпечення.

PHP інтерпретує HTML-код веб-сервера, який надсилається на сторінку клієнта. На відміну від мови сценаріїв JavaScript, користувач не бачить PHP-коду, оскільки браузер отримує підготовлений HTML-код. Це є перевага безпеки, але погіршує сторону інтерактивності. Але ніхто не забороняє, використовуючи генеровані PHP-кодери та JavaScript кодерів на стороні клієнта.

Наш прототип, регулятор за замовчуванням надсилає інформацію від датчиків до відповідних текстових файлів та останніх з інших файлів з відповідною інформацією керування. Тому ви спілкуєтеся з сервером, ви повинні написати PHP-скрипт, який буде надсилати інформацію на сервер і взяти на себе керуючі сигнали.Приведемо типовий php-сценарій для відправки та для отримання інформації.

Для відправки інформації:

```
<?php      if      (($paramfile      =
fopen("controls/params.txt", "r")) === FALSE ){
    echo "Ошибка открытия файла параметров!";
} else {
    if (($params = fread($paramfile, filesize("params.txt"))) ==
) {

        echo "Ошибка чтения файла параметров!";
```

```

        } else
    {
        echo
$params;
    }
    fclose($paramfile);
}
?>

```

Для отримання інформації:

```

<?php
    if(isset($_GET['c'])) {
        $command = $_GET['c'] . "\n" ;    if
(($commfile = fopen("controls/commands.txt", "a")) ===
FALSE ){
            echo "Ошибка открытия файла
комманд!";
        } else {
            if
(fwrite($commfile, $command) === FALSE) {
                echo "Ошибка записи команды!";
            } else {
                echo "Команда записана успешно.";
            }
        }
        fclose($commfile);
    }
}

```

Це лише найпростіший типовий приклад сценаріїв. Зазвичай на практиці таких сценаріїв багато, та вони можуть бути складніші. З повним текстом сценаріїв, що було використано при створенні нашого прототипу, можна ознайомитися на [GitHub](#).

4.1.3 Налаштування відео-потoku

Щоб отримати пульт дистанційного керування роботами з камери, для інших Інтернет-систем та інших веб-систем, необхідно створити відеопотік у режимі реального часу.

Є кілька способів отримати відеопотічок, який практично негайно працює. Це GStreamer і Netcat та VLC, а інший -. У реальних системах краще використовувати ці рішення або їх аналоги. Але через створення прототипу ми будемо використовувати потік MJPEG, оскільки він працює на широкому діапазоні апаратних платформ і не вимагає додаткової порівнянності. Багато IP-веб-камер використовують цей старий протокол потоку, який розшифровує як JPEG. Motion JPEG - це лише потік окремих фотографій JPEG, один за одним. Це не дивно, але більшість сучасних браузерів можуть відтворювати рідини MJPEG без додаткових (як зазначено вище). Зворотна сторона потоку MJPEG полягає в тому, що вони не такі ефективні, як H.264, що значно підвищує якість і зменшує розмір, кодуючи лише відмінності в одному кадрі з іншого. З MJPEG кожен кадр закодований для всього зображення JPEG. Але для нашого прототипу не має значення.

Таким чином, ви повинні налаштувати контролер для роботи з потоком MJPEG, щоб організувати відеоспостереження. По-перше, оновімо RPi, а потім встановімо деякі з залежності MJPGStreamer:

```
sudo apt-get update sudo apt-get upgrade sudo apt-get
install libjpeg8-dev imagemagick libv4l-dev
```

Треба зробити налаштування: файл videodev.h замінити на videodev2.h, але ця заміна має бути зроблена вручну. Щоб зробити її, просто створімо символічне посилання:

```
sudo ln -s /usr/include/linux/videodev2.h
/usr/include/linux/videodev.h
```

Завантаження та компіляція MJPG-Streamer. На жаль, MJPG-Streamer не доступний через apt-get, таким чином, ми повинні скомпілювати його з вихідного коду. Це трохи складніше, ніж налаштування веб-камери з використанням Motion, але MJPG-Streamer менш ресурсоємний, тому додаткова складність встановлення варта того.

Є кілька різних способів установки MJPG-Streamer. Найпростіший - за допомогою subversion(SNV).

Встановімо

```
subversion: sudo apt-get install subversion
```

Потім завантажимо копію вихідного коду MJPG-Streamer в домашню папку і перейдемо в каталог `mjpg-streamer`, коли завершиться завантаження:

```
cd ~ && svn co https://svn.code.sf.net/p/mjpg-streamer/code/mjpg-streamer/ mjpg-streamer && cd mjpg-streamer
```

В MJPG-Streamer входить багато плагінів, але ми скопілюємо лише три необхідні нам для прототипу. За бажанням, можна встановити і інші.

- `input_uvc.so` (для USB веб-камери): копіюємо JPG-файли з одного входу (в даному випадку, з веб-камери) до одного або більше вихідних модулів. Це добре для потокової передачі великих зображень з більш високою частотою кадрів і при меншому завантаженні процесора.
- `input_file.so` (для модуля камери): аналогічно описаному вище, але копіює їх з каталогу.
- `output_http.so`: потокова передача файлів на веб-сервер `make mjpg_streamer input_file.so input_uvc.so output_http.so` Крім того, запустивши `make`, ми можемо скопілювати все, що прийшло з MJPG-Streamer. Тут вибране для встановлення тільки те, що було необхідно, щоб заощадити об'єм сховища і ресурси.

Можна запустити MJPG-Streamer після компіляції прямо з папки (`./mjpg_streamer`), але було б краще перемістити його на постійне місце.

Виконаємо наступні команди, щоб скопіювати MJPG-Streamer в більш глобально-доступний каталог:

```
sudo cp mjpg_streamer /usr/local/bin && sudo cp output_http.so  
input_file.so input_uvc.so /usr/local/lib/ && sudo cp -R  
www/usr/local/www/usr/local/
```

Це є загальним місцем для сторонніх предметів або речей, доданих в систему адміністратором, але це не є необхідним кроком.

Якщо зараз спробувати запустити `mjpg_streamer`, то, ймовірно, повернеться повідомлення про помилку при пошуку шляхів (`error about search paths`). Це тому, що прямо зараз система не знає, де знайти файли. Це

легко виправити. Просто додаймо наступний рядок в `~/.bashrc` (якщо були викликані команди вище, щоб скопіювати файли в `/usr/local/`). Буде зроблена постійна його заміна, так що не треба буде робити це щоразу, коли входимо:

```
export LD_LIBRARY_PATH=/usr/local/lib/
```

Тепер можна вийти і зайти знову, але простіше виконати команду `source`, щоб застосувати зміни (це має той же ефект, що і перезавантаження):

```
source ~/.bashrc
```

Тепер можна просто викликати `mjpg_streamer`, незалежно від того, в якій директорії ми знаходимось.

Запустимо `mjpg_streamer` за допомогою команди:

```
/usr/local/bin/mjpg_streamer -i "/usr/local/lib/input_uvc.so"  
-o "/usr/local/lib/output_http.so -w /usr/local/www"
```

Ця команда має такі параметри:

- `-i`: використовує `input_uvc.so` (USB веб-камера) як вхід
- `-o`: `output_http.so` для виходу (відправка зображення на веб-сервер)
- `-w`: каталог, який має HTML, CSS, JS і файли:
`/usr/local/www`

Можна скасувати потік, натиснувши `Ctrl + C`. Також за потреби можна поекспериментувати з деякими іншими параметрами, що перераховані нижче:

- `-b`: запуск у фоновому режимі
- `-p`: встановити інший порт за замовчуванням, а не порт 8080 за замовчуванням
- `-c`: запитує ім'я користувача:пароль
- `-y YUYV`: дозволяє формат YUYV й відключає режим MJPEG
- `-f`: частота кадрів (у секундах)

Якщо використати параметр `-b`, то будемо отримувати підказку на екрані. Отже, як зупинити запущений потік? Після команди запуску,

побачимо рядок, типу: `forked to background (4979)`. Щоб зупинити потік, просто напишемо: `kill 4979`. Це «вбиває» процес ID (PID) `MJPEG-Streamer`.

Налаштуємо перегляд потокового відео з веб-камери через браузер (при підключенні до локальної мережі). Вище було описано налаштування веб-сервера, можна як і раніше, працювати з ним без проблем, тому що потік доступний на іншому порту (за замовчуванням): 8080. Тому, якщо зайти на нього з браузера, просто додаймо :8080 після IP-адреси:

– локально: `http://localhost:8080`

– з іншого пристрою в мережі: `http://<ip_адреса_raspberry_pi>:8080`

Перегляд зображення з веб-камери при підключенні до локальної мережі - це вже відеоспостереження, але покажемо, як отримати доступ до веб-камери з будь-якого місця, ввівши URL типу: `myDomain.com:8080`. Цей потік може бути захищений паролем, але це насправді не дуже безпечно, тому слід бути обережним, роблячи веб-камеру доступною онлайн.

Попередні умови:

– Перенаправлення порта 8080 на ваш RPi

– Включення служби динамічного DNS на маршрутизаторі

– (опціонально) Налаштування веб-сервера з каталогом, захищеним паролем

Після того, як описане вище завершено, зможемо отримати доступ до відео потоку зі своєї веб- або ip-вкамери з будь-якого браузера, перейшовши на `http://myDomainName.org:8080`. Але на неї ще немає пароля.

Ми можемо задати, щоб `MJPEG-Streamer` використовував пароль, за допомогою параметра `-c`. Даний пароль не надто безпечний (бо це тільки кодування base 64), але він може бути першою лінією захисту. Дехто зможе легко перехопити і розшифрувати пароль, але якщо ви знаходитесь в довірній мережі, то це не така вже й проблема.

Команда буде виглядати наступним чином:

```
/usr/local/bin/mjpg_streamer -i "/usr/local/lib/input_uvc.so"
-o "/usr/local/lib/output_http.so -w /usr/local/www -c
username:password"
```

Кращий спосіб добитися безпеки будь-якого інтернет-ресурсу - використання моделі пошарової оборони. Тому, перш ніж винести це в Інтернет, можна захиститися за допомогою інших інструментів, що надають інформаційну безпеку, але у цьому розділі це не розглядається.

Для простоти, зробимо сторінку в тому ж каталозі, в якій знаходяться HTML-файли. Це полегшує перевірку, що сторінка працює. Після того, як переконаємось, що все працює, сторінку можна перемістити будь-куди, але треба буде просто налаштувати шляхи. Наступний HTML-код створить файл `myscam.html`, і сторінка буде готова до роботи:

```
<html>
  <head><title>My Live-stream</title></head>
  <body>
    <center><h1>My Camera</h1></center>
    <center>
      
    </center>
  </body>
</html>
```

Тепер можна побачити її за адресами:

- <http://192.168.1.100:8080/myscam.html> (в локальній мережі) •

<http://myDomainName.org:8080/myscam.html> (через Інтернет де 192.168.1.100 треба замінити на IP-адресу RPi.

Тепер треба налаштувати запуск MJPG-Streamer як демона (фонову службу).

Нижче є два сценарії: дуже простий, який виконує команди, і просунутий сценарій, який дозволить використовувати команду служби, щоб контролювати його:

```
sudo service livestream.sh start sudo service
livestream.sh stop sudo service livestream.sh restart
```

Дуже простий сценарій

нижче:

```
sudo nano etc/init.d/livestream.sh
#!/bin/bash /usr/local/bin/mjpg_streamer -i
"/usr/local/lib/input_uvc.so" -o
"/usr/local/lib/output_http.so -w /usr/local/www -c
username:password"
```

Збережімо сценарій та зробимо його виконуваним:

```
sudo chmod 755 /usr/sbin/livestream.sh
```

Наступна команда зробить, щоб він виконувався під час завантаження: `sudo update-rc.d livestream.sh defaults`

Нижче наводиться сценарій, який буде завантажувати MJPG-Streamer при запуску RPi і дозволить використовувати команду служби:

```
#!/bin/sh
# /etc/init.d/livestream.sh
### BEGIN INIT INFO
# Provides:          livestream.sh
# Required-Start:    $network
# Required-Stop:     $network
# Default-Start:     2 3 4 5
# Default-Stop:      0 1 6
# Short-Description: mjpg_streamer for webcam
# Description:        Streams /dev/video0 to
http://IP/?action=stream
### END INIT INFO
f_message() {
    echo "[+] $1"
}

# Carry out specific functions when
asked to by the system case "$1" in
```

```

start)

    f_message "Starting mjpg_streamer"
    /usr/local/bin/mjpg_streamer -b -i
"/usr/local/lib/input_uvc.so" -o
    "/usr/local/lib/output_http.so -w /usr/local/www -c
username:password"

    sleep 2
f_message "mjpg_streamer
started"

;;
stop)

    f_message
"Stopping mjpg_streamer..."
killall mjpg_streamer
f_message "mjpg_streamer stopped"

;;
restart)
f_message
"Restartin
g daemon:
mjpg_strea
mer"
killall
mjpg_strea
mer

    /usr/local/bin/mjpg_streamer -b -i
"/usr/local/lib/input_uvc.so" -o
    "/usr/local/lib/output_http.so -w /usr/local/www -c
username:password"

    sleep 2
f_message "Restarted daemon:
mjpg_streamer"

```

```
;;
status)

                                pid=`ps -A | grep mjpg_streamer | grep
-v "grep" | grep -v
mjpg_streamer. | awk '{print $1}' | head -n 1`

if [ -n "$pid" ];
then
                                f_message "mjpg_streamer is running
                                with pid ${pid}"                                f_message
                                "mjpg_streamer was started with the following
                                command line"

                                cat
/proc/${pid}/cmdline      ;      echo      ""
else
                                f_message      "Could
not      find      mjpg_streamer      running"
fi

;;

*)

                                f_message      "Usage:  $0
{start|stop|status|restart}"
exit 1

;;

e

sa
c
ex
it
0
```

Після збереження, виконаємо ті ж команди, що й вище:

```
sudo chmod 755 /etc/init.d/livestream.sh
sudo update-rc.d livestream.sh defaults
```

Тепер наша камера доступна через Інтернет і запускається при включенні RPi. Потік з камери може бути прочитано клієнтом за допомогою ПО, що розроблено в ході роботи.

4.3.4 Налаштування сервера Ubuntu

На нашому сервері сервер виконує багато важливих функцій контролера технічного обслуговування та обробки, щоб забезпечити додатковий рівень безпеки, оскільки сервер може бути унікальним.

Перша позиція параметрів сервера відповідає встановленню контролера Маліну. Сервер повинен встановити FTP-сервер, веб-сервер, PHP та MySQL. Ці заходи тут не описані, оскільки вони описані у попередньому розділі.

Однак, на відміну від контролера, на сервері будуть використані деякі інші сценарії PHP. Сторінка PHP Server також матиме код JavaScript з JQUERY, AJAX та JSON.

Динамічний JavaScript (JS), мова програмування зосереджується на екзінхронно повторюваних даних з сервера, змінюючи структуру та зовнішній вигляд на сайті. " JavaScript також використовується на сервері програми (подібні мови програмування, такі як Java та C #), стаціонарні та мобільні програми, сценарії програмного забезпечення (наприклад, Adobe Creative Suite), PDFDocuments всередині та ін.

Javaquery - це популярна бібліотека JavaScript з відкритим кодом. Він був доставлений у 2006 році. У січні. Barcamp NYC John Resig. За словами W3Techs, JQuery використовує більше півмільйона, зазвичай сайти.

Javaquery є найпопулярнішою бібліотекою JavaScript, яка інтенсивно використовується. JQUERY SYNTAX призначений для зручної навігаційної орієнтації через домашні елементи, розвиток анімації, обробки та програмування подій. JQUERY також дає розробникам можливість створити

додаток JavaScript у верхній частині бібліотеки. Використовуючи ці об'єкти, розробники можуть створювати абстракцію в взаємодії з низьким рівнем та створювати високоякісний мультфільм. Це сприяє створенню потужних і динамічних сайтів.

AJAX (асинхронний JavaScript та XML) - підхід до розробки учасників інтернет-інтерфейсу, який веб-сайт та надсилання запитів сервера та лоюльних даних для користувачів. Ајах є одним з елементів динамічної HTML-концепції.

JSON (англійська маркування JavaScript маркування, Великобританія. Приєднатися до JavaScript, Bright Jason) - це текстовий формат для обміну даними між COM. JSON базується на тексті і може легко читати людину. Формат дозволяє описати об'єкти та інші структури даних. Цей формат в основному використовується для передачі структурної інформації в мережі (через процес, що називається серіалізація).

Формат Stekhord Douglas був розроблений та сприятливий.

JSON виявив основну мету написання веб-додатків, а саме використання технології AJAX. JSON, який використовується для АЈА (асинхронний JavaScript та JSON) як поправка XML (використовується в Ајах) асинхронно побудовано інформацію між клієнтом та сервером. У цьому випадку перевага JSON полягає в тому, що вона забезпечує складні структури в атрибутах, займає менше місця і безпосередньо інтерпретується з JavaScript.Приведемо типовий сценарій задля прийому інформації від контролера та передачі інформації контролеру за допомогою HTML/jQuery/Json:

```
<HTML>
<head>
<meta name="robots" content="noindex,nofollow">
<meta name="viewport" content="width=200">
<meta charset="UTF-8">
```

```

<script
src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.
min.js"></script> <script>
    $(document).ready(function() {
        // Форма обратной связи var regVr22 = "<div><img
width=\"28\" height=\"16\" style=\"margin-bottom:4px;\"
alt=\"...\" src=load.gif><span style=\"font: 11px Verdana;
color:#333; marginleft:6px;\">Сообщение
обрабатывается...</span></div><br>";
        $("#bget").click(function(){
            $("#loadBar1").html(regVr22).show();

            $.ajax({
                type:
"GET",
                url: " controllerURL
/debian.php?callback=?",
                jsonpCallback: "my_callback_function",
                dataType: "jsonp", data: {
            }, cache: false,
            success: function(json){

$("#loadBar1").html(json.message).fadeIn(30000);
                }
            });
            return false;
        });

        $("#bpost").click(function(){
            var c = $("#command").val();
            $("#loadBar2").html(regVr22).show();

            $.ajax({
                type:
"GET",

```

```

        url:
"controllerURL/debian.php?callback=?",
        jsonpCallback: "my_callback_function",
        dataType: "jsonp",
data: { "c" : c },
        cache:
false,
        success: function(json) {
            var all = (json.message);
            $("#loadBar2").html(all).fadeIn(30000);
        }
    });
    return false;
});
});
</script>
</head>
<body>
<center>
<form id="cForm2" action="" method="post">
<button id="bget" type="button">получить
параметры</button>
<DIV id="loadBar1" align="center">&nbsp;</DIV>
<br>
<input type="text" name="command" id="command"
value="">
<button id="bpost" type="button">Послать
команду</button> <DIV id="loadBar2" align="center"></DIV>
</form>
</center>
</body>
</HTML>

```

Де controllerURL – мережева адреса контролеру.

Як і випадку з контролером, це лише типовий сценарій отримання та відправки даних з сервера. Таких сценаріїв може бути багато та вони можуть бути складніше. Подібні сценарії використовуються й для зв'язку між

сервером та мобільним додатком, що розробляється у ході дипломного проекту. Повний код сценаріїв доступний на GitHub.

Використовуючи описані вище технології, було створено веб-сторінку для взаємодії користувача з контролером, а саме веб-сторінка дозволяє:

- Переглядати потокове відео (відеоспостереження)
- Отримувати онлайн інформацію з датчиків контролера
- Керувати пристроями, що підключені до GPIO контролера

Вся інформація відображається динамічно, з використанням принципу AJAX (тобто в реальному часі без перезавантаження веб-сторінки). Можна побачити затримку (приблизно 5-10 секунд) між отриманням нових значень температури та вологості, але це пов'язано з використанням у прототипі бюджетних датчиків, які не спроможні відправляти інформацію частіше.

Скріншот сторінки можна побачити на рисунку 4.4.



Рисунок 4.4 Розроблена сторінка

Висновки до розділу 4

На базі мікро-комп'ютера Raspberry – пі 3 був реалізований розумний будинок до функціоналу котрого входить управління чотирма реле, збір даних зі зовнішньою середою, а саме температуру та вологість повітря, трансляція відео-зображення у реальному часі та детектор руху. Але це не є кінцева реалізація проекту. У своїй дипломній роботі я аналізував різноманітні системи та обладнання, перевіряючи їх до відповідності з параметрами описаними у першому розділі. Тим самим я поступався часом на впровадження більш широких можливостей системи. Наприклад, до даної системи можна додати автоматичне керування термо-кліматом, яке опиралось би на встановлену користувачем температуру і в залежності від стану системи корегувало наступні свої дії. Наприклад, якщо у приміщенні холодно, то система би вмикала обігрів, чи навпаки, уникаючи одночасного використання регулюючого обладнання. Також у систему треба додати зв'язок між детектором руху та камерою, щоби економити ресурси мережі й концентрувати увагу користувача у разі виникнення непередбачених обставин, як грабіж.

Використання даного контролеру дало найбільш вражаючі результати, що в точності відповідають поставленій задачі проекту. А вбудований WiFi-модуль дозволяє й далі розвивати систему, підключаючи обладнання по всьому будинку, підключення нового обладнання по силам користувачам, які хоч віддалено знайомі з програмуванням контролерів.

Окрім цього, у цьому розділі було налаштовано FTP-сервер, веб-сервер, відеоспостереження, бази даних MySQL, встановлено пакети для роботи з мовою сценаріїв PHP.

Було вивчено сучасні технології веб-програмування та створено вебсторінку, що дозволяє взаємодіяти із контролером, а саме:

- Переглядати потокове відео (відеоспостереження)
- Отримувати онлайн інформацію з датчиків контролера

- Керувати пристроями, що підключені до GPIO контролера

5 ПОРІВНЯННЯ РОЗРОБЛЕНИХ РІШЕНЬ

Як вже зауважував у виводі другого розділу, використання системи побудованої з використанням промислового обладнання у межах розумного будинку, має як перевагу, так й явні мінуси. Розглянемо більш детально.

Кількість входів на двох контролерах дозволяє підключити до мережі більш ста різноманітних пристроїв, які працюють за технологією PROFinet на базі мережі PROFibus. Цей стандарт використовує Ethernet-кабелі для зв'язку обладнання. В теорії, така система може виконувати майже любі функції стосовно нагляду й керування розумним будинком. Але, як встановлює практика, середня система розумного будинку обмежується керуванням побутовими приладами, декількома датчиками температури, руху, й охоронною системою відео-спостереження, й пожежної безпеки. Тобто, основна потужність контролерів залишиться незадіюваною, якщо встановити їх до середньостатистичної квартири. Також, мало хто погодиться проводити глобальний ремонт приміщення задля встановлення такої системи, з'єднання приладів котрої виконується за допомогою дротів. Зважаючи на велику вартість самого обладнання, важкість його налаштування та монтажу було вирішено відмовитися від прототипування системи, яка вочевидь не підходить для широкого впровадження.

Тому, в другій частині, їх всього три, я розглянув систему з протилежними параметрами. Замість промислових контролерів – дешева плата Arduino Uno. Благо, на сьогоднішній день їх безліч у магазинах обладнання, чи в інтернеті. В моєму розпорядженні було: Плата Arduino Uno; Аналоговий датчик температури; Чотири механічних реле; Ethernet-шилд ENC28J60.

Сумарна вартість обладнання складає – 1820 грн. За умови, що все обладнання оригінальне. Підключення та налаштування обладнання не склало труднощів. Контролер з легкістю обробляє дані за датчика

температури, перетворюючи рівень сигналу напруги у цифрове значення, використовуючи для цього аналоговий вивід. Також легко відбувається керування реле, достатньо лише подати необхідний рівень напруги до конкретного pin-у. Складнощі праці з такою системою стали виникати, коли я спробував налаштувати керування системою хоча б на локальному рівні. Підключив Ethernet-модуль до плати та написав якийсь елементарний програмовий код, котрий присвоював ip-адресу контролеру, дозволяючи тим самим заходити до веб-інтерфейсу, який містив у собі керування чотирма реле, та відображував температуру приміщення кожні дві секунди, я звернув увагу на те, що ця програма використовує вісімдесят процентів пам'яті пристрою. Як виянилось пізніше, це проблема дешевого Ethernet-модулю, котрий підтримує бібліотеку лише EtherCard і не підтримує класичну бібліотеку Ethernet. Бібліотека EtherCard неспроможна переключатись при з'єднанні між режимами «клієнт» та «сервер», а завжди виступає у ролі «сервера». Модулі з підтримкою класичної бібліотеки стоять на порядок більше, більше навіть самої плати. Якщо скласти висновки по другій частині, то можна сказати, що використання у якості головного контролера Arduino не є розумним, якщо наша мета – це побудова повноцінного розумного будинку. Ресурсів одного контролера недостатньо для обробки відео, а вивід даних до мережі потребує додаткових затрат на обладнання, або використання додатної потужності у вигляді же однієї плати.

Оцінив все це і зробив для себе висновки, я став шукати рішення, спроможне все ж таки реалізувати систему з вже описаними параметрами. Третє і кінцеве рішення – використання мікро-комп'ютера Raspberry pi-3. Raspberry Pi - найпопулярніший в світі одноплатний комп'ютер розміром з кредитну картку.

Використання даного контролеру дало найбільш вражаючі результати, що в точності відповідають поставленій задачі проекту. А вбудований WiFi-модуль дозволяє й далі розвивати систему.

ВИСНОВКИ

Мета дипломного проекту – побудова протиптної моделі системи «розумний» будинок для подальшого дослідження і застосування інфокомунікаційних технологій.

Задачі проекту полягають в:

- опис загальної концепції;
- реалізації прототипу на різних платформах;
- вибір оптимальної платформи з перспективою можливості подальших досліджень.

Я визначив основні параметри для впровадження системи в життя. Такі як низька вартість обладнання, легкість налаштування та керування системою, а також наступний функціонал:

- Відеоспостереження в режимі реального часу.
- Керування електроприладами.
- Збір даних о стані зовнішньої середи.
- Автоматичний клімат-контроль будинку.

Використання TATUSmartLab дозволяє створити досить надійну систему розумного будинку, потужності процесорів достатньо, щоб організувати відео спостереження, роботу безліч датчиків та реле. Але у такого рішення є три серйозні мінуси:

- Складність налаштування системи та складність підключення нового обладнання.
- Обмеженість варіантів керування. Спілкування з системою може здійснюватися з використанням GSM/GPRS-модуля за смс, або email.
- Вартість обладнання. В межах розумного будинку використання потужного промислового контролера не має

сенсу, бо є більш дешевші та прості в використанні контролери, призначені саме для таких задач.

Використання плати Arduino в якості головного контролера дозволило створити систему керування реле та збору даних з датчика температури через веб-інтерфейс. Головним недоліком такої системи є мала ресурсоемність контролера, яка не дозволяє працювати з потоковим відео, та обмежує функціонал використання веб-інтерфейсу.

На базі мікро-комп'ютера Raspberry – пі 3 був реалізований розумний будинок до функціоналу якого входить управління чотирма реле, збір даних зі зовнішньої середовища, а саме температуру та вологість повітря, трансляція відео-зображення у реальному часі та детектор руху.

Кожна з розглянутих систем має свої плюси та мінуси, але найбільш точно відповідає нашим умовам саме система побудована з використанням у якості центрального контролера Raspberry-пі 3, потужності якого досить для організації повноцінного розумного будинку зі всіма нашими параметрами.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Умный дом— Режим доступа :
http://www.directinfo.net/index.php?option=com_content&view=article&id=139%3A2010-07-06-13-57-09&catid=1%3A2008-11-27-09-05-45&Itemid=84&lang=r.
2. Система умный дом – технология экономии, удобства и комфорта высокого уровня . – Режим доступа :
http://smarton.com.ua/smart_home/systema_umnyi_dom_intro .
3. Система умный дом. – Режим доступа :
http://intelcity.com.ua/comfort_house - Дата доступа : 14.05.2013.
4. Виртуальная реальность в шлеме Microsoft. – Режим доступа :
http://www.bbc.com/russian/multimedia/2015/01/150122_microsoft_helmet.
5. Кто отвечает за климат-контроль в доме?— Режим доступа :
http://smarton.com.ua/smart_home/klimat_kontrol_v_umnom_dome .
6. Microsoft's HoloLens Is "Something Different" Than Oculus or Morpheus – Режим доступа : <http://www.gamespot.com/articles/microsoft-s-hololens-issomething-different-than-o/1100-6424809/> .
7. Meizu выходит на рынок «Умного дома» – Режим доступа :
<http://itc.ua/news/meizu-vyihodit-na-ryinok-umnogo-doma-s-pomoshhyu-platformyilifekit-lineyki-raznogo-roda-ustroystv/> .
8. Orvibo Allone Wi-fi – Режим доступа :
<http://www.geekbuying.com/item/Orvibo-Allone-WiFi---IR---RF-Remote-ControlSmart-Home-Automation-Rechargeable-Battery-for-IOS-Android-Mobile-338990.html> .
9. Фрагментация Android – Режим доступа :
<http://habrahabr.ru/post/188738/>
10. Android Fragmentation Vizualized – Режим доступа :

http://opensignal.com/reports/2014/androidfragmentation/#android_version_ties

11. Android Studio. – Режим доступу :
<https://developer.android.com/sdk/index.html>

12. 14Хмарні сховища даних – Режим доступу :
<http://oksim.com.ua/index.php/167-khmarni-skhovishchadanikh>

13. SharedPreferences. – Режим доступу :
<http://developer.android.com/reference/android/content/SharedPreferences.html>

14. AsyncTask. – Режим доступу:
<http://developer.android.com/reference/android/os/AsyncTask.html>

15. OpenWeather Map – Режим доступу : <http://openweathermap.org/>

16. Material Design Introduction. – Режим доступу :
<https://www.google.com/design/spec/materialdesign/introduction.html>

ДОДАТОК А

Охорона праці

А.1 Служба охорони праці та її основні задачі

Служба охорони праці створюється на кожному підприємстві для організації виконання правових, організаційно–технічних, санітарно-гігієнічних, соціально-економічних і лікувально-профілактичних заходів, направлених на запобігання нещасних випадків, професійних захворювань і аварій в процесі праці у всіх структурних підрозділах підприємства.

Служба охорони праці відноситься до основних виробничо-технічних служб, входить в структуру підприємства і функціонує як самостійний структурний підрозділ.

У своїй діяльності працівники служби охорони праці керуються законодавством про охорону праці, міжгалузевими і галузевими нормативними актами про охорону праці, посадовими інструкціями, наказами і вказівками своїх керівників.

Служба охорони праці вирішує наступні задачі :

- контроль по забезпеченням безпеки виробничих процесів, устаткування, будівель і споруд, нормалізації санітарно - гігієнічних умов праці у всіх структурних підрозділах;
- забезпечення працюючих засобами індивідуального і колективного захисту;
- вибір оптимальних режимів праці і відпочинку працюючих.

Основними функціями служби охорони праці є:

- розробка ефективної цілісної системи управління охороною праці на виробництві, сприяння вдосконаленню діяльності в цьому напрямі кожного структурного підрозділу і кожного

посадовця підприємства, внесення документів з цих питань на твердження генеральному директору;

- вдосконалення системи навчання працівників ;
- організація забезпечення працюючих нормативними документами по охороні праці;
- організація оперативного обліку і аналізу нещасних випадків, професійних захворювань і аварій, а також збитку від цих подій; підготовка статистичний звітів підприємства з питань охорони праці;
- організація розробки перспективних і текучих планів профілактичної роботи, направленої на усунення причин нещасних випадків, профзахворювань і аварій і контролю за їх виконанням;
- допомоги комісії із питань охорони праці підприємства в підготовці необхідних матеріалів і реалізації її рекомендацій;
- підвищенні кваліфікації і перевірки знань посадовців і робітників з питань охорони праці і гігієни праці, уміння надавати допомоги захопленим аварією і потерпілим.

А.2. Характеристика підприємства і стану охорони праці на ньому

На підприємстві задіяні робітники різноманітних професій:

- системні адміністратори;
- програмісти;
- бухгалтера та секретарі;
- співробітники служби кол-центру та техпідримки.

Службові обов'язки вимагають від співробітників проводити майже весь робочий день за персональними комп'ютерами. Зважаючи на це,

необхідно визначити основні вимоги до приміщень де будуть розміщені робочі місця співробітників.

А.3 Основні шкідливі виробничі фактори, що впливають на умови праці

Експлуатація ПК супроводжується впливом на організм комплексу факторів трудового процесу й виробничого середовища. У результаті настають зміни функціонального стану центральних нервових, серцево-судинної систем організму, що характеризуються вираженою напругою. Зміни, що накопичуються в процесі роботи, приводять до ризику розвитку певних захворювань.

Основними шкідливими факторами, пов'язаними з роботою на ПК є:

- напруга зорових органів і пов'язані з ним стомлення, захворювання й побічні ефекти;
- значне навантаження на пальці й кисті рук, що при відсутності профілактики й медичного контролю можуть викликати професійні захворювання;
- тривале знаходження в одній і тій же позі, що викликає застійні явища в організмі, що може сприяти різним захворюванням;
- випромінювання різного виду при використанні відеомоніторів на електронно-променевих трубках (м'яке рентгенівське випромінювання, ультрафіолетове випромінювання, видиме випромінювання, інфрачервоне випромінювання, низько й високочастотне електромагнітне випромінювання, електростатичні поля;

- механічні шуми, пов'язані з роботою електронно-механічного друкувального пристрою (принтера), вентиляторів системи охолодження, приводів читання CD-дисків і вібрація;
- іонізація повітря;
- накопичення в повітрі робочого приміщення шкідливих хімічних речовин (біфеніли, озон, триметилфосфат, фуран).

Дослідження науково-дослідного інституту гігієни праці й профзахворювань указали на зміни у функціональному стані зорового аналізатора в ході виробничої діяльності фахівців, що працюють із відеотерміналами, наприкінці 4 години роботи. Вищенаведені фактори сприяють виникненню у робітників різноманітних професійних захворювань.

А.4 Заходи щодо поліпшення умов праці робітників

Робочим приміщенням будівля, що складається з 25 окремих кімнатах.

Загальна площа офісного приміщення становить 835 м², геометричні розміри 25х35 м, висота стелі становить 3,2 м. У приміщеннях розташована комп'ютерна техніка, споживана потужність якої становить 0,7 кВт, та устаткування кондиціонування та вентиляції

Приміщення знаходиться на 1 поверсі офісної будівлі цегельної забудови і по своїм характеристикам цілком відповідає вимогам СНіП 2.09.04.-87.

До приміщення й організації робочого місця з обліком шкідливих виробничих факторів пред'являється ряд вимог. (забезпечення необхідного мікроклімату згідно з ГОСТ 12.1.005-88, встановлення кондиціонера разом з іонізатором та очисником повітря, застосування антистатичного покриття для підлоги, виконання необхідних вимог до облаштування робочих місць з ПК та з іншими приладами, боротьба з шумом, застосування захисних екранів

для моніторів ПК і спеціальних окулярів з комп'ютерним спектральним фільтром та інш.). Виконання медичинських протипоказань до праці з ПК.

Приміщення у якому знаходиться робоче місце з ПК повинно мати природне освітлення, бажано з одnobічним розміщенням світопрорізів, площа осклянілості яких не повинна перевищувати 25% від площі стіни. Віконні прорізи в приміщенні з ПК повинні мати регульовані жалюзі чи занавеси, чи інші сонцезахисні пристрої.

Не допускається розташування робочих місць із ПК у підвальних і цокольних поверхах. Робочі місця з ПК рекомендується розміщати в окремих приміщеннях. Площа на одного працюючого з ПК повинна складати 6 м², обсяг - 20 м³ (НПАОП 0.00-1.31-99. Правила охорони праці під час експлуатації електронно-обчислювальних машин).

Неприпустиме розташування ПК, при якому працюючий звернений обличчям, або спиною до вікон чи кімнати задньої частини ПК, у яку монтуються вентилятори.

Забороняється застосовувати для обробки інтер'єра приміщень із ПК полімерні матеріали (древісностружечні плити, шпалери що миються, плівкові і рулоні синтетичні матеріали, шаруватий паперовий пластик і ін.), що виділяються в повітря шкідливі хімічні речовини, що перевищують гранично припустимі концентрації.

А.4.1 Аналіз розміщення робочих місць

Розглянемо облаштованість робочих місць. При аналізі облаштованості робочих місць із ПК порушень не виявлено. Облаштованість всіх кімнат відповідає ГОСТ:

- світло падає на робоче місце з боку;
- приміщення, у якому перебувають робочі місця із ПК мають природне висвітлення, з одnobічним розміщенням вікон, площа

вікон не перевищує 25% від площі стіни. Віконні прорізи в приміщення мають регульовані жалюзі;

- загальна площа всіх приміщень становить 547м^2 , об'єм – 1750м^3 , тобто можна розмістити 91 робоче місце з ПК;
- дотримується відстань між столами, що перебувають біля стіни - 1м;
- екран відеомонітора не використовується бо застосовані рідкокристалічні монітори;
- висота робочої поверхні стола регулюється в межах 680-800мм;
- робочий стіл має, простір для ніг висотою 600мм, шириною - 450мм;
- висота поверхні сидіння регулюється в межах 400-550мм. Ширина й глибина поверхні сидіння 400мм. Поверхня сидіння плоска, передній край - закругленим. Передбачена можливість зміни кута нахилу поверхні від 15 град уперед ,до 15 град назад;
- опорна поверхня спинки стільця має висоту 300 плюс, мінус 20мм, ширину - 300мм і радіус кривизни горизонтальної площини - 400мм. Кут нахилу спинки у вертикальній площині регулюється в межах 0 плюс-мінус 30 градусів від вертикального положення. Відстань спинки від переднього краю сидіння регулюється в межах 260-400мм;
- робоче місце обладнане підставкою для ніг, що має ширину 300мм, глибину - 400мм. регулювання по висоті в межах до 150мм по куту нахилу опорної поверхні підставки - до 20 град. Поверхня підставки рифлена, бортик висотою 100мм по нижньому краї.

А.4.2 Розрахунок параметрів системи кондиціонування та вентиляції для приміщення офісу

Проведемо розрахунок для приміщення під №1, де розташовані робочі місця операторів зв'язку. У приміщенні відділу є джерела тепловиділення, тому необхідно визначити необхідні умови його вентилування.

Витрату повітря в приміщенні з додатковим тепловиділенням визначаємо по формулі:

$$L = \frac{Q_{над}}{c \times p(t_v - t_n)}, \text{ м}^3/\text{год}, \quad (\text{A.1})$$

де: $Q_{над}$ - надлишкове виділення тепла в робочому приміщенні, ккал/год.;

c - теплоємність повітря (0,237 ккал/кг);

p - обсягова вага повітря (1,226 кг/ м³);

t_v - температура витяжного повітря (30°С);

t_n - температура приточного повітря (20°С).

Розрахуємо надлишкове надходження тепла по формулі:

$$Q_{над} = Q_{уст} + Q_{пер} + Q_{осв} + Q_{ср}, \quad (\text{A.2})$$

де: $Q_{уст}$ - виділення тепла від устаткування;

$Q_{пер}$ - виділення тепла робітниками;

$Q_{осв}$ - надходження тепла від електричного освітлення;

$Q_{ср}$ - надходження тепла від сонячної радіації через вікна.

Визначимо виділення тепла від устаткування по формулі:

$$Q_{уст} = P \times K_a \times K_{\sigma} \times 860, \quad (\text{A.3})$$

де: P - сумарна потужність устаткування, кВт/год;

K_a - коефіцієнт установленної потужності (0,95);

K_b - коефіцієнт одночасної роботи (1,0).

Сумарне виділення теплової енергії в приміщенні становитиме:

$$Q_{уст} = [(9 \cdot 0,7) + (1 \cdot 2,7)] \cdot 0,95 \cdot 1 \cdot 860 = 7353 \text{ ккал/год.}$$

Визначимо виділення тепла від обслуговуючого персоналу за допомогою формули:

$$Q_{пер} = n \times g, \quad (A.4)$$

де: n - кількість працюючих;

g - кількість тепла, що виділяє один працівник за годину (100 ккал/год.)

Розрахуємо:

$$Q_{пер} = 9 \times 100 = 900 \text{ ккал/год.}$$

Визначимо надходження тепла від електричного освітлення по формулі:

$$Q_{осв} = E_m \cdot g_l \cdot S, \quad (A.5)$$

де: E_m - нормована освітленість для цієї зорової роботи приймаємо рівним 400 лк;

g_l - питома тепловиділення на 1 м² підлоги при 1 лк освітленості (для люмінесцентних ламп - 0,05 ккал/год.).

S - площа приміщення, м².

Розрахуємо:

$$Q_{осв} = 400 \cdot 0,05 \cdot 160,65 = 3213 \text{ ккал / год.}$$

Визначимо надходження тепла від сонячної радіації через вікна по формулі:

$$Q_{\text{ср}} = F \cdot g_2 \cdot K_{\text{осл}}, \quad (\text{A.6})$$

де: F - площа віконних прорізів (2,9 м²).

g_2 - кількість тепла, що надходить через 1 м² віконного прорізу (65 ккал/год.).

$K_{\text{осл}}$ - коефіцієнт ослаблення, приймаємо - 0,4.

Розрахуємо:

$$Q_{\text{ср}} = 2 \cdot 2,9 \cdot 65 \cdot 0,4 = 151 \text{ ккал/год.}$$

Визначимо кількість надлишкового тепла:

$$Q_{\text{над}} = 7353 + 900 + 3213 + 151 = 3845,3 \text{ ккал/год.}$$

Визначимо витрати повітря в приміщенні:

$$L = \frac{11617}{0,237 \cdot 1,226 \cdot (30 - 20)} = 3998 \text{ м}^3 / \text{год}$$

Існуюча в наявності система кондиціонування і вентилявання має продуктивність 4000 куб. м./годину, що задовольняє необхідним нормативам.

Параметри мікроклімату на робочих місцях регламентуються ДСН 3.3.5.042-99 «Санітарні норми мікроклімату виробничих приміщень». Відповідно доданих санітарних норм температура повітря, швидкість руху повітря і відносна вологість у холодні періоди року повинна складати 22-24 градуса по Цельсію, 0,1 метра в секунду і 40-60 % відповідно. У теплі періоди року температура повітря повинна складати 23-25 градусів Цельсія, рухливість повітря 0,1-0,2 метрів секунду, вологість 40-60 %. Температура може коливатися від 22 до 26 градусів Цельсія при збереженні всіх інших

параметрів мікроклімату. Вище зазначені норми цілком відповідають фактичним даним приміщення відділу маркетингу.

А.4.3 Розрахунок системи загального рівномірного освітлення з лампами розжарювання для приміщення, в якому використовуються зорові роботи високої точності

Розміри приміщення: довжина ($a=17$ м), ширина ($b=9,45$ м), висота ($h=3,2$ м). Приміщення має світлу побілку: коефіцієнт відбиття - $P_{\text{стелі}} = 70\%$, $P_{\text{стін}} = 50\%$. Висота робочих поверхонь (столів) $h_p = 0,7$ м. Для освітлення прийнято світильники типу УПМ-15, які підвищуються до стелі, відстань від світильника до стелі $h_c = 0,5$ м. Мінімальна освітленість за нормами $E=400$ лк.

1) Визначимо висоту підвісу світильників над підлогою

$$h_0 = H - h_c = 3,2 - 0,5 = 2,7 \text{ м}, \quad (\text{A.7})$$

Для світильників загального освітлення з лампами розжарювання потужністю до 200 Вт мінімальна висота підвісу над підлогою відповідно до СНіП П-4-79 повинна бути у межах 2,5 - 4,0 м, залежно від характеристики світильника. В нашому випадку це відповідає цій вимозі.

2) Визначимо висоту підвісу світильника над робочою поверхнею

$$h = h_0 - h_p = 2,7 - 0,7 = 2 \text{ м}, \quad (\text{A.8})$$

Рівномірність освітлення досягається при відповідному співвідношенні відстані між світильниками (L) і висоти їх підвісу (h).

3) Визначимо рекомендовану відстань між світильниками

$$L = 0,7h = 0,7 \cdot 3,2 = 2,24 \text{ м}, \quad (\text{A.9})$$

4) Розрахуємо необхідну кількість світильників

$$N = \frac{ab}{L^2} = \frac{17 \cdot 9,45}{2,24^2} \approx 32, \quad (\text{A.10})$$

Приймаємо 50 світильники, враховуючи розміри приміщення розміщуємо їх у три ряди по 10 штук.

5) Світловий потік лампи світильника визначається за формулою:

$$\Phi_{\lambda} = \frac{E \cdot K_3 \cdot S \cdot Z}{N \cdot n \cdot \eta} \text{ккал/год.}, \quad (\text{A.11})$$

де: E - нормативна освітленість, лк;

K_3 - коефіцієнт запасу, що враховує зниження освітленості в результаті забруднення та старіння ламп ;

S - площа приміщення, що освітлюється, м²;

Z - коефіцієнт нерівномірності освітлення для ламп розжарювання (=1,15);

N - кількість світильників;

n - кількість ламп у світильнику;

η - коефіцієнт використання світового потоку, який визначається за світлотехнічними таблицями залежно від показника приміщення (i) та коефіцієнтів відбиття стін та стелі.

6) Визначимо показник приміщення:

$$i = \frac{ab}{h(a+b)} = \frac{17 \cdot 9,45}{3,2 \cdot (17 + 9,45)} = 1,9, \quad (\text{A.12})$$

7) Знаходимо коефіцієнт використання ($\eta = 0,78$) для світильника УПМ-15 (при $i = 1,9$, $P_{\text{стелі}} = 70\%$, $P_{\text{стін}} = 50\%$)

8) Світловий потік одного світильника, а значить і лампи, оскільки за конструктивним виконанням у світильнику цього типу встановлена лише одна лампа, дорівнює:

$$\Phi = \frac{E \cdot S \cdot K_3}{N \cdot \eta} = \frac{400 \cdot 160,65 \cdot 0,7}{30 \cdot 0,78} = 1922,3_{\text{лм}} \quad (\text{A.13})$$

9) Обираємо лампу з трехполосним люмінофором (світлова отдача до 110 Лм/Вт), потужністю 20 Вт, світловий потік якої становить 2100 лм. Хоча це значення на 10% більше розрахованого, однак не перевищує встановлену норму ($-0\% < \Delta\Phi_{\text{л}} < +20\%$). Сумарна електрична потужність усіх світильників, встановлених у приміщенні становить:

$$\Sigma P_{\text{св}} = P \cdot N = 20 \cdot 30 = 600 \text{ Вт}, \quad (\text{A.14})$$

А.5 Пожежна безпека

Найважливішою умовою роботи будь-якого підприємства є дотримання правил пожежної охорони.

У основні міри для забезпечення пожежної безпеки визначає Інструкція про заходи пожежної безпеки для службових приміщень. Вона є обов'язковою для виконання всіма співробітниками.

В інструкції про засоби пожежної безпеки для службових приміщень забороняється:

- улаштовувати тимчасові електромережі, застосовувати саморобні плавкі вставки в запобіжниках, прокладати електричні проводи безпосередньо по пальній основі, експлуатувати світильники зі знятими ковпаками

- (розсіювачами), використовувати саморобні подовжувачі, що не відповідають вимогам Правил пристрою електроустановок;
- пристосовувати вимикачі, штепсельні розетки для підвішування одягу й інших предметів, обгортати електролампи і світильники, заклеювати ділянки електромережі пальною тканиною, папером;
 - використовувати побутові електрокип'ятильники, чайники без непальних підставок, залишати без нагляду включеними в електромережу кондиціонери, комп'ютери, рахункові і друкарські машинки і т.п.;
 - захищати підступи до засобів пожежогасіння, використовувати пожежні крани, рукави і пожежний інвентар не по призначенню, зберігати документи, різні матеріали, предмети й інвентар у шафах (нішах) інженерних комунікацій;
 - курити (крім спеціально відведених для цього адміністрацією місць, позначених написом «Місце для паління» і забезпечених урною чи попільницею з непального матеріалу), проводити зварювальні та інші вогневі роботи без оформлення відповідного дозволу, застосовувати легкозаймисті рідини.

По закінченню роботи необхідно:

- оглянути приміщення, переконатися у відсутності порушень, що можуть спричинити пожежу;
- відключити освітлення, електроживлення приладів і устаткування.

Електромережі, електроприлади і апаратура повинні експлуатуватися тільки у справному стані. У випадку виявлення ушкоджень електромереж, вимикачів, розеток ті інших електровиробів варто негайно відключити їх і вжити необхідних заходів до приведення їх в пожаробезпечний стан.

А.6 Безпека при надзвичайних ситуаціях на підприємстві

Об'єктом забезпечення на предмет визначення надзвичайних небезпек та їх наслідків є будівля. Однією із вірогідних загроз може бути раптове виникнення пожежу внаслідок короткого замикання в електромережах або розрядів статистичної електрики, що може привести до пошкодження і руйнування будівлі, устаткування, комунікацій, виділення токсичних продуктів горіння.

Будинок, в якому розташовано офіс повинен бути обладнаний мережею протипожежного водо забезпечення, установками виявлення та гасіння пожеж відповідно вимогам нормативно-технічних документів. Кожний працівник повинен чітко знати та виконувати вимоги ППБ та протиаварійний режим на об'єкті, уміти користуватися наявними вогнегасниками, іншими первинними засобами пожежогасіння і знати місце їхнього перебування.

Меблі й устаткування повинні розміщатися таким чином, щоб забезпечувався вільний евакуаційний прохід до дверей виходу з приміщення (шириною не менше 1 м). Евакуаційні шляхи і виходи необхідно постійно держати вільними, нічим не захащувати. Засоби протипожежного захисту в приміщеннях потрібні триматися у справному стані.

У випадку виявлення пожежі слід:

- негайно повідомити державну пожежну охорону за телефоном «101», вказати при цьому адресу, кількість поверхів, місце виникнення пожежі, наявність людей, своє прізвище;
- повідомити про пожежу керівництву, а в нічний час черговому охоронцю.

У разі можливості почати гасіння пожежі наявними засобами, організувати зустріч пожежних підрозділів.

При виникненні пожежі у початковій стадії її розвитку випромінюється тепло, токсичні продукти згоряння, імовірні руйнування будівельних споруд. Тому слід як можна швидше провести евакуацію людей із палаючої будівлі. Показником ефективності евакуації є час, протягом якого працівники можуть при необхідності залишити окремі приміщення і будівлю в цілому. Безпека евакуації досягається тоді, коли час евакуації не перевищує час настання критичної фази розвитку пожежі, тобто часу від початку пожежі до досягнення граничних для людини впливів факторів пожежі (критичних температур, ступені задимлення, зниження концентрації кисню и т.п.). Число евакуаційних виходів повинно бути не менш двох. Вони повинні розташовуватися розосереджено. Мінімальна відстань l між найбільш віддаленими один від одного евакуаційними виходами із приміщення визначається за формулою:

$$l=1,5\sqrt{P} \quad , \quad (A.15)$$

де: P – периметр приміщення, м.

Двері на шляхах евакуації повинні відкриватися у напрямку виходу із будівлі (приміщення).

У кожному приміщенні на видному місці повинен бути вивішений план евакуації при пожежі (рисунок А.1).

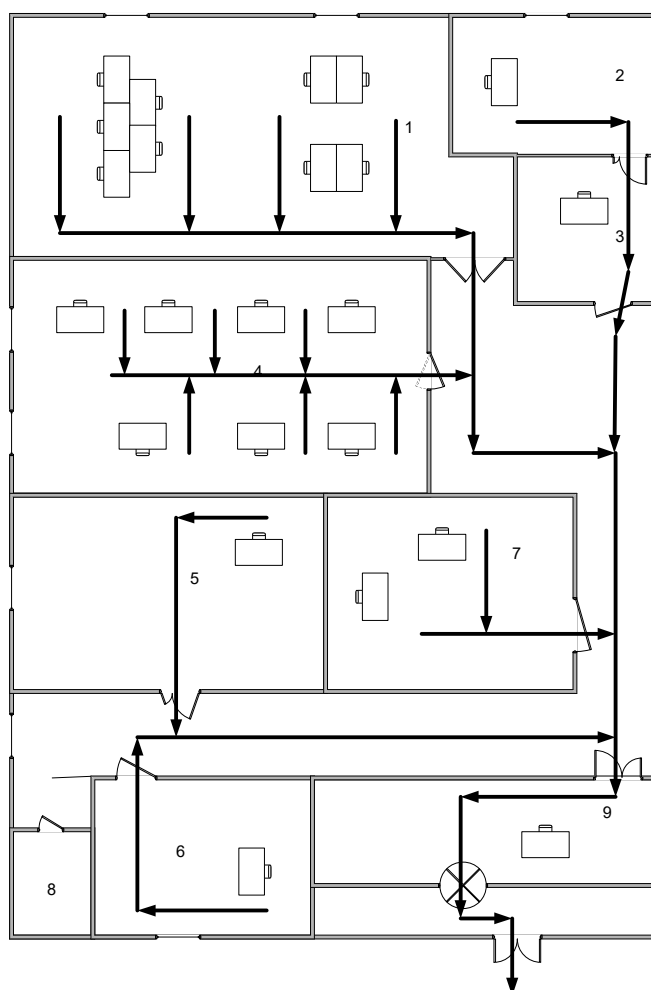


Рисунок А.1 – План евакуації при пожежі

При пожежі обов'язково необхідно враховувати небезпечні чинники і механізм їх дії на людину.

При виникненні пожежі, після виклику пожежної охорони, необхідно попередити про це усіх, хто знаходиться поруч, після чого евакуюватися самому і по можливості допомогти евакуюватися іншим, особливо особам літнього віку і дітям, попереджаючи при цьому виникнення паніки. З метою обмеження циркуляції повітря, яке здатне збільшувати швидкість горіння, покидаючи приміщення, закрийте за собою усі двері, якщо це можливо. Перекрийте газ і відключіть електрику, але слід пам'ятати, що в першу чергу завжди необхідно покинути активну зону горіння, тобто місце в районі

пожежі, де є присутнім задимлення і підвищена температура. Якщо пожежа виникла в приміщенні над вами і безпосередньої загрози для вас не спостерігається, то бажано виконати заходи по зниженню можливих втрат від води яку проливають при гасінні пожежі. Для цього необхідно відключити всі електроприводи та прикрити їх поліетиленовою плівкою. Значно гірше, якщо пожежа виникла в приміщенні над вами – потрібно оцінити обстановку и якщо є впевненість, що ще не має сильної задимленості з високою температурою, потрібно негайно покинути приміщення, рухаючись до виходу по коридорам і сходовим кліткам.

Користуватися ліфтом категорично забороняється, за винятком ліфтів, які спеціально призначені для транспортування пожежної охорони. Шахта ліфта є шляхом для поширення диму і отруйних продуктів горіння, до того ж при пожежі ліфт часто відключають і можна опинитися в пастці при пожежі.

Якщо ви знаходитесь в приміщенні де немає пожежі, але відрізани вогнем, димом, високою температурою від головних шляхів евакуації, то в першу чергу необхідно заважити доступу диму и продуктів горіння в це приміщення. Для чого необхідно негайно закрити усі щілини у дверях та під ними змоченими водою ганчірками, рушниками, робочими халатами та інш.

Якщо приміщення все ж заповнено димом, необхідно підповзти до вікна, закрити при цьому рот та ніс змоченою тканиною, яка грає роль фільтру та в певної мірі захищає від продуктів горіння.

Рухатись у задимленій зоні поповзом або максимально пригнувшись, необхідно тому що більшість нагрітих газоподібних отруйних речовин та дим збираються у верхній зоні приміщення, окрім цього, в приміщенні при горінні температура на рівні очей людини у 6 разів вище за температуру на рівні полу, до того ж внизу завжди зберігається більша концентрація кисню. Коли ви опинились біля вікна трохи відкрийте його та дихайте через щілину, очікуючи прибуття пожежників. При їх прибутті негайно зверніть на себе увагу. Ніколи не стрибайте через вікно без відомої на це необхідності (кожний другий стрибок з 4-г поверху при пожежі смертельний).

ДОДАТОК Б
Специфікація

ДОДАТОК А

Лістинг налаштувань платформи Arduino

```

#include <EtherCard.h> //Підключення бібліотеки для роботи з
Ethernetмодулем

#include <EEPROM.h> //Підключення бібліотеки для роботи з
регiстрами #include <math.h> //Підключення бібліотеки для роботи с мат.
моделями static byte mymac[] = {0x5A,0x5A,0x5A,0x5A,0x5A,0x5A
}; //Задання Масадреси

static byte myip[] = {192,168,0,222 }; //Задання ip-
адреси byte Ethernet::buffer[1150]; //Виділення буферної
пам'яті до сайту BufferFiller bfill; //Завантажуємо веб-
інтерфейс int LedPins[] = {2,3,4,5}; //Масив виходів
boolean PinStatus[3]; // З 0 до 3 - Перевірочні
коефіцієнти double Thermister(int RawADC) //Внутрішній алгоритм
обробки температури
{
double
Temp;

Temp = log(((10240000/RawADC) - 10000));
Temp = 1 / (0.001129148 + (0.000234125 + (0.0000000876741 *
Temp * Temp
)) * Temp );
Temp = Temp - 273.15; // Convert
Kelvin to Celcius return Temp; } int
InputParam1;

const char http_OK[] PROGMEM = //Стилiзація
"HTTP/1.0 200 OK\r\n"
"Content-Type:
text/html\r\n" "Pragma: no-
cache\r\n\r\n";

const char http_Found[] PROGMEM = "HTTP/1.0 302 Found\r\n"
"Location: /\r\n\r\n";

const char http_Unauthorized[] PROGMEM =
"HTTP/1.0 401 Unauthorized\r\n"
"Content-Type:
text/html\r\n\r\n" "<h1>401
Unauthorized</h1>"; void
homePage()
{

```

```

        int sensor1 = (Thermister(analogRead(A0))); //Прив'язка даних з
датчика bfill.emit_p(PSTR("$F" " <title>ArduinoPIN Webserver</title>"
"<font color=\\"#333333\\" face=\\"sans-serif\\" size=\\"5\\">Heater: <a
href=\\"?ArduinoPIN4=$F\\">$F</a></font><br />" "<font color=\\"#333333\\"
face=\\"sans-serif\\" size=\\"5\\">Cooler: <a
href=\\"?ArduinoPIN5=$F\\">$F</a></font><br />" "<font color=\\"#333333\\"
face=\\"sans-serif\\" size=\\"5\\">Lamp: <a
href=\\"?ArduinoPIN6=$F\\">$F</a></font><br
/>"), http_OK,
        PinStatus[2]?PSTR("off"):PSTR("on"),
PinStatus[2]?PSTR("<font face=\\"sans-serif\\"
color=\\"green\\">ON</font>"):PSTR("<font
color=\\"red\\">OFF</font>"),
        PinStatus[3]?PSTR("off"):PSTR("on"),
PinStatus[3]?PSTR("<font face=\\"sans-serif\\"
color=\\"green\\">ON</font>"):PSTR("<font
color=\\"red\\">OFF</font>"),
        PinStatus[4]?PSTR("off"):PSTR("on"),
        PinStatus[4]?PSTR("<font face=\\"sans-serif\\"
color=\\"green\\">ON</font>"):PSTR("<font
color=\\"red\\">OFF</font>")); //Створюємо чотири перемикача з
двома станами bfill.emit_p(PSTR(
        //"<meta http-equiv='refresh' content='5'>"
        "<font color=\\"#333333\\" face=\\"sans-serif\\"
size=\\"5\\">Temperature:
        ""$D"" &degC"), sensor1); //Виводимо дані з датчика температур
    }
    void setup() //Директива, що виконується лише один раз
    {
        Serial.begin(9600); //Виділяємо пам'ять під компорт
        if (ether.begin(sizeof Ethernet::buffer, mymac, 10) ==
0); //Перенаправляємо вихід
        if (!ether.dhcpSetup());
        ether.printIp("My Router IP: ", ether.myip); //Виводимо ip-
адресу ether.staticSetup(myip); //Заміняємо динамічну адресу статичною
        ether.printIp("My SET IP: ", ether.myip); //Виводимо ip-адресу for(int
i = 0; i <= 3; i++) //Організовуємо цикл, який перебирає послідовно
входи {
            pinMode(LedPins[i], OUTPUT); //Описуємо їх як виходи
            PinStatus[i] = EEPROM.read(i); //Зчитуємо у регістр
            digitalWrite(LedPins[i], PinStatus[i]); //Задаємо їм нульовий
рівень напруги

```

```

    }

}

void loop()//Нескінченний цикл
{

delay(
1);

    word len = ether.packetReceive(); //Перевіряємо цілісність
    пакетів word pos = ether.packetLoop(len);
    if (pos)//Перевіряємо домашню сторінку на оновлення
    {
        bfill = ether.tcpOffset();
        char *data = (char *)
Ethernet::buffer + pos; if (strncmp("GET
/", data, 5) != 0) {
bfill.emit_p(http_Unauthorized);
        } else {
data += 5; if
(data[0] == ' ')
{ homePage();
        }
        else if (strncmp("?ArduinoPIN2=on ", data, 16) == 0)//Умова на
перехід до стану 1
        {
            PinStatus[0] = true;
            digitalWrite(LedPins[0],PinStatus[0]);
            EEPROM.write(0,PinStatus[0]); // Записуємо до EEPROM №0, стан
LedPins[0]. bfill.emit_p(http_Found);
        }
        else if (strncmp("?ArduinoPIN3=on ", data, 16) == 0)
        {
            PinStatus[1] = true;
            digitalWrite(LedPins[1],PinStatus[1]);
            EEPROM.write(1,PinStatus[1]);
bfill.emit_p(http_Found);
        }
        else if (strncmp("?ArduinoPIN4=on ", data, 16) == 0)
        {
            PinStatus[2] = true;

```

```

        digitalWrite(LedPins[2],PinStatus[2]);
EEPROM.write(2,PinStatus[2]);
bfill.emit_p(http_Found);
    }
    else if (strncmp("?ArduinoPIN2=off ", data, 17) == 0)//Умова на
перехід до стану 0
    {
        PinStatus[0] = false;
        digitalWrite(LedPins[0],PinStatus[0]);
EEPROM.write(0,PinStatus[0]);
bfill.emit_p(http_Found);
    }
    else if (strncmp("?ArduinoPIN3=off ", data, 17) == 0)
    {
        PinStatus[1] = false;
        digitalWrite(LedPins[1],PinStatus[1]);
EEPROM.write(1,PinStatus[1]);
bfill.emit_p(http_Found);
    }
    else if (strncmp("?ArduinoPIN4=off ", data, 17) == 0)
    {
        PinStatus[2] = false;
        digitalWrite(LedPins[2],PinStatus[2]);
EEPROM.write(2,PinStatus[2]);
bfill.emit_p(http_Found);
    }

e
l
s
e
{
    bfill.emit_p(http_Unauthorized);//Перехід у разі помилки дії
}

ether.httpServerReply(bfill.position()); // send http response
}
}

```

ДОДАТОК В

Лістинг налаштувань мікро-комп'ютера Raspberry pi-3

```

import
Rpi.GPIO as GPIO
import time
import datetime

GPIO.setmode(GPIO.BCM)

GPIO_PIR = 5

GPIO_LED = 6

GPIO.setup(GPIO_PIR,GPIO.IN)

Current_State = 0

Previous_State = 0

GPIO.setup(GPIO_LED,GPIO.OUT)

GPIO.output(6, False)
try:
    while
GPIO.input(GPIO_PIR)==1:

    Current_State = 0

    while True :

        Current_State = GPIO.input(GPIO_PIR)

        if Current_State==1 and Previous_State==0:

            print 'Srabotalo, yra'

```

```

f = open('/var/www/data/motion/motion.txt', 'w')

f.write("1")

f.close

f = open('/var/www/data/motion/motion.txt')

f.read()
f.close

GPIO.output(6, True)

Previous_State=1

elif Current_State==0 and Previous_State==1:

f = open('/var/www/data/motion/motion.txt', 'w')

f.write("0")

f.close

f = open('/var/www/data/motion/motion.txt')

f.read()

f.close

GPIO.output(6, False)

print " Ready"

Previous_State=0

```

```

        time.sleep(0.01) except KeyboardInterrupt:
            print " Quit"
            GPIO.cleanup()


import RPi.GPIO as
GPIO import time
    GPIO.setmode(GPIO.B
CM)
        GPIO_PIR = 19
        GPIO.setup(GPIO_PIR
,GPIO.OUT)
GPIO.output(19, False)
try:      try:
while (1>0):
            f =
open('/var/www/data/GPIOState/state1.txt')          val
= f.read(1)

f.close          if
(val == '1'):

GPIO.output(GPIO_PIR, True)
time.sleep(0.5)          else:

GPIO.output(GPIO_PIR, False)
time.sleep(0.5)      except IOError:
            print("No
file") except
KeyboardInterrupt:

```

```

GPIO.output(GPIO_PIR, False)
print " Quit"

import RPi.GPIO as
GPIO import time
GPIO.setmode(GPIO.B
CM)
GPIO_PIR = 20
GPIO.setup(GPIO_PIR
,GPIO.OUT)
GPIO.output(20, False)
try:    try:
        while
(1>0):
            f =
open('/var/www/data/GPIOState/state2.txt')    val
= f.read(1)

f.close    if
(val == '1'):
        GPIO.output(GPIO_PIR, True)

time.sleep(0.5)    else:

GPIO.output(GPIO_PIR, False)
time.sleep(0.5)    except IOError:
        print("No
file") except
KeyboardInterrupt:
GPIO.output(GPIO_PIR, False)
print " Quit"

import RPi.GPIO as
GPIO import time
GPIO.setmode(GPIO.B
CM)
GPIO_PIR = 21

```

```

        GPIO.setup(GPIO_PIR
,GPIO.OUT)
GPIO.output(21, False)
try:    try:
while (1>0):
            f =
open('/var/www/data/GPIOSState/state3.txt')           val
= f.read(1)

f.close            if
(val == '1'):

GPIO.output(GPIO_PIR, True)
time.sleep(0.5)            else:

GPIO.output(GPIO_PIR, False)
time.sleep(0.5)    except IOError:
        print("No
file") except
KeyboardInterrupt:
GPIO.output(GPIO_PIR, False)
print " Quit"

import RPi.GPIO as
GPIO import time
GPIO.setmode(GPIO.BCM)
        GPIO_PIR = 26
        GPIO.setup(GPIO_PIR
,GPIO.OUT)
GPIO.output(26, False)
try:    try:
while (1>0):
            f =
open('/var/www/data/GPIOSState/state4.txt')           val
= f.read(1)

f.close            if
(val == '1'):

```

```
GPIO.output(GPIO_PIR, True)
time.sleep(0.5)                else:

GPIO.output(GPIO_PIR, False)
time.sleep(0.5)                except IOError:
    print("No
file") except
KeyboardInterrupt:
GPIO.output(GPIO_PIR, False)
print " Quit"
```