

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Західноукраїнський національний університет
Факультет комп'ютерних інформаційних технологій
Кафедра комп'ютерної інженерії

Батун Олег Васильович

**«Таблично-алгоритмічні засоби
обчислення скалярного добутку / Tabular-
algorithmic means of the scalar
product calculating»**

спеціальність: 123 - Комп'ютерна інженерія
освітньо-професійна програма - Комп'ютерна інженерія
Кваліфікаційна робота

Виконав студент групи КІм-21
О.В. Батун

Науковий керівник:
д.т.н., проф. В.М. Теслюк

Кваліфікаційну роботу допущено

до захисту:

"__" _____ 20__ р.

Завідувач кафедри

_____ О. М. Березький

Тернопіль – 2021

ВСТУП

Актуальність теми.

На етапі розвитку нейронауки з обробки інформації в реальному часі сфера їх застосування розширюється, і багато з них вимагають розвитку інтенсивних інформаційних потоків на обладнанні, що реагує на такі обмеження, як розмір, вага та енергоспоживання. Ці галузі включають промисловість (технологічний процес, управління складними установками), енергетику (оптимізація навантаження електричної системи), військову (технічне бачення, управління мобільною роботою), автомобільну (управління рухом і двигунами) та медицину (діагностика захворювань).) та інструменти (розпізнавання шаблонів та оптимізація керування). Більшість інсталяцій нейронних мереж мають універсальний тип структури, який є функціонально та структурно надлишковим, ігнорує специфічні вимоги до продуктивності з точки зору продуктивності, розміру та енергоспоживання та є неефективним.

Для створення високоефективної нейронної мережі в режимі з опрацюванням у даний час потрібно розробити та впровадити нову нейронну модель, алгоритм і спеціальну структуру NBSI, спрямовану на інтенсивне використання сучасних елементарних основ та реалізацію надвеликих інтегральних схем (НБСІ). нервових елементів і нервових клітин. мережі. У зв'язку з цим особливе значення набуває впровадження високопродуктивних НАДВЕЛИКИХ ІНТЕГРАЛЬНИХ СХЕМ для інтеграції нейронних елементів і нейронних мереж для обробки інформації в реальному часі та використання обладнання.

Реалізуючи нейронні мережі високої ємності з використанням обладнання з опрацюванням у даний час та NBSI, обчислювальний процес дозволяє паралельним і пластичним алгоритмам нейронних мереж мати архітектурні карти, які відповідають швидкості потоку інформації.

Реалізація VLSI вимагає зменшення витрат на зв'язок і апаратне забезпечення між голками інтерфейсу та нейронами для управління структурою нейронних елементів і нейронних мереж. Цим вимогам можуть задовольнити паралельні вертикальні методи проектування та використання нейронних елементів і нейронних мережевих структур з обчислювальними даними у двійкових операндах.

Для того щоб інтегрувати нейронні елементи та нейронні мережі великої ємності в режимі з опрацюванням у даний час за допомогою обладнання, потрібно розробити нові нейронні моделі, методи обробки інформаційних потоків, структури NBSI та методи їх інтеграції. Найкращим способом розробки таких нейронних елементів і нейронних мереж є використання сучасних будівельних блоків, проектування нейронних елементів, обчислювальних методів нейронних мереж, архітектури нейронних мереж, вимог специфічних для додатків та інформаційних підходів.

Нейронні елементи та технології нейронних мереж, синтезовані в цьому високоефективному пристрої, засновані на роботах відомих вчених, які впровадили теоретичних основ та практичні принципи проектування нейронних мереж.

Тому важливою є розробка нової моделі деяких компонентів системи обробки сигналів з опрацюванням у даний час.

Мета і завдання дослідження.

Метою кваліфікаційної роботи є розробка пристрою обчислення скалярного добутку. Для досягнення цієї мети потрібно вирішити наступні завдання. • Аналіз методів та інструментів виготовлення та впровадження пристрою обчислення скалярного добутку, виділення основи функціонування пристрою обчислення скалярного добутку, основних напрямків розвитку засобів виготовлення та впровадження пристрою обчислення скалярного добутку; вибір інструментів для формулювання та

виготовлення та впровадження вимог і принципів створення пристрою обчислення скалярного добутку; розробка методів паралельно-вертикальної обробки інформації у нейронних мережах; • моделювання паралельно-вертикальних формальних нейронів; інтеграція паралельно-вертикальних НБС-орієнтованих нейронних елементів;

Об'єктом дослідження є процеси обчислення скалярного добутку у реальному часі.

Предметом дослідження є моделі нейронних систем, методи генерації нейронних елементів і апаратних нейронних мереж з опрацюванням у даний час.

Методи дослідження. Для розв'язання поставлених у кваліфікаційній роботі задач використано: теоретичних основ та методи моделювання пристрою обчислення скалярного добутку, елементи законів побудови графів, теоретичних основ та методик для забезпечення паралельності алгоритмів, теоретичних основ проектування напівзамовних і замовних НАДВЕЛИКИХ ІНТЕГРАЛЬНИХ СХЕМ.

ЗМІСТ

1. АНАЛІЗ ГАЛУЗЕЙ ЗАСТОСУВАННЯ НЕЙРОННИХ МЕРЕЖ... ОШИБКА!**ЗАКЛАДКА НЕ ОПРЕДЕЛЕНА.**

1.1. Галузі застосування та задачі розв'язування нейронних мережами.....	6
1.2. Архітектура штучних нейронних мереж	10
1.3. Засоби обміну для штучних нейронних мереж.....	21
1.4. Засоби виготовлення та впровадження пристроїв обміну	23
<u>2. РОЗРОБКА ПРИСТРОЮ ОБМІНУ НА БАЗІ БАГАТОПОРТОВОЇ ПАМ'ЯТІ ДЛЯ НЕЙРОННИХ МЕРЕЖ.....</u>	25
2.1. Формування вимог до пристрою обміну	25
2.2. Вибір принципів та елементарної бази для виготовлення та впровадження пристрою обміну	30
2.3. Розробка структури пристрою обміну на базі багатопортової пам'яті	37
2.4. Розробка модуля контролераобміну.....	38
<u>3. РОЗРОБКА ПРОГРАМ ДЛЯ МОДЕЛЮВАННЯ ПРИСТРОЮ ОБМІНУ НА БАЗІ БАГАТОПОРТОВОЇ ПАМ'ЯТІ</u>	43
3.1. Синтез багатопортової пам'яті для комп'ютерних систем з опрацюванням у даний час	43
3.2. Розробка алгоритму роботи пристрою обміну на базі багатопортової пам'яті 46	
3.3. Моделювання компонентів бпп	50

1. АНАЛІЗ ЗАСТОСУВАННЯ НЕЙРОННИХ МЕРЕЖ

1.1. Задачі та можливості застосування застосування нейронних мереж.

Штучні нейронні мережі отримали широке поширення за останні 20 років і дозволили вирішувати складні завдання обробки даних, часто значно перевершуючи точність інших методів статистики і штучного інтелекту, або будучи єдиною можливим методом вирішення окремих завдань. Нейронних мережа відтворює структуру і властивості нервової системи живих організмів: нейронна мережа складається з великого числа простих обчислювальних елементів (нейронів) і володіє більш складною поведінкою в порівнянні з можливостями кожного окремого нейрона. Нейронних мережа отримує на вході набір вхідних сигналів і видає відповідний їм відповідь (вихідні сигнали), що є рішенням задачі. Сьогодні є безперечним значний науковий та практичний інтерес до обчислювальних структур нового типу — штучних нейронних мереж. Він спричинений низкою успішних застосувань цієї нової технології, яка дозволила розробити ефективні підходи до вирішення проблем, що вважалися складними для виготовлення та впровадження на традиційних комп'ютерах. На назву “нейронні мережі” зараз претендують усі обчислювальні структури, які в тій чи іншій мірі моделюють роботу мозку. Але таке моделювання, здебільшого, є дуже фрагментарним, і говорити про створення у найближчому майбутньому штучного мозку або навіть деякої його моделі, яка дублювала б роботу мозку найпримітивніших живих створінь, ще зарано. Такий висновок впливає з надзвичайної складності цього загадкового витвору природи. При побудові моделі мозку розглядають локальні та глобальні аспекти пізнання його функціонування. Основною глобальною характеристикою, яка істотно утруднює моделювання, є надзвичайно велика кількість базових структурних елементів.

Мозок людини містить близько сотні мільярдів нейронів, кожен з яких кількома тисячами зв'язків об'єднується з іншими нейронами. Використання

навіть найпростішої моделі нейрона не дозволяє побудувати модель мозку, що наближалася б за своїми глобальними показниками до реального об'єкта моделювання. До локальних характеристик слід віднести власне принципи, за якими будують модель нейрона. Останнім часом нейробіологія досягла значних успіхів у вивченні нейрона як елементарної структурної одиниці мозку. Відкрито велику кількість закономірностей, що описують його функціонування та взаємодію з іншими нейронами. Однак, як і раніше, залишаються без відповіді питання про те, яким чином реалізуються такі властиві мозку основні функції, як пам'ять та свідомість. Отже, залишається до кінця не з'ясованим зв'язок між локальними характеристиками нейрона та глобальними функціями мозку. Але саме такий зв'язок і є основою побудови штучних нейронних мереж, які моделюють функції мозку. Тому основною проблемою концептуального підходу до нейронних мережного моделювання є вертикальна стратифікація моделі, тобто з'ясування питання про взаємодію елементів на всіх рівнях знизу вгору. Лише шляхом вдалої координації дій великої кількості структурних елементів можливо досягти вияву якісно нової властивості всієї моделі.

Успішний розвиток теорії нейронних мереж за останнє десятиліття дозволив реалізувати низку цих глобальних властивостей. Найвідоміші з них – навчання, узагальнення та абстрагування. Властивість навчання проявляється у здатності нейронної мережі змінювати свою поведінку залежно від стану середовища. Завдяки цій властивості нейронні мережі привертають значну увагу. Існує велика різноманітність алгоритмів навчання нейронних мереж, кожен з яких має свої сильні та слабкі сторони, але сьогодні досі немає консенсусу щодо того, чому можна навчати нейронних мереж, і як це навчання має здійснюватися. Властивість узагальнення дозволяє нейронній мережі знизити чутливість до малих коливань вхідних сигналів.

Ця властивість дуже важлива для об'єктів, які існують у реальному світі. Особливістю нейронних мережевого підходу в даному випадку є те, що узагальнення є результатом властивостей структури, а не роботи спеціальної

програми на традиційному комп'ютері. Властивість абстракції дозволяє створити нову сутність у нейронній мережі на основі аналізу вхідної інформації. Ця властивість особливо очевидна для завдань розпізнавання зображень. Завдяки цьому нейронна мережа може створити ідеальне зображення, керуючись вхідною інформацією, яка має певні властивості цього зображення.

Сфери застосування нейронних мереж:

Економіка та бізнес: прогнозування ринку, автоматична торгівля, оцінка кредитного дефолту, прогнозування банкрутства, оцінка нерухомості, виявлення завищених і недооцінених компаній, автоматичний рейтинг, оптимізація портфеля, оптимізація товарних і грошових потоків, автоматичне за який здійснюється зчитування чеків і форм, безпека транзакцій пластикові картки.

- Медицина: обробка медичних зображень, моніторинг пацієнта, діагностика, факторний аналіз ефективності лікування, очищення показань приладу від шуму.

- Авіоніка: студентський автопілот, розпізнавання радіолокаційних сигналів, адаптивне пілотування сильно пошкоджених літаків.

- Зв'язок: стиснення відео, швидке кодування-декодування, оптимізація стільникової мережі та схеми маршрутизації пакетів.

- Інтернет: асоціативний пошук інформації, електронні секретарі та користувачські агенти в мережі, фільтрація інформації в push-системах, спільна фільтрація, заголовки новин, цільова реклама, цільовий маркетинг електронної комерції.

- Автоматизація виробництва: оптимізація режимів виробничого процесу, комплексна діагностика якості продукції (УЗД, оптика, гамма-випромінювання, ...), моніторинг та візуалізація багатовимірної інформації про відвантаження, запобігання аварій, робототехніка.

- Політичні технології: аналіз та узагальнення опитувань громадської думки, прогнозування динаміки оцінок, визначення важливих факторів, об'єктивне групування виборців, візуалізація соціальної динаміки населення.

- Системи безпеки та безпеки: системи ідентифікації особи, розпізнавання голосу, розпізнавання натовпу, розпізнавання номерних знаків, аналіз аерофотознімків, моніторинг потоку інформації, виявлення підробок.

- Введення та обробка інформації: обробка рукописних чеків, розпізнавання підпису, відбитків пальців та голосу. Введення фінансових та податкових документів до комп'ютера.

- Геологічна розвідка: аналіз сейсмічних даних, асоціативні методи розвідки корисних копалин, оцінка родових ресурсів.

Існує кілька поширених комерційних програмних пакетів для універсальних нейронних мереж (Statistica Neural Networks, NeuroShell, Matlab Neural Network Toolbox, NeuroSolutions, BrainMaker). Існує ще багато спеціалізованих, некомерційних або дослідницьких програм, розроблених вченими для власних потреб.

Нейронні мережі сьогодні широко використовуються: нейронні мережі – це не що інше, як новий інструмент для аналізу даних. І краще за інших нею може скористатися фахівець своєї справи. Основні труднощі на шляху ще більш широкого поширення нейротехнологій - нездатність широкого кола експертів сформулювати свої проблеми в термінах, які дозволяють просте рішення нейронної мережі.

Основними цікавими особливостями нейронних мереж на практиці є наступні:

- Наявність алгоритмів швидкого навчання: навіть із сотнями вхідних сигналів і десятками-сотнями тисяч опорних ситуацій нейронну мережу можна швидко вивчити на звичайному комп'ютері. Тому нейронні мережі мають широке застосування і дозволяють вирішувати складні задачі прогнозування, класифікації чи діагностики.

- Можливість роботи з великою кількістю неінформативних, зашумлених вхідних сигналів – не потрібно їх усувати завчасно, нейронна мережа сама визначає їх як непридатні для розв'язання задачі і може однозначно відкинути.

- Здатність працювати з корельованими незалежними змінними, з різними типами інформації – дискретною, кількісною та якісною, що часто ускладнює статистичні методи

- Нейронна мережа може вирішувати кілька завдань на одному наборі вхідних сигналів одночасно - має кілька виходів, прогнозує значення кількох показників.

Алгоритми навчання висувають дуже мало вимог до структури нейронної мережі та властивостей її нейронів. При наявності досвіду або в разі особливих вимог можна вибрати тип і властивості нейронів і нейронних мереж, скласти структуру нейронної мережі вручну, з окремих елементів і встановити необхідні властивості для кожного з них.

1.2. Архітектура штучних нейронних мереж

Оскільки всі штучні нейронні мережі засновані на концепції нейронів, зв'язків і функцій передачі, існує подібність між різними структурами або архітектурами нейронних мереж. Більшість змін походить від різних правил навчання. Розглянемо деякі з найвідоміших штучних нейронних мереж.

Перцептрон Розенбальта

Перцептрон Розенбалата вважається першою моделлю нейронних мереж. Теорія перцептрона є основою багатьох типів штучних нейронних мереж прямого поширення і є класичною для вивчення.

Одношаровий перцептрон здатний розпізнавати найпростіші зображення. Один нейрон обчислює зважену суму сигналів вхідного елемента, віднімає значення зміщення і передає результат через фіксовану порогову функцію, вихідний результат якої дорівнює +1 або -1. Залежно від значення вихідного сигналу вирішується:

- +1 - вхідний сигнал відноситься до класу А,
- -1 - вхідний сигнал відноситься до класу В.

На малюнку. 1.1 наведено схему одношарового перцептрона, графік передатної функції та діаграму областей прийняття рішень, створених у просторі багатовимірного вхідного сигналу. Домени рішень визначають, які вхідні зображення будуть класифіковані як клас А, а які — як клас В. Перцептрон, що складається з одного нейрона, утворює два домени рішень, які розділені надбудовою.

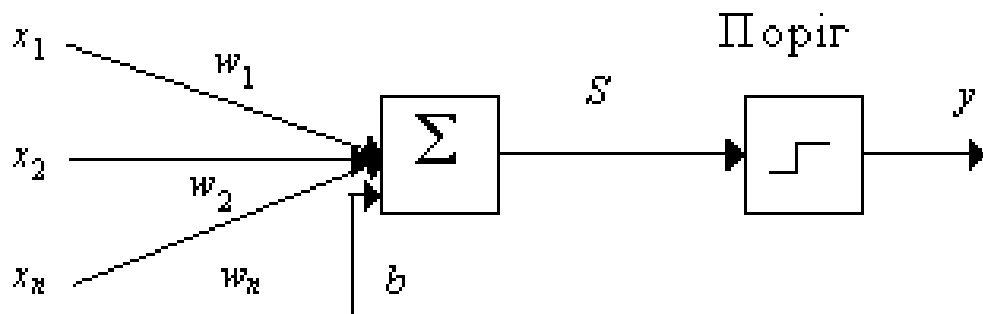


Рисунок. 1.1. Схема нейрона

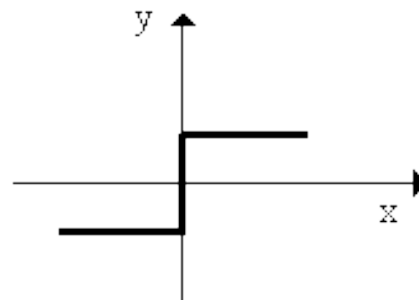


Рисунок. 1.2. Графік передатної функції

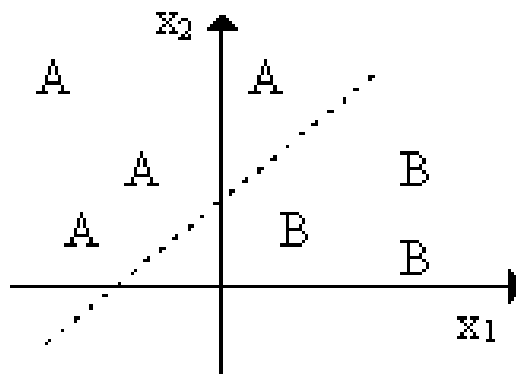


Рисунок. 1.3. Поділяюча поверхня

На рисунку 1.2 зображено випадок з розміром вихідного сигналу – 2. Розділювальною поверхнею є пряма на площині. Рівняння, яке визначає лінію поділу, залежить від значень синаптичних ваг і спотворення.

Нейронна мережа зворотних посилянь

Архітектура FeedForward BackPropagation була розроблена на початку 1970-х років кількома незалежними авторами: Werbor; Паркер; Румельхарт, Хінтон і Вільямс. Сьогодні парадигма BackPropagation є популярною, ефективною та легкою в освоєнні моделлю навчання для складних багаторівневих мереж. Він використовується в різних типах додатків і створив великий клас нейронних мереж з різними структурами і методами навчання.

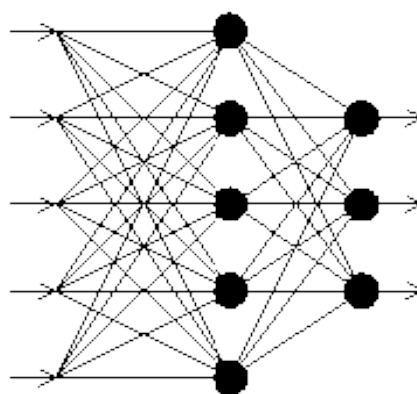


Рисунок. 1.4. Мережа зворотного поширення похибки

Типова мережа BackPropagation має вхідний шар, вихідний шар і принаймні один прихований шар. Теоретично немає обмежень на кількість прихованих шарів, але на практиці застосовуються один-два.

Нейрони організовані в шарувату структуру з прямою передачею сигналу (уперед). Кожен нейрон в мережі генерує зважену суму своїх вхідних даних, передає це значення через передатну функцію і виводить вихідне значення. Мережа може моделювати функцію практично будь-якої складності, а кількість шарів і кількість нейронів у кожному шарі визначає складність функції.

При моделюванні мережі важливо визначити кількість проміжних шарів і кількість нейронів у них. Більшість дослідників та інженерів використовують загальні правила, зокрема:

1. Кількість входів і виходів мережі задається кількістю вхідних і вихідних параметрів досліджуваного об'єкта, явища, процесу тощо. На відміну від зовнішніх шарів, кількість нейронів у латентному шарі n вибирається емпірично. У більшості випадків буде достатня кількість нейронів $n_{prh} \leq n_{vh} + n_{vuh}$, де n_{vh} , n_{vuh} - кількість нейронів у вхідних і відповідно вихідних шарах.

2. Якщо складність зв'язку між отриманими та необхідними вихідними даними зростає, кількість нейронів у латентному шарі також має збільшитися.

3. Якщо змодельований процес можна розділити на багато етапів, потрібен ще один прихований шар(и). Якщо процес не розділений на фази, інші рівні можуть дозволити запам'ятовувати і, як наслідок, прийняти неправильне загальне рішення.

Типова мережа BackPropagation має вхідний шар, вихідний шар і принаймні один прихований шар. Теоретично немає обмежень на кількість прихованих шарів, але на практиці використовуються один або два.

Нейрони організовані в шарувату структуру з прямою передачею сигналу (уперед). Кожен нейрон в мережі генерує зважену суму своїх вхідних даних, передає це значення через передатну функцію і виводить вихідне значення.

Мережа може моделювати функцію практично будь-якої складності, а кількість шарів і кількість нейронів у кожному шарі визначає складність функції.

При моделюванні мережі важливо визначити кількість проміжних шарів і кількість нейронів у них. Більшість дослідників та інженерів використовують загальні правила, зокрема:

1. Кількість входів і виходів мережі задається кількістю входних і вихідних параметрів об'єкта, явища, процесу тощо. На відміну від зовнішніх шарів, кількість нейронів у латентному шарі n вибирається емпірично. У більшості випадків буде достатня кількість нейронів $n_{prih} \leq n_{vh} + n_{vyh}$, де n_{vh} , n_{vyh} - кількість нейронів у входному та вихідному шарі.

2. Якщо складність зв'язку отриманих і необхідних вихідних даних зростає, то кількість нейронів у латентному шарі також має збільшитися.

3. Якщо змодельований процес можна розділити на багато етапів, потрібен ще один прихований шар(и). Якщо процес не розділений на фази, наступні рівні можуть дозволити вам запам'ятати і, як наслідок, прийняти погане рішення.

Після визначення кількості шарів і кількості нейронів у кожному з них потрібно знайти значення синаптичних ваг і порогів мережі, які здатні мінімізувати похибку результату. Тому існують алгоритми навчання, де мережева модель адаптується до наявних навчальних даних. Похибка для конкретної моделі мережі визначається шляхом опускання мережі всіх прикладів навчання та порівняння згенерованих вихідних значень з необхідними значеннями. Набір помилок створює функцію помилки, значення якої можна вважати помилкою мережі. Сума квадратів помилок найчастіше використовується як функція помилок.

Щоб краще зрозуміти алгоритм навчання зворотного поширення, потрібно уточнити поняття області стану. Кожне значення синаптичних ваг і порогів мережі (вільні параметри моделі після N) відповідає одному виміру в багатовимірному просторі. Розмір $N + 1$ відповідає помилці мережі. Для різних комбінацій масштабів відповідна помилка мережі може бути представлена

точкою в $N + 1$ -вимірному просторі, всі ці точки утворюють поверхневий стан станів. Мета навчання нейронної мережі - знайти найнижчу точку на багатовимірній поверхні.

Поверхня млинів має складну структуру і дещо неприємні властивості, зокрема наявність локальних мінімумів (точок, найнижчих у навколишньому середовищі, але вище глобального мінімуму), плоских поверхонь, сідластих точок і довгих вузьких ущелин. Аналітичні інструменти не можуть визначити положення глобального мінімуму на поверхні станів, тому вивчення нейронної мережі, по суті, є вивченням цієї поверхні.

Починаючи з початкової конфігурації ваг і порогів (з випадкової точки на поверхні), алгоритм навчання поступово знаходить глобальний мінімум. Розраховується вектор градієнта похибки поверхні, який вказує напрямок найкоротшого спуску на поверхні від заданої точки. Якщо трохи рухатися, похибка зменшиться. Зрештою, алгоритм зупиняється в кінці, який може бути лише локальним мінімумом (в ідеалі — глобальним мінімумом).

Проблема полягає у виборі довжини сходинок. При великій довжині кроку зближення буде швидшим, але є небезпека, що ви не приймете рішення або підете в неправильному напрямку. Невеликий крок визначає правильний напрямок, але кількість ітерацій збільшується. На практиці розмір кроку приймається пропорційно нахилу з певною постійною швидкістю навчання. Правильний вибір швидкості навчання залежить від конкретного завдання і проводиться експериментально. Ця константа також може залежати від часу, який зменшується з ходом роботи алгоритму.

Алгоритм працює ітераційно, його кроки називають епохами. У кожній епосі всі приклади навчання послідовно перераховуються на вході мережі, початкові значення мережі порівнюються з заданими значеннями, і обчислюється помилка. Значення помилок, а також стани градієнта поверхні використовуються для корекції ваг і повторюваних дій. Процес навчання припиняється або через певну кількість епох, або коли помилка досягає певного

рівня деталізації, або коли помилка перестає зменшуватися (користувач зазвичай вибирає правильний критерій зупинки).

Мережа була розроблена Тойво Кохоненом на початку 1980-х років і принципово відрізняється від мереж, про які йшлося вище, оскільки використовує неконтрольоване навчання, а навчальний набір складається лише із значень вхідних змінних.

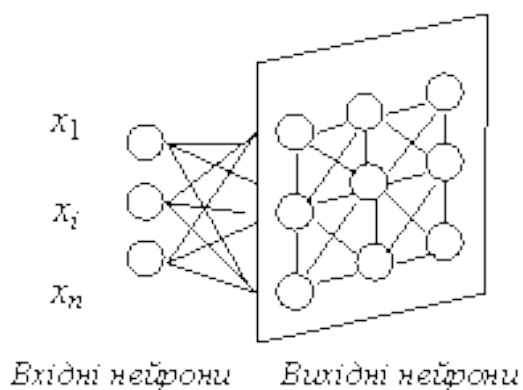


Рисунок. 1.5. Мережа Кохонена

Мережа розпізнає кластери в навчальних даних і розподіляє дані у відповідні кластери. Якщо інша мережа зустрине набір даних, не схожий ні на один із відомих шаблонів, вона відправить його до нового кластера. Якщо дані містять позначення класів, мережа може вирішити проблеми з класифікацією. Мережі Кохонена також можна використовувати в задачах, де класи відомі - перевага буде в здатності мережі виявляти схожість між різними класами.

Мережа Кохонена має лише два рівні: вхідний і вихідний. Елементи карти розташовані в деякому просторі, як правило, двовимірному. Методом послідовних наближень досліджується мережа Кохонена. У процесі навчання дані надходять на вхідні дані, але мережа адаптується не до вихідного опорного значення, а до шаблонів вхідних даних. Навчання починається з випадково вибраного початкового розташування центрів.

У процесі послідовного введення в мережу навчальних прикладів визначається найбільш подібний нейрон (той, у якого скалярний добуток ваг і вектор, що надходять на вхід, мінімальний). Цей нейрон оголошується

переможцем і є центром коригування ваг у сусідніх нейронах. Це правило навчання включає «змагальне» навчання з урахуванням відстані нейронів від «нейрона-переможця».

Навчання не мінімізує помилку, а налаштовує масштаби (внутрішні параметри нейронної мережі) для максимально можливої узгодження з вхідними даними.

Основний ітераційний алгоритм Кохонена проходить через серію епох, кожна з яких обробляє один приклад із навчальної вибірки. Вхідні сигнали поступово подаються в мережу, а необхідні вихідні сигнали не вказуються. Після подання достатньої кількості вхідних векторів синаптичні ваги мережі можуть ідентифікувати кластери. Масштаби організовані так, що топологічно близькі вузли реагують на подібні вхідні сигнали.

В результаті роботи алгоритму центр кластера встановлюється в певну позицію, що відповідає кластеризованим прикладам, для яких нейрон є «переможцем». В результаті мережевого навчання потрібно визначити ступінь сусідства нейронів, тобто оточення нейрона-переможця, яке представляє собою кілька нейронів, що оточують нейрон-переможець.

Спочатку велика кількість нейронів належить до середовища, потім її розміри поступово зменшуються. Мережа утворює топологічну структуру, в якій подібні приклади утворюють групи прикладів, близькі до топологічної карти.

Джон Хопфілд вперше представив свою асоціативну мережу в 1982 році в Національній академії наук. На честь Хопфілда та нового підходу до моделювання цю мережеву парадигму називають мережею Хопфілда. Мережа заснована на аналогії фізики динамічних систем. Початкові мережеві програми включали асоціативну або контент-орієнтовану пам'ять і вирішували проблеми оптимізації.

Мережа Хопфілда використовує три шари: вхідний, рівень Хопфілда та вихідний. Кожен шар має однакову кількість нейронів. Виходи нейронів вхідного шару надходять на входи відповідних нейронів шару Хопфілда. Тут

облігації мають фіксовану вагу. Виходи шару Хопфілда пов'язані між собою з входами всіх нейронів шару Хопфілда, крім них самих, а також з відповідними елементами вихідного шару. Під час навчання мережа маршрутизує дані з вхідного рівня на рівень Хопфілда. Шар Хопфілда коливається до тих пір, поки не буде завершено певну кількість циклів і поточний стан нейронів шару не буде передано на вихідний шар. Цей статус відповідає зображенню, яке буде збережено в Інтернеті.

Мережне навчання Хопфілда вимагає, щоб навчальне зображення було представлено на вхідному та вихідному рівнях одночасно. Рекурсивна природа шару Хопфілда забезпечує засіб коригування всіх ваг суглобів. Для правильного навчання мережі відповідні пари вводу-виводу повинні відрізнитися.

Якщо мережа Хопфілда використовується як адресна пам'ять, вона має два основних обмеження.

Кількість зображень, які можна зберегти та точно відтворити, суворо обмежена. Якщо зберігається забагато зображень, мережа може збігатися з новим неіснуючим зображенням з усіх запрограмованих зображень, а може і не відповідати. Обмеження ємності пам'яті для мережі становить приблизно 15% від кількості нейронів у шарі Хопфілда.

Якщо приклади дуже схожі, шар Хопфілда може стати нестабільним. Шаблон зображення вважається нестабільним, якщо він застосовано в нульовий час і мережа відповідає будь-якому іншому зображенню з навчального набору. Цю задачу можна вирішити, вибравши кілька прикладів ортогонального навчання.

Структурна схема мережі Хопфілда приведена на Рисунок.

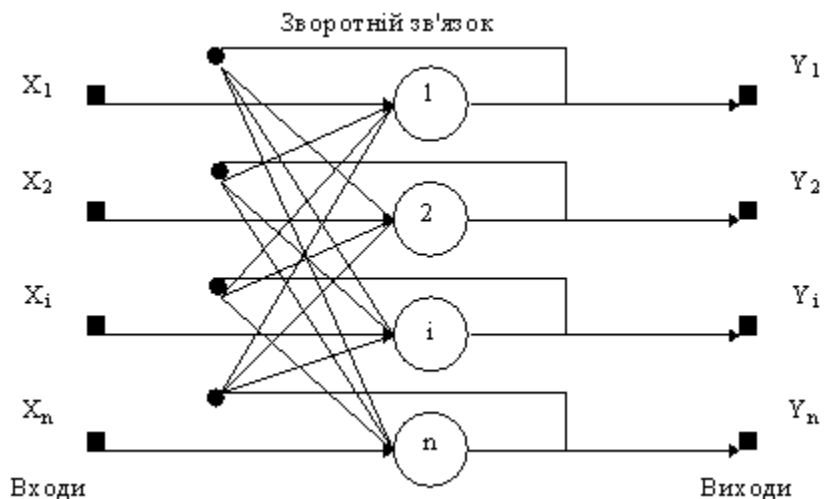


Рисунок. 1.6. Мережа Хопфілда

Для вирішення проблеми асоціативної пам'яті існує набір двійкових сигналів (малюнків, звукових чисел, інших даних, що описують об'єкти або характеристики процесу), який вважається зразковим. Мережа повинна мати можливість вибрати ("запам'ятати" з часткової інформації) відповідний зразок із зашумленого сигналу, який подається на її вхід, або "зробити висновок", що вхідні дані не відповідають жодній із вибірок.

У загальному випадку будь-який сигнал можна описати вектором x_1, x_i, x_n, \dots, n - кількість нейронів у мережі та розмір вхідного та вихідного векторів. Кожен елемент x_i дорівнює або $+1$, або -1 . Вектор, що описує k -ту вибірку, позначається X_k , або його складові, - $x_{ik}, k = 0, \dots, m-1, m$ - кількість вибірок. Якщо мережа розпізнає (або «запам'ятає») шаблон на основі поданих їй даних, то вона буде містити його виходи, тобто $Y = X_k$, де Y — вектор початкових значень мережі: y_1, y_i, y_n . В іншому випадку вихідний вектор не відповідає жодному із зразків.

Наприклад, якщо сигнали є певним зображенням, то при графічному відображенні даних з виходу мережі можна побачити зображення, яке повністю відповідає одній із моделей (у разі успіху) або «вільної імпровізації» мережі (в випадок невдачі).

Підшивна сітка

Мережа Хеммінга є розширенням мережі Хопфілда. Ця мережа була розроблена Річардом Ліппманом у середині 1980-х рр. Мережа Хемінга реалізує класифікатор на основі найменших помилок для двійкових вхідних векторів, де похибка визначається відстанню Хемінга. Відстань гемінування визначається як кількість бітів, які відрізняються між двома відповідними вхідними векторами фіксованої довжини. Один вхідний вектор — це мовчазний приклад зображення, інший — спотворене зображення. Вихідним вектором навчальної множини є вектор класів, до яких належать зображення. У режимі навчання вхідні вектори поділяються на категорії, для яких відстань між вибілковими вхідними векторами та поточним вхідним вектором мінімальна.

Мережа Хемінга має три шари: вхідний шар із кількістю вузлів, кількістю окремих подвійних зірок; шар категорії (шар Хопфілда), з кількістю вузлів, скільки категорій або класів; вихідний шар, який відповідає кількості вузлів у шарі категорії.

Мережа - це проста архітектура прямого розподілу з вхідним рівнем, повністю пов'язаним з рівнем категорії. Кожен нейрон у шарі категорії обернено пов'язаний з кожним нейроном у тому самому шарі та безпосередньо з'єднаний з вихідним нейроном. Вихід від шару категорій до початкового шару формується конкуренцією.

Мережне навчання Хемінга подібне до методології Хопфілда. Вхідний рівень отримує потрібне навчальне зображення, а вихідний рівень отримує значення потрібного класу, до якого належить вектор. Вихід містить лише значення класу, до якого належить вхідний вектор. Рекурсивна природа шару Хопфілда забезпечує засіб коригування всіх ваг суглобів.

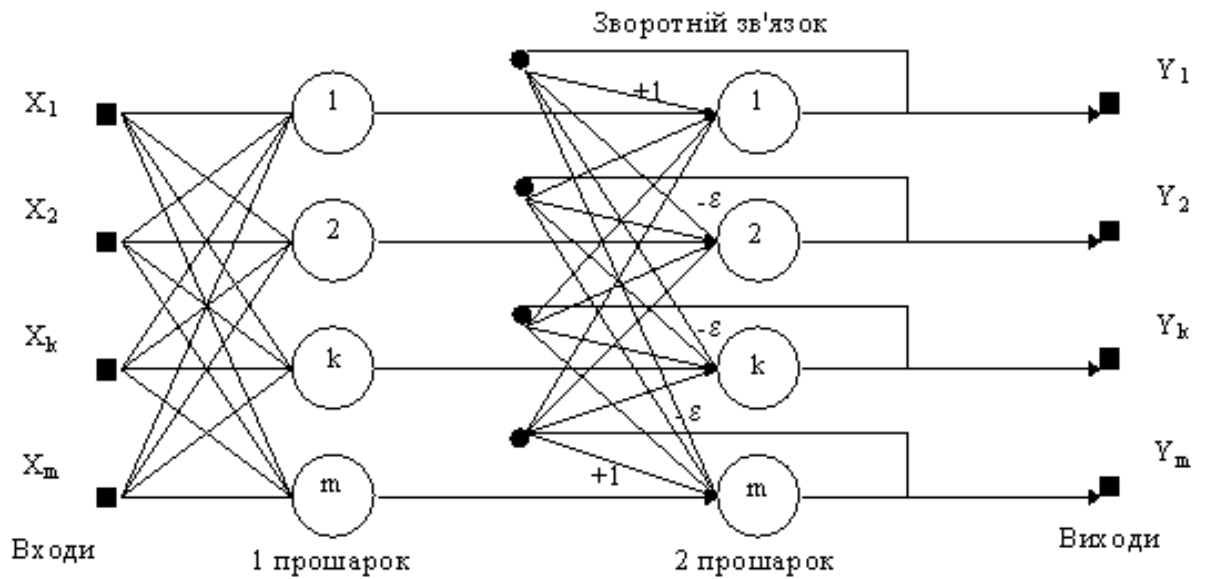


Рисунок. 1.7. Мережа Хемінга

1.3. Засоби обміну для штучних нейронних мереж

При виготовлення та впровадження змінних пристроїв потрібно враховувати принципи обміну інформацією. Обмін інформацією може бути синхронним і асинхронним. Вибір принципу обміну впливає не тільки на пропускну здатність, але й на фізичну довжину каналу обміну та кількість пристроїв, які можна підключити до нього.

Принцип синхронного зсуву

При використанні принципу синхронного обміну дані передаються блоками, обмеженими за часом. Передача даних заснована на узгодженні таймерів пристрою, що передає, і пристрою прийому. Сигнали синхронізації визначають певний інтервал часу, протягом якого зчитується інформація з каналу обміну. Для початку синхронізації використовуються спеціальні сигнали синхронізації. Фіксований протокол визначає розташування сигналів адреси та

даних щодо сигналів синхронізації. Практично не потрібна додаткова логіка, щоб вирішити, що робити далі, тому синхронна передача може бути швидкою та недорогою. Синхронна передача більш ефективно використовує канал обміну, підтримує високу ефективність, швидкість передачі даних і вбудований надійний механізм виявлення помилок. Основним недоліком є те, що через проблему спотворення тактового сигналу шина не може бути довгою під час синхронної передачі, а обладнання інтерфейсу є більш дорогим і складним.

Принцип асинхронного обміну

Асинхронна передача обмежена в часі. Через відсутність зв'язку між моментами доступу до передавального пристрою та моментами випуску готових даних - можливі подальші затримки в обміні даними. Сервісна інформація записується для вибору блоків даних на початку і в кінці кожного. Цей принцип полегшує підключення різних пристроїв на шині, і цей принцип має кращу стійкість до перешкод. Ви можете збільшити довжину шини, не турбуючись про спотворення сигналу синхронізації. До недоліків можна віднести те, що передається великий обсяг службової інформації і забезпечує низьку швидкість передачі даних порівняно з синхронною передачею. Асинхронний принцип передачі даних слід використовувати в системах, де обмін даними не є постійним і не вимагає високих швидкостей передачі.

Аналіз методів вирішення конфліктів у пристрої обміну

Конфлікти можуть виникати в пристрої обміну, коли кілька пристроїв у системі передають дані одночасно. Для вирішення подібних конфліктів використовуються різні методи, включаючи призначення кожному пристрою унікального пріоритету, призначення кожному пристрою фіксованих часових інтервалів та використання багатопортової пам'яті.

Розв'язання конфліктів багатопортової пам'яті

n-портова пам'ять має n незалежних наборів адрес, даних і шин керування, які забезпечують одночасний і незалежний доступ до пам'яті n пристроїв. Ця функція може значно спростити створення складних систем, але можуть

виникати конфлікти, і потрібно використовувати способи їх вирішення. Конфлікти виникають, коли кілька активних пристроїв звертаються до однієї комірки пам'яті під час запису даних з кількох портів або під час запису через один порт і читання через інший. Існують різні способи вирішення конфліктів синхронної та асинхронної багатопортової пам'яті (BP). Синхронний блок живлення передбачає синхронізацію активних пристроїв з одним системним таймером, тому вирішення конфліктів не вимагає використання додаткової логіки. Асинхронний блок живлення використовує: логіку арбітражу, семафори, запити на переривання та систему Master/Slave для вирішення конфліктів.

1.4. Засоби виготовлення та впровадження пристроїв обміну

Існують різні способи створення спільного пристрою залежно від схем розподілу пам'яті: спільна пам'ять і спільна пам'ять. Існує дві основні моделі обміну: обмін повідомленнями (для спільної пам'яті) і спільна пам'ять.

Спільна пам'ять

Розподіл спільного простору пам'яті між усіма вхідними портами дозволяє ефективно використовувати його відповідно до природи потоку вхідних даних. У цьому випадку всі головні обчислювальні вузли підключені до пам'яті через загальну шину. Перевага цього підходу полягає в тому, що обмін здійснюється шляхом запису/читання інформації з осередків спільної пам'яті, які доступні всім вузлам, і тому передача даних не займає часу. До недоліків можна віднести можливість конфліктів при одночасному зверненні до однієї комірки пам'яті, проблему повільного доступу до оперативної пам'яті та її обмеженої ємності, а також проблему масштабованості. Чим більше вузлів використовується, тим вищі витрати на розробку та нижча ефективність.

Існують системи з однорідним і гетерогенним доступом до пам'яті. У однорідних системах доступу всі вузли можуть отримати доступ до пам'яті

одночасно. Цей спосіб організації біржі найчастіше використовується для будівництва біржових приміщень. У системах з нерівним доступом до пам'яті кожному вузлу виділяється частина спільної пам'яті. Ця пам'ять має один адресний простір, тому ви можете отримати доступ до будь-якої комірки спільної пам'яті, використовуючи її адресу. Час доступу до модулів спільної пам'яті з різних вузлів різний.

Спільна пам'ять

У цьому випадку кожен вузол має доступ до фіксованої кількості виділених комірок пам'яті. Вузли з'єднані між собою каналами зв'язку для забезпечення обміну інформацією. Пакет, призначений для певного вихідного вузла, втрачається, якщо блок пам'яті, виділений цьому вихідному вузлу, переповнюється, хоча інші блоки в цей час можуть бути порожніми. Обмін в системі здійснюється шляхом надсилання посилань та отримання повідомлень. Ця схема розподілу пам'яті використовується для завдань, які вимагають малого обміну даними та великого обсягу пам'яті.

До переваг можна віднести можливість масштабування, тобто можна комбінувати велику кількість вузлів без істотного зниження ефективності їх взаємодії. Вартість системи буде пропорційна кількості вузлів. До недоліків можна віднести проблему обміну даними і високе енергоспоживання. Обмін даними в таких системах дуже повільний у порівнянні зі швидкістю обчислень (і з великими затримками). Тому ефективно вирішувати проблеми, які потребують інтенсивної заміни на таких системах, неможливо.

2. РОЗРОБКА ПРИСТРОЮ ОБМІНУ НА БАЗІ БАГАТОПОРТОВОЇ ПАМ'ЯТІ ДЛЯ НЕЙРОННИХ МЕРЕЖ

2.1. Формування вимог до пристрою обміну

Нейронні мережі використовуються для вирішення завдань класифікації або кластеризації багатовимірних даних. Основна ідея нейронних мереж - послідовне перетворення сигналу, що працює паралельно з елементарними функціональними елементами. Основою нейронної мережі є нейрон. Нейрон складається з 3 логічних блоків: входи, функції перетворення, вихід. Для кожного варіанта входу (вектора) функція перетворення нейронів генерує певний сигнал (вихід) (зазвичай скалярний) і передає його на входи інших нейронів мережі. Подаючи сигнали на входи деяких нейронів ззовні та записуючи виходи деяких нейронів як виходи мережі в цілому, ми отримуємо систему, яка виконує відображення $R_n \rightarrow R_k$, де n – вхідний розмір (зовнішня інформація), а k - вихідний розмір. Нейронні мережі розрізняються функцією перетворення в нейронах, внутрішньою архітектурою зв'язку між нейронами і методами налаштування (навчання).

Для того, щоб нейронна мережа виконувала необхідне відображення, потрібно особливим чином налаштувати функції нейронного перетворення та архітектуру внутрішнього з'єднання. Найчастіше архітектура підключення попередньо вибирається з відомих моделей і залишається незмінною під час роботи та налаштування мережі; налаштована функція перетворення [7].

Для побудови функції перетворення найчастіше використовується наступний підхід: усі елементи вхідного вектора підсумовуються з деякими ваговими коефіцієнтами (синаптичними ваговими коефіцієнтами), а потім сума стає нелінійною функцією (найпростіший випадок — порогове значення), а результатом є функція є вихід нейрона. У процесі налаштування мережі (навчання) ми коригуємо лише синаптичні ваги в нейронах. Для кожної архітектури нейронної мережі існують різні методи навчання. Основною

перевагою нейронних мереж є можливість розв'язування широкого класу алгоритмічно нерозв'язних задач або задач з нечіткими умовами. Теорія нейронних мереж (НС) забезпечує алгоритмічну основу для нейрокомп'ютерів. Нейронна мережа - це мережа з кінцевою кількістю шарів однотипних елементів - аналогів нейронів з різними типами зв'язку між шарами. Серед основних переваг НА в [2] відзначено: незмінність методів синтезу НА до розміру символічного простору та розміру НА, адекватність сучасних передових технологій, відмовостійкість у сенсі монотонних, а не катастрофічних змін якості проблеми в залежності на кількість зламаных елементів.

Розв'язання математичних задач у логічній основі нейронної мережі визначає теоретичні основи нейромематики. У [2] наступний етап вирішення практично будь-якої задачі в логічній основі нейронної мережі: створення вхідного сигналу НА, створення вихідного сигналу НА, створення потрібного вихідного сигналу НА, створення сигналу помилки та оптимізації функціональності, створення структура нейронної мережі, розробка алгоритму надзвичайних ситуацій, еквівалентний процес розв'язання задач у логічній основі нейронної мережі, проведення дослідження процесу розв'язання задач. Все вищезазначене робить проектування сучасних систем управління з використанням нейронних мережевого доступу і, виходячи з логічної основи нейронних мереж, одним з найбільш перспективних напрямків виготовлення та впровадження багатоканальних і багаторазово пов'язаних систем управління.

Як зазначалося, нейрокомп'ютер – це комп'ютерна система з архітектурою MSIMD, тобто з паралельними потоками одних і тих самих команд і кількома потоками даних. Давайте розберемося, яка сторона нейрокомп'ютерів відноситься до паралельних комп'ютерів. Сьогодні можна виділити три основні напрямки розвитку комп'ютерних систем з масовим паралелізмом:

Напрями розвитку обчислювальних систем з масовим паралелізмом

таблиця 1

№ Опис маршруту

1 OSMP на основі каскадного підключення універсальних мікропроцесорів SISD, SIMD, MISD Елементна база - універсальні процесори RISC або CISC: Intel, AMD, Sparc, Alpha, Power PC, MIPS тощо.

2 На основі процесорів з реалізація паралельних алгоритмів на апаратному рівні Елементна база - процесори DSP: TMS, ADSP, Motorola

3 OSMP на основі спеціалізованої елементної бази Елементна база від спеціалізованих однорозрядних процесорів до нейрочіпів

Для кожної з областей сьогодні існують рішення, які реалізують певні парадигми нейронних мереж. Для наочності будемо вважати, що нейронних мережеві системи, реалізовані на апаратних платформах першого ряду (хоч і багатопроцесорних), будемо називати нейроімуляторами – системами, які виконують типові нейрооперації на програмному рівні (зважене підсумовування та нелінійні перетворення).

Системи нейронних мереж, реалізовані на апаратних платформах другого та третього напрямків у вигляді плат розширення стандартних комп'ютерних систем (1-й напрямок) – ми будемо називати нейроприскорювачами – зважене підсумовування зазвичай реалізується апаратно, наприклад на основі перехресних фільтрів,) та реалізовані системи На третьому напрямі апаратну платформу у вигляді функціонально завершених обчислювальних пристроїв потрібно віднести до нейрокомп'ютерів (всі операції виконуються в логічній основі нейронної мережі). Підсумовуючи, можна констатувати, що нейрокомп'ютери можна безпечно віднести до високопаралельних обчислювальних систем (архітектура MSIMD), реалізованих на основі спеціалізованої елементної бази, орієнтованої на виконання нейронних мережевих операцій у логічній основі нейронної мережі [15].

Нейрокомп'ютерні системи другого та третього напрямів базуються на впорядкованих кристалах (ASIC), вбудованих мікроконтролерах (mC), універсальних процесорах (GPP), програмованій логіці (FPGA), перетворювачах, процесорах цифрових сигналів (DSP) та неврозах [1]. Крім того, використання

цих та інших сьогодні дозволяє реалізовувати нейрокомп'ютери в режимі з опрацюванням у даний час, але найбільше застосування в виготовлення та впровадження нейрокомп'ютерів знайшли FPGA, DSP і звичайно нейрочіпи.

DSP (Digital Signal Processor-digital signal processor) з потужною обчислювальною структурою дозволяє реалізувати різноманітні алгоритми обробки інформаційних потоків. Відносно низька вартість і передові засоби розробки програмного забезпечення дозволяють легко використовувати їх у побудові комп'ютерних систем з масовим паралелізмом.

Швидкий перехід сучасних систем управління на цифрові стандарти призвів до потреби обробки досить великих обсягів інформації з високою швидкістю. Складна обробка та фільтрація сигналів, наприклад, розпакування стиснених аудіо- та відеоданих, маршрутизація інформаційних потоків тощо, вимагає використання високопродуктивних комп'ютерних систем. Такі системи можуть бути реалізовані на різноманітних компонентах, але найпоширенішими є пристрої, що використовують цифрові сигнальні процесори та FPGA.

DSP - як елементарна основа нейрокомп'ютерів.

Протягом кількох десятиліть цифрові сигнальні процесори (ЦСП) були базовою основою для побудови нейроприскорювачів і логічної схеми загальносистемного керування нейрокомп'ютерами.

Який DSP можна використовувати для виготовлення та впровадження нейроакселераторів? - Але практично будь-який, все залежить від можливостей, ми розберемо лише депутатів трьох великих виробників: Analog Devices, Motorola і Texas Instruments, з точки зору побудови на їх комп'ютерних системах.

Вибір процесора є багатокритеріальним завданням, але перевагу Analog Devices [7] для додатків, які вимагають великої кількості математичних обчислень (наприклад, фільтрація цифрових сигналів, обчислення кореляційних функцій тощо), слід відзначити, оскільки їх продуктивність у подібних завдань вище, ніж у процесорів Motorola і Texas Instruments. У той же час для завдань, що вимагають інтенсивного обміну із зовнішніми пристроями

(багатопроесорними системами, різними контролерами), доцільніше використовувати процесори Texas Instruments [8], які мають високошвидкісні інтерфейсні підсистеми. Motorola є лідером у виробництві сигнальних мікропроцесорів, більшість з яких є недорогими, високопродуктивними 16- і 24-розрядними мікропроцесорами з фіксованою точкою. Розширені комунікаційні можливості, наявність достатньої кількості пам'яті на чіпі для даних і програм, можливість захисту програми від несанкціонованого доступу, підтримка режиму енергозбереження роблять ці мікропроцесори привабливими для використання не тільки як спеціалізовані комп'ютери, а й як блоки управління в побутовій електроніці. , адаптивні системи фільтрації тощо.

На виконання обмінного пристрою впливає архітектура зв'язку нервової системи, зокрема наявність зворотного зв'язку. Якщо нейросистема має структуру послідовних шарів, де кожен нейрон отримує інформацію тільки від попередніх шарів, то обробка інформації одностороння – без зворотного зв'язку. Вхідний сигнал обробляється послідовністю шарів і гарантується, що певна кількість циклів, що дорівнює кількості шарів, отримає відповідь. Для цього потрібно більше нейронів, ніж нейросистема зворотного зв'язку, а отже, більше каналів даних до пристрою обміну. У нейросистемі зворотного зв'язку нейрони посилаються частіше, ніж у нейросистемі без зворотного зв'язку, і тому обмінний курс вищий, ніж у нейросистемі без зворотного зв'язку.

Таким чином, ми можемо виділити наступні основні вимоги до змінних пристроїв:

- забезпечення високої продуктивності передачі даних;
- забезпечення високої загальної потужності;
- забезпечення роботи обмінного обладнання з великою кількістю входів і виходів;
- забезпечення можливості перемикання в обидві сторони;
- мінімізація затримки передачі;
- забезпечення безконфліктного обміну даними;

- забезпечення роботи із зовнішніми пристроями на різних швидкостях;
- можливість змінити час доступу до ПКП;
- можливість паралельного підключення потрібної кількості зовнішніх пристроїв;
- забезпечення високої ефективності використання обладнання.

Очевидно, що деякі з цих вимог є суперечливими, тому при побудові змінного обладнання потрібно враховувати їх оптимальне співвідношення.

Щоб забезпечити виконання цих вимог, розробка змінного обладнання повинна здійснюватися з урахуванням наступних принципів:

- модульність, що включає розробку простих компонентів обміну у вигляді готових модулів;
- змінний склад пристрою, що забезпечує гнучкість виготовлення та впровадження;
- конвеєрний і просторовий паралельність, що збільшує швидкість обміну;
- узгодженість інтенсивності даних від компонентів нейросистеми з інтенсивністю доступу до пристрою обміну, що підвищує ефективність виготовлення та впровадження.

2.2. Вибір принципів та елементарної бази для виготовлення та впровадження пристрою обміну

Вибір принципів конструкції пристрою обміну залежить від кількості каналів обміну, швидкості передачі даних і частоти їх прийому, потрібного обмінного курсу, максимальної частоти доступу до пам'яті та максимального обмінного курсу для кожного пристрою. Потрібно враховувати особливості нейросистеми. Зокрема, на реалізацію обмінного обладнання впливають способи функціонування нейросистеми. Є два режими роботи: режим навчання та робочий. У режимі навчання забезпечується налаштування певних параметрів нейросистеми - знаходження оптимальних значень шкал. Це означає, що

потрібно забезпечити велику кількість передачі даних. Це висуває додаткові вимоги до змінного пристрою. Зокрема, потрібно забезпечити двосторонню високошвидкісну передачу даних. Ваги встановлюються тільки під час навчання і в робочому режимі ваги фіксуються. Тому в цьому режимі кількість передачі даних значно менша в порівнянні з режимом навчання.

Паралельна структура пам'яті (РАМ) може бути представлена таким чином:

$$S_{III} = \langle l, r, a, m \rangle,$$

де l - набір модулів пам'яті; r - спосіб перемикання модуля пам'яті - зовнішній пристрій; i - метод адресації пам'яті; m - ширина доступу (кількість входів-виходів) до пам'яті.

Спосіб комутації r модуля пам'яті - зовнішні пристрої та маршрутизація даних забезпечується комутаційною мережею [38,133]. Характеристики комутаційної мережі істотно впливають на властивості ПП. У більшості випадків для з'єднання модулів пам'яті та зовнішніх пристроїв використовуються такі комутаційні мережі: ресурси, що виділяються за часом; просторове поділ; горіх; багаторівневий. Одним з головних параметрів таких мереж є пропускна здатність і підключення. Комутаційні мережі можуть передбачати будь-яку можливість комутації входів з виходами або вони обмежені кількома варіантами.

За способом адресації комірок ПП можна поділити на адресні, асоціативні та неадресовані. В адресі РР до комірки пам'яті звертаються відповідно до адреси $[A_i] iD_i$, де A_i - адреса комірки, D_i - вона буде містити осередки пам'яті за адресою A_i . Вибірка даних з асоціативного програмного забезпечення виконується на певних підставах, а з безадресного програмного забезпечення - шляхом звернення до попередньо визначеної комірки пам'яті та подальшої передачі даних.

Залежно від алгоритмів усунення несправностей і вимог додатків можна синтезувати багато паралельних структур пам'яті (РР), які відрізняються за способом проектування носія даних, мережею комутації, розташуванням і пошуком інформації, а також технічними параметрами [193-208]. Завдання опису та розробки ряду можливих структур ПП є складним, доцільно розробити та вивчити основні структури паралельної пам'яті, на основі яких можна було б синтезувати пам'ять для конкретних застосувань.

Аналіз паралельної структури пам'яті [196,197] дозволив виділити три основні паралельні структури пам'яті:

- з розподілом за часом ресурсів пам'яті;
- на основі кількох модулів пам'яті;
- на основі комірок пам'яті з багатьма входами.

Перша базова структура ПП з тимчасовим виділенням ресурсів пам'яті показана на рисунку 4.1, де ВЛА - блок адреси; Пристрій керування ПК [197]. Основним елементом такого програмного забезпечення є модуль пам'яті, який є звичайною класичною пам'яттю. При такій організації зв'язок між зовнішніми пристроями та модулем пам'яті здійснюється через мережевий комутатор. У більшості випадків для доступу до шини модуля пам'яті використовується алгоритм фіксованого часового інтервалу [199]. Цей алгоритм циклічно дає кожному пристрою однаковий часовий інтервал t_c , незалежно від того, потрібен пристрій доступу до пам'яті чи ні. Мінімальний час доступу до шини модуля пам'яті визначається часом циклу читання/запису в пам'ять. Цей доступ до шини пам'яті дозволяє ПК синхронізувати доступ до шини модуля пам'яті асинхронно запущених зовнішніх пристроїв. Ця структура використовує паралельне послідовне перетворення, яке називається масштабуванням паралельного доступу до спільного ресурсу - пам'яті. Це масштабування широко використовується для налаштування ширини доступу m до обмінного

курсу V . Максимальна швидкість V , якою зовнішні пристрої можуть обмінюватися пам'яттю:

$$V = \frac{1}{m \times t_u}$$

Однією з основних переваг цієї структури є гнучкість і легке збільшення або зменшення кількості пристроїв, підключених до шини модуля пам'яті. Апаратні витрати на реалізацію ПП з розподілом ресурсів за часом становлять:

$$W_1 = W_{\text{модП}} + mW_{\text{БлА}} + W_{\text{ПК}} + W_{\text{КМІ}},$$

де $W_{\text{модП}}$, $mW_{\text{БлА}}$, $W_{\text{ПК}}$, $W_{\text{КМІ}}$ - апаратні витрати відп. для виготовлення та впровадження модуля пам'яті, блоку адреси, пристрою керування та комутаційної мережі з розподілом ресурсів за часом.

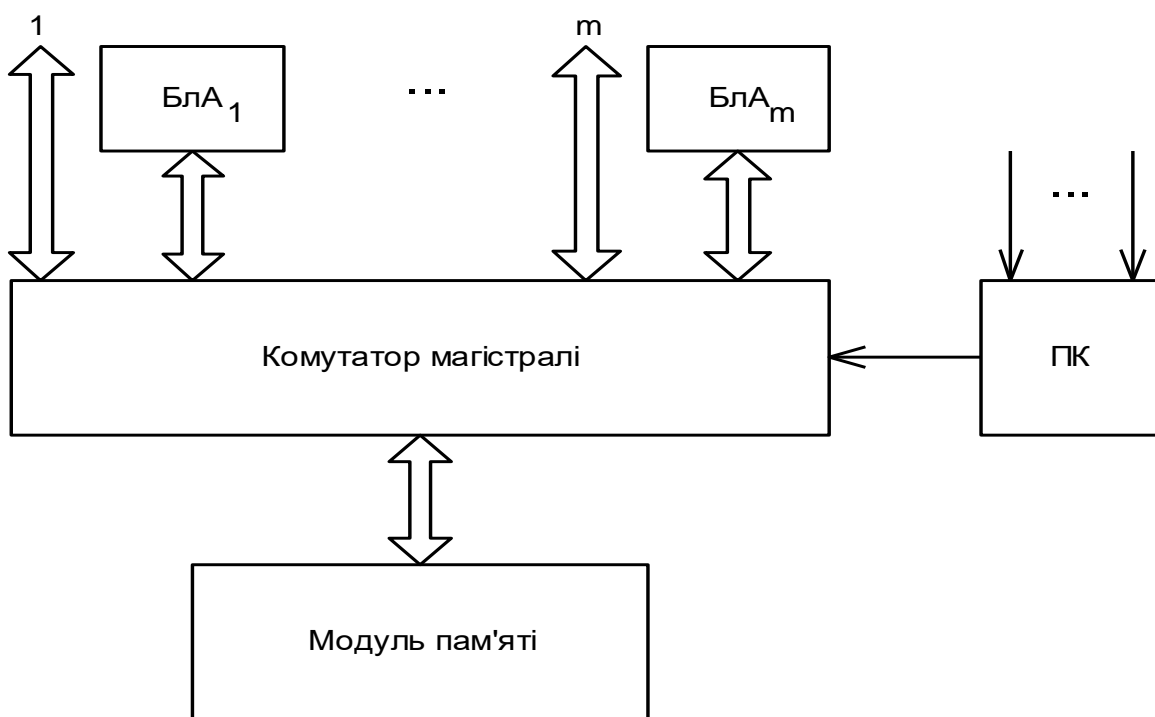


Рисунок 2.1. Базова структура паралельної пам'яті з часовим розподілом ресурсів пам'яті

Друга базова структура паралельної пам'яті на основі набору модулів пам'яті показана на рисунку 2.2, де ВU — блок управління. Основу структурної

організації такої пам'яті складають: L модулі пам'яті, блок адреси та комутаційна мережа. Ємність такого ПП дорівнює:

$$Q = \sum_{i=1}^L Q_i,$$

де Q_i - ємність i -го модуля пам'яті. У такому програмному забезпеченні комутаційна мережа реалізована на комутаторах з просторовим розподілом, які забезпечують підключення будь-якого входу до будь-якого виходу або підмножини виходів. Кожне з'єднання входів і виходів здійснюється за допомогою різного пристрою, комутатор з просторовим розподілом реалізований на мультиплексорах, кількість яких дорівнює кількості модулів пам'яті. Щоб забезпечити повне підключення графа, кожен вхід РР повинен бути підключений до відповідних входів усіх мультиплексорів. Комутаційна мережа управляється інформацією з виходу ВU.

Апаратні витрати на впровадження програмного забезпечення на основі L модулів пам'яті визначаються за формулою:

$$W_1 = LW_{\text{модП}} + W_{\text{БлА}} + W_{\text{ПК}} + W_{\text{КМ2}},$$

де $W_{\text{КМ2}}$ - витрати на апаратні засоби для виготовлення та впровадження комутаційної мережі з просторовим розподілом. Час доступу програмного забезпечення на основі модуля пам'яті L визначається наступним чином:

$$t_{\text{ППП}} = t_A + t_{\text{ц}} + t_{\text{КМ2}} + t_n,$$

де t_A , $t_{\text{ц}}$, $t_{\text{КМ2}}$, t_n - часи відповідно до генерації адреси, циклу доступу до модуля пам'яті, затримки даних комутаційної мережі з просторовим розподілом, налаштування комутаційної мережі.

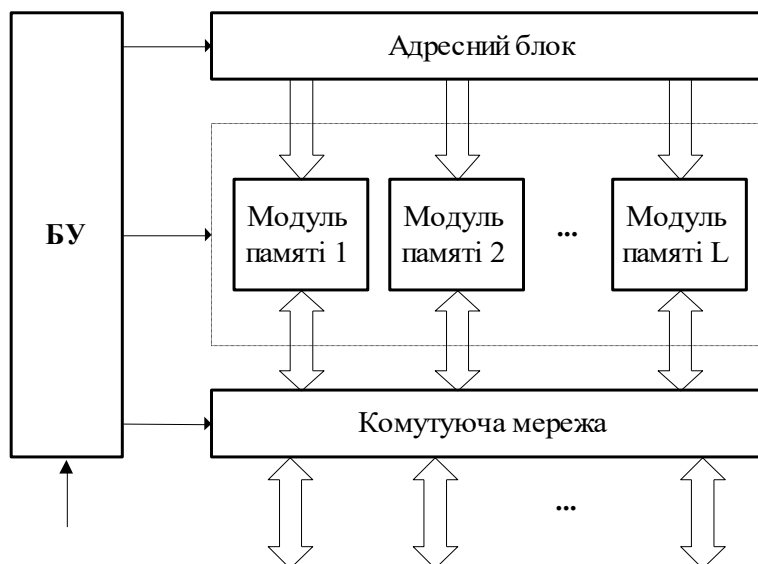


Рисунок 2.2. Базова структура паралельної пам'яті на основі множини модулів пам'яті

Показано третю базову структуру ПП на основі комірок пам'яті високого входу (рисунок 2.3), де КР — комірка пам'яті, LS — декодер, КМ — комутаційна мережа. Основою структурної організації такого середовища зберігання є набір комірок пам'яті, доступ до яких є довільним. Характерною особливістю одновимірного середовища з високим рівнем входу є можливість одночасного за який здійснюється зчитування інформації з однієї комірки пам'яті КРі на кілька виходів. Комутаційна мережа ПП реалізована на основі матричного комутатора, що забезпечує підключення m до N входів. Особливість матричного комутатора полягає в тому, що при відсутності конфліктів всі необхідні підключення забезпечуються одночасно. Конфлікти виникають, коли два або більше зовнішніх пристроїв вимагають доступу до одного ЦП.

Швидкість такого ПП визначається часом доступу до КП, який дорівнює:

$$t_{цПП_3} = t_A + t_{Дш} + t_{БЕ},$$

де $t_{Дш}$, $t_{БЕ}$ - часи затримки інформації на декодері та буферному елементі. Слід зазначити, що ПП на основі такого середовища зберігання дозволяє вибрати

будь-яку паралельну структуру даних. Основним недоліком високопродуктивного ПП на основі осередків пам'яті є високі витрати на апаратне забезпечення, пов'язані з реалізацією буферів, які лежать в основі комутаційної мережі. Апаратні витрати на впровадження такого програмного забезпечення визначаються за формулою:

$$W_{ППЗ} = NW_{КП} + W_{КМЗ} + W_{БЛА},$$

де $W_{КП}$, $W_{КМ}$ – вартість обладнання на комірку пам'яті та комутаційну мережу матриці.

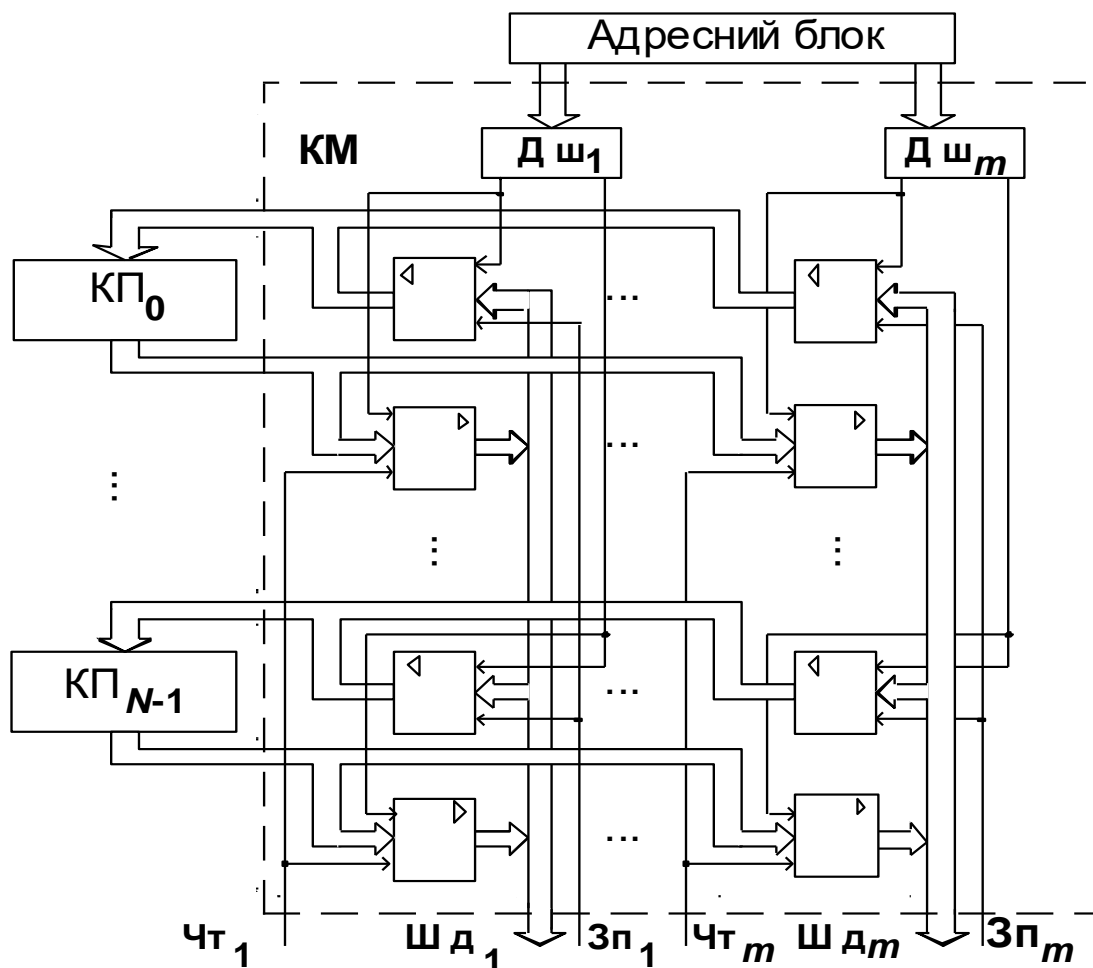


Рисунок 2.3. Базова структура паралельної пам'яті на основі багатовходових комірок пам'яті

Кількість значень, що надходять на вхід нейросистеми, також важлива. Це залежить від її способу навчання. За методами навчання розрізняють такі види: «з учителем» і «без вчителя». Такий спосіб навчання, де фактичний вихід нейросистеми порівнюється з еталонним, називається навчанням «вчителя». У такому навчанні вхідні значення та очікувані вихідні значення, які є елементом навчальної вибірки, надходять на вхід. При навчанні «без вчителя» значення очікуваного результату не враховується. Початкові значення такої нейросистеми формуються тільки під впливом вхідних значень.

2.3. Розробка структури пристрою обміну на базі багатопортової пам'яті

Структурна схема змінного пристрою показана на малюнку. 2.4. Пристрій заснований на двопортовій пам'яті, забезпечує обмін даними між процесором управління і процесором нейросистеми, і в той же час може взаємодіяти з одним із процесорів нейросистеми з блоком пам'яті. Сигнали взаємодії між процесором управління та пристроєм обміну наведені в табл.5.1.

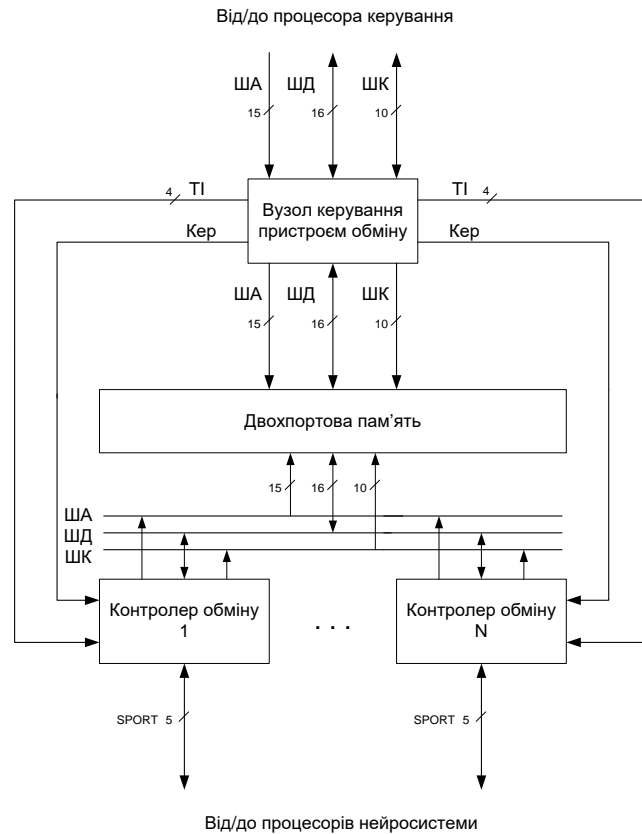


Рисунок. 2.4. Структурна схема пристрою обміну

З сигналів від процесорів нейросистеми, що надходять через інтерфейс SPORT, відповідний сигнал ПКП формує двоходовий сигнал доступу до пам'яті (шина Kerr), який надходить на блок ПКП. Якщо в даний момент немає запитів на доступ до пам'яті від інших процесорів нейросистеми, панель керування генерує сигнал дозволу доступу (шина Kerr) і тактову послідовність для генерування керуючих сигналів блоку пам'яті, приєднання даних, адреси та буферів керуючих сигналів.

2.4. Розробка модуля контролераобміну

Основними функціями обмінного контролера є:

- обмін даними з процесорами нейросистеми через інтерфейс SPORT;

- формування адресної шини, шини даних та шини для управління багатопортовою пам'яттю;
- перетворення паралельного інтерфейсу в послідовний і навпаки.

Для виконання цих завдань на малюнку. 2.5. показана схема обмінного контролера.

Основними функціональними блоками контролера є: блок синхронізації та управління, блоки перетворення інтерфейсу та схема формування сигналу управління багатопортовою пам'яттю, де SD BPP – порт даних багатопортової пам'яті, SH BPP – адреса багатопортової шини пам'яті, SHK BPP – шина управління багатопортовою пам'яттю.

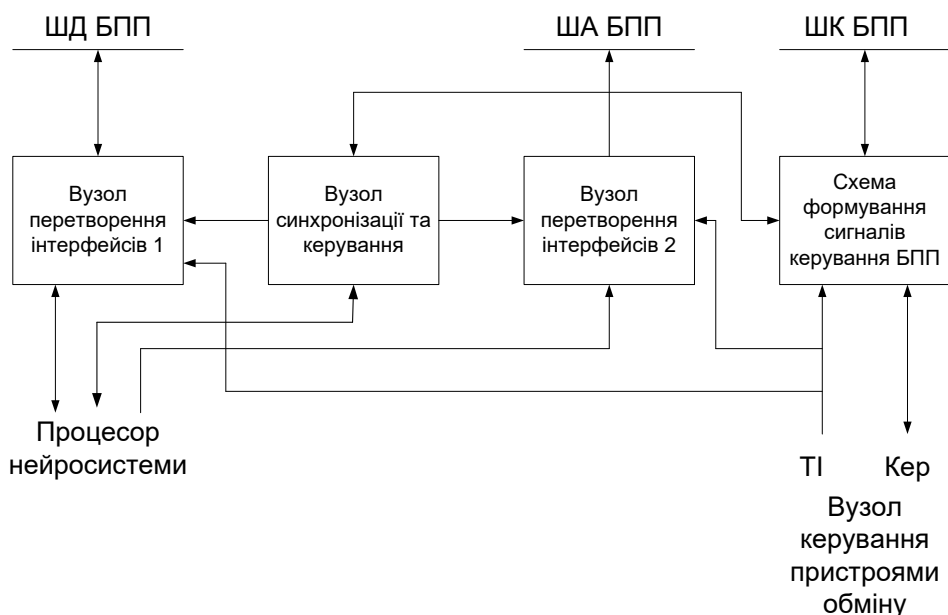


Рисунок. 2.5. Блок-схема контролера обміну

1278 / 5000

Результати перекладу

Вузол перетворення інтерфейсу 1 перетворює дані з послідовного коду (інтерфейс процесора нейросистеми SPORT) у паралельний (SD BPP) і навпаки.

Блок перетворення інтерфейсу 2 перетворює дані з послідовного коду (SPORT інтерфейс процесора нейросистеми) в паралельний (SA BPP).

Схема генерування керуючих сигналів БП забезпечує синхронізацію часу керуючих сигналів пам'яті при отриманні керуючих сигналів (Кер) і тактових імпульсів (ТИ). Сигнали керування дозволяють вибрати режим роботи. Тактові імпульси формують часову послідовність процесорів нейросистеми, які підключаються до багатопортової пам'яті.

Вузол синхронізації та управління формує імпульси синхронізації та керуючі сигнали, які надходять на інші вузли блоку керування обміном. Ці сигнали генеруються на основі сигналів, отриманих через інтерфейс SPORT від процесора нейросистеми та схеми генерації керуючого сигналу ВРР.

Схема працює наступним чином: з ПКП на вхід схеми керуючого сигналу на БПП надходить керуючий сигнал (Кер), який визначає режим і тактові імпульси (ТИ), які синхронізують процесор, нейросистему і мультипорт. пам'ять. Послідовний канал отримує адресу та дані від процесора нейросистеми (у режимі запису в багатопортову пам'ять). Коли ці сигнали приймаються на вузлах перетворення інтерфейсу, створюється адресна шина і шина даних блоку ВРР.

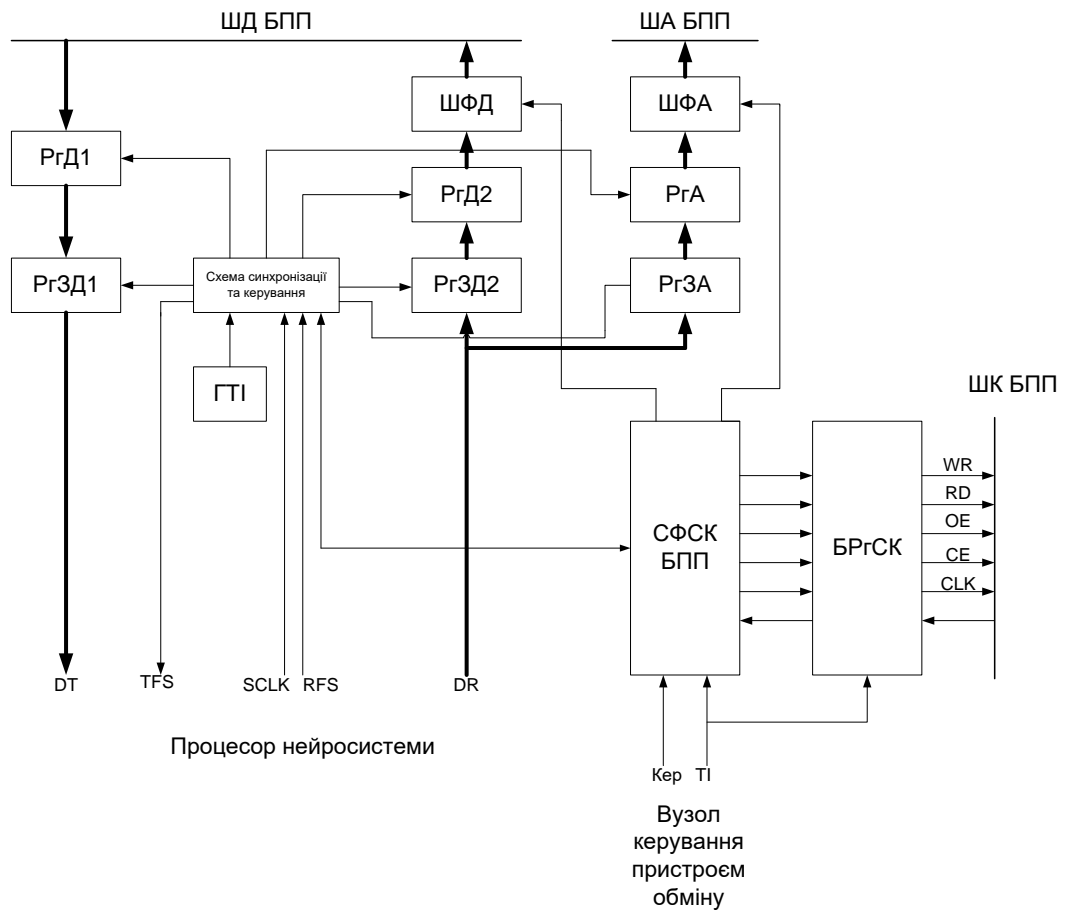


Рисунок. 2.6. Функціональна схема контролера обміну

На малюнку. 2.6. наведена функціональна схема контролера обміну, де RgD1, RgD2 — регістри даних; SFD - драйвер даних шини; RgZD1, RgZD2 - регістри зсуву даних; SFA - формувач адреси шини; RgA - адресний реєстр; RgZA - регістр зсуву адреси; GTI - тактовий генератор; SFSK BPP - схема формування сигналу управління багатопортовою пам'яттю; БРСК - регістр вирівнювання керуючого сигналу.

Вузол перетворення інтерфейсу 1 (див. рисунок 5.4.) Реалізовано на основі RGD1, RGD2, RGD1, RGD2 і СФД. Регістри Rg1 і RgZD1 використовуються для за який здійснюється зчитування даних з багатопортової пам'яті. Регістри RGD2 і RGD2 і SFD використовуються для запису даних у багатопортову пам'ять. SFD забезпечує підключення до шини даних при виборі конкретного процесора нейросистеми. Підключення здійснюється до сигналу, що надходить від SFSK BPP (цей сигнал генерується сигналом TI).

Інтерфейс перетворення вузла 2 реалізований на основі PgZA, PgA і SHFA. Генерація сигналу в ТК ТК подібна до генерації сигналу в ТК ТК.

Вузол синхронізації та управління отримує сигнали від процесора нейросистеми та від SFSK BPP. Крім того, для організації роботи використовуються тактові сигнали від GTI. На основі цих сигналів і керуючого сигналу (генерованого у вузлі SFSK BPP) формуються сигнали запису в регістрах RgD1, RgD2 і RgA, а також серія тактових імпульсів для організації зсуву даних в RgZD1, RgZD2 і RgZA. реєстри. .

Схема генерації керуючого сигналу BPP фактично включає в себе схему генерації багатопортової пам'яті та управління (SFSK BPP) і буферний регістр (BRgSK). Буферний регістр використовується для підключення сигналів схеми до SC BPP. Час підключення визначається сигналами Керг і TI, що надходять від контрольної панелі. Залежно від типу використовуваної багатопортової пам'яті список керуючих сигналів може відрізнятися.

У приймальній частині послідовного порту прийом даних ініціює сигнал синхронізації прийому кадрів (RFS). Послідовний потік даних (DR), отриманий від процесора нейросистеми, записується порозрядно в адреси прийому та регістри зсуву даних. Для бітової синхронізації використовуються годинники SCLK. Після отримання наступного слова воно перезаписується в регістри прийому даних RgD2 та адреси RgA.

Запис в регістр передачі даних (RGD1) готує послідовний порт до передачі даних. Початок передачі даних супроводжується сигналом синхронізації частоти кадрів (TFS). Потім слово з регістра передачі даних (PgD1) записується в регістр передачі зсуву (PgD1). Дані з регістра зсуву передачі біт надсилаються в процесор нейросистеми. Тактовий сигнал SCLK використовується для послідовної синхронізації послідовних даних (DT).

3. РОЗРОБКА ПРОГРАМ ДЛЯ МОДЕЛЮВАННЯ ПРИСТРОЮ ОБМІНУ НА БАЗІ БАГАТОПОРТОВОЇ ПАМ'ЯТІ

3.1. Синтез багатопортової пам'яті для комп'ютерних систем з опрацюванням у даний час

Основною метою синтезу CSP BPP в реальному часі є отримання структури, орієнтованої на VLSI, яка має високу ефективність використання пристрою та забезпечує інтенсивний доступ до пам'яті. Вихідною інформацією для синтезу комп'ютерних систем BPP в реальному часі є:

- узгоджена блок-схема алгоритму вирішення проблеми;
- кількість каналів передачі даних;
- розмір файлів даних;
- розрядні дані;
- максимальна частота доступу кожного каналу до пам'яті;
- основні структури паралельної пам'яті;
- техніко-економічні вимоги та обмеження.

На основі аналізу узгодженої блок-схеми алгоритму розв'язання задачі та вихідної інформації визначаються алгоритми формування адресних послідовностей, ємність та інтенсивність доступу до паралельної пам'яті. Інтенсивність доступу до паралелі визначається наступним чином:

де t_{PP} - ширина паралельного доступу до пам'яті, p_k - паралельний канал доступу до пам'яті, F_{PP} - частота паралельного доступу до пам'яті. З формули (с) видно, що інтенсивність доступу до паралельної пам'яті залежить як від ширини та частоти доступу, так і від бітрейту каналів. Вибір та оцінка синтезованих конструкцій БПП здійснюється за допомогою інтегрального параметра ефективності обладнання, який визначається наступним чином:

$$E = \frac{m_{\text{ПП}} n_{\text{к}} Q}{W_{\text{ПП}} \times t_{\text{цПП}}},$$

де $W_{\text{ПП}}$ - вартість обладнання для паралельної виготовлення та впровадження пам'яті; CCI - ширина паралельного доступу до пам'яті; Q - ємність паралельної пам'яті; $t_{\text{цПП}}$ - паралельний цикл доступу до пам'яті. Цей інтегрований параметр пов'язує ємність, інтенсивність доступу та витрати на обладнання для виготовлення та впровадження паралельної пам'яті та оцінює внесок кожного елемента схеми (клапана) у зберігання інформації та забезпечення інтенсивності доступу до пам'яті.

У синтезі БПП потрібно враховувати дві діалектичні протилежності – універсалізацію та спеціалізацію. Універсалізація розглядається як засіб зменшення обсягу, витрат на проектування та виготовлення, а спеціалізація - як засіб досягнення високої ефективності пристрою шляхом адаптації паралельної структури пам'яті до алгоритмів обробки та структури даних. Розглянемо методи паралельного синтезу пам'яті на основі базових структур з елементами універсалізації [7].

Інтенсивність введення даних, період (цикл) доступу до носія інформації та алгоритми формування адресних послідовностей при зверненні до пам'яті визначають способи проектування паралельної пам'яті. Для розробки паралельної пам'яті ОС у режимі з опрацюванням у даний час ми використовуємо просторово-часовий метод для відображення структури потоків даних у послідовному алгоритмі блок-схеми, щоб вирішити задачу щодо базових структур паралельної пам'яті. Виходячи з вимог конкретних додатків, вибирається структура ВРР та спосіб регулювання інтенсивності доступу до даних та пам'яті.

Інтенсивність доступу до ВРР, яка була визначена на основі аналізу узгодженої блок-схеми алгоритму розв'язання задачі, має вирішальне значення для вибору базової структури паралельної пам'яті та методу доступу до

середовища зберігання. ВРР і метод розподілу часу ресурсів носіїв даних використовуються для синтезу ВРР в наступних умовах:

$$F_{\text{ПР}} m_{\text{ПР}} n_{\text{ПР}} \leq n_{\text{ЗС}} F_{\text{ЗС}}, \quad (3.2)$$

де $F_{\text{ПР}}$ - частота доступу пристрою до паралельної пам'яті; $F_{\text{ЗС}}$ - частота доступу до носія інформації; $m_{\text{ПР}}$ - ширина паралельного доступу до пам'яті, $n_{\text{ПР}}$ - бітовий канал паралельного доступу до пам'яті; $n_{\text{ЗС}}$ — бітовий розмір носія. Умова виконання (5.2) дає змогу синтезувати паралельну пам'ять з інтенсивністю доступу, що забезпечує роботу в реальному часі.

Метод доступу, заснований на послідовно-паралельному перетворенні та стратифікації носія інформації, використовується в наступному випадку:

$$F_{\text{ПР}} m_{\text{ПР}} n_{\text{ПР}} \geq n_{\text{ЗС}} F_{\text{ЗС}}. \quad (3.3)$$

Виконання умови (5.3) вимагає зміни параметрів середовища зберігання для забезпечення потрібної інтенсивності доступу до паралельної пам'яті. Одним із способів збільшення інтенсивності доступу до паралельної пам'яті є використання послідовного в паралельне перетворення та збільшення розміру бітів носія даних шляхом розшарування K незалежних сегментів (модулів пам'яті). Значення K визначається співвідношенням:

$$K = \left\lceil \frac{F_{\text{ПР}} n_{\text{ПР}} m_{\text{ПР}}}{F_{\text{ЗС}} n_{\text{ЗС}}} \right\rceil,$$

де $\lceil \cdot \rceil$ — знак округлення до більшого цілого. Коли носій інформації розшаровується на K незалежних сегментів, це дозволяє збільшити бітовий розмір носія в K разів.

У разі:

$$F_{III} m_{III} n_{III} > n_{3C} F_{3C} > F_{III} n_{III}$$

Для побудови паралельної пам'яті використовується метод розподілу ресурсів і доступу за часом на основі послідовно-паралельного перетворення середовища зберігання. У цьому випадку потрібно стратифікувати його середовище зберігання, щоб забезпечити необхідну інтенсивність доступу до паралельної пам'яті. Кількість сегментів, необхідних для забезпечення потрібної інтенсивності доступу, визначається за такою формулою:

$$K \geq \left\lceil \frac{t_{ц3C} \times m_{III}}{T_{\min}} \right\rceil$$

Використовуючи комбінацію методу розподілу часових ресурсів і підходу, заснованого на послідовно-паралельному перетворенні середовища зберігання, можна синтезувати ВРР з високою ефективністю пристрою

3.2. Розробка алгоритму роботи пристрою обміну на базі багатопортової пам'яті

Коли ви використовуєте комп'ютер для інтенсивних обчислень з файлами даних, вам потрібно створити пам'ять з новими властивостями, які не завжди мають існуючі типи пам'яті. Розглянемо докладніше вимоги до пам'яті сучасного комп'ютера. Сучасні комп'ютери паралельні. Таким чином, пам'ять повинна бути в першу чергу паралельною і забезпечувати одночасний безконфліктний доступ до даних з багатьох каналів. Тільки наявність паралельної пам'яті відкриває шлях до створення принципово нових

комп'ютерів. Таким чином, виникли перші два критерії ефективності пам'яті: 1) пам'ять має бути багатоканальною (багатопортовою); 2) пам'ять повинна мати безконфліктний доступ до даних з багатьох портів. Крім того, при виконанні інтенсивних обчислень на полях даних потрібно забезпечити зберігання вхідних полів даних, при зчитуванні раніше отриманих полів виконувати операції реорганізації полів і сортувати дані по полях. Аналіз цих вимог показує, що потрібно реалізувати ще кілька критеріїв продуктивності пам'яті: 3) пам'ять повинна забезпечувати одночасний запис даних, що надходять на її входи, і за який здійснюється зчитування на виходах раніше записаних даних. У даному випадку можна говорити про два випадки - одночасний запис і за який здійснюється зчитування як наборів даних, так і окремих даних; 4) пам'ять повинна мати можливість виконувати операції реорганізації полів і сортування полів.

Для прискорення взаємодії комп'ютерних пристроїв із пам'яттю та зниження апаратної вартості елементів доступу до даних, особливо для масових сховищ, з якими збільшується вартість елементів доступу та зменшується їх швидкість, також постає питання зниження вимог до числових даних, у першу чергу. немає потреби в подвійному доступі під час запису та читання числа, що вимагає зберігання інформації про розташування даних на додаток до збільшення часу пошуку. Розгляньте можливість надання доступу до кожної комірки з введення та виводу пам'яті або з комбінованого введення та виводу пам'яті, що є дорогим і сповільнює пам'ять. Ця проблема, зокрема, стосується пам'яті, яка відповідає першому критерію, тобто відповідає першому критерію, тобто багатоканальної пам'яті, де потрібно забезпечити одночасний доступ з кожного каналу до кожної комірки пам'яті, що є проблемою, особливо з великою пам'яттю. потужності. Таким чином, є три додаткові критерії продуктивності пам'яті: 5) пам'ять не повинна вимагати доступу до даних і запису та читання; 6) організація пам'яті не повинна забезпечувати необхідність звернення до комірки

пам'яті при записі та читанні числа; 7) Організація пам'яті не повинна забезпечувати необхідність зберігання інформації про місцезнаходження

Перед записом або за який здійснюється зчитуванням інформації з ВРР кожен j -й зовнішній пристрій налаштовує генератор адрес відповідного j -го контролера ВРР для генерування потрібної адресної послідовності. Кожен j -й зовнішній пристрій може записувати або зчитувати інформацію з РР як послідовно, так і в полях. У разі словесного доступу j -го зовнішнього пристрою до паралельної пам'яті, для кожного запису або читання потрібна адреса попереднього запису в GA і, для інформаційного поля, адреса попереднього запису один раз на початку доступу до поля. . [12].

Процес попереднього запису адреси, до якої j -й зовнішній пристрій буде звертатися до РР, виглядає наступним чином. j -та шина даних SD_j від j -го зовнішнього пристрою отримує адресу, яка є імпульсом негативної полярності $\square\square ZpAdr_j$, записаним у лічильнику GA , крім того, цей сигнал записується схемою формування $F1$. Схеми формування керуючих сигналів $F1$, $F2$, $F3$ однакові і працюють наступним чином. Перед запуском тригери $Tr1.1$, $Tr1.2$, $Tr2.1$ і $Tr2.2$ знаходяться в журналі стану «0». На передньому фронті (перехід рівня сигналу з лог. "0" на лог. "1") керуючий сигнал перемикається на блок тригера $Tg1.1$ (тригер TgU - в стані лог. "0") або тригер, залежно від стан тригера $TgU.Tg2.1$ (тригер TgU - в стані лог. «1») і змінюємо стан тригера TgU на протилежне. Журнал сигналу «1» з виходу тригера $Tr1.1$ ($Tr2.1$) надходить на інформаційний вхід тригера $Tr1.2$ ($Tr2.2$) і записується в цей тригер тактового імпульсу $TI1_j$. Рівні логічної «1» на другому та третьому входах елемента $AND-NOT1.1$ (елемент $AND-NOT2.1$) дозволяють генерувати на його виході негативний імпульс приблизно тривалості тактового імпульсу $TI1_j$. Негативний імпульс з виходу елемента $I-NE1.1$ (елемент $I-NE2.1$) надходить на вихід ланцюга $ФК$, де $K = 1,2,3$, і проходить через елемент $I1.1$ (елемент $I2.1$) і передній фронт перемикає тригер $Tr1.1$ (тригер $Tr2.1$) у стан журналу «0». Після надходження наступних контрольних сигналів схеми $F1$, $F2$, $F3$ працюють аналогічно.

Керуючий сигнал, що формується на виході схеми F1, з'єднує виходи вихідного генератора з адресною шиною RAM і формує сигнали для за який здійснюється зчитування інформації RAM. Зчитана з ОЗП інформація надходить на вхід регістра Pr2 і передній фронт сигналу з виходу схеми F1 записується в цей регістр.

Розглянемо процес за який здійснюється зчитування інформації про j-й зовнішній пристрій з паралельної пам'яті. Для за який здійснюється зчитування інформації на вхід ЧтОЗПї потрібно подати імпульс з негативною полярністю. Цей імпульс надходить на вхід за який здійснюється зчитування регістра Pr2 і забезпечує передачу інформації з виходу цього регістра на шину даних SDj, тобто інформація зчитується за попередньо записаною адресою. Крім того, передній фронт цього сигналу збільшує вміст лічильника генератора адрес на одиницю і фіксує його в схемі F2. Після надходження тактового імпульсу TI1j на вихід схеми F2 формується сигнал, який підключається до адреси шини RAM і виходів GA і формує сигнали для за який здійснюється зчитування інформації з ОЗП. Інформація, зчитана з ОЗП за зміненою адресою, записується в Pr2. На цьому процедура читання одного слова закінчена. У разі за який здійснюється зчитування інформаційного поля наступні елементи поля читаються аналогічним чином. Попередній запис адреси та за який здійснюється зчитування інформації з ОЗП скорочує час вибірки інформації з ПП до часу роботи реєстру.

j-й зовнішній пристрій записує інформацію в паралельну пам'ять за попередньо записаною адресою наступним чином. Інформація для запису в пам'ять надходить на шину даних SDj і передній фронт сигналу ZPOZPj записується в Pr1, крім того сигнал ZPOZPj фіксується в схемі F3 і після надходження тактового імпульсу TI1j виводить відповідно GA і регістр Pr1 і формує керуючі сигнали для запису в оперативну пам'ять. Передній фронт сигналу з виходу схеми F3, вміст лічильника HA, збільшується на одиницю. На цьому процедура з одного слова закінчена. Якщо інформаційне поле записане - наступні елементи поля будуть записані аналогічним чином за адресами,

виданими Г.А. Час запису j -го зовнішнього інформаційного пристрою в програмному забезпеченні дорівнює часу роботи регістру [17].

3.3. Моделювання компонентів багатопортової пам'яті

Контролер ВРР був змодельований у VHDL у Active-HDL 8.1. Active-HDL — це повністю інтегроване середовище розробки цифрових пристроїв на основі текстових описів та графіки. Active-HDL є провідним середовищем проектування та моделювання для широкого спектру програмованих логічних інтегральних схем (FPGA), що забезпечує гнучкий підхід і забезпечує розширену підтримку найскладніших сучасних проектів. Середовище проектування Active-HDL надає розробникам незалежність у використанні інших інструментів проектування з єдиного, повністю інтегрованого середовища.

Асоціативна пам'ять доступу (АСУД) (або асоціативна пам'ять) передбачає зберігання даних та їх характеристик, якими можуть бути самі дані. Числа записуються в будь-які вільні (порожні) комірки пам'яті. Дані вибираються з такої пам'яті на основі збігу її знаків із набором. У паралельній асоціативній пам'яті всі цифри всіх полів пам'яті (так званий одночасний пошук слів) порівнюються з аргументом одночасно. У послідовної асоціативної пам'яті один біт кожного функціонального поля (так званий пошук бітів) одночасно порівнюється з бітом аргументу. Вміння працювати паралельно — головна перевага асоціативної пам'яті. Можливе читання всіх даних з ідентичними символами одночасно. Можливість швидкого перегляду даних у такій пам'яті дозволяє широко використовувати її в комп'ютерах, особливо за принципом пам'яті з асоціативним доступом часто є вбудована кеш-пам'ять [1]. Крім того, ця

пам'ять також має деякі недоліки, основними з яких є: порівняння всіх властивостей у клітинках з потрібними властивостями одночасно, а також забезпечення пошуку даних, властивості яких збігаються із зазначеними, що є проблема, особливо з великим об'ємом пам'яті. Необхідність подвійного доступу до однієї комірки пам'яті при записі та читанні числа уповільнює взаємодію пристроїв комп'ютера з пам'яттю асоціативного доступу. • Ця пам'ять має досить складну організацію, яка вимагає доступу до кожної комірки з введення-виводу пам'яті або з комбінованого введення/виводу пам'яті, і при зчитуванні потрібно порівняти вказану властивість з властивостями даних у пам'яті регістра, а також знайти дані, символи яких відповідають набору. Це коштує багато доступу і сповільнює пам'ять.

Пам'ять послідовного доступу. Послідовна пам'ять (послідовна пам'ять) побудована так, що дані зчитуються або записуються в цю пам'ять у послідовному порядку і формують чергу [1-5]. За який здійснюється зчитування виконуються з 85-ї черги слово до слова в порядку запису або навпаки. Прямий порядок читання забезпечується пам'яттю FIFO з службовою дисципліною First In, First Out. Зворотний порядок забезпечується пам'яттю LIFO з пам'яттю «Останній прийшов», «Першим вийшов» або FILO з дисципліною, яка прийшла першою, вона першою в черзі (Останній прийшов, перший вийшов). Першим увійшов, останнім вийшов. Пам'ять послідовного доступу, побудована на регістрах, часто називають стеком. Ця пам'ять швидко і легко реалізувати. Недоліком пам'яті послідовного доступу є значний час, необхідний для доступу до певної інформації. У гіршому випадку вам може знадобитися переглянути всі осередки пам'яті для цього доступу. Пам'ять з програмованим доступом. Подумайте про організацію нового типу пам'яті – пам'яті програмованого доступу (PPRD). Як і пам'ять послідовного доступу, PPRD зосереджено на роботі з наборами даних. Ця пам'ять забезпечує доступ до даних у програмно визначеному порядку, тобто код, який входить до пам'яті даних або при зчитуванні вказує розташування даних у полі джерела. У програмованій пам'яті немає прив'язки чисел до певної комірки пам'яті,

На малюнку. 3.1. показана схема контролера ВРР. Елементи U1, U2, U3 - відповідають схемам формування керуючого сигналу F1, F2, F3; U4 - генератор адреси GA; U5 і U6 відповідають буферним регістрам Pr1 і Pr2; BF1 і BF2 — водії автобусів. У схемі також є три логічні елементи «І».

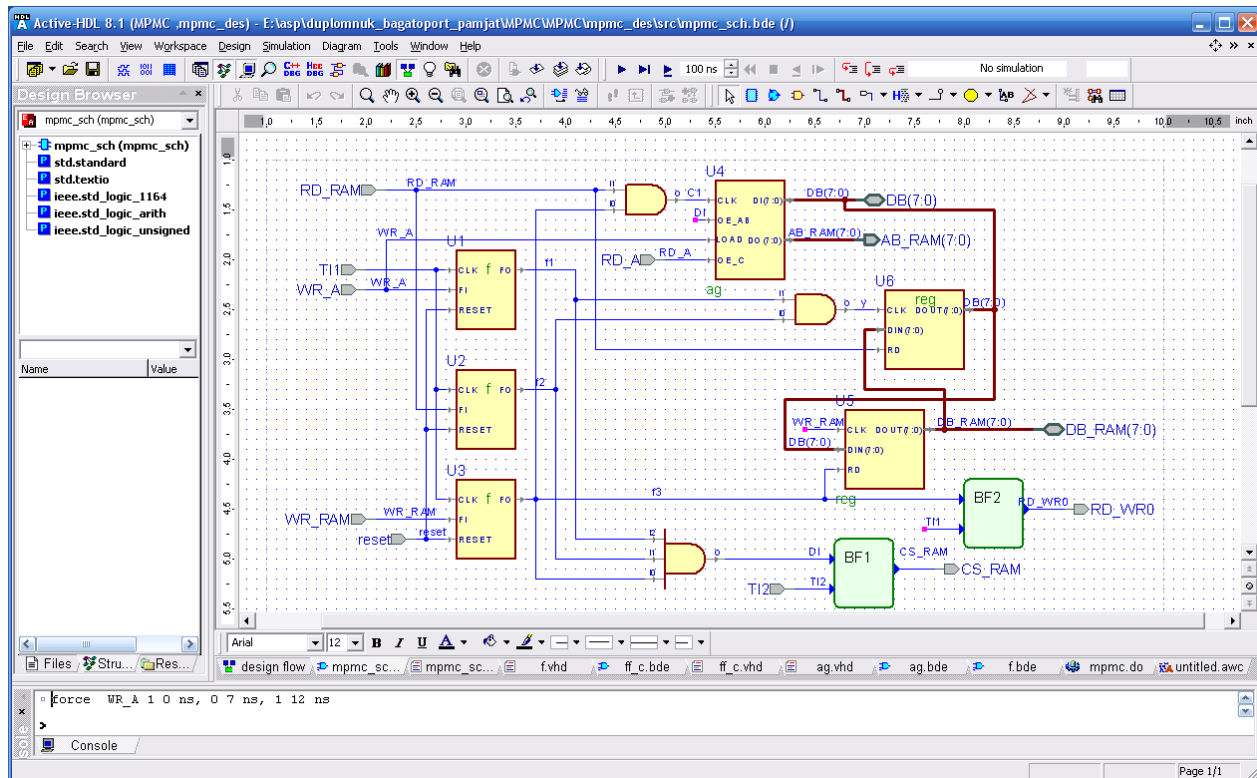


Рисунок. 3.1. Схема контролера БПП

Вхідними сигналами є: WR_RAM - сигнал запису сигналу ZpOZP, DB - шина даних, RD_RAM - сигнал за який здійснюється зчитування даних ChtOZP, RD_A - адреса сигналу читання ChtAdr, WR_A - адреса запису сигналу ZpAdr, T11 - імпульси, що надходять від тактового сигналу, вибірка вхідного сигналу T12 . Вихідні сигнали: DB_RAM - відповідає даним шини сигналу SD RAM, AB_RAM - відповідає шині адреси SHA RAM, RD_WR0 - відповідає сигналам читання/запису Thu / Zp RAM, CS_RAM - вибір кристала VC RAM. Шини DB, DB_RAM і AB_RAM вибираються як 8-розрядні [6].

На ісунку 3.2. представлена часова діаграма контролера БПП.

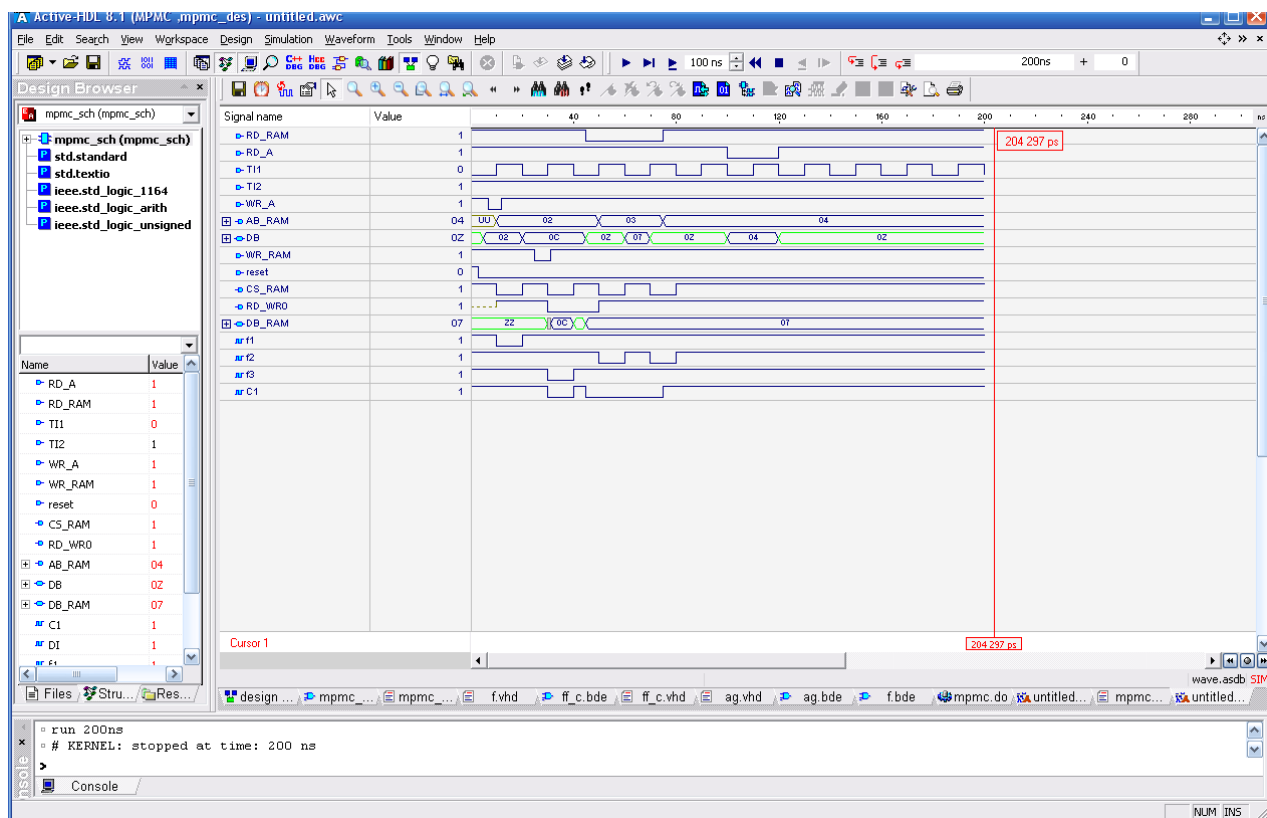


Рисунок. 3.2. Часова діаграма контролера БПП

Контролер ВРР може виконувати такі операції: записувати адресу задалегідь, зчитувати інформацію з паралельної пам'яті, записувати інформацію в паралельну пам'ять за попередньо записаною адресою і зчитувати вміст адреси зовнішнім пристроєм.

Попередній запис адреси виконується за імпульсом негативної полярності WR_A з DB (на який адреса надходить від зовнішнього пристрою і в даному випадку дорівнює 02) в генераторі адреси - U4. Імпульс фіксується за схемою виробництва F1. Сигнал f1, який формується на виході схеми формування F1, передає значення, що містяться в U4, в AB_RAM на передньому фронті. В результаті операції адреса (яка в даному випадку 02) надходить до AB_RAM.

Інформація з пам'яті зчитується на імпульсі негативної полярності входу RD_RAM. Передній фронт цього сигналу збільшує лічильник U4 на 1. При тактовому імпульсі TI1 на схемі F2 формується сигнал f2, який підключається до виходів AB_RAM генератора адрес U4 і змінює значення сигналу RD_WRO від

Від 0 до 1, тобто роздільна здатність читання. Інформація зчитується з DB_RAM (у цьому випадку - 07) і записується в U6, а звідти зчитується в DB на передньому фронті сигналу f2.

Інформація зберігається в паралельній пам'яті за попередньо записаною адресою переднім фронтом сигналу WR_RAM. Дані, які потрібно записати, знаходяться в БД (у цьому випадку це 0C) і за цим сигналом вони надходять до U5. Після надходження тактового імпульсу T11 на схему формування F3 на її виході формується сигнал f3. Він підключає до AB_RAM вихід лічильника U4 (попередньо збільшує його вміст на 1) і U5 до DB_RAM (куди передаються дані з БД). Тому сигнал DB_RAM у цьому випадку встановлюється на 0C відповідно до сигналу RD_WR0, який встановлюється на 0 (роздільна здатність запису) переднім фронтом сигналу f3 [12].

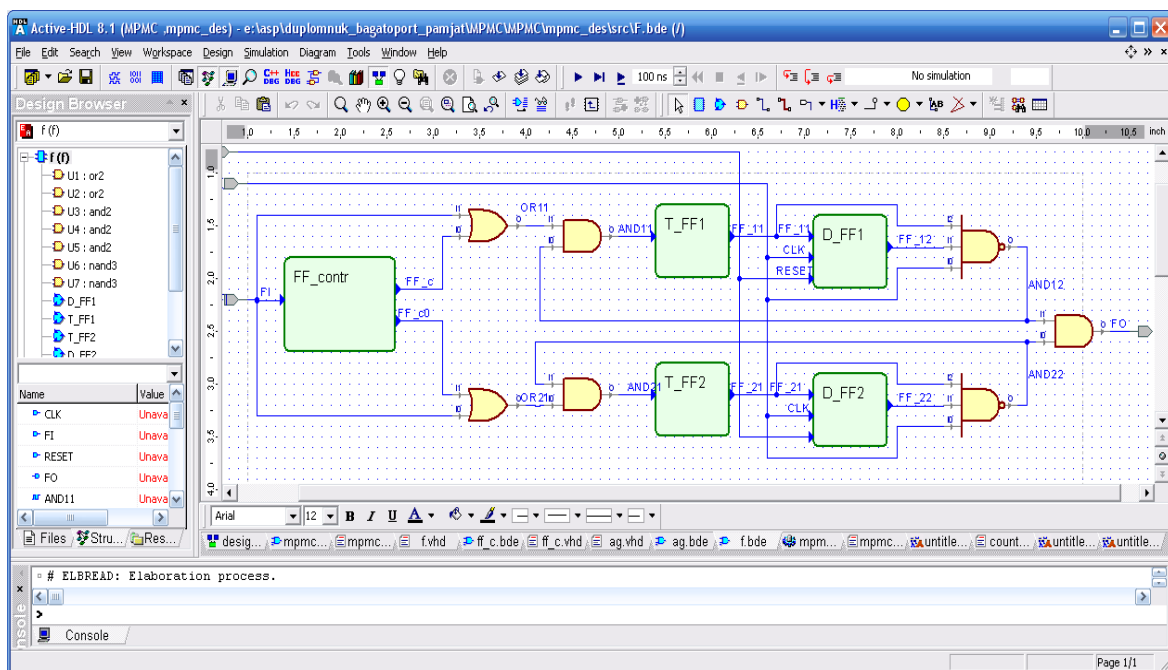


Рисунок. 3.3. Схема формувачів сигналів управління

Вміст адреси зчитується зовнішнім пристроєм за негативним імпульсним сигналом RD_A, що дозволяє передати адресу з лічильника AG (відповідного значенню AB_RAM) до DB (у цьому випадку значення - 04) через водій автобуса А.Г.

На рисунку 3.3. показана схема формувачів керуючого сигналу. Він складається з: керуючих тригерів - FF_contr, 2 елементів "АБО", 3 елементів "І", 2 тригерів Т - Т_FF1 і Т_FF2, 2 тригерів D - D_FF1 і D_FF2 і 2 елементів "І-НЕ".

Часова діаграма генераторів керуючих сигналів показана на малюнку. 3.4. Основними сигналами є: F1 - вхідний сигнал, FF_c, FF_c0 - вихідні сигнали тригера управління (FF_contr), OR11, OR21 - вихідні сигнали елементів "OR", AND11, AND21 - вихідні сигнали елементів "І", FF_11, FF_21 - вихідні сигнали Т-тригери - Т_FF1 і Т_FF2 в такому порядку, FF_12, FF_22 - вихідні сигнали тригерів D - D_FF1 і D_FF2 в такому порядку, AND12, AND22 - вихідні сигнали елементів "І-NOT", CLK - сигнал що надходить від тактового входу Т11, F0 - вихідний сигнал.

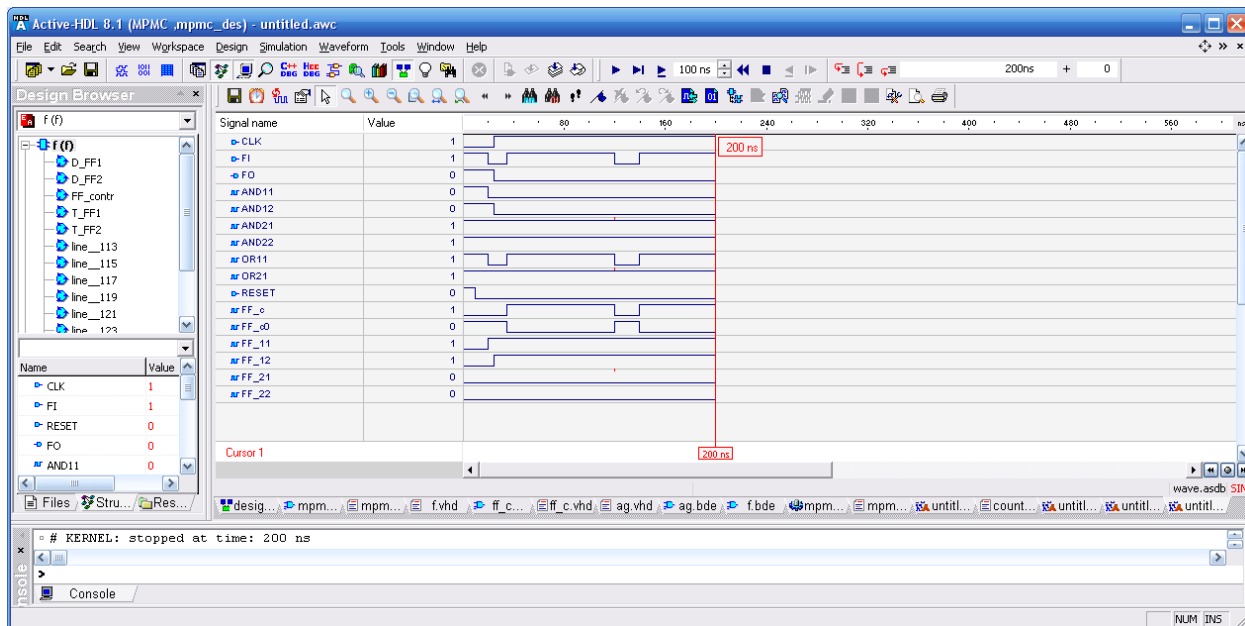


Рисунок. 3.4. Часова діаграма формувачів сигналів управління

На Рисунок. 3.5 показана схема генератора адрес. Основними елементами якої є: U1 – лічильник, перший шинний формувач – BF_1 та другий шинний формувач BF.

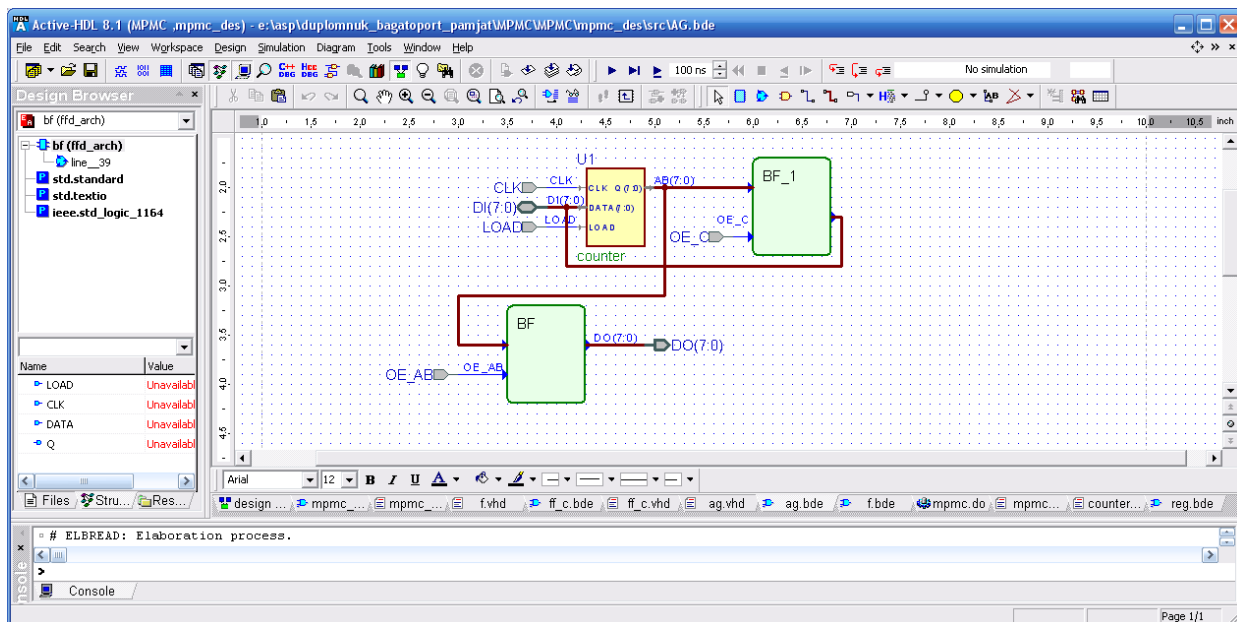


Рисунок. 3.5. Схема генератора адрес.

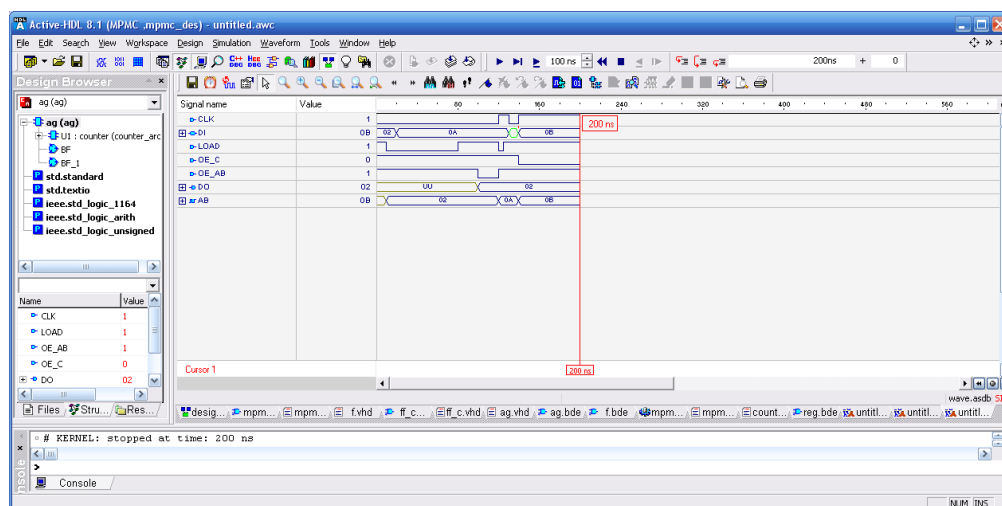


Рисунок. 3.6. Часова діаграма генератора адрес

Часова діаграма генератора адрес показана на малюнку. 3.6. Основними сигналами є: CLK - вхід синхронізації лічильника адрес, пов'язаний з виходом елемента "I" на схему контролера BPP, DI - сигнал, що надходить з шини даних (DB), LOAD - сигнал, що надходить від WR_A, AB - вихідний лічильник, який підключений до входів драйверів BF_1 і BF, OE_C - вхідний сигнал першого драйвера BF_1, що надходить від RD_A, OE_AB - вхідний сигнал другого

драйвера BF, що надходить від елемента "Г", який містить виходи сигнали керування драйвером, D0 - вихідний сигнал, відповідний сигналу AB_RAM на контролері схеми ВРР [8].

3.4. Розробка програми прикладу використання БПП в паралельно-потоківих системах

Структура паралельних потоківих систем (ППС) показана на рисунку 3.7, містить цифрові сигнальні процесори Kxm (PCOS), які поділяються відповідно до ВРР на K кроків [3]. Щоб ефективно розв'язувати задачі в такій системі, задачу потрібно розбити на K підзадач. При цьому кожна i -та підзадача ($i = 1, \dots, K$) вирішується паралельно СПКЯ i -го етапу системи. Час вирішення i -ї підзадачі має бути рівним або менше годин роботи системи, які визначаються виразом $T \leq N / (F \times g)$. При цьому кількість обчислень для кожної підзадачі не повинна перевищувати:

де P_j -потужність j -го процесора.

Щоб обробляти потоки даних у реальному часі відповідно до алгоритмів складності R , потужність P системи DSP повинна бути

$$P = gFR / N,$$

де g - кількість вхідних інформаційних каналів; F – частота введення даних у вхідні інформаційні канали, N – розмірність поля введення.

Перш ніж ви зможете запуснути РРР, ви повинні налаштувати його для виконання ряду завдань. Ця процедура виконується шляхом завантаження відповідних підпрограм, які є частиною загальної програми, в програмну пам'ять модулів процесора. Процес обробки інформації починається із завантаження першого поля вхідних даних та керуючої інформації, яка супроводжує це поле в першому модулі ВРР. У контрольній інформації вказується код завдання, яке

потрібно виконати, розмір i статус поля. Контрольна інформація записується при визначенні адреси модуля ВРР. Розмір пам'яті модулів ВРР визначається розміром оброблюваних полів даних і керуючою інформацією, яка їх супроводжує. Мінімальний розмір пам'яті ВРР повинен зберігати два максимальні набори даних з керуючою інформацією.

РРР орієнтований на обробку безперервного потоку даних з використанням алгоритмів, що характеризуються модульністю та розташуванням. Характерною особливістю цієї системи є можливість роботи в синхронному або асинхронному режимі. У синхронному режимі РРР синхронізується з сигналами переривання, які з'являються на модулях процесора. Період прийому цих сигналів дорівнює тактовій частоті системи T . В асинхронному режимі не існує загальної тактової частоти РРР. Перехід i -го кроку системи від обробки попереднього поля до обробки наступного виконується після завершення i -го кроку обробки попереднього поля та завантаження i -го поля. го в пам'ять. Наступне поле модуля ВРР. Управління трафіком РРР в асинхронному режимі здійснюється шляхом аналізу керуючої інформації, яка супроводжує масив даних, що передаються між сусідніми кроками системи [15].

Кожен раз, коли i -й крок закінчує обробку поля вхідних даних, результати обробки з керуючою інформацією зберігаються в пам'яті $(i + 1)$ модуля ВРР.

За допомогою керуючої інформації $(i + 1)$ -етап ППП отримує інформацію про готовність до обробки нового набору даних.

Основним параметром ВРР є час доступу пристрою до нього, який залежить від кількості пристроїв і часового циклу t_c читання (запису) в оперативну пам'ять. Поточний час для пристроїв з однаковою швидкістю роботи введення-виводу визначається виразом $T_c t_c n$.

Програма реалізована з використанням парадигми ООП і написана в середовищі Borland Delphi 7. Система Delphi дозволяє швидко та ефективно розробляти різноманітні програми. Delphi є відносно потужною візуальною системою об'єктно-орієнтованого програмування. Ця система дозволяє створити

зручний інтерфейс, а широкий спектр функцій, методів і можливостей дозволяє вирішувати прикладні обчислювальні задачі. Система також містить зручні інструменти для налагодження. Delphi традиційно називають системами швидкої розробки додатків (RAD). Це технологія візуального програмування, коли програміст проектує свою майбутню програму і бачить результати своєї роботи перед запуском програми.

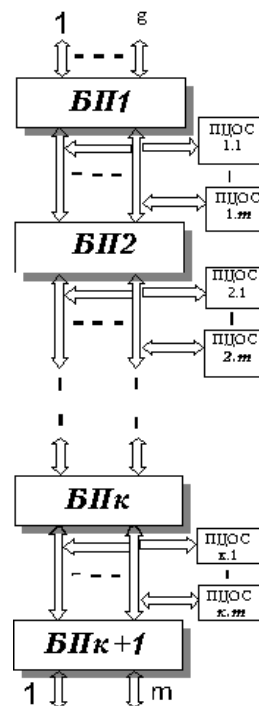


Рисунок. 3.7. Структура паралельно-поточної системи з обміном через багатопортову пам'ять

Програми Delphi створені в інтегрованому середовищі розробки (IDE). Інтерфейс користувача цього середовища використовується для взаємодії з програмістом і включає кілька вікон, які містять різні елементи керування. За допомогою інструментів IDE розробникам зручно розробляти частину інтерфейсу програми, а також писати програмний код і призначати його елементам керування. IDE проходить усі етапи створення програми, включаючи налагодження.

Delphi 7 IDE — це багатовіконна система. Для кожної програми автоматично пропонується вікно (форма), для якої вже створено два файли - з описом і модулем. Це дозволить створити найпростішу програму. Найпростіша програма — це порожній фреймворк, який надає розробникам усе необхідне для програми. Тому вам не доведеться писати власний обробник подій драйвера клавіатури чи миші або створювати пакет процедур для роботи з Windows. Під час розробки до найпростішої програми додаються нові форми, елементи керування та нові обробники подій.

Розробка програми Delphi включає два взаємопов'язаних кроки: створення інтерфейсу користувача та визначення функціональних можливостей програми. Інтерфейс користувача програми визначає, як користувач і програма взаємодіють, а саме вигляд форм під час виконання програми та спосіб, як користувач може керувати програмою. Інтерфейс розробляється шляхом розміщення його у формі компонентів, які називаються компонентами інтерфейсу або елементами керування. Функціональність програми визначається процедурами, які виконуються, коли відбуваються певні події, наприклад ті, що відбуваються під час дій користувача з елементами керування формою [20].

Розроблена програма призначена для виготовлення та впровадження прикладу використання БПП у ДПП. Загальний вигляд програми показаний на малюнку. 3.8. Робота починається з введення вхідних даних, які представляють собою кількість кроків (тобто кількість ВРР) у РРР, кількість РСOS на кожному кроці, кількість каналів вхідної інформації, частоту даних, розмір поля та тривалість циклу читання або запису. ».

При візуалізації РРР передбачається, що першим кроком є додавання двох полів, а другим — множення поля на значення, введене користувачем. Тому планується ввести два набори даних та зберегти їх за допомогою кнопки «Зберегти поле». Розмір полів даних відповідає розміру інформаційних полів, які раніше були введені у вхідні дані. Моделювання роботи ППП виконується за допомогою відповідної кнопки — «Моделювання роботи ППП». Відображає

кількість циклів, кількість кроків, що відповідають певній кількості ВРР, які були введені у вхідні дані, самі дані, результати кожного кроку та керуючу інформацію. Керуюча інформація включає код операції (код MUL у цьому case) та розміром поля. Відображається інформація, що міститься в кожному заході на кожному кроці РРР. Максимальна кількість даних для кожного кроку та кількість отриманих результатів відповідає кількості СПКЯ на кожному етапі.

Form1

Такт	БПі	Дані	Дані	Код	Розмір	Результат	Результат
1	БП1	10	12	Add	8		
	БП2						
	БП3						
2	БП1	14	16	Add	8		
	БП2	20	24	Mul	8		
	БП3						
3	БП1	18	20	Add	8		
	БП2	28	32	Mul	8		
	БП3	20	24	None	8		
4	БП1	22	24	Add	8		
	БП2	36	40	Mul	8		
	БП3	28	32	None	8	20	24
5	БП1						
	БП2	44	48	Mul	8		
	БП3	36	40	None	8	28	32
6	БП1						
	БП2						
	БП3	44	48	None	8	36	40
7	БП1						
	БП2						
	БП3					44	48

Вхідні дані

Кількість підзадач - K: 3

Кількість ПЦОС кожної сходинки - m: 2

Кількість вхідних інформаційних каналів - g: 2

Частота поступлення даних - F: 66

Величина масиву інформації - N: 8

Тривалість циклу читання/запису в ОЗП - T_ц: 0,09

Продуктивність процесора: 8448

Такт роботи системи: 0,06060606060606

Об'єм обчислень кожної підзадачі: 1024

Період звертання до БПП: 0,18

Зберегти

Порахувати

Масиви даних

0	1	2	3	4	5	6	7
10	11	12	13	14	15	16	17

Зберегти масиви

Етап1: Додавання двох масивів

Етап 2: Множення масиву на значення 2

Моделювання роботи ППС

Рисунок. 3.8. Загальний вигляд програми прикладу використання БПП в ППС.

Також можна порахувати значення продуктивності кожного процесора, такт роботи системи, об'єм обчислень кожної під задачі та період звертання бо БПП.

ВИСНОВКИ

На основі теоретичних та досліджень побудованих на експериментах у галузі спеціалізації вирішено наукові проблеми, удосконалено методи синтезу, створено високоефективні вертикальні нейронні мережі з опрацюванням у даний час з обладнанням аналізу нейронного спектру. Ось основні результати.

1. Визначити характеристики нейронних технологій з опрацюванням у даний час, визначити основу нейронних мереж та продемонструвати необхідність інтеграції нейронних мереж з опрацюванням у даний час та паралельних вертикальних нейронних мереж нейронних елементів.

2. Покращені критерії продуктивності обладнання, пов'язані з продуктивністю та вартістю обладнання, і використовуються для оцінки нейронних елементів і нейронних мереж. Проектування нейронних елементів та апаратних нейронних мереж базувалося на таких принципах, як паралельність, модульність, алгоритмічна структура, апаратне забезпечення, розробка програмного забезпечення, контроль інтенсивності інформації та можливості програмування. Архітектурний з використанням перепрограмованої FPGA.

3. Розроблено методику паралельно-вертикальної обробки інформації в нейронних мережах (нейронних елементах).

4. Формальні паралельно-вертикальні нейронні моделі були розроблені для розділення вхідної інформації та вагових показників у бітах, їх обробки, виготовлення та впровадження функцій табуляції та компіляції результатів у бітах.

5. Синтезований на базі NBSI на паралельно-вертикальних нервових елементах, що забезпечує високу ефективність роботи обладнання за рахунок координації та поєднання процесу паралельного прийому та обробки вхідної інформації.

6. Розширений метод інтеграції нейронних мереж в режимі з опрацюванням у даний час, що дозволяє розробляти потужні нейронні мережі в

режимі з опрацюванням у даний час на всіх етапах обробки інформації відповідно до обчислювальної потужності інформації.

7. Зосередьтеся на обчисленнях із фіксованою точкою та виготовлення та впровадження VLSI, що призвело до впровадження передового апаратного забезпечення високої ємності для спектрального аналізу нейронної мережі.

8. Узагальнити результати опрацювання нейронної мережі в реальному часі на основі паралельних і вертикальних типів нейронних елементів для створення масиву та підвищення точності динамічних вимірювань відстані за час, що дорівнює часу за який здійснюється зчитування з пам'яті масиву. його діяльність. . необов'язковий. , використання передового аналізу нейронного спектру може зменшити похибку вимірювання в 2,4 рази.

Список використаних джерел

1. Цмоць І.Г. Реалізація нейронного елемента на основі попередніх обчислень / І.Г. Цмоць, О.В. Скорохода, Б.І. Балич // Вісник Нац. ун-ту. «Львівська політехніка»: Комп'ютерні науки та інформаційні технології. – Львів, 2011. – № 710. – С. 11–18.
2. Цмоць І.Г. Синтез високоефективних багатошарових перцептронів з навчанням / І.Г. Цмоць, Р.О. Ткаченко, О.В. Скорохода // Вісник Нац. ун-ту. «Львівська політехніка»: Комп'ютерні науки та інформаційні технології. – Львів, 2009. – № 650. – С. 45–56.
3. Цмоць І.Г. Принципи побудови та способи ПРОЕКТУВАННЯ НАДВЕЛИКИХ ІНТЕГРАЛЬНИХ СХЕМ нейронних мереж з опрацюванням у даний час / І.Г. Цмоць, О.В. Скорохода, І.Є. Ваврук // Наук. вісник НЛТУ України: зб. наук.-техн. праць. – Львів: РВВ НЛТУ України, 2012. – Вип. 22.6. – С. 292–300.
4. Скорохода О.В. Особливості виготовлення та впровадження нейронних мереж з опрацюванням у даний час / О.В. Скорохода, І.Г. Цмоць, Б.І. Сенах // Науково-публіцистичний часопис «Технічні вісті». – Львів, 2011. – № 1(33)–2(34). – С. 28–30.
5. Цмоць І.Г. Методи та Надвеликі інтегральні схеми та структури пристроїв з паралельним та вертикальним обчисленням обчислення сум парних добутоків / І.Г. Цмоць, Б.І. Балич, О.В. Скорохода // Відбір і обробка інформації. – Львів, 2011. – № 33 (109). – С. 109–116.
6. Скорохода О.В. Засоби оцінювання параметрів динамічних об'єктів на основі нейронних мережевого сингулярного спектрального аналізу / О.В. Скорохода, Б.Р. Андрієцький, І.Г. Цмоць, Р.О. Ткаченко // Наук. вісник НЛТУ України: зб. наук.-техн. праць. – Львів: РВВ НЛТУ України, 2012. – Вип. 22.11. – С. 362–369.

7. Цмоць І.Г. Модифікований метод та НАДВЕЛИКИХ ІНТЕГРАЛЬНИХ СХЕМ-структура пристрою групового підсумовування для нейронний елемента / І.Г. Цмоць, О.В. Скорохода, Б.І. Балич // Вісник Нац. ун-ту «Львівська політехніка»: Комп'ютерні науки та інформаційні технології. – Львів, 2012. – № 732. – С. 51–57.
8. Ткаченко Р.О. Програмно-апаратна реалізація складений з багатьох шарів перцептрона з навчанням на базі різницевого вертикального таблично-алгоритмічного методу / Р.О. Ткаченко, І.Г. Цмоць, О.В. Скорохода, Б.І. Балич // Вісник Нац. ун-ту «Львівська політехніка»: Комп'ютерні науки та інформаційні технології. – Львів, 2010. – № 686. – С. 65–71.
9. Цмоць І.Г. Інтегрований підхід до синтезу високоефективних апаратних засобів нейронних мережевих технологій з опрацюванням у даний час / І.Г. Цмоць, Я.П. Кісь, О.В. Скорохода // Наук. вісник НЛТУ України: зб. наук.-техн. праць. – Львів: РВВ НЛТУ України, 2009. – Вип. 19.9. – С. 269–279.
10. Пат. № 66138, Україна, МПК G06F 7/38. Пристрій для обчислення сум парних добутків: Патент на корисну модель / І.Г. Цмоць, О.В. Скорохода; заявник і патентовласник Національний університет «Львівська політехніка». – № u201106811; заявл. 30.05.2011; опубл. 26.12.2011, Бюл. № 24. – 8 с.
11. Tsmots I. Methods and VLSI-structures for neural element implementation / I. Tsmots, O. Skorokhoda // Proc. of the VI International Scientific and Technical Conference «MEMSTECH'2010», Polyana, 20–23 April 2010. – Lviv, 2010. – P. 135.
12. Скорохода О.В. Таблично-алгоритмічна реалізація штучних нейронних мереж / О.В. Скорохода, П.В. Романюк, О.Р. Якимів // Матеріали міжнародної наукової конференції «Інтелектуальні системи прийняття рішень та проблеми обчислювального інтелекту ISDMCI'2010», Євпаторія, 17–21 травня 2010 р. – Євпаторія, 2010. – Т. 1. – С. 398.

13. Грицик В.В. Особливості технології нейрокомп'ютингу з опрацюванням у даний час / В.В. Грицик, Р.О. Ткаченко, І.Г. Цмоць, О.В. Скорохода // Матеріали науково-технічної конференції «Обчислювальні методи і системи перетворення інформації», Львів, 7–8 жовтня 2010 р. – Львів, 2010. – С. 229–232.
14. Skorokhoda O. Vertical-tabular implementation of neural element for the synthesis of multilayer perceptron with non-iterative learning / O. Skorokhoda, B. Balych, R. Tkachenko, I. Tsmots // Proc. of the V International Scientific and Technical Conference «CSIT'2010», Lviv, 14–16 October 2010. – Lviv, 2010. – P. 41.
15. Tsmots I. Hardware implementation of the real time neural network components / I. Tsmots, O. Skorokhoda // Proc. of the VII International Scientific and Technical Conf. «MEMSTECH'2011», Polyana, 11–14 May 2011. – Lviv, 2011. – P. 124–126.
16. Скорохода О.В. Вертикально-паралельний метод та структури для виготовлення та впровадження базових компонентів нейронний елемента з використанням попередніх обчислень / О.В. Скорохода, І.Г. Цмоць, Я.П. Кісь // Матеріали міжнародної наукової конференції «Інтелектуальні системи прийняття рішень та проблеми обчислювального інтелекту ISDMCI'2011», Євпаторія, 16–20 травня 2011 р. – Євпаторія, 2011. – Т. 1. – С. 311–313.
17. Skorokhoda O. Stages of FPGA-Based Neural Networks Construction / O. Skorokhoda, I. Tsmots // Proc. of the V International Scientific and Technical Conference «CSIT'2011», Lviv, 16–19 November 2011. – Lviv, 2011. – P. 28.
18. Скорохода О.В. Методологія розробки апаратних нейронних мереж з опрацюванням у даний час / О.В. Скорохода, І.Г. Цмоць, Р.О. Ткаченко // Матеріали міжнародної наукової конференції «Інтелектуальні системи прийняття рішень та проблеми обчислювального інтелекту ISDMCI'2012», Євпаторія, 27–31 травня 2012 р. – Євпаторія, 2012. – С. 415–416.

19. Андрієцький Б.Р. Використання нейронних мережевого сингулярного спектрального аналізу в задачах оцінювання та прогнозування параметрів динамічних об'єктів / Б.Р. Андрієцький, О.В. Скорохода, І.Г. Цмоць // Збірник тез доповідей шостої міжнародної науково-технічної конференції «Фотоніка ОДС-2012», Вінниця, 1–4 жовтня 2012 р. – Вінниця, 2012. – С. 48.
20. Haykin S. Neural networks and learning machines. Third Edition. / S. Haykin. – New York: Prentice Hall, 2009. – 936 p.
21. Круглов В.В. Искусственные нейронные сети. Теория и практика. 2-е изд. / В.В. Круглов, В.В. Борисов. – М.: Горячая линия-Телеком, 2002. – 382 с.
22. Галушкин А.И. Нейрокомпьютеры. Книга 3. / А.И. Галушкин. – М.: ИПРЖР, 2000. – 528 с.
23. Тимошук П.В. Штучні нейронні мережі: навч. посібник / П.В. Тимошук. - Львів: Видавництво Львівської політехніки, 2011. - 444 с.
24. Осовский С. Нейронные сети для обработки информации / С. Осовский; [пер. с польского]. – М.: Финансы и статистика, 2002. – 344 с.
25. Рутковская Д. Нейронные сети, генетические алгоритмы и нечеткие системы / Д. Рутковская, Л. Пилиньский, Л. Рутковский. [пер. с польского]. – М.: Горячая линия-Телеком, 2007. – 452 с.
26. Руденко О.Г., Бодянський Є.В. Штучні нейронних мережеві технології/ О.Г. Руденко, Є.В. Бодянський. – Харків: ТОВ «Компанія СМІТ», 2006. – 404 с.
27. Applications of adaptive systems [Електронний ресурс]. – Режим доступу: <http://www.peltarion.com/doc/index.php?title=Applications-of-adaptive-systems>
28. Nguyen N. Predicting housing value: a comparison of multiple regression analysis and artificial neural networks / N. Nguyen, A. Cripps // Journal of Real Estate Research. – 2001. – Vol. 22, Issue 3. – pp. 314–336.

- 29.Литвиненко В.І. Методологія синтезу колективу радіально-базисних мереж для розв'язання задач класифікації за допомогою алгоритму клонального відбору / В.І. Литвиненко, А.О. Фефелов, О.О. Дідик // Наукові праці: Науково-методичний журнал. - Миколаїв: Вид-во ЧДУ ім. П. Могили, 2009. - Т.106, Вип. 93. - С. 114-126.
- 30.Мисюрёв А.В. Использование искусственных нейронных сетей для распознавания рукопечатных символов / А.В. Мисюрёв // Интеллектуальные технологии ввода и обработки информации. – М.: Эдиториал УРСС, 1998. – 164 с.
- 31.Su T.-J. Medical Image Noise Reduction Using Cellular Neural Networks / T.-J. Su, J.-W. Jhang // ИН-MSP '06 International Conference on Intelligent Information Hiding and Multimedia Signal Processing, December, 2006. – pp. 228–231.
- 32.Bashir Z.A. Applying Wavelets to Short-Term Load Forecasting Using PSO-Based Neural Networks / Z.A. Bashir, M. E. El-Hawary // Power Systems. – 2009. – Vol. 24 , Issue 1. – pp. 20–27.
- 33.Інформаційні критерії параметричної ідентифікації динамічних систем / О.І. Михальов, А.І. Гуда // Адапт. системи автомат. упр.: Міжвід. наук.-техн. зб. – 2003. – № 6. – С. 105-112.
- 34.Анисимов А.А. Идентификация электромеханических систем с использованием искусственной нейронной сети / А.А. Анисимов, М.Н. Горячев // Вестник ИГЭУ. – 2008. – № 3. – С. 55–58.
- 35.Aizenberg N.N. Extraction and localization of important features on gray-scale images: implementation on the CNN NN / Aizenberg N.N., Aizenberg I.N., Belikova T.P. // Proc. of the Third IEEE International Workshop on Cellular Neural Networks and their Applications (CNNA-94). - Rome, 1994. - P. 207-212.
- 36.Dhar S. Performance evaluation of Neural Network approach in financial prediction: Evidence from Indian Market / S. Dhar, T. Mukherjee, A.K. Ghoshal

- // International Conference on Communication and Computational Intelligence (INCOCCI), December, 2010. – pp. 597–602.
37. Huang S.Y. Unsupervised neural networks approach for understanding fraudulent financial reporting / S.Y. Huang, R.H. Tsaih, W.Y. Lin // *Industrial Management & Data Systems*. – 2012. – Vol. 112, Issue 2. – pp. 224–244.
38. Bigné E. Mature market segmentation: a comparison of artificial neural networks and traditional methods / Bigné E., Aldas-Manzano J., Küster I., Vila N // *Neural Computing and Applications*. – 2010. – Vol. 19, Issue 1. – pp. 1–11.
39. Angelini E. A neural network approach for credit risk evaluation / E. Angelini, G. Di Tollo, A. Roli // *The Quarterly Review of Economics and Finance*. – 2008. – Vol. 48, Issue 4. – pp. 733–755.
40. Friesen D.D. A comparison of multiple regression and neural networks for forecasting real estate values / D.D. Friesen, M. Patterson, R. Harmel // *Regional Business Review*. – 2011. – Vol. 30. – pp. 114–136.
41. Kelemenis A. Support managers' selection using an extension of fuzzy TOPSIS / A. Kelemenis, K. Ergazakis, D. Askounis // *Expert Systems with Applications*. – 2011. – Vol. 38, Issue 3. – pp. 2774–2782.
42. Irwin, George William, Kevin Warwick, and Kenneth J. Hunt, eds. "Neural network applications in control." Institution of Electrical Engineers, 1995.
43. Fierro R. Control of a nonholonomic mobile robot using neural networks / R. Fierro, F.L. Lewis // *Neural Networks, IEEE Transactions on*. – 1998. – Vol. 9, Issue 4. – pp. 589–600.
44. Walter J.A. Implementation of self-organizing neural networks for visuo-motor control of an industrial robot / J.A. Walter, K.I. Schulten // *Neural Networks, IEEE Transactions on*. – 1993. – Vol. 4, Issue 1. – pp. 86–96.
45. Johnston S.P. An FPGA hardware/software co-design towards evolvable spiking neural networks for robotics application / S.P. Johnston, G. Prasad, L. Maguire,

- T.M. McGinnity // International journal of neural systems. – 2010. – Vol. 20, Issue 6. – pp. 447–461.
46. Application of artificial neural networks for the prediction of performance and exhaust emissions in SI engine using ethanol-gasoline blends / M.K. Deh Kiani, B. Ghobadian, T. Tavakoli [et al.] // Energy. – 2010. Vol. 35, Issue 1. – pp. 65–69.
47. Lucas C. Intelligent modeling and control of washing machine using locally linear neuro-fuzzy (llnf) modeling and modified brain emotional learning based intelligent controller (BELBIC) / C. Lucas, R.M. Milasi, B.N. Araabi // Asian Journal of Control. – 2006. – Vol. 8, Issue 4. – pp. 393–400.
48. Applications of Cellular Neural Networks to Noise Cancellation in Gray Images Based on Adaptive Particle-swarm Optimization / T.J. Su, J.C. Cheng, M.Y. Huang [et al.] // Circuits, Systems, and Signal Processing. – 2011. – Vol. 30, Issue 6. – pp. 1131–1148.
49. Smart meter systems detection & classification using artificial neural networks / [T. Bier, D.O. Abdeslam, J. Merckle, D. Benyoucef] // “IECON 2012” - 38th Annual Conference on IEEE Industrial Electronics Society, October, 2012. – pp. 3324–3329.
50. Köse E. The investigation of effects of digital proofing systems used in colour management on print quality with neural networks / E. Köse, T. Şahinbaşkan, I. Güler // Expert Systems with Applications. – 2009. – Vol. 36, Issue 1. – pp. 745–754.
51. Go J. Interpolation using neural networks for digital still cameras / J. Go, K. Sohn, C. Lee // Consumer Electronics, IEEE Transactions on. – 2000. – Vol. 46, Issue 3. – pp. 610–616.
52. Ince T. Evaluation of global and local training techniques over feed-forward neural network architecture spaces for computer-aided medical diagnosis / T. Ince, S. Kiranyaz, J. Pulkkinen, M. Gabbouj // Expert Systems with Applications. – 2010. – Vol. 37, Issue 12. – pp. 8450–8461.

53. Lancashire L.J. An introduction to artificial neural networks in bioinformatics— application to complex microarray and mass spectrometry datasets in cancer studies // L.J. Lancashire, C. Lemetre, G.R. Ball // *Briefings in bioinformatics*. – 2009. – Vol. 10, Issue 3. – pp. 315–329.
54. Edwards I. R. Adverse drug reactions: definitions, diagnosis, and management / I.R. Edwards, J.K. Aronson // *The Lancet*. – 2000. – Vol. 356 (9237). – pp. 1255–1259.
55. Ripley B.D. Neural networks as statistical methods in survival analysis / B.D. Ripley, R.M. Ripley // *Clinical applications of artificial neural networks*. – 2001 – pp. 237–255.
56. Куссуль Н.М. Виявлення аномальної діяльності користувачів комп'ютерних систем на основі нейронних мережного прогнозування / Н.М. Куссуль, О.М. Резнік, А.М. Соколов // *Abstracts of International Conference “Prediction and Decision Making under Uncertainties”*. - Київ, 2001. - С. 116-117
57. Chen C.Y. The exploration of internet marketing strategy by search engine optimization: a critical review and comparison / C.Y. Chen, B.Y. Shih, Z.S. Chen, T.H. Chen // *African Journal of Business Management*. – 2011. Vol. 5, Issue 12. – pp. 4644–4649.
58. Ripley B.D. *Pattern recognition and neural networks* / B.D. Ripley. – Cambridge university press, 2008. – 416 p.
59. Neural network ensembles for video game AI using evolutionary multi-objective optimization / [T.G. Tan, J. Teo, P. Anthony et al.] // *11th International Conference on Hybrid Intelligent Systems (HIS)*, December, 2011. – pp. 605–610.
60. McCulloch W.S. A logical calculus of the ideas immanent in nervous activity / W.S. McCulloch, W. Pitts // *The Bulletin of Mathematical Biophysics*. – Vol. 5, Issue 4. – pp. 115–133.

61. ADALINE (Adaptive linear) [Электронный ресурс]. – Режим доступа: <http://www.cs.utsa.edu/~bylander/cs4793/learns32.pdf>
62. Fukushima K. Cognitron: A self-organizing multilayered neural network / K Fukushima // *Biological cybernetics*. – 1975. – Vol. 20, Issue 3-4. – pp. 121–136.
63. Hopfield, J.J. Neural networks and physical systems with emergent collective computational abilities / J.J. Hopfield // *Proceedings of the national academy of sciences*. – 1982. – Vol. 79, Issue 8. – pp. 2554–2558.
64. Cao J. Boundedness and stability for Cohen–Grossberg neural network with time-varying delays / J. Cao, J. Liang // *Journal of Mathematical Analysis and Applications*. – 2004. – Vol 296, Issue 2. – pp. 665–685.
65. Нейронные сети [Электронный ресурс]. – Режим доступа: <http://www.statsoft.ru/home/textbook/modules/stneunet.html>
66. Rosenblatt F. Principles of neurodynamics: perceptrons and the theory of brain mechanisms / F. Rosenblatt. – Spartan Books, 1962. – 616 p.
67. Chen S. Orthogonal least squares learning algorithm for radial basis function networks / S. Chen, C.F.N. Cowan, P.M. Grant // *Neural Networks, IEEE Transactions on*. – 1991. – Vol. 2, Issue 2. – pp. 302–309.
68. Specht D.F. Probabilistic neural networks / D.F. Specht // *Neural networks*. – 1990. – Vol. 3, Issue 1. – pp. 109–118.
69. Specht D.F. A general regression neural network / D.F. Specht // *Neural Networks, IEEE Transactions on*. – 1991. – Vol. 2, Issue 6. – pp. 568–576.
70. Neighborhood behavior: a useful concept for validation of “molecular diversity” descriptors / D.E. Patterson, R.D. Cramer, A.M. Ferguson [et al.] // *Journal of medicinal chemistry*. – 1996. – Vol. 39, Issue 16. – pp. 3049–3059.
71. Bishop C.M. Neural networks for pattern recognition / C.M. Bishop. Oxford university press, 1995. – 482 p.
72. Kohonen T. The self-organizing map / T. Kohonen // *Proceedings of the IEEE*. – 1990. – Vol. 78, Issue 9. – pp. 1464–1480.

73. Хайкин С. Нейронные сети: полный курс / С. Хайкин; [пер. с англ]. – М.: Вильямс, 2006. – 1104 с.
74. Fausett L.V. Fundamentals of neural networks: architectures, algorithms, and applications / L.V. Fausett. – Englewood Cliffs: Prentice-Hall, 1994. – 461 p.
75. Ткаченко Р.О. Нова парадигма мереж з штучним інтелектом прямого поширення / Р.О. Ткаченко // Вісник Державного університету “Львівська політехніка”: Комп’ютерна інженерія та інформаційні технології. – 1999. – № 386. – С. 43–54.
76. Ткаченко Р.О. Моделювання методами нейронних мереж: навч.-метод. посібник / Р.О. Ткаченко, П.Р. Ткаченко, Н.О. Мельник. – Львів: ЛІБС УБС НБУ, 2010. – 114 с.
77. Ткаченко Р.О. Неітераційне навчання нейронних мереж прямого поширення / Р.О. Ткаченко, І.Ю. Юрчак, Ю.В. Цимбал // Вісник Державного університету “Львівська політехніка”: Комп’ютерна інженерія та інформаційні технології. – Львів. – 1999. – № 380. – С. 109–115.
78. Байков В.Д. Специализированные процессоры: Интегральные алгоритмы и структуры / В.Д. Байков, В.Б. Смолов. – М.: Радио и связи, 1985. – 288 с.
79. СуперЭВМ. Аппаратная и программная организация / под ред. С. Фернбаха; [пер. с англ]. – М.: Радио и связь, 1991. – 320 с.
80. Кун С. Матричные процессоры на СБИС / С. Кун. – М.: Мир, 1991. – 672 с.
81. Параллельная обработка информации: Т.4. Высокопроизводительные системы параллельной обработки информации / под ред. В.В. Грицыка. – Киев: Наук. думка, 1988. – 272 с.
82. Мельник А.А. Процессоры обработки сигналов / А.А. Мельник. – Львов, 1989. – 63 с. – (Препринт / АН УССР, ИППММ; 29-89).

Додаток Б
Довідка про використання

Додаток В
Текст програмної виготовлення та впровадження
пристрою С#