

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет \_\_\_\_\_ Інфокомунікацій \_\_\_\_\_  
(повна назва)  
Кафедра \_\_\_\_\_ Інформаційно-мережної інженерії \_\_\_\_\_  
(повна назва)

**АТЕСТАЦІЙНА РОБОТА**  
**Пояснювальна записка**

\_\_\_\_\_ другий (магістерський) \_\_\_\_\_  
(рівень вищої освіти)

\_\_\_\_\_ Дослідження засобів управління системами Smart House \_\_\_\_\_  
(тема)

Виконав: студент 2 курсу, групи ІМЗм-16-1  
спеціальності \_\_\_\_\_  
172 «Телекомунікації та радіотехніка»

\_\_\_\_\_ (код і повна назва спеціальності)  
спеціалізації \_\_\_\_\_  
«Інформаційні мережі зв'язку»

\_\_\_\_\_ (повна назва спеціалізації)  
Морковін Є.О.  
(прізвище, ініціали)

Керівник \_\_\_\_\_ доц. Костромицький А.І. \_\_\_\_\_  
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри

\_\_\_\_\_  
(підпис)

\_\_\_\_\_ Безрук В.М. \_\_\_\_\_  
(прізвище, ініціали)

2018 р.

.

Не містить відомостей, заборонених до відкритого публікування

Студент Морковін Є.О. \_\_\_\_\_

Керівник Костромицький А.І. \_\_\_\_\_

Харківський національний університет радіоелектроніки

Факультет \_\_\_\_\_ Інфокомунікацій \_\_\_\_\_  
Кафедра \_\_\_\_\_ Інформаційно-мережної інженерії \_\_\_\_\_  
Рівень вищої освіти \_\_\_\_\_ другий (магістерський) \_\_\_\_\_  
Спеціальність \_\_\_\_\_ 172 «Телекомунікації та радіотехніка» \_\_\_\_\_  
(код і повна назва)  
Спеціалізація \_\_\_\_\_ «Інформаційні мережі зв'язку» \_\_\_\_\_  
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_  
(підпис)

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

**ЗАВДАННЯ**  
НА АТЕСТАЦІЙНУ РОБОТУ

студентові \_\_\_\_\_ Морковіна Євгена Олександровича \_\_\_\_\_  
(прізвище, ім'я, по батькові)

1. Тема роботи \_\_\_\_\_ Дослідження засобів управління системами Smart House \_\_\_\_\_

затверджені наказом по університету від 06 квітня 2018 р. № 421 Ст \_\_\_\_\_

2. Термін подання студентом роботи до екзаменаційної комісії 11 червня 2018 р.

3. Вихідні дані до роботи В результаті порівняльного аналізу засобів управління Smart House, запропонувати свій варіант управління системою Smart House, вибір програмного забезпечення для свого варіанта, спроектувати та реалізувати його.

4. Перелік питань, що потрібно опрацювати в роботі \_\_\_\_\_

1 Огляд засобів управління системами Smart House

2 Опис обраної системи

3 Проектування та реалізація системи управління Smart House з використанням віртуальних машин

4 Аналіз спроектованої системи

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) \_\_\_\_\_

Слайди у форматі Power Point (назва та мета роботи, рисунки систем управління, висновки тощо) \_\_\_\_\_

---

---

---

---

---

---

---

---

---

---

### КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни Виконання етапів роботи	Примітка
1	Ознайомлення із завданням. Уточнення ТЗ.	09.04 – 10.04.18	
2	Підбір літератури за темою роботи.	11.04 – 13.04.18	
3	Виконання розділу 1	14.04 – 25.04.18	
4	Виконання розділу 2	25.04 – 05.05.18	
5	Виконання розділу 3	06.05 – 15.05.18	
6	Виконання розділу 4	16.05 – 06.06.18	
10	Оформлення презентаційного матеріалу, підготовка до захисту у ДЕК	07.06 – 11.06.18	

Дата видачі завдання \_\_\_\_\_ 20\_\_ р.

Студент \_\_\_\_\_  
( підпис )

Керівник роботи \_\_\_\_\_  
( підпис )

Костромицький А.І.  
(прізвище та ініціали)

## РЕФЕРАТ

Пояснювальна записка: 82 с., 28 рис., 5 табл., 8 джерел, 2 додатки

Мета роботи – дослідження засобів управління smart house та розробка централізованої системи управління smart house з використанням гіпервізора.

На основі аналізу проблем існуючих smart house систем було запропоновано рішення з використанням гіпервізора, а також були розглянуті варіанти скорочення витрат при реалізації даного рішення. Було проведено експериментальне дослідження можливості розгортання гіпервізора на персональному комп'ютері різних конфігурацій. Розглянуто програмне забезпечення для створення віртуального середовища розумного будинку. В результаті була створена модель, яка вирішує всі поставлені завдання і дозволяє розширити функціональні можливості системи управління будинку.

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ГІПЕРВІЗОР, ВІРТУАЛЬНА МАШИНА,  
РОЗУМНИЙ БУДИНОК, ПЕРСОНАЛЬНИЙ КОМП'ЮТЕР, SMART HOUSE.

## THE ABSTRACT

Explanatory note: 82p., 28 fig., 5 tabl., 8 sources, 2 app.

The purpose of the work is to research the smart house management tools and develop a smart house management system using the hypervisor.

On the basis of the analysis of the problems of existing smart house systems, solutions were proposed using the hypervisor, as well as options for reducing the cost of implementing this solution were considered. An experimental study was made of the possibility of deploying a hypervisor on PCs of various configurations. Software is considered to create a virtual environment of a smart home. As a result, a model was created that solves all tasks and allows you to expand the functionality of the home management system.

SOFTWARE,    HYPERVIRUS,    VIRTUAL    MACHINE,    CONSUMER    HOUSE,  
PERSONAL                      COMPUTER,                      SMART                      HOUSE.

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ .....	8
ВСТУП.....	9
1 ОГЛЯД ЗАСОБІВ УПРАВЛІННЯ СИСТЕМАМИ SMARTHOUSE .....	10
1.1 Аналіз централізованих систем управління Smart House .....	10
1.2 Аналіз децентралізованих систем управління Smart House .....	17
1.3 Аналіз змішаних систем управління Smart House .....	24
2 ОПИС ОБРАНОЇ СИСТЕМИ.....	26
2.1 Віртуальна машина .....	26
2.2 Системні віртуальні машини.....	29
2.3 Прикладні віртуальні машини .....	30
2.4 Аналіз існуючих гіпервізорів .....	31
3 ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ СИСТЕМИ УПРАВЛІННЯ SMART HOUSE З ВИКОРИСТАННЯМ ВІРТУАЛЬНИХ МАШИН.....	37
3.1 Базові технічні вимоги до проектування .....	37
3.2 Визначення структури моделі.....	37
3.3 Вибір гіпервізора методом аналізу ієрархії .....	40
3.4 Установка і налаштування Citrix XenServer .....	43
4 АНАЛІЗ СПРОЕКТОВАНОЇ СИСТЕМИ.....	51
ВИСНОВКИ.....	52
ПЕРЕЛІК ПОСИЛАНЬ .....	53
ДОДАТОК А СЛАЙДИ ПРЕЗЕНТАЦІЇ .....	54
ДОДАТОК Б ПУБЛІКАЦІЇ .....	61

## ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ПЗ – програмне забезпечення

ВМ – Віртуальна машина

KNX – комунікаційна шина асоціації KONNEX

ПК – персональний комп'ютер

ОС – операційна система

СУБД – система управління базами даних

ЛТ – just-in-time

AJAX – асинхронний JavaScript і ХМ

API – програмний інтерфейс програми

JSON – текстовий формат обміну даними

HTTP – протокол передачі гіпертексту

ОЗУ – Оперативна пам'ять

VMFS – файлова система віртуальної машини

Virtual SMP – віртуальне симетричне мультипроцесування

Web Access – веб-інтерфейс



## ВСТУП

Зараз стрімко розвивається і впроваджується технологія smart house, яка являє собою комплекс інженерних систем житлового будинку, створений для вирішення завдань життєзабезпечення, підвищення домашнього комфорту, безпеки та ресурсозбереження. Такі системи встановлюються по всьому будинку, і кожна з них виконує певне завдання, що призводить до необхідності централізованого управління комплексом.

Існує кілька підходів до вирішення даного завдання. Одним з підходів є виділення кожній системі окремого сервера, який організовує її роботу. Перевагою даного підходу є те, що при виході з ладу однієї з систем інші продовжать функціонування. Недоліками є витрати необхідні на встановлення окремих серверів, децентралізація всього комплексу і необхідність налаштування кожної системи. Інший підхід передбачає встановлення декількох серверних комплексів, які забезпечують спільне управління і роботу всіх систем. Як правило, такі сервери не працюють в режимі максимального завантаження, велика частина ресурсів не використовується і, отже, кошти витрачені на закупівлю серверного обладнання, не завжди виправдані.

Таким чином, метою роботи є вибір варіанта організації серверної частини комплексу інженерних систем "розумного будинку", а також дослідження можливості установки компонент комплексу на персональні комп'ютери користувачів, створення серверного комплексу для управління всіма системами в будинку.

## 1 ОГЛЯД ЗАСОБІВ УПРАВЛІННЯ СИСТЕМАМИ SMART HOUSE

Термін "розумний будинок" не має чіткого визначення, а тому під це поняття підпадає будь-яка система з автоматизованим керуванням приладами, яка спрощує життя людини та підвищує рівень її комфорту. Через нечіткі рамки виникло багато реалізацій з різним рівнем інтеграції та принципом роботи. Їх можна умовно поділити на три групи:

- вбудовані системи з центральним контролером;
- вбудовані системи без центрального контролера;
- системи з інтеграцією, що налаштовується.

Перша група представляє з себе повністю налаштовану і встановлену виробником систему, яка керується центральним обчислювальним пристроєм і не передбачає прямої взаємодії своїх компонентів між собою. Всі призначені для користувача налаштування зберігаються на центральному пристрої (сервері), а периферія лише виконує отримані від нього інструкції і зазвичай не має вбудованої пам'яті і обчислювальних потужностей. Принцип роботи системи зображений на рис. 1.1.



Рисунок 1.1 – Діаграма з центральним контролером

Друга група є системою з напіваавтономними пристроями. Алгоритми взаємодії прописуються з програми контролера безпосередньо в пам'ять кожного пристрою і для їх зміни пристрій буде необхідно перепрограмувати. У зв'язку з відсутністю центрального компонента, зв'язки між приладами відбуваються безпосередньо і є можливість створення автономних груп, замкнутих один на одного. Принцип роботи на рис. 1.2



Рисунок 1.2 – Діаграма, без центрального контролера

Третя група - це зовнішні контролери, які приєднуються до звичайних приладів і залежно від показання своїх сенсорів і вбудованого алгоритму регулюють їх роботу. Можуть мати центральний контролер, але часто керуються і налаштовуються з Інтернет або хмарного сервісу. Функціонують здебільшого як незалежні модулі і для налаштування прямого зв'язку можуть знадобитися додаткові датчики / сенсори. Принцип роботи на рис. 1.3.



Рисунок 1.3 – Діаграма що налаштовується

### 1.1 Аналіз централізованих систем управління Smart House

Централізовані системи будуються на основі центрального контролера і виконавчих блоків, іноді сюди додаються різні панелі управління. Подібні системи отримали широку популярність в приватному секторі. Центром системи є контролер, до якого підключаються всі пристрої. Компоненти системи можуть мати мікроконтролери, але управління здійснюється центральним контролером, на якому знаходиться програма, що управляє взаємодією всієї системи. До таких систем відносяться Crestron, Lutron і інші.

Для проведення аналізу були обрані 2 найбільш поширені системи - Crestron, Lutron.

AMX і Crestron - американські централізовані системи. Всі функції обробки інформації зосереджені в одному блоці - потужному центральному комп'ютері, що працює на своїй власній операційній системі. Цей комп'ютер приймає сигнали з усіх датчиків і вимикачів і пересилає їх далі - на прилади керування. Саме такими системами обладнані будинки багатьох голлівудських зірок і Білла Гейтса. По суті вони є різновидами локальної комп'ютерної мережі - провідний або бездротової, вирішувати замовнику. Централізовані системи надають власнику воістину багаті можливості, і, природно, вони більш надійні, ніж X10. Проте і з ними часом трапляються прикрі неприємності: вихід з ладу центрального комп'ютера - а працювати йому доводиться цілодобово, сім днів на тиждень - в даному випадку означатиме блокування всіх механізмів. До того ж вартість такої системи нерідко прирівнюється до середньостатистичного

гонорару голлівудської зірки. У кожної з перерахованих технологій, є як переваги, так і недоліки. Суть недоліків, підчас, зрозуміла лише фахівцям у цій галузі, але основна проблема очевидна. Це несумісність технологій, приладів, техніки, а також - у ряді випадків - погані можливості з розширення та модернізації системи. З питаннями сумісності люди стикаються щодня. Ми думаємо про те, чи сумісна материнська плата і відеокарта при складанні ПК, чи сумісні телевізор і DVD - програвач при покупці ДК, чи сумісний цей диск з цим приводом оптичних дисків. Прикладів багато. Так ось, справа в тому, що цієї долі не уникли і виробники технологій для «розумного дому». «Розумний дім» може управляти практично всіма електронними пристроями і функціями квартири. Все залежить лише від твоєї фантазії і спроможності. Дистанційне включення кавоварки, автоматичний набір води у ванну, випуск собаки на прогулянку в певні години, приглушення звуку музичної системи при надходженні вхідного дзвінка на телефон - все це цілком реально.

Компанія Crestron - одна з провідних фірм по створенню централізованих автоматизованих систем управління будинком. Рішення, що надається даною фірмою, засноване на застосування широкого спектра керуючих центральних контролерів і безлічі виконавчо-командних блоків. Контролери даної компанії високопродуктивні, а також мають широкий набір вбудованих можливостей. Найбільш популярними контролерами є RMC3 і CP2E. Контролери Crestron діляться на 2 і 3 серії. Найпотужнішими пристроями з другої серії були PRO2 і AV2, завдяки своїй можливості встановлення додаткової карти входу. У CP2E продуктивність приблизно дорівнює продуктивності цих двох контролерів, але в ньому встановлено меншу кількість портів і відсутня можливість установки карт розширення. Зараз випускають контролери 3 серії, які мають більшу продуктивність у порівнянні з 2 серією. Напрямок розробки, що проводиться Crestron, будується на бажанні зібрати в єдиний комплекс всі розрізнені системи будинку і інтеграція управління цих систем за допомогою єдиного контролера. Crestron має обмежений але досить широкий, набір панелей і вимикачів для управління, таких як сенсорні і клавішні панелі[1].

За допомогою систем Crestron координується робота конференцзалів з штаб-квартири Microsoft в Редмонді, штат Вашингтон. Аналогічні системи управління встановлені в семи з восьми американських університетів, що входять до престижної асоціації «Ліги плюща»; в пентхаусі Trump World Tower в Нью-Йорку і в залі оперативних нарад в Пентагоні.

Основним перевагами таких систем є можливість об'єднання управління всіма системами в єдиний центр.

До недоліків можна віднести дуже високу ціну пристроїв, необхідність вільного простору або технологічного приміщення для розміщення компонентів систем. Головним недоліком є те, що при аварійному стані або поломки контролера, вся система виходить з ладу і не може функціонувати. Але цього можна уникнути, встановивши кілька контролерів, що призводить до подорожчання системи.

Стандарт LonWorks підтримується асоціацією EIBG (European Intelligent Building Group), в яку входять компанії TAC, Andover Controls, Honeywell, Johnson Controls і багато інших. Перевагою стандарту є можливість побудови мережі за принципом вільної топології, головною її особливістю є те, що доступ до управління подібної мережею можна отримати майже з будь-якої точки такої системи. Так от, за допомогою архітектури LonWorks можна здійснювати контроль і регулювання таких функцій, як опалення, кондиціонування, вентиляція, зволоження повітря, підігрів підлоги, охоронна і пожежна сигналізація, управління світлом. Суть технології в тому, що всі інженерні системи об'єкта обмінюються даними про свій стан, виконаних діях або отриманих командах за допомогою єдиної мережі за спеціально розробленим в рамках LonWorks протоколу LonTalk. В якості фізичного інтерфейсу передачі даних часто використовується інтерфейс, подібний напівдуплексному багатоточковому послідовному RS 485.

На сьогоднішній день інтерфейс RS- 485 є одним з найбільш поширених стандартів передачі даних на фізичному рівні, тобто, на самому нижньому (першому) рівні в рамках моделі взаємодії відкритих систем OSI. Дуже часто цей стандарт використовується в разі потреби пов'язати між собою кілька різних пристроїв, виконаних на базі програмованих логічних контролерів (ПЛК). Нарівні з RS- 232, інтерфейс RS- 485 нерідко застосовується і в комп'ютерній індустрії. З допомогою RS- 485 можна побудувати мережу за участю 32 пар передавач/приймач, але вже сьогодні в специфікації стандарту введені зміни, що розширюють можливості цього RS- протоколу до 255 пристроїв, об'єднаних в одну мережу. Якщо ж є необхідність в об'єднанні ще більшої кількості пристроїв, то можна використовувати так звані повторювачі. У такому випадку, мережу на базі RS- 485 можна розширювати майже до нескінченності. А ось уже системи на основі LonWorks без особливих проблем

можна пов'язати з мережею Internet, і за допомогою стандартних засобів зв'язку здійснювати віддалений контроль і управління будь-якої інженерної підсистемою. Слід відзначити, що системи на базі LonWorks частіше застосовуються у великих будинках і приміщеннях.

Стандарт EIB (European Installation Bus - Європейська інсталяційна шина) розроблений асоціацією EIBA для автоматизації житла і малих об'єктів. Основними виробниками лінійки пристроїв EIB виступають компанії ABB, Berker, Gira, Jung, Merten, Siemens. Один двожильний кабель - шина EIB - об'єднує всі електричні пристрої будівлі, які обмінюються сигналами - «телеграмами» один з одним, тому система відрізняється високою гнучкістю і надійністю.

У EIB досить широкі можливості з розширення та перепрограмуванню окремих елементів вже функціонуючого на базі EIB «розумного будинку». До плюсів даного стандарту можна віднести:

- оптимально побудована для автономного, надійного управління освітленням, приводами механізмів, клімат контролю;
- система управління "Розумний дім", виконана на EIB, повністю автономна і незалежна від працездатності комп'ютера візуалізації, вона може працювати і зберігати в пам'яті логічних модулів всі режими незалежно від нього;
- електроустановка будівлі може бути виконана набагато простіше, а пізніше без проблем розширення і модифіковано;
- система безпечна, тому що до панелей управління походять тільки сигнали управління низької напруги, а всі силові мережі зведені в щиток;
- при зміні призначення або перерозподілі приміщень швидко і легко узгодження системи Instabus EIB здійснюється простим перерозподілом (змінюю параметрів) абонентів шини без необхідності прокладки нових електропроводок;
- EIB - стандарт де-факто в автоматизації кліматичних систем і електроприладів. Вибір зовнішнього вигляду універсальних вимикачів для EIB величезний і представлений різними виробниками;
- відмінна опрацювання програмно - апаратного забезпечення компонентів, величезні можливості розширення. А до основних

недоліків даної технології слід віднести дорожнечу при побудові систем автоматизації великих об'єктів.

Компанія Lutron (США) спеціалізується на випуску системи домашньої автоматизації, яка включає в себе управління освітленням, кліматом і управління шторами. На відміну від багатьох інших виробників, які прагнуть охопити весь спектр можливостей інтеграції інженерних і розважальних систем, Lutron вирішили зосередитися виключно на управлінні освітленням, кліматом і шторами. Lutron знайшов широке застосування в комерційних об'єктах - холах готелів, ресторанах, бізнес-центрах. На особливу увагу заслуговують штори Lutron, які забезпечені інноваційним, практично безшумним механізмом, можливо самим безшумним в світі.

Lutron - це в першу чергу, одна з найнадійніших систем управління освітленням. Американська компанія яка винайшла напівпровідниковий диммер, перша забезпечила підтримку AppleHomeKit і вже близько шістдесяти років втілює в життя ідею простого управління будинком для досягнення максимального комфорту, затишку і енергозбереження. Головна перевага "розумного освітлення" від Lutron полягає в тому, що при установці діє правило "поставив і забув", так як особливого обслуговування в процесі експлуатації ця система не вимагає.

Lutron - це ще й самі безшумні приводи штор з точним позиціонуванням і автоматичною системою підтримки постійного рівня освітленості в комбінації природного і штучного освітлення.

Освітленням, досить просто керувати за допомогою сценаріїв, які створюються для повсякденного використання і тим самим спрощують життя господарів. Для управління, не потрібно багато вимикачів, як це буває в класичному випадку, причому один і той же світильник можна включати і вимикати в різних місцях Вашого будинку, без монтажу складних електричних схем [2].

Lutron випускає велику кількість кейпадов в будь-якому кольорі і під будь-який дизайн інтер'єру. А також, існує можливість замовити індивідуальну гравіювання на клавішах вимикачів, з назвами сцен або окремих світильників в приміщенні для зручності управління.

Можливості системи:

- управління освітленням з різних місць;
- управління приводами штор і жалюзі;



- регулювання яскравості;
- вибір кольору LED підсвічування;
- включення сценаріїв за розкладом;
- інтеграція з аудіосистемами;
- управління з iPhone або iPad;
- голосове управління за допомогою Siri.

## 1.2 Аналіз децентралізованих систем управління smart house

У децентралізованих системах, на відміну від централізованих, відсутній центральний контролер. Всі пристрої з'єднані безпосередньо один з одним за допомогою шини. Кожен компонент системи має вбудований контролер, який діє самостійно завдяки закладеній в нього програмі. Кожен такий контролер підключений до шини, що управляє системою, і програмується незалежно від інших. Серед таких систем виділяються C-Bus і KNX.

Система управління KNX - це європейський стандарт, який представляє собою результат об'єднання трьох попередніх стандартів: EHS, EIB і BatiBUS. Дана система децентралізована. Пристрої в цій системі діляться на актуатори (виконавчі пристрої), сенсори (датчики) і системні пристрої. Сенсори збирають інформацію зі всієї системи і відправляють цю інформацію по шині у вигляді телеграм. Актуатори приймають телеграми, відправлені виконавчим пристроєм, і виконують відповідні дії. До системних пристроїв відносяться лінії живлення, USB, з'єднувальні пристрої. Оскільки KNX є децентралізованою системою, то всі пристрої, зв'язуються один з одним безпосередньо, без будь-якого мережного контролюючого пристрою. Обмін інформацією відбувається через загальний канал (через шину KNX).

KNX системи мають ряд переваг, серед яких: повна автономність і незалежність від комп'ютера або контролера, що підвищує надійність; гнучкість є можливість швидкого перепрограмування системи; компоненти системи монтуються в стандартному електрощиті, тобто система не вимагає додаткового місця в приміщенні. Одним з головних переваг є відкритий протокол, що дозволяє різним компаніям випускати обладнання сумісне з цією системою.

До недоліків таких систем можна віднести:

- невисока швидкість передачі даних;

- обмеженість адресного простору, що призводить до неможливості встановлення системи в дуже великих будівлях;
- низька перешкодозахищеність;
- висока ціна.



Рисунок 1.4 – Застосування стандарту KNX

Стандарт KNX набув широкої популярності в побудові інженерних систем для будинків та невеликих приміщень зокрема. Його перевага у взаємодії пристроїв, що працюють за цим стандартом та містяться в одній мережі. Таким чином, для побудови системи управління мікрокліматом в приміщенні достатньо мати відповідне обладнання, що реалізовує ці функції(рис. 1.4).

Підключені до шини (мережі) абоненти (пристрої) можуть обмінюватися інформацією через загальний канал передачі, шину. При цьому підлягає передачі інформація упаковується в телеграму і передається по кабелю від датчика (сенсора - відправника команд) до одного або кількох виконавчим механізмів. При успішній передачі та прийомі кожен пристрій-приймач підтверджує отримання телеграми. При відсутності підтвердження передача повторюється ще два рази. Якщо і після цього підтвердження відсутнє, то процес передачі закінчується. Саме тому протокол EIB не є "промисловим", тобто його не можна застосовувати в додатках, пов'язаних з небезпекою для людей.

Передача проводиться модулюванням напруги в мережі, причому логічний нуль пересилається у вигляді імпульсу, з амплітудою приблизно  $\pm 6$  В. Відсутність імпульсу інтерпретується як логічна одиниця. Телеграми пересилаються пакетами в яких 8 інформаційних байт. Пересилання синхронізується старт- і стоп-бітами, є біт контролю на парність.

Для вирішення колізій (зіткнень) телеграм в мережі TP1 застосовується метод CSMA / CA (Carrier Sensible Multiple Access with Collision Avoidance). Цей метод гарантує випадковий, безпроблемний доступ пристроїв до шини, при цьому без істотного зниження її максимальної пропускної здатності. При цьому гарантується, що спочатку будуть передані повідомлення з найвищим пріоритетом.

Для того, щоб система запрацювала необхідно не тільки встановити пристрої і з'єднати їх необхідними кабелями між собою і з силовою мережею, але і запрограмувати пристрою за допомогою інженерного ПО ETS. До завантаження необхідно провести наступні операції: призначити пристроям індивідуальні фізичні адреси, вибрати і налаштувати (параметризованих) прикладні програми пристроїв, створити структуру групових адрес і об'єднати на них об'єкти зв'язку, взявши один об'єкт в датчику та іншої у виконавчому пристрої. В межах однієї мережі кожен пристрій має мати фізичний (індивідуальний) адресу. Призначення адрес проводиться за допомогою ETS. Перед призначенням пристрою адреси воно переводиться в режим програмування, як правило, шляхом натискання на спеціальну кнопку на лицьовій частині корпусу, при цьому для підтвердження спалахує світлодіод (червоний). Групові адреси можуть бути призначені активним пристроям системи незалежно від їх розташування і значень фізичних адрес. Виконавчим пристроям (одержувачам телеграм) може бути призначено кілька групових адрес, але сенсори (датчики) можуть відправляти телеграму тільки за однією адресою. У складних системах, як правило, використовують трирівневу систему групової адресації (головна група / середня група / підгрупа). Об'єкти зв'язку, між якими встановлюється комутація, можуть мати розмір від 1 біта до 14 байт в залежності від функції, виконуваної цим об'єктом. У пристроїв кількість об'єктів зв'язку може бути різним.

До складу обладнання KNX входили наступні типи пристроїв:

- сенсори (датчики) - сенсорні настінні панелі та вимикачі; датчики фізичних величин - температури, вологості і т. д. ; датчики руху, таймери та інші. Вони відповідають за фіксування (реєстрацію) тих чи інших зовнішніх подій, настання яких повинно викликати певну відповідну реакцію системи. Після настання такої події (натискання кнопки, перевищення температурою порогового значення і т. П.)

Сенсор посилає по мережі керуючу команду відповідному виконавчому пристрою;

- виконавчі пристрої (актуатори, перетворювачі, суміжні модульні системи);
- світлові регулятори (диммери), релейні модулі; модулі управління жалюзі, та інші. Вони змінюють свій стан (включено-виключено, відкрито-закрито і т.п.) Відповідно до команд, які надходять від сенсорів, тим самим керуючи різним електрообладнанням;
- системні пристрої - блоки живлення, інтерфейсні модулі, шинні з'єднувачі, повторювачі і інші, включаючи панелі і логічні модулі. Системні пристрої забезпечують працездатність і можливість налаштування мережі KNX.

Прилади системи KNX з радіоканалом як середовищем передачі даних не відносяться до якої-небудь ієрархічної структури. Вони можуть встановлюватися в будь-якому місці, і будь-який сенсор може сполучатися з будь-яким виконавчим пристроєм при врахуванні радіусу дії радіоканалу.

Радіус дії радіоканалу в якості середовища передачі даних в просторовому відношенні неможливо точно встановити: KNX-радіотелеграми можуть бути отримані також іншими приладами сусідній KNX-радіосистеми.

У радіосистемах KNX в якості методу модуляції застосовується частотна модуляція (Frequency Shift Keying, FSK). При цьому логічні нуль і одиниця з незначними відхиленнями генеруються несучою частотою, також званої середньої частотою.

У радіосистемах KNX в якості середньої частоти застосовується частота в 868,30 МГц. Швидкість передачі даних складає 16 384 біт / сек і модулюється по методу манчестерського кодування. При такому кодуванні зміна фронту імпульсу з "0" на "1" і навпаки відбувається в середині кожного інформаційного біта. Завдяки такому методу кодування передавальне і приймає пристрої можуть бути легко синхронізовані, тому перехід 0/1 або 1/0 в середині кожного переданого біта дозволяє постійне узгодження тактових імпульсів.

Частота зв'язку в радіосистемі KNX лежить у смузі частот ISM (Industrial- Scientific-Medical). Діапазон частот для різних цілей застосування в рамках цієї смугистрого визначений. Максимальна потужність сигналу становить приблизно 12 мВт. Час передачі радіосигналу кожного приладу, зване також робочим циклом (Duty Cycle), становить приблизно 1%

(максимальна тривалість посилки сигналу - 0,6 сек. за хвилину). Строго регламентована тривалість посилки сигналу дозволяє уникнути окремих тривалих пересилань сигналу, а, отже, і безперервних тривалих сигналів перешкод, які блокують радіоканал. Таким чином, можна виходити з того, що відіслані повідомлення приймаються і обробляються тільки тими приймаючими пристроями, для яких вони призначені.

Перший інформаційний блок складається з контрольного поля (4 байти), серійного номера KNX пристрої (6 байтів) і поля захисту інформації (2 байти). У контрольному полі зазначаються дані про довжину телеграми, як передачі (приймальні здібності), а також про стан батарей радіокомпонентів. Серійний номер KNX служить однозначного розпізнавання приладу. Він програмується в пристрій при його виготовленні і надалі не змінюється. Серійний номер передається в кожній телеграмі і при введенні в експлуатацію або при зв'язку пристроїв по радіоканалу прописуються в приймальному пристрої як вихідний адреса передавача.

Таким чином, серійний номер KNX служить не тільки адресації шинних пристроїв, але й розмежуванню пристроїв сусідніх KNX-радіосистем. Поле захисту інформації повідомляє приймаючому пристрою про наявність або відсутність помилок при передачі телеграми.

У другому інформаційному блоці поряд зі старт-і стоп-бітами знаходяться також індивідуальний вихідний адреса, адреса призначення та повідомлення. Індивідуальний адреса джерела сигналу є фізичною адресою пристрою. Він використовується тільки вищими в ієрархії контролерами або з'єднувачами при програмуванні приладу і автоматично повідомляється ними при введенні пристрою в експлуатацію. Адреса призначення відрізняється за своєю функцією в залежності від типу доступу до пристроїв, отримують повідомлення. При фізичному доступі, тобто при програмуванні, адреса призначення є індивідуальним вихідною адресою пристрою. У нормальному режимі (напр., при передачі команди перемикання) адреса призначення містить номер запитуваної комунікаційного об'єкта в пристрої. Повідомлення містить такі дані, як, наприклад, команди, оповіщення, дані настройки, виміряні значення і т.д. Залежно від довжини повідомлення в одній KNX-радіотелеграм можуть передаватися та інші інформаційні блоки.

Режими конфігурації пристроїв KNX:

Режим A-mode KNX(Автоматична конфігурація) є найпростішим методом конфігурації. Він застосовується тільки там, де функції приладів мають однозначну прив'язку один до одного, і залишається лише встановити для користувача настройки. Базова конфігурація здійснюється автоматично за умови, що пристрій підключено до середовища передачі даних. Даний вид конфігурації призначений, насамперед, для самостійного введення в експлуатацію великих побутових електроприладів та радіоелектронної апаратури.

Конфігурація в режимі E-mode здійснюється через центральний контролер, кодове колесо або через кнопки, розташовані безпосередньо на пристроях KNX. Якщо контролер виконує додаткові функції (наприклад, світлові сцени або логічні операції), то він залишається в системі, в іншому випадку після введення в експлуатацію його можна видалити. Гарантована сумісність з S-mode.

Пристрої, сумісні з режимом E-mode, зазвичай мають обмежений набір функцій і призначені для малих або середньовеликих систем. ПК з інстальованим на ньому ETS не потрібно. Для введення в експлуатацію та конфігурації

необхідний запас базових знань, який можна отримати, ознайомившись з документацією або пройшовши короткий курс навчання.

Проектування та введення в експлуатацію системи KNX здійснюється з одного ПК, на якому встановлено ПЗ ETS. Даний вид конфігурації підходить для самого широкого спектра систем: від малих і простих до встановлених у великих комплексах з великим набором функцій.

На відміну від A-mode і E-mode для застосування режиму S-mode рекомендується пройти курс навчання. В S-mode може використовуватися весь спектр можливостей системи KNX.

S-mode KNX підходить для таких середовищ передачі даних, як вита пара і силовалінія PL.

Система управління будинком і будівлею C-Bus пропонує гнучкий, надійний і потужний спосіб управління практично будь-якими електричними навантаженнями.

C-Bus надає високий рівень гнучкості при інсталяції електрообладнання: економія електроенергії, зменшення часу інсталяції, проста реалізація складних схем підключення, зменшення вартості обслуговування.

C-Bus-система електропроводки, керована мікропроцесором, для управління освітленням та іншими електричними параметрами. C-Bus вільно управляє практично будь-яким типом електричного навантаження незалежно від того, чи забезпечується просте управління вкл. / викл. або змінне. Керуюча інформація завантажується і міститься в блоках C-Bus, а не зберігається цілком в центральному процесорі. Якщо комп'ютер не потрібен для нормального функціонування C-Bus, є кілька програмних засобів для забезпечення додаткової гнучкості роботи менеджерів, яким потрібно цей тип контролю [6].

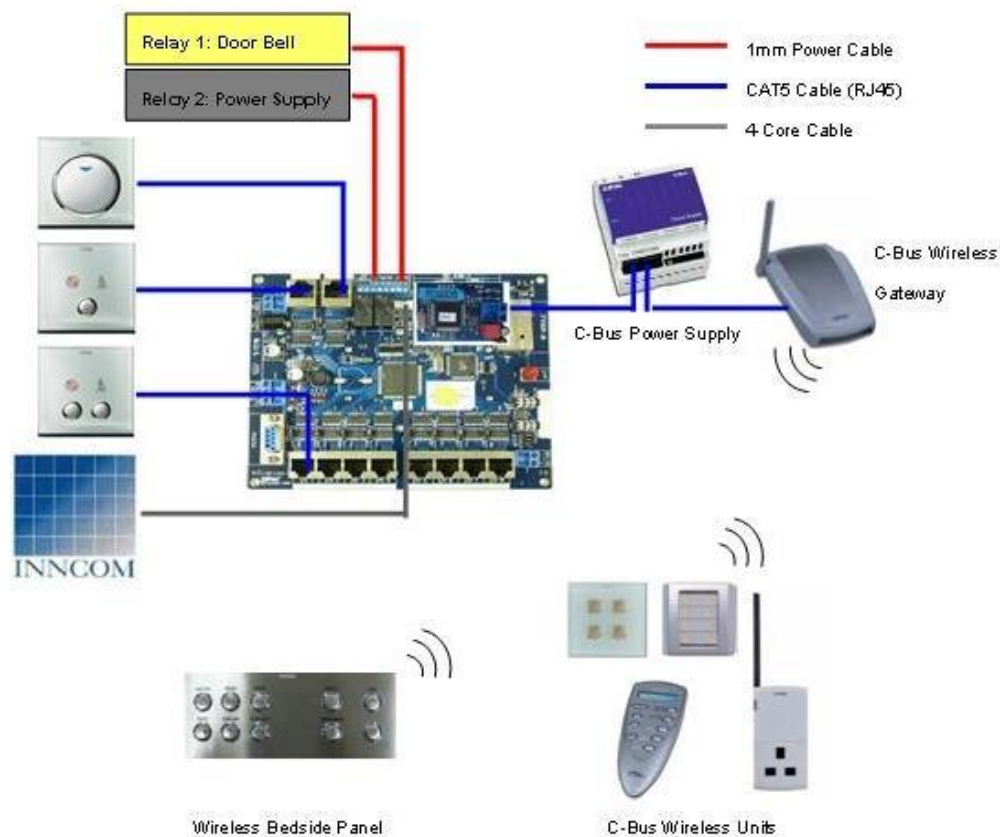


Рисунок 1.5 - Управління системою C-Bus

C-Bus - це мікропроцесорна система управління різними електричними навантаженнями - як світловими, так і інших типів. Керуюча інформація завантажується і міститься в кожному з пристроїв C-Bus, а не в центральному керуючому пристрої:

- інтегрує світло, пристрої безпеки, контроль доступу і керування будинком в одну систему;
- один кабель C-Bus може керувати необмеженою кількістю пристроїв;

- функції управління можуть бути додані, відключені, змінені або перепрограмовані в будь-який момент без перепідключення;
- C-Bus проста в установці і використанні.

### 1.3 Аналіз змішаних систем управління smart house

Змішані системи являють собою поєднання різних принципів і елементів централізованої і децентралізованої систем. Найбільшу популярність серед систем такого типу отримала X-10.

X10 - міжнародний відкритий промисловий стандарт, застосовуваний для зв'язку електронних пристроїв в системах домашньої автоматизації. Стандарт X10 визначає методи і протокол передачі сигналів управління електронними модулями, до яких підключені побутові прилади, з використанням звичайної електропроводки або бездротових каналів.

До недоліків відносяться: протокол X10 дуже повільний, пристрої захисного відключення можуть послаблювати сигнал настільки, що він не буде прочитана, деякі модулі X10 некоректно працюють (або не працюють взагалі), якщо керують пристроєм з низькою споживаної енергією (менше 50 Ватт), наприклад, флуоресцентними лампами[3].

При виконанні проекту X10 рекомендується наступна технологічна послідовність робіт:

- визначення об'єктів управління;
- підбір виконавчих пристроїв X10 по функціональності, здатності навантаження і вимогам до розміщення;
- визначення схеми управління і автоматизації;
- вибір контролерів і призначених для користувача засобів управління;
- призначення адрес пристроїв X10;
- визначення необхідного системного обладнання;
- підготовка ескізного (робочого) проекту;
- кабельні роботи (в необхідних випадках);
- монтаж, підключення та налаштування обладнання X10;
- підготовка виконавчої документації;
- введення системи в експлуатацію та здача робіт.

Використання системи автоматизації будинку X10 має багато переваг:



- натисненням однієї кнопки управляти освітленням в будь-якій частині будинку;
- встановлювати різну освітленість в будинку, наприклад, коли приймаєте гостей, зменшуючи або збільшуючи яскравість ламп;
- забезпечити безпеку і охорону будинку всередині і зовні за допомогою камер спостереження, а також детекторів руху і таймерів, які при спрацьовуванні будуть включати світло;
- вмикати і вимикати світло і електроприлади в випадковому порядку, коли ви знаходитесь далеко від дому, для імітації присутності в будинку мешканців;
- дистанційно керувати освітленням, побутовими електроприладами та іншими електричними пристроями за допомогою телефону, домашньої комп'ютерної мережі або широкосмугового з'єднання з Інтернетом.

Рухаючись по мережі пакет складається з адреси і команди, що відправляються контролером керованого модулю. Більш складні контролери також вміють опитувати такі ж керовані модулі про їх статус. Цей статус може бути досить простим ( «включено» або «вимкнено»), вказувати числове значення (поточне значення яскравості, температура або дані з інших датчиків).

Незалежно від середовища передачі (електрична мережа або радіосигнал), пакети X10 складаються з:

- 4 біта - код будинку;
- 4 біта - код модуля (може бути задано кілька модулів);
- 4 біта - команда.

Щоб уникнути плутанини і зручності користувачів код будинку задається латинськими буквами від А до Р, а код модуля - цифрами від 1 до 16.

Коли мережа X10 встановлена, кожен модуль налаштовується таким чином, щоб відгукуватися на один з 256 можливих адрес (16 кодів будинків × 16 кодів модулів = 256). Кожен модуль реагує тільки на команди, відправлені безпосередньо йому і на кілька широкомовних команд.

Наприклад, по мережі може прийти повідомлення виду: «модуль А3» а за ним команда «включитися» (turn on), що змушує модуль А3 включити приєднаний до нього пристрій. Керування кількома модулями здійснюється повідомленням виду: «модуль А3», «модуль А15» і «модуль А4», а потім команда «включитися». Результат - всі перераховані вище модулі повинні включити підключені до них пристрої.

## 2 ОПИС ОБРАНОЇ СИСТЕМИ

Зараз стрімко розвивається і впроваджується технологія SmartHouse, яка являє собою комплекс інженерних систем житлового будинку, створений для вирішення завдань життєзабезпечення, підвищення домашнього комфорту, безпеки та ресурсозбереження. Такі системи встановлюються по всьому будинку, і кожна з них виконує певне завдання, що призводить до необхідності централізованого управління [1]. Існує кілька підходів до вирішення даного завдання.

Одним з підходів є виділення кожній системі окремого сервера, який організовує її роботу. Перевагою даного підходу є те, що при виході з ладу однієї з систем інші продовжать функціонування. Недоліками є витрати необхідні на встановлення окремих серверів, децентралізація всього комплексу і необхідність налаштування кожної системи. Інший підхід передбачає встановлення декількох серверних комплексів, які забезпечують спільне управління і роботу всіх систем. Як правило, такі сервери не працюють в режимі максимального навантаження, велика частина ресурсів не використовується і як наслідок, кошти витрачені на закупівлю серверного обладнання, не завжди виправдані.

На основі проведеного аналізу інформації по сучасному стану впровадження технології smart house, було запропоновано рішення проблем і завдань централізації всіх систем на базі централізованого управління комплексом при установці декількох серверів. Був також проведений ряд експериментів по розгортанню серверів на звичайних ПК. Для вирішення проблем з ресурсами і несумісності деяких додатків при роботі на одній серверній ОС застосовувалася технологія віртуалізація, яка дозволила оптимізувати використання великої частини ресурсів сервера

### 2.1 Віртуальна машина

Віртуальна машина — модель обчислювальної машини, створеної шляхом віртуалізації обчислювальних ресурсів: процесора, оперативної пам'яті, пристроїв зберігання та вводу і виводу інформації. Віртуальна машина, на

відміну від програми емуляції конкретного пристрою, забезпечує повну емуляцію фізичної машини чи середовища виконання (для програми).

Раніше віртуальну машину визначали як «ефективну ізольовану копію реальної машини». Проте сучасні віртуальні машини можуть не мати прямого апаратного аналогу. Наприклад, в залежності від способу моделювання набору інструкцій віртуального центрального процесора, віртуальна машина може моделювати реальну або абстрактну обчислювальні машини. При моделюванні реальної обчислювальної машини набір інструкцій процесора віртуальної машини збігається з набором інструкцій обраного для моделювання центрального процесора. Віртуальні машини поділяються на 2 головні категорії, в залежності від їх використання та відповідності до реальної апаратури: системні (апаратні) віртуальні машини, що забезпечують повноцінну емуляцію всієї апаратної платформи і відповідно підтримують виконання операційної системи. прикладні віртуальні машини, які розроблені для виконання лише застосунків (прикладних програм), наприклад, Віртуальна машина Java[8].

Переваги технології віртуалізації:

- економічна ефективність;
- надійність;
- відновлюваність.

Віртуалізація являє собою процес відділення реалізації якого об'єкта чи процесу від його подання для користувача.

Види віртуалізації:

- віртуалізація платформ – створення програмних систем на основі існуючих АПК, залежних або не залежних від них;
- віртуалізація ресурсів – концентрація, абстрагування і спрощення управління групами ресурсів, таких як мережі, сховища даних.

Віртуалізація ОС – в ОС ПК як програма встановлюється платформа віртуалізації для створення ВМ, в яких встановлюються різні ОС(рис. 2.1).

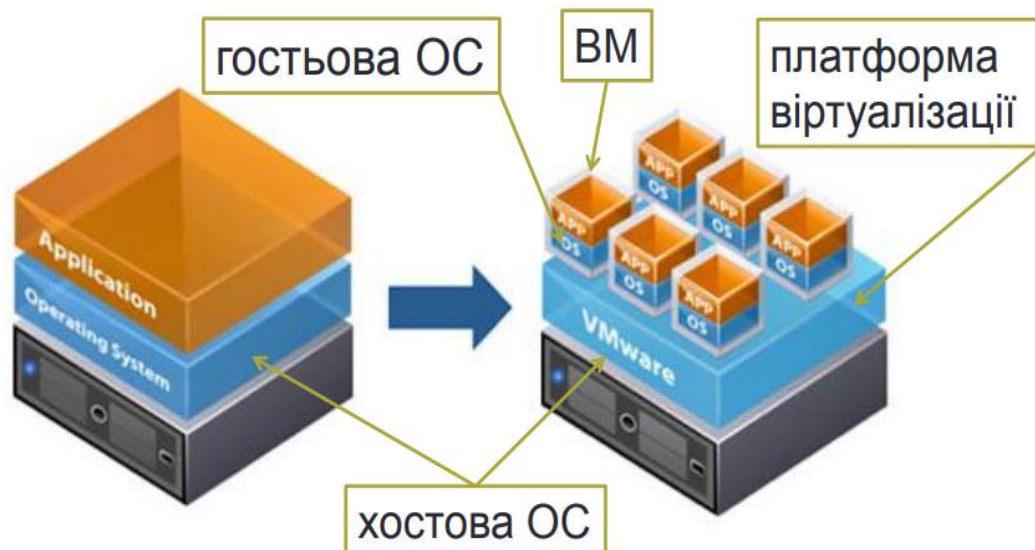


Рисунок 2.1 – Віртуалізація ОС

Віртуалізація платформ:

- Емуляція гостьової ОС – повна;
- Емуляція API гостьової ОС – часткова;
- Квазіемуляція гостьової ОС.

Віртуалізація ресурсів:

- об'єднання, агрегація і концентрація компонентів;
- кластеризація комп'ютерів та розподілені обчислення;
- поділ ресурсів (partitioning);
- інкапсуляція.

Класична архітектура з VM представлена на рис.2.2

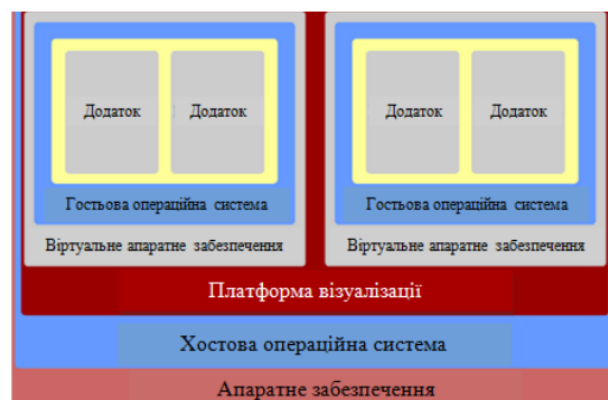


Рисунок 2.2 – Класична архітектура з VM

Недоліки використання ВМ:

- неможливість емуляції всіх пристроїв;
- віртуалізація вимагає додаткових апаратних ресурсів;
- деякі платформи віртуалізації вимогливі до конкретного апаратного забезпечення;
- вартість якісних платформ віртуалізації.

## 2.2 Системні віртуальні машини

Системні віртуальні машини дозволяють розподіл апаратних ресурсів фізичної машини між різними копіями віртуальних машин, на кожній з яких може бути встановлена своя операційна система. Пласт програмного забезпечення, що виконує віртуалізацію, називається гіпервізором. Гіпервізори поділяються на 2 типи: ті, що можуть виконуватися на «голій» апаратурі (1-й тип, або рідні (англ. native)), та ті, що виконуються в певній операційній системі (2-й тип, або хостові).

Основні переваги системних ВМ:

- Різні операційні системи можуть співіснувати на одному комп'ютері, і при цьому знаходитися в строгій ізоляції одна від одної;
- ВМ можуть забезпечувати розширений набір машинних інструкцій, адже при моделюванні абстрактної обчислювальної машини набір інструкцій процесора віртуальної машини може бути довільним;
- Широкі можливості контролю за програмами;
- Легкість модифікацій та відновлення.

Основний недолік:

- ВМ не така ефективна як реальна, тому що доступ до апаратури в ній відбувається опосередковано.

Різні ВМ, на кожній з яких може бути встановлена своя власна ОС (які також називається гостьовими ОС), часто використовуються для серверного об'єднання: різні сервіси (що повинні виконуватися на окремих машинах, щоб запобігти взаємовтручанню) запускаються в різних ВМ, проте на одній фізичній машині, що дозволяє економити апаратні ресурси.

### 2.3 Прикладні віртуальні машини

Прикладні віртуальні машини виконують звичайні програми всередині ОС. Вони зазвичай створюються коли програма запускається та знищуються після її завершення. Їхня ціль — забезпечити платформно-незалежне програмне середовище, яке дозволяє абстрагуватися від конкретної апаратури та операційної системи, на якій виконується програма. Прикладна ВМ забезпечує високорівневу абстракцію (наприклад, інтерпретатори високорівневих мов програмування — Lisp, Java, Python, Perl), в той час як системні ВМ зазвичай обмежуються низькорівневою абстракцією (машинним набором кодів). Сучасні прикладні ВМ, що реалізуються за допомогою інтерпретаторів, для підвищення швидкості виконання використовують компіляцію «на льоту» (англ. JIT — just-in-time) [6].

Прикладна віртуальна машина складається з машини состояний, брокера обробки запитів і трансляції подій, а також Web - сервісів, які обробляють стану інформаційних об'єктів в машині станів, і реалізується у вигляді спеціалізованого сервера Impress з динамічної інтерпретацією метамodelей.

Динамічна інтерпретація передбачає два режими обробки даних:

- зі зміною структури, коли разом з даними надходять нові метадані і потрібно повторно провести зв'язування абстрактних програмних компонентів, перетворення їх в машинний код і розгортання в оперативній пам'яті відповідних структур даних;
- без зміни структури, коли програмні компоненти метамodelей вже закеширувалися, підготовлені до роботи і динамічний компілятор проводить лише статистичну оптимізацію типів даних і класів сценарію.

Для реалізації двох режимів обробки, наведених вище, необхідні наступні функціональні модулі – інтерпретатор метамodelей, засоби синтаксичного і структурного аналізу метаданих, машина станів, сигнальний протокол і транспорт для поширення подій зміни метаданих, кеш програмного коду (підготовленого для виконання), кеш метамodelей, шаблонів і інтерфейсів (для побудованих динамічно програмних компонентів). Реалізація функціональних модулів являє собою три основних компоненти архітектури сервера з динамічної Інтерпретація метамodelей:

- брокер трансляції подій реалізований на базі двох протоколів, механізму IPC (Inter-Process Communication / міжпроцесовоє взаємодія), вбудованого в операційні системи (для взаємодії декількох потоків обробки в рамках одного сервера) і протоколу 0MQ (ZeroMQ), що забезпечує патерни швидкого обміну повідомленнями по TCP і надає уніфікує-ISSN 1560-8956 101;
- машина станів використовується для зберігання сесій і стану об'єктів інформаційної моделі предметної області, для чого пропонується використовувати безпосередньо оперативну пам'ять V8. Машина станів містить і дані та метадані, пов'язані на поточний момент часу з програмним кодом. Їх обробку здійснюють Web-сервіси, які звертаються до машини станів, а результати передаються відповідному брокеру обробки і трансляції;
- web-сервіс обробки даних забезпечує взаємодію між розподіленими додатками і між клієнтськими і серверними частинами прикладних програм по протоколам HTTP, HTTPS, WebSockets і SSE. Для тонких клієнтів (Web-інтерфейсів до прикладного ПЗ), Web-сервіс передбачає обмін даними в форматі JSON (формат серіалізації об'єктів JavaScript) за технологією AJAX (API асинхронних запитів за допомогою XMLHttpRequest, скорочено XHR-запитів).

## 2.4 Аналіз існуючих гіпервізорів

У моделі доцільно використовувати гіпервізор в якості основної операційної системи, тобто програмне середовище для віртуалізації повинне встановлюватися на сервер як базова ОС. Це дозволяє не витратити додаткові ресурси на запуск операційної системи, встановленої на серверній машині, а також з'являється можливість більш повного використання і розподілу наявних ресурсів між віртуальними машинами. При аналізі існуючих засобів були розглянуті наступні пакети для організації віртуальної середовища: Microsoft Hyper V, Xen server і VMware vSphere.

Microsoft Hyper-V надає програмну інфраструктуру і основні засоби управління, які можна використовувати для створення віртуалізованого середовища сервера та управління ним. Архітектура Hyper-V представлена на рис 2.3.

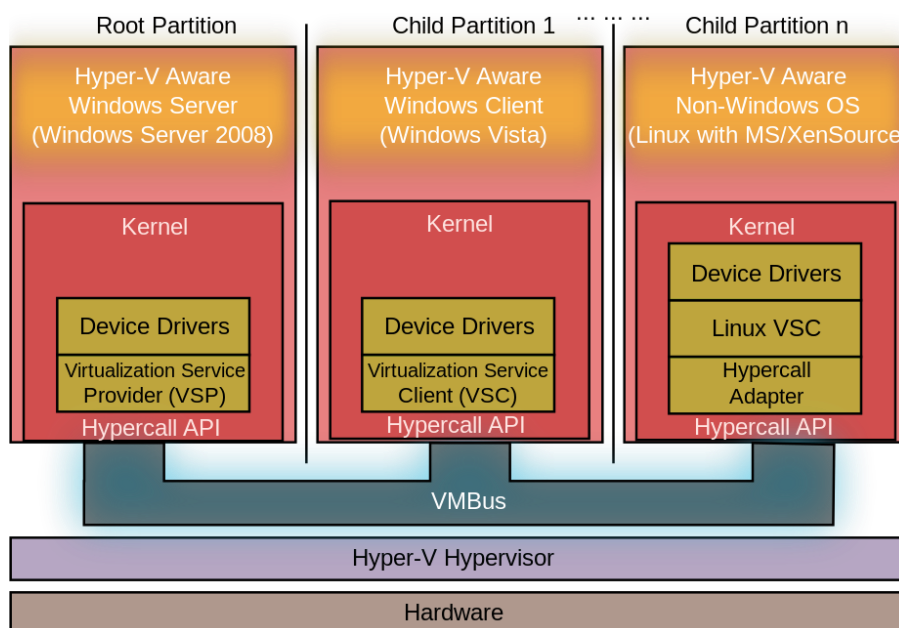


Рисунок 2.3 – Архитектура Hyper-V

Дана технологія віртуалізації середовища може використовуватися для багатьох цілей, спрямованих на підвищення ефективності і зниження витрат. Наприклад, віртуалізоване середовище сервера може допомогти в таких випадках, коли необхідне:

- Зменшення витрат на роботу і обслуговування фізичних серверів за рахунок збільшення використання обладнання (є можливість зменшення кількості обладнання, необхідного для виконання обсягу робіт сервера);
- Підвищення ефективності розробки та тестування за рахунок зменшення часу, потрібного для встановлення обладнання та програмного забезпечення з відтворенням середовища тестування.

Hyper-V Server передбачає ті ж вимоги до обладнання, що і Server 2008 з Hyper-V. Необхідний сервер з:

- x64-сумісних мікропроцесором Intel або AMD;
- апаратно забезпечується віртуалізацією (Intel VT або AMD-V);
- підтримкою апаратного запобігання виконання даних (DEP) через XD-розряд Intel або NX-розряд AMD.

Hyper-V Server встановлюється так само, як і MS Server 2008, з використанням аналогічної інтерактивної процедури установки. По завершенні установки пропонується вказати довгий пароль для облікового запису



адміністратора, після чого відкривається екран типу режиму Server Core з 2-я вікнами командного рядка на порожньому робочому столі.

Xen - програмне забезпечення для організації процесу віртуалізації, розроблене компанією Citrix. Даний пакет також є одним з найпопулярніших гіпервізорів. Він має можливість активної міграції, а також містить інструменти, необхідні для переміщення існуючих навантажень з фізичного середовища в віртуальне.(рис.2.4).

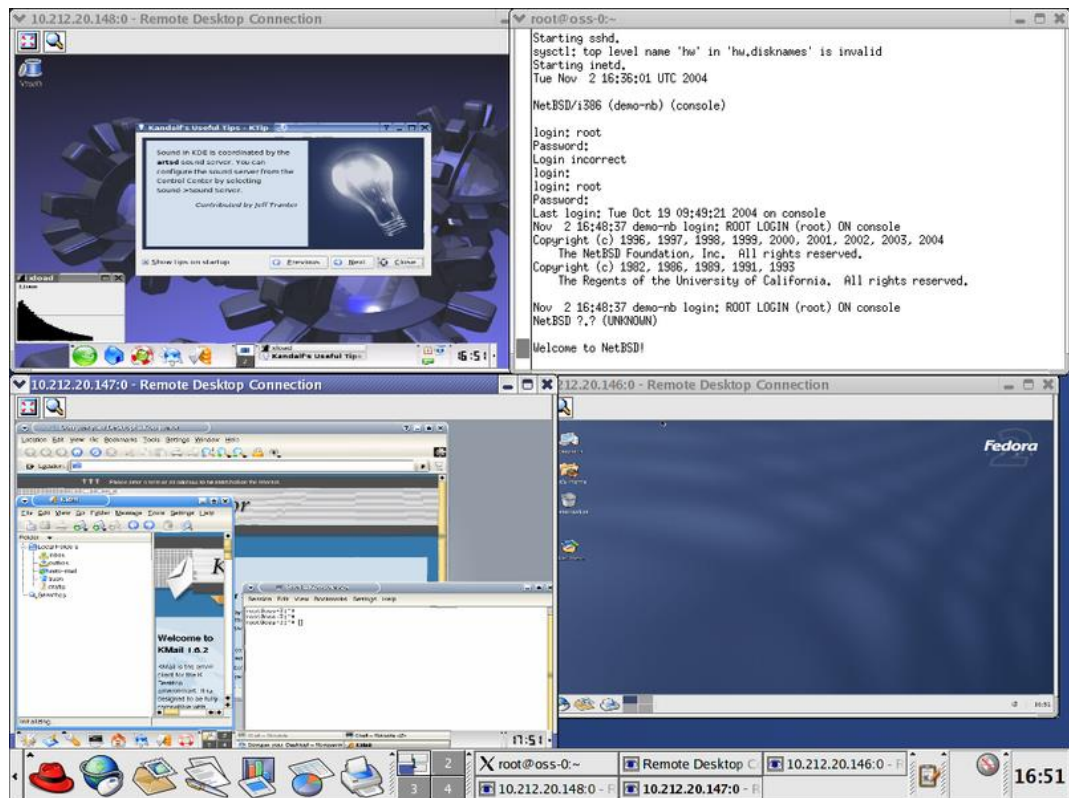


Рисунок 2.4 – Xen с запущенными NetBSD и тремя дистрибутивами Linux

Дозволяє користувачам створювати необмежену кількість серверів та віртуальних машин, якими можна безпечно і надійно управляти за допомогою єдиної консолі управління. Гіпервізор Xen server підтримує оптимізацію пам'яті, перерозподіляючи невикористану пам'ять сервера між віртуальними машинами, що знижує витрати і підвищує продуктивність і ступінь захищеності додатків. Також організована система балансування навантаження віртуальних машин на основі політики, яка визначається адміністратором системи. Найбільшою перевагою пакета є наявність безкоштовної версії, але з обмеженим функціоналом.

Гіпервізор Xen (для версії 3.4) реалізує мінімальний набір операцій для управління оперативною пам'яттю, станом процесора, таймерами реального часу і лічильниками тактів (TSC) процесора, переривань і контролем за DMA. Всі інші функції, такі як реалізація дискових і блокових пристроїв, створення та видалення віртуальних машин, їх міграція між серверами і т. Д. Реалізується в керуючому домені. За рахунок цього розмір гіпервізора виходить дуже малим (для версії 3.4 розмір двійкового коду за все гіпервізора менше 600 КБ), так само як і розмір його початкового тексту. За задумом авторів це збільшує стійкість системи віртуалізації, так як помилка в компонентах поза гіпервізора не призводить до компрометації / пошкодження самого гіпервізора і обмежує пошкодження тільки що вийшов з ладу компонентом, не заважаючи працювати іншим.

VMware vSphere - один з найпопулярніших гіпервізорів компанії VMware. Представляє собою dos оболонку, за допомогою якої можна налаштувати базові опції для сервера. Управління програмою здійснюється через клієнтську програму, яка встановлюється на комп'ютер користувача і здійснює підключення, як по локальній мережі, так і за допомогою мережі інтернет. Одним з головних переваг цього програмного пакету є можливість гнучкого налаштування резервного копіювання, а також можливість додавання віртуального сховища працюючих віртуальних машин без переривання роботи та простоїв сервера. Гіпервізор підтримує додаткові продукти і модулі, що розробляються компанією VMware, а також має дружній і інтуїтивно зрозумілий інтерфейс(рис. 2.5).

## VMware vSphere

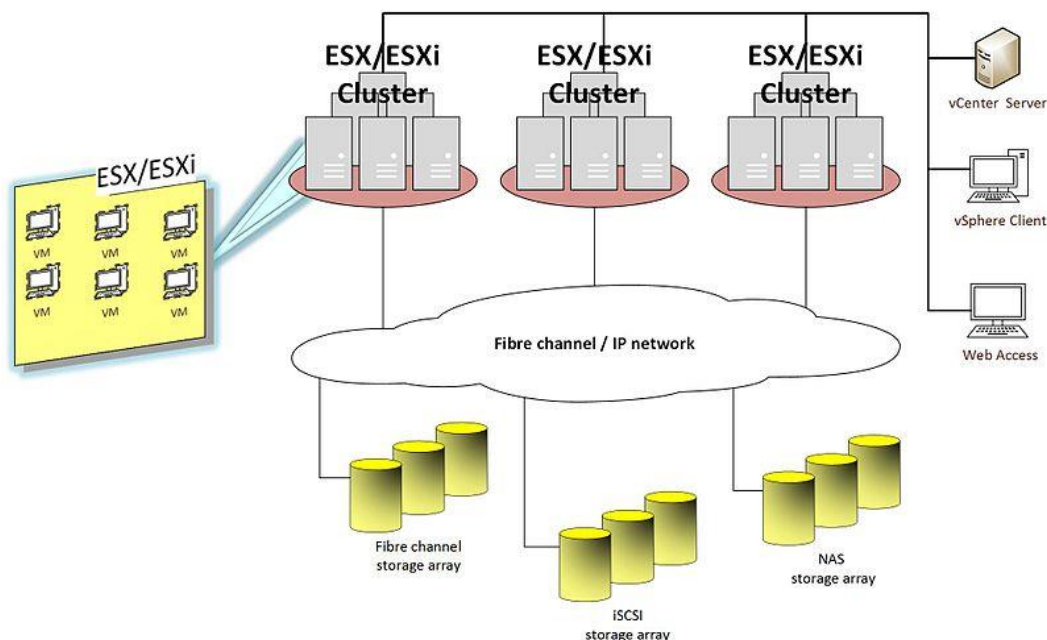


Рисунок 2.5 – VMware vSphere

Згідно з документацією компанії VMware система має наступні основні компоненти:

- ESX/ESXi— засіб віртуалізації, що абстрагує ресурси фізичних серверів такі як: процесор, пам'ять, жорсткі диски та інші, з метою створення віртуальних машин;
- vCenter Server — засіб для конфігурації, виділення ресурсів та управління платформою;
- vSphere Client — інтерфейс, який дозволяє користувачам віддалено підключатися до vCenter Server;
- Web Access — веб-інтерфейс, що надає можливість керування віртуальними машинами та надає доступ до віддалених консолів;
- VMFS — файлова система віртуальної машини (Virtual Machine File System), високопродуктивна кластерна файлова система[en] серверів (хостів) ESX/ESXi;
- Virtual SMP — віртуальне симетричне мультипроцесування (Symmetric Multiprocessing), функція, яка дозволяє одній віртуальній машині використовувати кілька фізичних процесорів одночасно;

- VMotion — функція, що забезпечує перенос віртуальної машини з одного фізичного серверу до іншого без зупинки її роботи;
- Storage VMotion — функція, що забезпечує перенос файлів віртуальної машини в середині фізичного дискового масиву, або навіть з одного дискового масиву до іншого, без втрати доступу до цих файлів;
- Fault tolerance — відмовостійкість, функція яка дозволяє мати повну копію віртуальної машини на іншому фізичному сервері в режимі онлайн, тобто будь-які зміни на конкретній віртуальній машині миттєво вносяться до її копії. В разі виходу з ладу основного фізичного серверу копія віртуальної машини на іншому фізичному сервері автоматично перехоплює навантаження без жодної затримки.

Слід зазначити, що значним недоліком цього програмного продукту є його вартість. Однак пакет має кілька редакцій, кожна з яких вирізняється з-поміж інших кількістю можливостей та спрямована на вирішення певних задач.

### 3 ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ СИСТЕМИ УПРАВЛІННЯ SMART HOUSE З ВИКОРИСТАННЯМ ВІРТУАЛЬНИХ МАШИН

#### 3.1 Базові технічні вимоги до проектування

В результаті проведених експериментів було встановлено, що розгорнути гіпервізор можливо не тільки на сервері, а й на персональному комп'ютері користувача, але з деякими модифікаціями такого комп'ютера, зокрема з заміною мережевої карти, оскільки гіпервізор не зміг виявити і задіяти стандартну мережеву карту, вбудовану в материнську плату. У таблиці 1 наведені основні характеристики тестового комп'ютера.

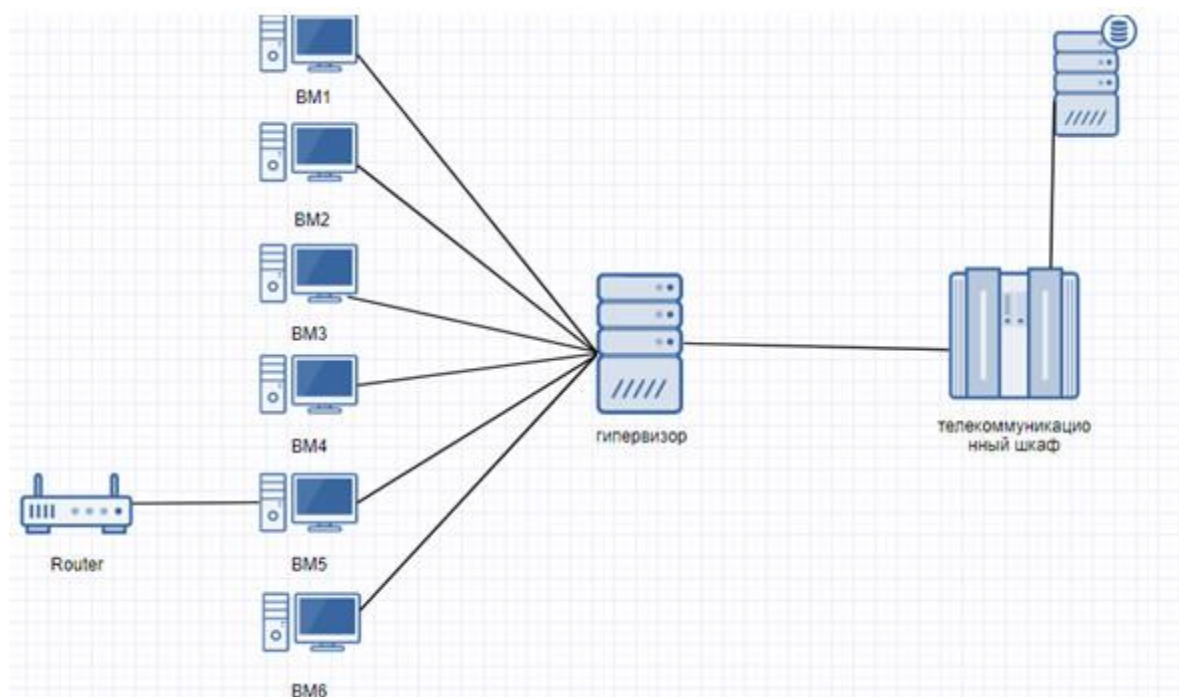
Таблиця 3.1 – Базові технічні вимоги до проектування

Процесор	Intel Core i3-2300
Материнська карта	intel dz68bc
ОЗУ	8-16Гб
Мережна карта	Достатньо вбудованої
Об'єм жорсткого диску	Від 40 Гб

Після розгортання гіпервізора на тестовому комп'ютері з вбудованою мережевою картою, система не змогла коректно визначити і підключити дану карту. Далі були протестовані зовнішні мережеві карти різних фірм проіхвідітелей. В результаті експериментів вдалося інстальювати і підключити мережеве обладнання комплексу шляхом установки мережевої карти фірми IBM[4].

#### 3.2 Визначення структури моделі

Після експериментальної перевірки, була розроблена базова система управління smart house з використанням гіпервізора(рис. 3.1).



Рисинок 3.1 – Система управління smart house використанням гіпервізора

На даній схемі зображено сервер з гіпервізором, де встановлено 6 віртуальних машин, кожна з яких виконує певну функцію, а саме:

Віртуальна машина 1 виконує функції серверу для управління підсистемою безпеки. Даний сервер забезпечує безпеку в приміщенні, при цьому існує можливість налаштування. У даному випадку можливі варіанти організації таких систем як охоронна сигналізація, світлова тривожна сигналізація, тривожне оповіщення за телефоном, світлова імітація присутності людини і т.д.

Віртуальна машина 2 відповідає за автоматизацію побутових процесів у будинку. Це можуть бути різноманітні одностійні дії які людина здатна доручити виконувати керівним механізмам;

Віртуальна машина 3 служить сервером для підсистеми мультимедіа. Можлива організація віддаленого управління різноманітними мультимедійними підсистемами і т.д.;

Віртуальна машина 4 є сервером відеоспостереження. Існує можливість організації бездротового аудіо/відео спостереження. Система представлена сукупністю відеокамер, встановлених у приміщенні та на території навколо будинку. Віртуальна машина забезпечує доступ до записів з камер відеоспостереження та надання віддаленого доступу до камер;

Віртуальна машина 5 забезпечує доступ до мережі Інтернет. Вона може надавати як дротовий так і бездротовий доступ до мережі з налаштуванням різних прав доступу;

Віртуальна машина 6 є сервером резервного копіювання, до завдання якої входить створення резервних копій всіх віртуальних машин, встановлених на сервері.

Після створення резервних копій віртуальна машина 6 відправляє їх в мережне сховище. У разі збою або виходу з ладу однієї або декількох віртуальних машин за допомогою даного сервера їх можна легко відновити з копій. Даний сервер може бути налаштований на моніторинг всіх важливих системних файлів і в разі їх пошкодження програма автоматично замінить пошкоджені файли і перезавантажить систему. Зв'язок сервера з мережним сховищем забезпечує маршрутизатор, який також може забезпечувати головний сервер з підключенням до мережі інтернет (рис. 3.2).

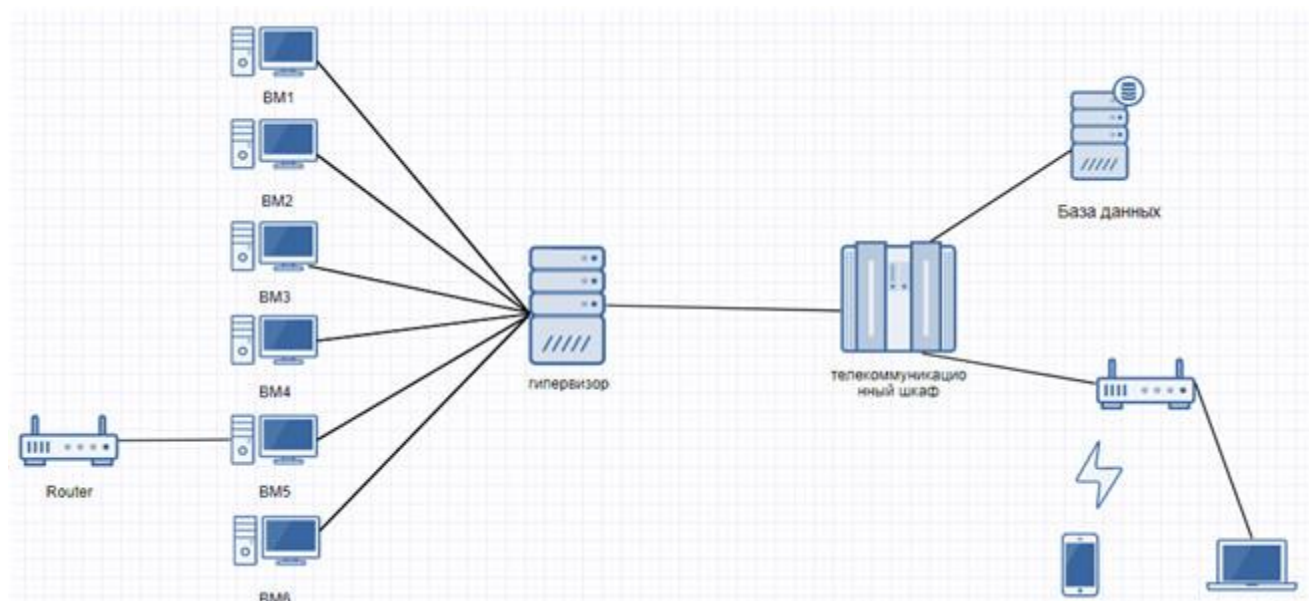


Рисунок 3.2 - Система управління на основі СКМ

Також в даній моделі був змінений порядок підключення пристроїв. У першій моделі до маршрутизатора підключався головний сервер та мережне сховище, а сам маршрутизатор підключався до інтернет. У другій моделі сервер та мережне сховище підключаються до телекомунікаційної шафи, а сам шафа підключається до маршрутизатора.

### 3.3 Вибір гіпервізора методом аналізу ієрархії

Метод аналізу ієрархій (MAI) складається в декомпозиції проблеми вибору кращого варіанту протоколу маршрутизації на прості складові частини отриманні суджень експертів по парним порівнянь елементів проблеми вибору, які представляються у вигляді матриці

$$A = \begin{pmatrix} a_{11} & \dots & a_{1j} & \dots & a_{1n} \\ a_{21} & \dots & a_{2j} & \dots & a_{2n} \\ \dots & & a_{ij} & & \dots \\ a_{n1} & \dots & a_{nj} & \dots & a_{nn} \end{pmatrix}, \quad (3.1)$$

Де  $a_{ij}=w_i/w_j$  - оцінки парних порівнянь елементів.

Діагональ цієї матриці заповнюється значеннями "1", а елементи матриці, що лежать нижче діагоналі, заповнюються відповідними зворотними значеннями.

Оцінки парних порівнянь елементів знаходяться з використанням суб'єктивних суджень експертів, чисельно визначених за шкалою відносної важливості елементів, яка представлена в табл. 3.2.

Таблиця 3.2 - Шкала відносної важливості елементів порівнянн

Інтенсивність відносної важливості	Визначення
1	Рівна важливість елементів порівняння
3	Помірна перевага одного елемента над іншим
5	Істотна перевага одного елемента над іншим
7	Значна перевага одного елемента над іншим
9	Дуже сильна перевага одного елемента над іншим
2,4,6,8	Проміжні рішення між двома судженнями

Оцінки парних порівнянь елементів знаходяться з використанням суб'єктивних суджень експертів, що чисельно визначаються за шкалою відносної важливості елементів.



Далі виконується обробка сформованих матриць парних порівнянь елементів ієрархій. З математичної точки зору це завдання обробки зводиться до обчислення головного власного вектора, відповідного максимальному власному значенню матриці. В результаті обробки отриманої матриці отримують компоненти власного вектора  $V_j$  і глобального вектора пріоритетів  $P_j$

$$P_j = \frac{V_j}{S}, \quad V_j = \sqrt[n]{\prod_{i=1}^n a_{ij}}, \quad S = \sum_{j=1}^n V_j. \quad (3.2)$$

де  $n$  - число варіантів алгоритмів самоорганізації, що порівнюється.

З використанням отриманих даних обчислюються значення компонент вектора глобальних пріоритетів  $C$  згідно

$$C_i = \sum_{j=1}^n P_j Q_{ij}, \quad i = \overline{1, N}, \quad (3.3)$$

За максимальним значенням компонент вектора глобальних пріоритетів (3.1) вибирається відповідний кращий варіант системи. У якості міри узгодженості суджень експертів розглядають два показника: - індекс узгодженості; - відношення узгодженості (ОС).

Щоб оцінити, чи є отримане узгодження суджень експертів прийнятним чи ні, його порівнюють з випадковим індексом  $C_{in}$ . Випадковим індексом називають індекс узгодженості, розрахований для квадратної  $n$ -мірної позитивної обернено симетричною матриці, елементи якої згенеровані випадковим чином за допомогою генератора випадкових чисел, розподілених по рівномірному закону для інтервалу значень від 1 до 15. У табл. 3.3 представлені серединні узгодженості для випадкових матриць порядку від 1 до 15.

Таблиця 3.3 - Величина випадкового індексу

Розмір матриці $n$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Випадковий індекс $C_{in}$	0	0	0,5 8	0,9 0	1,1 2	1, 24	1,3 2	1,4 1	1,4 5	1,4 9	1,5 1	1,5 4	1,5 6	1, 57	1,5 9

Отримавши в результаті розрахунку індекс узгодженості і вибравши з табл. 3 випадковий індекс для розмірності матриці 5( $C_n=1,12$ ) розраховують відношення узгодженості

$$OC = I_c/C_n$$

Якщо величина  $OC < 0,15$ , то ступінь узгодженості суджень експертів слід вважати прийнятною. В іншому випадку експертам рекомендується переглянути свої судження

Для вибору переважного гіпервізора, що проектується використовувалися дані чотирьох гіпервізорів: №1 – VMware vSphere; №2 – Xen server; №3 – Microsoft Hyper V; №4 – VMware ESX.

У табл. 3.4 приведена матриця парних порівнянь показників якості гіпервізорів до (1), потім обчислені власний вектор і вектор пріоритетів згідно (2). За показники якості були взяті такі характеристики гіпервізорів: K1 – Вартість, K2 – підтримка додаткових модулів, K3 – Узгодженість, K4 – Підтримка конвертора.

Таблиця 3.4 - Матриця парних порівнянь показників гіпервізорів

1 експ.	K1	K2	K3	K4	V1	P1
K1	1	2	3	4	2,21	0,46
K2	1/2	1	2	3	1,31	0,27
K3	1/3	1/2	1	2	0,75	0,16
K4	1/4	1/3	1/2	1	0,45	0,09

У табл. 3.5 показані результати обчислення значень компонент вектора глобальних пріоритетів  $C_j$  згідно співвідношенню (3)

Таблиця 3.5 - Обчислення значень компонента вектора глобальних пріоритетів

Протоколи позиціонування	K1	K2	K3	K4	$C_j$
Xen server	0.50	0.50	0.50	0.47	0.50
VMware vSphere	0.26	0.26	0.29	0.28	0.27
Microsoft Hyper V	0.13	0.14	0.12	0.15	0.13
VMware ESX	0.09	0.08	0.09	0.08	0.08

З табл. 3.5 видно, що згідно МАІ кращим варіантом гіпервізора є N1 – Xen server, який відповідає максимальній компоненті вектору пріоритетів.

В результаті проведеного порівняльного аналізу розглянутих пакетів був обраний гіпервізор на основі Xen server. Перевагами пакета є достатній для вирішення поставлених завдань набір функцій і наявність безкоштовної версії програми, що значно знижує витрати для організації віртуальної середовища.

### 3.4 Установка і налаштування Citrix XenServer

Вибираємо розкладку клавіатури(рис. 3.3).

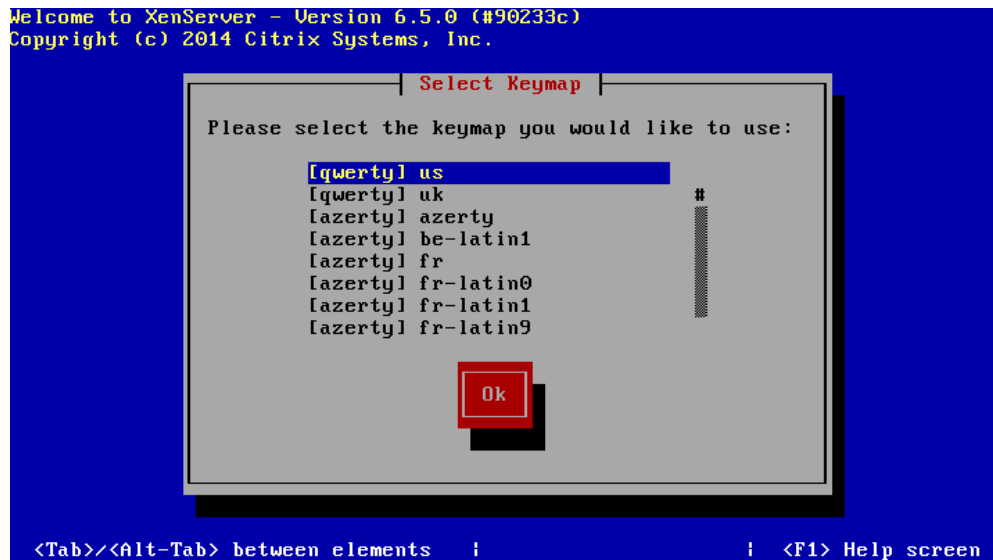


Рисунок 3.3 - Вибір розкладки клавіатури

Далі XenServer попереджає що всі дані на дисках будуть знищені, також можливо встановити спеціальні драйвера натиснувши F9 якщо це необхідно(рис. 3.4).

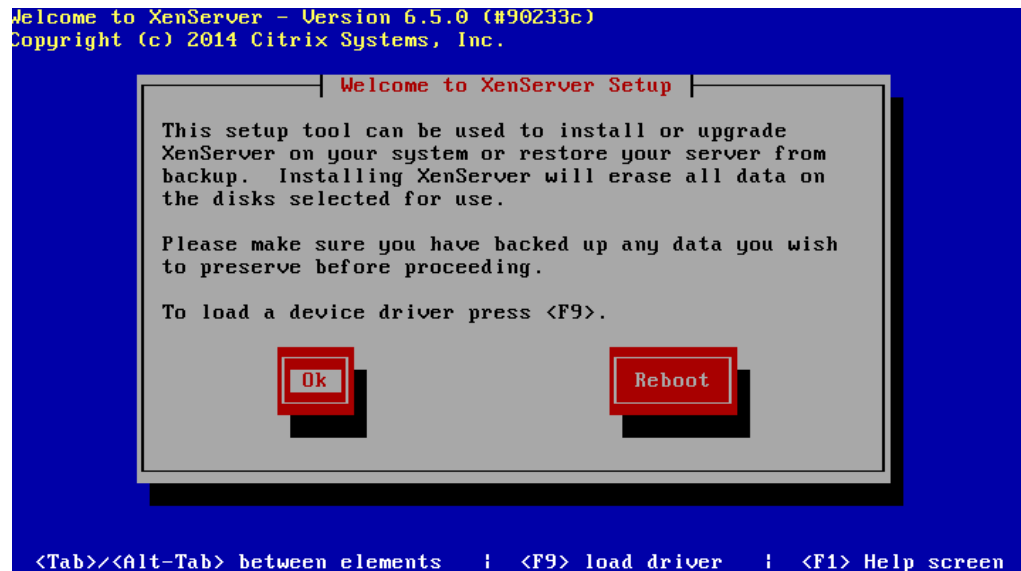


Рисунок 3.4 - Попереджає що всі дані на дисках будуть знищені

Приймаємо угоду(рис. 3.5).

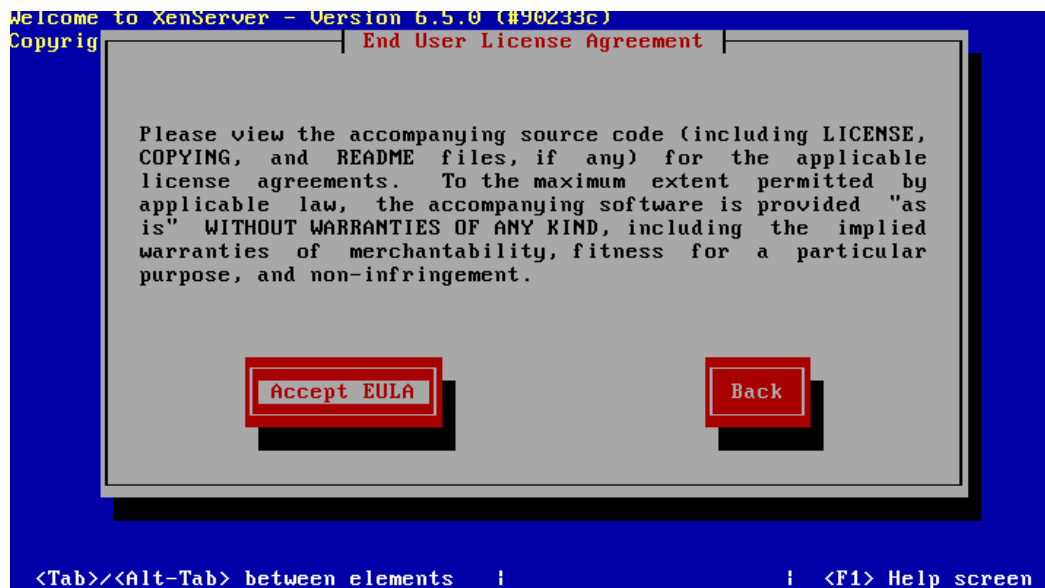


Рисунок 3.5 - Приймаємо угоду

Далі вибираємо диск на якому буде розташовуватися сховище віртуальних машин. Якщо необхідно сховище яке буде розширюватися динамічно, вибираємо Thin provisioning(рис. 3.6).

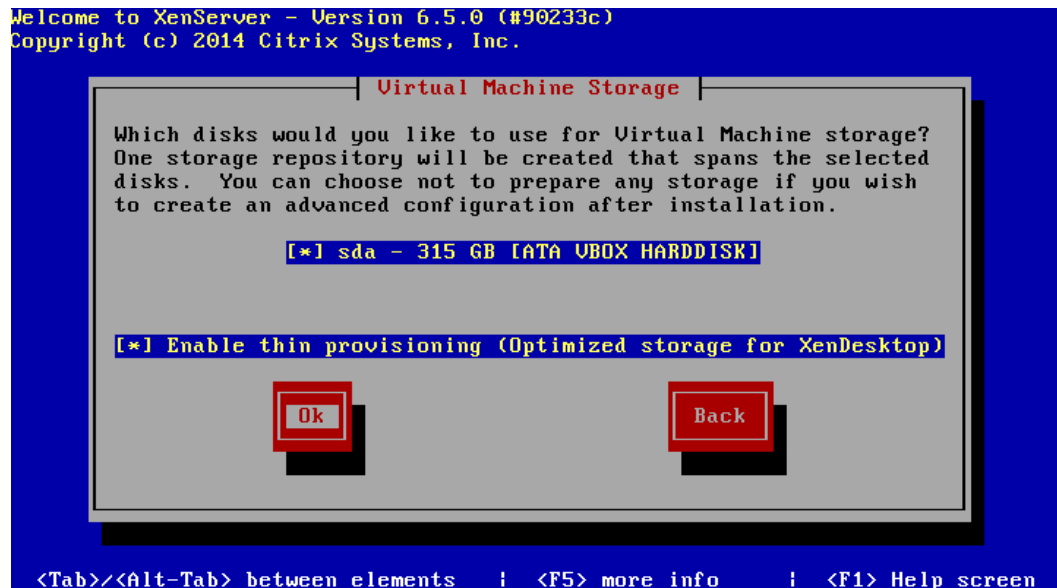


Рисунок 3.6 - Вибираємо диск

Далі вибираємо шлях файлів установки. У нашому випадку це локальне сховище(рис. 3.7).

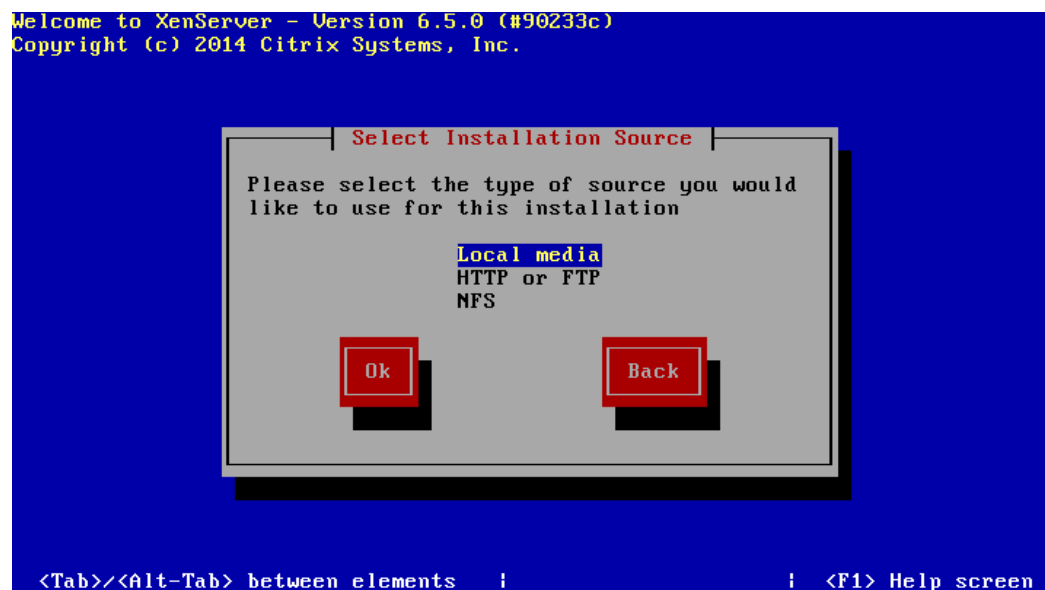


Рисунок 3.7 - Вибираємо шлях файлів установки

Вказуємо мережевий інтерфейс для управління сервером(рис. 3.8).

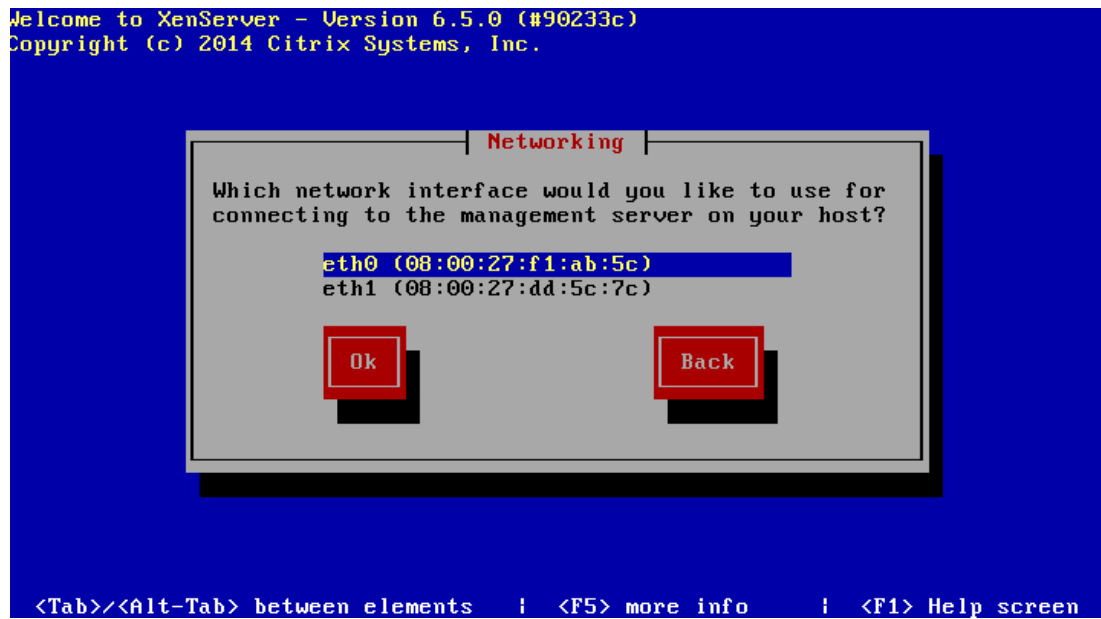


Рисунок 3.8 - Вказуємо мережевий інтерфейс

Вказуємо мережеві налаштування для керуючого інтерфейсу(рис. 3.9).

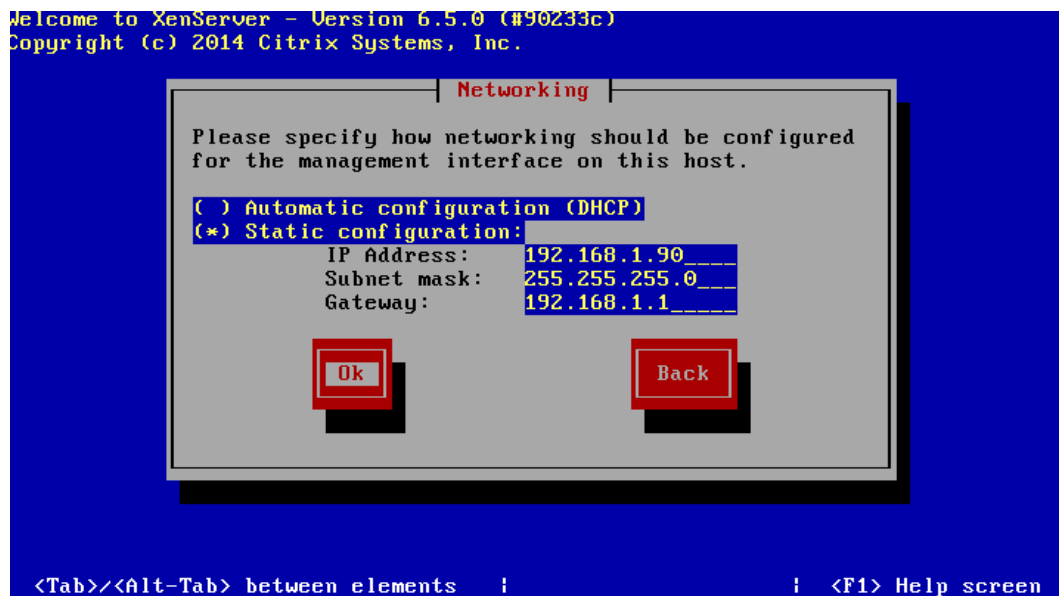


Рисунок 3.9 - Вказуємо мережеві налаштування

Вказуємо Hostname і DNS сервера(рис. 3.10).

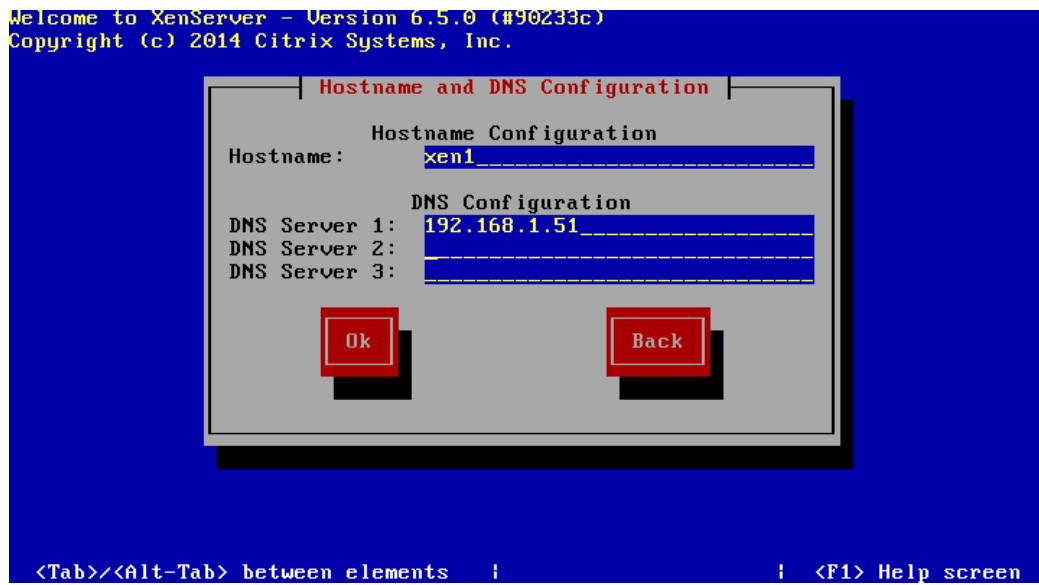


Рисунок 3.10 - Вказуємо Hostname і DNS сервера

Вказуємо адреси NTP серверів(рис. 3.11).

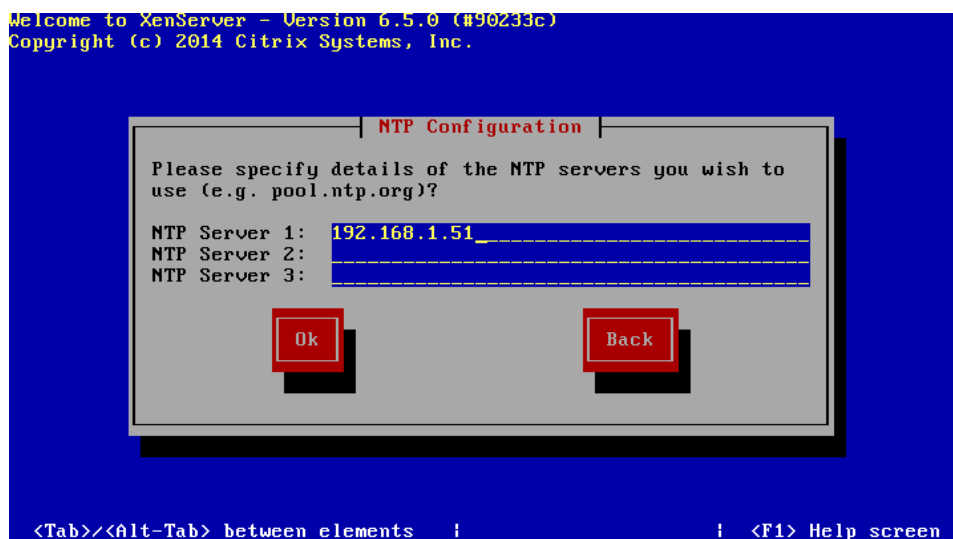


Рисунок 3.11 - Вказуємо адреси NTP серверів

Підтверджуємо установку XenServer(рис. 3.12).

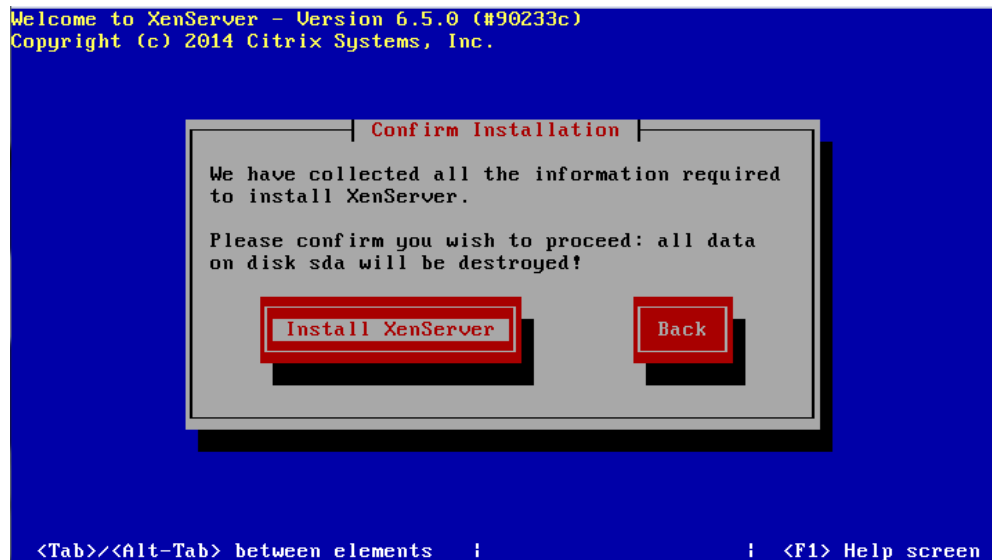


Рисунок 3.12 - Підтверджуємо установку XenServer

Після завантаження сервера встановлюємо Citrix XenCenter. Актуальну версію можна знайти тут (Windows Management Console). Запускаємо XenCenter, вибираємо Add New Server(рис. 3.13).

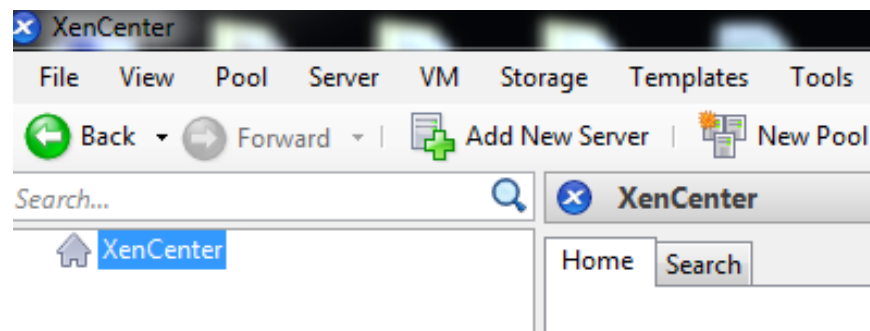


Рисунок 3.13 - Запуск XenCenter

Перевіряємо введені дані і створюємо віртуальну машину(рис. 3.14).



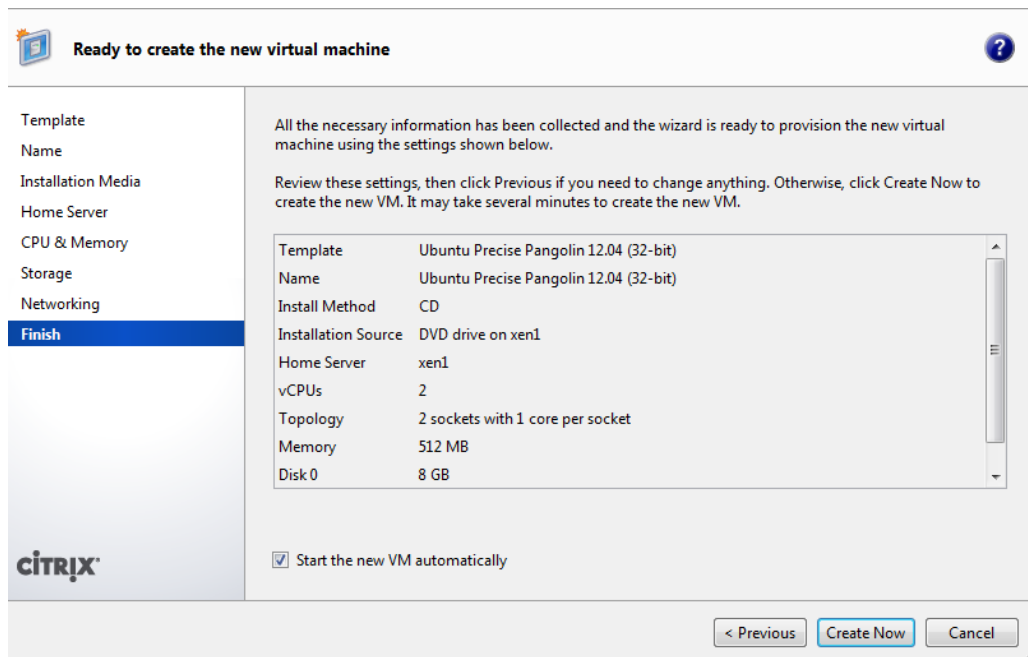


Рисунок 3.14 - Перевіряємо введені дані і створюємо віртуальну машину

Додамо загальне сховище ISCSI(рис. 3.15-3.18).

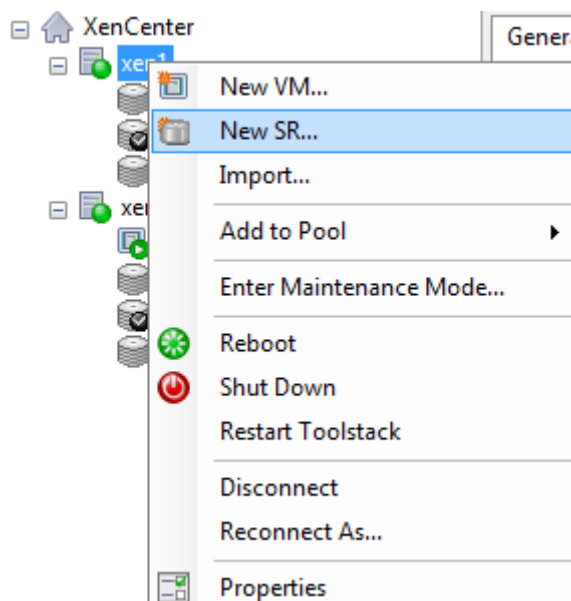


Рисунок 3.15 - Відкриваємо консоль XenCenter, вибираємо хост, далі натискаємо - New SR.

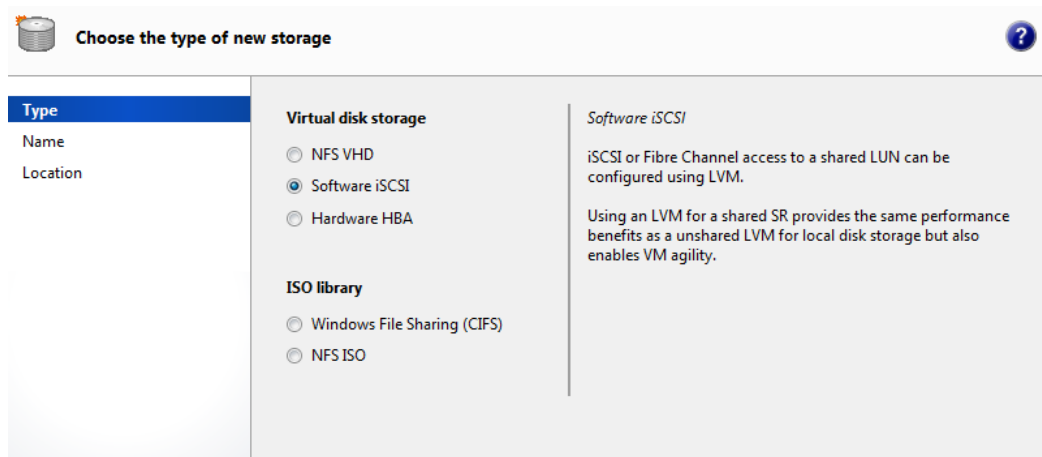


Рисунок 3.16 - Вибираємо Software iSCSI.

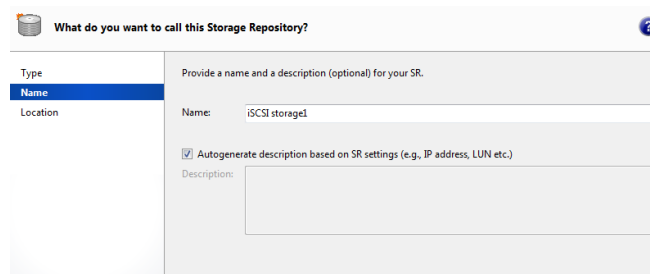


Рисунок 3.17 - Вказуємо ім'я сховища.

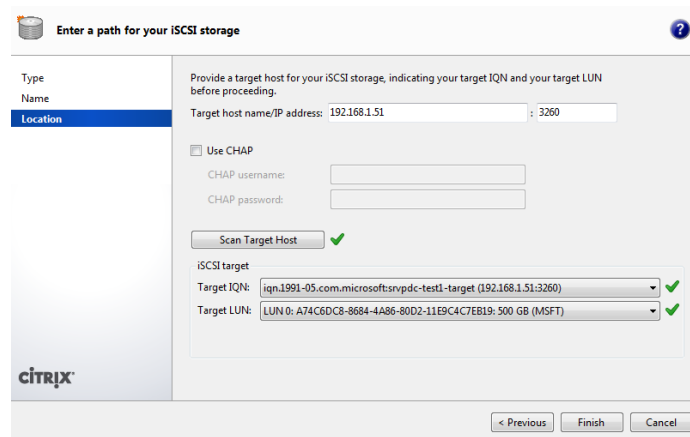


Рисунок 3.18 - Далі вказуємо адресу iSCSI сховища, target IQN і LUN.

Тепер сховище iSCSI з'явилося у хоста Xen1.

#### 4 АНАЛІЗ СПРОЕКТОВАНОЇ СИСТЕМИ

Отже, в даній системі управління SmartHouse вирішується проблема невикористовуваних серверних ресурсів, оскільки гіпервізор дозволяє задіяти всі ресурси фізичної системи, а також вирішує проблеми несумісності деяких програм. Дана система досить універсальна - її можна адаптувати під будь-які потреби системи SmartHouse та існуючу кількість віртуальних машин, що встановлюються на фізичний сервер. Вона залежить безпосередньо від характеристик сервера, чим потужніший сервер, тим більше віртуальних машин можна на нього поставити.

Гіпервізор можна контролювати за допомогою програми клієнта, яка може бути встановлена на будь-який комп'ютер користувача, який при цьому перебуває в одній мережі з сервером. Сервер резервного копіювання та відновлення забезпечує самообслуговування системи, завдяки чому система може стежити за своїм станом і в разі виникнення проблем провести автоматичне відновлення без втручання користувача. Мережне сховище при цьому забезпечує надійне зберігання резервних копій віртуальних машин, воно не залежить від стану головного сервера і в разі його відмови, і заміні, всі віртуальні машини, що існували на старому сервері, можуть бути відновлені на новому сервері зі збереженням всіх параметрів.

Вартість комплексу можна скоротити, шляхом використання замість серверної платформи стаціонарного призначеного для користувача ПК, використання безкоштовного ПЗ для організації віртуальної середовища. На ВМ можна встановити безкоштовні дистрибутиви Linux, але слід звернути увагу на те, що вибір дистрибутивів, повністю залежить від адміністратора комплексу.

## ВИСНОВОК

В даній атестаційній роботі було проведено аналіз централізованих та децентралізованих засобів управління системами Smart House різних систем, а також було запропоновано систему управління Smart House з використанням віртуальних машин. Був проведений аналіз існуючих гіпервізорів, і вибір гіпервізора для проектування системи управління Smart House.

Також було спроектовано та реалізовано централізовану систему управління Smart House з використанням віртуальних машин.

В результаті проведених експериментів було встановлено базові технічні вимоги до проектування та була розроблена базова система управління smart house з використанням гіпервізора.

В результаті роботи був зроблений аналіз спроектованої системи управління.

Результати даної атестаційної роботи можуть бути використані для роботи комплексної системи SmartHome з елементами штучного інтелекту, або як окремий модуль керування пристроями для розробки власної системи.

## ПЕРЕЛІК ПОСИЛАНЬ

1. Josyula V., orr V., Page G. Cloud Computing Automating the Virtualized Data Center – Cisco Press, 2012.
2. Carvalho L. Windows Server 2012 Hyper-V Cookbook. – Packt Publishing.
3. Лэнгоун Д., Лейбовичи А. Виртуализация настольных компьютеров с помощью VMware.
4. Bunch C. Automating vSphere: With VMware vCenter Orchestrator. – Vmware Press, 2012.
5. Tulloch M. Optimizing and Troubleshooting Hyper-V Networking – Microsoft Press, 2013.
6. Siron E. Microsoft Hyper-V Cluster Design.
7. Lutron Electronics, Inc. - Dimmers And Lighting Controls. – Режим доступа:<http://www.lutron.com/enUS/CompanyInfo/Pages/AboutUS/OurStory.aspx>.
8. Михеев М. Администрирование VMware vSphere 5. – М.: ДМК Пресс, 2012.