УДК 004.9 **doi: 10.26906/SUNZ.2023.2.106**

О. А. Ільяшов, К. В. Покора, В. О. Дяченко, А. А. Коваленко

Харківський національний технічний університет радіоелектроніки, Харків, Україна

КЛАСИФІКАЦІЯ ДАНИХ АПАРАТНИМИ ПРИСКОРЮВАЧАМИ FPGA У ЦЕНТРАХ ОБРОБКИ ДАНИХ ТА ХМАРАХ

Анотація. Актуальність. Аналіз даних, зокрема їх класифікація) часто виконується за допомогою методів машинного навчання. Часто задіяні алгоритми потрібні мати справу з великими наборами даних, що призводить до тривалого часу виконання. Таким чином, актуальним є дослідження апаратних прискорювачів, на базі програмованої вентильної матриці (FPGA) для покращення продуктивності. FPGA є перспективним рішенням для апаратного прискорення, конфігурації після виробництва та можливості перепрограмування. **Метою даної роботи** є дослідження та аналіз тенденцій у існуючих хмарних архітектурах FPGA, які підкреслюють складний зв'язок між архітектурами та системними вимогами та призначені для вирішення завдач класифікації даних методами машинного навчання. Це дозволяє нам ідентифікувати нові архітектури, які, ймовірно, запропонують значні переваги для хмарних робочих навантажень. **Об'єктом дослідження** є процес еволюції FPGA-прискорювачів для обчислень у центрах обробки даних (ЦОД) та хмарах. **Предметом дослідження** є методи та алгоритми дослідження хмарних архітектур FPGA на основі таксономічних категорій. **Результати.** У роботі обговорюється майбутнє використання FPGA у центрах обробки даних та хмарах. Також у роботі досліджуються поточні архітектури та обговорюється масштабованість і абстракції, які підтримуються операційними системами, проміжним програмним забезпеченням і віртуалізацією. Висновок. Розроблено класифікацію щодо дослідження хмарних архітектур FPGA на основі таксономічних категорій. Розглянута та запропонована архітектурна організація щодо розгортання додатків FPGA, що використовуються в хмарних середовищах і середовищах центрів обробки даних. Запропоновані дві моделі проектування додатків машинного навчання для класифікації даних з використанням апаратних FPGA-прискорювачів.

Ключові слова: реконфігурована логіка, FPGA прискорення, хмарні обчислення, центр обробки даних, віртуалізація, таксономічні категорії, класифікація даних, машинне навчання, програмне забезпечення.

Вступ

Майбутнє штучного інтелекту (AI), зокрема машинного навчання (МН), наближається, і воно залежить від нього. Штучний інтелект (AI) став однією з найпопулярніших обчислювальних технологій у світі, і компанії, такі як Intel, Google, Microsoft, IBM та інші, всі вони спираються на FPGAs. Масштабно-програмовані вентильні матриці (FPGAs) стали більш, ніж екзотичною та нішевою технологією, яку можуть осягнути тільки магіки апаратного забезпечення. Гравці такого рівня, як Alibaba, Amazon, Baidu, Huawei та Tencent, тепер використовують FPGAs для розробки додатків у своїх інфраструктурах дата-центрів. Інші використовують FPGAs для надання додатків як сервіс (Microsoft [2] та Nimbix) або для внутрішньо розроблених додатків. Крім того, по всьому світу здійснюється ряд проектів в академічних та інших дослідницьких організаціях з метою надання користувачам віддаленої можливості прискорення та гнучкості. Хоча FPGAs стають все більш доступними у дата-центрах, їх довгострокове прийняття в цих місцях не гарантоване. Архітектура та інтеграція на рівні плати та системи повинні забезпечити, щоб застосунки використовували переваги переконфігурації та пом'якшували його недоліки. Надання апаратних ресурсів вимагає відповідного проміжного програмного забезпечення, апаратної віртуалізації та механізмів розділення доменів.

З усього світу, в академічних та інших наукових організаціях, все більше запускається проектів, щоб надавати користувачам віддалену можливість прискорення та гнучкість. Хоча FPGAs стають все більш доступними у датацентрах, їх довгострокове використання у таких місцях не гарантується. Архітектура та інтеграція на рівні плати та системи повинна забезпечити те, що застосунки використовують переваги переконфігурування та зменшують його недоліки. Постачання апаратних ресурсів потребує відповідної проміжної програми, апаратної віртуалізації та механізмів розділення доменів. Ефективне та гнучке постачання апаратних ресурсів збільшує можливість безперешкодної інтеграції, ідеально без переконструювання, в існуючу інфраструктуру керування хмарою. Просторовий спільний доступ до FPGA збільшує складність планувальників та менеджерів ресурсів хмари. На щастя, великий обсяг робіт, розроблених за останнє десятиліття, може бути використаний у системах планування завдань хмари, які в даний час використовуються в датацентрах, таких як списокове планування, що використовується в Amazon Cloud. Просторовий спільний доступ до FPGA збільшує складність планування та менеджменту ресурсів хмари.

Метою цієї роботи є дослідження та аналіз тенденцій у існуючих хмарних архітектурах FPGA, які підкреслюють складний зв'язок між архітектурами та системними вимогами та призначені для вирішення завдач класифікації даних методами машинного навчання. Це дозволяє нам ідентифікувати нові архітектури, які, ймовірно, запропонують значні переваги для хмарних робочих навантажень.

Основна частина

На зміну універсальним обчисленням приходять гетерогенні: звичайні ЦП доповнюються спеціалізованими процесорами або прискорювачами, що підвищують продуктивність та енергоефективність серверів при ресурсомістких робочих навантаженнях, які стають звичайним явищем у сучасному датацентрі. В якості таких прискорювачів обчислень

використовуються графічні процесори (GPU), програмовані користувачем вентильні матриці (FPGA) та інтегральні схеми спеціального призначення (ASIC). Нові робочі навантаження визначають нові системні вимоги та функції, сприяють розвитку та впровадженню таких технологій, як NVMe, FPGA та бази даних у оперативній пам'яті (in-memory). За даними аналітиків Forrester, спеціалізовані прискорювачі, такі як FGPA та GPU, будуть використовуватися все ширше. Застосування графічних процесорів може приблизно в 25 разів прискорити виконання окремих видів обчислень, відзначають у 451 Research [6]. У ієрархії процесорів - від ЦП загального призначення до ASIC - існує компроміс між гнучкістю та ефективністю. При цьому, як стверджують у Accenture, коли якась конкретна програма реалізується за допомогою спеціалізованих рішень, продуктивність може збільшуватися на порядки (табл. 1).

Таблиця 1 – **Відносна ієрархія апаратних прискорювачів**

Апаратне рішення	CPU	GPU	FPGA	ASIC
Призначення	Для додатків загального призначення	Для обчислень пов'язаних з графікою	Масив програмованих блоків з програмованим міжз'єднанням	Розробляються для конкретної функціональності
Відносна продуктивність	1	100	1000	10000-100000
Гнучкість	Універсальний	Спеціалізований	Програмований	Для спеціального застосування
Ринок	Широкий	Обмежений	Обмежений	Конкретні ніші ринку
Простота програмування	Широко доступні навички програмування	Потребує спеціальних знань	Потребує спеціальних знань	Тільки інтерфейс
Ключові гравці	Intel, AMD, ARM	NVIDIA, AMD, Intel	Xilinx, Intel (Altera), Actel	NEC, LSI, Samsung

GPU, FPGA та ASIC можуть використовуватися для прискорення завдань віртуалізації та хмарних обчислень, 3D/2D-графіки, високопродуктивних обчислень (HPC), штучного інтелекту та машинного навчання, аналізу великих даних, віртуалізації мережевих функцій (NFV) тощо. Наприклад, їх можна успішно застосовувати в застосунках машинного навчання, призначених для розпізнавання голосу та зображень або аналізу тексту.

Комбінування універсальних процесорних ядер зі спеціалізованими блоками обчислень за нейромережевими алгоритмами – актуальна тенденція. Спеціалізовані мікросхеми для систем штучного інтелекту – ще один новий ринок, що стрімко розвивається.

Свій ASIC – Edge TPU – для задач ШІ випустила й Google. Проте зараз у сфері обробки даних з застосуванням ШІ найбільш відомі продукти NVIDIA з її графічними процесорами та рішення Intel, такі як Nervana Neural Network Processor (NNP-I), заявлена продуктивність якого щодо штучного інтелекту в 10 разів вища, ніж у графічних карт, а також лінійка FPGA Agilex для задач ШІ.

Кожен тип процесора, універсального (CPU), графічного (GPU) або FPGA, має свої переваги, інакше їх просто не виробляли б. CPU забезпечують хорошу продуктивність при найвищій універсальності та застосовності. Приблизно 99% всіх існуючих програм написані для виконання під CPU. GPU мають більшу кількість ядер та векторну архітектуру, високу швидкість обміну з пам'яттю та введення-виведення. FPGA мають найбільшу продуктивність на Ватт споживаної потужності завдяки властивостям апаратури, але потребують дуже ретельного та складного програмування.

Нижче про ці відмінності трохи детальніше:

Універсальні процесори CPU - це в суті, працюючі нині IT-індустрії. їх можна використовувати для найрізноманітніших завдань, але через свою архітектуру CPU не настільки ефективні для паралельних обчислень. Останнім часом ця проблема частково вирішується за рахунок імплементації в чипі процесора багатьох ядер. Однак навіть у найпродуктивніших CPU кількість ядер поки що вимірюється декількома десятками.

Графічні процесори (GPU) довгі роки працювали тільки в ніші відображення інформації на екрані. І лише відносно недавно GPU стали застосовуватися для завдань високопродуктивних обчислень, в тому числі майнінгу криптовалют. Робота з графікою як векторними завданнями призвела до такого розвитку архітектури GPU, яка стала пристосованою для цілей паралельних обчислень. Як результат, сучасна архітектура графічного процесора дозволяє прискорити прохід векторизованих даних через свої конвеєри, які в іншому випадку довелося б запускати через багато інших логічних блоків у ЦП з відповідною втратою продуктивності. Сучасні GPU містять у чипі кілька тисяч процесорних ядер.

FPGA, на відміну від універсальних і графічних процесорів, можна перепрограмувати згідно з особливостями вирішуваної на них обчислювальної задачі. В результаті отримується синтез спеціалізованого процесора для конкретної задачі. Іншими важливими відмінностями FPGA є знижене споживання енергії на одиницю обчислювальної потужності, а також архітектура з паралельним виконанням багатьох векторних операцій одночасно - так звана масивно-паралельна дрібнозерниста архітектура. Кількість ядер у чипі FPGA може досягати одного мільйона і більше.

FPGA-прискорювач представляє собою адаптер, який крім чипа FPGA містить на платі локальну оперативну пам'ять типу HBM (пам'ять DRAM з високою пропускною здатністю) та високошвидкісні інтерфейси введення-виведення, такі як 10/40 GE та PCI Express. FPGA-прискорювачі також випуска-

ються у форм-факторі SOM (як одномодульні системи). Кожен чіп FPGA містить понад мільйон комірок, що можуть бути перепрограмовані для різних функцій - кеш-пам'ять, сигнальні процесори, блоки цифрової обробки, блоки інтерфейсів.

На сьогодні FPGA перевершує ASIC у продуктивності завдяки більш сучасним технологічним процесам створення кристалів. Для FPGA використовуються техпроцеси рівня 20 нм і навіть 14 нм. Тоді як для створення кристалів ASIC використовуються більш "старі" техпроцеси рівня 60 нм. Відповідно, на тій же площі кристала у FPGA можна розмістити у декілька разів більше логічних комірок, ніж у ASIC, що забезпечує перевагу в продуктивності.

У хмарних обчисленнях FPGA застосовуються для швидкого розрахунку, прискорення мережевого трафіку та доступу до масивів даних. Сюди також можна віднести використання FPGA для високочастотної торгівлі на біржах. У сервери вставляють плати FPGA з PCI Express та оптичним мережевим інтерфейсом виробництва Intel (Altera) або Xilinx.

На FPGA чудово лягають криптографічні алгоритми, порівняння послідовностей ДНК та наукові задачі, такі як молекулярна динаміка. У Microsoft давно використовують FPGA для прискорення пошукового сервісу Bing, а також для організації Software Defined Networking всередині хмари Azure.

Бум машинного навчання також не обійшов FPGA стороною. Компанії Xilinx та Intel пропонують засоби на основі FPGA для роботи з глибокими нейронними мережами. Вони дозволяють отримувати прошивки FPGA, які реалізують ту чи іншу мережу безпосередньо з фреймворків, таких як Caffe та TensorFlow.

До того ж все це можна спробувати, не виходячи з дому та використовуючи хмарні сервіси. Наприклад, в Amazon можна орендувати віртуальну машину з доступом до плати FPGA та будь-яких засобів розробки, в тому числі й машинного навчання.

На сьогоднішній день більшість хмарних провайдерів зосередилися на платформах більш загального призначення, здаючи в оренду стандартизовану масштабовану інфраструктуру. Пропозиції в основному відрізняються кількістю (віртуальних) ядер, обсягом пам’яті та пропускною спроможністю мережі на ЦП і сховище. Пропозиції з більш спеціалізованим апаратним забезпеченням, насамперед із графічним процесором, є переважно окремими спеціалізованими пропозиціями, наприклад, для високопродуктивних обчислень (HPC), а не елементами, які забезпечують широку основу архітектури хмарної платформи.

Деякі хмарні провайдери почали використовувати спеціалізоване обладнання (включаючи FPGA) [2] і навіть почали розробляти власні SoC [5] замість того, щоб покладатися на кремнієвий процесор від постачальника, який відкриває двері для більшої диференціації платформ на основі апаратного забезпечення. Це передбачається як хмарна інфраструктура прогресує від інфраструктури як послуги (IaaS) до віртуалізованих платформ, а потім до мікросервісів і функції як послуги (FaaS), хмарні обчислення також можуть надати екосистеми програмного забезпечення, які сприятимуть зростанню впровадження спеціалізованого апаратного забезпечення.

Хмарні архітектури FPGA.

Попит на послуги дата-центрів має шанс продовжувати експоненційне зростання [1]. Оскільки закон Мура сповільнився, а обчислювальні витрати та складність хмарних завдань продовжують зростати та еволюціонувати, ПЛІС пропонує перспективний механізм для вирішення парадоксу, пов'язаного з поєднанням продуктивності, енергоефективності та гнучкості. Вони можуть бути встановлені майже будь-де в дата-центрі для прискорення обчислень, зберігання та мереж, як показано на рис. 1. Завдяки неоціненим перевагам ПЛІС їх використання в дата-центрах очікується зростати з гігантською річною зростанням в 48% між 2020 та 2027 роками [2,10].

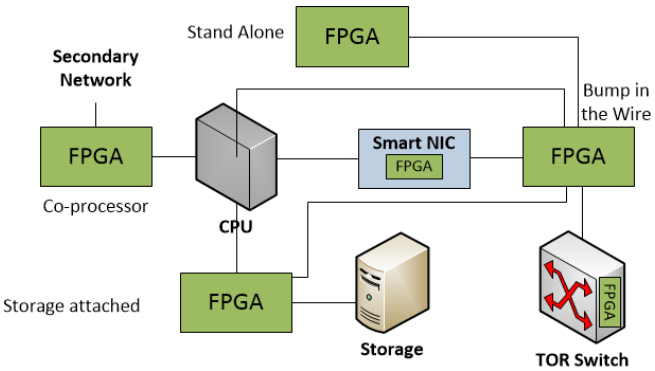

Рис. 1. Розгортання FPGAs в центрі обробки даних

Незважаючи на те, що ПЛІС пропонують низку переваг для дата-центрів, вибір певної архітектури для використання є складним та непростим; ми не можемо просто розміщувати будь-яку кількість та розмір цих пристроїв в будь-якому місці у хмарі. Екстремальна потреба в ефективності витрат приводить до акценту на розмірі, енергоспоживанні, охолодженні, сумісності, автоматизації та можливості оновлення на місці, забезпечуючи при цьому задоволення певних потреб хмарних завдань у термінах продуктивності, пам'яті, можливості сервера, надійності, безпеки та підключення до мережі.

Cloud FPGA - це інфраструктура пристроїв FPGA або програмних засобів проектування, які доступні в хмарі. Щоб використовувати переваги "розміщення FPGA в хмарі", ми пропонуємо класифікувати cloud FPGA за трьома рівнями сервісів:

1) FPGA програмні засоби як сервіс: з моделі хмарного SaaS засоби FPGA стали дематеріалізовані з 2010 року [4]. Ця модель отримала користь від масштабної обчислювальної потужності хмари без потреби в процесі ускладненого налаштування інструментів. Ця модель надає легкий доступ до хмари для апаратних проектувальників без складності інфраструктури та програмного забезпечення, але не дає доступу до фізичного FPGA.

2) FPGA платформи як сервіс: з моделі хмарного PaaS платформи FPGA були дематеріалізовані до 2010 року [5, 6]. Немає потреби в придбанні FPGA платформ. Ця модель дозволяє дизайнерам додатків отримувати доступ до однієї або кількох FPGA плат-

форм. Дизайнер може розробляти та впроваджувати свої хмарні додатки на старих або нових платформах FPGA.

3) FPGA ресурси як сервіс: з моделі хмарного IaaS FPGA та його ресурси стали віртуалізованими з 2014 року [7]. У цій моделі FPGA розділено на кілька незалежних віртуальних FPGA областей. Ці області можуть надаватися кільком дизайнерам додатків в багатокористувацькому середовищі з віртуальним доступом до фізичного FPGA.

Ця класифікація може змінюватись з часом в залежності від різних критеріїв через появу нових застосувань або зміни вимог. Залежно від сервісу, запропонованого для хмари FPGA, можна вирішити раніше поставлені виклики.

Використання FPGA в хмарі полягає у віддаленій оренді набору конкретних програмних засобів, платформ або ресурсів FPGA у вартісно-ефективний спосіб. Така FPGA-забезпечена хмара зберігає переваги FPGA (наприклад, низьке споживання енергії та програмованість) і забезпечує масштабованість, гнучкість та багатокористувацьку обслуговування.

Надання ресурсів FPGA подібно до надання традиційних хмар на основі центральних процесорів (CPU) та графічних процесорів (GPU). Щодо категорій сервісів у традиційному обчислюванні в хмарі, провайдери хмари FPGA пропонують FPGA як інфраструктуру як сервіс (IaaS) або програмне забезпечення як сервіс (SaaS) [6]. Рисунок 2 наводить приклад ієрархічного відображення в FPGA-хмарі. Немає стандартного визначення або класифікації для FPGA-хмар, і ієрархічне відображення може змінюватись з часом. "Vendor manage (optional)" та "User manage (optional)" вказують на те, що ця ієрархія не завжди існує в хмарі FPGA, і вона настроюється кожним виробником або користувачем хмари FPGA.

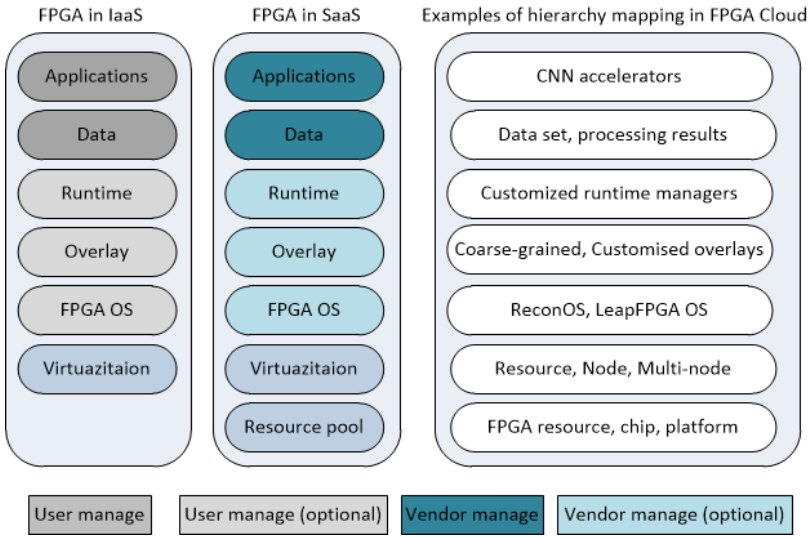

Рис. 2. IaaS та SaaS хмара FPGA

Фреймворк FPGA у хмарі відноситься до інфраструктури та програмного забезпечення, необхідного для використання FPGA в середовищі хмари. Фреймворк повинен забезпечувати спосіб відкриття FPGA як пулу ресурсів, які можуть бути виділені та звільнені арендарями. Це потребує відстеження використання FPGA для спрощення розрахунків.

Крім того, фреймворк повинен дозволяти арендарям програмувати FPGA. Традиційні системні стеки вважають FPGA нерухомими прискорювальними пристроями, не враховуючи їх програмованість. Для вирішення цих проблем потрібно додати нові модулі до обчислювальних вузлів, а планувальник в контролюючому вузлі повинен бути покращений. Системний стек обчислювальних вузлів можна розділити на чотири рівні: апаратний, гіпервізор, бібліотека та застосунок. Модулі FPGA повинні бути додані на рівень апаратного забезпечення та зроблені доступними як пул ресурсів для арендарів.

Рівень гіпервізора повинен бути змінений для підтримки віртуалізації FPGA, що дозволяє кільком арендарям ділитися однією FPGA без взаємодії. Рівень бібліотеки повинен забезпечувати програмні абстракції та інструменти, які дозволяють арендарям програмувати FPGA.

Нарешті, рівень застосунку повинен забезпечувати застосунки та сервіси, які можуть використовувати прискорення FPGA. Фреймворк також повинен вирішувати проблеми безпеки, такі як забезпечення того, що арендарі не можуть отримати доступ до FPGA.

FPGA мають бути представлені у стеку хмарних послуг як басейн ресурсів, який може бути активно керований, тобто його можна запитувати, виділяти та звільняти користувачами. Його використання повинно бути відстежене, щоб сприяти розрахунку, пов'язаному зі збором платежів за публічну хмарну модель. Крім того, якщо FPGA було надано орендареві, його повинно бути можливо програмувати орендарем, аналогічно до інших ресурсів, таких як ЦП та ГПУ. Однак, традиційні стеки системного програмного забезпечення, тобто операційна система та гіпервізор, вважають FPGA лише функціональними прискорювальними пристроями, ігноруючи їхню природу програмованості. Крім модулів, які зображені сірим кольором, на рис. 3 показані типові компоненти хмари на базі OpenStack.

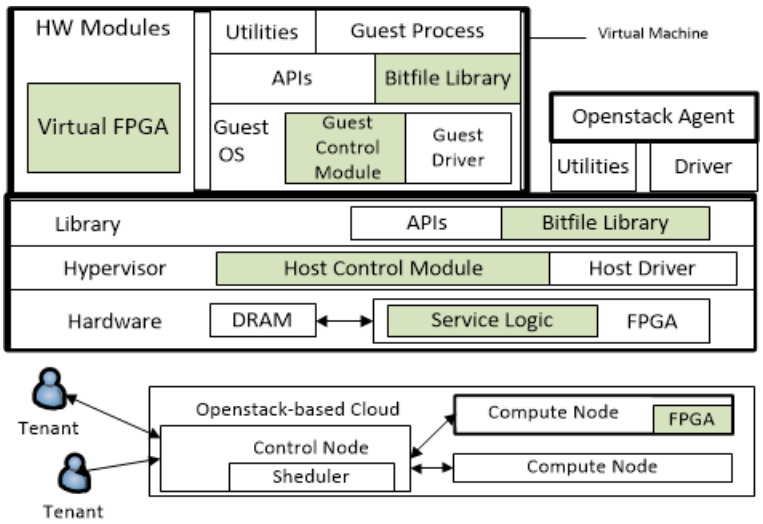

Рис. 3. Фреймворк FPGA у хмарі

Кілька вузлів обчислень, кожен з яких є фізичною машиною, надають фізичні ресурси, включаючи ЦП, пам'ять, диски та мережу. Керуючий вузол обробляє запити від користувачів, розподіляє ресурси та створює віртуальні машини (ВМ) на вибраних фізичних машинах. Після цього користувачі отримують доступ до своїх віртуальних машин та розгортають на них додатки.

Для введення FPGA в хмару ми надаємо нові модулі для вузлів обчислень, як показано сірим на рис. 2. Крім того, планувальник у керуючому вузлі OpenStack покращений. Щоб полегшити опис, ми розділили стек системи вузлів обчислень на 4 рівні: апаратний, гіпервізор, бібліотека та додаток.

Таксономія FPGA відноситься до системи класифікації, яка використовується для категоризації різних типів FPGAs на основі їх властивостей та можливостей. Існують кілька способів класифікації FPGAs, включаючи їх архітектуру, розмір, продуктивність та функціональність.

Один зі способів класифікації FPGA базується на їх архітектурі, яка може бути розподілена на три категорії: SRAM-базовані, anti-fuse-базовані та flash-базовані. SRAM-базовані FPGAs використовують лічильники пам'яті для зберігання конфігураційних даних, тоді як anti-fuse-базовані FPGAs використовують невідновлювальні плавки, які програмуються один раз і не можуть бути перепрограмовані. Flash-базовані FPGAs використовують клітини пам'яті flash для зберігання конфігураційних даних і можуть бути перепрограмовані багато разів.

FPGAs також можна класифікувати за їх розміром та продуктивністю, які зазвичай вимірюються за кількістю логічних елементів, тактовою частотою та споживанням енергії. Малі FPGAs зазвичай мають менше ніж 10 000 логічних елементів і використовуються для простих застосувань, тоді як великі FPGAs можуть мати понад 10 мільйонів логічних елементів і використовуються для складних застосувань, які вимагають високої продуктивності. Інший спосіб класифікації FPGAs базується на їх функціональності, яка може бути розподілена на загального призначення FPGAs та спеціалізовані FPGA (ASPs) [8-9].

Іншими словами таксономія базується на критичних аспектах будь-якої хмарної системи FPGA: тип плат FPGA, розміщення FPGA в системі, підключення до мережі, підключення до вузла та випадки використання. Ці таксономічні категорії не є ні взаємовиключними, ні комплексними: системи можуть мати кілька підкатегорій, а нові підкатегорії додаватимуться пізніше для інкапсуляції майбутніх інновацій.

Далі розглянемо аспект використання FPGA-прискорювачів методів машинного навчання при роботі з ЦОД та хмарами.

Однією з проблем реалізації машинного навчання є не тільки поділ на апаратне та програмне забезпечення. Як критична складова будь-якого рішення у галузі машинного навчання, проектування апаратного забезпечення становить серйозну проблему перед постачальниками FPGA-прискорювачів, які повинні мати можливість реалізовувати свою продукцію клієнтам за рахунок інструментальних програмних засобів, що дозволяють абстрагуватися від апаратного забезпечення. Таким чином, для виведення нових FPGA-прискорювачів на ринок їх розробникам доводиться залучати і фахівців у галузі створення програмного забезпечення (ПЗ). Апаратні структури, які критично важливі для обчислень машинного навчання, можуть створюватися розробниками апаратного забезпечення до того моменту, коли подальше вдосконалення моделі ще можна здійснити без проектування апаратної частини. Один з варіантів реалізації досягається просто шляхом створення екземпляра складного функціонального блоку, за допомогою якого буде керуватися інструментальним засобом вищого рівня. Деякі фірми-постачальники FPGA-прискорювачів поділяють процес проектування таким чином, щоб проектування апаратного забезпечення було значною мірою відокремлено від проектування моделі. Завдяки цьому FPGA стають доступнішими для розробників, які не є фахівцями у даній галузі розробки апаратного забезпечення. Ще одне питання стосується того, які інструментальні засоби слід використовувати, і яким чином проектування апаратного забезпечення було значною мірою відокремлено від проектування моделі.

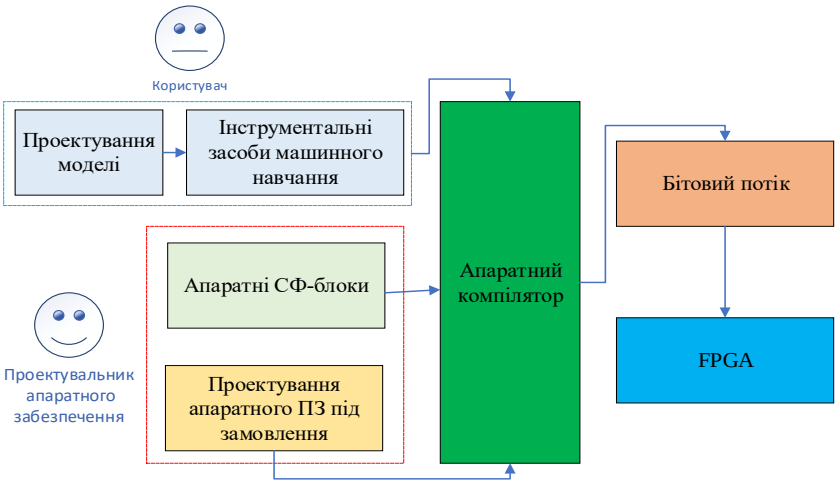

Рис. 4. Перша модель проектування додатків машинного навчання на FPGA

Детальна інформація про конструкцію FPGA зазвичай вказується в низькорівневому бітовому потоці, що завантажується в прилад. Але коли справа доходить до машинного навчання, деякі фірми реалізують у бітовому потоці всю модель, тоді як інші використовують його тільки для апаратної частини моделі, використовуючи для інформації про особливості моделі машинного навчання програмований двійковий файл. На рис. 4 всі аспекти конструкції відображені в одному бітовому потоці. Розробник може бути здатним здійснювати проектування на високому рівні, але інструментальні засоби все одно пропускатимуть інформацію через апаратний компілятор. На рис. 5 дані моделі високого рівня виділені в окремий двійковий файл.

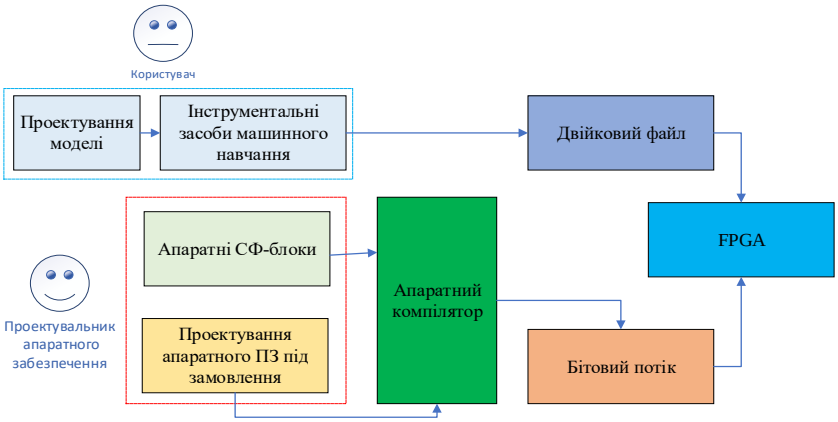

Рис. 5. Друга модель проектування додатків машинного навчання на FPGA

Сам момент при зміни моделі, якщо потрібна зміна бітового потоку апаратного забезпечення, визначається постачальником FPGA. Модель проектування має значення як для поточних модифікацій конструкції, та для оновлення приладів, які вже використовуються замовниками. У випадках, коли всі дані конструкції були включені в бітовий потік, можливі зміни в майбутньому для створення оновленого бітового потоку, який, в свою чергу, вимагатиме використання інструментальних апаратних засобів FPGA низького рівня, навіть якщо інструментальні засоби високого рівня дозволяють уникнути необхідності будь-якої явної зміни конструкції апаратного забезпечення. В інших випадках зміни в моделі призводять до змін лише у програмному двійковому файлі, який компілюється набагато швидше, ніж здійснюється повна рекомпіляція апаратного забезпечення. У цьому випадку основні апаратні засоби залишаться незмінними поряд із зміною аспектів моделі. Точна межа між змінами лише програмного забезпечення та змінами апаратного забезпечення залежатиме від конкретної архітектури.

Висновки

Поточні інфраструктури обчислення в хмарі на базі FPGA ефективно використовують наявні плати FPGA, встановлюють їх у сервери та надають доступ до них кінцевим користувачам. Це дозволило багатьом провайдерам хмарних послуг швидко розгорнути ці плати FPGA у своїх датацентрах і надати доступ до них кінцевим користувачам. Однак, на всіх рівнях архітектури апаратного та програмного забезпечення є багато можливостей для покращення. Від нових апаратних конструкцій (наприклад, нової інтеграції FPGA у ЦП), до конструкцій системи (наприклад, вибору типу бампера-на-проводі в порівнянні з типом прискорювача) та програмного забезпечення та програмування (наприклад, використання HLS або нових абстракцій). Крім того, безпека залишається відкритим завданням, оскільки низькорівневий доступ

до апаратного забезпечення FPGA дозволяє зловмисникам створювати нові датчикові схеми, які можуть сприяти витоку інформації.

Зокрема, доступність FPGA в хмарних датацентрах відкрила безпрецедентні можливості щодо гнучкості та продуктивності застосувань. Але FPGA в хмарі також надають унікальні шляхи для зловживань з боку зловмисників на інших користувачів хмари або інфраструктуру, що не є можливим з CPU або GPU, базованих на обчисленнях в хмарі. Безпека залишається викликом, де архітектури повинні балансувати низькорівневий доступ, який дозволяє користувачам створювати власні потужні прискорювачі.

Основною перешкодою для широкого застосування FPGA в центрах обробки даних є відсутність стека програмного забезпечення, яке дозволяє легко розгортати, керувати та масштабувати FPGA в хмарі. Надання стека програмного забезпечення, подібного до того, що привело до успіху центральних/графічних процесорів, збільшить диверсифікацію архітектури в хмарі, коли FPGA працюватимуть як рівноправні партнери на боці центральних і графічних процесорів.

У роботі був зроблений огляд архітектур хмарних FPGA, який показує складні та нетривіальні взаємозв'язки між вимогами системи та конфігураціями розгортання, і визначає можливі напрямки майбутньої інновації в цьому просторі. Для цього, було використано поняття таксономії, яке абстрагує низькорівневі деталі реалізації, при цьому виділяючи переваги та обмеження даної архітектури. Використовуючи цю таксономію, ми класифікували як виробничі, так і дослідницькі системи, що демонструє основні тенденції в архітектурі хмарних FPGA.

Список літератури

1. A. Putnam, A. M. Caulfield, E. S. Chung, D. Chiou, K. Constantinides, J. Demme, H. Esmaeilzadeh, J. Fowers, G. P. Gopal, J. Gray, M. Haselman, S. Hauck, S. Heil, A. Hormati, J. Kim, S. Lanka, J. Larus, E. Peterson, S. Pope, A. Smith, J. Thong, P. Y. Xiao, and D. Burger. 2014. A reconfigurable fabric for accelerating large-scale data center services. In ACM/IEEE 41st International Symposium on Computer Architecture (ISCA). 13–24. DOI:https://doi.org/10.1109/ISCA. 2014.6853195.
2. A. M. Caulfield, E. S. Chung, A. Putnam, H. Angepat, Jeremy Fowers, Michael Haselman, Stephen Heil, Matt Humphrey, Puneet Kaur, Joo-Young Kim, Daniel Lo, Todd Massengill, Kalin Ovtcharov, Lisa Woods, Sitaram Lanka, Derek Chiou, and Doug Burger. 2016. A cloud-scale acceleration architecture. In 49th IEEE/ACM Int. Symp. on Microarchitecture. 1–13.
3. J. Fowers, G. Brown, P. Cooke, and G. Stitt, “A performance and energy comparison of FPGAs,GPUs, and multicores for sliding-window applications,” in Proceedings of the ACM/ SIGDA International Symposium on Field Programmable Gate Arrays, pp. 47–56, ACM, Monterey, CA, USA, February 2012.
4. Chethan Ramesh, Shivukumar B. Patil, Siva Nishok Dhanuskodi, George Provelengios, Sébastien Pillement, Daniel Holcomb, and Russell Tessier. 2018. FPGA side channel attacks without physical access. In IEEE 26th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM). 45–52.
5. Kasper Rasmussen, Ilias Giechaskiel, and Ken Eguro. 2019. Leakier wires: Exploiting FPGA long wires for covert-and side-channel attacks. ACM Trans. Reconfig. Technol. Syst. 12, 3 (2019).
6. Ilias Giechaskiel, Kasper Rasmussen, and Jakub Szefer. 2020. CAPSULe: Cross-FPGA covert-channel attacks through power supply unit leakage. In IEEE Symposium on Security and Privacy. 1728–1741.
7. George Provelengios, Daniel Holcomb, and Russell Tessier. 2019. Characterizing power distribution attacks in multi-user FPGA environments. In International Conference on Field Programmable Logic and Applications (FPL). 194–201.
8. Xilinx Case Study. [n.d.]. Xilinx Powers Alibaba Cloud FaaS with AI Acceleration Solution for E-Commerce Business. Retrieved from https://www.xilinx.com/publications/powered-by-xilinx/xilinx-alibaba-case-study.pdf.
9. Norihisa Fujita, Ryohei Kobayashi, Yoshiki Yamaguchi, and Taisuke Boku. 2019. Parallel processing on FPGA combining computation and communication in OpenCL programming. In IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW). IEEE, 479–488.
10. Mohamed S Abdelfattah, David Han, Andrew Bitar, Roberto DiCecco, Shane O’Connell, Nitika Shanker, Joseph Chu, Ian Prins, Joshua Fender, Andrew C Ling, et al. Dla: Compiler and fpga overlay for neural network inference acceleration. In 2018 28th International Conference on Field Programmable Logic and Applications (FPL), pages 411–4117. IEEE, 2018.

Received (Надійшла) 31.01.2023
Accepted for publication (Прийнята до друку) 19.04.2023

Future of FPGA - accelerating computations in data processing centers and clouds

Oleksandr Ilyashov, Kostiantyn Pokora, Vladislav Diachenko, Andriy Kovalenko

Abstract. **Relevance**. Data analysis is often performed using machine learning methods. Often, the algorithms involved need to deal with large datasets, which leads to long execution times. Therefore, research into hardware accelerators based on field-programmable gate arrays (FPGAs) to improve performance is relevant. FPGAs are a promising solution for hardware acceleration, post-production configuration, and reprogramming capabilities. **The purpose** of this study is to investigate and analyze trends in existing cloud FPGA architectures, which highlight the complex relationship between architectures and system requirements. This allows us to identify new architectures that are likely to offer significant benefits for cloud workloads. **The object** of the study is the evolution of FPGA accelerators for data center (DC) and cloud computing. **The subject** of the study is methods and algorithms for researching cloud FPGA architectures based on taxonomic categories. **Results**. The paper discusses the future use of FPGAs in data centers and clouds. Current architectures are also investigated and scalability and abstractions supported by operating systems, middleware, and virtualization are discussed. Conclusion. A classification of cloud FPGA architectures based on taxonomic categories has been developed. An architectural organization for deploying FPGA applications used in cloud environments and data center environments is considered and proposed.

Keywords: reconfigurable logic, FPGA acceleration, cloud computing, data center, virtualization, taxonomic categories, data classification, machine learning, software.