

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ОДЕСЬКА ПОЛІТЕХНІКА
МІНІСТЕРСТВА ОСВІТИ І НАУКИ УКРАЇНИ
Кафедра комп'ютерних систем та мереж

НАУМОВ Олексій Денисович

ДИПЛОМНА РОБОТА МАГІСТРА
ДОСЛІДЖЕННЯ ЕНЕРГО-ОРІЄНТОВАНОЇ КОНТРОЛЕПРИДАТНОСТІ
БАГАТООПЕРАНДНОГО ДОДАВАЧА

Спеціальність 123 – Комп'ютерна інженерія
Спеціалізація - Комп'ютерні системи та мережі

Керівник: Защелкін Костянтин Вячеславович,
к.т.н., доцент

Одеса – 20

ЗМІСТ

Вступ	3
1 Аналітичний огляд методів та засобів контролю цифрових FPGA-компонентів у складі комп'ютерних системи критичного застосування	7
1.1 Огляд платформи FPGA та її місця серед програмованих компонентів комп'ютерних систем.....	7
1.2 Аналіз архітектурних особливостей мікросхем FPGA.....	9
1.3 Галузі застосування мікросхем FPGA.....	12
1.4 Особливості програмування мікросхем FPGA та формальних мов, задіяних в процесі програмування	15
1.5 Аналіз стандартів функціональної безпеки комп'ютерних системи критичного застосування.....	21
1.6 Визначення та підвищення показника контролепридатності цифрових схем комп'ютерних систем.....	25
1.7 висновки	27
2 Пропонований підхід до долідження енерго-орієнтованої контролепридатності багатооперандного додавача	28
2.1 Аналіз проблеми контролю цифрових FPGA-компонентів у складі комп'ютерних системи критичного застосування	28
2.2 Постановка мети та задач експериментального дослідження	34
2.3 Опис концепції дослідження енерго-орієнтованої контролепридатності багатооперандного додавача	34
2.4 Висновки	36
3 Експериментальне дослідження енерго-орієнтованої контролепридатності багатооперандного додавача	37
3.1 Середовище проведення експериментального дослідження.....	37
3.2 Опис схеми для проведення експериментального дослідження	39
3.3 Опис програмного забезпечення для проведення експерименту	48
3.4 Методика проведення експерименту.....	49
3.5 Процес отримання результатів експериментального дослідження.....	51
3.6 Аналіз отриманих результатів дослідження.....	68
3.6 Висновки	71
Загальні висновки.....	72
Перелік використаних джерел	73

ВСТУП

Людство незупинно розвивається, створюються нові технології та винаходи, населення на планеті також зростає. Для забезпечення бурхливого розвитку технологій та потреб людства необхідні ресурси. На даний момент забезпечення будь якими ресурсами нерозривно зв'язано з електричною енергією, бо виробництво більшої кількості необхідних ресурсів автоматизовано та його нарощування нерозривно зв'язано з використанням більшої кількості приладів, а це потребує більшої кількості електроенергії.

Для отримання електроенергії використовують електростанції, одними з найефективніших типів електростанцій вважаються атомні та гідроелектростанції. Вони дають можливість отримання енергії з мінімальною кількістю ресурсів, що витрачаються, та максимальною кількістю енергії, що отримується, також вважається, що такі типи електростанцій не забруднюють зовнішнє середовище. Це вірно лише, якщо не брати до уваги можливість аварії на таких електростанціях. Аварія на атомній електростанції спроможна зробити непридатними до життя людини тисячі квадратних кілометрів, разом з інфраструктурою, що залишиться в цій зоні та невинними наслідками для життя та здоров'я людей.

Для розрахунку ризику подій, у тому числі і техногенних катастроф, використовується два основних параметри, це імовірність виникнення події, та тяжкість наслідків від її траплення, в результаті бурхливого розвитку ми все більше и більше збільшуємо обсяги виробництва енергії, тобто збільшуємо потужність існуючих джерел та будуємо нові, що збільшує другий фактор розрахунку ризику події. Для утримання показника ризику придатного для забезпечення необхідної безпеки маємо тільки один шлях, а саме зменшення ймовірності виникнення події.

Для зменшення ймовірності події необхідно вдосконалення інформаційних технологій темпами, не меншими за швидкість зростання необхідності ресурсів, а ще краще випереджуючи її. Інформаційні технології імплементовані в комп'ютерні системи для керування небезпечними об'єктами підвищеного ризику та стримуванню аварій, що можуть траплятися на них.

Використання комп'ютерних систем для керування об'єктами підвищеного ризику перетворює їх на системи критичного застосування, що мають відмінність, яка проявляється при їх проектуванні, а саме розділення режимів їх роботи на два основних: нормальний режим та аварійний режим.

Беручи до уваги вищесказане, можна зробити висновок про те, що підвищення контролепридатності комп'ютерних систем критичного застосування складає актуальну науково-технічну проблему.

Мета роботи полягає в підвищенні контролепридатності комп'ютерних систем критичного застосування шляхом введення додаткового фактору контролю, а саме току споживання пристрою.

Відповідно до поставленої мети в роботі були вирішені наступні задачі:

- проведено аналіз методів та засобів контролю цифрових FPGA-компонентів;
- запропонований спосіб дослідження енергоорієнтованої контролепридатності;
- проведено експериментальне дослідження енергоорієнтованої контролепридатності цифрового компоненту;

Об'єктом дослідження виступає процес контролю комп'ютерних систем критичного застосування.

Предметом дослідження є методи та засоби контролю схем комп'ютерних систем критичного застосування за енергетичним проявом.

Методи дослідження базуються на основах цифрової схемотехніки, дискретній математиці, архітектурі комп'ютерних систем та теорії діагностики комп'ютерних систем.

Наукова новизна полягає в тому, що експериментально було доведено можливість відстеження режиму роботи пристрою за його енергоспоживанням для оцінки коректності роботи системи критичного застосування.

Практичне значення отриманих результатів полягає в тому, що в досліджених вдалося виявити несправності в складі комп'ютерних систем критичного застосування до необхідності використання ділянки схеми, що при використанні логічної контролепридатності були б виявленні під час необхідності використання конкретної ділянки схеми.

В першому розділі аналіз предметної області для виконання дипломної роботи. Були досліджені такі теми, як FPGA. Визначено що це за клас пристроїв, яку вони мають архітектуру, де застосовуються, які програмні засоби використовуються для їх конфігурування.

Було визначено поняття безпеки у контексті комп'ютерних систем та стандарти, що регулюють вимоги до функціональної безпеки пристрою.

У другому розділі було визначено проблематику роботи, проаналізовані проблеми контролю цифрових схем на базі FPGA.

Було виявлено, що основною проблемою логічної контролепридатності є її орієнтованість на поточні данні та ділянки схеми, що використовуються у даний конкретний момент.

Для визначення стану ділянок схеми, що не задіюються на поточний момент, але мають деякі логічні елементи, що виконують роботу не зважаючи на наявність вхідних даних було запропоновано відстеження току

споживання, що дає можливість визначити відключення від синхронізації елементів, що не задіяні на даний момент.

У третьому розділі на основі положень першого та другого розділу були розроблені умови та експеримент для дослідження току споживання семи при різних режимах.

Була побудована тестова схема, визначений набір вхідних параметрів та способи конфігурування та керування схемою.

Були проведені експерименти з використанням тестової схеми та отримані дані про енергоспоживання семи при різних режимах та початкових параметрах конфігурації.

Був проведений аналіз отриманих даних з побудовою графіків для оцінки результатів та формування висновків на базі аналізу отриманих даних при проведенні експериментів.

1 АНАЛІТИЧНИЙ ОГЛЯД МЕТОДІВ ТА ЗАСОБІВ КОНТРОЛЮ ЦИФРОВИХ FPGA-КОМПОНЕНТІВ У СКЛАДІ КОМП'ЮТЕРНИХ СИСТЕМИ КРИТИЧНОГО ЗАСТОСУВАННЯ

1.1 Огляд платформи FPGA та її місця серед програмованих компонентів комп'ютерних систем

Програмована користувачем вентильна матриця (field-programmable gate array, FPGA) представляє собою матрицю реконфігурованих елементів. З використанням спеціального програмного забезпечення користувач може описати свій проект на мові опису апаратури або у вигляді схеми, а потім реалізувати його в FPGA.

У FPGA можливо реалізувати як комбінаційні, так і послідовних схема. У них можна реалізовувати багаторівневі логічні схеми, тоді як в ПЛІМ можуть бути реалізовані тільки дворівневі схеми. В сучасні FPGA інтегровані інші корисні вузли, такі як помножувачі, високошвидкісні пристрої введення / виводу, ЦАП, АЦП, великі ОЗУ і процесори.

FPGA представляє собою матрицю логічних елементів (logic elements, LE, LE), які також називаються конфігуруються логічними блоками (configurable logic blocks, КЛБ, CLB). Кожен LE можна конфігурувати для виконання функцій деякої комбінаційної або послідовної схеми.

Логічні елементи складаються з невеликих таблиць перетворення і тригерів.

FPGA масштабуються до великих розмірностей з тисячами таблиць перетворення. Xilinx і Altera - два провідних виробника FPGA. Таблиці перетворення і програмовані межсоединения дозволяють створити практично будь-яку логічну функцію. Однак, вони на порядок менш ефективні за показником швидкості і собівартості (на одиницю площі

мікросхеми), ніж апаратно реалізовані версії тих же функцій. Тому FPGA часто включають в себе спеціалізовані блоки - пам'ять, помножувачі і навіть цілі мікропроцесори.

CPU універсальний, на ньому можна запустити будь-який алгоритм, він найбільш гнучкий, і використовувати його найлегше завдяки величезній кількості мов програмування і середовищ розробки.

Через універсальність і послідовне виконання інструкцій CPU знижується продуктивність і підвищується енергоспоживання схеми. Відбувається це тому, що на кожну корисну арифметичну операцію CPU робить багато додаткових операцій, пов'язаних з читанням інструкцій, переміщенням даних між регістрами і кешем, і т.д.

ASIC. На цій платформі необхідний алгоритм реалізується апаратно за рахунок прямого з'єднання транзисторів, всі операції пов'язані тільки з виконанням алгоритму і немає ніякої можливості змінити його. Це забезпечує максимальну продуктивність і найменше енергоспоживання платформи. Перепрограмувати ASIC неможливо.

GPU. Спочатку ці мікросхеми були розроблені для обробки графіки, але зараз використовуються і для обчислень загального призначення. Вони складаються з тисяч невеликих обчислювальних ядер і виконують паралельні операції над масивом даних.

Якщо алгоритм можна розпаралелити, то GPU виконає алгоритм значно швидше в порівнянні з CPU. З іншого боку, послідовні алгоритми будуть реалізовуватися гірше, тому платформа виявляється менш гнучкою, ніж CPU.

FPGA. Ця платформа поєднує ефективність ASIC з можливістю змінювати програму. ПЛІС не універсальні, але існує клас алгоритмів і завдань, які на них будуть показувати кращу продуктивність, ніж на CPU і навіть GPU. Складність розробки під FPGA вище, проте нові засоби розробки роблять цей розрив менше.

1.2. Аналіз архітектурних особливостей мікросхем FPGA

Архітектура мікросхеми FPGA закладається з трьох основних блоків:

- Логічні блоки, що конфігуруються
- Блоки вводу виводу
- Матриця комутації логічних блоків

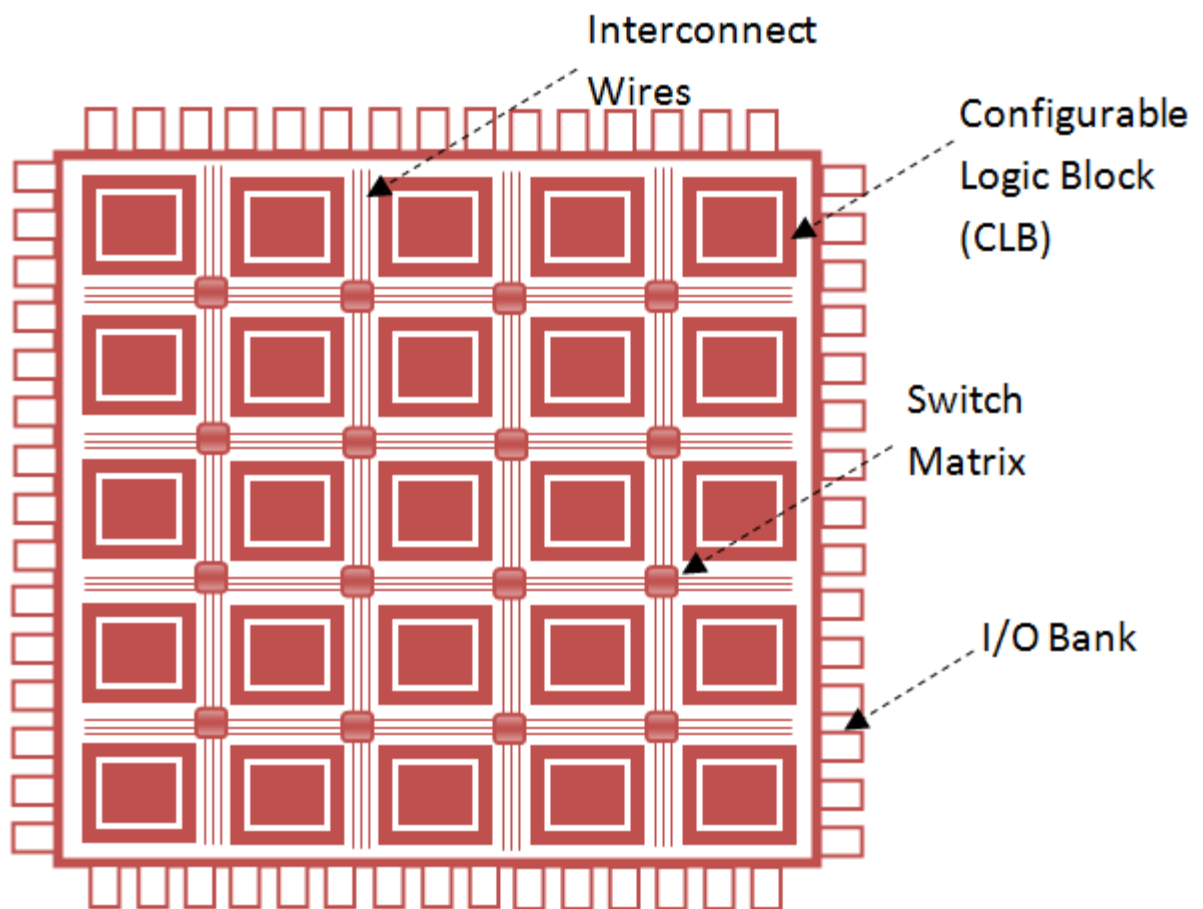


Рисунок 1.1 – Структурна схема архітектури FPGA

Логічні блоки, що конфігуруються складаються з одного або декількох базових логічних елементів. Базові логічні елементи можуть мати різну реалізацію в залежності від мікросхеми або покоління, але завжди мають у своєму складі наступні компоненти:

- LUT
- Триггер
- Конфігураційний реєстр

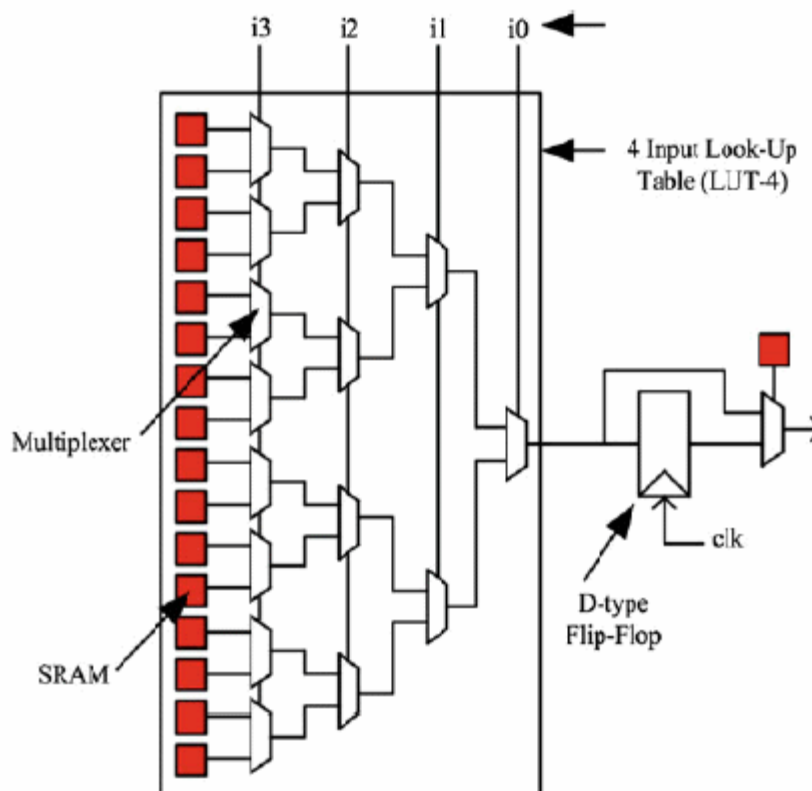


Рисунок 1.2 – Структурна схема базового логічного елемента FPGA

LUT це таблиця перетворення, що виконує зіставлення вхідних даних(у біти у даному прикладі) з одним бітом результату.

Триггер дає можливість запам'ятовувати попередній результат та змінювати свій тільки в моменти, що дозволені синхросигналом.

Конфігураційні реєстри дають можливість описати принцип роботи(таблицю істинності) для блоку LUT та необхідність використовувати триггер на виході схеми чи використовувати поточний результат.

Матриці комутації логічних блоків необхідні для передачі даних від одного логічного блоку до іншого, від входів мікросхеми до логічних блоків

та від логічних блоків до виходів мікросхеми. Існує два основних підходи до побудови зв'язків всередині FPGA: острівна та ієрархічна.

Острівна архітектура передбачає побудову зв'язків рівноправно між усіма блоками вводу виводу та базовими логічними елементами.

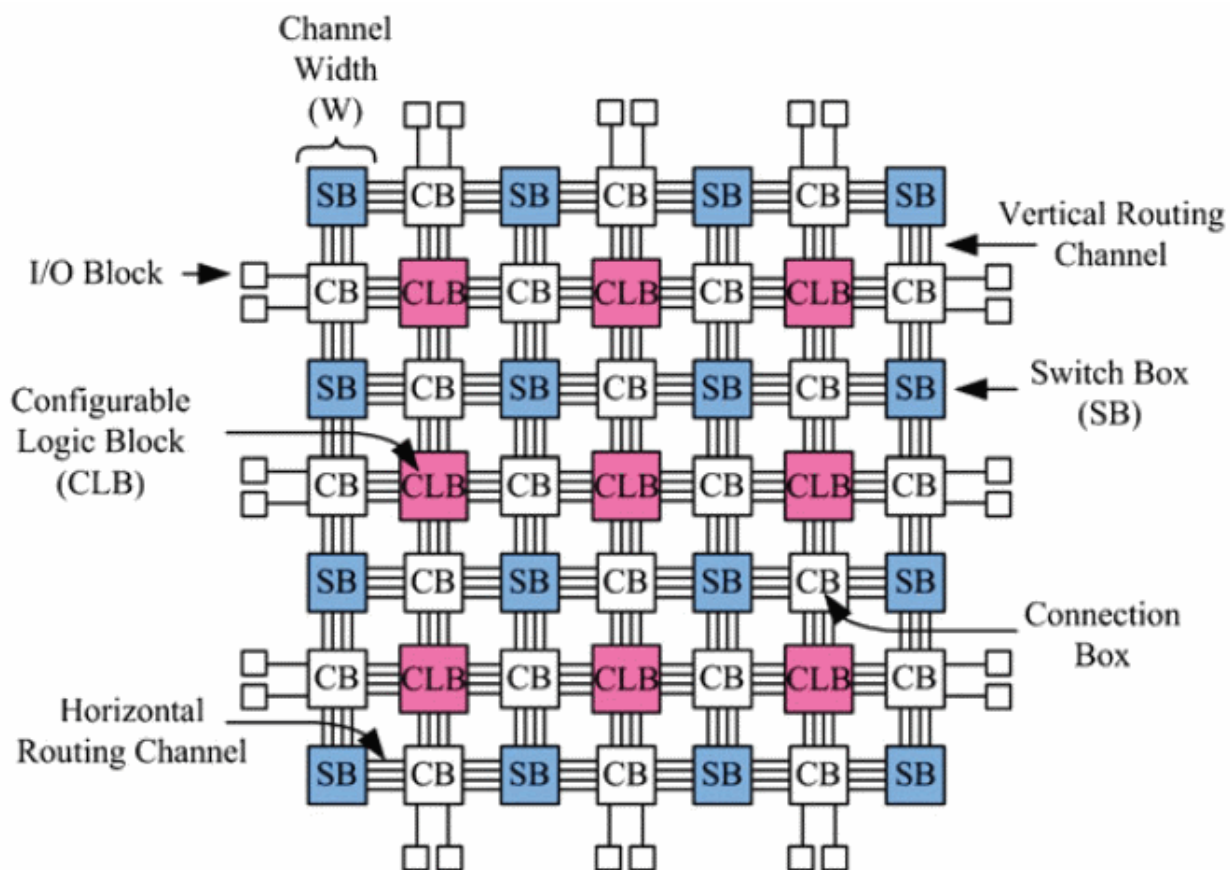


Рисунок 1.3 – Острівна архітектура побудови ліній зв'язку

Ієрархічна архітектура передбачає побудову таким чином, що декілька базових блоків формують осередок в якому комутація виконується набагато швидше ніж між блоками у острівній архітектурі, але в разі необхідності комутації віддалних блоків швидкість значно знижуються, тому високої продуктивності можна досягнути адаптацією схеми до роботи у блочному форматі згідно документації архітектури конкретної мікросхеми.

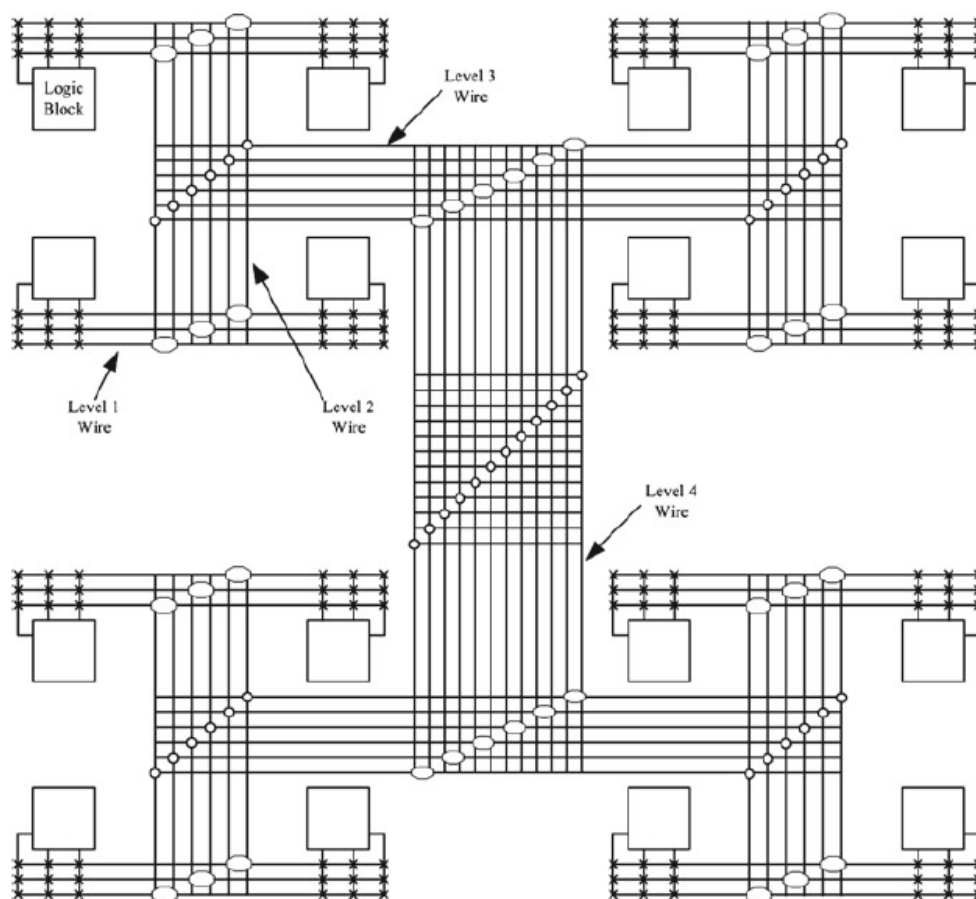


Рисунок 1.4 – Ієрархічна архітектура побудови ліній зв'язку

Блоки вводу виводу FPGA виконують функцію передачі результату з певної точки FPGA матриці на фізичний контакт мікросхеми.

1.3 Галузі застосування мікросхем FPGA

FPGA використовуються в багатьох споживчих продуктах, таких як автомобілі, медичне обладнання, пристрої обробки медіа-інформації. Наприклад, в системах навігації, круїз-контролю, звуковідтворення автомобілів Mercedes Benz S-класу використовується більше десяти FPGA і PLD фірми Xilinx. FPGA дозволяють швидше виводити вироби на ринок і

спрощують налагодження і додавання нових можливостей на пізніх етапах життєвого циклу продукту.

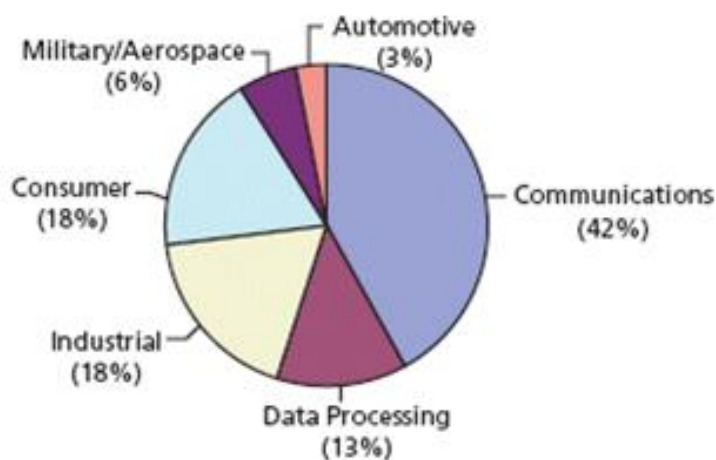


Рисунок 1.5 – Ієрархічна архітектура побудови ліній зв'язку

1.3.1 Зв'язок

Традиційно радіопередавач складається з трьох основних блоків, це антенна прийняття сигналу, блок обробки сигналу(зміна частоти та змісту передачі) та антена для передачі сигналу. При використанні жорстких апаратних методів модифікації сигналу функціональність пристрою обмежується закладеною функціональністю. Використання FPGA та блоків ADC та DAC дає можливість у будь який момент розширити функціональні можливості пристрою без втрати ефективності в зрівнянні з програмною реалізацією, бо FPGA реалізує алгоритм апаратно.

1.3.2 Системи штучного зору

Зараз все більше і більше зустрічаються пристрої, що використовують систему штучного зору. Прикладами є роботи, камери відеоспостереження, автопілоти машин, тощо. Функція аналізу зображення потребує великої продуктивності, що досягається можливістю FPGA створювати найбільш наближене до апаратної реалізації рішення, але дозволяє змінювати та модифікувати алгоритм аналізу.

1.3.3 Медичні системи візуалізації

Медичні системи потребують великої продуктивності обробки вхідних даних, нерідко у реальному часі. Задачі обробки біомедицинських зображень, триманих за допомогою ПЕТ, комп'ютерної томографії, рентгенівських промінів, тривимірних зображень для аналізу потребують багатократного повторення однотипних етапів, що відмінно реалізується за допомогою можливості FPGA виконувати паралельні обчислення.

1.3.4 Шифрування та криптографія

Можливість необмеженого розпаралелювання та здатність до апаратного конфігурування бітності ключа шифрування, низькою затримкою в роботі та достатня тактова частота дає можливість використовувати FPGA для задач шифрування та дешифрувці.

1.3.5 Радіоастрономія

Радіоастрономія це наука, що призначена для вивчення та аналізу явищ, що трапляються у далекому космосі, шляхом уловлювання з нього електромагнітного випромінення. Для аналізу цих даних необхідна велика швидкість та паралельність виконання алгоритмів, тому FPGA використовується для виконання аналізу в цій сфері.

1.3.6 Розпізнавання мови

Розпізнавання мови це метод, що використовується у двох основних напрямках, це пошук інформації та системи безпеки. FPGA дає можливість швидкого аналізу та зіставлення отриманої мови з записаним прикладом.

1.3.7 Аеронавтика та оборона

Існує багато задач по аналізу та керуванню повітряним судном, де необхідна висока надійність та швидкість роботи пристрою. У військових задачах часто може бути важлива характеристика енергоспоживання.

1.3.8 Хмарні центри обробки даних

Інтернет речей (IoT) і великі дані в цілому генерують експоненціальне зростання зібраних і оброблюваних даних, які разом з їх обчислювальним аналізом за допомогою методів глибокого навчання з декількома

паралельними / паралельними операціями. , Призводять до високого попиту на гнучкі і безпечні обчислювальні потужності з малою затримкою, які неможливо вирішити шляхом додавання додаткових серверів / блейд-серверів через шалений збільшення витрат на простір, споживання і гроші. Під цією панорамою двері в світ центрів обробки даних масово відкриваються для ПЛІС, завдяки їх здатності до прискорення обчислень, гнучкості конфігурації і безпеки, яку обладнання гарантує по відношенню до програмного забезпечення

1.3.9 Інжиніринг керування

Можливість реалізувати контролер на основі ПЛІС як систему жорсткого реального часу, яка може реагувати на будь-які критичні за часом зміни в керуючій середовищі протягом розрахованого детермінованого часу. Іншим аспектом є можливість переналаштувати FPGA під час виконання, що дозволяє адаптуватися до змінного середовища, вибираючи найбільш підходящий алгоритм контролера, скорочуючи при цьому необхідні логічні ресурси і час розгортання.

1.4 Особливості програмування мікросхем FPGA та формальних мов, задіяних в процесі програмування

VHDL є аббревіатурою від Very high speed integrated circuits Hardware Description Language, що перекладається як мова опису проєктів на надвеликих інтегральних схемах. Міністерство оборони США і IEEE спонсорували розробку цієї мови опису цифрової апаратури з метою отримання простого у використанні засобу проєктування і моделювання логічних схем для всіх етапів розробки електронних систем, починаючи від модулів мікросхем і закінчуючи великими обчислювальними системами. В даний час діє стандарт VHDL, викладений в документі IEEE 1076-2002 і є

промисловим стандартом, який широко використовується для описання роботи цифрових систем. Версія 3.0 проекту стандарту VHDL 2006, у якій увійшли всі дочірні стандарти, розроблені в ході створення стандартів VHDL (IEEE 1064, 1076.2, 1076.3), а також додані інші удосконалення, такі як інтерфейс з мовами високого рівня C / C ++ і ряд інших. У даній книзі розглянуті основні положення мови VHDL, які визначені в стандарті IEEE 1076-1993, оскільки попередні зміни (за винятком стандарту VHDL 2006, що розробляється) були несуттєвими. Крім того, всі наступні стандарти VHDL, включаючи VHDL 2006, забезпечують сумісність з проектами, розробленими відповідно до стандарту 1076-1993. Ще одним представником мов опису цифрової апаратури є мова Verilog, або Verilog HDL. Розробники Verilog зробили його синтаксис дуже схожим на синтаксис мови C. Verilog має препроцесор дуже схожий на препроцесор мови C, а основні керуючі конструкції Verilog також подібні однойменним конструкціям мови C. В даний час діє стандарт Verilog, викладений в документі IEEE 1364-2005, який являє собою допрацьований варіант дуже популярного серед прихильників Verilog стандарту IEEE 1364-2001. Мова Verilog застосовується в промисловості так само широко, як і VHDL, оскільки обидві ці мови дозволяють описувати та імітувати роботу складних цифрових систем. Третім представником мов VHDL є мова ABEL (Advanced Boolean Equation Language - розширена мова булевих рівнянь), який був розроблений в 1983 році для створення проектів цифрових пристроїв за допомогою програмованих логічних пристроїв (PLD - Programmable Logic Devices). Мова ABEL менш потужний, ніж VHDL і Verilog, зокрема не дозволяє проектувати пристрої на перепрограмованих логічних матрицях FPGA (Field Programmable Gate Arrays), тому він менш популярний в промисловості, хоча все ще знаходить своїх прихильників.

Хоча мова VHDL зовні виглядає так само, як і інші традиційні мови програмування (оскільки має літерали, роздільники, оператори і т. д.), він володіє деякими важливими відмітними характеристиками.

Проекти цифрових пристроїв (ЦУ), створені за допомогою мови VHDL, мають, як правило, ієрархічну структуру. Кожен автономний проєктований модуль (субблок ЦУ, що проєктується) має:

- жорстко визначений інтерфейс взаємодії з іншими модулями;
- точну специфікацію внутрішнього пристрою модуля, що проєктується, що описує концепцію і функціонування модуля.

Специфікації модулів VHDL проєктів можуть використовувати або математичні алгоритми, що описують їх роботу, або опис апаратної структури модуля, що проєктується. У відповідності до рівнів абстракцій проєктів опис модуля може мати поведінкову або структурну форму.

Моделювання алгоритму роботи проєкту ґрунтується на з подієвому принципі керування.

VHDL проєкт дозволяє виконувати моделювання протікання паралельних процесів в електричних схемах, часовий аналіз сигналів і їх параметрів.

VHDL підтримується інструментальними засобами синтезу і системами автоматизованого проєктування (САПР) багатьох виробників програмного забезпечення (ПЗ), які можуть створювати прямо з опису VHDL проєкту його апаратну реалізацію (пов'язані між собою структури логічних елементів, що містяться в НВІС).

Використовуючи VHDL, можна проєктувати, моделювати і синтезувати практично будь-яку, починаючи від простої комбінаційної схеми до закінченої мікропроцесорної системи на НВІС.

Перераховані вище характеристики мови VHDL як спеціалізованої мови опису ЦУ реалізуються за допомогою наступних мовних засобів:

Бібліотеки та пакети.

Проекти: інтерфейс і архітектура проекту.

Підпрограми: функції і процедури.

Скалярні типи даних: перелічуваних, числові, фізичні.

Програмні елементи даних: константи, змінні, сигнали, порти, ідентифікатори.

Математичні операції: логічні, відносин, арифметичні.

Програмні операції: установка значень сигналів, присвоєння значень змінним, реалізація зв'язку портів і сигналів.

Математичні вирази: логічні, алгебраїчні, логіко-алгебраїчні.

Оператори оголошення програмних елементів даних.

Оператори комбінаторної логіки: простий установки значення сигналу (ПУЗС), умовної установки значення сигналу (УУЗС), селективної установки значення сигналу (СУЗС), оператор process, оператор реалізації компонента port map.

Оператори реєстрової логіки: оператор process, ПУЗС, оператор умовної передачі управління, оператор циклу, оператор вибору.

На відміну від процедурних мов програмування, мовні засоби яких забезпечують виконання обчислень над абстрактними даними і управління ними, VHDL проект описує ЦУ, беручи до уваги його багатогранність, поведінку, структуру, функціональні і фізичні властивості, а також взаємодія зі спеціальною апаратурою, фізично реалізує проект ЦУ в НВІС.

VHDL по суті є мовою паралельного програмування, тобто в його конструкції існують оператори, відповідно до логічних вентилів. Ці оператори обробляються (обчислюються) за паралельним принципом. Суть даного принципу полягає в тому, що, як тільки сигнал, що міститься в описі проєктованого ЦУ, змінює своє значення (кажуть, що «відбувається подія на сигналі»), всі оператори, які беруть участь в його обслуговуванні, миттєво запускаються на виконання і одночасно видають кінцевий результат. Тому такі оператори називаються операторами паралельної обробки (concurrency

operator). Програма, написана на VHDL (як і на будь-якому іншому HDL мовою, наприклад Verilog), моделює фізичну поведінку системи (як правило, цифрової), сигнали в якій поширюються миттєво. Така програма дозволяє формувати тимчасову специфікацію (час затримки розповсюдження сигналу на логічне елементі), а також описувати систему як з'єднання різного роду компонентів, або функціональних блоків.

Мова Verilog HDL, скорочено Verilog, була розроблена фірмою Gateway Design Automaton як мова моделювання, орієнтована на внутрішнє використання.

Властивостями мови Verilog є близькість до синтаксису і семантиці мови C, менше в порівнянні з VHDL число службових слів. VHDL поступається Verilog в ефективності, тобто на опис однієї і тієї ж конструкції в Verilog потрібно менше символів ASCII, ніж в VHDL. Базовою одиницею мови Verilog є проектний модуль, що інтегрує, на відміну від відповідних конструкцій VHDL, як визначення інтерфейсу, так і правил функціонування пристрою або блоку. Елементами проектного модуля є декларації та оператори. Концепції мови дозволяють описувати пристрої з використанням різних рівнів абстракції, в тому числі у формі "чистої поведінки", уявлення реєстрових передач і вентильного рівня уявлення. Модуль може містити оператори ввімкнення інших модулів, що дозволяє створювати ієрархічні проекти. Забезпечується підтримка процесів як спадного, так і висхідного проектування.

Поведінкове уявлення дискретних пристроїв задається у формі арифметичних і логічних перетворень над вихідними і проміжними даними. Склад і позначення дозволених операцій відповідає складу операцій мови C його позначенням. Забезпечується можливість подання взаємодіючих підсистем, для чого в мові визначені як паралельні, так і послідовні оператори і процедури. Паралельні оператори відображають поведінку

ланцюгів без пам'яті, а при моделюванні виконуються при зміні будь-якого операнда в правій частині оператора.

Послідовні оператори полягають в виділені програмні блоки і при моделюванні виконуються послідовно один за одним в порядку запису. Результати перетворень доступні для інших блоків програми тільки після виконання всіх вкладених в блок операторів. Для кожного такого блоку можуть бути явно визначені функції блокування або ініціалізації запропонованих перетворень з боку інших блоків (аналогічно з поняттям PROCESS мови VHDL). У число послідовних операторів, зокрема, входять послідовні оператори присвоювання, умовний оператор IF, оператор вибору CASE, оператори повторення LOOP. Вводяться конструкції, які представляють дії, виконувані протягом деякого ненульового тимчасового інтервалу.

Структурний подання проектів забезпечується можливістю вбудовування в проект інших модулів, використовуються традиційні для алгоритмічних мов конструкції, аналогічні викликом відповідних підпрограм. В мові зумовлений широкий набір логічних примітивів, примітивів для подання двонаправлених ланцюгів передачі і резистивних ланцюгів.

Поняття макромодуля з точки зору принципів опису функціонування не відрізняється від поняття модуля. Макромодуль має незначні обмеження щодо використання деяких конструкцій. Відрізняється організація процедури моделювання ієрархічних проектів. Модуль компілюється таким чином, що при моделюванні вкладені програмні одиниці (підпрограми, оператори входження модулів) інтерпретуються як виклики відповідних процедур. Компіляція макромодуля передбачає пряме вкладення операторів макромодуля в програму. При цьому ряд внутрішніх змінних макромодуля може бути недоступними, бо їх визначення увійде в інші вирази.

Використання концепції макромодуля забезпечує зменшення часу моделювання.

У тілі модуля паралельні оператори і декларації розміщуються в довільному порядку, хоча будь-який об'єкт декларується раніше його використання в операторах.

Intel Quartus Prime Pro Edition призначена для роботи з поширеними опціями FPGA і SoC останнього покоління, такими як Intel Stratix 10 Intel Arria 10 Intel Cyclone 10 GX.

The Intel Quartus Prime Standard Edition включає в себе повну підтримку пристроїв попередніх поколінь, а також сімейства Intel Cyclone 10 LP.

The Intel Quartus Prime Lite Edition являє собою інструмент для роботи з сімейством масового сегмента.

Xilinx Vivado Design Suite - програмний пакет, створений Xilinx для синтезу та аналізу конструкцій HDL, який замінює Xilinx ISE додатковими функціями для розробки системи на кристалі і синтезу високого рівня.

1.5 Аналіз стандартів функціональної безпеки комп'ютерних системи критичного застосування

1.5.1 Визначення та галузі використання комп'ютерних системи критичного застосування

Комп'ютерні системи критичного застосування використовуються в разі необхідності виконувати закладені функції з великим показником надійності. Під керуванням систем критичного застосування знаходяться найбільш небезпечні об'єкти, що створила людина, такі як атомні та гідроелектростанції, нафтові вишки тощо.

Системи критичного застосування це програмно апаратний комплекс, що може використовувати в якості апаратного забезпечення мікроконтролери, FPGA пристрої, а в якості програмного забезпечення мови

програмування, такі як C, або мови описання апаратури, такі як VHDL або Verilog.

Системи критичного застосування відрізняються увагою до процесу проектування системи, щоб забезпечити необхідні показники обслуговуємості, діагностування, контролю та надійності.

Інформаційні технології не несуть безпосередньої загрози життю та здоров'ю людини, вони направлені на автоматизацію рутинних процесів, підвищенню ефективності роботи, швидкому розповсюдженню важливої інформації. Ми довіряємо їм свої особисті дані, керування найбільш складними та небезпечними об'єктами, що створила людина. В цей момент стає актуальними два основних поняття: функціональна(safety) та інформаційна(security) безпека.

Функціональна безпека означає можливість системи виконувати закладені в неї алгоритми без збоїв у необхідний час.

Інформаційна безпека передбачає запобігання спотворенню, зміні, несанкціонованому доступу, розкриттю, запису чи видаленню інформації.

1.5.2 Стандарти функціональної безпеки

Базовим стандартом, який визначає вимоги до функціональної безпеки є МЕК 61508 “Функціональна безпека систем електричних, електронних, програмованих електронних, пов'язаних з безпекою” (IEC 61508 Functional safety of electrical/electronic/programmable electronic safety-related systems).

МЕК 61508 визначає функціональну безпеку як коректне функціонування системи керування обладнанням та коректне функціонування обладнання, яке керується. Таким чином, для забезпечення функціональної безпеки необхідно спочатку визначити функції безпеки (safety functions), необхідні для зниження ризику керованого устаткування, а також для досягнення і збереження цим обладнанням безпечного стану (наприклад, функції протиаварійного захисту). Далі, система управління

повинна мати властивість повноти безпеки (safety integrity), під яким МЕК 61508 визначає, що система буде коректно виконувати функції безпеки при всіх заданих умовах протягом заданого інтервалу часу.

При забезпеченні повноти безпеки (safety integrity) враховуються два типи відмов: випадкові (random failures) і систематичні (systematic failures).

Випадкові відмови викликані виходом з ладу апаратних компонентів і усуваються такими методами, як резервування, самодіагностика, фізичне і електричний поділ компонентів, підвищення стійкості до зовнішніх впливів.

Систематичні відмови викликані помилками проектування, в тому числі, і помилками програмного забезпечення. Усунення систематичних відмов можливо шляхом вдосконалення процесів проектування і розробки, тестування, управління конфігурацією, проектного менеджменту тощо Крім того, оскільки класичне резервування не дозволяє уникнути систематичних відмов, застосовується диверсне (diversity) резервування, коли резервні канали розроблені із застосуванням різного програмного і апаратного забезпечення.

Положення МЕК 61508 деталізовані для потенційно небезпечних областей.

- IEC 61511, Functional safety - Safety instrumented systems for the process industry sector;

- IEC 62061, Safety of machinery - Functional safety of electrical, electronic and programmable electronic control systems;

- IEC 61513, Nuclear power plants - Instrumentation and control for systems important to safety;

- ISO 26262, Road vehicles - Functional safety;

- EN 50129, Railway Industry Specific - System Safety in Electronic Systems;

- IEC 62304, Medical Device Software.

1.6.3 МЕК 61508

Стандарт оперує терміном електрична / електронна / програмована електронна (Е / Е / ПЕ) система (electrical / electronic / programmable electronic).

Особливістю стандарту є ризик-орієнтований підхід. Залежно від ризику, який техногенний об'єкт створює для навколишнього середовища, життя і здоров'я людей, встановлюються ризики для відмов систем управління.

Для зниження значень ризиків реалізується комплекс організаційно-технічних заходів, які також регламентовані в МЕК 61508, в залежності від допустимої величини ризику відмови.

МЕК 61508 являє собою верхній рівень сімейства галузевих стандартів, які деталізують вимоги до функціональної безпеки для систем управління медичним обладнанням, автомобільним та залізничного транспортом, АСУ ТП і т.д.

Серія стандартів МЕК 61508 включає 7 частин, що продемонстровані на рисунку нижче.

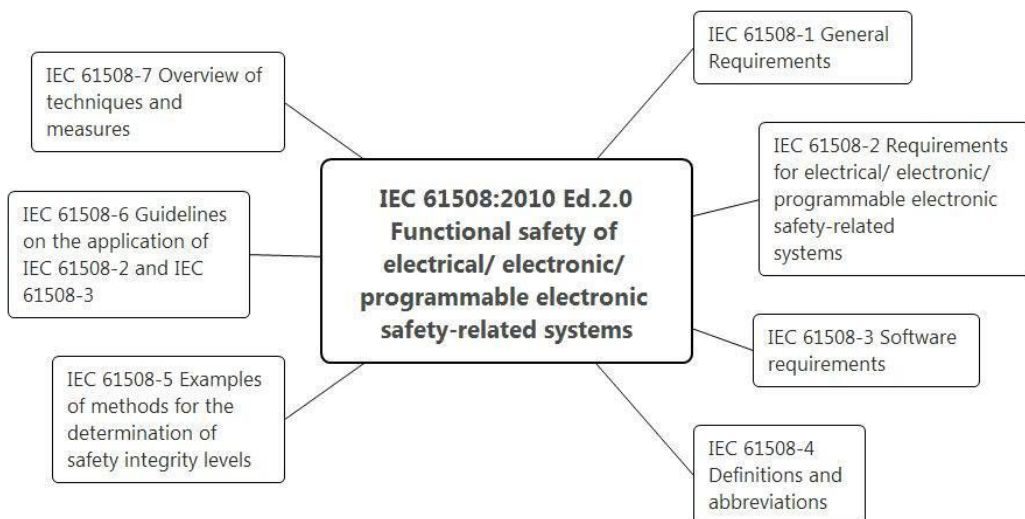


Рисунок 1.1 – Структурна схема стандарту МЕК 61508

Кожна з цих семи частин ділиться на підрозділи та має складну

структуру зв'язків з іншими частинами та підрозділами.

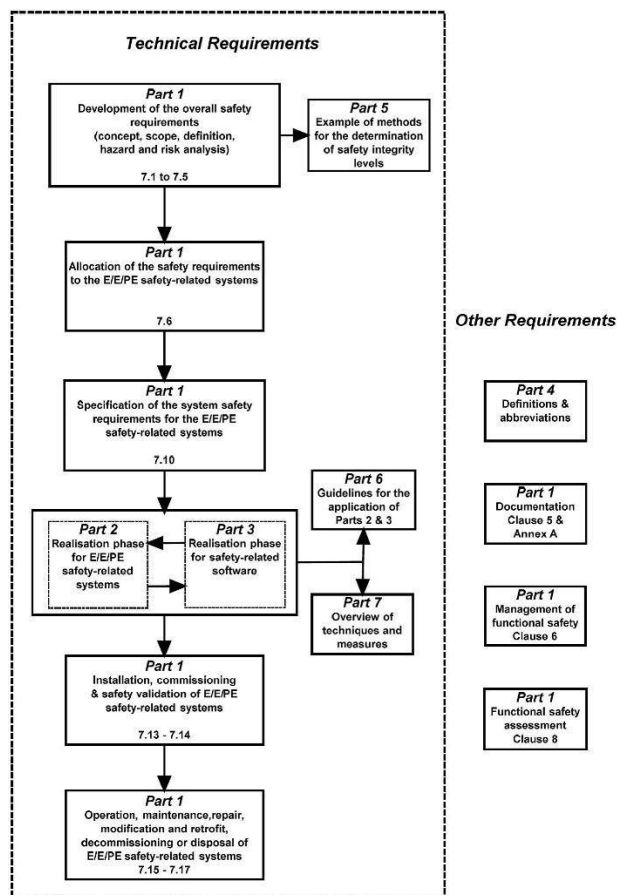


Рисунок 1.2 – Структурна схема взаємозв'язків в стандарті МЕК 61508

1.6 Визначення та підвищення показника контролепридатності цифрових схем комп'ютерних систем

1.6.1 Логічна контролепридатність

Логічна контролепридатність направлена на оцінку складності синтезу тестів для відстежування помилкової роботи пристрою.

Логічна контролепридатність може використовуватися в одному з підходів до функціонального відстеження роботи пристрою:

Тестове діагностування

Робоче діагностування

Дворежимне діагностування

Тестове діагностування гуртується на періодичному імітуванні аварійного режиму роботи для виявлення помилкового режиму роботи пристрою. Такий підхід вносить великі ризики щодо помилкового ввімкнення режиму імітації під час нормальної роботи пристрою. Для синтезу тестів це підхід використовує тільки функціональну модель пристрою.

Робоче діагностування не використовує поняття робочого режиму та режиму імітації аварійної ситуації, а дозволяє виконати аналіз цифрової схеми на предмет точок накопичення прихованих несправностей беручи до уваги функціональну модель пристрою та вхідні дані нормального режиму.

Дворежимне діагностування пропонує використання одного з двох шляхів виявлення точок потенційного накопичення прихованих несправностей. Перший шлях це зробити контролепридатність однаковою для двох режимів шляхом , щоб приховані несправності в нормальному режиму залишалися прихованими і в аварійному. Другий шлях це виявлення потенційно небезпечних точок та усунення тих з них, які можуть задіяні в аварійному режиму ґрунтуючись на даних, які можуть поступати на вхід пристрою в аварійному режимі.

1.6.2 Енергетична контролепридатність

Методи забезпечення функціональної відмовостійкості гуртуються на можливість пристрою виконувати обчислення в будь-якому з режимів роботи. Кожен пристрій складається з декількох логічних блоків:

Периферія

Блоки зосередженої пам'яті

Блоки PLL

Блоки Множення

Кожен з цих блоків має свою частоту роботи, це означає, що використовує свій тактовий генератор. В разі виходу зі строю тактового генератора виконання програми, що закладена в пристрій, зупиняється і данні на виході пристрою “застигають”. При таких умовах програмні модулі, що призначені для відлову та усунення помилок також перестають працювати.

В такому випадку енергоспоживання пристрою різко зростає, так як транзистори всередині кристалу перестають змінювати свій стан.

Відстеження зміни енергоспоживання дає можливість зробити висновки щодо коректності роботи без необхідності відстежувати процеси логічної обробки даних всередині пристрою.

Енергетична контролепридатість направлена на оцінку складності відстежування режиму роботи пристрою за його енергоспоживанням.

1.7 Висновки

В даному розділі магістерської кваліфікаційної роботи було виконано аналітичний огляд сфери проектування комп'ютерних систем критичного застосування, апаратних та програмних засобів, що використовуються, а також стандартів безпеки, яким повинні відповідати системи критичного застосування. Інформація, що були вивчена у цьому розділі буде використовуватися в подальших розділах в якості теоретично основ для розробки методики експерименту.

2 ПРОПОНОВАНИЙ ПІДХІД ДО ДОЛІДЖЕННЯ ЕНЕРГО-ОРІЄНТОВАНОЇ КОНТРОЛЕПРИДАТНОСТІ БАГАТООПЕРАНДНОГО ДОДАВАЧА

2.1 Аналіз проблеми контролю цифрових FPGA-компонентів у складі комп'ютерних системи критичного застосування

Людство незупинно розвивається, створюються нові технології та винаходи, населення на планеті також зростає. Для забезпечення бурхливого розвитку технологій та потреб людства необхідні ресурси. На даний момент забезпечення будь якими ресурсами нерозривно зв'язано з електричною енергією, бо виробництво більшої кількості необхідних ресурсів атоматизовано та його нарощування нерозривно зв'язано з використання більшої кількості приладів, а це потребує більшої кількості електроенергії.

Для отримання електроенергії використовують електростанції, одними з найефективніших типів електростанцій вважаються атомні та гідроелектростанції. Вони дають можливість отримання енергії з мінімальною кількістю ресурсів, що витрачаються, та максимальною кількістю енергії, що отримується, також вважається, що такі типи електростанцій не забруднюють зовнішнє середовище. Це вірно лише, якщо не брати до уваги можливість аварії на таких електростанціях. Аварія на атомній електростанції спроможна зробити непридатними до життя людини тисячі квадратних кілометрів, разом з інфраструктурою, що залишиться в цій зоні та не виправними наслідками для життя та здоров'я людей.

Для розрахунку ризику подій, у тому числі і техногенних катастроф, використовується два основних параметри, це імовірність виникнення події, та тяжкість наслідків від її траплення, в результаті бурхливого розвитку ми

все більше и більше збільшуємо обсяги виробництва енергії, тобто збільшуємо потужність існуючих джерел та будуємо нові, що збільшує другий фактор розрахунку ризику події. Для утримання показника ризику придатного для забезпечення необхідної безпеки маємо тільки один шлях, а саме зменшення ймовірності виникнення події.

Для зменшення ймовірності події необхідно вдосконалення інформаційних технологій темпами, не меншими за швидкість зростання необхідності ресурсів, а ще краще випереджуючи її. Інформаційні технології імплементовані в комп'ютерні системи для керування небезпечними об'єктами підвищеного ризику та стримуванню аварій, що можуть траплятися на них.

Використання комп'ютерних систем для керування об'єктами підвищеного ризику перетворює їх на системи критичного застосування, що мають відмінність, яка проявляється при їх проектуванні, а саме розділення режимів їх роботи на два основних: нормальний режим та аварійний режим.

Нормальний режим характеризується значною тривалістю - десятки років і незначною зміною та різноманітністю вхідних даних, часто на рівні шуму.

Аварійний режим відрізняється малою вивченістю - основний досвід дається дорогою ціною та набувається в результаті аварій, - а також підвищенням активності та різноманітності вхідних даних.

Для забезпечення необхідного рівня надійності роботи комп'ютерних систем критичного застосування необхідно забезпечити великий рівень контролепридатності.

Контролепридатність це можливість оцінки результатів роботи схеми для виявлення появи несправностей, та сигналізування про необхідність обслуговування, ремонту чи зміни модулів, що вийшли з ладу та в ході виконання функцій, для яких вони спроектовані, дають помилкові результати.

На даний момент контролепридатність комп'ютерних систем критичного застосування дає можливість оцінки тільки поточних результатів роботи системи, та не дає можливості оцінки коректності роботи модулів, що не завіюються на даний момент. Системи критичного застосування проектуються для роботи у тривалий проміжок часу, більше десяти років, та протягом усього періоду експлуатації вхідні дані можуть змінюватися не значно, та це буде свідчити про високий рівень роботи керуючих технологій, але така ситуація може привести до того, що буде задіяна тільки мала частина схеми пристрою. При такому розділенні маємо малу ділянку схеми, коректність роботи якої ми перевіряємо безперервно, та усю іншу схему, коректність роботи якої ніяк не перевіряється. В такому випадку з кожним днем роботи у режимі з невеликою задіяною ділянкою підвищується імовірність виникнення несправності у частині, схеми, що не перевіряється та неможливість визначення цього.

Будь яка схема комп'ютерних систем критичного застосування має в собі такі базові елементи як тригери, що дозволяють зберігати інформацію для подальших обчислень.

Будь який тригер має наступні групи входів та виходів:

- інформаційні входи
- входи синхронізації
- вихід стану

Інформаційні входи використовуються для обрання наступного стану тригеру.

Вхід синхронізації використовується для ініціації зміни стану тригером.

Вихід стану тригеру використовується для отримання поточного стану тригеру.

При появі сигналу на вході синхронізації тригер обов'язково виконає зміну свого стану на протилежний або на той самий. При відсутності синхро-сигналу.

Для цього буде витрачена електрична енергія. Синхро-сигнали наявні в схемі в будь який момент її роботи, це означає, що без залежності від даних, які обчислює схема виконується зміна стану кожного тригеру, що є її складовою.

При виникненні несправності та відмиканні синхронізації від одного чи багатьох тригерів це призведе до деяких варіантів наслідків, одні з яких описані нижче:

- Блокування отримання одного, деяких або усіх вхідних бітів
- Блокування системи контролю коректності обчислення

Обидва з наведених варіантів розвитку подій проявляться тільки при різкій зміні вхідних даних, що буде свідчити про аварію, але пристрій вже не буде мати можливості виконати закладені в нього функції.

Беручи до уваги особливість сигналів синхронізації завжди призводить до споживання енергії елементом, можна виконувати відстеження коректності роботи пристрою за його енергоспоживанням. Енергоспоживання схеми виражається у міліамперах.

Для виконання відстеження роботи схеми за її енергоспоживанням необхідно визначити які основні режими роботи схеми можуть бути розглянуті саме в залежності від енергоспоживання. В залежності від енергоспоживання можна виділити чотири основних режими:

- Режим нульової активності
- Режим нормальної роботи
- Режим роботи при над стабільних даних
- Режим аномально низького енергоспоживання

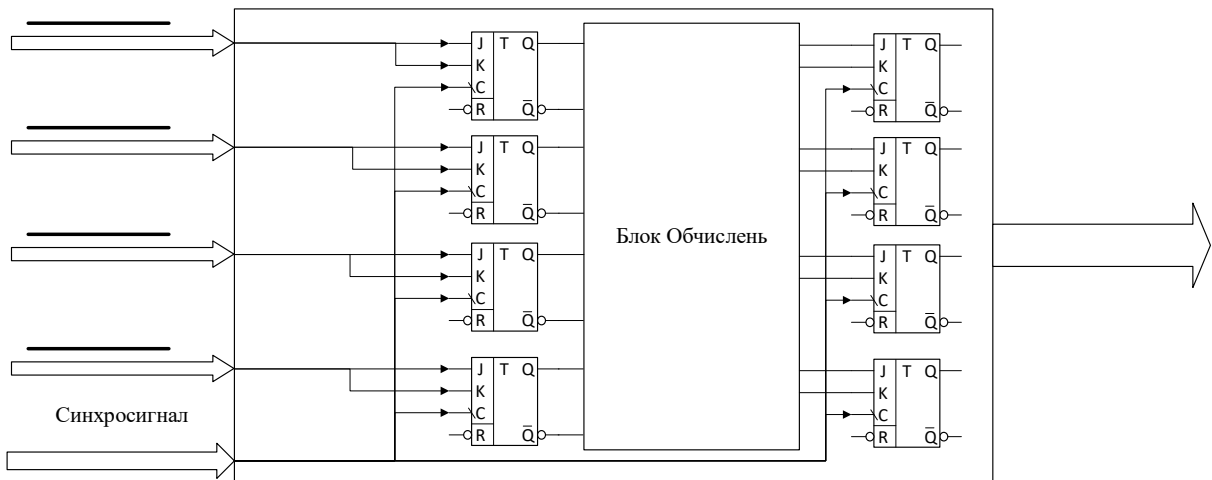


Рисунок 2.1 - Режим нульової активності

Режим нульової активності це режим при якому синхронізація працює справно, та всі елементи, що повинні мати залежність від неї мають з нею зв'язок. Цей режим проявляється при майже нульових даних на вході схеми, тобто при мінімальній її роботі.

Режим нормальної роботи характеризується значною зміною вхідних даних, та задіянням різних ділянок схеми, що дає можливість контролю за допомогою логічної контролепридатості.

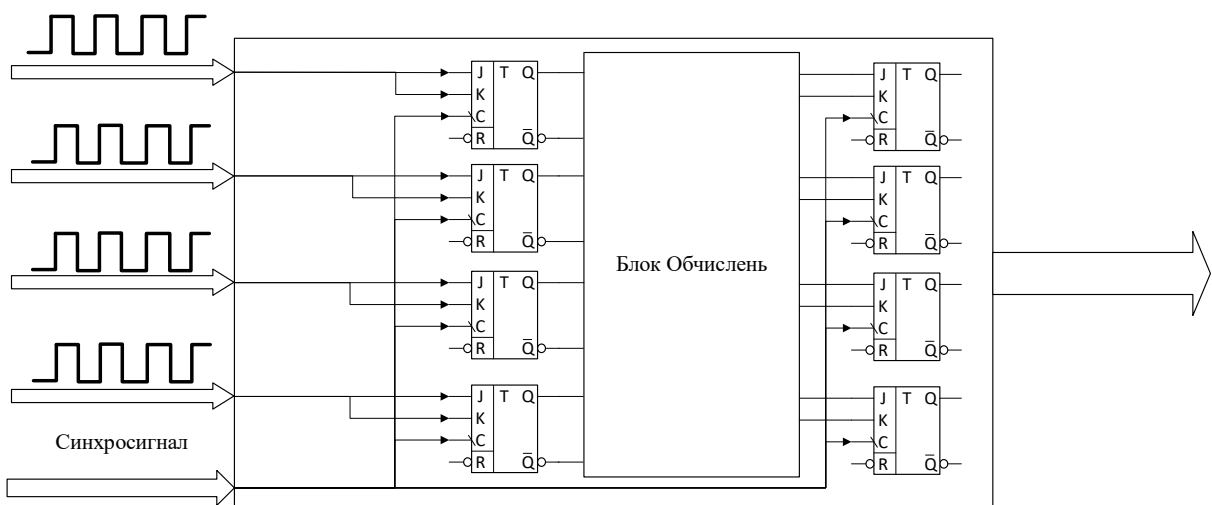


Рисунок 2.2 - Режим нормальної роботи

Режим роботи при над стабільних даних це режим при якому схема працює справно , але входні дані змінюються дуже слабо, що задіє для обчислення лише одну ділянку схеми.

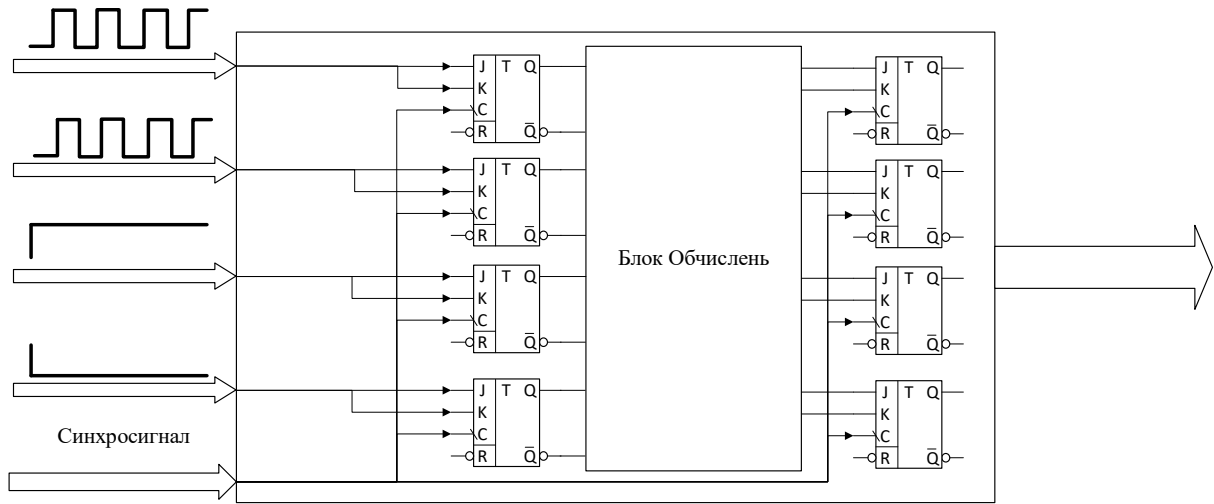


Рисунок 2.3 - Режим роботи при над стабільних даних

Режим аномально низького енергоспоживання характеризується зниженню енергоспоживання схеми менше ніж при режимі нульової активності, що свідчить, що деякі з обов'язкових споживачів не підключені до синхронізації.

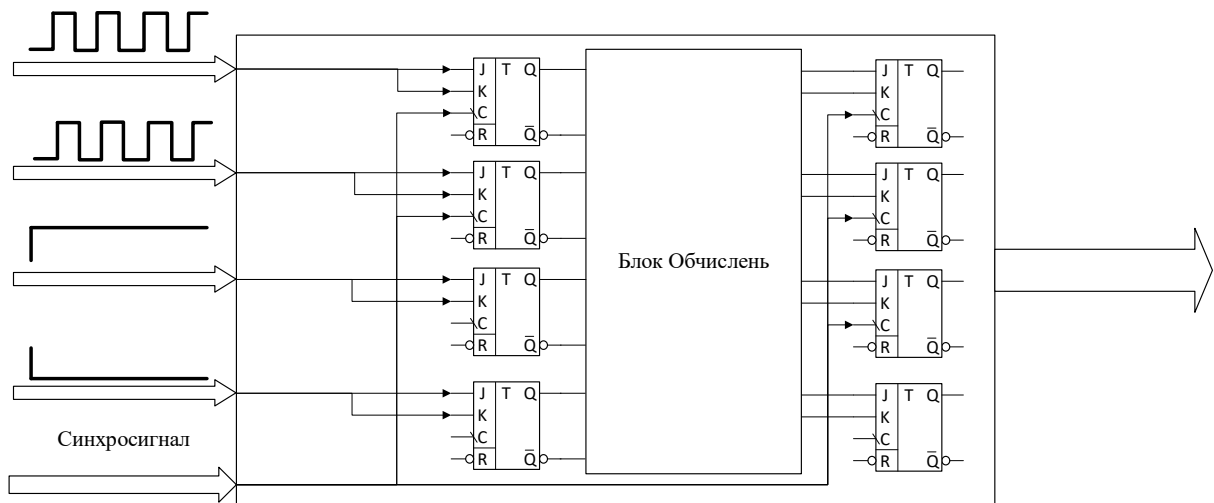


Рисунок 2.4 - Режим аномально низького енергоспоживання

2.2 Постановка мети та задач експериментального дослідження

Для розробки енерго-орієнтованого підходу до контролепридатності схем необхідно виконати дослідження впливу роботи логічних елементів, з яких може складатися схема пристрою на загальне енергоспоживання.

Для виконання даної магістерської роботи було обрано елемент багатооперандний суматор для оцінки його впливу на енергоспоживання схеми при різних режимах роботи.

Метою експериментального дослідження є отримання даних про енергоспоживання схеми при різних режимах роботи. Аналіз цих даних на предмет залежності від режимів роботи схеми.

2.3 Опис концепції дослідження енерго-орієнтованої контролепридатності багатооперандного додавача

Для виконання дослідження енерго-орієнтованої контролепридатності багатооперандного додавача необхідно виконати побудову тестової схеми, що буде відповідати усім потребам експерименту. Схема повина мати можливість налаштування по деяким параметрам, а саме:

- Кількість розрядів, що відмикаються від синхронізації
- Кількість розрядів, що змінюється
- Частота роботи схеми
- Розрядність операндів багатооперандного суматора

Частоту роботи схеми необхідно налаштовувати , бо при різних частотах логічні елементи та тригери споживають різну кількість енергії.

Кількість розрядів, що змінюється необхідно змінювати для дослідження впливу збільшення активності вхідних даних на можливість визначення відключення від синхронізації одного розряду.

Кількість розрядів, що відмикаються від синхронізації необхідно змінювати для дослідження динаміки зміни енергоспоживання від кількості тригерів, що відімкнулося від синхронізації.

Розрядність операндів багатооперандного суматора необхідно змінювати, бо при різній розрядності задіюється різна кількість базових логічних елементів, а це означає , що і енергоспоживання може змінюватися.

Для проведення експериментів необхідно побудувати схему, що надасть можливість імітування різних режимів роботи та зміни необхідних параметрів для дослідження струмів споживання.

Схему необхідно побудувати у програмному забезпеченні САПР, що має можливість формування конфігурації для FPGA мікросхеми.

Для проведення експерименту необхідно виконати аналіз лабораторного стенду, та обрати стенд з вбудованою можливістю вимірювання струму споживання.

Кожен експеримент повинен проходити в три основних етапи:

- Конфігурування схеми
- Дослідження струмів споживання
- Фіксація результатів

Конфігурування схеми має проводитися у програмному забезпеченні САПР, що має можливість програмування FPGA мікросхеми, що буде обрана для досліджень.

Дослідження струмів споживання має проводитися за допомогою програмного забезпечення, що може виконувати зчитування даних датчика струму, що вбудовано в тестовий стенд. Для зміни режимів роботи схми необхідно виконувати зовнішній вплив на поточний стан схеми.

Фіксація результатів виконується у табличному виді у матеріалах дипломної роботи для подальшого аналізу.

Для виконання аналізу отриманих результатів необхідно побудувати графічне відображення струму споживання в залежності від усіх зазначених факторів для можливості формування висновків про вплив змінних факторів на енергоспоживання схеми та можливість фіксації відключення від синхронізації одного і більше тригерів.

2.4 Висновки

В результаті проведення експериментів має бути отримано графічне та табличне представлення даних про енергоспоживання схеми при різних режимах роботи та проведений аналіз, що визначить можливість чи не можливість відстеження відключення від синхронізації одного чи більше тригерів, що мають входити в склад схеми для дослідження.

3 ЕКСПЕРИМЕНТАЛЬНЕ ДОСТІДЖЕННЯ ЕНЕРГО-ОРІЄНТОВАНОЇ КОНТРОЛЕПРИДАТНОСТІ БАГАТООПЕРАНДНОГО ДОДАВАЧА

3.1 Середовище проведення експериментального дослідження

Для проведення експериментів з дослідження контролепридатності багатооперандного додавача використовувалося наступне обладнання:

- Персональний комп'ютер
- Стенд Intel Cyclone 10 LP

Персональний комп'ютер використовується для виконання побудови та модифікації схеми для дослідження, отримання даних, щодо споживання току стендом, конвертації схеми до файлів конфігурації, завантаження конфігурації до стенду, та має наступні характеристики:

- Центральний процесор: Intel Core I5 4670K 2.4 GHz
- Об'єм оперативної пам'яті: 22 GB
- Операційна система : Windows 10

Стенд Intel Cyclone 10 LP використовується для виконання функцій, що закладено у схему FPGA, отримання даних від зовнішніх периферійних пристроїв, обчислення математичних операцій. Стенд Intel Cyclone 10 LP має наступну структурні схему.

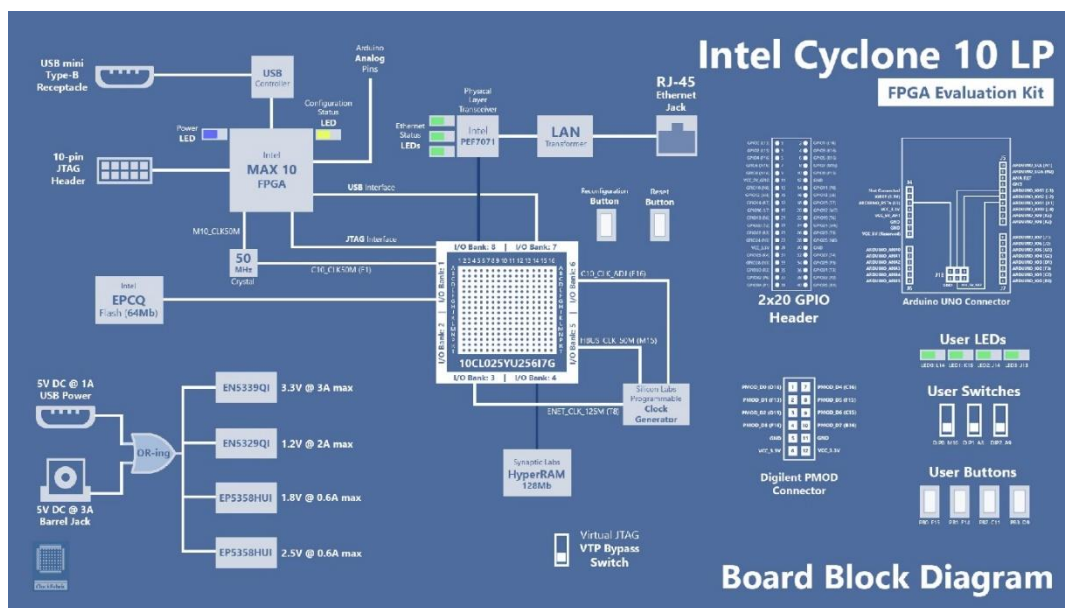


Рисунок 3.1 – Структурна схема Intel Cyclone 10 LP

Intel Cyclone 10 LP складається з декількох основних блоків:

- USB Контроллер – використовуються для підключення стенду до персонального комп'ютеру за допомогою USB кабелю А типу.
- LAN Контроллер – використовується для можливості взаємодії стенду з пристроями, що підтримують інтерфейс Ethernet , та протоколи передачі даних, такі як TCP, MQTT, MODBUS та інші.
- ОЗУ 128 MB – може використовуватися при обчислення великої кількості даних для зберігання проміжкових результатів та операндів
- Flash 64 MB – може використовуватися для зберігання конфігурації для подальшого зчитування мікросхемою 10CL025YU256I7G та конфігурації її належним чином.
- Блок перемикачів та кнопок – використовується в якості найпростішого методу впливу на роботу схеми стенду
- Блок Світлодіодів – використовується в якості найпростіших індикаторів роботи стенду та різних його режимів в залежності від схеми, що зіставлено.

- Intel MAX 10 FPGA – FPGA мікросхема, що використовується в якості допоміжної у стенду Intel Cyclone 10 LP та виконує функції відображення стану системи живлення стенду, реалізації можливості зчитування аналогових сигналів, конвертації USB інтерфейсу у JTAG інтерфейс та програмування основної FPGA мікросхеми.
- Intel 10CL025YU256I7G – основна FPGA мікросхема, що має в своєму складі 24624 базових логічних елемента та використовується для виконання функцій, що закладено в схемі проекту, що створено у середовище Quartus.

3.2 Опис схеми для проведення експериментального дослідження

3.2.1 Загальний опис експериментальної схеми

Для імітування різних режимів роботи та умов використання буде використовуватися наступна схема.

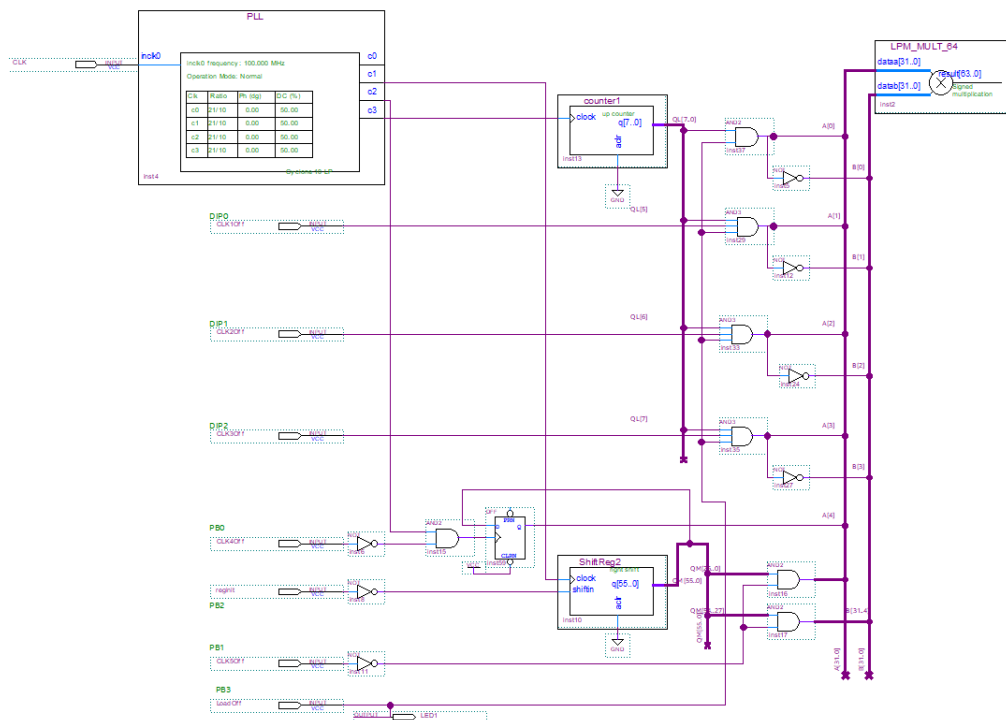


Рисунок 3.2 – Схема імітування різних умов використання мікросхеми.

Схема, що представлена складається з деяких основних логічних блоків:

- Блок формування базової частоти синхронізації схеми
- Блок формування змісту операндів багатооперандного суматора
- Блок обчислення

3.2.1.1 Блок формування базової частоти синхронізації схеми

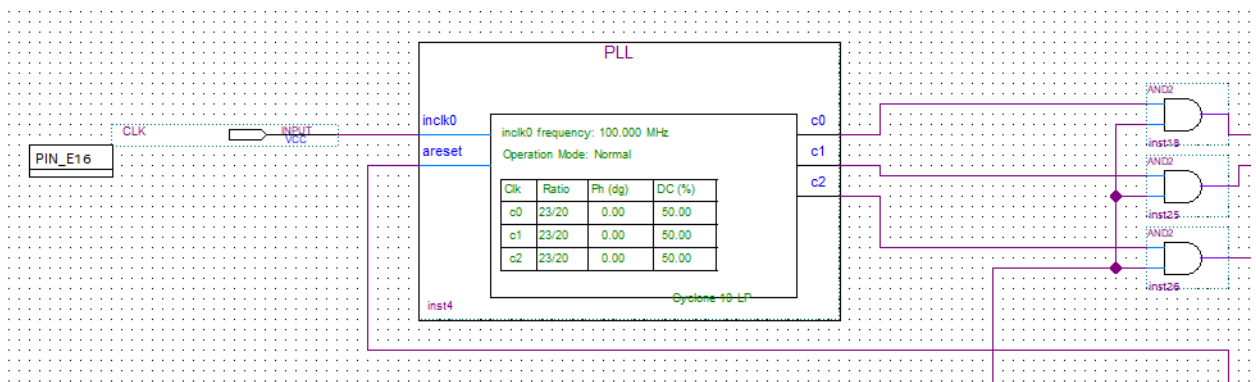


Рисунок 3.3 – Блок формування базової частоти синхронізації схеми.

Блок формування базової частоти функціонування складається з декількох елементів :

- Блок PLL
- 3 логічних елемента І
- Елемент отримання базової частоти з кварцового генератора

Блок PLL виконує перетворення одної частоти на іншу завдяки двом коефіцієнтам, коефіцієнт на який буде помножена вхідна частота та коефіцієнтна який буде розділено результат множення вхідної частоти на перший коефіцієнт.

Блок PLL має 2 входи та 3 виходи, перший вхід використовується для

отримання вхідної частоти. Другий вхід призначений для переводу блоку у режим скидання.

З виходи блоку PLL використовуються для отримання результату його роботи, а саме перетвореної частоти згідно з коефіцієнтами множення та поділу частоти.

З логічних елементи використовуються для відключення блоку PLL від схеми, для того, щоб таким чином імітувати вихід з ладу системи синхронізації.

Елемент отримання базової частоти з кварцового генератору використовується для отримання базової частоти від кварцового резонатору та передачі на вхід блоку перетворення частоти.

3.2.1.2 Блок формування змісту операндів багатооперандного суматора

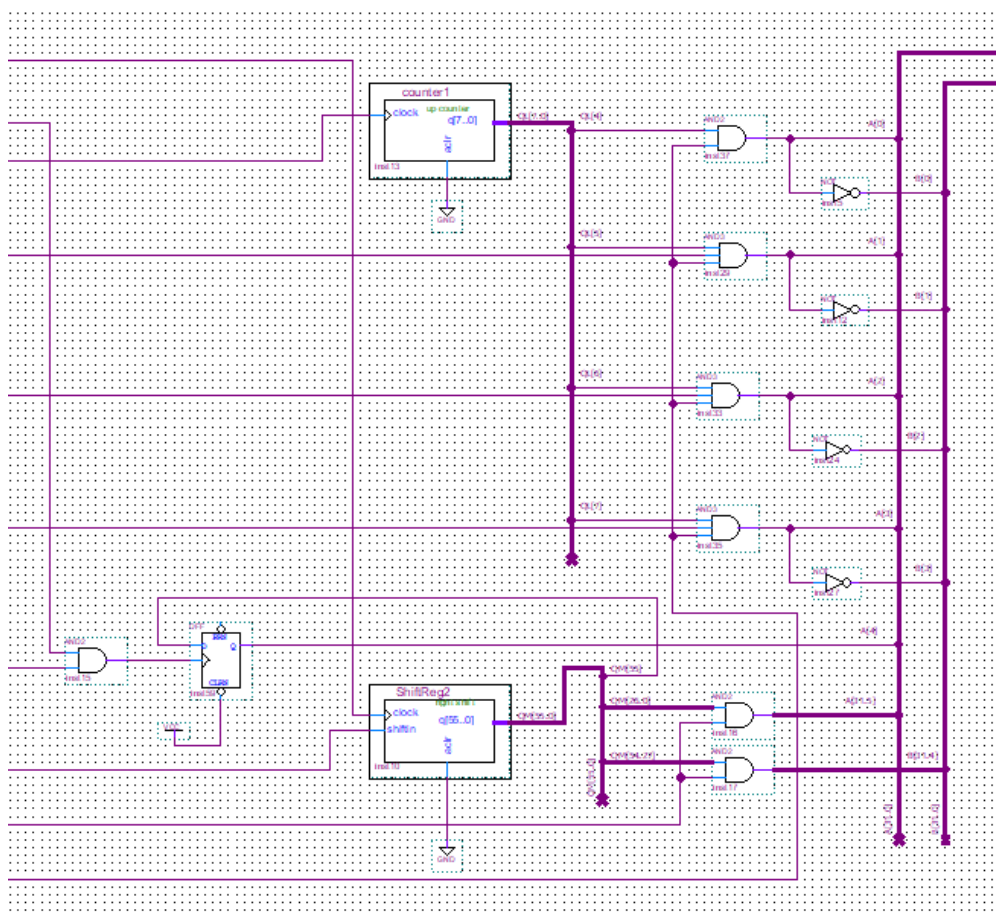


Рисунок 3.4 – Блок формування базової частоти синхронізації схеми

Блок формування базової частоти функціонування складається з декількох елементів :

- Лічильник
- Тригер
- Логічні елементи І
- Інвертори
- Зсувовий регістр

Блок формування операндів забезпечує формування операндів для трьох режимів роботи схеми:

- Режим нульової активності
- Режим нормального навантаження
- Аварійний стан

Для забезпечення режиму нульової активності виконується занулення результатів роботи лічильника за допомогою логічних елементів І, та тригер для імітування відключення старших розрядів ввімкнених.

Для забезпечення режиму нормального навантаження виконується передача результатів роботи лічильника в якості бітів для вхідних операндів батооперандного суматора, та тригер для імітування відключення старших розрядів ввімкнених.

Для забезпечення аварійного стану виконується вимкнення тригеру для імітування відключення старших розрядів.

Лічильник має 2 входи та один вихід. Перший вхід – тактовий, при зміні сигналу на цьому вході лічильник виконує операцію збільшення на одиницю свого внутрішнього регістра. Другий вхід використовується для очищення внутрішнього регістру лічильника. Вихід лічильника представляє собою поточне значення внутрішнього регістру.

Лічильник використовується для формування різних даних для

різних бітів операндів багатооперандного суматора, та для імітування виконання багатооперандним суматором корисної роботи.

Логічні елементи І призначені для імітування нульової активності чи корисної роботи на одному або декількох бітах чи групах бітів операндів багатооперандного суматора.

Тригер призначений для імітування виходу з ладу\відключення від синхронізації одного або декількох старших бітів операнду багатооперандного суматора.

Зсувсовий регістр має три входи та один вихід. Перший вхід тактовий, при зміні сигналу на цьому вході зсувсовий регістр виконує зсув вправо на один розряд. Другий вхід зсувсового регістру використовується для визначення яке значення буде записано на наймолодший біт при зсуві вправо. Вихід зсувсового регістру собою поточне значення внутрішнього регістру.

Зсувсовий регістр використовується для імітування нульової активності на старших розрядах, що відповідає поведінці при великому часі використання систем китичного застосування.

3.2.1.3 Блок обчислення

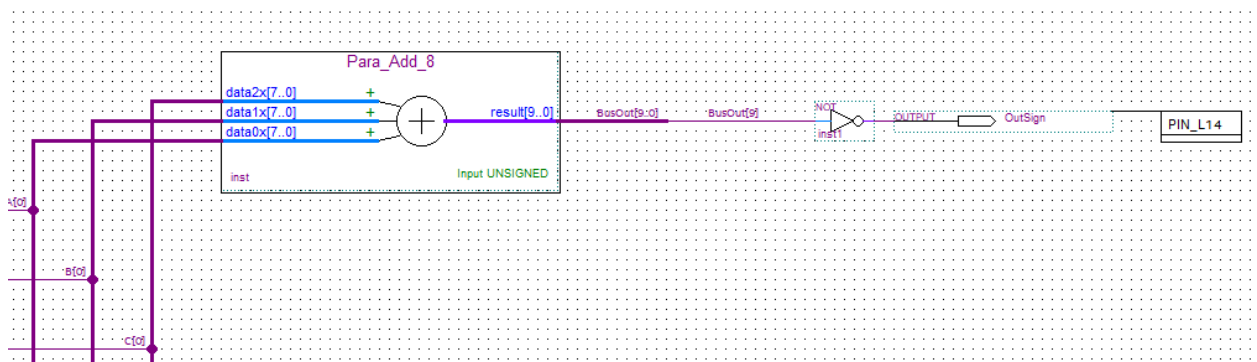


Рисунок 3.5 – Блок обчислення

Блок обчислення складається з наступних елементів:

- Багатооперандний суматор
- Елемент передачі одного з бітів результату на вихід

Багатооперандний суматор має три входи та один вихід. Три входи суматора використовуються для подачі на них вхідних операндів для обчислення. Вихід суматора використовується для отримання результату обчислення суми вхідних операндів.

Елемент передачі одного з бітів операнду на вихід використовується для трансляції стану одного з бітів результату на фізичний вихід мікросхеми.

3.2.2 Опис налаштування головних компонентів схеми

Налаштування основних компонентів схеми розглянуто на прикладі проведення експерименту. Експеримент з виявлення різниці між струмом споживання у нормальному, аварійному та режимі нульової активності при умовах розрядності операндів – 8, кількість розрядів, що відмикаються від синхронізації – 1 на частоті 115 МГц.

На вхід блоку перетворення частоти подається частота 100 МГц з вбудованого кварцового резонатору , що знаходиться на платі стенду. Для отримання з блоку перетворення частоти необхідну частоту, а саме 115 МГц необхідно виконати налаштування усіх виходів блоку наступним чином:

- Коефіцієнт множення – 23
- Коефіцієнт ділення – 20
- Кількість виходів – 3
- Вхідна частота – 100 МГц

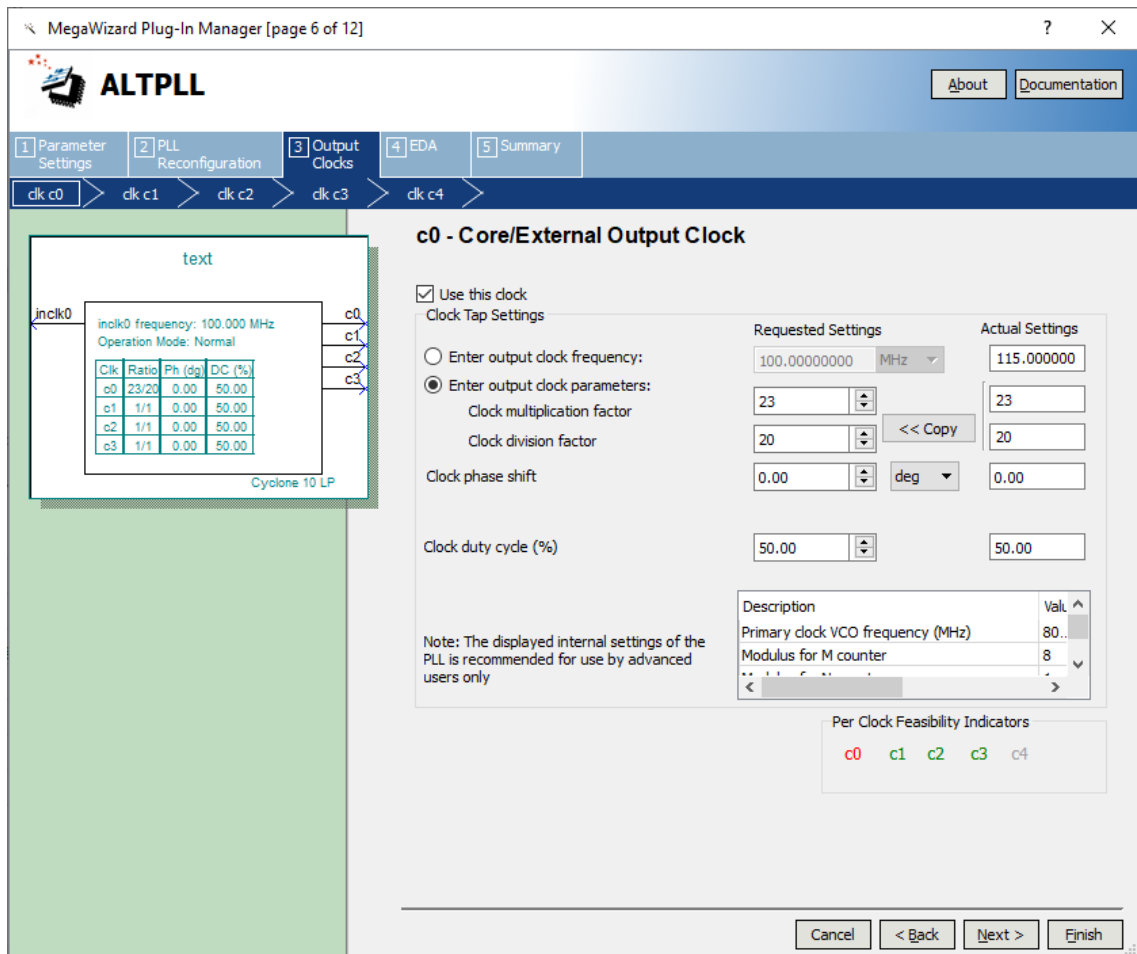


Рисунок 3.6 – Налаштування блоку PLL для проведення експерименту

Для виконання генерції бітів для заповнення молодших розрядів операндів та імітування зміни сигналу на рівну шуму використовується лічильник, який повинен мати 6 бітів, що подалі, за допомогою схеми блоку формування операндів будуть розподілені між молодшими бітами операндів багатооперандного суматора.

Для отримання суматора, що буде мати тактовий всіх, вхід скидання та вихідний регістр обсягом 6 біт використані наступні налаштування:

-Розрядність лічильника – 6 біт

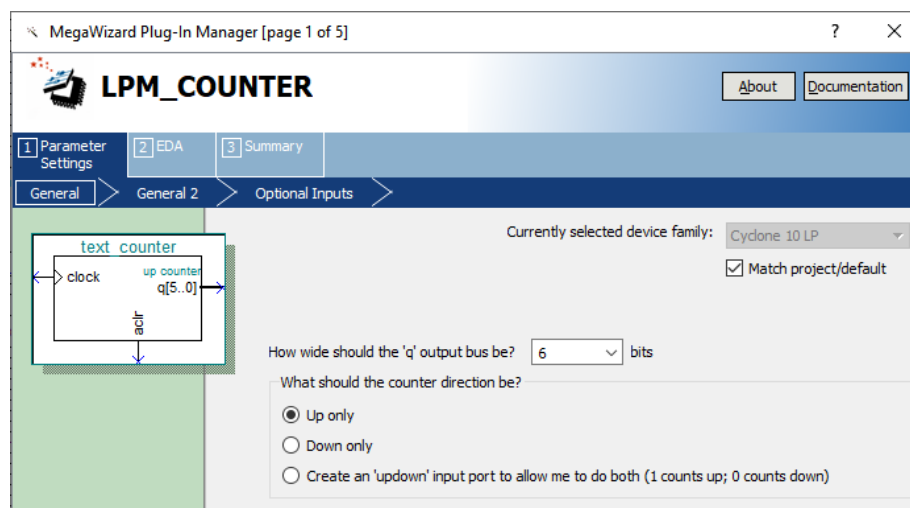


Рисунок 3.7 – Налаштування лічильника для формування молодших бітів

Для формування старших бітів операндів, за допомогою яких буде імітуватися відсутність різких змін сигналу у надстабільних системах необхідно використати зсувний регістр, біти якого будуть заповнюватися нульовими значеннями. Розрядність цього регістру обчислюється шляхом додавання кількості незадіяних у експерименті старших бітів операндів.

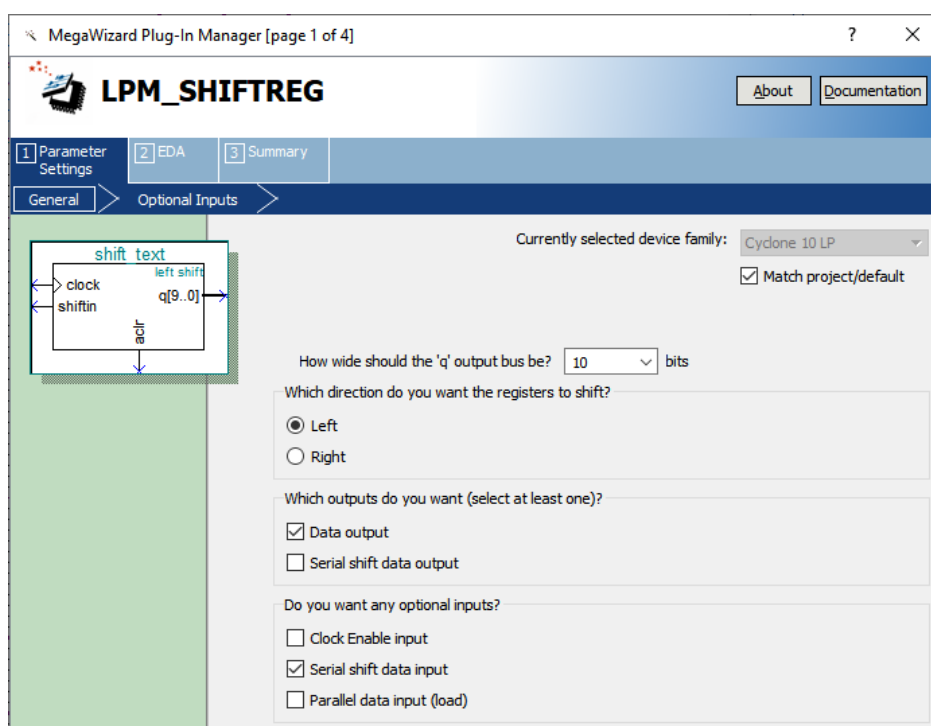


Рисунок 3.8 – Налаштування зсувowego регістру

В даному експерименті маємо необхідність зсувового регістру обсягом 10 біт. Для отримання зсувового регістру обсягом 10 біт виконуємо наступні налаштування:

- Розрядність регістру – 10 біт

Для виконання обчислення над трьома 8 бітними операндами необхідно використати багатооперандний суматор. Для його отримання необхідно виконати наступні налаштування:

- Розрядність операндів – 8 біт
- Кількість операндів – 3
- Розрядність результату – 10 біт

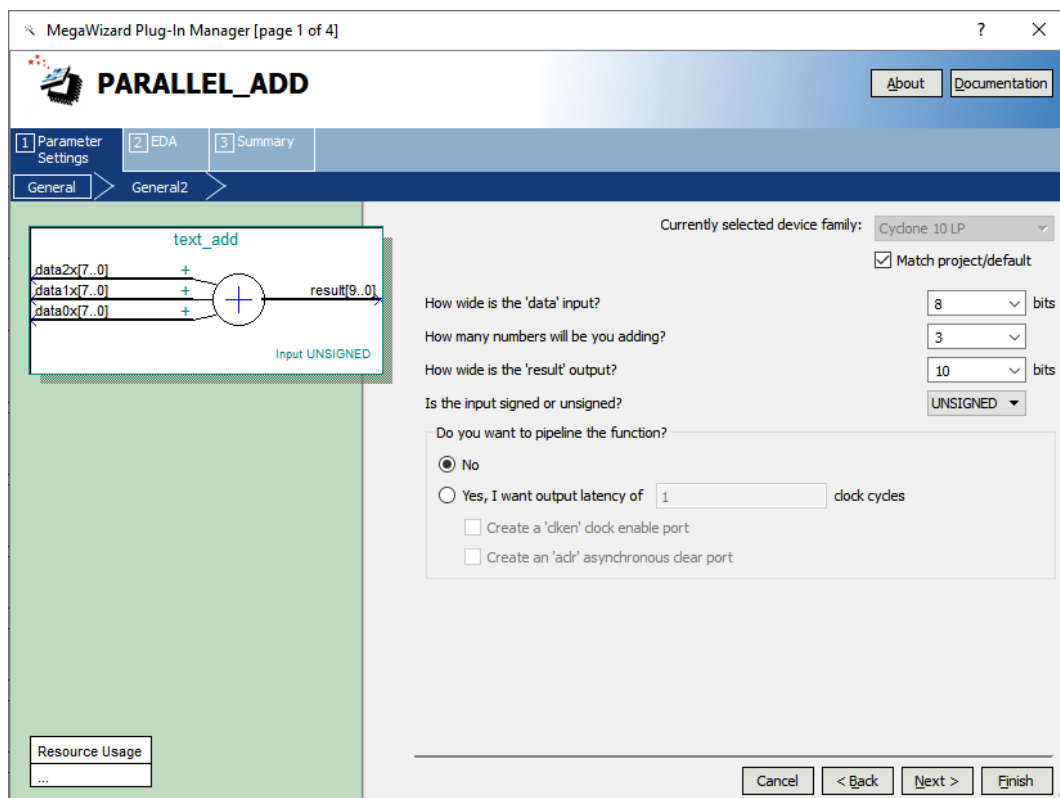


Рисунок 3.9 – Налаштування багатооперандного суматора

Для імітування відключення синхронізації одного біту операнду багатооперандного суматора використовується тригер, до входу синхронізації якого підключено результат логічного множення базової

частоти схеми та зовнішнього сигналу, за допомогою якого виконується відключення синхронізації від триггеру. Вихід триггеру підключено до 4-го розряду. Інформаційний вхід триггеру підключено до останнього біту зсувового регістру.

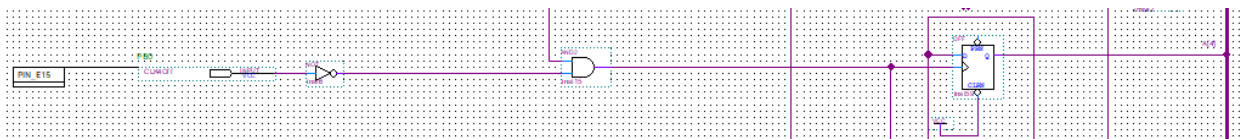


Рисунок 3.10 – Один розряд, що відмикається від синхронізації

3.3 Опис програмного забезпечення для проведення експерименту

Для проведення експериментів з вимірювання споживання струму лабораторним стендом Intel Cyclone 10 LP FPGA було використано три основних застосунки:

- Quartus Prime Lite Edition
- Programmer
- Power Monitor

Quartus Prime Lite Edition – програмне забезпечення, яке призначено для розробки схем на базі FPGA мікросхем, що виготовляє компанія Intel. Це програмне забезпечення використовувалось для створення та модифікації схем, що використовувалися для дослідження енерго-орієнтованої контролепридатності багатооперандного суматора.

Programmer – програмне забезпечення, яке призначення для генерації файлів конфігурації та налаштування мікросхеми FPGA згідно з проектом Quartus Prime Lite Edition. Це програмне забезпечення використовувалось для генерації та завантаження схем у стенд Intel Cyclone 10 LP.

Power Monitor – це програмне забезпечення, що призначене для отримання даних з вбудованого датчику струму споживання струму стендів на базі FPGA мікросхем, що виготовляє компанія Intel. Це програмне забезпечення використовувалось для вимірювання струму стенду Intel Cyclone 10 LP у різних режимах роботи, впливаючи на роботу стенду за допомогою зовнішніх периферійних пристроїв, таких як кнопки та перемикачі.

3.4 Методика проведення експерименту

Для проведення експериментів з дослідження енерго-орієнтованої контролепридатності багатооперандного додавача виконана побудова схем у середовищі Quartus Prime Lite Edition, що будуть завантажуватися за допомогою програмного забезпечення Programmer у стенд Intel Cyclone 10 LP FPGA. Схеми будувалися для зміни декількох ключових параметрів роботи багатооперандного додавача:

- Формування змісту операндів для першої групи розрядів, що представляють зміну вхідних даних
- Формування змісту операндів для другої групи розрядів, що представляють зміну вхідних даних
- Формування змісту операндів для третьої групи розрядів, що представляють зміну вхідних даних
- Відключення від синхронізації розрядів, що представляють зміну вхідних даних
- Відключення від синхронізації усієї схеми

Для ввімкнення або вимкнення формування змісту першої групи розрядів, що представляють зміну вхідних даних використовується

периферія тестового стенду, а саме перемикач, що позначений як DIP0, який підключено до PIN_M16.

Для ввімкнення або вимкнення формування змісту другої групи розрядів, що представляють зміну вхідних даних використовується периферія тестового стенду, а саме перемикач, що позначений як DIP1, який підключено до PIN_A8.

Для ввімкнення або вимкнення формування змісту третьої групи розрядів, що представляють зміну вхідних даних використовується периферія тестового стенду, а саме перемикач, що позначений як DIP2, який підключено до PIN_A9.

Для ввімкнення або вимкнення подачі базової частоти на розряди(тригери), завдяки яким виконується імітація відключення від синхронізації частини розрядів виконується подача сигналу у схему за допомогою периферії тестового стенду, а саме кнопки, що позначено PB0, що підключено до контакту PIN_E15.

Для отримання показань, що будуть відповідні нормальному режиму роботи необхідно ввімкнути синхронізацію усіх частин схеми за допомогою PB0, та ввімкнути що найменше одну групу розрядів за допомогою одного з трьох вимикачів DIP0, DIP1 або DIP2.

Для отримання показань, що будуть відповідні режиму нульової активності необхідно ввімкнути синхронізацію усіх частин схеми за допомогою PB0, та вимкнути усі перемикачі, що відповідають за формування змісту груп розрядів.

Для отримання показань, що відповідають аварійному стану необхідно вимкнути синхронізацію розрядів, за допомогою PB0 та по черзі вимкати DIP0, DIP1, та DIP2 для відстеження току споживання при різній кількості активних інформаційних розрядів.

3.5 Процес отримання результатів експериментального дослідження

Для отримання результатів експериментального дослідження буде проведено декілька експериментів з зміною різних параметрів та режимів роботи схеми та фіксацією енергоспоживання.

Для проведення першого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 8 біт
- Коефіцієнт множення частоти: 23
- Коефіцієнт ділення частоти: 20
- Частота синхронізації 115 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 11 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення першого експерименту отримано наступні результати:

Таблиця 3.1 – результати першого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	1,21 mA	1,21 mA	1,21 mA	1,21 mA	1,21 mA
	1.61 mA	1.61 mA	1.61 mA	1.61 mA	1.61 mA
2	1.21 mA	2.42 mA	2.83 mA	2.83 mA	2.83 mA
	1.61 mA		3.23 mA	3.23 mA	3.23 mA
3	1.21 mA	2.42 mA	2.83 mA	2.83 mA	2.83 mA
	1.61 mA		3.23 mA	3.23 mA	3.23 mA

Для проведення другого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 16 біт
- Коефіцієнт множення частоти: 23
- Коефіцієнт ділення частоти: 20
- Частота синхронізації 115 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 35 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення першого експерименту отримано наступні результати:

Таблиця 3.2 – результати другого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	2,02 mA	2,02 mA	2,02 mA	2,02 mA	2,02 mA
	2,42 mA	2,42 mA	2,42 mA	2,42 mA	2,42 mA
2	2,02 mA	3,23 mA	3,63 mA	3,63 mA	3,63 mA
	2,42 mA		4,04 mA	4,04 mA	4,04 mA
3	2,02 mA	3,23 mA	3,63 mA	3,63 mA	3,63 mA
	2,42 mA		4,04 mA	4,04 mA	4,04 mA

Для проведення третього експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 24 біт
- Коефіцієнт множення частоти: 23
- Коефіцієнт ділення частоти: 20
- Частота синхронізації 115 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 59 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення першого експерименту отримано наступні результати:

Таблиця 3.3 – результати третього експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	2,02 mA	2,02 mA	2,02 mA	2,02 mA	2,02 mA
	2,42 mA	2,42 mA	2,42 mA	2,42 mA	2,42 mA
2	2,02 mA	3,23 mA	3,63 mA	3,63 mA	3,63 mA
	2,42 mA		4,04 mA	4,04 mA	4,04 mA
3	2,02 mA	3,23 mA	3,63 mA	3,63 mA	3,63 mA
	2,42 mA		4,04 mA	4,04 mA	4,04 mA

Для проведення четвертого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 8 біт
- Коефіцієнт множення частоти: 69

- Коефіцієнт ділення частоти: 50
- Частота синхронізації 138 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 11 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення четвертого експерименту отримано наступні результати:

Таблиця 3.4 – результати четвертого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	2,02 mA	2,02 mA	2,02 mA	2,02 mA	2,02 mA
	2,42 mA	2,42 mA	2,42 mA	2,42 mA	2,42 mA
2	2,02 mA	2.83 mA	4,04 mA	4,04 mA	4,04 mA
	2,42 mA		4,44 mA	4,44 mA	4,44 mA
3	2,02 mA	2.83 mA	4,04 mA	4,04 mA	4,04 mA
	2,42 mA		4,44 mA	4,44 mA	4,44 mA

Для проведення п'ятого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 16 біт
- Коефіцієнт множення частоти: 69
- Коефіцієнт ділення частоти: 50
- Частота синхронізації 138 MHz
- Розрядність лічильника : 6 біт

- Розрядність зубового регістру : 35 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення першого експерименту отримано наступні результати:

Таблиця 3.5 – результати п'ятого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	2,02 мА	2,02 мА	2,02 мА	2,02 мА	2,02 мА
	2,42 мА	2,42 мА	2,42 мА	2,42 мА	2,42 мА
2	2,02 мА	2,83 мА	4,44 мА	4,44 мА	4,44 мА
	2,42 мА		4,84 мА	4,84 мА	4,84 мА
3	2,02 мА	2,83 мА	4,44 мА	4,44 мА	4,44 мА
	2,42 мА		4,84 мА	4,84 мА	4,84 мА

Для проведення шостого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 24 біт
- Коефіцієнт множення частоти: 69
- Коефіцієнт ділення частоти: 50
- Частота синхронізації 138 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 59 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення шостого експерименту отримано наступні результати:

Таблиця 3.6 – результати шостого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	2,42 mA	2,42 mA	2,42 mA	2,42 mA	2,42 mA
	2,83 mA	2,83 mA	2,83 mA	2,83 mA	2,83 mA
2	2,42 mA	3,23 mA	5,65 mA	5,65 mA	5,65 mA
	2,83 mA		6,05 mA	6,05 mA	6,05 mA
3	2,42 mA	3,23 mA	5,65 mA	5,65 mA	5,65 mA
	2,83 mA		6,05 mA	6,05 mA	6,05 mA

Для проведення сьомого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 8 біт
- Коефіцієнт множення частоти: 31
- Коефіцієнт ділення частоти: 20
- Частота синхронізації 155 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 11 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення сьомого експерименту отримано наступні результати:

Таблиця 3.7 – результати сьомого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	3,23 mA	3,23 mA	3,23 mA	3,23 mA	3,23 mA
	3,63 mA	3,63 mA	3,63 mA	3,63 mA	3,63 mA
2	3,23 mA	4,04 mA	5,65 mA	5,65 mA	5,65 mA
	3,63 mA		6,05 mA	6,05 mA	6,05 mA
3	3,23 mA	4,04 mA	5,65 mA	5,65 mA	5,65 mA
	3,63 mA		6,05 mA	6,05 mA	6,05 mA

Для проведення восьмого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 16 біт
- Коефіцієнт множення частоти: 31
- Коефіцієнт ділення частоти: 20
- Частота синхронізації 155 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 35 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення восьмого експерименту отримано наступні результати:

Таблиця 3.8 – результати восьмого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	2.83 mA	2.83 mA	2.83 mA	2.83 mA	2.83 mA
	3.23 mA	3.23 mA	3.23 mA	3.23 mA	3.23 mA
2	2.83 mA	3.63 mA	6.46 mA	6.46 mA	6.46 mA
	3.23 mA		6.86 mA	6.86 mA	6.86 mA
3	2.83 mA	3.63 mA	6.46 mA	6.46 mA	6.46 mA
	3.23 mA		6.86 mA	6.86 mA	6.86 mA

Для проведення дев'ятого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 24 біт
- Коефіцієнт множення частоти: 31
- Коефіцієнт ділення частоти: 20
- Частота синхронізації 155 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 59 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення дев'ятого експерименту отримано наступні результати:

Таблиця 3.9 – результати дев'ятого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	3,23 мА	3,23 мА	3,23 мА	3,23 мА	3,23 мА
	3,63 мА	3,63 мА	3,63 мА	3,63 мА	3,63 мА
2	3,23 мА	4,04 мА	6,86 мА	6,86 мА	6,86 мА
	3,63 мА		7,27 мА	7,27 мА	7,27 мА
3	3,23 мА	4,04 мА	6,86 мА	6,86 мА	6,86 мА
	3,63 мА		7,27 мА	7,27 мА	7,27 мА

Для проведення десятого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 8 біт
- Коефіцієнт множення частоти: 9
- Коефіцієнт ділення частоти: 5
- Частота синхронізації 180 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 11 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення десятого експерименту отримано наступні результати:

Таблиця 3.10 – результати десятого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	4,44 mA	4,44 mA	4,44 mA	4,44 mA	4,44 mA
	4,84 mA	4,84 mA	4,84 mA	4,84 mA	4,84 mA
2	4,44 mA	5,25 mA	7,27 mA	7,27 mA	7,27 mA
	4,84 mA		7,67 mA	7,67 mA	7,67 mA
3	4,44 mA	5,25 mA	7,27 mA	7,27 mA	7,27 mA
	4,84 mA		7,67 mA	7,67 mA	7,67 mA

Для проведення одинадцятого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 16 біт
- Коефіцієнт множення частоти: 9
- Коефіцієнт ділення частоти: 5
- Частота синхронізації 180 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 35 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення одинадцятого експерименту отримано наступні результати:

Таблиця 3.11 – результати одинадцятого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	4,44 mA	4,44 mA	4,44 mA	4,44 mA	4,44 mA
	4,84 mA	4,84 mA	4,84 mA	4,84 mA	4,84 mA
2	4,44 mA	7,27 mA	7,67 mA	7,67 mA	7,67 mA
	4,84 mA		8,07 mA	8,07 mA	8,07 mA
3	4,44 mA	7,27 mA	7,67 mA	7,67 mA	7,67 mA
	4,84 mA		8,07 mA	8,07 mA	8,07 mA

Для проведення дванадцятого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 24 біт
- Коефіцієнт множення частоти: 9
- Коефіцієнт ділення частоти: 5
- Частота синхронізації 180 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 59 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення дев'ятого експерименту отримано наступні результати:

Таблиця 3.12 – результати дванадцятого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	4,84 mA	4,84 mA	4,84 mA	4,84 mA	4,84 mA
	5,25 mA	5,25 mA	5,25 mA	5,25 mA	5,25 mA
2	4,84 mA	5,65 mA	8,88 mA	8,88 mA	8,88 mA
	5,25 mA		9,28 mA	9,28 mA	9,28 mA
3	4,84 mA	5,65 mA	8,88 mA	8,88 mA	8,88 mA
	5,25 mA		9,28 mA	9,28 mA	9,28 mA

Для проведення тринадцятого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 8 біт
- Коефіцієнт множення частоти: 21
- Коефіцієнт ділення частоти: 10
- Частота синхронізації 210 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 11 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення тринадцятого експерименту отримано наступні результати:

Таблиця 3.13 – результати тринадцятого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	6,46 мА	6,46 мА	6,46 мА	6,46 мА	6,46 мА
	6,86 мА	6,86 мА	6,86 мА	6,86 мА	6,86 мА
2	6,46 мА	7,27 мА	9,28 мА	9,28 мА	9,28 мА
	6,86 мА		9,69 мА	9,69 мА	9,69 мА
3	6,46 мА	7,27 мА	9,28 мА	9,28 мА	9,28 мА
	6,86 мА		9,69 мА	9,69 мА	9,69 мА

Для проведення чотирнадцятого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 16 біт
- Коефіцієнт множення частоти: 21
- Коефіцієнт ділення частоти: 10
- Частота синхронізації 210 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 35 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення чотирнадцятого експерименту отримано наступні результати:

Таблиця 3.14 – результати чотирнадцятого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	6,46 mA	6,46 mA	6,46 mA	6,46 mA	6,46 mA
	6,86 mA	6,86 mA	6,86 mA	6,86 mA	6,86 mA
2	6,46 mA	9,28 mA	9,69 mA	9,69 mA	9,69 mA
	6,86 mA		10,08 mA	10,08 mA	10,08 mA
3	6,46 mA	9,28 mA	9,69 mA	9,69 mA	9,69 mA
	6,86 mA		10,08 mA	10,08 mA	10,08 mA

Для проведення п'ятнадцятого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 24 біт
- Коефіцієнт множення частоти: 21
- Коефіцієнт ділення частоти: 10
- Частота синхронізації 180 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 59 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення п'ятнадцятого експерименту отримано наступні результати:

Таблиця 3.15 – результати п'ятнадцятого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	6,46 мА	6,46 мА	6,46 мА	6,46 мА	6,46 мА
	6,86 мА	6,86 мА	6,86 мА	6,86 мА	6,86 мА
2	6,46 мА	7,27 мА	11,30 мА	11,30 мА	11,30 мА
	6,86 мА		11,71 мА	11,71 мА	11,71 мА
3	6,46 мА	7,27 мА	11,30 мА	11,30 мА	11,30 мА
	6,86 мА		11,71 мА	11,71 мА	11,71 мА

Для проведення шістнадцятого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 8 біт
- Коефіцієнт множення частоти: 60
- Коефіцієнт ділення частоти: 19
- Частота синхронізації 316 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 11 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення шістнадцятого експерименту отримано наступні результати:

Таблиця 3.16 – результати шістнадцятого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	11,30 mA	11,30 mA	11,30 mA	11,30 mA	11,30 mA
	11,17 mA	11,17 mA	11,17 mA	11,17 mA	11,17 mA
2	11,30 mA	12,11 mA	16,55 mA	16,55 mA	16,55 mA
	11,17 mA		16,95 mA	16,95 mA	16,95 mA
3	11,30 mA	12,11 mA	16,55 mA	16,55 mA	16,55 mA
	11,17 mA		16,95 mA	16,95 mA	16,95 mA

Для проведення сімнадцятого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 16 біт
- Коефіцієнт множення частоти: 60
- Коефіцієнт ділення частоти: 19
- Частота синхронізації 319 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 35 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення сімнадцятого експерименту отримано наступні результати:

Таблиця 3.14 – результати сімнадцятого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	11,71 mA	11,71 mA	11,71 mA	11,71 mA	11,71 mA
	12,11 mA	12,11 mA	12,11 mA	12,11 mA	12,11 mA
2	11,71 mA	12,51 mA	17,36 mA	17,36 mA	17,36 mA
	12,11 mA		17,76 mA	17,76 mA	17,76 mA
3	11,71 mA	12,51 mA	17,36 mA	17,36 mA	17,36 mA
	12,11 mA		17,76 mA	17,76 mA	17,76 mA

Для проведення вісімнадцятого експерименту будуть використані наступні початкові параметри для налаштування тестової схеми:

- Розрядність операндів : 24 біт
- Коефіцієнт множення частоти: 21
- Коефіцієнт ділення частоти: 10
- Частота синхронізації 319 MHz
- Розрядність лічильника : 6 біт
- Розрядність зубового регістру : 59 біт
- Кількість розрядів, що відмикаються від синхронізації: 1, 2, 3

В ході проведення вісімнадцятого експерименту отримано наступні результати:

Таблиця 3.15 – результати вісімнадцятого експерименту

Кількість бітів, що відмикається	Аварійний режим	Режим нульової активності	Режим нормальної роботи		
			Зміна 1-го біту	Зміна 2-ох бітів	Зміна 3-ох бітів
1	12,51 mA	12,51 mA	12,51 mA	12,51 mA	12,51 mA
	12,92 mA	12,92 mA	12,92 mA	12,92 mA	12,92 mA
2	12,51 mA	13,32 mA	19,78 mA	19,78 mA	19,78 mA
	12,92 mA		20,18 mA	20,18 mA	20,18 mA
3	12,51 mA	13,32 mA	19,78 mA	19,78 mA	19,78 mA
	12,92 mA		20,18 mA	20,18 mA	20,18 mA

3.6 Аналіз отриманих результатів дослідження

Після проведення 18-ти експериментів були отримані дані про енергоспоживання тестової схеми при різних режимах роботи.

Згідно з отриманими результатами при відмиканні одного біту на усіх перевірених частотах та довжинах операндів енергоспоживання суттєво не змінюється. Виходячи з чого можна зробити висновок, що визначити відімкнення одного триггеру від синхронізації зафіксувати неможливо при прийнятих умовах.

При відмиканні двох і більше бітів спостерігається суттєве зниження енергоспоживання схеми при заданих умовах, для більш глибокого аналізу необхідно побудувати графіки залежності енергоспоживання від режиму роботи та базової частоти.

Згідно з результатами експериментів для операндів розрядністю 8 біт та кількістю розрядів, що відмикаються від синхронізації, отримано наступний графік залежності енергоспоживання від режиму та частоти.

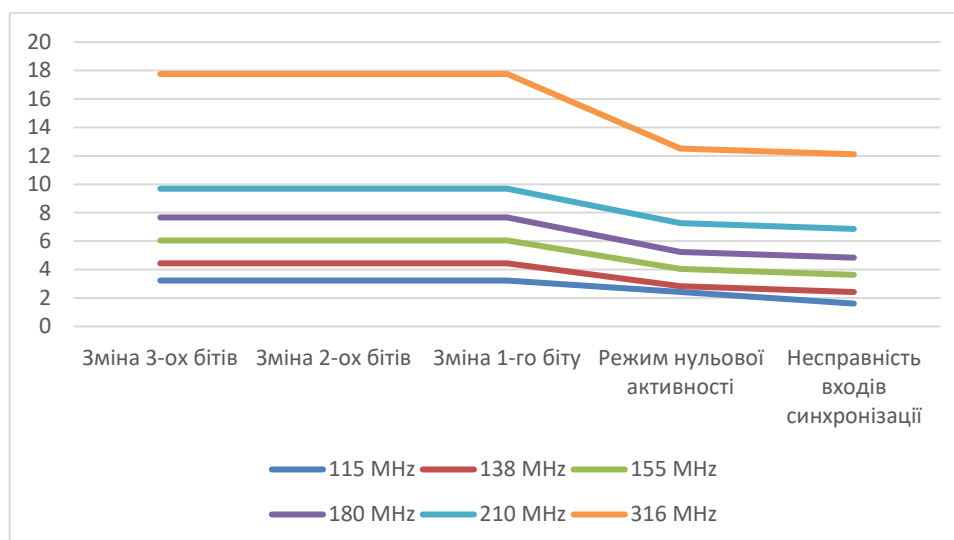


Рисунок 3.11 – залежність енергоспоживання від режиму роботи та частоти для розрядності операнду – 8 біт

Згідно з отриманим графіком можна зробити висновок, що енергоспоживання при відмиканні двох і більше бітів значно знижується з переходом у режим нульової активності, та ще більше знижується при переході в аварійний режим, що свідчить, про можливість відстежування відключення від синхронізації двох та більше бітів при розрядності операндів 8. Згідно з отриманими даними в залежності від частоти енергоспоживання схему у всіх режимах збільшується, та чим більше частота, тим більше і енергоспоживання. Різниця енергоспоживання при нормальному та аварійному режимах також стає більше з зростанням частоти. При зміні сигналу на прийнятому рівні в декілька молодших бітів впливу на енергоспоживання не виявлено.

Згідно з результатами експериментів для операндів розрядністю 16 біт та кількістю розрядів, що відмикаються від синхронізації, отримано наступний графік залежності енергоспоживання від режиму та частоти.

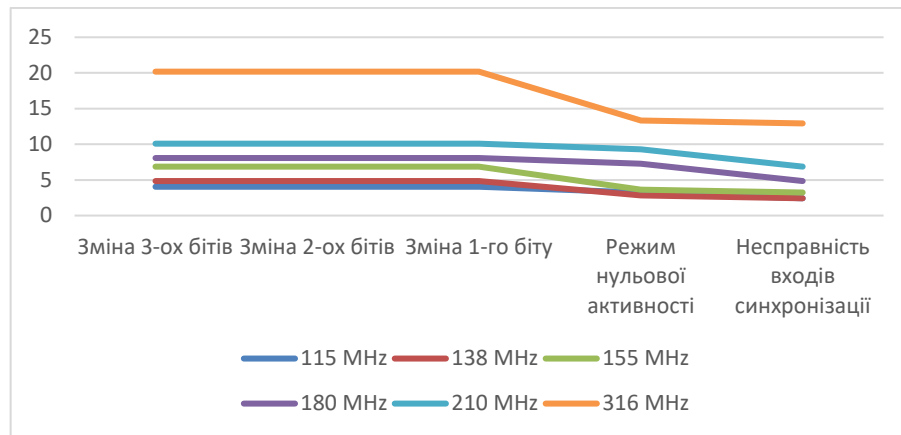


Рисунок 3.12 – залежність енергоспоживання від режиму роботи та частоти для розрядності операнду – 16 біт

Виходячи з отриманого графіки можна зробити висновок, що енергоспоживання всієї схеми зростає з збільшенням розрядності операндів, та частотах 316,210,180,138 MHz різниця між нормальним та аварійним режимом більше ніж при розрядності 8 біт.

Згідно з результатами експериментів для операндів розрядністю 24 біта та кількістю розрядів, що відмикаються від синхронізації, отримано наступний графік залежності енергоспоживання від режиму та частоти.

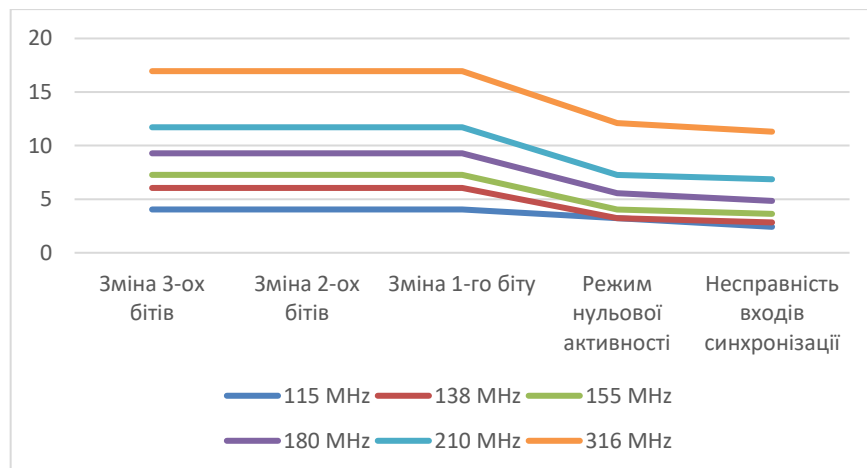


Рисунок 3.13 – залежність енергоспоживання від режиму роботи та частоти для розрядності операнду – 24 біта

Згідно з отриманим графіком можна зробити висновок, що різниця енергоспоживання при відмиканні двох і більше бітів при розрядності одного операнда 24 біти значно більша ніж при розрядності 16 та 8 біт на усіх частотах, що перевіряються, окрім 115 МГц. На частоті 115 МГц різниця між розрядностями 8, 16 та 24 біта найменша.

3.7 Висновки

Згідно з отриманих результатів можна зробити загальний висновок, що стабільне відстеження відімкнення можливо при відімкненні від синхронізації двох і більше тригерів. При зростанні частоти або розрядності загальне енергоспоживання та різниця між нормальним та аварійним режимом не змінюються. Зміна молодших бітів операндів не призводить до значної зміни енергоспоживання, що надає можливості чітко відстежувати тільки відімкнення тригерів від синхронізації.

ЗАГАЛЬНІ ВИСНОВКИ

В магістерській кваліфікаційній роботі дослідженна проблема необхідності підвищення контролепридатності комп'ютерних систем критичного застосування, через обмеженість можливостей логічної контролепридатності. Були проведений аналіз апаратної платформи FPGA, що використовується при побудові комп'ютерних систем критичного застосування.

Розроблений підхід до диверсифікації котролепридатності на логічну та енерго-орієнтовану, що дає можливість уникнути недоліків логічної контролеридатності завдяки можливості відстеження стану ділянок схеми, що не задіяні безпосередньо в поточний момент обчислень, що дає можливість виявити несправність до її впливу на результат роботи пристрою.

Були розроблені умови та принципи проведення експерименту, проведені експерименти та отримані дані , що дають можливість проаналізувати ступінь визначення наявності помилки в незадіяних ділянках схеми.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Девід М. Херріс і Сара Л. Херріс Цифровая схемотехника и архитектура компьютера. Издательство “Morgan Kaufman”, 2013 – 1627 с
2. Бабак В. П., Корченко А. Г., Тимошенко Н. П., Филоненко С. Ф. VHDL: Справочное пособие по основам языка. Издательский дом «Додэка-XXI». — 224 с.
3. Грушвицкий Р.И., Мурсаев А.Х., Угрюмов Е.П. Проектирование систем на микросхемах программируемой логики. Издательство “БВХ-Петербург”, 2002 – 608 с
4. Sarkany, Z.; Rencz, M. Design considerations to enhance thermal testability. Proceedings of the IEEE 14th Electronics Packaging Technology Conference (EPTC), Singapore, Singapore, December 2012. DOI: <http://dx.doi.org/10.1109/EPTC.2012.6507068>
5. Najem, M.; Benoit, P; Bruguier, F; Sassatelli, G; Torres, L. Method for Dynamic Power Monitoring on FPGAs. Conference Digest of 24th International Conference on Field Programmable Logic and Applications, Munich, Germany, September 1-5 2014.
6. Aminbakhsh, S.; Gunduz, M.; Sonmez, R. Safety risk assessment using analytic hierarchy process (AHP) during planning and budgeting of construction projects. *Journal of Safety Research* **2013**, Volume 46, 99–105. DOI: <http://dx.doi.org/10.1016/j.jsr.2013.05.003C>
7. Marvin Rausand. Risk Assessment. Common Cause Failures. Available online: <https://www.ntnu.edu/documents/624876/1277591044/ccf.pdf/f435f724-469d-4492-860a-66eca10e6bd2> (accessed on 20.01.20).
8. Kharchenko, V.; Sachenko, A.; Kochan, V.; Fesenko H. Reliability and Survivability Models of Integrated Drone-Based Systems for Post

- Emergency Monitoring of NPPs. Proceeding of the IEEE International Conference on Information and Digital Technologies, Rzeszov, Poland, 2016; 127 – 132 c. DOI: <http://dx.doi.org/10.1109/DT.2016.7557161>
9. IEC61508. *Functional Safety of Electrical, Electronic, Programmable Electronic Safety Related Systems. General requirements*; Rep. IEC: Geneva, Switzerland, 2010.
 10. IEC61513. *Nuclear Power Plants: Instrumentation & Control Systems Important to Safety. General Requirements for Systems*; Rep. IEC: Geneva, 2001.
 11. Park, Y.; Cho, Y.H.; Lee, K.; Jung, H.; Kim, H.; Kwon, S.; Park, H. Development of an FPGA-based Online Condition Monitoring System for Railway Catenary Application. Proceedings of the 8th World Congress on Railway Research, Seoul, Korea, May 2008.
 12. Smith, D; Simpson, K. *The Safety Critical Systems Handbook*, 4th ed.; Butterworth-Heinemann, 2016.
 13. Edstrom, J.; Tilevich, E. Reusable and Extensible Fault Tolerance for RESTful Applications. Proceedings of the 11th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, Liverpool, UK, 2012. DOI: <http://dx.doi.org/10.1109/TrustCom.2012.244>
 14. Jhavar, R.; Piuri, V. Fault Tolerance and Resilience in Cloud Computing Environments. In *Computer and Information Security Handbook*; 2017; pp. 165-181.