УДК 04.31

Я. М. Клятченко, О. С. Михайлюк, Л. М. Дудкова,
О. В. Тарасенко-Клятченко

ЗАХИСТ БЛОКІВ ІНТЕЛЕКТУАЛЬНОЇ ВЛАСНОСТІ У СПЕЦІАЛІЗОВАНИХ КОМП'ЮТЕРНИХ ЗАСОБАХ НА БАЗІ ПЛІС

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Київ

Анотація. Поточний рівень розвитку архітектур мікросхем програмовної логіки обумовлює не тільки доцільність, але і бажаність їхнього використання при розробці спеціалізованих комп'ютерних засобів або комбінаційної частини пристроїв обчислювальної техніки. Підвищення складності цифрових обчислювальних засобів, особливо в спеціалізованих системах критичного застосування, локалізує увагу розробників та компаній-виробників напівфабрикатів програмовних логічних інтегральних схем (ПЛІС) на виникненні ситуацій, які пов'язані із порушенням правильної роботи пристроїв, що обумовлені як зовнішніми впливами так і втручаннями. Якщо явища, що викликані негативними зовнішніми впливами, наприклад, як Single-Event Effect, можуть бути пов'язані із переходом на нові технологічні норми виготовлення напівпровідникової продукції, а саме, мікросхем ПЛІС, то різні втручання у функціонування пристроїв мають антропогенне коріння.

Широке використання ПЛІС для реалізації спеціалізованих комп'ютерних засобів спонукає до використання блоків інтелектуальної власності (intellectial property core, IP-core), оскільки для створення деяких екземплярів апаратних засобів необхідно реалізувати широкі функціональні можливості, що здійснюється завдяки IP. Такий підхід дозволяє втілити у спеціалізовані пристрої великий функціональний набір, подолати складності їхньої розробки та звузити часові рамки. В роботі наводиться частина огляду ефективних реалізацій захисту IP, який є складною та важливою задачею. Описано різні підходи та методи організації такого захисту. Наводяться посилання на приклади використання додаткових структур − доповнюючих шифрування та аутентифікації, які унеможливлюють несанкціонований доступ.

Ключові слова: Блоки інтелектуальної власності, IP-блоки, ПЛІС, bitstream encryption, AES, HMAC, ECC, CRC, SEU.

Аннотация. Текущий уровень развития архитектур микросхем программируемых логики обусловливает не только необходимость, но и желательность их использования при разработке специализированных компьютерных средств или комбинационной части устройств вычислительной техники. Повышение сложности цифровых вычислительных средств, особенно в специализированных системах критического назначения, локализует внимание разработчиков и компаний-производителей полуфабрикатов программируемых логических интегральных схем (ПЛИС) на возникновении ситуаций, связанных с нарушением правильнои роботы устройств, что обусловлены как внешними воздействиями, так и вмешательствами. Если явления, вызванные негативными внешними воздействиями, например, как Single-Event Effect, могут быть связаны с переходом на новые технологические нормы изготовления полупроводниковой продукции, а именно − микросхем ПЛИС, то вмешательство в функционирование устройств имеет антропогенные корни.

Широкое использование ПЛИС для реализации специализированных компьютерных средств побуждает к использованию блоков интеллектуальной собственности (intellectual property core, IP-core), поскольку для создания некоторых экземпляров аппаратных средств необходимо реализовать широкие функциональные возможности, то это осуществляется благодаря использованию IP. Такой подход позволяет воплотить в специализированные устройства большой функциональный набор, преодолеть сложности их разработки и сузить временные рамки. В работе приводится часть обзора эффективных реализаций защиты IP который является сложной и важной задачей. Определены различные подходы и методы организации такой защиты. Приводятся ссылки на примеры использования дополнительных структур − дополняющих шифрования и аутентификации, которые делают невозможным несанкционированный доступ.

Ключевые слова: Блоки интеллектуальной собственности, IP-блоки, ПЛИС, bitstream encryption, AES, HMAC, ECC, CRC, SEU.

Abstract. The current evolutionary stage of the microchips' architecture of programmable logic does precondition not only a rationale for but also desirability of its utilisation when developing specialised computer means or the combinatory part of devices of computing machines. The increase in complexity of digital computational devices, especially in the critical usage' computer systems, dramas and focuses the attention of developers and OEMs of FPGA to the occurrence of events related to the devices' correct operation' interruption, that may be caused both by external factors and intrusion. If events that are caused by negative external impacts such as a Single-Event Effect, may be related to the transition to new technological norms of the semiconductor products manufacturing, e.g. FPGA microchips, then any intrusions to the devices' operation have anthropological origins.

Widespread use of FPGA to implement the specialised computer means prompts the use of the intellectual property blocks (intellectual property core, IP-core) since to create certain samples of hardware the broad functional capabilities have to be implemented, which is effectuated by the IP. Such an approach enables materialisation of the substantial feature set in a specialised device, overcome the complexities in the devices' development and narrow down time-frames. A part of an overview of the efficient IP cores protection, being an important and complex task, is exemplified in the paper. Different approaches and methods are outlined for such protection organisation. Examples are given of the examples of the additional structures' usage, complementary to ciphering and authentication, that prohibit unauthorised access.

Key words: Intellectial property core, IP core, programmable logical devices, bitstream encryption, FPGA design protection, AES, HMAC, ECC, CRC, SEU.

DOI: https://doi.org/10.31649/1999-9941-2021-50-1-15-21.

Вступ

Наявність загроз та слабких сторін у апаратних засобів, що реалізовані на ПЛІС, висвітлює важливість забезпечення інформаційної стійкості в розрізі протидії негативним явищам, таким як несанкціонований доступ до інтелектуальної власності або її крадіжка. Розробка сучасних комп'ютерних засобів, а особливо спеціалізованих, супроводжується використанням так званих IP-блоків (блоків інтелектуальної власності, IP-core). Завдяки використанню IP значно скорочується час реалізації пристроїв та значно ро-

зширюється іхній функціонал. IP − це складний, протестований, верифікований та оптимізований функціональний модуль, що забезпечує необхідні алгоритми обробки інформації та, який може бути використано декілька разів. IP, як правило, може бути центральним компонентом в пристрої, що проєктується. Практика доводить той факт, що використання IP наближує момент появи цифрового пристрою на ринку. Процес проєктування мінімізується завдяки включенню вже створених та протестованих модулів, скорочується час на верифікацію проєкту. Статистичні дані говорять про те, що ринок IP-cores постійно розвивається та зростає [1].

Актуальність

Як не прикро, але існує зворотній бік використання IP: те, на що було витрачено інтелектуальні, матеріальні та часові ресурси, можна втратити за мить. Несанкціонований доступ до блоків інтелектуальної власності можливий завдяки копіюванню (клонуванню) пристрою та по причині здійснення дій із зворотньої розробки або інжинерингу (reverse engineering) [2]. Якщо у випадку клонування новий пристрій реалізується якомога більш схожим на оригінальний пристрій, то у другому випадку виконуються дії, що можуть призвести до отримання інформації про особливості інтелектуального продукту у вигляді алгоритмів роботи пристрою та дозволить створити нову специфікацію пристрою. До інших загроз для блоків інтелектуальної власності можна віднести такі ситуації, як несанкціонований надмірний випуск продукції OEM-виробниками; халатне ставлення інженерів до питань безпеки, наприклад, невиконання або повне ігнорування користувачами IP-cores інструкцій із заходів безпеки; наявність дірок у захисті (backdoor) які залишають розробники пристрою на етапі відлагодження або тестування; помилки у сфері безпеки проєктів.

До загроз неантропогенного характеру можна віднести одиничні збої SEU (англ. Single-Event Upset), що виникають внаслідок дії на чутливі структури пам'яті ПЛІС високоенергетичних часток із високою енергією (англ. Single-Event Effect, SEE) в результаті чого змінюється значення одного або декількох бітів пам'яті [3].

Мета

Метою даної роботи є визначення ефективних рішень у сфері захисту блоків інтелектуальної власності в спеціалізованих комп'ютерних засобах на ПЛІС за рахунок виявлення нових напрямків, огляду перспективних ідей, систематизації підходів, методів та засобів такого захисту.

Задачі

1. Провести огляд, систематизувати та визначити множину загроз, атак та впливів на IP-блоки у пристроях на ПЛІС. Узагальнити існуючі відомості про ефективність існуючих засобів протидії.
2. Оцінити множину існуючих рішень для захисту IP та, на основі цього визначити ефективність використовуваних підходів, методів та засобів.
3. Визначити найдієвіші підходи до захисту від одиночних збоїв та окреслити практичні рекомендації щодо їхньої організації.

Розв'язання задач

Негативні впливи у вигляді загроз та атак, що супутні процесу проєктування та виготовлення обчислювальних засобів на ПЛІС, в першу чергу приводять до проблем із використанням блоків інтелектуальної власності, захист яких від несанкціонованного ознайомлення, використання, підробки, модифікації тощо є актуальною задачею. Тому враховуючи наведені приклади, постає задача оцінки ективності множини рішень для захисту апаратних засобів на ПЛІС в умовах дії на них негативних факторів.

Ефективність реалізацій захисту IP

Компанії-виробники мікросхем програмовної логіки мають свої фірмові рішення для боротьби із загрозами та негативними впливами. Але, дивлячись на це, представляють загрозу саме атаки на проєкти (ПЛІС-реалізації) спеціалізованих комп'ютерних засобів. Такими атаками можна вважати дії по декодуванню bitstream (конфігураційної послідовності). Результатом цього є відновлення так званого netlist, що є ні що інше, як процес зворотнього інжинерингу. Оскільки процедура генерування bitstream вважається закритою, то у арсеналі виробників ПЛІС є свої фірмові рішення для унеможливлення зворотнього перетворення bitstream у netlist. Завдяки цьому стає дуже важко чітко співставити конфігураційну послідовність або її частину деякому списку з'днань netlist, оскільки для реалізації цього можуть знадобитись значні ресурси. Таким чином генерування конфігураційної послідовності відносять до одного із механізмів захисту блоків інтелектуальної власності, оскільки у вигляді bitstream проєкт пристрою є неочевидним, заплутаним та незрозумілим для аналізу.

До особих випадків зламу проєкту можна віднести так звану підміну, коли зловмисник змінює (змішує) або фрагмент, або всю частину bitstream FPGA-проєкту із власним кодом конфігураційної послідовності та видає за власний проект.

Апаратний троянець використовується для доступу до інформації, що зберігається в ПЛІС, і навіть для крадіжки IP. Зловмисники вбудовують цей інструмент у програмне забезпечення САПР або логіку проекту. В результаті проект стає вразливим до хакерських атак вже під час процесу розробки.

Так зване зворотнє зчитування потоку бітів з FPGA може бути віднесене до загроз IP, оскільки за допомогою цих даних можна отримати інформацію про стан пам'яті користувача та стан внутрішніх ре-

гістрів системи. Зворотнє зчитування потенційно може бути використано для відновлення конфігураційних даних проєкту.

При атаках за так званим стороннім каналом (Side-channel attack) [4] хакер бере за основу робочі характеристики системи та особливості протікання фізичних процесів у пристрої (атаки з використанням часових характеристик, параметрів енергоспоживання, електромагнітного випромінювання та помилки обчислень). За допомогою таких атак зловмисники можуть отримати ключі шифрування та дані щодо проєкту.

Атака шляхом внесення несправності − це метод, що використовується при тестуванні обладнання. Примушуючи пристрій перейти в режим тестування, відлагодження, аварійного стану або стану виведення службової інформації, забезпечується штучне введення різних несправностей для відмовостійкого тестування, тим паче в особливих ситуаціях обробки. Зловмисник змінює конструкцію, умови навколишнього середовища, напругу або температуру, щоб викликати надзвичайну ситуацію в роботі пристрою. Такі операції в пристроях на ПЛІС можуть змінювати біти в конфігураційній послідовності, що впливає на функціональність. Однак методи, що використовуються в сучасному апаратному проєктуванні (такі як визначення всіх станів та повний аналіз відмов) ускладнюють реалізацію таких атак на ПЛІС.

Ще одним класичним методом забезпечення безпеки є використання шифрування проєкту, де постачальник IP передає замовнику ключ дешифрування разом із IP, але знову ж таки, для досягнення згоди на використання IP та ключа необхідно укласти угоду. З точки зору захисту IP, ефективно розподіляти конфіденційні дані проєктів між різними компаніями, які беруть участь у виробництві комп'ютерного обладнання на ПЛІС, щоб не дати доступу до всієї інформації про проєкт. По суті, ПЛІС є надійною основою для реалізації обчислювальних засобів, оскільки при виробництві кінцевого продукту виробник обладнання (OEM) відокремлюється від проєктувальника.

Провідні постачальники напівфабрикатів ПЛІС [5] пропонують широкий спектр рішень щодо захисту проєктів (IP): від впровадження ідентифікатора пристрою (DNA) та шифрування конфігураційної послідовності до використання механізму перевірки цілісності інформації (HMAC) [6] для аутентифікації bitstream. У процесі реалізації комп'ютерних засобів на ПЛІС шифрування бітового потоку може не тільки запобігти клонуванню пристрою, але й захистити конфіденційні конфігураційні дані проєкту. Кожен програмовний логічний пристрій містить спеціальний блок дешифрування бітового потоку для підтримки стандартного AES-шифрування [7].

На основі прикладу, описаного в [8], реалізація шифрування bitstream проєкту полягає в наступному. Система шифрування потоку бітів складається з двох частин: програмного забезпечення для шифрування потоку бітів та системи дешифрування потоку бітів на основі мікросхеми пам'яті, що використовується для зберігання 256-бітового ключа шифрування. Розробники проєктів використовують програмне забезпечення для створення ключів шифрування та зашифрованих бітових потоків. В подальшому ключ шифрування зберігається в спеціальній енергонезалежній RAM або із застосуванням технології eFUSE [9]. Насправді eFUSE є компонентами вбудованої системи самодіагностики і відновлення, яка безперервно здійснює моніторинг функціональних можливостей пристрою. При виявленні ознак нештатної роботи eFUSE здійснює коригування характеристик мікросхеми, «випалюючи» мікроскопічні плавкі запобіжники, вбудовані в її структуру.

Ключ шифрування надсилається у пристрій лише через порт JTAG. Під час процесу конфігурації ПЛІС виконує зворотну операцію і використовує блок дешифрування ПЛІС для дешифрування вхідного бітового потоку відповідно до алгоритму AES. Блок дешифрування AES на мікросхемі ПЛІС недоступний розробникам проєктів і не може використовуватися для дешифрування будь-яких даних, крім послідовності конфігурації. В якості додаткового рівня безпеки забороняється завантаження зашифрованого бітового потоку в ПЛІС. Якщо пам'ять конфігурації повністю не очищена, зашифрований бітовий потік проєкту не можна завантажити в мікросхему ПЛІС. Подібно діє правило, за яким за умови того, що якщо ПЛІС завантажена зашифрованим bitstream, то не можна туди завантажити незашифрований проєкт. Це допомагає протидіяти маніпуляціям reverse engineering.

З метою протидії атакам типу підміни (спуфінга) та троянців, довело свою ефективність використання криптографічно потужних засобів автентифікації [10]. Для передачі зашифрованого алгоритмом AES бітового потоку проєкту використовується вбудований блок ПЛІС для хешування потоку методом автентифікації повідомлень HMAC [11]. Використовуючи механізм цілісності інформації HMAC, засоби проєктування використовуть ключ та саме повідомлення для створення коду автентифікації цього повідомлення (англ. Message Authentication Code, MAC). Приймаюча сторона (в даному контексті програмовне середовище, апаратна сторона) використовує той же самий ключ для обчислення цього хеш-коду (MAC) для отриманого повідомлення та порівняння результатів. Обидва ці компоненти генерують 256-бітний MAC на основі секретного ключа та захищеного хеш-алгоритму SHA256. Якщо ці два значення збігаються, то повідомлення вважається підтвердженим.

Отже, неможливо завантажити, змінити або клонувати бітовий поток IP, не знаючи ключів для AES та HMAC. Якщо алгоритм AES захищає вміст IP від копіювання або зворотного інжинірингу, то використання хеш-механізму HMAC може гарантувати, що бітовий потік конфігурації, що завантажений в ПЛІС, не змінився. Так виявляється будь-яка зміна потоку конфігурації, включаючи зміну значення принаймні одного біта.

Яскравим та ефективним прикладом забезпечення необхідного рівня захисту від апаратних троянських програм є так званий багатоваріантний метод реалізації [12], який пропонує задіювати декілька блоків IP від різних постачальників та використовувати блок виявлення троянських програм для ідентифікації підозрілих IP ядер. На рис.1 показано схему, що ілюструє механізм роботи такого захисту. Тут

порівнюються виходи множини IP ядер, і якщо існує розбіжність між значеннями на виходах, то, ймовірно, присутній троянець. Якщо IP блоків більше ніж три, можна скористатися спеціальною схемою, яка вибирає правильний результат і, при необхідності, активує в систему повідомлення про апаратного троянця.

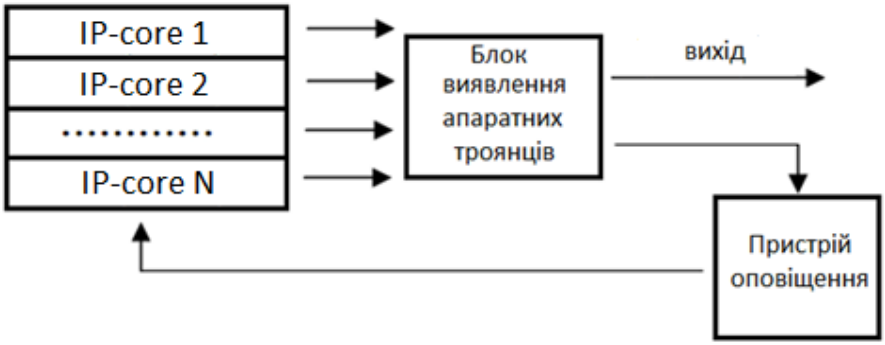

Рисунок 1 – Механізм захисту IP від апаратних троянців

Далі блок оповіщення щодо присутності троянських програм може частково реконфігорувати ПЛІС для видалення зараженого блоку та заміни його новим, поки решта логіки ПЛІС буде функціонувати. Такий блок виявлення троянців може використовуватись для виявлення IP від ненадійних або невідомих постачальників та їх позначення як підозріле IP.

Для оцінки ефективності застосування схеми виявлення апаратних троянців аналогічно із [12] проведено експеримент, що полягає у реалізації на ПЛІС п’яти IP, які представляють собою різні АЛП. Також, реалізовано збірний IP, що представляє собою всі АЛП разом із інтегрованою схемою виявлення троянців. Із таблиці 1 виходить, що час затримки збірного IP сумірний із максимальною затримкою одного із п’яти реалізованих ядер.

Таблиця 1 – Багатоваріантна реалізація

	LUT	**Затримка (нс)**
IP 1	54	11.0
IP 2	48	4.8
IP 3	172	4.1
IP 4	78	6.7
IP 5	68	5.1
Збірний IP	187	12.7

Підходи до організації захисту від одиночних збоїв

Сучасні дослідження [13] вказують на існування проблем в розрізі надійності у сучасних напівпровідникових кристалів, виготовлених за сучасними зменшеними техпроцесами. Серед причин, на які було вказано, є такі явища, як SEE − одноразовий вплив на роботу високоенергетичних частинок (важких іонів або протонів) чіпів [14]. Наслідки SEE можуть мати різний характер: руйнівні випадки (SEB, SEL − так звані hard errors) або неруйнівні (SEU, SET, SEFI − так звані soft errors). Оцінка достовірності функціонування цифрових обчислювальних засобів у мовах дії таких деструктивних факторів досліджувалась у [15].

Одиночні збої SEU (англ. Single-Event Upset) можуть впливати на комірки пам’яті та призводять дозміни значення одного або декількох бітів і тут важливо, в якому типі пам’яті сталися ці спотворення і де ця сама пам’ять розташовується. Спотворені біти можуть належати конфігураційній пам'яті ПЛІС, яка містить проєкт. Ця пам'ять найбільша за об’ємом та фізично розподілена по площині мікросхеми, але лише частина бітів має важливе значення для правильної роботи в пристрою, що реалізується на ПЛІС. Інші елементи пам’яті високої ємності, які використовуються для зберігання стану проекту називають блочною пам’яттю. Ця пам’ять є другою за ємністю. Елементи блочної пам’яті об’єднані в групи та розташовані по всій ПЛІС. До розподіленої пам'яті належать запам’ятовуючі елементи для зберігання стану проєкту на ПЛІС. Цей тип пам'яті реалізовано на базі матриці конфігуровних логічних блоків (англ. Configurable Logic Block, CLB) та розподілено по всьому програмовному кристалу [16].

В ПЛІС використовуються спеціальні структури для боротьби із із такими негативними явищами як одиночні збої (SEU), а також атаками за стороннім каналом та зламом. Такі засоби реалізують постійне зчитування у фоновому режимі конфігураційної послідовності проекту [17]. Також, застосовуються засоби виявлення та корекції помилок, що відомі як ECC (error correcting code) [18]. Цей відомий прийом збільшує ймовірність знаходження змін конфігурації у пам’яті внаслідок одиночного збою (SEU) або через атаку на проєкт.

Виділимо особливі риси таких засобів. У першу чергу визначимо характер функціонування цих засобів − реалізація пом'якшення впливів одиночних збоїв у розглянутих типах пам'яті ПЛІС, а не запобігання появі таких помилок. Це може бути створено шляхом внесення в проєкт пристрою додаткових апаратних структур для виявлення та корекції помилок або із застосуванням надмірності. Помилки в логічних ресурсах ПЛІС що не зайняті проєктом і які виникли в результаті дії SEU ігноруються. Крім цього відбувається класифікація помилок на «суттєві» та «неважливі». Це дає змогу враховувати та корегувати не всі помилки, а тільки значущі.

Характеристики реалізації відомих рішень [19] включають:

- малий час виявлення несправностей − близько 20-30 мілісекунд;
- застосування апаратних примітивів, що вбудовані в програмовну мікросхему для виявлення та виправлення помилок;
- застосування засобів відновлення , що основані на ECC або разом з алгоритмом CRC;
- виправлення шляхом повної заміни даних шляхом їхнього перезавантаження;
- визначення ступеня впливу помилки на функціонування проєкту;
- продовження часу безперервної роботи пристрою на ПЛІС за рахунок обробки «неважливих» помилок.

Додаткові засоби захисту IP

Останні моделі ПЛІС можуть видаляти ключ шифрування у відповідному блоці пам'яті, конфігураційній та тригерній пам'яті за спеціальним сигналом. Цей механізм може бути активований у відповідь на зловмисні дії.

Щоб запобігти клонуванню пристрою, останнє покоління ПЛІС також має вбудований унікальний ідентифікатор пристрою. Унікальний ідентифікатор (схожий на серійний номер) зберігається в енергонезалежній пам'яті. При розробці пристрою користувачі можуть використовувати цей унікальний ідентифікатор для реалізації нового механізму захисту IP від крадіжки.

Висновки

1. Не зважаючи на високий рівень оснащення сучасних зловмисників, які втручаються в intellectual property core, різноманіття їхнього арсеналу для зламу, а також, наявності множини загроз правильному функціонуванню обчислювальних засобів на ПЛІС, існує багата множина засобів захисту цих IP. Такі засоби вирішують завдання із захисту та дозволяють клієнтам створювати проєкти комп'ютерних засобів на ПЛІС, які є не тільки захищеними, але й стійкими до підробки та клонування. В роботі проведено огляд загроз та виділено найдієвіші напрямки для забезпечення захисту IP у проєктів на ПЛІС.

2. В доповнення до відомих методів шифрування та криптографічно стійкої аутентифікації, які дозволяють значно підвищити рівень захищенності проектів на ПЛІС запропоновано звернути увагу на використанням технології eFUSE, що дозволяє унеможливити злам.

3. Надано практичні рекомендації щодо захисту проєктів на ПЛІС від впливів, що спричинені, наприклад, одиночними збоями. Наслідків від одиночних збоїв можна уникнити якщо використовувати наявні апаратні засоби, що вбудовані в ПЛІС для виявлення та корекції помилок. На прикладі ефективної реалізації схеми захисту IP-блоків в проектах обчислювальних засобів на ПЛІС показано суттєве покращення рівня захисту, що може бути застосовно до сучасних ПЛІС з незначними внесеннями змін до набору функцій та архітектури пристрою.

4. Проведено експеримент із засобами виявлення апаратних троянців, що представляють собою багатоваріантну реалізацію. Дані цієї практичної реалізації показують, що впровадження разом із IP ядрами, які виконують однакову функцію, засобів виявлення апаратних троянців вносить несуттєвий вплив на швидкодію пристрою.

Список літератури

[1] Dylan. McGrath, «Report: Semiconductor IP market to double in five years», *EETIMES*, 2012, 2014. [Online]. Available: https://www.eetimes.com/report-semiconductor-ip-market-to-double-in-five-years.

[2] J.-B. Note, E. Rannaud, «From the bitstream to netlist», Proc. 16th Int.ACM/SIGDA Symp. On FPGA, N.Y.: ACM, 2008.

[3] Device Reliability Report. Second half 2020, *Xilinx Inc., User Guids*, 2020. [Online]. Available: https://www.xilinx.com/support/documentation/user_guides/ug116.pdf.

[4] M. McLean and J. Moore, «FPGA-Based Single Chip Cryptographic Solution», *Military Embedded Systems*, 2007.

[5] S. McNeil, «Solving Today's Design Security Concerns», *WP365*, (v1.2) July 30, 2012. [Online]. Available: https://www.xilinx.com/support/documentation/white_papers/wp365_Solving_Security_Concerns.

[6] HMAC: Keyed-Hashing for Message Authentication. [Online]. Available: https://www.ietf.org/rfc/rfc2104.txt .

[7] Advanced Encryption Standard (AES). (FIPS PUB 197). [Online]. Available: https://csrc.nist.gov/csrc/media/publications/fips/197/final/documents/fips-197.pdf.

[8] K. Wilkinson, «Using Encryption to Secure a 7 Series FPGA Bitstream», *XAPP1239* (v1.0) April 15, 2015. [Online]. Available: https://www.xilinx.com/support/documentation/application_notes/xapp1239-fpga-bitstream-encryption.pdf.

[9] Randal Kuramoto, *eFUSE Programming on a Device Programmer*, 2015. [Online]. Available: https://vdocuments.mx/xapp1260-efuse-programmer.html.
[10] Amir Moradi, Tobias Schneider, «Improved SideChannel Analysis Attacks», *Xilinx Bitstream Encryption of 5 6 and 7 Series, Constructive Side-Channel Analysis and Secure Design: 7th International Workshop, COSADE 2016*, Graz, Austria, April 14-15, 2016.
[11] FIPS-198-1, Keyed-Hash Message Authentication Code, Federal Information Processing Standards, U.S. National Institute of Standards and Technology. [Online]. Available: http://www.nist.gov/itl/upload/FIPS-198-1_final.pdf.
[12] A. Al-Anwar, Y. Alkabani, M. W. El-Kharashi, and H. Bedour, «Hardware Trojan detection methodology for FPGA», in *Proceedings of the 2013 IEEE Pacific Rim Conference on Communications, Computers, and Signal Processing (PacRim)*, Victoria, BC, Canada, pp. 177−182.
[13] Jameel Hussein and Gary Swif, «Mitigating Single-Event Upsets», *Xilinx Inc. WP395* (v1.1) May 19, 2015. [Online]. Available: http://www.xilinx.com/support/documentation/white_papers/wp395-Mitigating-SEUs.pdf.
[14] R. Rajaei, B. Asgari, M. Tabandeh, M. Fazeli, «Single Event Multiple Upset-Tolerant SRAM Cell Designs or Nano-scale CMOS Technology», *Turkish Journal of Electrical Engineering & Computer Sciences*, 2016.
[15] Я. М. Клятченко, «Визначення достовірності функціонування апаратних засобів на ПЛІС в умовах спотворення логічних сигналів», *ІТКІ*, вип. 34, т. 3, с. 9–12, Лют. 2016.
[16] О. Д. Азаров, В. А. Гарнага, Я. М. Клятченко, В. П. Тарасенко, *Комп'ютерна схемотехніка: підручник*. Вінниця, Україна: ВНТУ, 2018. 230 с.
[17] Soft Error Mitigation Controller v4.1 LogiCORE IP. Product Guide, 2017. [Online]. Available: https://www.xilinx.com/support/documentation/ip_documentation/sem/v4_1/pg036_sem.pdf.
[18] E. Gabidulin, N. Pilipchuk, «Error and erasure correcting algorithms for rank codes», *Des. Codes Cryptogr*, 2008.
[19] LogiCORE IP Soft Error Mitigation Controller v3.4.1. Product Guide. September 30, 2015. [Online]. Available: https://www.xilinx.com/support/documentation/ip_documentation/sem/v3_4/pg036_sem.pdf.

Стаття надійшла: 01.02.2021.

References

[1] Dylan. McGrath, «Report: Semiconductor IP market to double in five years», *EETIMES*, 2012, 2014. [Online]. Available: https://www.eetimes.com/report-semiconductor-ip-market-to-double-in-five-years.
[2] J.-B. Note, E. Rannaud, «From the bitstream to netlist», Proc. 16th Int.ACM/SIGDA Symp. On FPGA, N.Y.: ACM, 2008.
[3] Device Reliability Report. Second half 2020, *Xilinx Inc., User Guids*, 2020. [Online]. Available: https://www.xilinx.com/support/documentation/user_guides/ug116.pdf.
[4] M. McLean and J. Moore, «FPGA-Based Single Chip Cryptographic Solution», *Military Embedded Systems*, 2007.
[5] S. McNeil, «Solving Today's Design Security Concerns», *WP365*, (v1.2) July 30, 2012. [Online]. Available: https://www.xilinx.com/support/documentation/white_papers/wp365_Solving_Security_Concerns.
[6] HMAC: Keyed-Hashing for Message Authentication. [Online]. Available: https://www.ietf.org/rfc/rfc2104.txt .
[7] Advanced Encryption Standard (AES). (FIPS PUB 197). [Online]. Available: https://csrc.nist.gov/csrc/media/publications/fips/197/final/documents/fips-197.pdf.
[8] K. Wilkinson, «Using Encryption to Secure a 7 Series FPGA Bitstream», *XAPP1239* (v1.0) April 15, 2015. [Online]. Available: https://www.xilinx.com/support/documentation/application_notes/xapp1239-fpga-bitstream-encryption.pdf.
[9] Randal Kuramoto, *eFUSE Programming on a Device Programmer*, 2015. [Online]. Available: https://vdocuments.mx/xapp1260-efuse-programmer.html.
[10] Amir Moradi, Tobias Schneider, «Improved SideChannel Analysis Attacks», *Xilinx Bitstream Encryption of 5 6 and 7 Series, Constructive Side-Channel Analysis and Secure Design: 7th International Workshop, COSADE 2016*, Graz, Austria, April 14-15, 2016.
[11] FIPS-198-1, Keyed-Hash Message Authentication Code, Federal Information Processing Standards, U.S. National Institute of Standards and Technology. [Online]. Available: http://www.nist.gov/itl/upload/FIPS-198-1_final.pdf.
[12] A. Al-Anwar, Y. Alkabani, M. W. El-Kharashi, and H. Bedour, «Hardware Trojan detection methodology for FPGA», in *Proceedings of the 2013 IEEE Pacific Rim Conference on Communications, Computers, and Signal Processing (PacRim)*, Victoria, BC, Canada, pp. 177−182.

[13] Jameel Hussein and Gary Swif, «Mitigating Single-Event Upsets», *Xilinx Inc. WP395* (v1.1) May 19, 2015. [Online]. Available: http://www.xilinx.com/support/documentation/white_papers/wp395-Mitigating-SEUs.pdf.
[14] R. Rajaei, B. Asgari, M. Tabandeh, M. Fazeli, «Single Event Multiple Upset-Tolerant SRAM Cell Designs or Nano-scale CMOS Technology», *Turkish Journal of Electrical Engineering & Computer Sciences*, 2016.
[15] Y. M. Klyatchenko, «Vyznachinennya dostivirnosti funktsionuvannya aparnykh zazobiv na PLIS v umovakh spotvorennya lohichnykh syhnaliv», *ITKI*, vyp. 34, t. 3, s. 9–12, Lyut. 2016.
[16] O. D. Azarov, V. A. Harnaha, Y. M. Klyatchenko, V. P. Tarasenko, *Komp'yuterna skhemotekhnika: pidruchnyk.* Vinnytsya, Ukraina: VNTU, 2018. 230 s.
[17] Soft Error Mitigation Controller v4.1 LogiCORE IP. Product Guide, 2017. [Online]. Available: https://www.xilinx.com/support/documentation/ip_documentation/sem/v4_1/pg036_sem.pdf.
[18] E. Gabidulin, N. Pilipchuk, «Error and erasure correcting algorithms for rank codes», *Des. Codes Cryptogr*, 2008.
[19] LogiCORE IP Soft Error Mitigation Controller v3.4.1. Product Guide. September 30, 2015. [Online]. Available: https://www.xilinx.com/support/documentation/ip_documentation/sem/v3_4/pg036_sem.pdf.

Відомості про авторів

Клятченко Ярослав Михайлович – кандидат технічних наук, доцент кафедри системного програмування і спеціалізованих комп'ютерних систем.

Михайлюк Олена Станіславівна – асистент кафедри системного програмування і спеціалізованих комп'ютерних систем.

Дудкова Лариса Миколаївна – асистент кафедри системного програмування і спеціалізованих комп'ютерних систем.

Тарасенко-Клятченко Оксана Володимирівна – кандидат технічних наук, доцент кафедри системного програмування і спеціалізованих комп'ютерних систем.

Я. М. Клятченко, Е. С. Михайлюк, Л. Н. Дудкова,
О. В.Тарасенко-Клятченко

ЗАЩИТА БЛОКОВ ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ В СПЕЦИАЛИЗИРОВАННЫХ КОМПЬЮТЕРНЫХ СРЕДСТВАХ НА БАЗЕ ПЛИС

Национальный технический университет Украины «Киевский политехнический институт имени Игоря Сикорского», Киев

Y. M. Klyatchenko, O. S. Mykhailyuk, L. M. Dudkova,
O. V. Tarasenko-Klyatchenko

INTELLECTUAL PROPERTY CORES PROTECTION IN FPGA-BASED SPECIALIZED DEVICES

National Technical University of Ukraine «Igor Sikorsky Kyiv Polytechnic Institute», Kyiv