

АНОТАЦІЯ

У даній дипломній роботі розглянуті питання проектування, моделювання та розробки комп'ютерної мережі з використанням технології Інтернет речей. В якості об'єкту розробки був обраний одноповерховий житловий будинок, а для реалізації комп'ютерної мережі з інфраструктурою IoT вирішуються питання створення підсистем Smart Home, таких як клімат-контроль, управління освітленням і пожежогашінням, відеоспостереження і контроль загазованості повітря.

Для проектування та моделювання працездатності комп'ютерної мережі інтернет речей використовується середовище симулювання Cisco Packet Tracert 7.0.

У першому розділі дипломної роботи визначені базові поняття і спрощена архітектура мережі IoT, виконано огляд типових компонентів такої мережі.

Другий розділ присвячений проектуванню і моделюванню роботи підсистем Smart Home в Cisco Packet Tracert 7.0. В результаті отримана функціонуюча логічна схема мережі Інтернет речей з урахуванням вимог усіх підсистем Smart Home.

У третьому розділі описано обране типове обладнання комп'ютерної мережі та необхідні Інтернет речі для реалізації мережі Інтернет речей на що розробляється об'єкті, визначено кошторис проекту.

Отримані схеми моделювання в Cisco Packet Tracert 7.0 підсистем Smart Home можуть бути використані в навчальному процесі.

За результатами дипломної роботи опубліковані тези конференції.

ABSTRACT

In this abstract, the issues of designing, modeling and developing a computer network using the Internet of things technology are considered. As a development object, a one-story residential house has been chosen, and for the implementation of a computer network with an IoT infrastructure, the issues of creating Smart Home subsystems, such as climate control, lighting and fire fighting control, video surveillance and gas contamination control are being addressed.

To conduct the design and simulation of the operability of the computer network of things Internet, the Cisco Packet Tracer 7.0 simulation environment is used.

In the first chapter of the abstract the basic concepts and simplified architecture of the IoT network are defined, an overview of the typical components of such a network is made.

The second chapter is devoted to the design and modeling of Smart Home subsystems in Cisco Packet Tracer 7.0. As a result, we have obtained an effective logical scheme of the Internet of things, taking into account the requirements of all Smart Home subsystems.

The third chapter describes and selects the typical equipment of a computer network and the necessary Internet things for the realization of the Internet of things on the developed object, the estimated cost of the project is determined.

The resulting simulation schemes in the Cisco Packet Tracer 7.0 Smart Home subsystems can be used in the learning process.

According to the results of the thesis, the theses of the conference are published.

АННОТАЦИЯ

В данной дипломной работе рассмотрены вопросы проектирования, моделирования и разработки компьютерной сети с использованием технологии Интернет вещей. В качестве объекта разработки выбран одноэтажный жилой дом, а для реализации компьютерной сети с инфраструктурой IoT решаются вопросы создания подсистем Smart Home, таких как климат-контроль, управление освещением и пожаротушением, видеонаблюдение и контроль загазованности воздуха.

Для проведения проектирования и моделирования работоспособности компьютерной сети интернет вещей используется среда симулирования Cisco Packet Tracer 7.0.

В первой главе дипломной работы определены базовые понятия и упрощенная архитектура сети IoT, выполнен обзор типовых компонентов такой сети.

Вторая глава посвящена проектированию и моделированию работы подсистем Smart Home в Cisco Packet Tracer 7.0. В результате получена действующая логическая схема сети Интернет вещей с учетом требований всех подсистем Smart Home.

В третьей главе описаны и выбраны типовое оборудование компьютерной сети и необходимые Интернет вещи для реализации сети Интернет вещей на разрабатываемом объекте, определена сметная стоимость проекта.

Полученные схемы моделирования в Cisco Packet Tracer 7.0 подсистем Smart Home могут быть использованы в учебном процессе.

По результатам дипломной работы опубликованы тезисы конференции.

ЗМІСТ

	стр.
ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	7
ВСТУП	8
1 ТЕОРЕТИЧНІ АСПЕКТИ ПИТАНЬ ОРГАНІЗАЦІЇ КОП'ЮТЕРНОЇ МЕРЕЖІ З ІНФРАСТРУКТУРОЮ ІoT.	11
1.1 Поняття і визначення технології інтернет речей	11
1.2 Огляд ключових об'єктів контролю і управління в комп'ютерній мережі з Інтернет речами	16
2 ПРОЕКТУВАННЯ КОМП'ЮТЕРНОЇ МЕРЕЖІ ІНФРАСТРУКТУРОЮ ІoT	25
2.1 Постановка завдання проекту	25
2.2 Проектування комп'ютерної мережі в середовищі симуляції Cisco Packet Tracer	27
2.3 Проектування підмережі системи клімат-контролю	31
2.3.1 Побудова структурно-логічної схеми підмережі	33
2.3.2 Програмування функцій термостата	38
2.3.3 Реєстрація на сервері і моніторинг роботи підмережі	39
2.4 Проектування підмережі системи освітлення	43
2.4.1 Побудова структурно-логічної схеми підмережі	45
2.4.2 Програмування функцій роботи контролера	47
2.4.3 Реєстрація на домашньому шлюзі і моніторинг роботи підмережі	49
2.5 Проектування підмережі системи пожежогасіння	53
2.5.1 Построение структурно-логической подсети	54
2.5.2 Програмування функцій роботи контролера	57
2.5.3 Реєстрація на домашньому шлюзі і моніторинг роботи підмережі	60
2.6 Проектування підмережі системи відео спостереження	61

2.6.1 Побудова структурно-логічної схеми підмережі.....	63
2.6.2 Програмування функцій контролера	66
2.6.3 Реєстрація на сервері і моніторинг роботи підмережі	68
2.7 Проектування підмережі системи контролю рівня загазованості повітря	71
2.7.1 Побудова структурно-логічної схеми підмережі.....	73
2.7.2 Програмування функцій роботи контролера	76
2.7.3 Реєстрація на домашньому шлюзі і моніторинг роботи підмережі	76
2.8 Проект комп'ютерної мережі з інфраструктурою IoT для одноповерхового приватного будинку.....	78
3 РОЗРОБКА КОМП'ЮТЕРНОЇ МЕРЕЖІ З ІНФРАСТРУКТУРОЮ IoT	80
3.1 Опис і вибір обладнання для реалізації комп'ютерної мережі з інфраструктурою IoT.....	80
3.1.1 Пасивне обладнання комп'ютерної мережі	80
3.1.2 Активне обладнання комп'ютерної мережі	82
3.2 Розрахунок кошторису проекту	92
ВИСНОВОК.....	94
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	96
ДОДАТОК А.....	99
ДОДАТОК Б	102
ДОДАТОК В.....	103
ДОДАТОК Г	104
ДОДАТОК Д.....	105

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

Скорочення

IoT – Інтернет вещей (Internet of Things)

БД — бази даних

РП — Робочий простір

TLS — безпека транспортного рівня

ДК — дистанційне керування

SMS — служба коротких повідомлень

PoE(Power over Ethernet) — технологія, що дозволяє передавати віддаленого пристрою електричну енергію разом з даними, через стандартну виту пару в мережі Ethernet.

КМ(комп мережі) - система, що забезпечує обмін даними між обчислювальними пристроями (комп'ютери, сервери, маршрутизатори та інше обладнання).

ВСТУП

В наш час комп'ютерні мережі охоплюють практично всі сфери життєдіяльності людини. Їх впровадження в повсякденне життя повністю виправдало себе: наявність зв'язку між віддаленими станціями забезпечує високу швидкість обміну інформацією між офісами певної компанії не тільки в рамках одного приміщення, міста або країни, але і між континентами. Величезна кількість фінансових, юридичних та подібних операцій були перенесені до цифрового простору і при цьому якість обслуговування тільки примножилась. Віддалене адміністрування серверів, можливість поширення необхідної інформації між сотнями і тисячами співробітників, забезпечення віддалених конференцій, можливість спілкування між людьми з різних точок земної кулі - це далеко не всі переваги і унікальні можливості, які відкрили для людини комп'ютерні мережі.

Інтернет речей (ІоТ, Інтернет Речей) - це концепція «розумного будинку», де багато побутових приладів і систем керуються через мережу Інтернет.

Ідея інтернету речей вперше виникла ще в 1999 році у Кевіна Ештона [1] - дослідника з Массачусетського технологічного інституту, який запропонував концепцію системи управління через Інтернет промисловими об'єктами. Інтернет речей передбачає оснащення кожного пристрою, будь то пілосос, холодильник або пральна машина, модулем підключення до Інтернету з можливістю взаємодії його з домашнім комп'ютером або смартфоном домовласника.

З появою Інтернету речей автоматично вирішується безліч найрізноманітніших проблем: від індивідуального комфорту і безпеки, коли «розумний будинок» буде оцінювати і контролювати власний стан, до ліквідації пробок на дорогах, коли машини самі стануть домовлятися зі світлофорами про оптимальний трафік. Холодильники зможе стежити за

простроченими продуктами, ліки підкажуть час прийому, портфель в дощову погоду нагадує господареві, що той забув парасольку, а автомобіль сам витримає безпечна дистанцію в потоці автомобілів і покаже, де і як краще припаркуватися.

Інтернет речей - це не тільки безліч різних приладів і датчиків, об'єднаних між собою дротяними і бездротовими каналами зв'язку і під'єднаних до мережі Інтернет, а більш тісна інтеграція реального та віртуального світів, в якому спілкування здійснюється між людьми і пристроями.

Інтернет речей стає інструментом для створення принципово нового, «підключеного» світу, з більш тісною взаємодією між предметами, їх середовищем і людьми. Проте, щоб скористатися потенційними перевагами для людей, суспільства і економіки, необхідно вирішувати питання і проблеми, пов'язані з IoT.

Враховуючи вищесказане, стає очевидним те, що проекти і розробки з використанням технологій IoT набувають особливої актуальності і значущості.

Метою даної дипломної роботи є проектування і розробка комп'ютерної мережі з інфраструктурою інтернет речей для реалізації системи розумного будинку.

Для досягнення даної мети необхідно вирішити такі завдання:

- провести огляд сучасних принципів Всеохоплюючої інтернету і технологій реалізації Інтернет речей;
- обрати технології реалізації КМ і підмереж Інтернет речей;
- провести проектування логічної структури КМ(комп'ютерної мережі) і підсистем Smart Home з використанням технології інтернет речей;

- провести конфігурування вузлів мережі і моделювання роботи КМ і підмереж Інтернет речей в середовищі симуляції Cisco Packet Tracer 7,0;
- розробити алгоритми управління і відповідні мікропрограми роботи мікроконтролерів підсистем Smart Home, які розроблюються в середовищі симуляції Cisco Packet Tracer;
- провести тестування роботи інтегрованої КМ з інфраструктурою IoT в середовищі симуляції Cisco Packet Tracer;
- обрати необхідне активне і пасивне устаткування для реалізації КМ з інфраструктурою IoT і визначити кошторис проекту;

ВИСНОВОК

У даній дипломній роботі проведено моделювання працездатності та розробка комп'ютерної мережі з інфраструктурою інтернет речей для реалізації системи Smart Home в одноповерховому житловому будинку.

При виконанні дипломної роботи були вирішені наступні завдання:

1) проведено огляд сучасних принципів всеохоплюючого інтернету, розглянута архітектура мережі IoT і сучасні технології реалізації Інтернет речей;

2) для реалізації локальної КМ і підмереж Інтернет речей обрані технології Ethernet, Wi-Fi;

3) проведено проектування логічної структури КМ, конфігурація вузлів КМ, підмереж інтернет речей і підсистем Smart Home;

4) проведено моделювання роботи підсистем управління клімат-контролем, управління освітленням, управління відеоспостереженням, управління пожежною безпекою, управління рівнем загазованості повітря в середовищі симулювання Cisco Packet Tracer 7.0;

5) для мікроконтролерів підсистем Smart Home, розроблені діаграми процесів і керуючі програми на мові програмування Python, робота яких протестована в середовищі симулювання Cisco Packet Tracer 7.0;

6) тестування роботи КМ з інфраструктурою IoT, що включає всі підсистеми Smart Home проведено в середовищі симулювання Cisco Packet Tracer 7.0;

7) для реалізації мережі Інтернет речей вибрано необхідне активне і пасивне обладнання і визначений кошторис проекту;

Отримані схеми моделювання в Cisco Packet Tracer 7.0 підсистем Smart Home можуть бути використані в навчальному процесі.

На даній роботі представлена доповідь і опубліковані тези на Всеукраїнській конференції студентів та молодих вчених «Інформатика, інформаційні системи і технології» [24].

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Петин В. Датчики для Arduino и Raspberry Pi в проектах Internet Of Things – СПб.: «БХВ-Петербург», 2016. – 318 с.: ил.
2. Что такое интернет вещей [Электронный ресурс] – Режим доступа: [http://www.tadviser.ru/index.php/Статья:Что_такое_интернет_вещей_\(Internet_of_Things,_IoT\)](http://www.tadviser.ru/index.php/Статья:Что_такое_интернет_вещей_(Internet_of_Things,_IoT))
3. Статья главного футуролога компании Cisco Дэйва Эванса(Dave Evans) [Электронный ресурс] – Режим доступа: <http://www.cisco.com/web/UA/about/news/2011/06262011.html>
4. Всеобъемлющий интернет. Электронный курс cisco [Электронный ресурс] – Режим доступа: <https://static-course-assets.s3.amazonaws.com/IoE11/RU/index.html#2.1.3.4>
5. Интернет технологии [Электронный ресурс] – Режим доступа: <http://smages.com/internet-tehnologii/umnyj-dom/>
6. Контроллер Fibaro Home Center Lite [Электронный ресурс] – Режим доступа: <https://secur.ua/umniy-dom/controllers/homecenterlite.html>
7. Термостат Secure SRT322 [Электронный ресурс] – Режим доступа: <https://secur.ua/umniy-dom/climat-control/securestr322.html>
8. Умный сенсорный выключатель Broadlink TC2-1 [Электронный ресурс] – Режим доступа: <https://secur.ua/umniy-dom/upravleniye-osvescheniem/upravleniye-osvescheniem-vyiklyuchateli/umnyj-sensornyj-vykljuchatel-broadlink-tc2-1.html>
9. Умная розетка Belkin WeMo Switch [Электронный ресурс] – Режим доступа: <https://secur.ua/umniy-dom/upravleniye-electropriborami/rozetki/umnaja-rozetka-belkin-wemo-switch-s-schjotchikom-potreblenija-jelektrichestva-f7c029ea.html>

10. Датчик движения Broadlink Movement [Электронный ресурс] – Режим доступа: <https://secur.ua/umniy-dom/datchiki/datchik-dvizhenija-broadlink-movement.html>
11. LAN-кабель Одескабель Lan-кабель КПВ-ВП (100) 2x2x0,50 (UTP-cat.5) [Электронный ресурс] – Режим доступа http://tdok.com.ua/ru/products/details/lan_kabel__kpv_vp__100__2h2h0_50__utp
12. Розетка RJ-45 внешняя однопортовая Cor-X UC-2108B [Электронный ресурс] – Режим доступа: <http://deps.ua/katalog/aksessuaryi-ethernet-patchkordyi-razemyi/wall-jack-rj-45-uc-2108b.html#description>
13. Короб пластиковый [Электронный ресурс] – Режим доступа: http://www.elektrika.ua/catalog/korob_kabelny/korob_plastikovy_s033016/
14. Коннектор Digitus RJ45 Cat.5 UTP 100 шт (A-MO8/8SF) [Электронный ресурс] – Режим доступа: http://rozetka.com.ua/digitus_a-mo8_8sf/p232758/
15. Домашний шлюз Zipabox-G1 [Электронный ресурс] – Режим доступа: <https://prom.ua/p539009722-zipabox-kontroller-umnogo;all.html>
16. Коммутатор Cisco Catalyst 2960 [Электронный ресурс] – Режим доступа: https://prom.ua/p259530281-kommutator-cisco-catalyst.html?_openstat=prom_prosale%3Bкоммутаторы%3BКоммутатор+Cisco+Catalyst+2960+Plus+24+10%2F100+%288%3Bsearch
17. Беспроводный одноуровневый термостат AURATON 200RTH [Электронный ресурс] – Режим доступа: http://prom.ua/p333679512-besprovodnyj-odnourovnevujj-termostat.html?_openstat=prom_prosale%3BТовары%3BБеспроводный+одноуровневый+термостат+AURATON%3Bsearch
18. Контроллер Arduino Leonardo [Электронный ресурс] – Режим доступа: https://arduino-ua.com/prod260-Arduino_Leonardo

19. Беспроводной датчик движения Ajax MotionProtect

[Электронный ресурс] – Режим доступа:

http://prom.ua/p509990625-besprovodnoj-datchik-dvizheniya.html?_openstat=prom_prosale%3Bтовары%3BБеспроводной+датчик+движения+Ajax+MotionProtect%3Bsearch

20. Инфракрасный датчик пламени LifeSOS SM-3S

[Электронный ресурс] – Режим доступа:

<http://prom.ua/p514859678-besprovodnoj-datchik-dyma.html>

21. Беспроводной датчик дыма Ajax WS-502

[Электронный ресурс] – Режим доступа:

http://prom.ua/p399363125-besprovodnoj-datchik-dyma.html?_openstat=prom_prosale%3Bтовары%3BБеспроводной+датчик+дыма+Ajax+WS-502%3Bsearch

22. IP камера VSTARCAM C7815WIP [Электронный ресурс] – Режим доступа:

<https://prom.ua/p85582042-kamera-vstarcam-c7815wip;all.html>

23. Цепной привод для окон Aprimatic O2M [Электронный ресурс] – Режим доступа: <https://nhn.prom.ua/p5510351-privod-tsepnoj-dlya.html>

24. Гресь Д.О., Возможности программы «cisco packet tracer» для моделирования сетей интернет вещей / Гресь Д.О., Волощук Л.А // Тезисы докладов 14 Всеукраинской конференции студентов и молодых учёных «Информатика, информационные системы и технологии». – Одесса. – 2017. – С.151-152.

ДОДАТОК А

Описание алгоритма и скрипта работы термостата

```

var autoCoolTemp = 20;
var autoHeatTemp = 19;
var METRIC;

var HEATER_PIN = 1;
var COOLER_PIN = 2;
var state = 0; // 0 off, 1 cooling, 2 heating, 3 auto
var STATE_NAMES = ["OFF", "COOL", "HEAT", "AUTO"];
var temperature = 0;
var unit;

function setup() {
  IoEClient.setup({
    type: "Thermostat",
    states: [{name: "Status", type: "options", options: {
      "0": "Off",
      "1": "Cooling",
      "2": "Heating",
      "3": "Auto"}},
    controllable: true},

    {
      name: "Temperature", type: "number", unit: "&deg;C",
      imperialUnit: "&deg;F",
      toImperialConversion: "x*1.8+32",
      toMetricConversion: "(x-32)/1.8",
      decimalDigits: 1},
    {
      name: "Auto Cool Temperature",
      type: "number",
      unit: "&deg;C",
      imperialUnit: "&deg;F", toImperialConversion: "x*1.8+32",
      toMetricConversion: "(x-32)/1.8",
      decimalDigits: 1,
      controllable: true,
      minValue: 10,
      maxValue: 50},
    {
      name: "Auto Heat Temperature",
      type: "number",
      unit: "&deg;C",
      imperialUnit: "&deg;F",
      toImperialConversion: "x*1.8+32",
      toMetricConversion: "(x-32)/1.8",
      decimalDigits: 1,

```

```

controllable: true,
minValue: -100,
maxValue: 20}}]);

IoEClient.onInputReceive = function(input) {
processData(input, true);
};
attachInterrupt(0, function() {
processData(customRead(0), false);
});
temperature = Environment.get("Ambient Temperature");
measurementSystemChangeEvent();
setState(state);
Environment.removeCumulativeContribution("Ambient Temperature");
updateEnvironment();
sendReport(); //відправити звіт
}
function loop() { //цикл
temperature = Environment.get("Ambient Temperature");
updateEnvironment();
measurementSystemChangeEvent();
sendReport();
delay(1000);
}
function measurementSystemChangeEvent() {
METRIC = isUsingMetric();
unit = METRIC ? "C" : "F";
setCustomText(27, 15, 200, 100,
STATE_NAMES[state] + " " +convertTemperature(temperature) + " "
+ unit);}

function mouseEvent(pressed, x, y, firstPress) {
if mouse pressed, then flip it if (firstPress) {
state++;
    if (state >= 4)
        state = 0;
setState(state);}
}
function processData(data, bIsRemote) {
if (data) {
data = data.split(",");
setState(parseInt(data[0]));
autoCoolTemp = parseFloat(data[2]);
autoHeatTemp = parseFloat(data[3]);}
}
function setState(newState) {
state = newState;
analogWrite(A1, state); // change graphic
setCustomText(27, 15, 200, 100,
STATE_NAMES[state] + " " +
convertTemperature(temperature) + " " + unit);
sendReport();
}

```

```

}
function sendReport() {
var report = state + "," + temperature + "," + autoCoolTemp + ","
+ autoHeatTemp;
customWrite(0, report);
IoEClient.reportStates(report);
setDeviceProperty(getName(), "state", state);
setDeviceProperty(getName(), "temperature", temperature);
setDeviceProperty(getName(), "auto cool temp", autoCoolTemp);
setDeviceProperty(getName(), "auto heat temp", autoHeatTemp);}
function updateEnvironment()
{
var autoCoolTemp = getDeviceProperty(getName(), "auto cool
temp");
var autoHeatTemp = getDeviceProperty(getName(), "auto heat
temp");
digitalWrite(HEATER_PIN, LOW);
digitalWrite(COOLER_PIN, LOW);
if ( state === 0 )
{
digitalWrite(HEATER_PIN, LOW);
digitalWrite(COOLER_PIN, LOW);}
else if ( state === 1 )
{
digitalWrite(COOLER_PIN, HIGH);}
else if ( state == 2 )
{
digitalWrite(HEATER_PIN, HIGH);}
else if ( state == 3 )
{
if (temperature >= autoCoolTemp){
digitalWrite(COOLER_PIN, HIGH);}
else if (temperature <= autoHeatTemp){
digitalWrite(HEATER_PIN, HIGH);}
else{
digitalWrite(HEATER_PIN, LOW);
digitalWrite(COOLER_PIN, LOW);}
}
}
function convertTemperature(value) {
if (METRIC)
return value.toFixed(0);
else
return (value * 1.8 + 32).toFixed(0);
}

```

ДОДАТОК Б

Опис алгоритму і скрипта роботи контролера SBC в підсистема освітлення

```
from gpio import *
from time import *

def handleSensorData():
    value = digitalRead(0)
    if value == 0:
        customWrite(1, '0') //сигнал з цифрового порту 1
        customWrite(2, '0') //сигнал з цифрового порту 2
    else:
        customWrite(1, '1') //сигнал з цифрового порту 1
        customWrite(2, '1') //сигнал з цифрового порту 2

def main():
    add_event_detect(0, handleSensorData)
    while True:
        delay(1000)
    if __name__ == "__main__":
        main()
```

ДОДАТОК В

Опис алгоритму і скрипта роботи контролера MCU в підсистемі
пожежогасіння

```
from gpio import *
from time import *

def handleSensorData():
    value = digitalRead(0)
    if value == 0:
        customWrite(1, '0') //сигнал з цифрового порту 1
        customWrite(2, '0') //сигнал з цифрового порту 2
        customWrite(3, '0') //сигнал з цифрового порту 3
        customWrite(4, '0') //сигнал з цифрового порту 4

    else:
        customWrite(1, '1') //сигнал з цифрового порту 1
        customWrite(2, '1') //сигнал з цифрового порту 2
        customWrite(3, '1') //сигнал з цифрового порту 3
        customWrite(4, '1') //сигнал з цифрового порту 4

def main():
    add_event_detect(0, handleSensorData)
    while True:
        delay(1000)

if __name__ == "__main__":
    main()
```

ДОДАТОК Г

Опис алгоритму і скрипта роботи контролера MCU в підсистемі
відеоспостереження

```
from gpio import *
from time import *

def handleSensorData():
    value = digitalRead(0)
    if value == 0:
        customWrite(1, '0')
    else:
        customWrite(1, '1')

    value = digitalRead(4)
    if value == 0:
        customWrite(2, '0')
    else:
        customWrite(2, '1')

    value = digitalRead(5)
    if value == 0:
        customWrite(3, '0')
    else:
        customWrite(3, '1')

def main():
    add_event_detect(0, handleSensorData)
    add_event_detect(4, handleSensorData)
    add_event_detect(5, handleSensorData)

    while True:
        delay(1000)

if __name__ == "__main__":
    main()
```

ДОДАТОК Д

Опис алгоритму і скрипта роботи контролера MCU в підсистемі
загазованості

```
var smk_sensor = A0;
var grgDoor = 0;
var Window = 1;

function setup() {
  pinMode(grgDoor, OUTPUT);
  pinMode(Window, OUTPUT);
}

function loop() {
  // read from sensor
  var newValue = analogRead(smk_sensor);
  newValue = newValue/10;

  // map it from 1023 to 255
  //newValue = Math.floor(map(newValue, 0, 1023, 0, 255));

  if (newValue > 10) {
    // analog write to pins
    customWrite(grgDoor, 1);
    customWrite(Window, 1);
    //digitalWrite(livgFan, HIGH);
  } else {
    if (newValue < 3) {
      customWrite(grgDoor, 0);
      customWrite(Window, 0);
    }
  }
  Serial.println("Smoke level: " + newValue);
}

function loop() {
  // read from sensor
  var newValue = analogRead(smk_sensor);
  newValue = newValue/10;
```