

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Сумський державний університет
Кафедра електроніки і комп'ютерної техніки

ПОЯСНЮВАЛЬНА ЗАПИСКА

до дипломного проекту на тему:

«Пристрій комутації для локальної мережі
на базі протоколу RIP»

Завідувач кафедри

Керівник проекту

Проектував студент

А.С. Опанасюк

О.В. Бережна

Б.С. Сенько

Суми
2020 р.

Календарний план

№ п/п	Найменування етапів дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1.	Огляд технічної літератури	20.04.20	
2.	Розробка алгоритму функціонування та структурної схеми	25.04.20	
3.	Розробка схеми електричної функціональної	30.04.20	
4.	Розрахунок вузлів та блоків пристрою та розробка схеми електричної принципової	10.05.20	
5.	Оформлення графічної частини	20.05.20	
6.	Оформлення пояснювальної записки	25.05.20	
7.	Рецензування та підготовка до захисту	30.05.20	

Студент-дипломник _____

Керівник проекту _____

РЕФЕРАТ

Пояснювальна записка містить: 46 сторінок, 18 рисунків, 5 таблиць, 14 джерел літератури.

Дипломний проект на тему «Пристрій комутації для локальної мережі на базі протоколу RIP» містить пояснювальну записку та графічну частину.

Графічна частина містить у собі 4 креслення: схему алгоритму, електричну структурну, функціональну та принципову схеми.

Пояснювальна записка містить п'ять розділів.

У першому розділі «Огляд літератури та постановка завдання проектування» розглядаються функції маршрутизатора, протоколи маршрутизації та дистанційно-векторний протокол.

У другому розділі «Розробка алгоритму функціонування та структурної схеми маршрутизатора» виконується розроблення структурної схеми та алгоритму функціонування пристрою.

У третьому розділі «Розробка схеми електричної функціональної пристрою комутації на базі протоколу RIP» розроблюється функціональна схема пристрою та розглядаються характеристики блоків схеми.

У четвертому розділі «Розробка та розрахунок принципів електричних схем вузлів та блоків пристрою» розроблюється схема електрична принципова пристрою та розглядаються характеристики її елементів.

П'ятий розділ присвячений розробленню програмного забезпечення пристрою.

ЗМІСТ

	С.
Вступ	4
1 Огляд літератури та постановка завдання проектування.....	8
1.1 Маршрутизатор та його функції.....	8
1.2 Дистанційно-векторний протокол RIP.....	13
2 Розробка алгоритму функціонування та структурної схеми маршрутизатора.....	17
2.1 Протокол RIP. Алгоритм функціонування.....	17
2.2 Структурна схема пристрою комутації	22
3 Розробка схеми електричної функціональної пристрою комутації на базі протоколу RIP.....	27
4 Розробка та розрахунок принципових електричних схем вузлів та блоків пристрою.....	29
4.1 Мікроконтролер 38051F300.....	29
4.2 Ethernet-контролер CP2200.....	39
4.3 Блок узгодження та гальванічної розв'язки.....	42
5 Розроблення програмного забезпечення пристрою.....	44
Висновки.....	45
Список літератури.....	46

					ЦЗДВН 6.05080202.869 ПЗ			
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>				
<i>Разраб.</i>		Сенько Б.С.			Пристрій комутації для локальної мережі на базі протоколу RIP Пояснювальна записка	<i>Лит.</i>	<i>Лист</i>	<i>Листов</i>
<i>Провер.</i>		Бережна О.В.					3	46
<i>Реценз.</i>						СумДУ, гр. ЕСдн – 51п		
<i>Н. Контр.</i>		Бережна О.В.						
<i>Утверд.</i>		Опанасюк А.С						

ВСТУП

Передача інформації між комп'ютерами існує із моменту виникнення обчислювальної техніки, яка дозволяє організувати спільну роботу окремих комп'ютерів, вирішувати завдання за допомогою декількох комп'ютерів, спільно використовувати ресурси та вирішувати безліч інших проблем. При цьому кожний з комп'ютерів спеціалізується на виконанні однієї функції. Способів та засобів обміну інформацією за останнім часом запропоновано безліч: від найпростішого перенесення файлів за допомогою дисків до всесвітньої комп'ютерної мережі Internet, здатної зв'язати комп'ютери, які розташовані у різних містах, державах та континентах. Важливе ж місце у цієї ієрархії приділяється локальним мережам [1].

Локальна мережа - це складна система, що включає тисячі найрізноманітніших компонентів: комп'ютери різних типів, системне та прикладне програмне забезпечення, мережеві адаптери, концентратори, комутатори, маршрутизатори, а також кабельну систему [2].

Локальну мережу визначають як «систему для безпосереднього з'єднання багатьох комп'ютерів». При цьому мається на увазі, що інформація передається від комп'ютера до комп'ютера без посередників та по єдиному середовищу передачі інформації. Однак, говорити про єдине середовище передачі в сучасній локальній мережі не доводиться. Наприклад, у межах однієї мережі можуть використовуватися як електричні кабелі різних типів, так й оптоволоконні кабелі. Визначення передачі інформації «без посередників» також не занадто чітко, адже в сучасних локальних мережах використовуються найрізноманітніші концентратори, комутатори, маршрутизатори, мости, які часом виконують досить складне оброблення переданої інформації [2,3].

Напевно, найбільше точно було б визначити локальну таку мережу як таку що дозволяє користувачам не звертати увагу на принципи з'єднання ліній зв'язку. Комп'ютери, які зв'язані за допомогою локальної мережі, поєднуються, по суті, в один віртуальний комп'ютер, ресурси якого можуть бути доступні усім користувачам, причому цей доступ не менш зручний, ніж доступ до ресурсів безпосередньо кожного окремого комп'ютера. Під зручністю в першу чергу

розуміється висока реальна швидкість доступу, при якій обмін інформацією між додатками здійснюється непомітно для користувача.

З такого визначення виходить, що швидкість передачі по локальній мережі повинна обов'язково зростати в міру зростання швидкодії найпоширеніших комп'ютерів. Саме це й спостерігається: якщо ще порівняно недавно цілком прийнятною вважалася швидкість обміну в 1-10 Мбіт/с, то зараз середньошвидкісною вважається мережа, яка працює на швидкості 100 Мбіт/с та активно розроблюються засоби для швидкості 1000 Мбіт/с й навіть більше. При менших швидкостях передачі зв'язок стане вузьким місцем, буде надмірно сповільнювати роботу об'єднаних мережею комп'ютерів [3].

Таким чином, головна відмінність локальної мережі від будь-якої іншої - висока швидкість обміну. Але ця відмінність не є єдиною, не менш важливими є й інші фактори. Наприклад, принципово необхідним є низький рівень помилок передачі. Адже навіть дуже швидко передана, але спотворена помилками інформація безглузда - її потрібно буде передавати ще раз. Тому локальні мережі обов'язково використовують якісні лінії зв'язку, які прокладаються спеціально [4].

Принципове значення має й така характеристика мережі як можливість роботи з більшими навантаженнями, тобто з великою інтенсивністю обміну (або з більшим трафіком). Якщо механізм керування обміном, який використовується в мережі, не занадто ефективний, то комп'ютери можуть надмірно довго чекати своєї черги на передачу, й навіть якщо передача буде відбуватись потім на найвищій швидкості й повністю без помилок, то для користувача мережі це однаково обернеться неприйнятною затримкою доступу до всіх мережевих ресурсів [3,5].

Будь-який механізм керування обміном може гарантовано працювати тільки тоді, коли заздалегідь відома кількість комп'ютерів (абонентів, вузлів), які можуть бути підключені до мережі. При підключенні непередбаченої великої кількості абонентів, внаслідок перевантаження буде спостерігатися збій будь-якого пристрою або механізму. Нарешті, мережею можна назвати тільки таку систему передачі даних, що дозволяє з'єднувати хоча б декілька десятків комп'ютерів, але не два комп'ютери, як у випадку зв'язку через стандартні порти.

Таким чином, можна сформулювати наступні відмінні ознаки локальної мережі [2,3]:

- висока швидкість передачі, велика пропускна здатність;
- низький рівень помилок передачі (або, що є тим самим, високоякісні канали зв'язку). Припустима ймовірність помилок передачі даних повинна бути в межах $10^{-7} - 10^{-8}$;
- ефективний, швидкодіючий механізм керування обміном;
- обмежена, точно визначена кількість комп'ютерів, що підключаються до мережі.

За допомогою локальної мережі може передаватися різноманітна цифрова інформація: дані, зображення, телефонні розмови, електронні листи й т.п. Завдання передачі зображень, особливо повнокольорових динамічних зображень, пред'являє найвищі вимоги до швидкодії мережі. Найчастіше локальні мережі використовуються для розподілу (тобто спільного використання) таких ресурсів, як дисковий простір, принтери й вихід у глобальну мережу, але це всього лише незначна частина тих можливостей, які надають засоби локальних мереж. Наприклад, вони дозволяють здійснювати обмін інформацією між комп'ютерами різних типів. Абонентами (вузлами) мережі можуть бути не тільки комп'ютери, але й інші пристрої, наприклад принтери, плотери, сканери. Локальні мережі надають можливість організувати систему паралельних обчислень за допомогою усіх комп'ютерів мережі. Це дозволяє багаторазово прискорити вирішення складних математичних завдань. З їхньою допомогою можна також управляти роботою складної технологічної системи або дослідницького встаткування з декількох комп'ютерів одночасно [2,3].

В роботі розглядається один з компонентів комп'ютерної мережі – пристрій комутації на базі протоколу RIP. Цей комунікаційний пристрій регулює напрямки трафіку між кінцевими вузлами мережі.

У роботі необхідно розробити пристрій комутації для локальної мережі на базі протоколу RIP, яке має виконувати наступні функції:

- приймання та розподіл даних по портах;
- фільтрація пакетів;
- перевірка цілісності пакетів;
- перетворення мережевої адреси;
- ведення черг пакетів;
- передача пакетів у мережу.

При цьому необхідно використовувати мінімум апаратних та матеріальних ресурсів.

Для вирішення завдання необхідно розробити алгоритм роботи пристрою, а також схему електричну структурну, функціональну та принципову.

1 ОГЛЯД ЛІТЕРАТУРИ ТА ПОСТАНОВКА ЗАВДАННЯ ПРОЕКТУВАННЯ

1.1 Маршрутизатор та його функції

Комунікаційний пристрій для комп'ютерної мережі на базі протоколу RIP являє собою пристрій маршрутизації.

Маршрутизатори - це пристрій, який регулює напрямок трафіку між кінцевими вузлами мережі (хостами). Вони створюють маршрутні таблиці, які містять інформацію про всі можливі маршрути до всіх відомих вузлів [6,7].

Задача маршрутизації - це задача вибору оптимального шляху доставки пакета в комп'ютерній мережі, шляхом аналізу таблиць маршрутизації, розміщених у всіх маршрутизаторах і кінцевих вузлах мережі.

Основна функція маршрутизатора – це читання заголовків пакетів мережевих протоколів, що прийняті та буферизовані по кожному порту (наприклад, IPX, IP, AppleTalk або DECnet), та ухвалення рішення про подальший маршрут проходження пакета по його мережевій адресі, що включає, як правило, номер мережі та номер вузла [6].

Функції маршрутизатора можуть бути розбиті на три групи (рис. 1.1) [1].

Завдання маршрутизації зважається на основі аналізу таблиць маршрутизації, розміщених у всіх маршрутизаторах і кінцевих вузлах мережі. Основна робота зі створення таблиць маршрутизації виконується автоматично, але, як правило, передбачається й можливість вручну скорегувати або доповнити таблицю.

Для автоматичної побудови таблиць маршрутизації маршрутизатори обмінюються інформацією про топологію складеної мережі у відповідності зі спеціальним службовим протоколом – протоколом маршрутизації. Протоколи маршрутизації (наприклад, RIP, OSPF, NLSP) варто відрізнити від власне мережевих протоколів (наприклад, IP, IPX) [8].

Перші передають по мережі службову інформацію, другі призначені для передачі користувальницьких даних, як це роблять протоколи канального рівня.

Протоколи маршрутизації використовують мережеві протоколи як транспортний засіб.

Створення і ведення таблиць маршрутизації

Рівень
протоколу

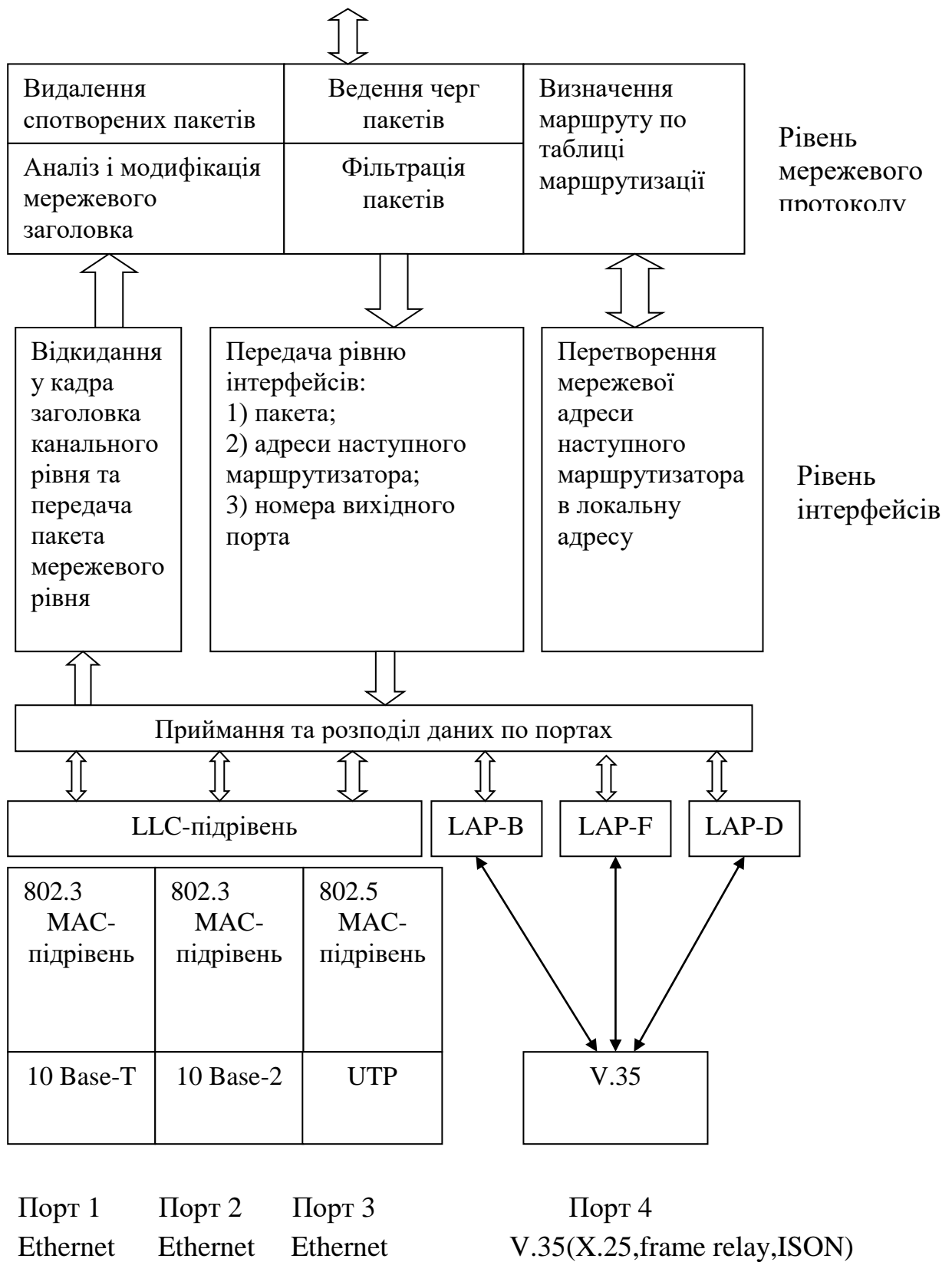


Рисунок 1.1 - Функціональна модель маршрутизатора

При обміні маршрутною інформацією пакети протоколу маршрутизації

містяться в поле даних пакетів мережевого рівня або навіть транспортного рівня, тому з погляду вкладеності пакетів протоколи маршрутизації формально варто було б віднести до більш високого рівня, чим мережевий [1].

У тому, що маршрутизатори для ухвалення рішення про просування пакета звертаються до адресних таблиць, можна побачити їх деяку подібність із мостами та комутаторами. Однак, природа використовуваних ними адресних таблиць сильно розрізняється. Замість MAC-адресів у таблицях маршрутизації вказуються номери мереж, які з'єднуються в інтермережу. Іншою відмінністю таблиць маршрутизації від адресних таблиць мостів є спосіб їхнього створення. У той час як міст будує таблицю, пасивно спостерігаючи за минаючими через нього інформаційними кадрами, що посилаються кінцевими вузлами мережі один одному, маршрутизатори зі своєї ініціативи обмінюються спеціальними службовими пакетами, повідомляючи сусідів про відомі їм мережах в інтермережі маршрутизатори та про зв'язки цих мереж з маршрутизаторами. Звичайно враховується не тільки топологія мережі, але й пропускна здатність та стан ліній зв'язку. Це дозволяє маршрутизаторам швидше адаптуватися до змін конфігурації мережі, а також правильно передавати пакети в мережах з довільною топологією, які допускають наявність замкнених контурів [1,8].

За допомогою протоколів маршрутизації маршрутизатори створюють карту зв'язків мережі того або іншого ступеня деталізації. На підставі цієї інформації для кожного номера мережі приймається рішення про те, якому наступному маршрутизатору треба передавати пакети, які направляються в цю мережу, щоб маршрут виявився раціональним. Результати цих рішень заносяться в таблицю маршрутизації. При зміні конфігурації мережі деякі записи в таблиці стають недійсними. У таких випадках пакети, що відправлені по помилкових маршрутах, можуть зациклюватися та губитися. Від того, наскільки швидко протокол маршрутизації приводить у відповідність вміст таблиці реальному стану мережі, залежить якість роботи мережі [8-10].

Протоколи маршрутизації можуть бути побудовані на базі різних алгоритмів, що відрізняються способами побудови таблиць маршрутизації, способами вибору найкоротшого маршруту та інших особливостей своєї роботи.

Маршрутизація виконується за розподіленою схемою - кожний маршрутизатор відповідальний за вибір тільки одного кроку маршруту, а остаточний маршрут складається в результаті роботи всіх маршрутизаторів, через які проходить

даний пакет. Такі алгоритми маршрутизації називаються однокроковими [1]. Існує й прямо протилежний, багатокроковий підхід - маршрутизація від джерела (Source Routing). Відповідно до нього вузол-джерело задає в пакеті, що відправляється в мережу, повний маршрут його проходження через усі проміжні маршрутизатори. При використанні багатокрокової маршрутизації немає необхідності створювати та аналізувати таблиці маршрутизації. Це прискорює проходження пакета по мережі, розвантажує маршрутизатори, але при цьому більше навантаження лягає на кінцеві вузли.

Однокрокові алгоритми залежно від способу формування таблиць маршрутизації діляться на три класи:

- алгоритми фіксованої (або статичної) маршрутизації;
- алгоритми простої маршрутизації;
- алгоритми адаптивної (або динамічної) маршрутизації.

В алгоритмах фіксованої маршрутизації всі записи в таблиці маршрутизації є статичними. Адміністратор мережі сам вирішує, на які маршрутизатори треба передавати пакети з тими або іншими адресами, й вручну (наприклад, за допомогою утиліти route ОС Unix або Windows NT) заносить відповідні записи в таблицю маршрутизації. Таблиця, як правило, створюється в процесі завантаження, надалі вона використовується без змін доти, поки її вміст не буде відредаговано вручну [1].

В алгоритмах простої маршрутизації таблиця маршрутизації або зовсім не використовується, або створюється без участі протоколів маршрутизації. Виділяють три типи простої маршрутизації [1]:

- випадкова маршрутизація, коли прибулий пакет посилається у випадковому напрямку, крім вихідного;
- лавинна маршрутизація, коли пакет ширококомовно посилається по всіх можливих напрямках, крім вихідного (аналогічно обробці мостами кадрів з невідомою адресою);
- маршрутизація по попередньому досвіду, коли вибір маршруту здійснюється по таблиці, але таблиця створюється за принципом мосту шляхом аналізу адресних полів пакетів, що з'являються на вхідних портах.

Найпоширенішими є алгоритми адаптивної (або динамічної) маршрутизації. Ці алгоритми забезпечують автоматичне відновлення таблиць маршрутизації після зміни конфігурації мережі. Протоколи, які побудовані на основі

адаптивних алгоритмів, дозволяють всім маршрутизаторам збирати інформацію про топологію зв'язків у мережі, оперативно відпрацьовуючи всі зміни конфігурації зв'язків. У таблицях маршрутизації при адаптивній маршрутизації звичайно є інформація про інтервал часу, протягом якого даний маршрут буде залишатися дійсним. Цей час називають часом життя маршруту (Time To Live, TTL) [1].

Адаптивні алгоритми звичайно мають розподілений характер, що виражається в тім, що в мережі відсутні які-небудь виділені маршрутизатори, які збирали б та узагальнювали топологічну інформацію: ця робота розподілена між всіма маршрутизаторами.

Адаптивні алгоритми маршрутизації повинні відповідати декільком важливим вимогам. По-перше, вони повинні забезпечувати, якщо не оптимальність, те хоча б раціональність маршруту. По-друге, алгоритми повинні бути досить простими, щоб при їхній реалізації не витрачалася занадто багато мережевих ресурсів, зокрема вони не повинні вимагати занадто великого обсягу обчислень або породжувати інтенсивний службовий трафік. І нарешті, алгоритми маршрутизації повинні мати властивість збіжності, тобто завжди приводити до однозначного результату за прийнятний час [11].

Адаптивні протоколи обміну маршрутною інформацією, які застосовуються в цей час в обчислювальних мережах, у свою чергу поділяються на дві групи, кожна з яких пов'язана з одним з наступних типів алгоритмів [1,12]:

- дистанційно-векторні алгоритми (Distance Vector Algorithms, DVA);
- алгоритми стану зв'язків (Link State Algorithms, LSA).

В алгоритмах дистанційно-векторного типу кожний маршрутизатор періодично й ширококомовно розсилає по мережі вектор, компонентами якого є відстані від даного маршрутизатора до усіх відомих йому мереж. Під відстанню звичайно розуміється число хопів (кроків). Можлива й інша метрика, що враховує не тільки число проміжних маршрутизаторів, але й час проходження пакетів по мережі між сусідніми маршрутизаторами. При одержанні вектору від сусіда маршрутизатор нарощує відстані до зазначених у векторі мереж на відстань до даного сусіда. Одержавши вектор від сусіднього маршрутизатора, кожний маршрутизатор додає до нього інформацію про відомі йому інші мережі, про які він довідався безпосередньо (якщо вони підключені до його портів) або з аналогічних оголошень інших маршрутизаторів, а потім знову розсилає нове значення вектору по мережі.

Зрештою, кожний маршрутизатор довідається інформацію про усі наявні в інтермережі мережі та про відстань до них через сусідні маршрутизатори [12]. Дистанційно-векторні алгоритми добре працюють тільки у невеликих мережах. У великих мережах вони засмічують лінії зв'язку інтенсивним широкомовним трафіком. До цього ж зміни конфігурації можуть оброблятися згідно цього алгоритму не завжди коректно, тому що маршрутизатори не мають точних відомостей про топологію зв'язків у мережі, а мають у своєму розпорядженні тільки узагальнену інформацію - вектор дистанцій, до того ж отриману через посередників. Робота маршрутизатора відповідно до дистанційно-векторного протоколу нагадує роботу моста, тому що точної топологічної картини мережі такий маршрутизатор не має [12].

Одним з поширених протоколів, заснованих на дистанційно-векторному алгоритмі, є протокол RIP, який розповсюджений у двох версіях - RIP IP, що працює із протоколом IP, та RIP IPX, що працює із протоколом IPX.

1.2 Дистанційно-векторний протокол RIP

Протокол RIP (Routing Information Protocol) являє собою один з найстарших протоколів обміну маршрутною інформацією, однак він дотепер надзвичайно розповсюджений в обчислювальних мережах. Крім версії RIP для мереж TCP/IP, існує також версія RIP для мереж IPX/SPX компанії Novell [1].

У цьому протоколі всі мережі мають номери (спосіб утворення номера залежить від використовуваного в мережі протоколу мережевого рівня), а всі маршрутизатори - ідентифікатори. Протокол RIP широко використовує поняття "вектор відстаней". Вектор відстаней являє собою набір пар чисел, що є номерами мереж та відстанями до них у хопах [11].

Вектори відстаней ітераційно поширюються маршрутизаторами по мережі й через кілька кроків кожний маршрутизатор має дані про досяжні для нього мережі та про відстані до них. Якщо зв'язок з якою-небудь мережею обривається, то маршрутизатор відзначає цей факт тим, що привласнює елементу вектору, який відповідає відстані до цієї мережі, максимально можливе значення, що має спеціальний сенс - "зв'язку немає". Таким значенням у протоколі RIP є число 16.

На рисунку 1.2 наведений приклад мережі, яка складається з шести маршрутизаторів, що мають ідентифікатори від 1 до 6, та з шести мереж від А до F, утворених прямими зв'язками типу "крапка-крапка" [1].

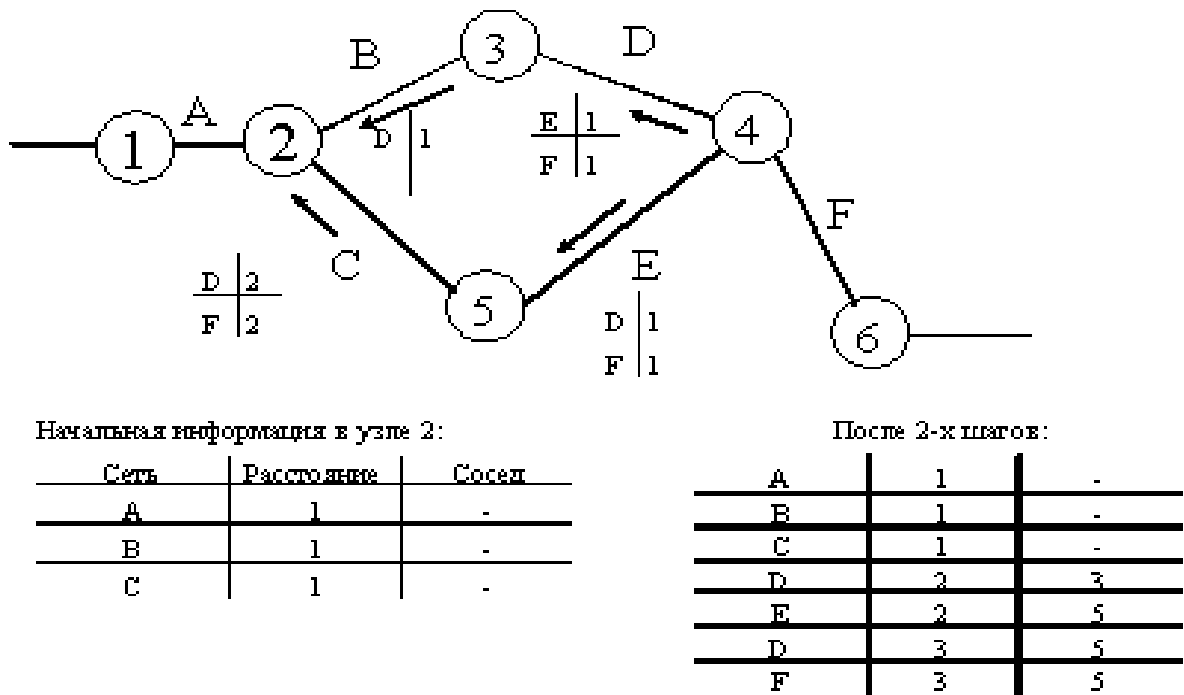


Рисунок 1.2 - Обмін маршрутною інформацією із протоколу RIP

На рисунку наведена початкова інформація, що втримується в топологічній базі маршрутизатора 2, а також інформація в цій же базі після двох ітерацій обміну маршрутними пакетами протоколу RIP. Після певного числа ітерацій маршрутизатор 2 буде знати про відстані до всіх мереж інтермережі, причому в нього може бути кілька альтернативних варіантів відправлення пакета до мережі призначення. Нехай у нашій прикладі мережею призначення є мережа D [11].

При необхідності відправити пакет у мережу D маршрутизатор переглядає свою базу даних маршрутів та вибирає порт, який має найменшу відстань до мережі призначення (у цьому випадку порт, що зв'язує його з маршрутизатором 3). Для адаптації до зміни стану зв'язків та встаткування з кожним записом таблиці маршрутизації зв'язаний таймер. Якщо за час тайм-ауту не прийде нове повідомлення, що підтверджує цей маршрут, то він видаляється з маршрутної таблиці.

При використанні протоколу RIP працює евристичний алгоритм динамічного програмування Беллмана-Форда, й рішення, яке знайдене за його допомогою, є не оптимальним, а є близьким до оптимального. Перевагою протоколу RIP є його обчислювальна простота, а недоліками - збільшення трафіка при періодичному розсиланні ширококомовних пакетів та неоптимальність знайденого маршруту [11,12].

На рисунку 1.3 показаний випадок нестійкої роботи мережі згідно протоколу RIP при зміні конфігурації - відмові лінії зв'язку маршрутизатора M1 з мережею 1 [1].

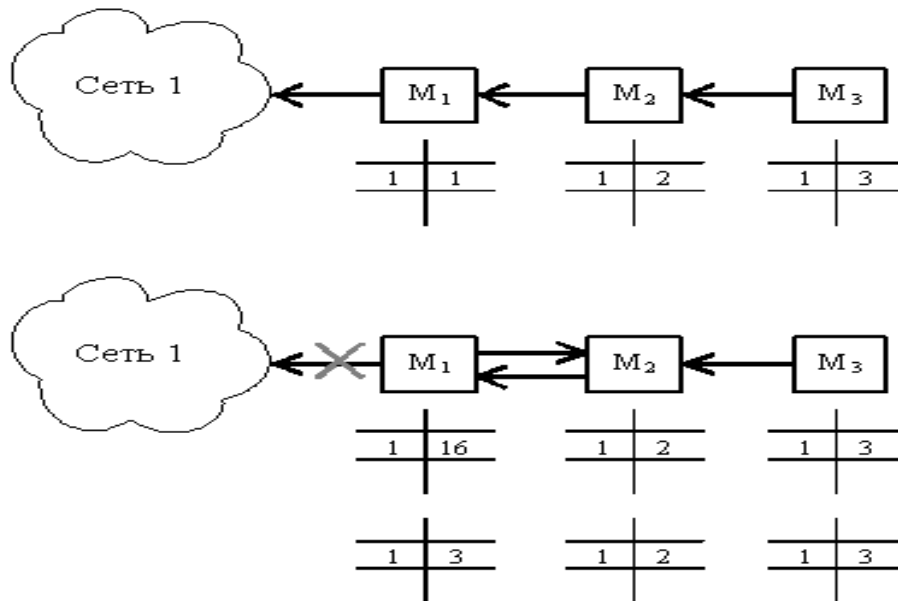


Рисунок 1.3 - Приклади нестійкої роботи мережі при використанні протоколу RIP

При обриві зв'язку з мережею 1 маршрутизатор M1 відзначає, що відстань до цієї мережі прийняла значення 16. Однак, одержавши через певний час від маршрутизатора M2 маршрутне повідомлення про те, що від нього до мережі 1 відстань становить 2 хопи, маршрутизатор M1 нарощує цю відстань на 1 й відзначає, що мережа 1 є досяжною через маршрутизатор 2. В результаті пакет, призначений для мережі 1, буде циркулювати між маршрутизаторами M1 та M2 доти, поки не мине час зберігання запису про мережу 1 у маршрутизаторі 2, й він не передасть цю інформацію маршрутизатору M1. Для виключення подібних ситуацій маршрутна інформація про відомий

маршрутизатору мережі не передається тому маршрутизатору, від якого прийшла [1,12].

2 РОЗРОБКА АЛГОРИТМУ ФУНКЦІОНУВАННЯ ТА СТРУКТУРНОЇ СХЕМИ МАРШРУТИЗАТОРА

2.1 Протокол RIP. Алгоритм функціонування

У роботі проектується пристрій на базі протоколу обміну інформацією про маршрутизацію RIP.

RIP повідомлення передаються в UDP датаграмах, як показано на рисунку 2.1 [11,12].



Рисунок 2.1 - Інкапсуляція RIP повідомлення в UDP датаграму

На рисунку 2.2 показаний формат RIP повідомлення, разом з адресами [1,11,12].

IP-

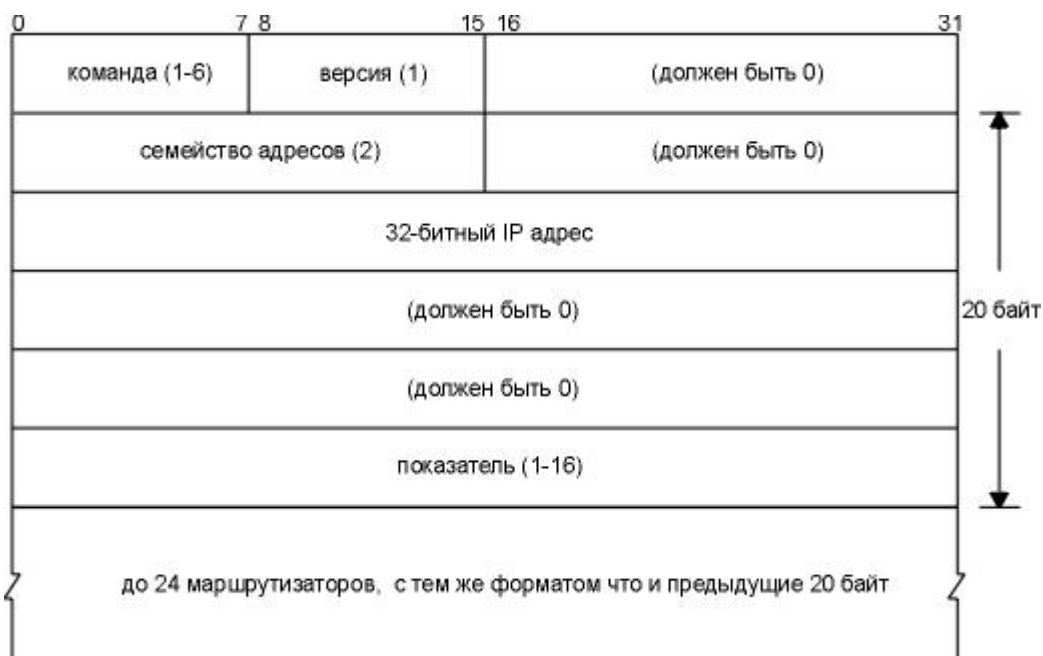


Рисунок 2.2 - Формат RIP повідомлення

Якщо поле «Команда» приймає значення 1 - це запит, якщо 2 - відгук. Існують

ще два значення поля команди (3 та 4), а також два недокументованих значення: опитування (5) та пункт опитування (6). У запиті знаходиться вимога до іншої системи надіслати усю або частину її таблиці маршрутизації. У відгуку втримується уся таблиця маршрутизації відправника або її частина. Поле «Версія» звичайно встановлене в 1. Наступні 20 байт містять: сімейство адрес (яке завжди дорівнює 2 для IP адрес), IP-адресу та відповідний показник. У ролі показника RIP виступає лічильник пересилань [11].

У RIP повідомленні може бути оголошено до 25 маршрутизаторів. Обмеження в 25 визначається повним розміром RIP повідомлення, $20 \times 25 + 4 = 504$, менше ніж 512 байт. Через обмеження в 25 маршрутизаторів, на один запит, як правило, потрібно надіслати кілька відгуків, щоб передати всю таблицю маршрутизації.

При використанні протоколу RIP працює евристичний алгоритм динамічного програмування Беллмана-Форда, й рішення, яке знайдене за його допомогою, є не оптимальним, а близьким до оптимального. Перевагою протоколу RIP є його обчислювальна простота, а недоліками - збільшення трафіка при періодичному розсиланні ширококомовних пакетів та неоптимальність знайденого маршруту [11].

Розглянемо, як працює маршрутизатор з використанням протоколу RIP. Номер зарезервованого порту для RIP - UDP - порт 520.

Ініціалізація. Коли домен стартує, він визначає усі активовані інтерфейси й посилає пакет із запитом на кожний інтерфейс, з вимогою до вилучених маршрутизаторів послати повні таблиці маршрутизації. У випадку каналу крапка-крапка цей запит відправляється на інший кінець каналу. Запит розсилається ширококомовними повідомленнями, якщо мережа їх підтримує. Порт призначення - UDP порт 520 (домен маршрутизації на іншому маршрутизаторі). Характеристики подібного запиту наступні: поле команди встановлене в 1, поле сімейство адрес установлене в 0 й показник встановлений в 16. Подібний формат відповідає спеціальному запиту, у відповідь на який потрібно надіслати повну таблицю маршрутизації [11].

Запит прийнятий. У випадку спеціального запиту, який ми тільки що описали, маршрутизатору, що запитує, відправляється повна таблиця маршрутизації. Інакше обробляється кожний пункт у запиті: якщо є присутнім маршрут на зазначену адресу, показник встановлюється в певне значення, інакше показник встановлюється в 16. Показник, встановлений в 16, - це спеціальне значення,

що означає "нескінчене" (infinity) й повідомляє, що маршруту до цього пункту призначення не існує. Повертається відповідь.

Відповідь прийнята. Якщо відповідь визнана коректною таблиця маршрутизації може бути оновлена. Можуть бути додані нові записи. Записи, які існують можуть бути модифіковані або вилучені.

Регулярне відновлення маршрутизації. Кожні 30 секунд вся таблиця маршрутизації або її частина відправляється кожному сусідньому маршрутизатору. Таблиця маршрутизації поширюється широкошовними повідомленнями (у випадку Ethernet) або відправляється на інший кінець каналу «крапка-крапка» [1].

Незаплановане відновлення. Відбувається у тому випадку коли змінюється показник маршруту. У цьому випадку немає необхідності посилати таблицю маршрутизації цілком, передається тільки той запис, що був змінений.

З кожним маршрутом зв'язаний тайм-аут. Якщо система, що використовує RIP, визначила, що маршрут не був оновлений протягом трьох хвилин, показник маршруту встановлюється в стан "нескінчене" (16) й позначається для видалення. Це означає, що було пропущено шість 30-секундних оновлень від маршрутизатора, який оголосив маршрут. Однак, видалення маршруту з локальної таблиці маршрутизації відкладається ще на 60 секунд, щоб переконатися що маршрут дійсно зник.

В якості показника у RIP використовується лічильник пересилань. Для усіх безпосередньо підключених інтерфейсів лічильник пересилань дорівнює 1. Розглянемо маршрутизатори та мережі, які наведені на рисунку 2.3 [12].

Чотири пунктирні лінії показують широкошовні повідомлення RIP. Маршрутизатор R1 оголошує маршрут до N2 з лічильником пересилань, що дорівнює 1, пославши широкошовне повідомлення на N1. (Безглуздо повідомляти маршрут до N1 у широкошовному повідомленні, переданому на N1.) Він також оголошує маршрут до N1 з лічильником пересилань рівним 1, надіславши широкошовне повідомлення на N2. Так саме, R2 оголошує маршрут до N2 з показником 1 та маршрут до N3 з показником 1.

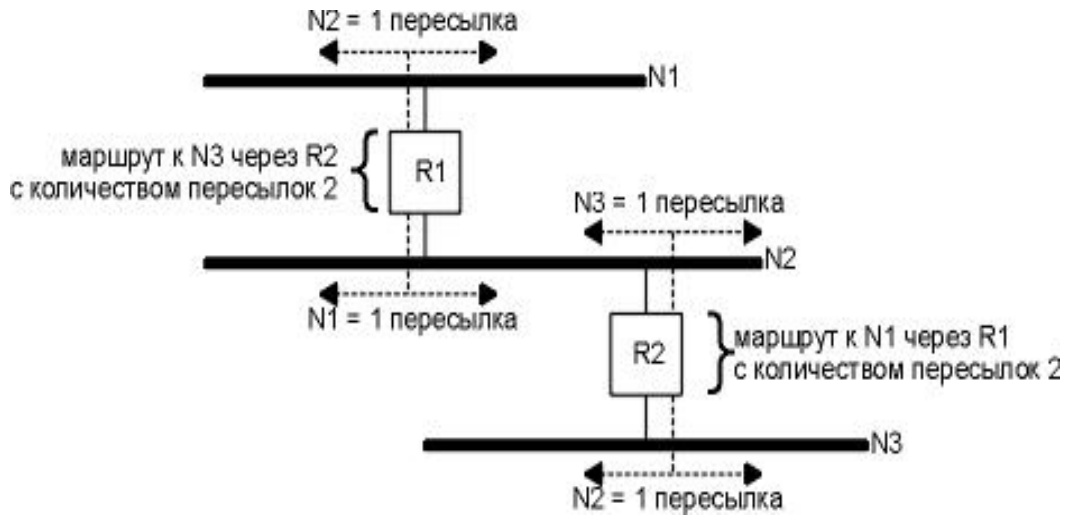


Рисунок 2.3 - Приклад маршрутизаторів та мереж

Якщо суміжний маршрутизатор оголосив маршрут до вилученої мережі з лічильником пересилань рівним 1, то для нас показник до цієї мережі буде дорівнює 2, так пакет необхідно послати спочатку на наш маршрутизатор, щоб одержати доступ до мережі. У прикладі, наведеному вище, показник до N1 для R2 дорівнює 2, так само як й показник до N3 для R1.

Тому що кожний маршрутизатор надсилає свої таблиці маршрутизації сусідам, визначається кожна мережа в кожній автономній системі (AS). Якщо усередині AS існує кілька шляхів від маршрутизатора до мережі, маршрутизатор вибирає шлях з найменшою кількістю пересилань та ігнорує інші шляхи [11,12].

Величина лічильника пересилань обмежена значенням 15, що означає, що RIP може бути використаний лише усередині AS, де максимальна кількість пересилань між хостами становить 15. Спеціальне значення показника, який дорівнює 16, вказує на те, що на дану IP-адресу не існує маршруту.

Як би просто це не звучало, усе ж існують проблеми. RIP не має уяви про розподіл на підмережі [12].

Блок-схема алгоритму маршрутизації наведена на рисунку 2.4.

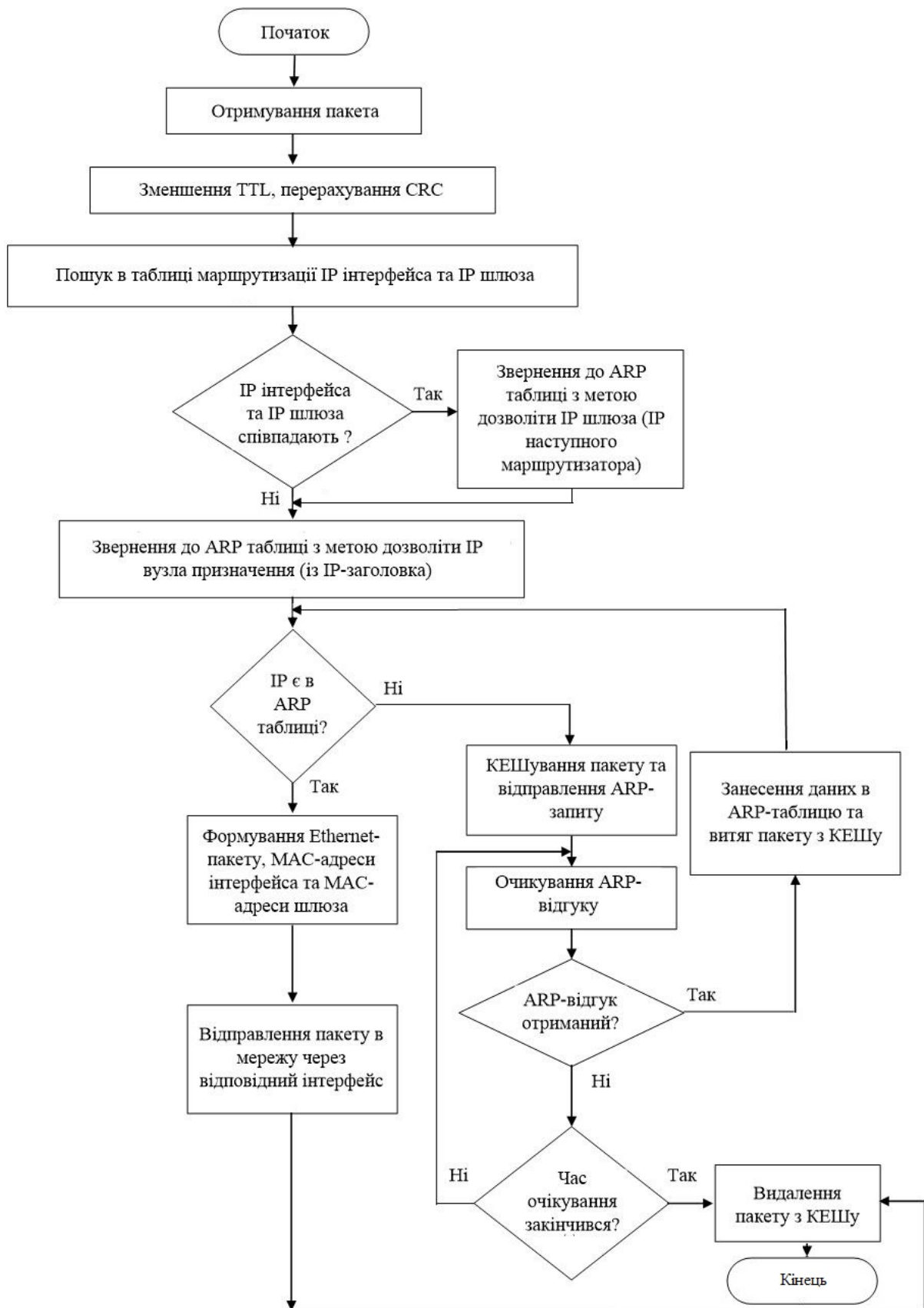


Рисунок 2.4 – Блок-схема алгоритму маршрутизації

2.2 Структурна схема пристрою комутації

Виходячи із структури мережі, алгоритму RIP та функцій маршрутизації пристрій комутації має наступний вигляд (рис. 2.5).

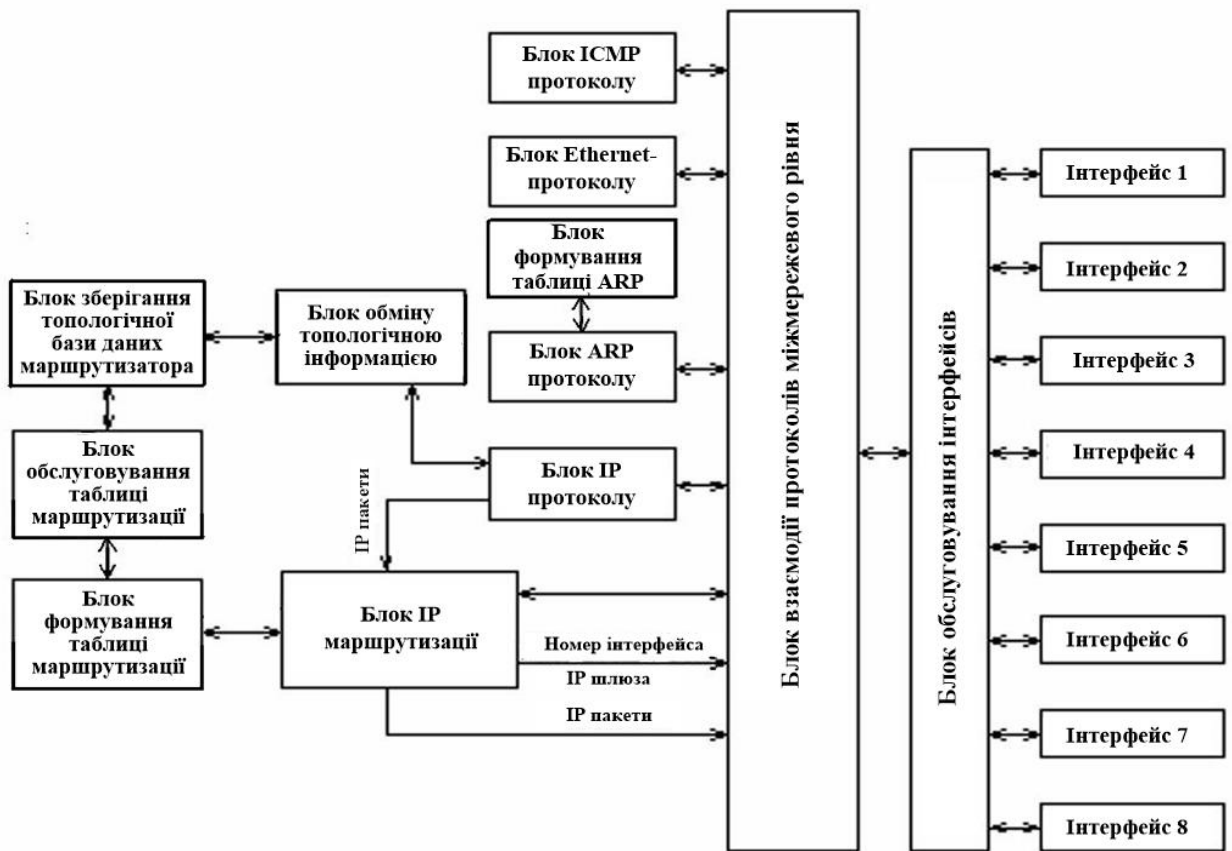


Рисунок 2.5 - Структурна схема маршрутизатора

Блок IP-протоколу виконує такі функції:

- переспрямування пакетів при маршрутизації;
- перевірка CRC заголовка (циклічного надлишкового коду) та вирішення питання ушкоджений пакет чи ні. Якщо пакет ушкоджений - він відкидається;
- зміна TTL (час життя), після кожного маршрутизатора його значення зменшується на одиницю, якщо воно стане рівним нулю - пакет знищується;
- перерахування CRC заголовка;
- забезпечення роботи OSPF протоколу.

Блок формування таблиці ARP (Address Resolution Protocol):

- зберігання бази даних, що зв'язує IP-адреси з відповідними їм MAC-адресами.

Блок ARP протоколу:

- оброблення звертань до ARP-таблиці, дозволяє локальну IP-адресу (адресу, яка належить підмережі, до якої належить даний інтерфейс). Процес дозволу локальної IP-адреси полягає в наданні блоку взаємодії протоколів міжмережевого рівня MAC-адреси, що відповідає IP-адресі;
- знищує застарілі записи після закінчення тайм-аута;
- при спробі дозволу IP-адреси, відсутньої у ARP таблиці, надсилає широкомовні ARP-запити, а пакет, який очікує необхідну йому MAC-адресу, розміщується в КЕШ, якщо приходить ARP-відповідь IP та відповідна йому MAC-адреса розміщується в таблицю;
- обробляє ARP-запити, й поміщає в таблицю IP-адресу й відповідну йому MAC-адресу хоста або маршрутизатора, який надіслав запит.

Блок Ethernet протоколу:

- формує для пакетів, що надійшли, Ethernet заголовок, який містить MAC-адресу одержувача, MAC-адресу відправника, тип наступного протоколу (поле має це значення тільки для Ethernet мережі й повинне перебувати в діапазоні від 0x05DC до 0xFFFF), а також звичайно підраховує CRC пакета (циклічного надлишкового коду) та розміщує його в кінець, але в нашому випадку CRC підраховує Ethernet контролер.

Блок обслуговування інтерфейсів:

- управляє низько рівневим доступом до Ethernet контролера та забезпечує механізм передачі та приймання пакетів. Ethernet контролер має реєстри для прямої та непрямой адресації, доступ до реєстрів для прямої адресації здійснюється наступним методом. Мікроконтролер виставляє на шину адреси – адресу, молодший байт якої управляє вибором необхідного Ethernet контролера (інтерфейсу), а старший байт якої являє собою номер прямого реєстра, до якого формується звернення, на шину даних виставляються дані, які необхідно записати в прямий реєстр, або Ethernet контролер сам виставить дані при операції читання. Також обслуговує переривання, які надходять від відповідного Ethernet контролера. Інформація про переривання, що виникле, втримується в спеціальних реєстрах. Прямі реєстри у свою чергу надають доступ до реєстрів статусу й контролю з непрямой адресацією, області пам'яті буфера передачі, приймання та Flash пам'яті.

Блок взаємодії протоколів міжмережевого рівня:

– блок, який здійснює взаємодію ARP, Ethernet, IP, ICMP, IGMP протоколів як між собою так й з драйвером інтерфейсів. Наприклад: передачу даних між цими протоколами та драйвером інтерфейсів; синхронізацію їхньої роботи. А також здійснює формування запитів на дозвіл IP-адреси, тимчасове їх кешування та знищення IP-пакетів у випадку неможливості дозволити IP-адресу, виконує оброблення даних, отриманих від блоку маршрутизації, та задіє в певному порядку протоколи для переспрямування пакетів, а також формування в кінці кінців Ethernet-пакетів, які передаються драйверу інтерфейсів.

Блок маршрутизації:

- служить для визначення згідно таблиці маршрутизації інтерфейсу (через який піде пакет у мережу) та IP-адреси наступного одержувача переспрямованого Ethernet-пакета (маршрутизатора або хоста одержувача);
- визначає чи є одержувач локальним хостом мережі, підключеної до одного з інтерфейсів, або пакет потрібно перенаправляти на наступний маршрутизатор;
- віддає блоку взаємодії протоколів міжмережевого рівня необхідну інформацію (IP-адресу інтерфейсу та IP-адресу одержувача) для формування Ethernet-пакету MAC-адресою одержувача, отриманою після дозволу IP, та MAC-адресою інтерфейса, який відправив пакет;
- обробляє маршрути, що накладаються;
- здійснює підтримку множинних маршрутів, обсяг трафіку, який передається по кожному з маршрутів, зворотно пропорційно відношенню метрик між маршрутами.

Блок формування таблиці маршрутизації:

- зберігає базу даних, необхідних для ухвалення рішення блоком маршрутизації по якому з маршрутів відправити прийняті пакети. У загальному випадку таблиця являє собою рядки з полями мережевої адреси та маскою мережі, які в парі формують деякий діапазон IP-адрес. Блок маршрутизації здійснює виявлення приналежності IP-адреси призначення (утримується в заголовку IP-пакета) одному з діапазонів, наведених у таблиці маршрутизації. Кожному діапазону ставиться у відповідність адреса шлюзу, адреса інтерфейсу та метрика маршруту. У випадку підтримки множинних маршрутів, одному діапазону може відповідати кілька записів адрес шлюзу, адрес інтерфейсів та метрик. Збігання адреси шлюзу з адресою інтерфейсу

означає приналежність IP-адреси призначення хосту, який належить до мережі, до якої підключений даний інтерфейс.

Блок зберігання топологічної бази даних маршрутизатора:

– база даних стану зв'язків, що представляє собою повний опис графа системи. При цьому вершинами графа є маршрутизатори (які можуть бути приграничними маршрутизаторами області, приграничними маршрутизаторами системи або кінцевою крапкою віртуального зв'язку), транзитні мережі, тупикові мережі, а ребрами - з'єднуючі їх зв'язки. Бази даних на усіх маршрутизаторах області ідентичні.

Блок обміну топологічною інформацією:

– служить для побудови топологічної бази даних маршрутизатора. Блок працює на RIP протоколі, який знаходиться безпосередньо над IP. Тим часом RIP протокол містить у собі наступні протоколи: Hello-протокол, протокол обміну (Exchange protocol), протокол заповнення (flooding). По вище описаних протоколах цей блок займається обміном топологічною інформацією між сусідніми маршрутизаторами, причому тільки тої, якої немає в його сусіда, тим самим синхронізуючи свої бази даних. Це відбувається за допомогою первісного обміну інформацією описового характеру про свою базу даних. Також цей блок займається цілим рядом заходів організаційного характеру. Наприклад, виявлення сусідів та встановлення з ними відносин суміжності, вибір виділеного та запасного виділеного маршрутизатора, у випадку неполадок створення віртуальних зв'язків, швидке реагування на зміну стану зв'язку за допомогою протоколу заповнення.

Блок обслуговування таблиці маршрутизації:

- формує таблицю маршрутизації на підставі найкоротших шляхів, обчислених за допомогою алгоритму Беллмана-Форда по топологічній базі даних маршрутизатора;
- обробляє звертання до таблиці маршрутизації;
- знищує застарілі записи після закінчення тайм-аута.

Блок ICMP (Internet Control Message Protocol) протоколу:

– виконує функцію сповіщення про помилки та управляє повідомленнями для протоколу IP. Наприклад, перевантажений маршрутизатор може послати пропозицію знизити швидкість передачі.

3 РОЗРОБКА СХЕМИ ЕЛЕКТРИЧНОЇ ФУНКЦІОНАЛЬНОЇ ПРИБРОЮ КОМУТАЦІЇ НА БАЗІ ПРОТОКОЛУ ІР

Проаналізувавши алгоритм функціонування пристрою та структурну схему пристрій маршрутизації повинен містити в собі:

- центральний процесорний модуль, що виконує основне завдання маршрутизації відповідно до програми, яка зашита в нього, та реалізований на базі мікроконтролера;
- Ethernet контролер, що здійснює взаємини між протоколами. А також здійснює формування запитів на дозвіл ІР-адреси, тимчасове їх кешування та знищення ІР-пакетів у випадку неможливості дозволити ІР-адресу, виконує обробку даних, отриманих від центрального процесорного модуля, та задіє в певному порядку протоколи для переспрямування пакетів;
- блок узгодження та гальванічної розв'язки. Цей блок має стандартний порт RJ-45 для підключення мережевого кабелю та виконує функції приймання-передачі пакета, а також індикації та гальванічної розв'язки.

На рисунку 3.1 зображена функціональна схема маршрутизатора.

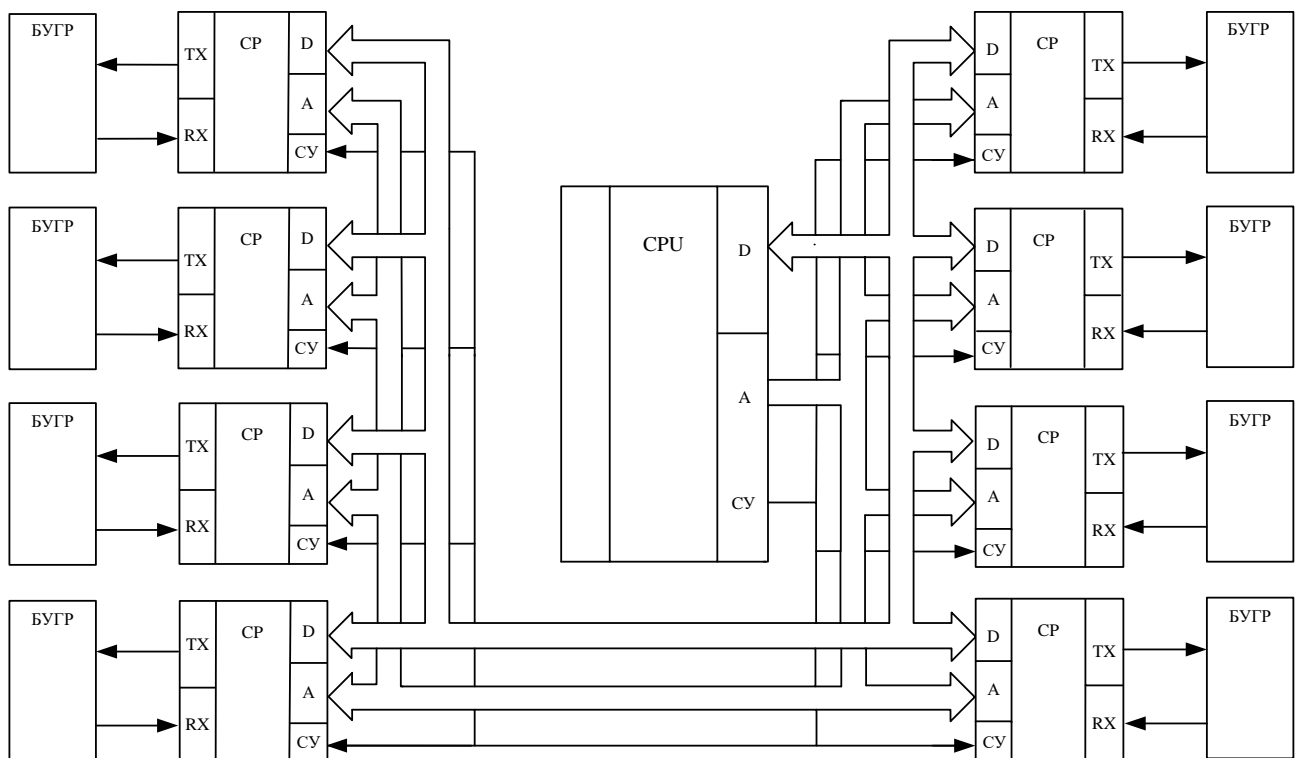


Рисунок 3.1 - Функціональна схема маршрутизатора

На рисунку зображені:

CPU - центральний процесорний модуль;

CP – Ethernet-контролер;

БУГР - блок узгодження та гальванічної розв'язки;

D - шина даних;

A - шина адреси;

SU - система управління (керування);

RX - буфер приймання інформації;

TX - буфер передачі інформації.

Детальний опис та розрахунок блоків, зображених на функціональній схемі маршрутизатора будуть проводитися далі.

4 РОЗРОБКА ТА РОЗРАХУНОК ПРИНЦИПОВИХ ЕЛЕКТРИЧНИХ СХЕМ ВУЗЛІВ ТА БЛОКІВ ПРИСТРОЮ

4.1 Мікроконтролер 38051F300

Спроекуємо пристрій комутації для локальної мережі на базі протоколу RIP із застосуванням мікроконтролера C8051F300, Ethernet-контролера CP2200, а також блоку узгодження та гальванічної розв'язки. Мікроконтролер C8051F300 працює на частотах до 25 МГц [13].

Аналогова периферія мікроконтролера містить швидкодіючий 8-бітний аналого-цифровий перетворювач із восьмиканальним мультиплексором. Крім того, він містить один аналоговий компаратор. Іншою особливістю є те, що входи аналогових вузлів не мають самостійних виводів корпусу та можуть бути налаштовані на кожній з виводів єдиного 8-бітного порту введення/виводу [13].

Цифрова периферія має усі вузли стандартного мікроконтролера 8052, а також програмувальний лічильник масив, розширений оброблювач переривань, охоронний таймер та монітор живлення, а також вбудований програмувальний генератор з підтримкою UART. У набір вбудованих апаратних інтерфейсів входять SMBus, сумісний з PC) та UART. Інтерфейс SPI відсутній [14].

Мікроконтролер C8051F300 має вбудовану Flash-пам'ять середнього обсягу 8К, яка дозволяє розмістити в ній програмні компоненти середньої складності. Перелічені достоїнства дозволяють зробити висновок, що мікроконтролер C8051F300 є незамінним при створенні мікроконтролерних систем виміру та контролю.

Узагальнена структура мікроконтролера C8051F300 наведена на рисунку 4.1. Вона складається з [13]:

- аналогової периферії;
- цифрової периферії;
- високопродуктивного контролерного ядра.

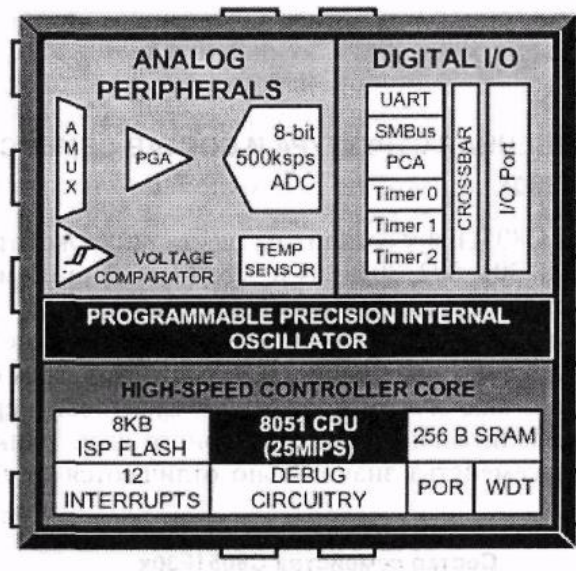


Рисунок 4.1 - Узагальнена структура мікроконтролера C8051F300

До складу аналогової периферії входять:

- швидкодіючий (до 500 ksps) 8-бітний аналого-цифровий перетворювач;
- вхідний аналоговий мультиплексор забезпечує комутацію до 8 зовнішніх входів;
- включений між аналоговим мультиплексором та аналогово-цифровим перетворювачем програмувальний попередній підсилювач із коефіцієнтами підсилення 4; 2; 1; та 0,5;
- аналогові вузли в якості опорної напруги можуть використовувати або зовнішнє джерело, що підключається через спеціальний вхід, або напругу живлення;
- до дев'ятого (внутрішнього) входу аналогового мультиплексора підключений вбудований датчик температури. До складу аналогової периферії також входить один аналоговий компаратор.

До складу цифрової периферії та ядра входять [13]:

- багатофункціональний послідовний порт UART;
- послідовний інтерфейс SMBus, сумісний з PC;
- три 16-розрядних таймери загального призначення;
- програмувальний масив-лічильник PCA із трьома модулями захоплення/порівняння та режимом таймера реального часу;
- вбудований тактовий генератор 24,5 Мгц з підтримкою UART;
- зовнішній генератор з можливістю роботи від кварцового або пьезокерамічного резонаторів, RC-ланцюжка або окремого конденсатора;

– один порт введення/виводу.

Мікроконтролер має вбудовану RAM обсягом 256 байт та Flash-пам'ять програм/даних обсягом 8Кб.

Мікроконтролер містить оригінальне ядро CIP-51, яке забезпечує пікову продуктивність до 25 MIPS при тактовій частоті 25 МГц, 70% інструкцій виконуються за 1-2 періоди тактової частоти [13].

Flash-пам'ять мікроконтролера програмується внутрисистемно через вбудований інтерфейс JTAG (I²C).

У складі цифрової периферії немає охоронного таймера WDT.

Мікроконтролер працює при напрузі живлення від 2,7 до 3,6 В (при типовому струмі споживання 5,8 мА) в індустріальному діапазоні температур від -45 до +85 °С. Лінії портів введення/виводу, скидання та JTAG працездатні при напрузі живлення 5 В [13].

Мікроконтролер C8051F300P випускається в корпусі DIP14 з 14-ю виводами (рис. 4.2). Функціональна схема мікроконтролера C8051F300 (з аналого-цифровим перетворювачем) показана на рис. 4.3 [13].

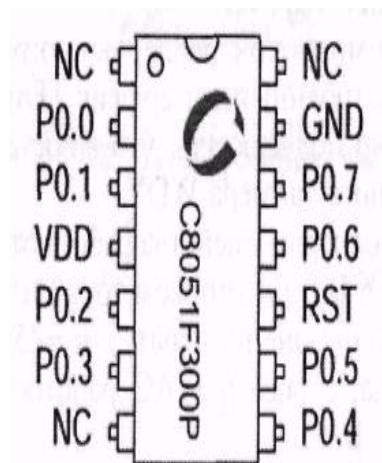


Рисунок 4.2 - Розташування виводів мікроконтролера C8051F300

Продовження таблиці 4.1

1	2	3	4	5
P0.3			D I/O	Лінія порту P0.3
P0.4	6	8	D I/O A In	Лінія порту P0.4
P0.5	7	9	D I/O A In	Лінія порту P0.5
C2CK	8	10	D I/O	Тактовий сигнал C2 відладочного інтерфейсу
RST			D I/O	Зовнішній вхід скидання або вхід/вихід з відкритим джерелом внутрішнього монітора
P0.6	9	11	D I/O A In	Лінія порту P0.6
CNVS TR			D I/O	Вхід зовнішнього запуску ADC
C2D	10	12	D I/O	Двонаправлена лінія даних C2 відладочного інтерфейсу
P0.7			D I/O A In	Лінія порту P0.7
GND	11	13		Загальний

Електричні параметри та граничні режими експлуатації наведені в таблиці 4.3 [13].

Перевищення параметрів, зазначених у таблиці, може привести до ушкодження виробу. Не рекомендується експлуатація виробу в граничних режимах, тому що це призводить до зниження надійності та ресурсу.

Підсистема мікроконтролера C8051F300. Мікроконтролер C8051F300 має типове ядро CIP-51 фірми SiLabs з підсистемою налагодження й програмування 32 та набором інструкцій. Ядро оснащено вбудованою пам'яттю даних з довільним доступом RAM обсягом 256 байт (0x00-0xFF).

Таблиця 4.2 - Електричні характеристики мікроконтролера C8051F300

Параметр	Умови	Мін	Норм	Макс
Напруга живлення аналогової частини	Напруга живлення аналогової частини повинна бути більше 1 В для роботи супервізора живлення	2,7	3,0	3,6
Струм споживання з активною аналоговою частиною, мА	ADC та компаратори включені при тактовій частоті 25 МГц		5,8	
Струм споживання цифрової частини при виключеному CPU, мА	VDD = 2,7В, Clock = 25МГц VDD = 2,7В, Clock = 1МГц VDD = 2,7В, Clock = 32МГц		2,1 83 мкА 2,8мкА	
Струм споживання цифрової частини в пасивному режимі, мА	Генератор виключений		< 0,1	
Напруги збереження даних в RAM, В			1,5	
Робочий температурний діапазон, С		-40		+85

Таблиця 4.3 - Граничні параметри мікроконтролера C8051F300

Гранична температура корпусу, С	-55...125
Гранична температура зберігання, С	-65...150
Граничні напруги на усіх виводах Port I/O та RST/ стосовно DGND, В	-0,3...5,8
Гранична напруга на виводі VDD стосовно к GND, В Максимальний загальний струм через VDD та GND, мА	0,3...4,2 500
Максимальний вихідний ток через будь-який з виводів Port I/O, мА	100

Молодші 128 байт (0x00-0x7F) доступні інструкціям з прямою та непрямою адресацією, регістри спеціальних функцій SFR доступні тільки інструкціям з прямою адресацією, а старші 128 байт (0x 80-0xFF) – тільки інструкціям з непрямою адресацією. Перші 32 байта (0x00- 0x1F) адресуються як чотири банки регістрів загального призначення, а наступний 16 байт (0x 20-0x2F) мають бітову адресацію. Карта пам'яті мікроконтролера C8051F300 показана на рисунку 4.4 [13].

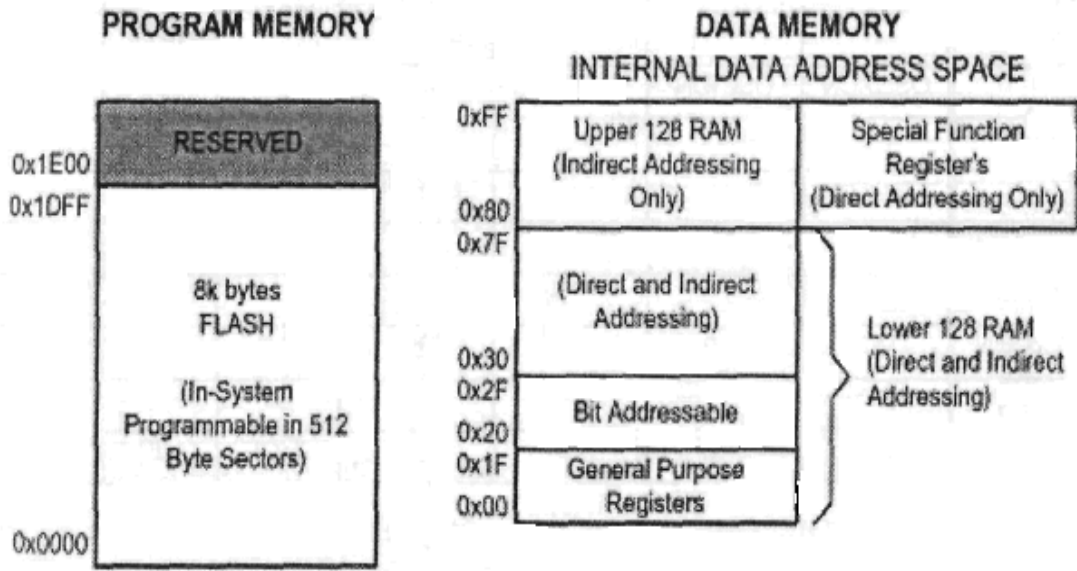


Рисунок 4.4 - Карта пам'яті сімейства C8051F300

Підсистема регістрів спеціальних функцій SFR. Адресний простір пам'яті даних 0x 80-0xFF з прямою адресацією в стандартному 8051 мікроконтролері зайнятий регістрами спеціальних функцій SFRs. За допомогою цих регістрів здійснюється керування та обмін даними між ресурсами ядра CIP-51 та периферією. Регістри спеціальних функцій ядра CIP-51 з однієї сторони відповідають регістрам стандартного 8051, а з іншого боку – доповнені можливостями конфігурування та обміну даними з оригінальними підсистемами мікроконтролерів SiLabs [13]. Оскільки мікроконтролер, який описується, має «полегшений» набір периферійних пристроїв, він має односторінковий SFR-простір адресів.

Таблиця 4.4 - Адреси регістрів спеціальних функцій мікроконтролера

Назва регістра	Адреса регістра	Опис функціонального призначення регістра	Розділ опису
ACC	0xE0	Акумулятор	3.6.24
ADC0CF	0XBC	Конфігурація ADC0	8.6.3
ADC0CN	0xE8	Управління ADC0	8.6.4
ADC0GT	0xC4	Байт верхнього порогу даних ADC0	3.6.8
ADC0LT	0xC6	Байт нижнього порогу даних ADC0	3.6.10
ADC0	0xBE	Байт даних ADC0	3.6.6
AMX0SL	0xBB	Вибір каналів мультиплексора MUX ADC0	8.6.2
B	0xFO	Регистр B	3.6.25
CKCON	0x8E	Регистр управління тактовою частотою таймерів	8.6.28

Особливості аналого-цифрового перетворювача. Аналого-цифровий перетворювач мікроконтролера C8051F300 має відмінності від аналогічних вузлів, що застосовуються в інших мікроконтролерах. У першу чергу це пов'язане з обмеженою кількістю виводів корпусу та необхідністю в цих умовах забезпечити й однополярний, й диференціальний режими. З цією метою у вузлі аналого-цифрового перетворювача була змінена структура аналогового мультиплексора. Фактично, у цьому мікроконтролері використовуються два мультиплексора, керованих одним регістром SFR, кожний з яких комутує свій вхід попереднього підсилювача PGA. Функціональна схема вузла аналого-цифрового перетворювача наведена на рисунку 4.5 [13].

Перша частина мультиплексора комутує джерело сигналу, а друга – крапку, щодо якої цей сигнал вимірюється.

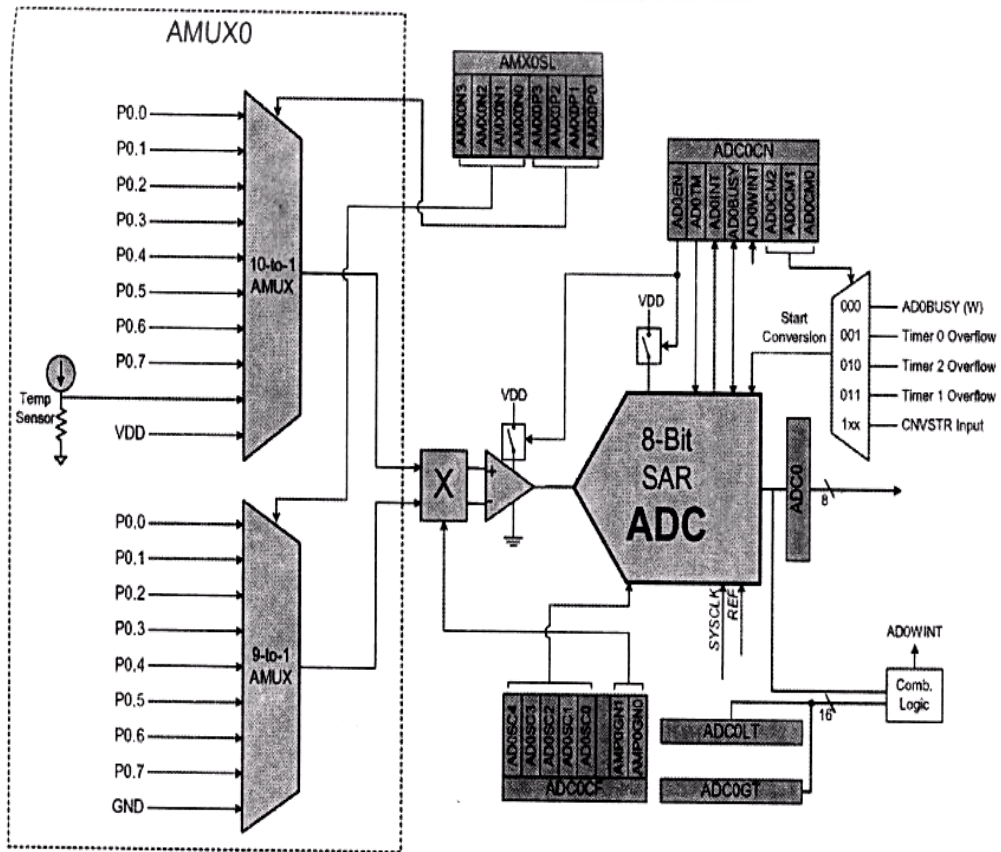


Рисунок 4.5 - Функціональна схема аналого-цифрового вузла

Якщо обидві частини мультиплексора настроєні на різні виводи мікроконтролера, маємо диференціальне включення, якщо ж друга частина мультиплексора з'єднана із землею – однополярне включення. При цьому, вимір здійснюється між землею та опорною напругою [13].

Перша половина мультиплексора має 10 входів, вісім з яких можуть бути підключені до входів порту 0, дев'ятий підключений до датчика температури, а десятий – до напруги живлення. Це дозволяє здійснювати вимір температури та напруги живлення.

Особливості компаратора. В мікроконтролері використовується оригінальна схема компаратора, що постачена вхідним мультиплексором, який складається з двох частин. Функціональна схема вузла компаратора наведена на рисунку 4.6 [13].

Джерела переривань. Оброблювач переривань мікроконтролера C8051F300 оброблює всього 13 джерел, наведених у таблиці 4.5.

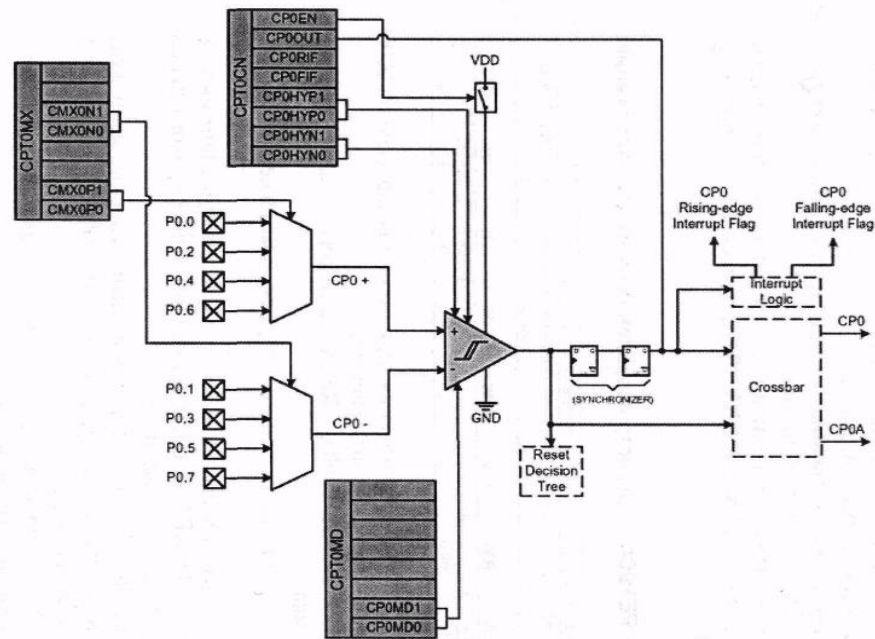


Рисунок 4.6 - Функціональна схема вузла компаратора

Таблиця 4.5 - Джерела переривань мікроконтролера C8051F300

1	2	3	4	5
Джерело переривань	Вектор	Пріоритет	Прапор переривання	Дозволено
Reset	0x0000	Вищий	Ні	Завжди дозволено
External Interrupt 0 (INT0)	0x0003	0	IE0 (TCON.1)	EX0 (IE.0)
Timer 0 Overflow	0x000B	1	TF0 (TCON.5)	ET0 (IE.1)
External Interrupt 1 (INT1)	0X0013	2	IE1 (TCON.3)	EX1 (IE.2)
Timer 1 Overflow	0X001B	3	TF1 (TCON.7)	ET1 (IE.3)
Serial Port (UART0)	0X0023	4	RI (SCON.0) TI (SCON.1)	ES (IE.4)

Продовження таблиці 4.5

1	2	3	4	5
SMBus	0X0033	6	SI (SMBOCN.0)	ESMB0 (EIE1.0)
ADC0 Window Comparison	0X003B	7	ADWINT (ADCOCN.3)	EWADC0 (EIE1.1)
ADC0 End of Conversion	0X0043	8	ADCINT (ADCOCN.5)	EADC0 (EIE1.2)
PCA	0X004B	9	CF (PCA0CN.7) CCFn (PCA0CN.n)	EPCA0 (EIE.3)
Comparator 0 Falling Edge	0X0053	10	CP0FIF (CPT0CN.4)	ECP0F (EIE1.4)
Comparator 0 Rising Edge	0X0058	11	CP0RIF (CPT0CN.5)	ECP0R (EIE1.5)

Flash-пам'ять та байт захисту мікроконтролера C8051F300. Байт захисту (таємності) розділений на два напівбайти. Запис хоча б одного нуля в біти від 7 до 4 приводить до захисту від запису або сторінкового стирання всієї Flash-пам'яті, минаючи інтерфейс I32. Запис хоча б одного нуля в біти від 3 до 0 призводить до захисту від читання, запису або сторінкового стирання всієї Flash-пам'яті, минаючи інтерфейс I32. Будь-яка спроба встановлення нульових бітів в одиницю в байті захисту призводить до повного стирання пам'яті програм [13]. FLKEY – реєстр блокування Flash-пам'яті. У цей реєстр необхідно здійснювати запис кодової послідовності 0xA5 & 0xF1 перед кожною операцією запису або стирання. Пам'ять буде замкнена до наступного скидання, якщо були записані неправильні коди або були початі спроби Flash-операцій перед записом у реєстр коректних кодів 0xA5 & 0xF1 [13].

4.2 Ethernet-контролер CP2200

CP2200 – єдиний чип Ethernet-контролер, який містить об'єднаний IEEE 802.3 Ethernet Media Access Controller (MAC), 10 BASE-T, фізичний шар (PHY) та енергонезалежну Flash-пам'ять на 8КБ. CP2200 може додати Ethernet

можливість приєднання до будь-якого мікроконтролеру або хост процесору з 11 або більшою кількістю портів введення/виводу. Паралельна 8-бітна шина інтерфейсу підтримує Intel та Motorola формати шин у мультиплексному та немультимплексному режимах. Швидкість передачі даних у немультимплексному режимі може перевищити 30 Mbps [14].

Flash-пам'ять може використовуватися для зберігання користувальницьких констант, змісту сервера мережі, або в якості енергонезалежної пам'яті. Flash-пам'ять містить унікальну MAC-адресу на 48 бітів, що втримується в останніх шести комірках пам'яті. Наявність унікальної MAC-адреси, яка міститься в CP2200, часто відокремлює крок виробничого процесу виробу більшості вкладених систем від перетворення в послідовну форму.

CP2200 має чотири режими роботи зі зміною рівнів функціональних можливостей, які дозволяють host-процесору повністю здійснювати керування. Додаткова кнопка переривань дозволяє host-процесору входити в режим "очікування" та виходити з нього коли пакет отриманий або коли CP2200 підключений до мережі. Автопереговори дозволяють пристрою автоматично вибирати найефективніший дуплексний режим (half/full duplex), який підтримується мережею.

На рисунку 4.7 показана функціональна схема Ethernet-контролера [14].

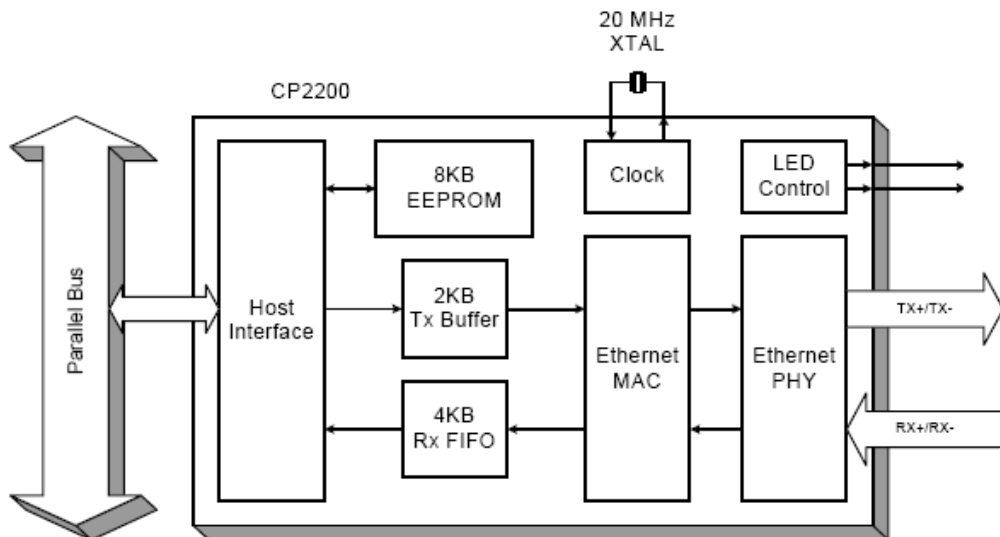


Рисунок 4.7 - Функціональна схема Ethernet-контролера

- Ethernet PHY – фізична реалізація Ethernet-рівня (реалізація за допомогою елементів);
- Ethernet MAC – MAC-адреса мікросхеми;
- Tx Buffer – буфер передачі;
- Rx FIFO – буфер приймання;
- Host Interface – інтерфейс сполучення з паралельною шиною;
- Parallel Bus – паралельна шина;
- EEPROM – енергонезалежна пам'ять;
- Clock – генератор;
- LED Control – контролер світлодіодів.

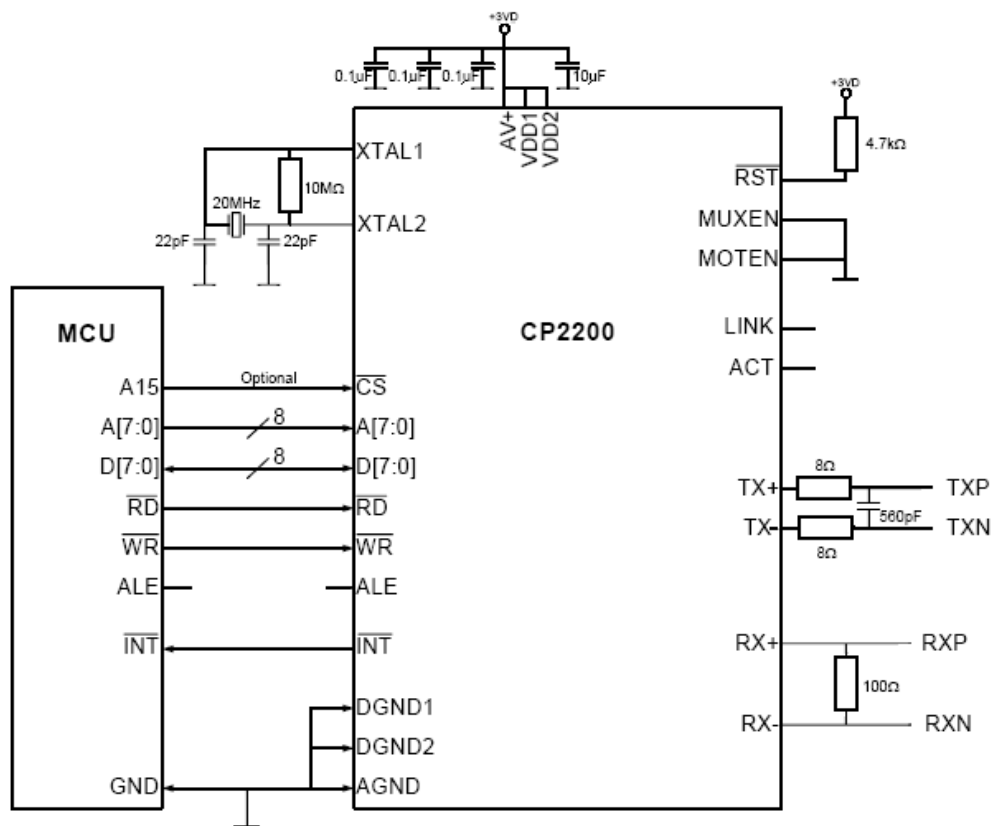


Рисунок 4.8 – Схема об'язки Ethernet-контролера, працюючого в немультіплексному режимі

4.3 Блок узгодження та гальванічної розв'язки

Блок узгодження та гальванічної розв'язки містить [14]:

- стандартний роз'єм RJ-45, який служить для підключення мережевого кабелю до пристрою;
- два телекомунікаційних трансформатори для передачі цифрового потоку (гальванічної розв'язки);
- два світлодіоди, які виконують функцію індикації режиму роботи пристрою (приймання/передачі).

Принципова схема БУГР наведена на рисунку 4.9.

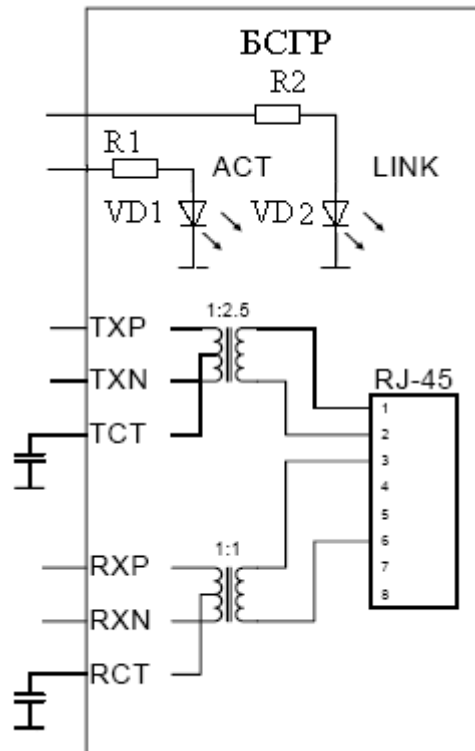


Рисунок 4.9 - Принципова схема БУГР

Для пристрою застосуємо телекомунікаційні трансформатори для передачі цифрового потоку марки TEW 5721-1 фірми Filtran LTD тому що вони досить дешеві та надійні [13].

Світлодіод застосуємо L-53 SGC фірми KING BRIGHT. Розрахуємо резистори R1 та R2. У зв'язку з тим що світлодіоди VD1 та VD2 однакові, резистори R1 та R2 також рівні.

$$R = \frac{U_{mc} + U_{VD}^{np}}{I}$$

де U_{mc} – напруга на виході мікросхеми;

U_{VD}^{np} – гранична напруга світлодіода.

$$R = \frac{3,3 - 2,1}{0,01} \approx 120(\text{Ом}).$$

5 РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПРИСТРОЮ

Основним елементом розробленої принципової схеми пристрою комутації є мікроконтролер С8051F300. Програмне забезпечення для даного мікропроцесора розроблюється на мові програмування АSМ-85. Лістинг розробленої підпрограми ініціалізації мікроконтролера має наступний вигляд.

BEGIN DI;	Заборона переривання
MVI A,40H;	Запис інструкції програмного скидання
OUT INSTR	
MVI A,7DH;	Запис інструкції режиму
OUT INSTR	
MVI A,31H ;	Запис інструкції команди передачі
OUT INSTR	
MVI B,N;	Встановлення лічильника масиву даних
LXI H,ADDR;	Завантаження початкової адреси масиву
ENTR: MOV A,M;	Передача елемента масиву в акумулятор
OUT DATA;	Запис елемента масиву
WAIT: IN INSTR;	Слово стану
MOV C,A;	Зберігання слова стану
ANI 08;	Виділення біта помилки парності
CNZ ERR;	Якщо помилка, то на програму оброблення
MOV A,C;	Відновлення слова стану
RAR;	Контроль готовності передавача
JNC WAIT;	Якщо не готовий, то повторення
DCR B;	Зміна лічильника елементів масиву
JZ EXIT;	Якщо все, то вихід з програми
INX H;	Наступний елемент масиву
JMP ENTR;	Повторення циклу передачі
EXIT: MVI A,38H ;	Запис інструкції команди кінця передачі
OUT INSTR EI;	Дозвіл переривання

ВИСНОВКИ

У роботі був розроблений пристрій комутації для локальної мережі на базі протоколу RIP. З міркувань швидкої адаптації до змін конфігурації мережі, а також правильної передачі пакетів у мережах з довільною топологією, яка допускає наявність замкнених контурів, в якості комунікаційного пристрою був обраний маршрутизатор.

Маршрутизатор на базі RIP протоколу виконує функції приймання та передачі пакета по найкоротшому маршруту. Цей пристрій створює маршрутні таблиці, які містять інформацію про усі можливі маршрути до усіх відомих вузлів. Для автоматичної побудови цих таблиць маршрутизатори обмінюються інформацією про топологію складеної мережі у відповідності зі спеціальним службовим протоколом – протоколом маршрутизації RIP. Протокол RIP підтримує тільки найкращі маршрути до пункту призначення. Якщо нова інформація забезпечує кращий маршрут, то ця інформація заміняє попередню маршрутну інформацію.

Також у ході роботи були спроектовані структурна, функціональна, принципова схеми та розроблений алгоритм роботи маршрутизатора.

Цей комунікаційний пристрій виконаний на базі мікропроцесорної системи: мікроконтролера 38051F300, Ethernet-контролера CP2200, блоку узгодження та гальванічної розв'язки. Усі блоки маршрутизатора вибиралися з максимально кращими характеристиками роботи, що забезпечує надійне функціонування даного пристрою.

СПИСОК ЛІТЕРАТУРИ

1. Олифер В.Г. Компьютерные сети. Принципы, технологии, протоколы: Учебник/ В.Г. Олифер, Н.А. Олифер. – СПб.: Питер, 2016. – 992 с.
2. Джеймс Ф. Куроуз, Кит В. Росс. Компьютерные сети. – СПб.: Питер, 2004. – 765 с.
3. Компьютерные сети. – 4-е изд. / Э. Танненбаум. – СПб.: Питер, 2005. – 992 с.
4. Хелеби С., Мак-Ферсон Д. Принципы маршрутизации в Internet. 2-е издание. Пер. с англ. М.: Издательский дом "Вильямс", 2001. – 448 с.
5. A.G. Loshkovskii. Teletraffic Theory. Methodical instructions for laboratory works on course «Teletraffic Theory in Telecommunications» / A.G. Loshkovskii, T.R. Shmeleva. – Odessa: ONAT after A.S. Popov, 2013. – 36 p.
6. https://cirspb.ru/blog/info-satellite/st_gps/, 20 октября 2018.
7. http://www.navgeocom.ru/infra/hard/leica_1200gg/ar25.htm, 2016.
8. <https://www.quora.com/What-is-network-technology/> Dec 20 / 2018.
9. <http://easy-code.com.ua/2016/08/ethernet-v-promislovosti-lokalni-merezhi>, 2016.
10. <http://www.ua5.org/lan//2018/12/124-lokaln-merezh.html>, 2018.
11. <https://infotel.ua/ua//2017/12/local-aria-network-data-transferring>, 2017.
12. Dilber M. N., Raza A. Analysis of successive Link Failures effect on RIP and OSPF Convergence time delay. International Journal of Advances in Science and Technology, 2014, pp. 42-48.
13. Жуйков В.Я. Мікропроцесори і мікроконтролери: електронний підручник з грифом МОН / Жуйков В.Я., Терещенко Т.О., Петергеря Ю.С., Хохлов Ю.В. – <http://www.kaf-pe.ntu-kpi.kiev.ua>, 2010.
14. Швець Є.Я. Матеріали і компоненти електроніки: навчальний посібник / Є.Я. Швець, І.Ф. Червоний, Ю.В. Головка. – Запоріжжя: ЗДІА, 2014. – 278 с.