

Ім'я користувача:
приховано налаштуваннями конфіденційності

ID перевірки:
1015644449

Дата перевірки:
19.06.2023 13:53:46 EEST

Тип перевірки:
Doc vs Library

Дата звіту:
19.06.2023 13:57:35 EEST

ID користувача:
100011372

Назва документа: ОК-42 Павло Сичов 2

Кількість сторінок: 27 Кількість слів: 4327 Кількість символів: 32412 Розмір файлу: 1.34 MB ID файлу: 1015290494

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

1.2% Схожість

Найбільша схожість: 0.6% з джерелом з Бібліотеки (ID файлу: 1000578359)

Пошук збігів з Інтернетом не проводився

1.2% Джерела з Бібліотеки

33

Сторінка 29

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0% Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи

35

Підозріле форматування

6
сторінок

ВСТУП

Програмовані логічні інтегральні схеми (ПЛІС, англ. programmable logic device, PLD) займають важливе місце на ринку цифрових інтегральних схем. Їх важливою відмінністю є те, що вони дозволяють створювати цифрові пристрої за індивідуальним проектом в короткі терміни завдяки модульній структурі цифрових компонентів та наявності схем комутації і конфігураційної пам'яті, яка на основі конфігураційного файлу створює зв'язки між цифровими комірками відповідно до проекту. Високу ефективність проектування цифрових схем на ПЛІС забезпечують комп'ютерні програмні засоби, які дозволяють створювати проекти як традиційним графічним методом з використанням бібліотечних компонентів так і з використанням мов опису апаратури, таких як Verilog, VHDL, AHDL та ін. Такі програмні засоби забезпечують верифікацію проекту засобами моделювання роботи цифрових схем, а також дозволяють легко програмувати мікросхеми ПЛІС завдяки їх функції внутрішньо системного програмування, що дозволяє перевірити роботу спроектованої схеми на фізичному рівні.

Зважаючи на зростаючий інтерес до ПЛІС та інтенсивний їх розвиток і широке застосування в комп'ютерній техніці, засобах телекомунікацій, автомобільній галузі і ін. актуальним є вивчення і здобуття практичних навиків роботи з такими пристроями у нашому коледжі. Вивчення ПЛІС у нашому коледжі в рамках навчальних дисциплін Комп'ютерна схемотехніка і Обчислювальна техніка та мікропроцесори було започатковано з 2000-х років.

Даний дипломний проєкт присвячений створенню інтерактивного автоматизованого робочого місця для проектування цифрових пристроїв на ПЛІС, що дозволить вдосконалити процес вивчення ПЛІС у нашому коледжі.

У першому розділі пояснювальної записки до дипломного проєкту приведено аналіз основних типів ПЛІС, а також наведено приклади ПЛІС фірми Altera, на використання яких орієнтоване робоче місце, що створюється в рамках даного проєкту.

У другому розділі приведена характеристика програмного середовища Quartus II для проектування пристроїв на ПЛІС фірми Altera, яке використовується у даному проєкті.

Третій розділ присвячений реалізації інтерактивного робочого місця для проектування пристроїв на ПЛІС фірми Altera з використанням персонального комп'ютера, на якому встановлене середовище **проектування Quartus II**, а також лабораторного стенда, що містить ПЛІС типу FPGA - EP1K10TC100-3 фірми Altera, програмувального кабеля USB Blaster Altera і USB осцилографа, що дає можливість верифікувати роботу проєктованої цифрової схеми на фізичному рівні.

Четвертий розділ присвячений техніко-економічному обґрунтуванню проєкту, а п'ятий розділ питанням охорони праці та безпеки життєдіяльності.

1 ОГЛЯД ПРОГРАМОВАНИХ ЛОГІЧНИХ ІНТЕГРАЛЬНИХ СХЕМ

1.1 Типи ПЛІС

Серед номенклатури ПЛІС найбільшого поширення набули мікросхеми, що виготовлені за технологіями FPGA та CPLD.

Абревіатура FPGA означає Field Programmable Gate Array – програмована користувачем вентильна матриця.

Абревіатура CPLD означає Complex Programmable Logic Device – комплексний програмований логічний пристрій.

ПЛІС FPGA і CPLD різняться внутрішньою архітектурою і способом конфігурування. Перші для зберігання конфігураційного файлу використовують вбудований оперативний запам'ятовуючий пристрій типу SRAM, що потребує використання зовнішньої енергонезалежної конфігураційної пам'яті. Другі для конфігурування використовують вбудовану енергонезалежну пам'ять типу EEPROM, яка дозволяє зберігати конфігурацію пристрою при вимкненні живлення але, з іншого боку, технологія EEPROM не може реалізовуватись на тонких напівпровідникових структурах. Це обмежує технологічні можливості ПЛІС типу CPLD – вони не можуть реалізовуватись у мікросхемах з високим ступенем інтеграції – в той час, як для пристроїв FPGA таких обмежень немає. Таким чином ПЛІС типу FPGA це, в першу чергу, пристрої з високим ступенем інтеграції, тому їхня внутрішня архітектура в першу чергу орієнтована на реалізацію складних алгоритмів паралельного оброблення сигналів. Ступінь інтеграції сучасних пристроїв FPGA сягає сотень тисяч вентилів, в той час як ступінь інтеграції пристроїв CPLD не перевищує 10 тисяч вентилів.

Ефективність застосування ПЛІС зумовлюють наступні фактори:

- Широка номенклатура ПЛІС різного ступеня інтеграції – від простих до складних з числом вентилів на кристал понад мільйон, що виготовлені за найсучаснішими технологіями;

- Можливість реалізації у їх структурі запам'ятовуючих пристроїв і обчислювальних структур;
- Широкий діапазон робочих частот, який сягає сотень МГц;
- Низький рівень споживання;
- Можливість створювати цифрові пристрої за індивідуальним проектом в короткі терміни завдяки модульній структурі цифрових компонентів та наявності схем комутації і конфігураційної пам'яті, яка на основі конфігураційного файлу створює зв'язки між цифровими комірками відповідно до проекту;
- Наявність комп'ютерних програмних засобів, які дозволяють створювати проекти як традиційним графічним методом з використанням бібліотечних компонентів так і з використанням мов опису апаратури, таких як Verilog, VHDL, AHDL та ін.;
- Можливість верифікації проекту засобами моделювання роботи цифрових схем, яку підтримує програмне забезпечення;
- Можливість внутрішньо системного програмування, що дозволяє перевірити роботу спроектованої схеми на фізичному рівні.

Проектуванням та виробництвом ПЛІС у даний час займається достатньо багато компаній, проте лідерами у цій галузі є фірма Intel, яка стала правонаступницею розробок фірми **Altera**, а також фірма Xilinx. У нашому проекті використовуються продукти фірми Altera.

1.2 ПЛІС на основі вбудованої конфігураційної пам'яті SRAM

ПЛІС на основі вбудованої конфігураційної пам'яті SRAM розглянемо на прикладі ПЛІС типу FPGA фірми Altera сімейства ACEX 1K. Мікросхеми ПЛІС цієї серії мають наступні характеристики:

- Енергозалежна вбудована конфігураційна пам'ять типу SRAM;
- Внутрішньо-системне програмування чи перепрограмування;
- Число циклів перепрограмування – необмежене;

- Число вентилів на кристалі: від 10 тис. до 100 тис.;
- Число логічних елементів (LE) – від 576 до 4992;
- Кількість блоків пам'яті (EAB) – від 3 до 12;
- Об'єм пам'яті – від 12288 біт до 49152 біт;
- Число виводів корпусів мікросхем – від 100 до 208 (484 для корпусу BGA);
- Робочий діапазон частот – до 285 МГц;
- Напруга живлення ядра – 2,5 В;
- Входи/виходи завдяки мультирівневому живленню підтримують сумісність з цифровими схемами, що живляться від напруги 5,0 В, 3,3 В або 2,5 В.

Пристрої ACEX 1K містять вбудовану область оперативної пам'яті (RAM) для їх конфігурування, тому вимагають зовнішнього конфігураційного пристрою. Це можуть бути серійні конфігураційні пристрої Altera такі як EPC16, EPC2, EPC1, і EPC1441, які завантажують пристрої ACEX 1K через послідовний потік даних при вмиканні живлення.

Конфігураційні дані можуть також бути завантажені з персонального комп'ютера через кабелі завантаження MasterBlaster TM, ByteBlasterMV TM або BitBlaster TM фірми Altera. Цей режим завантаження використовується в процесі проектування і налагодження схеми на основі ПЛІС.

Крім цього пристрої ACEX 1K містять інтерфейс, який дозволяє мікропроцесору завантажити пристрої ACEX 1K у послідовному або в паралельному режимах, синхронно або асинхронно. Інтерфейс також дозволяє мікропроцесору провести повторне завантаження у випадку виникнення помилки. Оскільки повторна конфігурація вимагає менш ніж 40 мс, зміни можуть бути виконані в реальному масштабі часу. Такий режим використовують у випадках, коли система, в якій використовуються пристрої на основі ПЛІС ACEX 1K, містить мікропроцесор.

Структурна схема ПЛІС серії ACEX 1K представлена на рис.1.1.

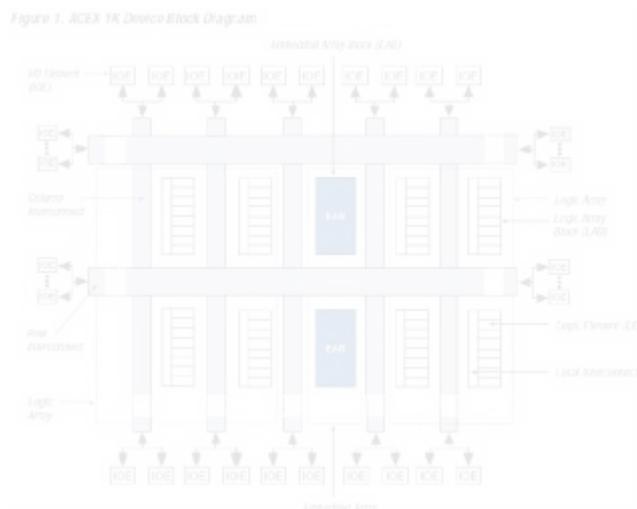


Рисунок 1.1 – Структурна схема ПЛІС серії ACEX 1K

Основу структури ПЛІС складають масиви логіки (Logic Array), які складаються з блоків логіки (LAB), кожен з яких містить вісім логічних елементів (LE). Блоки логіки в межах масиву логіки розташовуються рядами і стовпцями. Для організації зв'язків між блоками логіки використовуються швидкісні лінії (FastTrack), одні з яких розташовуються горизонтально (Row Interconnect), а інші вертикально (Column Interconnect). Лінії зв'язків на своїх кінцях сполучуються з елементами введення/виведення (I/O Element (IOE)), кожен з яких сполучений з нішкою корпусу і утворює зовнішній вхід/вихід.

Входи логічних елементів LE в межах кожного блоку логіки LAB сполучуються через локальні лінії зв'язків (Local Interconnect).

Схеми зв'язків між блоками та в межах кожного блоку формуються в процесі компілювання проекту у вигляді конфігураційного файлу, який повинен заноситися в конфігураційну оперативну пам'ять ПЛІС.

ПЛІС серії ACEX 1K містять також вбудовані блоки елементів пам'яті (Embedded Array Block (EAB)), які можуть використовуватися самостійно для організації оперативних запам'ятовуючих пристроїв, або в парі з логічними

блоками для реалізації складних логічних функцій. Блоки елементів пам'яті ЕАВ розташовуються по одному в кожному з рядків масиву логіки.

Кожен блок ЕАВ містить групу елементів оперативної пам'яті, а також вхідний і вихідний регістри. Така структура блоку дозволяє використовувати його не тільки для реалізації запам'ятовуючого пристрою, а й для реалізації мегафункцій (megafunctions). Це дає можливість реалізовувати, наприклад, функції множення, на яких базується робота цифрових фільтрів. При реалізації мегафункцій блок ЕАВ працює як великий LUT-процесор (LUT – Look-up Table), який працює за принципом табличних перетворень.

Структурна схема блоку логіки (LAB) представлена на рис.1.2.

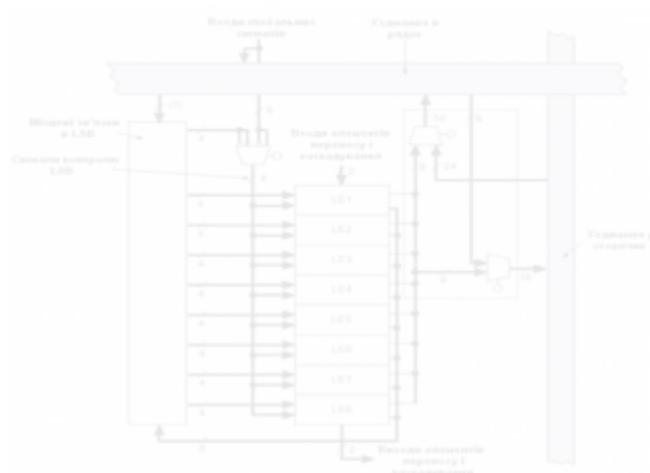


Рисунок 1.2 – Структурна схема блоку логіки (LAB)

Блок логіки LAB складається з восьми логічних елементів LE, які зв'язані між собою через схеми переносів і ланцюги каскадування. Сигнальні входи логічних елементів – по чотири на кожний – сполучені з локальною лінією зв'язків блоку логіки. Кожний логічний елемент крім цього має чотири входи, які сполучені з програмованими глобальними лініями керування, дві з яких можуть використовуватися для синхронізації, а дві інші для скидання і попереднього встановлення.

Структурна схема логічного елемента LE представлена на рис.1.3.

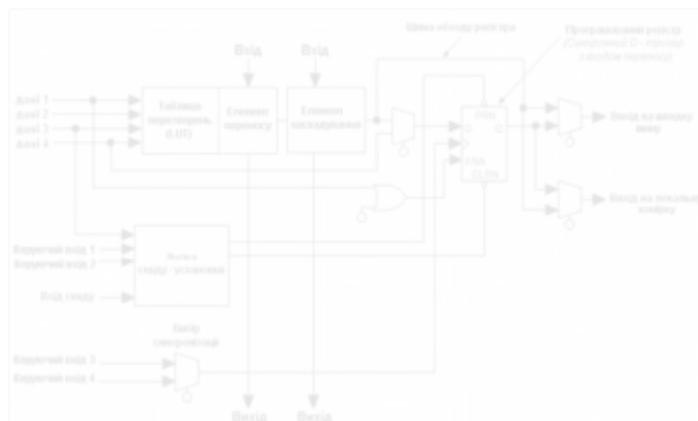


Рисунок 1.3 – Структурна схема логічного елемента LE

Логічний елемент LE – це найменша одиниця логіки в архітектурі ПЛІС ACEX 1K, яка має компактний розмір, що забезпечує ефективну компоновку логіки. Кожний такий елемент LE містить 4-входову LUT-таблицю, яка є генератором функції і може швидко обчислити будь-яку функцію чотирьох змінних. Крім того, кожен логічний елемент LE містить програмований синхронний тригер, ланцюг переносу, і ланцюг каскадування.

Програмований тригер в логічному елементі LE може бути налаштований для роботи в режимах D, T, JK або в режимі обходу. Установочні входи та виходи синхронізації можуть керуватися глобальними сигналами, що забезпечує високу якість синхронізації.

Архітектура ПЛІС ACEX 1K забезпечує два види швидкісної передачі даних між сусідніми елементами LE: через ланцюги переносу і ланцюги каскадування.

Вісім логічних елементів LEs, що утворюють один блок логіки LAB, можуть використовуватися для створення середнього розміру функціональних пристроїв, наприклад, 8-розрядного лічильника, дешифратора адреси і ін.

В табл.1.1 представлені затрати ресурсів ПЛІС серії ACEX 1K при реалізації різних функціональних вузлів.

Таблиця 1.1 – Приклади реалізації цифрових схем на ПЛІС серії ACEX 1K

Цифрова схема	Ресурси, що використовуються		Швидкодія			
			Градація швидкості			Одиниці
	LEs	EABs	-1	-2	-3	
16-розрядний лічильник	16	0	285	232	185	МГц
Мультиплексор 1x16	10	0	3,5	4,5	6,6	нс
256x16 RAM швидкість циклу читання	0	1	278	196	143	МГц

1.3 ПЛІС на основі вбудованої конфігураційної пам'яті EEPROM

ПЛІС на основі вбудованої конфігураційної пам'яті EEPROM розглянемо на прикладі CPLD фірми Altera серії MAX3000a. Мікросхеми ПЛІС цієї серії мають наступні характеристики:

- Енергонезалежна вбудована конфігураційна пам'ять типу EEPROM;
- Внутрішньо-системне програмування чи перепрограмування;
- Число циклів перепрограмування – до 100;
- Число вентилів на кристалі: від 600 до 10 тис.;
- Число блоків логіки (LAB) – від 2 до 32;
- Число макрокомірок (Macrocells) – від 32 до 512;
- Число виводів корпусів мікросхем – від 32 до 208;
- Робочий діапазон частот – до 227 МГц;
- Напруга живлення ядра – 3,3 В;
- Входи/виходи завдяки мультирівневому живленню підтримують сумісність з цифровими схемами, що живляться від напруги 5,0 В, 3,3 В або 2,5 В.

На рис.1.4 представлений зовнішній вигляд корпусу мікросхеми ПЛІС з числом виводів 208.



Рисунок 1.4 – Зовнішній вигляд корпусу мікросхеми ПЛІС з числом виводів 208

Структурна схема ПЛІС серії MAX3000a представлена на рис.1.5.

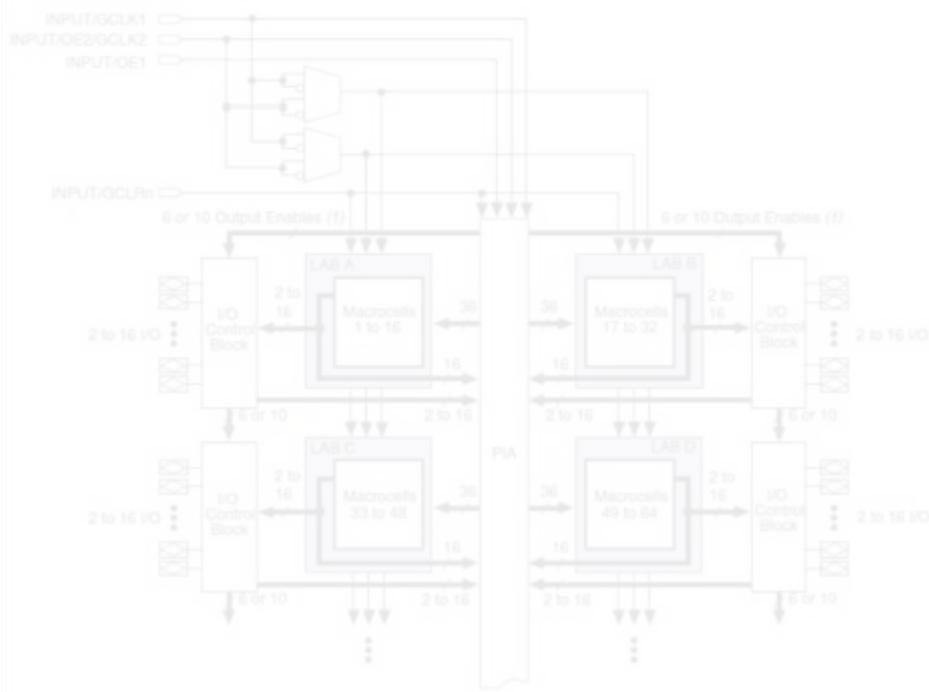


Рисунок 1.5 – Структурна схема ПЛІС серії MAX3000a

Архітектура ПЛІС MAX 3000 **А**включає наступні компоненти:

- Блоки логіки LAB (Logic Array Blocks);
- Макрокомірки (Macrocells);
- Засоби розширення Expander product terms (shareable and parallel);
- Програмований масив зв'язків PIA (Programmable interconnect array);
- Блоки контролю входів/виходів I/O.

Архітектура MAX 3000A включає також **чотири** спеціальні входи, які можуть бути використані як загальні входи або як високошвидкісні глобальні сигнали управління (синхронізації, скидання та два сигнали дозволу виведення) для кожної макрокомірки та ніжки входу/виходу.

Основу архітектури ПЛІС складають блоки логіки LAB (Logic Array Blocks), кожний з яких складається з 16-ти макрокомірок (Macrocells). Блоки логіки сполучаються між собою через програмований масив зв'язків PIA – глобальну шину, яка об'єднує всі ніжки спеціальних входів, ніжки входів/виходів та макрокомірки. На кожний блок логіки поступає 36 сигнальних ліній з шини PIA, які використовуються в якості логічних входів та глобальні сигнали керування.

Структурна схема макрокомірки представлена на рис.1.6.

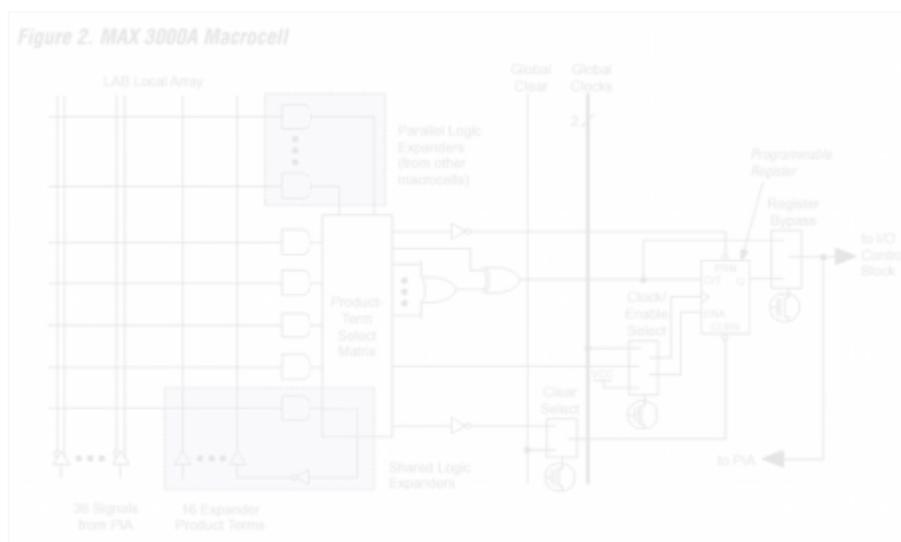


Рисунок 1.6 – Структурна схема макрокомірки ПЛІС серії MAX3000a

Кожна макрокомірка має програмований-AND/фіксований-OR масив (LAB Local Array) і конфігураційний регістр (Programmable Register) з незалежним програмуванням функцій синхронізації, дозволу синхронізації, скидання і початкового встановлення. Для побудови комплексних логічних функцій кожна макрокомірка може бути доповнена загальним розширювачем (Shared Logic Expander) і високошвидкісним паралельним розширювачем (Parallel Logic Expander) зв'язків для забезпечення до 32 зв'язків на макрокомірці.

Як бачимо з наведених прикладів ПЛІС фірми Fltera, структури ПЛІС мають гнучку структуру і широкі можливості для створення на їх основі цифрових пристроїв різного ступеню складності.

2 РОЗРОБЛЕННЯ СТРУКТУРНОЇ СХЕМИ ТА ВИБІР КОМПОНЕНТІВ ІНТЕРАКТИВНОГО РОБОЧОГО МІСЦЯ ДЛЯ ПРОЄКТУВАННЯ ПРИСТРОЇВ НА ПЛІС

2.1 Структурна схема робочого місця

Структурна схема інтерактивного робочого місця для проєктування пристроїв на ПЛІС представлена на рисунку 2.1.

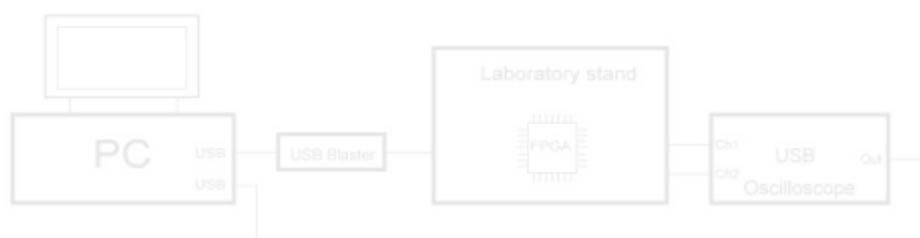


Рисунок 2.1 – Структурна схема інтерактивного робочого місця для проєктування пристроїв на ПЛІС

До складу робочого місця входять:

- Персональний комп'ютер, на який встановлюється програмний пакет QuartusII фірми Altera для створення проєктів на ПЛІС, і програмне забезпечення USB бластера і USB осцилографа;
- Лабораторний стенд, на друкованій платі якого міститься програмована логічна інтегральна схема типу FPGA - EP1K10TC100-3 фірми Altera;
- USB бластер, який через USB інтерфейс сполучений з персональним комп'ютером, а його вихід через зовнішній роз'єм лабораторного стенду з'єднаний з програмувальним входом ПЛІС, що є у стенді;

- Двоканальний USB осцилограф, входи якого підключені до виходів ПЛІС через відповідні роз'єми стенду, а вихід сполучений з персональним комп'ютером через USB інтерфейс.

Проектування цифрового пристрою на основі ПЛІС забезпечується середовищем Quartus II. Результатом проектування є конфігураційний файл, який з середовища Quartus II через USB бластер переписується у конфігураційну пам'ять ПЛІС. Формування вхідних впливів для проектованої схеми забезпечує генератор прямокутних імпульсів частотою 10 МГц, який розташований поруч з мікросхемою ПЛІС на друкованій платі стенду. Окрім проектованої схеми у проєкт включаються також допоміжні схеми, які полегшують дослідження схеми проєкту на фізичному рівні за допомогою USB осцилографа. Це подільники частоти з формувачами імпульсів, які дозволяють змінювати частоту вхідних імпульсів для досліджуваних схем, а також комутатори цифрових сигналів, які полегшують доступ входів осцилографа до контрольованих точок у схемі проектованого пристрою.

2.2 Персональний комп'ютер і середовище QUARTUS II

Вимоги до характеристик персонального комп'ютера в основному визначаються вимогами програмного забезпечення Quartus II. Основними з цих вимог є: обсяг оперативної пам'яті не менше 4 Гбайт, ємність жорсткого диску не менше 100 Гбайт, операційна система Windows 10. Для реалізації дипломного проєкту я вибрав версію Quartus II 9.0 webedition, яка підтримує мікросхему EP1K10TC100-3, що використана у стенді, і є у вільному доступі.

Для реалізації дипломного проєкту я обрав персональний комп'ютер **TechnicProz** наступними характеристиками:

- Процесор Celeron G3930 2,9Ghz;
- Обсяг оперативної пам'яті 4096M;
- Ємність жорсткого диску 500Gb;
- Форм фактор ATX400W;

- Монітор LG 19M37A;
- Операційна система Windows 10.

Характеристики персонального комп'ютера повністю відповідають вимогам середовища проектування Quartus II.

Розглянемо особливості середовища Quartus II.

Середовище проектування Quartus II було розроблено фірмою Altera для проектування пристроїв на ПЛІС цієї фірми. З 2018 року фірма Altera увійшла до складу відомої фірми Intel, яка, завдяки високим технологіям, сприяла покращенню технічних характеристик ПЛІС.

На рисунку 2.2 представлений логотип середовища Quartus II фірми Altera.



Рисунок 2.2 – Логотип середовища Quartus II фірми Altera

Середовище Quartus II містить наступні сервіси:

- Менеджер проекту: полегшує створення файлів проекту;
- Графічний редактор: дозволяє створювати проект графічним способом;
- Текстовий редактор: дозволяє створювати проект з використанням мови опису апаратури, наприклад, verilog HDL, VHDL;

- Компонувальник - дозволяє редагувати розташування компонентів проекту в структурі ПЛІС;
- Редактор часових діаграм: дозволяє задати вхідні часові діаграми для аналізу проекту симулятором;
- Компілятор: перевіряє синтаксис проекту, виконує логічний аналіз проекту та створює конфігураційний файл;
- Симулятор: моделює роботу проекту на основі заданих вхідних часових діаграм та формує вихідні часові діаграми;
- Програматор: завантажує конфігураційний файл у пам'ять ПЛІС через програмувальний кабель [USBBlasterAltera](#).

Важливими компонентами середовища Quartus II є його бібліотеки, які включають розділи примітивних, стандартних і багатофункціональних компонентів. Приклади бібліотечних компонентів представлені на рисунку 2.3.

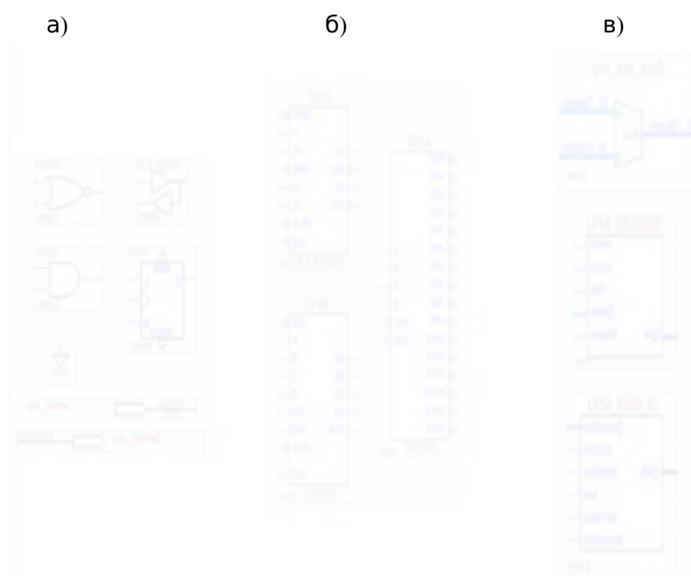


Рисунок 2.3 – Приклади бібліотечних компонентів:
а) примітивні; б) стандартні; в) багатофункціональні

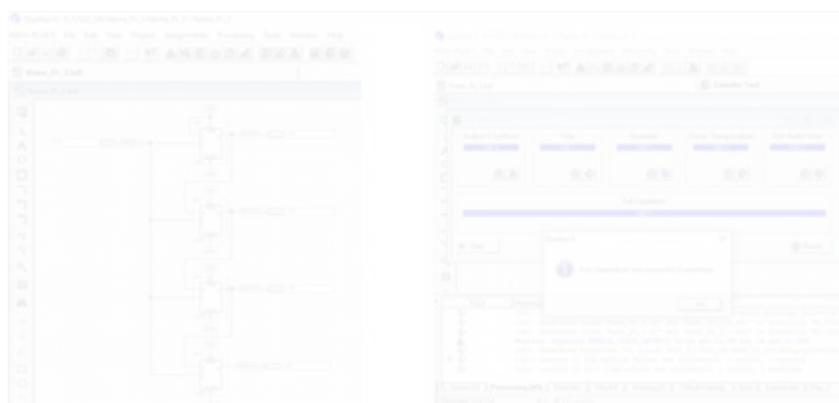
Наявність бібліотечних компонентів полегшує процес проектування цифрових пристроїв графічним методом. Високоєфективним є використання багатофункціональних компонентів, які дозволяють програмувати параметри компонента відповідно до потреб проекту, наприклад, вибирати вхідні та вихідні сигнали компонента зі списку доступних, задавати число розрядів тощо. Середовище дозволяє створювати власні бібліотечні компоненти на основі проекту, який виконано графічним чи програмним методом.

Створення проекту в середовищі Quartus II передбачає:

- Підготовку файлів проекту, а саме, файл проекту, графічний або текстовий файл для побудови схеми графічним методом чи програмним методом з використанням мови опису апаратури, файл часових діаграм **ТОЩО**;
- Побудову схеми проекту в графічному редакторі з використанням бібліотечних компонентів або написання програми в текстовому редакторі з використанням мови опису апаратури;
- Компілювання проекту, результатом якого є аналіз синтаксису проекту та створення файлів для симулятора і конфігураційного файлу проекту;
- Моделювання роботи проекту з використанням засобів середовища Quartus II, а саме, симулятора та часового аналізатора;
- Редагування, за потреби, розташування компонентів проекту в структурі ПЛІС з використанням редактора-компонувальника;
- Програмування ПЛІС і верифікацію проекту на фізичному рівні у випадку наявності відповідних технічних засобів.

Розглянемо приклад проектування цифрового лічильника з використанням простих бібліотечних компонентів. На рисунку 2.4 представлено приклад створення графічного проекту схеми 4-розрядного лічильника імпульсів на тригерах Т-типу і результат його компілювання в середовищі Quartus II. Проект створюється у графічному редакторі класичним методом. У схемі використано бібліотечні компоненти Т-тригера. Незадіяні установочні входи тригера **PRNi** **CLRN** повинні бути підключеними до шини живлення V_{cc} , що відповідає подачі

на них потенціалу логічної одиниці. Вхід рахунку імпульсів Clk має бути підключеним до вхідної клеми Input, а вихідні сигнали Q_i підключаються до вихідних клем Output. Компілятор перевіряє синтаксис проєкту і створює файли для симулятора, часового аналізатора і конфігураційний файл. При некоректній



побудові схеми проєкту компілятор видає повідомлення про помилки.

а)

б)

Рисунок 2.3 – Приклад створення графічного проєкту в середовищі Quartus II на простих компонентах:

а) схема проєкту; б) результат компілювання проєкту

2.3 Лабораторний стенд

У дипломному проєкті для реалізації інтерактивного робочого місця з проєктування пристроїв на ПЛІС використано лабораторний стенд, який був розроблений і виготовлений за участі викладачів і студентів коледжу [x]. Такими стендами обладнана лабораторія Мікропроцесорної техніки коледжу. Проте, через відсутність програмувальних кабелів і осцилографів на робочих місцях та відповідного методичного забезпечення неможливо було реалізувати під час навчального процесу режим інтерактивного проєктування пристроїв на ПЛІС. Стенди дозволяли проводити лабораторні роботи лише з дослідження цифрових пристроїв, які були створені і запрограмовані у ПЛІС стенду. Такі дослідження

проводилися у статичному режимі з використанням світлодіодної індикації кодів, яку підтримує стенд. Впровадження результатів дипломного проєкту у навчальний процес лабораторії Мікропроцесорної техніки дозволить підвищити ефективність навчального процесу. Стенд в режимі інтерактивного дослідження проєктованого цифрового пристрою на фізичному рівні має наступні характеристики:

- Тактова частота опорного генератора стенду, МГц 10;
- Число розрядів двонапрявленої шини введення/виведення стенду 8;
- Число розрядів світлодіодного індикатора 2x8;
- Число тумблерів для ручного введення кодів 8;
- Кількість вентилів ПЛІС, тис. 10;
- Максимальна тактова частота роботи компонентів ПЛІС, МГц 250;
- Напруга живлення, В +5;
- габаритні розміри стенду, мм 330x280x50.

2.4 USB осцилограф

Наявність персонального комп'ютера у складі робочого місця дозволила використати для контролю параметрів досліджуваних схем цифрових пристроїв USB осцилограф. Для реалізації завдання дипломного проєкту я вибрав двоканальний USB осцилограф ISDS 205A, який має наступні характеристики:

- Число вхідних каналів 2;
- Вхідний опір, МОм 1;
- Вхідна ємність, пФ 25;
- Діапазон вхідної напруги, В від мінус 5 до +5;
- Точність по вертикалі, % ± 3 ;
- Часова розгортка, нс/под. – мс/под. 10 – 100;
- Обсяг буферної пам'яті, Мбайт/канал 1;
- Смуга пропускання, МГц 16;
- Максимальне число вибірок, М/сек 48.

3 РЕАЛІЗАЦІЯ ІНТЕРАКТИВНОГО АВТОМАТИЗОВАНОГО РОБОЧОГО МІСЦЯ

3.1 Характеристика робочого місця

Інтерактивне автоматизоване робоче місце для проектування цифрових пристроїв на ПЛІС відповідно до структурної схеми, яка була розглянута у другому розділі, включає:

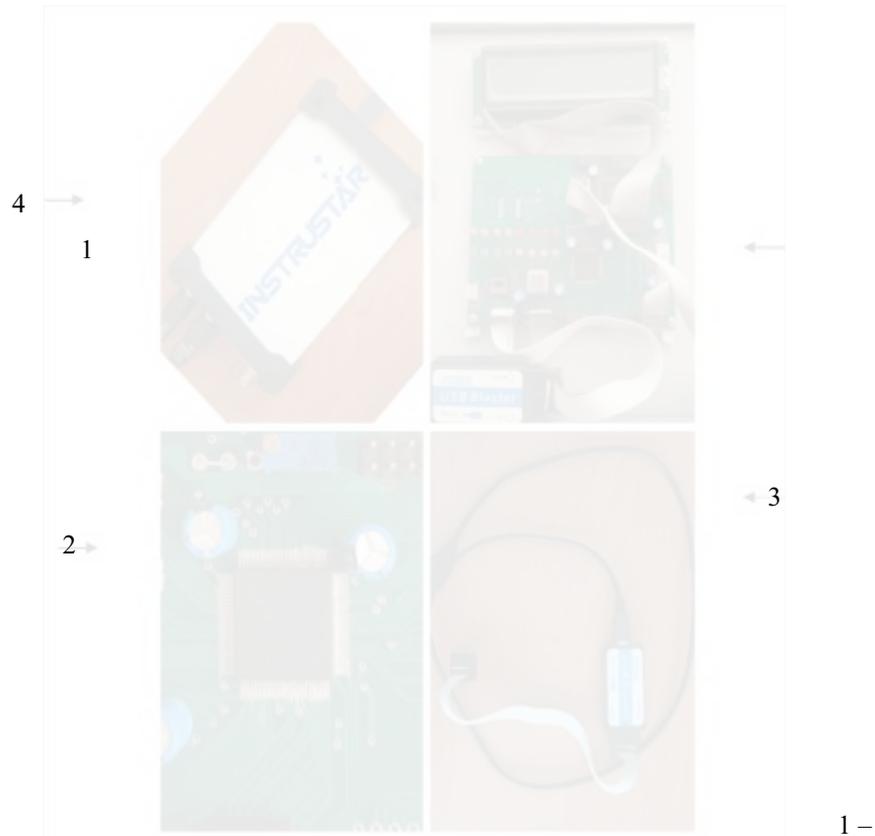
- Персональний комп'ютер з програмним забезпеченням QuartusII;
- Лабораторний стенд для досліджень, що містить ПЛІС типу FPGA - EP1K10TC100-3 фірми Altera;
- USBBlaster – кабель для програмування ПЛІС;
- Двоканальний USB-осцилограф.

Загальний вигляд робочого місця представлений на рисунку 3.1.



Рисунок 3.1 - Загальний вигляд робочого місця

На рисунку 3.2 представлені деякі основні компоненти робочого місця.



плата в стенді; 2 – ПЛІС;

3 – кабель для програмування ПЛІС;

4 – **USB**осцилограф

Рисунок 3.2 – Основні компоненти робочого місця

Робоче місце дозволяє:

- Створювати проекти цифрових пристроїв графічним або програмним методами;
- Верифікувати проект програмними засобами середовища Quartus;

- Записувати створений конфігураційний файл проєкту в пам'ять ПЛІС, що розміщена на платі стенду, через програмувальний USB-кабель;
- Досліджувати роботу спроектованого цифрового пристрою за допомогою USB-осцилографа та світлодіодних індикаторів, що розташовані на платі стенду.

Дванадцять таких робочих місць на даному етапі впроваджується в експлуатацію в лабораторії Мікропроцесорної техніки нашого коледжу і проходять апробацію в рамках технологічної практики студентів.

3.2 Графічний файл для дослідження проєктів на фізичному рівні

Схема електрична принципова графічного файлу, що створена в середовищі Quartus II і використовується при дослідженні проєктованих пристроїв на фізичному рівні, представлена на рисунку 3.3.

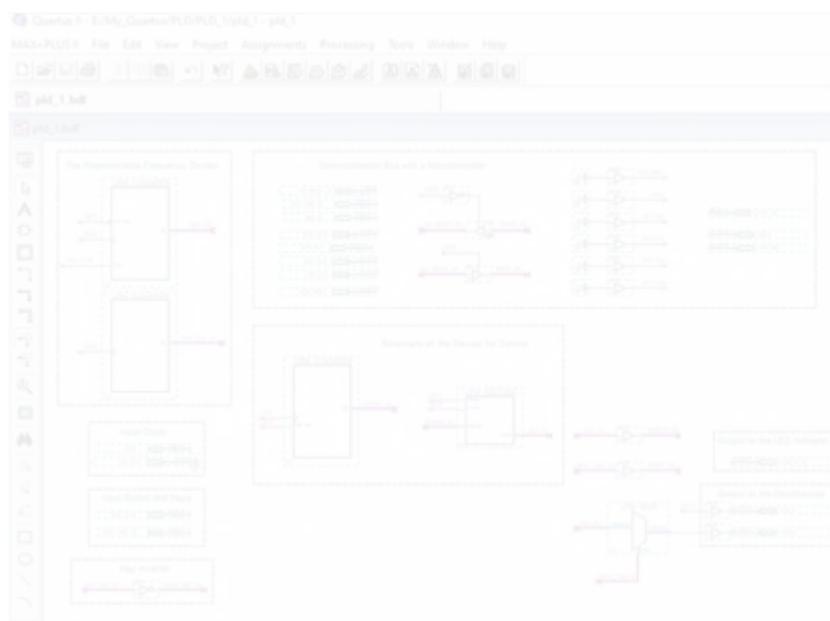


Рисунок 3.3 – Схема електрична принципова графічного файлу для дослідження проєктованих пристроїв

Представлена схема містить наступні вузли: формування вхідних сигналів; відображення інформації; комунікації з мікроконтролером стенду;схему проєктованого пристрою, що підлягає контролю. Розглянемо кожний з цих вузлів.

Вузол формування вхідних сигналів для проєктованого пристрою містить:

- Вхідний контакт від генератора $F_{clk}=10$ МГц (Input Clock);
- Кнопку But_Clk і 8 ключів Key_In[7..0] для ручного введення кодів (Input Button and Keys);
- Інвертори сигналів ключів (Key Invertors);
- Програмований дільник вхідної частоти Fclk з коефіцієнтами ділення 2, 2^2 , 2^3 , 2^4 , ..., 2^{32} (The Programmable Frequency Divider).

Вузол відображення інформації містить:

- Виходи на 16 світлодіодів Ind[15..0] для відображення коду з конфігурацією 2x8біт (Output to the LED Indicator);
- Виходи на 2-канальний осцилограф OSC1 і OSC2 (Output to the Oscilloscope);
- 8-канальний комутатор цифрових сигналів з ручним перемиканням каналів ключами Key_In[2..0], що підключений до виходу OSC2 (LPM_MUX).

Вузол комунікації з мікроконтролером (Communication Bus with a Microcontroller) стенду містить:

- Двонаправлену 8-розрядну шину, яка забезпечує комунікацію між мікроконтролером і зовнішніми пристроями.

Схема проєктованого пристрою, що підлягає контролю (Schematic of the Device for Control), в якості прикладу містить:

- 8-розрядний двійковий лічильник (LPM Counter) імпульсів і 3-розрядний двійковий дешифратор (LPM Decode).

3.3 Методика дослідження цифрових схем з використанням інтерактивного автоматизованого робочого місця

Розглянемо приклади дослідження схем лічильника імпульсів і двійкового дешифратора, що закладені у файлі (рисунок 3.3).

Для дослідження схем, що закладені у файлі, достатньо скористатися заготовкою проекту відповідно до схеми на рисунку 3.3, що зберігається на робочому диску комп'ютера. Проте, якщо ми хочемо використовувати таку заготовку проекту у навчальному процесі, то ми повинні зберігати її на диску як еталон, а студенти, чи будь хто інший, хто хоче опанувати роботу з ПЛІС, повинні створити свій власний проект, зробивши копію еталонного проекту. Така процедура є допустимою але, враховуючи те, що мікросхема ПЛІС у стенді, яку ми використовуємо для досліджень, є розпаяна на платі і ми не можемо змінювати первинне призначення виводів мікросхеми. Просте копіювання графічного файлу проекту не забезпечує збереження первинного призначення виводів мікросхеми ПЛІС. Копіювання скопійованої схеми у новому проекті приведе до втрати первинних налаштувань, що призведе до втрати працездатності схеми. Для цього, щоби зберегти первинні налаштування, ми повинні імпортувати призначення виводів з первинного проекту. Середовище Quartus II підтримує таку процедуру. Для цього потрібно зайти в меню Assignments > Import Assignments і в діалоговому вікні, яке відкриється, вказати файл з розширенням e_name.qsf еталонного проекту, з якого ми копіювали графічний файл. Після цього нам залишається скопіювати новий проект і у ньому будуть збережені усі призначення виводів, які були зроблені в еталонному проекті. У новоствореному проекті ми можемо вносити необхідні нам зміни, наприклад, змінювати сигнали, які ми хочемо бачити на моніторі комп'ютера тощо.

Наступне, якщо ми вже підготували свій графічний файл для дослідження схем, що є у файлі проекту, то для дослідження схем нам потрібно завантажити конфігураційний файл новоствореного проекту у конфігураційну пам'ять мікросхеми ПЛІС, що є на платі стенду, за допомогою програмувального кабелю,

скориставшись відповідною утилітою середовища Quartus II. Для відображення потрібних сигналів на екрані осцилографа необхідно у схемі проекту підключити їх до входу осцилографа безпосередньо чи через комутатор, що дозволить спостерігати їх на моніторі комп'ютера послідовно, перемикаючи входи комутатора.

На рисунку 3.4 показано процедуру програмування ПЛІС з середовища Quartus II через USB Blaster.

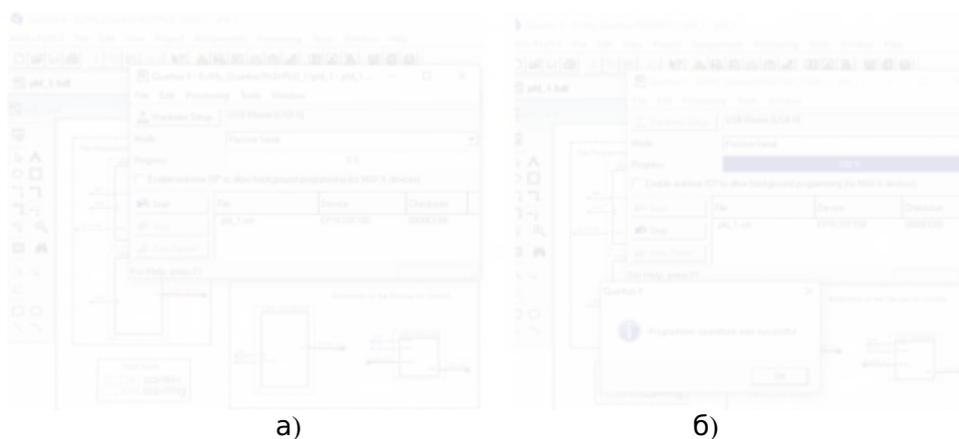


Рисунок 3.4 – Програмування ПЛІС з середовища Quartus II:

а) початок; б) завершення

Розглянемо процедуру дослідження двійкового дешифратора і лічильника імпульсів, які показані в якості досліджуваних пристроїв на рисунку 3.3. Для цього, щоби спостерігати сигнали дешифратора на моніторі нам потрібно його вихідні сигнали підключити до входів осцилографа. Це робиться через назви сигналів. У нашому прикладі (див. схему на рисунку 3.3) ми підключили старший розряд дешифратора $G[7]$ до одного з входів осцилографа, а всі вісім вихідних сигналів $G[7..0]$ через комутатор до іншого входу осцилографа. Вибір каналу комутатора забезпечується ключами $K1..K3$ стенду, що дає змогу контролювати всі вихідні сигнали дешифратора. Сигнал $G[7]$ використовується також в якості сигналу синхронізації (здається в меню налаштувань осцилографа). Це дає нам

змогу спостерігати положення вихідних сигналів дешифратора відносно сигналу на виході старшого розряду G[7].

На рисунку 3.5 показані часові діаграми, які ілюструють дослідження схеми двійкового дешифратора, що показана на рисунку 3.3 за допомогою USB осцилографа.



Рисунок 3.5 – Часові діаграми схеми двійкового дешифратора:

- а) сигнали G[0] (верхній) і G[7] (нижній);
- б) сигнали G[3] (верхній) і G[7] (нижній)

Як бачимо з рисунків 3.5,а і 3.5,б сигнали на виходах дешифратора зміщуються у часі відносно опорного сигналу G[7], який є сигналом синхронізації. Адже для трьох розрядного дешифратора, який має відповідно вісім виходів кожній кодовій комбінації буде відповідати логічна одиниця тільки на одному з восьми виходів.

Для того, щоби перейти до дослідження схеми двійкового лічильника імпульсів (див. схему на рисунку 3.3), необхідно на відповідних входах схем, що сполучають контрольовані сигнали з входами осцилографа, вказати відповідні назви вихідних сигналів лічильника CLKD[7..0], перекомпілювати проєкт і перепрограмувати ПЛІС.

На рисунку 3.6 показані часові діаграми, які ілюструють дослідження схеми лічильника імпульсів, що показана на рисунку 3.3 за допомогою USB осцилографа.



а)

б)

Рисунок 3.6 – Часові діаграми схеми лічильника імпульсів:

а) сигнали CLKD[2] (верхній) і CLKD[7] (нижній);

б) сигнали CLKD[4] (верхній) і CLKD[7] (нижній)

При дослідженнях роботи цифрового лічильника імпульсів за допомогою двоканального осцилографа в якості сигналу синхронізації потрібно вибирати сигнал з меншою частотою, що забезпечить якісну синхронізацію роботи осцилографа. У нашому прикладі найнижчу частоту має найстарший розряд лічильника CLKD[7], який відображається на одному з каналів осцилографа (нижній сигнал на графіках, що відображається синім кольором). Саме цей сигнал використовується для синхронізації роботи осцилографа. На іншому каналі осцилографа (верхній сигнал на графіках, що відображається жовтим кольором) завдяки восьми канальному комутатору ми можемо спостерігати по чергово вихідні сигнали всіх восьми вихідних сигналів досліджуваного лічильника імпульсів.

На верхньому графіку рисунку 3.6,а зображено жовтим кольором сигнал з виходу CLKD[2] лічильника імпульсів, а на верхньому графіку рисунку 3.6,б зображено жовтим кольором сигнал з виходу CLKD[4] лічильника імпульсів. Як бачимо з графіків частота сигналу на виході CLKD[4] є в чотири рази нижчою, ніж частота сигналу CLKD[2], адже кожний наступний розряд лічильника ділить частоту попереднього розряду вдвічі.

Схожість

Джерела з Бібліотеки

33

1	Студентська робота	ID файлу: 1000578359	Навчальний заклад: National University of Life and Environ	29 Джерело	0.6%
2	Студентська робота	ID файлу: 1006779879	Навчальний заклад: National Aviation University		0.23%
3	Студентська робота	ID файлу: 1011284832	Навчальний заклад: Cherkasy State Technological Univer	2 Джерело	0.18%
4	Студентська робота	ID файлу: 1000084267	Навчальний заклад: National Technical University of Ukraine "Kyi...		0.18%