

УДК: 681.325

В.П.Тарасенко, О.К.Тесленко, Я.М.Клятченко

Національний технічний університет України «Київський політехнічний інститут»

СУЧАСНІ ПЛІС-ОРІЄНТОВАНІ ТЕХНОЛОГІЇ В ОСВІТНІЙ ГАЛУЗІ

Розглядаються основні сучасні тенденції побудови апаратних засобів на ПЛІС, галузі та особливості їхнього застосування, напрямки розвитку технологій на базі програмованих логічних структур та, як приклад, варіант реалізації спеціалізованого пристрою з використанням сучасних ПЛІС.

Вступ

Програмовані логічні інтегральні схеми (ПЛІС) стали практично єдиною компонентною базою сучасної обчислювальної техніки [1], яка дозволяє користувачеві адаптувати серійний інструментарій інформаційних технологій під особливості вирішуваних задач (шляхом реконфігурування цього інструментарію). Крім того, ПЛІС стала єдиною компонентною базою для створення оригінальних обчислювальних структур в країнах, що не мають власної розвинутої електронної промисловості. В зв'язку з цим виникає питання про зміст відповідних навчальних програм по тематиці ПЛІС для студентів різних освітніх напрямів у вищій технічній школі України. Нижче обговорюються деякі найбільш важливі особливості і якості сучасних ПЛІС та пропонується варіант навчальної програми з тематики ПЛІС для студентів освітнього напрямку "Комп'ютерна інженерія", розроблений в НТУУ "КПІ".

Сфери застосування структур на ПЛІС

Розширення сфери традиційного застосування комп'ютерних систем, створення потужних пошукових систем, комп'ютерні експерименти по дослідженню когнітивних властивостей інформації та інше, обумовлюють необхідність включення до операційних ресурсів сучасних комп'ютерних систем операцій з обробки масивів, матриць, векторів тощо, що базуються на елементарних операціях. Інший клас задач, таких як моделювання складного руху об'єктів або сукупності об'єктів, обробка сигналів та зображень у реальному часі, складна реалістична комп'ютерна графіка тощо пов'язаний з багаторазовим виконанням цих елементарних операцій та вимагають виконання декількох мільярдів операцій на секунду.

Послідовне виконання великої кількості елементарних операцій займає відносно великий час, а також, навіть для найпростіших задач необхідно використовувати комп'ютер у цілому. Таким чином, прості задачі і задачі, які повинні вирішуватися в реальному часі, можуть привести до необхідності застосування апаратних варіантів. Для задач, що вирішуються в режимі реального часу, результат повинен бути отриманий за визначений інтервал часу, інакше він стає непотрібним [2]. Таким чином, апаратні методи можуть бути єдиним ефективним рішенням, і тому розглянемо варіанти застосування програмованих логічних структур в якості основи для цих рішень.

ПЛІС як єдина компонентна база нових розробок обчислювальної техніки в Україні

Для створення апаратних засобів для вирішення задач, що вказані вище, потрібен набір різноманітних елементів, вузлів і пристроїв. Такий набір можна реалізувати на мікросхемах різних ступенів інтеграції. Застосування ВІС/НВІС покращує такі параметри апаратури як вартість, швидкодія, надійність, економічність (по споживаній потужності) та ін. Але, як відомо з практики, на шляху реалізації апаратних засобів у вигляді ВІС/НВІС можуть виникнути великі економічні труднощі, оскільки розробка ВІС/НВІС надзвичайно дорога і вимагає великих витрат часу на розробку, що унеможливує таке застосування при створенні спеціалізованих обчислювальних засобів. Деякий час тому намітилася тенденція в галузі обчислювальної техніки та систем управління в спільному використанні мікропроцесорного пристрою разом із програмованими логічними інтегральними схемами, які змогли задовольнити вимоги широкого кола користувачів та започаткувати нові форми ефективних обчислювальних засобів. Крім того, ПЛІС дозволяють за допомогою використання тільки персонального комп'ютера та САПР ПЛІС реалізувати в кристалі практично будь-який проект цифрового пристрою за короткий проміжок часу. Застосування

сучасних ПЛІС, які за ступенем інтеграції досягли параметрів традиційних ВІС/НВІС, є єдиним можливим рішенням на шляху реалізації систем у цілому.

В сучасному застосуванні ПЛІС використовується при побудові:

– апаратури спеціалізованого призначення (реалізація складних алгоритмів та операцій, апаратні прискорювачі обчислень, нетрадиційна обробка сигналів в реальному часі, створення засобів моніторингу та керування виробничими процесами, кодування/декодування інформації в телекомунікаційних системах);

– інтерфейсних засобів сполучення з апаратурою керуючих систем;

– засобів захисту інформації. Різноманітні можливості застосування спеціалізованих апаратних засобів використовують для захисту інформації. Це можуть бути найпростіші засоби контролю доступу до інформації на твердих магнітних і лазерних дисках — так звані індивідуальні електронні ключі доступу. У інших випадках застосовується шифрація інформації перед записом на диск або передачею в мережу. Перевага апаратних засобів захисту в порівнянні із програмними, полягає в підвищенні швидкості роботи та неможливості скопіювати алгоритм шифрації;

– графічних прискорювачів для відтворення реалістичної графіки та обробки зображень;

– реалізація комунікаційних протоколів нижніх рівнів. Такі задачі традиційно вирішувалася з використанням апаратних засобів для формування бітових потоків і їхнього первинного декодування, встановлення з'єднань абонентів, тощо. Застосування програмованої логіки робить можливим і є економічною ефективною реалізацією в апаратному середовищі інтелектуальної обробки, включаючи ущільнення, буферизацію, пакетування даних і т.д.

Таким чином, у зв'язку з постійним ростом рівня інтеграції програмованих логічних інтегральних схем, зниженням їхньої вартості, розвитком засобів проектування галузі їх застосування постійно розширюються.

Властивості репрограмування ПЛІС

На сьогоднішній день ПЛІС, що репрограмуються, забезпечують нові можливості структур, побудованих на їх основі. В класичному випадку можливість реконфігурування системи на ПЛІС відбувається на завершальній стадії розробки проекту. Але сучасні засоби проектування та конструкція мікросхем програмованої логіки дозволяють виконувати повну чи часткову реконфігурацію в процесі роботи системи. Маючи в своєму розпорядженні плату, що містить одну або декілька ПЛІС, і підключаючи її до комп'ютера стандартними засобами, можна отримати кілька різних спеціалізованих пристроїв. Досить завантажити в мікросхему одну з можливих конфігурацій із числа, тих що зберігаються на твердому диску комп'ютера, щоб викликати до виконання той або інший спеціалізований алгоритм обробки даних. Також, користувач може самостійно модифікувати роботу цього, пристрою, забезпечуючи реалізацію тих чи інших специфічних функцій, що не передбачені заздалегідь.

IP-блоки та SoC

Однією із сучасних тенденцій у використанні ПЛІС є створення таких форм цифрових пристроїв як, наприклад, система на кристалі - SoC (System On a Chip), мережа на одному кристалі - NoC (Network On a Chip) та інші.

При проектуванні сучасних систем на кристалі широко використовуються готові модулі IP-блоки або IP core (intellectual property core), які реалізують досить складні функції перетворення і обробки даних. IP-блок це складний, вже протестований, верифікований і оптимізований функціональний модуль, який може бути багато разів використаний як компонент при проектуванні більш складних цифрових виробів з метою скорочення часу їх розробки. Як показала практика, використання IP дозволяє скоротити час до моменту появи цифрового виробу на ринку, спрощує процес проектування шляхом створення інтерфейсу для зв'язку системи з IP, мінімізує ризики проектування завдяки включенню вже відлагоджених модулів, зменшує час верифікації всієї системи. Сучасний ринок IP-компонет постійно розвивається та розширюється.

Система на кристалі - це спеціалізований цифровий виріб, конструктивно виконаний на кремнієвому кристалі, що характеризується низьким енергоспоживанням, малими розмірами, високою швидкодією і має наступні основні компоненти: процесор, пам'ять, логіку, інтерфейс вводу-виводу, перетворювачі інформації, шини обміну даними та інфраструктуру сервісного обслуговування.

Функціональні блоки SoC орієнтовані на ефективне вирішення спеціалізованої задачі на відмінність від мікроконтролера, що представляє собою універсальний обчислювач.

Перевагами SoC є прозорість щодо наявності вбудованих компонентів, висока швидкодія при низькому енергоспоживанні, мініатюрні розміри, ефективне використання різних типів пам'яті, висока надійність, низька вартість готового виробу та високий коефіцієнт використання функціональностей.

До недоліків SoC можна віднести значний час тестування та верифікації проекту, висока вартість розробки та макетування, складність відлагодження та виробництва, складність інтеграція IP-блоків з різних джерел та , іноді, висока їх вартість.

До однієї із форм еволюціонування систем на кристалі можна віднести мережу на кристалі (NoC). NoC це спеціалізований мультіядерний цифровий виріб конвеєрного типу, що має властивості системи на кристалі та доповнений інтелектуальною структурою комунікацій, що може підтримувати різні архітектури і рівні протоколів обміну даними. Як ядро може виступати IP-core (процесор або блок пам'яті), які організуються в масштабовані сегменти. Такій ієрархії ставиться у відповідність дворівнева система комунікацій і конвейеризації. Підтримка всіх рівнів протоколу обміну даними забезпечує високу швидкодію і пропускну здатність інформаційних каналів. Наявність інтелектуальних мультіядерної архітектури і комутаторів забезпечує високий рівень розпаралелювання обчислювальних процесів та вводу/виводу даних.

Програмні та апаратні процесорні ядра

Як вже було сказано, для реалізації систем на кристалі як елементної бази все частіше використовують ПЛІС. Основні виробники кристалів з програмованою логікою для підтримки даного напрямку (тобто SoC) розробили мікропроцесорні ядра (системи), а також спеціальні засоби для проектування та відлагодження вбудованих систем.

Використання передових технологій в процесі виробництва ПЛІС призвело до значного розширення їх функціональних можливостей У першу чергу ця тенденція стосується ПЛІС з архітектурою FPGA. Внаслідок цього, в процесі проектування цифрових пристроїв на їх основі, ПЛІС розглядається як деяке мікропроцесорне ядро, що оточене вільними логічними ресурсами, що конфігуруються. На основі цих ресурсів реалізуються будь-які необхідні периферійні пристрої в системі, що проектується.

За способом реалізації мікропроцесорні ядра, що використовуються при проектуванні систем на кристалах на основі ПЛІС поділяються на програмні (soft processor, soft-cores, soft-ядра) та апаратні (Hard Processor).

Програмні ядра або ядра, що конфігуруються, це ядра, які формуються на основі стандартних логічних ресурсів кристалів. Наприклад, сюди можна віднести сімейство 32-розрядних мікропроцесорних ядер MicroBlaze™ [3,4]. Перевагами мікропроцесорних ядер у порівнянні з апаратними є висока гнучкість, низька собівартість, відносно невеликий обсяг використовуваних ресурсів кристалів, можливість застосування в проектах, що реалізуються на базі найпоширеніших і доступних сімейств ПЛІС.

Апаратні процесори - мікропроцесорні ядра, які виконані у вигляді відповідних інтегрованих апаратних блоків ПЛІС. Головна перевага апаратних мікропроцесорних ядер, на прикладі сімейства PowerPC, полягає в можливості функціонування з високими тактовими частотами. Наслідком цього є більш висока швидкодія у порівнянні з мікропроцесорними ядрами, що конфігуруються. Недоліками апаратних ядер є обмежена кількість кристалів, в яких вони застосовуються, та їх висока вартість.

Приклад розробки багатопроцесорної системи на ПЛІС

Розглянемо, як приклад, варіант реалізації спеціалізованого пристрою з використанням особливостей сучасних ПЛІС. Виникненню проблем, що стосуються визначення запозичень в студентських роботах, наукових статтях, інтернет-ресурсах та інших текстах, які мають нести оригінальну складову, сприяє антропогенний вплив факторів широкого використання комп'ютерних інформаційно-пошукових сервісів, які дозволяють швидко знаходити та використовувати оригінальні тексти з порушенням існуючих законодавчих та моральних норм. Відомий ряд програмних систем, наприклад [5], що призначені автоматизувати процес знаходження запозичень, тобто визначення такої когнітивної властивості інформації, як її

оригінальність [6]. Основною операцією для значної частини функцій згаданих вище систем є операція порівняння інформаційних об'єктів. Більшість програмних сервісів бере її за основу, наприклад, системи для автоматизованого виявлення рівня запозичень в текстах. Порівняння є ключовою складовою, від якої залежить ефективне функціонування засобів порівняння інформаційних об'єктів на початковому рівні (рівні символів).

Одним з алгоритмів виявлення запозичень є алгоритм адаптивного порівняння [6], де запозичення виявляються шляхом порівняння окремих символів (наприклад – байтів) двох послідовностей з обмеженням деякої довжини співпадання, яка не береться до уваги. Оскільки в даному алгоритмі особливості семантичної структури даних не використовуються, то він є універсальним щодо інформаційних об'єктів різної природи. Тобто за його допомогою можна виявляти запозичення в україномовних текстах, так і більшості інших мов, запозичення в аудіо та відео файлах, програмах на мовах програмування та виконавчих програмах і т.п.

Основною проблемою систем виявлення запозичень є мінімізація часу, необхідного для обробки піддослідного файлу. Враховуючи великий обсяг бази зразків, який із кожним роком збільшується, для виявлення запозичень може знадобитись великий проміжок часу, недопустимий для широкого практичного використання. В алгоритмі адаптивного порівняння, також, був запропонований етап швидкісного порівняння [6] (швидкісне адаптивне порівняння). Основна ідея швидкісного етапу полягає у формуванні не точного, а приблизного рівня запозичень, але який гарантовано не менший за реально існуючий.

Використання пристроїв на базі сучасних ПЛІС для реалізації алгоритму детального адаптивного порівняння послідовностей символів може дозволити отримати спеціалізований апаратний комплекс, що значно прискорить процес виявлення запозичень і визначення їх кількісної величини та може використовуватись або окремо з доступом до даних на дискових накопичувачах, через обчислювальну мережу доступу до архівів даних, тощо.

Як було описано вище, основні виробники програмованих логічних пристроїв пропонують до використання в складі нових ПЛІС крім готових апаратних блоків ще й процесори, що можуть бути реалізовані програмним способом (soft процесори). Ці процесорні soft-ядра RISC архітектури, які мають свою систему команд, можуть бути оптимізовані за швидкодію для виконання таких завдань, як обробка символічних послідовностей за рахунок введення спеціалізованих інструкцій користувача. Зазвичай, спеціальні інструкції набагато краще враховують особливості алгоритмів для вирішення спеціалізованих задач.

В роботі [7] було запропоновано спеціалізовані команди soft процесора для реалізації на ПЛІС алгоритму швидкісного адаптивного порівняння не тільки символічних, а й будь-яких послідовностей, якщо вважати, що для представлення кожного елемента цих послідовностей достатньо 1 байта (символа). А при врахуванні можливості використання декількох soft процесорних ядер в складі системи разом з апаратними блоками, то це дає можливість отримати багато процесорну систему на одній ПЛІС, що має значну перевагу в швидкодії перед однопроцесорною системою.

Тепер розглянемо основні аспекти використання в такій системі декількох процесорних ядер. На сьогоднішній день використання багато процесорної програмованої логіки в існуючих системах, що пов'язані з обробкою великих масивів даних (системи зв'язку, мультимедіа, тощо) підтвердило свою ефективність. Так, наприклад, сучасні ПЛІС компанії Xilinx[®] дозволяють розмістити декілька [8] soft процесорів MicroBlaze[™]. Слід зауважити, що можливе використання такого soft-ядра та його системи команд не буде відрізнятися від реалізації цього алгоритму на базі системи аналогічних команд будь-якого іншого процесора. Додаткові процесори можуть виступати у якості сопроцесорів до основного або керуючого процесора, чи незалежно від нього виконувати різні обробки. Причому в такій системі можливо використовувати soft процесорні ядра на різних частотах. Можливі наступні комбінації побудови багато процесорної системи для вирішення поставленої задачі:

- система працює з однією послідовністю, обробка якої розпаралелюється між декількома процесорами керуючим модулем. Такий спосіб не вимагає зміни мікроінструкцій системи команд користувача;

- для обробки однієї піддослідної послідовності, що надходить в систему з високою швидкістю у вигляді потоку, можна організувати процесорний конвеєр, де кожний процесор буде виконувати свій етап обробки символічних даних або, іншими словами, виконувати визначену дію алгоритму швидкісного адаптивного порівняння, і передавати проміжні результати наступному процесору для наступної обробки. Цей спосіб дозволяє збільшити пропускну спроможність

системи. Кожне процесорне ядро буде оптимізовано під інструкцію, яка відображає певний етап алгоритму швидкісного порівняння.

Висновки

Протириччя між бажаністю і можливостями широкого застосування ВІС/НВІС у спеціалізованих мікропроцесорних пристроях для виконання нетрадиційних комп'ютерних операцій на базі елементарних, спонукало до широкого використання програмованих логічних структур. Завдяки програмуванню структур для електронної промисловості здійснилася універсалізація ВІС/НВІС, що зробило їх придатними для широкого кола споживачів. Це було досягнуто виготовленням промисловим способом деяких заготовок, які далі перетворюються споживачем у потрібний йому пристрій. У цьому випадку застосування таких, програмованих ВІС/НВІС для вирішення спеціалізованих задач навіть для малотиражних виробів стає можливим і ефективним. Створення програмованих логічних інтегральних схем, що мають назву ПЛІС, дозволила використати їх там, де раніше доводилося застосовувати спеціалізовані замовні ВІС/НВІС.

Розглянуто варіант вирішення актуальної проблеми для освітньої та наукової галузі – проблеми виявлення запозичень в роботах, що мають нести оригінальну складову за допомогою використання спеціалізованого процесорного пристрою на ПЛІС, що реалізує алгоритм швидкісного адаптивного порівняння інформаційних об'єктів і дозволяє досягти показників швидкодії обробки символічних послідовностей, що перевищують показники системи на базі стандартних інструкцій. Даний пристрій може бути виконаний у вигляді параметричного модуля з можливістю оперативного репрограмування. Один із способів удосконалення такої системи з точки зору підвищення швидкодії полягає в можливостях сучасних ПЛІС імплементувати в систему декілька soft процесорних ядер з використанням оптимізованих процесорних команд користувача.

Зважаючи на методичні і дидактичні особливості навчання в Технічній школі, а також на досвід НТУУ „КПІ” пропонується варіант розподілу лекційного матеріалу для навчальної програми з курсу "Проектування спеціалізованих комп'ютерних систем на ПЛІС" для студентів освітнього напрямку "Комп'ютерна інженерія" за наступними темами:

Тема 1. Основи архітектури ПЛІС.

Структура програмованих логічних матриць (PLA). Структура програмованих матриць логіки (PAL). Структура комплексних програмованих логічних пристроїв (CPLD). Структура програмованих користувачем вентильних матриць (FPGA).

Тема 2. Серії мікросхем ПЛІС.

Основні параметри та порівняльна характеристика серій мікросхем ПЛІС, які випускаються провідними фірмами світу.

Тема 3. Загальна методика проектування на основі ПЛІС.

Опис, синтез та моделювання проекту. Програмування ПЛІС. Алгоритми проектування на основі ПЛІС CPLD та FPGA.

Тема 4. Системи автоматизації проектування схем на основі ПЛІС.

Структура САПР ПЛІС. Функціональне призначення основних складових частин. Порівняльна характеристика САПР ПЛІС провідних фірм - Altera, Xilinx та Actel. Критерії вибору засобів проектування.

Тема 5. Синтезуючі описи схем в VHDL.

Подібність та розбіжності систем моделювання та синтезу в VHDL. Синтезуюча підмножина мови VHDL. Цільова бібліотека синтезу. Кодування даних при синтезі. Синтезовані описи комбінаційних схем та схем з пам'яттю.

Тема 6. Проектування типових схем на ПЛІС.

Проектування генераторів, тригерів, регістрів, лічильників, дешифраторів, мультиплекторів, компараторів, суматорів, перемножувачів тощо з використанням цільової бібліотеки в графічному режимі та з використанням описів на мові VHDL.

Тема 7. Апаратна реалізація довільних алгоритмів на ПЛІС.

Засоби ієрархічного подання складних схем. Особливості подання складних схем в графічному режимі, листи проекту та макроси. Особливості подання складних схем в синтезуючих описах в VHDL. Управління синтезом.

Тема 8. Особливості розробки цифрових пристроїв з використанням мікросхем ПЛІС.

Способи організації автоматичної настройки мікросхем ПЛІС. Організація динамічного реконфігурації. Питання синхронізації, ведучі та ведені мікросхеми.

Тема 9. Створення проекту цифрового пристрою на ПЛІС з використанням ІР-компонент.

ІР-ядра. Визначення архітектури та розбиття на окремі ІР-ядра. Адаптація проекту та окремих ІР-ядер на різні сімейства ПЛІС.

Тема 10. Розробка системи на кристалі (SoC).

Інтеграція цифрових елементів та елементів управління на одному кристалі (RISC-процесор, DSP, апаратні блоки обробки сигналів, периферійні пристрої, пам'ять та інтерфейс з пам'яттю).

Тема 11. Розробка одно- та багатопроцесорних систем на базі ПЛІС.

Вибір архітектури процесорного ядра. Інтеграція готових ядер від виробників кристалів програмованої логіки. Адаптація процесорних ядер під кристал, організація підсистеми ядра (інтерфейси, системні шини, контролери пам'яті, пристрої діагностики).

1. Тарасенко В.П. Сучасні ситуативно-методологічні аспекти створення спеціалізованих комп'ютерних систем/ В.П. Тарасенко, А.О. Мельник // Наукові вісті НТУУ "КПІ", 1997, -№1, - С.18-21
2. Грушвицкий, Р.И. Проектирование систем на микросхемах программируемой логики / Р.И. Грушвицкий, А.Х. Мурсаев, Е.П. Угрюмов. - СПб.: БХВ-Петербург, 2002. - 608 с.
3. Зотов, В.Ю. Проектирование встраиваемых микропроцессорных систем на основе ПЛИС фирмы Xilinx / В.Ю. Зотов. - М.: Горячая линия-Телеком, 2006. - 520 с.
4. Кузелин М.О. Современные семейства ПЛИС фирмы XILINX / М.О. Кузелин, Д.А. Кнышев, В.Ю. Зотов –М.: «Горячая линия – Телеком», 2004, 440 с.
5. <http://www.antiplagiat.ru>
6. Тарасенко В.П. Автоматизація оцінки оригінальності інформації./В.П.Тарасенко, А.Ю.Михайлюк, О.К. Тесленко, О.С. Осипов // Наукові записки українського науково-дослідного інституту зв'язку. №1, 2007.
7. Тарасенко В.П. Команди спеціалізованого процесора на ПЛІС для адаптивного порівняння інформаційних об'єктів / В.П.Тарасенко, О.К.Тесленко, Я.М.Клятченко А.Ю.Михайлюк// НАУ "ХАІ", науково-технічний журнал"Радіоелектронні і комп'ютерні системи", №7, 2010р. с.220-225.
8. <http://www.xilinx.com>