

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАДИМА ГЕТЬМАНА
Навчально-науковий інститут
«Інститут інформаційних технологій в економіці»
Кафедра інформаційних систем в економіці**

галузь знань	12 Інформаційні технології
спеціальність	122 Комп'ютерні науки
освітня програма	Інформаційні управляючі системи і технології

Форма навчання: денна

КВАЛІФІКАЦІЙНА МАГІСТЕРСЬКА РОБОТА

**на тему: “АВТОМАТИЗОВАНА СИСТЕМА БЕЗПЕКИ ПРИМІЩЕННЯ НА
ПЛАТФОРМІ ARDUINO”**

Здобувача Алешкова Анатолія Валерійовича

Науковий керівник: д.е.н., проф.

_____ Устенко С.В.

**Робота допущена до захисту перед
екзаменаційною комісією з атестації
здобувачів вищої освіти (ЕК)**

в.о. завідувача кафедри: к.е.н., доцент

_____ Тішков Б.О.

Київ 2023

АНОТАЦІЯ

Кваліфікаційна магістерська робота

здобувача 6 курсу

Навчально-наукового інституту «Інститут інформаційних технологій в економіці» Алешкова Анатолія Валерійовича,

виконана на тему:

«Автоматизована система безпеки приміщення на платформі Arduino»

Київ: кафедра інформаційних систем в економіці, 2023 р.

Кваліфікаційна магістерська робота присвячена проектуванню та розробці системи безпеки приміщення на підприємстві. В результаті проектування отримано оптимальну структурну модель для запобігання несанкціонованого доступу до приміщень, зокрема до комерційних та спеціалізованих приміщень.

Кваліфікаційна магістерська робота складається з трьох розділів, логічно пов'язаних між собою.

У першому розділі КРМ було проведено дослідження предметної області та аналіз аналогічних систем. Це дозволило отримати необхідні відомості та огляд існуючих рішень в цій галузі.

У другому розділі були визначені структурні та функціональні особливості автоматизованої системи безпеки приміщення. Також були розглянуті методи і моделі побудови систем безпеки. Цей аналіз сприяв визначенню оптимальних рішень для розробки системи.

У третьому розділі було розроблено проектування автоматизованої системи безпеки приміщення, зокрема розроблена архітектура системи, яка включила комплект необхідних датчиків, модуль GSM, Arduino IDE, п'єзоелектричний випромінювач звуку, перетворювач та інші пристрої. Крім того, було розроблене програмне забезпечення і проведено його тестування.

Як результат виконання КРМ була розроблена автоматизована система безпеки приміщення. Ця система повністю відповідає поставленим вимогам і пройшла успішне

тестування. Вона забезпечує безпеку до приміщення і сприяє забезпеченню безпеки спец-приміщень підприємств.

РЕФЕРАТ

Кваліфікаційна магістерська робота містить 82 сторінки, 8 таблиць, 19 рисунків, список літератури з 40 найменувань, 1 додаток.

“ Автоматизована система безпеки приміщення на платформі Arduino ”

Перелік ключових слів: система безпеки, ідентифікація, автентифікація, безпека, інформаційна безпека, RFID, датчик руху, модуль GSM, Arduino IDE, п'єзоелектричний випромінювач звуку, перетворювач та інші пристрої.

Об'єктом дослідження є система безпеки приміщення.

Предметом дослідження є методології проектування системи безпеки приміщення на платформі Arduino, яка відповідає основним вимогам та забезпечує ефективний та надійний захист зон та об'єктів приміщення.

Мета полягає в розробці та імплементації системи забезпечення безпеки приміщення на підприємстві.

Результати, досягнуті в процесі роботи: розроблена система безпеки приміщення на підприємстві на платформі Arduino та створено реальний макет і проведено його тестування.

Одержані результати можуть бути впроваджені на охоронних підприємствах та для охорони і безпеки власних приміщень.

ЗМІСТ

ВСТУП	5
РОЗДІЛ 1. ТЕОРЕТИЧНИЙ РОЗДІЛ. ДОСЛІДЖЕННЯ ТА АНАЛІЗ ПІДХОДІВ ДО ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ІНФОРМАЦІЙНИХ УПРАВЛЯЮЧИХ СИСТЕМ	11
1.1. Аналіз існуючих інформаційних систем. Дослідження предметної області	11
1.2. Аналіз автоматизованих систем безпеки приміщень	15
РОЗДІЛ 2. ОГЛЯД ТА ВИБІР АРХІТЕКТУРНОГО ЗАБЕЗПЕЧЕННЯ ПРОЄКТУ	24
2.1. Arduino – як платформа проєктування	24
2.2. Середовище розробки Arduino IDE	29
2.3. Програмування Arduino	32
2.4. Плати Arduino	35
2.5. Датчики та пристрої	46
2.6. Висновки до розділу 2	54
РОЗДІЛ 3 КОНСТРУКТИВНИЙ РОЗДІЛ.....	55
3.1. Проєктування та розробка апаратно-програмної частини системи	55
3.2. Тестування системи	68
3.3. Керівництво користувача.....	70
3.4. Технічне забезпечення системи.....	74
3.5. Реалізація автоматизованої системи безпеки приміщення.....	76
ВИСНОВКИ	77
ДОДАТОК	79

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ СИМВОЛІВ, ОДИНИЦЬ ВЕЛИЧИН І ТЕРМІНІВ

IAM – Identity and Access Management (Керування ідентифікацією та доступом)

2FA – Two-Factor Authentication (двоетапна аутентифікація)

MFA – Multi-Factor Authentication (багатофакторна аутентифікація)

IUC – інформаційна управляюча система

BLE – Bluetooth Low Energy (Bluetooth з низьким енергоспоживанням)

NFC – Near Field Communication (технологія зв'язку на невеликих відстанях)

BPMN – Business Process Model and Notation (система умовних позначень для моделювання бізнес-процесів.)

RFID – Radio Frequency IDentification (Радіочастотна ідентифікація)

ІЧ – інфрачервоний

ТЗ – технічне забезпечення

V/B – вольти

Ом – оми

LED – light-emitting diode (світловипромінюючий діод)

Ф – фарадеї

А – амперы

Гц/Hz – герцы

Кб – кілобайты

ВСТУП

Актуальність теми. Тема "Автоматизована система безпеки приміщення на платформі Arduino" є актуальною в сучасному світі, оскільки безпека і захист мають велике значення для людей і власності. Застосування платформи Arduino у системах безпеки набуває все більшого попиту завдяки її доступності, гнучкості та можливостям інтеграції з іншими пристроями.

Автоматизована система безпеки на базі Arduino може включати різноманітні компоненти, такі як датчики руху, датчики вогню, датчики газу, камери спостереження, датчики відкриття дверей тощо. Вона здатна виявляти небезпеку, спрацьовувати в разі виникнення загрози та передавати відповідну інформацію адміністраторам або відповідним службам безпеки.

Однією з переваг використання платформи Arduino у системах безпеки є можливість розробки та налаштування власних функцій і алгоритмів безпеки. Завдяки широкому спектру доступних модулів і компонентів, можна створити систему, яка відповідає конкретним потребам і вимогам користувача.

Крім того, зростаючий інтерес до "розумних" будинків та "інтернету речей" створює попит на розробку інтегрованих систем безпеки, які можуть бути керовані з використанням смартфонів або віддаленого доступу. КМР є актуальною з таких причин:

- **Гнучкість та розширюваність:** Платформа Arduino надає можливість розробки системи безпеки, яка може бути легко налаштована і розширена відповідно до потреб конкретного приміщення. Існує велика кількість датчиків, модулів та компонентів, які можна інтегрувати в систему безпеки з використанням платформи Arduino.
- **Доступність:** Платформа Arduino відома своєю доступністю та низькою вартістю, що робить її привабливою для широкого кола розробників і

ентузіастів. Це дозволяє більшій кількості людей використовувати платформу Arduino для створення власних систем безпеки без значних витрат.

- Інтеграція з іншими системами: Arduino може бути легко інтегрована з іншими системами автоматизації та "розумного" будинку. Це дає можливість створювати комплексні системи безпеки, які взаємодіють з іншими пристроями і системами, наприклад, системами освітлення, камерами спостереження, сенсорами, замками тощо.
- Персоналізація та індивідуальний підхід: Використання платформи Arduino дозволяє розробникам налаштовувати систему безпеки відповідно до власних потреб і вимог. Вони можуть вибирати та налаштовувати конкретні компоненти і функції, що робить систему більш ефективною і адаптованою до конкретного приміщення.
- Постійний розвиток: Платформа Arduino продовжує розвиватися, і нові моделі та компоненти постійно стають доступними. Це дозволяє використовувати останні досягнення технологій у створенні автоматизованих систем безпеки на платформі Arduino.

Мета і задачі дослідження.

Метою дослідження є вивчення та розробка індивідуальної системи безпеки, за власним трендом, яка базується на використанні платформи Arduino. Головною метою є створення ефективної та гнучкої системи, яка забезпечує безпеку приміщення та його мешканців, шляхом автоматизації різноманітних функцій і реагування на потенційні небезпеки, також система буде спрощеною для широкого та масового використання.

Задачі дослідження включають:

- Аналіз потреб і вимог безпеки приміщення: Ретельне дослідження та аналіз потреб безпеки різних типів приміщень (наприклад, житлові будинки, офіси, комерційні приміщення) для визначення основних вимог і функцій, які має виконувати автоматизована система безпеки.
- Вивчення можливостей платформи Arduino: Ознайомлення з функціональністю, характеристиками та можливостями платформи Arduino для створення системи

безпеки. Визначення технічних обмежень та можливостей для інтеграції різних модулів та датчиків безпеки.

- Розробка алгоритмів та програмного забезпечення: Розробка алгоритмів та програмного забезпечення для автоматизованої системи безпеки, зокрема для виявлення небезпечних ситуацій, виявлення вторгнень, контролю доступу, оповіщення та реагування на них. Розробка інтерфейсу користувача для налаштування та моніторингу системи.
- Реалізація та тестування системи безпеки: Фізична реалізація системи безпеки з використанням платформи Arduino та необхідних модулів та датчиків. Проведення тестування та аналіз ефективності системи в різних умовах та ситуаціях.
- Оцінка безпеки та аналіз результатів: Оцінка ефективності, надійності та безпеки розробленої системи. Аналіз результатів тестування, виявлення можливих проблем та пропозиції щодо подальшого вдосконалення системи безпеки.
- Отримані результати дослідження будуть використані для покращення безпеки приміщень, впровадження інноваційних технологій та автоматизації у сфері безпеки, а також для подальших наукових досліджень та розвитку в цій галузі.

Об'єктом дослідження є система безпеки в приміщенні, зокрема розробка та впровадження системи на базі Arduino, яка забезпечуватиме надійний, ефективний та спрощений варіант системи контролю безпеки в різних зонах приміщення комерційного та спеціалізованого типу.

Предметом дослідження є методології проєктування системи безпеки приміщення на платформі Arduino, яка відповідає основним вимогам та забезпечує ефективний та надійний захист зон та об'єктів приміщення. Головним об'єктом дослідження є розробка системи для приміщень, такі як житлові будинки, офіси, комерційні приміщення тощо. Предметом дослідження також є розробка та оцінка ефективності автоматизованої системи безпеки, яка забезпечує контроль, виявлення та реагування на потенційні небезпеки та небажані події, забезпечуючи безпеку приміщення та його мешканців.

Методи дослідження:

- Широкий аналіз літературних джерел та спеціалізованої літератури для отримання теоретичних знань та розуміння побудови систем безпеки на основі платформи Arduino можна провести аналіз наукової літератури, наукових статей, журналів та інших джерел, що стосуються автоматизованих систем безпеки та Arduino.
- Експериментальні дослідження: Для визначення ефективності та функціональності системи безпеки можна провести ряд експериментів. Це може включати встановлення та налаштування сенсорів (таких як датчики руху, диму, газу тощо) на платформі Arduino, проведення тестових сценаріїв, симуляцію небезпечних ситуацій та аналіз реакції системи на них.
- Конструювання прототипу: Одним із методів дослідження може бути розробка та конструювання прототипу автоматизованої системи безпеки на основі платформи Arduino. Це включає в себе вибір необхідних компонентів, збірку та програмування пристрою, а також тестування його роботи в реальних умовах.
- Опитування та анкетування: Для збору даних про задоволення користувачів системи безпеки на платформі Arduino можна використовувати опитування або анкетування. Це дозволяє отримати відгуки, пропозиції та рекомендації від користувачів, які допоможуть вдосконалити систему.
- Математичне моделювання: Для аналізу роботи системи безпеки можна використовувати математичні моделі, що дозволяють оцінити ймовірність виявлення небезпеки, швидкість реагування системи та інші характеристики.

Ці методи дослідження здійснюють комплексний аналіз та оцінку автоматизованої системи безпеки на платформі Arduino з різних аспектів, включаючи теоретичний, практичний та користувацький підходи.

Отримані результати в КМР з автоматизованої системи безпеки приміщення на платформі Arduino мають практичне значення, оскільки вони відкривають можливості для застосування розробленої системи в різних сферах діяльності, де важливим є

контроль та управління доступом до приміщень або обмеженого доступу до обладнання.

Отже, розроблена система безпеки приміщення на платформі Arduino стає ефективним інструментом для забезпечення безпеки в різних сферах діяльності. Вона відкриває перспективи для впровадження цієї технології в різних галузях, сприяючи покращенню безпеки та ефективності роботи.

Особистий внесок магістранта

Основні положення та результати цієї роботи були досягнуті автором самостійно. Розроблений пристрій ґрунтується на принципах, які вже використовуються в подібних системах. Проте, за допомогою використання різних груп сенсорів, автор зміг досягти зниження вартості пристрою та збільшення швидкості отримання результатів.

Було проведено дослідницьку роботу, систематичний аналіз літературних джерел і наукових досліджень, пов'язаних з автоматизованими системами безпеки та платформою Arduino. Виявлення найбільш актуальних методів та технологій, що застосовуються в області програмування, і внесення їх у роботу. Розробка концепції: Розроблення концептуальної моделі автоматизованої системи безпеки на платформі Arduino, враховуючи специфічні потреби та вимоги для захисту приміщення. Визначення необхідних компонентів, алгоритмів та функціональні можливості системи. Проектування та реалізація. Розроблення детального плану проекту, включаючи схему з'єднань, вибір сенсорів та пристроїв, програмування мікроконтролера Arduino та створення необхідного програмного забезпечення. Тестування та оцінка: Проведення експериментального тесту, для перевірки функціональності та ефективності системи безпеки. Аналіз результатів та висновки.

РОЗДІЛ І. ТЕОРЕТИЧНИЙ РОЗДІЛ.

ДОСЛІДЖЕННЯ ТА АНАЛІЗ ПІДХОДІВ ДО ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ІНФОРМАЦІЙНИХ УПРАВЛЯЮЧИХ СИСТЕМ

1.1. Дослідження предметної області

Забезпечення безпеки і захисту приміщень має велике значення для захисту життя та майна. Дослідження в цій області дозволить виявити ефективні способи використання платформи Arduino для створення автоматизованих систем безпеки та розробки нових рішень для захисту приміщень.

Метою дослідження є вивчення можливостей платформи Arduino у контексті систем безпеки приміщень, а також визначення оптимальних технологій, компонентів та методів реалізації системи. Дослідження передбачає вивчення важливих аспектів безпеки приміщень, аналіз існуючих систем безпеки, а також дослідження можливостей платформи Arduino і використання різних технологій та компонентів.

Дослідження включає:

- Аналіз загроз безпеці приміщень: проведення огляду різних типів загроз, з якими можуть стикатися приміщення, таких як пожежа, крадіжка, несанкціонований доступ і т.д. Визначення потенційних наслідків та ризиків, пов'язаних з цими загрозами.
- Огляд існуючих систем безпеки: аналіз існуючих рішень та систем безпеки, що використовуються в приміщеннях. Вивчення їх принципів роботи, переваг і недоліків, а також визначення можливостей для покращення та вдосконалення.
- Вивчення платформи Arduino: розгляд характеристик, можливостей програмування та компонентів, які надає платформа Arduino. Дослідження

існуючих проектів та рішень, реалізованих на базі Arduino у контексті систем безпеки.

- Вибір технологій та компонентів: визначення оптимальних технологій, сенсорів, модулів зв'язку та інших компонентів, які можна використовувати для створення автоматизованої системи безпеки на платформі Arduino.
- Розробка системи безпеки на платформі Arduino: проектування архітектури системи, написання програмного забезпечення, інтеграція компонентів та проведення тестування системи.
- Оцінка ефективності системи: проведення випробувань, аналіз результатів та оцінка ефективності розробленої автоматизованої системи безпеки. Визначення переваг, недоліків та можливостей для подальшого вдосконалення.

В результаті дослідження планується отримати практичні рекомендації щодо реалізації автоматизованої системи безпеки приміщення на платформі Arduino, а також виявити переваги цього підходу в порівнянні з існуючими системами безпеки.

На сьогоднішній день інформаційні технології стають все більш поширеними і використовуються в різних сферах життя, не лише для оптимізації виробничих та бізнес-процесів, але й у побуті людей. Прикладом таких технологій є розумні побутові пристрої, які можна керувати за допомогою смартфона, такі як кавоварки,пилососи, музичні пристрої тощо. Ці пристрої належать до концепції Інтернету речей (IoT).

Інтернет речей (IoT) - це мережа фізичних об'єктів, які мають вбудовані технології для взаємодії з оточуючим середовищем, передачі інформації про свій стан та отримання даних ззовні. Об'єднання всіх об'єктів IoT в одному приміщенні створює єдину систему, відому як "система безпеки". Автоматизована система безпеки приміщення є одним із пріоритетних напрямків розвитку безпеки майбутнього, яка охоплює управління приладами та моніторинг середовища. Технології, що використовуються в такій системі, дозволяють об'єднати управління всіма пристроями в приміщенні через один інтерфейс, такий як мобільний телефон, планшет, комп'ютер або сенсорна панель, що замінює потребу в керуванні з різних пультів дистанційного керування.

Однак, такі системи можуть застосовуватися не лише для житлових приміщень, квартир або будинків, але й для різних інших об'єктів, таких як офіси, склади, магазини, парковки, автомобілі та інше.

Розумні системи розробляються для різних цілей і задач, пов'язаних з моніторингом, дистанційним керуванням та забезпеченням безпеки об'єктів та приміщень. Приміщення можуть бути будь-якими будівлями, житловими або комерційними просторами, які потребують моніторингу різних параметрів, таких як температура, вологість, рівень газу або наявність диму, а також можуть бути керованими через Інтернет. Будь-який параметр приміщення, який можна виміряти, може бути об'єктом моніторингу.

Об'єкти включають в себе будь-які вузли приміщень, виробничі площі або об'єкти зберігання, які потребують збереження певних умов функціонування. Моніторинг об'єктів передбачає контроль зміни змінних параметрів, таких як температурні режими, вологість, енергопостачання тощо.

Залежно від поставлених завдань, розумні системи можуть автоматизувати дії, що означає, що система може самостійно виконувати запуск, вимкнення, зміну необхідних параметрів або функції моніторингу, які можуть надсилати необхідні повідомлення про контрольовані параметри. Технічні можливості розумних систем практично необмежені, що дозволяє не тільки контролювати необхідні параметри, але й управляти будь-якими механічними об'єктами, до яких можна отримати доступ вручну.

Розумні системи дозволяють контролювати та управляти різними параметрами, такими як управління енергією, контроль доступу, системи сповіщення, системи безпеки, протипожежна безпека, контроль протікання, дистанційне управління, моніторинг систем в будинку та технічних вузлах, включаючи каналізацію, опалення, температуру труб, кондиціонування, електропостачання, освітлення, водопостачання та газопостачання.

З технічного погляду, автоматизовані системи безпеки є складним програмно-апаратним комплексом, який виконує певні програмовані дії у відповідь на різні події, такі як зміна параметрів або відхилення від заданих значень. У системі безпеки приміщення зазвичай використовуються системи моніторингу, управління та безпеки, які працюють взаємозв'язано.

Система моніторингу стежить за станом об'єкта і повідомляє власника або систему про виникнення подій. В залежності від цих подій, власник або система можуть автоматично активувати задані послідовності дій, такі як відключення електропостачання або спрацювання сигналів безпеки. Система безпеки може працювати в автономному режимі або бути інтегрованою з іншими системами залежно від потреб користувача.

Кожен модуль системи безпеки приміщення може працювати самостійно, виконуючи свої функції, або спільно з іншими модулями. Кожна система може працювати незалежно, не залежно від інших.

Застосування розумних систем безпеки має безліч можливостей, і описати всі ці системи неможливо. В роботі розглядається розумна система моніторингу технічних характеристик приміщення.

1.2. Аналіз автоматизованих систем безпеки приміщення

На сьогоднішній день існує широкий спектр автоматизованих систем безпеки приміщення, розроблених різними компаніями та виробниками. Деякі з них базуються на платформі Arduino, що забезпечує гнучкість та можливість налаштування системи під конкретні потреби користувача.

При аналізі існуючих систем слід звернути увагу на їх функціональні можливості, такі як системи моніторингу стану приміщення, виявлення вторгнень, контроль доступу, системи сповіщення та інші. Важливим аспектом є також інтеграція з іншими системами, наприклад, системами відеоспостереження або пожежної безпеки.

Окрім функціональності, необхідно оцінити ефективність системи, її надійність та стабільність. Важливо врахувати також витрати на встановлення, обслуговування та розширення системи.

Під час аналізу існуючих систем можна вивчити відгуки користувачів, дослідження та огляди в цій галузі. Це допоможе зрозуміти переваги та недоліки різних рішень, а також виявити можливі напрямки для подальшого розвитку та вдосконалення.

1. Система Ajax

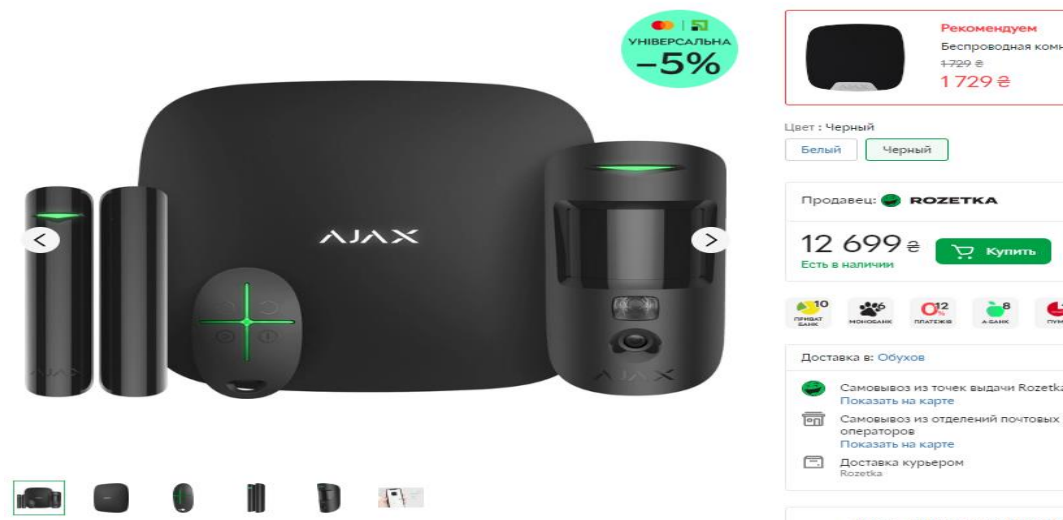


Рис. 1.1 Система безпеки Ajax

Таб. 1.1 Характеристика пристроїв системи безпеки “Аjах”

Тип пристрою	комплект охоронної сигналізації
Тип підключення	бездротовий
Канал зв'язку централь-датчики	радіоканал
Канал зв'язку централь-роутер	LAN
Канал передачі тривожних сповіщень	GSM, Ethernet, WCDMA (3G)
Повідомлення про тривогу	смс, дзвінок, push-повідомлення
Постановка на охорону	зі смартфона, брелока

2. Сигналізація GSM-015 безпроводна “Набори комплектів №1”



Сигнализация GSM-015 беспроводная Наборы комплектов №11

4 400 грн **4 000 грн**

Показать оптовые цены ▾

В наличии | Оптом и в розницу | Код: С-015-6

Комплект:

№11 ▾

Купить

+380 (96) 401-29-29 ▾

-9% скидка

Осталось 2 дня 09:04:37



Рис. 1.2 Система безпеки набори комплектів #1

Таб. 1.2 Характеристика пристроїв системи безпеки “Набори комплектів №1”

Тип пристрою:	Комплект сигналізації
Тип детекції:	Виявлення руху
Канали зв'язку:	GSM-2SIM
Можливість розширення охоронних зон:	Є (до 32)
Кількість дротових зон:	4
Кількість SIM-карт:	2

Кількість груп:	16
Дзвінок у разі тривоги:	є
SMS за тривоги:	є
Програмування:	3 Windows/Android
Імунітет від домашніх тварин:	є
Акустична потужність, дБ:	10
Тип оповіщення:	світлозвукове
Ємність акумулятора, Аг:	7

Повні технічні характеристики сигналізації Alarm System GSM-015:

Підтримка GSM частот: 850/900/1800/1900 МГц;

Частота бездротових датчиків: 433 МГц;

Кількість бездротових зон: 99;

Кількість провідних зон: 7;

Максимальна кількість бездротових датчиків: 99;

Кількість пультів брелоків: до 8;

Кількість номерів для обдзвону: 6;

Кількість номерів для відправлення SMS: 3;

Індикація режимів: дисплей;

Наявність клавіатури: так;

Гучність сирени: 110 дБ;

Підтримка бездротових датчиків EV1527/300K 2262/1.5-4.7M;

Бездротова відстань: до 100 м-код на відкритій місцевості;

Управління функціями сигналізації: СМС, пульт ДК (брелок), телефонний дзвінок;

Функція двосторонній зв'язок "СПКЕРФОН": є;

Функція акустичного контролю (прослуховування): є;

Системний годинник: так;

Таймер для постановки та зняття охорони за розкладом: є;

Керований релейний вихід: ні;

Вихід підключення зовнішнього динаміка: єсть;

Підключення зовнішньої бездротової сирени: так;

Запис голосового повідомлення: є;

Оповіщення про відключення зовнішнього живлення: є;

Голосові підказки: є;

Автовідповідач: так;

Системна пам'ять: не залежна від живлення;

Час автономної роботи від акумулятора: до 24 годин;

Характеристики акумулятора: NI-NI AAA*4 шт DC 7,4 В;

Скасування відключення брелоками: ні;

Захист GSM глушників: ні;

Антенa GSM: внутрішня;

Матеріал: пластик;

Напруга живлення: DC 12 Вольт 1А;

Струм у режимі очікування <55 мА;

Струм у режимі тривоги <450 мА.

Робоча температура: -20 ° C ... +40 ° C

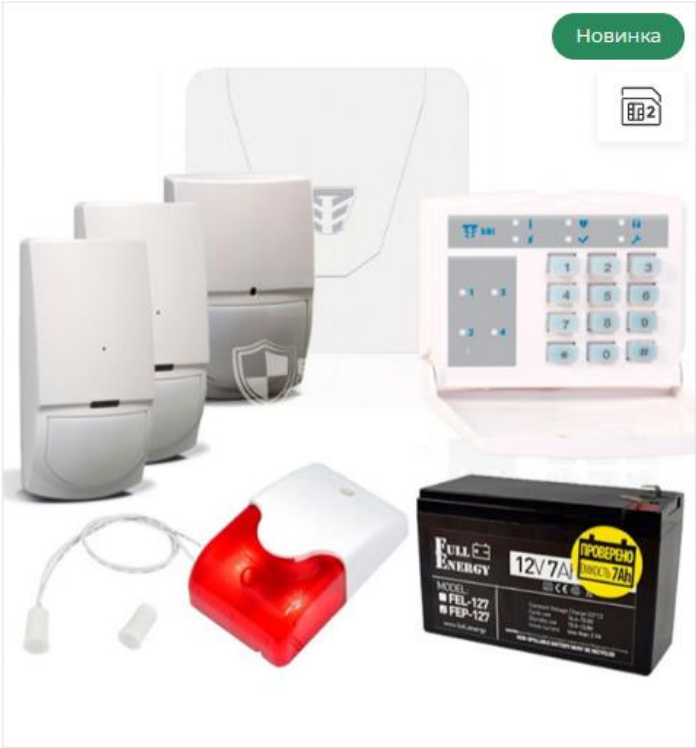
Вологість: до 90%

Габаритні розміри 18х12х2,8 см

Міжнародні сертифікати: CCC, CE, FCC

3. Комплект охороної сигналізації #1

Комплект охоронної сигналізації №1



Новинка

Комплект охоронної сигналізації №1 включає: ППКОП, клавіатуру, акумулятор, 2 комбінованих пасивних ІЧ-сповіщувача, 1 пасивний ІЧ-сповіщувач, геркон, світозвуковий оповіщувач, 50м сигнального кабелю, а також 10м силового кабелю.

Код товару: 1105782

В наявності

3

Ціна з ПДВ

13 281 грн

Купити

Додати до порівняння

Швидке замовлення

Кур'єром по Києву

Вартість доставки та обговорюється з менеджером

Нова Пошта

Доставка по всій Україні передоплаті

Гарантія

Офіційна гарантія від виробника

Обмін і повернення

Обмін / повернення

Оплата

Зручні варіанти оплати

Рис. 1.3 Система безпеки охоронної сигналізації №1

Таб. 1.3 Характеристика пристроїв системи безпеки “ Охоронної сигналізації №1”

Тип пристрою:	Комплект сигналізації
Тип детекції:	Виявлення руху
Канали зв'язку:	GSM-2SIM
Можливість розширення охоронних зон:	Є (до 32)
Кількість дротових зон:	4

Кількість SIM-карт:	2
Кількість груп:	16
Дзвінок у разі тривоги:	є
SMS за тривоги:	є
Програмування:	3 Windows/Android
Імунітет від домашніх тварин:	є
Акустична потужність, дБ:	10
Тип оповіщення:	світлозвукове
Ємність акумулятора, Аг:	7

Порівнюючи кілька автоматизованих систем безпеки приміщення, як дешеві, так і дорогі, можна зробити кілька важливих спостережень. Перш за все, вартість системи безпеки не завжди є визначальним фактором для її ефективності та надійності. Дешеві системи можуть виконувати базові функції безпеки, такі як виявлення руху або сповіщення про події, але можуть бути обмежені в інших аспектах, таких як можливості комунікації або інтеграція з іншими системами.

З іншого боку, дорогі системи безпеки часто пропонують більше функціональних можливостей, розширену інтеграцію та більшу надійність. Вони можуть мати продуману архітектуру, що дозволяє керувати багатьма аспектами безпеки одночасно та забезпечувати додаткові можливості, такі як відеоспостереження або взаємодія зі смарт-пристроями.

Важливо приділити увагу власним потребам та вимогам до системи безпеки. Дешева система може бути достатньою для простих рішень або для невеликих приміщень, тоді як більш складна або розподілена система може бути необхідною для великих приміщень або вимогливих вимог безпеки.

Окрім цього, необхідно ретельно вивчити можливості і характеристики кожної системи безпеки, враховуючи особливості власного середовища та конкретні потреби. Не завжди найдорожча система є найкращим варіантом, але відповідна система з адекватною ціною та функціональністю може забезпечити надійний рівень безпеки для різних потреб.

РОЗДІЛ 2. ОГЛЯД ТА ВИБІР АРХІТЕКТУРНОГО ЗАБЕЗПЕЧЕННЯ ПРОЄКТУ

2.1 Arduino – як платформа проєктування

Arduino - це інструмент для створення електронних пристроїв, який має більш глибоку взаємодію з фізичним середовищем, ніж звичайні персональні комп'ютери, що фактично працюють у віртуальному просторі. Проекти, засновані на цій платформі, можуть отримувати сигнали від різних цифрових і аналогових датчиків, а також керувати різними виконавчими пристроями. **Arduino** - це готова апаратно-програмна платформа, яка включає невелику плату-контролер введення/виведення та розробницьке середовище, засноване на Processing/Wiring. Проекти на Ардуіно можуть працювати автономно або взаємодіяти з програмним забезпеченням, що працює на персональному комп'ютері, наприклад, з додатками Adobe Flash, Processing або MaxMSP.

Існує багато різних моделей Ардуіно, кожна з яких призначена для виконання різних завдань. Хоча деякі плати можуть відрізнятися від тих, що зображені на малюнку нижче, більшість з них мають спільні компоненти (рисунок 2.1):

Роз'єм живлення (USB або роз'єм для адаптера): Кожна плата Ардуіно потребує джерела живлення. Наприклад, **Arduino UNO** може житися через USB-кабель від персонального комп'ютера або через зовнішній адаптер, підключений до відповідного роз'єму на платі. Рекомендована напруга живлення для Ардуіно - від 6 до 12 вольт. USB-з'єднання позначене на малюнку (1), а роз'єм для зовнішнього джерела живлення - (2).

Основна відмінність плати від попередніх полягає у способі взаємодії з USB. У випадку Arduino Uno використовується окремий мікроконтролер ATmega8U2 для цієї цілі. У попередніх версіях Arduino використовувалася мікросхема програматора FTDI.

Таб. 2.1 Характеристика Arduino UNO

Мікроконтролер	ATmega328
Робоча напруга	5В
Напруга живлення (рекомендований)	7-12В
Напруга живлення (граничне)	6-20В
Цифрові входи / виходи	14 (з них 6 можуть використовуватися в якості ШІМ-виходів)
Аналогові входи	6
Максимальний струм одного виведення	40 мА
Максимальний вихідний струм виводу 3.3V	50 мА
Flash-пам'ять	32 КБ (ATmega328) з яких 0.5 КБ використовуються загрузчиком
SRAM	2 КБ (ATmega328)
EEPROM	1 КБ (ATmega328)
Тактова частота	16 МГц



Рис. 2.1. Плата Arduino Uno

Vin (3) - цей пін використовується для вхідного живлення зовнішнього джерела. Рекомендована напруга для вхідного Vin становить від 7В до 20В, щоб уникнути перегріву і пошкодження вбудованого стабілізатора.

GND (4) - це з'єднання для заземлення. Воно необхідне для створення замкнутого кола, коли піни Vin, 5V або 3.3V підключені. На платі Arduino може бути кілька пінів GND, які можна використовувати для заземлення електричного кола.

5V (5) - цей пін надає стабільне живлення 5 вольт для зовнішніх пристроїв. Незалежно від джерела живлення плати (USB, роз'єм живлення або Vin), на цьому піні можна отримати постійну напругу 5 вольт. Ви можете вивести цей пін на макетну плату або підключити його безпосередньо до пристрою.

3.3V (6) - цей пін надає живлення 3.3 вольт для зовнішніх пристроїв. Режим роботи цього піна аналогічний до піна 5V.

3.3V (6) - цей пін надає живлення 3.3 вольт для зовнішніх пристроїв. Він працює за таким самим принципом, як і контакт 5V.

Analog (7) - ці піни розташовані під написом "Analog In" і використовуються для зчитування сигналів від аналогових датчиків, таких як датчик температури. Вони перетворюють аналогові значення на цифрові, з якими можна подальше працювати.

Digital (8) - це цифрові піни, які використовуються для отримання цифрових вхідних сигналів (наприклад, натискання кнопки) і генерації цифрових вихідних сигналів (наприклад, живлення світлодіода).

PWM (9) - ці піни можуть працювати як звичайні цифрові піни, а також у режимі широтно-імпульсної модуляції (ШИМ). Вони дозволяють створювати аналогові сигнали (наприклад, для плавного згасання світлодіода). Якщо на платі відсутня окрема позначка для PWM пінів, то поруч з номером піна використовується символ "~".

AREF (10) - цей контакт відповідає за опорну напругу аналогових входів на платформі Arduino.

Кнопка скидання (11) - при натисканні на цю кнопку відбувається скидання контролера Arduino, і виконання коду починається з початку. Це корисна опція, якщо ви хочете перезапустити вашу програму без вимикання живлення.

Індикатор живлення (12) - цей світлодіод запалюється, коли Arduino підключений до джерела живлення. Якщо цей індикатор не світиться, слід перевірити живлення та схему підключення.

Світлодіоди TX і RX (13) - TX означає передачу (transmit), а RX - прийом (receive). Ці світлодіоди візуально показують, коли Arduino передає або отримує дані (наприклад, при завантаженні програми на плату).

Головна інтегральна мікросхема (14) - це чорна деталь з металевими конекторами з обох сторін, яка представляє собою інтегральну мікросхему або мікропроцесор (Integrated Circuit). В різних моделях Arduino використовуються різні типи цих чіпів, але зазвичай вони належать до лінійки мікропроцесорів ATmega від компанії ATMEL.

Регулятор напруги (15) - це елемент, який контролює напругу, що надходить на плату Arduino. Він регулює напругу, щоб запобігти пошкодженню плати від надмірної напруги.

Робоча напруга плати Arduino Uno складає 5 В. На платі є вбудований стабілізатор напруги, що дозволяє жити їй з різних джерел. Крім того, плата може отримувати живлення від USB-пристроїв, при цьому джерело живлення обирається автоматично.

Якщо використовується зовнішній адаптер для живлення, рекомендована напруга становить від 7 до 12 В. Максимально допустима напруга складає 20 В, але значення, що перевищують 12 В, можуть швидко пошкодити плату. Напруга нижче 7 В може призвести до нестабільної роботи, оскільки на вхідному каскаді може втрачатися 1-2 В.

Для підключення живлення можна використовувати вбудований роз'єм DC 2.1 мм або безпосередньо підключити джерело живлення до входу VIN за допомогою дротів.

головне меню. Це програмне забезпечення дозволяє комп'ютеру взаємодіяти з Arduino як для передачі даних, так і для прошивки коду на контролер.

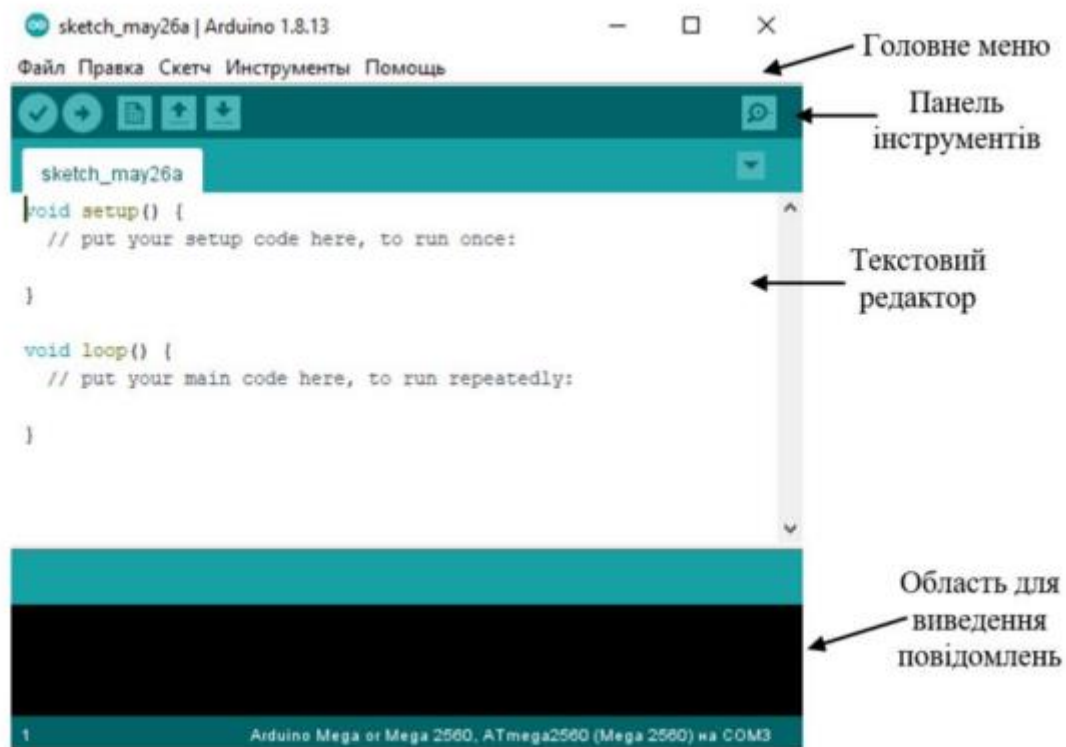


Рис. 2.2. Інтерфейс Arduino IDE

Програми, які створюються в середовищі розробки Arduino, часто називають "скетчами". Скетчі написані у текстовому редакторі і зберігаються у файлах з розширенням `.ino`. Вбудований текстовий редактор має стандартні інструменти копіювання, вставки, пошуку та заміни тексту.

Область повідомлень у вікні програми виконує роль зворотного зв'язку для користувача і надає інформацію про події, включаючи помилки, які виникають під час запису або експорту коду. В нижньому правому куті вікна програми вказана модель поточної плати і послідовний порт, до якого вона підключена.

На панелі інструментів знаходиться 6 кнопок (рис. 2.3):



Рис. 2.3. Панель інструментів

- **Перевірити:** Ця функція використовується для перевірки синтаксису вашого коду. У разі виявлення помилок, вони будуть відображені у вкладці повідомлень.
- **Завантажити:** Ця опція використовується для компіляції вашої програми та завантаження її на плату Arduino. Вона перетворює ваш код у виконуваний файл та пересилає його до мікроконтролера.
- **Створити:** Ця функція дозволяє створити новий файл у середовищі розробки Arduino.
- **Відкрити:** Використовується для відкриття існуючого файлу скетчу в середовищі розробки.
- **Зберегти:** Використовується для збереження поточного коду у файлі.
- **Монітор послідовного порту:** Ця опція відкриває програму "Serial Monitor", яка відображає дані, що надходять від Arduino до комп'ютера через послідовний інтерфейс (можна використовувати як USB, так і звичайні версії Arduino). Щоб надіслати дані зовнішнього пристрою, потрібно ввести текст у вікні програми і натиснути кнопку "Відправити" або клавішу Enter. Треба вибрати правильну швидкість передачі даних зі списку, відповідно до швидкості, вказаної у функції `Serial.begin()` вашого скетчу. За допомогою функції `Serial.print()` можна виводити текст у вікно програми.

- **Бібліотеки:** Бібліотеки розширюють функціональність програм та містять додаткові функції, наприклад, для роботи з апаратними засобами або обробки даних. Щоб підключити бібліотеку, потрібно обрати її зі списку у меню "Скетч" -> "Підключити бібліотеку". Після цього бібліотека буде компільована, і до початку програми будуть додані один або декілька операторів `#include`. Підключені бібліотеки завантажуються разом із скетчем на плату Arduino, тому кожна підключена бібліотека займає додаткове місце у пам'яті мікроконтролера.
- **Завантажити програму:** Ця опція використовується для завантаження вашої програми на плату Arduino. Перед завантаженням необхідно правильно вибрати плату в меню "Tools" -> "Board" та послідовний порт в меню "Tools" -> "Serial Port". Після вибору використовуваного порту і плати, натисніть кнопку "Завантажити" на панелі інструментів або оберіть пункт "Завантажити" у меню "Файл". Процес завантаження розпочнеться, і програма буде записана у пам'ять мікроконтролера. Під час завантаження на більшості моделей Arduino світлодіоди RX і TX можуть блимати. Після завершення процесу завантаження, програма виведе відповідне повідомлення або помилку.
- **Завантажувач:** Завантажувач - це невелика програма, яка вбудована у мікроконтролер Arduino і дозволяє завантажити код без необхідності використання зовнішніх апаратних засобів. Завантажувач активується на кілька секунд після перезавантаження плати і запускає останній завантажений скетч у контролері. Під час активації завантажувача світлодіод, підключений до 13-го піна контролера, може блимати.

2.3 Програмування Arduino

Для програмування Arduino використовується спрощена версія мови C++ з

деякими додатковими функціями. Усі прошивки для Arduino повинні містити щонайменше дві функції: `setup()` і `loop()`.

Функція `setup()` викликається при запуску скетчу і використовується для ініціалізації змінних, встановлення режимів роботи виходів, підключення до додаткових модулів та налаштування підключених бібліотек. Функція `setup()` виконується лише один раз, після кожного включення або скидання плати Arduino.

Функція `loop()` виконується після функції `setup()`. Як із назви випливає, весь код, що міститься між фігурними дужками у функції `loop()`, виконується в постійному циклі, дозволяючи вашій програмі здійснювати обчислення та реагувати на них. Наприклад, мікроконтролер ATmega328, який зазвичай використовується на більшості плат Arduino, виконує функцію `loop()` близько 10000 разів на секунду (якщо немає затримок і складних обчислень).

Основними типами функцій для взаємодії з Arduino є функції цифрового, аналогового та розширеного введення/виведення, а також функції часу.

До функцій цифрового введення/виведення належать:

`pinMode()` – використовується для налаштування режиму роботи певного виводу: як вхід або як вихід.

`digitalWrite()` – встановлює цифровому виводу значення HIGH або LOW. Якщо вивід встановлено як вихід (OUTPUT) за допомогою `pinMode()`, то при використанні `digitalWrite()` напруга на виводі змінюється на 5 В (або 3.3 В для плат, що працюють від 3.3 В) при встановленні значення HIGH, або на 0 В при встановленні значення LOW. Якщо вивід налаштовано як вхід (INPUT) за допомогою `pinMode()`, використання `digitalWrite()` зі значенням HIGH призводить до підключення внутрішнього підтягуючого резистора з опором 20 кОм. Запис значення LOW вимикає підтяжку.

`digitalRead()` – зчитує рівень сигналу HIGH або LOW з вказаного цифрового виводу.

До аналогових функцій введення/виведення в Arduino належать:

`analogWrite()` – ця функція дозволяє встановлювати задану напругу на аналоговому виводі шляхом створення ШІМ-сигналу. Після виклику `analogWrite()` на виводі буде неперервно генеруватися ШІМ-сигнал з заданим коефіцієнтом заповнення до наступного виклику `analogWrite()` (або до моменту виклику `digitalRead()` або `digitalWrite()`, що взаємодіють з цим же виводом). Частота ШІМ-сигналу становить приблизно 490 Гц.

`analogRead()` – ця функція зчитує значення напруги з вказаного аналогового виводу. За допомогою функції `analogReference()` можна змінювати джерело опорної напруги і роздільну здатність.

`analogReference()` – ця функція встановлює джерело опорної напруги, яке використовується при зчитуванні аналогового сигналу, тобто задає максимальне значення вхідного діапазону.

До розширених функцій введення/виведення в Arduino належать:

`tone()` – ця функція генерує прямокутний сигнал заданої частоти на виводі. Вона також дозволяє встановлювати тривалість сигналу. Якщо тривалість сигналу не вказана, він буде генеруватися до тих пір, поки не буде викликана функція `noTone()`. Одночасно може генеруватися лише один сигнал заданої частоти. Якщо сигнал вже генерується на будь-якому виводі, використання функції `tone()` для цього виводу просто змінить частоту сигналу, а виклик функції для іншого виводу не матиме жодного ефекту.

`noTone()` – ця функція припиняє генерування прямокутного сигналу після використання функції `tone()`. Якщо сигнал не генерується, функція `noTone()` ні до чого не призведе.

До функцій часу в Arduino належать:

`millis()` – ця функція повертає кількість мілісекунд, що пройшли з моменту запуску програми Arduino. Число, яке повертається, переповнюється (починається з нуля) приблизно через 50 днів.

`delay()` – ця функція призупиняє виконання програми на вказаний проміжок часу (у мілісекундах).

2.4 Плати Arduino

Існує широкий асортимент оригінальних плат Arduino, а також багато Arduino-сумісних плат, які виробляються різними сторонніми компаніями. Вони відрізняються за характеристиками мікроконтролерів, розмірами, кількістю портів введення-виведення та іншими параметрами, що дозволяє обрати плату, що найкраще підходить до поставлених вимог. Нижче наведено деякі з найпоширеніших плат Arduino.

Arduino Leonardo - це плата, яка базується на мікроконтролері ATmega32U4. Вона має роз'єм мікро-USB, роз'єм живлення, роз'єм для внутрішньосхемного програмування ICSP (In-Circuit Serial Programming) і кнопку скидання. Плата має 20 цифрових входів-виходів і 12 аналогових входів. Вбудована flash-пам'ять має об'єм 32 КБ, з яких чотири КБ виділені для завантажувача. Оперативна пам'ять (ОЗУ) пристрою складає 2,5 КБ, а тактова частота становить 16 МГц.

Основна відмінність Leonardo від попередніх плат полягає в тому, що його USB-контролер вбудований безпосередньо в мікроконтролер ATmega32U4, що усуває потребу в додатковому процесорі. Це дозволяє Leonardo бути визнаним комп'ютером як віртуальний COM-порт, а також може функціонувати як звичайна миша або клавіатура при підключенні до комп'ютера.

Arduino Due - це потужна плата, яка базується на 32-бітному ARM-процесорі AT91SAM3X8E від Atmel. Вона має тактову частоту 84 МГц, а 32-бітна архітектура дозволяє виконувати багато операцій з цілими числами у 4 байти за один такт. Плата має 54 цифрових виходи (з них 12 можуть працювати як ШІМ-виходи), 12 аналогових входів, 4 UART для послідовного зв'язку, USB з підтримкою технології OTG, 2 ЦАП, 2 TWI, роз'єм живлення, роз'єм SPI, роз'єм JTAG, кнопку скидання та кнопку очищення пам'яті. Особливість інтерфейсу SPI полягає в тому, що його не можна використовувати для внутрішньосхемного програмування, а лише для зв'язку з іншими пристроями. Відмінність Arduino Due від інших плат полягає в робочій напрузі, яка становить 3.3 В. На цій платі також є 512 КБ флеш-пам'яті та 96 КБ оперативної пам'яті.



Рис. 2.4. Плата Arduino Due

Arduino Leonardo - це платформа для розробки та прототипування електронних пристроїв, що базується на мікроконтролері ATmega32U4. Вона відрізняється від інших моделей Arduino завдяки вбудованій функціональності USB, що дозволяє

йому працювати як віртуальна клавіатура або миша, а також як віртуальний COM-порт для зв'язку з комп'ютером.

Основні характеристики Arduino Leonardo включають:

- Мікроконтролер: ATmega32U4 з тактовою частотою 16 МГц та 32 КБ флеш-пам'яттю для програмного забезпечення.
- Вбудований USB: Леонардо має вбудований контролер USB, що дозволяє платформі здійснювати зв'язок з комп'ютером та виконувати різноманітні функції, такі як віртуальна клавіатура або миша.
- Цифрові входи/виходи: Леонардо має 20 цифрових входів/виходів, серед яких 7 можуть бути використані для генерації ШИМ-сигналу.
- Аналогові входи: Платформа має 12 аналогових входів, кожен з яких має розширений діапазон напруги (0-5 вольт).
- Пам'ять: Леонардо має 2,5 КБ ОЗП (SRAM) та 1 КБ EEPROM для зберігання даних.
- Додаткові функції: Включають в себе вбудований генератор часу реального, можливість використання як пристроїв HID (Human Interface Device), підтримку Serial (COM) зв'язку, SPI та I2C.



Рис. 2.5 Плата Arduino Leonardo

Arduino Uno - це пристрій, що базується на мікроконтролері ATmega328. Він має все необхідне для зручної роботи з мікроконтролером: 14 цифрових входів/виходів (з них 6 можуть бути використані як ШІМ-виходи), 6 аналогових входів, роз'єм USB, роз'єм живлення, роз'єм для внутрішньосхемного програмування та кнопку скидання. Основні характеристики плати такі: тактова частота - 16 МГц, робоча напруга - 5 В, флеш-пам'ять - 32 КБ, оперативна пам'ять - 2 КБ.



Рис. 2.6. Плата Arduino Uno

В майбутньому в роботі ми будемо використовувати саме цю плату, а саме Arduino Mega 2560, тому що саме цей функціонал плати для нас цікавий найбільше з усіх.

Arduino Mega 2560 - це пристрій, що базується на мікроконтролері ATmega2560. Вона має все необхідне для зручної роботи з мікроконтролером: 54 цифрових входи/виходи (з них 15 можуть бути використані як ШІМ-виходи), 16 аналогових входів, 4 UART для послідовних інтерфейсів, кварцовий резонатор на 16 МГц, роз'єм USB, роз'єм живлення, роз'єм ICSP для внутрішньосхемного програмування та кнопка скидання. Робоча напруга цієї плати становить 5 В, об'єм флеш-пам'яті - 256 Кб, об'єм оперативної пам'яті - 8 Кб. Плати Arduino Mega використовуються там, де потрібна велика кількість входів і виходів. Я саме обрав цю плату для проектування системи безпеки.

Таб. 2.2 Характеристика Arduino Mega 2560, яка прийнята за основу

Мікроконтролер	ATmega2560
Робоча напруга	5В
Напруга живлення (рекомендована)	7-12В
Напруга живлення (гранична)	6-20В
Цифрові входи/виходи	54 (з яких 15 можуть використовуватися як ШІМ-виходи)
Аналогові входи	16
Максимальний струм одного виводу	40 мА
Максимальний вихідний струм виведення	3.3V 50 мА

Flash-пам'ять	256 КБ, з яких 8 КБ використовуються завантажувачем
SRAM	8 КБ
EEPROM	4 КБ
Тактова частота	16 МГц

Зв'язок

Arduino Mega 2560 надає низку можливостей для здійснення зв'язку з комп'ютером, ще одним Arduino або іншими мікроконтролерами. У ATmega2560 є чотири апаратні приймачі UART для реалізації послідовних інтерфейсів (з логічним рівнем TTL 5В). Мікроконтролер ATmega16U2 (або ATmega8U2 на платах версії R1 і R2) забезпечує зв'язок одного з приймачів з USB-портом комп'ютера, і при підключенні до ПК дозволяє Arduino визначатися як віртуальний COM-порт (для цього операційній системі Windows знадобиться відповідний .inf-файл, на відміну від OSX чи Linux, де розпізнавання плати як COM-порту відбувається автоматично). У пакет програмного забезпечення Arduino входить спеціальна програма SerialMonitor, що дозволяє зчитувати та відправляти на Arduino прості текстові дані. При передачі даних через мікросхему ATmega8U2/ATmega16U2 під час USB-з'єднання з комп'ютером, на платі блиматимуть світлодіоди RX та TX. (При послідовній передачі даних за допомогою висновків 0 і 1, без використання USB-перетворювача, дані світлодіоди не використовуються).

Програмування Arduino

Програмування Arduino Mega 2560 здійснюється за допомогою програмного забезпечення Arduino. Мікроконтролер ATmega2560 в Arduino Mega вже має завантажувальну програму, що дозволяє завантажувати нові

програми без використання зовнішнього програматора. Взаємодія з ним відбувається за допомогою оригінального протоколу STK500 (опис, заголовкові файли C).

Проте, мікроконтролер можна прошивати також через роз'єм для внутрішньо-схемного програмування ICSP (In-Circuit Serial Programming), не звертаючи уваги на завантажувальну програму; для докладнішої інформації див. відповідні інструкції.

Вихідний код прошивки мікроконтролера ATmega16U2 (або ATmega8U2 на платах R1 і R2) знаходиться у репозиторіях Arduino. Прошивка ATmega16U2/8U2 включає DFU-завантажувач (Device Firmware Update), що дозволяє оновлювати прошивку мікроконтролера. Для активації режиму DFU необхідно:

На платах версії R1: потрібно замкнути перемичку на зворотному боці плати (біля зображення Італії) і потім скинути 8U2. На платах версій R2 і вище - для спрощення переходу в режим DFU присутній резистор, що з'єднує лінію HWB мікроконтролера 8U2/16U2 з землею. Після переходу в режим DFU можна використовувати програмне забезпечення Atmel's FLIP (для Windows) або DFU programmer (для Mac OS X і Linux) для завантаження нової прошивки. Альтернативний варіант - прошивка мікроконтролера через роз'єм для внутрішньосхемного програмування ISP з використанням зовнішнього програматора, проте при цьому DFU-завантажувач буде видалено. Для отримання більш детальної інформації див. ці інструкції від користувачів.

Автоматичне (програмне) скидання

Для того щоб перезавантажити програму, не потрібно кожного разу натискати кнопку скидання перед завантаженням програми, Arduino Mega 2560 була спроектована з можливістю програмного скидання з підключеного комп'ютера.

Один з виводів ATmega8U2, який відповідає за керування потоком даних (DTR), з'єднаний з виводом RESET мікроконтролера ATmega2560 через конденсатор ємністю 100 нФ. Коли на лінії DTR з'являється нуль, вивід RESET також переходить до низького рівня на достатній проміжок часу для перезавантаження мікроконтролера. Ця особливість використовується для можливості прошивки мікроконтролера всього одним натисканням кнопки у середовищі Arduino. Така архітектура дозволяє зменшити час очікування завантажувача, оскільки процес прошивки завжди синхронізований зі спадом сигналу на лінії DTR.

Проте ця система може мати інші наслідки. При підключенні Mega 2560 до комп'ютерів, що працюють на Mac OS X або Linux, його мікроконтролер буде скидатися при кожному з'єднанні програмного забезпечення з платою. Після скидання на Arduino Mega 2560 активується завантажувач на протязі близько півсекунди. Незважаючи на те, що завантажувач запрограмований ігнорувати чужі дані (тобто всі дані, що не стосуються процесу прошивки нової програми), він може перехопити кілька перших байтів даних з пакету, що надсилається до плати одразу після встановлення з'єднання. Отже, якщо у програмі, що працює на Arduino, передбачено отримання від комп'ютера певних налаштувань або інших даних при першому запуску, переконайтеся, що програмне забезпечення, з яким взаємодіє Arduino, надсилає ці дані через секунду після встановлення з'єднання.

На платі Mega 2560 є доріжка (позначена як "RESET-EN"), від'єднавши яку, можна вимкнути автоматичне скидання мікроконтроллера. Для повторного відновлення функції автоматичного скидання необхідно з'єднати між собою виводи, розташовані по краях цієї доріжки. Автоматичне скидання також можна вимкнути, підключивши резистор ємністю 110 Ом між виводом RESET і 5В.

Фізичні характеристики та сумісність з розширювальними платами

Максимальна довжина та ширина друкованої плати Mega2560 становлять відповідно 10.2 см і 5.4 см, з урахуванням роз'єму USB та роз'єму живлення, що виступають за межі плати. Три отвори для кріплення дозволяють закріплювати плату на поверхні або корпусі. Зверніть увагу, що відстань між цифровими виводами 7 та 8 не кратна традиційним 2.54 мм і становить 4 мм.

Arduino Mega2560 спроектована з метою сумісності з більшістю розширювальних плат для Arduino Uno, Diecimila і Duemilanove. Для цього цифрові виводи 0 - 13 (а також суміжні виводи AREF та GND), аналогові входи 0 - 5, роз'єм живлення та ICSP-роз'єм розташовані однаково на всіх платах. Крім того, на вказаних пристроях лінії основного передавача-приймача UART з'єднані з одними й тими ж виводами (0 і 1), так само як і лінії зовнішніх переривань 0 і 1 (виводи 2 і 3 відповідно). Лінії SPI інтерфейсу виведені на ICSP-роз'єм на обох платах - як на Mega2560, так і на Duemilanove / Diecimila. Варто зауважити, що розташування виводів інтерфейсу I2C на Arduino Mega відрізняється від плат Duemilanove / Diecimila: на Arduino Mega це виводи 20 і 21, а на Duemilanove / Diecimila - аналогові входи 4 і 5.

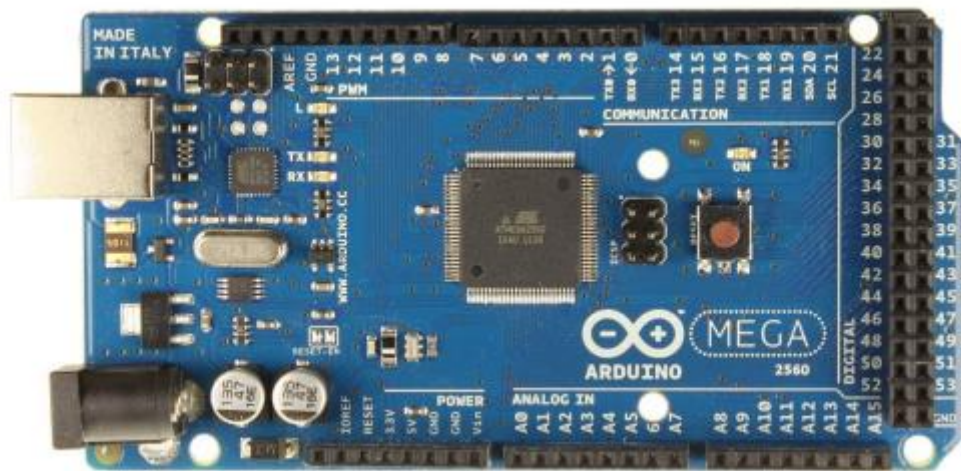


Рис. 2.7. Плата Arduino MEGA2560 (передня частина)

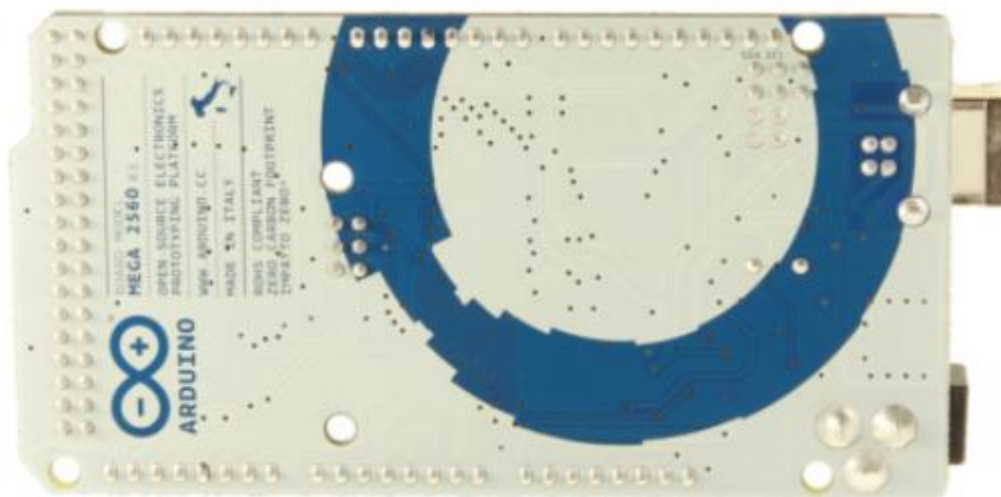


Рис. 2.8. Плата Arduino MEGA2560 (задня частина)

Arduino Ethernet - це пристрій, заснований на мікроконтролері ATmega328. Вона має 14 цифрових виходів, 6 аналогових входів, кварцовий резонатор на 16 МГц, роз'єм RJ45 для мережевого з'єднання, роз'єм живлення, роз'єм ICSP для внутрішньосхемного програмування, а також кнопку скидання. Характеристики цієї плати включають тактову частоту - 16 МГц, флеш-пам'ять - 32 Кб, оперативну пам'ять - 2 Кб, робочу напругу - 5 В. Arduino Ethernet також має додатковий модуль POE (Power Over Ethernet), який дозволяє жити платі безпосередньо від мережі Ethernet, якщо POE-живлення підключене на іншому

кінці Ethernet-кабелю, без необхідності використання зовнішнього джерела живлення.

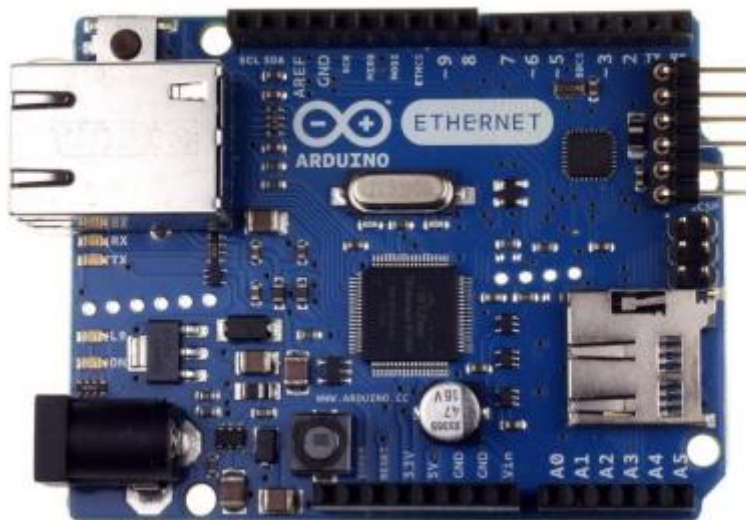


Рис. 2.9 Arduino Ethernet

Arduino Micro - це контролер, який повністю виправдовує свою назву; це одна з найменших плат у лінійці Arduino. Незважаючи на свій компактний розмір, вона все ж має велику кількість входних і вихідних виводів: 20 цифрових входів/виходів, з яких 7 можуть бути використані як ШІМ-виходи, а також 12 аналогових входів. Характеристики цієї плати включають: робочу напругу - 5 В, тактову частоту - 16 МГц, флеш-пам'ять - 32 Кб, оперативну пам'ять - 2.5 Кб.



Рис. 2.10 Плата Arduino Micro

2.5. Датчики та пристрої, які вибрані в проекті

Головним датчиком у системі безпеки є інфрачервоний датчик руху HCSR501. Інфрачервоні датчики дозволяють виявляти рух людей в контрольованій зоні, оскільки тіло людини випромінює тепло. Конструкція датчика руху PIR (пасивний інфрачервоний) не є складною - він складається з піроелектричного елемента, який відрізняється високою чутливістю (циліндрична деталь, у центрі якої розташований кристал), до наявності певного рівня інфрачервоного випромінювання у зоні дії. Чим вища температура об'єкта, тим більше випромінювання. Поверх PIR-датчика розміщена півсфера, розділена на кілька ділянок (лінз), кожна з яких фокусує теплове випромінювання на різні сегменти датчика руху. Найчастіше використовується лінза Френеля, яка, завдяки концентрації теплового випромінювання, розширює діапазон чутливості інфрачервоного датчика руху Arduino.

Основні технічні характеристики датчика руху Arduino наступні: зона виявлення рухливих об'єктів становить від 0 до 7 метрів; діапазон кута спостереження - 110° ; напруга живлення - від 4.5 до 6 В; робочий струм - до 0.05 мА; температурний режим - від -20° до $+50^{\circ}$ С; регульований час затримки від 0.3 до 300 секунд.



Рис. 2.11. Датчик руху HC-SR501

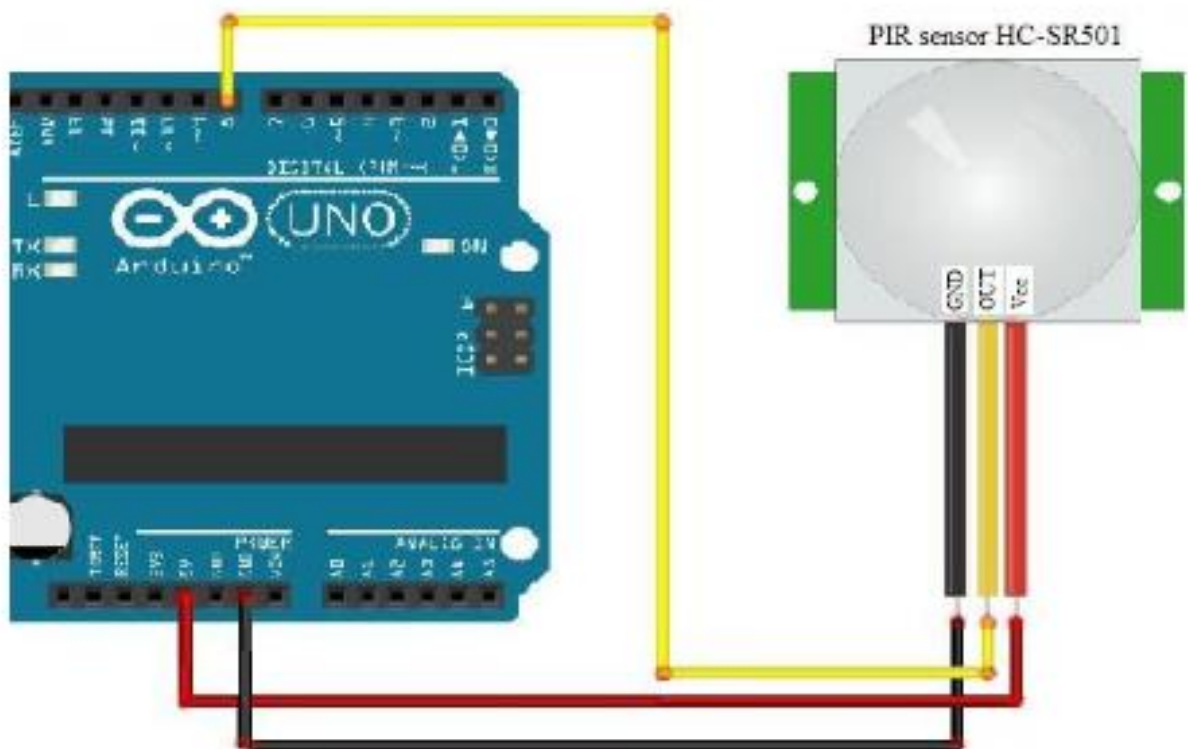


Рис. 2.12 Схема підключення датчика до плати Arduino

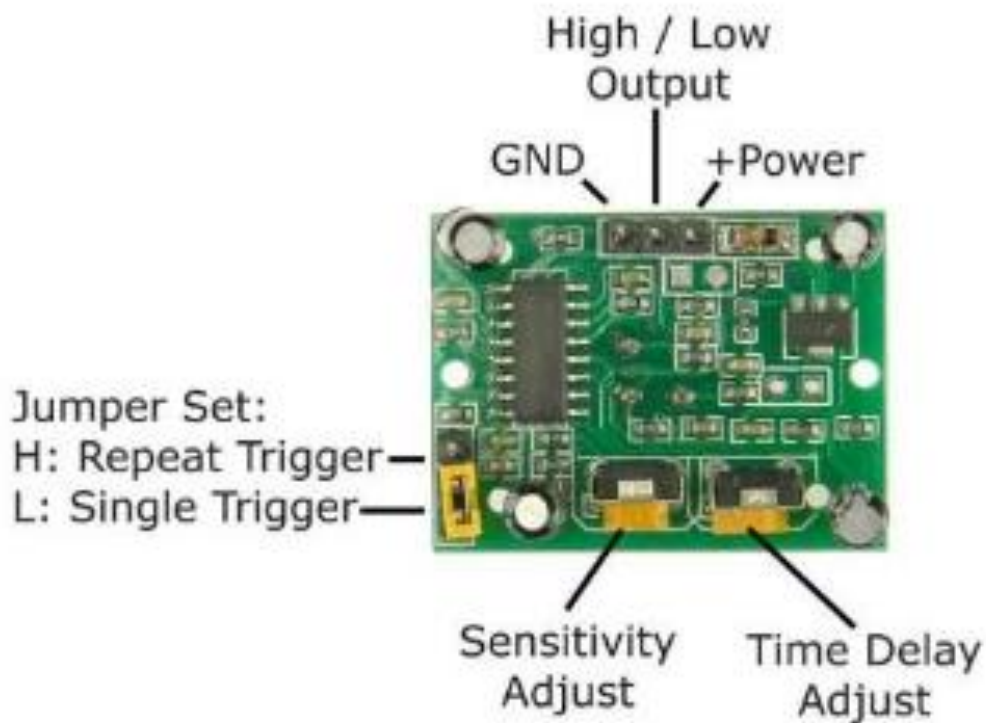


Рис. 2.13 Комплектація датчика руху

- Sensivity Adjust - Налаштування чутливості
- Time Delay Adjust - Регулювання затримки часу
- Jumper set – Вибір триггеру
- High Repeat Trigger - Тригер високого повтору
- Low Single Trigger – Тригер низького одного повтору
- GND - опорна точкою для всіх сигналів
- Power – Живлення

Для взаємодії користувача з системою потрібні пристрої вводу та виводу. В якості пристрою вводу була обрана 12-кнопочка мембранна клавіатура. Для використання мембранної клавіатури необхідно підключити її до пристрою, яким потрібно керувати, створити програму управління та забезпечити

живлення. Принцип роботи полягає у тому, що при натисканні будь-якої клавіші відбувається замикання певної пари проводів.

Характеристики цієї клавіатури наступні: час спрацювання - 1 мс; надійність - не менше 1 мільйона натискань; максимальна постійна робоча напруга - 35 В.



Рис. 2.12. 12-кнопкова мембранна клавіатура

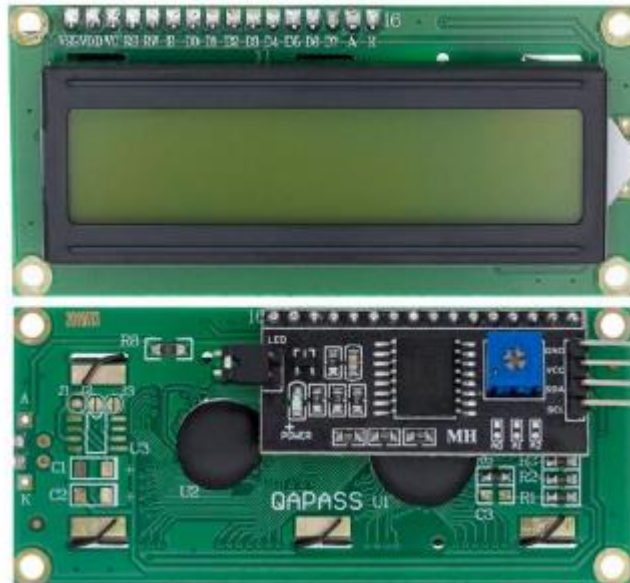


Рис. 2.13. Вигляд дисплею спереду та ззаду

Для забезпечення віддаленого керування та сповіщення користувача був обраний модуль SIM800L. Цей GSM-модуль (Global System for Mobile Communications) використовує мережу телефонного зв'язку оператора для передачі та отримання сигналу на віддалений об'єкт управління. В основі модуля знаходиться чіп SIM800L від SimCom. Робоча напруга чіпа становить від 3.4 В до 4.4 В. Основні технічні характеристики модуля такі:

- Формат SIM-карти: microSIM;
- Чотири діапазони частот 850/900/1800/1900 МГц;
- Підтримка 2G мережі;
- Струм в режимі очікування - 0.7 мА, граничний струм - 2 А.

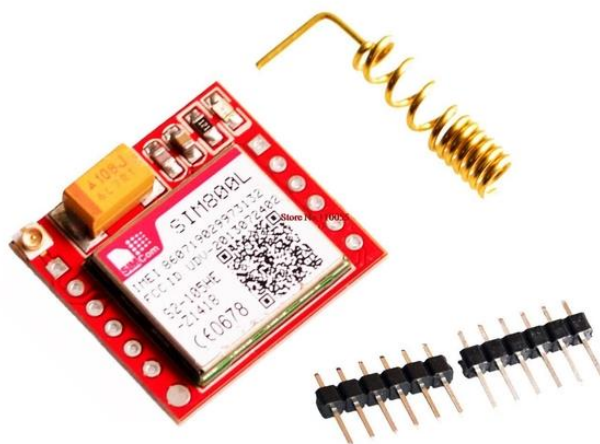


Рис. 2.14. Модуль GSM SIM800L

Модуль SIM800L вимагає нестандартну напругу живлення в діапазоні 3.4-4.4 В, тоді як Arduino плати надають живлення 3.3 В та 5 В. Для забезпечення необхідної напруги використовується окреме джерело живлення з напругою від 7 до 12 В та DC-DC понижуючий перетворювач MP1584, який дозволяє встановити потрібну напругу для GSM-модуля. Основні характеристики MP1584: вхідна напруга 4.5-28 В, вихідна напруга 0.8-20 В (можна налаштувати за допомогою спеціального резистора на платі), максимальний вихідний струм - 3 А (залежить від різниці між вхідною та вихідною напругами).



Рис. 2.15. DC-DC понижуючий перетворювач MP1584

П'єзоелектричний випромінювач звуку – п'ятивольтовий активний динамік безперервного сигналу з вихідною частотою 2.3 КГц. Даний модуль використовується для звукового сповіщення системи.



Рис. 2.16. П'єзоелектричний випромінювач звуку

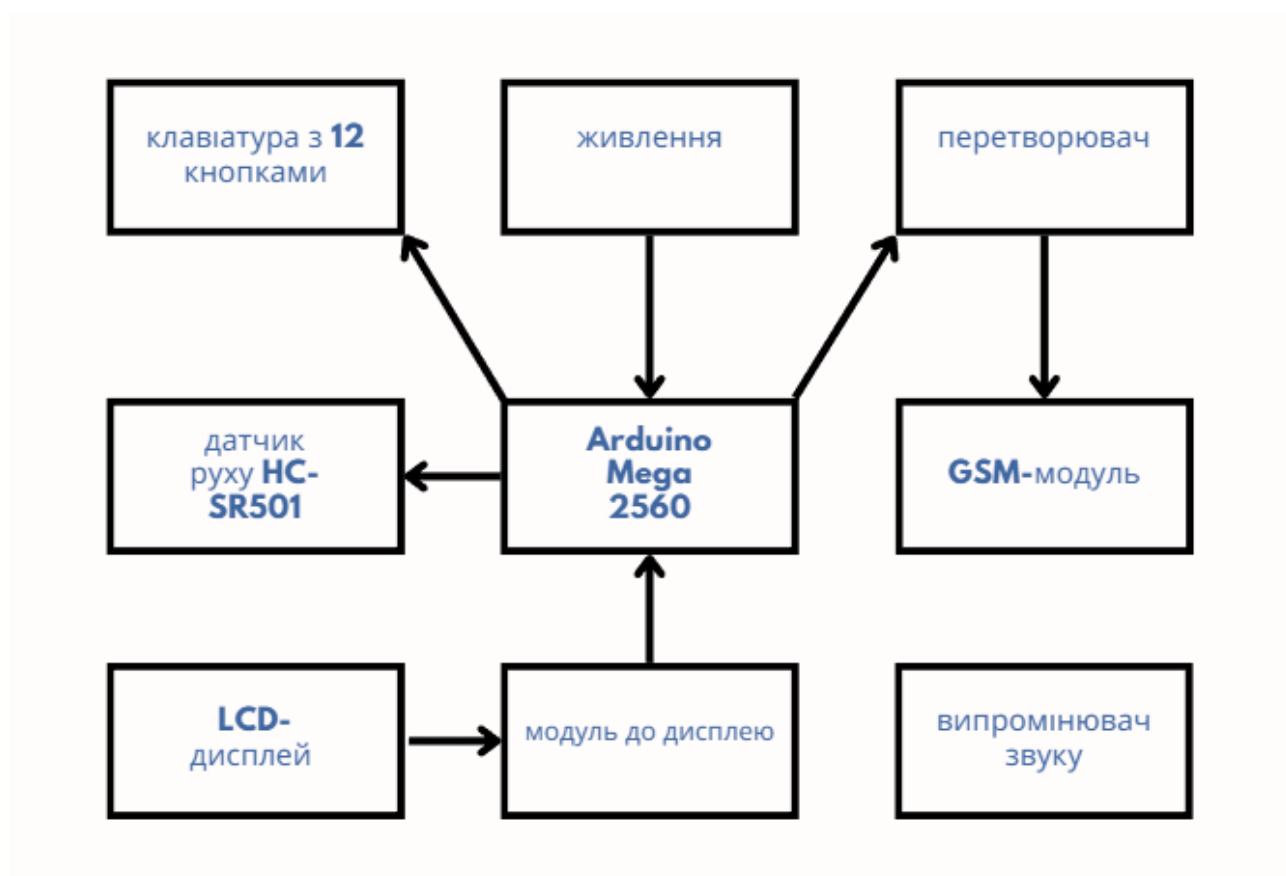


Рис. 2.17 Діаграма взаємодії компонентів та пристроїв системи безпеки

Система взаємодії Arduino MEGA 2560 з елементами, які включають датчик руху HC-SR501, LCD-дисплей з двома рядами по 16 символів, мембранну клавіатуру з 12 кнопками, GSM-модуль SIM800L, п'єзоелектричний випромінювач звуку, заземлення та перетворювач, є такою:

- **Arduino MEGA 2560:** Arduino MEGA 2560 є мікроконтролером, який забезпечує обробку даних та управління всіма елементами системи безпеки. Він має достатньо вхідних та вихідних портів для підключення всіх необхідних компонентів.
- **Датчик руху HC-SR501:** Цей датчик руху використовується для виявлення руху в зоні його покриття. Він може передавати сигнали до Arduino MEGA 2560, коли відбувається рух, що дозволяє активувати необхідні заходи безпеки.
- **LCD-дисплей:** LCD-дисплей з двома рядами по 16 символів використовується для відображення інформації, такої як статус системи, повідомлення про події або інструкції для користувача. Arduino MEGA 2560 забезпечує комунікацію з дисплеєм та керування відображеними даними.
- **Мембранна клавіатура:** Мембранна клавіатура з 12 кнопками використовується для введення команд або паролю для активації та керування системою безпеки. Arduino MEGA 2560 отримує сигнали від клавіатури та виконує відповідні дії в залежності від отриманої команди.
- **GSM-модуль SIM800L:** GSM-модуль забезпечує можливість передачі і отримання SMS-повідомлень. Це дозволяє системі безпеки повідомляти користувача про події, які відбуваються, а також приймати команди через SMS для керування системою.
- **П'єзоелектричний випромінювач звуку:** П'єзоелектричний випромінювач звуку використовується для генерації звукових сигналів, наприклад, для сирени або сигналізації про події. Arduino MEGA 2560 контролює роботу випромінювача звуку і активує його за необхідності.
- **Заземлення:** Заземлення є важливою складовою системи безпеки, оскільки воно допомагає уникнути електричного удару та забезпечує безпечну роботу всіх електричних пристроїв.
- **Перетворювач:** Перетворювач використовується для забезпечення необхідної енергії для системи безпеки, наприклад, для забезпечення

стабільного живлення від джерела енергії або для конвертації сигналів між різними форматами.

Ці компоненти працюють разом, спілкуючись з Arduino MEGA 2560, що дозволяє контролювати та керувати системою безпеки приміщення.

Програмний код на Arduino забезпечує взаємодію між всіма елементами та виконання необхідних дій залежно від отриманих сигналів і команд.

2.6. Висновки до розділу 2

У цьому розділі були розглянуті основні пристрої, комплекти та технічне забезпечення, яке необхідне для розробки системи безпеки в приміщенні.

При виборі плати Arduino необхідно враховувати функції майбутньої системи та модулі, які будуть підключатися до неї. На ринку доступно багато варіантів плат Arduino різних розмірів.

Інфрачервоні датчики дозволяють виявляти рух і часто використовуються у системах сигналізації. Вони мають малий розмір, низьку вартість, низьке енергоспоживання, легкі у використанні і малопіддаються зносу.

GSM-модуль в проектах Arduino дозволяє здійснювати зв'язок з віддаленими автономними пристроями через мобільну мережу. За допомогою цього модуля можна надсилати команди на пристрої та отримувати інформацію з них за допомогою SMS-команд.

Для проектування апаратної частини системи були обрані та обгрунтовані такі компоненти системи: Arduino Mega в якості основи пристрою, інфрачервоний датчик руху HC-SR501 для виявлення руху, GSM-модуль SIM800L для передачі повідомлень і віддаленого управління, 12-кнопкова клавіатура та LCD-дисплей як інтерфейс керування системою.

Для проєктування та розробки системи в Розділі 2 були вибрані необхідні компоненти, які використовуються в заданій структурі. На рис. 3.1 в якості основи пристрою була обрана платформа Arduino Mega. Для виявлення руху був використаний інфрачервоний датчик руху HC-SR501. Для керування системою використовується LCD-дисплей з двома рядами по 16 символів і мембранна клавіатура з 12 кнопками. Для надсилання повідомлень власнику був вибраний GSM-модуль SIM800L. Для реалізації звукової сирени використовується п'єзоелектричний випромінювач звуку.

Першим етапом проєктування системи є побудова схеми з'єднань між усіма компонентами системи та мікроконтролером. Схема з'єднань пристроїв наведена на рис. 3.2.

На цій схемі червоними лініями позначено живлення, чорними - заземлення. Зеленими та синіми лініями показано з'єднання між GSM-модулем та платою Arduino. Жовтими лініями показане з'єднання клавіатури і плати. Білими і сірими лініями показано з'єднання між Arduino і ІІС/I2C інтерфейсною шиною, яка підключена до дисплея. Коричневою лінією плату з'єднується з цифровим виходом датчика руху. Після побудови схеми з'єднань необхідно зібрати реальний пристрій, з'єднавши всі компоненти згідно з цією схемою.

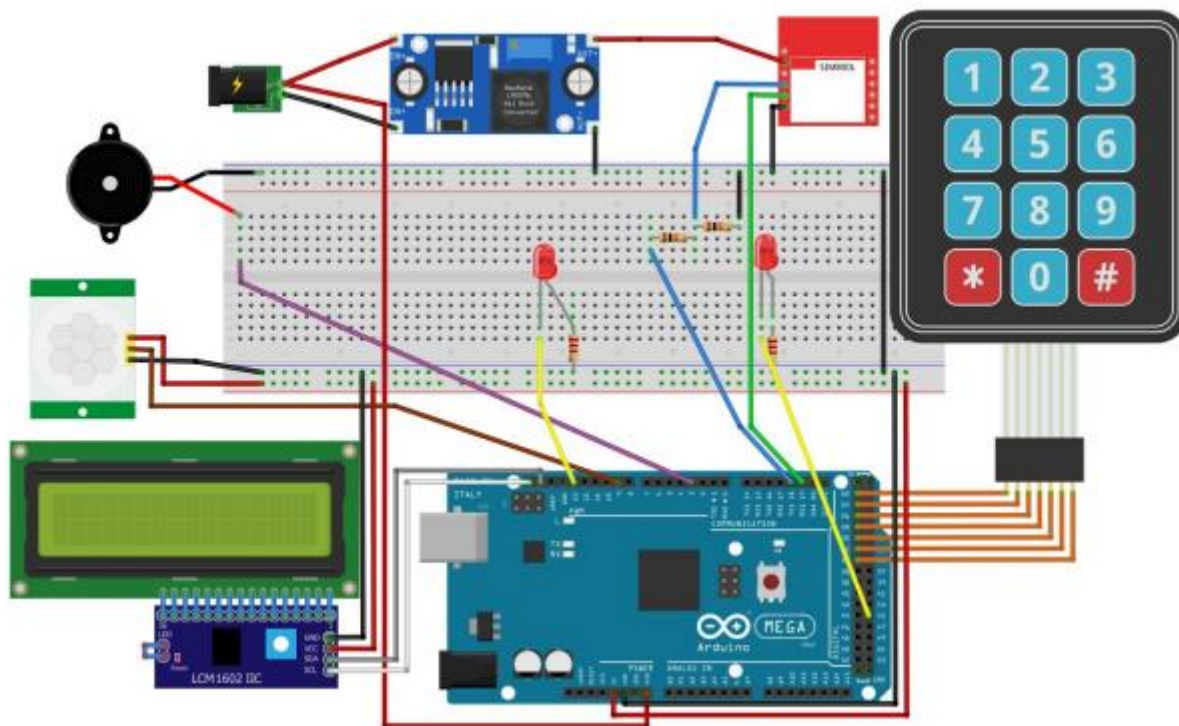


Рис. 3.2. Модель підключення пристроїв

В даному розділі при створенні заданої архітектури ІУС було здійснено програмування функціоналу системи здійснюється шляхом підключення мікроконтролера до комп'ютера та завантаження на нього скетчу з виконуваним кодом.

У початковій частині скетчу оголошуються підключені бібліотеки, макроси, константи та глобальні змінні.

В роботі наводяться розроблені автором коди програм:

```
#include
<LiquidCrystal_I2C.h>
#include <Wire.h>
#include <Keypad.h>
#include
<TinyGsmClient.h>
#include
```

```

<EEPROM.h> #define
LedPin 45
#define SoundPin
3 #define
SerialMon Serial
#define SerialAT
Serial1
TinyGsm
modem(SerialAT);
String _response;
String phone = "+38", _phone = "";
String
retrievedString;
String pin = "", _pin
= ""; String buttons;
const byte ROWS = 4; // число стовпчиків
клавіатури const byte COLS = 3; // число
стовпчиків клавіатури
char hexaKeys[ROWS][COLS] = {
    {'1', '2', '3'},
    {'4', '5', '6'},
    {'7', '8', '9'},
    {'*', '0', '#'}
};
byte rowPins[ROWS] = {22, 24, 26, 28};
byte colPins[COLS] = {30, 32, 34};
Keypad customKeypad = Keypad( makeKeymap(hexaKeys), rowPins,
colPins, ROWS, COLS);
LiquidCrystal_I2C lcd(0x27, 16, 2);
bool access = false,

```

```
alarm = false,  
waiting = false,  
smsSend = false,  
signalization = false;  
int count= 0;
```

У початкових рядках цього кодового фрагмента відбувається підключення необхідних бібліотек, які надають доступ до потрібних нам методів.

При програмуванні була вибрана бібліотека LiquidCrystal_I2C призначена для роботи з символьними графічними дисплеями, що підключаються до Arduino за допомогою апаратної шини I2C. Шина I2C є послідовною асиметричною шиною для зв'язку між електронними пристроями всередині пристроїв. Вона використовує дві двонаправлені лінії зв'язку (SDA і SCL), які з'єднують низькошвидкісні периферійні компоненти з процесорами і мікроконтролерами.

Також вибрана бібліотека Wire, яка використовується для зв'язку мікроконтролера з пристроями і модулями через інтерфейс I2C. Для зв'язку по I2C використовуються всього два контакти: лінія даних (SDA) і лінія тактового сигналу (SCL). За допомогою роз'ємів Arduino можна підключити до 120 пристроїв, які підтримують інтерфейс I2C. Для обміну даними з такими пристроями потрібна бібліотека Wire.

Бібліотека Keypad необхідна для використання матричних клавіатур разом з Arduino.

Бібліотека TinyGsmClient є бібліотекою Arduino для модулів GSM/GPRS. Бібліотека EEPROM використовується для роботи з енергонезалежною пам'яттю EEPROM. EEPROM (Electrically Erasable Programmable Read-Only Memory) - це тип пам'яті, в якому інформація зберігається навіть після вимкнення пристрою. Ця бібліотека дозволяє записувати і зчитувати дані з цієї пам'яті.

Далі, за допомогою директиви препроцесора #define, створюються макроси заміни коду для зручного доступу до виходів датчиків та для відправлення АТ-

команд GSM-модулю. Потім визначаються глобальні змінні типу String, в які записується інформація, що надходить від GSM-модулю, а також змінні для збереження номера телефону, пін-коду і натиснутих кнопок клавіатури. Константні змінні визначають кількість стовпців і рядків клавіатури, а також масив типу char, що містить значення кнопок клавіатури. Масиви типу byte містять номери входів плати мікроконтролера, до яких підключені проводи, що відповідають рядкам та стовпцям клавіатури. Потім створюється змінна типу Keypad, за допомогою якої буде визначатися, яка кнопка була натиснена. На останок визначаються змінні, які будуть потрібні під час роботи програми.

Далі йде блок коду, що задає основні налаштування:

```
pinMode(SoundPin,  
OUTPUT);  
pinMode(LedPin,  
OUTPUT); lcd.begin();  
lcd.backlight();
```

У початкових рядках коду використовується функція pinMode, щоб встановити режими роботи цифрових виходів на Arduino. Останні два рядки ініціалізують LCD екран та вмикають його підсвічування.

Для зберігання даних використовується EEPROM - енергонезалежна пам'ять. Вона призначена для зберігання інформації, такої як номер телефону, пін-код та статус сигналізації. Збереження цих даних в EEPROM необхідне в разі перебоїв енергопостачання. Таким чином, система завжди матиме доступ до номера телефону та пін-коду, а статус сигналізації дозволить системі активувати сигналізацію, якщо вона була активна до відключення живлення.

При кожному запуску системи відбувається зчитування енергонезалежної пам'яті та перевірка наявності номера телефону, пін-коду і статусу сигналізації, який був записаний в EEPROM при активації або деактивації сигналізації.

```
String newPhone, newPin, newSig;  
phoneAddrOffset = readStringFromEEPROM(0, &newPhone);
```

```

pinAddrOffset = readStringFromEEPROM(phoneAddrOffset,
&newPin); readStringFromEEPROM(pinAddrOffset, &newSig);
delay(1000);

if (newPhone.startsWith("+380"))
{
    phone = newPhone;
}
if (newPin.startsWith("pin"))
{
    pin = newPin.substring(3);
}

```

Функція *int readStringFromEEPROM (int addrOffset, String *strToRead)* зчитує з пам'яті рядок за вказаною адресою зсуву та повертає його. Якщо номер телефону та пін-код ще не записані, то запускаються функції для їх введення та запам'ятовування.

```

if (phone == "+38" )
{
    lcd.setCursor(0, 0);
    lcd.print("Enter phone
number"); while (phone ==
"+38")
{
        EnterPhone();
    }
}
if (pin == "")
{
    lcd.clear();
    lcd.setCursor(0, 0);

```

```

        lcd.print("Create a
        pin"); while (pin
        == "")
        {
            CreatePin();
        }
    }
}

```

Дві функції, EnterPhone() і CreatePin(), дають користувачу можливість ввести номер телефону та пін-код відповідно. Після введення даних викликається функція writeStringToEEPROM, яка здійснює запис рядка за вказаною адресою.

Функція writeStringToEEPROM має такий прототип: int writeStringToEEPROM(int addrOffset, String &strToWrite). Вона використовується для запису рядка в EEPROM за зсувом відносно початку пам'яті.

Налаштування GSM модуля

```

SerialAT.begin(9600);
lcd.setCursor(0,0);
lcd.print("Activating");
lcd.setCursor(0,1);
lcd.print("GSM module");
do {
    _response = sendATCommand("AT", true);
    _response.trim();
} while (_response !=
"OK"); do {

```

```

    _response = sendATCommand("AT+CMGF=1;&W", true);
    _response.trim();
} while (_response != "OK");
Serial.println("SIM800L
enabled"); lcd.clear();

```

Рядок `SerialAT.begin(9600)` встановлює швидкість обміну даними з GSM-модулем.

У рядку `_response = sendATCommand("AT", true)` функція `sendATCommand` відправляє команду "AT" GSM-модулю для налаштування швидкості обміну даними і зберігає відповідь модуля в змінну `_response`. Команда буде відправлятися до тих пір, поки модуль не відповість "OK".

У рядку `_response = sendATCommand("AT+CMGF=1;&W", true)` відправляється команда "AT+CMGF=1;&W" для увімкнення текстового режиму SMS. Після отримання відповіді "OK" від модуля, налаштування буде завершено.

Активація та деактивація сигналізації виконується шляхом введення пін-коду на матричній клавіатурі. Якщо пін-код введено правильно, то сигналізація буде активована або деактивована в залежності від поточного стану системи.

```

char customKey =
customKeypad.getKey(); if
(customKey)
{
    buttons      +=
        customKey;
    lcd.setCursor(
        k,      1);
}

```

```

        lcd.print('*');
k = k + 1; if (k == 4)
{
    lcd.setCursor(0, 1);
    currentMillis = millis();
    previousMillis =
    currentMillis;
    while (currentMillis - previousMillis < 5000)
    {
        currentMillis = millis();
        customKey =
        customKeypad.getKey();
        Serial.println(customKey);
        delay(100);
        if (customKey == '#')
        {
            access
            = true;
            break;
        }
        else
        {
            access = false;
        }
    }
}
}

```

Система постійно очікує введення пін-коду за допомогою рядка `char customKey = customKeypad.getKey()`. Значення натиснутої кнопки зберігається в змінній

customKey. Якщо натиснута кнопка є цифрою, вона додається до змінної buttons за допомогою рядка buttons += customKey.

Після введення 4 цифр система очікує натискання кнопки "#" протягом 5 секунд для перевірки пін-коду.

```
    if (buttons == pin)
{
    if (signalization == true)
    {
        waiting = false;
        lcd.print("Alarm
diactivated");
        SignalizationOff();
    }
    else
    {
        lcd.print("Alarm
activated"); waiting =
true;
        waitingOn    =
millis();
        waitingOff    =
millis();
        SignalizationO
n();
    }
    delay(
1000);
    lcd.cle
ar(); k
= 0;
```

```

        buttons = "";
    }
    else
    {
        lcd.clear();
        lcd.setCursor(0, 1);
        lcd.print("Access
        denied!");
        delay(1000);
        lcd.cl
        ear()
        ; k =
        0;
        buttons = "";
    }

```

Якщо коректний пін-код введено і сигналізація була увімкнена, виконується функція `SignalizationOff()`, яка вимикає сигналізацію та зберігає її стан у EEPROM. Змінна "waiting" вказує, чи потрібно активувати звуковий сигнал та надати час для виходу з приміщення.

Якщо коректний пін-код введено і сигналізація була вимкнена, виконується функція `SignalizationOn()`, яка увімкнює сигналізацію та зберігає її стан у EEPROM.

Взаємодія з GSM-модулем.

```

if (SerialAT.available()) {
    if (_response.startsWith("+CMTI:")) {
        int index = _response.lastIndexOf(",");
        String result = _response.substring(index+1, _response.length());
        result.trim();
        _response=sendATCommand("AT+CMGR="+result,
            true); parseSMS(_response);
        sendATCommand("AT+CMGDA=\"DEL ALL\"", true);
    }
}

```

У рядку `if (_response.startsWith("+CMTI:"))` перевіряється наявність нового SMS, а потім його вміст отримується. Далі викликається функція `parseSMS()`, якій передається зміст повідомлення для витягування номера телефону та команди. Якщо номер телефону належить власнику, виконується надіслана команда.

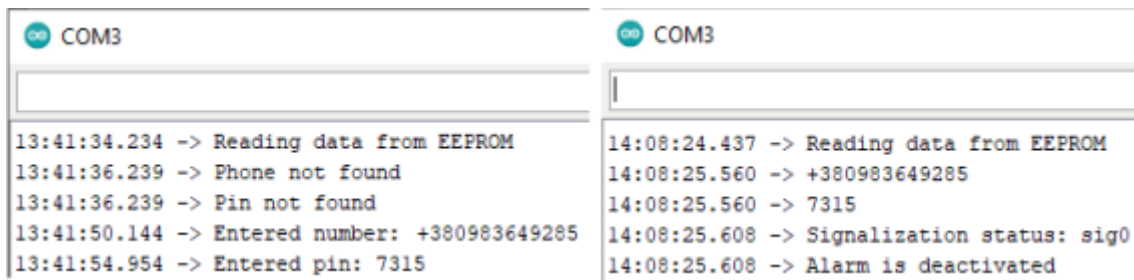
Після цього відправляється команда модулю для видалення повідомлення за допомогою `sendATCommand("AT+CMGDA="DEL ALL", true)`. Повідомлення потрібно видаляти, оскільки вони займають місце в обмеженій пам'яті SIM-карти.

Весь програмний код для системи безпеки розташований у додатку.

3.2. Тестування системи

Для тестування роботи системи використовується вбудоване середовище розробки Arduino IDE. Для цього було створено спеціальний скетч, який виводить інформацію в послідовний порт.

Тест №1. Перевірка запису та зчитування даних з енергонезалежної пам'яті. У пам'яті немає жодних записів. З клавіатури вводяться номер телефону та пін-код, після чого живлення вимикається.

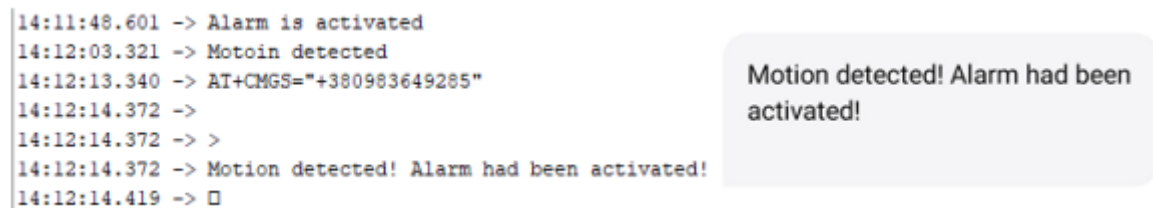


```
COM3
13:41:34.234 -> Reading data from EEPROM
13:41:36.239 -> Phone not found
13:41:36.239 -> Pin not found
13:41:50.144 -> Entered number: +380983649285
13:41:54.954 -> Entered pin: 7315

COM3
14:08:24.437 -> Reading data from EEPROM
14:08:25.560 -> +380983649285
14:08:25.560 -> 7315
14:08:25.608 -> Signalization status: sig0
14:08:25.608 -> Alarm is deactivated
```

Рис. 3.2. Результат тесту №1

З аналізу результатів тесту, який показаний на рис. 3.2, видно, що після відновлення живлення система успішно зчитала дані, які були введені раніше і збережені в пам'яті.



```
14:11:48.601 -> Alarm is activated
14:12:03.321 -> Motion detected
14:12:13.340 -> AT+CMGS="+380983649285"
14:12:14.372 -> 
14:12:14.372 -> >
14:12:14.372 -> Motion detected! Alarm had been activated!
14:12:14.419 -> □

Motion detected! Alarm had been activated!
```

Рис. 3.3. Результат тесту №2

За результатами, які представлені на рис. 3.3, система виявила рух і надала 10 секунд для вимкнення сигналізації. Після цього вона відправила повідомлення на мобільний номер власника.

Тест №3. Перевірка роботи команди «Turn off».

```

14:14:47.398 -> +CMTI: "SM",1
14:14:47.398 -> AT+CMGR=1
14:14:48.519 ->
14:14:48.519 -> +CMGR: "REC UNREAD","+380983649285","", "21/05/29,14:14:38+12"
14:14:48.613 -> Turn off
14:14:48.613 ->
14:14:48.613 -> OK
14:14:48.613 ->
14:14:48.613 -> Phone: +380983649285
14:14:48.660 -> Message: Turn off
14:14:48.660 -> Alarm deactivation
14:14:48.660 -> Alarm is deactivated
14:14:48.707 -> AT+CMGDA="DEL ALL"
14:14:49.735 ->
14:14:49.735 -> OK

```

Рис. 3.4. Результат тесту №3

За результатами, які показані на рис. 3.4, видно, що система успішно отримала і обробила повідомлення від номера власника, і після цього сигналізація була вимкнена. Тест №4. Перевірка роботи команди "Turn on".

```

14:15:29.724 -> +CMTI: "SM",1
14:15:29.771 -> AT+CMGR=1
14:15:30.852 ->
14:15:30.852 -> +CMGR: "REC UNREAD","+380983649285","", "21/05/29,14:15:20+12"
14:15:30.944 -> Turn on
14:15:30.944 ->
14:15:30.944 -> OK
14:15:30.944 ->
14:15:30.944 -> Phone: +380983649285
14:15:30.991 -> Message: Turn on
14:15:30.991 -> Alarm activation
14:15:30.991 -> Alarm is activated
14:15:31.037 -> AT+CMGDA="DEL ALL"
14:15:32.060 ->
14:15:32.060 -> OK

```

Рис. 3.5. Результат тесту №4

За результатами, які показані на рис. 3.5, видно, що система успішно отримала та обробила повідомлення від номера власника телефону, і після цього сигналізація була ввімкнена. Тест №5. Перевірка роботи команди "Turn off".

Тест №5. Перевірка роботи команди «Status».

```
14:20:21.436 -> +CMTI: "SM",2
14:20:21.436 -> AT+CMGR=2
14:20:22.557 ->
14:20:22.557 -> +CMGR: "REC UNREAD","+380983649285","", "21/05/29,14:20:12+12"
14:20:22.602 -> Status
14:20:22.602 ->
14:20:22.602 -> OK
14:20:22.649 ->
14:20:22.649 -> Phone: +380983649285
14:20:22.649 -> Message: Status
14:20:22.649 -> AT+CMGS="+380983649285"
14:20:23.672 ->
14:20:23.672 -> >
14:20:23.672 -> Signalization is inactive
14:20:23.719 -> □
14:20:24.757 -> Signalization is inactive
14:20:24.804 -> >
14:20:24.804 -> >
14:20:24.804 -> AT+CMGDA="DEL ALL"
14:20:27.515 ->
14:20:27.515 -> +CMGS: 47
14:20:27.515 ->
14:20:27.515 -> OK
```

Рис. 3.6. Результат тесту №5

За результатами, які показані на рис. 3.6, видно, що система успішно отримала і обробила команду від номера власника, і після цього відправила інформацію про статус системи. Тест №6. Перевірка роботи команди "Status".

3.3. Керівництво користувача

Налаштування системи. При першому включенні системи, вам буде запропоновано ввести номер телефону, на який будуть надсилатися повідомлення, а також з якого номера ви зможете відправляти команди для керування системою. Після введення номера ви зможете підтвердити, що номер введено правильно, натиснувши клавішу "#" або скасувати введення, натиснувши "*" у разі, якщо номер був введений неправильно.



Рис. 3.7. Введення номера телефону користувача

Після введення номера телефону, користувачеві пропонується створити 4-значний пін-код для активації/деактивації сигналізації. Після введення пін-коду, можна підтвердити правильність введення, натиснувши клавішу "#", або скасувати введення, натиснувши "*", якщо бажаєте змінити пін-код. Зараз система готова до роботи.



Рис. 3.8. Введення PIN-code користувачем

Для активації сигналізації потрібно ввести пін-код. Якщо пін-код введено правильно, на екрані з'явиться повідомлення про активацію сигналізації.



Рис. 3.9. Активація сигналізації системи безпеки

Після введення правильного пін-коду система активуватиме попереджувальний звуковий сигнал і надасть 30 секунд на виход з приміщення.

Для деактивації сигналізації, після повернення до дому, датчик руху виявить рух і вам буде надано 30 секунд на вимкнення сигналізації.

При правильному введенні пін-коду на екрані з'явиться повідомлення про деактивацію сигналізації.



Рис. 3.10. Деактивація сигналізації системи безпеки

Команди для керування системою доступні для віддаленого керування і відправляються на номер системи.

Нижче наведені доступні команди:

Команди для керування системою. Для віддаленого керування системою можна використати команди, які необхідно відправити на номер системи. Доступні команди:

«*Turn on*» увімкне сигналізацію;

«*Turn off*» вимкне сигналізацію;

«*Commands*» надасть список команд;

«*Status*» надасть інформацію про стан сигналізації на момент користування (активована або деактивована).

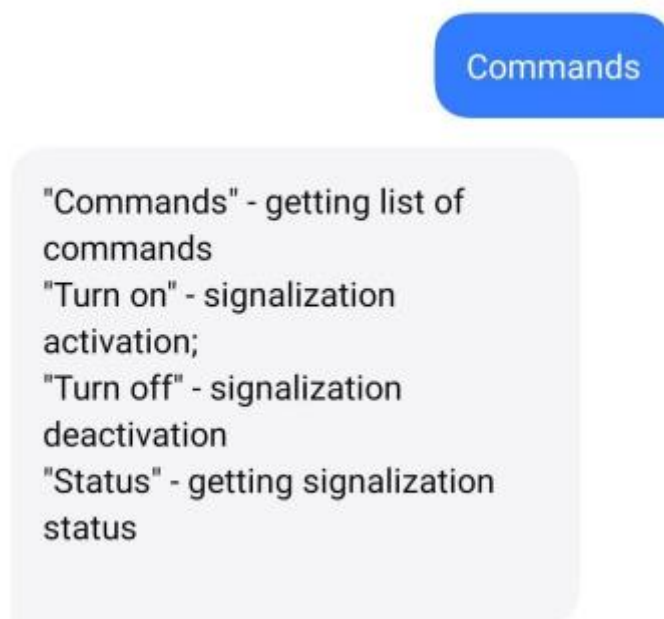


Рис. 3.11. Результат відправлення команди "Commands"

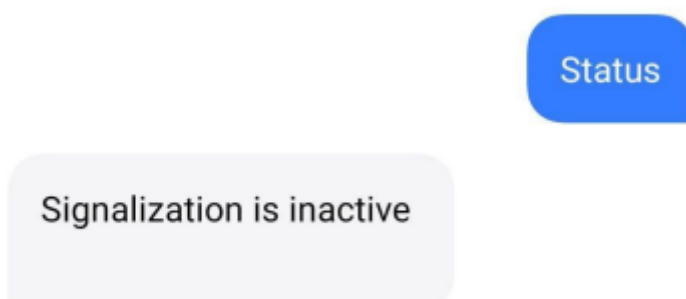


Рис. 3.12. Результат відправки команди "Status", на момент, якщо сигналізація деактивована.

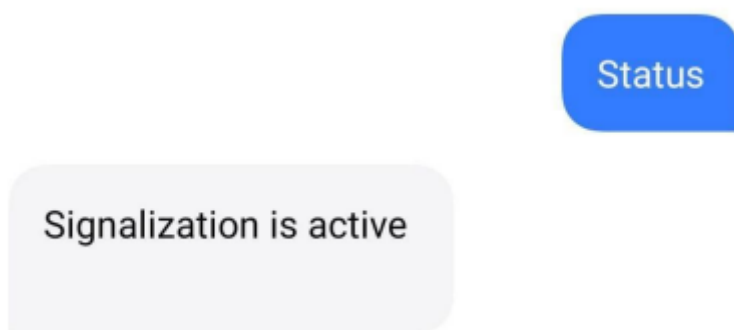


Рис. 3.13. Результат відправки команди "Status", на момент, якщо сигналізація активована

При виявленні руху і невимкненні сигналізації протягом 30 секунд, система активує постійний звуковий сигнал, а власнику системи буде надіслано повідомлення про виявлений рух.

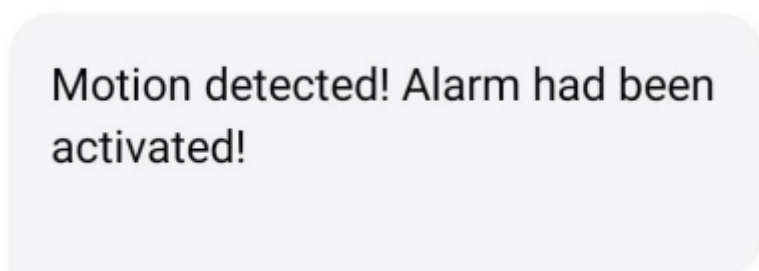


Рис. 3.14. Повідомлення сестеми про виявлення руху

3.4 Технічне забезпечення системи

Технічне забезпечення ІУС включає в себе серверну та інтерфейсну частини, а також систему моніторингу даних безпеки приміщення. Розглянути класичні сервера, інтерфейси, та систему моніторингу.

Вимоги до комп'ютера та операційної системи:

Система може бути використана або розроблена на персональному комп'ютері, або завантажена на окремий сервер.

Мінімальні характеристики:

ІВМ сумісний комп'ютер класу Intel;

Об'єм оперативної пам'яті (ОЗП) – 1 Гбайт;

Об'єм доступного дискового простору – не менше 250 Мбайт;

Операційна система – Microsoft Windows 8 і вище, Linux, Ubuntu, OS X;

Наявність програмних засобів – програмне забезпечення графічної візуалізації graphviz v.2.38 і вище та Java Runtime Environment (JRE) 8 і вище.

Рекомендовані характеристики:

Персональний ІВМ сумісний комп'ютер класу Intel Core i5 2xxx і вище;

Об'єм оперативної пам'яті (ОЗП) – 4 Гбайт;

Об'єм вільного дискового простору – 1 Гбайт;

Операційна система – Microsoft Windows 8.1, 10;

Наявність програмних засобів – програмне забезпечення графічної візуалізації graphviz v.2.38 і вище та Java Runtime Environment (JRE) 8 і вище.

Обробка даних безпеки приміщення накопичується в стандартній системі моніторингу.

3.5 Реалізація автоматизованої системи безпеки приміщення

Під час виконання роботи вирішені наступні завдання:

1. Проведено аналіз існуючих систем безпеки та докладно дослідженні різні підходи до забезпечення безпеки приміщень. Використовуючи цей аналіз, було ідентифіковано основні функції, які повинна реалізовувати розроблювана система безпеки. Додатково, були оцінені переваги і недоліки кожного підходу, а також виявлені потенційні виклики та обмеження, які можуть виникнути під час реалізації цих функцій.
2. Проведено детальний аналіз різних типів мікроконтролерів на базі Arduino з метою вибору найбільш ефективного для використання в особистому проєкті системі безпеки приміщення. При цьому розглядалися такі фактори, як продуктивність, вартість, доступність, розширюваність та екосистема підтримки. На основі цього аналізу було прийнято рішення про використання пристрою на базі Arduino, який найкраще задовольняв вимоги проєкту з точки зору функціональності, ефективності та простоти реалізації
3. Для забезпечення функціональності системи безпеки, у третьому розділі було написано програмний код для апаратної частини у середовищі Arduino IDE.
4. Розроблений програмний код був налаштований та протестований на контрольних прикладах з метою перевірки його працездатності та відповідності вимогам проєкту. В процесі тестування системи були проведені різноманітні експерименти, зокрема перевірка працездатності датчика руху та GSM-модуля, а також перевірка правильної роботи системи після відновлення живлення. Додатково, було проведено тестування віддаленого керування системою з метою переконатися в її зручності та надійності.
5. З метою забезпечення зручного користування розробленої системи, було розроблено докладне керівництво користувача, яке містить інструкції щодо

використання різних функціональностей системи. У цьому керівництві надаються пояснення щодо процедур налаштування, розпізнавання та реагування на події безпеки, а також наголошуються заходи безпеки, які необхідно дотримуватися під час експлуатації системи. Керівництво користувача доповнюється зрозумілими ілюстраціями та діаграмами, щоб забезпечити максимальну ясність та зручність для користувача.

6. Запропоновано архітектурне забезпечення побудови автоматизованої системи безпеки приміщення та здійснена його реалізація, особливо в частині апаратно-програмної платформи Arduino, зокрема сконструйовано з'єднання комплекту необхідних датчиків, модулю GSM, Arduino IDE, п'єзоелектричного випромінювача звуку, перетворювача та інші пристроїв.

ВИСНОВКИ

При написанні Комплексної Магістерської Роботи була спроектована автоматизована система безпеки приміщення на базі платформи Arduino та стандартних компонентів ІУС.

Під час виконання роботи були вирішені наступні завдання:

- Проведено аналіз існуючих систем безпеки, виділивши основні функції, які слід реалізувати.
- Проаналізовано різні типи мікроконтролерів на базі Arduino.
- Досліджено пристрої, що використовувалися під час розробки системи.
- Проаналізовано інструменти для проєктування програмної частини проєкту, що використовувалися під час розробки системи.
- Написано програмний код апаратної частини системи безпеки у середовищі Arduino IDE.
- Успішно протестовано систему і розроблено керівництво користувача.

Система безпеки приміщення - це система, яка забезпечує безпеку приміщень на підприємстві комерційного або спеціального призначення. Рівень безпеки залежить від набору встановлених датчиків. Кожна система безпеки передбачає наявність керуючого пристрою, клавіатури для управління сигналізацією, звукового сигналу тривоги і датчиків руху - по одному на кожну кімнату. В професійних системах також можуть бути встановлені датчики газу (у приміщенні з газовим обладнанням), датчики диму (у приміщеннях, де є ризик пожежі), датчики затоплення (у приміщеннях з комунікаціями), датчики розбиття скла і датчики відкриття дверей.

Запропоновано архітектурне забезпечення побудови автоматизованої системи безпеки приміщення та здійснена його реалізація, особливо в частині апаратно-програмної платформи Arduino, зокрема сконструйовано з'єднання комплекту необхідних датчиків, модулю GSM, Arduino IDE, п'єзоелектричного випромінювача звуку, перетворювача та інші пристрої.

У межах КМР була розроблена система безпеки на основі плати Arduino Mega, з використанням інфрачервоного датчика руху, GSM-модуля, клавіатури для введення пін-коду і номера телефону, дисплея і п'єзоелектричного динаміка, який відтворює звуковий сигнал тривоги. З'єднання всіх компонентів системи було здійснено за допомогою дротів, що є більш економічним рішенням, ніж бездротові технології зв'язку. Система здатна виявляти рух і повідомляти про це власника приміщення, а також має можливість дистанційного керування за допомогою SMS-команд.

Подальша рекомендація модифікації та використання системи; розроблена система буде корисна для широкого кола користувачів в соціально-побутовій сфері як фактор забезпечення безпеки їнього житла, яке тривалий час залишається без нагляду для переміщених осіб, що характерно для сучасних подій в режимі воєнного стані.

ДОДАТОК

```
include <LiquidCrystal.h>
include <Keypad.h>

include <SoftwareSerial.h>

const int motionSensorPin = 2;
const int buzzerPin = 3;

LiquidCrystal lcd(8, 9, 4, 5, 6, 7);

const byte ROWS = 4;
const byte COLS = 3;
char keys[ROWS][COLS] = {
  {'1', '2', '3'},
  {'4', '5', '6'},
  {'7', '8', '9'},
  {'*', '0', '#'}
};
byte rowPins[ROWS] = {A0, A1, A2, A3};
byte colPins[COLS] = {A4, A5, A6};
Keypad = Keypad(makeKeymap(keys), rowPins, colPins, ROWS, COLS);

SoftwareSerial gsmSerial(10, 11);

void setup() {
  pinMode(motionSensorPin, INPUT);
  pinMode(buzzerPin, OUTPUT);

  lcd.begin(16, 2);
  gsmSerial.begin(9600);

  lcd.setCursor(0, 0);
  lcd.print("Security System");
  lcd.setCursor(0, 1);
  lcd.print("Initializing...");
  delay(2000);
  lcd.clear();
}

void loop() {
  int motion = digitalRead(motionSensorPin);

  if (motion == HIGH) {
    activateAlarm();
  }
}
```

```

    }

    char key = keypad.getKey();

    if (key) {
        processCode(key);
    }
}

void activateAlarm() {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Intrusion Detected!");
    lcd.setCursor(0, 1);
    lcd.print("Alarm Activated");

    digitalWrite(buzzerPin, HIGH);

    gsmSerial.println("Intrusion Detected!");
    delay(1000);

    digitalWrite(buzzerPin, LOW);
    lcd.clear();
}

void processCode(char key) {

```